Alternative Digit Sets for Nonadjacent Representations

James A. Muir[1⋆] and Douglas R. Stinson[2⋆⋆]

[1] Department of Combinatorics and Optimization

[2] School of Computer Science

University of Waterloo
Waterloo, Ontario, N2L 3G1
Canada
{jamuir,dstinson}@uwaterloo.ca

Abstract. It is known that every positive integer n can be represented as a finite sum of the form $n = \sum a_i 2^i$, where $a_i \in \{0, 1, -1\}$ for all i, and such that no two consecutive a_i's are non-zero. This is called a *non-adjacent representation*, and the set of digits $\{0, 1, -1\}$ is a *nonadjacent digit set* (or NADS). Nonadjacent representations are useful in efficiently implementing elliptic curve arithmetic for cryptographic applications.
In this paper, we consider nonadjacent digit sets of the form $\{0, 1, x\}$, where x is an integer. We present an algorithm to determine if $\{0, 1, x\}$ is a NADS; and if it is, we present an algorithm to efficiently determine the nonadjacent representation of any positive integer n. We also present some necessary and sufficient conditions for $\{0, 1, x\}$ to be a NADS. These conditions are used to exhibit infinite families of integers x such that $\{0, 1, x\}$ is a NADS, as well as inifinte families of x such that $\{0, 1, x\}$ is not a NADS.

1 Introduction and History

In a base 2 (or *radix* 2) positional number system, representations of integers are converted into integers via the rule

$$(\ldots a_3 a_2 a_1 a_0)_2 = \cdots + a_3 2^3 + a_2 2^2 + a_1 2^1 + a_0 \ .$$

Each of the a_i's is called a *digit*. In the usual radix 2 positional number system the digits have the property that $a_i \in \{0, 1\}$, for all i. If we let $D = \{0, 1\}$ then we say that D is the *digit set* for this number system.

It is often advantageous to employ alternate digit sets. The digit set $D = \{0, 1, \overline{1}\}$, where $\overline{1}$ stands for -1, was studied as early as 1951 by Booth. In [1], Booth presents a technique whereby a binary computer can calculate a representation of the product of two integers without any extra steps to correct for

⋆ Supported in part by an NSERC Postgraduate Scholarship.
⋆⋆ Supported by NSERC grant RGPIN 203114-02.

its sign. His method is implicitly based on replacing one of the operands in the multiplication with a $\{0, 1, \bar{1}\}$ radix 2 representation. Later, in 1960, through his investigations on how to reduce the number of additions and subtractions used in binary multiplication and division, Reitwiesner [6] gave a constructive proof that every integer has a canonical $\{0, 1, \bar{1}\}$ radix 2 representation with a *minimal* number of nonzero digits.

Reitwiesner's canonical representations have a simple description. A $\{0, 1, \bar{1}\}$ radix 2 representation of an integer is in Reitwiesner's canonical form if and only if it satisfies the following property:

NA-1 *Of any two adjacent digits, at least one is zero.*

Said another way, for such representations, nonzero digits are nonadjacent. These representations have come to be called *nonadjacent forms* (NAFs).

Cryptographers came to be interested in NAFs through a study of exponentiation. Jedwab and Mitchell [3] noticed that it is possible to reduce the number of multiplications used in the square-and-multiply algorithm for exponentiation if a $\{0, 1, \bar{1}\}$ radix 2 representation of the exponent is used. This led them to an independent discovery of the NAF. However, in multiplicative groups, like those used for RSA and DSA, using the digit $\bar{1}$ requires the computation of an inverse which is more costly than a multiplication.

In elliptic curve groups this is not a problem since inverses can be computed essentially for free. Morain and Olivos [5] observed that in these groups the operation analogous to exponentiation could be made more efficient using $\{0, 1, \bar{1}\}$ representations. They give two algorithms for performing scalar-multiplication using addition and subtraction. The $\{0, 1, \bar{1}\}$ radix 2 representations upon which their algorithms are based are in fact the same ones that Booth and Reitwiesner studied. In the quest for efficient implementations of elliptic curve cryptosystems, NAFs and representations like them have become an important device; Gordon [2] and Solinas [8, 9] make this point quite convincingly.

If a finite length radix 2 representation has digit set D and satisfies **NA-1**, we call it a D-*nonadjacent form* (D-NAF). In this paper, we consider the question of which sets D provide nonadjacent forms for *every positive* integer. If D is such a digit set then we call it a *nonadjacent digit set* (NADS).

As a first investigation into this topic, we examine digit sets of the form $\{0, 1, x\}$ with $x \in \mathbb{Z}$. It is known that letting $x = \bar{1}$ gives a NADS, but it is somewhat surprising that there are many values of x with this property; for example, $x = \bar{5}$, $\overline{13}$, $\overline{1145}$ (note $\bar{5}$ means -5, etc.).

We give an infinite family of x's for which $\{0, 1, x\}$ is a NADS, and we also give an infinite family of x's for which $\{0, 1, x\}$ is not a NADS. We also give some results on the necessary conditions D must satisfy in order to be a NADS. The algorithms we present and analyze for computing D-NAFs might be of some interest as well.

2 Preliminaries

We start by introducing some definitions and notation which will facilitate our discussions.

