Contents

List of symbols

Symbol	Description
$\mathbb{N}$	set of nonnegative integers $\{0, 1, 2, 3, \dots\}$
$\mathbb{Z}$	set of integers $\{\dots, -2, -1, 0, 1, 2, \dots\}$
$\mathbb{R}$	set of real numbers
$\mathbb{C}$	set of complex numbers
$\mathbb{Q}$	set of rational numbers
$\mathbb{Q}(\beta)$	the smallest field containing the set $\mathbb{Q}$ and algebraic number β
$\#S$	number of elements of the finite set S
C^*	complex conjugation and transposition of the complex matrix C
m_β	monic minimal polynomial of the algebraic number β
$\deg \beta$	degree of the algebraic number β (over $\mathbb{Q}$)
$(\beta, \mathcal{A})$	numeration system with the base β and the alphabet $\mathcal{A}$
$(x)_{\beta,\mathcal{A}}$	$(\beta, \mathcal{A})$-representation of the number x
$\mathrm{Fin}_{\mathcal{A}}(\beta)$	set of all complex numbers with a finite $(\beta, \mathcal{A})$-representation
$\mathcal{A}^{\mathbb{Z}}$	set of all bi-infinite sequences of digits in $\mathcal{A}$
$\mathbb{Z}[\omega]$	the smallest ring containing $\mathbb{Z}$ and ω
π	isomorphism from $\mathbb{Z}[\omega]$ to $\mathbb{Z}^d$ $(d = \deg \omega)$
$\mathcal{B}$	alphabet of input digits
q_j	weight coefficient for the j-th position
$\mathcal{Q}$	weight coefficients set
$\mathcal{Q}_{[w_j,\dots,w_{j-m+1}]}$	set of possible weight coefficients for the input digits $w_j, \dots, w_{j-m+1}$
$\lfloor x \rfloor$	floor function of the number x
$\mathrm{Re}\, x$	real part of the complex number x
$\mathrm{Im}\, x$	imaginary part of the complex number x

Chapter 1

Preliminaries

1.1 Numeration systems

In the binary system, numbers are expressed as a sum of powers of 2 multiplied by 0 or 1. The following definition generalizes this concept.

Definition 1.1. Let $\beta \in \mathbb{C}, |\beta| > 1$ and $\mathcal{A} \subset \mathbb{C}$ be a finite set containing 0. A pair $(\beta, \mathcal{A})$ is called a *positional numeration system* with *base* β and *digit set* $\mathcal{A}$, usually called *alphabet*.

Sometimes, the term *radix* is used instead of base. So-called standard numeration systems have an integer base β and an alphabet $\mathcal{A}$ which is a set of contiguous integers. We restrict ourselves to a base β which is an algebraic integer and possibly non-integer alphabet $\mathcal{A}$.

Definition 1.2. Let $(\beta, \mathcal{A})$ be a positional numeration system. We say that a complex number x has a $(\beta, \mathcal{A})$-*representation* if there exist digits $x_n, x_{n-1}, x_{n-2}, \ldots \in \mathcal{A}, n \geq 0$ such that $x = \sum_{j=-\infty}^{n} x_j \beta^j$.

We write briefly a *representation* instead of a $(\beta, \mathcal{A})$-representation if the base β and alphabet $\mathcal{A}$ follow from context. The assumption $|\beta| > 1$ implies that the sum for a given representation converges.

Definition 1.3. Let $(\beta, \mathcal{A})$ be a positional numeration system. The set of all complex numbers with a finite $(\beta, \mathcal{A})$-representation is defined by

$$\mathrm{Fin}_{\mathcal{A}}(\beta) := \left\{ \sum_{j=-m}^{n} x_j \beta^j : n, m \in \mathbb{N}, x_j \in \mathcal{A} \right\}.$$

For $x \in \mathrm{Fin}_{\mathcal{A}}(\beta)$, we write

$$(x)_{\beta,\mathcal{A}} = {}^{\omega}0 x_n x_{n-1} \cdots x_1 x_0 \bullet x_{-1} x_{-2} \cdots x_{-m} 0^{\omega},$$

where ${}^{\omega}0$ and 0^{ω} denotes left and right-infinite sequence of zeros. From now on, we omit the starting and ending 0^{ω} when we work with numbers in $\mathrm{Fin}_{\mathcal{A}}(\beta)$, but we still consider a representaton as a bi-infinite sequence of digits. Note that indices are decreasing from left to right as it is usual to write the most significant digits first.

We remark that the set $\mathrm{Fin}_{\mathcal{A}}(\beta)$ is not necessarily closed under addition. Nevertheless, existence of a parallel addition algorithm in the sense of definitions in the next section implies that the set $\mathrm{Fin}_{\mathcal{A}}(\beta)$ is closed under addition, i.e.,

$$\mathrm{Fin}_{\mathcal{A}}(\beta) + \mathrm{Fin}_{\mathcal{A}}(\beta) \subset \mathrm{Fin}_{\mathcal{A}}(\beta)\,.$$

Designing an algorithm for parallel addition requires some redundancy in numeration system [7]. According to [9], a numeration system $(\beta, \mathcal{A})$ is called *redundant* if there exists $x \in \mathrm{Fin}_{\mathcal{A}}(\beta)$ which has two different $(\beta, \mathcal{A})$-representations. For instance, the number 1 has $(2, \{-1, 0, 1\})$-representations $1\bullet$ and $1(-1)\bullet$. Redundant numeration system may allow to avoid carry propagation in addition. On the other hand, redundancy bring some disadvantages. For example, comparison is problematic.

1.2 Parallel addition

Informally, parallel algorithm means that there are many threads or processes which run at the same time. Usually, the main task is to reduce communication among processes to minimum, since waiting for a result of another process slows down the whole computation. In the scope of parallel addition, parallelism is formalized by a local function, which is also often called a sliding block code.

Definition 1.4. Let $\mathcal{A}$ and $\mathcal{B}$ be alphabets. A function $\varphi : \mathcal{B}^{\mathbb{Z}} \to \mathcal{A}^{\mathbb{Z}}$ is said to be *p-local* if there exist $r, t \in \mathbb{N}$ satisfying $p = r + t + 1$ and a function $\phi : \mathcal{B}^p \to \mathcal{A}$ such that, for any $w = (w_j)_{j\in\mathbb{Z}} \in \mathcal{B}^{\mathbb{Z}}$ and its image $z = \varphi(w) = (z_j)_{j\in\mathbb{Z}} \in \mathcal{A}^{\mathbb{Z}}$, we have $z_j = \phi(w_{j+t}, \cdots, w_{j-r})$ for every $j \in \mathbb{Z}$. The parameter t, resp. r, is called *anticipation*, resp. *memory*.

This means that each digit of the image $\varphi(w)$ is computed from p digits of w in a sliding window.

Since two $(\beta, \mathcal{A})$-representations may be easily summed up digitwise in parallel, the crucial point of parallel addition is conversion of a $(\beta, \mathcal{A} + \mathcal{A})$-representation of the sum to a $(\beta, \mathcal{A})$-representation. The notion of a p-local function is applied to this conversion.

Definition 1.5. Let β be a base and let $\mathcal{A}$ and $\mathcal{B}$ be alphabets containing 0. A function $\varphi : \mathcal{B}^{\mathbb{Z}} \to \mathcal{A}^{\mathbb{Z}}$ such that

i) for any $w = (w_j)_{j\in\mathbb{Z}} \in \mathcal{B}^{\mathbb{Z}}$ with finitely many non-zero digits, $z = \varphi(w) = (z_j)_{j\in\mathbb{Z}} \in \mathcal{A}^{\mathbb{Z}}$ has only finite number of non-zero digits, and

ii) $\sum_{j\in\mathbb{Z}} w_j\beta^j = \sum_{j\in\mathbb{Z}} z_j\beta^j$

is called *digit set conversion* in the base β from $\mathcal{B}$ to $\mathcal{A}$. Such a conversion φ is said to be *computable in parallel* if φ is a p-local function for some $p \in \mathbb{N}$.

Suppose that there is a processing unit on each position with access to t input digits on the left and r input digits on the right. Then computation of $\varphi(w)$, where w has only finitely many non-zero digits, can be done in a constant time independent on the number of non-zeros in the sequence w.

We recall few results about parallel addition in a numeration system with an integer alphabet. C. Frougny, E. Pelantová and M. Svobodová proved the following sufficient condition of existence of an algorithm for parallel addition in [4].

Theorem 1.1. *Let $\beta \in \mathbb{C}$ be an algebraic number such that $|\beta| > 1$ and all its conjugates in modulus differ from 1. There exists an alphabet $\mathcal{A}$ of contiguous integers containing 0 such that addition on $Fin_{\mathcal{A}}(\beta)$ can be performed in parallel.*

An algorithm for an alphabet of the form $\{-a, -a+1, \ldots, 0, \ldots, a-1, a\}$ is provided in the proof, but in general, a is not minimal.

The same authors showed in [3] that the condition on the conjugates of the base β is also necessary:

Theorem 1.2. *Let the base $\beta \in \mathbb{C}, |\beta| > 1$, be an algebraic number with a conjugate β' such that $|\beta'| = 1$. Let $\mathcal{A} \subset \mathbb{Z}$ be an alphabet of contiguous integers containing 0. Then addition on $Fin_{\mathcal{A}}(\beta)$ cannot be computable in parallel.*

We see later that absolute value of conjugates of a base significantly influences also a method in the focus of this thesis.

A lower bound on the size of an integer alphabet is provided in [5].

Theorem 1.3. *Let $\beta \in \mathbb{C}, |\beta| > 1$, be an algebraic integer with the minimal polynomial m_β. Let $\mathcal{A} \subset \mathbb{Z}$ be an alphabet of contiguous integers containing 0 and 1. If addition on $Fin_{\mathcal{A}}(\beta)$ is computable in parallel, then $\#\mathcal{A} \geq |m_\beta(1)|$. Moreover, if β is a positive real number, $\beta > 1$, then $\#\mathcal{A} \geq |m_\beta(1)| + 2$.*

In Section 4.3, we prove the same bound for a larger class of alphabets. The most general alphabets, which are considered in this thesis, are finite subsets of the set from the next definition. For our purposes, a ring is commutative, associative under multiplication and there is a multiplicative identity.

Definition 1.6. Let ω be a complex number. The smallest ring which contains integers $\mathbb{Z}$ and ω is denoted by

$$\mathbb{Z}[\omega] = \left\{ \sum_{i=0}^{n} a_i \omega^i \colon n \in \mathbb{N}, a_i \in \mathbb{Z} \right\}.$$

In other words, the set $\mathbb{Z}[\omega]$ are all polynomials with integer coefficients evaluated in ω. Obviously, it is a subset of the field extension $\mathbb{Q}(\omega)$.

From now on, let ω be an algebraic integer which generates the set $\mathbb{Z}[\omega]$ and let $\beta \in \mathbb{Z}[\omega]$ be a base, i.e., $|\beta| > 1$. We remark that β is also an algebraic integer as all elements of $\mathbb{Z}[\omega]$ are algebraic integers. Finally, let $\mathcal{A} \subset \mathbb{Z}[\omega]$ be an alphabet, it means that it is finite set containing 0.

Few parallel addition algorithms for such numeration system with a non-integer alphabet were found ad hoc. We introduce the method for construction of a parallel addition algorithm for a given numeration system $(\beta, \mathcal{A})$ in Chapter **??**.

Chapter 2

Design of extending window method

We recall the general concept of addition at the beginning of this chapter and then we describe a so-called *extending window method* which is due to M. Svobodová [10]. The method attempts to construct a conversion algorithm computable in parallel for a given numeration system $(\beta, \mathcal{A})$. We recall that ω is an algebraic integer, $\beta \in \mathbb{Z}[\omega]$ is a base and $0 \in \mathcal{A} \subset \mathbb{Z}[\omega]$ is an alphabet.

2.1 Addition

The general idea of addition (standard or parallel) in any numeration system $(\beta, \mathcal{A})$ is the following: we sum up two numbers digitwise and then we convert the result with digits in $\mathcal{A}+\mathcal{A}$ into the alphabet $\mathcal{A}$. Obviously, digitwise addition is computable in parallel, thus the problematic part is the digit set conversion of the obtained result. It can be easily done in a standard way but a parallel digit set conversion is nontrivial. Parallel conversion is based on the same formulas as standard one but the choice of so called *weight coefficients* differs.

Now we go step by step more precisely. Let $x, y \in \mathrm{Fin}_{\mathcal{A}}(\beta)$ with $(\beta, \mathcal{A})$-representations $x_{n'}x_{n'-1}\cdots x_1x_0 \bullet x_{-1}x_{-2}\cdots x_{-m'}$ and $y_{n'}y_{n'-1}\cdots y_1y_0 \bullet y_{-1}y_{-2}\cdots y_{-m'}$ padded by zeros to have the same length. We set

$$\begin{aligned} w = x + y &= \sum_{i=-m'}^{n'} x_i\beta^i + \sum_{i=-m'}^{n'} y_i\beta^i = \sum_{i=-m'}^{n'} (x_i + y_i)\beta^i \\ &= \sum_{i=-m'}^{n'} w_i\beta^i\,, \end{aligned}$$

where $w_i = x_i + y_i \in \mathcal{A}+\mathcal{A}$. Thus, $w_{n'}w_{n'-1}\cdots w_1w_0 \bullet w_{-1}w_{-2}\cdots w_{-m'}$ is a $(\beta, \mathcal{A}+\mathcal{A})$-representation of $w \in \mathrm{Fin}_{\mathcal{A}+\mathcal{A}}(\beta)$.

We also use column notation for digitwise addition in what follows, e.g.,

$$\begin{array}{l} x_{n'}\ x_{n'-1}\cdots x_1\ x_0 \bullet x_{-1}\ x_{-2}\ \cdots x_{-m'} \\ y_{n'}\ y_{n'-1}\cdots y_1\ \ y_0 \bullet y_{-1}\ y_{-2}\ \ \cdots y_{-m'} \\ \hline w_{n'}w_{n'-1}\cdots w_1w_0 \bullet w_{-1}w_{-2}\cdots w_{-m'}\,. \end{array}$$

We search for a $(\beta, \mathcal{A})$-representation of w, i.e., a sequence

$$z_n z_{n-1} \cdots z_1 z_0 z_{-1} z_{-2} \cdots z_{-m}$$

such that $z_j \in \mathcal{A}$ and

$$z_n z_{n-1} \cdots z_1 z_0 \bullet z_{-1} z_{-2} \cdots z_{-m} = (w)_{\beta,\mathcal{A}} .$$

Note that the index of the first, resp. last, non-zero digit of the converted representation $z_n z_{n-1} \cdots z_1 z_0 \bullet z_{-1} z_{-2} \cdots z_{-m} = (w)_{\beta,\mathcal{A}}$ may differ from the original representation $w_{n'} w_{n'-1} \cdots w_1 w_0 \bullet w_{-1} w_{-2} \cdots w_{-m'}$. We assume that $n \geq n'$ and $m \geq m'$, otherwise we pad the converted representation by zeros.

Without lost of generality, we consider only conversion of so called β-integers – numbers from $\mathrm{Fin}_{\mathcal{A}+\mathcal{A}}(\beta)$ whose representations have all digits with negative indices equal zero. Modification for general representation $w_{n'} w_{n-1} \cdots w_1 w_0 \bullet w_{-1} w_{-2} \cdots w_{-m'}$ is obvious:

$$\beta^m \cdot w_{n'} w_{n'-1} \cdots w_1 w_0 \bullet w_{-1} w_{-2} \cdots w_{-m'} = w_n w_{n'-1} \cdots w_1 w_0 w_{-1} w_{-2} \cdots w_{-m'} \bullet$$

and after conversion

$$z_n z_{n-1} \cdots z_1 z_0 z_{-1} z_{-2} \cdots z_{-m'} \bullet \cdots z_{-m} = \beta^m \cdot z_n z_{n-1} \cdots z_1 z_0 \bullet z_{-1} z_{-2} \cdots z_{-m'} .$$

Digits w_j are converted into the alphabet $\mathcal{A}$ by adding a suitable representation of zero digitwise. For our purpose, we use the simplest possible representation which is deduced from the polynomial

$$x - \beta \in (\mathbb{Z}[\omega])\,[x] .$$

We remark that any polynomial $R(x) = r_s x^s + r_{s-1} x^{s-1} + \cdots + r_1 x + r_0$ with coefficients $r_i \in \mathbb{Z}[\omega]$ such that $R(\beta) = 0$ gives us a possible representation of zero. The polynomial R is called a *rewriting rule*. One of the coefficients of R which is greatest in modulus (so-called *core coefficient*) is used for the conversion of a digit w_j. Nevertheless, the extending window method is strongly dependent on the rewriting rule, so we focus only on the simplest possible rewriting rule $R(x) = x - \beta$. Usage of an arbitrary rewriting rule R is out of scope of this thesis.

