Optimised Versions of the Ate and Twisted Ate Pairings

Seiichi Matsuda[1], Naoki Kanayama[2], Florian Hess[3], and Eiji Okamoto[2]

[1] University of Tsukuba, Japan
seichi@cipher.risk.tsukuba.ac.jp
[2] University of Tsukuba, Japan
{kanayama, okamoto}@risk.tsukuba.ac.jp
[3] Technische Universität Berlin, Germany
hess@math.tu-berlin.de

Abstract. We observe a natural generalisation of the ate and twisted ate pairings, which allow for performance improvements in non standard applications of pairings to cryptography like composite group orders. We also give a performance comparison of our pairings and the Tate, ate and twisted ate pairings for certain polynomial families based on operation count estimations and on an implementation, showing that our pairings can achieve a speedup of a factor of up to two over the other pairings.

1 Introduction

Initiated by the pioneering works [18, 13, 5] on identity based key agreement, one-round tripartite Diffie-Hellman key exchange and identity based encryption respectively, the investigation of pairings has become one of the most attractive areas in contemporary cryptographic research. A host of pairing based protocols has been developed since 2001, offering superior efficiency or greater, novel functionality over classical protocols.

The currently only known instantiations of pairings suitable for cryptography are the Weil and Tate pairings on elliptic curves or on Jacobians of more general algebraic curves. In view of the applications, efficient algorithms for computing these pairings are of great importance.

The Tate pairing on elliptic curves is usually the most efficient choice. It is generally computed using Miller's algorithm [14, 15] or a much improved version of Miller's algorithm presented in Barreto *et al* [2]. Duursma and Lee [9] subsequently introduced a very special algorithm on a class of supersingular hyperelliptic curves over finite fields. Barreto *et al* [1] generalised this algorithm to efficiently compute a particular form of the Tate pairing, called η_T pairing, on supersingular elliptic and hyperelliptic curves. The main improvement here is that the loop length in Miller's algorithm for computing the Tate pairing can usually be reduced

to at most half the length when computing the η_T pairing. Hess *et al* [12] have generalised the η_T pairing in two ways to ordinary elliptic curves, retaining the efficiency advantage of the η_T pairing over the Tate pairing and at the same time enabling larger embedding degrees. The pairings from [12] are called ate pairing and twisted ate pairing.

The dramatic efficiency improvements of the η_T pairing and the ate and twisted ate pairings over the Tate pairing are not always possible. Let E be an elliptic curve over $\mathbb{F}_q$ with $\#E(\mathbb{F}_q) = q + 1 - t$ and let r be a large prime factor of $\#E(\mathbb{F}_q)$. The loop length in Miller's algorithm for the Tate pairing is roughly $\log_2(r)$, while the loop length of the η_T and ate pairings is roughly $\log_2(|t|)$. Since $|t| \leq 2\sqrt{q}$ by the theorem of Hasse, we have roughly $\log_2(|t|) \leq (1/2)\log_2(q)$. Now, in standard situations one has approximately $\log_2(r) = \log_2(q)$, so $\log_2(|t|) \leq (1/2)\log_2(r)$ and the statement on the loop lengths follows. But as soon as $r \leq (1/2)\log_2(q)$, the Tate pairing may actually become faster than the η_T or ate pairings.

In the present paper we observe a generalisation of the ate and twisted ate pairings which we call optimised ate and optimised twisted ate pairings. The loop length of these pairings is roughly equal to $\log_2(|S|)$, where S is any integer such that $S \equiv q \bmod r$ (we choose S to be of minimal absolute value). Note that $S \equiv t - 1 \bmod r$ because of $r|\#E(\mathbb{F}_q)$ and that roughly $|S| \leq r/2$. With this choice of S we thus obtain a pairing that is always at least as fast as the Tate pairing and the ate pairings. We also provide a performance comparison of our optimised pairings and the Tate, ate and twisted ate pairings for certain polynomial families, showing that our pairings can achieve a speedup of a factor of up to two over the other pairings.

