On Redundant τ-adic Expansions and Non-Adjacent Digit Sets

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 AT ruhr-uni-bochum.de
[2] Institut für Mathematik B, Technische Universität Graz, Austria
clemens.heuberger AT tugraz.at
[3] Department of Mathematics, University of Stellenbosch, South Africa
hproding AT sun.ac.za

Abstract. This paper studies τ-adic expansions of scalars, which are important in the design of scalar multiplication algorithms for Koblitz Curves, but are also less understood than their binary counterparts.

At Crypto '97 Solinas introduced the width-w τ-adic non-adjacent form for use with Koblitz curves. It is an expansion of integers $z = \sum_{i=0}^{\ell} z_i \tau^i$, where τ is a quadratic integer depending on the curve, such that $z_i \neq 0$ implies $z_{w+i-1} = \ldots = z_{i+1} = 0$, like the sliding window binary recodings of integers. We show that the digit sets described by Solinas, formed by elements of minimal norm in their residue classes, are uniquely determined. However, unlike for binary representations, syntactic constraints do not necessarily imply minimality of weight.

Digit sets that permit recoding of all inputs are characterized, thus extending the line of research begun by Muir and Stinson at SAC 2003 to the Koblitz Curve setting.

Two new digit sets are introduced with useful properties; one set makes precomputations easier, the second set is suitable for low-memory applications, generalising an approach started by Avanzi, Ciet, and Sica at PKC 2004 and continued by several authors since, including Okeya, Takagi and Vuillaume. Results by Solinas, and by Blake, Murty, and Xu are generalized.

Termination, optimality, and cryptographic applications are considered. The most important application is the ability to perform arbitrary windowed scalar multiplication on Koblitz curves without storing any precomputations first, thus reducing memory storage to just one point and the scalar itself.

1 Introduction

Elliptic curves (EC), as a cryptographic primitive [13, 11], are now well established and standardised [22, 23]. The performance of an EC cryptosystem depends on the efficiency of the fundamental operation, the *scalar multiplication*, i.e. the computation of the multiple $s \cdot P$ of a point P by an integer s. Among all EC, *Koblitz curves* [12], defined by the equation

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

over the finite field $\mathbb{F}_{2^n}$, permit particularly efficient implementation of scalar multiplication. Key to their good performance is the Frobenius endomorphism τ, i.e. the map induced on $E_a(\mathbb{F}_{2^n})$ by the Frobenius automorphism of the field extension $\mathbb{F}_{2^n}/\mathbb{F}_2$, that maps field elements to their squares.

Set $\mu = (-1)^{1-a}$. It is known [21, Section 4.1] that τ permutes the points on $E_a(\mathbb{F}_{2^n})$, and $(\tau^2 + 2)P = \mu\tau(P)$ for all points P. We identify τ with a root of

$$\tau^2 - \mu\tau + 2 = 0 \ . \tag{2}$$

If we write an integer z as $\sum_{i=0}^{\ell} z_i\tau^i$, where the digits z_i belong to a suitably defined digit set $\mathcal{D}$, then we can compute $z \cdot P$ as $\sum_{i=0}^{\ell} z_i\tau^i(P)$ via a Horner scheme. The resulting method [12, 20, 21] is called a "τ-and-add" method because it replaces the doubling with a Frobenius operation in the classic double-and-add scalar multiplication algorithm. Since a Frobenius operation is much faster than group doubling, scalar multiplication on Koblitz curves is a very fast operation.

The elements $d \cdot P$ for all $d \in \mathcal{D}$ must be computed before the main loop of the Horner scheme begins. Larger digit sets usually correspond to representations $\sum_{i=0}^{\ell} z_i\tau^i$ with fewer non-zero coefficients, which in turn translates to less group additions. The recipe for optimal performance is a balance between digit set size and number of non-zero coefficients.

Solinas [20, 21] considers the residue classes in $\mathbb{Z}[\tau]$ modulo τ^w which are coprime to τ, and forms a digit set comprising the zero and an element of minimal norm from each residue class that is coprime to τ. We prove in Theorem 2 that such elements are unique, hence Solinas' digit set is uniquely determined. It has cardinality $1 + 2^{w-1}$. Solinas' recoding enjoys the *width-w non-adjacent property*

$$z_i \neq 0 \qquad \text{implies} \qquad z_{w+i-1} = \ldots = z_{i+1} = 0 \ , \tag{3}$$

and is called the τ-adic width-w non-adjacent form (or τ-w-NAF for short). Every integer admits a unique τ-w-NAF.

We call a digit set that allows us to write each integer as a recoding satisfying property (3) a *(width-w) non-adjacent digit set*, or w-NADS for short. Our Theorem 1 is a criterion for establishing whether a given digit set is a w-NADS, which is very different in substance from the criterion of Blake, Murty, and Xu [6]. This line of research, i.e. the characterisation of digit sets which allow recoding with a non-adjacency condition, was initiated by Muir and Stinson in [14].

Our criterion is applied to digit sets introduced and analysed in §§ 2.3 and 2.4. We can prove under which conditions the first set is a w-NADS (Theorem 3), and give precise estimates of the length of the recoding (Theorem 4). The second digit set corresponds, in a suitable sense, to "repeated point halvings" (cf. Theorem 5) and is used to design a width-w scalar multiplication algorithm without precomputations. Among the other results in Section 2 are the facts that the τ-adic w-NAF as defined by Solinas is not optimal, and that it is not possible to compute minimal expansions by a deterministic finite automaton.

In Section 3 we discuss the relevance of our results for cryptographic applications. We conclude in Section 4. Some of the proofs are contained in Appendices.

2 Digit Sets

Let $\mu \in \{\pm 1\}$, τ be a root of equation (2) and $\bar{\tau}$ the complex conjugate of τ. Note that $2/\tau = \bar{\tau} = \mu - \tau = -\mu(1+\tau^2)$. We will consider digit expansions to the base of τ of integers in $\mathbb{Z}[\tau]$. Note that $\mathbb{Z}[\tau]$ is the ring of algebraic integers of $\mathbb{Q}(\sqrt{-7})$. It is well known that $\mathbb{Z}[\tau]$ is a Euclidean domain and therefore a factorial ring.

Definition 1. *Let $\mathcal{D}$ be a (finite) subset of $\mathbb{Z}[\tau]$ containing 0 and $w \geq 1$ be an integer. A $\mathcal{D}$-*expansion *of $z \in \mathbb{Z}[\tau]$ is a sequence $\boldsymbol{\varepsilon} = (\varepsilon_j)_{j \geq 0} \in \mathcal{D}^{\mathbb{N}_0}$ such that*

1. *Only a finite number of the digits ε_j is nonzero.*
2. $\mathsf{value}(\boldsymbol{\varepsilon}) := \sum_{j \geq 0} \varepsilon_j \tau^j = z$*, i.e., $\boldsymbol{\varepsilon}$ is indeed an expansion of z.*

The Hamming weight *of $\boldsymbol{\varepsilon}$ is the number of nonzero digits ε_j. The* length *of $\boldsymbol{\varepsilon}$ is defined as*

$$\mathsf{length}(\boldsymbol{\varepsilon}) := 1 + \max\{j : \varepsilon_j \neq 0\} \ .$$

*A $\mathcal{D}$-expansion of z is called a $\mathcal{D}$-w-*Non-Adjacent-Form ($\mathcal{D}$-w-NAF) of *z, if*

3. *Each block $(\varepsilon_{j+w-1}, \ldots, \varepsilon_j)$ of w consecutive digits contains at most one nonzero digit ε_k, $j \leq k \leq j+w-1$.*

*A $\{0, \pm 1\}$-2-NAF is also called a τ-*NAF.

*The set $\mathcal{D}$ is called a w-*Non-Adjacent-Digit-Set (w-NADS)*, if each $z \in \mathbb{Z}[\tau]$ has a $\mathcal{D}$-w-NAF.*

Typically, we will choose $\mathcal{D}$ to be a set of cardinality $1 + 2^{w-1}$, but we do not require this in the definition. One aim of this paper is to investigate which $\mathcal{D}$ are in fact w-NADS, and we shall usually restrict ourselves to digit sets formed by adjoining the 0 to a reduced residue system τ^w, which is defined as usual:

Definition 2. *Let $w \geq 1$ a natural number. A reduced residue system $\mathcal{D}'$ for the number ring $\mathbb{Z}[\tau]$ modulo τ^w is a set of representatives for the congruence classes of $\mathbb{Z}[\tau]$ modulo τ^w that are coprime to τ.*

For a digit set $\mathcal{D}$ for $\mathbb{Z}[\tau]$ formed by 0 together with a reduced residue system, the following algorithm either recodes an integer $z \in \mathbb{Z}[\tau]$ to the base of τ, or enters in a infinite loop for some inputs when $\mathcal{D}$ is not a NADS.

