Exponentiation in pairing-friendly groups using homomorphisms

Steven D. Galbraith* and Michael Scott**

[1] Mathematics Department,
Royal Holloway, University of London,
Egham, Surrey, TW20 0EX,
United Kingdom.
steven.galbraith@rhul.ac.uk
[2] School of Computing, Dublin City University.
Ballymun, Dublin 9, Ireland.
mike@computing.dcu.ie

Abstract. We present efficiently computable homomorphisms of the groups G_2 and G_T for pairings $G_1 \times G_2 \to G_T$. This allows exponentiation in G_2 and G_T to be accelerated using the Gallant-Lambert-Vanstone method.
Keywords: pairings.

1 Introduction

Let r be a prime and let G_1, G_2 and G_T be cyclic groups of order r with a bilinear pairing

$$e : G_1 \times G_2 \to G_T.$$

In practice G_1 is a set of points on some elliptic curve E over $\mathbb{F}_p$ and G_2 is a set of points on a twist E' of E over some field $\mathbb{F}_{p^e}$. The group G_T is a subgroup of $\mathbb{F}_{p^k}^*$, where k is the embedding degree, and is usually represented in a compressed form by using traces or algebraic tori.

Pairings over ordinary elliptic curves suffer in comparison to those over supersingular curves, in that a larger group G_2 is often required for one of the two parameters to the pairing. The quadratic twist is always an option if k is even, so $e = k/2$ and for the case $k = 2$ the quadratic twist is again over the base field. There is a family of pairing-friendly curves [10] of embedding degree $k = 6$ where the sextic twist applies, and again in this case $e = k/6 = 1$. However for most other cases of interest $e > 1$. For example with the BN curves [5], even though the sextic twist applies, G_2 is over the field $\mathbb{F}_{p^2}$. This suggests that manipulations of points over G_2 in some pairing-based protocols are in general likely to be more expensive than those over G_1, and perhaps much more expensive. Here we will demonstrate that this is not necessarily the case.

Gallant, Lambert and Vanstone (GLV) [12] gave a method to speed up operations in groups when a suitable group homomorphism is available. The main result of the paper is to get such a group homomorphism from the Frobenius map in $\mathbb{F}_{p^k}$. This particularly speeds up operations in G_2, but also has implications for G_T.

The main contributions of our paper are:

* This work supported by EPSRC grant EP/D069904/1.
** This author acknowledges support from the Science Foundation Ireland under Grant No. 06/MI/006

1. To speed up arithmetic in G_2 and G_T using the GLV method.
2. To show that simpler GLV decompositions of an exponent are often possible for pairing friendly curves (i.e., not requiring lattice reduction as a precomputation), especially for Ate friendly curves.
3. To remark that parameters for Ate-friendly curves give rise to good parameters for XTR and torus based cryptography.
4. To note that our methods can be used to obtain larger equivalence classes for the Pollard rho method.

We now outline the paper. Sections 2 and 3 recall basic facts about pairings and the GLV method. Section 4 analyses the methods of Stam and Lenstra when applied in the target group G_T for pairing-based cryptography. Section 5 contains our main result, namely the construction of a group homomorphism on G_2. Section 6 studies some specific examples. Section 7 summarises the costs and benefits of the GLV method. Sections 8 and 9 mention some consequences for trace/torus cryptography and the difficulty of the DLP in G_2, and we conclude in Section 10.

2 Elliptic Curves and Pairings

Let E be an elliptic curve over $\mathbb{F}_p$ where p is prime. Denote by ∞ the point at infinity on E. Let $\#E(\mathbb{F}_p) = p+1-t$ be the number of points on the curve, where t is the trace of the Frobenius. Let $r \mid \#E(\mathbb{F}_p)$ be a large prime. The embedding degree is the smallest integer k such that $r \mid (p^k - 1)$. We assume that no proper subfield of $\mathbb{F}_{p^k}^*$ contains elements of order r.

Let $G_1 = E(\mathbb{F}_p)[r]$ and let G_T be the subgroup of $\mathbb{F}_{p^k}^*$ of elements of order r. Denote by π_p the p-power Frobenius map on E. Define G_2 to be the subgroup of $E(\mathbb{F}_{p^k})[r]$ such that π_p acts as multiplication by p. We assume we have a non-degenerate bilinear pairing (such as the Ate pairing [15])

$$e : G_1 \times G_2 \to G_T.$$

Following Section 4 of [15] we represent G_2 as a group of points on a twist E' of E. This means there is an isomorphism $\phi : E' \to E$ with field of definition $\mathbb{F}_{p^d}$. It is necessary that the automorphism group $\mathrm{Aut}(E)$ contain an element of order d. Hence the only non-trivial possibilities are $d = 2$, $d = 4$ if $j(E) = 1728$ (CM discriminant $D = -4$) and $d = 3, 6$ if $j(E) = 0$ (CM discriminant $D = -3$). We assume $d \mid k$ and write $k = de$. Then $G_2 = E'(\mathbb{F}_{p^e})[r]$ and $\phi(G_2) \subset E(\mathbb{F}_{p^k})$. If $r > d$ then the image of $E'(\mathbb{F}_{p^e})[r]$ under ϕ does lie in the eigenspace of the q-power Frobenius on $E(\mathbb{F}_{p^k})$ with eigenvalue p.

