PAPER Special Section on Cryptography and Information Security

Radix-r Non-Adjacent Form and Its Application to Pairing-Based Cryptosystem*

Tsuyoshi TAKAGI[†a)], ***Member***, **David REIS, Jr.**[††b)], **Sung-Ming YEN**[†††c)], ***and*** **Bo-Ching WU**[†††d)], ***Nonmembers***

SUMMARY Recently, the radix-3 representation of integers is used for the efficient implementation of pairing based cryptosystems. In this paper, we propose non-adjacent form of radix-r representation (rNAF) and efficient algorithms for generating rNAF. The number of non-trivial digits is $(r-2)(r+1)/2$ and its average density of non-zero digit is asymptotically $(r-1)/(2r-1)$. For $r=3$, the non-trivial digits are $\{\pm 2, \pm 4\}$ and the non-zero density is 0.4. We then investigate the width-w version of rNAF for the general radix-r representation, which is a natural extension of the width-w NAF. Finally we compare the proposed algorithms with the generalized NAF (gNAF) discussed by Joye and Yen. The proposed scheme requires a larger table but its non-zero density is smaller even for large radix. We explain that gNAF is a simple degeneration of rNAF—we can consider that rNAF is a canonical form for the radix-r representation. Therefore, rNAF is a good alternative to gNAF.

key words: *non-adjacent form, radix-r representation, signed window method, elliptic curve cryptosystem, pairing based cryptosystem*

1. Introduction

Pairing based cryptosystems [15] are able to construct very attractive applications in cryptography, e.g., tripartite Diffie-Hellmann scheme [14], ID-based cryptosystems [4], short digital signature [5], etc. Barreto et al. and Galbraith et al. showed efficient algorithms for pairing based cryptosystems over supersingular elliptic curves [1], [8]. Several efficient arithmetic for elliptic curve with characteristic three have been investigated [2], [11], [13], [18], [20]. Particularly, the radix-3 representation of integers can be used for efficient implementation of these algorithms with characteristic three. Recently, Duursam and Lee proposed an efficient implementation of Tate pairing for hyper-elliptic curves constructed over general characteristic r [7]. In this case, the radix-r representation is utilized for the efficient implementation of the pairing based cryptosystems.

In order to achieve faster scalar multiplication, we have to exploit an efficient class of the radix-r representation, i.e., the number of non-zero digits is smaller. The generalized non-adjacent form (gNAF) is known as an efficient class of radix-r representation [6], [16]. The average density of non-zero digits (non-zero density) of the gNAF is asymptotically $\frac{r-1}{r+1}$ with $(r-1)$ pre-computed points among which $(r-2)$ points are non-trivial. For example, $r=3$ attains 0.5 non-zero density with 1 non-trivial pre-computed point. On the other hand, the non-zero density of the standard radix-r representation is $\frac{r-1}{r}$ with the same non-trivial pre-computed point, which is 0.67 for $r=3$. Therefore, the gNAF is able to improve the efficiency of computing the paring based cryptosystem, especially scalar multiplication. On the other hand, we can achieve lower non-zero density, if we have a larger digit set. Recently, Phillips and Burgess presented a generalized sliding window method for the radix-r representation [19]. The canonical form using a larger digit set for the binary representation is the width-w non-adjacent form (wNAF) [3], [17], [21]. However, there is no literature that reports a variation of wNAF for the radix-r representation.

In this paper, we present an efficient class of radix-r representation, called radix-r non-adjacent form (rNAF). The proposed algorithm is a natural extension of the classical non-adjacent form (NAF) for binary representation [3], [12], namely the adjacent bits are not simultaneously non-zero. In order to construct rNAF, we define the digit set D_r, whose elements are smaller than $\frac{r^2-1}{2}$ and are not divisible by the radix r. We prove that each integer can be uniquely represented by rNAF and the Hamming weight of rNAF representation is minimal among all signed radix-r representations using digit set D_r. We also prove that the average density of non-zero digits for rNAF is asymptotically $\frac{r-1}{2r-1}$ with $\frac{(r-2)(r+1)}{2}$ non-trivial digits. For $r=3$, we have 0.4 non-zero density with 2 non-trivial pre-computed points, which is faster than the radix-3 gNAF but requires 1 more point. Moreover, we extended this result to the width-w case. Our construction is similar to wNAF proposed by Solinas [21]. We prove that the proposed width-w rNAF has asymptotically $\frac{r-1}{w(r-1)+1}$ non-zero density with $\frac{r^w-r^{w-1}-2}{2}$ non-trivial digits. Finally, we investigate the relationship between the radix-r gNAF and rNAF. Interestingly, we show that gNAF is a degenerate form of rNAF, namely if some conversions for rNAF are ignored, then we can generate gNAF. Based on this observation we present a simple gener-

Manuscript received March 7, 2005.
Manuscript revised June 24, 2005.
Final manuscript received September 5, 2005.

[†]The author is with the School of Systems Information Science, Future University-Hakodate, Hakodate-shi, 041-8655 Japan.
[††]The author is with the School of Electrical and Computer Engineering, State University of Campinas, Caixa Postal 6101, Brazil.
[†††]The authors are with the Laboratory of Cryptography and Information Security (LCIS), Department of Computer Science and Information Engineering, National Central University, Taiwan 320, R.O.C.

*The preliminary version of this paper was presented at the 7th Information Security Conference (ISC 2004), held in Palo Alto.

a) E-mail: takagi@fun.ac.jp
b) E-mail: davidjr@dca.fee.unicamp.br
c) E-mail: yensm@csie.ncu.edu.tw
d) E-mail: wubq@csie.ncu.edu.tw

DOI: 10.1093/ietfec/e89–a.1.115

ation algorithm and a simple proof of the non-zero density for gNAF.

This paper is organized as follows: In Sect. 2 we shortly review gNAF. In Sect. 3 the radix-r NAF is proposed and some properties of rNAF are investigated. In Sect. 3.2 we develop the width-w version of rNAF. In Sect. 4 the proposed rNAF is compared with the previous schemes and the relationship between rNAF and gNAF is discussed. In Sect. 5 we apply the proposed scheme to the scalar multiplication used for the pairing-based cryptosystems and present some timings. In Sect. 6 we state the concluding remarks.

2. Generalized Non-Adjacent Form (gNAF)

In this section we discuss some known properties related to the radix-r representation.