Algorithm 1. General windowed integer recoding

INPUT: An element z from $\mathbb{Z}[\tau]$, a natural number $w \geq 1$ and a reduced residue system $\mathcal{D}'$ for the number ring R modulo τ^w.
OUTPUT: A representation $z = \sum_{j=0}^{\ell-1} z_j \tau^j$ of length ℓ of the integer z with the property that if $z_j \neq 0$ then $z_{j+i} = 0$ for $1 \leq i < w$.

```
$j \leftarrow 0, u \leftarrow z$
while $u \neq 0$ do
    if $\tau \mid u$ then
        $z_j \leftarrow 0$                                        [Output $0$]
    else
        Let $z_j \in \mathcal{D}'$ s.t. $s_j \equiv z \pmod{\tau^w}$                [Output $z_j$]
    $u \leftarrow u - z_j,\ u \leftarrow u/\tau,\ j \leftarrow j+1$
$\ell \leftarrow j$
return $(\{z_j\}_{j=0}^{\ell-1}, \ell)$
```

Example 1. Just having a valid digit set does not imply that the recoding algorithm terminates. This has been observed for NAF-like expansions of rational integers to the base of 2 by Muir and Stinson [14]. If we take $w = 1$ and the digit set $\{0, 1-\tau\}$ (here the corresponding reduced residue set modulo $\tau = \tau^1$ comprises the single element $1-\tau$) we see that the element 1 has an expansion $(1-\tau) + (1-\tau)\tau + (1-\tau)\tau^2 + (1-\tau)\tau^3 + \cdots$. Algorithm 1 does not terminate in this case.

2.1 Algorithmic Characterization

As we already mentioned above, one aim of this paper is to investigate which digit sets $\mathcal{D}$ are in fact w-NADS. For concrete $\mathcal{D}$ and w, this question can be decided algorithmically:

Theorem 1. *Let $\mathcal{D}$ be a finite subset of $\mathbb{Z}[\tau]$ containing 0 and $w \geq 1$ be an integer. Let*

$$M := \left\lfloor \frac{\max\{N(d) : d \in \mathcal{D}\}}{\left(2^{w/2}-1\right)^2} \right\rfloor ,$$

where $N(z)$ denotes the norm of z, i.e., $N(a+b\tau) = (a+b\tau)(a+b\bar{\tau}) = a^2 + \mu ab + 2b^2$ for $a, b \in \mathbb{Z}$.

Consider the directed graph $G = (V, A)$ defined by its set of vertices

$$V := \{z \in \mathbb{Z}[\tau] : N(z) \leq M\}$$

and set of arcs

$$\begin{aligned} A :=& \{(y,z) \in V^2 : \text{ There is a nonzero } d \in \mathcal{D} \text{ such that } z = \tau^w y + d\} \\ & \cup \{(y,z) \in V^2 : z = \tau y\} \ . \end{aligned}$$

Then $\mathcal{D}$ is a w-NADS if and only if the following two conditions are both satisfied.

1. *The set $\mathcal{D}$ contains a reduced residue system modulo τ^w.*
2. *In $G = (V, A)$, each vertex $z \in V$ is reachable from 0.*

If $\mathcal{D}$ is a w-NADS and $\mathcal{D} \setminus \{0\}$ is a reduced residue system modulo τ^w, then each $z \in \mathbb{Z}[\tau]$ has a unique $\mathcal{D}$-w-NAF.

This result is proved in Appendix A. We now make a few simple remarks and discuss two well-known examples.

Remark 1. A number $a + \tau b \in \mathbb{Z}[\tau]$ is relatively prime to τ if and only if a is odd. This follows from the fact that τ is a prime element in $\mathbb{Z}[\tau]$ and that τ divides a rational integer if and only if this rational integer is even.

Example 2. Let $w = 1$ and $\mathcal{D} = \{0, 1\}$. By Remark 1, there is only one residue class prime to τ. In this case $M = 5$, so $V = \{0, \pm 1, \pm 2, \pm\tau, \pm 1 \pm \tau, \pm(-\mu\tau + 2)\}$. The corresponding directed graph in the case $\mu = -1$ is shown in Figure 2. The case $\mu = 1$ is similar. We see that all 13 states are reachable from 0. Thus, $\{0, 1\}$ is a 1-NADS. This is equivalent to saying that τ is the base of a canonical number system in $\mathbb{Z}[\tau]$ in the sense of [9], and is a particular case of results from [8].

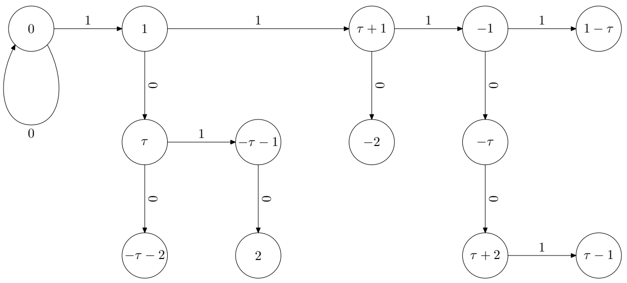

Fig. 1. Directed Graph G for $\mu = -1$, $w = 1$, $\mathcal{D} = \{0, 1\}$.

Remark 2. Example 2 immediately shows that there are exactly 2^w residue classes modulo τ^w; a complete residue system is given by $\sum_{j=0}^{w-1} \varepsilon_j \tau^j$ with $\varepsilon_j \in \{0, 1\}$ for $0 \leq j < w$. There are 2^{w-1} residue classes relatively prime to τ^w, a reduced residue system is given by $1 + \sum_{j=1}^{w-1} \varepsilon_j \tau^j$ with $\varepsilon_j \in \{0, 1\}$ for $1 \leq j < w$.

Example 3. Let $w = 2$ and $\mathcal{D} = \{0, \pm 1\}$. Using Remark 2, it is easily seen that $\{\pm 1\}$ is a reduced residue system modulo τ^2. In this case, $M = 1$, the graph G consists of the three states $V = \{0, \pm 1\}$ only, and those are obviously reachable from 0. Thus $\{0, \pm 1\}$ is a 2-NADS. This has been proved by Solinas [20, 21].

Example 4. One might consider the digit set $\mathcal{D} = \{0\} \cup \{\pm 1, \pm 3, \ldots, \pm(2^{w-1} - 1)\}$. The odd digits form a reduced residue system modulo τ^w, since τ^w divides a rational integer if and only if 2^w divides this rational integer (note that τ and $\bar{\tau}$ are coprime primes in $\mathbb{Z}[\tau]$). However, this digit set is not a w-NADS for all w. For instance, for $w = 6$, the number $1 - \mu\tau$ has no $\mathcal{D}$-6-NAF. However, using Theorem 1, it turns out that for $w \in \{2, 3, 4, 5, 7, 8, 9, 10\}$, this set $\mathcal{D}$ is a w-NADS.

2.2 Representatives of Minimal Norm

Theorem 2. *Let τ, $w \geq 2$ be as above, and $\mathcal{D}$ a digit set consisting of 0 together with one element of minimal norm from each odd residue class modulo τ^w.*

The digit set $\mathcal{D}$ is uniquely determined. In other words, in each odd residue class modulo τ^w there exists a unique element of minimal norm.

Proof. Let α, β be distinct elements of minimal norm in the same odd residue class modulo τ^w. Then, $\beta = \alpha + \gamma\tau^w$ with $\gamma \in \mathbb{Z}[\tau] \setminus \{0\}$. By [21, Corollary 59 and Equation 64] we have that $N(\alpha), N(\beta) \leq \frac{4}{7}N(\tau^w)$, hence $\sqrt{N(\gamma)N(\tau^w)} = \sqrt{N(\alpha - \beta)} \leq \sqrt{N(\alpha)} + \sqrt{N(\beta)} \leq \frac{4}{\sqrt{7}}\sqrt{N(\tau^w)}$. This implies $N(\gamma) \leq \frac{16}{7}$. Being $\gamma \neq 0$, it can only be $N(\gamma) = 1$ or 2.

Now we make use of the fact that τ is prime and does not divide α nor $\bar{\tau}$. Writing down the relation $N(\alpha + \gamma\tau^w) = N(\alpha)$ explicitly we obtain $\alpha\bar{\gamma}\bar{\tau}^w + \bar{\alpha}\gamma\tau^w + \gamma\bar{\gamma}\tau^w\bar{\tau}^w = 0$. This implies that τ^w divides $\alpha\bar{\gamma}\bar{\tau}^w$ and thus $\bar{\gamma}$. Therefore $2^w = N(\tau^w)$ divides $N(\bar{\gamma}) = N(\gamma)$ which we know to be either 1 or 2, implying in turn $w \leq 1$. This is a contradiction.

In [4] it has been shown that the τ-NAF has minimal weight among all the τ-adic expansions with digit set $\{0, \pm 1\}$. In fact, since the digit set $\mathcal{D} = \{0, \pm 1 \pm \bar{\tau}\}$ is also Solinas' set for $w = 3$, in the same paper it is shown that a $\mathcal{D}$-w-NAF with this digit set is actually always an optimal $\mathcal{D}$-expansion. In the binary case (where τ is replaced by 2), it turns out that the analogous result is true for all positive w [1, 15]. So one might conjecture that the same is also true for our choice of τ. But, the following example shows that this is in fact not the case:

Example 5. Consider $\mu = -1$, $w = 4$, and the set $\mathcal{D}$ of minimal norm representatives modulo τ^w. We have $\mathcal{D} = \{0, \pm 1, \pm 1 \pm \tau, \pm(3 + \tau)\}$ and note that

$$\mathsf{value}(1, 0, 0, 0, -1-\tau, 0, 0, 0, 1-\tau) = -9 = \mathsf{value}(-3-\tau, 0, 0, -1) \enspace .$$

The first expansion is the $\mathcal{D}$-w-NAF and has Hamming weight 3, whereas the second expansion does not satisfy the $\mathcal{D}$-w-NAF-condition, has Hamming weight 2 and is even shorter.

