J. Cryptology (2004) 17: 321–334
DOI: 10.1007/s00145-004-0311-z

Journal of
CRYPTOLOGY

Efficient Implementation of Pairing-Based Cryptosystems

Paulo S. L. M. Barreto
Escola Politécnica, Universidade de São Paulo,
Av. Prof. Luciano Gualberto, tr. 3, n. 158, s. C1-46,
BR 05508-900, São Paulo (SP), Brazil
pbarreto@larc.usp.br

Ben Lynn
Computer Science Department, Stanford University,
Stanford, CA 94305, U.S.A.
blynn@cs.stanford.edu

Michael Scott
School of Computer Applications, Dublin City University,
Ballymun, Dublin 9, Ireland
mscott@indigo.ie

Communicated by Arjen K. Lenstra

Received 29 November 2002 and revised 7 October 2003
Online publication 3 September 2004

Abstract. Pairing-based cryptosystems rely on the existence of bilinear, nondegenerate, efficiently computable maps (called pairings) over certain groups. Currently, all such pairings used in practice are related to the Tate pairing on elliptic curve groups whose embedding degree is large enough to maintain a good security level, but small enough for arithmetic operations to be feasible. In this paper we describe how to construct ordinary (non-supersingular) elliptic curves containing groups with arbitrary embedding degree, and show how to compute the Tate pairing on these groups efficiently.

Key words. Pairing-based cryptosystem, Elliptic curve construction, Efficient implementation, Tate pairing.

1. Introduction

Pairing-based cryptosystems depend on the existence of bilinear, nondegenerate, efficiently computable maps (called pairings) over certain groups. In some schemes, security depends on a new security assumption called the Bilinear Diffie–Hellman [6] or Tate–Diffie–Hellman [14] assumption. In other schemes, security depends on a standard assumption, but properties of the pairing are exploited. For example, pairings are used

to construct groups where the Computational Diffie–Hellman problem is believed to be hard, but the Decisional Diffie–Hellman assumption does not hold [18] (these groups have been called Gap Diffie–Hellman groups [7]).

Two basic problems naturally arise when implementing pairing-based systems, namely, generating groups where a suitable pairing exists, and effectively implementing the pairing. These problems can be solved in certain elliptic curve groups using maps based on the Tate pairing, which maps pairs of points to finite field elements.

A subgroup G of (the group of points of) an elliptic curve $E(\mathbb{F}_q)$ is said to have *embedding degree* k if the subgroup order r divides $q^k - 1$, but does not divide $q^i - 1$ for all $0 < i < k$. The Tate pairing [2], [13], [15] and the Weil pairing [18], [21], [26] map pairs of curve points to elements of the field $\mathbb{F}_{q^k}$, the former pairing being much more efficiently computable than the latter. The embedding degree for a random curve is usually enormous [1], rendering the computation of the Tate or Weil pairing infeasible. For a long time, the only curves known to contain groups whose embedding degree is small enough to make pairing computation feasible were supersingular curves, for example, over $\mathbb{F}_{3^m}$ where $k = 6$ [21]. It may be that the widespread suspicion that supersingular curves are unsafe is unjustified, but, nevertheless, these curves are often constructed over fields of low characteristic, making them more susceptible to discrete logarithm algorithms [9], [25]. Recently, Miyaji et al. [23] showed how to build non-supersingular curves over $\mathbb{F}_q$ of prime order via the complex multiplication method, as long as certain conditions hold for the field size q, the trace of Frobenius t [26, III.4.6], and the curve order n. Their method is based upon certain properties of the cyclotomic polynomials of order $k \in \{3, 4, 6\}$ (i.e., those of degree 2), so that here again the embedding degree is bound by $k \leq 6$, as in the supersingular case.

We investigate how to build curves with general embedding degree k, and present concrete methods for constructing them. We also describe a deterministic variant of Miller's algorithm to compute the Tate pairing that uses significantly fewer operations whenever the coordinates of one of the points is restricted to the base field $\mathbb{F}_q$.

These results extend and complement our previous work in [2] (which focuses on supersingular curves) and [3] (which only marginally considers pairing implementation). Another method for building curves with arbitrary k has been recently proposed by Dupont et al. [11]. Independent results on the implementation of the Tate pairing for supersingular curves have been obtained by Galbraith et al. [15] and especially by Duursma and Lee [12], and these results describe the best pairing algorithm presently known for supersingular curves in characteristic 3.

This paper is organized as follows. Section 2 summarizes the mathematical concepts we use in the remainder of the paper. Section 3 describes our methods to construct pairing-friendly elliptic curves. Section 4 presents our improvements for Tate pairing computation. Section 5 discusses experimental results.

2. Mathematical Preliminaries

Let p be a prime number, let m be a positive integer, and let $\mathbb{F}_{p^m}$ be the finite field with p^m elements; p is said to be the *characteristic* of $\mathbb{F}_{p^m}$, and m is its *extension degree*. We

simply write $\mathbb{F}_q$ with $q = p^m$ when the characteristic or the extension degree are known from the context or are irrelevant for the discussion. We also write $\mathbb{F}_q^* \equiv \mathbb{F}_q - \{0\}$.

An *elliptic curve* E defined over $\mathbb{F}_q$ is the set of solutions (x, y) over $\bar{\mathbb{F}}_q$ to an equation of the form $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$, where $a_i \in \mathbb{F}_q$, together with an additional *point at infinity*, denoted O. If K is an extension of the field $\mathbb{F}_q$, the set of K-rational points of E, which we denote by $E(K)$, is the set of points $(x, y) \in E$ such that $x, y \in K$.

If $p > 3$, then any elliptic curve can be affinely transformed so that its equation takes the form $y^2 = x^3 + ax + b$. The *twist* $E'(\mathbb{F}_q)$ of such a curve $E(\mathbb{F}_q)$ is given by $y^2 = x^3 + v^2ax + v^3b$ for any quadratic nonresidue in $v \in \mathbb{F}_q$.

