Some Explicit Formulae of NAF and its Left-to-Right Analogue

Dong-Guk Han[1], Tetsuya Izu[2], and Tsuyoshi Takagi[1]

[1] FUTURE UNIVERSITY-HAKODATE,
116-2 Kamedanakano-cho, Hakodate, Hokkaido, 041-8655, Japan
{christa,takagi}@fun.ac.jp
[2] FUJITSU LABORATORIES Ltd.,
4-1-1, Kamikodanaka, Nakahara-ku, Kawasaki, 211-8588, Japan
izu@labs.fujitsu.com

Abstract. Non-Adjacent Form (NAF) is a canonical form of signed binary representation of integers. We present some explicit formulae of NAF and its left-to-right analogue (FAN) for randomly chosen n-bit integers. Interestingly, we prove that the zero-run length appeared in FAN is asymptotically 16/7, which is longer than that of the standard NAF. We also apply the proposed formulae to the speed estimation of elliptic curve cryptosystems.

Keywords: *signed binary representation, non-adjacent form, Hamming weight, zero-run length.*

1 Introduction

In some exponentiation-based public-key cryptosystems including RSA and Elliptic Curve Cryptosystems (ECC), a binary representation of a given integer (which may be a secret in most cases) is commonly used as a standard technique. While a non-signed representation of an integer is unique, we have some ways for representing the integer in signed form. For example, an integer 13 can be represented in signed form such as $10\bar{1}01$, $100\bar{1}\bar{1}$, or $10\bar{1}1\bar{1}$, where $\bar{1}$ denotes -1. Such signed binary representations are especially useful in ECC, since inversions of arbitrary points can be obtained with almost free operations over elliptic curves. Some properties of such signed binary representations are related to the cost of an exponentiation. Especially, the number of nonzero bits (the Hamming weight) is important since this value rules the number of multiplications in the exponentiation. Thus analyzing signed representations implies a cost evaluate of exponentiations.

The non-adjacent form (NAF) is a well-known signed binary representation [Rei60]. A NAF of a positive integer a is an expression $a = \sum_{i=0}^{n-1} \nu_i 2^i$ where $\nu_i \in \{-1, 0, 1\}$, $\nu_{n-1} \neq 0$ and no two successive digits are nonzero, namely $\nu_i \cdot \nu_{i+1} = 0$ for $i = 0, 1, ..., n-2$ [Rei60]. Each integer a has a unique NAF representation denoted by $\textsc{Naf}(a)$. Moreover, $\textsc{Naf}(a)$ can be efficiently computed by right-to-left operations ([IEEE], for example). In [JY00], Joye-Yen proposed a left-to-right analogue "FAN" [1]. It is known that NAF and FAN can be generated by applying a sliding window method with width-2 to the Booth encoding [Boo51] in right-to-left and left-to-right, respectively [HKP+04,OSS+04]. Note that the Booth encoding was also introduced as the reversed binary representation by Knuth [Knu81, Exercise 4.1-27]. In this paper, the Booth encoding and FAN of an integer a are denoted by $\textsc{Booth}(a)$ and $\textsc{Fan}(a)$, respectively.

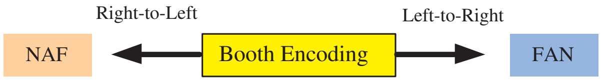

Fig. 1. A relation of the Booth encoding, NAF, and FAN.

NAF and FAN share some properties. Actually, they are generated by the similar manner as above, and the Hamming weights of NAF and FAN for the same integer are exactly same (Fact 1). In this paper, we prove an explicit formula derived from the Booth encoding (Theorem 1), which evaluates the average number of the Hamming weight of NAF and FAN representations as a first contribution. On the other hand, NAF and FAN have different properties. A fundamental observations is, while NAF does not have successive nonzero bits in the representation, FAN can have. Because of this difference, they have different significant length on average. We establish formulae for this value in Theorem 2. Moreover, we show the averaged length of zero runs in NAF and FAN in Theorem 3. In some implementations of ECC exponentiations, iterated elliptic curve doubling (wECDBL) is used for efficiency. With our analysis, a stricter evaluation of the averaged cost of exponentiations are possible. In fact, in ECC with 160-bit keys, FAN is about 15.97 multiplications (in a definition field) faster than NAF. Combined with a technique used in [SS01], FAN is about 317.47 multiplications faster than NAF.

An organization of this paper is as follows: section 2 defines some notations and the Booth encoding, NAF and FAN. In section 3, we prove some lemmas required for our theorems. Then, we establish an explicit evaluation formula for the Hamming weight in section 4. We also show evaluation formulae for the averaged significant length and the averaged zero runs (in NAF and FAN) in section 5, 6, respectively. Finally, in section 7, we apply our formulae to the cost evaluation of ECC exponentiations.

2 Preliminaries

2.1 Notations

For a given n-bit integer $a = \sum_{i=0}^{n-1} a_i 2^i$ with $a_i \in \{0, 1\}$, $a_{n-1} = 1$, we use the following notations:

[1] FAN comes from the reversed order of NAF.