Even worse, we exhibit chaotic behaviour in the following sense: for every positive integer k, it is possible to exhibit a pair of numbers which are congruent modulo τ^k, but whose optimal $\mathcal{D}$-expansions must differ even at the least significant position. Thus it is impossible to compute an optimal $\mathcal{D}$-expansion of z by a deterministic transducer automaton or an online algorithm.

Proposition 1. *Let $w = 4$, and $\mathcal{D} = \{0, \pm 1, \pm 1 \pm \tau, \pm(3 - \mu\tau)\}$ (all signs are independent) be the set of minimal norm representatives modulo τ^w. For every nonnegative integer ℓ, we define*

$$\begin{aligned} z_\ell &:= \mathsf{value}(\ 0, 0, 0, 0, \mu - \tau, (0, 0, 0, -3\mu + \tau)^{(\ell)}, 0, 0, 0, 0, 1 - \mu\tau, 0, 0, 0, -1) \enspace , \\ z'_\ell &:= \mathsf{value}(-\mu, 0, 0, 0, \mu - \tau, (0, 0, 0, -3\mu + \tau)^{(\ell)}, 0, 0, 0, 0, 1 - \mu\tau, 0, 0, 0, -1) \enspace , \end{aligned} \tag{4}$$

where $(0, 0, 0, -3\mu + \tau)^{(\ell)}$ means that this four-digit block is repeated ℓ times. Then $z_\ell \equiv z'_\ell \pmod{\tau^{4\ell+13}}$. All $\mathcal{D}$-optimal expansions of z_ℓ are given by

$$\big((0, 0, 0, 3 - \mu\tau)^{(\ell_2)}, 0, 0, \mu - \tau, (0, 0, 0, -3\mu + \tau)^{(\ell_1)}, 0, 0, 0, 0, 1 - \mu\tau, 0, 0, 0, -1\big) \enspace ,$$

where ℓ_1 and ℓ_2 are nonnegative integers summing up to ℓ. There is only one $\mathcal{D}$-optimal expansion of z'_ℓ, it is given by

$$\big((0,0,0,-3+\mu\tau)^{(\ell+1)},0,0,0,0,-3\mu+\tau,0,0,1+\mu\tau\big) \ .$$

Note that the $\mathcal{D}$-optimal expansion of z'_ℓ has Hamming weight $\ell+3$, whereas the $\mathcal{D}$-w-NAF of z'_ℓ given in (4) has Hamming weight $\ell+4$. The proof of this Proposition is based on the search of shortest paths in an auxiliary automaton.

2.3 Syntactic Sufficient Conditions

The aim of this section is to prove sufficient conditions for families of sets $\mathcal{D}$ to be a w-NADS at the level of digits of the τ-NAF. In contrast to Theorem 1, where a decision can be made for any concrete set $\mathcal{D}$, we will now focus on families of such sets. Blake, Murty, and Xu [6] gave such sufficient conditions based on the norm of the numbers involved.

Proposition 2. *Let $w \geq 1$ and ε, ε' two τ-NAFs. Then* $\mathsf{value}(\varepsilon) \equiv \mathsf{value}(\varepsilon') \pmod{\tau^w}$ *if and only if*

$$\varepsilon_j = \varepsilon'_j \textit{ for } 0 \leq j \leq w-2 \textit{ and } |\varepsilon_{w-1}| = |\varepsilon'_{w-1}| \ . \tag{5}$$

The proof is contained in Appendix B.

Definition 3. *Let w be a positive integer and $\mathcal{D}$ be a subset of*

$$\{\,0\,\} \cup \{\, \mathsf{value}(\varepsilon) : \varepsilon \textit{ is a } \tau\textit{-NAF of length at most } w \textit{ with } \varepsilon_0 \neq 0 \,\}$$

consisting of 0 *and a reduced residue system modulo τ^w. Then $\mathcal{D}$ is called a* set of short τ-NAF representatives *for τ^w.*

By Proposition 2, an example for a set of short τ-NAF representatives is

$$\begin{aligned}\mathcal{D} = \{\,0\,\} \cup \big\{\, \mathsf{value}(\varepsilon) : \varepsilon \text{ is a } \tau\text{-NAF of length at most } w \\ \text{with } \varepsilon_0 \neq 0 \text{ and } \varepsilon_{w-1} \in \{0, \varepsilon_0\} \,\big\} \ .\end{aligned} \tag{6}$$

All other sets of short τ-NAF representatives are obtained by changing the signs of ε_{w-1} without changing ε_0 in some of the ε. It is easy to check that the cardinality of $\mathcal{D}$ is indeed $1 + 2^{w-1}$.

The main result of this section is the following theorem, which states that in almost all cases, a set of short τ-NAF representatives is a w-NADS:

Theorem 3. *Let w be a positive integer and $\mathcal{D}$ a set of short τ-NAF representatives. Then $\mathcal{D}$ is a w-NADS if and only if it is not listed in Table 1. In particular, if $w \geq 4$, then $\mathcal{D}$ is a w-NADS.*

The proof of this theorem is contained in Appendix C.

μ	$\mathcal{D}$	Remark
-1	$\{1, -1, -\tau^2 + 1, -\tau^2 - 1\}$	$(-\tau - 1)\left(1 - \tau^3\right) = -\tau^2 + 1$
-1	$\{1, -1, -\tau^2 + 1, \tau^2 - 1\}$	$(-\tau - 1)\left(1 - \tau^3\right) = -\tau^2 + 1$
-1	$\{1, -1, \tau^2 + 1, \tau^2 - 1\}$	$(\tau + 1)\left(1 - \tau^3\right) = \tau^2 - 1$
1	$\{1, -1, -\tau^2 + 1, \tau^2 - 1\}$	$(-\tau + 1)\left(1 - \tau^6\right) = \left(-\tau^2 + 1\right)\tau^3 + \tau^2 - 1$

Table 1. List of sets of short τ-NAF representatives which are not a w-NADS. The remark column contains an example of an element which cannot be represented.

Theorem 4. *Let $w \geq 2$ be a positive integer, $\mathcal{D}$ a set of short τ-NAF representatives, and ε a $\mathcal{D}$-w-NAF of some $z \in \mathbb{Z}[\tau]$.*

Then the length of ε can be bounded by

$$2\log_2 |z| - w - 0.18829 < \mathsf{length}(\varepsilon) < 2\log_2 |z| + 7.08685 \ , \quad \textit{if } w \geq 4 \ , \tag{7}$$

$$2\log_2 |z| - 2.61267 < \mathsf{length}(\varepsilon) < 2\log_2 |z| + 5.01498 \ , \quad \textit{if } w = 3 \ , \tag{8}$$

$$2\log_2 |z| - 0.54627 < \mathsf{length}(\varepsilon) < 2\log_2 |z| + 3.51559 \ , \quad \textit{if } w = 2 \ . \tag{9}$$

Note that (9) is Solinas' [21] Equation (53). The proof of this Theorem uses (among other things) methods from [21]. It can be found in Appendix D.

2.4 Point Halving

For any given point P, point halving [10, 18, 19] consists in computing a point Q such that $2Q = P$. This inverse operation to point doubling applies to all elliptic curves over binary fields. Its evaluation is faster than that of a doubling and a halve-and-add scalar multiplication algorithm based on halving instead of doubling can be devised. This method is not useful for Koblitz curves because halving is slower than a Frobenius operation.

In [2] it is proposed to insert a halving in the "τ-and-add" method to speed up Koblitz curve scalar multiplication. This approach brings a non-negligible speedup and was further refined in [4], where the insertion of a halving was implicitly interpreted as a digit set extension. This interpretation is the following: Inserting a halving in the scalar multiplication is equivalent to adding $\pm\bar{\tau}$ to the digit set $\{0, \pm 1\}$. In fact, $\mathcal{D} = \{0, \pm 1, \pm\bar{\tau}\}$ is a valid 3-NADS. By Theorem 3, this is the only 3-NADS of short τ-NAF representatives for $w = 3$ and $\mu = -1$. In the following theorem we shall show that the set of cardinality $1 + 2^{w-1}$ defined by $\mathcal{D} := \{0\} \cup \{\pm\bar{\tau}^k : 0 \leq k < 2^{w-2}\}$ for $w \geq 2$ is a reduced residue system modulo τ^w. Later we shall discuss when it is a w-NADS, and we present a "precomputationless" width-w scalar multiplication algorithm generalising that of [2] that uses the above set.

Theorem 5. *Let $w \geq 2$. Then $\mathcal{D}' := \{\pm\bar{\tau}^k : 0 \leq k < 2^{w-2}\}$ is a reduced residue system modulo τ^w.*

The proof is found in Appendix E.

Theorem 6. *Let $w \in \{2, 3, 4, 5, 6\}$ and $\mathcal{D} := \{0\} \cup \{\pm\bar{\tau}^k : 0 \leq k < 2^{w-2}\}$. Then $\mathcal{D}$ is a w-NADS.*

Proof. For every pair (w, μ) the conditions of Theorem 1 have been verified by heavy symbolic computations.