There exists an abelian group law on E. Explicit formulas for computing the coordinates of a point $P_3 = P_1 + P_2$ from the coordinates of P_1 and P_2 are well known [26, algorithm III.2.3]. It is easy to show that $E(K)$ is a subgroup of E; for this reason we sometimes refer to $E(K)$ as "the curve $E(K)$."

The number of points of $E(K)$, denoted $\#E(K)$, is called its *order*. The *Hasse bound* states that $\#E(\mathbb{F}_q) = q + 1 - t$, where $|t| \leq 2\sqrt{q}$. The quantity t is called the *trace of Frobenius* (or simply "trace"). Curves whose trace t is a multiple of the characteristic p are called supersingular.

Let $n = \#E(K)$. The order of a point $P \in E$ is the smallest integer $r > 0$ such that $[r]P = O$. The set of r-torsion points of E, denoted $E(K)[r]$, is the set $\{P \in E(K) \mid [r]P = O\}$. The order of a point always divides the curve order. It follows that $\langle P \rangle$ is a subgroup of $E(K)[r]$, which in turn is a subgroup of $E(K)[n]$.

Let P be a point of $E(\mathbb{F}_q)$ of prime order r. The subgroup $\langle P \rangle$ is said to have *embedding degree* k for some $k > 0$ if $r \mid q^k - 1$ and $r \nmid q^s - 1$ for any $0 < s < k$. If E is supersingular, the value of k is bounded by $k \leq 6$ [21]. This bound is attained in characteristic 3 but not in characteristic 2, where the maximum achievable value is $k = 4$ [20, Section 5.2.2].

We assume $E(\mathbb{F}_{q^k})$ contains r^2 r-torsion points, that is, $E(\mathbb{F}_{q^k})[r] \cong \mathbb{Z}_r \oplus \mathbb{Z}_r$, and, in particular, there exists a point $Q \in E(\mathbb{F}_{q^k})$ of order r but linearly independent to P. This is always true when $k > 1$ [1].

For our purposes, a *divisor* is a formal sum of points on the curve $E(\mathbb{F}_{q^k})$. The *degree* of a divisor $\mathcal{A} = \sum_P a_P(P)$ is the sum $\sum_P a_P$. An abelian group structure is defined on the set of divisors by the addition of corresponding coefficients in their formal sums; in particular, $n\mathcal{A} = \sum_P (na_P)(P)$.

Let $f : E(\mathbb{F}_{q^k}) \to \mathbb{F}_{q^k}$ be a function on the curve and let $\mathcal{A} = \sum_P a_P(P)$ be a divisor of degree 0. We define $f(\mathcal{A}) \equiv \prod_P f(P)^{a_P}$. Note that, since $\sum_P a_P = 0$, $f(\mathcal{A}) = (cf)(\mathcal{A})$ for any factor $c \in \mathbb{F}_{q^k}^*$. The divisor of a function f is $(f) \equiv \sum_P \mathrm{ord}_P(f)(P)$ where $\mathrm{ord}_P(f)$ is the order of the zero or pole of f at P (if f has no zero or pole at P, then $\mathrm{ord}_P(f) = 0$). A divisor $\mathcal{A}$ is called *principal* if $\mathcal{A} = (f)$ for some function (f). It is known [20, Theorem 2.25] that a divisor $\mathcal{A} = \sum_P a_P(P)$ is principal if and only if the degree of $\mathcal{A}$ is zero and $\sum_P a_P P = O$. Two divisors $\mathcal{A}$ and $\mathcal{B}$ are equivalent, and we write $\mathcal{A} \sim \mathcal{B}$, if their difference $\mathcal{A} - \mathcal{B}$ is a principal divisor. Let $P \in E[n]$ where n is coprime to q, and let $\mathcal{A}_P$ be a divisor equivalent to $(P) - (O)$; under these circumstances the divisor $n\mathcal{A}_P$ is principal, and hence there is a function f_P such that $(f_P) = n\mathcal{A}_P = n(P) - n(O)$.

Let ℓ be a natural number coprime to q. The *Tate pairing* of order ℓ is the map $e_\ell : E(\mathbb{F}_q)[\ell] \times E(\mathbb{F}_{q^k})[\ell] \to \mathbb{F}_{q^k}^*$ given by $e_\ell(P, Q) = f_P(\mathcal{A}_Q)^{(q^k-1)/\ell}$. It satisfies the following properties:

— (Bilinearity) $e_\ell(P_1 + P_2, Q) = e_\ell(P_1, Q) \cdot e_\ell(P_2, Q)$ and $e_\ell(P, Q_1 + Q_2) = e_\ell(P, Q_1) \cdot e_\ell(P, Q_2)$ for all $P, P_1, P_2 \in E(\mathbb{F}_q)[\ell]$ and all $Q, Q_1, Q_2 \in E(\mathbb{F}_{q^k})[\ell]$. It follows that $e_\ell([a]P, Q) = e_\ell(P, [a]Q) = e_\ell(P, Q)^a$ for all $a \in \mathbb{Z}$.
— (Non-degeneracy) If $e_\ell(P, Q) = 1$ for all $Q \in E(\mathbb{F}_{q^k})[\ell]$, then $P = O$. Alternatively, for each $P \neq O$ there exists $Q \in E(\mathbb{F}_{q^k})[\ell]$ such that $e_\ell(P, Q) \neq 1$.
— (Compatibility) Let $\ell = h\ell'$. If $P \in E(\mathbb{F}_q)[\ell]$ and $Q \in E(\mathbb{F}_{q^k})[\ell']$, then $e_{\ell'}([h]P, Q) = e_\ell(P, Q)^h$.

Notice that because $P \in E(\mathbb{F}_q)$, f_P is a rational function with coefficients in $\mathbb{F}_q$.