– $\textsc{Booth}(a)$, $\textsc{Naf}(a)$, and $\textsc{Fan}(a)$ denote the Booth encoding, NAF, and FAN representation of the integer a respectively.
  - $\textsc{Booth}(a) := \sum_{i=0}^{n} \beta_i 2^i$ with $\beta_i \in \{-1, 0, 1\}$,
  - $\textsc{Naf}(a) := \sum_{i=0}^{n} \nu_i 2^i$ with $\nu_i \in \{-1, 0, 1\}$,
  - $\textsc{Fan}(a) = \sum_{i=0}^{n} \phi_i 2^i$ with $\phi_i \in \{-1, 0, 1\}$.
– $\mathcal{B}(n) := \{\textsc{Booth}(a) \mid 0 \le a \le 2^n - 1\}$.
  - $Case\_I\_\mathcal{B}(n) := \{\textsc{Booth}(a)$ with $\beta_n = 0 \mid 0 \le a \le 2^n - 1\}$.
  - $Case\_II\_\mathcal{B}(n) := \{\textsc{Booth}(a)$ with $(\beta_n, \beta_{n-1}) = (1, \bar{1}) \mid 0 \le a \le 2^n - 1\}$.
  - $Case\_III\_\mathcal{B}(n) := \{\textsc{Booth}(a)$ with $(\beta_n, \beta_{n-1}) = (1, 0) \mid 0 \le a \le 2^n - 1\}$.
– $\mathcal{N}(n) := \{\textsc{Naf}(a) \mid 0 \le a \le 2^n - 1\}$.
– $\mathcal{F}(n) := \{\textsc{Fan}(a) \mid 0 \le a \le 2^n - 1\}$.
– $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even.
– ε_n is the negligible function in n, namely for every constant $c \ge 0$ there exists an integer n_c such that $|\varepsilon_n| \le 1/n^c$ for all $n \ge n_c$.

2.2 Booth Encoding, NAF and FAN

The Booth encoding [Boo51] of an integer is defined as follows:

Definition 1 (Booth Encoding [OSS+04]). *The n-bit Booth encoding is an n-bit signed binary representation that satisfies the following two conditions:*

1. *Signs of adjacent nonzero bits (without considering zero bits) are opposite.*
2. *The most nonzero bit and the least nonzero bit are* 1 *and* $\bar{1}$*, respectively, unless all bits are zero.*

In [OSS+04], they showed a simple conversion method from an n-bit binary string to $(n+1)$-bit Booth encoding. Given an integer a, the Booth encoding of a is obtained by

$$\textsc{Booth}(a) = 2a \ominus a,$$

where $\ominus$ stands for a bitwise subtraction.

The non-adjacent form (NAF) also represents an integer in signed form. Since there is no successive nonzero bits in the representation, NAF is a standard technique for computing exponentiations [IEEE]. According to [HKP+04,OSS+04], NAF can be interpreted as a combination of the Booth encoding and a right-to-left sliding window method with width-2. For example, for an integer 13, we have $\textsc{Booth}(13) = 10\bar{1}1\bar{1}$. Then we divide $\textsc{Booth}(13)$ (as a string) into width-2 windows from right to left: 01, $0\bar{1}$, $1\bar{1}$ (the leftmost 0 was padded), and convert $1\bar{1}$ to 01 and $\bar{1}1$ to $0\bar{1}$, if exist. Thus we have $\textsc{Naf}(13) = 10\bar{1}01$.

FAN was introduced as a left-to-right analogue of NAF [JY00]. In fact, FAN can be also interpreted as a combination of the Booth encoding and a left-to-right sliding window method with width-2. For example, again, we divide the Booth encoding $\textsc{Booth}(13) = 10\bar{1}1\bar{1}$ into width-2 windows from left to right: 10, $\bar{1}1$, $\bar{1}0$ (the rightmost 0 was padded). Then, similarly to NAF, convert $1\bar{1}$ to 01 and $\bar{1}1$ to $0\bar{1}$, if exist. Thus we have $\textsc{Fan}(13) = 100\bar{1}\bar{1}$. Note that FAN can have successive nonzero bits unlike NAF.

Table 1 shows NAF, FAN, Booth, and non-signed binary representations of some positive integers.

Integer	Signed Binary			Non-signed
a	NAF(a)	FAN(a)	BOOTH(a)	Binary
0	00000	00000	00000	0000
1	00001	00001	$0001\bar{1}$	0001
2	00010	00010	$001\bar{1}0$	0010
3	$0010\bar{1}$	$0010\bar{1}$	$0010\bar{1}$	0011
4	00100	00100	$01\bar{1}00$	0100
5	00101	00101	$01\bar{1}1\bar{1}$	0101
6	$010\bar{1}0$	$010\bar{1}0$	$010\bar{1}0$	0110
7	$0100\bar{1}$	$0100\bar{1}$	$0100\bar{1}$	0111
8	01000	01000	$1\bar{1}000$	1000
9	01001	01001	$1\bar{1}01\bar{1}$	1001
10	01010	01010	$1\bar{1}1\bar{1}0$	1010
11	$10\bar{1}0\bar{1}$	$0110\bar{1}$	$1\bar{1}10\bar{1}$	1011
12	$10\bar{1}00$	$10\bar{1}00$	$10\bar{1}00$	1100
13	$10\bar{1}01$	$1001\bar{1}$	$10\bar{1}1\bar{1}$	1101
14	$100\bar{1}0$	$100\bar{1}0$	$100\bar{1}0$	1110
15	$1000\bar{1}$	$1000\bar{1}$	$1000\bar{1}$	1111

Table 1. NAF, FAN, Booth encoding representations of some integers

3 Lemmas

In this section, we prepare some lemmas required to prove our theorems.