We conjecture that Theorem 6 holds also for higher values of w, but verifying this using Theorem 1 seems to be too expensive.

2.5 Comparing the Digit Sets

So far, three digit sets have been studied: the minimal norm representatives, short NAF representatives, and powers of $\bar{\tau}$. It is a natural question to ask what are the relations between these sets.

As Table 2 shows, the minimal norm representatives are exactly the powers of $\bar{\tau}$ for $w \leq 4$. For the same range of w, all digits of these digit sets have a τ-NAF of length at most w, which implies that they are also digit sets of short NAF representatives.

If symmetry is required, i.e., if d is a digit, then $-d$ must also be a digit, there is only one digit set of short NAF representatives for $w \leq 3$ by Theorem 3, which therefore coincides with the digit set of minimal norm representatives and powers of $\bar{\tau}$. For $w = 4$, however, there is also a symmetric digit set of short NAF representatives distinct from the digit set of minimal norm representatives and powers of $\bar{\tau}$.

For $w \geq 5$, the three concepts are different: the lengths of the τ-NAFs of the powers of $\bar{\tau}$ grow exponentially in w, and the lengths of the some minimal norm representatives exceed w slightly (at most by 2).

w	MNR=P$\bar{\tau}$	Max τ-NAF length MNR	Max τ-NAF length P$\bar{\tau}$
2	True	1	1
3	True	3	3
4	True	4	4
5	False	6	8
6	False	8	17

Table 2. Comparison between minimal norm representatives, short NAF representatives, and powers of $\bar{\tau}$ digit sets. "MNR" stands for the minimal norm representatives digit set, whereas "P$\bar{\tau}$" stands for the powers of $\bar{\tau}$. The last two column show the maximum length of the τ-NAFs of the digits.

3 Applications

All digit sets seen so far can be used in a τ-and-add scalar multiplication, where we first precompute $d{\cdot}P$ for all $d \in \mathcal{D} \backslash \{0\}$ and then we evaluate the scheme $\sum z_i \tau^i(P)$; in fact, only a half of the precomputations usually suffice since in all cases that we explicitly described the non-zero elements of the digit set come in pairs of elements of opposite sign.

The digit set from § 2.3 simplifies the precomputation phase. The digit set from § 2.4 allows us to perform precomputations very quickly or to get rid of them completely. In the next two subsections we shall consider these facts in detail. In § 3.3 we explain how to use digit sets which are not w-NADS when they contain a subset that is a k-NADS for a smaller k.

3.1 Using the Short-NAF Digit Set

Let us consider here the digit set $\mathcal{D}$ defined in (6). With respect to Solinas' set it has the advantage of being syntactically defined. If a computer has to work with different curves, different scalar sizes and thus with different optimal choices for the window size, the representatives in Solinas' set must be recomputed – or they must be retrieved from a set of tables. In some cases, the time to compute representatives of minimal norm may have to be subsumed in the total scalar multiplication time. This is not the case with our set. This flexibility is also particularly important for computer algebra systems.

The scalar needs first to be recoded as a τ-NAF, and the elements of $\mathcal{D}$ are associated to NAFs of length at most w with non-vanishing least significant digit, and thus to certain *odd integers* in the interval $[-a_w, a_w]$ where $a_w = \frac{2^{w+1}-2(-1)^w}{3} - 1$ (the a_w form a generalized Jacobsthal sequence given by the recursion $a_w = a_{w-1} + 2a_{w-2} + 2$). These integers can be used to index the elements in the precomputation table. We need only to precompute the multiples of the base point by "positive" short NAFs (i.e. with most significant digit equal to 1) – and the integers are the odd integers in the interval $[0, a_{w-1}]$ together with the integers congruent to 1 modulo 4 in $[a_{w-1}+2, a_w]$. The indexes in the table are then obtained by easy compression. The precomputed elements for the scalar multiplication loop can thus be retrieved upon direct reading the τ-NAF, of which we need only to compute the least w significant places. If the least and the w-th least significant digits of this segment of the τ-NAF are both non-zero and have different signs, a carry is generated. Therefore the computation of the simple τ-NAF should be interleaved with its parsing for short NAFs. This can be done in a simple way by straightforward modifications to the algorithms for the τ-NAF in [20, 21].

3.2 τ-adic Scalar Multiplication with Repeated Halvings

Let $w \geq 2$ be an integer and $\mathcal{D}$ the digit set defined in § 2.4. Let P be a point on an elliptic curve and $Q_j := \tau^j(2^{-j}P)$ for $0 \leq j < 2^{w-2}$ and $R := Q_{2^{w-2}-1}$. To compute zP, we have to compute yR for $y := \bar{\tau}^{2^{w-2}-1}z$. Computing a $\mathcal{D}$-w-NAF of y, this can be done by using the points Q_j, $0 \leq j < 2^{w-2}$ as precomputations.

Now, a point halving on an elliptic curve is not only much faster than a point doubling – with affine coordinates a doubling and an addition have similar timings, and with other coordinate systems an addition is much slower than the doubling. But with more traditional digit sets the precomputations always involve at least one addition per digit set element. Therefore the approach just described with the points $\mathbb{Q}_j$ and halvings is already faster than traditional approaches.

But we can do even better, especially if normal bases are used to represent the field $\mathbb{F}_{2^n}$. Algorithm 2 computes $z \cdot P$ using an expansion $y = \sum_{i=0}^{\ell} y_i \tau^i$ of the integer $y := \bar{\tau}^{2^{w-2}-1} z$ where the digits y_i belong to the digit set introduced in Theorem 5, i.e. $\mathcal{D} := \{0\} \cup \{\pm\bar{\tau}^k : 0 \le k < 2^{w-2}\}$.

Algorithm 2. τ-adic Scalar Multiplication with Repeated Halvings

INPUT: A Koblitz curve E_a with corresponding parameter $\mu = (-1)^{1-a}$, a point P of odd order on E_a and an expansion $y = \sum_{i=0}^{\ell} y_i \tau^i$ where $y_i \in \mathcal{D} := \{\pm\bar{\tau}^k : 0 \le k < 2^{w-2}\}$ of the integer $y := \bar{\tau}^{2^{w-2}-1} z$. Write $y_i = \varepsilon_i \bar{\tau}^{k_i}$ with $\varepsilon \in \{0, \pm 1\}$.

OUTPUT: $z \cdot P$

```
1.  $\ell_k \leftarrow \max\left(\{-1\} \cup \{i : z_i = \pm\bar{\tau}^k \text{ for some } k\}\right)$
2.  $X \leftarrow 0$
3.  for $k = 0$ to $2^{w-2} - 1$ do
4.      if $k > 0$ then $X \leftarrow \tau^{n-\ell_k} X$, $X \leftarrow \frac{1}{2} X$
5.      for $i = \ell_k$ to $0$ do
6.          $X \leftarrow \tau X$
7.          if $y_i = \pm\bar{\tau}^k$ then $X \leftarrow X + \varepsilon_i P$
8.  return $(X)$
```

To explain how it works we introduce some notation. Write $y_i = \varepsilon_i \bar{\tau}^{k_i}$ with $\varepsilon_i \in \{0, \pm 1\}$. We also define

$$y^{(k)} = \sum_{i\,:\,0 \le i \le \ell,\; y_i = \pm\bar{\tau}^k} \varepsilon_i \tau^i \quad .$$

Now $y = \sum_{k=0}^{2^{w-2}-1} y^{(k)} \bar{\tau}^k$ and therefore

$$\begin{aligned} z \cdot P = \bar{\tau}^{-(2^{w-2}-1)} y \cdot P &= \left(\sum_{m=0}^{2^{w-2}-1} y^{(m)} \bar{\tau}^m \right) \bar{\tau}^{-(2^{w-2}-1)} \cdot P \\ &= \sum_{m=0}^{2^{w-2}-1} y^{(m)} \bar{\tau}^{m-(2^{w-2}-1)} \cdot P = \sum_{m=0}^{2^{w-2}-1} \left(\frac{\tau}{2}\right)^{2^{w-2}-1-m} (y^{(m)}) \cdot P \end{aligned}$$

and the last expression is evaluated by a Horner scheme in $\frac{\tau}{2}$, i.e. by repeated applications of τ and a point halving, interleaved with additions of $y^{(0)} \cdot P$, $y^{(1)} \cdot P$, etc. The elements $y^{(k)} \cdot P$ are computed by a τ-and-add loop as usual. To save a memory register, instead of computing $y^{(k)} \cdot P$ and then adding it to a partial evaluation of the Horner scheme, we apply τ to the negative of the length of $y^{(k)}$ (which is $1 + \ell_k$) to the intermediate result X and perform the τ-and-add loop to evaluate $y^{(k)} \cdot P$ starting with this X instead of a "clean" zero. In Step 4 there is an optimization already present in [2]: n is added to the exponent (since $n \approx \ell_k$ and τ^n acts like the identity on the curve) and the operation is also partially fused to the subsequent $\frac{\tau}{2}$. At the end of the internal loop the relation $X = \sum_{m=0}^{k} \left(\frac{\tau}{2}\right)^{k-m} y^{(m)} P$ holds, thus proving the correctness.