We note that our definition of the pairing differs slightly from the standard definition [13], [14]. We restrict the first argument of e_ℓ to $E(\mathbb{F}_q)[\ell]$ and we raise $f_P(\mathcal{A}_Q)$ to the power $(q^k - 1)/\ell$, so that e_ℓ maps to certain uniquely determined coset representatives. Our definition captures the essential properties needed for cryptographical purposes.

3. Curve Construction

3.1. *Generalized MNT Curves*

Any elliptic curve E over $\mathbb{F}_q$ of order n satisfies Hasse's theorem [26, V.1.1], which states that the trace t of the Frobenius endomorphism on E, related to q and n by the equation $n = q + 1 - t$, is restricted to $|t| \leq 2\sqrt{q}$.

Let Φ_k be the kth cyclotomic polynomial [19, Definition 2.44] for some $k > 0$. MNT curves [23] are non-supersingular curves built with the complex multiplication (CM) method [5, Chapter VIII] where the curve order has the form $n = mr$ for some m and a prime r, such that $r \mid \Phi_k(t-1)$ but $r \nmid \Phi_i(t-1)$ for all $0 < i < k$. This ensures that the following lemma holds.

Lemma 1. *For MNT curves, the subgroup of order r has embedding degree k.*

Proof. We show that $r \mid q^k - 1$ but $r \nmid q^i - 1$ for all $0 < i < k$. Clearly, $q \equiv t - 1 \pmod{r}$ due to the relation $n = q + 1 - t$, hence $r \mid q^s - 1$ iff $r \mid (t-1)^s - 1$ for any $s > 0$. It is well known [19, Theorem 2.45(i)] that $x^s - 1 = \prod_{d|s} \Phi_d(x)$. Thus, since r is prime, $r \mid x^s - 1$ iff $r \mid \Phi_d(x)$ for some $d \mid s$, and hence $r \mid (t-1)^s - 1$ iff $r \mid \Phi_d(t-1)$. Therefore, $r \mid q^s - 1$ iff $r \mid \Phi_d(t-1)$ for some $d \mid s$. As we assume from the definition of MNT curves that $r \nmid \Phi_i(t-1)$ for $i < k$, it follows that $r \nmid q^i - 1$. On the other hand, we also assume that $r \mid \Phi_k(t-1)$, and hence $r \mid q^k - 1$. □

The CM equation has the form

$$DV^2 = 4q - t^2 = 4mr - (t-2)^2. \tag{1}$$

The strategy to build MNT curves seems straightforward: choose k, t, and m so that $\Phi_k(t-1)$ contains a (large) prime factor r and $q = mr + t - 1$ is prime or a prime power,

then factor $4q - t^2$ as DV^2, where D is square-free, and use the CM method to compute the curve equation coefficients. Unfortunately, this approach is not practical, because in general the CM discriminant D is too large (comparable with q), and cryptographically significant parameters would have $q \approx 2^{160}$ at least.

Miyaji et al. originally considered only $k \in \{3, 4, 6\}$ and $r = \Phi_k(t - 1)$, for which (1) reduces to a Pell equation whose solution is well known [27]. The case of arbitrary k is much harder, since no general method is known to solve Diophantine equations of degree $\deg(\Phi_k) \geq 4$. Even $k \in \{5, 8, 10, 12\}$, which leads to quartic Diophantine equations potentially solvable by the method of Tzanakis [28], has proven unsuccessful in producing cryptographically significant curves for manageable D. However, even if this direct approach remains out of reach, there are ways to generate suitable field and curve parameters, as we show next. For simplicity, we concentrate on prime q.

3.2. *Algebraic Solutions*

One possible approach to solving (1) is to look for algebraic, closed-form solutions. The complexity of this equation does not suggest by itself an immediate answer; we propose two strategies to simplify the problem and obtain solutions in certain special cases.

First strategy: Consider (1) in the form

$$DV^2 = 4h\Phi_k(t - 1) - (t - 2)^2, \tag{2}$$

where $h\Phi_k(t - 1) = mr = n$, restricted so that h and m are integers and r is a large prime factor of $\Phi_k(t-1)$. Seek to factor out this equation by setting $h = \alpha Du^2$ for some square-free α and some u, and $t - 2 = \beta h$ for some β. For arbitrary square-free D, this yields possible curve parameters as functions of u, which can be varied so that r and $q = n + t - 1$ are both prime. It is easy to see that these choices satisfy (1), and that the Hasse bound $t \leq 2\sqrt{q}$ is satisfied. Table 1 summarizes the results of this strategy for $k = 2$ and $k = 6$.

Second strategy: Parameterize the CM equation (1) for $r = \Phi_k(x)$ as $DV(x)^2 = 4m(x)\Phi_k(x) - (x - 1)^2$, where $x = t - 1$, $V = V(x)$, and $m = m(x)$. Let $V(x) = (x - 1)V'(x)$ and $m(x) = (x - 1)^2 m'(x)$ for some $V'(x)$ and $m'(x)$. Multiplying both sides of this equation by D and defining $z(x) = Dm'(x)$ and $B(x) = DV'(x)$ leads to

$$B^2(x) = 4z(x)\Phi_k(x) - D. \tag{3}$$

Thus we are confronted with the somewhat simpler problem of finding $z(x)$ such that the polynomial $4z(x)\Phi_k(x) - D$ is a perfect square.