If n is an integer and we write $n = (\ldots a_2 a_1 a_0)_2$ then we are expressing n as the sum of an *infinite* series. If there is some ℓ such that $a_i = 0$ for all $i \geq \ell$ then n is the sum of a *finite* series and we indicate this by writing $n = (a_{\ell-1} \ldots a_2 a_1 a_0)_2$. If, in addition, $a_{\ell-1} \neq 0$ we say this representation has *length* ℓ.

Definition 1. *The* length *of a representation* $(\ldots a_2 a_1 a_0)_2$ *is the largest integer* ℓ *such that* $a_{\ell-1} \neq 0$ *but* $a_i = 0$ *for all* $i \geq \ell$*. The length of the all zero representation is defined to be zero.*

We will always use D to denote a digit set. The set of all *strings* of digits from D is denoted by D^*. The empty string is in D^* and is denoted by ϵ. Now, if D is the digit set for $(a_{\ell-1} \ldots a_1 a_0)_2$, then $a_{\ell-1} \ldots a_1 a_0$ is a string in D^*. Conversely, any string $\alpha \in D^*$ corresponds to a radix 2 representation with digit set D, namely $(\alpha)_2$. If $\alpha, \beta \in D^*$ then we denote their *concatenation* by $\alpha || \beta$.

We apply some of our terminology for representations to strings. If $0 \in D$ and a finite string $\alpha \in D^*$ satisfies the property **NA-1**, then we call α a D-NAF. If in addition, $(\alpha)_2 = n$ we say α is a D-NAF for n. Notice that if α is a D-NAF for n then α with any leading zeros removed is also a D-NAF for n. We denote the string formed by deleting the leading zeros from α by $\widehat{\alpha}$.

Given a digit set D and an integer n, we define

$$R_D(n) := \begin{cases} \widehat{\alpha} & \text{where } \alpha \in D^* \text{ is a } D\text{-NAF for } n\text{, if one exists} \\ \perp & \text{otherwise.} \end{cases}$$

Here, $\perp$ is just some symbol not in D. If $R_D(n)$ is a D-NAF for n, then by definition that string has no leading zeros. For example, if $D = \{0, 1, \overline{9}\}$ then $R_D(7) = 1000\overline{9}$ since $1000\overline{9}$ is a D-NAF, has no leading zeros, and $(1000\overline{9})_2 = 7$. However, $R_D(3) = \perp$ since 3 does not have a D-NAF, which we will prove later on.

We are interested in determining which integers have D-NAFs, so we define the set

$$\mathrm{NAF}(D) := \{n \in \mathbb{Z} : R_D(n) \neq \perp\} \ .$$

From our example with $D = \{0, 1, \overline{9}\}$ we see $7 \in \mathrm{NAF}(D)$ but $3 \notin \mathrm{NAF}(D)$. Using this notation, our definition of a nonadjacent digit set is as follows:

Definition 2. *D is a* nonadjacent digit set *if* $\mathbb{Z}^+ \subseteq \mathrm{NAF}(D)$.

3 Necessary Conditions for $\{0, 1, x\}$ to be a NADS

If we suppose $D = \{0, 1, x\}$ is a nonadjacent digit set then we can deduce necessary conditions on x.

Theorem 3. *Let* $D = \{0, 1, x\}$*. If* $3 \in \mathrm{NAF}(D)$*, then* $x \equiv 3 \pmod 4$*.*

Proof. For some particular D-NAF, say $(\ldots a_2 a_1 a_0)_2$, we have

$$\begin{aligned}(\ldots a_2 a_1 a_0)_2 &= 3 \\ &\implies a_0 \equiv 1 \pmod 2 \\ &\implies a_0 \neq 0\ .\end{aligned}$$

Since a_0 is nonzero and the representation is nonadjacent we have $a_1 = 0$. Thus

$$\begin{aligned}(\ldots a_2 0 a_0)_2 &= 3 \\ &\implies a_0 \equiv 3 \pmod 4 \\ &\implies a_0 \neq 1 \\ &\implies a_0 = x\ .\end{aligned}$$

So $x = a_0 \equiv 3 \pmod 4$. ∎

If $D = \{0, 1, x\}$ is a NADS then $3 \in \mathrm{NAF}(D)$, and by the previous result $x \equiv 3 \pmod 4$. So, if we are trying to find a value of x that makes $\{0, 1, x\}$ a NADS we need only consider those values congruent to 3 modulo 4.

3.1 The case $x > 0$

If we restrict x to be a positive integer, then we can give a complete characterization of all values which make $D = \{0, 1, x\}$ a NADS. It is well known that $x = 3$ is such a value, and this is remarked by Solinas [7]. We give a proof of this fact and then show that no other positive value of x makes $\{0, 1, x\}$ a NADS.

Theorem 4. *The only NADS of the form* $\{0, 1, x\}$ *with* $x > 0$ *is* $\{0, 1, 3\}$*.*

Proof. Let n be any positive integer. We want to show that n has a $\{0, 1, 3\}$-NAF. Let $(\ldots a_2 a_1 a_0)_2$ be the usual $\{0, 1\}$-radix 2 representation of n. If this representation satisfies **NA-1** there is nothing to prove, so suppose it does not. Let i be the smallest integer for which $a_{i+1} = a_i = 1$. Replace digits a_{i+1} and a_i by 0 and 3, respectively. Since $2^{i+1} + 2^i = 0 \cdot 2^{i+1} + 3 \cdot 2^i$, the resulting representation stands for the same integer.