An integer d is uniquely represented using the radix-r representation, namely

$$d = \sum_{j=0}^{n-1} d_j r^j, \quad d_j \in \{0, 1, \ldots, r-1\}. \tag{1}$$

We denote by $d = (d_{n-1}, \ldots, d_1, d_0)$ the radix-r representation of d. Here d_j and n are called the j-th digit and the digit length of the radix-r representation for d. The number of non-zero digits is called the Hamming weight of the radix-r representation of d. The average density of non-zero digits of the radix-r representation is obviously $\frac{r-1}{r}$.

If digit d_j is allowed to take a negative value (i.e., $d_j \in \{0, \pm 1, \ldots, \pm(r-1)\}$), it is called signed radix-r representation. In general the signed radix-r representation is not unique. However, the generalized non-adjacent form (gNAF) can uniquely represent each integer and is an optimal class for the signed radix-r representation [6]. gNAF is the signed radix-r representation which satisfied the following two conditions.

(1) $|d_i + d_{i+1}| < r$ for all i,
(2) $|d_i| < |d_{i+1}|$ if $d_i d_{i+1} < 0$.

If we choose $r = 2$, then the definition is equal to the classical NAF for binary representation. It is known that gNAF has the minimal Hamming weight among all signed radix-r representation with digit set $\{0, \pm 1, \ldots, \pm(r-1)\}$. The average density of the non-zero digits (non-zero density) is asymptotically $\frac{r-1}{r+1}$. For $r = 3$, the non-zero density is 0.5.

For a given radix-r representation of integer d, gNAF is generated by computing $(r+1)d \mathbin{\dot{-}} d$, where the minus $\dot{-}$ is a digit-wise subtraction of $(r+1)d$ by d. This construction is a generalization of Reitwiesner algorithm for generating NAF [12]. There is a carry for computing the radix-r representation of $(r+1)d$, and thus this algorithm is not computed in the left-to-right approach. Joye and Yen proposed a left-to-right based algorithm for generating a signed radix-r representation with same non-zero density and digit set as those of gNAF [16].

3. Radix-r Non-Adjacent Form (rNAF)

In this section, we define the radix-r non-adjacent form (rNAF) representation and prove some properties of rNAF.

We define the rNAF in the following.

Definition 1. *A signed radix-r representation $d = (d_{n-1}, \ldots, d_1, d_0)$ is called radix-r non-adjacent form (rNAF) if it satisfies the following conditions.*
(1) $d_j d_{j-1} = 0$ for all $j = 0, 1, \ldots, n$, where we define $d_n = d_{-1} = 0$.
(2) $d_j \in D_r = \{0, \pm 1, \pm 2, \ldots, \pm\lfloor \frac{r^2-1}{2} \rfloor\}$ $\backslash \{\pm 1r, \pm 2r, \ldots, \pm\lfloor \frac{r-1}{2} \rfloor r\}$.
(3) The leftmost non-zero digit is positive.

This definition is a natural extension of non-adjacent form for binary string to the radix-r representation. D_r is called the digit set of the rNAF. The set D_r is generated by right-to-left conversion of two consecutive unsigned digits (e_j, e_{j-1}) (for $e_{j-1} \neq 0$):

if $e_j r + e_{j-1} < \frac{r^2}{2}$, then $(0, e_j r + e_{j-1})$,
else $(1, 0, (e_j r + e_{j-1}) - r^2)$.

Therefore, all possible digits (except "0") are $(e_j r + e_{j-1})$ and $((e_j r + e_{j-1}) - r^2)$ for $e_j \in \{0, 1, \ldots, r-1\}$ and $e_{j-1} \in \{1, \ldots, r-1\}$, which are equal to $D_r \setminus \{0\}$. If r is an odd integer, then there are r^2 elements in $\{0, \pm 1, \pm 2, \ldots, \pm\lfloor \frac{r^2-1}{2} \rfloor\}$ and $r-1$ elements in $\{\pm 1r, \pm 2r, \ldots, \pm\lfloor \frac{r-1}{2} \rfloor r\}$, respectively. So, there are totally $r^2 - (r-1) = r^2 - r + 1$ elements in the set D_r. On the other hand, if r is an even integer, then there are $r^2 - 1$ elements in $\{0, \pm 1, \pm 2, \ldots, \pm\lfloor \frac{r^2-1}{2} \rfloor\}$ and $r-2$ elements in $\{\pm 1r, \pm 2r, \ldots, \pm\lfloor \frac{r-1}{2} \rfloor r\}$, respectively. So, there are totally $(r^2 - 1) - (r-2) = r^2 - r + 1$ elements in D_r.

Note that if we choose $r = 2$, then D_r is just the digits of NAF for binary string, namely $\{0, \pm 1\}$. We can prove the following theorem:

Theorem 1. *(1) Every positive integer d has a unique rNAF representation.*
(2) The rNAF representation of d has the smallest Hamming weight among all signed representations of d with digit set D_r.

Proof. We start with the proof for (1). Assume that r is odd (the even case can be similarly proven).

We prove it by induction of digit length n for the unique unsigned radix-r representation $d = (e_{n-1}, \ldots, e_1, e_0)$. For $n = 2$, d is uniquely represented by

$$0 = (0, 0), 1 = (0, 1), 2 = (0, 2), \ldots,$$
$$r - 1 = (0, r-1),$$
$$r = (1, 0), r + 1 = (0, r+1), \ldots,$$
$$r + (r-1) = (0, 2r-1),$$
$$\ldots$$
$$k_r r = (k_r, 0), k_r r + 1 = (0, k_r r + 1), \ldots,$$

$$k_r r + k_r = \left(0, \left\lfloor \frac{r^2-1}{2} \right\rfloor\right),$$
$$k_r r + (k_r + 1) = \left(1, 0, -\left\lfloor \frac{r^2-1}{2} \right\rfloor\right), \ldots,$$
$$k_r r + (r-1) = (1, 0, k_r r + r - 1 - r^2),$$
$$\ldots$$
$$(r-1)r = (r-1, 0), (r-1)r + 1 = (1, 0, -r+1), \ldots,$$
$$(r-1)r + (r-1) = (1, 0, -1).$$

where $k_r = \lfloor \frac{r-1}{2} \rfloor$ and we have $k_r r + k_r = \lfloor \frac{r^2-1}{2} \rfloor$. Note that the radix-$r$ representation of 2-digit integers can be uniquely represented by 3-digit rNAF (the leftmost digit is $\{0, 1\}$).