The significance of our result is twofold. First, our pairings are very natural generalisations of the ate and twisted ate pairings. Second, while our pairings do not offer a performance improvement for standard applications of pairings in cryptography, they may prove useful for special embedding degrees or composite group orders. The use of composite group orders in pairing based protocols has recently attracted much interest, see for example [6, 7]. If pairing values are to be computed in prime order subgroups with known subgroup orders (we are currently not aware of any protocols based on this situation) then our pairings can offer performance improvements.

This paper is organised as follows. Section 2 gives a brief mathematical description of the Tate pairing, η_T pairing and the ate and twisted ate pairings. Section 3 contains our main theorem about the optimised ate and optimised twisted ate pairings. Section 4 contains a performance

comparison of our pairings against the Tate, ate and twisted ate pairings, using operation count estimations and our implementation of these pairings. We draw conclusions in Section 5.

2 Background

2.1 Tate Pairing

Let $\mathbb{F}_q$ be a finite field of characteristic p and E an elliptic curve defined over $\mathbb{F}_q$. Let r be a large prime coprime to q such that $r|\#E(\mathbb{F}_q)$. The embedding degree k with respect to q and r is the smallest positive integer k such that $r|(q^k-1)$. We also require $r^2 \nmid \#E(\mathbb{F}_{q^k})$. The point at infinity of E is denoted by O.

For every $P \in E(\mathbb{F}_{q^k})$ and integer s let $f_{s,P}$ be an $\mathbb{F}_{q^k}$-rational function with divisor $\mathrm{div}(f_{s,P}) = s(P)-(sP)-(s-1)(O)$. Note that $f_{s,P}$ is uniquely defined up to non-zero scalar multiples from $\mathbb{F}_{q^k}$.

Let $P \in E(\mathbb{F}_{q^k})[r]$ and $Q \in E(\mathbb{F}_{q^k})$. Choose an arbitrary (random) $R \in E(\mathbb{F}_{q^k})$ such that $\#\{P, O, Q+R, R\} = 4$, and let $D = (Q+R)-(R)$. Then the Tate pairing is a non-degenerate bilinear pairing defined by

$$\begin{aligned}\langle\cdot,\cdot\rangle_r : E(\mathbb{F}_{q^k})[r] \times E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k}) &\rightarrow \mathbb{F}_{q^k}^{\times}/(\mathbb{F}_{q^k}^{\times})^r,\\ \langle P, Q\rangle_r &= f_{r,P}(D)\cdot\mathbb{F}_{q^k}^{\times}.\end{aligned}$$

Pairing-based protocols require unique elements (and not classes) in the domain and range of the Tate pairing. Using the isomorphisms $\phi_r : E(\mathbb{F}_{q^k})[r] \rightarrow E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})$, $Q \mapsto Q + rE(\mathbb{F}_{q^k})$ and $\chi_r : \mathbb{F}_{q^k}^{\times}/(\mathbb{F}_{q^k}^{\times})^r \rightarrow \mu_r$, $x\cdot\mathbb{F}_{q^k}^{\times} \mapsto x^{(q^k-1)/r}$ with $\mu_r = \mathbb{F}_{q^k}^{\times}[r]$, the group of r-th roots of unity, we obtain the reduced Tate pairing as

$$\begin{aligned}t : E(\mathbb{F}_{q^k})[r] \times E(\mathbb{F}_{q^k})[r] &\rightarrow \mu_r,\\ t(P,Q) = \chi_r(\langle P, \phi_r(Q)\rangle_r) &= \langle P, Q\rangle_r^{(q^k-1)/r}.\end{aligned}$$

Galbraith *et al* [11] shows that r can be replaced by any integer N such that $r|N|(q^k-1)$, i.e. $t(P,Q) = \langle P, Q\rangle_N^{(q^k-1)/N}$.

One can compute $f_{r,P}(Q)$ for $Q \in E(\mathbb{F}_{q^k})$ using Miller's algorithm. For a description of Miller's algorithm and numerous optimisations see Barreto *et al* [2]. A particularly noteworthy optimisation from [2] is

$$t(P,Q) = f_{r,P}(Q)^{(q^k-1)/r},$$

which holds for $P \in E(\mathbb{F}_q)$.

