Algorithmen & mathematische Modellierung

Forschungsschwerpunkt

Algorithmen und mathematische Modellierung

Graz University of Technology

Optimality of Digital Expansions to the Base of the Frobenius Endomorphism on Koblitz Curves in Characteristic Three

Markus Kröll

Project Area(s):

Analysis of Digital Expansions with Applications in Cryptography

Institut für Optimierung und Diskrete Mathematik (Math B)

Report 2010-9, August 2010

Optimality of Digital Expansions to the Base of the Frobenius Endomorphism on Koblitz Curves in Characteristic Three

Markus Kröll

August 11, 2010

Contents

1 Introduction **2**

2 τ-adic Expansions **3**
2.1 Background 3
2.2 Properties of τ 4
2.3 Digit Sets 5
2.4 Optimality 9
2.5 Examples for Optimality 11

3 The Transducer **11**
3.1 Basic Idea of the Transducer 11
3.2 Buildup in detail 15
3.3 Correctness of the Transducer 18

4 Fast algorithmic realization of the Transducer **19**
4.1 Reducing the graph to a sixth part 19
4.2 Adjustment of the Bellman-Ford algorithm 21

5 Arithmetic in $\mathbb{Z}[\tau]$ **24**
5.1 Reducing to elements in $\mathcal{D}_w$ 24
5.2 Needful Operations 27

6 Results **30**

1 Introduction

Since both Koblitz [7] and Miller [9] suggested the use of elliptic curve cryptography in 1985, it has become a more and more popular cryptosystem. Elliptic curve cryptography is based on the algebraic structure of elliptic curves over finite fields. One can introduce an addition law for two points on the elliptic curve $\mathcal{E}$ in a certain way. By appending the set of points of an elliptic curve $\mathcal{E}(\mathbb{F})$ by a special point which represents the point 'at infinity', $\mathcal{E}(\mathbb{F})$ turns into an additive abelian group. Because an elliptic curve $\mathcal{E}$ over a finite field has finitely many points, the discrete logarithm problem (DLP) can be applied to the points on $\mathcal{E}$. The discrete logarithm problem is to find a number $n \in \mathbb{N}$ for given points P and Q on $\mathcal{E}(\mathbb{F})$, such that

$$nP = Q.$$

Because it is believed that there is no fast algorithm for solving the DLP, there are a few cryptosystems which rely on the problem of finding a number n as above. For both encryption and decryption in such systems many scalar multiplications of points on the elliptic curve are performed.

Based on the goal of making the usage of elliptic curve cryptosystems as efficient as possible, there have been several attempts in accelerating the computation of group operations on the group of points of an elliptic curve over a finite field. Koblitz showed in [8] that the elliptic curve

$$\mathcal{E}_{3,\mu} : Y^2 = X^3 - X - \mu, \qquad \text{with } \mu \in \{-1, 1\}$$

defined over $\mathbb{F}_3$ offers an alternative approach for scalar multiplication of its points, by taking a special digit expansion of the scalar. The approach of Avanzi, Heuberger and Prodinger [1] is motivated by the work of Koblitz. By using a τ-adic expansion of the scalar, where τ is a root of the characteristic polynomial of the Frobenius endomorphism, a Horner scheme is used to make a scalar multiplication. In this thesis, I present a method how to show the optimality of a τ-adic expansion under certain aspects.

First of all, the Frobenius Endomorphism τ is introduced, which maps points on an elliptic curve. This endomorphism can be identified with a complex number, which will also be denoted by τ. It is shown that with a τ-adic digital expansion the scalar multiplication on elliptic curves can be performed in an efficient way when choosing an adequate digit set.

Next a digit set $\mathcal{D}_w$ is introduced. It is shown that every integer admits a $\mathcal{D}_w$-w-NAF, which is a digital expansion with only one non-zero element in every factor of length w. To determine whether the set $\mathcal{D}_w$ is a proper digit set for the goal of fast scalar multiplication, optimality has to be defined. There is also an example of non-optimality when introducing τ on an elliptic curve in characteristic 2.

To prove optimality, it is shown that the buildup of a transducer $\mathscr{T}$ is needed, where a transducer is a finite automaton with one input and one output tape. Based on empirical observations, such a transducer grows very fast for growing w. Because of that, two ways are introduced for a fast algorithmic implementation. On the one hand, the transducer can be reduced to a sixth part, on the other hand, the Bellman-Ford algorithm for finding shortest paths can be modified to achieve better results.

To make evaluations in $\mathbb{Z}[\tau]$, it is crucial to know facts about the arithmetic in this ring. In chapter 5, efficient ways for reducing elements $a+b\tau$ in $\mathbb{Z}[\tau]$ to elements in $\mathcal{D}_w$ or to reduce elements $\alpha_k\tau^k + \alpha_{k-1}\tau^{k-1} + \cdots + \alpha_1\tau + \alpha_0$ to their shortest possible form are presented.

In the end numerical results and the set of w for which it was proven that the $\mathcal{D}_w$-w-NAF is optimal are presented.

2 τ-adic Expansions

2.1 Background

Let $\mathcal{E}_{3,\mu}(\mathbb{F}_{3^m})$ be the group of points of the elliptic curve $\mathcal{E}_{3,\mu}$, where m is a natural number coprime to 6. The Frobenius endomorphism

$$\varphi : \mathbb{F}_{3^m} \to \mathbb{F}_{3^m}, \quad x \mapsto x^3$$

is also a function on the points of the elliptic curve

$$\varphi : \mathcal{E}_{3,\mu}(\mathbb{F}_{3^m}) \to \mathcal{E}_{3,\mu}(\mathbb{F}_{3^m}), \quad (x,y) \mapsto (x^3, y^3), \text{ with } (x,y) = P \in \mathcal{E}_{3,\mu}(\mathbb{F}_{3^m}), \tag{2.1}$$

One can show that φ satisfies the relation

$$\begin{aligned} & \varphi(\varphi(P)) - 3\mu\varphi(P) + 3P = 0 \\ \iff \quad & (\varphi^2 - 3\mu\varphi + 3\mathsf{id})(P) = 0 \quad \text{for every } P \in \mathcal{E}_{3,\mu}(\mathbb{F}_{3^m}), \end{aligned}$$

and thus it follows that

$$\varphi^2 - 3\mu\varphi + 3\mathsf{id} = 0 \quad . \tag{2.2}$$

The key step in creating an alternative way for scalar multiplication is to introduce the complex number $\tau \in \mathbb{C}$ with the same minimal polynomial as the one of φ in (2.2):

$$\tau^2 - 3\mu\tau + 3 = 0 \quad \text{leads to} \quad \tau = \frac{3\mu + \sqrt{-3}}{2} = \sqrt{-3}\frac{1 - \mu\sqrt{-3}}{2} \quad .$$

By defining a group action from $\mathbb{Z}[\tau]$ to $\mathcal{E}_{3,\mu}(\mathbb{F}_{3^m})$ where

$$(a + b\tau)P = aP + b\varphi(P), \qquad a, b \in \mathbb{Z},$$

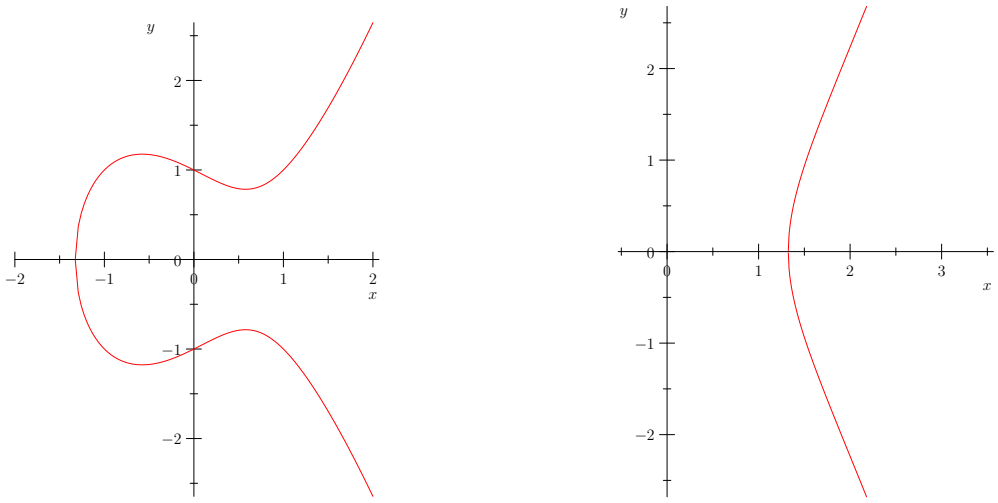
the Frobenius endomorphism φ can be identified with the imaginary quadratic number τ.

If now an integer z can be written in a τ-adic expansion

$$z = \sum_{j=0}^{l} a_j \tau^j, \qquad \text{where } a_j \in \mathcal{D} \subseteq \mathbb{Z}[\tau] \text{ and } l \in \mathbb{N}, \tag{2.3}$$

the scalar multiplication of z with a point $P \in \mathcal{E}_{3,\mu}(\mathbb{F}_{3^m})$ can be evaluated by

$$zP = (\sum_{j=0}^{l} a_j \tau^j)P = \sum_{j=0}^{l} a_j \varphi^j(P) = \varphi(\ldots(\varphi(a_l P) + a_{l-1}P) + \ldots).$$



(a) $\mu = -1$ (b) $\mu = 1$

Figure 1: The elliptic curve $\mathcal{E} : Y^2 = X^3 - X - \mu$ over $\mathbb{R}$

The right hand side of the equation is the Horner scheme of the sum, which is used for efficiency reasons. Making a first rough breakdown of the time costs of the scalar multiplication in this way, it can be stated that the Frobenius endomorphism has to be applied l times, which is a cheap operation (see (2.1)). The other operation that has to be done is to add points on the elliptic curve $H(a_l a_{l-1} \dots a_1 a_0)$ times, where H is the Hamming weight function, which is defined as

$$H(a_l a_{l-1} \dots a_1 a_0) := \#\{j \in \mathbb{N} : a_j \neq 0\} \quad .$$

Here we can see the basic approach in improving the performance of scalar multiplication on an elliptic curve. In contrary to the double-and-add algorithm for example, where $\mathcal{O}(\log z)$ additions are needed to evaluate zP, the number of additions to be done depends on the number of non-zero digits a_j in the expansion (2.3), and thus on the digit set $\mathcal{D} \subseteq \mathbb{Z}[\tau]$ that is chosen.

2.2 Properties of τ

First of all some basic properties of τ have to be stated.
Let $\zeta \in \mathbb{C}$ be the complex number with

$$\zeta := \frac{1 - \mu\sqrt{-3}}{2} \quad . \tag{2.4}$$

ζ is also an element of $\mathbb{Z}[\tau]$ and a primitive sixth root of unity, because

$$\frac{1 - \mu\sqrt{-3}}{2} = 2 - \frac{-3 - \mu\sqrt{-3}}{2} = 2 - \mu\tau \quad .$$

A multiplication ζz with $z \in \mathbb{Z}[\tau]$ corresponds to a rotation of the point z in the complex plane by $\frac{\pi}{3}$, where the direction is positive with negative μ and negative with $\mu = 1$. Because every $z \in \mathbb{Z}[\tau]$ is still in $\mathbb{Z}[\tau]$ after such a rotation, it leaves $\mathbb{Z}[\tau]$ globally invariant. Directly

from the definition of ζ in (2.4), it follows that ζ is related to τ in the following way:

$$\tau = \sqrt{-3}\frac{1-\mu\sqrt{-3}}{2} = \sqrt{-3}\zeta$$

Moreover if we denote the complex conjugate of τ by $\bar{\tau}$, $\bar{\tau} = \zeta\tau$ is imperative. Thus both ζ and $\bar{\tau}$ are beneficial, because it can be shown that they correspond to the functions on the endomorphism ring of $\mathcal{E}_{3,\mu}$

$$\begin{aligned} \zeta &: (x,y) \mapsto (x+\mu, -y) \quad \text{and} \\ \bar{\tau} &: (x,y) \mapsto (x^3+\mu, -y^3) \end{aligned}$$

and so these operations can be computed efficiently. One can evaluate that $\tau\bar{\tau} = 3$ holds. Another fact about τ has to be stated in the following proposition, which appears in [1]:

Proposition 2.1. *Let $z = a + b\tau$ be an element of $\mathbb{Z}[\tau]$, where a and b are rational integers. Then $\tau \mid z$ if and only if $3 \mid a$.*

Proof.

"$\Longrightarrow$" Let z by divisible by τ. Then there are some integers $\bar{a}$,$\bar{b}$ with

$$a + b\tau = z = (\bar{a} + \bar{b}\tau)\tau = \bar{a}\tau + \bar{b}\tau^2 = \bar{a}\tau + 3\mu\bar{b}\tau - 3\bar{b}.$$

Because

$$a + b\tau = -3\bar{b} + \left(\bar{a} + 3\mu\bar{b}\right)\tau$$

the integer a has to be divisible by 3.