For efficient pairing computation, much work has been done to find viable bilinear pairings, with the minimum number of iterations in Miller's algorithm. Starting with the Duursma-Lee method [9] and subsequent work by Barreto et al. [4] (in the context of supersingular curves), Hess al. extended the idea to ordinary elliptic curves with the discovery of the Ate pairing. Now the main Miller loop in the pairing computation iterates only $\lg(|t-1|)$ times, rather than $\lg(r)$ times as required by the Tate pairing. An "Ate pairing friendly curve" is defined as one where $|t-1|$ is as small as possible compared to r. It has been conjectured that the minimum possible ratio between $|t-1|$ and r is $1/\varphi(r)$ (where φ is the Euler totient function), and indeed this ideal condition is met by some pairing-friendly families of curves. Recently Lee, Lee and Park [18], Hess [16] and Vercauteren [28] have shown how to achieve the same level of loop truncation on curves, even if they are not Ate pairing friendly.

Many families of pairing-friendly curves have been found - see [10] for a survey. The most sought after curves are those with the minimum value of ρ, which is defined as the rounded fraction $\lg(p)/\lg r$. It is relatively easy to find families of curves with $\rho \approx 2$, but it is much preferred that $\rho \approx 1$, as this leads to more efficient implementations.

3 The GLV method

Gallant, Lambert and Vanstone [12] introduced a method to speed up general point multiplication nP in $E(\mathbb{F}_p)[r]$. In its simplest form their method works if, given a point P, one can somehow have knowledge of a non-trivial multiple of P. This extra information is available if there is an efficiently computable endomorphism ψ on E defined over $\mathbb{F}_p$ such that $\psi(P) = \lambda P$. One can then compute nP efficiently by writing $n \equiv n_0 + n_1\lambda \pmod r$ with $|n_i| < \sqrt{r}$ and performing the double exponentiation $n_0P + n_1\psi(P)$. Decomposing n as $n_0 + n_1\lambda \pmod r$ is done by solving a closest vector problem in a lattice and the Euclidean algorithm can be used to compute a suitable lattice basis, see [12, 23] for the details. We call this the GLV method.

Double exponentiation algorithms require precomputation and storage, but their efficiency comes from halving the number of doublings. One can simultaneously reduce the number of additions by using window methods, but this adds further precomputation and storage. Another method to reduce the number of additions is to allow signed representations for n_0 and n_1 and compute their joint sparse form (that is such that the signed expansions of n_0 and n_1 both have i-th bit equal to 0 with probability approximately 1/2). We refer to Section 9.1.5 of [1] for further details.

The idea generalises to m-dimensional expansions $n \equiv n_0 + n_1\lambda + \cdots + n_{m-1}\lambda^{m-1} \pmod r$ assuming that the powers of λ are sufficiently different modulo r (the typical requirement is that the endomorphism ψ satisfies a characteristic polynomial of degree $\geq m$; see the discussion below). We call this the m-dimensional GLV method.

The task of decomposing n is again solving a closest vector problem in a lattice. This problem can be efficiently solved using Babai's rounding method [2] if an LLL-reduced lattice basis is precomputed. More precisely, define the modular lattice

$$L = \left\{ x \in \mathbb{Z}^m : \sum_{i=0}^{m-1} x_i\lambda^i \equiv 0 \pmod r \right\}. \tag{1}$$

The $2m$ vectors $(0, \ldots, 0, r, 0, \ldots, 0)$ and $(0, \ldots, 0, \lambda, -1, 0, \ldots, 0)$ generate the (row) lattice L if $\gcd(\lambda, r) = 1$. Run LLL on this basis to obtain a new basis. Given an exponent n use the Babai rounding technique to find a lattice vector $x = (x_0, \ldots, x_{m-1})$ close to $w = (n, 0, \ldots, 0)$. Define $u = w - x$. Then $\sum_{i=0}^{m-1} u_i\lambda^i \equiv n \pmod r$ by definition. If the LLL-reduced basis is sufficiently good then the coefficients u_i will be such that $|u_i| \approx r^{1/m}$. The practical performance of this approach depends on the particular parameters under consideration.

We stress that the lattice reduction is a pre-computation; the online cost in point multiplication is just the Babai rounding step. An alternative approach (when a random multiple of a point P is required) is to simply choose random coefficients $n_0, \ldots, n_{m-1}$ instead of choosing n first and then decomposing it.

We remark that there are natural boundaries on the size of m. For example, let $r \mid (p^2 - p + 1)$ and let ψ be the p-power Frobenius map in the subgroup G_T of $\mathbb{F}_{p^6}^*$ of order r. Then $\lambda \equiv p \pmod r$ satisfies $\lambda^6 \equiv 1 \pmod r$ and one might expect to be able to take $m = 6$. However, since $\lambda^2 \equiv \lambda - 1$

(mod r) it follows that $n_0 + n_1\lambda + n_2\lambda^2 \equiv (n_0 - n_2) + (n_1 + n_2)\lambda \pmod{r}$. Therefore the size of the largest coefficient n_i in the 3-dimensional expansion cannot be significantly smaller than the size of the largest coefficient in the 2-dimensional case.