3.1 Some Properties of Booth Encoding

Property 1. Due to the definition of Booth encoding, the number of Hamming weight of *Booth*(a) is always even, if the original integer a is positive.

Let $\langle 1\bar{1}\rangle^k$ be a pattern of nonzero bits in Booth encoding such that $\overbrace{1,\bar{1}}^{k},\ldots,\overbrace{1,\bar{1}}^{2},\overbrace{1,\bar{1}}^{1}$ (k-times) without considering zero bits between 1 and $\bar{1}$. Let $\#[\langle 1\bar{1}\rangle^k]$ be the total number of strings with $\langle 1\bar{1}\rangle^k$ pattern. For example, in $\mathcal{B}(3)\setminus\{0\}$, all elements except $1\bar{1}1\bar{1}$ have the same pattern $\langle 1\bar{1}\rangle^2$. Thus $\#[\langle 1\bar{1}\rangle^2] = 6$ and $\#[\langle 1\bar{1}\rangle^4] = 1$.

Lemma 1 (Pattern Lemma). $\mathcal{B}(n)$ *contains all possible representations with* $\langle 1\bar{1}\rangle^k$ *pattern for* $0 \le k \le \lceil n/2 \rceil$.

Proof. For $1 \le k \le \lceil n/2 \rceil$,

$$\#[\langle 1\bar{1}\rangle^0] = \binom{n+1}{0}, \quad \#[\langle 1\bar{1}\rangle^1] = \binom{n+1}{2}, \quad \ldots$$
$$\#[\langle 1\bar{1}\rangle^k] = \binom{n+1}{2k}, \quad \ldots \quad \#[\langle 1\bar{1}\rangle^{\lceil n/2 \rceil}] = \binom{n+1}{2\lceil n/2 \rceil}.$$

Thus $\sum_{k=0}^{\lceil n/2\rceil}\#[\langle 1\bar{1}\rangle^k] = \sum_{k=0}^{\lceil n/2\rceil}\binom{n+1}{2k} = 2^n$. This implies that there are 2^n different representations with $\langle 1\bar{1}\rangle^k$ pattern. As the total number of integers with n-bit is 2^n and Property 1, the assertion is proved. □

Lemma 2 (Classification Lemma). *$\mathcal{B}(n)$ can be divided into the following three cases;*

- *$Case\_I\_\mathcal{B}(n)$ with $\#[Case\_I\_\mathcal{B}(n)]=2^{n-1}$,*
- *$Case\_II\_\mathcal{B}(n)$ with $\#[Case\_II\_\mathcal{B}(n)] = 2^{n-2}$,*
- *$Case\_III\_\mathcal{B}(n)$ with $\#[Case\_III\_\mathcal{B}(n)] = 2^{n-2}$.*

Proof. From Property 1 and Lemma 1,

$$\#[Case\_I\_\mathcal{B}(n)] = \sum_{k=0}^{\lfloor \frac{n}{2}\rfloor}\binom{n}{2k} = 2^{n-1},$$

$$\#[Case\_II\_\mathcal{B}(n)] = \sum_{k=0}^{\lfloor \frac{n-1}{2}\rfloor}\binom{n-1}{2k} = 2^{n-2},$$

$$\#[Case\_III\_\mathcal{B}(n)] = \begin{cases}\sum_{k=0}^{\lfloor \frac{n-1}{2}\rfloor}\binom{n-1}{2k+1}, & \text{(if } n \text{ is even)}\\ \sum_{k=0}^{\lfloor \frac{n-1}{2}\rfloor-1}\binom{n-1}{2k+1}, & \text{(otherwise)}\end{cases} = 2^{n-2}.$$

□

Lemma 3 (Extension Lemma). *$\mathcal{B}(n)$ can be constructed from $\mathcal{B}(n-1)$ according to the following rules;*

- *$Case\_I\_\mathcal{B}(n) = \{(\beta_n = 0)\|(\beta_{n-1},\ldots,\beta_0) \mid (\beta_{n-1},\ldots,\beta_0) \in \mathcal{B}(n-1)\}$,*
- *$Case\_II\_\mathcal{B}(n)=\big\{(\beta_n,\beta_{n-1})=(1,\bar{1})\|(\beta_{n-2},\ldots,\beta_0) \mid (\beta_{n-2},\ldots,\beta_0) \in \mathcal{B}(n-2)\big\}$,*
- *$Case\_III\_\mathcal{B}(n)=\big\{(\beta_n,\beta_{n-1})=(1,0)\|(\beta_{n-2},\ldots,\beta_0) \mid (1,\beta_{n-2},\ldots,\beta_0)\in\{Case\_II\_\mathcal{B}(n-1) \cup Case\_III\_\mathcal{B}(n-1)\}\big\}$.*

Here, $x\|y$ denotes concatenation between two bit strings x and y.

Proof. From Property 1 and Lemma 1, 2, we can see that the assertion is true.

