

SCALAR MULTIPLICATION ON KOBLITZ CURVES USING THE FROBENIUS ENDOMORPHISM AND ITS COMBINATION WITH POINT HALVING: EXTENSIONS AND MATHEMATICAL ANALYSIS

ROBERTO M. AVANZI[†], CLEMENS HEUBERGER[‡], AND HELMUT PRODINGER^{*}

ABSTRACT. In this paper we prove the optimality and other properties of the τ -adic non-adjacent form: this expansion has been introduced in order to efficiently compute scalar multiplications on Koblitz curves. We also refine and extend results about double expansions of scalars introduced by Avanzi, Ciet and Sica in order to further improve scalar multiplications. Our double expansions are optimal and their properties are carefully analysed. In particular we provide first and second order terms for the expected weight, determine the variance and prove a central limit theorem. Transducers for all the involved expansions are provided, as well as automata accepting all expansions of minimal weight.

1. INTRODUCTION

In 1985 Miller [11] and Koblitz [9] independently proposed to design cryptosystems around the group of rational points of an elliptic curve over a finite field. The performance of any such cryptosystem depends on the efficiency of their fundamental operation, the *scalar multiplication*: Given a point \mathbf{P} and an integer s , compute $s\mathbf{P}$. The generic way of computing scalar multiplications is to use a double-and-add method (isomorphic to a Horner scheme) and a (possibly redundant) base 2 representation of the scalar, the representation of choice being the *width- w non-adjacent form*, or *w -NAF*, independently introduced by Miyaji et al. [12] and Solinas [18] (see also [1, 13, 14] for more properties and similar methods).

Some families of elliptic curves have arithmetic properties which can be successfully exploited to considerably speed up this operation. Noteworthy are the curves defined by

$$(1) \quad E_a : y^2 + xy = x^3 + ax^2 + 1 \quad \text{with} \quad a \in \{0, 1\}$$

over a finite field \mathbb{F}_{2^n} . They were first proposed by Koblitz [10], whence their name *Koblitz curves*. Solinas [17, 18] called them *anomalous binary curves*. The benefit of using them comes from the fact that a scalar multiplication can be performed very efficiently using the Frobenius endomorphism τ . This is the map induced on the curve by the Frobenius automorphism of the field extension $\mathbb{F}_{2^n}/\mathbb{F}_2$, which maps a field element to its square. In fact, the evaluation of the Frobenius is much faster than the addition of two distinct points

[†] This paper was written while this author was a visitor at the John Knopfmacher Centre for Applicable Analysis and Number Theory, School of Mathematics, University of the Witwatersrand, Johannesburg. He thanks the centre for its hospitality.

[‡] This paper was written while this author was a visitor at the John Knopfmacher Centre for Applicable Analysis and Number Theory, School of Mathematics, University of the Witwatersrand, Johannesburg. He thanks the centre for its hospitality. He was also supported by the grant S8307-MAT of the Austrian Science Fund.

^{*} This author is supported by the grant NRF 2053748 of the South African National Research Foundation.

on the curve or the computation of the double of a given point: τ consists just in the squaring of the coordinates, and if a suitable representation of the field \mathbb{F}_{2^n} is chosen squarings are computationally almost free operations.

Instead of computing $s\mathbf{P}$ using a double-and-add method, one uses an expression $\sum_{i=0}^{n'} s_i \tau^i$ with $n' \lesssim n$ —by this it is understood that $n' \leq n + c$ for a small absolute constant c —and such that $s\mathbf{P} = \sum_{i=0}^{n'} s_i \tau^i(\mathbf{P})$. Such an expression can be evaluated easily via a Horner scheme, resulting in an algorithm using only repeated applications of τ interspersed with additions of the base point \mathbf{P} . Solinas introduced a method for computing the s_i 's efficiently leading to a “representation” of any given scalar s where on average $n/3$ of the s_i are non-zero. His representation satisfies the syntactical property $s_i s_{i+1} = 0$, akin to that of the non adjacent form of any integer. Because of this his recoding is called the τ -NAF.

Knudsen [8] and Schroepel [16] independently proposed a technique to speed up scalar multiplication on all elliptic curves over binary fields based on *point halving*. This method computes the multiple \mathbf{R} of any point \mathbf{P} of odd order such that $2\mathbf{R} = \mathbf{P}$ and $\mathbf{R} \in \langle \mathbf{P} \rangle$. The point \mathbf{R} is denoted as $\frac{1}{2}\mathbf{P}$. Since for curves of order twice a prime point halving is up to three times as fast as point doubling, it is possible to improve performance of scalar multiplication by expanding the scalar using “powers of $1/2$ ” and replacing the double-and-add algorithm with a halve-and-add method.

The present paper deals with properties of integer expansions which are associated to Koblitz curves, as well as with techniques combining point halving and Frobenius expansions introduced in [2]. The latter, which are used to improve the speed of scalar multiplication, are extended in an optimal way, and a complexity analysis of the resulting methods is provided.

The techniques in [2] stem from the following observation. Let $\mathbf{Q} := \tau(\frac{1}{2}\mathbf{P})$. There exist expressions of the form $\sum_{i=0}^k \varepsilon_i \tau^i$ where $\varepsilon_i \in \{0, \pm 1\}$, $\varepsilon_0 \varepsilon_k \neq 0$ with the property that $\sum_{i=0}^k \varepsilon_i \tau^i(\mathbf{P}) = \sum_{i=0}^{k'} \delta_i \tau^i(\mathbf{Q})$ for suitable δ_i 's in $\{0, \pm 1\}$ —but the number of non-zero δ_i 's is considerably smaller than the amount of non-zero ε_i 's. In [2] three different types of such expressions are presented which are then used to compute $s\mathbf{P}$ as

$$(2) \quad \sum_{i=0}^{n_1} s_i^{(1)} \tau^i(\mathbf{P}) + \sum_{i=0}^{n_2} s_i^{(2)} \tau^i(\mathbf{Q})$$

with $n_1, n_2 \lesssim n$. The difference with Solinas' method is that the number of non-zero coefficients among the $s_i^{(j)}$ is now about $2n/7$. The scalar multiplication can be performed without additional precomputations but doubling the number of Frobenius applications. This still leads to a non-negligible speed-up (cf. [2, Algorithm 3]). Alternatively, \mathbf{Q} can be precomputed and the number of Frobenius applications corresponds to Solinas' method.

All the expressions presented in [2] share the property that only two of the δ_i 's are non-zero. Ciet's thesis [3] contains a heuristic approach for deriving more complicated expressions from the given ones—but the resulting improvement is minimal. A brute-force search on a computer reveals that there is at least another family of τ -adic expressions simplifying to an expression with two non-zero coefficients, and this has been the starting point for the present research. The main aim of this paper is to refine the methods in [2] and [3] giving an optimal splitting of the type (2), and to give a more precise complexity analysis.

It also turned out that some information about the τ -NAF was apparently missing from the literature. In particular, we found no proof that the τ -NAF is a τ -adic recoding of minimal

weight. Also, explicit transducers for computing the τ -NAF have not been described. Exactly as for the NAF, the τ -NAF is some recoding of minimal weight, but it is not the only minimal expansion. However, the non-adjacency property implies that every number has a unique τ -NAF. We then provide automata that accept as valid inputs only the recodings of minimal weight. These results are collected in Section 2.

In the section that follows we turn our attention to the double expansions of type (2). We introduce a double expansion which we call the wide-double-NAF, prove that it has minimal weight among all the double expansions, and provide automata that validate the double expansions of minimal weight. The average weight of the wide-double-NAF is $n/4$. Transducers computing the wide-double-NAF of scalars are also provided, which can be trivially transformed in recoding algorithms that employ table look-ups. Finally, a complexity analysis with second order terms, variance and a central limit theorem is given.

Section 4 contains a refined analysis of the “double digits” in the wide-double-NAF.

An appendix contains some illustrations showing fractals that appear when evaluating some τ -NAFs, as well as odometers for adding ± 1 to an existing τ -NAF.

2. τ -EXPANSIONS

We consider here a curve defined by equation (1) over a finite field \mathbb{F}_{2^n} and set $\mu = (-1)^{1-a}$. Let τ denote the Frobenius automorphism of the field extension $\mathbb{F}_{2^n}/\mathbb{F}_2$, i.e., $\tau(x) = x^2$. Since the equation of the curve E_a is invariant under τ , this map permutes the \mathbb{F}_{2^n} -rational points of the curve. It is well-known (cf. Solinas [18, Section 4.1]) that for each point $\mathbf{P} \in E_a(\mathbb{F}_{2^n})$, we have $(\tau^2 + 2)\mathbf{P} = \mu\tau(\mathbf{P})$, which implies that we can identify τ with the complex number satisfying

$$(3) \quad \tau^2 + 2 = \mu\tau.$$

Now, for any $z \in \mathbb{Z}[\tau]$, a τ -expansion of z is an expression $\mathbf{s} = (\dots, s_2, s_1, s_0) \in \{-1, 0, 1\}^{\mathbb{N}_0}$ such that only finitely many $s_j \neq 0$ and $\text{value}_\tau(\mathbf{s}) := \sum_{j \geq 0} s_j \tau^j = z$. We will identify finite and (left) infinite sequences in the natural way by padding with leading zeros. The Hamming weight of \mathbf{s} is the number of $j \geq 0$ such that $s_j \neq 0$.