The original proposal of Gallant, Lambert and Vanstone specifically proposed using the automorphisms of elliptic curves E with $j(E) = 0, 1728$. Hence it is standard that the GLV method can be used to speed up point multiplication in G_1 and G_2 in the cases for which using twists gives good compression of G_2. In both cases the automorphisms satisfy a characteristic polynomial of degree 2 with coefficients in $\{0, 1\}$, so only the two-dimensional GLV method applies.

4 Using the Frobenius to speed up operations in G_T

In this case much of the work has already been done by Stam and Lenstra. However here we consider their results in the context of pairings.

We call the subgroup of $\mathbb{F}_{p^k}^*$ of order $\Phi_k(p)$ (where $\Phi_k(x)$ is the k-th cyclotomic polynomial) the "cyclotomic group". The group G_T of order r is a subgroup of the cyclotomic group in $\mathbb{F}_{p^k}^*$. For the case $k = 6$, $r \mid (p^2 - p + 1)$ and G_T is a subgroup of the well studied "XTR subgroup" of $\mathbb{F}_{p^6}^*$. For the case $k = 2$, the cyclotomic group is of order $p + 1$, and was used in the LUC cryptosystem (see Stam and Lenstra [25]).

There are three approaches for efficient arithmetic in cyclotomic subgroups. The simplest approach is to perform arithmetic using a standard representation for $\mathbb{F}_{p^k}$ and to exploit tricks which arise from elements having order dividing $\Phi_k(p)$ (for example, the fact that the inverse of an element can be computed efficiently). The other approaches are based on compression of field elements using traces or algebraic tori respectively. All three methods can be applied for efficient exponentiation in G_T (for example see [13]). The latter two methods are also useful for minimising bandwidth in pairing-based cryptography.

Stam and Lenstra [26] discuss the first approach. They exploit the fact that elements in the cylotomic group have some extra properties that do not hold for general elements in $\mathbb{F}_{p^k}$. Specifically field inversion is a simple conjugation, and thus effectively free, and the field squaring operation can be significantly cheaper. Also as inversion is free, faster NAF methods of windowing are applicable [22].

For exponentiation in the XTR subgroup the most efficient method is to use traces. For XTR the trace is over $\mathbb{F}_{p^2}$, so the compression is by a factor of 3. For LUC the trace is over $\mathbb{F}_p$, and the compression is by a factor of two. However traces can only be manipulated in limited ways: for example multiplication of subgroup elements, if required by a protocol, is non-trivial. When using compression by a factor of 2 then exponentiating using a torus representation is competitive with LUC [13]. One advantage of tori is that one can efficiently multiply group elements as well as exponentiate them. In [8] the applications of higher dimensional tori are considered, and indeed it is suggested that in principle a degree 8 Frobenius automorphism can be used to split the exponent, and then use multi-exponentiation, in much the same way as suggested here.

In [25] a method for double exponentiation using traces is proposed, for both the LUC and XTR cases. This is required for example for the application of LUC/XTR to ElGamal-like digital signature verification schemes. But the authors also point out that the Frobenius endomorphism can be used to implement a single exponentiation using a variant of the GLV idea (independently discovered) with their double exponentiation algorithm, and indeed this is the fastest way to do it. In Section 4.4 of [25] it is pointed out that if $p \bmod r \approx \sqrt{r}$ then the 2-dimensional decomposition

of the exponent is particularly easy, and the decomposition can be found at the cost a division and a remainder. In the sequel we will refer to such a decomposition as "natural". As we will see, in the context of pairings, natural decompositions arise quite frequently.

It is apparently non-trivial to extend the double exponentiation of traces to general multi-exponentiation [25], and so if multi-exponentiation is possibly beneficial then we must either use torus methods or else work in the full $\mathbb{F}_{p^k}^*$ (see Stam and Lenstra [26]).

Pairings evaluate as elements in G_T, often in higher degree cyclotomic fields than those considered by Stam and Lenstra. Many of the same ideas apply immediately if the embedding degree is a multiple of 2 or 6. However in the context of pairings, since we know that the pair (p, r) arise in the context of an elliptic curve, we know that $p \bmod r = t - 1$. Fortunately for us, for many pairing friendly curves $|t - 1|$ is often rather small compared with r, in which case higher dimensional natural decompositions will also be possible. Application of the Frobenius to an element x of order r gives us the value $x^p \equiv x^{t-1}$, so the exponent n can be expressed to the base $(t-1)$ and multi-exponentiation applied as $x^{n_0}.(x^p)^{n_1}.(x^{p^2})^{n_3} \cdots$. See the examples below for more details.

5 A homomorphism on G_2

As described above, the group G_2 is a subgroup of $E'(\mathbb{F}_{p^e})$ and there is a group homomorphism $\phi : E'(\mathbb{F}_{p^e}) \to E(\mathbb{F}_{p^k})$. We now explain how to use the p-power Frobenius on $E(\mathbb{F}_{p^k})$ to get an efficiently computed group homomorphism on G_2. Iijima et al [17] used essentially the same ideas to construct a homomorphism for a different application.

