Minimality of the Hamming Weight of the τ-NAF for Koblitz Curves and Improved Combination with Point Halving

Roberto Maria Avanzi[1,⋆], Clemens Heuberger[2,⋆⋆], and Helmut Prodinger[3,⋆⋆⋆]

[1] Faculty of Mathematics and Horst Görtz Institute for IT Security, Ruhr-University Bochum, Germany
roberto.avanzi@ruhr-uni-bochum.de
[2] Institut für Mathematik B, Technische Universität Graz, Austria
clemens.heuberger@tugraz.at
[3] Department of Mathematics, University of Stellenbosch, South Africa
hproding@sun.ac.za

Abstract. In order to efficiently perform scalar multiplications on elliptic Koblitz curves, expansions of the scalar to a complex base associated with the Frobenius endomorphism are commonly used. One such expansion is the τ-adic NAF, introduced by Solinas. Some properties of this expansion, such as the average weight, are well known, but in the literature there is no proof of its *optimality*, i.e. that it always has minimal weight. In this paper we provide the first proof of this fact.

Point halving, being faster than doubling, is also used to perform fast scalar multiplications on generic elliptic curves over binary fields. Since its computation is more expensive than that of the Frobenius, halving was thought to be uninteresting for Koblitz curves. At PKC 2004, Avanzi, Ciet, and Sica combined Frobenius operations with one point halving to compute scalar multiplications on Koblitz curves using on average 14% less group additions than with the usual τ-and-add method without increasing memory usage. The second result of this paper is an improvement over their expansion. The new representation, called the *wide-double-NAF*, is not only simpler to compute, but it is also optimal in a suitable sense. In fact, it has minimal Hamming weight among all τ-adic expansions with digits $\{0, \pm 1\}$ that allow one halving to be inserted in the corresponding scalar multiplication algorithm. The resulting scalar multiplication requires on average 25% less group operations than the

⋆ This paper was in part written while this author was visiting the Institut für Mathematik, Technische Universität Graz, supported by the START-project Y96-MAT of the Austrian Science Fund. The author's research described in this paper has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT. The information in this document reflects only the author's views, is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

⋆⋆ Supported by the grant S8307-MAT of the Austrian Science Fund.

⋆⋆⋆ Supported by the grant NRF 2053748 of the South African National Research Foundation.

B. Preneel and S. Tavares (Eds.): SAC 2005, LNCS 3897, pp. 332–344, 2006.

Frobenius method, and is thus 12.5% faster than the previously known combination.

Keywords. Koblitz curves, scalar multiplication, point halving, τ-adic expansion, integer decomposition.

1 Introduction

The use of elliptic curves to design cryptosystems [8, 6] has become increasingly relevant in the last years and it is nowadays regulated by standards [15, 16]. The basic operation of such a cryptosystem is the *scalar multiplication*, i.e. given a point $\boldsymbol{P}$ and an integer s, to compute $s\boldsymbol{P}$. Such an operation is usually performed by a method called double-and-add: it consists in writing the scalar s as $\sum_{i=0}^{\ell} s_i 2^i$ and in evaluating $s\boldsymbol{P} = \sum_{i=0}^{\ell} s_i 2^i \boldsymbol{P}$ by a Horner-like scheme.

Some families of elliptic curves have arithmetic properties that permit very fast implementations of the scalar multiplication, making them especially attractive for the applications. Among them, the *Koblitz curves* [7] defined by

$$E_a\colon y^2 + xy = x^3 + ax^2 + 1 \qquad \text{with} \qquad a \in \{0, 1\} \tag{1}$$

over a finite field $\mathbb{F}_{2^n}$ are of particular relevance. The good performance of Koblitz curves is due to the Frobenius endomorphism τ. This is the map induced on the curve by the Frobenius automorphism of the field extension $\mathbb{F}_{2^n}/\mathbb{F}_2$, that maps a field element to its square. The evaluation of τ is much faster than the usual group law on the curve: τ is performed by squaring the coordinates, and if a suitable representation of the field $\mathbb{F}_{2^n}$ is chosen, this operation has almost negligible computational cost. The basic idea is to rewrite the scalar to the "base of τ" instead of to the base of 2, so that a "τ-and-add" scalar multiplication method using τ in place of doublings [13, 14] can be deployed. *In this paper we give a proof that the commonly used expansion to the base of τ, Solinas' τ-NAF, is optimal, i.e. its weight is minimal among all the τ-adic representations of the same integer with digits $\{0, \pm 1\}$ – in fact we prove the stronger result that the sum of the absolute values of its digits is minimal among all τ-adic expansions with rational integer coefficients.*

Point halving [5, 10] is the inverse operation to point doubling and applies to *all* elliptic curves over binary fields, not only to Koblitz curves. Its evaluation is 2 to 3 times faster than that of a doubling and it is possible to rewrite the scalar multiplication algorithm using halving instead of doubling. The resulting method is very fast, but on Koblitz curves it is slower than the τ-and-add method.

In [2] it is proposed to insert a halving in the "τ-and-add" method to further speed up scalar multiplication. This approach brings a non-negligible speedup (on average 14% with suitable representations of the fields) with respect to the use of the τ-NAF, but it is not optimal. *We show how to get an optimal representation of the scalar under the same assumptions, and we analyse the complexity.* The scalar multiplication performed using our representation is now on average 25% faster than the Frobenius method, up from 14%.

In the next section some mathematical background is recalled. Sections 3 and 4 are respectively devoted to the minimality of the τ-NAF and to the wide-double-NAF, our optimal improvement of the results from [2]. In particular § 4.5 contains a simple analysis of the Hamming weight of the wide-double-NAF. In Section 5 we conclude.