As $0 = \beta^j \cdot R(\beta) = 1 \cdot \beta^{j+1} - \beta \cdot \beta^j$, we have a representation of zero

$$1(-\beta)\underbrace{0 \cdots 0}_{j}\bullet = (0)_\beta .$$

for all $j \in \mathbb{N}$. We multiply this representation by $q_j \in \mathbb{Z}[\omega]$, which is called a *weight coefficient*, to obtain another representation of zero $q_j(-q_j\beta)0 \cdots 0\bullet = (0)_\beta$. This is digitwise added to $w_n w_{n-1} \cdots w_1 w_0 \bullet$ to convert the digit w_j into the alphabet $\mathcal{A}$. The conversion of j-th digit causes a *carry* q_j on the $(j+1)$-th position.

In standard addition, the digit set conversion runs from the right ($j = 0$) to the left until all non-zero digits and carries are converted into the alphabet $\mathcal{A}$:

$$\begin{array}{ccccccccccccc}
w_{n'} & w_{n'-1} & \cdots & w_{j+1} & w_j & w_{j-1} & \cdots & w_1 & w_0 & \bullet & = & (w)_{\beta,\mathcal{A}+\mathcal{A}} \\
 & & & & & q_{j-2} & \iddots & & & & & \\
 & & & & q_{j-1} & -\beta q_{j-1} & & & & & & \\
 & & & \iddots & & & & & & & & \\
 & & & q_j & -\beta q_j & & & & & & & \\
\hline
z_n \cdots z_{n'} & z_{n'-1} & \cdots & z_{j+1} & z_j & z_{j-1} & \cdots & z_1 & z_0 & \bullet & = & (w)_{\beta,\mathcal{A}}
\end{array}$$

Hence, the desired formula for conversion on the j-th position is

$$z_j = w_j + q_{j-1} - q_j\beta$$

for $j \in \mathbb{N}$. We set $q_{-1} = 0$ as there is no carry from the right on the 0-th position. To clarify, the terms carry and weight coefficient are related to a position. While q_{j-1} is a carry from the right and q_j is a chosen weight coefficient on the j-th position, q_j is a carry from the right on the $(j+1)$-th position etc.

We remark that the conversion with rewriting rule $x - \beta$ prolongs the part of non-zero digits only to the left as there is no carry from the left. Thus, all digits with negative indices of the converted sequence are zero.

The fact that the conversion preserves the value of w follows from adding a representation of zero or we may formally verified it.

$$\begin{aligned}\sum_{j\geq 0} z_j\beta^j &= w_0 - \beta q_0 + \sum_{j>0}(w_j + q_{j-1} - q_j\beta)\beta^j \\ &= \sum_{j\geq 0} w_j\beta^j + \sum_{j>0} q_{j-1}\beta^j - \sum_{j\geq 0} q_j \cdot \beta^{j+1} \\ &= \sum_{j\geq 0} w_j\beta^j + \sum_{j>0} q_{j-1}\beta^j - \sum_{j>0} q_{j-1} \cdot \beta^j \\ &= \sum_{j\geq 0} w_j\beta^j = w\,.\end{aligned} \tag{2.1}$$

The weight coefficient q_j must be chosen so that the converted digit is in the alphabet $\mathcal{A}$, i.e.,

$$z_j = w_j + q_{j-1} - q_j\beta \in \mathcal{A}\,. \tag{2.2}$$

The choice of weight coefficients is a crucial part of construction of addition algorithms which are computable in parallel. The extending window method determining weight coefficients for a given input is described in Section 2.2.

On the other hand, the following example shows that determining weight coefficients is trivial for standard numeration systems.

Example 2.1. Assume now a standard numeration system $(\beta, \mathcal{A})$, where

$$\beta \in \mathbb{N}\,, \beta \geq 2\,, \mathcal{A} = \{0, 1, 2, \ldots, \beta - 1\}\,.$$

Notice that

$$z_j \equiv w_j + q_{j-1} \mod \beta\,.$$

There is only one representative of each class modulo β in the standard numeration system $(\beta, \mathcal{A})$. Therefore, the digit z_j is uniquely determined for a given digit $w_j \in \mathcal{A} + \mathcal{A}$ and carry q_{j-1} and thus so is the weight coefficient q_j. This means that $q_j = q_j(w_j, q_{j-1})$ for all $j \geq 0$. Generally,

$$q_j = q_j(w_j, q_{j-1}(w_{j-1}, q_{j-2})) = \cdots = q_j(w_j, \ldots, w_1, w_0)$$

and

$$z_j = z_j(w_j, \ldots, w_1, w_0)\,,$$

which implies that addition runs in linear time.

We require that the digit set conversion from $\mathcal{A}+\mathcal{A}$ into $\mathcal{A}$ is computable in parallel, i.e., there exist constants $r, t \in \mathbb{N}_0$ such that for all $j \geq 0$ is $z_j = z_j(w_{j+t}, \ldots, w_{j-r})$. Anticipation t equals zero since we use the rewriting rule $x - \beta$. To avoid the dependency on all less significant digits, we need variety in the choice of weight coefficient q_j. This implies that the used numeration system must be redundant.

2.2 Extending window method

In order to construct a digit set conversion in numeration system $(\beta, \mathcal{A})$ which is computable in parallel, we consider a more general case of digit set conversion from an *input alphabet* $\mathcal{B}$ such that $\mathcal{A} \subsetneq \mathcal{B} \subset \mathcal{A}+\mathcal{A}$ instead of the alphabet $\mathcal{A}+\mathcal{A}$. As mentioned above, the key problem is to find for every $j \geq 0$ a weight coefficient q_j such that

$$z_j = \underbrace{w_j}_{\in \mathcal{B}} + q_{j-1} - q_j\beta \in \mathcal{A}$$

for any input $w_{n'}w_{n'-1}\ldots w_1w_0\bullet = (w)_{\beta,\mathcal{B}}, w \in \mathrm{Fin}_{\mathcal{B}}(\beta)$. We remark that the weight coefficient q_{j-1} is determined by the input $w_{j-1}\ldots w_1w_0\bullet$. For a digit set conversion with the rewriting rule $x - \beta$ to be computable in parallel, the digit z_j is required to satisfy $z_j = z_j(w_j, \ldots, w_{j-r})$ for a fixed memory r in $\mathbb{N}$.

Note that the digit z_j is given by the input digit w_j and carry q_{j-1} which is determined by input digits $w_{j-1}w_{j-2}\ldots$. Thus, if we find a weight coefficient q_j all possible combinations of input digits $w_jw_{j-1}w_{j-2}\ldots$, then the position j is not important. Therefore, we may strongly simplify our notation if we omit j in subscripts. From now on, $w_0 \in \mathcal{B}$ is a converted digit, $w_{-1}w_{-2}\cdots \in \mathcal{B}$ are digits on right, $q_{-1} \in \mathbb{Z}[\omega]$ is a carry from the right and we search for a weight coefficient $q_0 \in \mathbb{Z}[\omega]$ such that

$$z_0 = w_0 + q_{-1} - q_0\beta \in \mathcal{A}\,.$$

We introduce two definitions before we describe the extending window method.

Definition 2.1. Let $\mathcal{B}$ be a set such that $\mathcal{A} \subsetneq \mathcal{B} \subset \mathcal{A}+\mathcal{A}$. Then any finite set $\mathcal{Q} \subset \mathbb{Z}[\omega]$ containing 0 such that

$$\mathcal{B} + \mathcal{Q} \subset \mathcal{A} + \beta\mathcal{Q}$$

is called a *weight coefficients set*.

We see that if $\mathcal{Q}$ is a weight coefficients set, then

$$(\forall w_0 \in \mathcal{B})(\forall q_{-1} \in \mathcal{Q})(\exists q_0 \in \mathcal{Q})(\underbrace{w_0 + q_{-1} - q_0\beta}_{z_0} \in \mathcal{A})\,.$$

In other words, there is a weight coefficient $q_0 \in \mathcal{Q}$ for a carry from the right $q_{-1} \in \mathcal{Q}$ and a digit w_0 in the input alphabet $\mathcal{B}$ such that z_0 is in the alphabet $\mathcal{A}$. Notice that the carry from the right for the right most non-zero digit of a converted sequence which is 0 is in $\mathcal{Q}$ by the definition.

Definition 2.2. Let r be an integer and $q : \mathcal{B}^r \to \mathcal{Q}$ be a mapping such that

$$w_0 + q(w_{-1}, \ldots, w_{-r}) - \beta q(w_0, \ldots, w_{-(r-1)}) \in \mathcal{A}$$

for all $w_0, w_{-1}, \ldots, w_{-r} \in \mathcal{B}$, and $q(0, 0, \ldots, 0) = 0$. Then q is called a *weight function* and r is called a *length of window*.

Having a weight function q, we define a function $\phi : \mathcal{B}^{r+1} \to \mathcal{A}$ by

$$\phi(w_0, \ldots, w_{-r}) = w_0 + \underbrace{q(w_{-1}, \ldots, w_{-r})}_{=q_{-1}} - \beta \underbrace{q(w_0, \ldots, w_{-(r-1)})}_{=q_0} =: z_0 \,, \tag{2.3}$$

which verifies that the digit set conversion is indeed a $(r+1)$-local function and memory r. The requirement of zero output of the weight function q for the input of r zeros guarantees that $\phi(0, 0, \ldots, 0) = 0$. Thus, the first condition of Definition 1.5 is satisfied. The second one follows from the equation (2.1).

Let us summarize the construction of the digit set conversion by the rewriting rule $x - \beta$. We need to find weight coefficients for all possible combinations of digits of the input alphabet $\mathcal{B}$. The rewriting rules multiplied by the weight coefficients are digitwise added to an input sequence. In fact, it means that the equation (2.2) is applied on each position. If the digit set conversion is computable in parallel, the weight coefficients are determined as the outputs of the weight function q with some fixed length of window r.

We search for a weight function q for a given base β and input alphabet $\mathcal{B}$ by the extending window method. It consists of two phases. First, we find some weight coefficients set $\mathcal{Q}$. We know that it is possible to convert an input sequence by choosing the weight coefficients from the set $\mathcal{Q}$. The set $\mathcal{Q}$ serves as the starting point for the second phase in which we increment the expected length of the window r until the weight function q is uniquely defined for each $(w_0, \ldots, w_{-(r-1)}) \in \mathcal{B}^r$. Then, the local conversion is determined – we use the weight function outputs as weight coefficients in the formula (2.3).

We describe the general concept of the extending window method in this chapter, while various possibilities of construction of sets during both phases are discussed in Chapter **??**. Note that convergence of both phases is studied in Chapter 4.

2.3 Phase 1 – Weight coefficients set

The goal of the first phase is to compute a weight coefficients set $\mathcal{Q}$, i.e., to find a set $\mathcal{Q} \ni 0$ such that

$$\mathcal{B} + \mathcal{Q} \subset \mathcal{A} + \beta\mathcal{Q} \,.$$

We build the sequence $\mathcal{Q}_0, \mathcal{Q}_1, \mathcal{Q}_2, \ldots$ iteratively so that we extend $\mathcal{Q}_k$ to $\mathcal{Q}_{k+1}$ in a way to cover all elements of the set $\mathcal{B} + \mathcal{Q}_k$ by elements of the extended set $\mathcal{Q}_{k+1}$, i.e.,

$$\mathcal{B} + \mathcal{Q}_k \subset \mathcal{A} + \beta\mathcal{Q}_{k+1} \,.$$

This procedure is repeated until the extended weight coefficients set $\mathcal{Q}_{k+1}$ is the same as the original set $\mathcal{Q}_k$. We remark that the expression "a weight coefficient q covers an element x" means that there is $a \in \mathcal{A}$ such that $x = a + \beta q$.

In other words, we start with $\mathcal{Q}_0 = \{0\}$ meaning that we search all weight coefficients q_j necessary for digit set conversion for the case where there is no carry from the right, i.e., $q_{j-1} = 0$. We add them to the weight coefficients set $\mathcal{Q}_0$ to obtain the set $\mathcal{Q}_1$. Assume now that we have the set $\mathcal{Q}_k$ for some $k \geq 1$. The weight coefficients in $\mathcal{Q}_k$ now may appear as a carry q_{j-1}. If there are no suitable weight coefficients q_j in the weight coefficients set $\mathcal{Q}_k$ to cover all sums of added coefficients and digits of the input alphabet $\mathcal{B}$, we extend $\mathcal{Q}_k$ to $\mathcal{Q}_{k+1}$ by suitable coefficients using Algorithm 3. And so on until there is no need to add more

elements, i.e., the extended set $\mathcal{Q}_{k+1}$ equals $\mathcal{Q}_k$. Then the weight coefficients set $\mathcal{Q} := \mathcal{Q}_{k+1}$ satisfies Definition 2.1.

The precise description of the algorithm in a pseudocode is in Algorithm 1. For better understanding, see Figures **??**–**??** in Appendix **??** which illustrate the construction of the weight coefficients set $\mathcal{Q}$ for the Eisenstein base and a complex alphabet (see Example **??** for its description).

Section **??** discusses the convergence of Phase 1, i.e. whether it happens that $\mathcal{Q}_{k+1} = \mathcal{Q}_k$ for some k.

Algorithm 1 Search for weight coefficients set (Phase 1)

1: $k := 0$
2: $Q_0 := \{0\}$
3: **repeat**
4: By Algorithm 3, extend $\mathcal{Q}_k$ to $\mathcal{Q}_{k+1}$ in a minimal possible way so that

$$\mathcal{B} + \mathcal{Q}_k \subset \mathcal{A} + \beta\mathcal{Q}_{k+1}$$

5: $k := k + 1$
6: **until** $\mathcal{Q}_k = \mathcal{Q}_{k+1}$
7: $\mathcal{Q} := \mathcal{Q}_k$
8: **return** $\mathcal{Q}$

2.4 Phase 2 – Weight function

We want to find a length of the window M and a weight function $q : \mathcal{B}^M \to \mathcal{Q}$. We start with the weight coefficients set $\mathcal{Q}$ obtained in Phase 1. The idea is to reduce necessary weight coefficients for the conversion of a given digit up to single value. This is done by enlarging the number of considered input digits (extending the length of window) – there might be less possible carries from the right if we know which digits on the right are converted.

We introduce the following notation. Let $\mathcal{Q}$ be a weight coefficients set and $w_j \in \mathcal{B}$. Denote by $\mathcal{Q}_{[w_j]}$ any set such that

$$(\forall q_{j-1} \in \mathcal{Q})(\exists q_j \in \mathcal{Q}_{[w_j]})(w_j + q_{j-1} - q_j\beta \in \mathcal{A})\,.$$

By induction with respect to $m \in \mathbb{N}, m > 1$, for all $(w_j, \ldots, w_{j-m+1}) \in \mathcal{B}^m$ denote by $\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]}$ any subset of $\mathcal{Q}_{[w_j,\ldots,w_{j-m+2}]}$ such that

$$(\forall q_{j-1} \in \mathcal{Q}_{[w_{j-1},\ldots,w_{j-m+1}]})(\exists q_j \in \mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]})(w_j + q_{j-1} - q_j\beta \in \mathcal{A})\,.$$

Recall the scheme (**??**) of the digit set conversion for better understanding of the notation and method:

$$\begin{array}{cccc}
\cdots w_{j+1} & w_j & w_{j-1} & \cdots w_{j-M+1} w_{j-M} \cdots \\
 & & q_{j-2} & \\
 & q_{j-1} & -\beta q_{j-1} & \\
q_j & -\beta q_j & & \\
-\beta q_{j+1} & & & \\
\hline
\cdots z_{j+1} & z_j & z_{j-1} & \cdots z_{j-M+1}\ z_{j-M} \cdots
\end{array}$$

The idea is to check all possible right carries $q_{j-1} \in \mathcal{Q}$ and determine values $q_j \in \mathcal{Q}$ such that

$$z_j = w_j + q_{j-1} - q_j\beta \in \mathcal{A}\,.$$

So we obtain a set $\mathcal{Q}_{[w_j]} \subset \mathcal{Q}$ of weight coefficients which are necessary to convert the digit w_j with any carry $q_{j-1} \in \mathcal{Q}$. Assuming that we know the input digit w_{j-1}, the set of possible carries from the right is also reduced to $\mathcal{Q}_{[w_{j-1}]}$. Thus we may reduce the set $\mathcal{Q}_{[w_j]}$ to a set $\mathcal{Q}_{[w_j,w_{j-1}]} \subset \mathcal{Q}_{[w_j]}$ which is necessary to cover all elements of $w_j + \mathcal{Q}_{[w_{j-1}]}$. Prolonging the length of window in this manner may lead to a unique weight coefficient q_j for enough given input digits.

Accordingly, the weight function q is found if there is $M \in \mathbb{N}$ such that

$$\#\mathcal{Q}_{[w_j,\ldots,w_{j-M+1}]} = 1$$

for all $w_j, \ldots, w_{j-M+1} \in \mathcal{B}^M$. The precise description of the construction of the weight function is in Algorithm 2. Figures **??**–**??** in Appendix **??** illustrate the construction of the set $\mathcal{Q}_{[\omega,1,2]}$ for the Eisenstein numeration system.