We assume that the radix-r representation of n-digit integers can be uniquely represented by $(n+1)$-digit rNAF (the most significant digit is $\{0, 1\}$). Then we try to prove that it is also true for $(n+1)$-digit integers. Let $d = (e_n, e_{n-1}, \ldots, e_0)$ be the unique unsigned radix-r representation of $(n+1)$-digit integer d. From the assumption, the first n-digit $(e_{n-1}, \ldots, e_0)$ has the unique rNAF representation $(b_n, b_{n-1}, \ldots, b_1, b_0)$. Assume that $b_{n-1} = 0$ holds, then the rNAF representation of d is $(a_{n+1}, a_n, a_{n-1}, b_{n-2}, \ldots, b_1, b_0)$, where $a_{n+1} = 1, a_n = 0, a_{n-1} = 0$ if $e_n + b_n = r$, otherwise $a_{n+1} = 0, a_n = e_n + b_n, a_{n-1} = 0$. Assume that $b_{n-1} \neq 0$ holds, then the rNAF representation of d is $(a_{n+1}, a_n, a_{n-1}, b_{n-2}, \ldots, b_1, b_0)$, where $a_{n+1} = a_n = 0, a_{n-1} = re_n + b_{n-1}$ if $re_n + b_{n-1} < \frac{r^2}{2}$, otherwise $a_{n+1} = 1, a_n = 0, a_{n-1} = re_n + b_{n-1} - r^2$. The representation of last three digits (a_{n+1}, a_n, a_{n-1}) is obviously unique due to the definition of rNAF. Thus the rNAF representation of d for $(n+1)$-digit integers is unique. Consequently, all positive integers can be uniquely represented by rNAF representation.

Next we prove assertion (2). For a give integer d, we assume that there is a radix-r representation $R(d)$ of d with digit set D_r, whose Hamming weight is smaller than that of rNAF representation of d. Then there is a non-zero digit a_i of rNAF representation of d, which should be converted to zero in representation $R(d)$, namely there are some non-zero digits c_j such that $a_i r^i = \sum_j c_j r^j$, where $i \neq j$ and $c_j \in D_r$. However, there is no solution c_j for $a_i r^i = \sum_j c_j r^j$, because $\sum_j c_j r^j \bmod r^i \neq 0$ and $a_i \neq 0 \bmod r$. Consequently, the rNAF representation has minimal Hamming weight among all radix-r representations with digit set D_r. □

3.1 Proposed Generation Algorithm for rNAF

In this section, we explain the proposed algorithm that generates the rNAF from an integer or a usual radix-r representation.

The notation 'mods' stands for the signed modulo, namely d mods r is equal to $(d \bmod r) - r$ if $(d \bmod r) \geq r/2$, otherwise $(d \bmod r)$. Note that the set of all possible digits in Step 2.2 is exactly equal to D_r, because we eliminate the integers divisible by r at Step 2.1. The computation of $d \leftarrow d - cd_i$ causes a carry +1 if cd_i is a negative digit. At Step 2.3, we lift to the next digit of the rNAF representation of d.

Algorithm 1: Proposed Algorithm (Integer to rNAF)

Input : An integer d.
Output: The rNAF of d: $cd = (\ldots, cd_1, cd_0)$.

$i \leftarrow 0$;
while $d > 0$ **do**
  if $d \bmod r = 0$ **then** $cd_i \leftarrow 0$;
  else
    $cd_i \leftarrow d \ mods \ r^2$;
    $d \leftarrow d - cd_i$;
  end
  $d \leftarrow d/r$;
  $i \leftarrow i + 1$;
end
return $(\ldots, cd_1, cd_0)$;

Next we investigate the average density of non-zero digits (non-zero density) appeared in rNAF representation for $n \to \infty$. It is obvious that the non-zero density of rNAF is smaller than that of the usual radix-r representation, i.e., $\frac{r-1}{r}$. Indeed, we prove the following theorem.

Theorem 2. *The average non-zero density of the rNAF is asymptotically* $\frac{r-1}{2r-1}$. *The number of non-trivial digits (except* $\{0, \pm 1\}$ *and ignoring their sign) is* $\frac{(r-2)(r+1)}{2}$.

Proof. We investigate the distribution of each digit after the conversion $d_{i+1} r + d_i$ mods r. If non-zero digit d_i appears, the next digit d_{i+1} is always zero. Then there are two cases $(d_i) = (0)$ and $(d_{i+1}, d_i) = (0, x)$ with non-zero digit x. If case $(0, x)$ with a negative digit x appears, then there is a carry +1 to the next bits. The carry propagates to the higher bits. However, we can assume that each digit of radix-r representation d is randomly distributed in $\{0, 1, \ldots, r-1\}$, namely each digit of d appear with probability $1/r$. We can also assume that each digit of $d+1$ appears with probability $1/r$, because we deal with the asymptotical estimation. Therefore, the zero digit with probability $1/r$ and non-zero digit appears with $(r-1)/r$ after both cases (0) and $(0, x)$. Thus, the Markov chain of the two case $(0), (0, x)$ is as follows:

$$\begin{pmatrix} (0) & : & 1/r & (r-1)/r \\ (0, x) & : & 1/r & (r-1)/r \end{pmatrix}.$$

This Markov chain is aperiodic and irreducible, and thus there is the stationary distribution: $((0), (0, x)) = (1/r, (r-1)/r)$. Thus non-zero digit asymptotically appears $r-1$ out of $1 + 2(r-1)$. Consequently, we prove the assertion about the non-zero density. Next, D_r has $(r^2-1)-(r-2) = r^2-r+1$ elements and the non-zero digits always have their opposite sign. Therefore second assertion is true. □

Note that if we choose the classical binary case $r = 2$, then we obtain the famous non-zero density of NAF, namely $1/3$.

3.2 Extension to Higher Width

We define the width-w radix-r non-adjacent form (wrNAF) in the following.

