# Minimal Weight and Colexicographically Minimal Integer Representations

Clemens Heuberger*
Institut für Mathematik B
Technische Universität Graz, Graz, Austria
clemens.heuberger@tugraz.at

James A. Muir†
School of Computer Science
Carleton University, Ottawa, Canada
jamuir@scs.carleton.ca

21 June 2006 16:53:29 EDT

## Abstract

Redundant number systems (e.g., signed binary representations) have been utilized to efficiently implement algebraic operations required by public-key cryptosystems, especially those based on elliptic curves. Several families of integer representations have been proposed that have a minimal number of nonzero digits (so-called *minimal weight* representations). We observe that many of the constructions for minimal weight representations actually work by building representations which are minimal in another sense. For a given set of digits, these constructions build *colexicographically minimal* representations; that is, they build representations where each nonzero digit is positioned as far left (toward the most significant digit) as possible. We utilize this strategy in a new algorithm which constructs a very general family of minimal weight dimension-*d* *joint* representations for any $d \geq 1$. The digits we use are from the set $\{a \in \mathbb{Z} : \ell \leq a \leq u\}$ where $\ell \leq 0$ and $u \geq 1$ are integers. By selecting particular values of $\ell$ and $u$, it is easily seen that our algorithm generalizes many of the minimal weight representations previously described in the literature. From our algorithm, we obtain a syntactical description of a particular family of dimension-*d* joint representations; any representation which obeys this syntax must be both colexicographically minimal and have minimal weight; moreover, every vector of integers has exactly one representation that satisfies this syntax. We utilize this syntax in a combinatorial analysis of the weight of the representations.

**Key words.** redundant number systems, signed digits, integer representations, joint representations, minimal weight, colexicographic order, Joint Sparse Form.

**AMS classification.** Primary 11A63; Secondary 94A60, 68W40.

## 1 Introduction and Background

In this paper, we deal with a class of integer representations known as *joint representations*.

**Definition 1.1.** Let $d \geq 1$ and $r \geq 2$ be integers. A *dimension-d radix-r joint representation* is a sum of the form $\sum_{j=0}^{s-1} A_j r^j$ where each $A_j \in \mathbb{Z}^{d\times 1}$.

[*] C. Heuberger is supported by the Austrian Science Foundation FWF, project S9606, that is part of the Austrian National Research Network "Analytic Combinatorics and Probabilistic Number Theory."

[†] J.A. Muir is supported by a Natural Sciences and Engineering Research Council of Canada Postdoctoral Fellowship.

Joint representations are representations of *vectors* of integers. We are particularly interested in the case when the radix $r$ is equal to 2. If $N \in \mathbb{Z}^{d\times 1}$ is a vector such that $N = \sum_{j=0}^{s-1} A_j 2^j$, then we say that $\sum_{j=0}^{s-1} A_j 2^j$ is a *radix-*2 *joint representation of* $N$. To denote radix-2 joint representations, we use the following notation:

$$(A_{s-1}\dots A_1A_0)_2 := A_{s-1}2^{s-1} + \cdots + A_1 2^1 + A_0.$$

Each $A_j$ is a column vector and the entries in these column vectors are called *digits*. Note that a dimension-1 joint representation is an ordinary integer representation.

**Example 1.2.** Here is a dimension-4 radix-2 joint representation with digits from the set $\{0, 1\}$:

$$\begin{pmatrix}0111\\1011\\1101\\1110\end{pmatrix}_2 = \begin{pmatrix}0\\1\\1\\1\end{pmatrix}2^3 + \begin{pmatrix}1\\0\\1\\1\end{pmatrix}2^2 + \begin{pmatrix}1\\1\\0\\1\end{pmatrix}2^1 + \begin{pmatrix}1\\1\\1\\0\end{pmatrix}2^0 = \begin{pmatrix}7\\11\\13\\14\end{pmatrix}.$$

The values of $A_3, A_2, A_1, A_0$ are as listed above. $\diamondsuit$

**Example 1.3.** Here is a dimension-4 radix-2 joint representation with digits from the set $\{0, 1, 2, 3\}$:

$$\begin{pmatrix}103\\203\\301\\302\end{pmatrix}_2 = \begin{pmatrix}1\\2\\3\\3\end{pmatrix}2^2 + \begin{pmatrix}0\\0\\0\\0\end{pmatrix}2^1 + \begin{pmatrix}3\\3\\1\\2\end{pmatrix}2^0 = \begin{pmatrix}7\\11\\13\\14\end{pmatrix}.$$

The column vector $N = (7, 11, 13, 14)^\mathsf{T}$ has several other joint representations which use the digits $\{0, 1, 2, 3\}$. Notice that the representation above has just two nonzero columns, namely $A_2$ and $A_0$. The representation in Example 1.2 has four nonzero columns. $\diamondsuit$

Joint representations were introduced by Solinas [18] when he considered how to compute a linear combination of two elliptic curve points efficiently; i.e., he considered the computation of $n_1P_1 + n_2P_2$, where $n_1, n_2 \in \mathbb{Z}$ and $P_1, P_2$ are elements of an elliptic curve group. What motivated Solinas to consider this particular algebraic operation was its use in the Elliptic Curve Digital Signature Algorithm's signature verification operation [4].

The algorithm for computing $n_1P_1 + n_2P_2$ Solinas investigated is a special case of an algorithm due to Straus [19, see the proof at the bottom of page 807].[1] The general form of Straus' algorithm is presented in Appendix A and the special case is presented as Algorithm 1. The general algorithm computes $\sum_{i=1}^{d} n_iP_i$ using a dimension-$d$ radix-$2^k$ joint representation of $N = (n_1, n_2, \dots, n_d)^\mathsf{T}$ with digits in $\{0, 1, \dots, 2^k - 1\}$. Setting the parameters $d = 2$ and $k = 1$, we get an algorithm that computes $n_1P_1 + n_2P_2$ using a dimension-2 joint representation with digits in $\{0, 1\}$; i.e., the rows of the joint representation are just ordinary *binary* representations. For each nonzero column of this joint representation (not counting $A_{s-1}$), an elliptic curve addition operation is performed (see lines 6–9). These addition operations are computationally expensive, so it is desirable to do only as few of them as necessary.

For a given value of $N$, we could reduce the number of addition operations in Algorithm 1 by utilizing a joint representation of $N$ with fewer nonzero columns. However, when restricted to the digits $\{0, 1\}$, this observation is of little consequence; every vector $N$ of nonnegative integers has exactly one radix-2 joint representation with digits in $\{0, 1\}$. But if we instead consider joint representations which use the digits $\{0, \pm 1\}$, the situation changes.

**Algorithm 1** Straus' algorithm for $d = 2, k = 1$

```
Input: $N = (n_1, n_2)^\mathsf{T}$, $P = (P_1, P_2)$
Output: $Q = n_1P_1 + n_2P_2$
$R \leftarrow P_1 + P_2$
$A_{s-1} \dots A_1 A_0 \leftarrow$ the columns of the dimension-2 radix-2 joint rep. of $N$ with digits from $\{0, 1\}$
based on the value of $A_{s-1}$, initialize $Q$ to one of $P_1$, $P_2$, $R$
for $j = s - 2 \dots 0$ do
    $Q \leftarrow 2Q$
    if $A_j \neq \vec{0}$ then
        if $A_j = (1, 0)^\mathsf{T}$ then $Q \leftarrow Q + P_1$
        else if $A_j = (0, 1)^\mathsf{T}$ then $Q \leftarrow Q + P_2$
        else if $A_j = (1, 1)^\mathsf{T}$ then $Q \leftarrow Q + R$
return $Q$
```

It is possible to modify Algorithm 1 so that it processes a radix-2 joint representation of $N$ with digits in $\{0, \pm 1\}$.[2] This is done in Algorithm 2. Notice now that for every nonzero column in $A_{s-1} \dots A_1 A_0$ (not counting $A_{s-1}$) an addition or *subtraction* operation is carried out. In elliptic curve groups, point subtraction can be done just as efficiently as point addition, so utilizing this operation does not carry any extra cost. Every nonzero vector $N \in \mathbb{Z}^{2\times 1}$ has infinitely many radix-2 joint representation with digits from $\{0, \pm 1\}$, and any one of these can be used in Algorithm 2. This led Solinas to the following problem:

**Problem 1.4.** *Given $N \in \mathbb{Z}^{2\times 1}$, construct a radix-2 joint representation of $N$ using the digits $\{0, \pm 1\}$ that has a minimal number of nonzero columns.*

The number of nonzero columns in a joint representation is often referred to as its *weight*.

Solinas solved Problem 1.4 by presenting an algorithm that constructs a canonical joint representation for any pair of integers called the *joint sparse form* (JSF). The JSF was developed as a generalization of the well-known *nonadjacent form* (NAF) due to Reitwiesner [17]. The NAF is a family of radix-2 representations with digits in $\{0, \pm 1\}$ that have the property that *of any two consecutive digits, at most one is nonzero* (i.e., their nonzero digits are nonadjacent). Reitwiesner showed that every integer has exactly one NAF, and that this representation has a minimal number of nonzero digits. Solinas showed every pair of integers has exactly one JSF, and that this representation has a minimal number of nonzero columns.

**Example 1.5.** Here are two radix-2 joint representations of $(602, 1365)^\mathsf{T}$ with digits from $\{0, \pm 1\}$:

$$\begin{pmatrix} 001010\bar{1}0\bar{1}010 \\ 10\bar{1}0\bar{1}0\bar{1}0\bar{1}0\overline{11} \end{pmatrix}_2, \begin{pmatrix} 01010\bar{1}0\bar{1}010 \\ 10101010101 \end{pmatrix}_2.$$

Note that we use "$\bar{1}$" to denote "$-1$". The first representation is composed of 12 columns, 7 of which are nonzero. The second representation has 11 columns and all 11 are nonzero. Each row of the second representation is a NAF. This demonstrates that taking each row of a joint representation to be a NAF does not necessarily give a minimal number of nonzero columns. ◊

Solinas suggested some additional research problems involving joint representations. The ones most relevant to the work presented here are the following:

[1] This special case of Straus' algorithm is often incorrectly attributed to Shamir. Bernstein explains this and a number of other misconceptions regarding exponentiation algorithms in a manuscript [2].

[2] The benefits of using radix-2 representations with digits $\{0, \pm 1\}$ in elliptic curve arithmetic were first demonstrated by Morain and Olivos [12].

**Algorithm 2** Straus' algorithm for $d = 2, k = 1$ modified to use the digits $\{0, \pm 1\}$

```
Input: $N = (n_1, n_2)^\mathsf{T}$, $P = (P_1, P_2)$
Output: $Q = n_1 P_1 + n_2 P_2$
$R \leftarrow P_1 + P_2$, $S \leftarrow P_1 - P_2$
$A_{s-1} \ldots A_1 A_0 \leftarrow$ the columns of a dimension-2 radix-2 joint rep. of $N$ with digits from $\{0, \pm 1\}$
based on the value of $A_{s-1}$, initialize $Q$ to one of $\pm P_1, \pm P_2, \pm R, \pm S$
for $j = s - 2 \ldots 0$ do
    $Q \leftarrow 2Q$
    if $A_j \neq \vec{0}$ then
        if $A_j = (1, 0)^\mathsf{T}$ then $Q \leftarrow Q + P_1$
        else if $A_j = (\bar{1}, 0)^\mathsf{T}$ then $Q \leftarrow Q - P_1$
        else if $A_j = (0, 1)^\mathsf{T}$ then $Q \leftarrow Q + P_2$
        else if $A_j = (0, \bar{1})^\mathsf{T}$ then $Q \leftarrow Q - P_2$
        else if $A_j = (1, 1)^\mathsf{T}$ then $Q \leftarrow Q + R$
        else if $A_j = (\bar{1}, \bar{1})^\mathsf{T}$ then $Q \leftarrow Q - R$
        else if $A_j = (1, \bar{1})^\mathsf{T}$ then $Q \leftarrow Q + S$
        else if $A_j = (\bar{1}, 1)^\mathsf{T}$ then $Q \leftarrow Q - S$
return $Q$
```

**Problem 1.6.** *Generalize the JSF to dimension $d$ where $d \geq 3$.*

**Problem 1.7.** *Give an analogue of the JSF which uses digits other than* $\{0, \pm 1\}$.

Problem 1.6 was solved independently by Proos [16] and by Grabner, Heuberger and Prodinger [6]. Both works demonstrate how to build arbitrary dimension-$d$ radix-2 joint representations using the digits $\{0, \pm 1\}$ that have minimal weight. To date, there has been little progress made on Problem 1.7.

**Our contributions.** We consider the problem of constructing minimal weight dimension-$d$ radix-2 joint representations, for arbitrary $d \geq 1$, which use the digits $\{a \in \mathbb{Z} : \ell \leq a \leq u\}$, where $\ell \leq 0$ and $u \geq 1$ are integers. We provide an efficient algorithm which constructs such representations. By selecting particular values of $d, \ell, u$, it can be seen that our construction generalizes a number of previously known minimal weight representations (see Table 1). One unusual property of the digit sets we consider is that they are not necessarily symmetric about zero; i.e., they can contain an unequal number of negative and positive digits.

| *minimal weight representation* | $d$ | $\ell$ | $u$ |
|---|---|---|---|
| nonadjacent form [17] | $1$ | $-1$ | $1$ |
| width-$w$ nonadjacent form [15] [1] [14] | $1$ | $-(2^{w-1} - 1)$ | $2^{w-1} - 1$ |
| signed fractional window representation [15] [11] | $1$ | $-m$ | $m$ |
| simple joint sparse form [6] | $\geq 1$ | $-1$ | $1$ |

TABLE 1: Families of minimal weight integer representations (citations are given to minimality proofs).