By working from right to left, repeating this substitution as necessary, we transform $(\ldots a_2 a_1 a_0)_2$ into a $\{0, 1, 3\}$-NAF. This proves that $\{0, 1, 3\}$ is a NADS.

Now consider x with $x > 3$. We show $n = 3$ does not have a $\{0, 1, x\}$-NAF. Suppose to the contrary that for some $\{0, 1, x\}$-NAF we have $(\ldots a_2 a_1 a_0)_2 = 3$. Since 3 is odd, $a_0 \neq 0$ and so $a_1 = 0$. Now $a_0 \equiv 3 \pmod 4$ so it must be that $a_0 = x$. However, since each of the digits in $\{0, 1, x\}$ is nonnegative we have

$$3 = (\ldots a_2 0 x)_2 = \cdots + a_2 2^2 + 0 \cdot 2^1 + x \geq x > 3\ ,$$

which is a contradiction. So, 3 does not have a $\{0, 1, x\}$-NAF when $x > 3$. ∎

An example helps illustrate the construction used in the above proof. Suppose $n = 237$. To find a $\{0, 1, 3\}$-NAF for 237 we start with its usual binary representation and then, working from right to left, replace any occurrences of the digits 11 with 03:

$$237 = (11101101)_2 = (10300301)_2 \ .$$

A natural question to ask is if this is the only $\{0, 1, 3\}$-NAF for 237. We give the answer in the next section.

3.2 Uniqueness

We show that every integer, not only just the positive ones, has at most one $\{0, 1, x\}$-NAF where $x \equiv 3 \pmod 4$.

Theorem 5. *If $x \equiv 3 \pmod 4$, then any integer has at most one finite length $\{0, 1, x\}$-nonadjacent form.*

Proof. Let $D = \{0, 1, x\}$ and suppose the result is false. Then it must be that

$$(a_{\ell-1} \dots a_2 a_1 a_0)_2 = (b_{\ell'-1} \dots b_2 b_1 b_0)_2$$

where $(a_{\ell-1} \dots a_2 a_1 a_0)_2$ and $(b_{\ell'-1} \dots b_2 b_1 b_0)_2$ are two different D-NAFs with lengths ℓ and ℓ' respectively. These representations stand for the same integer, call it n. We can assume that ℓ is as small as possible.

If $a_0 = b_0$, then

$$(a_{\ell-1} \dots a_2 a_1)_2 = (b_{\ell'-1} \dots b_2 b_1)_2 \ ,$$

and so we have two different, and shorter, D-NAFs which stand for the same integer, contrary to the minimality of ℓ. So it must be that $a_0 \neq b_0$.

If one of a_0 or b_0 is 0, then n is even, and so both a_0 and b_0 are 0. But a_0 and b_0 are different so it must be that a_0 is equal to 1 or x. Without loss of generality, we can assume the representations have the form

$$(a_{\ell-1} \dots a_2 0 x)_2 = (b_{\ell'-1} \dots b_2 01)_2 \ .$$

This implies $x \equiv 1 \pmod 4$, contrary to our hypothesis that $x \equiv 3 \pmod 4$. Thus every integer has at most one D-NAF. ■

4 Recognizing NADS of the form $\{0, 1, x\}$

From now on we fix $D = \{0, 1, x\}$ with $x \equiv 3 \pmod 4$. In this section we work towards a method of deciding if $\{0, 1, x\}$ is a NADS. By Theorem 4, this is easy when $x > 0$, so we will assume $x < 0$.

Recall that $R_D(n)$ either evaluates to the symbol $\perp$ or a finite string, with no leading zeros, that is a D-NAF for n. Theorem 5 tells us that any n has at most one D-NAF, so in the second case, the string returned by $R_D(n)$ is unique. Thus, $R_D(n)$ is well defined (i.e., for every input n there is exactly one output.).

The ability to evaluate $R_D(n)$ can be useful in deciding if D is a NADS. If we can find $n \in \mathbb{Z}^+$ such that $R_D(n) = \perp$ then we know that D is not a NADS. Also, if we have an algorithmic description of $R_D(n)$, we might be able to analyze this algorithm and show that for any $n \in \mathbb{Z}^+$, $R_D(n) \neq \perp$, thus proving that D is a NADS.

We show that $R_D(n)$ can be computed recursively and give an algorithm which evaluates $R_D(n)$ in this manner. We begin with some lemmas:

Lemma 6. *If $n \equiv 0 \pmod 4$ then $n \in \mathrm{NAF}(D)$ if and only if $n/4 \in \mathrm{NAF}(D)$. Further, if $n \in \mathrm{NAF}(D)$ then $R_D(n) = R_D(n/4)||00$.*

Proof. Since $n \equiv 0 \pmod 4$, the definition of the digit set D implies that any D-NAF for n is of the form $(a_{\ell-1} \ldots a_3 a_2 00)_2$, where $a_{\ell-1} \neq 0$. Now,

$$\begin{aligned} n \in \mathrm{NAF}(D) &\iff n \text{ has a } D\text{-NAF of the form } (a_{\ell-1} \ldots a_3 a_2 00)_2 \\ &\iff n/4 \text{ has a } D\text{-NAF of the form } (a_{\ell-1} \ldots a_3 a_2)_2 \\ &\iff n/4 \in \mathrm{NAF}(D) \ , \end{aligned}$$

which proves the first part of the lemma. If $n \in \mathrm{NAF}(D)$ then

$$R_D(n) = a_{\ell-1} \ldots a_3 a_2 00 = a_{\ell-1} \ldots a_3 a_2 || 00 = R_D(n/4)||00 \ ,$$

which proves the second part of the lemma. ■

We omit the proofs of the next three lemmas since they can be established by making only minor changes to the proof of Lemma 6.

