Generalized Far-Difference Representations

Prakod Ngamlamai

Generalized Far-Difference Representations

Prakod Ngamlamai

Peter Kagey, Advisor

Michael Orrison, Reader

HARVEY MUDD COLLEGE

Department of Mathematics

May, 2023

Abstract

Integers are often represented as a base-b representation by the sum $\sum c_i b^i$. Lekkerkerker and Zeckendorf later provided the rules for representing integers as the sum of Fibonacci numbers. Hannah Alpert then introduced the far-difference representation by providing rules for writing an integer with both positive and negative multiples of Fibonacci numbers. Our work aims to generalize her work to a broader family of linear recurrences. To do so, we describe desired properties of the representations, such as lexicographic ordering, and provide a family of algorithms for each linear recurrence that generate unique representations for any integer. We then prove other interesting properties of these representations such as summand-minimality.

Contents

Abstract **iii**

Acknowledgments **ix**

1 Introduction and Overview **1**
1.1 Base-b Representations and Signed Representations 1
1.2 Summand Minimality 2
1.3 Representations from Linear Recurrences 2
1.4 Signed Representations from Linear Recurrences 3
1.5 Diagrams at a high-level view 4

2 Major Definitions and Preliminaries **7**
2.1 Notation 7
2.2 PLRS 8
2.3 Diagrams 9
2.4 Allowable Blocks 14

3 Algorithm **19**
3.1 Signed Allowable Block 19
3.2 Properties 22

4 Summand Minimality **25**
4.1 Skipponacci Sequences 25
4.2 Base-b Representations 26
4.3 Tribonacci 33

5 Conjectures and data **41**
5.1 Almost Balanced Base-$(2b)$ 41
5.2 Existence of Summand Minimal Diagrams 42

6 Future work **45**
6.1 Statistics of representations 45
6.2 Descriptions of representations for general diagram 45
6.3 Representations that are not described by diagrams 46

A Known Diagrams **47**
A.1 Base-3 47
A.2 Balanced Ternary 47
A.3 Zeckendorf 48
A.4 Far-Difference 48
A.5 Tribonacci Far-Difference 48
A.6 Binary 48
A.7 Width-2 Non-Adjacent Form 48

B Data of representations of integers for some given diagrams **49**
B.1 Base-2 Representation 49
B.2 Base-3 Representations 50
B.3 Fibonacci Representations 51
B.4 Tribonacci Representations 52

Bibliography **53**

List of Figures

1.1 Diagram for the base-3 representation 4
1.2 Diagram for the balanced ternary representation. 5
1.3 Diagram for the Zeckendorf representation. 5
1.4 Diagram for the far-difference representation. 5
1.5 Diagram for an arbitrary representation. 5

2.1 Diagram for the base-3 representation 10
2.2 A possible diagram of an arbitrary PLRS 12

A.1 Diagram of the base-3 representation 47
A.2 Diagram of the balanced ternary representation 47
A.3 Diagram of the Zeckendorf decomposition 48
A.4 Diagram of the far-difference representation 48
A.5 Diagram of the Tribonacci far-difference representation . . . 48
A.6 Diagram of the binary representation 48
A.7 Diagram of the width-2 non-adjacent form representation . . 48

Acknowledgments

I would like to thank my advisor Professor Peter Kagey for being extremely patient with my lack of clarity and delays in finalizing the codes, as well as for the help in generating interesting questions and ideas throughout the project. I would also like to thank my reader Professor Michael Orrison as well as my peers who provided feedback on my works in this project. I am also indebted to Professor Steven Miller and my colleagues in SMALL REU for introducing and working on a related project that familiarize me with this topic.

Chapter 1

Introduction and Overview

In this thesis, we are going to talk about integer representations, a way to write each integer N as a linear sum of integer multiples of elements from a set $S \subset \mathbb{Z}$. The choice of S and how the linear sums are constructed for each number determine the type of integer representations, whether it is base-b representations, signed representations, generalized Zeckendorf decomposition, or generalized far-difference representations. We will also introduce a diagrammatic framework for understanding all of these different integer representations.

1.1 Base-b Representations and Signed Representations

Integer representations vary in their ease of use and readability. The most familiar family of representations are the base-b representations. Base-b representations are a way of representing integers as sums of powers of b, $\{1, b, b^2, b^3, \ldots\}$. For example, the regular base-10 representation would represent 203 as $2 \cdot 10^2 + 0 \cdot 10^1 + 3 \cdot 10^0$. We will refer to multiples of each digit value as 'coefficients' in the rest of the paper. Common among all representations that we will study in this paper, we put in constraints so that the representation for each integer is uniquely determined. To ensure this property in the case of the base-b representation, we restrict the coefficients to $\{0, 1, 2, \ldots, b-1\}$.

We notice, however, that using a different restriction on the coefficients may yield a unique and interesting representation. For example, balanced ternary is a base-3 representation but with the restriction that the coefficients are from $\{-1, 0, 1\}$ instead of $\{0, 1, 2\}$, and was once a candidate along with binary for use in early computing, according to Knuth (1997). Multiplication and addition within the balanced ternary representation is computationally simple, with the benefit of decreasing the number of digits by a factor of $\log_3(2) \approx 0.63$.

Similarly, for base-2, if we expand the coefficient set to $\{-1, 0, 1\}$, then we will lose uniqueness; for example, 1 can be written as $1 = 1 \cdot 2^1 + (-1) \cdot 2^0$ or $1 = 1 \cdot 2^0$. We may impose another condition that we can no longer use powers of 2 which are 'next to each other.' That is, if we have 2^k or -2^k in a representation of a number, then we must have the coefficients of 2^{k-1} and 2^{k+1} be 0. This is called the sparse signed binary representation by Jörg Arndt (2010), and also considered a special case of the non-adjacent forms, which are discussed in Section 6.3.

1.2 Summand Minimality

Later, in Section 4.2, we will prove that the balanced ternary representation shares an interesting property with the sparse signed binary representation, called *summand minimality*.

Definition 1. *Given a set $\{a_1, a_2, \dots\}$, we call a representation $n = \sum_{i=1}^{N} c_i a_i$ summand minimal if the sum of the absolute value of the coefficients $\sum_{i=1}^{N} |c_i|$ is minimal over all such representations.*

Example 1. *As an example, base-3 is not summand minimal because we have $8 = 3 + 3 + 1 + 1$ which makes use of 4 summands, while balanced ternary represents 8 as $9 - 1$ which makes use of $|1| + |-1| = 2$ summands.*

This property has been studied in particular by Cordwell et al. (2018) for a more generalized type of positive representations than base-b representation. In this paper, we will expand the scope of representations considered in their results on summand minimality.

1.3 Representations from Linear Recurrences

One approach to generalizing from base-b representations can be found in the work of Zeckendorf (1972) and Lekkerkerker (1951), who defined a representation of any integer using the Fibonacci sequence. Both base-b and Zeckendorf representations make use of sequences that satisfy linear recurrences. In order to generalize base-b representations, we note that the sequence $\{a_n\}$ used in each base-b representation has the initial term $a_1 = 1$ and recurrence relation $a_{i+1} = ba_i$ for $i \geq 1$. Using a different recurrence relation such as $a_{i+1} = a_i + a_{i-1}$ yields the familiar Fibonacci sequence, and the resulting representation is called the Zeckendorf decomposition.

We generalize both the Fibonacci numbers and powers of integers, with a family of sequences based on linear recurrences called Positive Linear Recurrence Sequences, often abbreviated as PLRS. The integer representations based on the terms within these PLRS using nonnegative coefficients were studied by Miller and Wang Miller and Wang (2012), who coined the resulting representation the Generalized Zeckendorf Decomposition.

In order to discuss the coefficients of a particular representation of a number into terms of these PLRSs, we introduce the following notation that mirrors the base-b notation.

Definition 2. *We write each coefficient as an entry within the tuple, starting from the left with the coefficient of the smallest term $a_1 = 1$ and increasing toward the right.*

For example, in the context of Zeckendorf decomposition, the following tuple represents the unique Zeckendorf decomposition of 19

$$(1, 0, 0, 1, 0, 1) = 1 \cdot F_1 + 1 \cdot F_4 + 1 \cdot F_6 = 1 + 5 + 13 = 19$$

while in base-10, we have

$$19 = (9, 1) = 9 \cdot 1 + 1 \cdot 10$$

Additionally, Cordwell et al. (2018) studied the summand minimal property of the Generalized Zeckendorf Decomposition. We note, however, that summand minimality in this context is limited to comparisons with other representations which also have nonnegative coefficients. The base-3 representation, while not summand minimal when compared to balanced ternary, is summand minimal over other representations which use nonnegative coefficients.

1.4 Signed Representations from Linear Recurrences

Thus, we require a framework which encompasses representations that permit negative coefficients while also preserving uniqueness. While simple recurrence sequences that generate base-b representations can be generalized without much difficulty, we need rather intricate rules for more complex

PLRSs. Our attempt follows the work of Hannah Alpert (2009), who coined the term far-difference representation for the case of Fibonacci numbers with positive or negative coefficients. To ensure uniqueness, Alpert requires that terms that are less than 4 indices apart cannot both be of the same sign, and we are forbidden from simultaneously using terms that are less than 3 indices apart, which means that our coefficients are either -1, 0, or 1. Examples can be found in Table B.5.

Under these conditions, Alpert also proved that the far-difference representation is summand minimal, using a technique which we will employ for our proof of summand minimality of balanced ternary and some other generalized far-difference representations.

1.5 Diagrams at a high-level view

As the example of the far-difference representation shows, we require a framework which lets us produce necessary restrictions for each PLRS to ensure uniqueness. However, we also see that each PLRS contains many distinct generalized far-difference representations depending on the choice of restrictions. To keep track of the coefficients and the terms of the PLRS, we employ a diagrammatic framework inspired by the recent work of John Lentfer (2021). We present the diagrams as examples of how our understanding of representations translate to this framework.

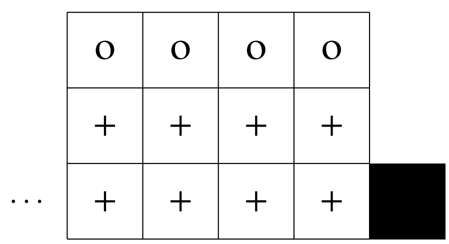


Figure 1.1 Diagram for the base-3 representation

The first diagram shown in Figure 1.1 corresponds to the familiar base-3 representation of integers. We see that each column has 2 + signs, indicating that the maximum coefficient of each digit is 2.

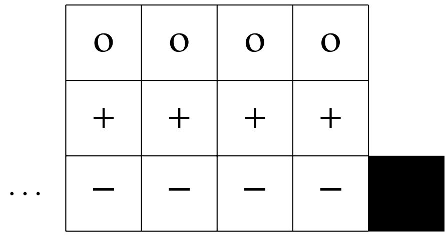


Figure 1.2 Diagram for the balanced ternary representation.

In Figure 1.2 for the balanced ternary representation, there is a − sign in place of a + sign on each column, indicating that we no longer allows +2 as coefficient, instead allowing the negative coefficient −1.

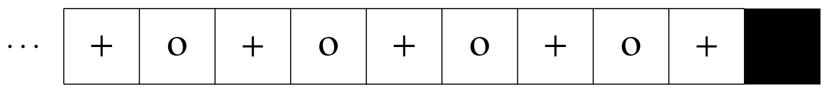


Figure 1.3 Diagram for the Zeckendorf representation.

Figure 1.3 is the diagram for the Zeckendorf decomposition, which is based on the Fibonacci numbers. The arrangement of the + signs where there is an *o* mark in between each pair corresponds to the restriction that we may not use terms that are next to each other like $F_6 = 13$ and $F_7 = 21$.

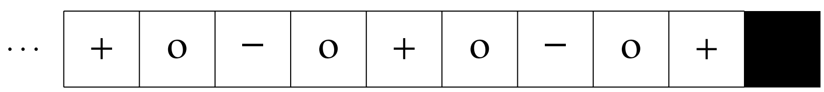


Figure 1.4 Diagram for the far-difference representation.