An important concept we emphasize is the commonality between minimal weight representations and *colexicographically* minimal representations.[3] For a fixed set of digits, the set of all joint representations of a vector $N \in \mathbb{Z}$ can be ordered by comparing the *positions* of their nonzero columns, as read right-to-left. Representations which are minimal with respect to this (colexicographic) ordering share a number of

[3] Common properties of minimal weight and colexicographically minimal integer representations were described in Muir's Ph.D. thesis [13, see Ch.4].

properties with those that have minimal weight. Thus, for a given set of digits, it is natural to ask whether a colexicographically minimal representation has minimal weight. For the digit sets we consider, this is indeed true, and the design of our algorithm exploits this fact.

The main results presented herein can be summarized as follows:

- the outputs of our algorithm are minimal weight representations (Theorem 1).
- the outputs of our algorithm are colexicographically minimal representations (Theorem 2).
- any representation with digits restricted to $\{a \in \mathbb{Z} : \ell \leq a \leq u\}$ that is colexicographically minimal must also have minimal weight (Corollary 3).
- any representation with digits restricted to $\{a \in \mathbb{Z} : \ell \leq a \leq u\}$ which satisfies three syntactic properties must be colexicographically minimal and have minimal weight. Every integer vector admits exactly one representation satisfying these three syntactic properties (Theorem 4).
- the probability distribution of the weight of the $n$ least significant columns of a colexicographically minimal representation of $N$ can be explicitly determined; from this, asymptotic formulae for its expected value and variance follow (Theorem 5).

**Related Work.** Integer representations using digit sets of the form $\{a \in \mathbb{Z} : \ell \leq a \leq u\}$ where $\ell \leq 0$ and $u \geq 1$ have been proposed previously in the literature. Phillips and Burgess [15] introduced a "generalized sliding window" transformation which is applied to an integer's standard radix-$r$ representation where $r \geq 2$. If the parameters $\ell$ and $u$ satisfy $\ell = 0$ or $\ell \equiv 1 \pmod{r}$ and $u \equiv -1 \pmod{r}$, then they are able to prove that their transformation produces a minimal weight representation. In the case where $r = 2$, which is the only radix value considered in our work, these two conditions can always be satisfied because of the fact that only the odd digits from $\{a \in \mathbb{Z} : \ell \leq a \leq u\}$ are utilized (e.g., if $u$ is even, it can be replaced with $u - 1$ and then their proof will go through). Although Phillips and Burgess consider only integer representations (not joint representations), our technique for proving minimality is similar to theirs (both works use induction and the properties of addition) with the exception that we do not require any extra conditions on $\ell$ and $u$; this is important since our joint representations, in general, utilize both even and odd digits from $\{a \in \mathbb{Z} : \ell \leq a \leq u\}$.

The connection between colexicographically minimal representations and minimal weight representations is unique to our work. This observation provides some perspective on sliding window transformations, including the one proposed by Phillips and Burgess. Sliding window transformations tend to produce colexicographically minimal representations, and this is why they often give minimal weight representations.

**Outline.** We begin by presenting some preliminary concepts and notations in §2. In §3 we explain the design of our algorithm. A number of properties common to both minimal weight and colexicographically minimal representations are presented in §4. That the outputs of our algorithm are minimal is established in §5. A syntax which characterizes the outputs, along with an analysis of their weight, is given in §6. We end with some remarks in §7.

# 2 Preliminaries

## 2.1 Column-strings

Let $D \subset \mathbb{Z}$ be a finite set of digits with $0 \in D$. $D^{d\times 1}$ denotes the set of all dimension-$d$ column vectors with entries (digits) from $D$. We use $\vec{0}$ to denote the all-zero column vector. Column vectors can be concatenated together to form strings of column vectors.

Given $N \in \mathbb{Z}^{d\times 1}$, when looking for a column-string $\mathcal{A} = A_{s-1}\dots A_1A_0$ such that $(\mathcal{A})_2 = N$, leading zeros do not matter since we obviously have $(\mathcal{A})_2 = N$ if and only if $(\vec{0}\mathcal{A})_2 = N$. We denote the number of nonzero columns in the string $\mathcal{A}$ by $\mathsf{wt}(\mathcal{A})$. This value is often referred to as the *joint Hamming weight*, or simply, the *weight*, of $\mathcal{A}$.

To denote the columns of a joint representation, we use capital letters; e.g., $A_{s-1}\dots A_1A_0$ where each $A_j \in D^{d\times 1}$. To denote the digits of an integer representation, rather than a joint representation, we use lower case letters; e.g., $a_{s-1}\dots a_1a_0$ where each $a_j \in D$.

## 2.2 Colexicographic Order

For a vector $N \in \mathbb{Z}^{d\times 1}$ and a digit set $D \subset \mathbb{Z}$, consider the set of all dimension-$d$ radix-2 joint representations of $N$ with digits restricted to $D$. We can order the representations in this set by considering the positions of their nonzero columns.

Suppose $(A_{s-1}\dots A_1A_0)_2 = N$. From $\mathcal{A} = A_{s-1}\dots A_1A_0$, we derive a binary string $\mathsf{char}(\mathcal{A})$ defined as follows: $\mathsf{char}(\mathcal{A}) = a_{s-1}\dots a_2a_1a_0$ where

$$a_i := \begin{cases} 0 & \text{if } A_i \text{ is a zero column} \\ 1 & \text{otherwise.} \end{cases}$$

Now, if $\mathcal{B} = B_{s-1}\dots B_1B_0$ and $(\mathcal{B})_2 = N$, we write $\mathcal{A} \preceq \mathcal{B}$ if $\mathsf{char}(\mathcal{A})$ is less than or equal to $\mathsf{char}(\mathcal{B})$ when they are compared *colexicographically*. Colexicographic order is similar to lexicographic order except that strings are compared by reading their symbols *right-to-left* rather than left-to-right. Here is an example to illustrate:

```
column                   string
digit                   minimal
minimal                  column
radix            representation
representation           weight
string                    digit
weight                    radix
```

The strings in the left column are ordered lexicographically, and the strings in the right column are ordered colexicographically.

Comparing integer representations using colexicographic order has been utilized previously in the literature. Grabner, Heuberger and Prodinger [6, see p. 330] used colexicographic order to prove that their Simple Joint Sparse Form has minimal weight. Muir and Stinson [14] showed that the width-$w$ nonadjacent form of an integer is uniquely determined as its colexicographically minimal representation.

## 2.3 The Digit Set $D_{\ell,u}$

For integers $\ell \le 0$ and $u \ge 1$, we define the digit set

$$D_{\ell,u} := \{a \in \mathbb{Z} : \ell \le a \le u\}.$$

Notice that because of the bounds on $\ell$ and $u$, $D_{\ell,u}$ always contains the digits $0, 1$. Also, if $D_{\ell,u}$ contains negative digits, then $-1 \in D_{\ell,u}$. On the other hand, if $\ell = 0$, then the digits in $D_{0,u}$ can certainly only be used to represent nonnegative numbers. Observe that $\#D_{\ell,u} = u - \ell + 1$.

Given a set of digits $D_{\ell,u}$, we define $w$ to be the unique positive integer that satisfies

$$2^{w-1} \le \#D_{\ell,u} < 2^w.$$

This implies that $D_{\ell,u}$ contains a complete system of residues modulo $2^{w-1}$. Two such systems are

$$\begin{aligned} \mathsf{lower}(D_{\ell,u}) &:= \{a \in D_{\ell,u} : \ell \leq a < \ell + 2^{w-1}\}, \\ \mathsf{upper}(D_{\ell,u}) &:= \{a \in D_{\ell,u} : u - 2^{w-1} < a \leq u\}. \end{aligned}$$

Depending on the values of $\ell$ and $u$, these two sets might coincide. Note that $D_{\ell,u}$ does *not* contain a complete system of residues modulo $2^w$ because $\#D_{\ell,u} < 2^w$. Since $\{0, 1\} \subseteq D_{\ell,u}$, it is always the case that $2 \leq \#D_{\ell,u}$; from this, we see that $w \geq 2$.

**Example 2.1.** For $\ell = -3$ and $u = 7$, we have

$$D_{-3,7} = \{-3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7\},$$

and $2^3 \leq \#D_{-3,7} < 2^4$ (i.e., $w = 4$). Each of the eight congruence classes modulo $2^3$ has either one or two representatives in $D_{-3,7}$:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
|  |  |  |  |  | $\bar{3}$ | $\bar{2}$ | $\bar{1}$ |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 . |

Each of the sixteen congruence classes modulo $2^4$ has either zero or one representative in $D_{-3,7}$:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |  |  |  |  |  | $\bar{3}$ | $\bar{2}$ | $\bar{1}$ . |

$\diamondsuit$

We say that a digit $a \in D_{\ell,u}$ is *unique modulo* $2^{w-1}$ if there is no other digit $a' \in D_{\ell,u}$ such that $a \equiv a' \pmod{2^{w-1}}$ and $a' \neq a$. The set of digits of $D_{\ell,u}$ which are unique modulo $2^{w-1}$ is denoted by $\mathsf{unique}(D_{\ell,u})$. It is easily seen that

$$\mathsf{unique}(D_{\ell,u}) = \mathsf{lower}(D_{\ell,u}) \cap \mathsf{upper}(D_{\ell,u}) = \{a \in D_{\ell,u} : u - 2^{w-1} < a < \ell + 2^{w-1}\}. \tag{1}$$

The set of digits $D_{\ell,u} \setminus \mathsf{unique}(D_{\ell,u})$ is denoted by $\mathsf{nonunique}(D_{\ell,u})$. We have

$$\mathsf{nonunique}(D_{\ell,u}) = \mathsf{lower}(D_{\ell,u}) \triangle \mathsf{upper}(D_{\ell,u}) = \{a \in D_{\ell,u} : a \leq u - 2^{w-1} \text{ or } \ell + 2^{w-1} \leq a\}; \tag{2}$$

here, $\triangle$ denotes the symmetric difference of two sets. From Example 2.1, we see that

$$\begin{aligned} \mathsf{unique}(D_{-3,7}) &= \mathsf{lower}(D_{-3,7}) \cap \mathsf{upper}(D_{-3,7}) = \{0, 1, 2, 3, 4\}, \\ \mathsf{nonunique}(D_{-3,7}) &= \mathsf{lower}(D_{-3,7}) \triangle \mathsf{upper}(D_{-3,7}) = \{-3, -2, -1, 5, 6, 7\}. \end{aligned}$$

Given $n \in \mathbb{Z}$, to compute a digit $a \in D_{\ell,u}$ such that $n \equiv a \pmod{2^{w-1}}$, we can take either

$$a \leftarrow \ell + ((n - \ell) \bmod 2^{w-1}), \quad \text{or} \tag{3}$$

$$a \leftarrow u - ((u - n) \bmod 2^{w-1}). \tag{4}$$

Since $0 \leq x \bmod 2^{w-1} < 2^{w-1}$ for any $x \in \mathbb{Z}$, it is easily seen that both assignments yield a digit in $D_{\ell,u}$. Moreover, for the first assignment we have $a \in \mathsf{lower}(D_{\ell,u})$, and for the second we have $a \in \mathsf{upper}(D_{\ell,u})$.

# 3 The Algorithm

There are two main tasks ahead of us:

1. give an algorithm which builds minimal weight radix-2 joint representations where digits are restricted to the set $D_{\ell,u}$.
2. prove that the outputs of this algorithm do in fact have minimal weight.

In this section, we concentrate on task 1. Our strategy, which may initially seem misguided, will be to develop an algorithm which builds *colexicographically* minimal joint representations. We will see later on that colexicographically minimal representations and minimal weight representations have a number of common properties. This fact will hopefully postpone any misgivings about our approach until, with the completion of task 2, they can be laid to rest completely.