2 Background

2.1 Koblitz Curves

We consider a curve E_a defined over $\mathbb{F}_{2^n}$ by equation (1), with a (unique) subgroup G of large prime order p (standards require the cofactor to be at most 4). Set $\mu = (-1)^{1-a}$. Recall that τ is induced by the map $x \mapsto x^2$. The equation of E_a is invariant under τ, hence the map permutes the $\mathbb{F}_{2^n}$-rational points on it. G is invariant, too. It is well-known [14, Section 4.1] that for each $\boldsymbol{P} \in E_a(\mathbb{F}_{2^n})$, we have $(\tau^2 + 2)\boldsymbol{P} = \mu\tau(\boldsymbol{P})$. Thus we can identify τ with a complex number satisfying

$$\tau^2 + 2 = \mu\tau \ . \tag{2}$$

For $z \in \mathbb{Z}[\tau]$, a τ-expansion of z is an expression $\mathbf{s} = (\ldots, s_2, s_1, s_0) \in \{-1, 0, 1\}^{\mathbb{N}_0}$ where only finitely many $s_j \neq 0$ and $\mathsf{val}_\tau(\mathbf{s}) := \sum_{j\geq 0} s_j \tau^j = z$. The Hamming weight of $\mathbf{s}$ is the number of $j \geq 0$ such that $s_j \neq 0$.

For simplicity, when there is no risk of ambiguity, we write $\mathbf{s}\boldsymbol{P}$ *in place of* $\mathsf{val}(\mathbf{s})P = \sum_{j\geq 0} s_i \tau^i(\boldsymbol{P})$*, for any point* $\boldsymbol{P}$ *and* τ*-adic expansion* $\mathbf{s}$*.*

The τ-adic non-adjacent form (τ-NAF for short) of an integer $z \in \mathbb{Z}[\tau]$ is a decomposition $\mathbf{s}$ as above with the *non-adjacency* property $s_j s_{j+1} = 0$ for $j \geq 0$, similarly to the classical NAF [9]. The average *density* (that is the average ratio of non-zero bits related to the total number of bits) of a τ-NAF is $1/3$. Each integer z admits a unique τ-NAF. Its computation is easy (see for example [14]).

If $m \in \mathbb{Z}$ has a τ-expansion $\mathbf{s}$ and $\boldsymbol{P} \in E_a(\mathbb{F}_{2^n})$, $m\boldsymbol{P}$ can be computed by evaluating $\sum_{j\geq 0} s_j \tau^j(\boldsymbol{P})$ by a Horner-like scheme called τ-and-add. Clearly, the Hamming weight corresponds to the number (plus 1) of additions on the curve E_a.

Before using the τ-adic expansion in scalar multiplication we have to reduce the integer m modulo $(\tau^n - 1)/(\tau - 1)$ (note that τ^n is the identity on the curve) to keep the length of the expansion bounded by n plus a small constant. The τ-NAF of $m \bmod (\tau^n - 1)/(\tau - 1)$ is called the *reduced* τ-NAF of m. Solinas [13, 14] has a different, in practice faster approach.

2.2 Point Halving

Let now $E/\mathbb{F}_{2^n}$ be a generic elliptic curve defined by an equation

$$E \ : \ y^2 + xy = x^3 + ax^2 + b \qquad \text{with} \qquad a, b \in \mathbb{F}_{2^n}$$

(it is not necessarily a Koblitz curve) and a subgroup $G \leq E(\mathbb{F}_{2^n})$ of large prime order. Halving a point $\boldsymbol{P}$ means to find a point $\boldsymbol{R}$ such that $2\boldsymbol{R} = \boldsymbol{P}$. As described in [5, 10, 11], point halving can be performed by two field multiplications

(denoted by M) in the field $\mathbb{F}_{2^n}$, solving an equation of the type $\lambda^2+\lambda=c$ for λ (EQ) and extraction of a square root ($\sqrt{}$). An elliptic curve addition (in affine coordinates, usually the fastest system for elliptic curves in even characteristic) is done by one field inversion (I), two multiplications and one squaring (S). A point doubling requires $I+2M+2S$.

With a polynomial basis, according to [4], $S\approx\frac{1}{7.5}M$ for $n=163$ and $\frac{1}{9}M$ for $n=233$. Following [3] we assume that, on average, $I\approx 8\,M$ when $n=163$ and $I\approx 10\,M$ when $n=233$. In $\mathbb{F}_{2^{233}}$, a field defined by a trinomial, a square root can be computed in $\approx\frac{1}{8}M$ [3, Example 3.12]. For $\mathbb{F}_{2^{163}}$ only a generic method is currently known, so $\sqrt{}\approx\frac{1}{2}M$. EQ takes, experimentally $\approx\frac{2}{3}M$. If a normal basis is used S, $\sqrt{}$ and EQ have negligible costs, $I\approx 3\,M$. It is then clear that a point halving can be performed in a fraction of the time required by an addition or of a doubling.

According to the very thorough analysis in [3], halving is about two times faster than doubling. We refer the interested reader to [5, 10, 11, 3] for details.

2.3 Frobenius-Cum-Halving

Avanzi, Ciet, and Sica [2] combine the τ-NAF with a single point halving, thereby reducing the amount of point additions from $n/3$ to $2n/7$. They can therefore claim an asymptotic speed-up of $\approx 14.29\%$ on average. Their fundamental idea is that it is possible, using a single point halving, to replace some sequences of a τ-NAF having density $1/2$ and containing at least three non-zero coefficients with sequences having weight 2. Their starting point is the observation that (2) implies $\tau^3+2\tau=\mu\tau^2=\mu(\mu\tau-2)=\tau-2\mu$, hence

$$2=-\mu\big(1+\tau^2\big)\tau\ . \tag{3}$$

Therefore $2\boldsymbol{P}$ can be computed as $-\mu\big(1+\tau^2\big)\tau\boldsymbol{P}$ – which is in fact computationally more expensive, hence this relation is per se not useful. But it can be used to build telescopic sums: In fact, if $\boldsymbol{P}=2\boldsymbol{R}$ and $\boldsymbol{Q}=\tau\boldsymbol{R}$, then, for example, $(1-\tau^2+\tau^4-\tau^6+\tau^8)\boldsymbol{P}=-\mu(1+\tau^{10})\boldsymbol{Q}$, and the second expression requires less group operations than the first one even if we consider the cost of computing $\boldsymbol{Q}$.