Figure 1.4 shows the diagram for the far-difference representation. Compared to 1.3, some of the + signs are replaced with − signs. We note that identical signs are spaced at least 4 index apart, while the first − sign appears at the 3rd column from the black square, which corresponds to the restriction on the indices of terms of the same or different signs in the far-difference representation.

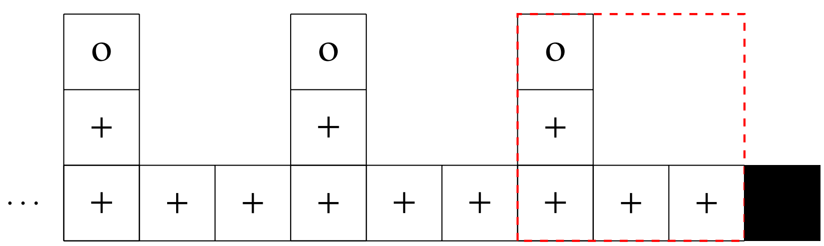


Figure 1.5 Diagram for an arbitrary representation.

Finally, we have an example of a PLRS with recurrence relation a_i =

$a_{i-1} + a_{i-2} + 3a_{i-3}$ within Figure 1.5. One "copy" of the recurrence relation is highlighted in the dotted red square and we see that the columns follows a height pattern of 1, 1, then 3. This particular diagram has no − signs and will therefore give us a representation with nonnegative coefficients. Furthermore, we may expect to see a coefficient of +2, but the fact that the column with 2 + signs is preceded by columns with only 1 + sign complicates our intuition.

From these diagrams, we can summarize our generalization of representations with all coefficients being nonnegative as the simple act of flipping the signs of some squares within the frame defined by a particular PLRS. We note, however, that this generalization is not exhaustive, as will be discussed in Section 6.3.

Chapter 2

Major Definitions and Preliminaries

In this section, we will provide definitions for important terms, some of which have been mentioned within the introduction, which will see extensive use within the rest of the paper.

Note that the word **representation** may refer to the representation of an integer as a linear sum under certain rules of numbers within a given sequence, or it may refer to the map itself from the integers to the possible linear sums. For example, the representation of 8 under the base-3 representation is $2 \cdot 1 + 2 \cdot 3$.

2.1 Notation

We start with some notations for writing representations of a given integer N. As explained in Definition 2, we may write

$$N = (c_1, c_2, \dots, c_M) = \sum_{i=1}^{M} c_i a_i$$

as a tuple or sum where c_i is called the coefficient of a_i, which is the i^{th} term within a given sequence. a_i may also be referred to as the i^{th} digit value. We write N explicitly as a sum when the clarity of the size of each term is required over the ease of comparison when using tuples, such as in calculations with a given small integer.

Example 2. *Suppose we have the sequence of powers of three* $\{1, 3, 9, 27, \dots\}$ *and we wish to find base-*3 *representation of* 17*. We get that* $17 = (2, 2, 1)$*, but might also write* $17 = 2 \cdot 1 + 2 \cdot 3 + 1 \cdot 9$ *or* $17 = 2 \cdot 1 + 2 \cdot 3 + 9$ *for clarity. Note that the order of coefficients, from coefficients of small terms to coefficients of larger terms, is in reverse to how the usual base-b representations are written.*

Furthermore, when writing the first few terms of a sequence, we use curly braces $\{a_1, a_2, \dots\}$. Note, however, that for convenience we may index the terms as $\{a_0, a_1, \dots\}$ in such cases like the powers of b in base-b representation. This is because we wish to have $a_i = b^i$ as the terms of the sequence and have c_i represents the coefficient of $a_i = b^i$.

Finally we say that a representation is legal if the coefficients c_i obey certain given rules, and illegal otherwise. For example, $(1, 1, 2)$ is a legal representation for 22 written in base-3, but (1,4,1) is not.

2.2 PLRS

Positive Linear Recurrence Sequences, which we will abbreviate as PLRS, are defined in the following way:

Definition 3 (Miller and Wang (2012))**.** *A Positive Linear Recurrence Sequence* (*PLRS*) *is a sequence* $\{a_n\}_{n=1}^{\infty}$ *of positive integers with the following properties*:

- *Recurrence relation*: *There are non-negative integers* ℓ, $d_1, d_2, \dots, d_\ell$ *such that*

$$a_n = d_1 a_{n-1} + d_2 a_{n-2} + \cdots + d_\ell a_{n-\ell}$$

  with ℓ, d_1, d_ℓ *positive.*

- *Initial conditions*: $a_1 = 1$*, and for* $2 \le n \le \ell$*, we have*

$$a_n = d_1 a_{n-1} + d_2 a_{n-2} + \cdots + d_{n-1} a_1 + 1$$

  .

We have given examples of powers of b and the Fibonacci numbers as examples of PLRSs. We will demonstrate how to generate the terms of a PLRS with an unusual recurrence relation in the following concrete example.

Example 3. *Consider the recurrence relation* $a_n = a_{n-1} + a_{n-2} + 3a_{n-3}$*, we first note that* $\ell = 3, d_1 = 1, d_2 = 1,$ *and* $d_3 = 3$*. Thus, we generate the initial* 3 *terms using the initial conditions.*

We have that $a_1 = 1$*, and the remaining initial terms are*:

$$a_2 = d_1 a_1 + 1 = 2$$
$$a_3 = d_1 a_2 + d_2 a_1 + 1 = 2 + 1 + 1 = 4$$

Now, we are able to generate the remaining terms within the sequence using the full recurrence relation.

$$a_4 = d_1 a_3 + d_2 a_2 + d_3 a_1 = 4 + 2 + 3 \cdot 1 = 9$$
$$a_5 = d_1 a_4 + d_2 a_3 + d_3 a_2 = 9 + 4 + 3 \cdot 2 = 19$$
$$\ldots$$

Notice, however, that the PLRS with recurrence relation $a_n = a_{n-1} + a_{n-2} + 2a_{n-3} + a_{n-4} + a_{n-5} + 3a_{n-6}$ has exactly the same terms as the PLRS in the above example. Indeed, if the recurrence relation $a_n = a_{n-1} + a_{n-2} + 3a_{n-3}$ holds, then we may substitute a_{n-3} with $a_{n-4} + a_{n-5} + 3a_{n-6}$ and obtain the PLRS with 6 terms recurrence relation instead. On the other hand, if $n - 3$ is too small for the full recurrence relation, then applying the initial conditions gives $a_{n-3} = 1$ or $a_{n-3} = a_{n-4} + 1$ or $a_{n-3} = a_{n-4} + a_{n-5} + 1$, which in all cases satisfy the initial conditions of the PLRS with 6 terms recurrence relation.

Furthermore, the same argument holds when considering the PLRS with 9 terms recurrence relation where we apply the recurrence relation to a_{n-6}, and so on. Thus, while a recurrence relation does not uniquely determines the terms of a PLRS, we have some understanding of a family of recurrence relations that all produces the same terms when used to define PLRSs.

Now that we are able to generate terms of any desired PLRS, we turn to the notion of diagrams to utilize these terms to represent integers. Note that, as shown with the base-3 and balanced ternary examples, a single PLRS may result in multiple ways of writing an integer N as a sum $\sum c_i a_i$. Indeed, we can come up with a diagram for the normal base-3 representation and a different diagram for the balanced ternary representation, even though both representations make use of the same sequence of powers of 3. This is because different diagrams correspond to different maps $N \mapsto (c_1, c_2, \ldots, c_M)$.

2.3 Diagrams

In this subsection, we will introduce the diagrams of any given Positive Linear Recurrence Sequence.

Definition 4. *A diagram for a PLRS with recurrence sequence* $a_n = d_1 a_{n-1} + d_2 a_{n-2} + \cdots + d_\ell a_{n-\ell}$ *is a sequence of pairs of nonnegative integers* $[(p_i, n_i)]_{i=1}^{\infty}$ *with the following requirements*:

- $p_1 \geq 1$
- $p_i + n_i = d_i$ *for* $i \equiv k \pmod{\ell}$ *where* $1 \leq k < \ell$.
- $p_i + n_i = d_i - 1$ *for* $i \equiv 0 \pmod{\ell}$.

We illustrate a diagram in the case of a base-3 representation, which is given by the PLRS $a_n = 3a_{n-1}$.

2.3.1 Base-3 Representation

We note that the numbers used in base-3 representations, the powers of 3, form the Positive Linear Recurrence Sequence $a_n = 3a_{n-1}$. Using this recurrence relation, we generate the skeleton of the diagram:

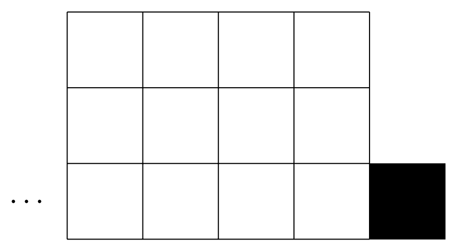


Figure 2.1 Diagram for the base-3 representation

Notice that we have an infinite number of columns to the left, and each column has 3 squares.

Definition 5. *The number of squares within a column is called the height of the column.*

We will index the columns from right to left with the shaded square being in column 0, the next column being column 1. The height of column 1 is equal to the coefficient of a_{n-1} within the recurrence relation, which is 3 in this particular case. As there are no more terms within the recurrence relation, we then repeat the columns $1, 2, \ldots, \ell$ for columns $\ell+1, \ell+2, \ldots, 2\ell$ and so on.

Definition 6. *A copy of recurrence block for the PLRS* $a_n = d_1 a_{n-1} + d_2 a_{n-2} + \cdots + d_\ell a_{n-\ell}$ *an arrangement of columns where column* i *has height* d_i *for* $1 \leq i \leq \ell$.

Thus, the skeleton of the diagram is simply composed of copies of recurrence block appended together with a shaded square on the rightmost column.

Definition 7. *Next, we fill in the skeleton using the diagram under the following rules*:

- *The column with the highest index within a copy of recurrence block must have exactly one square denoted* O*.*
- *Column* 1 *must have at least one square denoted* $+$*.*
- *Any other squares must be denoted with either* $+$ *or* $-$ *such that for column* i*, the number of square denoted* $+$ *is* p_i *and the number of square denoted* $-$ *is* n_i*.*

which will give us a representation map.

This would give us many different representation maps for a given skeleton of a PLRS depending on the diagram. We claim that the following valid representation map encodes the base-3 representation:

	O	O	O	O	
	+	+	+	+	
...	+	+	+	+	■

Indeed, the arrangement of + signs within a column denote the maximum number of terms of a particular value allowed, which in this case is a constant 2. That is, we may only use the numbers $\{0, 1, 2\}$ as the coefficients of our powers of three within base-3 representation.

Another way of filling in the entries might yield:

	O	O	O	O	
	+	+	+	+	
...	−	−	−	−	■

Indeed, one could see that this particular representation is none other than the balanced ternary representation, where for each digit (which here corresponds to a column,) we may put in the coefficient $\{+1, 0, \text{or } -1\}$. Knuth (1997)

In which case we are not allowed numbers such as $33 = 27 + 2 \cdot 3$ or $14 = 2 \cdot 1 + 3 + 9$, but must instead write them as $33 = 27 + 9 - 3$ and $14 = 27 - 9 - 3 - 1$ respectively instead.

2.3.2 Diagram Information

We consider another PLRS $a_n = a_{n-1} + a_{n-2} + 3a_{n-3}$, with the following first few terms $\{1, 2, 4, 9, 19, 40, 86, 183, 389, 830, 1768, \dots\}$, can have the following representation map out of many:

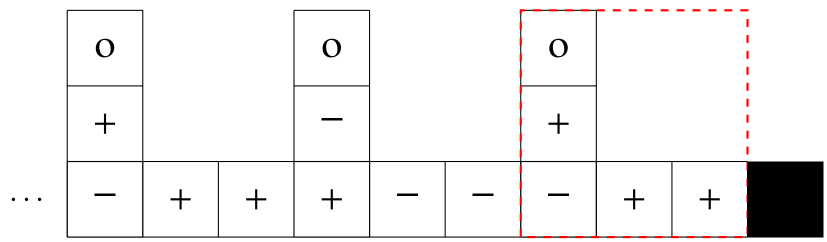


Figure 2.2 A possible diagram of an arbitrary PLRS

Note that a **copy of recurrence block** is highlighted with the dashed red lines, with the height of both column 1 and 2 being 1, while the height of column 3 is 3 since the coefficient of a_{n-3} is 3.