Definition 2. *A signed radix-r representation* $d = (d_{n-1}, \ldots, d_1, d_0)$ *is called the width-w radix-r non-adjacent form (wrNAF) if it satisfies the following conditions.*
(1) there is at most 1 non-zero digit among any w adjacent digits
(2) $d_j \in D_{w,r} = \{0, \pm 1, \pm 2, \ldots, \pm\lfloor \frac{r^w-1}{2} \rfloor\} \setminus \{\pm 1r, \pm 2r, \ldots, \pm\lfloor \frac{r^{w-1}-1}{2} \rfloor r\}$.
(3) the leftmost non-zero digit is positive.

This definition is a natural extension of width-w non-adjacent form for binary string to the radix-r representation. The set $D_{w,r}$ is generated by right-to-left conversion of w consecutive unsigned digits $(e_{j+w-1}, \ldots, e_{j+1}, e_j)$ (for $e_j \neq 0$):

$$\text{let } e_w = e_{j+w-1}r^{w-1} + \ldots + e_{j+1}r + e_j,$$
$$\text{if } e_w < \tfrac{r^w}{2}, \text{ then } (\underbrace{0, \ldots, 0}_{w-1}, e_w),$$
$$\text{else } (1, \underbrace{0, \ldots, 0}_{w-1}, e_w - r^w).$$

Therefore all possible digits (except "0") are (e_w) and $(e_w - r^w)$ for $e_j \in \{1, \ldots, r-1\}$ and $e_{j+1}, \ldots, e_{j+w-1} \in \{0, 1, \ldots, r-1\}$, which are equal to $D_{w,r} \setminus \{0\}$. The number of elements in $D_{w,r}$ is $r^w - r^{w-1} + 1$. Note that if we choose $r = 2$, then $D_{w,r}$ is just the digits of NAF for binary string, namely $\{0, \pm 1, \pm 3, \ldots, \pm(2^{w-1} - 1)\}$.

We can prove the following theorem:

Theorem 3. *(1) Every positive integer d has a unique wrNAF representation.*
(2) The wrNAF representation of d has the smallest Hamming weight among all signed representations for d with digit set $D_{w,r}$.

Proof. The proof of these assertions is similar to that of Theorem 1. □

In the following, we explain the proposed algorithm that generates the width-w rNAF (i.e., wrNAF) from an integer or from a usual radix-r representation.

This is a simple generalization of rNAF to the width-w approach. The difference from the width-2 case is the signed modulus r^w operation at Step 2.2. This algorithm is also a natural extension of the wNAF generation algorithm proposed by Solinas [21].

We can prove the following theorem about the non-zero density of the wrNAF.

Theorem 4. *The non-zero density of the wrNAF is asymptotically* $\frac{r-1}{w(r-1)+1}$. *The number of non-trivial digits (except* $\{0, \pm 1\}$ *and ignoring their sign) is* $\frac{r^w - r^{w-1} - 2}{2}$.

Algorithm 2: Proposed Algorithm (Integer to wrNAF)

Input : An integer d and width w.
Output: The wrNAF of d: $wcd = (\ldots, wcd_1, wcd_0)$.

```
i ← 0;
while d > 0 do
    if d mod r = 0 then wcd_i ← 0;
    else wcd_i ← d mods r^w;
    d ← d − wcd_i;
    d ← d/r;
    i ← i + 1;
end
return (..., wcd_1, wcd_0);
```

Proof. The proof is similar to the case of $w = 2$. The only difference is that we deal with two statuses $(d_i) = (0)$ and status $(\underbrace{0, \ldots, 0}_{w-1}, x)$. The Markov chain of the two statuses $(0), (0, \ldots, 0, x)$ is as follows:

$$\begin{pmatrix} (0) & : & 1/r & (r-1)/r \\ (0, \ldots, 0, x) & : & 1/r & (r-1)/r \end{pmatrix}.$$

This Markov chain is aperiodic and irreducible, and thus there is the stationary distribution: $((0), (0, \ldots, 0, x)) = (1/r, (r-1)/r)$. Thus non-zero digit asymptotically appears $r - 1$ out of $1 + w(r-1)$. The second assertion is trivial from the definition of $D_{w,r}$. Consequently, we prove the theorem. □

4. Comparisons

Arithmetic weight (or non-zero density) of the representation of secret key usually reflects performance of an implementation for cryptographic operation, e.g., scalar multiplication over elliptic curve or exponentiation over finite group. Performance comparisons of the proposed rNAF and wrNAF with gNAF and the well known sliding window technique will be given.

4.1 Comparison with gNAF

The gNAF representation is an approach to reduce the arithmetic weight in order to enhance the performance of computing scalar multiplication. In the following comparison, we consider only the case of width-2 rNAF since existing gNAF does not consider a width larger than two.

The gNAF achieves $\frac{r-1}{r+1}$ non-zero density recoding and needs to store $r - 2$ non-trivial precomputed values within a table. On the other hand, the proposed rNAF has $\frac{r-1}{2r-1}$ asymptotical non-zero density and needs to store $\frac{(r-2)(r+1)}{2}$ non-trivial precomputed values. Numerical enumeration of the above comparisons are provided in Table 1. The result is that the proposed rNAF has better performance for implementing scalar multiplication with the cost of additional storage space. For practical applications, a noticeable computational speedup with only one additional precomputed

Table 1 Comparisons of gNAF and the proposed rNAF.

	gNAF		rNAF	
radix	non-zero density	number of non-trivial digits	non-zero density	number of non-trivial digits
2	$\frac{1}{3} \approx 0.333$	0	$\frac{1}{3} = 0.333$	0
3	$\frac{1}{2} = 0.5$	1	$\frac{2}{5} = 0.4$	2
4	$\frac{3}{5} = 0.6$	2	$\frac{3}{7} = 0.428$	5
5	$\frac{2}{3} \approx 0.666$	3	$\frac{4}{9} \approx 0.444$	9
6	$\frac{5}{7} \approx 0.714$	4	$\frac{5}{11} \approx 0.454$	14

value is possible by selecting $r = 3$. In this case, non-trivial digits $\{\pm 2, \pm 4\}$ are selected and a 0.4 non-zero density recoding is achieved. By using gNAF, a 0.5 non-zero density recoding is obtained with non-trivial digit $\{\pm 2\}$. Note that we count the number of non-trivial digits ignoring their sign.