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

A τ -Nonadjacent-Form (τ -NAF) of z is a τ -expansion \mathbf{s} of z satisfying $s_j s_{j+1} = 0$ for all $j \geq 0$, i.e., an expansion that does not contain adjacent nonzero digits. Solinas [18, Section 4.2, Theorem 1] showed that each $z \in \mathbb{Z}[\tau]$ has a unique τ -NAF.

In Figures 1 and 2, there is a transducer which computes the τ -NAF of an integer from any other τ -expansion from right to left for $\mu = -1$ and $\mu = 1$, respectively. In various places, we write $\bar{1}$ for -1 and ε for the empty word. We note that all transducers in this paper read their input and write their output from right to left. For all transitions $i \xrightarrow{d|\mathbf{o}} j$ in the transducers in Figures 1 and 2, we have $d + \text{value}_\tau(i) = \tau(\text{value}_\tau(j) + \text{value}_\tau(\mathbf{o}))$, where $\text{value}_\tau(s_k \dots s_0.s_{-1} \dots s_{-\ell}) = \sum_{j=-\ell}^k s_j \tau^j$. Furthermore, if i has ℓ_i and j has ℓ_j digits after the “ τ -point”, then \mathbf{o} has length $1 + \ell_i - \ell_j$. This implies that the transducers indeed produce a τ expansion of the value of its input. Obviously, the transducers produce a NAF.

hypothesis without increasing its cost c . We conclude that $\text{value}_\tau(\mathbf{s}) = \text{value}_\tau(\mathbf{s}'')$ for some \mathbf{s}'' such that $s_0'' \in \{1, 2\}$, $(\dots, s_3'', s_2'', s_1'')$ is in τ -NAF and $c(\mathbf{s}'') \leq c(\mathbf{s})$.

We note that for arbitrary t_3, t_4 , we have

$$(4a) \quad \text{value}_\tau(1, 0, 2) = \text{value}_\tau(0, \mu, 0),$$

$$(4b) \quad \text{value}_\tau(0, -\mu, 2) = \text{value}_\tau(-1, 0, 0),$$

$$(4c) \quad \text{value}_\tau(t_3, 0, \mu, 2) = \text{value}_\tau(-\mu + t_3, 0, 0, 0)$$

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

$$(5a) \quad \text{value}_\tau(t_3, 0, 0, 2) = \text{value}_\tau(-\mu + t_3, 0, -\mu, 0),$$

$$(5b) \quad \text{value}_\tau(t_4, 0, -1, 0, 2) = \text{value}_\tau(1 + t_4, -\mu, 0, \mu, 0),$$

$$(5c) \quad \text{value}_\tau(t_3, 0, \mu, 1) = \text{value}_\tau(-\mu + t_3, 0, 0, -1),$$

$$(5d) \quad \text{value}_\tau(0, -\mu, 1) = \text{value}_\tau(-1, 0, -1).$$

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 (4) and (5) 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 (4)) 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 (5), our claim is proved. \square

“Automatic” proof. The same result can also be proved using the argument of [5, Lemma 19] and computations involving the transducers in Figures 1 and 2:

We consider the weighted digraph induced by the transducers in Figures 1 and 2, respectively, with edge weights

$$w(i \xrightarrow{d|o} j) := c(d) - c(o),$$

where $c(\mathbf{s}) := \sum_{j \geq 0} |s_j|$ and $c(\varepsilon) = 0$. By using the Ford-Bellman algorithm (cf. Cook et al. [4]), we conclude that there is no negative cost cycle, which implies that the shortest path (in terms of the costs c) from 0 to 0 has weight 0, i.e., the τ -NAF is a τ -expansion of minimal weight. \square

As in [5, Remark 20], we conclude that any optimal τ -expansion corresponds to edges satisfying $\pi(i) + w(i \xrightarrow{d|o} j) = \pi(j)$, where $\pi(i)$ denotes the vertex potential of vertex i , i.e., the weight of the shortest path from the initial vertex 0 to vertex i . Therefore, the set of optimal τ -expansions equals the set of expansions recognized by the automata in Figures 3 and 4 for $\mu = -1$ and $\mu = 1$, respectively.

Theorem 2. *Let \mathbf{s} be a τ -expansion of some $z \in \mathbb{Z}[\tau]$. Then the Hamming weight of \mathbf{s} is minimum amongst all τ -expansions of z if and only if \mathbf{s} is accepted by the automaton in Figure 3 for $\mu = -1$ or Figure 4 for $\mu = 1$.*

3. NEW SCALAR DECOMPOSITION AND SCALAR MULTIPLICATION

Avanzi, Ciet, and Sica [2] proposed the following method for computing $m\mathbf{P}$, where $m \in \mathbb{Z}$ and $\mathbf{P} \in E_a(\mathbb{F}_{2^n})$. They set $\mathbf{Q} := \tau(\frac{1}{2}\mathbf{P})$, which is easy to compute (cf. [2]), and compute

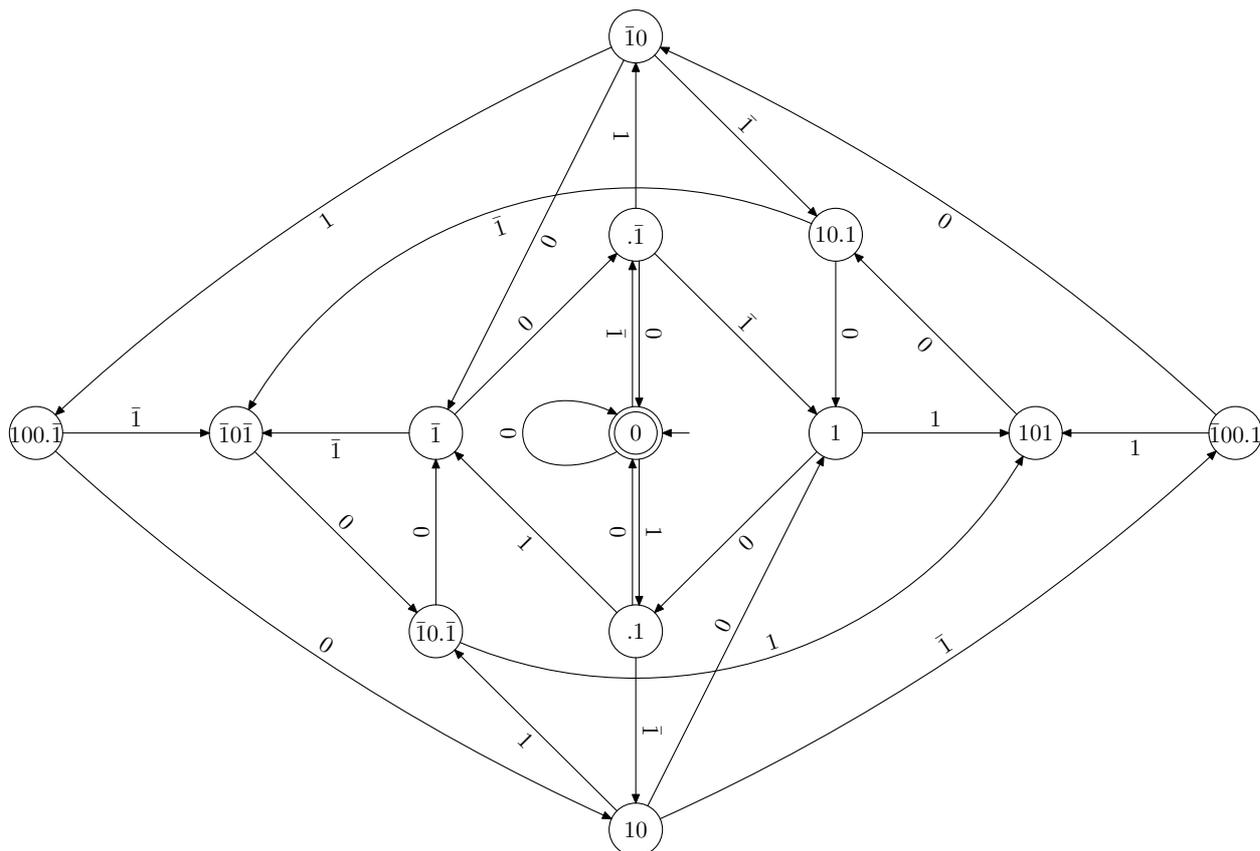


FIGURE 3. Automaton accepting all τ -expansions of minimal weight from right to left for $\mu = -1$.

elements $m_P, m_Q \in \mathbb{Z}[\tau]$ such that $mP = m_P P + m_Q Q$, choosing τ -expansions of m_P and m_Q such that the sum of their Hamming weights is small. We will refine their method giving an optimal such splitting, give a precise analysis involving second order terms, the variance, and a central limit theorem.

Equation (3) implies that $\tau^3 + 2\tau = \mu\tau^2 = \mu(\mu\tau - 2) = \tau - 2\mu$, hence

$$(6) \quad 2 = -\mu(1 + \tau^2)\tau.$$

In particular, this means that we can compute $2P$ as $-\mu(1 + \tau^2)\tau P$. This alone is not very useful, since it replaces a point doubling with one addition and three Frobenius operations. However, these relations become interesting if we can make repeated use of them.