Apart from the input, we need only storage for the additional variable X and the recoding of the scalar. The multiplication of z by $\bar{\tau}^{2^{w-2}-1}$ is an easy operation, and the negative powers of τ can be easily eliminated by multiplying by a suitable power of τ^n, which operates trivially on the points of the curve. Reduction of this scalar by $(\tau^n - 1)/(\tau - 1)$ following Solinas [20, 21] is also necessary.

An issue with Algorithm 2 is that the number of Frobenius operations may increase exponentially with w, since the internal loop is repeated up to 2^{w-2} times. This is not a problem if a normal basis is used to represent the field, but may induce a performance penalty with a polynomial basis. A similar problem was faced by Okeya, Takagi and Vuillaume in [16], and they solved it adapting an idea by Park, Sim and Lee [17]. The technique consists in keeping a copy R of the point P in normal basis representation. Instead of computing $y^{(k)} \cdot P$ by a Horner scheme in τ, the summands $\varepsilon_i \tau^i \cdot P$ are just added together. The power of the Frobenius is applied to R *before* converting the result back to a polynomial basis representation and adding it to an accumulation variable. According to [7] converting a field element between the two bases takes about the same time as one polynomial basis multiplication, and the conversion routines require each a matrix that occupies $O(n^2)$ bits of memory.

Algorithm 3 is our realisation of this approach. It is particularly well suited for context where a polynomial basis is used for a field where the cost of an inversion is not prohibitive. The routines $\mathrm{normal\_basis}$ and $\mathrm{polynomial\_basis}$ perform the conversion of coordinates of the points between polynomial and normal bases.

Algorithm 3. Low-memory τ-adic Scalar Multiplication on Koblitz Curves with Repeated Halvings, for Fast Inversion

INPUT: $P \in E(\mathbb{F}_{2^n})$, scalar z
OUTPUT: $z \cdot P$

```
1. $y \leftarrow \bar{\tau}^{2^{w-2}+m-1} z$
   Write $y = \sum_{i=0}^{\ell} y_i \tau^i$ where $y_i \in \mathcal{D} := \{0\} \cup \pm\{\bar{\tau}^k : 0 \le k < 2^{w-2}\}$
   Write $y_i = \varepsilon_i \bar{\tau}^{k_i}$ with $\varepsilon_i \in \{0, \pm 1\}$
2. $R \leftarrow \mathrm{normal\_basis}(P)$
3. $Q \leftarrow 0$
4. for $k = 0$ to $2^{w-2} - 1$
5.     if $k > 0$ then $Q \leftarrow \tau Q,\ Q \leftarrow {}^1\!/\!_2 Q$
6.     for $i = 0$ to $\ell$
7.         if $y_i = \pm\bar{\tau}^k$ then $Q \leftarrow Q + \varepsilon_i \cdot \mathrm{polynomial\_basis}(\tau^i R)$
8. return $Q$
```

Algorithm 4 is designed for fields with a slow inversion (such a large fields). It uses inversion-free coordinate systems, and for this purpose, since there is no halving formula known in such coordinates, a doubling is used. Not only this is not a problem, since using Projective or López-Dahab coordinates (see [3, § 15.1]) a doubling followed by an application of τ^{-1} (which amount to three square root extractions). is about twice as fast as a mixed-coordinate addition preceded by a basis conversion – therefore the situation is advantageous as the previous one. Furthermore, this dispenses us with the need of using a modified scalar y.

Algorithm 4. Low-memory τ-adic Scalar Multiplication on Koblitz Curves with Repeated Doublings, for Slow Inversion

```
INPUT: $P \in E(\mathbb{F}_{2^n})$, scalar $z$
OUTPUT: $z \cdot P$

Write $z = \sum_{i=0}^{\ell} z_i \tau^i$ where $z_i \in \mathcal{D} := \{0\} \cup \pm\{\bar{\tau}^k : 0 \le k < 2^{w-2}\}$
Write $z_i = \varepsilon_i \bar{\tau}^{k_i}$ with $\varepsilon_i \in \{0, \pm 1\}$
$R \leftarrow \mathrm{normal\_basis}(P)$                                 [Keep in affine coordinates]
$Q \leftarrow 0$                                                      [$Q$ is in López-Dahab coodinates]
for $k = 2^{w-2} - 1$ to 0
    if $k > 0$ then $Q \leftarrow \tau^{-1} Q,\ Q \leftarrow 2 \cdot Q$                [$\tau^{-1}$ is three square roots]
    for $i = 0$ to $\ell$
        if $z_i = \pm\bar{\tau}^k$ then $Q \leftarrow Q + \varepsilon_i \cdot \mathrm{polynomial\_basis}(\tau^i R)$ [Mixed coordinates]
return $Q$                                                            [Convert to affine coordinates]
```

The digit set $\mathcal{D}$ introduced in Theorem 5 may not be a w-NADS for all w. The technique presented in the next Subsection shows how to save the situation.

3.3 Stepping Down Window Size

Suppose we have a digit set $\mathcal{D}$, and a recoding like Algorithm 1 parametrized by an integer w, and something is causing the recoding to stop or to enter a loop – our set is not a w-NADS. For other inputs, and for the digits generated so far, the algorithm delivers a nice, low density. How can we save it? One possible answer is to lower the value of the parameter w and settle for a smaller digit set which is a subset of $\mathcal{D}$, which we know is a w-NADS, for the rest of the computation. We call this operation *stepping down*. The resulting recoding may have a slightly higher weight, but the algorithm is guaranteed to terminate.

Non termination can happen in Algorithm 1 when the set $\mathcal{D}$ is not a w-NADS and the norm of the variable u gets too small in comparison to the chosen digit, so that it may be that $|u| \le \left|\frac{u-z_j}{\tau^w}\right| \le \frac{|u|+|z_j|}{2^{w/2}}$, i.e. $|z_j| \ge |u|(2^{w/2} - 1)$. This is usually caused by the appearance of “large” digits towards the end of the main loop of the recoding algorithm, and stepping down must then hold until the end of the algorithm. Solinas is able to prove termination of his τ-adic w-NAF because his digits have norm bounded by $\frac{4}{7}2^w$ and are minimal representants in their classes. A large norm or non-minimality of digits are necessary but not sufficient conditions for non-termination. In fact digit sets with digits of norm larger than 2^w can be w-NADS. For example, the elements in the digit sets in Example 4 have larger norm and are not all minimal representants, but for $w \in \{2, 3, 4, 5, 7, 8, 9, 10\}$ they form w-NADS. In all cases we tested, the digit set from § 2.4 is a w-NADS.

Algorithm 5. Windowed Integer Recoding With Termination Guarantee

INPUT: An element z from $\mathbb{Z}[\tau]$, a natural number $w \geq 1$ and a set of reduced residue systems $\mathcal{D}'_k \subset \mathcal{D}'_{k+1} \subset \ldots \mathcal{D}'_w$ modulo τ^k, τ^{k+1}, . . ., τ^w respectively, $(1 \leq k < w)$ where $\mathcal{D}'_k \cup \{0\}$ is a k-NADS.
OUTPUT: A representation $z = \sum_{j=0}^{\ell-1} z_j \tau^j$ of length ℓ.

```
1.  $j \leftarrow 0, u \leftarrow z, v \leftarrow w$
2.  while $u \neq 0$ do
3.      if $\tau \mid u$ then
4.          $z_j \leftarrow 0$
5.      else
6.          Let $z_j \in \mathcal{D}'_v$ s.t. $z_j \equiv u \pmod{\tau^v}$
7.          if $(\,|z_j| \geq |u|(2^{v/2} - 1)$ AND $v > k\,)$ then decrease $v$ and retry:
8.              $v \leftarrow v - 1$, go to Step 6
9.      $u \leftarrow u - z_j,\ u \leftarrow u/\tau,\ j \leftarrow j + 1$
10. $\ell \leftarrow j$
11. return $(\{z_j\}_{j=0}^{\ell-1}, \ell)$
```

Remark 3. There are variants of this algorithm. Instead of checking norms in Step 7 – which can be expensive even if done smartly – we can just ignore the test and check *later* if the algorithm has entered in a loop. This can be done by checking if j has become larger than $\log_2 N(z)$ plus some small constant, and if this is the case, we decrease v to k and continue with the guarantee that the recoding will work. In fact, this is the variant we chose to implement, as a tight bound for $\log_2 N(z)$ is always known in the applications, and most random scalars have nearly maximal norm, hence almost no additional computational costs are involved.

Remark 4. Note that in the digit set from Example 4, the syntactically defined set of § 2.3 and the set of Theorem 5 all have the property that each set is contained in the sets with larger w – hence this enhanced recoding algorithm can be applied.

In our experiments, the recodings done with the different digit sets have similar length and the average density is, as expected, $1/(w + 1)$. Stepping down makes the weight higher, but only in relatively few cases. The highest increase in weight is about $w/2$ and there are no changes in the average asymptotic density. Therefore the new digit sets bring their advantages with *de facto* no performance penalty.

3.4 A Performance Remark

Algorithms 2, 3 and 4 all perform a scalar multiplication by $2^{w-2} - 1$ "faster" operation blocks and on average $n/(w+1)$ "slower" operation blocks. In the first algorithm (with normal bases) these two block types are given by a halving and an addition. In the second, resp. third algorithm these two block types are given by a Frobenius operation and a halving (resp. by an inverse Frobenius and a doubling), and by a basis conversion followed by an addition. In all cases we have remarked that the first block costs α times the second, where $\alpha \leq 1/2$.