In [2] there are three different types of sums like the one we have just seen, each of arbitrary length. For example, the first such family is of the following form

$$\left(\sum_{j=0}^{k-1}(-1)^j\tau^{2j}\right)\boldsymbol{P}=-\mu(1+(-1)^{k-1}\tau^{2k})\boldsymbol{Q}\ .$$

Their algorithm takes an input τ-NAF $\mathbf{s}$. The output is a pair of τ-adic expressions $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$ with the property that

$$\begin{aligned}\mathbf{s}\boldsymbol{P}=\mathbf{s}^{(1)}\boldsymbol{P}+\mathbf{s}^{(2)}\boldsymbol{Q}&=((-\mu)(1+\tau^2)\mathbf{s}^{(1)}+\mathbf{s}^{(2)})\boldsymbol{Q}\\&=((\mu-\tau)\mathbf{s}^{(1)}+\mathbf{s}^{(2)})\boldsymbol{Q}=(\bar\tau\mathbf{s}^{(1)}+\mathbf{s}^{(2)})\boldsymbol{Q}\ ,\end{aligned} \tag{4}$$

where $\bar\tau$ denotes the *complex conjugate* of τ. Because of this, we call the expression $\left(\begin{smallmatrix}\mathbf{s}^{(1)}\\ \mathbf{s}^{(2)}\end{smallmatrix}\right)$ a $(\bar\tau,1)$-*double expansion*. Note that $\bar\tau$, being an element of $\mathbb{Z}[\tau]$,

operates on the points of the curve, and it is natural to ask what it does: It corresponds to the operation, which we may also denote by $\bar{\tau}$, such that $\tau(\bar{\tau}P) = \bar{\tau}(\tau P) = 2P$.

The *Hamming weight* of a double expansion $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ is defined to be the sum of the Hamming weights of $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$. The input is scanned from right to left and whenever one of the above blocks is found in $\mathbf{s}$, then it is removed from $\mathbf{s}$ and the corresponding equivalent expression for $\boldsymbol{Q}$ placed in $\mathbf{s}^{(2)}$. At the end, what "remains" of $\mathbf{s}$ constitutes $\mathbf{s}^{(1)}$. We call the resulting expansion $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ the *ACS expansion*, from the initials of its inventors. The ACS expansion has an average density of 2/7.

The method can be interleaved with the τ-NAF recoding, because the latter also operated from right to left, and can be performed twice to generate $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$ independently without having to store them. A variant of the τ-and-add method is proposed that uses $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$ independently to perform the scalar multiplication without precomputing $\boldsymbol{Q}$ or storing intermediate representations. We present it as Algorithm 1 in a simpler "non interleaved" form for ease of reading, but also because we shall use it with a different recoding in Subsection 4.2. Note that it uses the inverse Frobenius operation τ^{-1} in place of τ, which is fine because it is an operation of negligible cost (like squaring) in the cases when a normal basis for $\mathbb{F}_{2^n}$ is used [1], and still very fast if the field extension is defined by a trinomial [3, § 3.2].

Note that the values ℓ_i are NOT needed in advance. In fact, the recoding of $\mathbf{s}$ into $\mathbf{s}^{(2)}$ first and $\mathbf{s}^{(1)}$ later can be done without knowing ℓ_i in advance: the results in [14] guarantee that the length of $\mathbf{s}$ will be $\approx n$, those in [2] that $\ell_i \approx n$ and the value will be known at the end of the recoding (and of the corresponding τ-and-add loop) so that they can be used immediately after that.

For the other cases, a right-to-left method based on τ is proposed. In this case the recoding must be stored first.

3 Optimality of the τ-NAF

Let τ-NAF($\mathbf{s}$) denote the τ-NAF of $\mathsf{val}_\tau(\mathbf{s})$. For any τ-expansion $\mathbf{s}$ with any (rational) integer digits, define its *cost function* as $c(\mathbf{s}) := \sum_{j\geq 0} |s_j|$. As in the case of the binary nonadjacent form introduced by Reitwiesner [9], the τ-NAF minimizes the Hamming weight. In fact, we prove the following stronger result.

Theorem 1. *Let $z \in \mathbb{Z}[\tau]$. The cost of the τ-NAF of z is minimum over all τ-expansions of z. In particular, the τ-NAF has minimal Hamming weight among all expansions with digits $\{0, \pm 1\}$.*

Proof. We prove this claim by induction on $c(\mathbf{s})$.

Without loss of generality, we may assume that $s_0 > 0$. We choose $k \in \mathbb{Z}$ such that $1 \leq s_0 - 2k \leq 2$. We have

$$\mathsf{val}_\tau(\ldots, s_3, s_2, s_1, s_0) = \mathsf{val}_\tau(\ldots, s_3, s_2 - k, s_1 + \mu k, s_0 - 2k) =: \mathsf{val}_\tau(\mathbf{s}').$$

Algorithm 1. Scalar multiplication algorithm from [2]

INPUT: A Koblitz curve E_a with corresponding parameter $\mu = (-1)^{1-a}$, a point $\boldsymbol{P}$ of odd order on E_a and a joint expansion $\left(\begin{smallmatrix}\mathbf{s}^{(1)}\\ \mathbf{s}^{(2)}\end{smallmatrix}\right)$ of length approximately n
OUTPUT: $\mathbf{s}^{(1)}\boldsymbol{P} + \mathbf{s}^{(2)}\boldsymbol{Q}$

1. ℓ_i will contain the length of $\mathbf{s}^{(i)}$
2. $\boldsymbol{X} \leftarrow s_0^{(2)}\boldsymbol{P}$
3. **for** $j = 1$ **to** $\ell_2 - 1$ **do**
4. $\quad \boldsymbol{X} \leftarrow \tau^{-1}\boldsymbol{X}$, and $\boldsymbol{X} \leftarrow \boldsymbol{X} + s_j^{(2)}\boldsymbol{P}$ $\quad \left[\text{Now } \boldsymbol{X} = \tau^{-\ell_2+1}\mathbf{s}^{(2)}\boldsymbol{P}\right]$
5. $\boldsymbol{X} \leftarrow \tau^{\ell_2-n}\boldsymbol{X}$, $\boldsymbol{X} \leftarrow \frac{1}{2}\boldsymbol{X}$ $\quad \left[\text{Now } \boldsymbol{X} = \mathbf{s}^{(2)}\tau\left(\frac{1}{2}\boldsymbol{P}\right)\right]$
6. $\boldsymbol{X} \leftarrow \boldsymbol{X} + s_0^{(1)}\boldsymbol{P}$
7. **for** $j = 1$ **to** $\ell_1 - 1$ **do**
8. $\quad \boldsymbol{X} \leftarrow \tau^{-1}\boldsymbol{X}$, and $\boldsymbol{X} \leftarrow \boldsymbol{X} + s_j^{(1)}\boldsymbol{P}$
$\qquad \left[\text{Now } X = \tau^{-\ell_1+1}\left(\mathbf{s}^{(1)}\boldsymbol{P} + \mathbf{s}^{(2)}\tau\left(\frac{1}{2}\boldsymbol{P}\right)\right) = \tau^{-\ell_1+1}s\boldsymbol{P}\right]$
9. $\boldsymbol{X} \leftarrow \tau^{\ell_1-1-n}\boldsymbol{X}$
10. **return** $(\boldsymbol{X})$