## 3.1 Building Colexicographically Minimal Representations

Given $N \in \mathbb{Z}^{d\times 1}$ and a set of digits $D_{\ell,u}$ (if $\ell = 0$, we require all components of $N$ to be nonnegative), we will construct a joint representation, $(A_{s-1} \ldots A_1 A_0)_2$, of $N$ by setting the value of each column in turn from least- to most-significant (i.e., right-to-left). If we can correctly set the digits of the least significant column, then this leads to an algorithm of the following form:

```
$s \leftarrow 0$
**while** $N \neq \vec{0}$ **do**
        select digits from $D_{\ell,u}$ to form $A$, the least significant column of a representation of $N$
        $A_s \leftarrow A$
        $N \leftarrow (N - A)/2$
        $s \leftarrow s + 1$
**return** $A_{s-1} \ldots A_1 A_0$
```

We start with the column $A_0$. So that $A_{s-1} \ldots A_1 A_0$ has low colexicographic rank, we try to apply the following rule: *If possible, make $A_0$ a zero column; otherwise, choose the digits of $A_0$ so that the number of zero columns which follow $A_0$ is maximized.*

If $N = (A_{s-1} \ldots A_1 A_0)_2$, then $N \equiv A_0 \pmod 2$. Thus, a condition under which it is not possible to make $A_0$ a zero column is $N \not\equiv \vec{0} \pmod 2$. If this condition does not hold (i.e., if $N \equiv \vec{0} \pmod 2$), then we will set $A_0 \leftarrow \vec{0}$. But suppose it is the case that $N \not\equiv \vec{0} \pmod 2$. Since $D_{\ell,u}$ contains a complete system of residues modulo $2^{w-1}$, we can choose $A_0$ so that $N \equiv A_0 \pmod{2^{w-1}}$. Setting the digits of $A_0$ in this manner allows *at least* $w - 2$ zero columns to follow $A_0$. But, depending on the values of $u$, $\ell$ and $N$, there can be more than one possibility for $A_0$; our choice can influence the number of zero columns following $A_0$.

Using the expression in (3), we initially set $A_0 \leftarrow L + ((N - L) \bmod 2^{w-1})$ where $L = (\ell, \ell, \ldots, \ell)^{\mathsf{T}}$. Each digit of $A_0$ is either unique modulo $2^{w-1}$ (in $D_{\ell,u}$) or not. The next possible nonzero column will occur no sooner than $A_{w-1}$. By computing $M \leftarrow (N - A_0)/2^{w-1}$ and checking if $M \equiv \vec{0} \pmod 2$, we can determine if the initial value of $A_0$ causes $A_{w-1}$ to be nonzero. However, it is only the digits of $A_0$ which are unique modulo $2^{w-1}$ which determine whether or not $A_{w-1}$ must be nonzero. This is because a digit of $A_0 = (a_1, a_2, \ldots, a_d)^{\mathsf{T}}$ which is not unique can be replaced with $a_i + 2^{w-1}$, and this changes the parity of $m_i$ where $M = (m_1, m_2, \ldots, m_d)^{\mathsf{T}}$. These replacements can sometimes be used to make $A_{w-1}$ a zero column.

Using the sets $\mathsf{unique}(D_{\ell,u})$, $\mathsf{nonunique}(D_{\ell,u})$ introduced in (1) and (2), our observations so far on how to compute $A_0$ are incorporated in the method below:

```
**if** $N \equiv \vec{0} \pmod 2$ **then**
        $A \leftarrow \vec{0}$
**else**
```

```
    $A \leftarrow L + ((N - L) \bmod 2^{w-1})$
    $\mathcal{I}_{\text{unique}} \leftarrow \{i \in \{1, 2, \ldots, d\} : a_i \in \text{unique}(D_{\ell,u})\}$
    $\mathcal{I}_{\text{nonunique}} \leftarrow \{i \in \{1, 2, \ldots, d\} : a_i \in \text{nonunique}(D_{\ell,u})\}$
    $M \leftarrow (N - A)/2^{w-1}$
    if $m_i \equiv 0 \pmod 2$ for all $i \in \mathcal{I}_{\text{unique}}$ then
        for $i \in \mathcal{I}_{\text{nonunique}}$ such that $m_i \equiv 1 \pmod 2$ do
            $a_i \leftarrow a_i + 2^{w-1}$
            $m_i \leftarrow m_i - 1$
$A_0 \leftarrow A$
```

However, as the following example shows, we are not done yet.

**Example 3.1.** Consider the vector $N = (3, 5)^{\mathsf{T}}$ and the digit set $D_{-3,1} = \{-3, -2, -1, 0, 1\}$. We have unique$(D_{-3,1}) = \{-2, -1, 0\}$ and nonunique$(D_{-3,1}) = \{-3, 1\}$. By iterating the method above, we obtain the following representation:

$$\begin{pmatrix} 10\bar{3}0\bar{1} \\ 10\bar{2}0\bar{3} \end{pmatrix}_2 = \begin{pmatrix} 3 \\ 5 \end{pmatrix}.$$

However, this representation is not colexicographically minimal because

$$\begin{pmatrix} 10\bar{1} \\ 101 \end{pmatrix}_2 = \begin{pmatrix} 3 \\ 5 \end{pmatrix}.$$

has lower colexicographic rank. Our method sets the least significant column to $(-1, -3)^{\mathsf{T}}$. The digit 1 could be used in place of $-3$, but our method does not recognize the advantage of doing so. ♢

If it is true that both $A_0$ and $A_{w-1}$ must be nonzero columns (i.e., if $N \not\equiv \vec{0} \pmod 2$ and $m_i \equiv 1 \pmod 2$ for some $i \in \mathcal{I}_{\text{unique}}$), then our current method does not make any changes to the initial value of $A_0$. But there is another reason to change the initial value of $A_0$, aside from making $A_{w-1}$ a zero column. Doing so may result in more choices for the digits of column $A_{w-1}$; this in turn may allow us to prevent a nonzero column when choosing digits for $A_{2w-2}$.

The vector $M$ above determines which digits can be used in column $A_{w-1}$. By computing $\ell + ((m_i - \ell) \bmod 2^{w-1})$ and checking if this digit is in unique$(D_{\ell,u})$ or nonunique$(D_{\ell,u})$, we can determine whether or not we have a choice for the digit at coordinate $i$ of $A_{w-1}$. By replacing $m_i$ with $m_i - 1$ (after updating $A_0$), it is sometimes possible to move coordinate $i$ of $A_{w-1}$ from unique$(D_{\ell,u})$ into nonunique$(D_{\ell,u})$. From (1), we see that the minimum digit of unique$(D_{\ell,u})$ is $u - 2^{w-1} + 1$. It is easily seen that if nonunique$(D_{\ell,u})$ is nonempty, then the following implication holds for $m_i$:

$$\begin{aligned} &\ell + ((m_i - \ell) \bmod 2^{w-1}) = u - 2^{w-1} + 1 \in \text{unique}(D_{\ell,u}) \\ \Longrightarrow\ &\ell + ((m_i - 1 - \ell) \bmod 2^{w-1}) = u - 2^{w-1} \in \text{nonunique}(D_{\ell,u}). \end{aligned}$$

We will test for the condition $\ell + ((m_i - \ell) \bmod 2^{w-1}) = u - 2^{w-1} + 1$ and make the changes necessary to allow two choices for digit $i$ of $A_{w-1}$.[4] We do this by adding an "else" clause to the second "if" statement in the previous pseudocode listing. This is the only condition under which replacing $m_i$ with $m_i - 1$ moves us from unique$(D_{\ell,u})$ into nonunique$(D_{\ell,u})$.

Here is the modified "if" statement:

```
if $m_i \equiv 0 \pmod 2$ for all $i \in \mathcal{I}_{\text{unique}}$ then
    for $i \in \mathcal{I}_{\text{nonunique}}$ such that $m_i \equiv 1 \pmod 2$ do
```

[4] The test condition can be simplified (e.g., it is equivalent to $m_i \equiv u + 1 \pmod{2^{w-1}}$ ), but, for the sake of clarity, we leave it as is.

```
            $a_i \leftarrow a_i + 2^{w-1}$
            $m_i \leftarrow m_i - 1$
    else
        for $i \in \mathcal{I}_{\mathsf{nonunique}}$ such that $\ell + ((m_i - \ell) \mod 2^{w-1}) = u - 2^{w-1} + 1$ do
            $a_i \leftarrow a_i + 2^{w-1}$
            $m_i \leftarrow m_i - 1$
```

This change completes our algorithm.

**Example 3.2.** Repeating Example 3.1 with our modified pseudocode results in the following representation:

$$\begin{pmatrix} 10\bar{1} \\ 101 \end{pmatrix}_2 = \begin{pmatrix} 3 \\ 5 \end{pmatrix}.$$

It is easily seen that this representation is colexicographically minimal. ◊

## 3.2 The Final Algorithm

Our final algorithm is listed as Algorithm 3. The claim made in the caption there (i.e., that the outputs are colexicographically minimal and have minimal weight) will be justified later on. Here, we show only that Algorithm 3 terminates for all its inputs (i.e., we show that it really is an algorithm).

**Lemma 3.3.** *For any valid input $N \in \mathbb{Z}^{d\times 1}$, Algorithm 3 terminates.*

By "valid input", we mean that if $\ell = 0$, then all components of $N$ must be nonnegative; if $\ell < 0$, then any $N \in \mathbb{Z}^{d\times 1}$ is valid.

*Proof.* We first consider the case that $\ell < 0$ (i.e., we also have negative digits). Then we obviously have $\max\{|u|, |\ell|\} \le u - \ell - 1 < 2^w - 2$. We note that if $A \neq \vec{0}$ in some step of the algorithm, we have $N \equiv A \pmod{2^{w-1}}$. This implies that in the subsequent $w - 2$ steps of the algorithm, we will have $A = \vec{0}$. We temporarily call these $w - 2$ steps "insignificant steps" as opposed to the other steps, which we call "significant steps".

We claim that $\|N\|_\infty$ strictly decreases from one significant step to the next significant step. Here, $\|N\|_\infty$ denotes the infinity norm of $N$, i.e., $\max_i\{|n_i|\}$. If $A = \vec{0}$ in some step of the algorithm, it is clear that $\|(N - A)/2\|_\infty = \|N/2\|_\infty < \|N\|_\infty$. If $A \neq \vec{0}$, we have to consider the next significant step, i.e., the next number will be $(N - A)/2^{w-1}$. If $\|N\|_\infty \ge 2$, we have

$$\|(N - A)/2^{w-1}\|_\infty \le (\|N\|_\infty + \max\{|u|, |\ell|\})/2^{w-1} < (\|N\|_\infty + 2(2^{w-1} - 1))/2^{w-1} \le \|N\|_\infty,$$

as claimed. We still have to consider the case that $\|N\|_\infty = 1$. The algorithm will choose $A = N$ in this case, since all entries of $N$ belong to the digit set. Thus the algorithm terminates in this case.

We now turn to the case $\ell = 0$. Here, we have to show that during the execution of the algorithm, no component of $N$ ever becomes negative. This could only happen if $n_i - u < 0$ for some $i$. This means that $n_i$ itself is a digit. The situation can only be dangerous if $n_i + 2^{w-1}$ is also a digit, thus $n_i \in \mathsf{nonunique}(D_{0,u})$ with $n_i \in \mathsf{lower}(D_{0,u})$. Thus, at line 16, the quantity $m_i$ equals 0. But $n_i + 2^{w-1}$ will neither be taken to make $m_i$ even (at line 19) nor will it happen that $0 = m_i \mod 2^{w-1} = u - 2^{w-1} + 1$ (at line 25) because $u \ge 2^{w-1}$ (since we have a digit in $\mathsf{nonunique}(D_{0,u})$). Thus $a_i = n_i$ in this case.

This means that in the case $\ell = 0$, all intermediate numbers $N$ will be nonnegative and the components of $N$ will strictly decrease until they reach 0, where they remain. □

When Algorithm 3 terminates for an input $N \in \mathbb{Z}^{d\times 1}$, from line 30 it is clear that the columns returned form a joint representation of $N$.

**Algorithm 3** Computation of a colexicographically minimal & minimal weight joint representation

**Input:** $N = (n_1, n_2, \ldots, n_d)^\mathsf{T} \in \mathbb{Z}^{d\times 1}$, $\ell \leq 0$, $u \geq 1$ (with all components of $N$ nonnegative if $\ell = 0$).
**Output:** $A_{s-1} \ldots A_1 A_0$, a colexicographically minimal & minimal weight representation of $N$.

```
$D_{\ell,u} \leftarrow \{a \in \mathbb{Z} : \ell \leq a \leq u\}$
$w \leftarrow$ the integer such that $2^{w-1} \leq \#D_{\ell,u} < 2^w$
unique$(D_{\ell,u}) \leftarrow \{a \in D_{\ell,u} : u - 2^{w-1} < a < \ell + 2^{w-1}\}$
nonunique$(D_{\ell,u}) \leftarrow \{a \in D_{\ell,u} : a \leq u - 2^{w-1}$ or $\ell + 2^{w-1} \leq a\}$
{these sets respectively consist of the digits which are unique and nonunique modulo $2^{w-1}$.}
$s \leftarrow 0,\ L \leftarrow (\ell, \ell, \ldots, \ell)^\mathsf{T}$
while $N \neq \vec{0}$ do
    if $N \equiv \vec{0} \pmod 2$ then
        {We can make column $s$ zero, so we do this.}
        $A \leftarrow \vec{0}$
    else
        {We cannot make column $s$ zero, thus it will be nonzero.}
        $A \leftarrow L + ((N - L) \bmod 2^{w-1})$
        $\mathcal{I}_{\text{unique}} \leftarrow \{i \in \{1, 2, \ldots, d\} : a_i \in$ unique$(D_{\ell,u})\}$
        $\mathcal{I}_{\text{nonunique}} \leftarrow \{i \in \{1, 2, \ldots, d\} : a_i \in$ nonunique$(D_{\ell,u})\}$
        $M \leftarrow (N - A)/2^{w-1}$
        if $m_i \equiv 0 \pmod 2$ for all $i \in \mathcal{I}_{\text{unique}}$ then
            {We can make column $s + w - 1$ zero.}
            for $i \in \mathcal{I}_{\text{nonunique}}$ such that $m_i$ is odd do
                $a_i \leftarrow a_i + 2^{w-1}$
                $m_i \leftarrow m_i - 1$
        else
            {Column $s + w - 1$ will be nonzero.}
            {Use redundancy at column $s$ to increase redundancy at column $s+w-1$.}
            for $i \in \mathcal{I}_{\text{nonunique}}$ such that $\ell + ((m_i - \ell) \bmod 2^{w-1}) = u - 2^{w-1} + 1$ do
                $a_i \leftarrow a_i + 2^{w-1}$
                $m_i \leftarrow m_i - 1$
        {We have $N \equiv A \pmod{2^{w-1}}$ and $M = (N - A)/2^{w-1}$.}
    $A_s \leftarrow A$
    $N \leftarrow (N - A)/2$
    $s \leftarrow s + 1$
return $A_{s-1} \ldots A_1 A_0$
```

## 4 Common Properties

Here, we describe some common properties of colexicographically minimal representations and minimal weight representations. The properties are presented as a sequence of lemmas. We first show that both kinds of representations are recursive.