Lemma 1. *Let notation be as above. Denote by π_p the p-power Frobenius map on E. Then $\psi = \phi^{-1}\pi_p\phi$ is an endomorphism of E' such that $\psi : G_2 \to G_2$. Further, for $Q \in G_2$ we have $\psi^k(Q) = Q$, $\psi(Q) = pQ$ and $\Phi_k(\psi)(Q) = \infty$ where $\Phi_k(x)$ is the k-th cyclotomic polynomial.*

Proof. Clearly ψ is a morphism from E' to E' which fixes the point at infinity. Hence ψ is an endomorphism of E'.

Let $Q \in E'(\mathbb{F}_{p^e})[r]$. Then $\phi(Q) \in E(\mathbb{F}_{p^k})$ and, as mentioned in Section 2, we have $\pi_p(\phi(Q)) = p\phi(Q)$. Hence $Q' = \pi_p(\phi(Q))$ lies in the image of $E'(\mathbb{F}_{p^e})$ under ϕ and so $Q'' = \phi^{-1}(Q') \in E'(\mathbb{F}_{p^e})$.

Clearly $\psi^k = \phi^{-1}\pi_p^k\phi = \phi^{-1}\pi_{p^k}\phi$. Since $\pi_p^k = 1$ on $E(\mathbb{F}_{p^k})$ it follows that $\psi^k(Q) = Q$. Further, as noted above, $\pi_p(\phi(Q)) = p\phi(Q)$ and so

$$\psi(Q) = \phi^{-1}\pi_p\phi(Q) = \phi^{-1}p\phi(Q) = pQ.$$

Finally, since Q has order r and $r \mid \Phi_k(p)$ it follows that $\Phi_k(\psi)(Q) = \Phi_k(p)Q = \infty$. This completes the proof. □

The group homomorphism ψ can be computed efficiently and so is potentially useful for the GLV method. However, there are cases when this map is just a familiar homomorphism arising in an unfamiliar way. Our main interest is when the construction gives something which was not previously used for efficient computation. The following result shows that if $e = 1$ then we are just recovering elements of the automorphism group of the curve.

Lemma 2. *If $e = 1$ then ψ is equal to $\rho\pi_p'$ where π_p' is the p-power Frobenius on E' and where ρ is an element of $Aut(E')$.*

Proof. By Corollary 2.12 of [24] ψ can be written as $\rho\pi_p$ where $\pi_p : E' \to E'^{(p)}$ is the p-power Frobenius to a Galois conjugate of E' and $\rho : E'^{(p)} \to E'$ is an isomorphism. In the case $e = 1$ we have $E' = E'^{(p)}$ and so $\rho \in \mathrm{Aut}(E')$. $\square$

This result shows that our methods give no new result in the case $e = 1$ (although decomposition of a random exponent is always simpler than the general case of GLV). The case $e > 1$ is interesting as it gives potential for new and improved applications of the GLV method. In particular, we have homomorphisms which do not come from $\mathrm{Aut}(E')$.

We mention that a similar optimisation for G_1 was proposed by Granger, Page and Stam in Section 4 of [13]. They considered a supersingular elliptic curve $E(\mathbb{F}_{3^m})$ and used the fact that multiplication by 3^m on E is given by a simple and easy to compute formula. Since $3^m \equiv \pm 3^{(m+1)/2} - 1 \pmod{r}$ they remarked that it is easy to obtain a GLV decomposition in this case.

6 Examples

Pairing friendly families vary significantly in detail, so the benefits of our methods are best considered on a case-by-case basis. The first two examples correspond to the case $e = 1$ and, as explained earlier, our methods give nothing new in this case. However, it is useful to demonstrate how simple the GLV decomposition is in these cases.

Example 1. Consider the pairing-friendly family of $k = 6, \rho = 2$ curves (see Section 6.7 of [10]), with $D = -3$ and $j(E) = 0$

$$p = 27x^4 + 9x^3 + 3x^2 + 3x + 1 \qquad r = 9x^2 + 3x + 1 \qquad t = 3x + 2$$

One can construct an elliptic curve $E : Y^2 = X^3 + B$ over $\mathbb{F}_p$ having $r \mid \#E(\mathbb{F}_p)$. The embedding degree is 6 and one can identify G_2 with $E'(\mathbb{F}_p)$ where E' is the sextic twist of E defined over $\mathbb{F}_p$ (in other words, $e = 1$).

Since $j(E) = 0$ the standard GLV method applies immediately to G_1. However observe that r is of the form $\lambda^2 + \lambda + 1$, with $\lambda = 3x$. Therefore the standard automorphism $\rho(x, y) = (\zeta_3 x, y)$ applied to a point $P = (x, y)$, gives us the point λP, and presents us with a natural 2-dimensional decomposition of a point multiplier into its quotient and remainder modulo $3x$.

Now consider the homomorphism ψ of Lemma 1. For $Q \in G_2$ we have $\psi(Q) = TQ$ where $T = t - 1 = 3x + 1$. Hence $\psi = \rho + 1$, which can naturally be interpreted as $-\rho^2$. The point multiplication by $n < r$ can be written as $n_0 Q + n_1 \psi(Q)$ by taking the base T representation of n.