4.2 Comparison with Sliding Window Technique

Sliding window technique [10], [19], [22] is an enhanced windowing technique by exploiting space-time trade-off in order to speedup scalar multiplication or exponentiation computations. With the same characteristic that a larger table with appropriate precomputed values can lead to a reduction of computational load. The conventional sliding window technique for binary representation can lead to a generalized form for any radix r larger than 2, and we called this the generalized sliding window form (gSWF).

We describe the width-w sliding window method for the radix-r representation (gSWF).

We scan the digits of the radix-r representation from the most significant bit, and if a non-zero digit appears, then we convert the w-consecutive digits using the following conversion table $\mathcal{T}_{SW}$:

$$
\begin{aligned}
&(1, 0, \ldots, 0) \to (1, 0, \ldots, 0), \ldots, \\
&(r-1, 0, \ldots, 0) \to (r-1, 0, \ldots, 0), \\
&(1, 1, 0, \ldots, 0) \to (0, r+1, 0, \ldots, 0), \ldots, \\
&(r-1, r-1, 0, \ldots, 0) \to (0, r^2-1, 0, \ldots, 0), \\
&\quad \ldots \\
&(r-1, \ldots, r-1, 1) \to (0, \ldots, 0, r^w - r + 1), \ldots, \\
&(r-1, \ldots, r-1, r-1) \to (0, \ldots, 0, r^w - 1).
\end{aligned}
$$

Then, we convert the radix-r representation to gSWF as Algorithm 3.

Each integer is uniquely converted to gSWF by this algorithm. We can prove that the number of non-trivial digits of gSWF is $r^w - r^{w-1} - 1$ (excluding $\{1\}$) and the average density of non-zero digits of gSWF is asymptotically $\frac{r-1}{(r-1)w+1}$.

In the proposed wrNAF, larger width w may reduce the non-zero density. Similarly, in the gSWF, a larger windowing width (usually denoted as w) will also reduce the computational load by reducing the non-zero density. However, storage space for both the wrNAF and the gSWF increase

Algorithm 3: Proposed Algorithm (Radix-r to gSWF)

Input : An integer in radix-r representation $d = (d_{n-1}, \ldots, d_1, d_0)$ and width-w.
Output: The width-w SW chain of d: $swd = (swd_{n-1}, \ldots, swd_1, swd_0)$.

```
i ← n − 1;
d_0 ← 0, ..., d_{−w+1} ← 0;
while i > 0 do
    if d_i = 0 then
        swd_i ← 0;
        i ← i − 1;
    end
    else
        (swd_i, ..., swd_{i−w+1}) ← T_SW(d_i, ..., d_{i−w+1});
        i ← i − w;
    end
end
return (swd_{n−1}, ..., swd_1, swd_0);
```

Table 2 Comparisons of sliding window technique and the proposed wrNAF.

	gSWF		wrNAF	
(r, w)	non-zero density	number of non-trivial	non-zero density table elements	number of non-trivial digits
(2,2)	0.3333	1	0.3333	0
(2,3)	0.25	3	0.25	1
(2,4)	0.2	7	0.2	3
(2,5)	0.1667	15	0.1667	7
(2,6)	0.1429	31	0.1429	15
(3,2)	0.4	5	0.4	2
(3,3)	0.2857	17	0.2857	8
(4,2)	0.4286	11	0.4286	5
(5,2)	0.4444	19	0.4444	9

for larger width w. It is therefore interesting to compare both the wrNAF and the gSWF.

Consider the case of $r = 3$ and $w = 2$, the elements stored within the gSWF precomputed table are $\{2, 4, 5, 7, 8\}$ and the elements of the wrNAF digit set are $\{\pm 2, \pm 4\}$ (or $\{2, 4\}$ is sufficient for scalar multiplication over elliptic curve), respectively. An interesting fact is that both wrNAF and gSWF remove elements divisible by the radix r from their tables. Evidently, storage space requirement for the proposed wrNAF is much smaller. It will be clear from the following paragraph that the non-zero densities of both wrNAF and gSWF are the same.

The number of non-trivial elements in the gSWF table is $r^w - r^{w-1} - 1$ (excluding $\{1\}$) and the non-zero density of gSWF is $\frac{r-1}{(r-1)w+1}$. Recall that there are $\frac{r^w - r^{w-1} - 2}{2}$ non-trivial elements (excluding $\{\pm 1\}$) in the wrNAF digit set (or a corresponding precomputed table) and the non-zero density of wrNAF is $\frac{r-1}{w(r-1)+1}$. With the above results, numerical enumeration of comparisons between gSWF and wrNAF are listed in Table 2. The result is that both gSWF and wrNAF have identical non-zero density and thus have equivalent computational performance. However, wrNAF is superior to gSWF due to much less storage space requirement. The wrNAF based approach needs less than half memory space

Table 3 Example of gNAF and rNAF.

integer	radix-3	rNAF	gNAF	gSWF
1	0001	0001	0001	0001
2	0002	0002	0002	0002
3	0010	0010	0010	0010
4	0011	0004	0011	0004
5	0012	$010\bar{4}$	$002\bar{1}$	0005
6	0020	0020	0020	0020
7	0021	$010\bar{2}$	$010\bar{2}$	0007
8	0022	$010\bar{1}$	$010\bar{1}$	0008
9	0100	0100	0100	0100
10	0101	0101	0101	0101
11	0102	0102	0102	0102
12	0110	0040	0110	0040
13	0111	0104	0111	0041
14	0112	$020\bar{4}$	$02\bar{1}\,\bar{1}$	0042
15	0120	$10\bar{4}0$	$02\bar{1}0$	0050
16	0121	$020\bar{2}$	$020\bar{2}$	0051
17	0122	$020\bar{1}$	$020\bar{1}$	0052
18	0200	0200	0200	0200
19	0201	0201	0201	0201
20	0202	0202	0202	0202
21	0210	$10\bar{2}0$	$10\bar{2}0$	0070
22	0211	0204	$10\bar{2}1$	0071
23	0212	$100\bar{4}$	$10\bar{1}\,\bar{1}$	0072
24	0220	$10\bar{1}0$	$10\bar{1}0$	0080
25	0221	$100\bar{2}$	$100\bar{2}$	0081
26	0222	$100\bar{1}$	$100\bar{1}$	0082

as gSWF.

4.3 Example of gNAF, rNAF and gSWF

In Table 3, we show examples of gNAF, rNAF in Sect. 3, and gSWF in Sect. 4.2 up to 3-digit radix-r representation for $r = 3$ and width 2.