Algorithm 2 Search for weight function (Phase 2)

Input: weight coefficients set $\mathcal{Q}$

$m := 1$
for all $w_j \in \mathcal{B}$ **do**
    By Algorithm 5, find set $\mathcal{Q}_{[w_j]} \subset \mathcal{Q}$ such that

$$w_j + \mathcal{Q} \subset \mathcal{A} + \beta\mathcal{Q}_{[w_j]}$$

end for
while $\max\{\#\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]} : (w_j, \ldots, w_{j-m+1}) \in \mathcal{B}^m\} > 1$ **do**
    $m := m + 1$
    for all $(w_j, \ldots, w_{j-m+1}) \in \mathcal{B}^m$ **do**
        By Algorithm 5, find set $\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]} \subset \mathcal{Q}_{[w_j,\ldots,w_{j-m+2}]}$ such that

$$w_j + \mathcal{Q}_{[w_{j-1},\ldots,w_{j-m+1}]} \subset \mathcal{A} + \beta\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]}\,,$$

    end for
end while
$M := m$
for all $(w_j, \ldots, w_{j-M+1}) \in \mathcal{B}^M$ **do**
    $q(w_j, \ldots, w_{j-M+1}) :=$ only element of $\mathcal{Q}_{[w_j,\ldots,w_{j-M+1}]}$
end for
return q

There is space to improve Phase 2 by a modification of Algorithm 5. It is possible that the effort to reduce the size of $\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]}$ as much as possible is not the best for convergence of Phase 2.

Unfortunately, we do not know when Phase 2 converges. But we may reveal the nonconvergence of Phase 2 with deterministic Algorithm 5 for some numeration systems by Algorithm 7, which is described in Section **??**.

Notice that for a given length of window M, the number of calls of Algorithm 5 within Algorithm 2 is

$$\sum_{m=1}^{M} \#\mathcal{B}^m = \#\mathcal{B} \sum_{m=0}^{M-1} \#\mathcal{B}^m = \#\mathcal{B} \frac{\#\mathcal{B}^M - 1}{\#\mathcal{B} - 1}\,.$$

It implies that the time complexity grows exponentially as about $\#\mathcal{B}^M$. The required memory is also exponential because we have to store sets $\mathcal{Q}_{[w_j,\dots,w_{j-m+1}]}$ at least for $m \in \{M-1, M\}$ for all $w_j, \dots, w_{j-m+1} \in \mathcal{B}$.

We may reduce the number of the combinations of the input digits so that if for some $(w_j, \dots, w_{j-m+1}) \in \mathcal{B}^m, m < M$ we have $\#\mathcal{Q}_{[w_j,\dots,w_{j-m+1}]} = 1$, we do not extend the window for these digits but we set the ouput of $q(w_j, \dots, w_{j-m+1}, w_{j-m}, \dots w_{j-M+1})$ to the single element of $\mathcal{Q}_{[w_j,\dots,w_{j-m+1}]}$ for all $(w_{j-m}, \dots w_{j-M+1}) \in \mathcal{B}^{M-m}$. KONEC VLOZENI Z VYZKUMAKU

Chapter 3

Properties of $\mathbb{Z}[\omega]$

3.1 Isomorphism of $\mathbb{Z}[\omega]$ and $\mathbb{Z}^d$

In this section, we recall that the ring $\mathbb{Z}[\omega]$ is isomorphic to the set $\mathbb{Z}^d$ equiped with multiplication, where d is the degree of an algebraic integer ω. This structure is useful as it allows to handle elements of $\mathbb{Z}[\omega]$ as vectors. For example, division in $\mathbb{Z}[\omega]$ can be replaced by searching for an integer solution of a linear system (Theorem 3.2) which is used in our implementation of the extending window method.

For defining multiplication in $\mathbb{Z}^d$, we recall the notion of companion matrix.

Definition 3.1. Let $p(x) = x^d + p_{d-1}x^{d-1} + \cdots + p_1 x + p_0 \in \mathbb{Z}[x]$ be a monic polynomial with integer coefficients, $d \geq 1$. The matrix

$$S := \begin{pmatrix} 0 & 0 & \cdots & 0 & -p_0 \\ 1 & 0 & \cdots & 0 & -p_1 \\ 0 & 1 & \cdots & 0 & -p_2 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & -p_{d-1} \end{pmatrix} \in \mathbb{Z}^{d\times d}$$

is called *companion matrix* of the polynomial p.

Let S be the companion matrix of a polynomial p. It is well known (see for instance [6]) that the characteristic polynomial of the companion matrix S is p. The matrix S is also root of the polynomial p.

We remark that the minimal polynomial of an algebraic integer ω is denoted by m_ω and is always meant to be monic. Multiplication in $\mathbb{Z}^d$ is defined in the following way.

Definition 3.2. Let ω be an algebraic integer of degree $d \geq 1$ and let S be the companion matrix of m_ω. We define a mapping $\odot_\omega : \mathbb{Z}^d \times \mathbb{Z}^d \to \mathbb{Z}^d$ by

$$u \odot_\omega v := \left(\sum_{i=0}^{d-1} u_i S^i\right) \cdot \begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{d-1} \end{pmatrix} \quad \text{for all } u = \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{d-1} \end{pmatrix}, v = \begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{d-1} \end{pmatrix} \in \mathbb{Z}^d .$$

The technical proof of Theorem 3.1, which shows that $\mathbb{Z}^d$ with multiplication $\odot_\omega$ is a ring isomorphic to $\mathbb{Z}[\omega]$, can be found in [8].

Theorem 3.1. *Let ω be an algebraic integer of degree d. Then*

$$\mathbb{Z}[\omega] = \left\{ \sum_{i=0}^{d-1} a_i \omega^i \colon a_i \in \mathbb{Z} \right\},$$

$(\mathbb{Z}^d, +, \odot_\omega)$ is a commutative ring and the mapping $\pi : \mathbb{Z}[\omega] \to \mathbb{Z}^d$ defined by

$$\pi(u) = \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{d-1} \end{pmatrix} \quad \text{for every } u = \sum_{i=0}^{d-1} u_i \omega^i \in \mathbb{Z}[\omega]$$

is a ring isomorphism.

This theorem provides simple representation of elements of $\mathbb{Z}[\omega]$ in computer – they are represented by integer vectors and multiplication in $\mathbb{Z}[\omega]$ is replaced by multiplying by an appropriate matrix.

Divisibility in $\mathbb{Z}[\omega]$ can be also transformed into $\mathbb{Z}^d$. To check whether an element of $\mathbb{Z}[\omega]$ is divisible by another element, we look for an integer solution of a linear system given by Theorem 3.2. Moreover, this solution provides the result of division in the positive case.

Theorem 3.2. *Let ω be an algebraic integer of degree d and let S be the companion matrix of its minimal polynomial. Let $\beta = \sum_{i=0}^{d-1} b_i \omega^i$ be a nonzero element of $\mathbb{Z}[\omega]$. Then for every $u \in \mathbb{Z}[\omega]$*

$$u \in \beta\mathbb{Z}[\omega] \iff S_\beta^{-1} \cdot \pi(u) \in \mathbb{Z}^d,$$

where $S_\beta = \sum_{i=0}^{d-1} b_i S^i$.

The proof can be found in [8].

3.2 β-norm

The goal of this section is to construct a norm in $\mathbb{Z}[\omega]$. We use the isomorphism with $\mathbb{Z}^d$ and some results from matrix theory. This norm is used for the proof of convergence of Phase 1 in Chapter 4.

First, we recall a simple way how a new norm can be constructed from another one.

Lemma 3.3. *Let ν be a norm of the vector space $\mathbb{C}^d$ and P be a nonsingular matrix in $\mathbb{C}^d$. Then the mapping $\mu : \mathbb{C}^d \to \mathbb{R}_0^+$ defined by $\mu(x) = \nu(Px)$ is also a norm of the vector space $\mathbb{C}^d$.*

Proof. Let x and y be vectors in $\mathbb{C}^d$ and $\alpha \in \mathbb{C}$. We use linearity of matrix multiplication, nonsingularity of matrix P and the fact that ν is a norm to prove the following statements:

1. $\mu(x) = \nu(Px) \geq 0$,
2. $\mu(x) = 0 \iff \nu(Px) = 0 \iff Px = 0 \iff x = 0$,
3. $\mu(\alpha x) = \nu(P(\alpha x)) = \nu(\alpha Px) = |\alpha|\nu(Px) = |\alpha|\mu(x)$,

4. $\mu(x+y) = \nu(P(x+y)) = \nu(Px+Py) \leq \nu(Px) + \nu(Py) = \mu(x) + \mu(y)\,.$

This verifies that μ is a norm. □

Now we use Lemma 3.3 to define a new norm for a given diagonalizable matrix.

Definition 3.3. Let $M \in \mathbb{C}^{n\times n}$ be a diagonalizable matrix and $P \in \mathbb{C}^{n\times n}$ be a nonsingular matrix which diagonalizes M, i.e., $M = P^{-1}DP$ for some diagonal matrix $D \in \mathbb{C}^{n\times n}$. We define a vector norm $\|\cdot\|_M$ by

$$\|x\|_M := \|Px\|_2 \tag{3.1}$$

for all $x \in \mathbb{C}^n$, where $\|\cdot\|_2$ is Euclidean norm. A matrix norm $|||\cdot|||_M$ is induced by the norm $\|\cdot\|_M$, i.e,

$$|||A|||_M = \sup_{\|x\|_M=1} \|Ax\|_M$$

for all $A \in \mathbb{C}^{n\times n}$.

The following theorem is a known result from matrix theory – for a given diagonalizable matrix, there is a matrix norm such that the norm of the matrix equals its spectral radius. JE POTREBA NEJAKOU CITACI?

Theorem 3.4. *Let $M \in \mathbb{C}^{n\times n}$ be a diagonalizable matrix. Then*

$$\rho(M) = |||M|||_M\,,$$

where $\rho(M)$ is the spectral radius of the matrix M.

Proof. First, we prove that $|||M|||_M \geq \rho(M)$. For all eigenvalues λ in the spectrum $\sigma(M)$ with a respective eigenvector u such that $\|u\|_M = 1$, we have

$$|||M|||_M = \sup_{\|x\|_M=1} \|Mx\|_M \geq \|Mu\|_M = \|\lambda u\|_M = |\lambda| \cdot \|u\|_M = |\lambda|\,.$$

Secondly, we show that $|||M|||_M \leq \rho(M)$. Following Definition 3.3, let $P \in \mathbb{C}^{n\times n}$ be a nonsingular matrix and $D \in \mathbb{C}^{n\times n}$ a diagonal matrix with the eigenvalues of M on the diagonal such that $PMP^{-1} = D$.

Let y be a vector such that $\|y\|_M = 1$ and set $z = Py$. Notice that

$$\sqrt{z^*z} = \|z\|_2 = \|Py\|_2 = \|y\|_M = 1\,.$$

Consider

$$\begin{aligned}\|My\|_M = \|PMy\|_2 = \|DPy\|_2 = \|Dz\|_2 = \sqrt{z^*D^*Dz} \\ \leq \sqrt{\max_{\lambda\in\sigma(M)} |\lambda|^2 z^*z} = \max_{\lambda\in\sigma(M)} |\lambda| = \rho(M)\,.\end{aligned}$$

which implies the statement. □

Before we define a norm in $\mathbb{Z}[\omega]$, we verify that a specific matrix given by an algebraic number $\beta \in \mathbb{Z}[\omega]$ is diagonalizable. Lemma 3.5 summarizes also some other properties of this matrix and the norm which it induces according to Theorem 3.4.

Lemma 3.5. *Let ω be an algebraic integer of degree d and let S be the companion matrix of its minimal polynomial m_ω. Let $\beta = \sum_{i=0}^{d-1} b_i \omega^i$, where $b_i \in \mathbb{Z}$, be a nonzero element of $\mathbb{Z}[\omega]$. Set $S_\beta = \sum_{i=0}^{d-1} b_i S^i$. Then*

i) The matrix S_β is diagonalizable.

ii) The characteristic polynomial of S_β is m_β^k with $k = d/\deg \beta$.

iii) $|\det S_\beta| = |m_\beta(0)|^k$.

iv) $\|x\|_{S_\beta} = \|x\|_{S_\beta^{-1}}$ *for all* $x \in \mathbb{C}^d$ *and* $|||X|||_{S_\beta} = |||X|||_{S_\beta^{-1}}$ *for all* $X \in \mathbb{C}^{d\times d}$.

v) $|||S_\beta|||_{S_\beta} = \max\{|\beta'| \colon \beta' \text{ is conjugate of } \beta\}$ *and* $\left|\left|\left|S_\beta^{-1}\right|\right|\right|_{S_\beta} = \max\{\frac{1}{|\beta'|} \colon \beta' \text{ is conjugate of } \beta\}$.

Proof. The characteristic polynomial of the companion matrix S is the same as minimal polynomial of ω which has no multiple roots. Hence, S is diagonalizable, i.e., $S = P^{-1}DP$ where D is diagonal matrix with the conjugates of ω on the diagonal and P is a nonsingular complex matrix. The matrix S_β is also diagonalized by P:

$$S_\beta = \sum_{i=0}^{d-1} b_i S^i = \sum_{i=0}^{d-1} b_i \left(P^{-1}DP\right)^i = P^{-1} \underbrace{\left(\sum_{i=0}^{d-1} b_i D^i\right)}_{D_\beta} P\,.$$

There is a result in theory of algebraic numbers about conjugates under a field isomorphism – let α be an algebraic number with a conjugate α' and $\sigma : \mathbb{Q}(\alpha) \to \mathbb{Q}(\alpha')$ be a field isomorphism. If $\gamma \in \mathbb{Q}(\alpha)$, then $\sigma(\gamma)$ is a conjugate of γ. JE POTREBA NEJAKOU CITACI?

Therefore, the diagonal elements of the diagonal matrix D_β are conjugates of β. Since $S_\beta \in \mathbb{Z}^{d\times d}$, its characteristic polynomial p_{S_β} has integer coefficients. There exists $k \in \mathbb{N}, k \geq 1$ such that $p_{S_\beta} = m_\beta^k$ as all roots of p_{S_β} must be conjugates of β. The value k follows from the equality $d = \deg(m_\beta^k) = k \deg m_\beta$.

The modulus of the determinant of S_β equals the modulus of the absolute coefficient of the characteristic polynomial p_{S_β} which is $|m_\beta(0)|^k$.

The matrix S_β^{-1} is also diagonalized by P since $S_\beta^{-1} = (P^{-1}D_\beta P)^{-1} = P^{-1}D_\beta^{-1}P$. Thus, the norms $\|\cdot\|_{S_\beta}$ and $\|\cdot\|_{S_\beta^{-1}}$ are the same and so are the induced matrix norms $|||\cdot|||_{S_\beta}$ and $|||\cdot|||_{S_\beta^{-1}}$.

The matrix S_β is diagonalizable and its eigenvalues are the conjugates of β. Theorem 3.4 implies that

$$|||S_\beta|||_{S_\beta} = \rho(S_\beta) = \max\{|\beta'| \colon \beta' \text{ is conjugate of } \beta\}\,.$$

For the second part of the last statement, we use the part *iv)*, Theorem 3.4 and the fact that the eigenvalues of S_β^{-1} are reciprocal of the conjugates of β. □

Finally, we may define a norm in $\mathbb{Z}[\omega]$.

Definition 3.4. Let π be the isomorphism between $\mathbb{Z}[\omega]$ and $(\mathbb{Z}^d, +, \odot_\omega)$. Using the notation of the previous lemma, we define *β-norm* $\|\cdot\|_\beta : \mathbb{Z}[\omega] \to \mathbb{R}_0^+$ by

$$\|x\|_\beta = \|\pi(x)\|_{S_\beta}$$

for all $x \in \mathbb{Z}[\omega]$.

We can easily verify that β-norm is a norm in $\mathbb{Z}[\omega]$:

1. $\|x\|_\beta = \|\pi(x)\|_{S_\beta} \geq 0$,
2. $\|x\|_\beta = 0 \iff \|\pi(x)\|_{S_\beta} = 0 \iff \pi(x) = 0 \iff x = 0$,
3. $\|\alpha x\|_\beta = \|\pi(\alpha x)\|_{S_\beta} = |\alpha| \, \|\pi(x)\|_{S_\beta} = |\alpha| \, \|x\|_\beta$,
4. $\|x + y\|_\beta = \|\pi(x + y)\|_{S_\beta} = \|\pi(x) + \pi(y)\|_{S_\beta} \leq \|\pi(x)\|_{S_\beta} + \|\pi(y)\|_{S_\beta} = \|x\|_\beta + \|y\|_\beta$,

for all $x, y \in \mathbb{Z}[\omega]$ and $\alpha \in \mathbb{Z}[\omega]$.

The important property of β-norm is that there is only finitely many elements of $\mathbb{Z}[\omega]$ which are bounded in this norm by a given constant. The explanation is following – images of elements of $\mathbb{Z}[\omega]$ under the isomorphism π are integer vectors and there are only finitely many integer vectors in any finite dimensional vector space bounded by any norm. It follows from equivalence of all norms a finite dimensional vector space.