**Lemma 4.1.** *If $(\ldots A_2 A_1 A_0)_2$ is a colexicographically minimal representation of a vector $N \in \mathbb{Z}^{d\times 1}$, then $(\ldots A_2 A_1)_2$ is a colexicographically minimal representation of $(N - A_0)/2$.*

*Proof.* Let $\mathcal{A} = \ldots A_2 A_1 A_0$ and $\mathcal{A}' = \ldots A_2 A_1$. Let $\mathcal{B}'$ be the columns of a colexicographically minimal representation of $(N - A_0)/2$. Suppose $\mathsf{char}(\mathcal{B}')$ is strictly less than $\mathsf{char}(\mathcal{A}')$, colexicographically. Then

$$\begin{aligned} & \mathcal{B}' \prec \mathcal{A}' \\ \Longrightarrow\ & \mathcal{B}'A_0 \prec \mathcal{A}'A_0 \\ \Longrightarrow\ & \mathcal{B}'A_0 \prec \mathcal{A}. \end{aligned}$$

Since $(\mathcal{B}'A_0)_2 = N$, we see that $\mathcal{A}$ is not a colexicographically minimal representation of $N$. □

The same recursive property is true for minimal weight representations.

**Lemma 4.2.** *If $(\ldots A_2A_1A_0)_2$ is a minimal weight representation of a vector $N \in \mathbb{Z}^{d\times 1}$, then $(\ldots A_2A_1)_2$ is a minimal weight representation of $(N - A_0)/2$.*

*Proof.* Let $\mathcal{A} = \ldots A_2A_1A_0$ and $\mathcal{A}' = \ldots A_2A_1$. Let $\mathcal{B}'$ be the columns of a representation of $(N - A_0)/2$ that has fewer nonzero columns than $\mathcal{A}'$. Then

$$\begin{aligned} & \mathsf{wt}(\mathcal{B}') < \mathsf{wt}(\mathcal{A}') \\ \Longrightarrow\ & \mathsf{wt}(\mathcal{B}'A_0) < \mathsf{wt}(\mathcal{A}'A_0) \\ \Longrightarrow\ & \mathsf{wt}(\mathcal{B}'A_0) < \mathsf{wt}(\mathcal{A}). \end{aligned}$$

Since $(\mathcal{B}'A_0)_2 = N$, we see that $\mathcal{A}$ is not a minimal weight representation of $N$. □

Notice that the above lemmas are true for *any* digit set $D \subset \mathbb{Z}$. For the digit set $D_{\ell,u}$, other commonalities can be demonstrated. Before we get to those, we establish two short facts.

**Fact 4.3.** *For any representation $(a_{w-2}\ldots a_1a_0)_2$ where each $a_j \in D_{\ell,u}$, the Diophantine equation*

$$(a_{w-2}\ldots a_1a_0)_2 = x \cdot 2^{w-1} + y \tag{5}$$

*has a solution, $(x, y)$, with $x, y \in D_{\ell,u}$.*

*Proof.* Observe that all integers in the range

$$\ell \cdot 2^{w-1} + \ell, \ldots, u \cdot 2^{w-1} + u$$

can be expressed as $x \cdot 2^{w-1} + y$ with $x, y \in D_{\ell,u}$. Now, since $\ell \leq 0$ and $u \geq 1$, we have

$$\ell \cdot 2^{w-1} + \ell \leq \ell \cdot 2^{w-1} - \ell \leq (a_{w-2}\ldots a_1a_0)_2 \leq u \cdot 2^{w-1} - u \leq u \cdot 2^{w-1} + u,$$

and so $(a_{w-2}\ldots a_1a_0)_2$ is an integer in this range. □

**Fact 4.4.** *For any representation $(0a_{s-1}\ldots a_1a_0)_2$, with each $a_j \in D_{\ell,u}$, and integer $a$, with $a \in D_{\ell,u}$, there exists a representation $(b_sb_{s-1}\ldots b_1b_0)_2$, with each $b_j \in D_{\ell,u}$, such that*

$$(b_sb_{s-1}\ldots b_1b_0)_2 = (0a_{s-1}\ldots a_1a_0)_2 + a,$$

*and*

$$\mathsf{wt}(b_sb_{s-1}\ldots b_1b_0) \leq \mathsf{wt}(0a_{s-1}\ldots a_1a_0) + 1.$$

*Proof.* Use the classical addition algorithm to add $a$ to $(0a_{s-1}\ldots a_1a_0)_2$. This may trigger a carry, however, carries stop when reaching the first zero column from the right, at the latest. Therefore, the Hamming weight will be increased by at most one. □

Fact 4.4 extends to joint representations as well; that is, we can carry out the addition $(0A_{s-1}\ldots A_1A_0)_2+A$ with only increasing the number of nonzero columns by at most one. We use this to establish additional common properties.

**Lemma 4.5.** *If $(\ldots A_2A_1A_0)_2$ is a colexicographically minimal representation of a vector $N\in\mathbb{Z}^{d\times 1}$ with digits restricted to $D_{\ell,u}$, then every nonzero column of $(\ldots A_2A_1A_0)_2$ must contain an odd digit.*

*Proof.* Let $\mathcal{A}=\ldots A_2A_1A_0$ and suppose $\mathcal{A}$ contains a nonzero column consisting of only even digits. By Lemma 4.1, we can assume that $A_0$ is such a nonzero column. Let $A_t$ be the first zero column that precedes $A_0$. Then we have

$$\mathcal{A}=\ldots\vec{0}\,\underbrace{A_{t-1}\ldots A_1A_0}_{\text{nonzero columns}}.$$

By Fact 4.4, we can replace the columns $\vec{0}A_{t-1}\ldots A_1A_0$ with $B_tB_{t-1}\ldots B_1\vec{0}$ where

$$(B_tB_{t-1}\ldots B_1)_2=(\vec{0}A_{t-1}\ldots A_1)_2+A_0/2.$$

Thus, we have

$$N=(\ldots A_{t+2}A_{t+1}\vec{0}A_{t-1}\ldots A_1A_0)_2=(\ldots A_{t+2}A_{t+1}B_tB_{t-1}\ldots B_1\vec{0})_2;$$

but this new representation contradicts the fact that $(\mathcal{A})_2$ is colexicographically minimal (it has lower colexicographic rank because $A_0\neq\vec{0}$). □

**Lemma 4.6.** *Every vector $N\in\mathbb{Z}^{d\times 1}$ (with $N\geq\vec{0}$ if $\ell=0$) has a minimal weight representation with digits restricted to $D_{\ell,u}$ where each nonzero column contains an odd digit.*

*Proof.* Suppose $(\ldots A_2A_1A_0)_2$ is a minimal weight representation of $N$ that has a nonzero column consisting of only even digits. Let $\mathcal{A}=\ldots A_2A_1A_0$. By Lemma 4.2, we can assume that $A_0$ is such a nonzero column. We describe how to modify $\mathcal{A}$ so that $A_0$ becomes a zero column while the joint weight does not change.

Let $A_t$ be the first zero column that precedes $A_0$. Then we have

$$\mathcal{A}=\ldots\vec{0}\,\underbrace{A_{t-1}\ldots A_1A_0}_{\text{nonzero columns}}.$$

By Fact 4.4, we can replace the columns $\vec{0}A_{t-1}\ldots A_1A_0$ with $B_tB_{t-1}\ldots B_1\vec{0}$ where

$$(B_tB_{t-1}\ldots B_1)_2=(\vec{0}A_{t-1}\ldots A_1)_2+A_0/2,$$

and

$$\begin{aligned}&\mathsf{wt}(B_tB_{t-1}\ldots B_1)\leq\mathsf{wt}(\vec{0}A_{t-1}\ldots A_1)+1\\ \Longrightarrow\ &\mathsf{wt}(B_tB_{t-1}\ldots B_1\vec{0})\leq\mathsf{wt}(\vec{0}A_{t-1}\ldots A_1A_0).\end{aligned}$$

Because of the bound above, this replacement cannot increase the number of nonzero columns; thus, we maintain the minimal weight property. □

**Example 4.7.** A dimension-$d$ joint representation with $d=1$ is just a radix-2 representation of an integer. Suppose we want to build minimal weight integer representations using the digit set

$$D_{-7,7}=\{0,\pm1,\pm2,\pm3,\pm4,\pm5,\pm6,\pm7\}.$$

According to Lemma 4.6, if we restrict ourselves to the smaller digit set

$$\{0, \pm 1, \pm 3, \pm 5, \pm 7\}$$

we will still be able to construct minimal weight representations. This is because any minimal weight representation with digits in $D_{-7,7}$ can be transformed into one with digits in $\{0, \pm 1, \pm 3, \pm 5, \pm 7\}$ which has the same weight. This property underlies the construction of the width-$w$ nonadjacent forms. The digits $\{0, \pm 1, \pm 3, \pm 5, \pm 7\}$ are the width-4 nonadjacent form digits. Note however, that all this is only true in dimension 1. $\lozenge$

**Lemma 4.8.** *Let $(\dots A_2 A_1 A_0)_2$ be a colexicographically minimal representation of a vector $N \in \mathbb{Z}^{d\times 1}$ with digits restricted to $D_{\ell,u}$. If $A_j \neq \vec{0}$, then $A_{j+w-2} = \cdots = A_{j+1} = \vec{0}$ (i.e., immediately preceding any nonzero column, there must be at least $w-2$ zero columns).*

*Proof.* Suppose the result is false. Let $\mathcal{A} = \dots A_2 A_1 A_0$. By Lemma 4.2, we can assume that $A_0$ is nonzero and one of $A_{w-2} \dots A_1$ is also nonzero. By Fact 4.3, there exist $X, Y \in {D_{\ell,u}}^{d\times 1}$ such that

$$(A_{w-2} \dots A_1 A_0)_2 = X \cdot 2^{w-1} + Y.$$

Now we have

$$\begin{aligned} N &= (\dots A_w A_{w-1} A_{w-2} \dots A_1 A_0)_2 \\ &= (\dots A_w A_{w-1})_2 \cdot 2^{w-1} + (A_{w-2} \dots A_1 A_0)_2 \\ &= (\dots A_w A_{w-1})_2 \cdot 2^{w-1} + X \cdot 2^{w-1} + Y \\ &= \big((\dots A_w A_{w-1})_2 + X\big) \cdot 2^{w-1} + Y \\ &= (\dots B_w B_{w-1} \vec{0} \dots \vec{0} Y)_2. \end{aligned}$$

Note that the addition $(\dots A_w A_{w-1})_2 + X$ is carried out using Fact 4.4. But this new representation of $N$ contradicts the fact that $(\dots A_2 A_1 A_0)_2$ is colexicographically minimal because it has lower colexicographic rank. □

**Lemma 4.9.** *Every vector $N \in \mathbb{Z}^{d\times 1}$ (with $N \geq \vec{0}$ if $\ell = 0$) has a minimal weight representation with digits restricted to $D_{\ell,u}$ where each nonzero column is immediately preceded by at least $w-2$ zero columns.*

*Proof.* The statement can be proved in essentially the same way as Lemma 4.8. The only difference is that we use the weight bound of Fact 4.4 to conclude that the newly constructed representation also has minimal weight. □

In the next section, we show that there is an even stronger connection between colexicographically minimal representations and minimal weight representation with digits restricted to $D_{\ell,u}$: any colexicographically minimal representation is a minimal weight representation.

## 5 Minimality

We now return to the second task described in §3: proving that the outputs of Algorithm 3 have minimal weight. To accomplish this, we apply some of the results in §4. We also show that the outputs of the algorithm are colexicographically minimal.

**Theorem 1.** *For any input $N \in \mathbb{Z}^{d\times 1}$ (with $N \geq \vec{0}$ if $\ell = 0$), the representation constructed by Algorithm 3 has minimal weight.*

*Proof.* Suppose the result is false. Then Algorithm 3 produces a representation which does not have minimal weight for some input $N \in \mathbb{Z}^{d\times 1}$. Of all such $N$, choose one such that the number of columns returned by Algorithm 3 is minimal.