Of course, $c(\mathbf{s}') = c(\mathbf{s}) + |s_2 - k| - |s_2| + |s_1 + \mu k| - |s_1| + (s_0 - 2k) - s_0 \le c(\mathbf{s})$. Since $c(\ldots, s_3', s_2', s_1') < c(\mathbf{s}') \le c(\mathbf{s})$, we may replace this expansion by its τ-NAF by induction hypothesis without increasing its cost c. We conclude that $\mathsf{val}_\tau(\mathbf{s}) = \mathsf{val}_\tau(\mathbf{s}'')$ for some $\mathbf{s}''$ such that $s_0'' \in \{1, 2\}$, $(\ldots, s_3'', s_2'', s_1'')$ is in τ-NAF and $c(\mathbf{s}'') \le c(\mathbf{s})$.

We note that for arbitrary t_3, t_4, we have

$$\mathsf{val}_\tau(1, 0, 2) = \mathsf{val}_\tau(0, \mu, 0), \tag{5a}$$
$$\mathsf{val}_\tau(0, -\mu, 2) = \mathsf{val}_\tau(-1, 0, 0), \tag{5b}$$
$$\mathsf{val}_\tau(t_3, 0, \mu, 2) = \mathsf{val}_\tau(-\mu + t_3, 0, 0, 0) \tag{5c}$$

(note that the cost c of the left hand side is always larger than that of the right hand side) and

$$\mathsf{val}_\tau(t_3, 0, 0, 2) = \mathsf{val}_\tau(-\mu + t_3, 0, -\mu, 0), \tag{6a}$$
$$\mathsf{val}_\tau(t_4, 0, -1, 0, 2) = \mathsf{val}_\tau(1 + t_4, -\mu, 0, \mu, 0), \tag{6b}$$
$$\mathsf{val}_\tau(t_3, 0, \mu, 1) = \mathsf{val}_\tau(-\mu + t_3, 0, 0, -1), \tag{6c}$$
$$\mathsf{val}_\tau(0, -\mu, 1) = \mathsf{val}_\tau(-1, 0, -1). \tag{6d}$$

In the last four equalities, the cost c of the left hand side is not smaller than that of the right hand side and the last three or two digits of the right hand side are

already in nonadjacent form. We consider the equivalences (5) and (6) as replacement rules: "replace an occurrence of the left hand side by the corresponding right hand side". Applying these rules on $\mathbf{s}''$ and then using the induction hypothesis for the resulting expansion (in the case of the rules in (5)) or on the left part of the resulting expansion (i.e., excluding the last two or three digits) in the case of the rules in (6), our claim is proved.

4 The Wide-Double-NAF

4.1 Definition and Uniqueness

We consider $(\bar{\tau}, 1)$-*double expansions* $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$, where $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$ are just any τ-expansions of arbitrary elements of $\mathbb{Z}[\tau]$. We say that two such expansions $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ and $\binom{\mathbf{s}'^{(1)}}{\mathbf{s}'^{(2)}}$ are *equivalent* if $\bar{\tau}\,\mathsf{val}_\tau(\mathbf{s}^{(1)}) + \mathsf{val}_\tau(\mathbf{s}^{(2)}) = \bar{\tau}\,\mathsf{val}_\tau(\mathbf{s}'^{(1)}) + \mathsf{val}_\tau(\mathbf{s}'^{(2)})$; in this case we write $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}} \equiv \binom{\mathbf{s}'^{(1)}}{\mathbf{s}'^{(2)}}$.

If we have a point $\boldsymbol{P} \in E_a(\mathbb{F}_{2^n})$ and set $\boldsymbol{Q} = \tau(\frac{1}{2}\boldsymbol{P})$, the relation $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}} \equiv \binom{\mathbf{s}'^{(1)}}{\mathbf{s}'^{(2)}}$ implies that $\mathsf{val}_\tau(\mathbf{s}^{(1)})\boldsymbol{P} + \mathsf{val}_\tau(\mathbf{s}^{(2)})\boldsymbol{Q} = \mathsf{val}_\tau(\mathbf{s}'^{(1)})\boldsymbol{P} + \mathsf{val}_\tau(\mathbf{s}'^{(2)})\boldsymbol{Q}$.

The Hamming weight of a double expansion $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ is defined to be the sum of the Hamming weights of $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$.

Let now $\mathbf{s}$ be the τ-NAF of an $m \in \mathbb{Z}$. We will construct a double expansion $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ such that $\binom{\mathbf{s}}{\mathbf{0}} \equiv \binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ and with minimal Hamming weight.