Exponentiation in G_T can use the fast trace methods of [25]. However the decomposition is again simple to obtain, as $p \bmod r = 3x + 1 \approx \sqrt{r}$. Note that the fast squaring operations of [25, 26] do not apply since $p \not\equiv 2 \pmod 3$, but one can still obtain very efficient field arithmetic in this case.

Example 2. Miyaji, Nakabayashi and Takano [21] gave parameters for curves of prime order r over $\mathbb{F}_p$ with embedding degree 6. These curves are ideal, in the sense that $\rho = 1$.

$$p = x^2 + 1 \qquad r = x^2 - x + 1 \qquad t = x + 1$$

One major drawback of the MNT method is the necessity of solving Pell equations to generate the curves. Furthermore, certain CM discriminants cannot be used. Indeed, it is not possible to

generate a suitable curve with $j(E) = 0$. In the more general setting we have $\mathrm{Aut}(E) = \{1, -1\}$ and the GLV method cannot usefully be applied. The best representation for G_2 is then as a subgroup of $E'(\mathbb{F}_{p^3})$ where E' is a quadratic twist of E which is defined over $\mathbb{F}_p$.

In this case nothing can be done for G_1, but E' is now a "subfield curve" so it is natural to use the Frobenius map π'_p on E' to speed up arithmetic on $E'(\mathbb{F}_{p^3})$. For the subgroup of relevance π_p satisfies $\pi'^2_p + \pi'_p + 1 = 0$ and so a 2-dimensional GLV method is the best on can hope for.

As with the previous example, our approach gives the same performance with simpler decomposition of the exponents. The group homomorphism ψ on G_2 defined above satisfies $\psi^2 - \psi + 1 = 0$ and acts as multiplication by $t - 1$.

Example 3. Consider this family of Ate pairing-friendly curves [3], with $k = 12, D = -3, \rho = 3/2$.

$$p = (x^6 - 2x^5 + 2x^3 + x + 1)/3 \qquad r = x^4 - x^2 + 1 \qquad t = x + 1$$

In this case standard GLV applies to G_1, and again a natural 2-dimensional decomposition is possible with the standard automorphism, given the special form of r. The group G_2 is a subgroup of the sextic twist $E'(\mathbb{F}_{p^2})$. Since $j(E') = 0$ we could use the standard GLV method, but in this case $e = 2$ so it is possible to do better. In this case for G_2 and G_T we get a natural 4-dimensional decomposition, as any multiplier in G_2 or exponent in G_T can be written as a degree 4 polynomial in $T = t-1 = x$. For G_T trace methods are probably not practical for a degree 4 multi-exponentiation, so fast non-trace based methods should be used here instead.

Example 4. Consider this family of Ate pairing-friendly curves [3], with $k = 24, D = -3, \rho = 5/4$. This curve might be appropriate at the highest levels of security.

$$p = (x^{10} - 2x^9 + x^8 - x^6 + 2x^5 - x^4 + x^2 + x + 1)/3 \quad r = x^8 - x^4 + 1 \quad t = x + 1$$

As before standard GLV applies to G_1, again with a natural 2-dimensional decomposition. G_2 is a subgroup of the sextic twist $E'(\mathbb{F}_{p^4})$. In this case for G_2 and G_T we get a natural 8-dimensional decomposition, as any multiplier in G_2 or exponent in G_T can be written as a degree 8 polynomial in $T = t - 1 = x$. Again for G_T fast non-trace-based methods should be used.

Example 5. (BN curves [5]) Consider the BN parameters

$$t = 6x^2 + 1, \qquad p = 36x^4 + 36x^3 + 24x^2 + 6x + 1, \qquad r = p + 1 - t.$$

One can construct an elliptic curve $E : Y^2 = X^3 + a$ over $\mathbb{F}_p$ having r points. The embedding degree is 12 and one can identify G_2 with a subgroup of $E'(\mathbb{F}_{p^2})$ where E' is a twist of E defined over $\mathbb{F}_{p^2}$.

Taking $\phi^{-1}\pi_p^2\phi$ gives the usual automorphism $\zeta_6(x, y) = (\zeta_3 x, -y)$ which satisfies the characteristic polynomial $\zeta_6^2 - \zeta_6 + 1 = 0$. It is standard that the GLV method using this automorphism speeds up point multiplication on E'.

Now consider $\psi = \phi^{-1}\pi_p\phi$, which satisfies $\psi^4 - \psi^2 + 1 = 0$ and so behaves as ζ_{12}. Note that $\mathrm{Aut}(E')$ does not contain an element of order 12. Since ψ acts as multiplication by p and $p \equiv (t-1) \pmod r$ one can naturally decompose n as $n_0 + n_1(t-1)$ such that $|n_0| < |t-1|$ and n_1 is a similar size. Hence one gets the 2-dimensional GLV method with natural decomposition.

Unlike the previous two examples, $|t-1| \not\approx r^{1/m}$ and so obtaining the GLV expansion is not as simple as writing the exponent n in base $(t-1)$. In this case it is necessary to use lattice reduction.