We consider so-called $((-\mu)(1 + \tau^2), 1)$ -double expansions $\begin{pmatrix} s^{(1)} \\ s^{(2)} \end{pmatrix}$, where $s^{(1)}$ and $s^{(2)}$ are just any τ -expansions of arbitrary elements of $\mathbb{Z}[\tau]$. We call two such expansions $\begin{pmatrix} s^{(1)} \\ s^{(2)} \end{pmatrix}$ and $\begin{pmatrix} s'^{(1)} \\ s'^{(2)} \end{pmatrix}$ equivalent and write $\begin{pmatrix} s^{(1)} \\ s^{(2)} \end{pmatrix} \equiv \begin{pmatrix} s'^{(1)} \\ s'^{(2)} \end{pmatrix}$, if $\text{value}_\tau(s^{(1)})(-\mu)(1 + \tau^2) + \text{value}_\tau(s^{(2)}) = \text{value}_\tau(s'^{(1)})(-\mu)(1 + \tau^2) + \text{value}_\tau(s'^{(2)})$. If we have a point $P \in E_a(\mathbb{F}_{2^n})$ and set $Q = \tau(\frac{1}{2}P)$, the relation $\begin{pmatrix} s^{(1)} \\ s^{(2)} \end{pmatrix} \equiv \begin{pmatrix} s'^{(1)} \\ s'^{(2)} \end{pmatrix}$ implies that $\text{value}_\tau(s^{(1)})P + \text{value}_\tau(s^{(2)})Q = \text{value}_\tau(s'^{(1)})P + \text{value}_\tau(s'^{(2)})Q$.

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

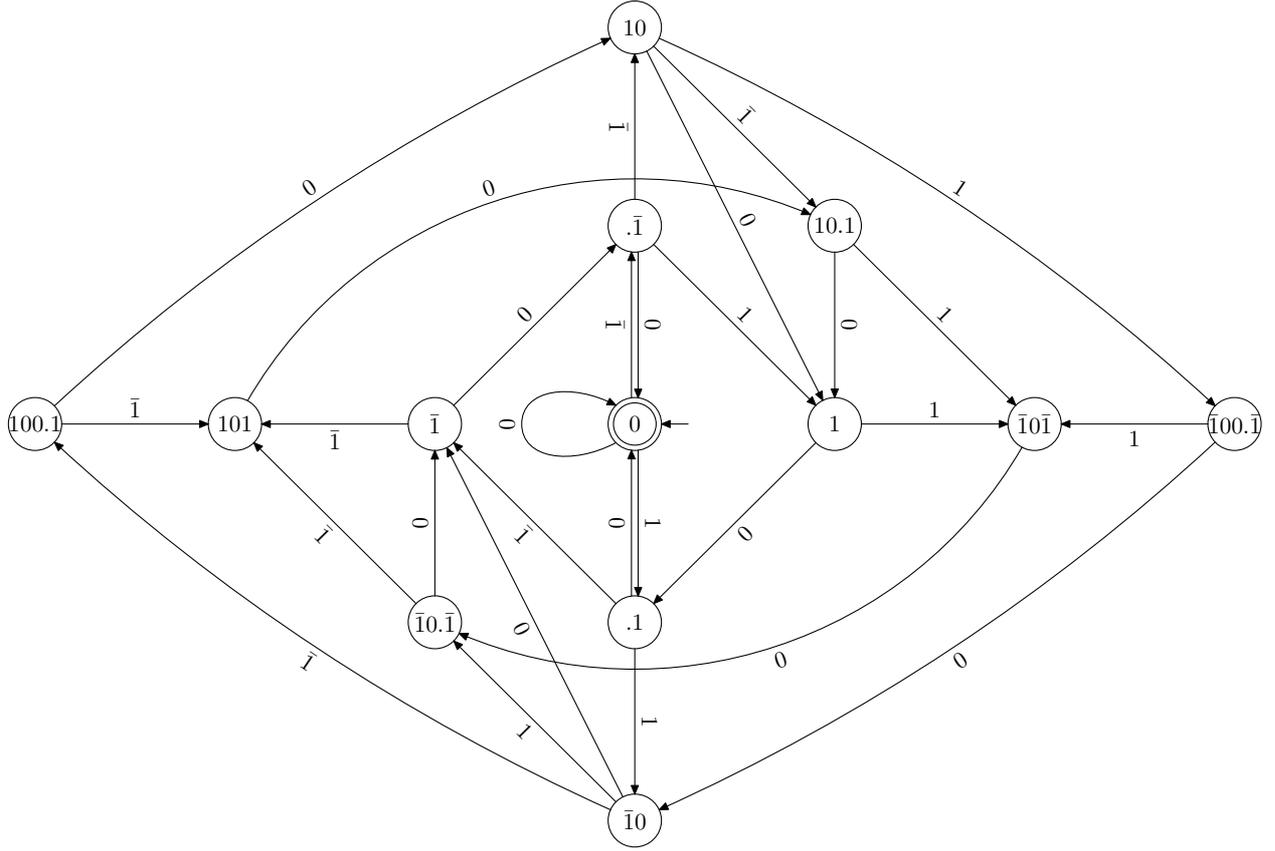


FIGURE 4. Automaton accepting all τ -expansions of minimal weight from right to left for $\mu = 1$.

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 such that the Hamming weight of $\binom{\mathbf{s}^{(1)}}{\mathbf{s}^{(2)}}$ is minimum.

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 $i' = 2 - i$ and $j \geq 0$.

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

$$\left(\varepsilon + \begin{matrix} 1 & \bar{1} & 0 & 0 & 01 & 0\bar{1} & 00 & 00 \\ 0 & 0 & 1 & \bar{1} & 00 & 00 & 01 & 0\bar{1} \end{matrix} \right) \left(\begin{matrix} 0 & 001 & 00\bar{1} & 000 & 000 \\ 0 & 000 & 000 & 001 & 00\bar{1} \end{matrix} \right)^*.$$

We first prove a uniqueness result.

Lemma 1. If \mathbf{s} and \mathbf{s}' are equivalent wide-double-NAFs, then they are equal.

The proof relies on the following extension of Solinas' [18] Lemma 28, which he used to prove the uniqueness of the τ -NAF.

Lemma 2. Consider $z = \sum_{j \geq 0} s_j \tau^j \in \mathbb{Z}[\tau]$. Then

- (1) z is divisible by τ in $\mathbb{Z}[\tau]$ if and only if $s_0 \equiv 0 \pmod{2}$,
- (2) z is divisible by τ^2 in $\mathbb{Z}[\tau]$ if and only if $s_0 + 2s_1 \equiv 0 \pmod{4}$,

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

Proof of Lemma 2. The first two assertions have been proved by Solinas [18]. To prove the last assertion, we see that by definition, the element z is divisible by τ^3 if and only if there are rational integers a_0, a_1 such that $s_0 + s_1\tau + s_2\tau^2 = \tau^3(a_0 + a_1\tau)$. Multiplying with the complex conjugate $\bar{\tau}^3$ of τ^3 , we get $s_0\bar{\tau}^3 + 2s_1\bar{\tau}^2 + 4s_2\bar{\tau} = 8(a_0 + a_1\tau)$ using $\tau\bar{\tau} = 2$. Reducing this equation using $\bar{\tau} = \mu - \tau$ and the minimal polynomial, we obtain

$$8(a_0 + a_1\tau) = -3\mu(s_0 - 2\mu s_1 - 4s_2) - 8s_1 - 8\mu s_2 + \tau(s_0 - 2\mu s_1 - 4s_2).$$

Thus divisibility is equivalent to $s_0 - 2\mu s_1 - 4s_2 \equiv 0 \pmod{8}$. \square

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

By definition of equivalence, we have

$$(7) \quad \sum_{j \geq 0} (s_j^{(1)} - s_j'^{(1)})(-\mu)(1 + \tau^2)\tau^j + \sum_{j \geq 0} (s_j^{(2)} - s_j'^{(2)})\tau^j = 0.$$

From the first assertion of Lemma 2 we conclude that

$$(s_0^{(1)} - s_0'^{(1)})(-\mu) + (s_0^{(2)} - s_0'^{(2)}) \equiv 0 \pmod{2}.$$

Since $s_0^{(i)} = 1$ and $s_0^{(i')} = 0$, we conclude that $\begin{pmatrix} s_0'^{(1)} \\ s_0'^{(2)} \end{pmatrix} \neq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$. This implies that $s_j^{(k)} = s_j'^{(k)} = 0$ for $1 \leq j, k \leq 2$. We set $c = -\mu(s_0^{(1)} - s_0'^{(1)})$ and $d = (s_0^{(2)} - s_0'^{(2)})$.

From (7) we conclude that $c(1 + \tau^2) + d$ is divisible by τ^3 , which implies

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

by Lemma 2. By assumption, we have $(c, d) \neq (0, 0)$ and $|c| + |d| = 2$. This contradicts (8). \square

Now we can prove that a wide-double-NAF indeed exists and minimizes the Hamming weight in its equivalence class.