Lemma 7. *If $n \equiv 1 \pmod 4$ then $n \in \mathrm{NAF}(D)$ if and only if $(n-1)/4 \in \mathrm{NAF}(D)$. Further, if $n \in \mathrm{NAF}(D)$ then $R_D(n) = R_D(\frac{n-1}{4})||01$.*

Lemma 8. *If $n \equiv 2 \pmod 4$ then $n \in \mathrm{NAF}(D)$ if and only if $n/2 \in \mathrm{NAF}(D)$. Further, if $n \in \mathrm{NAF}(D)$ then $R_D(n) = R_D(n/2)||0$.*

Lemma 9. *If $n \equiv 3 \pmod 4$ then $n \in \mathrm{NAF}(D)$ if and only if $(n-x)/4 \in \mathrm{NAF}(D)$. Further, if $n \in \mathrm{NAF}(D)$ then $R_D(n) = R_D(\frac{n-x}{4})||0x$.*

Given an integer n, if we somehow know that $n \in \mathrm{NAF}(D)$ then Lemmas 6–9 suggest a recursive procedure that we can use to evaluate $R_D(n)$. To illustrate suppose $D = \{0, 1, \overline{9}\}$. It was shown in an earlier example that $7 \in \mathrm{NAF}(D)$. Using these lemmas, we have:

$$R_D(7) = R_D(4)||0\overline{9} = R_D(1)||00||0\overline{9} = 1||00||0\overline{9} = 1000\overline{9} \ .$$

To describe the general procedure for computing $R_D(n)$, given that $n \in \mathrm{NAF}(D)$, we use the following two functions:

$$f_D(n) := \begin{cases} n/4 & \text{if } n \equiv 0 \pmod 4 \\ (n-1)/4 & \text{if } n \equiv 1 \pmod 4 \\ n/2 & \text{if } n \equiv 2 \pmod 4 \\ (n-x)/4 & \text{if } n \equiv 3 \pmod 4 \ , \end{cases}$$

$$g_D(n) := \begin{cases} 00 & \text{if } n \equiv 0 \pmod 4 \\ 01 & \text{if } n \equiv 1 \pmod 4 \\ 0 & \text{if } n \equiv 2 \pmod 4 \\ 0x & \text{if } n \equiv 3 \pmod 4 \ . \end{cases}$$

Note that f_D returns an integer, and g_D returns a string. Here is the procedure described in pseudocode:

Procedure 10: EVAL_α-$R_D(n)$

$\alpha \leftarrow \epsilon$
while $n \neq 0$
  do $\begin{cases} \alpha \leftarrow g_D(n) \mid\mid \alpha \\ n \leftarrow f_D(n) \end{cases}$
return $\widehat{\alpha}$

Procedure 10 terminates on input n if and only if ${f_D}^i(n) = 0$ for some positive integer i. An easy calculation shows that, for $D = \{0, 1, \overline{9}\}$, ${f_D}^3(7) = 0$, and so the procedure terminates on input $n = 7$. However, $f_D(3) = 3$ and so ${f_D}^i(3) = 3 \neq 0$ for all i, thus the procedure does not terminate on input $n = 3$.

Using the previous lemmas, we can show Procedure 10 terminates on input n if and only if $n \in \text{NAF}(D)$. Instead of making use of the lemmas individually, it is more convenient to summarize them as follows:

Lemma 11. *For all $n \in \mathbb{Z}$, $n \in \text{NAF}(D)$ if and only if $f_D(n) \in \text{NAF}(D)$. Further, if $n \in \text{NAF}(D)$ then $R_D(n) = R_D(f_D(n))||g_D(n)$.*

Now, suppose $n \in \text{NAF}(D)$. Then the finite string $R_D(n)$ can be computed with a finite number of recursive steps. This implies that there is some positive integer i such that ${f_D}^i(n) = 0$, which in turn implies that the procedure terminates. Conversely, suppose the procedure terminates. Then ${f_D}^i(n) = 0$ for some i, and clearly $0 \in \text{NAF}(D)$. Thus, ${f_D}^i(n) \in \text{NAF}(D)$, and by the lemma $n \in \text{NAF}(D)$.

Procedure 10 is named EVAL_α-$R_D(n)$. We justify this name by noting that if the procedure terminates, it returns a string with no leading zeros (i.e., $\widehat{\alpha}$) equal to $R_D(n)$. We are not able to evaluate $R_D(n)$ for all values of n using this procedure because we have not yet described a way to recognize when $R_D(n) = \perp$. We proceed to do this now.