Let $\mathcal{A} = A_{s-1} \dots A_1 A_0$ be the columns returned by Algorithm 3 and let $\mathcal{B} = B_{t-1} \dots B_1 B_0$ be the columns of a minimal weight representation of $N$. We have

$$(\mathcal{A})_2 = N = (\mathcal{B})_2, \quad \text{and} \quad \mathsf{wt}(\mathcal{A}) > \mathsf{wt}(\mathcal{B})\,.$$

By Lemma 4.6 and Lemma 4.9, we can choose $\mathcal{B}$ so that each nonzero column contains an odd digit and each nonzero column is preceded by at least $w-2$ zero columns. We will show how to construct a new input, $N'$, for which the output of Algorithm 3 is shorter and does not have minimal weight. This contradicts our choice of $N$ and thereby establishes the result.

Suppose $A_0 = B_0$. The output of Algorithm 3 on input $N' = (N - A_0)/2$ is $A_{s-1} \dots A_1$. Now, $(A_{s-1} \dots A_1)_2 = N' = (B_{t-1} \dots B_1)_2$ and $\mathsf{wt}(A_{s-1} \dots A_1) > \mathsf{wt}(B_{t-1} \dots B_1)$. But this contradicts our choice of $N$. Thus it must be that $A_0 \neq B_0$.

If $N \equiv \vec{0} \pmod 2$, then both $A_0 = \vec{0}$ (by the definition of Algorithm 3) and $B_0 = \vec{0}$ (because each nonzero column of $\mathcal{B}$ contains an odd digit). But $A_0 \neq B_0$, so it must be that $N \not\equiv \vec{0} \pmod 2$. Hence both $A_0$ and $B_0$ are nonzero columns.

So far, we have the following picture:

$$\mathcal{A} = A_{s-1} \dots A_w A_{w-1} \overbrace{A_{w-2} \dots A_1}^{\text{zero columns}} A_0, \qquad \mathcal{B} = B_{t-1} \dots B_w B_{w-1} \overbrace{B_{w-2} \dots B_1}^{\text{zero columns}} B_0.$$

Thus, $A_0 \equiv B_0 \pmod{2^{w-1}}$ but $A_0 \neq B_0$. However, if $\#D_{\ell,u} = 2^{w-1}$, then $A_0 \equiv B_0 \pmod{2^{w-1}}$ implies $A_0 = B_0$. So for the special case $\#D_{\ell,u} = 2^{w-1}$, we are done. We will continue under the assumption that $\#D_{\ell,u} \neq 2^{w-1}$ (this implies that $\mathsf{nonunique}(D_{\ell,u})$ is nonempty).

We denote the digits of columns $w-1$ and $0$ of $\mathcal{A}$ and $\mathcal{B}$ like so:

$$A_{w-1} = \begin{matrix} a_{1(w-1)} \\ a_{2(w-1)} \\ \vdots \\ a_{d(w-1)} \end{matrix}, \quad A_0 = \begin{matrix} a_{10} \\ a_{20} \\ \vdots \\ a_{d0} \end{matrix}, \quad B_{w-1} = \begin{matrix} b_{1(w-1)} \\ b_{2(w-1)} \\ \vdots \\ b_{d(w-1)} \end{matrix}, \quad B_0 = \begin{matrix} b_{10} \\ b_{20} \\ \vdots \\ b_{d0} \end{matrix}.$$

We argue that both $A_{w-1}$ and $B_{w-1}$ must be nonzero columns.

If $B_{w-1} = \vec{0}$, then we have that $N \equiv B_0 \pmod{2^w}$. But then Algorithm 3 would have set $A_0 = B_0$. So it must be that $B_{w-1} \neq \vec{0}$. This implies that columns $B_{2w-3} \dots B_w$ are all zero.

Suppose that $A_{w-1} = \vec{0}$. We have $B_{w-1}2^{w-1} + B_0 \equiv A_0 \pmod{2^w}$, which implies that $B'_w := B_{w-1}/2 + (B_0 - A_0)/2^w$ is an integer vector. Observe that every entry $b'_{iw}$ of $B'_w$ is an element of $D_{\ell,u}$, since

$$\ell - 1 \leq \ell/2 - 1 < b_{i(w-1)}/2 + (b_{i0} - a_{i0})/2^w < u/2 + 1 \leq u;$$

note that the strict inequalities follow because $b_{i0} - a_{i0} = \pm 2^{w-1}$. We set $B'_{w-1} = \vec{0}$, $B'_0 = A_0$, and $B'_j := B_j$ for $j \notin \{0, w-1, w\}$. For $\mathcal{B}' := \dots B'_w B'_{w-1} \dots B'_0$ we have that $(\mathcal{B}')_2 = (\mathcal{B})_2 = N$. Thus we can replace $\mathcal{B}$ by $\mathcal{B}'$ and proceed as above. Hence, we can assume that $A_{w-1} \neq \vec{0}$.

Consider the vector $(N - A_0)/2^{w-1}$. We have

$$\begin{aligned} \frac{N - A_0}{2^{w-1}} &= (B_{t-1} \dots B_{2w-2} \overbrace{B_{2w-3} \dots B_w}^{\text{zero columns}} B_{w-1})_2 + \frac{B_0 - A_0}{2^{w-1}} \\ &= (A_{s-1} \dots A_{2w-2} \underbrace{A_{2w-3} \dots A_w}_{\text{zero columns}} A_{w-1})_2. \end{aligned}$$

Each coordinate of the vector $(B_0 - A_0)/2^{w-1}$ is in $\{0, \pm 1\}$. We will show that we can perform the addition $(B_{t-1} \dots B_{w-1})_2 + \frac{B_0 - A_0}{2^{w-1}}$ by changing only column $B_{w-1}$ (thus no new nonzero columns are created).

Consider $b_{i(w-1)}$, the $i$th coordinate of $B_{w-1}$. We must either add 1 to $b_{i(w-1)}$, subtract 1 from $b_{i(w-1)}$ or leave $b_{i(w-1)}$ as it is. The only difficulty in computing $b_{i(w-1)} + 1$ occurs when $b_{i(w-1)} = u$ (because $u + 1 \notin D_{\ell,u}$). In this case, we have $i \in \mathcal{I}_{\mathsf{nonunique}}$ since $a_{i0} \neq b_{i0}$ and

$$\begin{aligned} b_{i(w-1)} + 1 &\equiv a_{i(w-1)} \pmod{2^{w-1}} \\ \implies u + 1 &\equiv a_{i(w-1)} \pmod{2^{w-1}} \\ \implies a_{i(w-1)} &= u - 2^{w-1} + 1 . \end{aligned}$$

But this cannot happen: At line 25 of Algorithm 3, the value of $a_{i0}$ was set to avoid $a_{i(w-1)} = u - 2^{w-1} + 1$ (i.e., Algorithm 3 would have set $a_{i0} = b_{i0}$). So the problem of computing $u + 1$ does not occur.

The only difficulty in computing $b_{i(w-1)} - 1$ occurs when $b_{i(w-1)} = \ell$ (because $\ell - 1 \notin D_{\ell,u}$). In this case, we have

$$\begin{aligned} b_{i(w-1)} - 1 &\equiv a_{i(w-1)} \pmod{2^{w-1}} \\ \implies \ell - 1 &\equiv a_{i(w-1)} \pmod{2^{w-1}} \\ \implies a_{i(w-1)} &= \ell + 2^{w-1} - 1 . \end{aligned}$$

However, because we are subtracting 1, it must be that $a_{i0} = b_{i0} + 2^{w-1}$; so we have $a_{i0} \in \mathsf{upper}(D_{\ell,u})$. But Algorithm 3, by default, selects digits from $\mathsf{lower}(D_{\ell,u})$; it would only do otherwise if it could make $A_{w-1}$ a zero column or increase the redundancy at digit $a_{i(w-1)}$. But neither of those conditions occur ($A_{w-1} \neq \vec{0}$ and $a_{i(w-1)} = \ell + 2^{w-1} - 1 \neq u - 2^{w-1}$). Thus, we never need to compute $\ell - 1$.

So we can carry out the addition by replacing $B_{w-1}$ with a new column $B'_{w-1}$ with digits from $D_{\ell,u}$. Let $N' = (N - A_0)/2^{w-1}$. We have

$$\begin{aligned} &(A_{s-1} \dots A_w A_{w-1})_2 = N' = (B_{t-1} \dots B_w B'_{w-1})_2 \text{ and} \\ &\mathsf{wt}(A_{s-1} \dots A_w A_{w-1}) > \mathsf{wt}\big(B_{t-1} \dots B_w B'_{w-1}\big) . \end{aligned} \tag{6}$$

But then $N'$ contradicts our choice of $N$ and we are done. □

**Theorem 2.** *For any input $N \in \mathbb{Z}^{d \times 1}$ (with $N \geq \vec{0}$ if $\ell = 0$), the representation constructed by Algorithm 3 is colexicographically minimal.*

*Proof.* This can be established using essentially the same proof as Theorem 1. The only difference comes at (6): we obtain the desired contradiction by noting that $A_{s-1} \dots A_w A_{w-1} \succ B_{t-1} \dots B_w B'_{w-1}$ (i.e., the newly constructed representation has lower colexicographic rank). □

**Corollary 3.** *Any colexicographically minimal representation of a vector $N \in \mathbb{Z}^{d \times 1}$ with digits restricted to $D_{\ell,u}$ has minimal weight.*

*Proof.* By Theorems 1 and 2, we know that at least one colexicographically minimal representation of $N$ has minimal weight. However, all colexicographically minimal representations of $N$ have the same number of nonzero columns, so the result follows. □

# 6 Analysis

We now analyze the weight of the outputs of Algorithm 3. We do so by defining a probability distribution on a set $\mathcal{M}$ of infinite sequences over ${D_{\ell,u}}^{d \times 1}$. Each sequence in $\mathcal{M}$ satisfies the same syntax as the outputs of Algorithm 3 (e.g., each nonzero column contains an odd digit). An explicit description of this syntax is presented in §6.1. By considering the number of nonzero columns out of the first $n$ columns of a sequence drawn from $\mathcal{M}$, we obtain a random variable, $W_n$. The expected value and the variance of $W_n$ are computed in §6.2.

## 6.1 Combinatorial Characterization

Here we provide a precise combinatorial description of the outputs of Algorithm 3.

**Theorem 4.** *Let $N \in \mathbb{Z}^{d\times 1}$ (with $N \geq \vec{0}$ if $\ell = 0$). Then there is exactly one representation $(A_{s-1} \dots A_1 A_0)_2$ (up to leading zeros) of $N$, such that the following conditions are satisfied:*

1. *Each column $A_j$ is zero or contains an odd digit.*
2. *If $A_j \neq \vec{0}$ for some $j$, then $A_{j+w-2} = \cdots = A_{j+1} = \vec{0}$.*
3. *If $A_j \neq \vec{0}$ and $A_{j+w-1} \neq \vec{0}$ for some $j$, then*
    - *(a) there is an $i \in \{1, \dots, d\}$ such that $a_{i(j+w-1)}$ is odd and $a_{ij} \in \mathsf{unique}(D_{\ell,u})$,*
    - *(b) if $a_{ij} \in \mathsf{nonunique}(D_{\ell,u})$, then $a_{i(j+w-1)} \not\equiv u+1 \pmod{2^{w-1}}$,*
    - *(c) if $a_{ij} \in \mathsf{upper}(D_{\ell,u}) \cap \mathsf{nonunique}(D_{\ell,u})$, then $a_{i(j+w-1)} \equiv u \pmod{2^{w-1}}$.*

*Furthermore, $A_{s-1} \dots A_1 A_0$ is the output of Algorithm 3 on input $N$.*

*Proof.* It is easily seen that if $A_{s-1} \dots A_1 A_0$ is the output of Algorithm 3 on input $N \in \mathbb{Z}^{d\times 1}$, then the conditions above are satisfied because of the decisions made in the Algorithm.

To complete the argument, assume now that $N \in \mathbb{Z}^{d\times 1}$ admits two different (not only up to leading zeros) representations $(\mathcal{A})_2$ and $(\mathcal{B})_2$ that satisfy the conditions stated in the Theorem. Without loss of generality, we choose the triple $(N, \mathcal{A}, \mathcal{B})$ so that the minimum of the lengths of $\mathcal{A}$ and $\mathcal{B}$ is minimum. This ensures that $A_0 \neq B_0$.

We have $N \equiv A_0 \equiv B_0 \pmod 2$. If $A_0 = \vec{0}$, then Condition 1 implies that $B_0 = \vec{0}$, which contradicts $A_0 \neq B_0$. Thus, $A_0$ and $B_0$ are both nonzero. By Condition 2, we conclude that $A_{w-2} = \cdots = A_1 = B_{w-2} = \cdots = B_1 = \vec{0}$. Therefore, we have $A_0 \equiv B_0 \pmod{2^{w-1}}$. This implies that for all indices $i$ such that $a_{i0} \in \mathsf{unique}(D_{\ell,u})$, we have $b_{i0} = a_{i0}$.