The η_T pairing and ate and twisted ate pairings, which are discussed in the next sections, are all restrictions of some power of the reduced Tate pairing to suitable subgroups of $E(\mathbb{F}_{q^k})$.

2.2 η_T Pairing

Let E be a supersingular elliptic curve with distortion map $\psi : E(\mathbb{F}_q) \to E(\mathbb{F}_{q^k})$ and let $\#E(\mathbb{F}_q) = q + 1 - t$. Barreto *et al* [1] have introduced the (reduced) η_T pairing which is defined by

$$\eta_T(P, Q) = f_{T,P}(\psi(Q))^{(q^k-1)/r}$$

for $T = q - \#E(\mathbb{F}_q) = t - 1$ and $P, Q \in E(\mathbb{F}_q)[r]$. It is a bilinear and non-degenerate pairing if certain conditions are met.

The main improvement of the η_T pairing over the Tate pairing is that the loop length in Miller's algorithm for the evaluation of $f_{T,P}$ at a point is at most only half the loop length required for the evaluation of $f_{r,P}$ at a point, if $r \approx \#E(\mathbb{F}_q)$ holds true. The reason for this is that T has at most only half the bit length of $\#E(\mathbb{F}_q)$ according to the theorem of Hasse.

2.3 Ate Pairing and Twisted Ate Pairing

The ate and twisted ate pairings have been introduced by Hess *et al* [12]. These pairings can be regarded as variations or generalisations of the η_T pairing for ordinary elliptic curves.

Let E be an ordinary elliptic curve. As in the case of the η_T pairing, let $\#E(\mathbb{F}_q) = q + 1 - t$ and $T = t - 1$. Also write $N = \gcd(T^k - 1, q^k - 1) > 0$ and $T^k - 1 = LN$. Let $\pi_q : (x, y) \mapsto (x^q, y^q)$ be the q-power Frobenius endomorphism on E and define two groups $\mathbb{G}_1 = E(\mathbb{F}_q)[r] = E[r] \cap \mathrm{Ker}(\pi_q - 1)$, $\mathbb{G}_2 = E[r] \cap \mathrm{Ker}(\pi_q - q)$. Moreover, define $f_{T,Q}^{\mathrm{norm}} = f_{T,Q}/(z^r f_{T,Q})(O)$, where z is a local uniformiser at O. Finally, let $c_T = \sum_{i=0}^{k-1} T^{k-1-i} q^i \equiv kq^{k-1} \bmod r$ (these definitions hold equally for positive or negative T).

The ate pairing is defined as

$$a_T(Q, P) = f_{T,Q}^{\mathrm{norm}}(P)^{c_T(q^k-1)/N}$$

for $Q \in \mathbb{G}_2$ and $P \in \mathbb{G}_1$. If $k | \#\mathrm{Aut}(E)$, then the twisted ate pairing is defined as

$$a_T^{\mathrm{twist}}(P, Q) = f_{T,P}(Q)^{c_T(q^k-1)/N}$$

for $P \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$. The ate pairing and twisted ate pairing are bilinear and non-degenerate if and only if $r \nmid L$.

Like the η_T pairing, the ate and twisted ate pairing can be computed using at most half the loop length in Miller's algorithm in comparison with the Tate pairing, if $r \approx \#E(\mathbb{F}_q)$.

2.4 Twists

The fairly restrictive condition $k|\#\mathrm{Aut}(E)$ for the twisted ate pairing (note $\#\mathrm{Aut}(E) \leq 6$ for ordinary E) is related to the existence of twists of E.

Let E and E' be two ordinary elliptic curves over $\mathbb{F}_q$. The curve E' is called a twist of degree d of E if there exists an isomorphism $\psi : E' \to E$ defined over $\mathbb{F}_{q^d}$ and d is minimal with this property. Then the condition $d \,|\, \#\mathrm{Aut}(E)$ holds true if and only if E admits a twist of degree d.

Table 1 contains information about the various twists of elliptic curves in characteristic ≥ 5 together with the twisting isomorphisms. The element D has to be chosen from $\mathbb{F}_q$ such that the twisting isomorphisms are properly defined over $\mathbb{F}_{q^d}$.