"$\Longleftarrow$" Let $a = 3\alpha$ for an integer α. Then

$$a + b\tau = 3\alpha + b\tau = \left(3\mu\tau - \tau^2\right)\alpha + b\tau = \tau(3\mu\alpha - \tau\alpha + b).$$

□

2.3 Digit Sets

The number of non-zero digits in τ-adic digit expansions depends on the chosen digit set $\mathcal{D} \subset \mathbb{Z}[\tau]$. To make further investigations, the following definitions have to be made:

Definition 2.1. Let $\mathcal{D}$ be a finite subset of $\mathbb{Z}[\tau]$ and $w \in \mathbb{N}$. A word $\eta_{l-1}\eta_{l-2}\ldots\eta_0 \in \mathcal{D}^*$ is called a $\mathcal{D}$-w-NAF of a $z \in \mathbb{Z}[\tau]$, if

(1) $\mathsf{value}(\eta_{l-1}\ldots\eta_0) := \sum_{j=0}^{l-1} \eta_j\tau^j = z$

(2) Each factor $\eta_{j+w-1}\ldots\eta_j$ contains at most one non-zero.

$\mathcal{D}$ is called a w-Non-Adjacent-Digit-Set (or simply w-NADS), if every integer $z \in \mathbb{Z}[\tau]$ admits a $\mathcal{D}$-w-NAF.

Definition 2.2. A reduced residue system modulo τ^w is a set containing exactly one representative for each residue class of $\mathbb{Z}[\tau]$ modulo τ^w that is not divisible by τ.

If the digit set $\mathcal{D}$ consists of 0 and a reduced residue system modulo τ^w, each $z \in \mathbb{Z}[\tau]$ is either divisible by τ or congruent modulo τ^w to exactly one element of the digit set, because τ is a prime element in $\mathbb{Z}[\tau]$. So if there is a $\mathcal{D}$-w-NAF for z, it is uniquely determined. Further Solinas [12, 13], Blake, Kumar Murty and Xu [2] show that a digit set, which consists of 0 and so-called representatives of minimal norm from each residue class modulo τ^w which are not divisible by τ, is a w-NADS.

Definition 2.3. Let $\eta \in \mathbb{Z}[\tau]$ be not divisible by τ, $w \in \mathbb{N}$ and assume that

$$|\eta| \leq |\lambda| \text{ for all } \lambda \in \mathbb{Z}[\tau] \text{ with } \lambda \equiv \eta \mod \tau^w.$$

Then η is called a representative of minimum norm of its residue class modulo τ^w.

The following theorem from [1] gives an explicit description of a digit set $\mathcal{D}_w$, which consists of 0 and one representative of minimal norm from each residue class modulo τ^w which is not divisible by τ.

Theorem 2.1. *Let $w \geq 2$ and set*

$$\mathcal{D}_{w,0} = \left\{ a + b\mu\tau : a \in \mathbb{Z}, b \in \mathbb{Z}, 3 \nmid a, 1 \leq a \leq 3^{w/2} - 2 \text{ and } -\frac{a}{3} < b < 3^{w/2-1} - \frac{2a}{3} \right\} \quad (2.5)$$

if w is even and

$$\begin{aligned} \mathcal{D}_{w,0} = \left\{ a + b\mu\tau : a \in \mathbb{Z}, b \in \mathbb{Z}, 3 \nmid a, -3^{\lfloor \frac{w}{2} \rfloor} + 2 \leq b \leq 0, 1 - 2b \leq a \leq 3^{\lfloor \frac{w}{2} \rfloor} - b - 1 \right\} \quad (2.6) \\ \cup \left\{ 3^{\lfloor \frac{w}{2} \rfloor} - b + b\mu\tau : b \in \mathbb{Z}, 3 \nmid b, -\frac{3^{\lfloor \frac{w}{2} \rfloor} - 1}{2} \leq b \leq 0 \right\} \end{aligned}$$

if w is odd. Set

$$\mathcal{D}_w := \{0\} \cup \bigcup_{0 \leq k \leq 5} \zeta^k \mathcal{D}_{w,0} \quad .$$

Then $\mathcal{D}_w$ consists of 0 and exactly one representative of minimum norm of every residue class modulo τ^w. Moreover, $\mathcal{D}_w$ is a w-NADS.

Proof. In [1] a proof is given, but the set $\mathcal{D}_{w,0}$ is not derived in detail for the case of an odd w. It is shown that

$$\mathcal{D}_{w,0} = \begin{cases} \{\alpha \in \mathbb{Z}[\tau] \cap 3^{w/2}\widetilde{V}_0 : \tau \nmid \alpha\} & \text{if } w \text{ is even,} \\ \{\alpha \in \mathbb{Z}[\tau] \cap \mu i 3^{w/2}\zeta^2\widetilde{V}_0 : \tau \nmid \alpha\} & \text{if } w \text{ is odd.} \end{cases}$$

where

$$\begin{aligned} 3^{w/2}\widetilde{V}_0 = \left\{ x + \sqrt{-3} : 0 < x < \frac{3^{w/2}}{2} \text{ and } -\frac{x}{3} < \mu y \leq \frac{x}{3} \right\} \cup \\ \cup \left\{ \frac{3^{w/2}}{2} + \sqrt{-3}y : 0 < \mu y \leq \frac{3^{w/2-1}}{2} \right\}. \end{aligned}$$

The second set in the union corresponds to points on the outer boundary of $3^{w/2}\widetilde{V}$. The cell $\widetilde{V}_0$ for $\mu = 1$, which is the plot of the set $\widetilde{V}_0$ in the complex plane, is given by

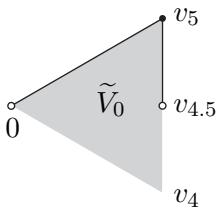


Figure 2: The cell $\widetilde{V}_0$ for $\mu = 1$

where $v_0 = \frac{\sqrt{-3}}{3}$, $v_k = v_0\zeta^{-\mu k}$ and $v_{k+1/2}$ lies on the line from v_k to v_{k+1}. For the odd case, a rotation induced by the factor $\mu i\zeta^2$ is needed. To simplify matters, we assume that $\mu = 1$ (the other case can be treated analogously). We obtain the rotated points $\tilde{v}_4$ and $\tilde{v}_5$ with

$$\tilde{v}_4 = v_4 i\zeta^2 = \frac{\sqrt{3}}{6} - \frac{1}{2}i \quad \text{and} \quad \tilde{v}_5 = v_5 i\zeta^2 = \frac{1}{\sqrt{3}}$$

and with a multiplication by $3^{w/2}$

$$3^{w/2}\tilde{v}_4 = \frac{3^{w/2}\sqrt{3}}{6} - \frac{3^{w/2}}{2}i \quad \text{and} \quad 3^{w/2}\tilde{v}_5 = \sqrt{3}3^{w/2-1} = 3^{\lfloor w/2 \rfloor}.$$

This leads to the cells $\widetilde{V}_0'$ for positive μ and $\widetilde{V}_0''$ for negative μ.

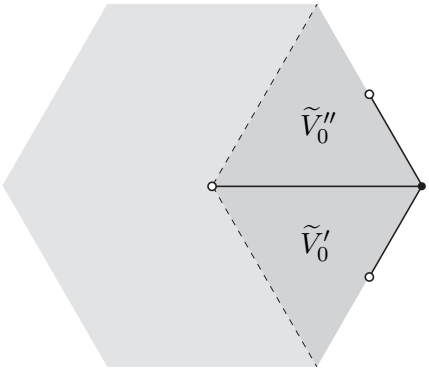


Figure 3: Rotated cells for $\mu \in \{-1, 1\}$

Because the set $\widetilde{V}_0'$ is bounded by an equilateral triangle, where the endpoint of one side is the origin and this side is congruent with the axis of the real numbers, $3^{w/2}\widetilde{V}_0'$ without the outer boundary is given by

$$\left\{ x + iy : -\frac{3^{w/2}}{2} < y \leq 0 \text{ and } -\frac{\sqrt{3}}{3}y < x < \frac{\sqrt{3}}{3}y + 3^{\lfloor w/2 \rfloor} \right\}. \tag{2.7}$$

Because

$$a + b\tau = \left(a + \frac{3}{2}b\right) + i\left(\frac{\sqrt{3}}{2}b\right),$$

we can plug in $a + \frac{3}{2}b$ for x and $\frac{\sqrt{3}}{2}b$ for y. This leads to

$$-\frac{3^{w/2}}{2} < \frac{\sqrt{3}}{2}b \leq 0 \quad \Longleftrightarrow \quad -3^{\lfloor w/2 \rfloor} < b \leq 0 \quad \Longleftrightarrow \quad -3^{\lfloor w/2 \rfloor} + 1 \leq b \leq 0$$

for the first inequality in the set given in (2.7) and

$$\begin{aligned} & -\frac{\frac{\sqrt{3}}{2}b\sqrt{3}}{3} < a + \frac{3}{2}b < \frac{\frac{\sqrt{3}}{2}b\sqrt{3}}{3} + 3^{\lfloor w/2 \rfloor} \quad \Longleftrightarrow \quad -\frac{b}{2} < a + \frac{3}{2}b < \frac{b}{2} + 3^{\lfloor w/2 \rfloor} \\ \Longleftrightarrow \quad & -2b < a < 3^{\lfloor w/2 \rfloor} - b \quad \Longleftrightarrow \quad 1 - 2b \leq a \leq 3^{\lfloor w/2 \rfloor} - b - 1 \end{aligned}$$

for the second one. There are no possible values for a if $b = -3^{\lfloor w/2 \rfloor} + 1$, so the first set in the union of both $3^{w/2}\widetilde{V}_0'$ and the set $\mathcal{D}_{w,0}$ in (2.6) are the same. Because a is bounded to the right by $3^{\lfloor w/2 \rfloor} - b$ and $-3^{\lfloor w/2 \rfloor} + 1 \leq b \leq 0$, the line from $\tilde{v}_5$ to $\tilde{v}_{4.5}$, without the point $\tilde{v}_{4.5}$, is given by

$$\left\{ 3^{\lfloor \frac{w}{2} \rfloor} - b + b\mu\tau : b \in \mathbb{Z}, 3 \nmid b, -\frac{3^{\lfloor \frac{w}{2} \rfloor} - 1}{2} \leq b \leq 0 \right\}.$$

□

Example 2.1. Set $w = 4$. Then

$$\mathcal{D}_{w,0} = \{-1 + 2\tau, -2 + 3\tau, 1 + 3\tau, 1, 1 + \tau, 1 + 2\tau, 2, 2 + \tau, 4\} \quad .$$

For further investigation, it is very useful to know how many elements a certain set $\mathcal{D}_w$ contains.

Theorem 2.2. *Let w be a natural number greater or equal 2. Then $|\mathcal{D}_w| = 6 \cdot 3^{w-2} + 1$.*

Proof. There is a direct way for proving the theorem above:
The set $\mathcal{D}_w \backslash \{0\}$ evolves of the set $\mathcal{D}_{w,0}$ defined in (2.5) for even and in (2.6) for odd w, by making the union of all sets $\zeta^k \mathcal{D}_{w,0}$ for $k \in \{0, \ldots, 5\}$. Because of the construction of $\mathcal{D}_{w,0}$ there is an $\alpha \in \left[0, \frac{5\pi}{3}\right)$ so that for every element $z \in \mathcal{D}_{w,0}$ the argument of z is in $\left[\alpha, \alpha + \frac{\pi}{3}\right)$. Thus it is only needed to show that $|\mathcal{D}_{w,0} \backslash \{0\}| = 3^{w-2}$, because the sets $\zeta^k \mathcal{D}_{w,0}$ are disjoint for distinct values of k. Let w be even. For an element $a + b\tau \in \mathcal{D}_{w,0}$, where a and b are both in $\mathbb{Z}$, the values of a are in $\{1 \leq a \leq 3^{\frac{w}{2}} - 2 : 3 \nmid a\} := A$. For every $a \in A$ a possible b is bounded by

$$-\frac{a}{3} < b < 3^{\frac{w}{2}-1} - \frac{2a}{3} \quad \Longleftrightarrow \quad -a < 3b < 3^{\frac{w}{2}} - 2a.$$

For $a = 1$, there are $3^{\frac{w}{2}-1}$ different values for b and for $a = 2$ there are $3^{\frac{w}{2}-1} - 1$ values. Let K be the number of values for b for a fixed $a = 3n + 1$. For $a = 3n + 2$, the inequality is

$$-3n - 2 < 3b < 3^{\frac{w}{2}} - 6n - 3 - 1,$$

so the number of values for b is $K - 1$. The next value for a is $3n + 4$. Because the inequality is

$$-3(n + 1) - 1 < 3b < 3^{\frac{w}{2}} - 6n - 6 - 2,$$

the number of values for b is also $K - 1$. Because there is a $m \in \mathbb{N}$ with $3n + 4 = 3m + 1$, the number of values for b, starting with $a = 3n + 1$, are

$$K, K - 1, K - 1, K - 2, K - 2, K - 3, K - 3, \ldots$$

For both $a = 3^{\frac{w}{2}} - 2$ and $a = 3^{\frac{w}{2}} - 1$ there is exactly one value for b which satisfies the inequality, so the number of values for $a + b\tau$ is

$$3^{\frac{w}{2}-1} + 2(3^{\frac{w}{2}-1} - 1) + 2(3^{\frac{w}{2}-1} - 2) + \cdots + 4 + 2 =$$
$$= 3^{\frac{w}{2}-1} + 2\frac{(3^{\frac{w}{2}-1} - 1)3^{\frac{w}{2}-1}}{2} = 3^{w-2} \quad .$$

Now let w be odd. The two sets in (2.6) have to be treated separately. The number of elements in the first set, counted in the way above is

$$2 \cdot 3^{\lfloor w/2 \rfloor - 1} + 6 \sum_{j=1}^{3^{k-1}} j = 2 \cdot 3^{\lfloor w/2 \rfloor - 1} + 3 \cdot 3^{\lfloor w/2 \rfloor - 1}(3^{\lfloor w \rfloor/2-1} - 1) =$$
$$= 3^{2\lfloor w/2 \rfloor - 1} - 3^{\lfloor w/2 \rfloor - 1} = 3^{w-2} - 3^{\lfloor w/2 \rfloor - 1}.$$

The second set in (2.6) has $3^{\lfloor w/2 \rfloor - 1}$ elements, so the theorem is also true for odd w. □

2.4 Optimality

The question is, if the $\mathcal{D}_w$-w-NAF is the best choice of a digit expansion with the given digit set $\mathcal{D}_w$. The goal to achieve is to minimize the number of non-zero digits in an expansion. To formalize this, we define the following:

Definition 2.4. A $\mathcal{D}$-w-NAF is called optimal, when for every word $\mathcal{W} = \eta_k \eta_{k-1} \dots \eta_1 \eta_0$ which is a $\mathcal{D}$-w-NAF with $z = \mathsf{value}(\mathcal{W})$, $\mathcal{W}$ has minimum Hamming weight among all words with digits in $\mathcal{D}_w$ and value z.