Table 1. CM parameters obtained by the first algebraic strategy.

k	α	β	m	r	t	q	V
2	2	4	$4Du^2$	$m + 1$	$2m + 2$	$m^2 + 3m + 1$	$4u$
6	1	12	Du^2	$144m^2 + 12m + 1$	$12m + 2$	$144m^3 + 12m^2 + 13m + 1$	$2u(12m - 1)$

Table 2. Some solutions to (3) ($i, j > 0$).

k	D	$z(x)$	$B(x)$
3^i	3	1	$2x^{k/3}+1$
2^i3^j	3	1	$2x^{k/6}-1$
7^i	7	$x^{2k/7}-x^{k/7}+2$	$2x^{4k/7}+2x^{2k/7}+2x^{k/7}+1$
2^i7^j	7	$x^{k/7}+x^{k/14}+2$	$2x^{2k/7}+2x^{k/7}-2x^{k/14}+1$

The cyclotomic polynomials are known [24] to satisfy the following recurrence relations. If v is any prime dividing u, then $\Phi_{uv}(x) = \Phi_u(x^v)$. On the other hand, if $v \nmid u$, then $\Phi_{uv}(x) = \Phi_u(x^v)/\Phi_u(x)$. These recurrence relations provide a means to obtain new solutions to (3) from existing ones, as whenever $B^2(x) = 4z(x)\Phi_u(x) - D$, clearly $B^2(x^v) = 4z(x^v)\Phi_{uv}(x) - D$ if v is a prime dividing u, and $B^2(x^v) = 4[z(x^v)\Phi_u(x)]\Phi_{uv}(x) - D$ if $v \nmid u$.

Table 2 lists some noteworthy examples. Corresponding CM parameters are constructed for $t \equiv 2 \pmod{D}$ (i.e. restricted so that all divisions are exact), $m = (t-2)^2 z(t-1)/D$, a large prime factor r of $\Phi_k(t-1)$, $n = mr$, and $q = n + t - 1$. In all cases it is necessary to ensure that q is prime and that the embedding degree of the subgroup of order r is indeed k (it is at most k, but can be smaller). We point out that, in principle, this method enables the ratio m/r to get arbitrarily small for the listed D and large k.

Appendix 6 contains a detailed example of this method.

It is unclear whether these strategies can be generalized in a simple way for arbitrary k and D, since the corresponding expressions get very involved (new results along these lines have been recently found by Brezing and Weng [8]). Many such solutions probably exist. For example, for $k = D = 11$ we found the particular solution $m(x) = (4x^8 + x^6 - 3x^5 + 3x^4 + x^3 + 6x^2 - 6x + 5)/11$ where $x = t - 1$. However, the second strategy only yields solutions for small D, which could potentially have a lower security level [5, Section VIII.2], even though no specific vulnerability is known at the time of this writing. To address these limitations, the next method we propose is suitable for general k and D.

3.3. *A General Method*

Assume D and t are chosen in (1) so that $\gcd(D, t) = 1$ (otherwise there is no hope that q is prime). For convenience, let $A = 4r$ and $B = (t-2)^2$, so that (1) takes the form

$$DV^2 = Am - B.$$

We want to solve this quadratic Diophantine equation for m and V, ensuring that $q = mr + t - 1$ is prime and trying to keep m as small as possible. We do so in three steps.

First, we solve for m and z the linear Diophantine equation $Dz = Am - B$. This equation has solutions [17] for the given choice of t and D iff $\gcd(A, D) \mid B$, namely,

$$m_i = m_0 + i(D/g),$$
$$z_i = z_0 + i(A/g),$$

where $g = \gcd(A, D)$, $m_0 = (B/g)(A/g)^{-1} \bmod (D/g)$, and $z_0 = (Am_0 - B)/D$ (notice that A/g is invertible modulo D/g, because $\gcd(A/g, D/g) = 1$). All of the values g, m_0, and z_0 can be computed at once with the extended Euclidean algorithm.

Next, we solve for V and i the quadratic Diophantine equation $V^2 = z_0 + i(A/g)$. This clearly requires z_0 to be a quadratic residue (QR) modulo A/g. If so, let $\{s_u\}$ be the set of square roots of z_0 modulo A/g, that is, the set of values in the range $0 \le s_u < A/g$ for which $s_u^2 \equiv z_0 \pmod{A/g}$. Thus we can write $V = s_u + (A/g)\alpha$ where α is any integer, implying $i = (V^2 - z_0)/(A/g) = (A/g)\alpha^2 + 2s_u\alpha + (s_u^2 - z_0)/(A/g)$. Therefore the complete solution is

$$\begin{aligned} V_\alpha &= s_u + \alpha(A/g), \\ i_\alpha &= (A/g)\alpha^2 + 2s_u\alpha + (s_u^2 - z_0)/(A/g), \end{aligned}$$

for each square root s_u of z_0 modulo A/g.

Finally, we pick up any solution i_α such that $q = m_{i_\alpha} r + t - 1$ is prime. Although we can vary α to obtain this solution, to make the ratio between $\log q$ and $\log r$ as tight as possible one may prefer to restrict the search for q to $\alpha = 0$, that is, by considering $i_0 = (s_u^2 - z_0)/(A/g)$ alone and varying only t.

Experiments we conducted showed that, in practice, m tends to be close to r. Nevertheless, such solutions are perfectly suitable for most pairing-based cryptosystems, a noticeable exception being the short signature scheme of [7].

Appendix 6 contains examples of this method.

3.4. *Choosing Sparse Group Orders*

There are circumstances where one need not strictly minimize m, but rather find r with some structure, for instance, low Hamming weight, as this considerably speeds up pairing computations.

It is possible to choose a reasonably sparse r using any of the constructions we propose. This can be achieved when $\Phi_k(t-1)$ is reasonably sparse (as it happens if all prime factors of k are very small) by strengthening the condition $r \mid \Phi_k(t-1)$ to $r = \Phi_k(t-1)$, and by restricting $t - 1$ to values of low Hamming weight.

4. Computing the Tate Pairing

In this section we propose several improvements to Miller's algorithm [22] to compute the Tate pairing as defined in Section 2, in cases of cryptographical interest.

Let r be the order of the subgroup of $E(\mathbb{F}_q)$ with embedding degree $k > 1$, let $P \in E(\mathbb{F}_q)[r]$ and $Q \in E(\mathbb{F}_{q^k})$ be linearly independent points, and let f_P be the rational function with divisor $(f_P) = r(P) - r(O)$. We wish to compute the Tate pairing $e_r(P, Q) = f_P(\mathcal{A}_Q)^{(q^k-1)/r}$, where $\mathcal{A}_Q$ satisfies $\mathcal{A}_Q \sim (Q) - (O)$, and the support of $\mathcal{A}_Q$ does not contain P or O.