Theorem 3. *Let \mathbf{s} be a $((-\mu)(1 + \tau^2), 1)$ -double expansion. Then there exists a unique wide-double-NAF which is equivalent to \mathbf{s} . Its Hamming weight is not larger than that of \mathbf{s} .*

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

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

By (the ‘‘direct’’ proof of) Theorem 1, we may replace $(s_j^{(i)})_{j \geq 0}$ by its τ -NAF $(s_j'^{(i)})_{j \geq 0}$ 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

$$(9a) \quad \begin{pmatrix} t_2^{(1)} & 0 & 1 \\ t_2^{(2)} & 0 & \mu \end{pmatrix} \equiv \begin{pmatrix} t_2^{(1)} & 0 & 0 \\ (\bar{\mu} + t_2^{(2)}) & 0 & 0 \end{pmatrix},$$

$$(9b) \quad \begin{pmatrix} 0 & 1 \\ 0 & \bar{\mu} \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 \\ \bar{1} & 0 \end{pmatrix},$$

$$\begin{aligned}
(9c) \quad & \begin{pmatrix} 0 & t_1^{(1)} & 0 \\ 1 & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} 0 & t_1^{(1)} & \bar{\mu} \\ 0 & 0 & 0 \end{pmatrix}, \\
(9d) \quad & \begin{pmatrix} 0 & \bar{1} & 0 \\ t_2^{(2)} & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 & 0 \\ t_2^{(2)} & 0 & \bar{1} \end{pmatrix}, \\
(9e) \quad & \begin{pmatrix} t_2^{(1)} & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \equiv \begin{pmatrix} t_2^{(1)} & 0 & 0 \\ 0 & 0 & \mu \end{pmatrix}, \\
(9f) \quad & \begin{pmatrix} 0 & 0 & 1 \\ \mu & 0 & 0 \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & \bar{\mu} \end{pmatrix}, \\
(9g) \quad & \begin{pmatrix} t_5^{(1)} & t_4^{(1)} & t_3^{(1)} & 0 & 1 & 0 \\ t_5^{(2)} & t_4^{(2)} & 0 & \bar{1} & 0 & 1 \end{pmatrix} \equiv \begin{pmatrix} t_5^{(1)} & t_4^{(1)} & t_3^{(1)} & 0 & 0 & 0 \\ (\mu + t_5^{(2)}) & t_4^{(2)} & 0 & 0 & 0 & \bar{1} \end{pmatrix}.
\end{aligned}$$

We note that in all the above equivalences, the costs c decrease from the left hand side to the right hand side. 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 (9) do not occur (at least in the rightmost digits). Furthermore, we have

$$\begin{aligned}
(10a) \quad & \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}, \\
(10b) \quad & \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 & 0 \\ (\bar{\mu} + t_6^{(2)}) & t_5^{(2)} & t_4^{(2)} & (\bar{1} + t_3^{(2)}) & 0 & 0 & \mu \end{pmatrix}, \\
(10c) \quad & \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}, \\
(10d) \quad & \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}, \\
(10e) \quad & \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}, \\
(10f) \quad & \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}, \\
(10g) \quad & \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}, \\
(10h) \quad & \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}, \\
(10i) \quad & \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}.
\end{aligned}$$

We observe that in each of the above equivalences, the costs do not increase from left to right and that the last three digits of the right hand side is always a block which is allowed in a wide-double-NAF. This means that we can apply the induction hypothesis to the right hand sides with the last three digits removed. Finally, we note that for every s' found above, exactly one of the listed equivalences (or its negative) can be applied. \square

It is straightforward to write down explicitly a computer program that performs the re-coding simply by table look-ups. Of course, the rules (9) and (10) can also be implemented by a transducer. This transducer has 153 states, hence it is not shown in this paper. The transducer can however be used to give an “automatic” optimality proof following the lines of the “automatic” proof of Theorem 1. As a consequence, we also get a characterization of optimal expansions. In this case, the automaton recognizing the optimal expansions can be simplified to 28 states and is shown in Figures 5 and 6 for $\mu = -1$ and $\mu = 1$, respectively.

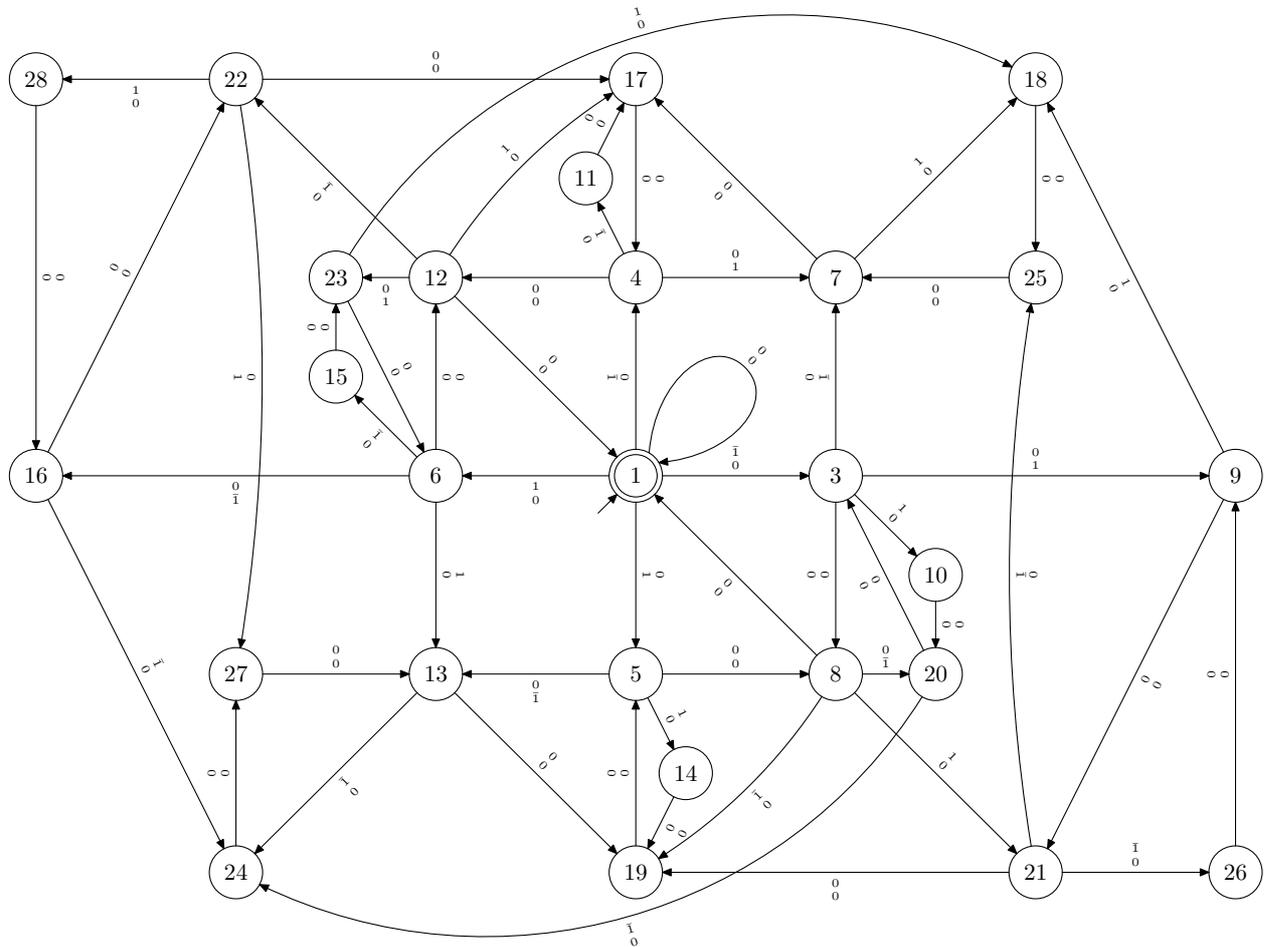


FIGURE 5. Automaton accepting all double expansions of minimal Hamming weight for $\mu = -1$

In our situation, we are given the τ -NAF s of an integer $m \in \mathbb{Z}$ and we are looking for the wide-double-NAF which is equivalent to $\begin{pmatrix} s \\ 0 \end{pmatrix}$. In this case, the above mentioned transducer with 153 states can be considerably reduced—only the inputs $\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \bar{1} \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ can occur, furthermore,