3.3 Number of congruence classes

Congruence classes play important role in the structure of an alphabet which allows parallel addition. We have seen that the isomorphism with $\mathbb{Z}^d$ is an efficient tool for handling elements of $\mathbb{Z}[\omega]$. We use it also for counting number of congruence classes. The definition of congruence in $\mathbb{Z}^d$ is following.

Definition 3.5. Let $M \in \mathbb{Z}^{d\times d}$ be a nonsingular integer matrix. Vectors $x, y \in \mathbb{Z}^d$ are *congruent modulo M in $\mathbb{Z}^d$* if $x - y \in M\mathbb{Z}^d$.

Let $x, y, z \in \mathbb{Z}^d$. We verify that congruence modulo M is an equivalence.

i) reflexivity: $x - x = 0 = M \cdot 0$,

ii) symetry: if $\exists\, v \in \mathbb{Z}^d$ such that $x - y = M \cdot v$, then $y - x = M \cdot (-v)$,

iii) transitivity: if $\exists\, v, v' \in \mathbb{Z}^d$ such that $x - y = M \cdot v$ and $y - z = M \cdot v'$, then $z - x = (z - y) + (y - x) = M \cdot (v' + v)$.

JE POTREBA TEN PRIKLAD?

In Theorem 3.7, we will see that a congruence class modulo β in $\mathbb{Z}[\omega]$ corresponds to a congruence class modulo S_β in $\mathbb{Z}^d$, where we use the notation from the previous section. Therefore, we count number of congruence classes modulo a matrix M in Lemma 3.6.

Lemma 3.6. *Let $M \in \mathbb{Z}^{d\times d}$ be a nonsingular integer matrix. The number of congruence classes modulo M in $\mathbb{Z}^d$ is $|\det M|$.*

Proof. Set $y_i := Me_i$ for $i \in \{0, \ldots, d-1\}$ and

$$B_{(\alpha_0,\ldots,\alpha_{d-1})} := \left\{ \sum_{i=0}^{d-1} (\alpha_i + t_i) y_i \colon t_i \in [0, 1) \right\}$$

for $(\alpha_0, \ldots, \alpha_{d-1}) \in \mathbb{Z}^d$. Obviously,

$$\mathbb{R}^d = \bigcup_{(\alpha_0,\ldots,\alpha_{d-1})\in\mathbb{Z}^d} B_{(\alpha_0,\ldots,\alpha_{d-1})} \, .$$

For fixed $(\alpha_0, \ldots, \alpha_{d-1}) \in \mathbb{Z}^d$, the number of points of $\mathbb{Z}^d$ in $B_{(\alpha_0,\ldots,\alpha_{d-1})}$ is the volume of $B_{(\alpha_0,\ldots,\alpha_{d-1})}$ which is $|\det M|$. Hence, it is enough to prove that there is exactly one representative of each congruence class in $B_{(\alpha_0,\ldots,\alpha_{d-1})}$.

To show that there are representatives of all classes, assume an arbitrary vector $x \in \mathbb{Z}^d$. Since $(y_0, \ldots, y_{d-1})$ is a basis of $\mathbb{R}^d$, there are scalars $s_0, \ldots, s_{d-1} \in \mathbb{R}$ such that $x = \sum_{i=0}^{d-1} s_i y_i$. Set $\gamma_i := \lfloor s_i \rfloor$ and $t_i := s_i - \gamma_i$. Now

$$x = \sum_{i=0}^{d-1} (\gamma_i + t_i) y_i = \sum_{i=0}^{d-1} t_i y_i + \sum_{i=0}^{d-1} (\gamma_i - \alpha_i) y_i + \sum_{i=0}^{d-1} \alpha_i y_i = \underbrace{\sum_{i=0}^{d-1} (\alpha_i + t_i) y_i}_{\in B_{(\alpha_0,\ldots,\alpha_{d-1})}} + M \underbrace{(\gamma - \alpha)}_{\in \mathbb{Z}^d},$$

where $\alpha = (\alpha_0, \ldots, \alpha_{d-1})^T$ and $\gamma = (\gamma_0, \ldots, \gamma_{d-1})^T$. Hence, there is an integer vector $\sum_{i=0}^{d-1} (\alpha_i + t_i) y_i$ in $B_{(\alpha_0,\ldots,\alpha_{d-1})}$ which is congruent to x modulo M.

Let $x' = \sum_{i=0}^{d-1} s'_i y_i \in \mathbb{Z}^d$ and $x'' = \sum_{i=0}^{d-1} s''_i y_i \in \mathbb{Z}^d$ be distinct elements of $B_{(\alpha_0,\ldots,\alpha_{d-1})}$ which are congruent modulo M, i.e., there exists $z = (z_0, \ldots, z_{d-1})^T \in \mathbb{Z}^d$ such that $x' = x'' + Mz$. There is $i_0 \in \{0, \ldots, d-1\}$ such that $|z_{i_0}| \geq 1$ as $x' \neq x''$. Thus, $|s'_{i_0} - s''_{i_0}| = |z_{i_0}| \geq 1$ which contradicts that $x', x'' \in B_{(\alpha_0,\ldots,\alpha_{d-1})}$. □

Now we compute number of congruence classes modulo β in $\mathbb{Z}[\omega]$ since two elements of $\mathbb{Z}[\omega]$ are congruent modulo β if and only if the corresponding vectors in $\mathbb{Z}^d$ are congruent modulo S_β.

Theorem 3.7. *Let ω be an algebraic integer of degree d and $\beta = \sum_{i=0}^{d} b_i \omega^i$, where $b_i \in \mathbb{Z}$, be such that $\deg \omega = \deg \beta$. The number of congruence classes modulo β in $\mathbb{Z}[\omega]$ is $|m_\beta(0)|$.*

Proof. Let $x, y \in \mathbb{Z}[\omega]$ and let S be the companion matrix of the minimal polynomial m_ω. Set $S_\beta = \sum_{i=0}^{d-1} b_i S^i$. Then

$$\begin{aligned} x \equiv y \mod \beta &\iff \exists z \in \mathbb{Z}[\omega] \colon x - y = \beta z \\ &\iff \exists z \in \mathbb{Z}[\omega] \colon \pi(x - y) = S_\beta \pi(z) \\ &\iff \pi(x) \equiv \pi(y) \mod S_\beta . \end{aligned}$$

Thus, the number of congruence classes modulo β is $|\det S_\beta|$ by Lemma 3.6. The statement follows from Lemma 3.5. □

Chapter 4

Convergence

4.1 Convergence of Phase 1

In this section, we show that if the extending window method converges, then the base β must be expanding, i.e., all its conjugates are greater than one in modulus. Then we prove that it is also sufficient condition for convergence of Phase 1.

We use the following notation:

Definition 4.1. Let ω be a complex number and $\beta \in \mathbb{Z}[\omega]$ be such that $|\beta| > 1$. Let $\mathcal{A} \subset \mathbb{Z}[\omega]$ be an alphabet. Set

$$\mathcal{A}[\beta] := \left\{ \sum_{i=0}^{N} a_i \beta^i \colon a_i \in \mathcal{A}, N \in \mathbb{N} \right\} .$$

The essential part of the proof that β must be expanding is Theorem 4.1 which is based on the papers of Akiyama, Thuswaldner and Zäimi [1, 2].

Theorem 4.1. *Let ω be a complex number and $\beta \in \mathbb{Z}[\omega]$ be such that $|\beta| > 1$. Let $\mathcal{A} \subset \mathbb{Z}[\omega]$ be an alphabet. If $\mathbb{N} \subset \mathcal{A}[\beta]$, then the number β is expanding.*

Proof. For all $n \in \mathbb{N}$ we may write

$$n = \sum_{i=0}^{N} a_i \beta^i ,$$

where $N \in \mathbb{N}$, $a_i \in \mathcal{A}$ and $a_N \neq 0$.

Set $m := \max\{|a| \colon a \in \mathcal{A}\}$. We take $n \in \mathbb{N}$ such that $n > m$. Since $|a_0| \leq m < n$, we have $N \geq 1$ and there is $i_0 \in \{1, 2, \ldots, N\}$ such that $a_{i_0} \neq 0$. Thus, ω is an algebraic number as $a_i \in \mathcal{A} \subset \mathbb{Z}[\omega]$ and β can be expressed as an integer combination of powers of ω. Therefore, β is also an algebraic number.

Let β' be an algebraic conjugate of β. Since $\beta \in \mathbb{Z}[\omega] \subset \mathbb{Q}(\omega)$, there is an algebraic conjugate ω' of ω and an isomorphism $\sigma : \mathbb{Q}(\omega) \to \mathbb{Q}(\omega')$ such that $\sigma(\beta) = \beta'$. Now

$$n = \sigma(n) = \sum_{i=0}^{N} \sigma(a_i)(\beta')^i .$$

Set $\tilde{m} := \max\{|\sigma(a)| \colon a \in \mathcal{A}\}$. For all $n \in \mathbb{N}$, we have

$$n = |n| \leq \sum_{i=0}^{N} |\sigma(a_i)| \cdot |\beta'|^i \leq \sum_{i=0}^{\infty} |\sigma(a_i)| \cdot |\beta'|^i \leq \tilde{m} \sum_{i=0}^{\infty} |\beta'|^i .$$

Hence, the sum on the right side diverges which implies that $|\beta'| \geq 1$. Thus, all conjugates of β are at least one in modulus.

If the degree of β is one, the statement is obvious. Therefore, we may assume that $\deg \beta \geq 2$.

Suppose for contradiction that $|\beta'| = 1$ for an algebraic conjugate β' of β. The complex conjugate $\overline{\beta'}$ is also an algebraic conjugate of β. Take any algebraic conjugate γ of β and the isomorphism $\sigma' : \mathbb{Q}(\beta') \to \mathbb{Q}(\gamma)$ given by $\sigma'(\beta') = \gamma$. Now

$$\frac{1}{\gamma} = \frac{1}{\sigma'(\beta')} = \sigma'\left(\frac{1}{\beta'}\right) = \sigma'\left(\frac{\overline{\beta'}}{\beta'\overline{\beta'}}\right) = \sigma'\left(\frac{\overline{\beta'}}{|\beta'|^2}\right) = \sigma'(\overline{\beta'})\,.$$

Hence, $\frac{1}{\gamma}$ is also an algebraic conjugate of β. Moreover, $\left|\frac{1}{\gamma}\right| \geq 1$ and $|\gamma| \geq 1$ which implies that $|\gamma| = 1$. We may set $\gamma = \beta$ which contradicts $|\beta| > 1$. Thus all conjugates of β are greater than one in modulus, i.e., β is an expanding algebraic number. □

Now we can easily prove that existence of a parallel addition algorithm with rewriting rule $x - \beta$ implies that β is expanding.

Theorem 4.2. *Let $\mathcal{A} \subset \mathbb{Z}[\omega]$ be an alphabet such that $1 \in \mathcal{A}[\beta]$. If the extending window method with the rewriting rule $x - \beta$ converges for the numeration system $(\beta, \mathcal{A})$, then the base β is expanding.*

Proof. The existence of an algorithm for addition which is computable in parallel implies that the set $\mathrm{Fin}_{\mathcal{A}}(\beta)$ is closed under addition. Moreover, the set $\mathcal{A}[\beta]$ is closed under addition since there is no carry to the right when the rewriting rule $x - \beta$ is used. For any $n \in \mathbb{N}$, the sum $1 + 1 + \cdots + 1 = n$ is in $\mathcal{A}[\beta]$ by the assumption $1 \in \mathcal{A}[\beta]$. Therefore, $\mathbb{N} \subset \mathcal{A}[\beta]$ and thus the base β is expanding by Theorem 4.1. □

Since we know that it makes sense to consider only expanding base, we may ask if Phase 1 converges for such a base. The answer is positive, with some natural assumption on the alphabet $\mathcal{A}$. The following lemma provides a finite set of weight coefficients $\mathcal{Q}$.

Lemma 4.3. *Let ω be an algebraic integer, $\deg \omega = d$, and β be an expanding algebraic integer in $\mathbb{Z}[\omega]$. Let $\mathcal{A}$ and $\mathcal{B}$ be finite subsets of $\mathbb{Z}[\omega]$ such that $\mathcal{A}$ contains at least one representative of each congruence class modulo β in $\mathbb{Z}[\omega]$. There exists a finite set $\mathcal{Q} \subset \mathbb{Z}[\omega]$ such that $\mathcal{B} + \mathcal{Q} \subset \mathcal{A} + \beta\mathcal{Q}$.*

Proof. We use the isomorphism $\pi : \mathbb{Z}[\omega] \to \mathbb{Z}^d$ and β-norm $\|\cdot\|_\beta$ to give a bound on the elements of $\mathbb{Z}[\omega]$. Let γ be the smallest conjugate of β in modulus. Denote $C := \max\{\|b - a\|_\beta : a \in \mathcal{A}, b \in \mathcal{B}\}$. Consequently, set $R := \frac{C}{|\gamma|-1}$ and $\mathcal{Q} := \{q \in \mathbb{Z}[\omega] \colon \|q\|_\beta \leq R\}$. By Lemma 3.5, we have

$$\left|\left|\left|S_\beta^{-1}\right|\right|\right|_{S_\beta} = \max\{\frac{1}{|\beta'|} \colon \beta' \text{ is conjugate of } \beta\} = \frac{1}{|\gamma|}\,.$$

Also, $|\gamma| > 1$ as β is an expanding integer. Since $C > 0$, the set $\mathcal{Q}$ is nonempty. Any element $x = b + q \in \mathbb{Z}[\omega]$ with $b \in \mathcal{B}$ and $q \in \mathcal{Q}$ can be written as $x = a + \beta q'$ for some $a \in \mathcal{A}$ and $q' \in \mathbb{Z}[\omega]$ due to existence of a representative of each congruence class in $\mathcal{A}$. Using the

isomorphism π, we may write $\pi(q') = S_\beta^{-1} \cdot \pi(b - a + q)$. We prove that q' is in $\mathcal{Q}$:

$$\begin{aligned}\|q'\|_\beta = \|\pi(q')\|_{S_\beta} = \left\|S_\beta^{-1} \cdot \pi(b-a+q)\right\|_{S_\beta} &\leq \left\|\left|S_\beta^{-1}\right|\right\|_{S_\beta} \|b-a+q\|_\beta \\ \leq \frac{1}{|\gamma|}(\|b-a\|_\beta + \|q\|_\beta) = \frac{1}{|\gamma|}(C+R) &= \frac{C}{|\gamma|}(1+\frac{1}{|\gamma|-1}) = R\,.\end{aligned}$$

Hence $q' \in \mathcal{Q}$ and thus $x = b + q \in \mathcal{A} + \beta\mathcal{Q}$.

Since there are only finitely many elements of $\mathbb{Z}^d$ bounded by the constant R, the set $\mathcal{Q}$ is finite. □

The way how candidates for the weight coefficients are chosen in Algorithm 4 is the same as in the proof of Lemma 4.3. Therefore, the convergence of Phase 1 is guaranteed by the following theorem.

Theorem 4.4. *Let ω be an algebraic integer and $\beta \in \mathbb{Z}[\omega]$. Let the alphabet $\mathcal{A} \subset \mathbb{Z}[\omega]$ be such that $\mathcal{A}$ contains at least one representative of each congruence class modulo β in $\mathbb{Z}[\omega]$. Let $\mathcal{B} \subset \mathbb{Z}[\omega]$ be the input alphabet.*

If β is expanding, then Phase 1 of the extending window method converges.

Proof. Let R be a constant and $\mathcal{Q}$ be a finite set from Lemma 4.3 for the alphabet $\mathcal{A}$ and input alphabet $\mathcal{B}$. We prove by induction that all intermediate weight coefficient sets $\mathcal{Q}_k$ in Algorithm 1 are subsets of the finite set $\mathcal{Q}$.

We start with $\mathcal{Q}_0 = \{0\}$ whose elements are bounded by any positive constant. Suppose that the intermediate weight coefficients set $\mathcal{Q}_k$ has elements bounded by the constant R. We see from the previous proof that the candidates obtained by Algorithm 4 for the set $\mathcal{Q}_k$ are also bounded by R. Thus, the next intermediate weight coefficients set $\mathcal{Q}_{k+1}$ has elements bounded by the constant R, i.e., $\mathcal{Q}_{k+1} \subset \mathcal{Q}$.

Since $\#\mathcal{Q}$ is finite and $\mathcal{Q}_0 \subset \mathcal{Q}_1 \subset \mathcal{Q}_2 \subset \cdots \subset \mathcal{Q}$, Phase 1 successfully ends. □

4.2 Convergence of Phase 2

We have no simple sufficient or necessary condition of convergence of Phase 2 in the sense of properties of a base β or an alphabet $\mathcal{A}$. Nevertheless, the convergence can be controled during a run of algorithm. An easy check of non-convergence can be done by searching $\mathcal{Q}_{[b,\ldots,b]}$ for each $b \in \mathcal{B}$ separately. This was already described in [8], but we recall it with a simpler proof. For its purposes, we introduce a notion of stable Phase 2, which is used also in the main result of this section – the control of convergence during Phase 2 is transformed into searching for a cycle in an oriented graph.