Definition 1. *A double expansion* $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ *is called a* wide-double-NAF*, if* $s_j^{(i)} = \pm 1$ *implies that* $s_{j+2} = s_{j+1} = \binom{0}{0}$ *and* $s_j^{(i')} = 0$*, where* $i' = 3 - i$ *and* $j \geq 0$*.*

This means that in the language of regular expressions, a wide-double-NAF can be written as

$$\left(\varepsilon + \begin{smallmatrix}1\\0\end{smallmatrix} + \begin{smallmatrix}\bar{1}\\0\end{smallmatrix} + \begin{smallmatrix}0\\1\end{smallmatrix} + \begin{smallmatrix}0\\\bar{1}\end{smallmatrix} + \begin{smallmatrix}0&1\\0&0\end{smallmatrix} + \begin{smallmatrix}0&\bar{1}\\0&0\end{smallmatrix} + \begin{smallmatrix}0&0\\0&1\end{smallmatrix} + \begin{smallmatrix}0&0\\0&\bar{1}\end{smallmatrix}\right)\left(\begin{smallmatrix}0\\0\end{smallmatrix} + \begin{smallmatrix}0&0&1\\0&0&0\end{smallmatrix} + \begin{smallmatrix}0&0&\bar{1}\\0&0&0\end{smallmatrix} + \begin{smallmatrix}0&0&0\\0&0&1\end{smallmatrix} + \begin{smallmatrix}0&0&0\\0&0&\bar{1}\end{smallmatrix}\right)^*. \tag{7}$$

where as customary $\bar{1}$ denotes -1 (here, and in what follows the bar over 1 and μ will denote negation and not complex conjugation).

We first prove a uniqueness result.

Theorem 2. *If* $\mathbf{s}$ *and* $\mathbf{s}'$ *are* equivalent *wide-double-NAFs, then they are equal.*

The proof relies on the following extension of Solinas' [14] Lemma 28, that he used to prove the uniqueness of the τ-NAF.

Lemma 1. *Consider* $z = \sum_{j\geq 0} s_j \tau^j \in \mathbb{Z}[\tau]$*. Then*

(i) z *is divisible by* τ *in* $\mathbb{Z}[\tau]$ *if and only if* $s_0 \equiv 0 \pmod 2$*,*
(ii) z *is divisible by* τ^2 *in* $\mathbb{Z}[\tau]$ *if and only if* $s_0 + 2s_1 \equiv 0 \pmod 4$*,*
(iii) z *is divisible by* τ^3 *in* $\mathbb{Z}[\tau]$ *if and only if* $s_0 - 2\mu s_1 - 4s_2 \equiv 0 \pmod 8$*.*

The first two assertions of Lemma 1 are in Solinas' paper, the proof of the third one is straightforward and we omit it.

Proof of Theorem 2. Let $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}} \equiv \binom{\mathbf{s}'^{(1)}}{\mathbf{s}'^{(2)}}$ be two wide-double-NAFs. Without loss of generality, we may assume that $\binom{s_0^{(1)}}{s_0^{(2)}} \neq \binom{s'^{(1)}_0}{s'^{(2)}_0}$ and that $s_0^{(i)} = 1$ for some $i \in \{1, 2\}$, which implies $s_0^{(i')} = 0$ for $i' = 3 - i$ by definition of a wide-double-NAF. By (4), we have

$$\sum_{j\geq 0} \left(s_j^{(1)} - s'^{(1)}_j\right)(-\mu)(1+\tau^2)\tau^j + \sum_{j\geq 0} \left(s_j^{(2)} - s'^{(2)}_j\right)\tau^j = 0 \ . \tag{8}$$

From Lemma 1(i) we conclude that $\left(s_0^{(1)} - s'^{(1)}_0\right)(-\mu) + \left(s_0^{(2)} - s'^{(2)}_0\right) \equiv 0 \pmod 2$. Since $s_0^{(i)} = 1$ and $s_0^{(i')} = 0$, we conclude that $\binom{s'^{(1)}_0}{s'^{(2)}_0} \neq \binom{0}{0}$. This implies that $s_j^{(k)} = s'^{(k)}_j = 0$ for $1 \leq j, k \leq 2$. We set $c = -\mu(s_0^{(1)} - s'^{(1)}_0)$ and $d = (s_0^{(2)} - s'^{(2)}_0)$. From (8) we conclude that $c(1+\tau^2) + d$ is divisible by τ^3. Hence by Lemma 1

$$0 \equiv (c+d) - 4c \equiv d - 3c \pmod 8 \tag{9}$$

but, by assumption, $(c, d) \neq (0, 0)$ and $|c| + |d| = 2$. This contradicts (9).

4.2 Existence and Use

One important property of the ACS expansion $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ is that for each column at most one digit is not vanishing. The same property is, by definition, satisfied by the wide-double-NAF. Any $(\bar{\tau}, 1)$-double expansion with this property can be easily viewed, by virtue of (4), as a recoding of the number $\frac{2}{\tau}z$ to the base τ with digit set $\{0, \pm 1, \pm\bar{\tau}\}$. This allows us to make a very simple computation of the wide-double-NAF.

We have the following result:

Lemma 2. *Consider $z = s_0 + s_1\tau \in \mathbb{Z}[\tau]$. Then $\frac{2}{\tau}z = (\mu s_0 + 2s_1) - s_0\tau$. Also*

(i) $z \equiv 1 \pmod{\tau^3}$ *if and only if* $s_0 - 2\mu s_1 \equiv 1 \pmod 8$,
(ii) $z \equiv -1 \pmod{\tau^3}$ *if and only if* $s_0 - 2\mu s_1 \equiv -1 \pmod 8$,
(iii) $z \equiv \bar{\tau} \pmod{\tau^3}$ *if and only if* $s_0 - 2\mu s_1 \equiv 3\mu \pmod 8$,
(iv) $z \equiv -\bar{\tau} \pmod{\tau^3}$ *if and only if* $s_0 - 2\mu s_1 \equiv -3\mu \pmod 8$.

Sketch of the proof. The first assertion is easily verified. For (i) we have $s_0 + s_1\tau \equiv 1 \pmod{\tau^3}$ if and only if τ^3 divides $(s_0 - 1) + s_1\tau$, and at this point Lemma 1 can be applied. To prove (iii) and (iv) it is better to work with $\mu - \tau$ in place of $\bar{\tau}$.

Note that there are *eight* congruence classes modulo τ^3 (cfr. [14]) of which four correspond to elements that are divisible by τ (see Lemma 1 above). It is now clear how we can produce the wide-double-NAF of any element of $\mathbb{Z}[\tau]$: Algorithm 2 serves the purpose thereby giving also an existence proof.