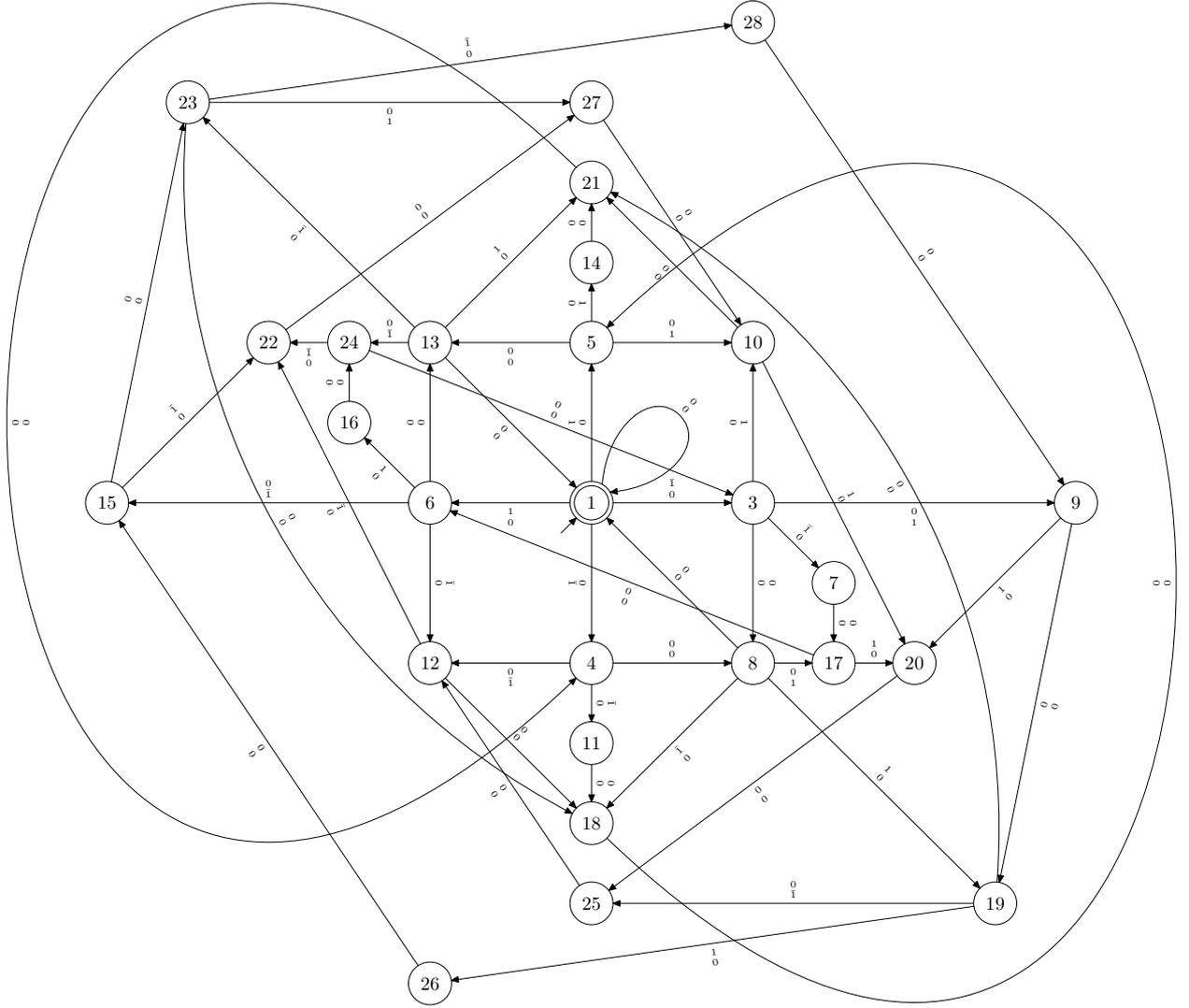


FIGURE 6. Automaton accepting all double expansions of minimal Hamming weight for $\mu = 1$

the NAF-condition on the input can be imposed: The resulting transducer only has 9 states and is shown in Figures 7 and 8 for $\mu = -1$ and $\mu = 1$, respectively.

The labels of the states correspond to carries (before the “ τ -point”) and stored input where no decision could be made up to now (after the “ τ -point”). More precisely, for a transition

$$\mathbf{s}.s' \xrightarrow{d|o} \mathbf{t}.t'$$

we always have

$$\left(\mathbf{s} + \begin{pmatrix} \mathbf{d} \\ \mathbf{0} \end{pmatrix} \right) s' \equiv \mathbf{t}t'$$

and the sum of the lengths of \mathbf{d} and s' equals the sum of the lengths of t' and \mathbf{o} . The output is always a wide-double-NAF.

Not every wide-double-NAF can be reached by applying the transducers in Figures 7 and 8. If we consider the output of these transducers as the input of an nondeterministic

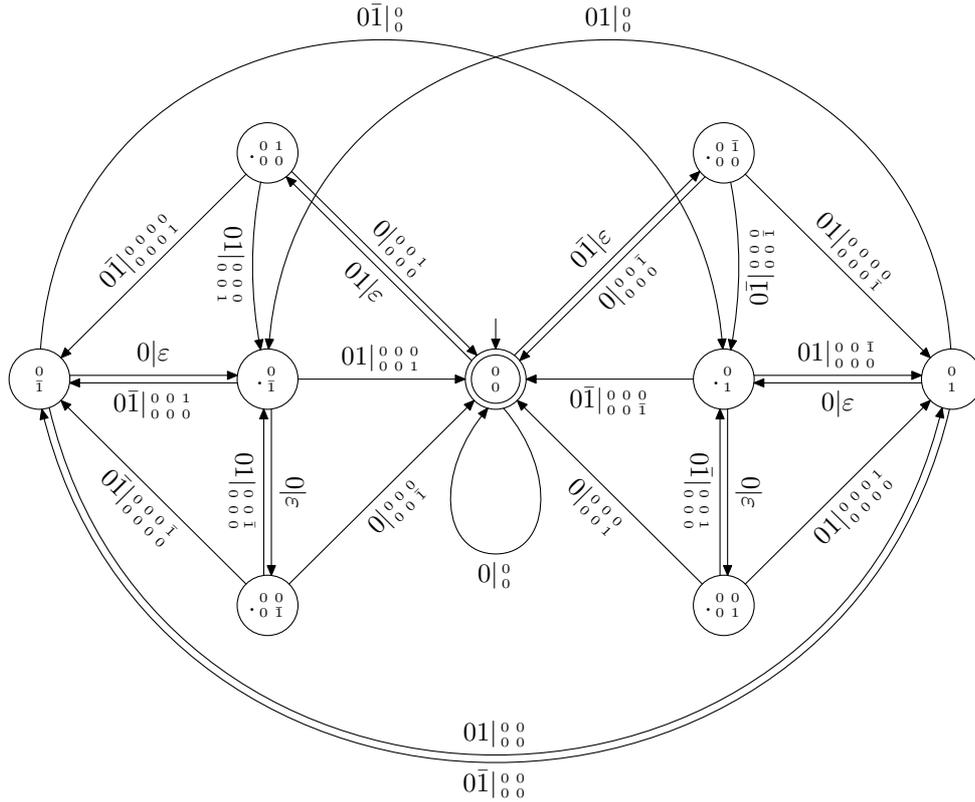


FIGURE 7. Transducer computing the wide-double-NAF equivalent to the (single) input NAF for $\mu = -1$

automaton, convert it to a deterministic automaton and simplify it, we obtain the automaton in Figure 9 (for both values of μ). This means that a wide-double-NAF \mathbf{s} is equivalent to a $\begin{pmatrix} \mathbf{t} \\ \mathbf{0} \end{pmatrix}$ for some simple τ -NAF \mathbf{t} if and only if the number of nonzero entries in the second row is even.

We can also ask which elements can be represented by a double expansion, when we define the value of a double expansion $\begin{pmatrix} \mathbf{s}^{(1)} \\ \mathbf{s}^{(2)} \end{pmatrix}$ to be $\text{value}_\tau(\mathbf{s}^{(1)}) + \frac{1}{(-\mu)(1+\tau^2)} \text{value}_\tau(\mathbf{s}^{(2)})$, i.e., the value in the “ \mathbf{P} -world.” By (6), we see that $\frac{1}{(-\mu)(1+\tau^2)} = \tau/2$. So double expansions represent the elements of $\mathbb{Z}[\tau] + \frac{\tau}{2}\mathbb{Z}[\tau]$. The set $\mathbb{Z}[\tau]$ equals the set $\mathbb{Z} + \tau\mathbb{Z}$, so that we have $\frac{\tau}{2}\mathbb{Z}[\tau] = \frac{\tau}{2}\mathbb{Z} + \frac{\tau^2}{2}\mathbb{Z} = \frac{\tau}{2}\mathbb{Z} + (\frac{\mu}{2}\tau - 1)\mathbb{Z} = \mathbb{Z} + \frac{\tau}{2}\mathbb{Z}$. We conclude that double expansions correspond exactly to the set $\mathbb{Z} + \frac{\tau}{2}\mathbb{Z}$.