In the third case, in order to construct $Case\_III\_\mathcal{B}(n)$ the most bit 1 of the strings sampled from $\{Case\_II\_\mathcal{B}(n-1) \cup Case\_III\_\mathcal{B}(n-1)\}$ is changed to $\beta_{n-1} = 0$ and $\beta_n = 1$ is concatenated. Refer to Fig. 2.

Lemma 4 (Case II-Classification Lemma).
$\#\big[Case\_II\_\mathcal{B}(n)$ with $\#($the most consecutive nonzero bits$) = even\big] = \frac{2^{n-1}+\kappa_n}{3}$,
$\#\big[Case\_II\_\mathcal{B}(n)$ with $\#($the most consecutive nonzero bits$) = odd\big] = \frac{2^{n-2}-\kappa_n}{3}$,
where $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even. Especially,
$\#\big[Case\_II\_\mathcal{B}(n)$ with $\#($the most consecutive nonzero bits$) = 2\big] = 2^{n-3}$.

Proof. Straightforward because of Property 1 and Lemma 1.

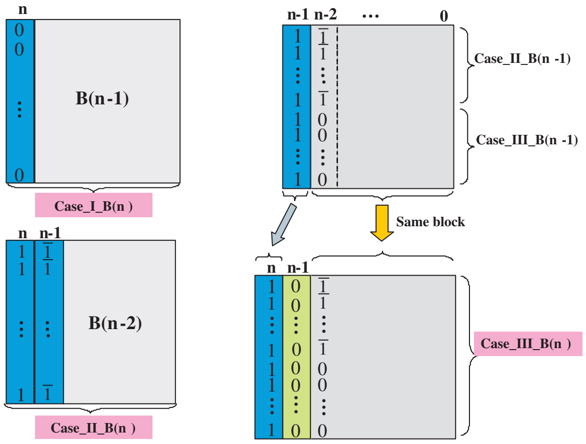

Fig. 2. Construct $\mathcal{B}(n)$ from $\mathcal{B}(n-1)$

3.2 Relations among Booth, NAF, and FAN

Lemma 5 (Adjacent Lemma). *The string with odd (>1) number of consecutive nonzero bits in Booth representations is converted into a sting with* 11 *or* $\bar{1}\bar{1}$ *in FAN representation.*

$$\overbrace{\dots 0\,\underbrace{1\bar{1}1\bar{1}\cdots 1\bar{1}1}_{\#odd}\,0\dots}^{Booth} \Rightarrow \overbrace{\dots 0\,\underbrace{0101\cdots 011}\,0\dots}^{FAN},$$

$$\overbrace{\dots 0\,\underbrace{\bar{1}1\bar{1}1\cdots \bar{1}1\bar{1}}_{\#odd}\,0\dots}^{Booth} \Rightarrow \overbrace{\dots 0\,\underbrace{0\bar{1}0\bar{1}\cdots 0\bar{1}\bar{1}}\,0\dots}^{FAN}$$

However, the even number of consecutive nonzero bits in Booth representations is not converted into 11 *or* $\bar{1}\bar{1}$.

Lemma 6 (Length Lemma). *For a given n-bit integer a, i.e. $a_{n-1}=1$,*

$$\begin{aligned} &L[\textsc{Booth}(a) \text{ with } (b_n, b_{n-1}) = (1,0)] \\ =\ &L[\textsc{Naf}(a)] = L[\textsc{Fan}(a)], \end{aligned}$$

$$\begin{aligned} &L[\textsc{Booth}(a) \text{ with } \#(\textit{the most consecutive nonzero bits}) = \textit{even}] \\ =\ &L[\textsc{Naf}(a)] + 1 = L[\textsc{Fan}(a)] + 1, \end{aligned}$$

$$\begin{aligned} &L[\textsc{Booth}(a) \text{ with } \#(\textit{the most consecutive nonzero bits}) = \textit{odd}\ (>1)] \\ =\ &L[\textsc{Naf}(a)] = L[\textsc{Fan}(a)] + 1. \end{aligned}$$

Here, $L[a]$ denotes the bit length of an integer a, for example, if $a = (10110)_2$ then $L[a] = 5$.

From the results of Lemma 6, we prove the following *NAF Carry Formula* of NAF.

Lemma 7 (NAF Carry Formula). *Assume that NAF is converted from integers of n bits. The carry at $(n+1)$-th bit of NAF occurs with probability*

$$C_n = \frac{1}{3} - \frac{\kappa_n}{3}\left(\frac{1}{2}\right)^n, \tag{1}$$

where $\kappa_n = 2$ for odd n and $\kappa_n = 1$ for even n.

Proof. Consider $\mathcal{B}(n)$. From Lemma 6, the elements of $Case\_III\_\mathcal{B}(n)$ and $Case\_II\_\mathcal{B}(n)$ with #(the most consecutive nonzero bits)= odd (> 1) are converted into NAF with a carry at $(n+1)$-bit. From Lemma 2 and 4, the total number of integers with a carry at $(n+1)$-bit is equal to

$$2^{n-2} + \frac{2^{n-2} - \kappa_n}{3} = \frac{2^n - \kappa_n}{3},$$

where $\kappa_n = 2$ for odd n and $\kappa_n = 1$ for even n. Thus the carry probability C_n is $\frac{1}{3} - \frac{\kappa_n}{3}\left(\frac{1}{2}\right)^n$.