Here is an example of two such words:

Example 2.2. Suppose that $w = 3$ and set $\mathcal{W} = (-2)(-1)(4 - \tau)$. Then

$$\mathsf{value}(\mathcal{W}) = 10 - 8\tau = \mathsf{value}((-1)00(1 - 2\tau)) \quad .$$

Because $\mathcal{W}$ has 3 non-zero digits, and the equivalent $\mathcal{D}$-3-NAF has only 2, this is not a counterexample for the optimality. In fact, there is none, because it can be shown that the $\mathcal{D}$-3-NAF is optimal.

Although there is no proof for the optimality for all possible $w \in \mathbb{N}$ known yet, there is a way in proving the optimality of a fixed w once at a time, by the use of extensive computations. For this, the following definitions have to be made.

Definition 2.5. A finite state transducer is a finite state machine with one input and one output tape.

To simplify matters, a finite state transducer is called transducer in this work. Moreover, the state transitioning function of a transducer is seen as the edges between states.
Another definition is needed:

Definition 2.6. Let $\mathscr{T}$ be a transducer, where every edge from a state S_i to a state S_j has a unique pair (α, β), where α is the digit read when going from state S_i to state S_j, and β is the word written. Then a graph $\mathrm{G}(\mathscr{T})$ is called the related graph to $\mathscr{T}$, if the states of $\mathscr{T}$ are

the nodes of $\mathrm{G}(\mathscr{T})$ and the edges are those of the transducer. Each edge (S_i, S_j) has weight w, where

$$w = H(\alpha) - H(\beta),$$

and H is the Hamming weight function.

Now we have everything we need for the following theorem. It shows how to test the optimality of a $\mathcal{D}$-w-NAF for a certain $w \in \mathbb{N}$ in an intuitively easy accessible way:

Theorem 2.3. *Let $\mathscr{T}$ be a transducer, transforming any word with digits in $\mathcal{D}_w$ into a $\mathcal{D}$-w-NAF, with related graph $G(\mathscr{T})$. Then*

The $\mathcal{D}$-w-NAF is optimal $\iff$ *All paths from the start to the accept state in $G(\mathscr{T})$ have non-negative weight.*

Proof. A $\mathcal{D}$-w-NAF is optimal for a certain w, when it has the smallest Hamming weight among all possible digit expansions with digits in $\mathcal{D}_w$. This is equivalent to the property that the difference $H(A) - H(B)$ is greater or equal 0, where A is any word in $\mathcal{D}_w^*$, and B is the $\mathcal{D}$-w-NAF of the value of A. When translating a word $\mathcal{W} \in \mathcal{D}_w^*$ with $\mathscr{T}$ into $\tilde{\mathcal{W}}$, the finite automaton runs along a certain path (the transition function of $\mathscr{T}$ corresponds to edges in $G(\mathscr{T})$). The sum of the weights related to this edges in $G(\mathscr{T})$ is

$$\sum_{\text{digits } \lambda \in \mathcal{W}} [H(\lambda) - H(\tilde{\lambda})], \quad \text{where } \tilde{\lambda} := \text{digit-block written by } \mathscr{T} \text{ when } \lambda \text{ is read.}$$

Because the output finally written by $\mathscr{T}$ (without overwriting something) is $\tilde{\mathcal{W}}$, the sum above is equal

$$\sum_{\text{digits } \lambda \in \mathcal{W}} H(\lambda) - \sum_{\text{blocks } \tilde{\lambda} \in \mathcal{W}} H(\tilde{\lambda}) = H(\mathcal{W}) - H(\tilde{\mathcal{W}}).$$

Hence the weight of a path in $G(\mathscr{T})$ which occurs when translating a word with $\mathscr{T}$ gives the difference of the Hamming weights of the original and computed word.
So on the one hand, if there are no paths with negative weight, there can not be any word $A \in \mathcal{D}_w^*$ with smaller Hamming weight than B, which is the $\mathcal{D}$-w-NAF of $\mathsf{value}(A)$. On the other hand, if there is a path with negative weight, there is a word triggering this translation into one word with a higher Hamming weight. Hence the $\mathcal{D}$-w-NAF is not optimal. □

The theorem, useful as it is, has a much more handy corollary that follows directly out of it:

Corollary 2.1. *Let $\mathscr{T}$ be a transducer, transforming any word with digits in $\mathcal{D}_w$ into a $\mathcal{D}$-w-NAF, with related graph $G(\mathscr{T})$. Then*

The $\mathcal{D}$-w-NAF is optimal $\iff$ *The shortest paths from the start to the accept state have non-negative weight.*

So given a transducer $\mathscr{T}$ translating any word, the optimality of the $\mathcal{D}$-w-NAF for a certain $w \in \mathbb{N}$ can be proved by showing that a shortest path has non-negative weight. There are several algorithms for doing this; the real problems are how to construct the transducer, and how to check for optimality in a fast way.

2.5 Examples for Optimality

So far τ and the digit set $\mathcal{D}_w$ were based on the elliptic curve

$$\mathcal{E}_{3,\mu} : Y^2 = X^3 - X - \mu, \qquad \text{with } \mu \in \{-1, 1\}$$

in characteristic three. In [5], Heuberger gives attention to the number $\tau \in \mathbb{C}$ by looking at the elliptic curve

$$\mathcal{E}_a : Y^2 + XY = X^3 + aX^2 + 1, \qquad \text{with } a \in \{0, 1\},$$

which is defined over $\mathbb{F}_2$ and with point group $\mathcal{E}_a(\mathbb{F}_{2^n})$. The Frobenius endomorphism φ in characteristic 2 satisfies the equation

$$\varphi^2 - \mu\varphi + 2 = 0 \qquad \text{with } \mu = (-1)^{1-a}.$$

Similar to our approach, φ is identified with the root τ of this equation and this τ is again used for scalar multiplication. With a digit set $\mathcal{D}$, which is a w-NADS, the question arises again for which $w \in \mathbb{N}$ the $\mathcal{D}$-w-NAF is optimal.
Unlike the binary case, where the digit set of minimal norm representatives consists of zero and all odd integers of absolute value less than 2^{w-1} and where w-NAFs with this digit set minimises the Hamming weight, see Phillips and Burgess [10], there is no optimality for $w \in \{4, 5, 6\}$.

3 The Transducer

3.1 Basic Idea of the Transducer

Let $z \in \mathbb{Z}[\tau]$, z not divisible by τ, be an element with the two different digit representations

$$\mathsf{value}(\lambda_n\lambda_{n-1}\dots\lambda_1\lambda_0) = z = \mathsf{value}(\eta_k\eta_{k-1}\dots\eta_1\eta_0) \qquad \text{with} \qquad \lambda_i, \eta_j \in \mathcal{D}_w,$$

and let $\eta_k\eta_{k-1}\dots\eta_1\eta_0$ be a $\mathcal{D}$-w-NAF. Further assume that n, $k > w$. We can write the equation above as

$$z = \lambda_n\tau^n + \lambda_{n-1}\tau^{n-1} + \dots + \lambda_1\tau + \lambda_0 = \eta_k\tau^k + \eta_{k-1}\tau^{k-1} + \dots + \eta_1\tau + \eta_0 \tag{3.1}$$

and take a look at $z \mod \tau^w$:

$$\lambda_{w-1}\tau^{w-1}+\lambda_{w-2}\tau^{w-2}+\dots+\lambda_1\tau+\lambda_0 \equiv \eta_{w-1}\tau^{w-1}+\eta_{w-2}\tau^{w-2}+\dots+\eta_1\tau+\eta_0 \pmod{\tau^w}.$$

Because $\tau \nmid z$ and the η-digits are representing a $\mathcal{D}$-w-NAF, η_0 has to be an element in $\mathcal{D}_w\backslash\{0\}$. Hence $\eta_{w-1}, \eta_{w-2}, \dots, \eta_2, \eta_1$ have to be 0, and the above equation becomes

$$\lambda_{w-1}\tau^{w-1} + \lambda_{w-2}\tau^{w-2} + \dots + \lambda_1\tau + \lambda_0 \equiv \eta_0. \pmod{\tau^w}. \tag{3.2}$$

Thinking of a finite state machine or rather a transducer, the first digit of the $\mathcal{D}$-w-NAF can be computed by reading the first w digits of a given digit representation one-by-one, going from one state to another, where each of those states represents a value $a + b\tau \in \mathbb{Z}[\tau]$. When the w-th digit is read, the calculated value has to be reduced modulo τ^w to determine η_0. Although a finite state machine could be built to make all evaluations, a plain transducer is

all we need, because there are only finitely many possibilities for the coefficients $\lambda_0, \ldots, \lambda_{w-1}$ and therefore finitely many values for $\lambda_{w-1}\tau^{w-1} + \cdots + \lambda_0$.
The important step is to write Equation (3.2) in its equivalent form

$$\exists \delta_w \in \mathbb{Z}[\tau] \quad \text{with} \quad \delta_w \tau^w = \lambda_{w-1}\tau^{w-1} + \lambda_{w-2}\tau^{w-2} + \ldots + \lambda_1\tau + \lambda_0 - \eta_0.$$

We now define such $\delta \in \mathbb{Z}[\tau]$ in general:

Definition 3.1. Let $\lambda_n\lambda_{n-1}\ldots\lambda_1\lambda_0$ be a word in $\mathcal{D}_w^*$ with value $z \in \mathbb{Z}[\tau]$ and $\eta_k\eta_{k-1}\ldots\eta_1\eta_0$ the $\mathcal{D}$-w-NAF of z. Then define

$$\delta_j := \left(\sum_{i=0}^{j-1} \lambda_i\tau^i - \sum_{i=0}^{j-1} \eta_i\tau^i\right)\frac{1}{\tau^j} \qquad \text{for } 1 \le j \le \max\{n,k\}+1,$$

with $\lambda_r := 0$ for $r > n$ and $\eta_s := 0$ for $s > k$.

Further the set Δ is defined as the union of all sets of δ_j for every word in $\mathcal{D}_w^*$ without zero.

For the general case, a lemma is needed:

Lemma 3.1. *Let $z \in \mathbb{Z}[\tau]$ with $z = \eta_n\tau^n + \eta_{n-1}\tau^{n-1}\ldots\eta_1\tau + \eta_0 + \delta$, $\eta_i \in \mathcal{D}_w$, $\delta \in \Delta$. Then*

$$z \text{ is divisible by } \tau \iff \eta_0 + \delta \text{ is divisible by } \tau$$

Proof. This follows directly from the basic principles of divisibility. □

Now the following theorem can be stated:

Theorem 3.1. *Let $\lambda_n\lambda_{n-1}\ldots\lambda_1\lambda_0$ be a word in $\mathcal{D}_w^*$ with value $z \in \mathbb{Z}[\tau]$ and $\eta_k\eta_{k-1}\ldots\eta_1\eta_0$ the $\mathcal{D}$-w-NAF of z where the first m digits of the $\mathcal{D}$-w-NAF are known. There are two different cases an unknown value η_m can be derived from a given δ_m:*

1. *$\tau \mid \delta_m + \lambda_m$*
   Then $\eta_m = 0$ and $\delta_{m+1} = \frac{1}{\tau}(\delta_m + \lambda_m)$

2. *$\tau \nmid \delta_m + \lambda_m$*
   Then $\eta_m \neq 0$,

$$\eta_m \equiv \sum_{j=0}^{w-1} \lambda_{m+j}\tau^j + \delta_m \pmod{\tau^w},$$

   the digits $\eta_{m+1} = \ldots = \eta_{m+w-1} = 0$ and

$$\delta_{m+w} = \frac{1}{\tau^w}\left(\sum_{j=0}^{w-1} \lambda_{m+j}\tau^j + \delta_m - \eta_m\right)$$

Proof. From (3.1) we conclude that

$$\mathsf{value}(\eta_k\ldots\eta_m) = \frac{1}{\tau^m}\left(z - \sum_{j=0}^{m-1}\eta_j\tau^j\right) = \frac{1}{\tau^m}\left(\sum_{j=m}^{n}\lambda_j\tau^j\right) + \delta_m = \sum_{j=m}^{n}\lambda_j\tau^{j-m} + \delta_m.$$

