International Journal of Security and Its Applications
Vol.8, No.3 (2014), pp.31-38
http://dx.doi.org/10.14257/ijsia.2014.8.3.04

A New Left-to-Right Scalar Multiplication Algorithm Using a New Recoding Technique

Abdalhossein Rezai[1] and Parviz Keshavarzi[2]

[1]*Academic Center for Education, Culture and Research (ACECR), Isfahan University of Technology (IUT) branch, Isfahan, Iran*
[2]*Electrical and Computer Engineering Faculty, Semnan University, Semnan, Iran*
[1]*rezaie@acecr.ac.ir*, [2]*pkeshavarzi@semnan.ac.ir*

Abstract

Elliptic Curve Cryptosystem (ECC) is a well-known cryptosystem for securing the communications. The most important operation in ECC is scalar multiplication. The integer representation plays an important role in the performance of this operation. This paper presents and evaluates a novel recoding technique which reduces the average Hamming weight of integers. The Left-to-Right (L2R) scalar multiplication is modified to utilize this new integer representation. Our analysis shows that the computation cost (the number of required point addition/subtraction operation) in the proposed L2R scalar multiplication algorithm is effectively reduced in comparison with other modified L2R binary scalar multiplication algorithms.

Keywords: *Network security, Elliptic Curve Cryptosystem (ECC), cryptography algorithm, scalar multiplication algorithm, recoding technique*

1. Introduction

Pairing based cryptosystems [1] and Elliptic Curve Cryptosystems (ECCs) [2, 3], two new Public Key Cryptosystems (PKCs), have recently been used for application in cryptography [4-6]. The basic operation in the pairing based cryptosystems and ECCs is scalar multiplication. As a result, increasing the performance of this operation is a challenging issue [5-8].

Scalar multiplication is performed by repeating point addition/subtraction and point doubling operations. Hamming weight of scalar determines the number of the required point addition/subtraction operation. Reducing the Hamming weight of the scalar can affect the performance of scalar multiplication [7-9].

There is much attempts to enhance the efficiency of this operation by reducing the Hamming weight such as scalar multiplication algorithm using complement recoding technique [10], modified complementary recoding technique [11], and hybrid complementary and 1's complement recoding technique [12].

This paper presents and evaluates a novel scalar multiplication algorithm to reduce the computation cost (the number of required point addition/subtraction operation). This novel scalar multiplication algorithm utilized a new recoding technique to reduce the Hamming weight of the scalar. Using Markov chain, we prove that the average Hamming weight of the proposed scalar multiplication is $\frac{3n}{13}$.

The rest of this paper is organized as follows: Section 2 briefly describes the scalar multiplication algorithm. Section 3 outlines two typical recoding techniques. The proposed

ISSN: 1738-9976 IJSIA

algorithms are presented in Section 4. In Section 5, the proposed scalar multiplication is evaluated. Finally, Section 6 concludes this paper.

2. The Scalar Multiplication Algorithm

Scalar multiplication is defined by $Q = kP = \underbrace{P + P + \cdots + P}_{k \text{ times}}$ where P and Q are the elliptic curve points and k is a scalar. This operation plays a major role in the pairing based cryptosystems and ECCs. A widely used method for computing scalar multiplication is binary (double-and-add) method [9, 13, 14, 15]. There are two common algorithms in the binary method: the Right-to-Left (R2L) algorithm and the Left-to-Right (L2R) algorithm. The R2L algorithm processes the scalar bits from the Least Significant Bit (LSB), while the L2R algorithm scans the scalar bits from the Most Significant Bit (MSB). The L2R algorithm is a widely used algorithm. Algorithm 1 shows the L2R binary scalar multiplication algorithm [13-15].

Algorithm 1: The L2R binary scalar multiplication algorithm
INPUT: $k=(k_{n-1}k_{n-2}...k_1k_0)_2$;P=(x,y);
OUTPUT: Q=(x',y')=kP;
1. $Q \leftarrow 0$;
2. For i= n-1 Downto 0
3. Q=2Q;
4. If k_i=1 then $Q \leftarrow Q+P$;
5. Return Q;

In this algorithm, $k = \sum_{i=0}^{n-1} k_i.2^i$ where $k_i \in \{0,1\}$. This algorithm scans the scalar bits from the MSB to LSB. For $k_i \neq 0$, both point operations are computed, while for $k_i = 0$, the point doubling operation is only computed [9, 14].