We also confirm that the requirements from Definition 7 have been fulfilled, with the O entries as well as column 1 having exactly one square with + entry. Also, all the remaining squares are filled in with either + or − sign.

The diagram encodes the largest number with a given **largest summand** in its representation, and the smallest number with a given largest summand in its representation. Here the largest summand within a representation of the number N refers to a_k within the representation of

$$N = \sum_{i=1}^{i=\infty} c_i a_i$$

where k is the largest number such that $c_k \neq 0$. For example, the largest summand of 35, 947 in a base-10 representation is 10, 000.

Definition 8. *For a given diagram $D = [(p_i, n_i)]_{i=1}^{\infty}$, the largest number with a given largest summand a_n, is denoted S_n^D or S_n, is the number with representation*:

$$S_n = \sum_{i=1}^{\infty} p_i a_{n-i+1}.$$

Example 4. *For the diagram in Figure 2.2, we have the sum $S_n = a_n + a_{n-1} + a_{n-2} + a_{n-5} + a_{n-6} + a_{n-7} + a_{n-8} + a_{n-11} + \dots$ as the largest number with given largest summand a_n. We may calculate S_6, for example, as the sum $a_6 + a_5 + a_4 + a_1 = 40 + 19 + 9 + 1 = 69$.*

Definition 9. *For a given diagram $D = [(p_i, n_i)]_{i=1}^{\infty}$, the smallest number with a given largest summand a_n, which we denote by D_n, is the number with representation*:

$$D_n = a_n - \sum_{i=1}^{\infty} n_i a_{n-i}$$

where n_i is the number of squares with $-$ entry within column i.

Example 5. *Using the same diagram in Figure 2.2, we have the sum $D_n = a_n - a_{n-3} - a_{n-4} - a_{n-5} - a_{n-6} - a_{n-9} - a_{n-10} - \dots$ as the smallest number with given largest summand a_n. Thus, we have that $D_7 = a_7 - a_4 - a_3 - a_2 - a_1 = 86 - 9 - 4 - 2 - 1 = 70$.*

Notice that D_7 is equal to $S_6 + 1$. Indeed, this relationship holds for any diagram generated according to Definition 7. This essentially gives us the carry-over rule of the representation, since adding 1 to the largest "n-digit" number yields the smallest "$n + 1$-digit" number.

Lemma 1. *For any PLRS and any diagram $D = [(p_i, n_i)]_{i=1}^{\infty}$, we have $D_{n+1} = S_n + 1$, and thus $\mathbb{Z}$ is partitioned by the sets of numbers whose representation have a given largest summand a_k.*

Proof. Note that $D_{n+1} - S_n$ is equal to

$$a_{n+1} - \sum_{i=1}^{\infty} n_i a_{n+1-i} - \sum_{i=1}^{\infty} p_i a_{n-i+1} = a_{n+1} - \sum_{i=1}^{\infty} (p_i + n_i) a_{n-i+1}$$

Since for each a_{n-i+1} we have a coefficient of $-(p_i + n_i)$, then by Definition 4 the expression can be understood as the value of the shaded cell minus the sum of all the remaining non-O entries. We claim that this difference is exactly 1, and will prove this using induction on n.

Let the PLRS $a_n = d_1 a_{n-1} + d_2 a_{n-2} + \cdots + d_\ell a_{n-\ell}$ have length ℓ. If $n+1 < \ell$, then

$$a_{n+1} = d_1 a_n + d_2 a_{n-1} + \cdots + d_n a_1 + 1$$

by the definition of PLRS. Thus, a_{n+1} is the sum of all squares within the columns of index less than ℓ plus 1 as required.

Suppose now that $n + 1 = \ell$, then

$$a_{n+1} = d_1 a_n + d_2 a_{n-1} + \cdots + d_\ell a_1$$

while the sum of squares with non-O entry is

$$d_1 a_n + d_2 a_{n-1} + \cdots + d_{\ell-1} a_2 + (d_\ell - 1) a_1$$

as the square with O entry is in the column for a_1. Thus the difference between the two is $a_1 = 1$ as required. This concludes the base cases.

In the inductive step, suppose that if $n \leq k$ then a_n is equal to the sum of all squares with non-O entry plus 1 for some $k > \ell$. Consider a_{k+1}, then by the recurrence relation

$$a_{k+1} = d_1 a_k + d_2 a_{k-1} + \cdots + d_\ell a_{k-\ell+1}$$

while the sum of all squares with non-O entry is

$$d_1 a_k + d_2 a_{k-1} + \cdots + (d_\ell - 1) a_{k-\ell+1} + d_1 a_{k-\ell} + d_2 a_{k-\ell-1} + \ldots$$

Thus, their difference is

$$a_{k-\ell+1} - d_1 a_{k-\ell} - d_2 a_{k-\ell-1} - \ldots$$

which is $a_{k-\ell+1}$ minus the sum of all non-O entries to the left. Since $k + 1 - \ell \leq k$, we apply the inductive hypothesis and we are done. □

This lemma explains the particular reason why we force the input O at the leftmost column of each copy of the recurrence block.

2.4 Allowable Blocks

We wish to generalize the notion of an allowable block from Hamlin and Webb into the context of signed representations. They utilized the following definition in studying representations with only nonnegative coefficients.

Definition 10 (Hamlin and Webb (2012))**.** *A set S of allowable blocks of a particular PLRS with recurrence relation $a_n = d_1a_{n-1} + d_2a_{n-2} + \cdots + d_\ell a_{n-\ell}$ is a set of sets $S = \{S_0, S_1, \ldots, S_{A-1}\}$ such that $A = d_1 + d_2 + \cdots + d_\ell$, $S_0 = 0$, and for $i > 0$ we have S_i consisting of all strings of integers of length k for all $1 \le k \le \ell$ such that $d_1d_2 \ldots d_{k-1} \le S_i < d_1d_2 \ldots d_k$ in lexicographic order.*

Hamlin and Webb use this construction instead of a greedy algorithm because of the initial terms within the PLRS. For example, if we have the PLRS $a_n = a_{n-1} + a_{n-2} + 3a_{n-3}$ with the following first few terms: $1, 2, 4, 9, 19, 40, \ldots$ then the greedy algorithm for representing 80 would give the result $80 = 2\cdot 40$. However, if we allow $80 = 2 \cdot 40$, then it seems that we should also allow $2 = 2 \cdot 1$ instead of writing 2 as being represented by 2 a number within the sequence. Similarly, we would have had $78 = 40 + 2 \cdot 19$ which might justify $4 = 2 + 2 \cdot 1$.

This is the same rationale as never using 2 in binary representation. Although the greedy algorithm would also never assign 2 as a digit within binary representation, in this particular more complicated example, we see that the greedy algorithm does assign a digit that is too high when dealing with later terms of the sequence, but not with the initial terms. Thus, we need to define allowable blocks of digits to make our representation less arbitrary.

Indeed, if we have a general PLRS with the recurrence relation $a_n = d_1a_{n-1} + d_2a_{n-2} + \cdots + d_\ell a_{n-\ell}$ then we cannot have a leading coefficient greater than d_1, as representing $N = a_2$ with $(d_1 + 1) \cdot a_1$ would be justified. Thus, we would expect the set of allowable blocks for PLRS with $(d_1, d_2, d_3) = (1, 1, 3)$ to be:

$$S = \{0, 1, 11, 111, 112\}$$

Example 6. *We consider how to decompose* 16 *with the PLRS $a_n = a_{n-1} + a_{n-2} + 3a_{n-3}$. Suppose we want to look at the following candidate*:

$$16 = 9 + 4 + 2 + 1$$

Now, it seems like 1111 *is not explicitly an allowable string, and in more complicated case we would need a way to check the legality of some long strings of digit values. Miller and Wang* (2012) *describes the process as starting from the leftmost entry* (*the largest entry*) *within the string as follows*:

When we encounter the first term, we compare its coefficient with the maximum first digit within all the allowable blocks.

- *If it is greater, the decomposition is illegal.*

- *If it is equal, we would check the next entry with the maximum second digit.*
- *If it is less, we "finish" the comparison and look at the next number in the string and compare it to the maximum first digit.*

For $16 = 9 + 4 + 2 + 1$*, we check* $1 \cdot 9$*, which is the maximum possible value for the first digit, thus we look at* $1 \cdot 4$*, which is also the maximum possible value for the second digit. We thus need to check the third digit* $1 \cdot 2$*, which is less than the maximum and thus we restart the process at* $1 \cdot 1$*, which is allowed. Thus, this decomposition is legal.*

Example 7. *As a non-example, consider the decomposition of* $17 = 9+4+2+1+1$ *which can be seen as a string* 1112*. We check this string from the leftmost entry with the maximum first digit allowed. Since it is equal, we check the second and get the same result. We check the third entry and see that it is less than the maximum third digit allowed and thus we finish the comparison and proceed to check the fourth entry. Since the fourth entry* $2 \cdot 1$ *exceeds the maximum first digit allowed, this decomposition is illegal.*

Instead, the legal decomposition of 17 *would be the sum* $9 + 4 + 2 + 2$*, which is indeed the string* 112 *within* $S = \{0, 1, 11, 111, 112\}$*.*

This definition of allowable blocks satisfies the algorithm for the generalized Zeckendorf decompositions Cordwell et al. (2018), which we would want to replicate in the case where our diagrams have all n_i equal to zero. We call such diagrams the Generalized Zeckendorf Decomposition diagrams.

2.4.1 Generalized Zeckendorf Decomposition's Algorithm

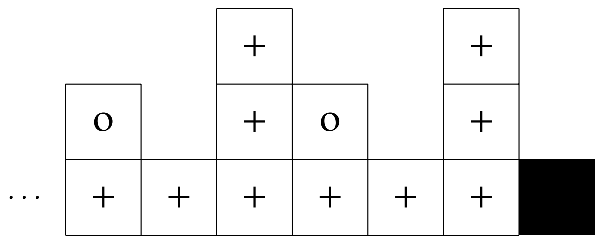
We will now explain the algorithm based on the concept of allowed blocks which is used in determining the Generalized Zeckendorf Decomposition for any non-negative integers N and PLRS $a_n = d_1 a_{n-1} + d_2 a_{n-2} + \cdots + d_\ell a_{n-\ell}$.

Recall that the largest summand for any diagram D can be calculated from the integer k such that $D_k \leq N \leq S_k$ where $D_k = S_{k-1} + 1$ from Lemma 1. Now, our diagram $D = [(p_i, 0)]$ is a Generalized Zeckendorf Decomposition diagram with $n_i = 0$ for all n_i.

Thus, we have that D_k is simply a_k and $S_k = D_{k+1} - 1 = a_{k+1} - 1$. That is, the largest summand is the largest term a_k such that $a_k \leq N \leq a_{k+1} - 1$. We greedily choose the largest term to begin our blocks!

Next, we consider the largest allowable block $S_i = s_1 s_2 \ldots s_n$ out of the set S of allowable blocks such that the associated sum $s_1 a_k + s_2 a_{k-1} + \cdots + s_n a_{k-n+1} \leq N$. We then record the linear combinations corresponding the chosen block S_i starting at the chosen summand a_k and repeat the above steps with $N - S_i$ until we reach 0.

Example 8. *We will now try to apply the above algorithm to a more complicated PLRS with recurrence relation* $a_n = 3a_{n-1} + a_{n-2} + 2a_{n-4}$ *which has the following first few terms*: $1, 4, 14, 47, 157, \ldots$ *and set of allowable blocks* $S = \{1, 2, 3, 31, 3101, 3102\}$. *We may visualize this as part of a diagram with only* + *or O entries*:



Suppose we want to decompose 217, *we greedily have the largest summand* 157. *Note that the only block we can insert is simply* 157, *as the next allowable block is* $2 \cdot 157 = 314$, *which is too large. Thus, we record* $1 \cdot 157$ *and continue with* $217 - 157 = 60$.