$d = 2$	$E : y^2 = x^3 + Ax + B,$
	$E' : y^2 = x^3 + A/D^2 x + B/D^3,$
	$\psi : E' \to E : (x, y) \mapsto (Dx, D^{3/2} y),$
$d = 4$	$E : y^2 = x^3 + Ax,$
	$E' : y^2 = x^3 + A/Dx,$
	$\psi : E' \to E : (x, y) \mapsto (D^{1/2} x, D^{3/4} y),$
$d = 3, 6$	$E : y^2 = x^3 + B,$
	$E' : y^2 = x^3 + B/D,$
	$\psi : E' \to E : (x, y) \mapsto (D^{1/3} x, D^{1/2} y).$

Table 1. Twists of Elliptic Curves in Characteristic ≥ 5

Twists can be used in conjunction with the ate and twisted ate pairing to achieve point compression and a protocol depending speed up. If E' is a twist of E of degree d and $de = k$, then it is possible to choose a twisting isomorphism $\psi : E' \to E$ such that $\psi(E'(\mathbb{F}_{q^e})) = \mathbb{G}_2$. This allows to work with $Q' = \psi^{-1}(Q)$ instead of Q. Note that in the supersingular case we

can have $E' = E$ and the twisted ate pairing then coincides with the η_T pairing.

3 Optimised Versions of the Ate and Twisted Ate Pairings

Let E be an ordinary elliptic curve over $\mathbb{F}_q$. The next theorem provides a generalisation of the ate and twisted ate pairings by replacing $T = t - 1$ with any integer S such that $S \equiv q \bmod r$.

Theorem 1. *Let S be any integer with $S \equiv q \bmod r$. Define $N = \gcd(S^k - 1, q^k - 1) > 0$ and $L = (S^k - 1)/N$. Let $c_S = \sum_{i=0}^{k-1} S^{k-1-i} q^i \bmod N$. Then*

$$a_S : \mathbb{G}_2 \times \mathbb{G}_1 \to \mu_r, \quad (Q, P) \mapsto f_{S,Q}^{\mathrm{norm}}(P)^{c_S(q^k-1)/N}$$

defines a bilinear pairing. If $k \mid \#\mathrm{Aut}(E)$ then

$$a_S^{\mathrm{twist}} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mu_r, \quad (P, Q) \mapsto f_{S,P}(Q)^{c_S(q^k-1)/N}$$

also defines a bilinear pairing. Both pairings a_S and a_S^{twist} are non-degenerate if and only if $r \nmid L$.

The relation with the reduced Tate pairing is

$$a_S(Q, P) = t(Q, P)^L \ \text{ and } \ a_S^{\mathrm{twist}}(P, Q) = t(P, Q)^L.$$

We remark that if $P = O$ or $Q = O$ then the pairing values are defined to be equal to 1. Also, if $P = Q$ (only possible for $k = 1$) then P needs to be replaced by any divisor $(P + R) - (R)$ coprime to $(Q) - (O)$ for the first pairing and by $(Q + R) - (R)$ coprime to $(P) - (O)$ for the second pairing, with $R \in E(\overline{\mathbb{F}}_q)$.

We will show that under certain conditions a suitable choice of S yields pairings a_S and a_S^{twist} which are more efficient than the ate pairing a_T and the twisted ate pairing a_T^{twist} for $T = t - 1$. For these choices of S, we call a_S and a_S^{twist} optimised ate and optimised twisted ate pairing.

Theorem 1 can also be applied to the base extensions E_e of E over $\mathbb{F}_{q^e}$ (i.e. E regarded as an elliptic curve over $\mathbb{F}_{q^e}$) for $1 \le e \le k - 1$. The subgroups $\mathbb{G}_1$ and $\mathbb{G}_2$ of $E[r]$ remain invariant under such base extensions and the embedding degree of E_e with respect to r is $k_e = k/\gcd(k, e)$. Hence Theorem 1 also holds true after replacing q by q^e and k by k_e for any e with $1 \le e \le k-1$. This observation allows for example to apply the twisted ate pairing in the case $k \nmid \#\mathrm{Aut}(E)$ and $\gcd(k, \#\mathrm{Aut}(E)) \neq 1$, where we choose $e = k/\gcd(k, \#\mathrm{Aut}(E))$ so that $k_e = \gcd(k, \#\mathrm{Aut}(E))$

and $k_e \,|\, \#\mathrm{Aut}(E_e)$. Another application is to further minimise the absolute value of S. This has been observed in [22], where a number of interesting examples are given.