We now want to analyze the Hamming weight of the output of the transducers in Figure 7 and 8. We set $a_{k,\ell}^{(\mu)}$ to be the number of τ -NAFs of length ℓ whose output after running the transducers in Figure 7 or 8 has Hamming weight k . We consider the generating function $G^{(\mu)}(U, Z)$ defined as

$$G^{(\mu)}(U, Z) := \sum_{\substack{k \geq 0 \\ \ell \geq 0}} a_{k,\ell}^{(\mu)} U^k Z^\ell.$$

For computing this generating function, we consider the labelled adjacency matrix

$$B^{(\mu)}(U, Z_1, Z_0) := (b_{ij}^{(\mu)}(U, Z_1, Z_0))_{\substack{1 \leq i \leq 9, \\ 1 \leq j \leq 9}}$$

For $\mu = -1$, we have

$$B^{(\mu)}(U, Z_1, Z_0) = \begin{pmatrix} Z_0 & Z_1 & Z_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ UZ_0 & 0 & 0 & UZ_1 & UZ_1 & 0 & 0 & 0 & 0 \\ UZ_0 & 0 & 0 & 0 & 0 & UZ_1 & UZ_1 & 0 & 0 \\ UZ_1 & 0 & 0 & 0 & UZ_1 & 0 & 0 & Z_0 & 0 \\ 0 & 0 & 0 & Z_0 & 0 & Z_1 & Z_1 & 0 & 0 \\ 0 & 0 & 0 & Z_1 & Z_1 & 0 & Z_0 & 0 & 0 \\ UZ_1 & 0 & 0 & 0 & 0 & UZ_1 & 0 & 0 & Z_0 \\ UZ_0 & 0 & 0 & UZ_1 & UZ_1 & 0 & 0 & 0 & 0 \\ UZ_0 & 0 & 0 & 0 & 0 & UZ_1 & UZ_1 & 0 & 0 \end{pmatrix}.$$

In writing down the generating function, we have to care about the fact that our transducer always reads nonzero digits together with its subsequent zero. Thus we have to label such a transition with Z^2 , whereas a transition with input label 0 has to be multiplied by Z . There is one exception to this rule: At the end of our input NAF of given length ℓ , we may have to read a single 1 or -1 without its associated 0, more precisely, we may of course read the associated 0, but we are not allowed to count it in the input length. In the language of regular expressions, we can write a NAF (from right-to-left) as

$$(1 + \bar{1} + \varepsilon)(0 + (01) + (0\bar{1}))^*.$$

Therefore, the generating function is (we have to reverse the direction by definition of the matrix product)

$$G^{(\mu)}(U, Z) = (1, 0, \dots, 0)(I - B^{(\mu)}(U, Z^2, Z))B(U, Z, 1)v,$$

where

$$v = (1, U, U, U, U, U, U, U, U)^T$$

takes care of the output which has to be written after the last input digit. We note that by chance, v is independent of μ .

Using a computer algebra program, we get

$$G^{(\mu)}(U, Z) = \frac{1 + 2UZ - Z^2 + 2UZ^2 - 4UZ^3 + 4U^2Z^3 - 4UZ^4 - 2U^2Z^5 + 2U^3Z^5}{(-1 + Z)(1 + Z)(-1 + Z + 4UZ^3)},$$

again independently of μ . We will therefore drop the index μ for the remainder of the analysis.

We now consider the random variable H_ℓ which we define to be the Hamming weight of the output of the transducers in Figures 7 and 8 (i.e., the number of elliptic curve summands) reading a random τ -NAF of length ℓ , where we consider equidistribution on the set of all τ -NAFs of length ℓ .

We note that the number of τ -NAFs equals the number of binary NAFs and can be calculated as

$$[Z^\ell] G(1, Z) = [Z^\ell] \frac{1 + 2Z}{1 - (Z + 2Z^2)} = \frac{4}{3}2^\ell - \frac{1}{3}(-1)^\ell = \frac{4}{3}2^\ell(1 + O(2^{-\ell})),$$

cf. for instance [6]. Therefore, each NAF of length ℓ occurs with probability $\frac{3}{4}2^{-\ell}(1 + O(2^{-\ell}))$.

Thus the expectation $\mathbb{E}(H_\ell)$ equals

$$\mathbb{E}(H_\ell) = \frac{1}{\frac{4}{3}2^\ell(1 + O(2^{-\ell}))} [Z^\ell] \left(\frac{\partial G(U, Z)}{\partial U} \right)_{U=1},$$

which can be calculated to be

$$\mathbb{E}(H_\ell) = \frac{\ell}{4} + \frac{53}{96} + O(2^{-\ell/2}) \approx 0.25\ell + 0.552083 + O(2^{-\ell/2}).$$

This should be compared with the average Hamming weight of NAF's of length ℓ , being

$$\frac{\ell}{3} + \frac{2}{9} + O(2^{-\ell}).$$

For computing the variance $\mathbb{V}(H_\ell)$, we calculate

$$\begin{aligned} \mathbb{E}(H_\ell(H_\ell - 1)) &= \frac{1}{\frac{4}{3}2^\ell(1 + O(2^{-\ell}))} [Z^\ell] \left(\frac{\partial^2 G(U, Z)}{\partial U^2} \right)_{U=1} \\ &= \frac{1}{16}\ell^2 + \frac{11}{192}\ell - \frac{23}{288} + O(2^{-\ell/2}). \end{aligned}$$

We conclude that

$$\begin{aligned} \mathbb{V}(H_\ell) &= \mathbb{E}(H_\ell(H_\ell - 1)) + \mathbb{E}(H_\ell) - (\mathbb{E}(H_\ell))^2 \\ &= \frac{\ell}{32} + \frac{1543}{9216} + O(\ell 2^{-\ell/2}) \approx 0.03125\ell + 0.167426 + O(\ell 2^{-\ell/2}). \end{aligned}$$

From Hwang's [7] "quasi-power theorem," we conclude that we have the following central limit theorem:

$$\lim_{\ell \rightarrow \infty} \mathbb{P}\left(H_\ell \leq \frac{\ell}{4} + h\frac{1}{8}\sqrt{2\ell}\right) = \frac{1}{\sqrt{2\pi}} \int_0^h e^{-t^2/2} dt.$$

We summarize our findings in the following theorem.

Theorem 4. *Let \mathbf{P} be a point on $E_a(\mathbb{F}_{2^n})$, $m \in \mathbb{Z}$ and \mathbf{s} a τ -NAF of m with ℓ digits. Then the transducers in Figures 7 and 8 compute an expansion $(\mathbf{s}^{(1)})_{\mathbf{s}^{(2)}}$ such that*

$$m\mathbf{P} = \text{value}_\tau(\mathbf{s}^{(1)})\mathbf{P} + \text{value}_\tau(\mathbf{s}^{(2)})\tau\left(\frac{1}{2}\mathbf{P}\right).$$

The right hand side can be written as a sum of H_ℓ summands. The random variable H_ℓ (where all τ -NAFs of length ℓ are considered to be equally likely) satisfies

$$\begin{aligned} \mathbb{E}(H_\ell) &= \frac{\ell}{4} + \frac{53}{96} + O(2^{-\ell/2}) \approx 0.25\ell + 0.552083 + O(2^{-\ell/2}), \\ \mathbb{V}(H_\ell) &= \frac{\ell}{32} + \frac{1543}{9216} + O(\ell 2^{-\ell/2}) \approx 0.03125\ell + 0.167426 + O(\ell 2^{-\ell/2}), \\ \lim_{\ell \rightarrow \infty} \mathbb{P}\left(H_\ell \leq \frac{\ell}{4} + h\frac{1}{8}\sqrt{2\ell}\right) &= \frac{1}{\sqrt{2\pi}} \int_0^h e^{-t^2/2} dt. \end{aligned}$$

4. REFINED ANALYSIS

Instead of considering the Hamming weight of the output of the algorithm, we produce here a refined analysis, by counting the “digits” in the output. The following four digits will be counted: $\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}$, $\begin{smallmatrix} \bar{1} \\ 0 \end{smallmatrix}$, $\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}$, $\begin{smallmatrix} 0 \\ \bar{1} \end{smallmatrix}$ (in that order). The following two tables give, for $\mu = -1$ resp. $\mu = 1$, the average and variances of these counts (the second line is always the floating point representation). This is easy to achieve from the transducer, by appropriately labelling the edges with auxiliary variables (instead of just one variable U as before).

		$\mu = -1$	
		Average	Variance
$\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}$	$\frac{\ell}{16} + \frac{1}{12} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{1193}{18432} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.08333333 + O(2^{-\ell/2})$
$\begin{smallmatrix} \bar{1} \\ 0 \end{smallmatrix}$	$\frac{\ell}{16} + \frac{1}{12} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{1193}{18432} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.08333333 + O(2^{-\ell/2})$
$\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}$	$\frac{\ell}{16} + \frac{37}{192} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{2413}{36864} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.192708 + O(2^{-\ell/2})$
$\begin{smallmatrix} 0 \\ \bar{1} \end{smallmatrix}$	$\frac{\ell}{16} + \frac{37}{192} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{2413}{36864} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.192708 + O(2^{-\ell/2})$

		$\mu = 1$	
		Average	Variance
$\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}$	$\frac{\ell}{16} + \frac{1}{12} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{1205}{18432} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.08333333 + O(2^{-\ell/2})$
$\begin{smallmatrix} \bar{1} \\ 0 \end{smallmatrix}$	$\frac{\ell}{16} + \frac{1}{12} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{1205}{18432} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.08333333 + O(2^{-\ell/2})$
$\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}$	$\frac{\ell}{16} + \frac{37}{192} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{5149}{36864} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.192708 + O(2^{-\ell/2})$
$\begin{smallmatrix} 0 \\ \bar{1} \end{smallmatrix}$	$\frac{\ell}{16} + \frac{37}{192} + O(2^{-\ell/2})$	$\frac{25\ell}{512} + \frac{5149}{36864} + O(\ell 2^{-\ell/2})$	$0.0625\ell + 0.192708 + O(2^{-\ell/2})$

REFERENCES

- [1] R. Avanzi, *A Note on the Signed Sliding Window Integer Recoding and a Left-to-Right Analogue*, Proceedings of Selected Areas in Cryptography 2004, Waterloo, ON, Canada, 9–10 August 2004, Lecture Notes in Comput. Sci., Springer-Verlag, Berlin, 2004.
- [2] R. Avanzi, M. Ciet, and F. Sica, *Faster scalar multiplication on Koblitz curves combining point halving with the Frobenius endomorphism*, Proceedings of Public Key Cryptography 2004, Singapore, March 1–4, 2004, Lecture Notes in Comput. Sci., vol. 2947, Springer, 2004, pp. 28–40.
- [3] M. Ciet, *Aspects of secure and efficient implementation of elliptic curve cryptosystems*, Ph.D. thesis, Université Catholique Louvain-la-Neuve, 2003.
- [4] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver, *Combinatorial optimization*, Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons Inc., New York, 1998.
- [5] C. Heuberger and H. Prodinger, *Analysis of alternative digit sets for nonadjacent representations*, Preprint available at <http://www.opt.math.tu-graz.ac.at/~cheub/publications/dnaf-1.pdf>.