Firstly, we menshion some equivalent conditions of non-convergence of Phase 2. It enables us to handle easier with non-convergence in later proofs.

Lemma 4.5. *The following statements are equivalent:*

i) Phase 2 does not converge,

ii) $\forall\, k \in \mathbb{N}\, \exists\, (w_0, \ldots, w_{-k}) \in \mathcal{B}^{k+1} \colon \#\mathcal{Q}_{[w_0,\ldots,w_{-k}]} \geq 2$,

iii) $\exists\, (w_{-k})_{k\geq 0}, w_{-k} \in \mathcal{B}\, \exists\, k_0 \forall k \geq k_0 \colon \#\mathcal{Q}_{[w_0,\ldots,w_{-k}]} = \#\mathcal{Q}_{[w_0,\ldots,w_{-(k-1)}]} \geq 2$.

Proof. i) $\iff$ ii): The while loop in Algorithm 2 ends if and only if there exist $k \in \mathbb{N}$ such that $\#\mathcal{Q}_{[w_0,\dots,w_{-k}]} = 1$ for all $(w_0,\dots,w_{-k}) \in \mathcal{B}^{k+1}$.

ii) $\iff$ iii): There is an infinite sequence $(w_{-k})_{k\geq 0}$ such that $\#\mathcal{Q}_{[w_0,\dots,w_{-k}]} \geq 2$ for all $k \in \mathbb{N}$ since $\mathcal{Q}_{[w_0,\dots,w_{-k}]} \supset \mathcal{Q}_{[w_0,\dots,w_{-(k+1)}]}$. Hence, the sequence of integers $(\#\mathcal{Q}_{[w_0,\dots,w_{-k}]})_{k\geq 0}$ is eventually constant. The opposite implication is trivial. □

We need to ensure that choice of a possible weight coefficient set $\mathcal{Q}_{[w_0,\dots,w_{-k}]} \subset \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is determined by an input digit w_0 and a set $\mathcal{Q}_{[w_{-1},\dots,w_{-0}]}$, while the influence of the set $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is limited. It is formalized in the following definition.

Definition 4.2. Let $\mathcal{B}$ be an alphabet of input digits. We say that Phase 2 is *stable*, if the following condition on the sets of possible weight coefficients which are produced by the algorithm holds:

$$\mathcal{Q}_{[w_{-1},\dots,w_{-k}]} = \mathcal{Q}_{[w_{-1},\dots,w_{-(k-1)}]} \implies \mathcal{Q}_{[w_0,\dots,w_{-k}]} = \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$$

for all $k \in \mathbb{N}, k \geq 2$ and for all $(w_0,\dots,w_{-(k+1)}) \in \mathcal{B}^{k+1}$.

II)$\#\mathcal{Q}_{[w_{-1},\dots,w_{-K}]} = 1 \implies \#\mathcal{Q}_{[w_0,\dots,w_{-K}]} = 1$, NENI POTREBA POKUD NEBUDE OPACNA IMPLIKACE

The definition may seem too restrictive, but note that $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ if fully determined by $\mathcal{Q}_{[w_{-1},\dots,w_{-k}]}$, $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ and w_0 for a fixed deterministic way of choice of possible weight coefficients sets. Therefore, the assumption $\mathcal{Q}_{[w_{-1},\dots,w_{-k}]} = \mathcal{Q}_{[w_{-1},\dots,w_{-(k-1)}]}$ implies that the only difference in the choice of $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ an $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is that $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ is a subset of $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$, while $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is chosen as a subset of $\mathcal{Q}_{[w_0,\dots,w_{-(k-2)}]}$. At the same time, $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is a subset of $\mathcal{Q}_{[w_0,\dots,w_{-(k-2)}]}$. Thus, the requirement that Phase 2 is stable means that the same possible weight coefficients set is found even if it is searched as a subset of smaller set. This is natural way how an algorithm of choice should be constructed – the set $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ is searched such that

$$\mathcal{B} + \mathcal{Q}_{[w_{-1},\dots,w_{-k}]} \subset \mathcal{A} + \beta\mathcal{Q}_{[w_0,\dots,w_{-k}]}\,,$$

i.e., there is no reason to choose the set $\mathcal{Q}_{[w_0,\dots,w_{-k}]} \subsetneq \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ as we know that

$$\mathcal{B} + \underbrace{\mathcal{Q}_{[w_{-1},\dots,w_{-(k-1)}]}}_{=\mathcal{Q}_{[w_{-1},\dots,w_{-k}]}} \subset \mathcal{A} + \beta\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}\,.$$

In other word, if $\mathcal{Q}_{[w_0,\dots,w_{-k}]} \subsetneq \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$, the set $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ might have been chosen smaller.

Now we use that finiteness of Phase 2 implies that there exists a length of window m such that the set $\mathcal{Q}^m_{[b]}$ contains only one element for all $b \in \mathcal{B}$, where $\mathcal{Q}^m_{[b]}$ is a shorter notation for

$$\mathcal{Q}_{\underbrace{[b,\dots,b]}_{m}}\,.$$

The following theorem was proved in [8] with the assumption that Phase 2 is deterministic. Briefly, it says that $\#\mathcal{Q}^m_{[b]}$ must decrease every time we increase m, otherwise Phase 2 does not converge. When we consider only inputs of the form $bb\dots b$ for some $b \in \mathcal{B}$, determinism implies that Phase 2 is stable. The given proof with Phase 2 being stable is slightly shorter.

Theorem 4.6. *Let $m_0 \in \mathbb{N}$ and $b \in \mathcal{B}$ be such that sets $\mathcal{Q}_{[b]}^{m_0}$ and $\mathcal{Q}_{[b]}^{m_0-1}$ produced by stable Phase 2 have the same size. Then*

$$\#\mathcal{Q}_{[b]}^{m} = \#\mathcal{Q}_{[b]}^{m_0} \qquad \forall m \geq m_0 - 1\,.$$

Particularly, if $\#\mathcal{Q}_{[b]}^{m_0} \geq 2$, Phase 2 does not converge.

Proof. As $\mathcal{Q}_{[b]}^{m_0} \subset \mathcal{Q}_{[b]}^{m_0-1}$, the assumption of the same size implies

$$\mathcal{Q}_{[b]}^{m_0} = \mathcal{Q}_{[b]}^{m_0-1}\,.$$

By the assumption that Phase 2 is stable, we have

$$\begin{aligned} \mathcal{Q}_{[b]}^{m_0} = \mathcal{Q}_{[b]}^{m_0-1} &\implies \mathcal{Q}_{[b]}^{m_0+1} = \mathcal{Q}_{[b]}^{m_0} \\ &\implies \mathcal{Q}_{[b]}^{m_0+2} = \mathcal{Q}_{[b]}^{m_0+1} \\ &\vdots \end{aligned}$$

This implies the statement.

If $\#\mathcal{Q}_{[b]}^{m_0} \geq 2$, then the statement *iii)* in Lemma 4.5 holds for the sequence $(b)_{k\geq 0}$. □

The condition of convergence during Phase 2 is formulated as a searching for an infinite path in a so-called Rauzy graph. This term comes from combinatorics on words. The vertices of our graph are combinations of input digits for which the size of their possible weight coefficients sets did not decreased with an increment of lenght of the window. Whereas in combinatorics on words, vertices are given as factors of some language. But the edges are placed identically – if some combination of digits without the first one equals another one without the last digit.

Definition 4.3. Let $\mathcal{B}$ be an alphabet of input digits. Let Phase 2 is stable. Let $k \in \mathbb{N}, k \geq 2$. We set

$$V := \left\{(w_{-1}, \ldots, w_{-k}) \in \mathcal{B}^k \colon \#\mathcal{Q}_{[w_{-1},\ldots,w_{-k}]} = \#\mathcal{Q}_{[w_{-1},\ldots,w_{-(k-1)}]}\right\}$$

and

$$E := \left\{(w_{-1}, \ldots, w_{-k}) \to (w'_{-1}, \ldots, w'_{-k}) \in V \times V \colon (w_{-2}, \ldots, w_{-k}) = (w'_{-1}, \ldots, w'_{-(k-1)})\right\}\,.$$

The oriented graph $G_k = (V, E)$ is called *Rauzy graph of Phase 2 (for the window of length k).*

The structure of Rauzy graph G_k signifies whether the non-decreasing combinations are such that they cause nonconvergence of Phase 2. Existence of an infinite walk in G_k implies that Phase 2 does not converge:

Theorem 4.7. *Let Phase 2 is stable. If there exists $k_0 \in \mathbb{N}, k_0 \geq 2$, and $(w_0, \ldots, w_{-k_0}) \in \mathcal{B}^{k_0+1}$ such that*

i) $\#\mathcal{Q}_{[w_0,\ldots,w_{-(k_0-1)}]} > 1$ *and*

ii) there exist an infinite walk $((w_{-1}^{(i)}, \ldots, w_{-k_0}^{(i)}))_{i\geq 1}$ in G_{k_0} which starts in the vertex

$$(w_{-1}^{(1)}, \ldots, w_{-k_0}^{(1)}) = (w_{-1}, \ldots, w_{-k_0})\,,$$

then Phase 2 does not converge.

Proof. Set

$$(w_k)_{k\geq 0} := w_0, w_1^{(1)}, \ldots, w_{k_0-1}^{(1)}, w_{k_0}^{(1)}, w_{k_0}^{(2)}, w_{k_0}^{(3)}, w_{k_0}^{(4)}, \ldots$$

We prove that $\#\mathcal{Q}_{[w_0,\ldots,w_{-k}]} = \#\mathcal{Q}_{[w_0,\ldots,w_{-(k_0-1)}]} > 1$ for all $k \geq k_0 - 1$, i.e., the condition *iii)* in Lemma 4.5 is satisfied.

Let $l \in \mathbb{N}$. Since $(w_{-(1+l)}, \ldots, w_{-(k_0+l)})$ is a vertex of G_{k_0}, the set $\mathcal{Q}_{[w_{-l},\ldots,w_{-(k_0+l)}]}$ equals $\mathcal{Q}_{[w_{-l},\ldots,w_{-(k_0+l-1)}]}$. As Phase 2 is stable, we have

$$\begin{aligned}
&\mathcal{Q}_{[w_{-l},\ldots,w_{-(k_0+l)}]} = \mathcal{Q}_{[w_{-l},\ldots,w_{-(k_0+l-1)}]} \\
\implies &\mathcal{Q}_{[w_{-(l-1)},\ldots,w_{-(k_0+l)}]} = \mathcal{Q}_{[w_{-(l-1)},\ldots,w_{-(k_0+l-1)}]} \\
&\vdots \\
\implies &\mathcal{Q}_{[w_{-1},\ldots,w_{-(k_0+l)}]} = \mathcal{Q}_{[w_{-1},\ldots,w_{-(k_0+l-1)}]} \\
\implies &\mathcal{Q}_{[w_0,\ldots,w_{-(k_0+l)}]} = \mathcal{Q}_{[w_0,\ldots,w_{-(k_0+l-1)}]}\,.
\end{aligned}$$

Hence, $\#\mathcal{Q}_{[w_0,\ldots,w_{-k}]} = \#\mathcal{Q}_{[w_0,\ldots,w_{-(k_0-1)}]} > 1$ for all $k \geq k_0 - 1$. □

BYLO BY FAJN DOKAZAT I OPACNY SMER, JEDINE, PRES CO SE NEUMIM DOSTAT JE KDYBY EXISTOVALA JEN APERIODICKA POSLOUPNOUST, KVULI KTERE TO NEKONVERGUJE

We remark that existence of an infinite walk in a finite graph is equivalent to existence of a cycle in the graph. Thus, if there is an infinite walk, we may find another one whose sequence of vertices is eventually periodic. We use this fact in Section 6.1 which describes modified algorithm for Phase 2 which implements the result of Theorem 4.7.

4.3 Minimal alphabet $\mathcal{A}$

Frougny, Pelantová and Svodová [5] proved a lower bound on the size of an alphabet $0 \in \mathcal{A} \subset \mathbb{Z}$ of consecutive integers which enables parallel addition. In this sextion, we prove the same bound for an arbitrary alphabet $\mathcal{A} \in \mathbb{Z}[\beta]$. We recall their auxiliary results in Theorem 4.8 an Lemma 4.9, but only for a parallel digit set conversion without anticipation as our rewriting rule $x - \beta$ does not require memory. The following theorem says that all classes modulo β which are contained in $\mathcal{A} + \mathcal{A}$ must have their representatives in $\mathcal{A}$.

Theorem 4.8. *Let ω be an algebraic integer. Let the base $\beta \in \mathbb{Z}[\omega]$ be such that $|\beta| > 1$ and the alphabet $\mathcal{A} \subset \mathbb{Z}[\omega]$ be such that $0 \in \mathcal{A}$. If there exists a p-local digit set conversion defined by the function $\phi\colon (\mathcal{A}+\mathcal{A})^p \to \mathcal{A}$ and $p = r + 1$, then the number $\phi(b, \ldots, b) - b$ belongs to the set $(\beta - 1)\mathbb{Z}[\omega]$ for any $b \in \mathcal{A} + \mathcal{A}$.*

Proof. Let $b \in \mathcal{A} + \mathcal{A}$ and $a = \phi(b, \ldots, b)$. For $n \in \mathbb{N}, n \geq 1$, we denote S_n the number represented by

$${}^{\omega}0\underbrace{b\ldots b}_{n}\bullet\underbrace{b\ldots b}_{r}0^{\omega}\,.$$

The representation of S_n after the digit set conversion is of the form

$$^{\omega}0\,\underbrace{w_r \dots w_1}_{\beta^n W}\,\underbrace{a \dots a}_{n} \bullet \underbrace{\widetilde{w_1} \dots \widetilde{w_r}}_{\beta^{-r}\widetilde{W}}\,0^{\omega}\,,$$

where

$$W = \sum_{j=1}^{r} w_j \beta^{j-1} \qquad \text{and} \qquad \widetilde{W} = \sum_{j=1}^{r} \widetilde{w_j} \beta^{r-j}\,.$$

Since both representations have same value, we have

$$b \sum_{j=-r}^{n-1} \beta^j = W\beta^n + a \sum_{j=0}^{n-1} \beta^j + \beta^{-r}\widetilde{W}$$

$$b \sum_{j=-r}^{-1} \beta^j + b\frac{\beta^n - 1}{\beta - 1} = W\beta^n + a\frac{\beta^n - 1}{\beta - 1} + \beta^{-r}\widetilde{W}\,, \tag{4.1}$$

for all $n \geq 1$. We substract this equation for n and $n-1$ to obtain

$$b\frac{\beta^n - \beta^{n-1}}{\beta - 1} = W(\beta^n - \beta^{n-1}) + a\frac{\beta^n - \beta^{n-1}}{\beta - 1}\,.$$

We simplify it to

$$b = W(\beta - 1) + a\,. \tag{4.2}$$

Hence, $a = \phi(b, \dots, b) \equiv b$ modulo $\beta - 1$. □

If a base β has a real conjugate greater than one, then there are some extra requirements on the alphabet $\mathcal{A}$. For simplicity, we assume that the base β itself is real and greater than one. We show later that this assumption is without loss of generality.

Lemma 4.9. *Let ω be a real algebraic integer and the base $\beta \in \mathbb{Z}[\omega]$ be such that $\beta > 1$. Let the alphabet $\mathcal{A} \subset \mathbb{Z}[\omega]$ be such that $0 \in \mathcal{A}$ and denote $\lambda = \min \mathcal{A}$ and $\Lambda = \max \mathcal{A}$. If there exists a p-local digit set conversion defined by the function $\phi\colon (\mathcal{A}+\mathcal{A})^p \to \mathcal{A}$ and $p = r+1$, then:*

i) $\phi(b, \dots, b) \neq \lambda$ for all $b \in \mathcal{A} + \mathcal{A}$ such that $b > \lambda$.

ii) $\phi(b, \dots, b) \neq \Lambda$ for all $b \in \mathcal{A} + \mathcal{A}$ such that $b < \Lambda$.

iii) If $\Lambda \neq 0$, then $\phi(\Lambda, \dots, \Lambda) \neq \Lambda$.

iv) If $\lambda \neq 0$, then $\phi(\lambda, \dots, \lambda) \neq \lambda$.

Proof. To prove *i)*, assume in contradiction that $\phi(b, \dots, b) = \lambda$. We proceed in the same manner as in Theorem 4.8, the equation (4.1) implies

$$b \sum_{j=-r}^{-1} \beta^j + b\frac{\beta^n - 1}{\beta - 1} = \beta^n W + \lambda\frac{\beta^n - 1}{\beta - 1} + \beta^{-r}\widetilde{W}\,.$$

We apply also the equation c to obtain

$$b\sum_{j=-r}^{-1}\beta^j + b\frac{\beta^n-1}{\beta-1} = \beta^n\frac{b-\lambda}{\beta-1} + \lambda\frac{\beta^n-1}{\beta-1} + \beta^{-r}\widetilde{W}\,.$$

Now we may simplify and estimate

$$b\sum_{j=-r}^{-1}\beta^j + \frac{-b}{\beta-1} = \frac{-\lambda}{\beta-1} + \beta^{-r}\sum_{j=1}^{r}\widetilde{w_j}\beta^{r-j}$$

$$\underbrace{b\left(\sum_{j=1}^{r}\frac{1}{\beta^j} - \frac{1}{\beta-1}\right)}_{-\sum_{j=r+1}^{\infty}\frac{1}{\beta^j}} = -\lambda\frac{1}{\beta-1} + \sum_{j=1}^{r}\widetilde{w_j}\beta^{-j} \geq \lambda\underbrace{\left(-\frac{1}{\beta-1} + \sum_{j=1}^{r}\frac{1}{\beta^j}\right)}_{-\sum_{j=r+1}^{\infty}\frac{1}{\beta^j}}\,.$$

Hence $b \leq \lambda$ which is a contradiction. The proof of *ii)* can be done in the same way.