The largest summand for 60 *is* 47, *and again we choose the smallest block* $1 \cdot 47$. *We continue with* $60 - 47 = 13$.

Now, the largest summand is 4 *and the block we choose is* 31 *since* $3 \cdot 4 + 1 \cdot 1 = 13$ *which concludes the algorithm.*

Thus, 217 *decomposes into* $157 + 47 + 3 \cdot 4 + 1$.

We are now ready to proceed with our generalization of the concept of allowable blocks.

Chapter 3

Algorithm

This chapter will talk about how we can generate representations from diagrams. Concrete examples can be found in Appendix B.

3.1 Signed Allowable Block

We would want our generalizations to at least have the following properties:

1. The positive diagrams and the derived allowable blocks should correspond to the Generalized Zeckendorf Decomposition.

2. The representation of N and $-N$ should be negatives of each other.

3. The representations using the diagrams for other known representations such as the Far-Difference Representation and Balanced Ternary should also agree.

4. The representation is unique for a given diagram. This property is guaranteed due to the fact that we make use of an algorithm.

To do so, we generalize the concept of allowable blocks as follow:

For a given diagram $D = [(p_i, n_i)]_{i=1}^{\infty}$, and largest summand a_n, we consider the **signed allowable blocks** to refer to a set of sums of terms within the associated PLRS, all of which possess the same sign.

There will be two such sets, depending on whether we are inserting a block of the same or opposite signs to the block we have immediately inserted before.

Definition 11. *Rule* (1) *We are inserting a block of the same sign as the previous block. The signed allowable blocks are of the form*

$$\sum_{i=1}^{M} c_i a_{n-i+1}$$

where $c_k > 0$ *implies* $c_m = p_m$ *for all* $m < k$, *and* $0 \leq c_i \leq p_i$ *for all* i. *We order the signed allowable blocks by sum,* $a_n = B_1 < B_2 < \cdots < B_{max} = S_n$ *the value of the sum of all* $+$ *entries within the diagram. Similar to Definition* 10, *the difference between consecutive blocks* B_i *and* B_{i+1} *will be equal to a term within the recurrence sequence.*

Note that unlike in the Generalized Zeckendorf case in Section 2.4.1, we are not simply defining the signed allowable blocks within a single copy of the recurrence relation, but must continue the sum down to the smallest terms as there are no guarantees that the values of p_i will have a nice pattern.

The second case follows a similar approach but uses the numbers n_i instead of p_i. Since the negative signs do not always appear at the first column of the diagram, we need to skip all the s leading zeroes.

Definition 12. *Rule* (2) *We are inserting a block of different signs from the previous block. Let* s *be the smallest number such that* $n_s > 0$, *so that we can skip the leading zeroes in the sequence of* $\{n_i\}$. *We have that the signed allowable blocks are of the form*

$$\sum_{i=s}^{M} c_i a_{n-i+s}$$

Note that the index for our c_i *starts at* s. *We have that, similar to Definition* 11, $c_k > 0$ *implies* $c_m = n_m$ *for all* $m < k$ *and* $0 \leq d_i \leq n_i$ *for all* $i \geq s$. *We also order the signed allowable blocks by sum,* $a_n = B'_1 < B'_2 < \cdots < B'_{max} = a_{n+s} - D_{n+s}$ *the value of the sum of all* $-$ *entries within the diagram. Again, we have that the difference between consecutive blocks* B'_i *and* B'_{i+1} *will be equal to some term within the recurrence sequence.*

Example 9. *Given diagram*: $D = [(p_i, n_i)]_{i=1}^{\infty}$ *for the PLRS* $a_n = a_{n-1} + a_{n-2} + 2a_{n-3}$. *Let* $p_i = 1$ *for* $i \equiv 0, 1, 2, 3 \pmod 6$ *and* 0 *otherwise. Also, let* $n_i = 1$ *for* $i \equiv 0, 3, 4, 5 \pmod 6$ *and* 0 *otherwise. That is,* D *is a loop of* $[(1,0),(1,0),(1,1),(0,1),(0,1),(1,1),\ldots]$. *Note then that we have* $s = 3$ *for the second rule.*

Suppose we are given a largest summand a_7. The set of blocks corresponding to rule (1) *is*

$$S = \{a_7, a_7 + a_6, a_7 + a_6 + a_5, a_7 + a_6 + a_5 + a_2, a_7 + a_6 + a_5 + a_2 + a_1, \dots\}$$

and the set of blocks corresponding to rule (2) *is*

$$S' = \{a_7, a_7 + a_6, a_7 + a_6 + a_5, a_7 + a_6 + a_5 + a_4, a_7 + a_6 + a_5 + a_4 + a_1, \dots\}$$

Definition 13. *At each step of the algorithm, we determine a signed block with the following iterative process starting from $i = 1$.*

At the i^{th} iteration: if any of the following conditions are fulfilled

1. *i is the final maximum index, which means we are considering the largest signed allowable block available.*
2. *$N - B_{i+1}$ is of the opposite sign with the negative largest summand $a_{neg} > a_{\delta-s}$ where $a_\delta = B_{i+1} - B_i$*
3. *$N - B_i$ has a largest* (*negative or positive depending on the sign relations*) *summand $a_d < a_\delta = B_{i+1} - B_i$*

then B_i is our signed block. Otherwise, proceed to the $(i+1)^{th}$ iteration.

The algorithm consists of repeatedly applying process in Definition 13, starting with the integer $N_0 = N$ the number we wish to find the representation of. After computing and appending the signed block B_0 to the representation, we start the next iteration on the number $N_1 = N_0 - B_0$, comparing the sign of N_1 to B_0. This is done until we are left with 0, at which point we terminate the algorithm.

Example 10. *We will demonstrate how the algorithm is applied with the following PLRS $a_n = a_{n-1} + a_{n-2} + 3a_{n-3}$ with the corresponding diagram $D = [(1,0),(1,0),(1,1),(0,1),(0,1),(1,1),\dots]$ from Example 9. We will find the representation for $N = 677$, and we note that $s = 3$ the number of leading zeroes for the sequence of n_i.*

First, we determined that $S_9 = 389 + 183 + 86 + 9 + 4 + 2 + 1 = 674$. Thus, the largest summand for $N = 677$ is $a_{10} = 830$. We start the algorithm with the first signed allowable block which is 830 itself and see that $N - B_{1+1} = 677 - (830+389) = -542$ which is of the opposite sign (*that is, it is negative*)*. The largest negative summand for 542 is $389 = a_9$ our a_{neg}*

We note that $B_2 - B_1 = 830 + 389 - 830 = 389 = a_9$, which is our a_δ here. We have that $a_{\delta-s} = a_6 < a_{neg}$, thus we satisfied the second condition for halting *the block search. Therefore,* 830 *is the first block and we will now append the representation of* -153 *to it.*

As we are now inserting a block of the opposite sign to the previous block, we find the largest negative summand of 153*, which is* 86*. We start with the first block* -86*, and we have* $-153 - B_2' = -153 - (-86 - 40) = -27$ *which is still negative. Also,* $N - B_1' = -67$ *has largest negative summand* $40 = a_6$ *which is the same as* a_δ*. Since none of the* 3 *conditions are fulfilled, we proceed to the next signed allowable block.*

We now consider $-153 - B_3'$ *which is still negative, with* $-153 - B_2'$ *having the same largest summand as* a_δ*. Thus, none of the conditions are fulfilled and we proceed to the next block.*

We get that $-153 - B_4' = -153 - (-86 - 40 - 19 - 9) = 1$ *which is of the opposite sign. We have* $a_\delta = 9 = a_4$ *while* $a_{neg} = 1 = a_1$ *the largest negative summand of* 1*. Thus, we have* $a_{neg} = a_{\delta-s} = a_1$ *and our second condition is not fulfilled. Our third condition is also not fulfilled and we proceed to the next iteration.*

We have $-153 - B_5' = 2$ *which is of the opposite sign. Now, however, we have* $a_\delta = a_1$ *and thus* $a_{\delta-s} = 0$ *is definitely less than* $a_{neg} = 2$*. Our second condition has been fulfilled and thus we have our next block* $-86 - 40 - 19 - 9$*. We have a remainder of* 1*, which is the third and final block.*

Therefore, we get the far-difference representation $677 = 830 - 86 - 40 - 19 - 9 + 1$*.*

3.2 Properties

In this section, we will prove the following theorem:

Theorem 1. *The representation obtained using the Far-Difference Representation Algorithm is lexicographically increasing.*

Proof. We consider a PLRS $a_n = d_1 a_{n-1} + d_2 a_{n-2} + \cdots + d_\ell a_{n-\ell}$ with diagram D which has the first nonzero n_i at entry (p_s, n_s). Assume for the sake of contradiction that there exist two numbers $N_1 < N_2$ such that N_1 is lexicographically larger than N_2, where both numbers first differ at the coefficient of a_k. Thus, we consider their representation according to the algorithm as follows:

$$N_1 = \text{prefix} + m_1 a_k + \text{suffix}_1$$
$$N_2 = \text{prefix} + m_2 a_k + \text{suffix}_2$$

where the prefix is a potentially empty linear combination of a_i terms where $i > k$. That is, we consider k the largest term that differs in coefficient between N_1 and N_2. Since N_1 is lexicographically larger, we have $m_1 > m_2$.

The final step is to prove that $\text{suffix}_2 - \text{suffix}_1 < (m_1 - m_2)a_k$, since this implies $N_1 - N_2 = (m_1 - m_2)a_k - (\text{suffix}_2 - \text{suffix}_1) > (m_1 - m_2)a_k - (m_1 - m_2)a_k = 0$ a contradiction. We consider 2 cases:

Case 1: $m_1 \leq 0$. We immediately have that $m_2 < m_1$ must be negative.

- We first consider how the algorithm runs on N_1. Since choosing a signed block with a larger negative coefficient of a_k is possible, as demonstrated by the existence of m_2, the fact that m_1 was chosen means that the signed block of negative value must have the smallest summand at least a_k, and we now begin another signed block starting with largest summand at most a_{k-1}. Since we seek to minimize suffix_1, we have that it is greater than or equal to $-S_{k-1}$ the largest negative signed block with largest summand a_{k-1}.

- Now, we have that m_2 is negative, and thus suffix_2 is bounded above by the largest possible signed block of positive value. To get this upper bound, the algorithm must have terminated also at a_k which has a negative coefficient, and immediately start another signed block of the opposite (that is, positive) sign. This signed block, by the second condition of Definition 13, must have largest summand at most a_{k-s}. Since the signed block is chosen from the set of allowable signed blocks according to Definition 12, it must have size at most $a_{k-s+s} - D_{k-s+s} = a_k - D_k$. We thus get that in this case, $\text{suffix}_2 - \text{suffix}_1 \leq a_k - D_k + S_{k-1} = a_k - 1$ which is strictly less than $1 \cdot a_k \leq (m_1 - m_2)a_k$ as we desired.

Case 2: $m_1 > 0$.

- We have that suffix_1 is bounded below by the largest possible signed block of negative value with largest summand a_{k-s}. Using the same arguments as case $1.b$, we get that $\text{suffix}_1 \geq -(a_k - D_k) = D_k - a_k$.

- We then work with bounding suffix_2 above on two cases:
  - Consider $m_2 < 0$; we have that $m_1 - m_2 \geq 2$. Also, we have that suffix_2 is bounded above by the largest possible signed block of positive value with largest summand a_{k-s} with the familiar

argument. We get that $\text{suffix}_2 \leq a_k - D_k$ and $\text{suffix}_2 - \text{suffix}_1 \leq 2a_k - 2D_k < 2a_k \leq (m_1 - m_2)a_k$ as required.

– Now, if m_2 is nonnegative then by the argument in case $1.a$ we bound suffix_2 above by S_{k-1} and we obtain $\text{suffix}_2 - \text{suffix}_1 \leq a_k + S_{k-1} - D_k = a_k - 1$ which is less than $(m_1 - m_2)a_k$ as we desired.

□

This theorem thus demonstrates that our algorithm provides a resulting representation with some patterns and that it is not entirely arbitrary.