Let $\delta_m + \lambda_m \equiv 0$ modulo τ. Because of Lemma 3.1, η_m has to be zero. So

$$\delta_{m+1} = \left(\sum_{i=0}^{m} \lambda_i \tau^i - \sum_{i=0}^{m} \eta_i \tau^i\right) \frac{1}{\tau^{m+1}} = \left(\sum_{i=0}^{m-1} \lambda_i \tau^i - \sum_{i=0}^{m-1} \eta_i \tau^i + \tau^m(\lambda_m - \eta_m)\right) \frac{1}{\tau^m} \frac{1}{\tau} =$$
$$= (\delta_m + \lambda_m - \eta_m)\frac{1}{\tau} = (\delta_m + \lambda_m)\frac{1}{\tau}.$$

Now assume that $\tau \nmid (\delta_m + \lambda_m)$. By the definition of δ_m we have

$$\delta_m \tau^m := \sum_{i=0}^{m-1} \lambda_i \tau^i - \sum_{i=0}^{m-1} \eta_i \tau^i.$$

So δ_{m+w} can be evaluated by

$$\delta_{m+w} = \left(\sum_{i=0}^{m+w-1} \lambda_i \tau^i - \sum_{i=0}^{m+w-1} \eta_i \tau^i\right) \frac{1}{\tau^{m+w}} = \left(\sum_{j=0}^{w-1} \lambda_{m+j} \tau^j - \sum_{j=0}^{w-1} \eta_{m+j} \tau^j + \delta_m\right) \frac{\tau^m}{\tau^{m+w}} =$$
$$= \left(\sum_{j=0}^{w-1} \lambda_{m+j} \tau^j + \delta_m - \eta_m\right) \frac{1}{\tau^w}$$

because the η digits form a w-NAF. Further η_m, which has to be in $\mathcal{D}_w \backslash \{0\}$ because of Lemma 3.1, can be obtained by

$$\sum_{j=0}^{m+w-1} \lambda_j \tau^j \equiv \sum_{j=0}^{m+w-1} \eta_j \tau^j \iff \sum_{j=0}^{m+w-1} \lambda_j \tau^j - \sum_{j=0}^{m-1} \eta_j \tau^j \equiv \sum_{j=m}^{m+w-1} \eta_j \tau^j \pmod{\tau^{m+w}},$$

which leads to

$$\tau^m \left(\sum_{j=0}^{w-1} \lambda_{m+j} \tau^j + \delta_m\right) \equiv \tau^m \left(\sum_{j=0}^{w-1} \eta_{m+j} \tau^j\right) \pmod{\tau^{m+w}}.$$

Because the η digits form a w-NAF, dividing this equation by τ^m leads to the second part of the theorem.

□

In Theorem 3.1 we can see that the δ_i are evaluated recursively. Because a transducer automaton built with the idea described above must have finitely many states and hence finitely many possible values for $\lambda_{m-1}\tau^{w-1} + \lambda_{m-2}\tau^{w-2} + \ldots + \lambda_{m-w+2}\tau + \lambda_{m-w+1} + \delta_{m-w+1}$, it is crucial that the set of all δs is finite.

Theorem 3.2. *The set $\Delta \subset \mathbb{Z}[\tau]$ is finite.*

Proof. For each bound $M \in \mathbb{N}$, there are only finitely many $a + b\tau \in \mathbb{Z}[\tau]$ with $|a + b\tau| \leq M$, and because $\mathcal{D}_w$ is finite, there is a $L_w \in \mathbb{N}$ for each w, such that $|\lambda_j| \leq L_w$ for all $\lambda_j \in \mathcal{D}_w$. It can be shown by induction, that $3L_w$ is an upper bound for the absolute values of all $\delta_i \in \Delta$:

Let $\mathcal{W} \in \mathcal{D}_w^*$ be a word of arbitrary length, where the first digit (from right) is λ_0 and for any digit λ_i, the next digit is λ_{i+1}. Further let the w-NAF of $\mathsf{value}\,(\mathcal{W})$ have digits η_i. Then

$$\delta_1 = (\lambda_0 - \eta_0)\frac{1}{\tau}$$

is bounded by

$$|\delta_1| = \left|(\lambda_0 - \eta_0)\frac{1}{\tau}\right| \leq \frac{1}{\sqrt{3}}(|\lambda_0| + |\eta_0|) < 3L_w$$

because of the triangle inequality and the fact that $|\tau| = \sqrt{3}$. Now suppose that

$$|\delta_i| \leq 3L_w \quad \text{for all } i \in \{1, \ldots, n\}.$$

Then

$$\begin{aligned} |\delta_{n+1}| &= \left|\left(\sum_{i=0}^{n} \lambda_i\tau^i - \sum_{i=0}^{n} \eta_i\tau^i\right)\frac{1}{\tau^{n+1}}\right| = \left|\left(\sum_{i=0}^{n-1} \lambda_i\tau^i - \sum_{i=0}^{n-1} \eta_i\tau^i + \lambda_n\tau^n - \eta_n\tau^n\right)\frac{1}{\tau^n\tau}\right| \leq \\ &\leq \left|\left(\sum_{i=0}^{n-1} \lambda_i\tau^i - \sum_{i=0}^{n-1} \eta_i\tau^i\right)\frac{1}{\tau^n\tau}\right| + \left|(\lambda_n - \eta_n)\,\frac{1}{\tau}\right| \leq |\delta_{n-1}|\frac{1}{\sqrt{3}} + \frac{2}{\sqrt{3}}L_w < 3L_w. \end{aligned}$$

Now we have to show that $\Delta \subseteq \mathbb{Z}[\tau]$. Because of (3.1), for any $k \in \mathbb{N}$

$$\lambda_{k-1}\tau^{k-1} + \lambda_{k-2}\tau^{k-2} + \cdots + \lambda_1\tau + \lambda_0 \equiv \eta_{k-1}\tau^{k-1} + \eta_{k-2}\tau^{k-2} + \cdots + \eta_1\tau + \eta_0 \pmod{\tau^k},$$

which implies that there is an element $\gamma \in \mathbb{Z}[\tau]$ such that

$$\tau^k\gamma = \sum_{i=0}^{k-1} \lambda_i\tau^i - \sum_{i=0}^{k-1} \eta_i\tau^i \quad \Longleftrightarrow \quad \gamma = \frac{1}{\tau^k}\left(\sum_{i=0}^{k-1} \lambda_i\tau^i - \sum_{i=0}^{k-1} \eta_i\tau^i\right)$$

and because $\gamma = \delta_k$ it follows that $\delta_k \in \mathbb{Z}[\tau]$. □

3.2 Buildup in detail

By putting all together from the previous section, the following sketch of the transducer $\mathscr{T}$ can be drawn. The states of this transducer are partitioned in $w-2$ different so called levels, which are Level 0, Level 1,...,Level $(w-2)$ and the Δ Level. Each state is given by a unique pair (z, j), where z is an element in $\mathbb{Z}[\tau]$, and $j \in \{0, 1, 2, \ldots, w-3, w-2, \Delta\}$, which stands for one of the possible levels.

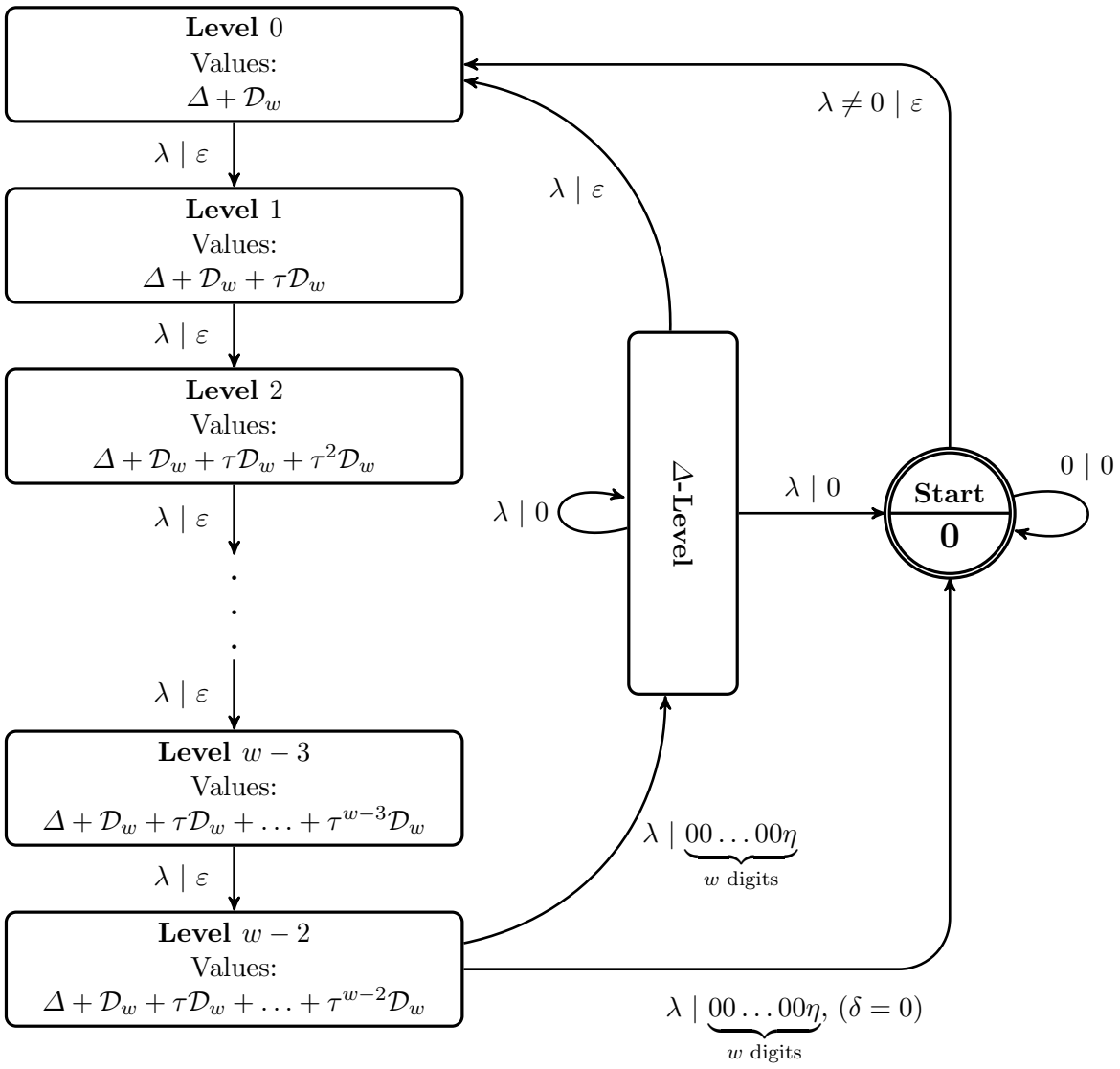


Figure 4: Sketch of transducer $\mathscr{T}$

To take a closer look at the sketch of the transducer $\mathscr{T}$, it has to be stated that $\Delta + \mathcal{D}_w + \tau\mathcal{D}_w + \ldots + \tau^j\mathcal{D}_w := \{\delta + \eta_0 + \eta_1\tau + \ldots + \eta_j\tau^j | \delta \in \Delta, \eta_i \in \mathcal{D}_w\}$. Moreover the node which is the start and accept state in one can also be reached without the transducer to be halted. The transducer can be stuck at this state arbitrarily, when it is on this state and the following input values are zeros.

The assembling of two consecutive levels Level $(j-1)$ and Level j is as follows:

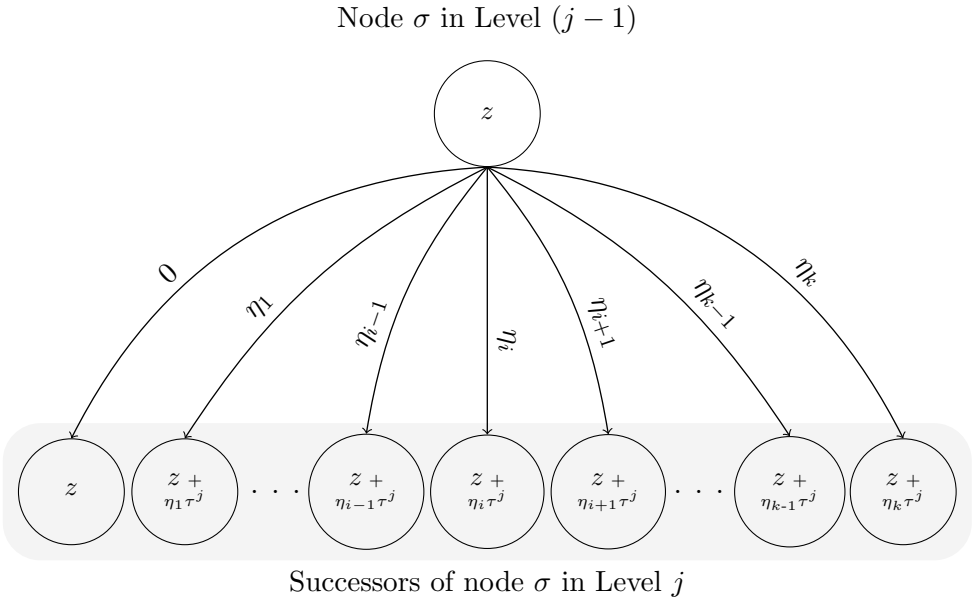


Figure 5: Node in Level $(j-1)$ and its successors

Level $(j-1)$ contains nodes with certain values, obtained by reading in a word. From every node in Level $(j-1)$, there are $|\mathcal{D}_w|$ many edges, one for every possible input digit read, going out and leading into a node in Level j. In Figure 5, only the edges going out of the node with value z are shown. One can see that the values on the next level are the old one plus all possible elements $\eta\tau^m$, where $\eta \in \mathcal{D}_w$ and m is the number of the new level.
The weights of nearly all the edges of the related graph $G(\mathscr{T})$ are one; all possible $\eta \in \mathcal{D}_w$ are read, whereas nothing is written at all. The only edges with weight 0 are those with the same values in the head and tail of it. Hence there are exactly |Level $(j-1)$| edges with weight 0 in $G(\mathscr{T})$ between Level $(j-1)$ and Level j, and $(|\mathcal{D}_w|-1)|$Level $(j-1)|$ with weight 1.