The correctness is easy to prove (it is an almost immediate consequence of Lemmas 1 and 2 and of the definition of wide-double-NAF). The termination

Algorithm 2. Wide-double-NAF recoding

INPUT: An integer $s_0 + s_1\tau \in \mathbb{Z}[\tau]$

OUTPUT: Its wide-double-NAF $\left(\begin{smallmatrix}\mathbf{s}^{(1)}\\ \mathbf{s}^{(2)}\end{smallmatrix}\right)$

$(s_0, s_1) \leftarrow (\mu s_0 + 2s_1, -s_0)$ [Multiply by $\frac{2}{\tau}$]
while $((s_0, s_1)$!= $(0, 0))$ **do**
  if $s_0 \equiv 0 \pmod 2$
    output $\left(\begin{smallmatrix}0\\0\end{smallmatrix}\right)$
    $(s_0, s_1) \leftarrow (\mu\frac{s_0}{2} + s_1, -\frac{s_0}{2})$ [Divide by τ]
  else
    switch $(s_0 - 2\mu s_1 \pmod 8)$
      case 1 : $s_0 \leftarrow s_0 - 1$, **output** $\left(\begin{smallmatrix}0&0&0\\0&0&1\end{smallmatrix}\right)$
      case -1 : $s_0 \leftarrow s_0 + 1$, **output** $\left(\begin{smallmatrix}0&0&0\\0&0&\bar{1}\end{smallmatrix}\right)$
      case 3μ : $s_0 \leftarrow s_0 - \mu$, $s_1 \leftarrow s_1 + 1$, **output** $\left(\begin{smallmatrix}0&0&1\\0&0&0\end{smallmatrix}\right)$
      case -3μ : $s_0 \leftarrow s_0 + \mu$, $s_1 \leftarrow s_1 - 1$, **output** $\left(\begin{smallmatrix}0&0&\bar{1}\\0&0&0\end{smallmatrix}\right)$
    $(s_0, s_1) \leftarrow \left(\frac{1}{8}(-3\mu s_0 - 2s_1), \frac{1}{8}(s_0 - 2\mu s_1)\right)$ [Divide by τ^3]

proof follows the same arguments as in Solinas' paper [14] and in fact the length has a similar bound than that for the τ-NAF. The recoding is easy to implement and can be used with Algorithm 1, because all remarks following Algorithm 1 apply also here.

It must be noted that, in order for this recoding to be used efficiently, the quantity $s_0 + s_1\tau$ should be reduced modulo $\tau^n - 1$ as for the NAF and the width-w τ-NAF, as explained in [14] (even partial reduction is fine).

4.3 Optimality

In this section we prove that the wide-double-NAF minimizes the Hamming weight in its equivalence class. This provides also a second, but more complicated, construction of the form.

Theorem 3. *Let* $\mathbf{s}$ *be a* $(\bar{\tau}, 1)$*-double expansion. Then there exists a wide-double-NAF that is equivalent to* $\mathbf{s}$*. Its Hamming weight is not larger than that of* $\mathbf{s}$*.*

Proof. We allow arbitrary (rational) integer digits in $\mathbf{s}$ and prove the theorem by induction on

$$c(\mathbf{s}) := \sum_{j\geq 0} \left(\left|s_j^{(1)}\right| + \left|s_j^{(2)}\right|\right) .$$

By the proof of Theorem 1, we may replace $(s_j^{(i)})_{j\geq 0}$ by its τ-NAF $(s'^{(i)}_j)_{j\geq 0}$ for $i \in \{1, 2\}$ without increasing the costs c. Of course, we have $\mathbf{s} \equiv \mathbf{s}'$.

We easily check that for all $t_j^{(i)}$, we have

$$\begin{pmatrix} t_2^{(1)} & 0 & 1 \\ t_2^{(2)} & 0 & \mu \end{pmatrix} \equiv \begin{pmatrix} t_2^{(1)} & 0 & 0 \\ (\bar{\mu}+t_2^{(2)}) & 0 & 0 \end{pmatrix} , \qquad \begin{pmatrix} 0 & \bar{1} & 0 \\ t_2^{(2)} & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 & 0 \\ t_2^{(2)} & 0 & \bar{1} \end{pmatrix} ,$$

$$\begin{pmatrix} 0 & 1 \\ 0 & \bar{\mu} \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 \\ \bar{1} & 0 \end{pmatrix} , \qquad \begin{pmatrix} t_2^{(1)} & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \equiv \begin{pmatrix} t_2^{(1)} & 0 & 0 \\ 0 & 0 & \mu \end{pmatrix} ,$$

$$\begin{pmatrix} 0 & t_1^{(1)} & 0 \\ 1 & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} 0 & t_1^{(1)} & \bar{\mu} \\ 0 & 0 & 0 \end{pmatrix} , \qquad \begin{pmatrix} 0 & 0 & 1 \\ \mu & 0 & 0 \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & \bar{\mu} \end{pmatrix} , \tag{10}$$

$$\begin{pmatrix} t_5^{(1)} & t_4^{(1)} & t_3^{(1)} & 0 & 1 & 0 \\ t_5^{(2)} & t_4^{(2)} & 0 & \bar{1} & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} t_5^{(1)} & t_4^{(1)} & t_3^{(1)} & 0 & 0 & 0 \\ (\mu+t_5^{(2)}) & t_4^{(2)} & 0 & 0 & 0 & \bar{1} \end{pmatrix} .$$

We note that in all the above equivalences, the costs c decrease from the left hand side to the right hand side (even if, occasionally, digits with absolute value 2 may appear on the r.h.s). This means that if we find one of the left hand sides (or its negatives, of course) as subblocks in our double expansion $\mathbf{s}'$, we can replace this subblock by the corresponding right hand side and use the induction hypothesis to convert the resulting expansion to a wide-double-NAF not increasing the costs.