4 Hamming Weight of NAF and FAN

This section shows an explicit evaluation formula for the Hamming weight of NAF and FAN. The following fact is fundamental for our discussion.

Fact 1 (Theorem 12. [JY00,HKP+04]) *The Hamming weight of NAF is exactly equal to that of FAN for each integer.*

Let $H(a)$ be the Hamming weight of $\textsc{Naf}(a)$. Let $\mathcal{H}(n)$ be the average Hamming of NAF for n-bit integers, which is defined by

$$\mathcal{H}(n) = \frac{\sum_{k=0}^{2^n-1} H(k)}{2^n}. \tag{2}$$

For example, $\mathcal{H}(2) = 1, \mathcal{H}(3) = \frac{11}{8}, \mathcal{H}(4) = \frac{7}{4}, \mathcal{H}(5) = \frac{67}{32}$. Then, we have the following theorem.

Theorem 1. *Let n be any integer larger than 1. The average Hamming weight of the NAF and FAN of n-bit integers is*

$$\mathcal{H}(n) = \frac{1}{3}n + \frac{4}{9} - \frac{\kappa_n + 3}{9}\left(\frac{1}{2}\right)^n, \tag{3}$$

where $\kappa_n = 2$ for odd n and $\kappa_n = 1$ for even n.

Proof. Due to Fact 1, we will only consider the average Hamming weight of NAF. Recall that NAF is generated by the right-to-left width-2 sliding window method to Booth. The plan of this proof is as follows;

1. First find the average Hamming weight of $\mathcal{B}(n)$. (Refer to Lemma 8)
2. Next find the average number of two consecutive nonzero bits appeared in $\mathcal{B}(n)$. (Refer to Lemma 9)
3. The wanted average Hamming weight of NAF is

$$[Result\ of\ Step\ 1] - [Result\ of\ Step\ 2].$$

$\square$

Lemma 8. *The average Hamming weight of Booth representations in $\mathcal{B}(n)$ is*

$$\mathcal{H}_{Booth}(n) = \frac{1}{2}n + \frac{1}{2}. \tag{4}$$

Proof. From Lemma 1, the total number of Hamming weight of Booth is

$$2 \cdot \binom{n+1}{2} + 4 \cdot \binom{n+1}{4} + \ldots + 2k \cdot \binom{n+1}{2k} + \ldots + 2\lceil n/2 \rceil \cdot \binom{n+1}{\lceil n/2 \rceil} = (n+1)2^{n-1}.$$

Therefore $\mathcal{H}_{Booth}(n) = \frac{(n+1)2^{n-1}}{2^n} = \frac{n}{2} + \frac{1}{2}$. $\square$

We investigate the average number of two consecutive nonzero bits appeared in the representation of Booth $\mathcal{B}(n)$. Indeed we prove the following theorem.

Lemma 9. *Assume that Booth is converted from integers of n bits. The average number of two consecutive nonzero bits appeared in the representation of Booth is*

$$A_n = \frac{1}{6}n + \frac{1}{18} + \frac{\kappa_n + 3}{18}\left(\frac{1}{2}\right)^{n-1}, \tag{5}$$

where $\kappa_n = 2$ for odd n and $\kappa_n = 1$ for even n.

Proof. Let B_n be the total number of two consecutive nonzero bits appeared in $\mathcal{B}(n)$. For example, we know $B_2 = 2$, $B_3 = 5$, $B_4 = 12$, and $B_5 = 29$.

Now we prove that the following relationship holds.

$$B_n = 2B_{n-1} + \frac{2^{n-1} - \kappa_n}{3}, \tag{6}$$

where $\kappa_n = 2$ if odd n and $\kappa_n = 1$ if even n. From Lemma 2, there are three cases: (1) the most bit is 0, (2) the most two bits are $1\bar{1}$, and (3) the most two bits are 10.

- In the case of (1), the number of two consecutive nonzero bits is B_{n-1}.
- In the case of (2), the lower bits are exactly equal to $\mathcal{B}(n-2)$, and the most two bits are always two consecutive nonzero bits. Therefore, the number two consecutive nonzero bits is $B_{n-2} + 2^{n-2}$. (Refer to Lemma 2 and 3.)

– In the case of (3), we know that it is related with the Case II and III of $\mathcal{B}(n-1)$. The number of two consecutive nonzero in $\mathcal{B}(n)$ derived from $Case\_III\_\mathcal{B}(n-1)$ is exactly equal to that of $Case\_III\_\mathcal{B}(n-1)$. However, there are some changes if the target part of $\mathcal{B}(n)$ which is derived from $Case\_II\_\mathcal{B}(n-1)$. That is if the number of the most consecutive nonzero bits is even in the $Case\_II\_\mathcal{B}(n-1)$ then the number of two consecutive nonzero bits is decreased by one in $\mathcal{B}(n)$. (Refer to Lemma 2 and 3.)
From Lemma 4, $\#[Case\_II\_\mathcal{B}(n-1)$ with $\#$(the most consecutive nonzero bits) = even] = $\frac{2^{n-2}+\kappa_n}{3}$, where $\kappa_n = 2$ if odd n and $\kappa_n = 1$ if even n. Therefore it is equal to $B_{n-1} - B_{n-2} - \left(\frac{2^{n-2}+\kappa_n}{3}\right)$.