Proof (of Theorem 1). The proof is essentially the same as in [12], but slightly more general. In the following, we only adapt the main arguments of [12] to our setting.

We let $\psi = \pi_q$ for the ate pairing case and $\psi = \gamma\pi_q$ for the twisted ate pairing case, where $\gamma \in \mathrm{Aut}(E)$ is an automorphism of order k such that $(\gamma\pi_q)(Q) = Q$ and $(\gamma\pi_q)(P) = qP$. If we interchange P and Q for the twisted ate pairing we have $\psi(P) = P$, $\psi(Q) = qQ = SQ$ and need to consider $f_{S,Q}(P)^{c_S(q^k-1)/N}$ like for the ate pairing. This allows us to deal with both cases simultaneously.

From Lemma 1 of [12] we obtain

$$t(Q,P) = f_{r,Q}(P)^{(q^k-1)/r} = f_{N,Q}(P)^{(q^k-1)/N}$$

and

$$\begin{aligned} t(Q,P)^L &= f_{N,Q}(P)^{L(q^k-1)/N} = f_{LN,Q}(P)^{(q^k-1)/N} \\ &= f_{S^k-1,Q}(P)^{(q^k-1)/N} \\ &= f_{S^k,Q}(P)^{(q^k-1)/N}. \end{aligned} \tag{1}$$

Lemma 2 of [1] yields

$$f_{S^k,Q} = f_{S,Q}^{S^{k-1}} f_{S,SQ}^{S^{k-2}} \cdots f_{S,S^{k-1}Q}. \tag{2}$$

Since ψ is purely inseparable of degree q, we obtain from Lemma 4 in [12]

$$f_{S,\psi^i(Q)} \circ \psi^i = f_{S,Q}^{q^i}. \tag{3}$$

We have $\psi^i(Q) = S^iQ$ and $\psi^i(P) = P$. Combining this with (2) and (3) gives

$$f_{S^k,Q}(P) = f_{S,Q}(P)^{\sum_{i=0}^{k-1} S^{k-1-i}q^i}. \tag{4}$$

Substituting (4) into (1) gives

$$t(Q,P)^L = f_{S,Q}(P)^{c_S(q^k-1)/N}. \tag{5}$$

Now (5) shows that a_S and a_S^{twist} are bilinear pairings, which are non-degenerate if and only if $r \nmid L$. □

4 Performance Evaluation

We provide some families of elliptic curves admitting a twist of degree 4 and 6, and compare the costs of our optimised pairings with the standard pairings.

4.1 Polynomial Families

Assume $q = p$. If $\Delta = 1, 2, 3$ in the CM equation $4p - t^2 = \Delta V^2$, then the corresponding elliptic curves can be generated without the full CM algorithm [10], for example by randomly choosing α and β in the equation below until the correct curve is found (but see also [17]).

$$\begin{aligned} E_1 &: y^2 = x^3 + \alpha x && (\Delta = 1) \\ E_2 &: y^2 = x^3 - 30\alpha x^2 + 56\alpha^3 && (\Delta = 2) \\ E_3 &: y^2 = x^3 + \beta && (\Delta = 3) \end{aligned}$$

The endomorphism rings are isomorphic to $\mathbb{Z}[\sqrt{-\Delta}]$ and E_1, E_2 and E_3 admit twists of degree 4, 2 and 6 respectively.

Let $\rho \equiv \log p / \log r$ be the ratio between the bit lengths of the finite field and the order of the subgroup. Some polynomial families such that $4p - t^2$ is a square polynomial have been presented

- in [8], for $k = 4$ and $\rho \sim 2$,
- in [19], for $k = 6$ and $\rho \sim 2$,
- in [10], for $k = 8$ and $\rho \sim 3/2$,
- in [4], for $k = 12$ and $\rho \sim 1$.