Lemma 2. *For any proper factor d of k, $q^d - 1$ is a factor of $(q^k - 1)/r$.*

Proof. We start with the factorization $q^k - 1 = (q^d - 1)\sum_{i=0}^{k/d-1} q^{id}$. Since the

embedding degree is $k > 1$, we have $r \mid q^k - 1$ and $r \nmid q^d - 1$. Thus $r \mid \sum_{i=0}^{k/d-1} q^{id}$, and $q^d - 1$ survives as a factor of $(q^k - 1)/r$. □

Corollary 1 (Irrelevant Factors). *For any proper factor d of k, one can freely multiply $f_P(Q)$ by any nonzero factor $x \in \mathbb{F}_{q^d}$ without affecting the pairing value.*

Proof. To compute the pairing, $f_P(Q)$ is raised to the exponent $(q^k - 1)/r$. By Lemma 2, this exponent contains a factor $q^d - 1$, thus by Fermat's Little Theorem for finite fields [19, Lemma 2.3], $x^{(q^k-1)/r} = 1$. □

Theorem 1. *Let $P \in E(\mathbb{F}_q)[r]$ and $Q \in E(\mathbb{F}_{q^k})$ be linearly independent points. Then $e_r(P, Q) = f_P(Q)^{(q^k-1)/r}$.*

Proof. Let $R \notin \{O, -P, Q, Q - P\}$ be some point on $E(\mathbb{F}_q)$, and let f'_P be a function with divisor $(f'_P) = r(P+R)-r(R) \sim (f_P)$, so that $e_r(P, Q) = f'_P((Q)-(O))^{(q^k-1)/r}$. Because f'_P does not have a zero or pole at O, we have $f'_P((Q)-(O)) = f'_P(Q)/f'_P(O)$; moreover, since $P \in E(\mathbb{F}_q)$, f'_P is defined over $\mathbb{F}_q$, and hence $f'_P(O) \in \mathbb{F}_q^*$. Corollary 1 then ensures that $f'_P(O)$ is an irrelevant factor and can be omitted from the Tate pairing computation. Therefore, $e_r(P, Q) = f'_P(Q)^{(q^k-1)/r}$.

Now $(f'_P) = r((P + R) - (R)) = r((P) - (O) + (g)) = (f_P) + r(g)$ for some rational function g, since $(P + R) - (R) \sim (P) - (O)$. Thus $f'_P = f_P g^r$, and because Q is not a zero or pole of f_P or f'_P (so that $g(Q) \in \mathbb{F}_{q^k}^*$ is well defined) it follows that $f'_P(Q)^{(q^k-1)/r} = f_P(Q)^{(q^k-1)/r} g(Q)^{q^k-1} = f_P(Q)^{(q^k-1)/r}$. □

In the next theorem, for each pair $U, V \in E(\mathbb{F}_q)$ we define the *line function* $g_{U,V}$ to be (the equation of) the line through points U and V (if $U = V$, then $g_{U,V}$ is the tangent to the curve at U, and if either one of U, V is the point at infinity O, then $g_{U,V}$ is the vertical line at the other point). The shorthand g_U stands for $g_{U,-U}$. Notice that the $g_{U,V}$ functions are defined over $\mathbb{F}_q$.

Theorem 2 (Miller's Formula). *Let P be a point on $E(\mathbb{F}_q)$ and let f_c be a function with divisor $(f_c) = c(P) - ([c]P) - (c - 1)(O)$, $c \in \mathbb{Z}$. For all $a, b \in \mathbb{Z}$, $f_{a+b}(Q) = f_a(Q) \cdot f_b(Q) \cdot g_{[a]P,[b]P}(Q)/g_{[a+b]P}(Q)$ up to a constant factor in $\mathbb{F}_q^*$.*

Proof. The divisors of the line functions satisfy

$$\begin{aligned} (g_{[a]P,[b]P}) &= ([a]P) + ([b]P) + (-[a + b]P) - 3(O), \\ (g_{[a+b]P}) &= ([a + b]P) + (-[a + b]P) - 2(O). \end{aligned}$$

Hence, $(g_{[a]P,[b]P})-(g_{[a+b]P}) = ([a]P)+([b]P)-([a+b]P)-(O)$. From the definition of f_c we see that

$$(f_{a+b}) = (a + b)(P) - ([a + b]P) - (a + b - 1)(O)$$

$$\begin{aligned} &= a(P) - ([a]P) - (a-1)(O) \\ &\quad + b(P) - ([b]P) - (b-1)(O) \\ &\quad + ([a]P) + ([b]P) - ([a+b]P) - (O) \\ &= (f_a) + (f_b) + (g_{[a]P,[b]P}) - (g_{[a+b]P}). \end{aligned}$$

Therefore $f_{a+b}(Q) = f_a(Q) \cdot f_b(Q) \cdot g_{[a]P,[b]P}(Q)/g_{[a+b]P}(Q)$. □

The f_c functions are defined over $\mathbb{F}_q$; besides, $(f_0) = (f_1) = 0$ which means that f_0 and f_1 are constants, so by Corollary 1 we can set $f_0(Q) = f_1(Q) = 1$ for simplicity. Furthermore, $f_{a+1}(Q) = f_a(Q) \cdot g_{[a]P,P}(Q)/g_{[a+1]P}(Q)$ and $f_{2a}(Q) = f_a(Q)^2 \cdot g_{[a]P,[a]P}(Q)/g_{[2a]P}(Q)$.