To decide if $D = \{0, 1, x\}$ is a NADS, it suffices to determine if there are any $n \in \mathbb{Z}^+$ for which Procedure 10 fails to terminate. We can determine if the procedure will terminate by examining the iterates of f_D.

Let n be a positive integer. Observe that, for $n \not\equiv 3 \pmod 4$, we have that

$$n > f_D(n) \geq 0 \ , \tag{1}$$

and, for $n \equiv 3 \pmod 4$, that

$$n > f_D(n) \iff n > \frac{-x}{3} \tag{2}$$

$$f_D(n) \geq 0 \iff n \geq x \ . \tag{3}$$

Since x is negative, we see that any iterate of the function f_D, on input n, always results in a nonnegative integer. Consider the graph G_n having directed edges

$$n \to f_D(n) \to {f_D}^2(n) \to {f_D}^3(n) \to \cdots \quad .$$

The vertices of G_n are nonnegative integers. Inequalities (1) and (2) tell us that there must be some vertex of G_n that is less than $\frac{-x}{3}$. Suppose ${f_D}^i(n) < \frac{-x}{3}$. We claim ${f_D}^{i+1}(n) < \frac{-x}{3}$ as well. This is clearly true if ${f_D}^i(n) \equiv 0, 1, 2 \pmod 4$. If ${f_D}^i(n) \equiv 3 \pmod 4$ then

$$\begin{aligned} {f_D}^i(n) < \frac{-x}{3} &\implies \frac{{f_D}^i(n) - x}{4} < \frac{\frac{-x}{3} - x}{4} \\ &\implies {f_D}^{i+1}(n) < \frac{-x - 3x}{12} = \frac{-x}{3} \ , \end{aligned}$$

and so the claim is true. The claim also tells us that if ${f_D}^i(n) < \frac{-x}{3}$, then any subsequent iterate of f_D must be less than $\frac{-x}{3}$.

From the preceding discussion it is clear that for a positive integer n, either:

1. G_n is a path terminating at 0, or
2. G_n contains a directed cycle of integers in the interval $\{1, 2, \ldots, \lfloor \frac{-x}{3} \rfloor\}$.

If we can detect a directed cycle in G_n then we can determine whether or not Procedure 10 will terminate on input n. To do this we need to compute and store some of the vertices of G_n. However, as Procedure 10 executes, it computes all the vertices of G_n, so we might as well modify the procedure to detect a directed cycle in G_n on its own. This modification is described as Algorithm 12.

Algorithm 12: EVAL-$R_D(n)$

comment: n is a positive integer

$\alpha \leftarrow \epsilon$
while $n > \frac{-x}{3}$
do $\begin{cases} \alpha \leftarrow g_D(n) \mid\mid \alpha \\ n \leftarrow f_D(n) \end{cases}$
$\mathcal{S} \leftarrow \varnothing$
while $n \neq 0$
do $\begin{cases} \textbf{if } n \in \mathcal{S} \\ \quad \textbf{then return } \perp \\ \mathcal{S} \leftarrow \mathcal{S} \cup \{n\} \\ \alpha \leftarrow g_D(n) \mid\mid \alpha \\ n \leftarrow f_D(n) \end{cases}$
return $\widehat{\alpha}$

Now we can use the title "Algorithm" rather than "Procedure", because EVAL-$R_D(n)$ terminates for every $n \in \mathbb{Z}^+$. (For some positive integers, it was shown that EVAL$_\alpha$-$R_D(n)$ fails to terminate, which is why it cannot technically be called an algorithm.) As its name suggests, Algorithm 12 evaluates $R_D(n)$ for any $n \in \mathbb{Z}^+$. It is possible to show that the running time of EVAL-$R_D(n)$ is $O(\lg n + |x|)$.

Returning to our main task of recognizing when $\{0, 1, x\}$ is a NADS, Algorithm 12 and the preceding analysis are very helpful since they lead us to the following result:

Theorem 13. *Suppose x is a negative integer and $x \equiv 3 \pmod 4$. If every element in the set $\{n \in \mathbb{Z}^+ : n \leq \lfloor -x/3 \rfloor\}$ has a $\{0, 1, x\}$-NAF, then $\{0, 1, x\}$ is a NADS.*

Proof. From inspection of Algorithm 12 this result is almost immediate, however we can give a formal argument using the graph G_n.

Suppose the hypothesis is true. We must argue that $\{0, 1, x\}$ is a NADS. Take any $n \in \mathbb{Z}^+$ and consider the graph G_n. Suppose G_n contains a directed cycle. Let n_0 be a vertex in this cycle. Then $1 \leq n_0 \leq \lfloor -x/3 \rfloor$, and G_{n_0} must contain the same directed cycle. This implies that n_0 does not have a $\{0, 1, x\}$-NAF, contrary to our hypothesis. So, G_n is a path terminating at 0, and thus n has a $\{0, 1, x\}$-NAF. ■

Theorem 13 suggests a computational method of determining if $\{0, 1, x\}$ is a NADS. For each $n \in \mathbb{Z}^+, n \leq \lfloor -x/3 \rfloor$, compute EVAL-$R_D(n)$. If all of these values have $\{0, 1, x\}$-NAFs then $\{0, 1, x\}$ is a NADS; otherwise, we find a value which does not have a $\{0, 1, x\}$-NAF which proves that $\{0, 1, x\}$ is not a NADS. To recognize a NADS, this method requires $\lfloor -x/3 \rfloor$ calls to EVAL-$R_D(n)$. However, we can decrease this number, as the next result shows.