4.4 Relationship between gNAF and rNAF

We explain that gNAF is a simple degeneration of rNAF. Recall that rNAF is generated by the conversion table of two consecutive digits described in Sect. 3.1. We show that the conversion table for gNAF can be obtained by degenerating that of rNAF. For example, the degenerated conversion table for $r = 3$ is as follows: $(0,1) \leftarrow (0,1)$, $(0,2) \leftarrow (0,2)$, $(1,0,\bar{2}) \leftarrow (2,1)$, $(1,0,\bar{1}) \leftarrow (2,2)$. The difference from the rNAF generation algorithm is to eliminate the tables $(0,\bar{4}) \leftarrow (1,2)$ and $(0,4) \leftarrow (1,1)$. In other words, if the consecutive digits $(1,2)$ or $(1,1)$ appear, we do not convert it, but slide 1 bit to the left.

Indeed the gNAF can be generated by the following algorithm, which is a simple modification of the generation algorithm for rNAF. For the sake of simplicity, we use the radix-r representation for input d.

The only difference from the rNAF generation algorithm is the if-condition "if $d_{i+1} + d_i/r \bmod r = 0$ or $r-1$" appeared at Step 2.2 and its branch (i.e., the whole process of Step 2.3). In order to satisfy the property of gNAF, we have an additional treatment at Step 2.3, namely we perform $(d_{i+1}, d_i) \to (d_{i+1}+1, d_i - r)$ for $(d_i \bmod r + d_{i+1} \bmod r) > r$.

Algorithm 4: Proposed Algorithm (Radix-r to gNAF)

Input : An integer in radix-r representation
$d = (d_{n-1}, \ldots, d_1, d_0) \ldots$
Output: The radix-r gNAF of d: $cd = (cd_n, \ldots, cd_1, cd_0)$.

```
i ← 0;
d_n ← 0;
while i < n do
    if d_i mod r = 0 then
        cd_i ← 0;
        d_{i+1} ← d_{i+1} + d_i/r;
        i ← i + 1;
    end
    else if d_{i+1} + d_i/r mod r = 0 or r − 1  cd_{i+1} ← 0;
    cd_i ← d_{i+1} r + d_i mods r;
    d_{i+2} ← d_{i+2} + (1 − sign(cd_i))/2;
    i ← i + 2;
    else
        d_{i+1} ← d_{i+1} + d_i/r;
        if (d_i mod r + d_{i+1} mod r) > r then
            cd_i ← d_i mod r − r;
            d_{i+1} ← d_{i+1} + 1;
            i ← i + 1;
        end
        else
            cd_i ← d_i mod r;
            i ← i + 1;
        end
    end
end
return (cd_n, ..., cd_1, cd_0);
```

In the following, we prove that this algorithm correctly returns the gNAF representation of the radix-r representation of d.

Theorem 5. *The algorithm (*Radix-r to gNAF*) generates radix-r gNAF.*

Proof. Let $d'_i = d_i + b \bmod r$, where b is a carry in the algorithm from right hand side, namely $d'_i = d_i + d_{i-1}/r \bmod r$.

At step 2.1., we have a branch, if digit d'_i is equal to zero, then we skip to the next bit after computing carry. If digit d'_i is non-zero, we check digit d'_{i+1} with carry is 0 or $r-1$. If $d'_{i+1} = 0$, we assign $(cd_{i+1}, cd_i) = (0, d'_i)$. If $d'_{i+1} = r-1$, then we perform the conversion $(cd_{i+1}, cd_i) = (1, 0, d'_{i+1} r + d'_i - r^2)$. Note that $|d'_{i+1} r + d'_i - r^2| < r$, and thus the converted digits (cd_{i+1}, cd_i) after Step 2.2. satisfy the condition of gNAF.

If d'_{i+1} is neither 0 nor $r-1$, then we assign d'_i based on the size of $|d'_{i+1} + d'_i|$ at Step 2.3. If $|d'_{i+1} + d'_i| > r$ holds, we assign $cd_i = d'_i - r$ with carry to $d'_{i+1} = d'_{i+1} + 1$. Otherwise, $cd_i = d'_i$. Therefore, after Step 2.3. the converted digits (d'_{i+1}, cd_i) satisfy the condition of gNAF. Then we have to consider the case that the digit d'_{i+1} arisen from Step 2.3 is converted by the next Step 2.2 or Step 2.3. Denote by cd_{i+1} the converted digit, and we check whether two consecutive bits (cd_{i+1}, cd_i) satisfy the condition of gNAF. Note that $d'_{i+1} > 0$ and $cd_{i+1} = d'_{i+1} - r < 0$. Recall that $|d'_{i+1} + cd_i| < r$ and $|d'_{i+1}| > |cd_i|$ for $cd_i < 0$ from the above discussion. If

$cd_i < 0$, then we have $|cd_{i+1} + cd_i| = r - |d'_{i+1}| + |cd_i| < r$. In the case of $cd_i > 0$, we have $|cd_{i+1} + cd_i| = |d'_{i+1} + cd_i - r| < r$ due to $d'_{i+1} + cd_i \neq 0$, and $|cd_{i+1}| - |cd_i| = r - (d'_{i+1} + cd_i) > 0$.

Consequently, any two consecutive digits (cd_{i+1}, cd_i) obtained by the algorithm satisfied the condition of gNAF. □

The proposed algorithm has the minimal non-zero density $(r-1)/(r+1)$ with digit set $\{0, \pm 1, \ldots, \pm(r-1)\}$ due to the uniqueness of gNAF. However, our algorithm is able to show an easier proof about the non-zero density of gNAF.

Theorem 6. *The average density of non-zero digits arisen from the algorithm (*Radix-r to gNAF*) is asymptotically* $\frac{r-1}{r+1}$.