Summing up those three values we obtain equation (6). From $A_n = B_n/2^n$ and equation (6), we obtain

$$A_n = A_{n-1} + \frac{1}{3}\left(\frac{1}{2} - \kappa_n\left(\frac{1}{2}\right)^n\right). \tag{7}$$

Then we know

$$\begin{aligned} A_n &= A_2 + \frac{1}{3}\sum_{i=3}^{n}\left(\frac{1}{2} - \kappa_n\left(\frac{1}{2}\right)^i\right) \\ &= \frac{1}{6}n + \frac{1}{18} + \frac{\kappa_n+3}{18}\left(\frac{1}{2}\right)^{n-1}, \end{aligned}$$

where $\kappa_n = 2$ for odd n and $\kappa_n = 1$ for even n. $\square$

5 Bit Length of NAF and FAN

This section shows an evaluation formulae for the averaged siginificant length of NAF and FAN, which are summarizes in the following theorem.

Theorem 2. *Let n be any integer larger than 1. The average significant length of NAF and FAN is*

$$\mathcal{LN}(n) = n - \frac{1}{3} + \left(-\frac{1}{2}n + \frac{1}{3\kappa_n}\right)\left(\frac{1}{2}\right)^n, \tag{8}$$

$$\mathcal{LF}(n) = n - \frac{1}{2}, \tag{9}$$

where $\kappa_n = 2$ for odd n and $\kappa_n = 1$ for even n.

Proof. First we estimate the average significant length of $\mathcal{F}(n)$ denoted as $\mathcal{LN}(n)$. From Lemma 2 and 6, $\#[\mathcal{F}(n)$ with $\phi_n = 1] = 2^{n-2}$. Similarly, $\#[\mathcal{F}(n)$ with the most significant bit $\phi_{i-1} = 1] = 2^{i-2} + 2^{i-3}$ for $i = 3, 4, ..., n$. Finally, there is only 1 whose length is 1 or 2. Therefore we have the following relationship.

$$
\begin{aligned}
2^n\mathcal{LF}(n) &= (n+1)2^{n-2} + \sum_{i=1}^{n-2}(i+2)(2^i + 2^{i-1}) + 2\cdot 1 + 1\cdot 1 \\
&= n2^n - 2^{n-1}, \\
\mathcal{LF}(n) &= n - \frac{1}{2}.
\end{aligned}
$$

Next we estimate the average significant length of NAF denoted as $\mathcal{LN}(n)$. From Lemma 6, the following equation holds.

$$
\begin{aligned}
2^n\mathcal{LN}(n) &= 2^n\mathcal{LF}(n) \\
&\quad + \sum_{i=4}^{n} \#[Case\_II\_\mathcal{B}(i) \ with \ \#(the\ most\ \ consecutive\ nonzero\ bits) = odd].
\end{aligned}
$$

From Lemma 4,

$$
\begin{aligned}
&\sum_{i=4}^{n} \#[Case\_II\_\mathcal{B}(i) \ with \ \#(the\ most\ \ consecutive\ nonzero\ bits) = odd] \\
&= \begin{cases} \sum_{i=1}^{\frac{n-3}{2}} \frac{2^{2i}-1}{3} + \sum_{i=1}^{\frac{n-3}{2}} \frac{2^{2i+1}-2}{3} = \frac{2^{n-1}}{3} - \frac{n}{2} + \frac{1}{6} & \text{(if } n \text{ is odd)} \\ \sum_{i=1}^{\frac{n-2}{2}} \frac{2^{2i}-1}{3} + \sum_{i=1}^{\frac{n-4}{2}} \frac{2^{2i+1}-2}{3} = \frac{2^{n-1}}{3} - \frac{n}{2} + \frac{1}{3} & \text{(if } n \text{ is even)} \end{cases} \\
&= \frac{2^{n-1}}{3} - \frac{n}{2} + \frac{1}{3\kappa_n},
\end{aligned}
$$

where $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even. Thus

$$
\begin{aligned}
2^n\mathcal{LN}(n) &= 2^n\left(n - \frac{1}{2}\right) + \frac{2^{n-1}}{3} - \frac{n}{2} + \frac{1}{3\kappa_n}, \\
\mathcal{LN}(n) &= n - \frac{1}{3} + \left(-\frac{1}{2}n + \frac{1}{3\kappa_n}\right)\left(\frac{1}{2}\right)^n.
\end{aligned}
$$

6 Zero Run Length of NAF and FAN

In this section, we investigate the averaged length of zero run regarding to NAF and FAN. For example, the average length of zero run for $10\bar{1}0\bar{1}01$ is 1. The corresponding FAN is $1100\bar{1}\bar{1}$ and its average length of zero run is 2. In general FAN has a longer zero run on average. Indeed we prove the following theorem.