- [6] ———, *Carry propagation in signed digit representations*, European J. Combin. **24** (2003), 293–320.
- [7] H.-K. Hwang, *On convergence rates in the central limit theorems for combinatorial structures*, European J. Combin. **19** (1998), 329–343.
- [8] E. W. Knudsen, *Elliptic Scalar Multiplication Using Point Halving*, Advances in Cryptology – Asia-crypt’99, Lecture Notes in Comput. Sci., vol. 1716, Springer-Verlag, Berlin, 1999, pp. 135–149.
- [9] N. Koblitz, *Elliptic curve cryptosystems*, Math. Comp. **48** (1987), no. 177, 203–209.
- [10] N. Koblitz, *CM-curves with good cryptographic properties*, Advances in cryptology—CRYPTO ’91 (Santa Barbara, CA, 1991), Lecture Notes in Comput. Sci., vol. 576, Springer, Berlin, 1992, pp. 279–287.
- [11] V. S. Miller, *Use of elliptic curves in cryptography*, Advances in cryptology – CRYPTO ’85, Lecture Notes in Comput. Sci., vol. 218, Springer-Verlag, Berlin, 1986, pp. 417–426.
- [12] A. Miyaji, T. Ono, and H. Cohen, *Efficient elliptic curve exponentiation*, Information and communications security. 1st international conference, ICICS ’97, Beijing, China, November 11–14, 1997. Proceedings (Y. et al. Han, ed.), LNCS, vol. 1334, Springer-Verlag, 1997, pp. 282–290.
- [13] J. A. Muir and D. R. Stinson, *Minimality and other properties of the width- w nonadjacent form*, Tech. Report CORR 2004-08, Centre for Applied Cryptographic Research, 2004, available at <http://www.cacr.math.uwaterloo.ca/techreports/2004/>.
- [14] ———, *New minimal weight representations for left-to-right window methods*, Tech. Report CACR 2004-03, Centre for Applied Cryptographic Research, 2004, available at <http://www.cacr.math.uwaterloo.ca/techreports/2004/>.
- [15] G. W. Reitwiesner, *Binary arithmetic*, Advances in computers, vol. 1, Academic Press, New York, 1960, pp. 231–308.
- [16] R. Schroepel, *Elliptic curve point ambiguity resolution apparatus and method*, International Application Number PCT/US00/31014, filed 9 November 2000.
- [17] J. A. Solinas, *An improved algorithm for arithmetic on a family of elliptic curves*, Advances in Cryptology - CRYPTO ’97. 17th annual international cryptology conference. Santa Barbara, CA, USA. August 17-21, 1997. Proceedings (B. S. jun. Kaliski, ed.), LNCS, vol. 1294, Springer, Berlin, 1997, pp. 357–371.
- [18] ———, *Efficient arithmetic on Koblitz curves*, Des. Codes Cryptogr. **19** (2000), no. 2-3, 195–249, Towards a quarter-century of public key cryptography.

APPENDIX. ILLUSTRATIONS

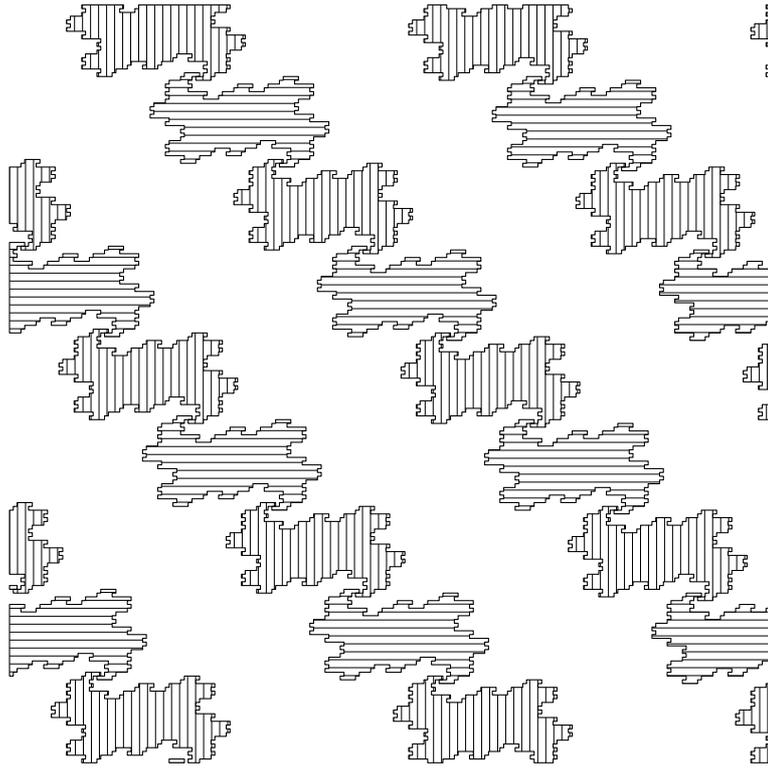
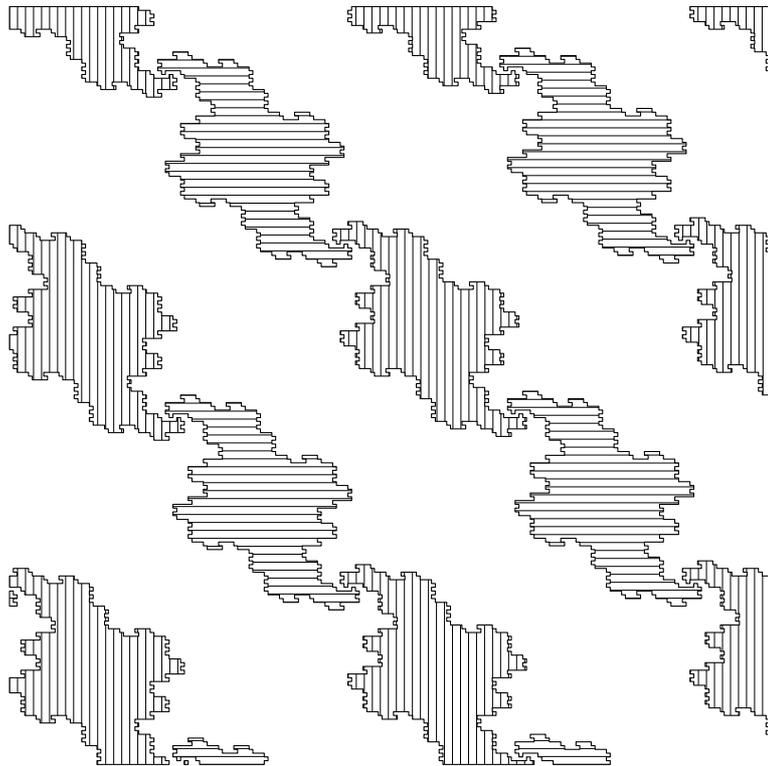
In this appendix, we illustrate a few properties of the τ -NAF, for the reader’s benefit. We use the following representations of (double-)digits by colors:

$\frac{0}{1}$	“Magenta”	
$-1, \frac{1}{0}$	“Blue”	
$1, \frac{1}{0}$	“Red”	
$\frac{0}{1}$	“Green”	

Figure 10 describes all numbers $a + b\tau$, for $0 \leq a, b < 200$ and $\mu = -1$, scaled into a square, and the digit with index 10 (the rightmost digit has index 0).

Figure 11 describes all numbers $a + b\tau$, for $0 \leq a, b < 200$ and $\mu = 1$, scaled into a square, and the digit with index 11.

Figure 12 describes all numbers $a + b\tau$, for $0 \leq a, b < 200$ and $\mu = -1$, scaled into a square, and the double digit of the wide-double-NAF with index 10.