Proof. The above algorithm has four statuses of digit d_i, namely digit (0), digit (i) for $i = 1, 2, \ldots, r-2$, digit $(r-1, y)$ with carry from the right, and digit $(r-1, n)$ without carry from the right. The statuses are transited by the following Markov chain:

$$\begin{pmatrix} (0) & : & 1/r & (r-2)/r & 0 & 1/r \\ (i) & : & 1/r & (r-2)/r & 1/r & 0 \\ (r-1,y) & : & 1/r & (r-2)/r & 0 & 1/r \\ (r-1,n) & : & 1/r & (r-2)/r & 1/r & 0 \end{pmatrix}.$$

This Markov chain is aperiodic and irreducible, and thus there is the stationary distribution: $((0), (i), (r-1, y), (r-1, n)) = (\frac{1}{r}, \frac{r-2}{r}, \frac{r-1}{(r+1)r}, \frac{2}{(r+1)r})$. Thus zero digit appears at statuses (0) and $(r-1, y)$, and thus the non-zero density is $1 - \frac{1}{r} - \frac{r-1}{(r+1)r} = \frac{r-1}{r+1}$. Consequently, we prove the theorem. □

5. Experimental Results

Main operations appeared in pairing based cryptosystems are as follows;

(1) Tate pairing $e(Q, R)$,

(2) scalar multiplication dP,

where P, Q, R are points on the underlying elliptic curve and d is an integer scalar [4], [5], [14], [15]. The proposed scheme in this paper aims at improving the efficiency of the scalar multiplication dP of elliptic curves over a finite field with characteristic $r > 2$.

In order to evaluate the performance of the proposed scheme, we implemented the scalar multiplication dP on supersingular curve $E : y^2 = x^3 - x + 1$ over finite field $GF(3^{97})$ used in the references [8], [9], [11]. In this curve, the tripling operation is very efficient, i.e., $3P = ((x^3)^3 - 1, -(y^3)^3)$ for a given point $P = (x, y)$, which requires no multiplications but 4 cubings in the base field $GF(3^{97})$. The finite field operations were implemented following the algorithms from [23], but the multiplication used the well-known comb method. As the irreducible trinomial for $GF(3^{97})$, we use $x^{97} + x^{12} + 1$.

Our experiment environment was a Pentium 4 2.66 GHz desktop, with 512 MBytes of RAM, running Linux Gentoo 2.6.10-r4. We wrote the program in C and compiled it with GCC version 3.3.3 using the flags `-O3 -fomit-frame-pointer -funroll-loops`. In our implementation the ratio of multiplication time to inversion time in finite field is about 5, in this case, the affine coordinates are more interesting. The conversion algorithm from an exponent in base 2 to wrNAF was implemented using GNU Multiple Precision Arithmetic Library (GMP).

Table 4 Timings for the scalar multiplication using wrNAF.

(Radix,Width)	Precomp.	Evaluation	Total
(2,1)	0.00 μs	5690.61 μs	5690.61 μs
(2,2)	0.00 μs	4990.20 μs	4990.20 μs
(2,3)	3.02 μs	4619.18 μs	4622.20 μs
(2,4)	80.87 μs	4410.00 μs	4490.87 μs
(2,5)	188.20 μs	4269.61 μs	4457.81 μs
(2,6)	446.59 μs	4198.47 μs	4645.06 μs
(3,1)	0.00 μs	2040.71 μs	2040.71 μs
(3,2)	47.54 μs	1321.59 μs	1369.13 μs
(3,3)	216.42 μs	986.48 μs	1202.90 μs
(3,4)	704.61 μs	812.27 μs	1516.87 μs
(3,5)	2167.39 μs	709.31 μs	2876.71 μs

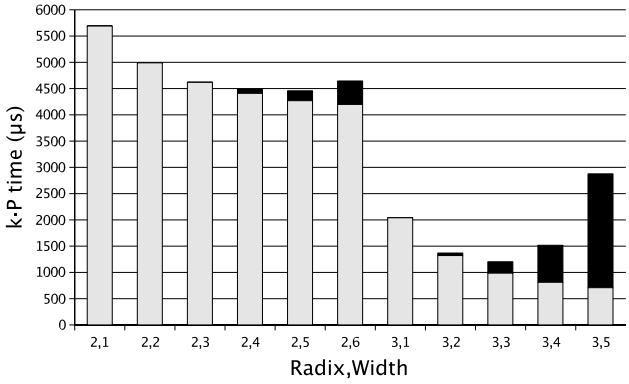


Fig. 1 Precomputation (black) and evaluation (gray) times for wrNAF.

Table 4 shows the pre-computation and the evaluation time for radix-2 and -3 with different widths. To evaluate timings of the scalar multiplication dP, we executed it 100000 times for each pair (radix,width) with a random 155-bit exponent. We added the width 1 as reference, and it corresponds to the double-and-add for radix-2 and triple-and-add for radix-3. As expected the radix-3 representation is much faster then radix-2, because in characteristic 3 the point tripling can be efficiently computed. For comparison, the timing of computing Tate pairing $e(Q, R)$ using Duursma-Lee algorithm [7], [13] requires 7.71 ms in our experiment.

In Fig. 1, we show the precomputation time, in black, and the evaluation time, in gray. The precomputation time grows faster in characteristic 3 leading to different optimal width for radix-2 and -3. In our implementation the optimal width, in terms of speed, for radix-2 is 5 and for radix-3 is 3. In our implementation, the wrNAF representation is about 21% faster when using radix-2 and 41% when using radix-3 then a standard scalar multiplication.

6. Conclusion

In this paper, we extended the width-w non-adjacent form (wNAF) of binary representation to that of the radix-r rep-

resentation, called rNAF. Our construction inherits the property of the classical width-w NAF, namely there is at most 1 non-zero digit among w consecutive digits, and the digits are not divisible by r. We estimated the required size of digit set and the average density of non-zero digits using Markov chain. We compared the proposed scheme with the previously known gNAF representation discussed by Joye-Yen. Our scheme has smaller non-zero density with a larger digit set. For radix 3, the proposed algorithm with width $w = 2$ attains non-zero density 0.4 with two additional digits, where gNAF has 0.5 with one additional digit.

Moreover, we showed that gNAF is a degenerated form of rNAF—if some conversion tables for rNAF are removed, we can obtain gNAF with a small modification. Based on this observation, we presented a simple generation algorithm of gNAF. Therefore, the proposed scheme can be considered as a canonical class for the signed radix-r representation. Indeed, if we choose $r = 2$, then we are able to obtain the classical NAF. The proposed scheme is a good alternative to gNAF.

The radix-r representation is used for the efficient computation of pairing-based cryptosystem constructed over (hyper-)elliptic curve with characteristic r. The proposed rNAF is particularly able to improve the speed of computing scalar multiplications.