Let x be the parameter in the BN polynomial family. Then a reduced basis for the lattice L of equation (1) with $\lambda = T = 6x^2$ is

$$B = \begin{pmatrix} x+1 & x & x & -2x \\ 2x+1 & -x & -(x+1) & -x \\ 2x & 2x+1 & 2x+1 & 2x+1 \\ x-1 & 4x+2 & -(2x-1) & x-1 \end{pmatrix}.$$

The determinant of B is $-3r(x)$.

To decompose an integer n one needs to find a vector x close to $w = (n, 0, 0, 0)$ in the lattice L. One first computes a vector $v \approx wB^{-1}$. As pointed out to us by Barreto, for the above choice of B one has

$$wB^{-1} = \left(\frac{n(2x^2+3x+1)}{r}, \frac{n(12x^3+8x^2+x)}{r}, \frac{n(6x^3+4x^2+x)}{r}, \frac{n(-2x^2-x)}{r} \right)$$

and so computing v can be done using integer multiplication and division by r. One then computes the vector $u = w - vB$ whose entries are the coefficients n_i for the decomposition of n.

We illustrate the method with a toy example. Let $x = 10267$ and choose the "random" exponent $n = 123456789123456789$. The first step is to decompose the vector $(n, 0, 0, 0)$ with respect to the basis formed by the rows of B. This gives

$$\begin{aligned}(n,0,0,0)B^{-1} = (&260312812706281012445968 20/r, 160344884510280497561435613 2115/r, \\ &801724423185167914772443492389/r, -260287460854634510594347 05/r)\end{aligned}$$

Rounding these coefficients to the nearest integer gives a vector v such that vB is a close vector in the lattice to $(n, 0, 0, 0)$. Finally, compute

$$u = (n, 0, 0, 0) - vB = (-11418, -5569, -4753, -8683)$$

and one can check that $n \equiv \sum_{i=0}^{3} u_i T^i \pmod{r}$ as required. Note that all the entries in the vector u satisfy $|u_i| < r^{1/4}$. Experiments with 64-bit x (i.e., 258-bit p) always had coefficients u_i satisfying $|u_i| < 2^{65}$ as desired.

Example 6. Pairing friendly elliptic curves with $k = 9$ were considered by [19]. Since $\varphi(9) = 6$ one would get a 6-dimensional GLV method in this case.

7 Multi-exponentiation

As high-dimensional exponent decompositions are now possible, it is a useful exercise to see just how much improvement can be expected from using them. Here we follow the analysis and methods of Möller [22]. In particular we consider the wNAF-based interleaving windowed exponentiation method, which applies both for G_2 and for G_T. NAF methods apply when inversion is easy in the group. It is well known that inversion is easy for points on an elliptic curve, but perhaps not as well known that this also applies to elements in G_T. Indeed as part of the final exponentiation of the pairing, there is a component in that exponentiation of $p^{k/2} - 1$. After this exponentiation elements become "unitary" (i.e., norm 1), and with this property inversion becomes a simple conjugation, and field squaring becomes significantly cheaper [26].

We stress that we are considering exponentiation for a variable base. Hence our estimates and timings include the cost of any "precomputation" required. If the base in exponentiation is fixed then there are all sorts of different optimisations based on precomputation which can be adopted.

Here for simplicity, we do not further consider trace-based methods, as they are limited by the extent to which they can exploit multi-exponentiation. But we will of course exploit the "unitary" property of elements in G_T.

When estimating the cost of multi-exponentiation, it is important to estimate the relative costs of field multiplication and squaring in G_T, and of point doubling and addition in G_2. So we make the assumption that a point addition/field multiplication is c times the cost of a point doubling/field squaring, where we will keep c as a variable.

In fact the relative costs of these operations for an elliptic curve over a prime field is the subject of much debate, and improved formulae for both doubling and addition are still being found, often using novel coordinate systems [6]. On the other hand, for curves over larger extension fields the subject has not received much attention. Indeed it seems likely that affine coordinates may be faster than projective coordinates for higher extensions. In G_T the matter is also not so clear cut - but the fast methods for field squaring of unitary elements [26] are certainly relevant. (Even just exploiting their quadratic extension formulae leads to significant improvements when applied over large even extension fields; see [14].)

Assuming that the same window size is used for all exponents, the cost of multi-exponentiation [22] is approximated by

$$(mc(2^{w-1} - 1 + b/(w+2)) + b)$$

point doublings/field squarings for an m-dimensional decomposition, using a window size of w, and exponents of constant size b bits. Here w is simply chosen to minimise this cost – we ignore the space required for the precomputation. Clearly we have a choice as to the extent to exploit the possible decomposition, so we might double m (which will halve the size of b) to see how this effects the cost.

Our estimates (based on the above formula) are given in Table 1, for a group whose order r is 256-bits, assuming that a 1, 2, or 4 dimensional decomposition is possible (as is the case for the BN curve). We conclude that it is beneficial to decompose to the maximum extent possible, assuming that space for precomputation is not an issue.