The maybe most interesting part of $\mathscr{T}$ is the Δ Level, which can be seen as a sort of switchman between two blocks of length w of the input-word. It has ingoing edges from Level $w-2$, which are those where the transducer $\mathscr{T}$ is writing something on its output tape. There is an edge from a node in Level $w-2$ with value λ to a node in the Δ Level with value z_s, if there is an $\eta_k \in \mathcal{D}_w$ so that z_s is evaluated as in Figure 6 and is not zero.

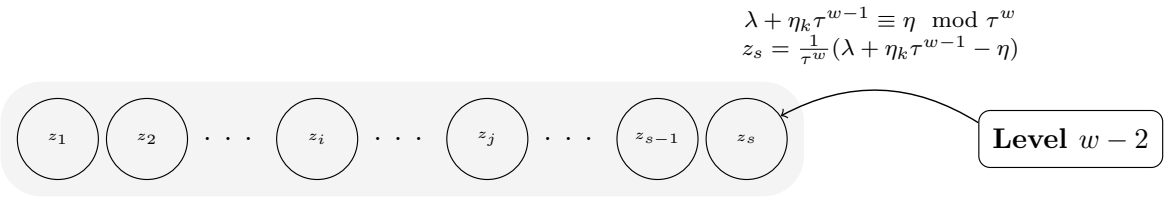


Figure 6: Δ Level: Ingoing edges

The outgoing edges lead to two different levels. There is an edge from the state with value z_1 in the Δ Level to Level 0, if there is an element $\eta_k \in \mathcal{D}_w$ with $\tau \nmid (z_1 + \eta_k)$.

Figure 7: Δ Level: Edges to Level 0

There are also edges from the Δ Level to the start/accept state. Such an edge is an outgoing edge of a node with value z_2, and there is a $\eta_k \in \mathcal{D}_w$ with $\eta_k + z_2 = 0$.

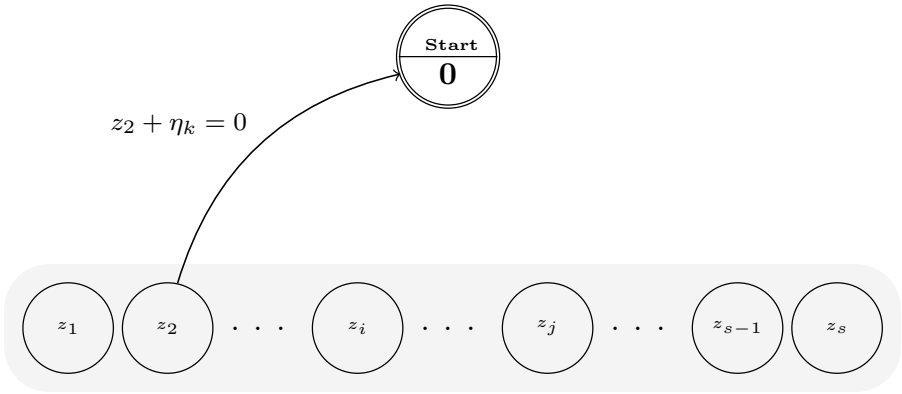


Figure 8: Δ Level: Edges to start/accept state

When the value $a + b\tau \in \mathbb{Z}[\tau]$ of a block of length w is divisible by τ after reducing it modulo τ^w, subtracting the outcome from it, dividing it by τ^w and adding the new read digit $\eta_k \in \mathcal{D}_w$ to it, a situation to the new input block would appear as stated in Proposition 3.1. This is handled by edges within the Δ Level, as shown in the following figure.

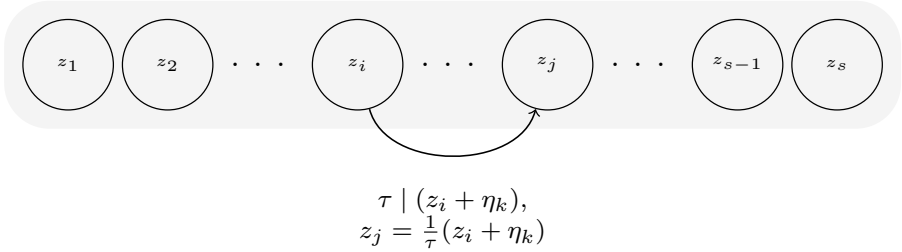


Figure 9: Δ Level: Elements divisible by τ

3.3 Correctness of the Transducer

The description of the transducer does not imply that it translates any word in $\mathcal{D}_w^*$ in the way it has to. To prove its correctness, we unite the start/accept state and the Δ Level to one level, which we denote Level -1.

Theorem 3.3. *Let $\mathcal{W} = 0^*\lambda_n\lambda_{n-1}\dots\lambda_1\lambda_0$ be a word in $\mathcal{D}_w^*$ and $\eta_k\eta_{k-1}\dots\eta_1\eta_0$ the w-NAF of $\mathsf{value}(\mathcal{W})$. Further let $\lambda_m\lambda_{m-1}\dots\lambda_1\lambda_0$ be the digits read by the transducer $\mathscr{T}$ and (z,j) the current state. Then $\eta_{m-j-1}\dots\eta_0$ are the digits written and*

$$z = \delta_{m-j} + \sum_{i=0}^{j} \lambda_{m-j+i}\tau^i.$$

Proof. We prove this by induction:
Let $m = 0$, which means that (z,j) is the first state of the transducer $\mathscr{T}$ after the start state. There are two different cases for λ_0:

1. If $\lambda_0 = 0$, $\mathcal{W}$ is divisible by τ and the first digit of the w-NAF has to be 0. The transducer stays in Level -1, $\eta_0 = 0$ is written and $z = 0$.
2. If $\lambda_0 \neq 0$, the current level is Level 0, which means that nothing is written and $z = \lambda_0$.

Now assume that the theorem holds for all states before the state (z,j), in particular the state when λ_{m-1} is read. We denote the predecessor state of (z,j) as $(\tilde{z},\tilde{j})$. The number of the current level is either -1 or in $\{0,1,\dots,w-2\}$. First assume that $j = -1$. There are two different ways in reaching Level -1:

- The state which leads to (z,j) is in Level -1. Then $\tilde{z} = \delta_m$ and $\eta_{m-1}\dots\eta_0$ is on the output tape. The transducer stays in Level -1, because

  $$\delta_m + \lambda_m \equiv 0 \pmod{\tau},$$

  and it follows by Theorem 3.1, that $z = \frac{1}{\tau}(\delta_m + \lambda_m) = \delta_{m+1}$. Further η_m has to be 0 because of Lemma 3.1.

- The state $(\tilde{z},\tilde{j})$ is in not in Level -1. Then it has to be in Level $w-2$, which means that

  $$\tilde{z} = \delta_{m-w+1} + \sum_{i=0}^{w-2} \lambda_{m-w+1+i}\tau^i.$$

  Because of Theorem 3.1, the next digits of the w-NAF are written and

  $$z = \left(\delta_{m-w+1} + \sum_{i=0}^{w-1} \lambda_{m-w+1+i}\tau^i - \eta_m\right)\frac{1}{\tau^w} = \delta_m.$$

Now assume that the state (z,j) is given with $j \in \{0,\dots,w-2\}$. At the transition to state (z,j), nothing was written, so by the induction hypothesis, the digits $\eta_{m-j-1}\dots\eta_0$ are from the w-NAF. The value z is given by

$$z = \delta_{m-1-j+1} + \sum_{i=0}^{j-1} \lambda_{m-1-j+1+i}\tau^i + \lambda_m\tau^j = \delta_{m-j} + \sum_{i=0}^{j} \lambda_{m-j+i}\tau^i,$$

and hence the proof is complete. □

4 Fast algorithmic realization of the Transducer

When trying to show the optimality of a $\mathcal{D}$-w-NAF for a certain w, it is crucial to be very efficient when implementing the evaluation of the finite automaton, and certainly when computing the paths from the start to the accepting state. Based on empirical observations, the time needed for the proof of the optimality for $w+1$ is about 40-80 times higher than the time needed for w. The cardinality of $\mathcal{D}_w \backslash \{0\}$ grows exactly by a factor 3, and the number of states of $\mathscr{T}$ by a factor of about 10.

4.1 Reducing the graph to a sixth part

The first step that can be realised when trying to lower the time needed to show the optimality of a $\mathcal{D}$-w-NAF for a certain w, is to simplify the graph $G(\mathscr{T})$ of the transducer as much as possible. The smaller the graph is, the less time is needed for evaluating the length of paths from the start to the accept state. In fact, there is a very basic way for making the graph smaller, based on the structure of $\mathcal{D}_w$.

Theorem 4.1. *Let $\mathscr{T}$ be a Transducer built like shown in Figure 4. Then the values of the nodes of the related graph $G(\mathscr{T})$, which are not the start/accept state, are symmetric under a rotation of $\frac{\pi}{3}$.*

Proof. First we can show that the set $\mathcal{D}_w + \tau\mathcal{D}_w + \ldots + \tau^n\mathcal{D}_w = \{\eta_0 + \eta_1\tau + \ldots + \eta_n\tau^n | \eta_i \in \mathcal{D}_w\}$ is symmetric under a rotation of $\frac{\pi}{3}$ by induction:
$\mathcal{D}_w$ is symmetric because of its construction, as done in Theorem 2.1. Now assume that the assumption holds for $\mathcal{D}_w + \ldots + \tau^{n-1}\mathcal{D}_w$. So this set can be written as

$$\mathcal{D}_w + \ldots + \tau^{n-1}\mathcal{D}_w = \bigcup_{0\le k\le 5} \zeta^k \tilde{\mathcal{D}}^w_{n-1}, \tag{4.1}$$

for some set $\tilde{\mathcal{D}}^w_{n-1}$. Adding $\tau^n\mathcal{D}_w$ leads to

$$\begin{aligned}\bigcup_{0\le k\le 5} \zeta^k \tilde{\mathcal{D}}^w_{n-1} + \tau^n\mathcal{D}_w &= \bigcup_{0\le k\le 5} \zeta^k \tilde{\mathcal{D}}^w_{n-1} + \{0\} \cup \bigcup_{0\le j\le 5} \zeta^j \mathcal{D}_{w,0} = \\ &= \bigcup_{0\le k\le 5} \zeta^k (\tilde{\mathcal{D}}^w_{n-1} + \{0\} \cup \bigcup_{0\le j\le 5} \zeta^j \mathcal{D}_{w,0}).\end{aligned}$$

Hence $\mathcal{D}_w + \tau\mathcal{D}_w + \ldots + \tau^n\mathcal{D}_w$ is symmetric.
The next thing to show is that Δ_1 is symmetric, which is the set of all deltas constructed like in chapter 3.1 out of the set $\mathcal{D}_w + \tau\mathcal{D}_w + \ldots + \tau^n\mathcal{D}_w$. Like in 4.1, $\mathcal{D}_w + \tau\mathcal{D}_w + \ldots + \tau^n\mathcal{D}_w$ consists of sets $\zeta^k\tilde{\mathcal{D}}^w_n$. Let $\lambda_0 \in \tilde{\mathcal{D}}^w_n$, and let δ_0 in Δ_1 be

$$\delta_0 = \frac{\lambda_0 - \eta_0}{\tau^w} \quad \text{for } \eta_0 \in \mathcal{D}_w \text{ with } \quad \lambda_0 \equiv \eta_0 \mod \tau^w.$$

For $k \in \{1, \ldots 5\}$ we get

$$\zeta^k\lambda_0 \equiv \zeta^k\eta_0 \mod \tau^w \quad \text{and} \quad \delta_k = \frac{\zeta^k\lambda_0 - \zeta^k\eta_0}{\tau^w} = \zeta^k\frac{\lambda_0 - \eta_0}{\tau^w} = \zeta^k\delta_0$$

and therefore Δ_1 is symmetric. Because it has been shown above that the sum of sets which are symmetric under a rotation of $\frac{\pi}{3}$ is symmetric, $\mathcal{D}_w + \ldots + \tau^n\mathcal{D}_w + \Delta_1$ has this property,

and because the Δ_l sets are evaluated iteratively like above, $\Delta + \sum_{k=0}^{m} \tau^k \mathcal{D}_w$ is symmetric for every $m \in \{0, \ldots, n\}$, and so are the values of the nodes of $G(\mathscr{T})$. □

Because of the theorem above, for every node with value λ_0 in Level i with $i \in \{0, \ldots, w-2\}$ or the Δ Level, there are the nodes with values $\zeta\lambda_0, \zeta^2\lambda_0, \zeta^3\lambda_0, \zeta^4\lambda_0, \zeta^5\lambda_0$ in the same level. Suppose that the transducer $\mathscr{T}$ is in the state λ_0 and the next digit that is read is a η, the output of $\mathscr{T}$ at this single input is an $a \in \mathcal{D}_w^*$, and the next state is a node with value λ_1 in a following level. Then for every $j \in \{1, \ldots 5\}$, when $\mathscr{T}$ is in in the state with node-value $\zeta^j\lambda$ and the input digit is $\zeta^j\eta$, the output is $\zeta^j a$ and the next state is $\zeta^j\lambda_1$, which is in the same level as λ_1. This follows directly from the fact that

$$\zeta^j\lambda_0 + \zeta^j\eta = \zeta^j(\lambda_0 + \eta) \quad \text{and} \quad \zeta\varepsilon = \varepsilon \text{ when } a = \varepsilon,$$
$$\text{and} \quad \lambda_0 + \eta \equiv \tilde{\eta} \mod \tau^w \Rightarrow \zeta^j(\lambda_0 + \eta) \equiv \zeta^j\tilde{\eta} \mod \tau^w.$$