Recall that $r > 0$ is the order of P. Let its binary representation be $r = (r_t, \ldots, r_1, r_0)$ where $r_i \in \{0, 1\}$ and $r_t \neq 0$. Miller's algorithm (simplified via Theorem 1) computes $f_P(Q) = f_r(Q)$ for $Q \notin \{O, P\}$ by coupling the above formulas with the double-and-add method to calculate $[r]P$:

Miller's algorithm (simplified):

set $f \leftarrow 1$ and $V \leftarrow P$
for $i \leftarrow t-1, t-2, \ldots, 1, 0$ do {
  set $f \leftarrow f^2 \cdot g_{V,V}(Q)/g_{[2]V}(Q)$ and $V \leftarrow [2]V$
  if $r_i = 1$ then set $f \leftarrow f \cdot g_{V,P}(Q)/g_{V+P}(Q)$ and $V \leftarrow V + P$
}
return f

Miller's algorithm can be simplified even further if k is even, as established by the following observations [4]. Let $d = k/2$, and suppose the characteristic of the field $p > 3$. Consider the twist of the curve $E(\mathbb{F}_{q^d}) : y^2 = x^3 + ax + b$, which is given by $E'(\mathbb{F}_{q^d}) : y^2 = x^3 + v^2ax + v^3b$ where v is a quadratic nonresidue in $\mathbb{F}_{q^d}$. Pick any $Q' \in E'(\mathbb{F}_{q^d})$. The map $\Psi : (\mathrm{X}, \mathrm{Y}) \mapsto (\mathrm{v}^{-1}\mathrm{X}, (\mathrm{v}\sqrt{\mathrm{v}})^{-1}\mathrm{Y})$ describes how to map points of $E'(\mathbb{F}_{q^d})$ to points of $E(\mathbb{F}_{q^k}) - E(\mathbb{F}_q)$. Set $Q = \Psi(\mathrm{Q}')$, and note that its x-coordinate lies in $\mathbb{F}_{q^d}$.

Theorem 3 (Denominator Elimination). *Choosing Q according to these observations for even k, the $g_{[2]V}$ and g_{V+P} denominators in Miller's formula can be discarded altogether without changing the value of $e_r(P, Q)$.*

Proof. The denominators in Miller's formula have the form $g_U(Q)$. From the definition of the line functions, it is clear that $g_U(Q) \equiv x - u$, where $x \in \mathbb{F}_{q^d}$ is the abscissa of Q and $u \in \mathbb{F}_q$ is the abscissa of U. Hence $g_U(Q) \in \mathbb{F}_{q^d}$. By Corollary 1, they can be discarded without changing the pairing value. □

There is no guarantee that r divides the order of Q, though this is very likely. To determine whether Q is suitable, it suffices to check that $e_r(P, Q) \neq 1$.

In the cases $p = 2, 3$, if particular supersingular curves are used, then there exist distortion maps that take points of $E(\mathbb{F}_q)$ to linearly independent points of $E(\mathbb{F}_{q^k})$, and it turns out that the denominators can be eliminated for similar reasons [2]. We may also apply point doubling and tripling to improve performance [2], [15].

For efficiency, the map Ψ can be delayed until it is required for a pairing computation. In other words, operations on Q that do not involve the pairing can be performed on Q' instead, which lies in a smaller field, and Ψ is applied only when necessary.

Thus, if $k = 2$ pairing-based protocols can be almost completely implemented using $E(\mathbb{F}_q)$ arithmetic. Only simple support for $\mathbb{F}_{q^2}$ arithmetic is required for the pairing computation. For higher k, we suggest implementing $\mathbb{F}_{q^k}$ as $\mathbb{F}_q[x]/R_k(x)$, where $R_k(x)$ is the sparsest possible polynomial containing only terms of even degree, so that elements of $\mathbb{F}_{q^d}$ are polynomials lacking any term of odd degree and can be represented as polynomials of degree d.

5. Experimental Results

There are several opportunities for further optimization that lead to noticeable performance improvements under certain circumstances, depending on the details of the actual pairing-based cryptosystem.

For instance, raising a finite field element to a power h is usually much faster than multiplying a curve point by the scalar α, so one can benefit from the pairing bilinearity and compute $e_r(P, [\alpha]Q)$ or $e_r([\alpha]P, Q)$ as $e_r(P, Q)^\alpha$.

Besides, one often needs to compute pairings $e_r(P, Q)$ where P is either fixed (e.g., the base point on the curve) or used repeatedly (e.g., a public key). In these cases the underlying scalar multiplication in Miller's algorithm can be executed only once to precompute the coefficients of the line functions $g_U(Q)$.

However, the heaviest operation in any pairing-based cryptosystem is the pairing computation. For a 512-bit prime p, computing the Tate pairing takes 20.0 ms (8.6 ms with preprocessing) on a PIII 1 GHz machine.

For illustrative purposes, Table 3 compares the signing and verification times for RSA, DSA (without precomputation), ECDSA (without precomputation), and BLS [7] at the same security level. Timings for Boneh–Franklin identity-based encryption (IBE) are listed in Table 4. The data refers to a curve over $\mathbb{F}_p$ with a 512-bit prime p, using a subgroup of 160-bit sparse prime order r.

6. Conclusions

We have shown how to construct curves containing subgroups of arbitrary embedding degree k. Such curves are suitable for most pairing-based cryptosystems. The ratio

Table 3. Comparison of signing and verification times (in ms) on a PIII 1 GHz.

Algorithm	Signing	Verification
RSA, $\|n\| = 1024$ bits, $\|d\| = 1007$ bits	7.90	0.40
DSA, $\|p\| = 1024$ bits, $\|q\| = 160$ bits	4.09	4.87
$\mathbb{F}_p$ ECDSA, $\|p\| = 160$ bits	4.00	5.17
$\mathbb{F}_{2^{163}}$ ECDSA	5.77	7.15
$\mathbb{F}_p$ BLS (MNT), $\|p\| = 157$ bits	2.22	45.8

Table 4. IBE times on a PIII 1 GHz.