The details of these polynomial families are given in Appendix 1.

4.2 Efficiency Comparison

We follow the analysis of [16] and compare the Tate pairing $f_{r,P}(Q)$, ate pairing $f_{T,Q}(P)$, twisted ate pairing $f_{T^e,P}(Q)$, optimised ate pairing $f_{S,Q}(P)$ and optimised twisted ate pairing $f_{S_e,P}(Q)$ on ordinary elliptic curves admitting a twist of degree 6 when $k = 6, 12$ and of degree 4 when $k = 4, 8$. We refer to $f_{N,P}(Q)$ as a Miller-Lite operation and $f_{N,Q}(P)$ as a Miller-Full operation. We denote the cost of the Miller-Lite operation by C_{Lite} and the cost of the Miller-Full operation by C_{Full}. Assume both operations use projective coordinates. On the form $Y^2 = X^3 + AX + B$, the costs for Miller-operations are estimated as follows [12].

When $A = -3$:

$$C_{\text{Lite}} = (4S_1 + (2e+7)M_1 + S_k + M_k)\log_2 N$$
$$C_{\text{Full}} = (4S_e + 6M_e + 2eM_1 + S_k + M_k)\log_2 N$$

When $A = 0$:

$$C_{\text{Lite}} = (5S_1 + (2e+6)M_1 + S_k + M_k)\log_2 N$$
$$C_{\text{Full}} = (5S_e + 6M_e + 2eM_1 + S_k + M_k)\log_2 N$$

where $s = 2^i 3^j, M_s = 3^i 5^j M_1, S_s = M_s$ with respect to multiplication in $\mathbb{F}_{q^s}^{\times}$.

Using the parameters in Appendix 1, we estimate the loop length for each pairing. The results are given in Table 2.

Security Level	Method	Cost (average size t) Standard	Optimised
$k = 4, d = 4$	Tate	4960	
$\log_2 p \sim 320$	ate	4800	2400
$\log_2 r \sim 160$	twisted ate	4960	2480
$k = 6, d = 6$	Tate	11008	
$\log_2 p \sim 512$	ate	5504	5504
$\log_2 r \sim 256$	twisted ate	5504	5504
$k = 8, d = 4$	Tate	17664	
$\log_2 p \sim 384$	ate	16896	16896
$\log_2 r \sim 256$	twisted ate	26496	13248
$k = 12, d = 6$	Tate	26880	
$\log_2 p \sim 256$	ate	16256	16256
$\log_2 r \sim 256$	twisted ate	26880	20160

Table 2. The Costs Required for the Different Pairings

When $k = 4$ the optimised ate and optimised twisted ate pairing are twice as fast as the Tate, ate and twisted ate pairing. When $k = 8$ the optimised twisted ate pairing is more efficient than the optimised ate pairing. We conclude that our optimised pairings always run at least as fast as the Tate pairing, and the loop length of the optimised (twisted) ate pairing can be at least reduced to $\frac{\deg(r)-1}{\deg(r)}$ of the loop length of the Tate pairing when $t-1 \geq r$ for the optimised ate pairing and $(t-1)^e \geq r$ for the optimised twisted ate pairing.

4.3 Implementation Evaluation

We have implemented all pairings on $\mathbb{G}_1 \times \mathbb{G}_2$ for $k = 4$ using the GNU MP library in C++ to demonstrate the effectiveness of our proposal. Table 3 presents the running times for Tate pairing, twisted ate pairing and optimised twisted ate pairing excluding the final powering. The detailed parameters are given in Appendix 2.

Security Level	Method	Running time
$k = 4$	Tate	11.3
$\log_2 q \sim 320, \log_2 r \sim 160$	Twisted ate	11.1
MOV security ~ 1280	Optimised twisted ate	5.7

Table 3. The Running Time for the Different Pairings

5 Acknowledgement

We thank Xavier Boyen for pointing us to [6, 7].

6 Conclusion

We have described very natural optimised variants of the ate and twisted ate pairing which are simultaneous improvements over the Tate, ate and twisted ate pairings. We have provided some sample polynomial families for which the loop length in Miller's algorithm for our optimised pairings is shorter by a factor of $\frac{\deg(r)-1}{\deg(r)}$ in comparison to the loop length for the Tate pairing when $t-1 \geq r$ for the optimised ate pairing and $(t-1)^e \geq r$ for the optimised twisted ate pairing.