If $A_{w-1} = B_{w-1} = \vec{0}$, we have $A_0 \equiv B_0 \pmod{2^w}$, which contradicts $A_0 \neq B_0$. Consequently, we have $A_{w-1} \neq \vec{0}$, say. For the index $i$ described in Condition 3a, we have $b_{i0} = a_{i0}$ and therefore $b_{i(w-1)} \equiv a_{i(w-1)} \equiv 1 \pmod 2$. Thus we have $B_{w-1} \neq \vec{0}$, too. This yields $A_{2w-3} = \cdots = A_w = B_{2w-3} = \cdots = B_w = \vec{0}$ and therefore

$$B_{w-1} \equiv A_{w-1} + (A_0 - B_0)/2^{w-1} \pmod{2^{w-1}}. \tag{7}$$

Since $A_0 \neq B_0$, there is an index $i \in \mathsf{nonunique}(D_{\ell,u})$ such that $a_{i0} \in \mathsf{upper}(D_{\ell,u}) \cap \mathsf{nonunique}(D_{\ell,u})$ and $b_{i0} = a_{i0} - 2^{w-1}$ (we swap $\mathcal{A}$ and $\mathcal{B}$, if necessary). From Condition 3c we conclude that $a_{i(w-1)} \equiv u \pmod{2^{w-1}}$, thus (7) yields $b_{i(w-1)} \equiv u + 1 \pmod{2^{w-1}}$. But this contradicts Condition 3b. □

Note that the well-known syntaxes of the nonadjacent form and width-$w$ nonadjacent form and the simple joint sparse form are special cases of the syntax above.

### 6.2 Weight

We set up a probabilistic model as follows. Consider the space

$$\mathcal{M} := \{\dots A_2 A_1 A_0 : \text{ for all } j \geq 0,\ A_j \in {D_{\ell,u}}^{d\times 1} \text{ and } A_{j-1}\dots A_1 A_0 \text{ satisfies the conditions of Thm. 4}\}.$$

The elements of $\mathcal{M}$ are infinite sequences over ${D_{\ell,u}}^{d\times 1}$. We define random variables $\dots X_2 X_1 X_0$ to be the corresponding columns of a sequence $\dots A_2 A_1 A_0$ drawn from $\mathcal{M}$. The probability measure, Pr, we utilize is defined by the following property: for any nonnegative integer $n$ and vector $A \in \{0,\dots,2^n-1\}^{d\times 1}$,

$$\Pr\left(\sum_{j=0}^{n-1} X_j 2^j \mod 2^n = A\right) = \frac{1}{2^{nd}}.$$

Note that this measure is simply the image of the Haar measure on the space of infinite sequences over $\{0,1\}^{d\times 1}$ (which can be identified with $d$-tuples of 2-adic integers) via the map given by Algorithm 3. In fact, Algorithm 3 describes a continuous map from the space of $d$-tuples of 2-adic integers to $\mathcal{M}$.

We are interested in the random variable

$$W_n := \sum_{j=0}^{n-1} [X_j \neq \vec{0}];$$

note that here, we have used Iverson's notation: [*expression*] equals 1 if *expression* is true and 0 otherwise. Thus, we see that $W_n$ equals the number of nonzero columns among the first $n$ columns of a sequence in $\mathcal{M}$. The $X_j$'s are not independent random variables; the value of $X_j$ is influenced by the value of some of $X_{j-1}\dots X_0$ (e.g., see Condition 2 of Theorem 4). We determine the expected value and variance of $W_n$ by carrying out an analysis similar to the one done by Grabner, Heuberger, Prodinger and Thuswaldner [5] which combines techniques from the analysis of Markov processes with generating functions.

We begin by deriving a number of transition probabilities. The following notation facilitates this:

**Definition 6.1.** For a vector $A = (a_1,\dots,a_d)^{\mathsf{T}} \in \mathbb{Z}^{d\times 1}$ and a set $R \subseteq \mathbb{Z}$, we define

$$\mathcal{I}_R(A) := \{i \in \{1,2,\dots,d\} : a_i \equiv r \pmod{2^{w-1}} \text{ for some } r \in R\}.$$

Observe that $\mathcal{I}_R(A)$ equals some subset of the index set, $\{1,2,\dots,d\}$, of $A$. We also define

$$\begin{aligned}\mathcal{I}_{\mathsf{odd}}(A) &:= \{i \in \{1,2,\dots,d\} : a_i \text{ is odd}\},\\ \mathcal{I}_{\mathsf{unique}}(A) &:= \mathcal{I}_{\mathsf{unique}(D_{\ell,u})}(A),\\ \mathcal{I}_{\mathsf{nonunique}}(A) &:= \mathcal{I}_{\mathsf{nonunique}(D_{\ell,u})}(A).\end{aligned}$$

Now, let $A$ be a vector from $\{0,\dots,2^{w-1}-1\}^{d\times 1}$ containing at least one odd element. We compute conditional probabilities for $X_{j+w-1}$ under the assumption that $X_j \mod 2^{w-1} = A$. We have

$$p_{AE} := \Pr(X_{j+w-1} = \vec{0} \mid X_j \mod 2^{w-1} = A) = \frac{1}{2^{\#\mathcal{I}_{\mathsf{unique}}(A)}}; \tag{8}$$

here, the index $E$ in $p_{AE}$ stands for "even". $p_{AE}$ is the probability that we transition from the state $X_j \mod 2^{w-1} = A$ to $X_{j+w-1} = \vec{0}$. Let $B$ also be a vector from $\{0,\dots,2^{w-1}-1\}^{d\times 1}$ containing at least one odd element. Assuming that $\mathcal{I}_{\mathsf{odd}}(B) \cap \mathcal{I}_{\mathsf{unique}}(A) \neq \emptyset$ and $\mathcal{I}_{\mathsf{nonunique}}(B) \cap \mathcal{I}_{\{u+1\}}(A) = \emptyset$ (so that Conditions 3a and 3b of Theorem 4 are satisfied), we have

$$p_{AB} := \Pr(X_{j+w-1} \mod 2^{w-1} = B \mid X_j \mod 2^{w-1} = A) = \frac{2^{\#(\mathcal{I}_{\{u\}}(B) \cap \mathcal{I}_{\mathsf{nonunique}}(A))}}{2^{(w-1)d}}. \tag{9}$$

This can be seen as follows. For each $i \in \mathcal{I}_{\{u\}}(B) \cap \mathcal{I}_{\textsf{nonunique}}(A)$, Algorithm 3 can make $x_{i(j+w-1)} \equiv u \pmod{2^{w-1}}$ in two ways: either lines 26–27 are executed, wherein $m_i \equiv u+1$ is changed so that $m_i \equiv u$, or they are not executed because $m_i \equiv u$ already. Now, if either $\mathcal{I}_{\textsf{odd}}(B) \cap \mathcal{I}_{\textsf{unique}}(A) \neq \emptyset$ or $\mathcal{I}_{\textsf{nonunique}}(B) \cap \mathcal{I}_{\{u+1\}}(A) = \emptyset$ does not hold, then $X_{j+w-1} \bmod 2^{w-1}$ is never equal to $B$; so we have

$$p_{AB} = 0.$$

Observe,

$$\begin{aligned} p_{EB} &:= \Pr(X_{j+1} \bmod 2^{w-1} = B \mid X_j = X_{j-1} = \cdots = X_{j-w+2} = \vec{0}) = \frac{1}{2^{(w-1)d}}, \\ p_{EE} &:= \Pr(X_{j+1} = \vec{0} \mid X_j = X_{j-1} = \cdots = X_{j-w+2} = \vec{0}) = \frac{1}{2^d}. \end{aligned} \tag{10}$$

We set $X_{-1} = \cdots = X_{-w+1} = \vec{0}$, so that (10) holds for all $j \geq -1$ ( $X_{-1} = \cdots = X_{-w+1} = \vec{0}$ can be viewed as an initial state). Finally, it is clear that

$$\Pr(X_{j+w-2} = X_{j+w-3} = \cdots = X_{j+1} = \vec{0} \mid X_j \bmod 2^{w-1} = A) = 1.$$

The transition probabilities calculated so far are more detailed than necessary (and useful). They can be aggregated into similar cases. For $0 \leq s \leq d$, we set

$$S_s := \{A \in \{0, \ldots, 2^{w-1}-1\}^{d \times 1} : \mathcal{I}_{\textsf{odd}}(A) \neq \emptyset \text{ and } \#\mathcal{I}_{\textsf{nonunique}}(A) = s\}.$$

Fix some $s$ and $A \in S_s$. We compute

$$p_{At} := \Pr(X_{j+w-1} \bmod 2^{w-1} \in S_t \mid X_j \bmod 2^{w-1} = A)$$

for $t = 0, \ldots, d$. We do this by considering the generating function

$$F_A(Z) := \sum_{t=0}^{d} p_{At} Z^t = \sum_{\substack{B \in \{0,\ldots,2^{w-1}-1\}^{d \times 1} \\ \mathcal{I}_{\textsf{odd}}(B) \neq \emptyset}} p_{AB} Z^{\#\mathcal{I}_{\textsf{nonunique}}(B)}.$$

One way to compute the coefficients of $F_A(Z)$ is to use (9) directly; however, it is less cumbersome to take a different route. Write $B = (b_1, \ldots, b_d)^{\mathsf{T}}$. For each $i \in \{1, \ldots, d\}$, we choose which of the four (disjoint) sets

$$\mathcal{I}_{\{u\}}(B),\ \ \mathcal{I}_{\textsf{nonunique}}(B) \setminus \mathcal{I}_{\{u\}}(B),\ \ \mathcal{I}_{\{u+1\}}(B),\ \ \mathcal{I}_{\textsf{unique}}(B) \setminus \mathcal{I}_{\{u+1\}}(B),$$

the index $i$ will belong to. Each choice has a certain number of values of $b_i$ associated to it, carries probabilities, and possibly contributions to the exponent of $Z$; the product of these quantities ($\Pi_{i=1}^{d} \ldots$) gives us $\sum_B p_{AB} Z^{\#\mathcal{I}_{\textsf{nonunique}}(B)}$ for a subset of values of $B$. If we omit the requirement that $\mathcal{I}_{\textsf{odd}}(B) \cap \mathcal{I}_{\textsf{unique}}(A) \neq \emptyset$ for the moment, then each factor in the product $\Pi_{i=1}^{d} \ldots$ can be extracted from the table below:

| | $i \in \mathcal{I}_{\{u\}}(B)$ | $i \in \mathcal{I}_{\textsf{nonunique}}(B)$ | $i \in \mathcal{I}_{\{u+1\}}(B)$ | $i \in \mathcal{I}_{\textsf{unique}}(B)$ |
|---|---|---|---|---|
| | | $i \notin \mathcal{I}_{\{u\}}(B)$ | | $i \notin \mathcal{I}_{\{u+1\}}(B)$ |
| $i \in \mathcal{I}_{\textsf{nonunique}}(A)$ | $1 \cdot \frac{2}{2^{w-1}} \cdot Z^1$ | $(\#\textsf{nonunique}/2 - 1) \cdot \frac{1}{2^{w-1}} \cdot Z^1$ | $1 \cdot 0 \cdot Z^0$ | $(\#\textsf{unique} - 1) \cdot \frac{1}{2^{w-1}} \cdot Z^0$ |
| $i \in \mathcal{I}_{\textsf{unique}}(A)$ | $1 \cdot \frac{1}{2^{w-1}} \cdot Z^1$ | $(\#\textsf{nonunique}/2 - 1) \cdot \frac{1}{2^{w-1}} \cdot Z^1$ | $1 \cdot \frac{1}{2^{w-1}} \cdot Z^0$ | $(\#\textsf{unique} - 1) \cdot \frac{1}{2^{w-1}} \cdot Z^0$ |

Note that #nonunique is short for $\#\textsf{nonunique}(D_{\ell,u})$, and similarly for #unique. All possible products are generated by the following polynomial:

$$2^{-(w-1)d}\,(\#\textsf{unique} - 1 + (\#\textsf{nonunique}/2 + 1)Z)^{\#\mathcal{I}_{\textsf{nonunique}}(A)}\,(\#\textsf{unique} + (\#\textsf{nonunique}/2)Z)^{\#\mathcal{I}_{\textsf{unique}}(A)}\,.$$

Recall that $A \in S_s$; thus we have $\#\mathcal{I}_{\mathsf{nonunique}}(A) = s$ and $\#\mathcal{I}_{\mathsf{unique}}(A) = d - s$.