Operation	Time (ms)
IBE encryption	35 (preprocessed: 22)
IBE decryption	27 (preprocessed: 17)

$\log(q)/\log(r)$ is always larger than 1, and in general close to 2. Systematically constructing groups with arbitrary k where $\log(q)/\log(r)$ is close to 1 (or better yet, curves of prime order and prescribed k) remains an open problem.

We have also proposed efficient algorithms to implement the Tate pairing, as needed in pairing-based cryptosystems. Our algorithms are practical and lead to significant performance improvements.

Acknowledgements

We thank Dan Boneh, Steven Galbraith, Frederik Vercauteren, Nigel Smart, and the anonymous referees for their valuable comments and feedback.

Appendix A. An Example of the Algebraic Construction

This simple construction implements the second strategy of Section 3.2 and quickly yields a curve and a point of large prime order r, with embedding degree k and function $z(x)$ given by Table 3.

1. Choose $t \equiv 2 \pmod{D}$ of appropriate size at random.
2. Set $r \leftarrow \Phi_k(t-1)$. If r is not prime, restart at step 1.
3. Set $m \leftarrow (t-2)^2 \cdot z(t-1)/D$ and $n = mr$.
4. Set $q = n + t - 1$. If q is not prime, restart at step 1.
5. Use the CM method to find the curve of the form $y^2 = x^3 + B$ with discriminant D (in practice small values of B can be tested to find the correct curve [10]), and find a point of order r on the curve using, e.g., the method described in Section A11.3 of [16].

An example run of this algorithm for $k = 12$ and $D = 3$ yields

$$
\begin{aligned}
t &= 203247593909,\\
r &= 170648176572900637805671583469251009431023883 3,\\
m &= 13769861476328261174883,\\
n &= 234980175259684736902960831138646770636883178734845136408169108315 39,\\
q &= 234980175259684736902960831138646770636883178734845136410201584254 47.
\end{aligned}
$$

Here r is a 151-bit prime, and q is a 224-bit prime. The curve is quickly found as $E : y^2 = x^3 + 4$ over $\mathbb{F}_q$.

Appendix B. An Example of the General Construction

Let σ be the approximate desired size (in bits) of the subgroup order r, let D be the chosen CM discriminant, and let k be the desired embedding degree. The following procedure implements the general construction method described in Section 3.3 restricted to $\alpha = 0$, and yields a suitable field size q, a prime subgroup order r, and the curve order n (it also indirectly provides the cofactor m, which it seeks to minimize, and the trace of Frobenius t). It is straightforward, but hardly necessary in practice, to modify step 6 to consider other small values of α, say, $\alpha < 128$.

1. Choose $t \approx 2^{\sigma/\delta}$ at random, where $\delta \equiv \deg(\Phi_k)$.
2. Compute $r \leftarrow \Phi_k(t-1)$, $A \leftarrow 4r$, $B \leftarrow (t-2)^2$, and $g \leftarrow \gcd(A, D)$.
3. Check that r is prime, that $r \nmid q^d - 1$ for any $d > 0$ such that $d \mid k$, and that $g \mid B$. If either of these conditions fail, choose another t in step 1.
4. Solve for m and z the linear Diophantine equation $Dz - Am + B = 0$, namely, set $m_0 \leftarrow (B/g)(A/g)^{-1} \bmod (D/g)$, and $z_0 \leftarrow (Am_0 - B)/D$.
5. Compute all square roots s_u of z_0 modulo A/g. If z_0 is not a QR mod A/g, choose another t in step 1.
6. For each square root s_u, set $i_0 \leftarrow (s_u^2 - z_0)/(A/g)$, $m \leftarrow m_0 + i_0(D/g)$, $n \leftarrow mr$, and $q \leftarrow n + t - 1$. If q is not prime, restart with another t at step 1. Otherwise, construct a curve over $\mathbb{F}_q$ of order n and trace of Frobenius t using the CM method.

An example run of this algorithm for $k = 7$ and $D = 500003$ yields

$$
\begin{aligned}
t &= 67329606,\\
r &= 931614857617431861361911956993265396021487251 31,\\
m &= 134250909401898068393989981874150935048866951703 32,\\
n &= 125070141847460013396986527273692733814291536913611095852428963052461\backslash\\
&\quad 4109630975056367228694013492,\\
q &= 125070141847460013396986527273692733814291536913611095852428963052461\backslash\\
&\quad 4109630975056367228761343097.
\end{aligned}
$$

Here r is a 157-bit prime, and q is a 320-bit prime. The curve is quickly found as $E : y^2 = x^3 - 3x + b$ over $\mathbb{F}_q$, where

$$
\begin{aligned}
b &= 315283565391589690418903185062076693159181569566876474809008162248459\backslash\\
&\quad 256213526466473404332175506.
\end{aligned}
$$

Another example, this time for $k = 11$ and $D = 500003$:

$$
\begin{aligned}
t &= 5651493,\\
r &= 3323772180632929247773347289228681738347763229928181779465948192 2677,\\
m &= 1942952980732001725092951978115817809844683873108566791665866709 4871,\\
n &= 6457933065634855138129650480350987789632015379681348132364272137169 36\backslash\\
&\quad 8688315605252369389645580297676565301354953627247078356010452896 67,
\end{aligned}
$$

$$q = 645793306563485513812965048035098778963201537968134813236427213716936\backslash$$
$$868831560525236938964558029767656530135495362724707835601050941159.$$

Here r is a 225-bit prime, q is a 448-bit prime, and the curve is $E : y^2 = x^3 + x + b$ over $\mathbb{F}_q$, where

$$b = 114943390928260306683630109134459805121604770378075777246396805900505\backslash$$
$$342675782177132483241141214345727335963156146267084855055515119955.$$