To achieve optimal performance we need to find the minimum of

$$f(n;\alpha) = \alpha(2^{w-2} - 1) + \frac{n}{w+1} \quad .$$

It is a well known fact that the minimum is attained for

$$\widehat{w} = \frac{2\,\mathrm{W}\left(\sqrt{2\ln(2)\,n/\alpha}\right)}{\ln 2} - 1$$

where W is the main branch of Lambert's omega function. This $\widehat{w}$ can be well approximated as

$$\log_2(n/\alpha) - 2\log_2(\log_2(n/\alpha)) + c$$

where, asymptotically, $c = 3 - \ln(\ln 2)/\ln 2 - \ln 2/2$ and for $n/\alpha < 1000$ one can take a slightly larger value, for example $c = \frac{10}{3}$ to get a good approximation. The optimal value of w for the applications is thus the closest integer to $\widehat{w}$.

The important aspect here is the following: Taking into account the fact that not only $\alpha = O(1)$, but that in practice α is bounded also from below, and setting $w = \log_2(n/\alpha) - 2\log_2(\log_2(n/\alpha)) + O(1)$ in $f(n;\alpha)$, we easily obtain that $f(n;\alpha) = O(n/\log n)$. In other words, *Algorithms 2, 3 and 4 are instances of sublinear scalar multiplication algorithms on Koblitz Curves with constant memory consumption.* The method in [5] is interesting theoretically but its practical relevance stil has to be assessed - the authors warn that the involved constants may be quite large.

Previous algorithms, such as traditional windowed methods with precomputations, have of course similar complexity but require storage for $2^{w-2} - 1$ points [20, 21]. The method of [16] has a small memory footprint but works for $w = 5$ only.

Furthermore, our algorithms perform better than the aforementioned techniques using precomputations, for the same values of w. In fact, performing the required precomputations with Solinas' digit set requires one addition and possibly some Frobenius operations per precomputed point (there are $2^{w-2} - 1$ of them). In any case, we replace these operations with much cheaper ones, whereas in Algorithms 3 and 4 the increase in cost associated to the addition in the main loop is relatively small and the increase in recoding weight is marginal. In fact, we can use also larger window sizes and better balance performance. Hence it easy to verify that our methods run faster. An exact performance evaluation lies outside the scope of this paper.

4 Conclusions

The paper at hand presents several new results about τ-adic recodings.

We characterise digit sets allowing a w-NAF to be computed for all inputs, and we study several such sets with interesting properties for Koblitz curves.

Solinas' digit set, characterised by the property that the elements have minimal norm in their residue classes, is also considered. We present a surprising example showing that the non adjacency property does not imply minimality of weight, and enunciate a result implying that optimal expansions cannot be computed by a deterministic finite automaton.

In § 2.3 we introduce a new digit set characterised by syntactic properties. Its usage is described in § 3.1.

The digit set introduced in § 2.4 together with Algorithms 2, 3 and 4 from § 3.2 permit to perform a "windowed" τ-adic scalar multiplication without requiring storage for precomputed points. The result is potentially ground-breaking for implementation on restricted devices. In fact, our methods easily perform better than the previous methods that made use of precomputations. Our method works for all values for the window size. A thorough performance assessment will be part of future work.

References

1. R. Avanzi. *A Note on the Signed Sliding Window Integer Recoding and its Left-to-Right Analogue.* In Proceedings of SAC 2004. LNCS 3357, 130–143. Springer, 2005.
2. R. 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. R. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange, K. Nguyen, and F. Vercauteren. *The Handbook of Elliptic and Hyperelliptic Curve Cryptography.* CRC Press, 2005.
4. R. Avanzi, C. Heuberger, and H. Prodinger. *Minimality of the Hamming Weight of the τ-NAF for Koblitz Curves and Improved Combination with Point Halving.* Proceedings of SAC 2005. LNCS 3897, pages 332-344. Springer, 2006.
5. R. Avanzi and F. Sica. Scalar Multiplication on Koblitz Curves Using Double Bases, 2006. Cryptology ePrint Archive, Report 2006/067, 2006. Available at `http://eprint.iacr.org/`.
6. I.F. Blake, V. Kumar Murty, G. Xu. *A note on window τ-NAF algorithm.* Information Processing Letters **95** (2005) 496–502.
7. J.-S. Coron, D. M'Raïhi, and C. Tymen. Fast generation of pairs $(k, [k]p)$ for koblitz elliptic curves. In *Proceedings of SAC 2001*, volume 2259 of *Lecture Notes in Computer Science*, pages 151–164. Springer, 2001.
8. I. Kátai and B. Kovács. *Canonical number systems in imaginary quadratic fields*, Acta Math. Hungar. **37** (1981), 159–164.
9. I. Kátai and J. Szabó. *Canonical Number Systems for Complex Integers.* Acta Scientiarum Mathematicarum, 1975, pp. 255–260
10. E.W. Knudsen. *Elliptic Scalar Multiplication Using Point Halving.* In: *Proocedings of ASIACRYPT 1999*, LNCS 1716, pp. 135–149. Springer, 1999.
11. N. Koblitz. *Elliptic curve cryptosystems.* Mathematics of computation **48**, pp. 203–209, 1987.
12. N. Koblitz. *CM-curves with good cryptographic properties.* In: *Proceedings of CRYPTO 1991*, LNCS 576, pp. 279–287. Springer, 1991.
13. V. S. Miller. *Use of elliptic curves in cryptography.* In: *Proceedings of CRYPTO '85.* LNCS 218, pp. 417–426. Springer, 1986.
14. J.A. Muir and D.R. Stinson. *Alternative digit sets for nonadjacent representations.* In: Proceedings of SAC 2003. LNCS 3006, pp. 306–319. Springer, 2004.

15. J.A. Muir and D.R. Stinson. *Minimality and other properties of the width-w nonadjacent form*. Mathematics of Computation **75** (2006), 369–384.
16. K. Okeya, T. Takagi, and C. Vuillaume. Short Memory Scalar Multiplication on Koblitz Curves. In *Proceedings of CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*, pages 91–105. Springer, 2005.
17. D. J. Park, S. G. Sim, and P. J. Lee. Fast scalar multiplication method using change-of-basis matrix to prevent power analysis attacks on koblitz curves. In Springer Verlag, editor, *Proceedings of WISA 2003*, volume 2908 of *Lecture Notes in Computer Science*, pages 474–488, 2003.
18. 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.
19. R. Schroeppel. *Elliptic curve point ambiguity resolution apparatus and method.* International Application Number PCT/US00/31014, filed 9 November 2000.
20. 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.
21. J. A. Solinas. *Efficient Arithmetic on Koblitz Curves.* Designs, Codes and Cryptography **19** (2/3), pp. 125–179, 2000.
22. IEEE Std 1363-2000. *IEEE Standard Specifications for Public-Key Cryptography*. IEEE Computer Society, August 29, 2000.
23. National Institute of Standards and Technology. *Digital Signature Standard.* FIPS Publication 186-2, February 2000.

Appendices

A Proof of Theorem 1

Lemma 1. *Let $z \in V$ and $d \in \mathcal{D}$ with $d \equiv z \pmod{\tau^w}$. Then $(z-d)/\tau^k \in V$.*

Proof. By construction, $(z-d)/\tau^w$ is an element of $\mathbb{Z}[\tau]$. We have

$$\sqrt{N\left(\frac{z-d}{\tau^w}\right)} = \frac{|z-d|}{2^{w/2}} \le \frac{|z|+|d|}{2^{w/2}} \le \frac{\frac{\max\{|d|:d\in\mathcal{D}\}}{(2^{w/2}-1)}+|d|}{2^{w/2}} \le \frac{\max\{|d|:d\in\mathcal{D}\}}{(2^{w/2}-1)} \;.$$

Since $N((z-d)/\tau^w)$ is an integer, we conclude that $N((z-d)/\tau^w) \le M$.

Proof. (Theorem 1) We first assume that $\mathcal{D}$ is a w-NADS. Let $z = a + b\tau$ be relatively prime to τ and let $\boldsymbol{\varepsilon}$ be its $\mathcal{D}$-w-NADS. Obviously, we have $1 \equiv a \equiv z \equiv \varepsilon_0 \pmod{\tau}$. Thus, $\varepsilon_0 \neq 0$ and therefore $\varepsilon_1 = \cdots = \varepsilon_{w-1} = 0$, whence $z \equiv \varepsilon_0 \pmod{\tau^w}$. Thus $\mathcal{D}$ contains a reduced residue system modulo τ^w.

Assume that $0 \neq z \in V$. Let $\boldsymbol{\varepsilon}$ be the $\mathcal{D}$-w-NADS of z. If $\varepsilon_0 = 0$, we set $y = z/\tau$. Clearly, $y \in V$ and $(y, z) \in A$. If $\varepsilon_0 \neq 0$, we set $y = (z - \varepsilon_0)/\tau^w$, which is an element of V by Lemma 1. Again, $(y, z) \in A$. In both cases, a $\mathcal{D}$-w-NAF of y can be obtained by omitting the last digit(s) of $\boldsymbol{\varepsilon}$. Repeating this finitely often, we arrive at 0. Using the arcs in reverse order we see that z is reachable from 0.