3. The Recoding Techniques

Based on investigations in the previous section, the integer representation (the length and Hamming weight of the scalar k) has an important role in the performance of this algorithm. Several recoding techniques have been proposed to reduce the Hamming weight of the scalar [9-12, 16, 17]. This section outlines two important recoding techniques, Canonical Recoding (CR) technique and complementary recoding technique.

3.1. The Canonical Recoding Technique

A canonical representation [18] of an integer A is defined as $A = \sum_{i=0}^{n-1} a_i.2^i$ where $a_i \in \{-1,0,1\}$. Algorithm 2 is utilized to convert an integer from the binary representation to its canonical representation [9, 14].

Algorithm 2: The Canonical Recoding (CR) algorithm
Input: $A= (a_{n-1}a_{n-2}\ldots a_1a_0)_2$
Output: $D= (d_nd_{n-1}\ldots d_1d_0)_{SD}$
1. c_0:= 0;
2. For i = 0 to n
3. $c_{i+1} := \lfloor (a_i + a_{i+1} + c_i)/2 \rfloor$;
4. $d_i := a_i + c_i - 2c_{i+1}$;
5. Return D;

This algorithm scans input integer A from the LSB to the MSB. The average Hamming weight of an n-bit canonical recoded integer is $\frac{n}{3}$ and this integer representation guarantees the minimal Hamming weight. [14, 19, 20, 21].

3.2. The Complementary Recoding Technique

A complementary representation of an integer k is a sequence of bits $k = (k_{n-1}k_{n-2}...k_1k_0)_2$ where $k_i \in \{0,1\}$ and satisfies the following equation [9-11]:

$$k = \sum_{i=0}^{n-1} k_i.2^i = 2^n - \bar{k} - 1 \quad (1)$$

where $\bar{k}$ is 1's complement of k, $\bar{k} = \bar{k}_{n-1}\bar{k}_{n-2}\cdots\bar{k}_1\bar{k}_0$, and satisfies the following equation:

$$\begin{cases} \bar{k}_i = 0 & \text{if } k_i = 1 \\ \bar{k}_i = 1 & \text{if } k_i = 0 \end{cases} \quad \text{for } i = 0,1,...,\ n-1 \quad (2)$$

4. The Proposed Algorithms

This section outlines the proposed recoding technique, and the proposed scalar multiplication algorithm.

4.1. The Proposed Recoding Technique

The proposed representation for an integer B, which is output of the proposed recoding algorithm, is a sequence of digits $B = (b_{n-1}, b_{n-2}, \ldots, b_1, b_0)_{SD}$ where $b_i \in \{-1, 0, 1\}$. Figure 1 shows the steps to convert an integer from its binary representation to the proposed representation.

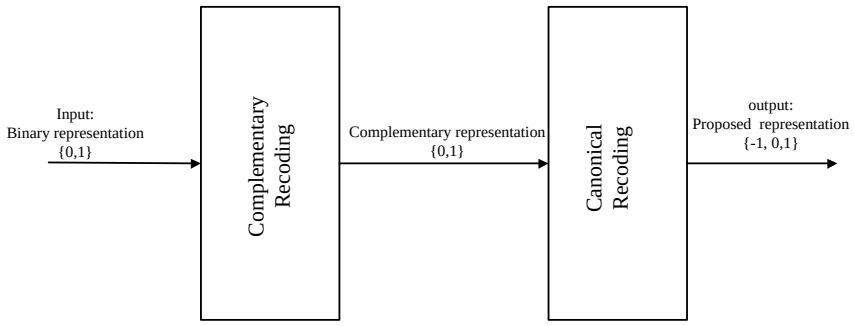

Figure 1. The Block Diagram of the Proposed Recoding Technique

The proposed integer representation uses two recoding techniques: canonical recoding technique and complementary recoding technique. Algorithm 3 is proposed for converting an n-bit integer k from its binary representation to the proposed representation.