In the final example, for $k = 2$ and $D = 40003$, the order r has low Hamming weight. Generating it needs a slight modification to the above algorithm, namely, choosing t with low Hamming weight in step 1. The size of r was chosen so that its discrete logarithm security matches the index calculus security of $\mathbb{F}^*_{q^2}$. This example is designed to provide a non-supersingular replacement curve for IBE as originally described [6]:

$$t = 146150163733090291820368483271628301965593255 3443,$$
$$r = 146150163733090291820368483271628301965593255 3443,$$
$$m = 787416166498841206906296612827344790048771921628788394310882904386034\backslash$$
$$623950169737757901981677331705791003 7,$$
$$n = 115081001659887928983183333439006823966686089624358183845462930047742\backslash$$
$$449061542192527617451661960091622653320553434335100383879042640029916\backslash$$
$$65661870388607391,$$
$$q = 115081001659887928983183333439006823966686089624358183845462930047742\backslash$$
$$449061542192527617451661960091622653335168450708409413061079488357079\backslash$$
$$48681526321160833.$$

Here r is a 161-bit prime of Hamming weight 8, q is a 512-bit prime, and the curve is $E : y^2 = x^3 - 3x + b$ over $\mathbb{F}_q$, where

$$b = 845063354749524237003327740674758499794269970671427244291565289026454\backslash$$
$$691206522422878864529897191976309217068909606291590475187329776939910\backslash$$
$$7743038519465095.$$

References

[1] R. Balasubramanian and N. Koblitz. The improbability that an elliptic curve has subexponential discrete log problem under the Menezes–Okamoto–Vanstone algorithm. *Journal of Cryptology*, 11(2):141–145, 1998.

[2] P.S.L.M. Barreto, H.Y. Kim, B. Lynn, and M. Scott. Efficient algorithms for pairing-based cryptosystems. In *Advances in Cryptology – Crypto* 2002, pages 354–368. Volume 2442 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2002.

[3] P.S.L.M. Barreto, B. Lynn, and M. Scott. Constructing elliptic curves with prescribed embedding degrees. In *Security in Communication Networks – SCN* 2002, pages 263–273. Volume 2576 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2002.

[4] P.S.L.M. Barreto, B. Lynn, and M. Scott. On the selection of pairing-friendly groups. In *Selected Areas in Cryptography – SAC* 2003, pages 17–25. Volume 3006 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2003.
[5] I. Blake, G. Seroussi, and N. Smart. *Elliptic Curves in Cryptography*. Cambridge University Press, London, 1999.
[6] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.
[7] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the Weil pairing. In *Advances in Cryptology – Asiacrypt* 2001, pages 514–532. Volume 2248 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, 2002.
[8] F. Brezing and A. Weng. Elliptic curves suitable for pairing based cryptography. Cryptology ePrint Archive, Report 2003/143, 2001. http://eprint.iacr.org/2003/143.
[9] D. Coppersmith. Fast evaluation of logarithms in fields of characteristics two. In *IEEE Transactions on Information Theory*, 30: 587–594, 1984.
[10] R. Crandall and C. Pomerance. *Prime Numbers*: *a Computational Perspective*. Springer-Verlag, Berlin, 2001.
[11] R. Dupont, A. Enge, and F. Morain. Building curves with arbitrary small MOV degree over finite prime fields. Cryptology ePrint Archive, Report 2002/094, 2002. http://eprint.iacr.org/2002/094.
[12] I. Duursma and H.-S. Lee. Tate pairing implementation for hyperelliptic curves $y^2 = x^p - x + d$. In *Advances in Cryptology – Asiacrypt* 2003, pages 111–123. Volume 2894 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2003.
[13] G. Frey, M. Müller, and H. Rück. The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems. *IEEE Transactions on Information Theory*, 45(5):1717–1719, 1999.
[14] S. Galbraith. Supersingular curves in cryptography. In *Advances in Cryptology – Asiacrypt* 2001, pages 495–513. Volume 2248 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2002.
[15] S. Galbraith, K. Harrison, and D. Soldera. Implementing the Tate pairing. In *Algorithm Number Theory Symposium – ANTS V*, pages 324–337. Volume 2369 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2002.
[16] IEEE P1363 Working Group. *Standard Specifications for Public-Key Cryptography – IEEE Std* 1363-2000, 2000.
[17] K. Ireland and M. Rosen. *A Classical Introduction to Modern Number Theory*, 2nd edition. Springer-Verlag, Berlin, 1990.
[18] A. Joux and K. Nguyen. Separating decision Diffie–Hellman from Diffie–Hellman in cryptographic groups. Cryptology ePrint Archive, Report 2001/003, 2001. http://eprint.iacr.org/2001/003.
[19] R. Lidl and H. Niederreiter. *Finite Fields*, 2nd edition. Number 20 in Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1997.
[20] A. Menezes. *Elliptic Curve Public Key Cryptosystems*. Kluwer, Dordrecht, 1993.
[21] A. Menezes, T. Okamoto, and S. Vanstone. Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Transactions on Information Theory*, 39:1639–1646, 1993.
[22] V. Miller. Short programs for functions on curves. Unpublished manuscript, 1986.
[23] A. Miyaji, M. Nakabayashi, and S. Takano. New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Transactions on Fundamentals*, E84-A(5):1234–1243, 2001.
[24] T. Nagell. *Introduction to Number Theory*, 2nd edition. Chelsea, New York, 2001.
[25] O. Schirokauer, D. Weber, and T. Denny. Discrete logarithms: the effectiveness of the index calculus method. In *Algorithm Number Theory Symposium – ANTS II*, pages 337–361. Volume 1122 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 1996.
[26] J.H. Silverman. *The Arithmetic of Elliptic Curves*. Number 106 in Graduate Texts in Mathematics. Springer-Verlag, Berlin, 1986.
[27] N.P. Smart. *The Algorithmic Resolution of Diophantine Equations*. Number 41 in London Mathematical Society Student Texts. Cambridge University Press, London, 1998.
[28] N. Tzanakis. Solving elliptic diophantine equations by estimating linear forms in elliptic logarithms: the case of quartic equations. *Acta Arithmetica*, 75:165–190, 1996.