So we may assume that the left hand sides of (10) do not occur (at least in the rightmost digits). Furthermore, we have

$$\begin{pmatrix} t_4^{(1)} & t_3^{(1)} & 0 & 0 & 1 \\ t_4^{(2)} & t_3^{(2)} & 0 & \bar{1} & 0 \end{pmatrix} \equiv \begin{pmatrix} t_4^{(1)} & t_3^{(1)} & 0 & 0 & \bar{1} \\ (\mu+t_4^{(2)}) & t_3^{(2)} & 0 & 0 & 0 \end{pmatrix} , \qquad \begin{pmatrix} 0 & 1 & 0 & 1 \\ t_3^{(2)} & 0 & 0 & 0 \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 & 0 & 0 \\ (\bar{1}+t_3^{(2)}) & 0 & 0 & \bar{\mu} \end{pmatrix} ,$$

$$\begin{pmatrix} t_3^{(1)} & 0 & 1 & 0 \\ t_3^{(2)} & 0 & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} t_3^{(1)} & 0 & 0 & \mu \\ (\bar{\mu}+t_3^{(2)}) & 0 & 0 & 0 \end{pmatrix} , \qquad \begin{pmatrix} t_3^{(1)} & 0 & 0 & 0 \\ 0 & \bar{1} & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} (\bar{1}+t_3^{(1)}) & 0 & 0 & \bar{\mu} \\ 0 & 0 & 0 & 0 \end{pmatrix} ,$$

$$\begin{pmatrix} t_3^{(1)} & 0 & 0 & 1 \\ 0 & \bar{\mu} & 0 & 0 \end{pmatrix} \equiv \begin{pmatrix} (\bar{\mu}+t_3^{(1)}) & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{\mu} \end{pmatrix} , \qquad \begin{pmatrix} 0 & \mu & 0 & 0 \\ t_3^{(2)} & 0 & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 & 0 & \bar{\mu} \\ (\bar{\mu}+t_3^{(2)}) & 0 & 0 & 0 \end{pmatrix} , \tag{11}$$

$$\begin{pmatrix} t_4^{(1)} & 0 & \bar{1} & 0 & 1 \\ t_4^{(2)} & t_3^{(2)} & 0 & 0 & 0 \end{pmatrix} \equiv \begin{pmatrix} t_4^{(1)} & 0 & 0 & 0 & 0 \\ (\mu+t_4^{(2)}) & t_3^{(2)} & 0 & 0 & \bar{\mu} \end{pmatrix} , \qquad \begin{pmatrix} t_4^{(1)} & 0 & \bar{\mu} & 0 & 0 \\ t_4^{(2)} & t_3^{(2)} & 0 & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} t_4^{(1)} & 0 & 0 & 0 & \bar{\mu} \\ (1+t_4^{(2)}) & t_3^{(2)} & 0 & 0 & 0 \end{pmatrix} ,$$

$$\begin{pmatrix} t_6^{(1)} & t_5^{(1)} & t_4^{(1)} & 0 & \bar{1} & 0 & 1 \\ t_6^{(2)} & t_5^{(2)} & t_4^{(2)} & t_3^{(2)} & 0 & \bar{1} & 0 \end{pmatrix} \equiv \begin{pmatrix} t_6^{(1)} & t_5^{(1)} & t_4^{(1)} & 0 & 0 & 0 \\ (\bar{\mu}+t_6^{(2)}) & t_5^{(2)} & t_4^{(2)} & (\bar{1}+t_3^{(2)}) & 0 & 0 & \mu \end{pmatrix} .$$

Note that for every $\mathbf{s}'$ found above, the least significant columns of $\mathbf{s}'$ are found in the l.h.s. of exactly one of the equivalences (11). Thus we can replace them with the corresponding block in the r.h.s. to obtain a new expansion $\mathbf{s}''$. In each of these equivalences, the costs do not increase from left to right and the last three digits of the right hand side always form a block that is allowed in a wide-double-NAF. In particular $\mathbf{s}''$ has cost not larger than $\mathbf{s}'$ (and thus not larger than the cost of $\mathbf{s}$). Hence we can apply the induction hypothesis to $\mathbf{s}''$ with the last three digits removed. This proves the Theorem. (Note that after applying one of the replacements (11), patterns of the l.h.s.'s of (10) may appear again and these should be replaced with the corresponding r.h.s.'s too, should one desire to formulate a constructive version of this theorem.)

4.4 An Example

Let us consider the rational integer 195. If $a = 1$ in (1), then the τ-NAF of 195 is $\tau^{16} + \tau^{14} + \tau^{10} + \tau^7 - \tau^5 + \tau^2 - 1$ or

$$\mathsf{val}_\tau(10100010010\bar{1}00010\bar{1}) = 195 \ .$$

The weight is 7 and the ACS recoding has also weight 7 (in fact, no subsequence of the given τ-NAF can be simplified by the ACS method, hence the output is identical with the input).

However, Algorithm 2 gives the following wide-double-NAF

$$\begin{pmatrix} 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \bar{1}\ 0\ 0\ 0\ 0 \\ 0\ 0\ \bar{1}\ 0\ 0\ \bar{1}\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1 \end{pmatrix} \ ,$$

that has weight 5.

4.5 Analysis

We now analyze the wide-double-NAF using methods from average case analysis of algorithms, cf. for instance [12].

To calculate the asymptotic density of the wide-double-NAF it is sufficient to sum up the Hamming weights of all wide-double-NAF's of a certain length N and divide it by the number of all wide-double-NAF's of the same length. A more detailed analysis based on the τ-NAF or the unique expansion with digits $\{0, 1\}$ for the input statistics is beyond the scope of this paper, but the main term would agree.

We define $a_{M,N}$ to be the number of wide-double-NAF's of length N and Hamming weight M and consider the generating function

$$G(Y, Z) := \sum_{N \geq 0} \sum_{M \geq 0} a_{M,N} Y^M Z^N \ .$$

This function can be derived from our regular expression in (7) by labelling contributions to the Hamming weight by Y and to the length by Z and by transforming $(\cdots)^*$ to $(1 - (\cdots))^{-1}$. Thus we get

$$G(Y, Z) = \frac{1 + 4YZ + 4YZ^2}{1 - (Z + 4YZ^3)} \ .$$

Obviously, the number W_N of wide-double-NAF's of length N equals the coefficient of Z^N in

$$G(1, Z) = \frac{1 + 4Z + 4Z^2}{1 - Z - 4Z^3} = \frac{2}{1 - 2\,Z} - \frac{1}{1 + Z + 2\,Z^2} \ .$$