Chapter 4

Summand Minimality

4.1 Skipponacci Sequences

The Skipponacci sequences are Positive Linear Recurrent Sequences of the form:

$$a_{n+1} = a_n + a_{n-k}$$

where $k \in \mathbb{N}$. Note that when $k = 1$, we get the Fibonacci sequence. We follow Demontigny et al. in denoting these as the k-Skipponaccis.

They have also discovered the following far-difference decomposition rules for the Skipponaccis:

Theorem 2 (Demontigny et al. (2014))**.** *Every $x \in \mathbb{Z}$ has a unique far-difference representation for the k-Skipponaccis such that every two terms of the same sign are at least $2k + 2$ apart in index and every two terms of opposite sign are at least $k + 2$ apart in index.*

We will focus on cases where $k \geq 2$ and prove the following results:

Lemma 2. *When $k \geq 2$, the far-difference representation of k-Skipponaccis is not summand minimal.*

Proof. We claim that $N = a_n + a_{n-2k}$ cannot be expressed with fewer than 3 terms using the above decomposition rules. Note that N is larger than the largest number whose far-difference representation has leading summand a_n, which is $S_n = a_n + a_{n-2k-2} + a_{n-4k-4} + \dots$. Thus, the largest summand of N is a_{n+1} since $a_n < N = a_n + a_{n-2k} < a_n + a_{n-k} = a_{n+1}$.

Thus, we have that $N = a_{n+1} - D$ for some positive integer $D = a_{n+1} - a_n - a_{n-2k} = a_{n-k} - a_{n-2k}$. We claim that, for sufficiently large n, D is not a term within the k-Skipponaccis.

Note that

$$D = a_{n-k} - a_{n-2k} = (a_{n-k-1} + a_{n-2k-1}) - a_{n-2k} < a_{n-k-1}$$

when $a_{n-2k-1} < a_{n-2k}$, which is always true for large enough n.

Thus, we only have to prove that $D > a_{n-k-2}$ to complete our proof. Notice that

$$\begin{aligned} D - a_{n-k-2} &= a_{n-k} - a_{n-k-2} - a_{n-2k} \\ &= (a_{n-k-1} + a_{n-2k-1}) - a_{n-k-2} - a_{n-2k} \\ &= (a_{n-k-2} + a_{n-2k-2} + a_{n-2k-1}) - a_{n-k-2} - a_{n-2k} \\ &= a_{n-2k-1} + a_{n-2k-2} - a_{n-2k} \\ &= a_{n-2k-2} - a_{n-3k-1} \\ &> 0 \end{aligned}$$

since $k > 1$. Thus, we require at least 2 terms to represent D and the far-difference representation of $a_n + a_{n-2k}$ requires at least 3 terms. Therefore, the far-difference representation of k-Skipponaccis is not summand minimal for $k > 1$.

□

4.2 Base-b Representations

Equipped with the algorithm, we present the following results and conjectures on the number of summands in various far-difference representations.

Lemma 3. *The balanced ternary representation corresponding to the diagram* $D = [(1,1),(1,1),(1,1),\dots]$*, is summand minimal.*

Proof. We will prove this result by presenting a process to transform any arbitrary representation of an arbitrary integer $N = \sum_{i=0}^{\infty} c_i 3^i$ where each coefficient $c_i \in \mathbb{Z}$ can be positive or negative, into the balanced ternary representation of N.

We represent the coefficients of the sum with the infinite sequence $(c_0, c_1, c_2, \dots)$ with a finite number of non-zero terms. We will make use of the following transformations which do not increase the number of summands, defined as $\sum_{i=0}^{n} |c_i|$, while preserving the sum:

- We change $c_k 3^k$ into $(c_k - 3)3^k + 3^{k+1}$ when $c_k \geq 2$.

- We change $c_k 3^k$ into $(c_k+3)3^k - 3^{k+1}$ when $c_k \le -2$.

Note that both transformations preserve the number of summands when $|c_k| = 2$ and reduce the number of summand by at least 1 otherwise. Also, if we are unable to perform both moves, then we have that $|c_k| \le 1$ which implies that the resulting sum is the unique balanced ternary representation of N.

Assume for the sake of contradiction that this process does not terminate, we get that after a finite number of process the number of summands must stay constant. Thus, we are only applying the transformations to case where $|c_k| = 2$. Note that if the algorithm does not terminate, our largest summand must be unbounded since the absolute sum increases by $2 \cdot 3^k$ every time we perform a transformation on the k^{th} digit.

Now, without loss of generality assume that the largest summand has a positive coefficient. Since we are now restricted to coefficients of values $-2, -1, 0, 1, 2$, if we have m summands and our largest summand is 3^M, then we get

$$\begin{aligned} N = \sum_{i=0}^{\infty} c_i' 3^i &\ge 3^M - 2 \cdot (3^{M-1} + 3^{M-2} + \cdots + 3^{M-\lfloor \frac{m-1}{2} \rfloor}) - r \cdot 3^{M-1-\lfloor \frac{m-1}{2} \rfloor} \\ &> 3^{M-1-\lfloor \frac{m-1}{2} \rfloor} \end{aligned}$$

where r is the remainder of $m-1$ divided by 2. This results from greedily taking the largest summands of digit less than M as negative coefficients. Now, we have that m is constant while M is unbounded, which means that N is unbounded, an impossibility. Therefore, the process must terminate and the proof is completed.

□

Example 11. *We consider the following representation of* 173 *as*

⊖	2⊖	5⊕	⊖	2⊕

with the leftmost square representing the coefficient of the smallest digit 1 *and the rightmost square representing the coefficient of the largest digit, here being* $3^4 = 81$*. We apply the transformation on* $5 \cdot 3^2$ *and get* $2 \cdot 3^2 + 3^3$ *which cancels out*

⊖	2⊖	2⊕		2⊕

We continue by applying the transformations starting from the left to get:

⊖	⊕	⊕		2⊕

and then finally

⊖	⊕	⊕		⊖	⊕

Theorem 3. *All balanced base-$(2b+1)$ representation corresponding to the diagram $[(b,b),(b,b),(b,b),\dots]$ are summand-minimal.*

Proof. We generalizes the idea in the proof of Lemma 3. Our transformations which do not increase the number of summands are:

- We change $c_k(2b+1)^k$ into $(c_k - 2b - 1)(2b+1)^k + (2b+1)^{k+1}$ when $c_k \geq b+1$.
- We change $c_k b^k$ into $(c_k + 2b + 1)(2b+1)^k - (2b+1)^{k+1}$ when $c_k \leq -b-1$.

Again, we assume that the process of applying these two transformations does not terminate, and therefore after a finite number of moves the number of summand no longer decreases. We are then only applying transformations to the case where $|c_k| = b+1$, which increases the absolute sum by $2b(2b+1)^k$ per transformation. Thus, the largest summand is unbounded.

Assume without loss of generality that the largest summand has a positive coefficient. Since we are now restricted to coefficients of values $\{-b-1, -b, \dots, b, b+1\}$, suppose we have m summands and our largest summand is $(2b+1)^M$, we get

$$\begin{aligned} N = &\sum_{i=0}^{\infty} c_i' 3^i \\ \geq &(2b+1)^M - (b+1)((2b+1)^{M-1} + (2b+1)^{M-2} + \dots \\ &+ (2b+1)^{M-\lfloor \frac{m-1}{b+1} \rfloor}) - r(2b+1)^{M-1-\lfloor \frac{m-1}{b+1} \rfloor} \\ > &(2b+1)^{M-1} \end{aligned}$$

where r is the remainder when $m-1$ is divided by $b+1$. Since M is unbounded, we have that N is unbounded, an impossibility. Thus, the process must terminate at the desired balanced base-$(2b+1)$ representation of N, which must then be summand minimal. □

Note, however, that there is not necessarily a unique summand minimal diagram. Different diagrams may all have the same number of summands as shown in the following result for an alternative summand minimal base-3 representation.

We will consider the diagram $D = [(2,0),(0,2),(2,0),(0,2),\dots]$ which we will prove is summand minimal just like the balanced ternary representation. First, however, we will describe this particular representation, which we call the far ternary representation.

Lemma 4. *The diagram* $D = [(2,0),(0,2),(2,0),(0,2),\dots]$ *results in a representation which satisfies the following conditions*

1. *All coefficients are elements of* $\{-2,-1,0,1,2\}$.
2. *No terms of the opposite signs may be consecutive.*
3. *If the coefficient* $c_k = \pm 2$ *for some* $k \geq 1$, *then* $c_{k-1} = 0$.

Proof. The first condition is satisfied by the fact that the allowable blocks are $0,1,2$ and thus no coefficients of size 3 or more are ever added in the process.

The second condition in the lemma follows from the second condition of Definition 13 in determining a signed block. That is, since the first column in D is $(2,0)$, with the second entry being 0, terms of the opposite signs have to be at least 2 index apart.

The third condition follows from the fact that, without loss of generality, if $c_k = 2$ then $c_{k-1} \leq 0$ if both c_k and c_{k-1} belongs to the same block, since our allowable blocks are $\{1,2,201,202,20201,\dots\}$. The same argument holds for the negative case.

Suppose otherwise that $c_{k-1} \geq 1$ is the beginning of a new block, which we call B'. Since the previous block ends with $c_k = 2$, it is of the form $B = (\dots,2,0,2,0,2,\dots,2) = 2\cdot 3^k + 2\cdot 3^{k+2} + 2\cdot 3^{k+4} + \cdots + 2\cdot 3^{k+2M}$. Also, we note that $c_{k-2} \geq 0$ since it cannot be negative by the second condition. At the step determining the largest summand of B' to be c_{k-1}, we must have $c_{k-1}3^{k-1} + c_{k-2}3^{k-2} + \cdots > 2\cdot 3^{k-2} + 2\cdot 3^{k-4} + \dots$. This then implies

$$\begin{aligned} B + c_{k-1}3^{k-1} + c_{k-2}3^{k-2} + \dots &> 2\cdot 3^{k+2M} + 2\cdot 3^{k+2M-2} + \dots \\ &= S_{k+2M}. \end{aligned}$$

However, the process of determining the largest summand of B to be 3^{k+2M} requires that $B + c_{k-1}3^{k-1} + c_{k-2}3^{k-2} + c_{k-3}3^{k-3} + c_{k-4}3^{k-4} + \cdots \leq S_{k+2M}$, which gives us a contradiction. Thus, c_{k-1} must be at most 0. However, due to the second condition, it cannot be negative and thus c_{k-1} is exactly 0 as desired. □

Now, we will prove that this representation is unique.

Lemma 5. *The far ternary representation, which is described by the diagram* $D = [(2,0),(0,2),(2,0),(0,2),\ldots]$ *is unique.*

Proof. Suppose that there exists an integer N such that $N = (c_0, c_1, c_2, \ldots) = (c'_0, c'_1, c'_2, \ldots)$ where both $\{c\}$ and $\{c'\}$ satisfy the three conditions. Without loss of generality, assume $N > 0$ and assume for the sake of contradiction that there exists k, the largest integer such that $c_k > c'_k$.

Now, we have that at most one of c_k or c'_k can be 0 and either

- $|c_{k-1}3^{k-1} + c_{k-2}3^{k-2} + \ldots| \leq 2 \cdot 3^{k-2} + 2 \cdot 3^{k-4} + \ldots$ when $c_k \neq 0$ or
- $|c'_{k-1}3^{k-1} + c'_{k-2}3^{k-2} + \ldots| \leq 2 \cdot 3^{k-2} + 2 \cdot 3^{k-4} + \ldots$ when $c'_k \neq 0$.

while

- $|c_{k-1}3^{k-1} + c_{k-2}3^{k-2} + \ldots| \leq 2 \cdot 3^{k-1} + 2 \cdot 3^{k-3} + \ldots$ when $c_k = 0$ or
- $|c'_{k-1}3^{k-1} + c'_{k-2}3^{k-2} + \ldots| \leq 2 \cdot 3^{k-1} + 2 \cdot 3^{k-3} + \ldots$ when $c'_k = 0$.