Theorem 3. *The average zero run of NAF and FAN converted from n bits integers is equal to*

$$
\begin{aligned}
\frac{\mathcal{LN}(n) - \mathcal{H}(n)}{\mathcal{H}(n) - \frac{1}{2}} &= \frac{\frac{2}{3}n - \frac{7}{9} + \varepsilon_n}{\frac{1}{3}n - \frac{1}{18} + \varepsilon_n} = 2 + \mathcal{O}(n^{-1}), \\
\frac{\mathcal{LF}(n) - \mathcal{H}(n)}{\mathcal{H}(n) - \frac{1}{2} - \frac{E_n}{2^n}} &= \frac{\frac{2}{3}n - \frac{17}{18} + \varepsilon_n}{\frac{7}{24}n - \frac{5}{72} + \varepsilon_n} = \frac{16}{7} + \mathcal{O}(n^{-1}),
\end{aligned}
$$

respectively. Here we define

$$E_n = \frac{1}{24}n2^n - \frac{6+2\kappa_n}{9} + (\frac{1}{72}) \cdot 2^n,$$

where $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even, and ε_n is the negligible function in n.

Proof. At first we estimate the average zero run of NAF. The total number of zero appeared in NAF converted from n bits integers is $\mathcal{LN}(n)2^n - \mathcal{H}(n)2^n$. Because there is no consecutive nonzero bits, the number of zero runs can be estimated by Hamming weight, namely there are $\mathcal{H}(n)2^n - 2^{n-1}$ different zero runs, where 2^{n-1} is the number of NAF whose least bit is nonzero. Therefore, the average length of zero run for NAF is evaluated as follows:

$$\frac{\mathcal{LN}(n)2^n - \mathcal{H}(n)2^n}{\mathcal{H}(n)2^n - 2^{n-1}} = \frac{\mathcal{LN}(n) - \mathcal{H}(n)}{\mathcal{H}(n) - \frac{1}{2}} = \frac{\frac{2}{3}n - \frac{7}{9} + \varepsilon_n}{\frac{1}{3}n - \frac{1}{18} + \varepsilon_n} = 2 + \mathcal{O}(n^{-1}),$$

where ε_n is the negligible function in n. Next we estimate the average number of zero run for FAN. The number of two consecutive nonzero bits 11 and $\bar{1}1$ should be excluded from the number of different consecutive zeroes appeared in the denominator above. In the following we estimate the number these exceptional consecutive bits. Let E_n be the number of 11 and $\bar{1}1$ appeared in FAN converted from n bits integers. We know $E_4 = 2, E_5 = 6$, and $E_6 = 16$.

We use the three cases appeared in Lemma 2: (1) the most bit is 0, (2) the most two bits are $1\bar{1}$, and (3) the most two bits are 10. The estimation is similar to Lemma 9.

- In the case of (1), the number is E_{n-1}.
- In the case of (2), from Lemma 3 the target part of $\mathcal{B}(n)$ is $(1\bar{1})\|\mathcal{B}(n-2)$. From Lemma 5 we can see that only the strings such that $(\beta_{n-2}, \beta_{n-3}) = (1,0)$ in $\mathcal{B}(n-2)$ generate new 11 in $\mathcal{B}(n)$. Thus the number is $E_{n-2} + 2^{n-4}$.
- In the case of (3), the number of two consecutive nonzero bits in $\mathcal{B}(n)$ derived from $Case\_III\_\mathcal{B}(n-1)$ is exactly equal to that of $Case\_III\_\mathcal{B}(n-1)$. However, there are some changes if the target part of $\mathcal{B}(n)$ which is derived from $Case\_II\_\mathcal{B}(n-1)$ because #(the most consecutive nonzero bits) ≥ 2. We consider $Case\_II\_\mathcal{B}(n-1)$. From Lemma 5, we can derive the following results;
  - If #(the most consecutive nonzero bits) $= 2$ then there is no change of the number of two consecutive nonzero bits.
  - If #(the most consecutive nonzero bits) $=$ *even*(> 2) then new 11 is generated after conversion to FAN.
  - If #(the most consecutive nonzero bits) $=$ *odd* then 11 which was existed is disappeared after conversion to FAN.

Thus the number is

$$\begin{aligned}
&E_{n-1} - E_{n-2} \\
&+ \#\big[Case_II_B(n-1) \textit{ with } \#(\textit{the most consecutive nonzero bits}) = \textit{even } (> 2)\big] \\
&- \#\big[Case_II_B(n-1) \textit{ with } \#(\textit{the most consecutive nonzero bits}) = \textit{odd}\big] \\
&= E_{n-1} - E_{n-2} + \frac{4 - \kappa_n 2^{n-4}}{3\kappa_n},
\end{aligned}$$

where $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even. (Refer to Lemma 4.)

Therefore we have the following relationship:

$$E_n = 2E_{n-1} + \frac{\kappa_n 2^{n-3} + 4}{3\kappa_n},$$

where $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even.

Thus we obtain

$$E_n = \frac{1}{24} n 2^n - \frac{6 + 2\kappa_n}{9} + (\frac{1}{72}) \cdot 2^n,$$

where $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even.

Therefore, the average length of zero run for FAN is evaluated as follows:

$$\frac{\mathcal{LF}(n)2^n - \mathcal{H}(n)2^n}{\mathcal{H}(n)2^n - 2^{n-1} - E_n} = \frac{\frac{2}{3}n - \frac{17}{18} + \varepsilon_n}{\frac{7}{24}n - \frac{5}{72} + \varepsilon_n} = \frac{16}{7} + \mathcal{O}(n^{-1}).$$

7 Application to ECC

In this section we estimate the efficiency of scalar multiplication used for ECC with NAF or FAN. We assume that the scalar is a randomly chosen n-bit integer.