Conversely, we assume that the two conditions are fulfilled. We first show that every $z \in V$ has a $\mathcal{D}$-w-NAF by induction on the distance from 0 to z in G. Let $z \in V$. Then there is a $y \in V$ with has a smaller distance from 0 than z and is a predecessor of z in G. By induction, y has a $\mathcal{D}$-w-NAF. Depending on whether

$z = \tau^w y + d$ for some nonzero $d \in \mathcal{D}$ or $z = \tau y$, we get a $\mathcal{D}$-w-NAF of z by appending $(0, 0, \ldots, 0, d)$ ($w - 1$ zeros) or 0 to the $\mathcal{D}$-w-NAF of y.

Next, we prove that all $z \in \mathbb{Z}[\tau]$ have a $\mathcal{D}$-w-NAF by induction on $N(z)$. Let $z \in \mathbb{Z}[\tau]$. We may assume that $N(z) > M$ and therefore

$$|z|(2^{w/2} - 1) > \max\{|d| : d \in \mathcal{D}\} \ .$$

If τ divides z, we set $y = z/\tau$ with $N(y) = N(z)/2$. If τ does not divide z, we have $\gcd(z, \tau) = 1$. Thus there are $d \in \mathcal{D}$ and $y \in \mathbb{Z}[\tau]$ with $z = \tau^w y + d$. We have

$$\sqrt{N(y)} = \frac{|z - d|}{2^{w/2}} < \frac{|z| + |z|(2^{w/2} - 1)}{2^{w/2}} = \sqrt{N(z)} \ .$$

Thus we may take a $\mathcal{D}$-w-NAF of y and append 0 or $(0, 0, \ldots, 0, d)$ ($w - 1$ zeros) respectively to obtain a $\mathcal{D}$-w-NAF of z.

Finally we assume that $\mathcal{D} \setminus \{0\}$ is a reduced residue system modulo τ^w and $\mathcal{D}$ is a w-NADS. Assume that some z has two $\mathcal{D}$-w-NAFs $\boldsymbol{\varepsilon}$ and $\boldsymbol{\eta}$. If $z \equiv 0 \pmod{\tau}$, we must have $\varepsilon_0 = \eta_0 = 0$ and we continue with z/τ. If $z \not\equiv 0 \pmod{\tau}$, then we must have $\varepsilon_0 \neq 0$ and $\eta_0 \neq 0$, and the w-NAF property implies that $\varepsilon_j = \eta_j = 0$ for $1 \leq j < w$. Therefore, we have $\varepsilon_0 \equiv \eta_0 \pmod{\tau^w}$, whence $\varepsilon_0 = \eta_0$. Thus we continue with $(z - \varepsilon_0)/\tau^w$. By induction, we see that $\boldsymbol{\varepsilon} = \boldsymbol{\eta}$.

B Proof of Proposition 2

Assume first that (5) holds. If $\varepsilon_{w-1} = \varepsilon'_{w-1}$, then it is clear that $\mathsf{value}(\boldsymbol{\varepsilon}) \equiv \mathsf{value}(\boldsymbol{\varepsilon}') \pmod{\tau^w}$. W.l.o.g., we may now assume that $\varepsilon_{w-1} = 1$ and $\varepsilon'_{w-1} = -1$. In this case we have $\mathsf{value}(\boldsymbol{\varepsilon}) - \mathsf{value}(\boldsymbol{\varepsilon}') \equiv 2\tau^{w-1} \equiv 0 \pmod{\tau^w}$ by (2).

To prove the converse direction, we proceed by induction on w. For $w = 1$, we note that $\varepsilon_0 \equiv \mathsf{value}(\boldsymbol{\varepsilon}) \equiv \mathsf{value}(\boldsymbol{\varepsilon}') \equiv \varepsilon'_0 \pmod{\tau}$ implies $|\varepsilon_0| = |\varepsilon'_0|$ since both least significant digits are elements of $\{0, \pm 1\}$. We now consider the case of general w. Assume that $\mathsf{value}(\boldsymbol{\varepsilon}) \equiv \mathsf{value}(\boldsymbol{\varepsilon}') \pmod{\tau^w}$. By induction hypothesis, we have $\varepsilon_j = \varepsilon'_j$ for $0 \leq j \leq w - 3$ and $|\varepsilon_{w-2}| = |\varepsilon'_{w-2}|$.

We first consider the case that $\varepsilon_{w-2} = \varepsilon'_{w-2}$. In that case we conclude that $\mathsf{value}(\boldsymbol{\varepsilon}) - \mathsf{value}(\boldsymbol{\varepsilon}') \equiv (\varepsilon_{w-1} - \varepsilon'_{w-1})\tau^{w-1} \pmod{\tau^w}$, which implies that $\varepsilon_{w-1} \equiv \varepsilon'_{w-1} \pmod{\tau}$ and therefore $|\varepsilon_{w-1}| = |\varepsilon'_{w-1}|$. Thus (5) is proved in this case.

Finally, we consider the case that $\varepsilon_{w-2} \neq \varepsilon'_{w-2}$. W.l.o.g., we may assume that $\varepsilon_{w-2} = 1$ and $\varepsilon'_{w-2} = -1$. Since $\boldsymbol{\varepsilon}$ and $\boldsymbol{\varepsilon}'$ are both τ-NAFs, the subsequent digits ε_{w-1} and ε'_{w-1} must both vanish. But this implies that $\mathsf{value}(\boldsymbol{\varepsilon}) - \mathsf{value}(\boldsymbol{\varepsilon}') \equiv 2\tau^{w-2} \equiv \mu\tau^{w-1} \pmod{\tau^w}$, a contradiction. Thus this case cannot occur.

C Proof of Theorem 3

For $w \in \{1, 2\}$, all choices of $\mathcal{D}$ are those studied in Examples 2 and 3. These turned out to be w-NADS. For $w = 3$, there are only the possibilities $\mathcal{D} = \{0, 1, -1, \pm\tau^2 + 1, \pm\tau^2 - 1\}$ for independent signs in front of τ^2. Using Theorem 1, these have been checked and Table 1 has been established based on the results.

So the only remaining case is that of $w \geq 4$. Let $z \in \mathbb{Z}[\tau]$ be relatively prime to τ, choose $d = \mathsf{value}(\boldsymbol{\varepsilon}) \in \mathcal{D}$ such that $d \equiv z \pmod{\tau^w}$ and set $y = (z-d)/\tau^w$. Denote the τ-NAF of z by $\boldsymbol{\eta}$. We set $y' := \sum_{j\geq 0} \eta_{j+w}\tau^j$, i.e., the number created by truncating the least significant w digits of the τ-NAF of z.

We claim that either $y = y'$ or y' is *even* (i.e. it is a multiple of τ) and $y \in \{y', y' \pm \bar{\tau}\}$. If $\eta_{w-1} = 0$ then the number formed by the w least significant digits of $\boldsymbol{\eta}$, which is $(0\,\eta_{w-2}\,\ldots\,\eta_1\,\eta_0)_\tau$, is in $\mathcal{D}$, hence it is d and $y = y'$. Otherwise $\eta_w = 0$ and y' is even, and from Proposition 2 together with the fact that $\bar{\tau} = 2 \cdot \tau^{-1}$ we see that $y \in \{y', y' \pm \bar{\tau}\}$.

Next, we want to show that the length of the τ-NAF of $y' \pm \bar{\tau}$ is at most the length of the τ-NAF of y' increased by 3. If we can prove this, since the length of the τ-NAF of y' equals the length of the τ-NAF of z decreased by w, we conclude that the length of the τ-NAF of y is smaller than the length of the τ-NAF of z. From this it follows that repeatedly choosing $d \equiv z \pmod{\tau^w}$ in $\mathcal{D}$ and replacing z with $(z-d)/\tau^w$ will eventually terminate with 0 and yield a $\mathcal{D}$-w-NAF of z.

To prove our claim about the length of the τ-NAF of $y' \pm \bar{\tau}$ we study the behaviour of even τ-adic NAFs upon addition or subtraction of $\bar{\tau}$. We therefore consider *transducers* which compute the τ-NAF of $y' \pm \bar{\tau}$ from the τ-NAF of y'. One such transducer for the case $\mu = -1$ is shown in Figure 2. The transducer for $\mu = 1$ is similar and has been omitted for space reasons. The labels have been chosen to represent the carry, and the "τ-point" corresponds to the look-ahead.

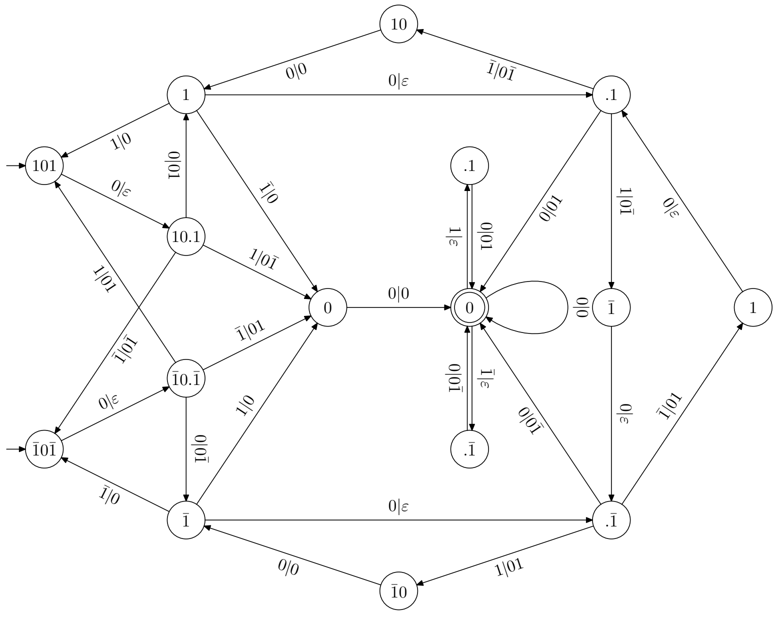

Fig. 2. Transducer for the addition of $\pm\bar{\tau}$ for $\mu = -1$. Addition of $\bar{\tau}$ and $-\bar{\tau}$ corresponds to starting at states 101 and $\bar{1}0\bar{1}$, respectively.