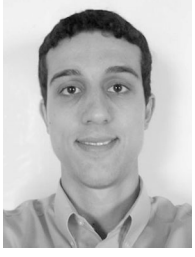
Acknowledgement

The research of S.M. Yen was supported in part by the National Science Council R.O.C. under contract NSC 92-2213-E-008-007. D. Reis Jr and T. Takagi were supported by SicAri Project (http://www.sicari.de/).

References

[1] P. Barreto, H. Kim, B. Lynn, and M. Scott, "Efficient algorithms for pairing-based cryptosystems," CRYPTO2002, LNCS 2442, pp.354–368, 2002.
[2] G. Bertoni, J. Guajardo, S. Kumar, G. Orlando, C. Paar, and T. Wollinger, "Efficient $GF(p^m)$ arithmetic architectures for cryptographic applications," CT-RSA2003, LNCS 2612, pp.158–175, 2003.
[3] I. Blake, G. Seroussi, and N. Smart, Elliptic Curves in Cryptography, Cambridge University Press, 1999.
[4] D. Boneh and M. Franklin, "Identity based encryption from the Weil pairing," SIAM J. Comput., vol.32, no.3, pp.586–615, 2001.
[5] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," ASIACRYPT2001, LNCS 2248, pp.514–532, 2001.
[6] W. Clark and J. Liang, "On arithmetic weight for a general radix representation of integers," IEEE Trans. Inf. Theory, vol.IT-19, no.6, pp.823–826, 1973.
[7] I. Duursma and H.-S. Lee, "Tate pairing implementation for hyperelliptic curves $y^2 = x^p - x + d$," ASIACRYPT2003, LNCS 2894, pp.111–123, 2003.
[8] S. Galbraith, K. Harrison, and D. Soldera, "Implementing the Tate pairing," ANTS V, LNCS 2369, pp.324–337, Springer-Verlag, 2002.
[9] R. Granger, D. Page, and M. Stam, "On small characteristic algebraic tori in pairing-based cryptography," Cryptology ePrint Archive: Report 2004/132.
[10] D. Gordon, "A survey of fast exponentiation methods," J. Algorithms, vol.27, pp.129–146, 1998.
[11] K. Harrison, D. Page, and N. Smart, "Software implementation of finite fields of characteristic three," LMS J. Comput. Math., vol.5, pp.181–193, 2002
[12] IEEE P1363, Standard Specifications for Public-Key Cryptography, 2000.
[13] S. Kwon, "Efficient Tate pairing computation for supersingular elliptic curves over binary fields," Cryptology ePrint Archive, Report 2004/303.
[14] A. Joux, "A one round protocol for tripartite Diffie-Hellman," ANTS V, LNCS 1838, pp.385–394, 2000.
[15] A. Joux, "The Weil and Tate pairings as building blocks for public key cryptosystems (survey)," ANTS V, LNCS 2369, pp.20–32, 2002.
[16] M. Joye and S.-M. Yen, "New minimal modified radix-r representation with applications to smart cards," PKC2002, LNCS 2274, pp.375–384, 2002.
[17] A. Miyaji, T. Ono, and H. Cohen, "Efficient elliptic curve exponentiation," ICICS'97, LNCS 1334, pp.282–291, 1997.
[18] D. Page and N. Smart, "Hardware implementation of finite fields of characteristic three," CHES2002, LNCS 2523, pp.529–539, 2002.
[19] B. Phillips and N. Burgess, "Minimal weight digit set conversions," IEEE Trans. Comput., vol.53, no.6, pp.666–677, 2004.
[20] N. Smart and J. Westwood, "Point multiplication on ordinary elliptic curves over fields of characteristic three," Appl. Algebra Eng. Commun. Comput., vol.13, no.6, pp.485–497, 2003.
[21] J. Solinas, "Efficient arithmetic on Koblitz curves," Des. Codes Cryptogr., vol.19, no.2/3, pp.195–249, 2000.
[22] E.G. Thurber, "On addition chains $l(mn) \leq l(n) - b$ and lower bounds for $c(r)$," Duke Math. J., vol.40, pp.907–913, 1973.
[23] K. Harrison, D. Page, and N. Smart, "Software implementation of finite fields of characteristic 3," LMSJ Comput. Math., vol.5, pp.181–193, 2002.

Tsuyoshi Takagi received the B.Sc. and M.Sc. degrees in mathematics from Nagoya University in 1993 and 1995, respectively. He had engaged in the research on network security at NTT Laboratories from 1995 to 2001. He received the Dr.rer.nat degree from Technische Universität Darmstadt in 2001. He was an Assistant Professor in the Department of Computer Science at Technische Universität Darmstadt until 2005. He is currently an Associate Professor in the School of Systems Science Information at Future University-Hakodate. His current research interests are information security and cryptography. Dr. Takagi is a member of International Association for Cryptologic Research (IACR).



David Reis, Jr. received the B.Eng. degree from the School of Electrical and Computer Engineering at State University of Campinas (UNICAMP) in 2005. From 2004 to 2005 he visited the Department of Computer Science at Technische Universität Darmstadt (TUD) in the exchange program between UNICAMP and TUD. During his stay at TUD he worked for SicAri project on secure implementation of elliptic curve cryptosystems.

Sung-Ming Yen received his Ph.D. degree in electrical engineering from the National Cheng-Kung University, Taiwan, in 1994. He is a professor of the Dept of Computer Science and Information Engineering, National Central University, Taiwan, where he directs the Laboratory of Cryptography and Information Security (LCIS). His research interests include cryptography, information and network security, and fast computer arithmetic. Dr. Yen received the best paper awards of WISA 2002, ICS (Taiwan) 1998, and ISC (Taiwan) 2002 to 2005. He is the inventor of about 20 local or foreign patents, and authored or coauthored about 80 international journal or conference papers. He served in numerous program committees of international conferences or workshops of cryptology and network security.

Bo-Ching Wu completed his B.Sc. degree in mathematics from National Central University (NCU) in 2002, then he joins the Laboratory of Cryptography and Information Security (LCIS) in the Department of Computer Science and Information Engineering of NCU. Under the supervision of Prof. Sung-Ming Yen, Mr. Wu works towards his Ph.D. degree. His research interests include information security, cryptography, and side-channel cryptanalysis on smart card.