Corollary 14. *Suppose x is a negative integer and $x \equiv 3 \pmod 4$. If every element in the set $\{n \in \mathbb{Z}^+ : n \leq \lfloor -x/3 \rfloor, n \equiv 3 \pmod 4\}$ has a $\{0, 1, x\}$-NAF, then $\{0, 1, x\}$ is a NADS.*

Proof. If $\{0, 1, x\}$ is not a NADS then choose the smallest integer $n_0 \in \mathbb{Z}^+$ such that G_{n_0} contains a directed cycle. By Theorem 13 it must be that $n_0 \leq \lfloor -x/3 \rfloor$. Let $n_1 = f_D(n_0)$, then (n_0, n_1) is an arc of G_n. If $n_0 \not\equiv 3 \pmod 4$ then $n_0 < n_1$ and G_{n_1} contains the same directed cycle, contrary to the choice of n_0. Thus, it must be that $n_0 \equiv 3 \pmod 4$. So, if the hypothesis is true, there can be no smallest positive integer which does not have a $\{0, 1, x\}$-NAF. Hence $\{0, 1, x\}$ is a NADS. ■

Now we can detect a NADS of the form $\{0, 1, x\}$ with $\lfloor -x/12 \rfloor$ calls to EVAL-$R_D(n)$. We have used this approach to find all the values of x greater than -10^6 such that $\{0, 1, x\}$ is a NADS. The Appendix lists all the negative values of x greater than -10^4 such that $\{0, 1, x\}$ is a NADS.

5 Some Infinite Families of NADS and non-NADS

Again, we fix $D = \{0, 1, x\}$ where $x \equiv 3 \pmod 4$. In this section, we give an infinite family of values for x which makes D a NADS. We also give some infinite families of values for x for which D is not a NADS.

If n is a nonnegative integer, $w(n)$ denotes the number of ones in the usual $\{0, 1\}$-radix 2 representation of n (i.e., the Hamming weight of n). We use the function $w(n)$ to describe our first infinite family.

Theorem 15. *Let x be a negative integer with $x \equiv 3 \pmod 4$. If $w\left(\frac{3-x}{4}\right) = 1$, then $\{0, 1, x\}$ is a NADS.*

Proof. Suppose $\{0, 1, x\}$ is not a NADS. Then there is some $n \in \mathbb{Z}^+$ for which the graph G_n contains a directed cycle. We can assume n is a vertex of this cycle. Let t be the number of vertices in the cycle, then

$$n \to f_D(n) \to {f_D}^2(n) \to \cdots \to {f_D}^{t-1}(n) \to n \ .$$

Let $n' = f_D(n)$. We want to relate $w(n')$ to $w(n)$. There are four possible residues of n modulo 4, and for the residues $0, 1, 2$ we can determine $w(n')$ exactly:

$n \bmod 4$	n'	$w(n')$
0	$\frac{n}{4}$	$w(n)$
1	$\frac{n-1}{4}$	$w(n) - 1$
2	$\frac{n}{2}$	$w(n)$

If $n \equiv 3 \pmod 4$, we have

$$n' = \frac{n-x}{4} = \frac{n-3}{4} + \frac{3-x}{4} \ .$$

By hypothesis $w\left(\frac{3-x}{4}\right) = 1$, and so

$$\begin{aligned} w(n') &= w\left(\frac{n-3}{4} + \frac{3-x}{4}\right) \\ &\le w\left(\frac{n-3}{4}\right) + w\left(\frac{3-x}{4}\right) \\ &= w(n) - 2 + 1 \\ &= w(n) - 1 \ . \end{aligned}$$

So, in any case, $w(n') \le w(n)$, but if n is odd then we have the strict inequality $w(n') < w(n)$. Applying this inequality to the integers in the cycle of G_n we see

$$w(n) \ge w(f_D(n)) \ge w({f_D}^2(n)) \ge \cdots \ge w({f_D}^{t-1}(n)) \ge w(n) \ .$$

However, some vertex in this cycle must be congruent to 3 modulo 4. If not, then the iterates of f_D are strictly decreasing on this cycle and we get

$$n > f_D(n) > {f_D}^2(n) > \cdots > {f_D}^{t-1}(n) > n \ ,$$

which is a contradiction. So, there is some odd vertex in the cycle which means one of the inequalities relating the Hamming weights of adjacent vertices is strict. This implies that $w(n) > w(n)$, which is a contradiction.

So, G_n cannot contain a directed cycle, and hence $\{0, 1, x\}$ is a NADS. ■

Theorem 15 asserts that $x = -5, -13, -29, -61$, etc. all yield NADS. We now present an infinite family of integers x such that $\{0, 1, x\}$ is not a NADS.