For *iii)*, assume that $\phi(\Lambda,\ldots,\Lambda) = \Lambda$. Now consider a number T_q represented by

$$^{\omega}0 \bullet \underbrace{\Lambda\ldots\Lambda}_{r}\underbrace{(2\Lambda)\ldots(2\Lambda)}_{q}0^{\omega}\,.$$

After the digit set conversion, a representation is

$$^{\omega}0\underbrace{w_r\ldots w_1}_{W}\bullet z_1\ldots z_{r+q}0^{\omega}\,.$$

The value T_q preserves, thus,

$$\Lambda\sum_{j=1}^{r}\beta^{-j} + 2\Lambda\sum_{j=r+1}^{r+q}\beta^{-j} = W + \sum_{j=1}^{r+q}z_j\beta^{-j}\,.$$

But $W = 0$ from the equation (4.2). We estimate

$$\Lambda\sum_{j=1}^{r+q}\beta^{-j} + \Lambda\sum_{j=r+1}^{r+q}\beta^{-j} = \sum_{j=1}^{r+q}z_j\beta^{-j} \leq \Lambda\sum_{j=1}^{r+q}\beta^{-j}$$

$$\Lambda\sum_{j=r+1}^{r+q}\beta^{-j} \leq 0\,.$$

This contradicts that Λ is positive as it is a nonzero, maximal element of the alphabet $\mathcal{A}$ which contains 0. The proof of *iv)* is analogous. □

In order to prove the lower bound, we need to show that the alphabet $\mathcal{A}$ must contain all representatives modulo β and $\beta - 1$.

Theorem 4.10. *Let β be an algebraic integer such that $|\beta| > 1$. Let $0 \in \mathcal{A} \subset \mathbb{Z}[\beta]$ be an alphabet such that $1 \in \mathcal{A}[\beta]$. If addition in the numeration system $(\beta, \mathcal{A})$ which uses the rewriting rule $x - \beta$ is computable in parallel, then the alphabet $\mathcal{A}$ contains at least one representative of each congruence class modulo β and $\beta - 1$ in $\mathbb{Z}[\beta]$.*

Proof. The existence of an algorithm for addition with the rewriting rule $x - \beta$ implies that the set $\mathcal{A}[\beta]$ is closed under addition. By the assumption $1 \in \mathcal{A}[\beta]$, the set $\mathbb{N}$ is subset of $\mathcal{A}[\beta]$. Since $0 \in \mathcal{A}$, we have $\beta \cdot \mathcal{A}[\beta] \subset \mathcal{A}[\beta]$. Hence, $\mathbb{N}[\beta] \subset \mathcal{A}[\beta]$.

For any element $x = \sum_{i=0}^{N} x_i \beta^i \in \mathbb{Z}[\beta]$ there is an element $x' = \sum_{i=0}^{N} x'_i \beta^i \in \mathbb{N}[\beta]$ such that $x \equiv_\beta x'$ since $m_\beta(0) \equiv_\beta 0$ and $\beta^i \equiv_\beta 0$. As $x' \in \mathbb{N}[\beta] \subset \mathcal{A}[\beta]$, we have

$$x \equiv_\beta x' = \sum_{i=0}^{n} a_i \beta^i \equiv_\beta a_0 \,,$$

where $a_i \in \mathcal{A}$. Hence, for any element $x \in \mathbb{Z}[\omega]$, there is a letter $a_0 \in \mathcal{A}$ such that $x \equiv_\beta a_0$.

In order to prove that there is at least one representative of each congruence class modulo $\beta - 1$ in the alphabet $\mathcal{A}$, we consider again an element $x = \sum_{i=0}^{N} x_i \beta^i \in \mathbb{Z}[\beta]$. Similarly, there is an element $x' = \sum_{i=0}^{N} x'_i \beta^i \in \mathbb{N}[\beta]$ such that $x \equiv_{\beta-1} x'$ since $m_{\beta-1}(0) \equiv_{\beta-1} 0$ and $(\beta-1)^i \equiv_{\beta-1} 0$.

Since $x' \in \mathbb{N}[\beta] \subset \mathcal{A}[\beta]$,

$$x' = \sum_{i=0}^{n} a_i \beta^i \,,$$

where $a_i \in \mathcal{A}$. We prove by induction with respect to n that $x' \equiv_{\beta-1} a$ for some $a \in \mathcal{A}$. If $n = 0$, $x' = a_0$. Now we use the fact that if there is a parallel addition algorithm, for each letter $b \in \mathcal{A} + \mathcal{A}$, there is $a \in \mathcal{A}$ such that $b \equiv_{\beta-1} a$ (Theorem 4.8). For $n+1$, we have

$$\begin{aligned} x' &= \sum_{i=0}^{n+1} a_i \beta^i = a_0 + \sum_{i=1}^{n+1} a_i \beta^i \\ &= a_0 + \beta \sum_{i=0}^{n} a_{i+1} \beta^i - \sum_{i=0}^{n} a_{i+1} \beta^i + \sum_{i=0}^{n} a_{i+1} \beta^i \\ &\equiv_{\beta-1} a_0 + (\beta-1) \sum_{i=0}^{n} a_{i+1} \beta^i + a \equiv_{\beta-1} a_0 + a \equiv_{\beta-1} a' \in \mathcal{A} \,, \end{aligned}$$

where we use the induction assumption

$$\sum_{i=0}^{n} a_{i+1} \beta^i \equiv_{\beta-1} a \,.$$

□

The following lemma summarizes that if we have a parallel addition algorithm for a base β, then we easily obtain an algorithm also for conjugates of β.

Lemma 4.11. *Let ω be an algebraic integer with a conjugate ω'. Let $\beta \in \mathbb{Z}[\omega], |\beta| > 1$ and let $\sigma : \mathbb{Q}(\omega) \to \mathbb{Q}(\omega')$ be an isomorphism such that $|\sigma(\beta)| > 1$. Let φ is a digit set conversion in the base β from $\mathcal{A} + \mathcal{A}$ to $\mathcal{A}$. There exists is a digit set conversion φ' in the base β' from $\mathcal{A}' + \mathcal{A}'$ to $\mathcal{A}'$ where $\beta' = \sigma(\beta)$ and $\mathcal{A}' = \{\sigma(a) \colon a \in \mathcal{A}\}$.*

Proof. Let $\phi : \mathcal{A}^p \to \mathcal{A}$ be a mapping which defines φ with $p = r + t + 1$. We define a mapping $\phi' : \mathcal{A}^p \to \mathcal{A}$ by

$$\phi'(w'_{j+t}, \ldots, w'_{j-r}) = \sigma\left(\phi\left(\sigma^{-1}(w'_{j+t}), \ldots, \sigma^{-1}(w'_{j-r})\right)\right) \,.$$

Next, we define a digit set conversion $\varphi' : (\mathcal{A}' + \mathcal{A}') \to \mathcal{A}'$ by $\varphi'(w') = (z'_j)_{j\in\mathbb{Z}}$ where $w' = (w'_j)_{j\in\mathbb{Z}}$ and $z'_j = \phi'(w'_{j+t}, \ldots, w'_{j-r})$. Obviously, if w' has only finitely many nonzero entries, then there is only finitely many nonzeros in $(z'_j)_{j\in\mathbb{Z}}$ since

$$\phi'(0,\ldots,0) = \sigma\left(\phi\left(\sigma^{-1}(0),\ldots,\sigma^{-1}(0)\right)\right) = \sigma\left(\phi\left(0,\ldots,0\right)\right) = \sigma\left(0\right) = 0\,.$$

The value of the number represented by w' is also preserved:

$$\begin{aligned}\sum_{j\in\mathbb{Z}} w'_j\beta'^j &= \sum_{j\in\mathbb{Z}} \sigma(w_j)\sigma(\beta)^j = \sigma(\sum_{j\in\mathbb{Z}} w_j\beta^j)\\ &= \sigma(\sum_{j\in\mathbb{Z}} z_j\beta^j) = \sigma(\sum_{j\in\mathbb{Z}} \phi\left(w_{j+t},\ldots,w_{j-r}\right)\beta^j)\\ &= \sum_{j\in\mathbb{Z}} \sigma(\phi\left(w_{j+t},\ldots,w_{j-r}\right))\beta'^j = \sum_{j\in\mathbb{Z}} z'_j\beta'^j\end{aligned}$$

where $w_j = \sigma^{-1}(w'_j)$ for $j \in \mathbb{Z}$ and $\varphi((w_j)_{j\in\mathbb{Z}}) = (z_j)_{j\in\mathbb{Z}}$. □

Finally, we put together that the alphabet $\mathcal{A}$ contains all representative modulo β and $\beta - 1$, number of congruence classes and restrictions on the alphabet for a base with a real conjugate greater than one.

Theorem 4.12. *Let β be an algebraic integer such that $|\beta| > 1$. Let $0 \in \mathcal{A} \subset \mathbb{Z}[\beta]$ be an alphabet such that $1 \in \mathcal{A}[\beta]$. If addition in the numeration system $(\beta, \mathcal{A})$ which uses the rewriting rule $x - \beta$ is computable in parallel, then*

$$\#\mathcal{A} \geq \max\{|m_\beta(0)|, |m_\beta(1)|\}\,.$$

Moreover, if β is such that it has a real conjugate greater than 1, then

$$\#\mathcal{A} \geq \max\{|m_\beta(0)|, |m_\beta(1)| + 2\}\,.$$

Proof. By Theorem 4.10, there are all representatives modulo β and modulo $\beta - 1$ in the alphabet $\mathcal{A}$. The numbers of congruence classes are $|m_\beta(0)|$ and $|m_{\beta-1}(0)|$ by Theorem 3.7. Obviously, $m_{\beta-1}(x) = m_\beta(x + 1)$. Thus $m_{\beta-1}(0) = m_\beta(1)$.

Let ϕ be a mapping which defines the parallel addition. According to Lemma 4.11, we may asssume that β is real and greater than 1 in the proof of the second part. The assumption $1 \in \mathcal{A}[\beta]$ implies that $\Lambda > 0$. Thus, there are at least three elements in the alphabet $\mathcal{A}$, because $\mathcal{A} \ni \phi(\Lambda, \ldots, \Lambda) \neq \lambda$ and $\mathcal{A} \ni \phi(\Lambda, \ldots, \Lambda) \neq \Lambda$ by Lemma 4.9. It also implies that there are at least two representatives modulo $\beta-1$ in the alphabet in the class which contains Λ, since $\phi(\Lambda, \ldots, \Lambda) \equiv_{\beta-1} \Lambda$.

If $\lambda \equiv_{\beta-1} \Lambda$, there must be one more element of the alphabet $\mathcal{A}$ in this class, since $\lambda \neq \Lambda$. Therefore, $\#\mathcal{A} \geq |m_\beta(1)| + 2$.

The case that $\lambda \not\equiv_{\beta-1} \Lambda$ is divided into two subcases. Suppose now that $\lambda \neq 0$. Then $\phi(\lambda, \ldots, \lambda) \neq \lambda$ and hence there is one more element in the alphabet in the class containing λ. Thus, there are at least two congruence classes which contain at least two elements of the alphabet $\mathcal{A}$. Therefore, $\#\mathcal{A} \geq |m_\beta(1)| + 2$.

If $\lambda = 0$, then all elements of $\mathcal{A}+\mathcal{A}$ are nonnegative and $\phi(b, \ldots, b) \neq 0$ for all $b \in (\mathcal{A}+\mathcal{A})\setminus 0$. Suppose for contradiction, that there is no nonzero element of the alphabet $\mathcal{A}$ congruent to 0. We know that there is at least one represenative of each congruence class class modulo

$\beta - 1$ in $\mathcal{A}$ and at least two representatives in the congruence class which contains Λ. Let $k \in \mathbb{N}$ denote the number of elements which are in $\mathcal{A}$ extra to one element in each congruence class, i.e., $\#A = |m_\beta(1)| + k$. For $d \in \Lambda + \mathcal{A}$, the value $\phi(d, \ldots, d) \in \mathcal{A}$ is not congruent to 0 as it is nonzero and the class containing zero has only one element by the assumption. Therefore, the values $\phi(d, \ldots, d) \in \mathcal{A}$ for $|m_\beta(1)| + k$ distinct letters $d \in \Lambda + \mathcal{A}$ belong to only $|m_\beta(1)| - 1$ congruence classes. Hence, there exists $e_1, \ldots e_k, e_{k+1} \in \mathcal{A}$, pairwise distinct, and $f_1, \ldots f_k, f_{k+1} \in \mathcal{A}$ such that $e_i \neq f_i$ and $\phi(e_i + \Lambda, \ldots, e_i + \Lambda) \equiv_{\beta-1} \phi(f_i + \Lambda, \ldots, f_i + \Lambda)$ for all $i, 1 \leq i \leq k + 1$. Since

$$e_i + \Lambda \equiv_{\beta-1} \phi(e_i + \Lambda, \ldots, e_i + \Lambda) \equiv_{\beta-1} \phi(f_i + \Lambda, \ldots, f_i + \Lambda) \equiv_{\beta-1} f_i + \Lambda\,.$$

also $e_i \equiv_{\beta-1} f_i$ for all $i, 1 \leq i \leq k + 1$. This is a contradiction since it implies that $\#\mathcal{A} = |m_\beta(1)| + k + 1$. Hence, classes containing λ and Λ have both at least one more element of the alphabet $\mathcal{A}$, i.e., $\#\mathcal{A} \geq |m_\beta(1)| + 2$. □

Note that the larger necessary alphabet for a base with a real conjugate greater than one is caused by the fact that minimal and maximal element of $\mathcal{A}$ both require another element in $\mathcal{A}$ which is in the same congruence class. This should be considered when an alphabet for such a base is generated.

BYLO BY FAJN TO JESTE ZOBECNIT NA ABECEDU ZE Z[OMEGA]
NEJAKY POPIS GENEROVANI?

Chapter 5

Different methods of choice in the extending window method

5.1 Different methods in Phase 1

TADY POPSAT RUZNE METODY VYBERU? NEBO TO DAT NEKAM JINAM?

There may be more possible weight coefficients which cover some element of the set $\mathcal{B}+\mathcal{Q}_k$. Let us suppose that we have the list which contains the lists of these candidates for each element of the set $\mathcal{B}+\mathcal{Q}_k$. This list of lists is saved in the variable `candidates` in Algorithm 3. Now, for each element, we check the list of candidates which cover this element and if there is none of them contained in the set $\mathcal{Q}_k$, the smallest (in absolute value) weight coefficient from the list of candidates is added to the set $\mathcal{Q}_k$. If there are more elements with the same absolute value, we deterministically choose one of them. The extension $\mathcal{Q}_{k+1}$ of the set $\mathcal{Q}_k$ is obtained in this manner.

We may slightly improve this procedure: for example we may first extend $\mathcal{Q}_k$ by all single-element lists of `candidates`. These elements may be enough to cover also other elements of $\mathcal{B}+\mathcal{Q}_k$. It implies that the resulting $\mathcal{Q}$ is dependent on the way of selection from `candidates`.

Algorithm 4 describes the search for the list of lists of candidates. For each element $x \in \mathcal{B}+\mathcal{Q}_k$ we build the list of candidates (in the variable `cand_for_x`) so that we test the divisibility of $x-a$ by the base β for all letters $a \in \mathcal{A}$. In the positive case, the result of division is appended to `cand_for_x` as a possible weight coefficient. We remark that Theorem 3.2 is used to check the divisibility.

We can improve the performance of Algorithm 4 by substituting the set $\mathcal{B}+\mathcal{Q}_k$ by $(\mathcal{B}+\mathcal{Q}_k) \setminus (\mathcal{B}+\mathcal{Q}_{k-1})$ on the line 2 because

$$\mathcal{B}+\mathcal{Q}_{k-1} \subset \mathcal{A}+\beta\mathcal{Q}_k \subset \mathcal{A}+\beta\mathcal{Q}_{k+1}$$

for any $\mathcal{Q}_{k+1} \supset \mathcal{Q}_k$. Thus there is no need to check whether the elements of $\mathcal{B}+\mathcal{Q}_{k-1}$ are covered by some weight coefficient from $\mathcal{Q}_k$ in Algorithm 3.