In order to correct the error made by omitting the condition $\mathcal{I}_{\mathsf{odd}}(B) \cap \mathcal{I}_{\mathsf{unique}}(A) \neq \emptyset$, we have to subtract the contribution of those $B$ where all components with indices $i \in \mathcal{I}_{\mathsf{unique}}(A)$ are even. Let $\mathsf{even}(\mathsf{nonunique})$ be the set of even digits in $\mathsf{nonunique}(D_{\ell,u})$, and let $\mathsf{even}(\mathsf{unique})$ be similarly defined. Thus we obtain

$$\begin{aligned} F_A(Z) = 2^{-(w-1)d} \, (\#\mathsf{unique} - 1 + (\#\mathsf{nonunique}/2 + 1)Z)^s \cdot \\ \Big( (\#\mathsf{unique} + \#\mathsf{nonunique}/2 \cdot Z)^{d-s} - (\#\mathsf{even}(\mathsf{unique}) + \#\mathsf{even}(\mathsf{nonunique})/2 \cdot Z)^{d-s} \Big) . \end{aligned} \tag{11}$$

Note that $\#\mathsf{nonunique}/2$ and $\#\mathsf{even}(\mathsf{nonunique})/2$ are integers because $a \in \mathsf{nonunique}(D_{\ell,u})$ implies that either $a + 2^{w-1}$ or $a - 2^{w-1}$ is also in $\mathsf{nonunique}(D_{\ell,u})$. From (11), it is clear that $F_A(Z)$ (and hence the probabilities $p_{At}$) does not really depend on $A$, but only on $s = \#\mathcal{I}_{\mathsf{nonunique}}(A)$ (i.e., $F_A(Z)$ is the same for all $A \in S_s$). Thus, we define

$$F_s(Z) := F_A(Z) \ \text{ where } \ A \in S_s.$$

It follows that

$$p_{st} := \Pr(X_{j+w-1} \bmod 2^{w-1} \in S_t \mid X_j \bmod 2^{w-1} \in S_s) = \text{Coefficient of } Z^t \text{ in } F_s(Z).$$

We remark that setting $s = d$ in (11) yields $F_d(Z) = 0$ which coincides with Condition 3a of Theorem 4 (i.e., if $X_j \neq \vec{0}$ and each of its digits is in $\mathsf{nonunique}(D_{\ell,u})$, then $X_{j+w-1}$ must be a zero column).

**Example 6.2.** Consider $D_{-1,3} = \{-1, 0, 1, 2, 3\}$. For each $d \in \{1, 2, 3\}$ and $s \in \{0, \dots, d\}$, we give $F_s(Z)$.

$$\begin{array}{lll|lll|lll}
d = 1, & s = 0, & \frac{1}{4} + \frac{1}{4}Z & d = 2, & s = 0, & \frac{5}{16} + \frac{3}{8}Z + \frac{1}{16}Z^2 & d = 3, & s = 0, & \frac{19}{64} + \frac{27}{64}Z + \frac{9}{64}Z^2 + \frac{1}{64}Z^3 \\
 & s = 1, & 0 & & s = 1, & \frac{1}{8} + \frac{1}{4}Z + \frac{1}{8}Z^2 & & s = 1, & \frac{5}{32} + \frac{11}{32}Z + \frac{7}{32}Z^2 + \frac{1}{32}Z^3 \\
 & & & & s = 2, & 0 & & s = 2, & \frac{1}{16} + \frac{3}{16}Z + \frac{3}{16}Z^2 + \frac{1}{16}Z^3 \\
 & & & & & & & s = 3, & 0.
\end{array}$$

For this digit set, we have $w = 3$. Consider the case $d = 2$ and suppose $X_0 \neq \vec{0}$ and $\#\mathcal{I}_{\mathsf{nonunique}}(X_0) = 1$. Then, from the corresponding polynomial, we see that the probability that $X_2 \neq \vec{0}$ and $\#\mathcal{I}_{\mathsf{nonunique}}(X_2) = 0$ is $\frac{1}{8}$. Similarly, the probability that $X_2 \neq \vec{0}$ and $\#\mathcal{I}_{\mathsf{nonunique}}(X_2) = 1$ is $\frac{1}{4}$. Note that the probabilities $\frac{1}{8}, \frac{1}{4}, \frac{1}{8}$ do not sum to 1. This is because we have not accounted for the possibility that $X_2 = \vec{0}$. This must happen with probability $1 - F_1(1) = \frac{1}{2}$. ♢

The transition probability $p_{AE}$ computed in (8) also depends on $\#\mathcal{I}_{\mathsf{unique}}(A) = d - s$ only. Thus, we have

$$p_{sE} := \Pr(X_{j+w-1} = \vec{0} \mid X_j \bmod 2^{w-1} \in S_s) = \frac{1}{2^{d-s}}.$$

And the final transition probability we require is

$$p_{Et} := \Pr(X_{j+1} \bmod 2^{w-1} \in S_t \mid X_j = X_{j-1} = \cdots = X_{j-w+2} = \vec{0}) = \text{Coefficient of } Z^t \text{ in } F_E(Z),$$

where

$$F_E(Z) = 2^{-(w-1)d} \left( (\#\mathsf{unique} + \#\mathsf{nonunique}/2 \cdot Z)^d - (\#\mathsf{even}(\mathsf{unique}) + \#\mathsf{even}(\mathsf{nonunique})/2 \cdot Z)^d \right).$$

Now we are able to describe the distribution of the $X_j$'s using a $(d+2) \times (d+2)$ probability transition matrix

$$P := \begin{pmatrix} p_{EE} & p_{E0} & p_{E1} & \cdots & p_{Ed} \\ p_{0E} & p_{00} & p_{01} & \cdots & p_{0d} \\ \vdots & & & & \\ p_{dE} & p_{d0} & p_{d1} & \cdots & p_{dd} \end{pmatrix}$$

Note, however, that the step size (i.e., the number of columns output between states) is not constant: the transition probabilities in all but the first row of $P$ describe the state in $w-1$ steps (because all intermediate columns are $\vec{0}$), whereas the probabilities in the first row describe the state in the next step.

**Example 6.3.** Here are the transition matrices for the digits $D_{-1,3} = \{-1, 0, 1, 2, 3\}$ when $d \in \{1, 2, 3\}$:

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ 1 & 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} \frac{1}{4} & \frac{5}{16} & \frac{3}{8} & \frac{1}{16} \\ \frac{1}{4} & \frac{5}{16} & \frac{3}{8} & \frac{1}{16} \\ \frac{1}{2} & \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} \frac{1}{8} & \frac{19}{64} & \frac{27}{64} & \frac{9}{64} & \frac{1}{64} \\ \frac{1}{8} & \frac{19}{64} & \frac{27}{64} & \frac{9}{64} & \frac{1}{64} \\ \frac{1}{4} & \frac{5}{32} & \frac{11}{32} & \frac{7}{32} & \frac{1}{32} \\ \frac{1}{2} & \frac{1}{16} & \frac{3}{16} & \frac{3}{16} & \frac{1}{16} \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Notice that the probabilities in row sum to 1, as they should. Remark also that the first two rows in each matrix are identical; this is true in general since $F_E(Z) = F_0(Z)$ and $p_{EE} = p_{0E} = 1/2^d$. However, when we enter state $E$ or 0 from any other state, the number of columns output as a result is different when $w \neq 2$ (1 column is output in state $E$, and $w-1$ columns in state 0). Because of this difference, the two states cannot (in general) be identified. ◊

We describe the distribution of $W_n$ using a bivariate generating function:

$$G(Y, Z) := \sum_{n,k \geq 0} \Pr(W_n = k) Y^k Z^n.$$

Observe that

$$G(1, Z) = \frac{1}{1-Z} \qquad \text{and} \qquad \left. \frac{\partial G(Y, Z)}{\partial Y} \right|_{Y=1} = \sum_{n \geq 0} \mathrm{E}(W_n) Z^n.$$

To evaluate the coefficients of $G(Y, Z)$, we note that our sequence $X_{n-1} \dots X_0$ of random variables can be described by the regular expression

$$(\varepsilon + (\vec{0}^{(w-3)} + \vec{0}^{(w-4)} + \cdots + \vec{0}^{(1)} + \varepsilon)A)(\vec{0} + \vec{0}^{(w-2)}A)^*;$$

here, $A$ stands for a nonzero column, $\vec{0}^{(r)}$ for $r$ consecutive zero columns, $\varepsilon$ for the empty word, and $(\dots)^*$ for Kleene's star (finite repetition). The sequences generated by this expression will satisfy Conditions 1 and 2 of Theorem 4; of course, Condition 3 must also be satisfied, but this has already been taken into account in the derivation of the matrix $P$.

We use the regular expression (read right-to-left) to obtain an expression for $G(Y, Z)$. This gives us

$$G(Y, Z) = (p_{EE}\ p_{E0}\ p_{E1}\ \cdots\ p_{Ed}) \cdot \left(I - \mathrm{diag}(Z, YZ^{w-1}, \dots, YZ^{w-1})P\right)^{-1} \cdot \left((1\ 1\ \dots\ 1)^{\mathsf{T}} + YZ\frac{Z^{w-2}-1}{Z-1}(0\ 1\ \dots\ 1)^{\mathsf{T}}\right). \quad (12)$$

In the product above, each column of the sequence generated by the regular expression is marked with the variable $Z$, and nonzero columns are additionally marked with the variable $Y$. The dimensions of the vectors/matrices in the product are, from left-to-right, $1 \times (d+2)$, $(d+2) \times (d+2)$ and $(d+2) \times 1$.

**Example 6.4.** Using (12), we can give a general expression for $G(Y, Z)$ for any given $d$ in terms of $w$, #unique, #nonunique, #even(unique) and #even(nonunique). If desired, the latter four constants can be replaced with functions of $\ell, u, w$. For the case $d = 1$, we get

$$G(Y, Z) = \frac{2Z - YZ - 2 + YZ^{w-1}(Z\lambda - \lambda - Z + 2)}{(1-Z)\left(Z - 2 + YZ^{w-1}(Z\lambda - \lambda - Z + 2)\right)},$$

where $\lambda = \#\mathsf{odd}(D_{\ell,u})/2^{w-2}$; note that $1 \leq \lambda \leq 2$. To verify this expression, the fact that $\#\mathsf{odd}(D_{\ell,u}) = 1/4 \cdot \bigl(2(u-\ell+1) - (-1)^{\ell} - (-1)^{u}\bigr)$ can be utilized. ◊

**Example 6.5.** For $d = 2$, we give $G(Y, Z)$ for the special case when the digits $D_{-1,3} = \{-1, 0, 1, 2, 3\}$ are used:

$$\begin{aligned} G(Y,Z) &= \frac{Y^2Z^4 + 6\,YZ^2 + 24\,YZ + 32}{-Y^2Z^5 + Y^2Z^4 - 6\,YZ^3 - 18\,YZ^2 - 8\,Z + 32} \\ &= 1 + \left(\frac{1}{4} + \frac{3}{4}Y\right)Z + \left(\frac{1}{16} + \frac{15}{16}Y\right)Z^2 + \left(\frac{1}{64} + \frac{9}{16}Y + \frac{27}{64}Y^2\right)Z^3 + \cdots . \end{aligned}$$

The general expression for $G(Y, Z)$ when $d = 2$ is too long to display here. ◊

Write $G(Y, Z) = f(Y, Z)/g(Y, Z)$ where $f$ and $g$ are polynomials. Because $G(1, Z) = 1/(1 - Z)$, we have that $g(1, Z) = (1 - Z) \cdot f(1, Z)$. From the properties of derivatives, it follows that

$$\left.\frac{\partial G(Y,Z)}{\partial Y}\right|_{Y=1} = \frac{\left.\frac{\partial f(Y,Z)}{\partial Y}\right|_{Y=1}}{(1-Z)\cdot f(1,Z)} - \frac{\left.\frac{\partial g(Y,Z)}{\partial Y}\right|_{Y=1}}{(1-Z)^2\cdot f(1,Z)}.$$

Using a partial fraction expansion, we obtain

$$\left.\frac{\partial G(Y,Z)}{\partial Y}\right|_{Y=1} = \frac{e_{\ell,u,d}}{(1-Z)^2} + \frac{c_{\ell,u,d}}{1-Z} + \text{power series in } (1-Z)$$

for suitable constants $e_{\ell,u,d}$ and $c_{\ell,u,d}$. Extracting the coefficient of $Z^n$ yields the expectation of $W_n$ as

$$\begin{aligned} \mathrm{E}(W_n) &= e_{\ell,u,d}(n+1) + c_{\ell,u,d} + O(1) \\ &= e_{\ell,u,d}n + O(1). \end{aligned}$$

The constant $e_{\ell,u,d}$ is the *asymptotic density* of the representations; observe that $\lim_{n\to\infty} \mathrm{E}(W_n/n) = e_{\ell,u,d}$. A general formula for $e_{\ell,u,1}$ (listed below) can be obtained from the generating function listed in Example 6.4.

To determine $\mathrm{Var}(W_n)$, the variance of $W_n$, we compute $\mathrm{E}(W_n^2) - \mathrm{E}(W_n)^2$. Observe that

$$\left.\frac{\partial^2 G(Y,Z)}{\partial Y^2}\right|_{Y=1} + \left.\frac{\partial G(Y,Z)}{\partial Y}\right|_{Y=1} = \sum_{n\geq 0} E(W_n^2)Z^n.$$

Extracting the coefficient of $Z^n$ and subtracting $\mathrm{E}(W_n)^2 = (e_{\ell,u,d}(n+1) + c_{\ell,u,d} + O(1))^2$ gives

$$\mathrm{Var}(W_n) = v_{\ell,u,d}n + O(1)$$

for a suitable constant $v_{\ell,u,d}$. A general formula for $v_{\ell,u,1}$ (listed below) can be deduced. For other values of $d$, general formulae for $e_{\ell,u,d}$ and $v_{\ell,u,d}$ are quite long; however, Table 2 displays values for $e_{\ell,u,d}$ and $v_{\ell,u,d}$ in a number of special cases.