The cases when one of the two levels is the start/accept state, or both levels are the Δ-Level can be treated in the same way.
The left hand side of figure 10 is a sketch of the described feature:

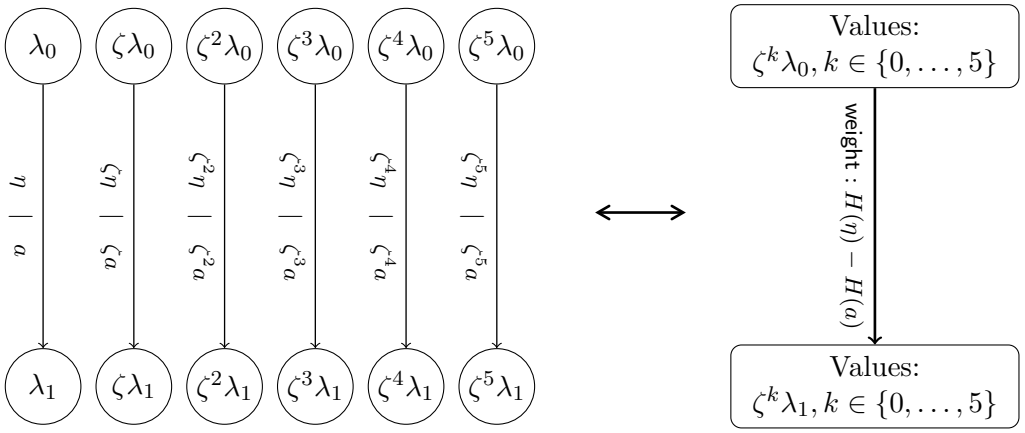


Figure 10: Compression of nodes

The related graph $G(\mathscr{T})$ has no information about input and output, instead the weights of the edges are the differences of the Hamming weights. For a single input digit $\eta \neq 0$, and thus for the input we get at each step of the finite automaton, the Hamming weight $H(\eta)$ is 1, which is the same as $H(\zeta^k\eta)$ for any $k \in \mathbb{N}$. For $\eta = 0$, it is clear that the product $\zeta^k\eta$ and η both have Hamming weight of 0. So the Hamming weights of the input digits of the above Figure 10 is the same. The two possible outputs that can occur are 0 and a word $00\ldots0\bar{\eta}$, and they too have the same Hamming weight when rotated by $\frac{\pi}{3}$. So when interested in a shortest path from the start to the accept state, the graph $G(\mathscr{T})$ can be compressed to a sixth part as in Figure 10, because the difference $H(\eta) - H(a)$ is invariant under a rotation of $\frac{\pi}{3}$.
To realize this simplification, a representative for each tuple $(\lambda, \zeta\lambda, \ldots, \zeta^5\lambda)$ has to be chosen. An easy way to do this is to take only the nodes for $V(G(\mathscr{T}))$, whose arguments of the values of the nodes seen as a complex number are in $[0, \frac{\pi}{3})$. Numerical inaccurateness and

nonessential additional computing time when evaluating the arguments with the arctan can be avoided by finding the representatives with discrete equations:

Proposition 4.1. *Let $z = a + b\tau$ be an element in $\mathbb{Z}[\tau]$, and define the left-closed, right-open intervals $A_k := [k\frac{\pi}{3}, (k+1)\frac{\pi}{3})$ with $k \in \{-3, -2, -1, 0, 1, 2\}$. Further set $c := 2a + 3\mu b$. Then*

$$\begin{aligned}
&\arg(z) \in A_{-1} \Leftrightarrow -c \leq b < 0 && \arg(z) \in A_0 \Leftrightarrow 0 \leq b < c \\
&\arg(z) \in A_{-3} \Leftrightarrow c < b \leq 0 && \arg(z) \in A_1 \Leftrightarrow c \leq b, c \geq 0 \text{ or } b > -c, c \leq 0 \\
&\arg(z) \in A_2 \Leftrightarrow 0 < b \leq -c && \arg(z) \in A_{-2} \Leftrightarrow b \leq c, c \leq 0 \text{ or } b < -c, c \geq 0
\end{aligned}$$

where $\arg(z)$ *is the argument of* $z \in \mathbb{Z}[\tau] \subset \mathbb{C}$.

Proof. Looking at z as a complex number rather than an element of $\mathbb{Z}[\tau]$, z can be written as

$$z = a + b\tau = a + b\frac{3\mu + \sqrt{-3}}{2} = a + \frac{3\mu b}{2} + i\frac{\sqrt{3}b}{2}.$$

Now assume that $\mathrm{Re}(z) \geq 0$ (the other case can be treated analogously). The argument of z is

$$\arg(z) = \arctan\left(\frac{\frac{\sqrt{3}b}{2}}{a + \frac{3\mu b}{2}}\right) = \arctan\left(\frac{\sqrt{3}b}{2a + 3\mu b}\right).$$

Let $b \geq 0$ (again, the case $b < 0$ can be treated analogously). For a positive numerator x,

$$\varphi \in A_0 \Leftrightarrow \frac{x}{y} \in [0, \sqrt{3}), \quad \varphi \in A_1 \Leftrightarrow \frac{x}{y} \in (-\infty, -\sqrt{3}) \cup [\sqrt{3}, \infty), \quad \varphi \in A_2 \Leftrightarrow \frac{x}{y} \in [-\sqrt{3}, 0),$$

where $\varphi := \arctan\left(\frac{x}{y}\right)$. The theorem follows directly by plugging in $\sqrt{3}b$ for x and $2a + 3\mu b$ for y.

□

4.2 Adjustment of the Bellman-Ford algorithm

The optimality of a $\mathcal{D}_w$-w-NAF can be checked by searching a graph for paths with negative weight from the start to the accept state. This can be reduced to following equivalent problem:

Proposition 4.2. *Let $\mathscr{T}$ be the transducer of a $\mathcal{D}$-w-NAF with related graph $G(\mathscr{T})$. Then*

The $\mathcal{D}$-w-NAF is optimal $\Longleftrightarrow$ There is no negative cycle in $G(\mathscr{T})$.

Proof.

- Suppose that the $\mathcal{D}$-w-NAF is optimal. If $G(\mathscr{T})$ had a negative cycle, there would be a path from the start state of the transducer to it, because there is a path from it to every single node. So there would be a negative path from the start to the accept state of $G(\mathscr{T})$, in contradiction to Theorem 2.3.

- Suppose that the $\mathcal{D}$-w-NAF is not optimal. There is a word $\mathcal{W} = \eta_k\eta_{k-1}\dots\eta_1\eta_0$ which is a $\mathcal{D}$-w-NAF and a Word $\mathcal{V} = \lambda_m\lambda_{m-1}\dots\lambda_1\lambda_0$ with $\mathsf{value}(\mathcal{W}) = \mathsf{value}(\mathcal{V})$ and $\mathcal{W}$ has a higher Hamming weight than $\mathcal{V}$. Thus the transducer $\mathscr{T}$ would work along a negative path with weight N. With the word $\lambda_m\lambda_{m-1}\dots\lambda_1\lambda_0 00\dots 00\lambda_m\lambda_{m-1}\dots\lambda_1\lambda_0$ as input, where the number of zeros has to be large enough to be finished with writing $\mathcal{W}$ when reading λ_0 for the second time, the path would have weight $2N$. Because the word $\mathcal{V}$ can be extended arbitrarily long in this way, there is a path with arbitrary large negative weight, and so there has to be a negative cycle.

□

A common algorithm for finding shortest paths in a graph with negative weights and also for revealing negative cycles is the Bellman-Ford algorithm. The usual algorithm is shown in Algorithm 1, which can be found in [3]. By scanning every single edge in a graph G at every single loop run, the node potentials, which are equivalent with the distance, are updated when a certain inequality applies. The check for a negative cycle of G is in the end of the algorithm. Obviously, the algorithm can answer the question of optimality of a $\mathcal{D}$-w-NAF. However it is not very fast in the present form. With a direct implementation of Algorithm 1, there are $n \cdot m$ if-queries needed, where n is the number of nodes and m is the number of edges in G. This is based on the fact that the primal version of the algorithm finds a shortest path from a root to all nodes in G, thus a shortest-path-arborescence. But the result we are interested in is only whether there is a negative path from the start to the accept state in $G(\mathscr{T})$. Also, if a negative path exists, the path itself is worth knowing.

Input *A graph $G = (V, E)$, with a node s as the source and weighted edges*

Initialize:
Set $\mathsf{distance}(s) := 0$ and $\mathsf{predecessor}(s) := \mathsf{null}$
For all $v \in V \backslash \{s\}$ set $\mathsf{distance}(v) := \infty$ and $\mathsf{predecessor}(v) := \mathsf{null}$

for $i \in \{1, \dots, |V| - 1\}$ **do**
  for $(u, v) \in E$ **do**
    if $\mathsf{distance}(u) + \mathsf{weight}((u, v)) < \mathsf{distance}(v)$ **then**
      $\mathsf{distance}(v) := \mathsf{distance}(u) + \mathsf{weight}((u, v))$
      $\mathsf{predecessor}(v) := u$

for $v \in V$ **do**
  for $(u, v) \in E$ **do**
    if $\mathsf{distance}(u) + \mathsf{weight}((u, v)) < \mathsf{distance}(v)$ **then**
      return 'Negative cycle detected!'

Algorithm 1: Bellman-Ford

When looking at the distance function of the Bellman-Ford algorithm, it is evident that it is monotonically decreasing for each node in G. So it is sufficient to take notice of the distance of the start/accept state. When it is negative at any time, there is a negative cycle in G, and the algorithm can stop. Another step in the algorithm that can be improved is the selection of edges in the main loop. Every single edge is scanned, even if the distance doesn't

change. In fact, if there is a valid assignment for the node potentials, there are nodes in every level from Level 1 to Level $w-2$ with a potential that is already known:

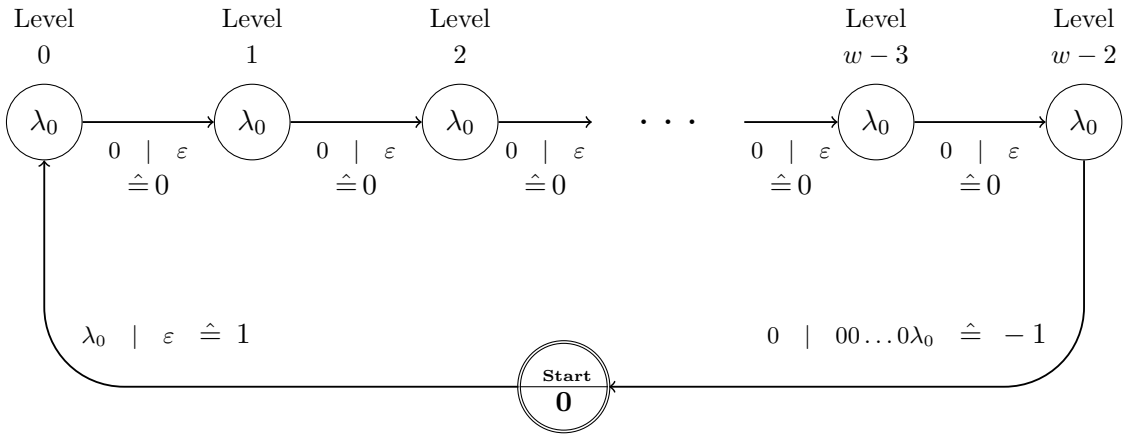


Figure 11: A priori node potentials

For Figure 11 let λ_0 be in $\mathcal{D}_w/\{0\}$. Every level from Level 0 to Level $w-2$ contains a node with value λ_0, because there is the possibility of a word as an input that is just the digit λ_0, which corresponds to $00\ldots0\lambda_0$, where the number of zeros is arbitrarily long. The path of the executions of the transducer for exactly this input is shown above. Now define the following node potentials:

$$\begin{aligned}&\pi(s) := 0, \quad \text{potential of start/accept state}\\ &\pi_j(\lambda_0) \in \mathbb{R}, \quad \text{potential of the node in level } j \text{ with value } \lambda_0.\end{aligned}$$

For valid node potentials in a graph G, following inequality has to hold:

$$\pi(u) + \mathsf{weight}((u,v)) \leq \pi(v) \qquad \text{for all edges } (u,v) \in E(G).$$

In figure 11, the weights of the edges are given. It follows directly from this:

$$\begin{aligned}\pi(s)+1 \leq &\pi_0(\lambda_0) \leq \pi_1(\lambda_0) \leq \ldots \leq \pi_{w-2}(\lambda_0) \quad \text{and} \quad \pi_{w-2}(\lambda_0) - 1 \leq \pi(s)\\ &\Longrightarrow \pi(s)+1 = \pi_0(\lambda_0) = \pi_1(\lambda_0) = \ldots = \pi_{w-2}(\lambda_0) \quad .\end{aligned}$$

Thus for every level of the relating graph of the transducer, there are $|\mathcal{D}_w|$ many nodes with known node potential. The node potential of the start/accept state can be chosen arbitrarily, eg. 0.
This means for the Bellman-Ford algorithm, that there are at least $(w-3)|\mathcal{D}_w|^2$ edges which do not have to be checked more than once. Every single node in a certain level has $|\mathcal{D}_w|$ many outgoing edges. When $\pi(u)$ does not change, the inequality $\pi(u) + \mathsf{weight}((u,v)) < \pi(v)$ can only apply once, because π is a monotonically decreasing function.
A feasible solution for this problem of inefficiency is to let the main loop of the Bellman-Ford algorithm only check edges which have the possibility of changing node potentials. This is done in Algorithm 2. The idea is that an edge $(u,v) \in E(G)$ is relevant, if $\pi(u)$ has recently changed. Therefore two stacks K_0 and K_1 are introduced for storing nodes. In one of them,

nodes of the current run of the main-loop are stored. When a node is tested for all its outgoing edges, it can be removed from the stack. Whenever the potential of a node changes, the node itself is stored in the stack which is not used at the time. In the beginning, only the start/accept node is stored in K_1. This causes the modified Bellman-Ford algorithm to execute in the same order as the transducer can be built (by starting with the $\mathcal{D}_w$ set on Level 1, and adding elements to the Δ-Level iteratively).