PODLE LEMMA 4.2 DENOTE $C := \mathrm{MAX}\{\|B-A\|_\beta : A \in \mathcal{A}, B \in \mathcal{B}\}$. CONSEQUENTLY, SET $R := \frac{C}{|\gamma|-1}$ AND $\mathcal{Q} := \{Q \in \mathbb{Z}[\omega] : \|Q\|_\beta \leq R\}$.

Algorithm 3 Extending intermediate weight coefficients set

Input: previous weight coefficients set $\mathcal{Q}_k$, method number $M \in \{12, 13, 14, 15, 16\}$

$\widetilde{\mathcal{Q}} := \mathcal{Q}_k$
if $M \in \{12, 13, 14\}$ **then**
    for all $D_x \in D$ **do**
        if $\#D_x = 1$ **then**
            Add the element of D_x to $\widetilde{\mathcal{Q}}$
        end if
    end for
end if
$\mathcal{Q}_{k+1} := \widetilde{\mathcal{Q}}$
for all $D_x \in D$ **do**
    if $D_x \cap \widetilde{\mathcal{Q}} = \emptyset$ **then**
        if $M = 12$ **then**
            Add all smallest elements in absolute value of D_x to $\mathcal{Q}_{k+1}$
        else if $M = 13$ **then**
            Add all smallest elements in β-norm of D_x to $\mathcal{Q}_{k+1}$
        else if $M = 14$ **then**
            Add all elements of D_x to$\mathcal{Q}_{k+1}$
        else if $M = 15$ **then**
            Add all smallest elements in β-norm of D_x to $\mathcal{Q}_{k+1}$
        else if $M = 16$ **then**
            Add all smallest elements in absolute value of D_x to $\mathcal{Q}_{k+1}$
        end if
    end if
end for
return $\mathcal{Q}_{k+1}$

Algorithm 4 Search for set of candidates D

Input: the previous weight coefficients set $\mathcal{Q}_k$, alternatively also the set $\mathcal{Q}_{k-1}$

$D := \{\emptyset\}$
for all $x \in \mathcal{B} + \mathcal{Q}_k$ **do** {Alternatively, $x \in (\mathcal{B} + \mathcal{Q}_k) \setminus (\mathcal{B} + \mathcal{Q}_{k-1})$}
    $D_x := \emptyset$
    for all $a \in \mathcal{A}$ **do**
        if $(x - a)$ is divisible by β in $\mathbb{Z}[\omega]$ (using Theorem 3.2) **then**
            Add $\frac{x-a}{\beta}$ to D_x
        end if
    end for
    Add the set D_x to D
end for
return D

5.2 Different methods in Phase 2

TADY POPSAT RUZNE METODY VYBERU? NEBO TO DAT NEKAM JINAM?

Definition 5.1. Let n be a positive integer. Let $(x_1, \ldots, x_n)$ and $(y_1, \ldots, y_n)$ are elements of $\mathbb{Z}^n$. We say that $(x_1, \ldots, x_n) \preceq (y_1, \ldots, y_n)$ if

$$x_1 < y_1 \text{ or } (x_1 = y_1 \wedge (x_2, \ldots, x_n) \preceq (y_2, \ldots, y_n)) .$$

If x and y are elements of $\mathbb{Z}[\omega]$, then we say that x is *lexicographically smaller* than y if

$$\pi(x)^T \preceq \pi(y)^T ,$$

where ^{T} indicates transpose of column vector.

For construction of the set $\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]}$ we first choose such elements of $\mathcal{Q}_{[w_j,\ldots,w_{j-m+2}]}$ which are the only possible to cover some value $x \in w_0 + \mathcal{Q}_{[w_{j-1},\ldots,w_{j-m+1}]}$. Other elements from $\mathcal{Q}_{[w_j,\ldots,w_{j-m+2}]}$ which cover an uncovered value are added one by one to $\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]}$ until each x equals $a + \beta q_j$ for some q_j in $\mathcal{Q}_{[w_j,\ldots,w_{j-m+1}]}$ and $a \in \mathcal{A}$. The pseudocode is in Algorithm 5.

Algorithm 5 Search for set $\mathcal{Q}_{[w_0,\ldots,w_{-k}]}$

Input: Input digit w_0, set of possible carries $\mathcal{Q}_{[w_{-1},\ldots,w_{-k}]}$, previous set of possible weight coefficients $\mathcal{Q}_{[w_0,\ldots,w_{-(k-1)}]}$

1: $C := \{\emptyset\}$
2: **for all** $x \in w_0 + \mathcal{Q}_{[w_{-1},\ldots,w_{-k}]}$ **do**
3: $\quad C_x := \{q_0 \in \mathcal{Q}_{[w_0,\ldots,w_{-(k-1)}]} \colon \exists\, a \in \mathcal{A} : x = a + \beta q_0\}$
4: $\quad$ Add C_x to C
5: **end for**
6: $\mathcal{Q}'_{[w_0,\ldots,w_{-k}]} := \emptyset$
7: **for all** $C_x \in C$ **do**
8: $\quad$ **if** $\#C_x = 1$ **then**
9: $\quad\quad$ Add the element $q \in C_x$ to $\mathcal{Q}'_{[w_0,\ldots,w_{-k}]}$
10: $\quad\quad$ Remove sets $C_{x'}$ such that $q \in C_{x'}$ from the set C
11: $\quad$ **end if**
12: **end for**
13: **while** $C \neq \emptyset$ **do**
14: $\quad$ By Algorithm 6, pick an element q from $\bigcup C$
15: $\quad$ Add the element q to $\mathcal{Q}'_{[w_0,\ldots,w_{-k}]}$
16: $\quad$ Remove sets C_x such that $q \in C_x$ from the set C
17: **end while**
18: $\mathcal{Q}_{[w_0,\ldots,w_{-k}]} := \mathcal{Q}'_{[w_0,\ldots,w_{-k}]}$
19: **return** $\mathcal{Q}_{[w_0,\ldots,w_{-k}]}$

Algorithm 6 Choose one element from the set of covering C

Input: set of coverings C, method number $M \in \{9, 15, 22, 23\}$, already added elements $\mathcal{Q}'_{[w_0,\ldots,w_{-k}]}$

if $M = 9$ **then**
    $g :=$ point of gravity of elements of $\bigcup C$ as complex numbers
    $T :=$ elements of $\bigcup C$ which are closest to g in absolute value
else
    $m := \min\{\#C_x \colon C_x \in C\}$
    $C' := \{C_x \in C \colon \#C_x = m\}$
    if $M = 15$ **then**
        $g :=$ point of gravity of elements of $\mathcal{Q}'_{[w_0,\ldots,w_{-k}]}$ as complex numbers
        $T :=$ elements of $\bigcup C'$ which are closest to g in absolute value
    else if $M = 22$ **then**
        $T :=$ elements of $\bigcup C'$ which are smallest in absolute value
    else if $M = 23$ **then**
        $T :=$ elements of $\bigcup C'$ which are smallest in β-norm
    end if
end if
return the lexicographically smallest element of T according to Definition 5.1

Chapter 6

Design and implementation

6.1 Modified Phase 2

In this section, we introduce algorithms based on the theorems proven in Chapter 4. We recall an algorithm for checking convergence of Phase 2 for $bb\dots b$ inputs. Next, we show that it is possible to make Phase 2 stable by wrapping the choice of a set of possible weight coefficients into a simple while loop. Finally, we present an algorithm for Phase 2 which includes all modifications – the menshioned check for $bb\dots b$ inputs and control of convergence by searching for an infinite walk in Rauzy graph G_k.

Checking $bb\dots b$ inputs

Algorithm 7 was proposed in [8]. It checks whether Phase 2 stops when it processes input digits $bb\dots b$. Sets $\mathcal{Q}^m_{[b]}$ can be easily constructed separately for each $b \in \mathcal{B}$ for given m. We build the set $\mathcal{Q}^m_{[b]}$ for input digits $bb\dots b$ in the same way as in Phase 2. This means that we first search for $\mathcal{Q}^1_{[b]}$ such that

$$b + \mathcal{Q} \subset \mathcal{A} + \beta\mathcal{Q}^1_{[b]}\,.$$

Until the set $\mathcal{Q}^m_{[b]}$ contains only one element, we increment the length of window m and, using Algorithm 8, we find a subset $\mathcal{Q}^{m+1}_{[b]}$ of the set $\mathcal{Q}^m_{[b]}$ such that

$$b + \mathcal{Q}^m_{[b]} \subset \mathcal{A} + \beta\mathcal{Q}^{m+1}_{[b]}\,.$$

In each iteration, we check whether the set $\mathcal{Q}^{m+1}_{[b]}$ is strictly smaller than the set $\mathcal{Q}^m_{[b]}$. If not, we know by Theorem 4.6 that Phase 2 does not converge because $\#\mathcal{Q}^m_{[b]}$ is eventually a constant greater than 2.

Hence, non-finiteness of Phase 2 can be revealed by running Algorithm 7 for each input digit $b \in \mathcal{B}$.

Stable Phase 2

We remind that for checking whether Phase 2 converges we assume that it is stable. An algorithm of choice of possible weight coefficients set can be easily modified to ensure that

$$\mathcal{Q}_{[w_{-1},\dots,w_{-k}]} = \mathcal{Q}_{[w_{-1},\dots,w_{-(k-1)}]} \implies \mathcal{Q}_{[w_0,\dots,w_{-k}]} = \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$$

Algorithm 7 Check the input $bb\dots b$

Input: Weight coefficient set $\mathcal{Q}$, digit $b \in \mathcal{B}$

Ouput: TRUE if there is a unique weight coefficient for input $bb\dots b$, FALSE otherwise

1: Find minimal set $\mathcal{Q}^1_{[b]} \subset \mathcal{Q}$ such that

$$b + \mathcal{Q} \subset \mathcal{A} + \beta\mathcal{Q}^1_{[b]}\,.$$

2: $m := 1$
3: **while** $\#\mathcal{Q}^m_{[b]} > 1$ **do**
4: $\quad m := m + 1$
5: $\quad$ By Algorithm 8, find minimal set $\mathcal{Q}^m_{[b]} \subset \mathcal{Q}^{m-1}_{[b]}$ such that

$$b + \mathcal{Q}^{m-1}_{[b]} \subset \mathcal{A} + \beta\mathcal{Q}^m_{[b]}\,.$$

6: $\quad$ **if** $\#\mathcal{Q}^m_{[b]} = \#\mathcal{Q}^{m-1}_{[b]}$ **then**
7: $\quad\quad$ **return** FALSE
8: $\quad$ **end if**
9: **end while**
10: **return** TRUE

for all $(w_{-1}, \dots, w_{-k}) \in \mathcal{B}^k$. If the algorithm is deterministic, then the set $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ is determined by the set $\mathcal{Q}_{[w_{-1},\dots,w_{-k}]}$, $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ and w_0. Similarly, the set $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is determined by the set $\mathcal{Q}_{[w_{-1},\dots,w_{-(k-1)}]}$, $\mathcal{Q}_{[w_0,\dots,w_{-(k-2)}]}$ and w_0.

Suppose that $\mathcal{Q}_{[w_{-1},\dots,w_{-k}]} = \mathcal{Q}_{[w_{-1},\dots,w_{-(k-1)}]}$. Now, the only difference between $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ and $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is that $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ is searched as a subset of $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$, whereas $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ as a subset of $\mathcal{Q}_{[w_0,\dots,w_{-(k-2)}]}$. In orded to find $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$, we use Algorithm 5 repeatedly instead of once. In each iteration, the input digit w_0 and the set $\mathcal{Q}_{[w_{-1},\dots,w_{-(k-1)}]}$ remains the same but we search for $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ as a subset of $\mathcal{Q}'_w$, where $\mathcal{Q}'_w$ is the output of the previous iteration or $\mathcal{Q}_{[w_0,\dots,w_{-(k-2)}]}$ in the first iteration. The algorithm stops when $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]} = \mathcal{Q}'_w$. This loop is desribed in Algorithm 8.

Now, when the set $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ is searched as a subset of $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ in the same manner, it runs with the same inputs as the last iteration of the search for $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$. Hence, $\mathcal{Q}_{[w_0,\dots,w_{-k}]} = \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$.

Algorithm 8 Stable search for possible weight coefficient set $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$

Input: Input digit w_0, set of possible carries $\mathcal{Q}_{[w_{-1},\dots,w_{-k}]}$, previous set of possible weight coefficients $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$

1: $\mathcal{Q}'_w := \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$
2: **while** TRUE **do**
3: $\quad$ By Algorithm 5, find the set of possible weight coefficients $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ as a subset of $\mathcal{Q}'_w$ instead of the previous set of weight coefficients $\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$.
4: $\quad$ **if** $\mathcal{Q}_{[w_0,\dots,w_{-k}]} = \mathcal{Q}'_w$ **then**
5: $\quad\quad$ **return** $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$
6: $\quad$ **end if**
7: $\quad \mathcal{Q}'_w := \mathcal{Q}_{[w_0,\dots,w_{-k}]}$
8: **end while**

Infinite walk in Rauzy graph G_k

As the Rauzy graph G_k is finite, there exist an infinite walk in G_k if and only if there exists an oriented cycle. Algorithm 9 checks whether some walk starting in $(w_{-1}, \ldots, w_{-k})$ enters such a cycle eventually. Firts, it checks if the vertex $(w_{-1}, \ldots, w_{-k})$ is in the graph G_k. If yes, all vertices which are accessible by appropriately oriented edge from $(w_{-1}, \ldots, w_{-k})$ are entered by Algorithm 10. It is called recursively to enter all accessible vertices from the given one. The already traversed path is passed in each call and if an vertex is visited second time, then the cycle is found and algorithm ends. If all branches of the recursion ends up without visiting some vertex twice, then there is no cycle and thus no infinite path.

Note that saving of the traversed path requires only the label of the first vertex and last digits of the labels of next visited vertices due to the construction of Rauzy graph.

Algorithm 9 Check if there is in an infinite walk in G_k starting in $(w_{-1}, \ldots, w_{-k})$

Input: Rauzy graph G_k, combination of input digits $(w_{-1}, \ldots, w_{-k})$
Ouput: Return TRUE if TRUE is returned in any step of the recursion, that is when a walk $w_1, w_2, \ldots$ enters an oriented cycle in G_k. Otherwise return FALSE.

1: **if** $(w_{-1}, \ldots, w_{-k}) \in G_k$ **then**
2: By Algorithm 10, enter next vertices from $(w_{-1}, \ldots, w_{-k})$ with the traversed path $(w_{-1}, \ldots, w_{-k})$.
3: **else**
4: **return** FALSE
5: **end if**
6: **return** FALSE

Algorithm 10 Enter vertices from $(w'_{-1}, \ldots, w'_{-k})$

Input: Rauzy graph G_k, vertex $(w'_{-1}, \ldots, w'_{-k})$, traversed path $(w_1, \ldots, w_l)$.

1: **for all** $w'_{k+1} \in \mathcal{B}$ such that $(w'_{-1}, \ldots, w'_{-k}) \to (w'_{-2}, \ldots, w'_{-(k+1)}) \in G_k$ **do**
2: **if** $(w'_{-2}, \ldots, w'_{-(k+1)})$ is in the traversed path $(w_1, \ldots, w_l)$ **then**
3: **return** TRUE
4: **else**
5: By Algorithm 10, enter next vertices from $(w'_{-2}, \ldots, w'_{-(k+1)})$ with the traversed path $(w_1, \ldots, w_l, w'_{(k+1)})$.
6: **end if**
7: **end for**

Phase 2 with convergence control

Algorithm 11 modifies the basic proposal of Phase 2 to reveal its possible non-convergence. First, the necessary condition given by Theorem 4.6 is checked, i.e., the convergence of Phase 2 for inputs $bb \ldots b$ is verified by Algorithm 7 for all $b \in \mathcal{B}$.

Then we proceed in the same way as in Algorithm 2 with the following modifications: it is sufficient to process only $(w_0, \ldots, w_{-k}) \in \mathcal{B}^{k+1}$ such that $\#\mathcal{Q}_{[w_0,\ldots,w_{-(k-1)}]} > 1$ since if $\#\mathcal{Q}_{[w_0,\ldots,w_{-(k-1)}]} = 1$, then $\mathcal{Q}_{[w_0,\ldots,w_{-k'}]} = \mathcal{Q}_{[w_0,\ldots,w_{-(k-1)}]}$ for all $k' > k$.

Moreover, we check possible non-convergence according to Theorem 4.7. If $\mathcal{Q}_{[w_0,\dots,w_{-k}]} = \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$, then the vertex $\mathcal{Q}_{[w_0,\dots,w_{-k}]}$ is added into the Rauzy graph G_{k+1} and the Rauzy graph G_k is examined by Algorithm 9 whether it contains an infinite walk starting in $(w_{-1},\dots,w_{-k})$. Note that $\mathcal{Q}_{[w_0,\dots,w_{-k}]} = \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ is a necessary condition for $(w_{-1},\dots,w_{-k})$ be a vertex of G_k. It also implies that $\#\mathcal{Q}_{[w_0,\dots,w_{-k}]} > 1$.