References

1. P.S.L.M. Barreto, S. Galbraith, C. O'hEigeartaigh, and M. Scott, "Efficient pairing computation on supersingular abelian varieties," Designs, Codes and Cryptography, Vol. 42, No. 3, (2007) pp. 239–271.
2. P.S.L.M. Barreto, H.Y. Kim, B. Lynn, and M. Scott, "Efficient algorithms for pairing-based cryptosystems," *Advances in Cryptology–CRYPTO 2002*, Lecture Notes in Computer Science 2442, Springer-Verlag (2002) pp. 354–368.
3. P.S.L.M. Barreto, Ben Lynn, and M. Scott, "On the Selection of Pairing-Friendly Groups," *Selected Areas in Cryptography–SAC 2003*, Lecture Notes in Computer Science 3006, Springer-Verlag (2003) pp. 17–25.

4. P.S.L.M. Barreto and M. Naehrig, "Pairing-Friendly Elliptic Curve of Prime Order", *Selected Areas in Cryptography–SAC 2005*, Lecture Notes in Computer Science 3897, Springer (2006) pp. 319–331.
5. D. Boneh and M. Franklin, "Identity-based Encryption from the Weil pairing", *Advances in Cryptology–CRYPTO 2001*, Lecture Notes in Computer Science 2139, Springer-Verlag (2001) pp. 213–229.
6. X. Boyen and B. Waters, "Compact Group Signatures Without Random Oracles", *Advances in Cryptology—EUROCRYPT 2006*, Lecture Notes in Computer Science 4004, Springer-Verlag (2006) pp. 427–444.
7. X. Boyen and B. Waters, "Full-Domain Subgroup Hiding and Constant-Size Group Signatures", *Public Key Cryptography—PKC 2007*, Lecture Notes in Computer Science 4450, Springer-Verlag (2007) pp. 1–15.
8. P. Duan, S. Cui and C.W. Chan, "Effective Polynomial Families for Generating More Pairing-friendly Elliptic Curve", *Cryptology ePrint Archive*, Report 2005/236, 2005. `http://eprint.iacr.org/2005/236`
9. I. Duursma and H.S. Lee, "Tate Pairing Implementation for Hyperelliptic Curves $y^2 = x^p - x + d$", *Advances in Cryptology–ASIACRYPT 2003*, Lecture Notes in Computer Science 2894, Springer-Verlag (2003) pp. 111–123.
10. D. Freeman, M. Scott and E. Teske, "A taxonomy of pairing-friendly elliptic curves", *Cryptology ePrint Archive*, Report 2006/372, 2006. `http://eprint.iacr.org/2006/372`
11. S. Galbraith, K. Harrison and S. Soldera, "Implementing the Tate pairing", *Algorithmic Number Theory Symposium–ANTS V*, Lecture Notes in Computer Science 2369, Springer-Verlag (2002) pp. 324–337.
12. F. Hess, N.P. Smart and F. Vercauteren, "The Eta Pairing Revisited", *IEEE Transaction on Information Theory*, Vol. 52, No. 10 (2006) pp. 4595–4602.
13. A. Joux, "A One Round Protocol for Tripartite Diffie-Hellman", *Algorithmic Number Theory Symposium–ANTS IV*, Lecture Notes in Computer Science 1838, Springer-Verlag (2000) pp. 385–394,
14. V.S. Miller, "Short Programs for functions on Curves", (1986) `http://crypto.stanford.edu/miller/miller.pdf`
15. V.S. Miller. "The Weil pairing and its efficient calculation", *Journal of Cryptology*, Vol. 17, No. 4 (2004) pp. 235–261.
16. N. Koblitz and A. Menezes, "Pairing-based cryptography at high security level", *Cryptography and Coding: 10th IMA International Conference*, Lecture Notes in Computer Science 3796, Springer-Verlag (2005) pp. 13–36.
17. K. Rubin and A. Silverberg, "Choosing the correct elliptic curve in the CM method", *Cryptology ePrint Archive*, Report 2007/253, 2007. `http://eprint.iacr.org/2007/253`
18. R. Sakai, K. Ohgishi and M. Kasahara, "Cryptosystems based on pairing", *Symposium on Cryptography and Information Security–SCIS 2000*, 2000.
19. M. Scott, Private communication.
20. M. Scott, "Scaling security in pairing-based protocols", *Cryptology ePrint Archive*, Report 2005/139, 2005. `http://eprint.iacr.org/2005/139`
21. M. Scott, N. Costigan, and W. Abdulwahab, "Implementing Cryptographic Pairings on Smartcards", *Workshop on Cryptographic Hardware and Embedded System–CHES 2006*, Lecture Notes in Computer Science 4249, Springer-Verlag (2006), pp. 134–147.
22. C.-A. Zhao, F. Zhang and J. Huang "A Note on the Ate Pairing", *Cryptology ePrint Archive*, Report 2007/247, 2007. `http://eprint.iacr.org/2007/247`