FIGURE 10. τ -NAF, $\mu = -1$, $k = 10$, $M = 200$ FIGURE 11. NAF, $\mu = +1$, $k = 11$, $M = 200$

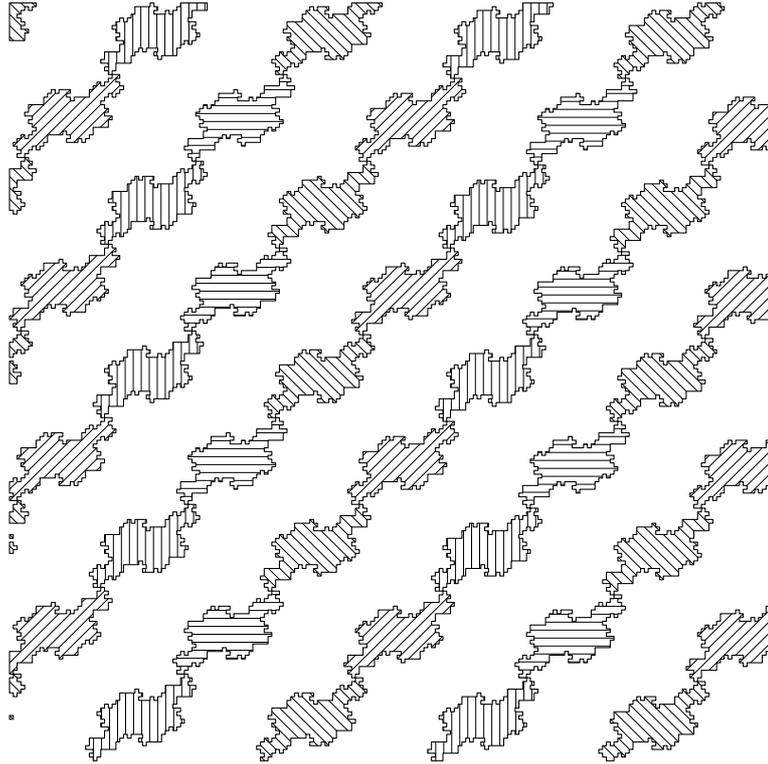


FIGURE 12. Wide-Double-NAF, $\mu = -1$, $k = 10$, $M = 200$

Figure 13 describes all numbers $a + b\tau$, for $0 \leq a, b < 200$ and $\mu = 1$, scaled into a square, and the double digit of the wide-double-NAF with index 11.

Another interesting construction when dealing with number systems, is the *odometer*. This is a transducer that describes how to add (or subtract) 1. We will give these odometers for the τ -NAF's with $\mu = \pm 1$ in the Figures 14 and 15. The meaning of the states is as in Figures 1 and 2. Note that there are two states labelled 1 (and $\bar{1}$), since the transducer checks that the input is indeed a nonadjacent form.

(R. Avanzi) INSTITUTE FOR EXPERIMENTAL MATHEMATICS (IEM) – UNIVERSITÄT DUISBURG–ESSEN, ELLERNSTRASSE 29, D-45326 ESSEN, GERMANY, COMMUNICATION SECURITY (COSY) – ELECTRICAL ENGINEERING AND INFORMATION TECHNOLOGY, RUHR-UNIVERSITÄT BOCHUM, UNIVERSITÄTSSTRASSE 150, D-44780 BOCHUM, GERMANY

E-mail address: mocenigo@exp-math.uni-essen.de

(C. Heuberger) INSTITUT FÜR MATHEMATIK B, TECHNISCHE UNIVERSITÄT GRAZ, STEYRERGASSE 30, 8010 GRAZ, AUSTRIA

E-mail address: clemens.heuberger@tugraz.at

(H. Prodinger) THE JOHN KNOPFMACHER CENTRE FOR APPLICABLE ANALYSIS AND NUMBER THEORY, SCHOOL OF MATHEMATICS, UNIVERSITY OF THE WITWATERSRAND, P. O. WITS, 2050 JOHANNESBURG, SOUTH AFRICA

E-mail address: helmut@maths.wits.ac.za

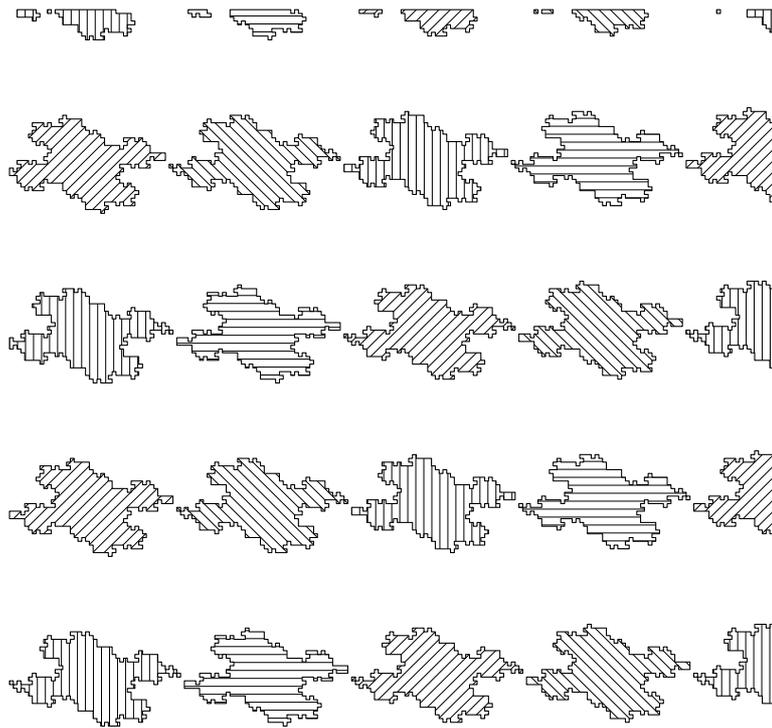


FIGURE 13. Wide-Double-NAF, $\mu = +1$, $k = 11$, $M = 200$

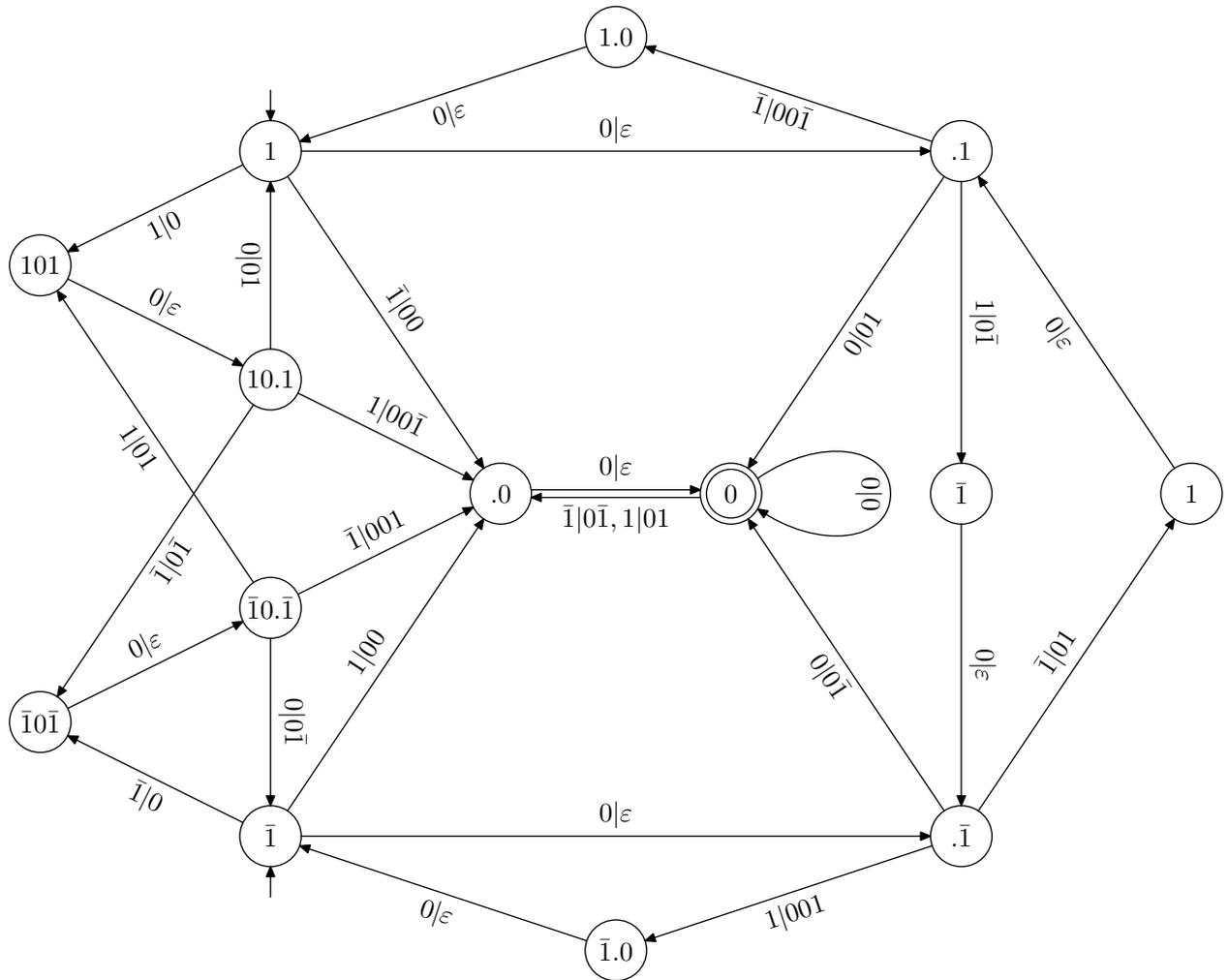


FIGURE 14. Addition of ± 1 for NAFs with $\mu = -1$. Use initial state 1 for addition of 1 and initial state $\bar{1}$ for addition of -1 .