Taking the difference between these two representations yields

$$\begin{aligned} 0 &= (c_k - c'_k)3^k + (c_{k-1} - c'_{k-1})3^{k-1} + (c_{k-2} - c'_{k-2})3^{k-2} + \ldots \\ &\geq (c_k - c'_k)3^k - 2 \cdot 3^{k-1} - 2 \cdot 3^{k-2} - \ldots \\ &\geq 3^k - 2 \cdot 3^{k-1} - 2 \cdot 3^{k-2} - \ldots \\ &= 1, \end{aligned}$$

which is a contradiction. Thus, no such k exist and N has a unique representation which satisfies the three conditions. □

Now, since we are able to describe the resulting representation, we can then prove the following property:

Theorem 4. *The far ternary representation is summand minimal.*

Proof. We will prove this result by showing that one can always transform any representation into the representation corresponding to the diagram D without increasing the number of summands.

Since we can always reduce the number of summands whenever we have a coefficient not in the set $\{-2,-1,0,1,2\}$, we will focus on the forbidden case where summands of opposing signs are next to each other, or when the coefficient of a summand has magnitude 2, while the smaller summand has a nonzero coefficient. Whenever such a pair of the first kind is found, starting from the left, apply the following transformations depending on their signs:

- We perform $c_k 3^k + c_{k+1}3^{k+1} = (c_k + 3)3^k + (c_{k+1} - 1)3^{k+1}$ when $c_k < 0$ and $c_{k+1} > 0$. Note that the resulting coefficients will always have digit values $\{-2, -1, 0, 1, 2\}$ if all coefficients were already in the same range.

$c_k\ominus$	$c_{k+1}\oplus$	$\rightarrow$	$(3 + c_k)\oplus$	$(c_{k+1} - 1)\oplus$

- We perform $c_k 3^k + c_{k+1}3^{k+1} = (c_k - 3)3^k + (c_{k+1} + 1)3^{k+1}$ when $c_k > 0$ and $c_{k+1} < 0$. Note that the resulting coefficients will always have digit values $\{-2, -1, 0, 1, 2\}$ if all coefficients were already in the same range.

$c_k\oplus$	$c_{k+1}\ominus$	$\rightarrow$	$(3 - c_k)\ominus$	$(c_{k+1} + 1)\ominus$

Again, we claim that such a process must terminate. Consider labeling **runs** within the original balanced ternary as follows:

- Start from the leftmost (smallest) digit, and add the first digit with a nonzero coefficient as the beginning of the run.
- Keep appending the next digit to the run until you come across a digit with zero coefficient or with a different sign.
- If you stop at a zero coefficient, this particular **run** consists of consecutive digits with all coefficients +1 or all −1. Otherwise, add the final digit with the opposite sign as the end of the run.
- Continue the labeling on the next digit with nonzero coefficient.

Without loss of generality (we simply invert the sign in the other case), we consider the leftmost run of the form

$$(\ldots, 1, 1, 1, \ldots, 1, -1, \ldots).$$

Since this is the smallest illegal run, our algorithm would start applying the second transformation at the right end of this run to yield

$$(\ldots, 1, 1, 1, \ldots, -2, 0, \ldots).$$

We repeatedly apply the second transformation within this run to get the final arrangement

$$(\ldots, -2, -1, -1\ldots, -1, 0, \ldots).$$

We continue with the next smallest run of this form until they are all transformed into the final legal arrangement. Note that in doing so, all digits with zero as coefficient remain unchanged.

Now consider the digit immediately to the left of any of both kinds of runs. It was either initially zero, or, since it is nonzero and not part of the run we are immediately interested in, it was the rightmost digit of another run of the initially illegal kind. If the digit was zero, then after the above process it remains zero and the run we are interested in is legal, since it is preceded and succeeded by zeroes. Otherwise, the digit becomes zero anyway and we achieve the same result.

Thus, all the runs become legal, since they all have a preceding and succeeding digits of 0 and the middle digits have coefficients of the same sign with possibly a single entry of ±2 on the leftmost digit. Thus, our algorithm terminates and we are done. □

Example 12. *We will demonstrate the result by converting the balanced ternary representation of* $173 = (-1, 1, 1, 0, -1, 1)$, *which we have derived in Example 11, into its far ternary representation. Its balanced ternary representation is as follows*:

⊖	⊕	⊕		⊖	⊕

We have that the leftmost two boxes constitute a run, as well as the third box on its own and the fifth and sixth boxes. Thus our algorithm runs as follow:

⊖	⊕	⊕		⊖	⊕
2⊕		⊕		⊖	⊕
2⊕		⊕		2⊕	

Thus the far ternary representation of $173 = 2 \cdots 3^0 + 3^2 + 2 \cdots 3^4$ *is summand minimal as desired.*

Remark 1. *A natural question to ask would be whether the diagram*

$$D = [(b, 0), (0, b), (b, 0), (0, b), \ldots]$$

for base-$(b+1)$ *is summand minimal when* $b \geq 3$, *thus filling in the gap of Theorem 3 which only provides the summand minimal representation for base-odd numbers.*

However, we do not have summand minimality for $b > 3$, *since we can simply consider the number b itself which will be written as* $b \cdot 1$ *using* $b \geq 3$ *summands instead of the summand minimal* $-1 \cdot 1 + 1 \cdot (b+1)$ *which uses only* 2 *summands.*

4.3 Tribonacci

We will be proving in this section that the representation corresponding to the diagram

$$D = [(1,0),(0,1),(0,0),(1,0),(0,1),(0,0),\dots]$$

corresponds to a Tribonacci far-difference representation which is also summand minimal.

First, we will describe the representation in terms of rules for limiting the coefficients that may be used. We obtain a similar description to the case of Fibonacci far-difference representation.

Lemma 6. *The following rules uniquely determine the representation of any integer, and result from applying the algorithm to the diagram*

$$D = [(1,0),(0,1),(0,0),(1,0),(0,1),(0,0),\dots].$$

- *The coefficients are from the set* $\{-1,0,1\}$*.*
- *Two summands of the same sign must be at least* 3 *apart in index.*
- *Two summands of the opposite signs must be at least* 2 *apart in index.*

We first prove that applying the algorithm does indeed result in these rules.

Proposition 1. *Applying the algorithm to the diagram*

$$D = [(1,0),(0,1),(0,0),(1,0),(0,1),(0,0),\dots]$$

satisfies the rules in Lemma 6*.*

Proof. Note that the first rule is satisfied by the fact that for any entry within the diagram, we have $p_i \leq 1$ and $n_i \leq 1$. Thus, we cannot have 2 within the string of allowable blocks.

Indeed, the signed allowable blocks are the same for both inserting blocks of the same sign and inserting blocks of a different sign, which are $\{1, 1001, 1001001, \dots\}$ which means that in each insertion we cannot have terms of the same sign being less than 3 indices apart.

Also, suppose that at a step in the algorithm, we have inserted the next block of the same sign as the previous block at index less than 3 apart from the previous block. We get that the previous block cannot have been the chosen signed block. It cannot be chosen based on the first condition of Definition 13 since otherwise we get that the previous block has a largest summand that is too small. Also, we have that it would not have yielded an opposite sign if subtracted with the next larger signed allowable block so it could not fulfill the second condition. Finally, a_δ which is less than 3 indices away is larger than the summand that would have been added which is 3 indices away.

The final condition is fulfilled by the fact that $n_1 = 0$ and thus the second condition of Definition 13 would prevent blocks of the opposite sign to be placed next to the previous block. □

Now, we will prove the uniqueness that results from the 3 rules.

Proposition 2. *The rules in Lemma* 6 *uniquely determine the representation of any integer.*

Proof. Suppose that there exists an integer N such that $(c_1 c_2 \dots c_n)$ and $(c'_1 c'_2 \dots c'_m)$ are representations of N which both satisfy the conditions of Lemma 6. We consider the largest integer k such that $c_k \neq c'_k$. Without loss of generality, assume $c_k > c'_k$. Thus, since our coefficients are either $-1, 0$ or 1, we have that $c_k = 0$ or $c_k = 1$. In the following cases, we try to minimize $\sum_{i=1}^{k-1} c_i T_i$ and maximize $\sum_{j=1}^{k-1} c'_j T_j$ according to the value of c_k.

Case 1: Suppose that $c_k = 0$, we have $c'_k < 0$ and thus c'_{k-1} must be 0. We get the bound using the rules as follow:

$$\begin{aligned}
N - N &= (c_k - c'_k)T_k + \left(\sum_{i=1}^{k-1} c_i T_i\right) - \left(\sum_{j=1}^{k-1} c'_j T_j\right) \\
&\geq T_k + (-T_{k-1} - T_{k-4} - T_{k-7} - \dots) - (T_{k-2} + T_{k-5} + T_{k-8} + \dots) \\
&= T_k - (T_{k-1} + T_{k-2} + T_{k-4} + T_{k-5} + \dots) \\
&= 1
\end{aligned}$$

which is a contradiction.

Case 2: Suppose that $c_k = 1$, we get that $c_{k-1} = 0$ and our bounds are as follow:

$$\begin{aligned}
N - N &= (c_k - c_k')T_k + \left(\sum_{i=1}^{k-1} c_i T_i\right) - \left(\sum_{j=1}^{k-1} c_j' T_j\right) \\
&\geq T_k + (-T_{k-2} - T_{k-5} - T_{k-8} - \dots) - (T_{k-1} + T_{k-4} + T_{k-7} + \dots) \\
&= T_k - (T_{k-1} + T_{k-2} + T_{k-4} + T_{k-5} + \dots) \\
&= 1
\end{aligned}$$

which is also a contradiction. Therefore, no such N exists and our representation is uniquely determined for any integer. □

In order to prove that the far-ternary representation is summand minimal, we will need the following result:

Lemma 7. *Let T_n be the n^{th} Tribonacci number with $T_1 = 1$, $T_2 = 2$, $T_3 = 4$, and $T_n = T_{n-1} + T_{n-2} + T_{n-3}$ for $n \geq 4$. We have that*

$$T_n + T_{n-3} + T_{n-6} + \cdots = 2T_{n-1} + T_{n-4} + T_{n-7} + \dots$$

for all $n \geq 2$.

Proof. We use mathematical induction to prove the above result. The base case for $n = 2, 3, 4$ can be checked directly with simple computations. Suppose that for some $k \geq 4$ we have that the equation holds, we consider

$$\begin{aligned}
T_{k+1} + T_{k-2} + T_{k-5} + \dots &= (T_k + T_{k-1} + T_{k-2}) + T_{k-2} + T_{k-5} + \dots \\
&= T_k + T_{k-1} + (2T_{k-2} + T_{k-5} + \dots) \\
&= T_k + T_{k-1} + T_{k-1} + T_{k-4} + \dots \\
&= T_k + (2T_{k-1} + T_{k-4} + \dots) \\
&= T_k + T_k + T_{k-3} + T_{k-6} + \dots \\
&= 2T_k + T_{k-3} + T_{k-6} + \dots
\end{aligned}$$

which is what we wish to prove. □

We are now ready to prove the theorem.

Theorem 5. *The Tribonacci far-difference representation, which corresponds to the diagram*

$$D = [(1,0), (0,1), (0,0), (1,0), (0,1), (0,0), \dots],$$

is summand minimal.

Proof. We will prove summand minimality by applying the four following transformations to an arbitrary Tribonacci representation, each having positive and negative versions:

- The first transformation performs $2T_n \mapsto T_{n+1} + T_{n-3}$ at index n.
- The second transformation performs $T_n + T_{n-1} \mapsto T_{n+1} - T_{n-2}$ at index n.
- The third transformation performs $T_n + T_{n-2} \mapsto T_{n+1} - T_{n-1}$ at index n.
- The fourth transformation performs $T_n - T_{n-1} \mapsto T_{n-1} - T_{n-4}$ at index n.

Note that if we cannot apply any of the transformations, then we have arrived at the Tribonacci far-difference representation without increasing the number of summands, since all of the four transformations weakly decrease the number of summands.