Appendix 1

Polynomial families for $k = 4, 6, 8, 12$ from [8, 4, 10] and the values of $e = k/\gcd(k, \#\mathrm{Aut}(E))$, $T = t - 1$, $S \equiv p \bmod r$ and $S_e \equiv p^e \bmod r$.

$k = 4$
$p = 8z^4 + 6z^2 + 2z + 1$
$r = 4z^2 + 1$
$t = 4z^2 + 2z + 2$
$\Delta V^2 = 4z^2(2z - 1)^2$
$\Delta = 1$
$e = 1$
$T = 4z^2 + 2z + 1$
$S = 2z$

$k = 6$
$p = 27z^4 + 9z^3 + 3z^2 + 3z + 1$
$r = 9z^2 + 3z + 1$
$t = 3z + 2$
$\Delta V^2 = 3z^2(6z + 1)^2$
$\Delta = 3$
$e = 1$
$T = 3z + 1$
$S = T$

$k = 8$
$p = \frac{1}{4}(81z^6 + 54z^5 + 45z^4 + 12z^3 + 13z^2 + 6z + 1)$
$r = 9z^4 + 12z^3 + 8z^2 + 4z + 1$
$t = -9z^3 - 3z^2 - 2z$
$\Delta V^2 = (3z + 1)^2$
$\Delta = 1$
$e = 2$
$T = -9z^3 - 3z^2 - 2z - 1$
$T^2 = 81z^6 + 54z^5 + 45z^4 + 30z^3 + 10z^2 + 4z + 1$
$S = T$
$S_e = p^2 \bmod r = -18z^3 - 15z^2 - 10z - 4$

$k = 12$
$p = 36z^4 + 36z^3 + 24z^2 + 6z + 1$
$r = 36z^4 + 36z^3 + 18z^2 + 6z + 1$

$t = 6z^2 + 1$
$\Delta V^2 = 3(6z^2 + 4z + 1)^2$
$\Delta = 3$
$e = 2$
$T = 6z^2$
$T^2 = 36z^4$
$S = T$
$S_e = p^2 \bmod r = -36z^3 - 18z^2 - 6z - 1$

Appendix 2

The parameters for the pairing implementation in Section 4.3.

$k = 4$
$p = 68024122034851547747794925989419190236993965539150456815120701699466168905058761705253618722974 9$ (319 bit)
$E : y^2 = x^3 + 3x$
$E' : y^2 = x^3 + (3/D)x$, where $1/D = v^2$ and $v^2 - 2 = 0$
$\#E(\mathbb{F}_p) = 68024122034851547747794925989419190236993965539033817094583612321760641102231722226473506156493 6$ (319 bit)
$\#E'(\mathbb{F}_p) = 68024122034851547747794925989419190236993965539267096535657791077171696491886059946143006166737 0$ (319 bit)
$r = 1166397205370893777055276948271688598347500051217$ (160 bit)
$t = 11663972053708937770552780282703947878011256648 14$ (160 bit)
$T = 11663972053708937770552780282703947878011256648 13$ (160 bit)
$S = 1079998706189453625613596$ (80 bit)