They work as follows: Suppose that $\mu = -1$ and that we want to add $\bar{\tau}$ to $(100\bar{1}0\bar{1}0)_\tau = \tau^6 - \tau^3 - \tau$. We start with the state labeled 101. From each state we go to the next following the edge whose label begins with next digit, and the corresponding output is the part of the label after the | sign. Following the edges with labels beginning with $0, \bar{1}, 0, \bar{1}, 0, 0, 1$ (and possibly additional "most significant" zeros until the output becomes composed exclusively by zeros too) we record the outputs $\varepsilon, 0\bar{1}, \varepsilon, 01, 0, 0, \varepsilon$ and 01, corresponding to the number $(100010\bar{1})_\tau = \tau^6 + \tau^2 - 1$. Indeed, $(100\bar{1}0\bar{1}0)_\tau + \bar{\tau} = (100010\bar{1})_\tau$. Transducers can be easily made into explicit algorithms employing a table look up.

From these transducers, it is easily seen that the length of the τ-NAF of $y' \pm \bar{\tau}$ is at most the length of the τ-NAF of y' increased by 3. This concludes the proof of the theorem.

D Proof of Theorem 4

We first consider the case $w \geq 4$. By definition of $\mathcal{D}$, every nonzero digit of $\mathcal{D}$ has a τ-NAF of length at most w. Replacing each block $(0, \ldots, 0, d)$ of $\boldsymbol{\varepsilon}$ by the τ-NAF of $d \in \mathcal{D}$ yields a $\{0, \pm 1\}$-expansion $\boldsymbol{\eta}$ of z. By construction, this expansion $\boldsymbol{\eta}$ has the following property: if $|\eta_j| = |\eta_{j+1}|$ holds for some j, then the block $(\eta_{j+w}, \ldots, \eta_{j+1})$ satisfies the 2-NAF condition, i.e., $\eta_{k+1} \cdot \eta_k = 0$ for $j+1 \leq k \leq j+w-1$. Furthermore, we have

$$\mathsf{length}(\boldsymbol{\varepsilon}) \leq \mathsf{length}(\boldsymbol{\eta}) \leq \mathsf{length}(\boldsymbol{\varepsilon}) + w - 1 \ . \qquad (10)$$

We now derive a bound for $\mathsf{length}(\boldsymbol{\eta})$ which is independent of w. To that aim, we relax the above syntactical condition. More precisely, we only use that

$$\begin{aligned}\boldsymbol{\eta} \in \mathcal{L} := \Big\{ \boldsymbol{\theta} \in \{0, \pm 1\}^{\mathbb{N}_0} \ &: \text{There is no } j \in \mathbb{N}_0 \text{ such that} \\ &\quad |\theta_j| = |\theta_{j+1}| = |\theta_{j+2}| = 1 \text{ or such that} \\ &\quad |\theta_j| = |\theta_{j+1}| = |\theta_{j+3}| = |\theta_{j+4}| = 1 \Big\} \ .\end{aligned}$$

We denote the maximum and the minimum of the norm of words of $\mathcal{L}$ of length d by

$$\begin{aligned} N^{\mathcal{L}}_{\max}(d) &:= \max\{N(\mathsf{value}(\boldsymbol{\theta})) : \boldsymbol{\theta} \in \mathcal{L} \text{ and } \mathsf{length}(\boldsymbol{\theta}) = d\} \ , \\ N^{\mathcal{L}}_{\min}(d) &:= \min\{N(\mathsf{value}(\boldsymbol{\theta})) : \boldsymbol{\theta} \in \mathcal{L} \text{ and } \mathsf{length}(\boldsymbol{\theta}) = d\} \ . \end{aligned}$$

Solinas' [21] estimates (from Lemma 35 to Corollary 51) in the case of the τ-NAF remain valid for our quantities $N^{\mathcal{L}}_{\max}(d)$ and $N^{\mathcal{L}}_{\min}(d)$. Thus in our case, Solinas' Theorem 2 reads

$$\left(\sqrt{N^{\mathcal{L}}_{\min}(d)} - \frac{\sqrt{N^{\mathcal{L}}_{\max}(d)}}{2^{d/2} - 1}\right)^2 \cdot 2^{\mathsf{length}(\boldsymbol{\eta}) - d} < N(z) < \frac{N^{\mathcal{L}}_{\max}(d)}{(2^{d/2} - 1)^2} \cdot 2^{\mathsf{length}(\boldsymbol{\eta})}$$

for $\mathsf{length}(\boldsymbol{\eta}) > 2d$. We calculate that

$$\begin{aligned} N^{\mathcal{L}}_{\min}(13) &= 86 \ , & N^{\mathcal{L}}_{\max}(13) &= 18288 \ , \\ N^{\mathcal{L}}_{\min}(15) &= 289 \ , & N^{\mathcal{L}}_{\max}(15) &= 73850 \ . \end{aligned}$$

This yields

$$2\log_2|z| - 1.18830 < \mathsf{length}(\boldsymbol{\eta}) < 2\log_2|z| + 7.08685 \tag{11}$$

for $\mathsf{length}(\boldsymbol{\eta}) > 30$. This bound is present also in Solinas [21, Eq. (53) and § 9], but it is an unnecessary restriction: If $\mathsf{length}(\boldsymbol{\eta}) \le 30$, we consider $\tau^k z$ for a sufficiently large integer k and the expansion $\boldsymbol{\eta}' \in \mathcal{L}$ defined by

$$\eta'_j = \begin{cases} 0, & \text{if } j < k, \\ \eta_{j-k}, & \text{if } j \ge k, \end{cases}$$

i.e., $\boldsymbol{\eta}'$ is $\boldsymbol{\eta}$ shifted left by k digits. Since (11) holds for $\tau^k z$ and $\boldsymbol{\eta}'$, it also holds for z and $\boldsymbol{\eta}$.

Together with (10), we obtain (7) for $w \ge 4$.

To obtain the bound for $w = 3$, we consider all 3-NADS (by Theorem 3, there are 4 of them). In these concrete cases, the above calculations can be performed directly, yielding our bound. The case $w = 2$ is contained in Solinas [21, Equation (53)].

E Proof of Theorem 5

The assertion has already been proved for $w = 2$ in Example 3, so we assume that $w \ge 3$ in the sequel. We first claim that for $w \ge 3$, we have

$$v_\tau(\bar{\tau}^{2^{w-2}} - 1) = w \ , \tag{12a}$$

$$v_\tau(\bar{\tau}^{2^{w-2}} + 1) = 1 \ , \tag{12b}$$

where for $z \in \mathbb{Z}[\tau]$, $v_\tau(z)$ denotes the maximal integer k such that τ^k divides z.

Now, (12b) is an immediate consequence of (12a) and the fact that $v_\tau(2) = 1$. For $w = 3$, we have $\bar{\tau}^2 = \mu\tau^3 + 1$, which proves (12a) in this case. For $w \ge 4$, we note that $v_\tau(\bar{\tau}^{2^{w-2}} - 1) = v_\tau(\bar{\tau}^{2^{w-3}} - 1) + v_\tau(\bar{\tau}^{2^{w-3}} + 1) = (w-1) + 1 = w$, thus (12a) is proved by induction.

Since the unit group of $\mathbb{Z}[\tau]/\tau^w\mathbb{Z}[\tau]$ has order 2^{w-1} by Remark 2, the order of $\bar{\tau}$ modulo τ^w is a power of 2. By (12a), we have $\bar{\tau}^{2^k} \equiv 1 \pmod{\tau^w}$ if and only if $k \ge w-2$, thus

$$\bar{\tau} \text{ has order } 2^{w-2} \text{ modulo } \tau^w \ . \tag{13}$$

Assume that $\bar{\tau}^\ell \equiv -\bar{\tau}^k \pmod{\tau^w}$ for some $0 \le k < \ell < 2^{w-2}$. We get $\bar{\tau}^{\ell-k} \equiv -1 \pmod{\tau^w}$. By (13), squaring this congruence shows that 2^{w-2} divides $2(\ell-k) < 2^{w-1}$, thus $(\ell-k) \in \{0, 2^{w-3}\}$. Taking into account (12b), we see that both cases lead to a contradiction.

Since all elements of $\mathcal{D}'$ are relatively prime to τ^w, they are pairwise incongruent modulo τ^w and the cardinality of $\mathcal{D}'$ equals 2^{w-1}, the proof is completed.