Thus, we will prove that it is always possible to transform any arbitrary representation of any number into its Tribonaci far-difference representation using a finite amount of these transformations. To do so, we conceive of an algorithm that we then prove must terminate. We describe the algorithm as follows:

1. If at any point the number of summands can be decreased, do so and restart the algorithm at this step.
2. Repeatedly perform only the first, second, and third transformations in this order of priority until none can be performed. (We will prove that this step must terminate after a finite number of transformations.)
3. Perform the fourth transformation at the smallest possible index, which we numbered $i_1 = i$.
4. After that, if we can perform the fourth transformation at index $i - 3$, do so and update the value of i to $i_2 = i - 3$. Repeat this step until we perform the final fourth transformation at i_f the final index.
5. Run this algorithm but this time only on indices less than $i_f - 1$. After which, return to step 3.

To prove that it terminates, we use induction on the number of summands. For the base case of 1 summand, the algorithm immediately terminates at the Tribonacci far-difference representation as we desired. Now, suppose that for some $k \in \mathbb{N}$ we have that the algorithm terminates whenever we perform it on at most k summands. We now work on an arbitrary sum of $k+1$ Tribonacci numbers, and any reduction in the number of summands would immediately lead to the algorithm eventually terminating by the inductive hypothesis.

We first characterize some arrangements which lead to a possible reduction in the number of summands in the first step. These are as follows:

1.

⊕	⊕	⊕		→				⊕

2.

		⊕	⊕		⊕	⊕	
→	⊖			⊕	⊕	⊕	
→	⊖						⊕

We will now prove that the second step of the algorithm must terminate. Note that the first, second, and third transformations increase the lexicographical order of the representation, thus if we perform infinitely many of these 3 transformations then the largest summand must be arbitrarily large.

Since we give the first transformation priority, we first prove the claim that we can only perform finitely many first transformations before we are forced to perform other transformations. Notice that the first transformation, when performed at index $n \geq 3$, reduces the following sum

$$\sum_{j=1}^{\infty} j|c_j|$$

which is the sum of the index of each summand, by at least 1 when $n = 3$ and 2 otherwise. Furthermore, if the first transformation is performed at indices 1 or 2, we have that the number of summands decreases, as in $1+1 = 2+0 = 2$ and $2+2 = 4+0 = 4$.

Thus, after performing all the possible first transformations, our coefficients are either $-1, 0$, or 1. Suppose we start with at most m summands representing the integer N. We note that we cannot have $\pm(T_n + T_{n-1} + T_{n-2})$ or $\pm(T_n + T_{n-1} + T_{n-3} + T_{n-4})$ since we can reduce the number of summands otherwise. We then get the following inequality with a given largest positive summand T_M:

$$\begin{aligned} N &\geq T_M - T_{M-1} - T_{M-2} - T_{M-4} - T_{M-6} - T_{M-7} - \dots \\ &= T_{M-3} - T_{M-4} - T_{M-6} - T_{M-7} - T_{M-9} - T_{M-11} \dots \\ &= T_{M-5} - T_{M-7} - T_{M-9} - T_{M-11} - T_{M-12} - \dots \\ &\geq T_{M-6} \end{aligned}$$

However, since N is fixed, we have that M, and also the lexicographic size of the representation, is bounded above. Thus, our second step of the algorithm must terminate. We are left with a sum where terms of the same signs must be at least 3 indices apart, but terms of different signs may be next to each other.

At the third and fourth step, we start by applying the fourth transformation at the smallest possible index i. Now, without loss of generality since we can invert the signs otherwise, assume that $c_i = 1$ and $c_{i-1} = -1$. We have that $c_{i-2} = 0$ as we would be able to perform the third transformation at i if $c_{i-2} = 1$ and second transformation at $i-1$ if $c_{i-2} = -1$. We have two possibilities for c_{i-3}.

- If $c_{i-3} = 1$, then $c_{i-4} = 0$ since it cannot be 1 and if it is -1 then the smallest possible index that we can apply the fourth transformation would be $i-3$. Also, c_{i-5} cannot be 1 and if $c_{i-5} = -1$, then we can reduce the number of summands after performing the fourth transformation on i by performing the second transformation on $i-4$. By the same argument, $c_{i-6} \neq -1$. We thus have the following:

$i-6$	$i-5$	$i-4$	$i-3$	$i-2$	$i-1$	i
0 or $\oplus$			$\oplus$		$\ominus$	$\oplus$

  After performing the fourth transformation on i, we have $c_i = 0$, $c_{i-1} = 1$, $c_{i-3} = 1$, $c_{i-4} = -1$ and we are at step 4. Whether we return to step 4 again depends on whether c_{i-6} is 1 or 0 using the exact same reasoning but at 3 indices lower.

$i-6$	$i-5$	$i-4$	$i-3$	$i-2$	$i-1$	i
0 or $\oplus$		$\ominus$	$\oplus$		$\oplus$	

- We eventually have that $c_{i-3} = 0$ after repeating step 4 at lower and lower indices and updating the values of i.

$i-4$	$i-3$	$i-2$	$i-1$	i
0 or$\ominus$			$\ominus$	$\oplus$

Note that if $c_{i-4} = 1$ then we get a cancellation after we perform the fourth transformation for the final time in step 4. After finishing step 4 without cancellations, we have that c_{i-4} can be -2 or -1 while $c_{i-3} = 0, c_{i-2} = 0, c_{i-1} = 1$.

$i-4$	$i-3$	$i-2$	$i-1$
$\ominus$ or 2$\ominus$			$\oplus$

We then proceed to step 5. Note that the entries on indices i_f to i_1 is composed of $1 = c_{i_1-1} = c_{i_2-1} = \cdots = c_{i_f-1}$ and all other entries being 0, which satisfies the Tribonacci representation rules.

Now, since we are running the algorithm only on indices $i-2$ or lower, while we have isolated at least one summand with $c_{i-2} = 1$, we have that step 5 must terminate by the inductive hypothesis. Note that we are running the algorithm on a sum S with a negative coefficient for the largest summand T_{i-4}.

If in the process of performing step 5, we have that the largest summand increased to T_{i-1} after performing some transformations, then it canceled with the positive T_{i-1} which remains after step 4. This is because the largest summand cannot be of a different sign if there have been no cancellations, and thus it remains negative throughout step 5.

Now, we claim that after step 5 terminates without cancellations, the largest summand is at most T_{i-3}. This is because

$$\begin{aligned} S &\geq -2T_{i-4} - T_{i-7} - T_{i-10} - \ldots \\ &= -T_{i-3} - T_{i-6} - T_{i-9} - \ldots \end{aligned}$$

which is proved in Lemma 7, and

$$\begin{aligned} S &\leq -T_{i-4} + T_{i-6} + T_{i-9} + \ldots \\ &< 0 \end{aligned}$$

Therefore, after step 5, all the entries from index 1 through index $i-1$ are legal, and from index $i-1$ to index i_1 the index we start the fourth step on, is also legal. We return to step 3, but this time we perform the fourth transformation at a larger i_1 initial index. Therefore, the algorithm eventually terminates and our proof is complete. ◻

Chapter 5

Conjectures and data

In this chapter, we discuss unresolved conjectures and supporting data.

5.1 Almost Balanced Base-($2b$)

Given the result in Remark 1 and Theorem 3, it may seem that the approach to finding a summand minimal representation for base-($2b$) numbers should follow the odd case as much as possible. However, the fact that we have an even base means that we cannot have an entirely balanced representation, which leads to the following conjecture.

Conjecture 1. *The representation for base-*($2b$) *corresponding to the diagram* $[(b, b-1), (b-1, b), (b, b-1), (b-1, b), \dots]$ *is summand minimal.*

This conjecture is supported by the following result and preliminary data:

- The conjecture is true for the case $b = 1$ the binary representation with no consecutive terms.
- When $b = 2$, we have that the average number of summands needed for numbers in $[4^6, 4^7)$ is as follows

Repeating Diagram	Average Number of Summands
$[(3,0),\dots]$	11.0
$[(2,1),\dots]$	7.166666666666667
$[(1,2),\dots]$	9.0
$[(2,1),(1,2),(2,1),(2,1),\dots]$	7.082275390625
$[(2,1),(1,2),\dots]$	7.079996744791667
$[(2,1),(1,2),(1,2),(2,1),\dots]$	7.130208333333333

From the table, we can see that the diagram $[(2,1),(1,2),\dots]$ has the lowest average number of summands compared to some possible alternative diagrams.

- Also, for base-$(2b)$, if the diagram is to correspond to a summand minimal representation, then the initial entry must be $(p_1, n_1) = (b, b-1)$. Consider the number $2b^2 - 1$ which requires at least $b + 1$ summands (we see this by simply considering when the largest summand is $(2b)^2, (2b)^3$, or greater) and is summand minimal in the sum $2b^2 - 1 = -1 \cdot 1 + b \cdot (2b)$. This requires that in the diagram $p_1 \geq b$ to allow $b \cdot (2b)$, as otherwise we have largest summand $(2b)^2$ at which point we require more than $b + 1$ summands.

  Now, consider the number $b + 1$ which requires at least b summands and is summand minimal in the sum $-(b-1) \cdot 1 + 1 \cdot (2b)$. If $p_1 \geq b + 1$, then we have that in that diagram the number $b + 1$ must be written $(b+1) \cdot 1$ which uses $b+1$ summands and thus is not summand minimal. Thus, we have that $p_1 \leq b$ and therefore we have $p_1 = b$ with $n_1 = b - 1$ as claimed.

  Unfortunately, this analysis becomes much more difficult starting with (p_2, n_2) since it makes use of well-selected counterexamples. The conjecture thus remains unsolved.

5.2 Existence of Summand Minimal Diagrams

The second conjecture concerns the existence of summand minimal diagrams. We note that in the case of Generalized Zeckendorf decompositions, summand minimality is only achieved according to the following result, rewritten using the concept of diagrams.

Theorem 6 (Cordwell et al. (2018))**.** *For a given positive linear recurrence* *sequence with the recurrence relation*

$$a_n = d_1 a_{n-1} + d_2 a_{n-2} + \dots d_\ell a_{n-\ell}$$

the diagram $D = [(d_1, 0), (d_2, 0), \dots, (d_{\ell-1}, 0), (d_\ell - 1, 0), (d_1, 0), \dots]$ *which has all* n_i *set to* 0*, is summand minimal over all linear sums with nonnegative coefficients if and only if* $d_1 \geq d_2 \geq \cdots \geq d_\ell$*.*

This theorem gives a clear condition for the nonnegative coefficients case. However, we believe that since each PLRS has infinitely many diagrams associated to it in the generalized far-difference case, then there might be enough possibilities compared to just a single diagram for the nonnegative case. Thus, we formulate the following conjecture:

Conjecture 2. *For any PLRS, there exists a summand minimal representation corresponding to some diagram associated with the sequence.*

Chapter 6

Future work

6.1 Statistics of representations

While the number of summands is interesting to study, we may also turn our attention to well-established statistics such as the Hamming Weight mentioned in Muir and Stinson (2006), which is the number of digits with nonzero coefficients. We may also be interested in the distribution of the number of summands and whether it is Gaussian as studied in Miller and Wang (2012).

6.2 Descriptions of representations for general diagram

We note that being able to describe the resulting representations from a given diagram is crucial in proving interesting properties about the diagram such as summand minimality in Chapter 4. Furthermore, we would need to be able to describe diagrams like $D = [(b, b-1), (b-1, b), (b, b-1), (b-1, b), \dots]$ to make progress on Conjecture 1.

Furthermore, the (s, d)-representations in Demontigny et al. (2014) with $s < d$ does not correspond to any PLRS but may correspond to diagrams with non-repeating patterns of (p_i, n_i). For example, the diagram $D = [(1, 0), (0, 0), (1, 1), (0, 0), (1, 1), \dots]$ belongs to the $(2, 3)$-representation and will have the entry $(1, 0)$ only at the beginning, and thus correspond to no PLRS. However, we may also say that this diagram corresponds, in some sense, to an infinitely long PLRS with recurrence relation $a_n = a_{n-1} + 2a_{n-3} + 2a_{n-5} + \dots$. How do we make sense of these irrational diagrams?

6.3 Representations that are not described by diagrams

There are other representations that appear not to be describable by diagrams such as the width-w non-adjacent forms.