In summary, our analysis can be taken as proof of the following theorem:

**Theorem 5.** *There are constants $e_{\ell,u,d}$ and $v_{\ell,u,d}$ such that the expectation and variance of the random variable $W_n$, defined to be the weight of the $n$ least significant columns of a colexicographically minimal dimension-$d$ joint representation with digits from $D_{\ell,u}$, are given by*

$$\mathrm{E}(W_n) = e_{\ell,u,d}n + O(1) \quad \textit{and} \quad \mathrm{Var}(W_n) = v_{\ell,u,d}n + O(1).$$

| $\ell$ | $u$ | $d$ | $w$ | $e_{\ell,u,d}$ | $v_{\ell,u,d}$ |
|---|---|---|---|---|---|
| $-1$ | 1 | 1 | 2 | $\frac{1}{3}$ | $\frac{2}{27}$ |
| $-1$ | 1 | 2 | 2 | $\frac{1}{2}$ | $\frac{1}{16}$ |
| $-1$ | 1 | 3 | 2 | $\frac{23}{39}$ | $\frac{2800}{59319}$ |
| 0 | 5 | 1 | 3 | $\frac{2}{7}$ | $\frac{18}{343}$ |
| 0 | 5 | 2 | 3 | $\frac{32}{89}$ | $\frac{63200}{2114907}$ |
| 0 | 5 | 3 | 3 | $\frac{586}{1487}$ | $\frac{68928570}{3288008303}$ |
| $-3$ | 7 | 1 | 4 | $\frac{2}{9}$ | $\frac{2}{81}$ |
| $-3$ | 7 | 2 | 4 | $\frac{16}{59}$ | $\frac{2640}{205379}$ |
| $-3$ | 7 | 3 | 4 | $\frac{13942}{47595}$ | $\frac{354835806}{42033603575}$ |

TABLE 2: Coefficients of the dominant term in the asymptotic mean and variance of $W_n$.

| $\ell$ | $u$ | $e_{\ell,u,2}$ |
|---|---|---|
| even | even | $\frac{3+3\cdot 2^{-w}(2\delta-3)}{3w+\delta^2-3+2^{-w}\bigl(3w(2\delta-3)+3\delta^2-10\delta+9\bigr)}$ |
| even | odd | $\frac{3+3\cdot 2^{-w}(2\delta-3)}{3w+\delta^2-3+2^{-w}\bigl(3w(2\delta-3)+3\delta^2-10\delta+9\bigr)}$ |
| odd | even | $\frac{3+2^{-w+1}(2\delta-3)-2^{-2w+3}}{3w+\delta^2-3+2^{-w+1}\bigl(w(2\delta-3)+\delta^2-4\delta+3\bigr)-2^{-2w+2}(2w+2\delta-3)+2^{-3w+3}}$ |
| odd | odd | $\frac{3+2^{-w+2}(2\delta-3)+2^{-2w+4}}{3w+\delta^2-3+2^{-w+2}\bigl(w(2\delta-3)+\delta^2-3\delta+3\bigr)+2^{-2w+2}(4w+4\delta-7)+2^{-3w+4}}$ |

TABLE 3: General formula for $e_{\ell,u,2}$ where $u-\ell+1=\delta 2^{w-1}$ with $1 \le \delta < 2$.

*For $d=1$, we have*

$$e_{\ell,u,1} = \frac{1}{w-1+\lambda} \quad \textit{and} \quad v_{\ell,u,1} = \frac{(3-\lambda)\lambda}{(w-1+\lambda)^3},$$

*where*

$$\lambda = \frac{\#\mathsf{odd}(D_{\ell,u})}{2^{w-2}} = \frac{2(u-\ell+1)-(-1)^\ell-(-1)^u}{2^w}.$$

*General formulae for $e_{\ell,u,d}$ for $d=2$ are given in Table 3. For $d \in \{1,2,3,4\}$, general formulae for $e_{\ell,u,d}$ and $v_{\ell,u,d}$ are given on the accompanying web page [8].*

*Furthermore, the random variable $W_n$ satisfies the central limit law*

$$\lim_{n\to\infty} \Pr\Bigl(W_n \le \mathrm{E}(W_n) + x\sqrt{\mathrm{Var}(W_n)}\Bigr) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{t^2}{2}}\,dt.$$

**Remark 6.6.** The central limit law follows from Hwang's quasi-power theorem [9]. Also, the same expression for $e_{\ell,u,1}$ was obtained by Phillips and Burgess using a steady-state analysis of a Markov chain [15, see equation (13)].

# 7 Remarks

For $d=1$, it is easily seen that every integer has at most one colexicographically minimal representation with digits from $D_{\ell,u}$. The fact that every integer has a unique nonadjacent form and width-$w$ nonadjacent form can be viewed as special cases of this result. However, for $d > 1$, colexicographically minimal joint representations are not necessarily unique.

**Example 7.1.** For the digit set $D_{-3,5} = \{-3, -2, -1, 0, 1, 2, 3, 4, 5\}$, we have

$$\begin{pmatrix}5\\9\end{pmatrix} = \begin{pmatrix}0005\\1001\end{pmatrix}_2 = \begin{pmatrix}100\bar{3}\\1001\end{pmatrix}_2 .$$

It is easily seen that these two representations are both colexicographically minimal. ◊

Some authors [3] [10] have sought an algorithm which constructs minimal weight joint representations with digits restricted to the set $\{0, 1, 3\}$ or $\{0, \pm1, \pm3\}$. It seems natural that since Solinas was able to generalize the $\{0, \pm1\}$-nonadjacent form to joint representations that there would also be a generalization of the $\{0, 1, 3\}$-nonadjacent form, which is also known to have minimal weight (cf. [7, see Lemma 19]). But building minimal weight representations (for $d = 1$) with digits from $\{0, 1, 3\}$ is actually equivalent to building minimal weight representations with digits from $\{0, 1, 2, 3\}$ (recall Lemma 4.6). So perhaps the appropriate generalization is from the $\{0, 1, 3\}$-nonadjacent form to joint representations with digits from $\{0, 1, 2, 3\}$.

Nevertheless, it is possible that there may be a simple strategy for building minimal weight $\{0, 1, 3\}$-joint representations. However, all we can say for certain about such a strategy is that it is not the one that builds a colexicographically minimal representation.

**Example 7.2.** Suppose $(\dots A_2 A_1 A_0)_2$ is a colexicographically minimal representation of $N = (5, 9)^\mathsf{T}$ which uses the digits $\{0, 1, 3\}$. If we were trying to construct this representation, we would first try to make $A_0$ a zero column. However, since both 5 and 9 are odd, this is not possible. So, we try to make $A_1$ a zero column. This can only be done by setting $A_0$ to $(1, 1)^\mathsf{T} = (5, 9)^\mathsf{T} \mod 4$. If we continue in this manner, we arrive at the following representation:

$$\begin{pmatrix}0101\\1001\end{pmatrix}_2 = \begin{pmatrix}5\\9\end{pmatrix} .$$

This is a colexicographically minimal representation of $(5, 9)^\mathsf{T}$ and it has weight 3. However,

$$\begin{pmatrix}0013\\0033\end{pmatrix}_2 = \begin{pmatrix}5\\9\end{pmatrix} .$$

and this representation has weight 2. So, for the digits $\{0, 1, 3\}$, the strategy of building a colexicographically minimal representation does not necessarily give a minimal weight representation. ◊

Although the new family of minimal weight joint representations we have introduced (i.e., the outputs of Algorithm 3) can be viewed as generalizations of Solinas' Joint Sparse Form (JSF), Algorithm 3 cannot (in general) be used to build the JSF. When the parameters $d = 2, \ell = -1, u = 1$ are used, the output of Algorithm 3 may contain $1\bar{1}$ or $\bar{1}1$ which are not allowed in the JSF (e.g., this happens with the input $N = (1, 2)^\mathsf{T}$). For these parameters, the outputs of Algorithm 3 are exactly the dimension-2 Simple Joint Sparse Forms (SJSF) [6]. The SJSF and JSF have their zero columns in the same positions [6]; thus, because the SJSF is colexicographically minimal, so is the JSF.

## References

[1] R. AVANZI. A note on the signed sliding window integer recoding and a left-to-right analogue, in "Selected Areas in Cryptography – SAC 2004", *Lecture Notes in Computer Science* **3357** (2005), 130–143.

[2] D. BERNSTEIN. Pippenger's exponentiation algorithm. Preprint. `http://cr.yp.to/papers.html`.

[3] E. DAHMEN, K. OKEYA AND T. TAKAGI. An advanced method for joint scalar multiplications on memory constraint devices, in "Security and Privacy in Ad-hoc and Sensor Networks – ESAS 2005", *Lecture Notes in Computer Science* **3813** (2005), 189–204.

[4] FIPS 186-2. *Digital Signature Standard (DSS)*. Federal Information Processing Standards Publication 186-2, U.S. Department Of Commerce / National Institute of Standards and Technology, 2000. `http://www.csrc.nist.gov/publications/fips/`

[5] P. GRABNER, C. HEUBERGER, H. PRODINGER AND J. THUSWALDNER. Analysis of linear combination algorithms in cryptography. *ACM Transactions on Algorithms* **1** (2005), 123–142.

[6] P. GRABNER, C. HEUBERGER AND H. PRODINGER. Distribution results for low-weight binary representations for pairs of integers. *Theoretical Computer Science* **319** (2004), 307–331.

[7] C. HEUBERGER AND H. PRODINGER. Analysis of alternative digit sets for nonadjacent representations. *Monatshefte für Mathematik* **147** (2006), 219–248.

[8] C. HEUBERGER AND J. MUIR. Minimal weight and colexicographically minimal integer representations – online resources. `http://www.opt.math.tu-graz.ac.at/~cheub/publications/colexi/`

[9] H. HWANG. On convergence rates in the central limit theorems for combinatorial structures. *European Journal of Combinatorics* **19** (1998), 329–343.

[10] B. KUANG, Y. ZHU AND Y. ZHANG. An improved algorithm for $uP + vQ$ using $\mathrm{JSF}_3^1$, in "Applied Cryptography and Network Security – ACNS 2004", *Lecture Notes in Computer Science* **3089** (2004), 467–478.

[11] B. MÖLLER. Fractional windows revisited: improved signed-digit representations for efficient exponentiation, in "Information Security and Cryptology – ICISC 2004", *Lecture Notes in Computer Science* **3506** (2004), 137–153.

[12] F. MORAIN AND J. OLIVOS. Speeding up the computations on an elliptic curve using addition-subtraction chains, *RAIRO Theoretical Informatics and Applications* **24** (1990), 531–543.

[13] J. MUIR. *Efficient Integer Representations for Cryptographic Operations*, Ph.D. thesis, University of Waterloo, 2004. `http://etd.uwaterloo.ca/etd/jamuir2004.pdf`

[14] J. MUIR AND D. STINSON. Minimality and other properties of the width-$w$ nonadjacent form. *Mathematics of Computation* **75** (2006), 369–384.

[15] B. PHILLIPS AND N. BURGESS. Minimal weight digit set conversions. *IEEE Transactions on Computers* **53** (2004), 666–667.

[16] J. PROOS. Joint sparse forms and generating zero columns when combing. Technical Report CORR 2003-23, Centre for Applied Cryptographic Research, 2003. `http://www.cacr.math.uwaterloo.ca/techreports/2003/tech_reports2003.html`

[17] G. REITWIESNER. Binary arithmetic, in *Advances in Computers, Vol. 1*, Academic Press, 1960, pp. 231–308.

[18] J. SOLINAS. Low-weight binary representations for pairs of integers. Technical Report CORR 2001-41, Centre for Applied Cryptographic Research, 2001.

`http://www.cacr.math.uwaterloo.ca/techreports/2001/tech_reports2001.html`

[19] E. STRAUS. Addition chains of vectors (problem 5125). *American Mathematical Monthly* **71** (1964), 806–808.

# A Straus' Algorithm

The general form of Straus' algorithm is presented as Algorithm 4. Note that the multiplication indicated at line 2 (i.e., $P \cdot A$) is a matrix multiplication. The matrix (row vector) $P$ has dimension $1 \times d$, and the matrix (column vector) $A$ has dimension $d \times 1$.

**Algorithm 4** Straus' algorithm

**Input:** $N = (n_1, n_2, \ldots, n_d)^\mathsf{T},\ P = (P_1, P_2, \ldots P_d),\ k \in \mathbb{Z}^+$
**Output:** $Q = \sum_{i=1}^{d} n_i P_i$

```
1: for all $A \in \{0, 1, \ldots, 2^k - 1\}^{d \times 1} \setminus \{\vec{0}\}$ do
2:     $R_A \leftarrow P \cdot A$
3: $A_{s-1} \ldots A_1 A_0 \leftarrow$ the cols. of the dimension-$d$ radix-$2^k$ joint rep. of $N$ with digits from $\{0, 1, \ldots, 2^k - 1\}$
4: $Q \leftarrow R_{A_{s-1}}$
5: for $j = s - 2 \ldots 0$ do
6:     for $i = 1 \ldots k$ do $Q \leftarrow 2Q$
7:     if $A_j \neq \vec{0}$ then
8:         $Q \leftarrow Q + R_{A_j}$
9: return $Q$
```