Table 1. Cost of multi-exponentiation (Optimal w in brackets)

m	c=1.0	c=1.33	c=1.66	c=2.0	c=3.0
1	306 (4)	322 (4)	338 (4)	355 (4)	405 (4)
2	185 (4)	203 (4)	222 (4)	241 (4)	298 (4)
4	127 (3)	148 (3)	169 (3)	190 (3)	254 (3)

8 Hashing to G_2

Some pairing based protocols, for example the original Boneh and Franklin IBE scheme [7], require hashing of identities to G_1 or G_2. In the latter case this might be considered inefficient, as a large

co-factor multiplication would be required. For example consider hashing an identity to the group $G_2 \subset E'(\mathbb{F}_{p^2})$ on a BN curve. The number of points on $E'(\mathbb{F}_{p^2})$ is $r(p-1+t)$ (see [5]) where t is the trace of Frobenius of $E(\mathbb{F}_p)$. To hash-and-map an identity to a point of order r, the approach might be to hash the identity to an x coordinate, solve the quadratic curve equation to find a y coordinate (and iterate on x if one should not exist), and finally multiply this point by the co-factor $p-1+t$.

However, in this case the homomorphism ψ of Lemma 1 can be exploited to advantage. As we have seen, ψ satisfies the equation

$$\psi^2(P) - [t]\psi(P) + [p]P = 0$$

for $P \in E'(\mathbb{F}_{p^2})$. Therefore by simple substitution

$$[p-1+t]P = [t](\pi(P) + P) - \pi^2(P) - P.$$

The major cost of the cofactor calculation is therefore a multiplication by t, which is "half-sized" compared to a full multiplication by $p-1+t$.

9 Application to XTR and torus-based cryptography

As mentioned in Section 4.7 of Stam's thesis [27], a natural problem is to develop the XTR cryptosystem in $\mathbb{F}_{p^{6m}}$. The main obstacle is efficient key generation. A key generation algorithm was given in [20] but it requires factoring integers so is not very practical for large security levels.

A fact (which does not seem to have been noted before) is that polynomial families of parameters for pairing-friendly curves give efficient key generation algorithms for XTR or torus based cryptography over extension fields. Once such parameters are available then one can immediately apply the GLV method to speed up exponentiation (see [25, 26, 13]).

Furthermore, if one works in a subgroup of order r where $r = p - T$ is "super Ate friendly" then one can also benefit from the easy decomposition of exponents using the base-$|T|$ expansion and hence get very efficient multi-exponentiation in dimension > 2.

10 Security implications

Gallant, Lambert and Vanstone [11] and Wiener and Zuccherato [29] showed how to speed up the parallel Pollard rho algorithm by using equivalence classes coming from efficiently computable endomorphisms on elliptic curves. One can always work with equivalence classes of size $\#\mathrm{Aut}(E)$.

Such methods can also exploit our homomorphism, giving a slight lowering of security for the group G_2 compared with what was previously believed. As shown in Lemma 1, the homomorphism ψ on G_2 has order k and so we can partition $G_2 - \{0\}$ into equivalence classes of size k. Similarly $G_T - \{1\}$ can be partitioned into equivalence classes of size k.

The size of equivalence classes for G_2 and G_T is therefore k, while the size of equivalence classes for G_1 is $\mathrm{Aut}(E)$. When $e = 1$ then $k = \#\mathrm{Aut}(E)$ and so our result is not new, but when $e > 1$ then $k > \#\mathrm{Aut}(E)$. For example, with BN curves the size of equivalence classes is 6 for G_1 and 12 for G_2 and G_T. This does not imply that the DLP is easier by a factor $\sqrt{2}$ in G_2 and G_T than G_1, since those groups are defined over larger fields; in practice it will still be quicker to solve the DLP in $G_1 = E(\mathbb{F}_p)[r]$ than in G_2 or G_T.

11 Conclusion

In the deployment of pairing-based cryptography there has been much emphasis on the efficiency of the pairing itself. But in real protocols the efficiency of operations in the groups G_1, G_2 and G_T are also of significance, but have been rather overlooked. In this paper we address this imbalance by suggesting faster algorithms for group operations in G_T, and particularly in G_2. The latter is of particular significance for pairing-friendly ordinary elliptic curves, where G_2 may be defined over an extension field. Further work is required to determine more precisely the speed-up that can be achieved in practise.

Acknowledgements

The authors thank Paulo Barreto, Rob Granger, Xibin Lin, Nigel Smart, Martijn Stam and the anonymous referees for comments and suggestions.