Definition 14 (Muir and Stinson (2006))**.** *A width-w non-adjacent form is a representation of integers with powers of* 2 *where for any w consecutive digits, at most one digit may have a nonzero coefficient.*

While the width-2 non-adjacent form can be captured by the diagram $D = [(1,0),(0,1),(1,0),(0,1),\dots]$, all higher widths require digits with coefficients of size larger than 2, which our diagrams cannot provide. Thus, there possibly exists a further generalization of the concept of diagrams presented in this paper.

Appendix A

Known Diagrams

This appendix contains several examples of diagrams corresponding to various named representations.

A.1 Base-3

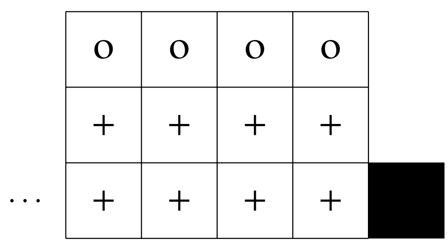


Figure A.1 Diagram of the base-3 representation

A.2 Balanced Ternary

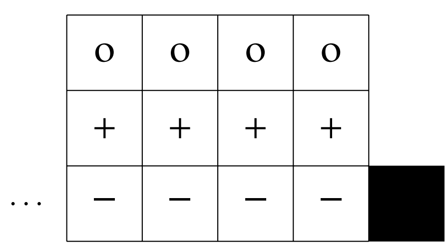


Figure A.2 Diagram of the balanced ternary representation

A.3 Zeckendorf

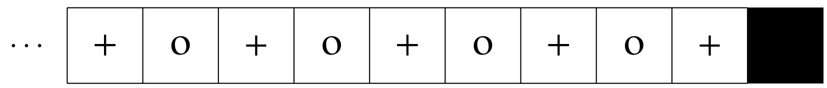


Figure A.3 Diagram of the Zeckendorf decomposition

A.4 Far-Difference

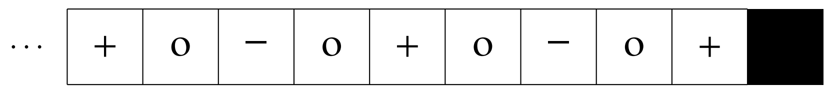


Figure A.4 Diagram of the far-difference representation

A.5 Tribonacci Far-Difference

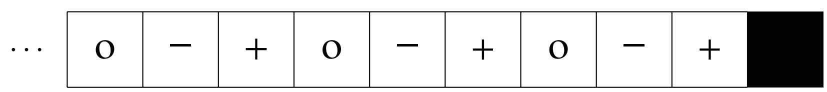


Figure A.5 Diagram of the Tribonacci far-difference representation

A.6 Binary

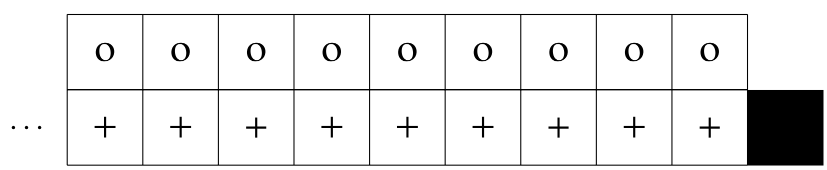


Figure A.6 Diagram of the binary representation

A.7 Width-2 Non-Adjacent Form

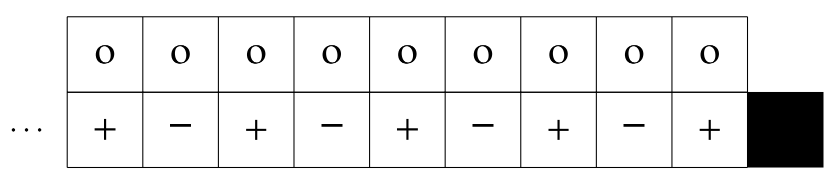


Figure A.7 Diagram of the width-2 non-adjacent form representation

Appendix B

Data of representations of integers for some given diagrams

The diagrams for the following representations can be found in Appendix A.

B.1 Base-2 Representation

B.1.1 Width-2 Non-Adjacent Form

n	representation	n	representation	n	representation
0	0	11	$16-4-1$	22	$32-8-2$
1	1	12	$16-4$	23	$32-8-1$
2	2	13	$16-4+1$	24	$32-8$
3	$4-1$	14	$16-2$	25	$32-8+1$
4	4	15	$16-1$	26	$32-8+2$
5	$4+1$	16	16	27	$32-4-1$
6	$8-2$	17	$16+1$	28	$32-4$
7	$8-1$	18	$16+2$	29	$32-4+1$
8	8	19	$16+4-1$	30	$32-2$
9	$8+1$	20	$16+4$	31	$32-1$
10	$8+2$	21	$16+4+1$	32	32

Table B.1 Outputs for Width-2 Non-Adjacent Form

B.2 Base-3 Representations

B.2.1 Balanced Ternary

n	representation	n	representation	n	representation
0	0	11	$9+3-1$	22	$27-9+3+1$
1	1	12	$9+3$	23	$27-3-1$
2	$3-1$	13	$9+3+1$	24	$27-3$
3	3	14	$27-9-3-1$	25	$27-3+1$
4	$3+1$	15	$27-9-3$	26	$27-1$
5	$9-3-1$	16	$27-9-3+1$	27	27
6	$9-3$	17	$27-9-1$	28	$27+1$
7	$9-3+1$	18	$27-9$	29	$27+3-1$
8	$9-1$	19	$27-9+1$	30	$27+3$
9	9	20	$27-9+3-1$	31	$27+3+1$
10	$9+1$	21	$27-9+3$	32	$27+9-3-1$

Table B.2 Outputs for Balanced Ternary

B.2.2 Far Ternary

n	representation	n	representation	n	representation
0	0	11	$9+1+1$	22	$27-3-1-1$
1	1	12	$9+3$	23	$27-3-1$
2	$1+1$	13	$9+3+1$	24	$27-3$
3	3	14	$9+3+1+1$	25	$27-1-1$
4	$3+1$	15	$9+3+3$	26	$27-1$
5	$3+1+1$	16	$9+9-1-1$	27	27
6	$3+3$	17	$9+9-1$	28	$27+1$
7	$9-1-1$	18	$9+9$	29	$27+1+1$
8	$9-1$	19	$9+9+1$	30	$27+3$
9	9	20	$9+9+1+1$	31	$27+3+1$
10	$9+1$	21	$27-3-3$	32	$27+3+1+1$

Table B.3 Outputs for Far Ternary

B.3 Fibonacci Representations

B.3.1 Zeckendorf

n	representation	n	representation	n	representation
0	0	11	8 + 3	22	21 + 1
1	1	12	8 + 3 + 1	23	21 + 2
2	2	13	13	24	21 + 3
3	3	14	13 + 1	25	21 + 3 + 1
4	3 + 1	15	13 + 2	26	21 + 5
5	5	16	13 + 3	27	21 + 5 + 1
6	5 + 1	17	13 + 3 + 1	28	21 + 5 + 2
7	5 + 2	18	13 + 5	29	21 + 8
8	8	19	13 + 5 + 1	30	21 + 8 + 1
9	8 + 1	20	13 + 5 + 2	31	21 + 8 + 2
10	8 + 2	21	21	32	21 + 8 + 3

Table B.4 Outputs for Zeckendorf

B.3.2 Far Difference

n	representation	n	representation	n	representation
0	0	11	13 − 2	22	21 + 1
1	1	12	13 − 1	23	21 + 2
2	2	13	13	24	21 + 3
3	3	14	13 + 1	25	34 − 8 − 1
4	5 − 1	15	13 + 2	26	34 − 8
5	5	16	21 − 5	27	34 − 8 + 1
6	8 − 2	17	21 − 5 + 1	28	34 − 8 + 2
7	8 − 1	18	21 − 3	29	34 − 5
8	8	19	21 − 2	30	34 − 5 + 1
9	8 + 1	20	21 − 1	31	34 − 3
10	13 − 3	21	21	32	34 − 2

Table B.5 Outputs for Far Difference

B.4 Tribonacci Representations

B.4.1 Tribonacci Zeckendorf

n	representation	n	representation	n	representation
0	0	11	7 + 4	22	13 + 7 + 2
1	1	12	7 + 4 + 1	23	13 + 7 + 2 + 1
2	2	13	13	24	24
3	2 + 1	14	13 + 1	25	24 + 1
4	4	15	13 + 2	26	24 + 2
5	4 + 1	16	13 + 2 + 1	27	24 + 2 + 1
6	4 + 2	17	13 + 4	28	24 + 4
7	7	18	13 + 4 + 1	29	24 + 4 + 1
8	7 + 1	19	13 + 4 + 2	30	24 + 4 + 2
9	7 + 2	20	13 + 7	31	24 + 7
10	7 + 2 + 1	21	13 + 7 + 1	32	24 + 7 + 1

Table B.6 Outputs for Tribonacci Zeckendorf

B.4.2 Tribonacci Far Difference

n	representation	n	representation	n	representation
0	0	11	13 − 2	22	24 − 2
1	1	12	13 − 1	23	24 − 1
2	2	13	13	24	24
3	4 − 1	14	13 + 1	25	24 + 1
4	4	15	13 + 2	26	24 + 2
5	7 − 2	16	24 − 7 − 1	27	24 + 4 − 1
6	7 − 1	17	24 − 7	28	24 + 4
7	7	18	24 − 7 + 1	29	44 − 13 − 2
8	7 + 1	19	24 − 7 + 2	30	44 − 13 − 1
9	13 − 4	20	24 − 4	31	44 − 13
10	13 − 4 + 1	21	24 − 4 + 1	32	44 − 13 + 1

Table B.7 Outputs for Far Difference

Bibliography

Alpert, Hannah. 2009. Differences of multiple fibonacci numbers. *Integers* 9(6):745–749. doi:10.1515/INTEG.2009.061.

Arndt, J. 2010. *Matters Computational: Ideas, Algorithms, Source Code*. Springer Berlin Heidelberg. URL https://books.google.com/books?id=HsRHS6u7e80C.

Cordwell, Katherine, Max Hlavacek, Chi Huynh, Steven J. Miller, Carsten Peterson, and Yen Nhi Truong Vu. 2018. Summand minimality and asymptotic convergence of generalized zeckendorf decompositions. *Research in Number Theory* 4(4):43. doi:10.1007/s40993-018-0137-7.

Demontigny, Philippe, Thao Do, Archit Kulkarni, Steven J. Miller, and Umang Varma. 2014. A generalization of fibonacci far-difference representations and gaussian behavior. doi:10.48550/arXiv.1309.5600. 1309.5600 [math].

Grabner, P. J, and R. F Tichy. 1990. Contributions to digit expansions with respect to linear recurrences. *Journal of Number Theory* 36(2):160–169. doi:10.1016/0022-314X(90)90070-8.

Hamlin, Nathan, and William Webb. 2012. Representing positive integers as a sum of linear recurrence sequences. *The Fibonacci Quarterly* 50.

Knuth, Donald E. 1997. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. USA: Addison-Wesley Longman Publishing Co., Inc.

Lekkerkerker, C.G. 1951. Voorstelling van natuurlijke getallen door een som van getallen van fibonacci. *Stichting Mathematisch Centrum* URL https://ir.cwi.nl/pub/6922. Issue: ZW 30/51.

Lentfer, John. 2021. Tiling representations of zeckendorf decompositions. *HMC Senior Theses* .

Miller, Steven J., and Yinghui Wang. 2012. From fibonacci numbers to central limit type theorems. *Journal of Combinatorial Theory, Series A* 119(7):1398–1413. doi:10.1016/j.jcta.2012.03.014.

Muir, James A., and Douglas R. Stinson. 2006. Minimality and other properties of the width-w nonadjacent form. *Mathematics of Computation* 75(253):369–384. URL https://www.jstor.org/stable/4100158.

Zeckendorf, Edouard. 1972. Representations des nombres naturels par une somme de nombres de fibonacci on de nombres de lucas. *Bulletin de La Society Royale des Sciences de Liege* 179–182. URL https://cir.nii.ac.jp/crid/1570009749187075840.