Theorem 16. *Let x be a negative integer with $x \equiv 3 \pmod 4$. If $(2^s - 1)|x$ for any $s \geq 2$, then $\{0, 1, x\}$ is not a NADS.*

Proof. Let $n = -x/(2^s - 1)$. We show G_n contains a directed cycle. We have

$$\begin{aligned} & n(2^s - 1) \equiv -x \pmod 4 \\ \implies & n(0 - 1) \equiv -3 \pmod 4 \\ \implies & n \equiv 3 \pmod 4 . \end{aligned}$$

Note that,

$$n - x = \frac{-x}{2^s - 1} - x = \frac{-x - x\ 2^s + x}{2^s - 1} = 2^s \frac{-x}{2^s - 1} = 2^s n .$$

Now,

$$f_D(n) = \frac{n - x}{4} = 2^{s-2} n$$

Subsequent iterates of f_D will cancel out the factor 2^{s-2}. Thus, for some i, ${f_D}^i(n) = n$ and so G_n contains a directed cycle. ■

Theorem 16 says that many sets $\{0, 1, x\}$ are not NADS. In particular, it rules out values of x that are divisible by 3, 7, 31, etc.

Theorem 17. *If $x = 3 - 11 \cdot 2^i$, where $i \geq 2$, then $\{0, 1, x\}$ is not a NADS.*

Proof. We show that G_3 contains a directed cycle, and hence 3 does not have a $\{0, 1, x\}$-NAF. Let $y = \frac{3-x}{4} = 11 \cdot 2^{i-2}$. Note that, if $n \equiv 3 \pmod 4$, then

$$f_D(n) = \frac{n - x}{4} = \frac{n - 3}{4} + \frac{3 - x}{4} = \frac{n - 3}{4} + y .$$

Now, consider the iterates of f_D on input 3. We have

$$\begin{aligned} f_D(3) &= 11 \cdot 2^{i-2} \\ {f_D}^2(3) &= 11 \cdot 2^{i-4} \\ &\vdots \\ {f_D}^j(3) &= 11, \text{ for some } j \\ {f_D}^{j+1}(3) &= 2 + y . \end{aligned}$$

We proceed using case analysis.

Case $i \geq 5$: We have $2 + y = 2 + 11 \cdot 2^{i-2} = (10110\ldots010)_2$, and subsequent iterates of f_D will effectively strip off the nonadjacent representation $(0\ldots010)_2$ at the right of $2+y$, and will return to the value $(1011)_2 = 11$. Hence, 11 is in a directed cycle of G_3.

Case $i = 4$*:* Here $y = 44$, and so $2 + y = 46$. By iterating f_D we see that in G_3 we have:

$$46 \to 23 \to 49 \to 12 \to 3\ .$$

So, an iterate of f_D returns to the vertex 3 in G_3. Thus we have a directed cycle.

Case $i = 3$*:* Here $y = 22$, and so $2+y = 24$. The iterates of f_D are as follows:

$$24 \to 6 \to 3\ .$$

So, we see G_3 contains a directed cycle.

Case $i = 2$*:* Here $y = 11$, and so $2+y = 13$. The iterates of f_D are as follows:

$$24 \to 6 \to 3\ .$$

So, we see G_3 contains a directed cycle.

In all the above cases, G_3 contains a directed cycle, and $\{0, 1, x\}$ is not an NADS. ■

Theorem 18. *Let* $x = 3 - 7 \cdot 2^i$*, where* $i \geq 2$*. Then* $\{0, 1, x\}$ *is an NADS if and only if* $i \in \{2, 3\}$*.*

Proof. When $i = 2$, $x = \overline{25}$, and when $i = 3$, $x = \overline{53}$. We first show that both $\{0, 1, \overline{25}\}$ and $\{0, 1, \overline{53}\}$ are NADSs.

For $x = \overline{25}$, we have $\lfloor -x/3 \rfloor = 8$. By Corollary 14, it is sufficient to check that 3 and 7 have $\{0, 1, \overline{25}\}$-NAFs. Consider the graph G_3. Iterating f_D on input $n = 3$ we see G_3 is equal to:

$$3 \to 7 \to 8 \to 2 \to 1 \to 0\ .$$

Notice that 7 is a vertex of G_3, and so G_7 is contained in G_3. Both G_3 and G_7 are paths terminating at 0, and hence both 3 and 7 have $\{0, 1, \overline{25}\}$-NAFs. Thus $\{0, 1, \overline{25}\}$ is a NADS.

For $x = \overline{53}$, we have $\lfloor -x/3 \rfloor = 17$. Arguing as before, we see that it is sufficient to demonstrate that G_n, where $n = 3, 7, 11, 15$, consists of a path. By iterating f_D on input $n = 3$ we see G_3 is equal to:

$$3 \to 14 \to 7 \to 15 \to 17 \to 4 \to 1 \to 0$$

This takes care of G_3, G_7 and G_{15}. Consider G_{11}:

$$11 \to 16 \to 4 \to 1 \to 0$$

Thus $\{0, 1, \overline{53}\}$ is an NADS.

Now, when $i \geq 4$ we must show that $\{0, 1, x\}$ is not an NADS. We do so by showing G_7 contains directed cycle. Let $y = \frac{3-x}{4} = 7 \cdot 2^{i-2}$. We have

$$f_D(7) = \frac{7-3}{4} + y = 1 + y \ .$$

Since $i \geq 4$ we have $1 + y = (1110\ldots001)_2$. Subsequent iterates of f_D will effectively right shift $1 + y$, leaving the representation $(111)_2 = 7$. So, we see that 7 is in a directed cycle. ■

6 Open Problems

It is an interesting question if there exists a simply stated set of necessary and sufficient conditions for $\{0, 1, x\}$ to be a NADS. A particularly interesting special case is addressed in the following conjecture.