References

1. R. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange, K. Nguyen and F. Vercauteren, Handbook of elliptic and hyperelliptic cryptography, Chapman and Hall/CRC, 2006.
2. L. Babai, On Lovasz lattice reduction and the nearest lattice point problem, *Combinatorica*, 6(1) (1986) 1–13.
3. P.S.L.M. Barreto, B. Lynn and M. Scott. Constructing Elliptic Curves with Prescribed Embedding Degrees. Security in Communication Networks – SCN 2002. Springer-Verlag LNCS 2576, 263–273, 2002
4. P. S. L. M. Barreto, S. Galbraith, C. O hEigeartaigh and M. Scott. Efficient Pairing Computation on Supersingular Abelian Varieties. Designs, Codes and Cryptography, 42: 239-271, 2007
5. P. S. L. M. Barreto and M. Naehrig. Pairing-friendly elliptic curves of prime order. In B. Preneel and S. E. Tavares (Eds.), SAC 2005, Springer LNCS 3897 (2005), 319–331.
6. D. J. Bernstein and T. Lange, Inverted Edwards coordinates. in S. Boztas, H.-F. Lu (eds), AAECC 2007, Springer LNCS 4851 (2007) 20–27.
7. D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.
8. M. van Dijk, R. Granger, D. Page, K. Rubin, A. Silverberg, M. Stam and D. Woodruff, Practical Cryptography in High Dimensional Tori. In *Advances in Cryptology – Eurocrypt'2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 234–250. Springer-Verlag, 2005.
9. I. Duursma and H-S. Lee, Tate pairing implementation for hyperelliptic curves $y^2 = x^p - x + d$. In *Advances in Cryptology – Asiacrypt'2003*, volume 2894 of *Lecture Notes in Computer Science*, pages 111–123. Springer-Verlag, 2003.
10. D. Freeman, M. Scott and E. Teske, A taxonomy of pairing-friendly elliptic curves. Cryptology ePrint Archive, Report 2006/372, 2006.
11. R. P. Gallant, R. J. Lambert and S. A. Vanstone, Improving the parallelized Pollard lambda search on anomalous binary curves, *Math. Comp.*, **69** (2000), 1699-1705.
12. R. P. Gallant, R. J. Lambert and S. A. Vanstone, Faster Point Multiplication on Elliptic Curves with Efficient Endomorphisms. In J. Kilian (Ed.), CRYPTO 2001, Springer LNCS 2139 (2001), 190–200.
13. R. Granger, D. Page, and M. Stam. On small characteristic algebraic tori in pairing-based cryptography. In *LMS Journal of Computation and Mathematics*, volume 9, pages 64–85, 2006.
14. D. Hankerson, A. Menezes and M. Scott. Software Implementation of Pairings. University of Waterloo, Centre for Applied Cryptographic Research, Technical report CACR 2008-08.

15. F. Hess, N. P. Smart and F. Vercauteren, The Eta Pairing Revisited. *IEEE Trans. Information Theory*, 52(10): 4595–4602, 2006.
16. F. Hess, Pairing lattices, to appear in this volume.
17. T. Iijima, K. Matsuo, J. Chao and S. Tsujii, Construction of Frobenius maps of twists elliptic curves and its application to elliptic scalar multiplication, SCIS 2002.
18. E. Lee, H-S. Lee, and C-M Park, Efficient and Generalized Pairing Computation on Abelian Varieties. Cryptology ePrint Archive, Report 2008/040, 2008.
19. X. Lin, C.-A. Zhao, F. Zhang and Y. Wang, Computing the Ate Pairing on Elliptic Curves with Embedding Degree $k = 9$, to appear in *IEICE transactions A*, Vol. E91-A, No.9 (2008).
20. S.-G. Lim, S.-J. Kim, I.-W. Yie, J.-M. Kim, H.-S. Lee, XTR Extended to GF(p^{6m}). In S. Vaudenay and A. M. Youssef (Eds.), SAC 2001, Springer LNCS 2259 (2001), 301–312.
21. A. Miyaji, M. Nakabayashi and S. Takano, New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Trans. Fundamentals*, E84 (2001), 1234–1243.
22. B. Möller, Algorithms for multi-exponentiation. In S. Vaudenay and A. M. Youssef (Eds.), SAC 2001, Springer LNCS 2259 (2001), 165–180.
23. F. Sica, M. Ciet, J.-J. Quisquater, Analysis of the Gallant-Lambert-Vanstone method based on efficient endomorphisms: Elliptic and hyperelliptic curves, in K. Nyberg and H. M. Heys (eds.), SAC 2002, Springer LNCS 2595 (2003) 21–36.
24. J. H. Silverman. *The Arithmetic of Elliptic Curves*. Graduate Texts in Mathematics 106. Springer-Verlag, 1986.
25. M. Stam and A. K. Lenstra, Speeding Up XTR. In C. Boyd (ed.), ASIACRYPT 2001, Springer LNCS 2248 (2001), 125–143.
26. M. Stam and A. K. Lenstra, Efficient Subgroup Exponentiation in Quadratic and Sixth Degree Extensions. In B. S. Kaliski Jr., C. K. Koc and C. Paar (Eds.), CHES 2002, Springer LNCS 2523 (2002), 318–332.
27. M. Stam, Speeding up Subgroup Cryptosystems. PhD thesis, available from http://www.cs.bris.ac.uk/Publications/Papers/2000036.pdf, 2003.
28. F. Vercauteren, Optimal pairings. Cryptology ePrint Archive, Report 2008/096, 2008.
29. M. J. Wiener and R. J. Zuccherato, Faster Attacks on Elliptic Curve Cryptosystems. In S. Tavares and H. Meijer (Eds.), SAC 1998, Springer LNCS 1556 (1999), 190–200.