Input *A graph $G = (V, E)$, with a node s as the source and weighted edges*

Initialize:
$K_1 = \{(start/acceptstate)\}$, $K_0 = \emptyset$
Set $\mathsf{distance}(s) := 0$ and $\mathsf{predecessor}(s) := \mathsf{null}$
For all $v \in V \backslash \{s\}$ set $\mathsf{distance}(v) := \infty$ and $\mathsf{predecessor}(v) := \mathsf{null}$

$i = 1$
while *True* **do**
  if $\mathsf{distance}(s) < 0$ **then**
    return 'Negative cycle detected!'
  if $K_0 = \emptyset$ *and* $K_1 = \emptyset$ **then**
    return
  while $K_i \neq \emptyset$ **do**
    $u = \mathsf{pop}(K_i)$
    for *every edge* (u, v) *in* $\delta^+(u)$ **do**
      if $\mathsf{distance}(u) + \mathsf{weight}((u, v)) < \mathsf{distance}(v)$ **then**
        $\mathsf{distance}(v) := \mathsf{distance}(u) + \mathsf{weight}((u, v))$
        $\mathsf{predecessor}(v) := u$
        $\mathsf{push}(v, K_j)$
  $j = i$
  $i = i + 1 \mod 2$

Algorithm 2: Modified Bellman-Ford

The last instance that changes to the original Bellman-Ford algorithm is to check if both K_1 and K_2 are empty before an execution of the main loop. If both are empty, no element was added to the current appropriate set K_i in the last loop run, which means that not a single node potential has changed. This is the case when the Bellman-Ford algorithm is finished, and thus there can be an early break.

5 Arithmetic in $\mathbb{Z}[\tau]$

5.1 Reducing to elements in $\mathcal{D}_w$

Plenty depends on the reduction of elements in $\mathbb{Z}[\tau]$ modulo τ^w to an element in $\mathcal{D}_w$. One way to do this is based on the division of an element in $\mathbb{Z}[\tau]$ by τ^k.

Proposition 5.1. *Let $\alpha \in \mathbb{Z}[\tau]$ and $k \in \mathbb{N}$. Then*

$$\frac{\alpha}{\tau^k} = \frac{1}{3^k}\alpha\bar{\tau}^k$$

Proof. This is trivial due to the fact that

$$\frac{\alpha}{\tau^k} = \frac{\alpha}{\tau^k}\frac{\bar{\tau}^k}{\bar{\tau}^k} = \frac{\alpha\bar{\tau}^k}{(\underbrace{\bar{\tau}\tau}_{=3})^k} = \frac{1}{3^k}\alpha\bar{\tau}^k \quad .$$

□

For an element $\alpha \in \mathbb{Z}[\tau]$, an element $\eta \in \mathcal{D}_w$ with $\alpha \equiv \eta \pmod{\tau^w}$ can be determined by dividing $\alpha - \eta_i$ by τ^w for every $\eta_i \in \mathcal{D}_w$. If $z = (\alpha - \eta_i)/\tau^w$ is an element $a + b\tau$ with both a and b in $\mathbb{Z}$, z is an element in $\mathbb{Z}[\tau]$ and thus $\alpha \equiv \eta_i \pmod{\tau^w}$ holds. The drawback of this method is its worst case running time: To find an appropriate η_i, it can happen that every element in $\mathcal{D}_w$ has to be checked.
However, there is a more efficient way in reducing, deduced from following theorem:

Theorem 5.1. *Let $a + b\tau$ be an element of $\mathbb{Z}[\tau]$, $\tau \nmid (a + b\tau)$, with balanced ternary representation*

$$a + b\tau = \sum_{j=0}^{n} a_j 3^j + \left(\sum_{i=0}^{m} b_i 3^i\right)\tau, \qquad a_0 \neq 0, a_j, b_i \in \{-1, 0, 1\} \text{ and } n, m \in \mathbb{N}.$$

Then

$$\begin{aligned} a + b\tau \equiv c + d\tau \pmod{\tau^w} \iff \quad & a_i = c_i \quad \forall i \in \{0, 1, \ldots \left\lceil \frac{w}{2} \right\rceil - 1\}, \\ & b_j = d_j \quad \forall j \in \{0, 1, \ldots \left\lfloor \frac{w}{2} \right\rfloor - 1\}, \end{aligned}$$

where c_i, d_j are digits from the balanced ternary representation of $c + d\tau \in \mathcal{D}_w$.

To make a proof, the following lemma is needed:

Lemma 5.1. *Let $a + b\tau \in \mathbb{Z}[\tau]$ with a, $b \in \mathbb{Z}[\tau]$ and $w \geq 1$. Then*

$$\tau^w \mid (a + b\tau) \quad \iff \quad 3^{\lceil w/2 \rceil} \mid a \text{ and } 3^{\lfloor w/2 \rfloor} \mid b.$$

Proof. The proof can be made by induction. For $w = 1$, the lemma is the same as Proposition 2.1. Further it can be shown that $a \in \mathbb{Z}$ is divisible by 3 iff a is divisible by τ^2. This is due to the fact that $3 = \tau^2\zeta$. Now assume that the lemma holds for $n \in \mathbb{N}$. Then

$$\begin{aligned} \tau^{n+1} \mid (a + b\tau) &\iff \exists \tilde{a} + \tilde{b}\tau \in \mathbb{Z}[\tau] \text{ with } a + b\tau = (\tilde{a} + \tilde{b}\tau)\tau^{n-1}\tau^2 \\ &\iff \exists a' + b'\tau \in \mathbb{Z}[\tau] \text{ with } a + b\tau = \left(3^{\lceil (n-1)/2 \rceil}a' + 3^{\lfloor (n-1)/2 \rfloor}b'\tau\right)\tau^2 \end{aligned}$$

and

$$\begin{aligned} a + b\tau &= \left(3^{\lceil (n-1)/2 \rceil}a' + 3^{\lfloor (n-1)/2 \rfloor}b'\tau\right)\tau^2 = \\ &= -3^{\lceil (n+1)/2 \rceil}(a' + 3^{[2\nmid n]}\mu b') + 3^{\lfloor (n+1)/2 \rfloor}(3^{[2|n]}\mu a' + 2b')\tau \end{aligned}$$

where $[expr]$ denotes the Iversonian notation, cf. Graham, Knuth and Patashnik [4]. □

Proof of Theorem 5.1.

"$\Longrightarrow$" Because $c + d\tau \in \mathcal{D}_w$, $c + d\tau$ is a representative of minimum norm of its residue class modulo τ^w. So $a + b\tau$ can be written as

$$a + b\tau = c + d\tau + e\tau^w \qquad \text{for an element } e \in \mathbb{Z}[\tau].$$

With the balanced ternary representation of $c + d\tau$,

$$c + d\tau = \sum_{j=0}^{u} c_j 3^j + \left(\sum_{i=0}^{v} d_i 3^i \right) \tau, \qquad c_j, d_i \in \{-1, 0, 1\} \text{ and } u, v \in \mathbb{N},$$

it follows that

$$\begin{aligned} a + b\tau &= \sum_{j=0}^{\lceil w/2 \rceil - 1} a_j 3^j + \left(\sum_{i=0}^{\lfloor w/2 \rfloor - 1} b_i 3^i \right) \tau + \tilde{e}\tau^w = c + d\tau + e\tau^w = \\ &= \sum_{j=0}^{\lceil w/2 \rceil - 1} c_j 3^j + \left(\sum_{i=0}^{\lfloor w/2 \rfloor - 1} d_i 3^i \right) \tau + f\tau^w + e\tau^w \qquad \text{for } \tilde{e}, f \in \mathbb{Z}[\tau] \end{aligned}$$

Because of the lemma above, there is a balanced ternary representation for $\tilde{e}\tau^w$ and $(f + e)\tau^w$:

$$\begin{aligned} \tilde{e}\tau^w &= \sum_{i=\lceil w/2 \rceil}^{E_1} \tilde{e}_{i,1} 3^i + \left(\sum_{i=\lfloor w/2 \rfloor}^{E_2} \tilde{e}_{i,2} 3^i \right) \tau \quad \text{for } E_1, E_2 \in \mathbb{N} \\ (f + e)\tau^w &= \sum_{i=\lceil w/2 \rceil}^{F_1} \tilde{f}_{i,1} 3^i + \left(\sum_{i=\lfloor w/2 \rfloor}^{F_2} \tilde{f}_{i,2} 3^i \right) \tau \quad \text{for } F_1, F_2 \in \mathbb{N} \end{aligned}$$

We obtain the implication by comparing the coefficients of the balanced ternary representations of

$$\sum_{j=0}^{\lceil w/2 \rceil - 1} a_j 3^j + \sum_{i=\lceil w/2 \rceil}^{E_1} \tilde{e}_{i,1} 3^i = \sum_{j=0}^{\lceil w/2 \rceil - 1} c_j 3^j + \sum_{i=\lceil w/2 \rceil}^{F_1} \tilde{f}_{i,1} 3^i$$

and

$$\sum_{i=0}^{\lfloor w/2 \rfloor - 1} b_i 3^i + \sum_{i=\lfloor w/2 \rfloor}^{E_2} \tilde{e}_{i,2} 3^i = \sum_{i=0}^{\lfloor w/2 \rfloor - 1} d_i 3^i + \sum_{i=\lfloor w/2 \rfloor}^{F_2} \tilde{f}_{i,2} 3^i.$$

"$\Longleftarrow$"

$$\begin{aligned} a + b\tau &= \sum_{j=0}^{\lceil w/2 \rceil - 1} a_j 3^j + \left(\sum_{i=0}^{\lfloor w/2 \rfloor - 1} b_i 3^i \right) \tau + \sum_{j=\lceil w/2 \rceil}^{n} a_j 3^j + \left(\sum_{i=\lfloor w/2 \rfloor}^{m} b_i 3^i \right) \tau \equiv \\ &\equiv \sum_{j=0}^{\lceil w/2 \rceil - 1} a_j 3^j + \left(\sum_{i=0}^{\lfloor w/2 \rfloor - 1} b_i 3^i \right) \tau = \sum_{j=0}^{\lceil w/2 \rceil - 1} c_j 3^j + \left(\sum_{i=0}^{\lfloor w/2 \rfloor - 1} d_i 3^i \right) \tau \equiv \\ &\equiv c + d\tau \pmod{\tau^w} \end{aligned}$$

□

Remark. Knuth [6] wrote about balanced ternary integer representations: *Perhaps the prettiest number system of all is the balanced ternary notation, which consists of radix-3 representation using* -1, 0, *and* $+1$ *as "trits" (ternary digits) instead of* 0, 1, *and* 2.

So there is also the way of reducing an element in $\mathbb{Z}[\tau]$ to one in $\mathcal{D}_w$ by computing the balanced ternary digit expansions and comparing them. The following corollary states that there is still an easier way:

Corollary 5.1. *Let* $z \in \mathbb{Z}[\tau]$ *with* $z = a + b\tau$ *and* a, b *are rational integers. Then*

$$z \equiv \eta \mod \tau^w \text{ with } \eta \in \mathcal{D}_w \iff \eta = c + d\tau \text{ and } a \equiv c \mod 3^{\lceil w/2 \rceil}, \quad b \equiv d \mod 3^{\lfloor w/2 \rfloor}$$

Proof. This follows directly from the theorem above. The first few digits of a balanced ternary expansion have to be the same, and thus the numbers reduced by 3 raised to a certain power, which is in this case either $\lceil \frac{w}{2} \rceil$ or $\lfloor \frac{w}{2} \rfloor$. □

From the corollary above a possible way in evaluating z modulo τ^w is given. A dictionary $\bar{\mathcal{D}}_w$ can be computed with

$$\bar{\mathcal{D}}_w = \Big\{ (\tilde{c} + \tilde{d}\tau, c + d\tau) \in \mathbb{Z}[\tau]^2 : (c + b\tau) \in \mathcal{D}_w \text{ with } c \equiv \tilde{c} \pmod{3^{\lceil w/2 \rceil}} \text{ and } d \equiv \tilde{d} \pmod{3^{\lfloor w/2 \rfloor}} \Big\},$$

where for every element $c + d\tau \in \mathcal{D}_w$ exactly one pair $(\tilde{c} + \tilde{d}\tau, c + d\tau)$ with $0 < \tilde{c} < 3^{\lceil w/2 \rceil}$ and $0 < \tilde{d} < 3^{\lfloor w/2 \rfloor}$ is stored. An element $z = a + b\tau$ can be reduced modulo τ^w by computing

$$a \equiv \tilde{c} \mod 3^{\lceil w/2 \rceil}, \quad b \equiv \tilde{d} \mod 3^{\lfloor w/2 \rfloor},$$

and finding the element $(\tilde{c} + \tilde{d}\tau, c + d\tau)$ in $\bar{\mathcal{D}}_w$. Then

$$a + b\tau \equiv c + d\tau \pmod{\tau^w} \qquad \text{with } c + d\tau \in \mathcal{D}_w.$$

5.2 Needful Operations

When computing the set $\mathcal{D}_w$, it is crucial to make evaluations of the form ζz, where z is an element in $\mathbb{Z}[\tau]$. Of course this can be done without great effort because both ζ and z are complex numbers. With the following proposition, there is a way of making this kind of multiplication and staying with elements $a + b\tau$, where both a and b are integers.