Conjecture 19. Let x be a negative integer with $x \equiv 3 \pmod 4$. If $w\left(\frac{3-x}{4}\right) = 2$ and $2^s - 1$ does not divide x for any $s \in \mathbb{Z}^+$, $s \geq 2$, then $\{0, 1, x\}$ is an NADS.

References

1. A. D. Booth. A Signed Binary Multiplication Technique, *Quarterly Journal of Mechanics and Applied Mathematics* **4** (1951), 236–240.
2. D. M. Gordon. A Survey of Fast Exponentiation Methods, *Journal of Algorithms* **27** (1998), 129–146.
3. J. Jedwab and C. J. Mitchell. Minimum Weight Modified Signed-Digit Representations and Fast Exponentiation, *Electronic Letters* **25** (1989), 1171–1172.
4. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*, CRC Press, 1996.
5. 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.
6. G. W. Reitwiesner. Binary Arithmetic, In *Advances in Computers, Vol. 1*, Academic Press, 1960, pp. 231–308.
7. J. A. Solinas. Low-Weight Binary Representations for Pairs of Integers. Technical Report CORR 2001-41, Centre for Applied Cryptographic Research. Available from `http://www.cacr.math.uwaterloo.ca/techreports/2001/corr2001-41.ps`.
8. J. A. Solinas. An Improved Algorithm for Arithmetic on a Family of Elliptic Curves. In "Advances in Cryptology – CRYPTO '97", *Lecture Notes in Computer Science* **1294** (1997), 357–371. An extended version of the paper is available from `http://-www.cacr.math.uwaterloo.ca/techreports/1999/corr99-46.ps`.
9. J. A. Solinas. Efficient arithmetic on Koblitz curves. *Designs, Codes and Cryptography* **19** (2000), 195–249.

Appendix

We list the all values of x from -1 to -10000 for which $\{0, 1, x\}$ is a NADS:

-1	-5	-13	-17	-25	-29	-37	-53	-61	-65
-113	-121	-125	-137	-145	-149	-157	-233	-241	-253
-257	-265	-269	-277	-281	-305	-317	-325	-437	-481
-485	-493	-505	-509	-517	-521	-533	-541	-557	-565
-601	-605	-613	-629	-641	-653	-673	-821	-869	-913
-937	-977	-989	-1013	-1021	-1025	-1033	-1037	-1045	-1061
-1073	-1081	-1097	-1117	-1133	-1145	-1165	-1265	-1273	-1277
-1289	-1297	-1325	-1345	-1349	-1357	-1621	-1637	-1733	-1745
-1765	-1885	-1933	-1949	-1985	-1993	-2017	-2021	-2033	-2041
-2045	-2053	-2069	-2101	-2105	-2113	-2129	-2137	-2141	-2153
-2161	-2165	-2173	-2185	-2189	-2197	-2237	-2273	-2285	-2293
-2297	-2321	-2353	-2365	-2369	-2381	-2393	-2405	-2425	-2497
-2525	-2533	-2557	-2593	-2609	-2621	-2641	-2645	-2669	-2677
-2693	-3245	-3265	-3337	-3385	-3421	-3509	-3541	-3557	-3629
-3653	-3673	-3761	-3797	-3853	-3877	-3881	-3917	-3925	-3929
-3961	-4001	-4033	-4037	-4085	-4093	-4097	-4105	-4117	-4121
-4133	-4141	-4145	-4153	-4157	-4201	-4205	-4217	-4253	-4261
-4273	-4285	-4297	-4337	-4345	-4349	-4373	-4393	-4397	-4469
-4537	-4541	-4573	-4589	-4597	-4601	-4621	-4633	-4645	-4649
-4661	-4693	-4777	-4801	-5021	-5077	-5093	-5101	-5105	-5113
-5129	-5137	-5153	-5165	-5189	-5197	-5213	-5273	-5281	-5365
-5377	-5381	-5393	-5405	-5437	-5441	-6565	-6613	-6773	-6805
-6929	-6973	-7033	-7277	-7333	-7345	-7381	-7393	-7397	-7465
-7477	-7561	-7597	-7613	-7621	-7649	-7741	-7817	-7865	-7877
-7901	-7949	-8045	-8053	-8065	-8069	-8081	-8093	-8101	-8117
-8129	-8165	-8173	-8177	-8185	-8189	-8201	-8213	-8221	-8233
-8237	-8297	-8305	-8317	-8333	-8341	-8369	-8417	-8429	-8437
-8441	-8453	-8485	-8497	-8501	-8573	-8581	-8593	-8597	-8665
-8669	-8681	-8693	-8717	-8725	-8741	-8753	-8789	-8797	-8825
-8837	-8921	-8977	-9089	-9101	-9133	-9157	-9161	-9181	-9209
-9221	-9245	-9341	-9353	-9421	-9425	-9433	-9461	-9473	-9497
-9505	-9509	-9581	-9665	-9673	-9677	-9697	-9761	-9925	-9997