We remark that the non-convergence caused by an input $bb\dots b$ for some $b \in \mathcal{B}$ would be revealed also as a cycle in Rauzy graph. Nevertheless, the number of calls of Algorithm 5 during Algorithm 7 is at most $\#Q$ for each $b \in \mathcal{B}$, while it grows exponentially with the length of window in search for a weight function. Thus, we save a lot of computational time in cases which fail already on $bb\dots b$ inputs. At the same time, the costs of this check are low in other cases.

CASTECNE VYUZIT Z VYZKUMAKU??
PRIDAT POZNAMKU O GOOGLE TABULCE, HROMADNE TESTOVANI

Algorithm 11 Modified search for a weight function (Phase 2)

Input: weight coefficients set $\mathcal{Q}$

for all $b \in \mathcal{B}$ **do**
  if not Check the input $bb\dots b$ by Algorithm 7 **then**
    return Phase 2 does not converge for input $bb\dots b$.
  end if
end for
for all $w_0 \in \mathcal{B}$ **do**
  By Algorithm 5, find set $\mathcal{Q}_{[w_0]} \subset \mathcal{Q}$ such that

$$w_0 + \mathcal{Q} \subset \mathcal{A} + \beta\mathcal{Q}_{[w_0]}$$

end for
$k := 0$
while $\max\{\#\mathcal{Q}_{[w_0,\dots,w_{-k}]} : (w_0,\dots,w_{-k}) \in \mathcal{B}^{k+1}\} > 1$ **do**
  $k := k + 1$
  for all $\{(w_0,\dots,w_{-k}) \in \mathcal{B}^{k+1} \colon \#\mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]} > 1\}$ **do**
    By Algorithm 8, find set $\mathcal{Q}_{[w_0,\dots,w_{-k}]} \subset \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ such that

$$w_j + \mathcal{Q}_{[w_{-1},\dots,w_{-k}]} \subset \mathcal{A} + \beta\mathcal{Q}_{[w_0,\dots,w_{-k}]}\,.$$

    if $\mathcal{Q}_{[w_0,\dots,w_{-k}]} = \mathcal{Q}_{[w_0,\dots,w_{-(k-1)}]}$ **and** $k \geq 2$ **then**
      Add the vertex $(w_0,\dots,w_{-k})$ to the Rauzy graph G_{k+1}.
      if Check infinite walks in G_k starting in $(w_{-1},\dots,w_{-k})$ by Alg. 9 **then**
        return Phase 2 does not converge.
      end if
    end if
  end for
end while
$r := k + 1$??????????????????????????
for all $l \in \mathbb{N}, l \leq r$ **do**
  for all $\{(w_0,\dots,w_{-l}) \in \mathcal{B}^{l+1} \colon \mathcal{Q}_{[w_0,\dots,w_{-l}]}$ was found, $\#\mathcal{Q}_{[w_0,\dots,w_{-l}]} = 1\}$ **do**
    for all $(w_{-(l+1)},\dots,w_{-r}) \in \mathcal{B}^{r-l}$ **do**
      $q(w_0,\dots,w_{-r}) :=$ only element of $\mathcal{Q}_{[w_0,\dots,w_{-l}]}$
    end for
  end for
end for
return q

Chapter 7

Testing

TABULKA VSECH USPESNYCH PLUS NEJAKE NEUSPESNE A OKOMENTOVAT

7.1 Comparing different choices in Phase 1 and 2

VSECHNY MILENINY RUCNE SPOCTENE, TAKY BY TADY MOHLY BYT I NEJAKE NEUSPESNE

7.2 Quadratic bases with non-integer alphabet

Ex.	ω	m_ω	β	m_β	conj.	$\#\mathcal{A}$	min.	Phase 1	$\#\mathcal{Q}$	$bb\dots b$	Phase
D.5	$\frac{1}{2}i\sqrt{11}+\frac{1}{2}$	t^2-t+3	$-\omega+1$	x^2-x+3	no	3	yes	✓	9	✓	✗

Table 7.1: Quadratic bases with non-integer alphabet

7.3 Quadratic bases with integer alphabet

Ex.	ω	m_ω	β	m_β	conj.	$\#\mathcal{A}$	min.	$\#\mathcal{Q}$	$bb\dots b$	Phase 2	m
D.5	$\frac{1}{2}i\sqrt{11}-\frac{1}{2}$	t^2+t+3	$-\omega-1$	x^2+x+3	yes	7	no	9	✓	✓	2

Table 7.2: Quadratic bases with integer alphabet

7.4 Cubic bases

Name	ω	m_ω	β	m_β	conj.	$\#\mathcal{A}$	min.	$\#\mathcal{Q}$				
								12	13	14	15	16
Eisenstein_1–block_complex	$\frac{1}{2}i\sqrt{3}-\frac{1}{2}$	t^2+t+1	$\omega-1$	$x^2+3\,x+3$	no	7	yes	19	19	19	19	19
Eisenstein_1–block_integer	$\frac{1}{2}i\sqrt{3}-\frac{1}{2}$	t^2+t+1	$\omega-1$	$x^2+3\,x+3$	no	7	yes	57	57	139	57	57
Eisenstein_2–block_complex	$\frac{1}{2}i\sqrt{3}-\frac{1}{2}$	t^2+t+1	$-3\,\omega$	$x^2-3\,x+9$	no	14	no	17	17	17	17	17
Eisenstein_2–block_integer	$\frac{1}{2}i\sqrt{3}-\frac{1}{2}$	t^2+t+1	$-3\,\omega$	$x^2-3\,x+9$	no	16	no	26	26	26	26	26
Penney_1–block_complex	$i-1$	$t^2+2\,t+2$	ω	$x^2+2\,x+2$	no	5	yes	45	45	45	45	45
Penney_1–block_integer	i	t^2+1	$\omega-1$	$x^2+2\,x+2$	no	5	yes	27	27	95	27	27
Penney_2–block_integer	i	t^2+1	$-2\,\omega$	x^2+4	no	9	no	27	27	27	27	27
Quadratic+1+0–2_integer	$\sqrt{2}$	t^2-2	ω	x^2-2	yes	3	yes	9	9	9	9	9
Quadratic+1+0–21_integer	$-\frac{1}{2}\sqrt{21}+\frac{3}{2}$	$t^2-3\,t-3$	$2\,\omega-3$	x^2-21	yes	22	yes	9	9	9	9	9
Quadratic+1+0–3_integer	$\sqrt{3}-1$	$t^2+2\,t-2$	$-\omega-1$	x^2-3	yes	4	yes	9	9	9	9	9
Quadratic+1+0–5_integer	$\frac{1}{2}\sqrt{5}-\frac{1}{2}$	t^2+t-1	$2\,\omega+1$	x^2-5	yes	8	no	9	9	9	9	9
Quadratic+1+2+3_complex	$i\sqrt{2}-1$	$t^2+2\,t+3$	$-\omega-2$	$x^2+2\,x+3$	no	6	yes	27	27	27	27	27
Quadratic+1+3+4_complex	$\frac{1}{2}i\sqrt{7}-\frac{1}{2}$	t^2+t+2	$\omega-1$	$x^2+3\,x+4$	no	8	yes	20	20	21	20	19
Quadratic+1+3+5_complex1	$\frac{1}{2}i\sqrt{11}-\frac{3}{2}$	$t^2+3\,t+5$	ω	$x^2+3\,x+5$	no	9	yes	11	17	19	17	11
Quadratic+1+3+5_complex2	$\frac{1}{2}i\sqrt{11}-\frac{3}{2}$	$t^2+3\,t+5$	ω	$x^2+3\,x+5$	no	9	yes	33	39	43	39	33
Quadratic+1+4+5_complex1	i	t^2+1	$\omega-2$	$x^2+4\,x+5$	no	10	yes	17	17	19	17	17
Quadratic+1+4+5_complex2	i	t^2+1	$\omega-2$	$x^2+4\,x+5$	no	10	yes	17	17	17	17	17

Table 7.3: sdvhbsdjhvjsd

Name	Methods Phase 1	$\#\mathcal{Q}$	9			15			22			23		
			bbb	Ph.2	r	*bbb*	Ph.2	r	*bbb*	Ph.2	r	*bbb*	Ph.2	r
Eisenstein_1–block_complex	12, 13, 14, 15, 16	19	✓	✓	3	✓	✓	3	✓	✓	3	✓	✓	3
Eisenstein_1–block_integer	12, 13, 15, 16	57	✗	-	-	✗	-	-	✗	-	-	✗	-	-
	14	139	✗	-	-	✗	-	-	✗	-	-	✗	-	-
Eisenstein_2–block_complex	12, 13, 14, 15, 16	17	✗	-	-	✗	-	-	✗	-	-	✗	-	-
Eisenstein_2–block_integer	12, 13, 14, 15, 16	26	✗	-	-	✗	-	-	✗	-	-	✗	-	-
Penney_1–block_complex	12, 13, 14, 15, 16	45	✓	✓	6	✓	✓	6	✓	✓	6	✓	✓	6
Penney_1–block_integer	12, 13, 15, 16	27	✗	-	-	✗	-	-	✗	-	-	✗	-	-
	14	95	✗	-	-	✗	-	-	✗	-	-	✗	-	-
Penney_2–block_integer	12, 13, 14, 15, 16	27	✓	✓	5	✓	✓	5	✓	✓	5	✓	✓	5
Quadratic+1+0–2_integer	12, 13, 14, 15, 16	9	✓	✓	5	✓	✓	5	✓	✓	5	✓	✓	4
Quadratic+1+0–21_integer	12, 13, 14, 15, 16	9	✓	✓	4	✓	✓	4	✓	✓	4	✓	✓	4
Quadratic+1+0–3_integer	12, 13, 14, 15, 16	9	✓	✓	4	✓	✓	5	✓	✓	5	✓	✓	5
Quadratic+1+0–5_integer	12, 13, 14, 15, 16	9	✗	-	-	✓	✓	3	✓	✓	2	✓	✓	2
Quadratic+1+2+3_complex	12, 13, 14, 15, 16	27	✓	✗	-	✓	✓	7	✓	✗	-	✓	✗	-
Quadratic+1+3+4_complex	12	20	✓	✓	7	✓	✓	7	✓	✗	-	✗	-	-
	16	19	✓	✗	-	✓	✓	7	✓	✗	-	✗	-	-
	13, 15	20	✓	✗	-	✓	✓	7	✓	✗	-	✗	-	-
	14	21	✓	✓	7	✓	✓	7	✓	✗	-	✗	-	-
Quadratic+1+3+5_complex1	14	19	✗	-	-	✗	-	-	✗	-	-	✗	-	-
	12, 16	11	✗	-	-	✓	✗	-	✗	-	-	✗	-	-
	13, 15	17	✗	-	-	✗	-	-	✗	-	-	✗	-	-
Quadratic+1+3+5_complex2	12, 16	33	✗	-	-	✓	✗	-	✗	-	-	✗	-	-
	13, 15	39	✗	-	-	✓	✗	-	✓	✗	-	✗	-	-
	14	43	✗	-	-	✓	✗	-	✓	✗	-	✗	-	-
Quadratic+1+4+5_complex1	14	19	✗	-	-	✗	-	-	✗	-	-	✗	-	-
	12, 13, 15, 16	17	✗	-	-	✗	-	-	✗	-	-	✗	-	-
Quadratic+1+4+5_complex2	12, 13, 14, 15, 16	17	✓	✓	3	✓	✓	3	✓	✓	3	✓	✓	3

Table 7.4: sdvhbsdjhvjsd

Ex.	ω	m_ω	β	m_β	conj.	$\#\mathcal{A}$	min.	Phase 1	$\#\mathcal{Q}$	$bb\ldots b$	Phase 2	m
D.5	$\frac{1}{2}i\sqrt{11}+\frac{1}{2}$	t^2-t+3	$-\omega+1$	x^2-x+3	no	3	yes	✓	9	✓	✗	-

Table 7.5: xxx

Bibliography

[1] S. Akiyama, J.M. Thuswaldner, and T. Zaïmi, *Comments on the height reducing property II*, Indagationes Mathematicae **26** (2015), no. 1, 28–39.

[2] S. Akiyama and T. Zaïmi, *Comments on the height reducing property*, Central European Journal of Mathematics **11** (2013), no. 9, 1616–1627.

[3] C. Frougny, P. Heller, E. Pelantová, and M. Svobodová, *k-block parallel addition versus 1-block parallel addition in non-standard numeration systems*, Theoret. Comput. Sci. **543** (2014), 52–67.

[4] C. Frougny, E. Pelantová, and M. Svobodová, *Parallel addition in non-standard numeration systems*, Theoret. Comput. Sci. **412** (2011), 5714–5727.

[5] C. Frougny, E. Pelantová, and M. Svobodová, *Minimal digit sets for parallel addition in non-standard numeration systems*, J. Integer Seq. **16** (2013), 36.

[6] R. A. Horn and C. R. Johnson, *Matrix analysis*, Cambridge University Press, 1990.

[7] P. Kornerup, *Necessary and sufficient conditions for parallel, constant time conversion and addition*, Proc. 14th IEEE Symposium on Computer Arithmetic (1999), 152–155.

[8] J. Legerský, *Construction of algorithms for parallel addition*, Research thesis, Czech Technical University in Prague, Faculty of Nuclear Science, Czech Republic, 2015.

[9] A. M. Nielsen and P. Kornerup, *Redundant radix representations of rings*, IEEE Trans. Comput. **48** (1999), 1153–1165.

[10] M. Svobodová, Private communication, 2014–2015.

Appendices

A Illustration of Phase 1

STEJNE JAK VE VYZKUMAKU?

B Illustration of Phase 2

STEJNE JAK VE VYZKUMAKU?

C GUI

CLOUD, GOOGLE TABULKA?

D Tested examples

Quadratic bases with non-integer alphabet

Example D.1. Parameters:

- Minimal polynomial of ω: $t^2 + t + 2$
- Base $\beta = \omega$
- Minimal polynomial of base: $x^2 + x + 2$
- Alphabet $\mathcal{A} = \{0, \omega + 1, 1, -1\}$
- Input alphabet $\mathcal{B} = \mathcal{A} + \mathcal{A}$

The result of the extending window method is:

1. Phase 1 was successful. The number of elements in the weight coefficient set $\mathcal{Q}$ is 29.
2. There is a unique weight coefficient for input $b, b, \ldots, b$ for all $b \in \mathcal{B}$.
3. Phase 2 was successful. The length of window m of the weight function q is 8.

Example D.2. Parameters:

- Minimal polynomial of ω: $t^2 + t + 2$
- Base $\beta = \omega$
- Minimal polynomial of base: $x^2 + x + 2$
- Alphabet $\mathcal{A} = \{0, \omega + 1, 1, -1\}$
- Input alphabet $\mathcal{B} = \mathcal{A} + \mathcal{A}$

The result of the extending window method is:

1. Phase 1 was successful. The number of elements in the weight coefficient set $\mathcal{Q}$ is 29.
2. There is a unique weight coefficient for input $b, b, \ldots, b$ for all $b \in \mathcal{B}$.
3. Phase 2 was successful. The length of window m of the weight function q is 8.

Example D.3. Parameters:

- Minimal polynomial of ω: $t^2 + 2$
- Base $\beta = \omega$
- Minimal polynomial of base: $x^2 + 2$
- Alphabet $\mathcal{A} = \{0, \omega + 1, -\omega - 1\}$
- Input alphabet $\mathcal{B} = \mathcal{A} + \mathcal{A}$

The result of the extending window method is:

1. Phase 1 was successful. The number of elements in the weight coefficient set $\mathcal{Q}$ is 9.
2. There is a unique weight coefficient for input $b, b, \ldots, b$ for all $b \in \mathcal{B}$.
3. Phase 2 was successful. The length of window m of the weight function q is 4.

Example D.4. Parameters:

- Minimal polynomial of ω: $t^2 + 2$
- Base $\beta = \omega$
- Minimal polynomial of base: $x^2 + 2$
- Alphabet $\mathcal{A} = \{0, 1, -1\}$
- Input alphabet $\mathcal{B} = \mathcal{A} + \mathcal{A}$

The result of the extending window method is:

1. Phase 1 was successful. The number of elements in the weight coefficient set $\mathcal{Q}$ is 9.
2. There is a unique weight coefficient for input $b, b, \ldots, b$ for all $b \in \mathcal{B}$.
3. Phase 2 was successful. The length of window m of the weight function q is 4.

Quadratic bases with integer alphabet

Example D.5. Parameters:

- Minimal polynomial of ω: $t^2 + t + 3$
- Base $\beta = -\omega - 1$
- Minimal polynomial of base: $x^2 + x + 3$
- Alphabet $\mathcal{A} = \{-3, -2, -1, 0, 1, 2, 3\}$
- Input alphabet $\mathcal{B} = \{-4, -3, -2, -1, 0, 1, 2, 3, 4\}$

The result of the extending window method is:

1. Phase 1 was successful. The number of elements in the weight coefficient set $\mathcal{Q}$ is 9.
2. There is a unique weight coefficient for input $b, b, \ldots, b$ for all $b \in \mathcal{B}$.
3. Phase 2 was successful. The length of window m of the weight function q is 2.

Cubic bases

dfb