Proposition 5.2. *Let* $z = a + b\tau$ *be in* $\mathbb{Z}[\tau]$. *Then* $\tilde{z} = z\zeta^k$ *is*

$$\tilde{z} = a + b\tau \text{ for } m = 0 \qquad \tilde{z} = 2a + 3\mu b + (-\mu a - b)\tau \text{ for } m = 1$$
$$\tilde{z} = a + 3\mu b + (-\mu a - 2b)\tau \text{ for } m = 2 \qquad \tilde{z} = -a - b\tau \text{ for } m = 3$$
$$\tilde{z} = -2a - 3\mu b + (\mu a + b)\tau \text{ for } m = 4 \qquad \tilde{z} = -a - 3\mu b + (\mu a + 2b)\tau \text{ for } m = 5$$

for $k \in \mathbb{N}$ *and* $k \equiv m \mod 6$.

Proof. It is clear that $\zeta^k = \zeta^{k \pmod 6}$ because ζ is a sixth root of unity. Because $\zeta = 2 - \mu\tau$,

$$(a + b\tau)(2 - \mu\tau) = 2a + 2\tau b - \mu\tau a - \mu\tau^2 b$$

and with $\tau^2 = 3\mu\tau - 3$ the result follows for $m = 1$. The other results can be computed in the same way by simply multiplying $(2 - \mu\tau)$. □

Another computation that has to be executed more than once is the reduction from an element $z = \alpha_{w-1}\tau^{w-1} + \alpha_{w-2}\tau^{w-2} + \ldots + \alpha_1\tau + \alpha_0$ to the shortest possible form $z = a + b\tau$. This has to be done for both efficiency reasons and for the possibility of reducing z modulo τ^w. A way in doing this is to go via recursions. For this purpose, the following well-known theorem, see for example [11], can be used:

Theorem 5.2. *Let $(a_n)_{n\in\mathbb{N}}$ be a recursion with*

$$a_n = \sum_{j=1}^{d} \alpha_j a_{n-j} + \sum_{j=1}^{k} p_j(n)\beta_j^n, \qquad n \geq j, n \in \mathbb{N},$$

where $d \in \mathbb{N}$, $\alpha_j \in \mathbb{R}$, $j = 1, 2, \ldots, d$, $\alpha_d \neq 0$, $p_j(n)$ are polynomials of degree d_j, $k \in \mathbb{N}_0$ and $\beta_j \in \mathbb{R}$. Moreover let $\lambda_1, \lambda_2, \ldots, \lambda_j$ be the distinct roots of the polynomial

$$x^d - \sum_{j=1}^{d} \alpha_j x^{d-j},$$

and let $\mu(\lambda_j)$ be the multiplicity of a root λ_j.
Then there are r polynomials $q_1(n), \ldots, q_r(n)$, where $deg(q_j) \leq \mu(\lambda_j) - 1$ and polynomials $R_1(n), \ldots, R_k(n)$ where $deg(R_j) \leq d_j$, such that

$$a_n = \sum_{j=1}^{r} q_j(n)\lambda_j^n + \sum_{j=1}^{k} n^{\mu(\beta_j)}\beta_j^n R_j(n) \quad .$$

We obtain the following theorem:

Theorem 5.3. *Let $z = \alpha_{w-1}\tau^{w-1} + \alpha_{w-2}\tau^{w-2} + \ldots + \alpha_1\tau + \alpha_0$ be in $\mathbb{Z}[\tau]$, $\alpha_j \in \mathbb{Z}$, and let $\mu = 1$. Define*

$$\begin{aligned} f(n) := \frac{1}{3\cdot 2^{n+1}}\Big(2i\sqrt{3}((3 - i\sqrt{3})^n - (3 + i\sqrt{3})^n)\alpha_{w-2} + 3\Big(((1 + i\sqrt{3})(3 - i\sqrt{3})^n - \\ (1 - i\sqrt{3})(3 + i\sqrt{3})^n)\alpha_{w-1} + 2\sum_{j=0}^{n-1}\Big[-\frac{1}{2^{j+1}}3^{-\frac{3j}{2}}\Big((3 - i\sqrt{3})^n(\sqrt{3} - 3i)(3 + i\sqrt{3})^j \\ + (3 + i\sqrt{3})^n(\sqrt{3} + 3i)(3 - i\sqrt{3})^j\Big)\alpha_{w-3-j}\Big]\Big)\Big) \quad . \end{aligned}$$

Then $a = f(w - 2)$, $b = -3f(w - 3) + \alpha_0$, a and b are both integers and $z = a + b\tau$.

Proof. We know that $\tau^2 = 3\tau - 3$, which implies that $\tau^m = 3\tau^{m-1} - 3\tau^{m-2}$ for any $m \in \mathbb{N}$ which is greater or equal 2. Thus z can be written as

$$\begin{aligned} z &= \alpha_{w-1}(3\tau^{w-2} - 3\tau^{w-3}) + \alpha_{w-2}\tau^{w-2} + \ldots + \alpha_1\tau + \alpha_0 = \\ &= (3\alpha_{w-1} + \alpha_{w-2})\tau^{w-2} + (-3\alpha_{w-1} + \alpha_{w-3})\tau^{w-3} + \alpha_{w-4}\tau^{w-4} + \ldots + \alpha_1\tau + \alpha_0 \quad . \end{aligned}$$

Now let $f(n)$ be the coefficient of the term with τ^{w-1-n} for $z = \sum_{j=1}^{w-1-n} \tilde{\alpha}_j \tau^j$. Then, as we can see above, $f(n)$ is given by the recursion

$$\begin{aligned} f(0) &= \alpha_{w-1} \\ f(1) &= 3\alpha_{w-1} + \alpha_{w-2} \\ f(n) &= 3f(n-1) - 3f(n-2) + \alpha_{w-3-n} \quad \forall n \geq 2. \end{aligned}$$

The theorem follows directly with the recursive representation of $f(n)$ and the general theorem about recursions. □

Remark. Theorem 5.3 assumes that μ has to be one. This is because the formula in this theorem gets much more complicated with μ as a variable. With $\mu = -1$, a very similar result can be obtained in the same way as above.
However, the theorem above seems to be useful in theory at most. For most practical purposes, the formulas for $a + b\tau$ has the disadvantages in both being too long and numerical risky. Because of this, there is always the opportunity in precomputing formulas for transforming $z = \alpha_k\tau^k + \alpha_{k-1}\tau^{k-1} + \ldots + \alpha_1\tau + \alpha_0$ in a shorter form for several $n \in \mathbb{N}$ simply with the recursion and not the formula of theorem 5.3.
Another way in transforming z to a shorter form is to evaluate $\tau^k = c + d\tau$ for all $k \in \{2, 3, \ldots, w-1\}$. Both approaches are demonstrated in the following example:

Example 5.1. When building the transducer for $w = 5$ with $\mu = 1$, the only $z \in \mathbb{Z}[\tau]$ that appear have the form

$$z = \alpha_4\tau^4 + \alpha_3\tau^3 + \alpha_2\tau^2 + \alpha_1\tau^1 + \alpha_0,$$

where $\alpha_j \in \mathcal{D}_w$ for $j \in \{1, 2, 3, 4\}$ and $\alpha_0 \in \mathcal{D}_w \backslash \{0\}$. So the formulas that have to be precomputed are

$$\begin{aligned} z &= \alpha_4\tau^4 + \ldots + \alpha_0 = (\alpha_0 - 3\alpha_2 - 9\alpha_3 - 18\alpha_4) + (\alpha_1 + 3\alpha_2 + 6\alpha_3 + 9\alpha_4)\tau \\ z &= \alpha_3\tau^3 + \ldots + \alpha_0 = (\alpha_0 - 3\alpha_2 - 9\alpha_3) + (\alpha_1 + 3\alpha_2 + 6\alpha_3)\tau \\ z &= \alpha_2\tau^2 + \alpha_1\tau + \alpha_0 = (\alpha_0 - 3\alpha_2) + (\alpha_1 + 3\alpha_2)\tau \end{aligned}$$

Next we compute τ^k for $k \in \{2, 3, 4\}$:

$$\tau^2 = -3 + 3\tau \qquad \tau^3 = -9 + 6\tau \qquad \tau^4 = -18 + 9\tau$$

This leads to the same formulas for z as above.

6 Results

Here some numerical results from evaluations that have been made are listed. First of all the numbers of digits for digit sets $\mathcal{D}_w$:

w	2	3	4	5	6	7
$\lvert\mathcal{D}_w\rvert$	7	19	55	163	487	1459

Table 1: Size of digit set $\mathcal{D}_w$

Here we can see that the formula

$$|\mathcal{D}_w| = 6 \cdot 3^{w-2} + 1$$

holds.
The next table shows the size of each level of the transducer $\mathscr{T}$, and the number of states of the whole transducer. Here we can observe that the number of states for a specific level rises by a factor of ≈ 3 for growing w (except $w = 2$). The number of states for $\mathscr{T}_w$, which denotes the transducer for a certain w, seems to be

$$\mathscr{T}_w \approx 10\mathscr{T}_{w-1} \quad .$$

Level	start	Δ	0	1	2	3	4	5	$\sum$
$w = 2$	1	6	12	-	-	-	-	-	19
$w = 3$	1	60	102	282	-	-	-	-	445
$w = 4$	1	150	282	792	2280	-	-	-	3505
$w = 5$	1	492	942	2766	8172	24330	-	-	367043
$w = 6$	1	1428	2784	8190	24258	72300	215976	-	324937
$w = 7$	1	4404	8658	25794	76956	230340	689700	2067540	3103393

Table 2: Number of states of transducer $\mathscr{T}$

Every node of $G(\mathscr{T})$ has $|\mathcal{D}_w|$ outgoing edges, which leads to

$$|E(\mathscr{T}_7)| = 3103393 \cdot 1459 = 4,528 \cdot 10^9.$$

The transducer $\mathscr{T}_8$ has not been built yet because of storage reasons. If the factor of the number of states from $\mathscr{T}_w$ to $\mathscr{T}_{w+1}$ is approximative 10, we can make a guess for the number of edges for $\mathscr{T}_8$, because we know that $|\mathcal{D}_8| = 4375$:

$$|E(\mathscr{T}_8)| \approx 31000000 \cdot 4375 = 1,356 \cdot 10^{11}.$$

The transducer and the relating graph $G(\mathscr{T})$ were built for $w \in \{2, 3, 4, 5, 6, 7\}$. There are no non-negative paths from the start- to the end states in these graphs, which proves the following theorem:

Theorem 6.1. *The $\mathcal{D}$-w-NAF is optimal for $w \in \{2, 3, 4, 5, 6, 7\}$.*

References

[1] Roberto M. Avanzi, Clemens Heuberger, and Helmut Prodinger. Arithmetic of Supersingular Koblitz Curves in Characteristic Three. Technical Report 2010-8, Graz University of Technology, 2010. `http://www.math.tugraz.at/fosp/pdfs/tugraz_0166.pdf`.

[2] Ian F. Blake, V. Kumar Murty, and Guangwu Xu. Efficient algorithms for Koblitz curves over fields of characteristic three. *Discrete Algorithms 3*, 1294:113–124, 2005.

[3] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 3rd edition, 2009.

[4] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete mathematics*. Addison-Wesley, Reading, MA, 1989.

[5] C. Heuberger. Redundant τ-Adic Expansions II: Non-Optimality and Chaotic Behaviour. *Mathematics in Computer Science*, 3:141–157, 2010.

[6] Donald E. Knuth. *The Art of Computer Programming*, volume 2. Addison-Wesley Longman Publishing Co., third edition, 1997.

[7] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48:203–209, 1987.

[8] Neal Koblitz. An elliptic curve implementation of the finite field digital signature algorithm. *Lecture Notes in Computer Science*, 1462:327–337, 1998.

[9] Victor S. Miller. Use of Elliptic Curves in Cryptography. *Lecture Notes in Computer Science*, 218:417–426, 1985.

[10] B. Phillips and N. Burgess. Minimal weight digit set conversions. *IEEE Trans. Comput.*, 53:666–677, 2004.

[11] Keneth H. Rosen. *Discrete Mathematics and its Applications*. McGraw-Hill, fourth edition, 1999. Chinese Edition.

[12] J.A Solinas. An improved algorithm for arithmetic on a family of elliptic curves. *Advances in Cryptology - CRYPTO '97*, 1294:357–371, 1997.

[13] J.A Solinas. Efficient arithmetic on Koblitz curves. *Des. Codes Cryptography*, 19:195–249, 2000.