Let ECDBL and ECADD be the efficiency of computing elliptic doubling and addition, respectively. The average number of multiplication for computing scalar multiplications using NAF or FAN is estimated by

$$(\mathcal{LN}(n) - 1)\mathsf{ECDBL} + (\mathcal{H}(n) - 1)\mathsf{ECADD} \tag{10}$$

$$(\mathcal{LF}(n) - 1)\mathsf{ECDBL} + (\mathcal{H}(n) - 1)\mathsf{ECADD}. \tag{11}$$

From Theorem 2, the difference of average efficiency is equal to

$$\left(\frac{1}{6} + \left(-\frac{1}{2}n + \frac{1}{3\kappa_n}\right)\left(\frac{1}{2}\right)^n\right)\mathsf{ECDBL}, \tag{12}$$

where $\kappa_n = 2$ for odd n and $\kappa_n = 1$ for even n.

Let M, S, I denote the computation time of a multiplication, a squaring, and an inverse. We assume that $1S = 0.8M$ and $1I = 30M$. When we use a Jacobian coordinate for standard curves over prime field, ECDBL and ECADD require 8.8

multiplications (4M+6S) [CMO98]. For example, ECC with 160 bits ($n = 160$) using FAN is about 1.47 multiplications faster on average than that using NAF.

If ECDBL is repeatedly computed w times, we have an efficient variation, called wECDBL that can be computed with $4wM + (4w + 2)S$ [ITT+99].

The difference of average efficiency from Theorem 3 is equal to

$$\frac{n}{w_1} * (7.2 * w_1 + 1.6) - \frac{n}{w_2} * (7.2 * w_2 + 1.6), \tag{13}$$

where w_1 and w_2 denote the expected number of w for NAF and FAN respectively, and actually $w_1 = 1.987$ and $w_2 = 2.269$ for $n = 160$. Therefore the result equation (14) is 15.97. This implies that FAN is about 15.97 multiplications faster than NAF.

If we consider Sakai-Sakurai method [SS01] of multidoubling for Weierstrass elliptic curves in terms of affine coordinates that can be computed with $(4w + 1)M + (4w + 1)S + I$. When $n = 160$ the difference of average efficiency is

$$\frac{n}{w_1} * (7.2 * w_1 + 31.8) - \frac{n}{w_2} * (7.2 * w_2 + 31.8) = 317.47,$$

i.e. FAN is about 317.47 multiplications faster. Here, $w_1 = 1.987$ and $w_2 = 2.269$ for $n = 160$.

Acknowledgements

Dong-Guk Han was supported by the Korea Research Foundation Grant. (KRF-2005-214-C00016)

References

[Boo51] A. Booth, "A signed binary multiplication technique", Journ. Mech. and Applied Math., 4(2), pp.236-240, 1951.

[CMO98] H. Cohen, A. Miyaji, and T. Ono, "Efficient elliptic curve exponentiation using mixed coordinates," *Asiacrypt'98*, LNCS 1514, pp.51-65, Springer-Verlag, 1998.

[HKP+04] C. Heuberger, R. Katti, H. Prodinger, and X. Ruan, "The alternating greedy expansion and applications to left-to-right algorithms in cryptography", To appear in Theor. Comput. Sci. A. Preprint version is available at http://www.opt.math.tu-graz.ac.at/ cheub/publications/alg1.pdf

[IEEE] IEEE 1363-2000, IEEE Standard Specifications for Public-Key Cryptography, 2000.

[ITT+99] K. Itoh, M. Takenaka, N. Torii, S. Temma, and Y. Kurihara, "Fast Implementation of Public-Key Cryptography ona DSP TMS320C6201," *CHES'99*, LNCS 1717, pp.61-72, Springer-Verlag, 1999.

[JY00] M. Joye, and S.-M. Yen, "Optimal Left-to-Right Binary Signed-digit Exponent Recoding", IEEE Transactions on Computers 49(7), pp.740-748, 2000.

[JY02] M. Joye, and S.-M. Yen, “New minimal modified radix-r representation with applications to smart-cards”, Proc. of PKC 2002, LNCS 2274, pp. 375-383, Springer-Verlag, 2002.

[Knu81] D.E. Knuth, ”*The Art of Computer Programming, vol. 2, Seminumerical Algorithms*, 2nd ed., Addison-Wesley, Reading, Mass, 1981.

[OSS+04] K. Okeya, K. Schmidt-Samoa, C. Spahn, and T. Takagi, “Signed Binary Representations Revisited”, IACR Cryptology ePrint Archive, 2004. Available at http://eprint.iacr.org/2004/195

[Rei60] G.W. Reitwiesner, *Binary arithmetic*, Advances in Computers, vol.1, pp.231-308, 1960.

[SS01] Y. Sakai and K. Sakurai, “On the Power of Multidoubling in Speeding Up Elliptic Scalar Multiplication,” *SAC'01*, LNCS 2259, pp.268-283, Springer-Verlag, 2001.