We obtain

$$W_N = 2^{N+1} + O(2^{N/2}) \ .$$

We differentiate $G(Y,Z)$ with respect to Y and set $Y=1$,

$$\left.\frac{\partial G(Y,Z)}{\partial Y}\right|_{Y=1} = \frac{1}{2(1-2Z)^2} + \frac{1}{4(1-2Z)} + \frac{-3-Z}{4(1+Z+2Z^2)^2} + \frac{Z}{4(1+Z+2Z^2)} \ ,$$

and extract the coefficient of Z^N to obtain the sum H_N of the Hamming weights of all wide-double-NAF's of length N as

$$H_N = \left(N+\frac{3}{2}\right)\cdot 2^{N-1} + O(N2^{N/2}) \ .$$

Dividing H_N by W_N, we proved

Theorem 4. *The expected Hamming weight of a wide-double-NAF of length N equals*

$$\frac{1}{4}N + \frac{3}{8} + O\left(\frac{N}{2^{N/2}}\right) \ .$$

Hence the wide-double-NAF achieves an average improvement of 25 % over the τ-and-add algorithm.

5 Final Remarks and Conclusions

In this paper we consider a few problems related to τ-adic expansions associated to Koblitz curves.

The first problem is the optimality of Solinas' τ-NAF. We give the first proof that the τ-NAF has minimal weight among all the τ-adic expansions with digit set $\{0,\pm 1\}$. In fact we prove a stronger result, namely that the τ-NAF has cost (sum of the absolute values of the digits) minimal among the costs of all τ-adic representations with arbitrary rational integer digits (Theorem 1).

Then we consider a result presented at PKC 2004. There, Avanzi, Ciet, and Sica [2] showed that one could perform a scalar multiplication on Koblitz curves by "inserting" a point halving in a τ-and-add scalar multiplications, thereby reducing the number of group additions required. They attained, under suitable conditions, a reduction of 14.3% of the number of group additions with respect to the plain τ-and-add method on average without increasing memory requirements. The method is thus just a faster drop-in replacement for the τ-and-add method. We improve on this result under the same assumptions they made, bringing the reduction to 25% on average. The corresponding expansion of the scalar is the *wide-double-NAF*: we construct it (cf. Subsection 4.2), we prove its uniqueness (Theorem 2) and a suitable minimality property (Theorem 3), and we carefully analyse its expected Hamming weight (Theorem 4).

A speed-up is achieved using the new recoding together with the scalar multiplication algorithm from [2] (cf. Algorithm 1). Due to the increased number of Frobenius applications, the speed-up in a real world implementations may be smaller

than 25%: in [2] an effective speedup of 12.5% was found on standard Koblitz curves over the field $\mathbb{F}_{2^{233}}$ using normal bases. By repeating the computations done in [2, § 4.2] using the new density $1/4$ in place of $2/7$ we expect our method to bring at least an improvement of 23% on average under the same conditions. Like the method in [2] it is a drop-in replacement for the τ-and-add method.

We also note that the decrease of group operations from $2/7n$ to $1/4n$ represents a reduction of 12.5%, i.e. our method can perform on average 12.5% faster than the previous combination of the Frobenius with the halving.

Acknowledgements. The authors wish to express their gratitude to the anonymous reviewers for their remarks and suggestions.

References

1. D. W. Ash, I. F. Blake and S. Vanstone. *Low complexity normal bases.* Discrete Applied Math. **25**, pp. 191–210, 1989.
2. R. M. Avanzi, M. Ciet, and F. Sica. *Faster Scalar Multiplication on Koblitz Curves combining Point Halving with the Frobenius Endomorphism.* Proceedings of PKC 2004, LNCS 2947, 28–40. Springer, 2004.
3. K. Fong, D. Hankerson, J. Lopez and A. Menezes. *Field inversion and point halving revisited.* IEEE Trans. on Computers **53** (8), pp. 1047–1059. August 2004.
4. D. Hankerson, J. Lopez-Hernandez, and A. Menezes. *Software Implementatin of Elliptic Curve Cryprography over Binary Fields.* In: *Proceedings of CHES 2000.* LNCS 1965, pp. 1–24. Springer, 2001.
5. E. W. Knudsen. *Elliptic Scalar Multiplication Using Point Halving.* In: *Proocedings of ASIACRYPT 1999*, LNCS 1716, pp. 135–149. Springer, 1999.
6. N. Koblitz. *Elliptic curve cryptosystems.* Mathematics of computation **48**, pp. 203–209, 1987.
7. N. Koblitz. *CM-curves with good cryptographic properties.* In: *Proceedings of CRYPTO 1991*, LNCS 576, pp. 279–287. Springer, 1991.
8. V. S. Miller. *Use of elliptic curves in cryptography.* In: *Proceedings of CRYPTO '85.* LNCS 218, pp. 417–426. Springer, 1986.
9. G. W. Reitwiesner. *Binary arithmetic.* Advances in Computers **1**, pp. 231–308, 1960.
10. R. Schroeppel. *Point halving wins big.* Talks at: (i) Midwest Arithmetical Geometry in Cryptography Workshop, November 17–19, 2000, University of Illinois at Urbana-Champaign; and (ii) ECC 2001 Workshop, October 29–31, 2001, University of Waterloo, Ontario, Canada.
11. R. Schroeppel. *Elliptic curve point ambiguity resolution apparatus and method.* International Application Number PCT/US00/31014, filed 9 November 2000.
12. R. Sedgewick and P. Flajolet. *An Introduction to the Analysis of Algorithms.* Addison-Wesley, 1996.
13. J. A. Solinas. *An improved algorithm for arithmetic on a family of elliptic curves.* In: *Proceedings of CRYPTO 1997*, LNCS 1294, pp. 357–371. Springer, 1997.
14. J. A. Solinas. *Efficient Arithmetic on Koblitz Curves.* Designs, Codes and Cryptography **19** (2/3), pp. 125–179, 2000.
15. IEEE Std 1363-2000. *IEEE Standard Specifications for Public-Key Cryptography.* IEEE Computer Society, August 29, 2000.
16. National Institute of Standards and Technology. *Digital Signature Standard.* FIPS Publication 186-2, February 2000.