Algorithm 3: The proposed recoding algorithm
Input: $k = (k_{n-1}, k_{n-2}, \ldots, k_1, k_0)_2$;
Output: $B = (b_n, b_{n-1}, \ldots, b_1, b_0)_{SD}$;
1. If H(k)>n/2 Then $A = \bar{k}$;
2. Else A=k;
3. D=CR(A);

4. If H(k)>n/2 Then $B = 2^n - D - 1$;
5. Else B=D;
6. Return B;

H(k) in this algorithm denotes the Hamming weight of k and the output of this algorithm is $B = (b_n, b_{n-1}, \ldots, b_1, b_0)_{SD}$. First, H(k) is compared to the average Hamming weight of the binary representation. Then, if H(k) is greater than n/2, the proposed representation is represented based on 2's complement as follows:

$B = 2^n - D - 1$.

where D is computed by applying CR algorithm to 1's complement of k, $A = \bar{k}$.

On the other hand, the proposed representation is computed as follows:

B=D

where D is computed by applying CR algorithm to A=k.

Theorem 1. The Hamming weight arisen from algorithm 3 is asymptotically $\frac{3n}{13}$.

Proof. Assume that, k is an n-bit binary integer. To compute the average Hamming weight of the proposed representation, we model it using Markov chain.

According to [9, 19], all possible inputs and outputs of the CR algorithm are listed in Table 1.

Table 1. The State Transition Table for the CR Algorithm

Current State		Output	Next State	
s_i	(a_{i+1}, a_i, c_i)	(d_i, c_{i+1})	$a_{i+2} = 0$	$a_{i+2} = 1$
s_0	(0, 0, 0)	(0, 0)	s_0	s_4
s_1	(0, 0, 1)	(1, 0)	s_0	s_4
s_2	(0, 1, 0)	(1, 0)	s_0	s_4
s_3	(0, 1, 1)	(0, 1)	s_1	s_5
s_4	(1, 0, 0)	(0, 0)	s_2	s_6
s_5	(1, 0, 1)	(1, 1)	s_3	s_7
s_6	(1, 1, 0)	(1, 1)	s_3	s_7
s_7	(1, 1, 1)	(0, 1)	s_3	s_7

All 8 states in this table are labeled $s_0 - s_7$. For example s_4 represents $(a_{i+1}, a_i, c_i) = (1, 0, 0)$. In this state, the output is $(d_i, c_{i+1}) = (0, 0)$. So, the next state is $(a_{i+2}, a_{i+1}, c_{i+1}) = (a_{i+2}, 1, 0)$. Since the complementary recoding is applied to the integer k for computing A, $P(a_{i+2} = 1) = \frac{1}{4}$ and $P(a_{i+2} = 0) = \frac{3}{4}$ [9-10]. Thus, the probability transitions from the state $s_4 = (1, 0, 0)$ to the states $s_6 = (1, 1, 0)$ and $s_2 = (0, 1, 0)$ are $\frac{1}{4}$ and $\frac{3}{4}$ respectively. The probability when the state s_i succeeds the state s_j is denoted by P_{ij}. So, $P_{46} = \frac{1}{4}$, $P_{42} = \frac{3}{4}$ and $P_{4j} = 0$ for $j = 0,1, 3, 4, 5, 7$. Therefore, the one step transition probability matrix is given as follows:

$$P = \begin{bmatrix} 3/4 & 0 & 0 & 0 & 1/4 & 0 & 0 & 0 \\ 3/4 & 0 & 0 & 0 & 1/4 & 0 & 0 & 0 \\ 3/4 & 0 & 0 & 0 & 1/4 & 0 & 0 & 0 \\ 0 & 3/4 & 0 & 0 & 0 & 1/4 & 0 & 0 \\ 0 & 0 & 3/4 & 0 & 0 & 0 & 1/4 & 0 \\ 0 & 0 & 0 & 3/4 & 0 & 0 & 0 & 1/4 \\ 0 & 0 & 0 & 3/4 & 0 & 0 & 0 & 1/4 \\ 0 & 0 & 0 & 3/4 & 0 & 0 & 0 & 1/4 \end{bmatrix} \quad (3)$$

If π_i shows the probability of the state s_i for $i = 0, 1, \ldots, 7$, the probability for each state is found by solving the following equations [9, 19]:

$$\begin{cases} \pi . P = \pi \\ \sum_{i=0}^{7} \pi_i = 1 \end{cases} \quad (4)$$

As a result:

$$\pi = [\frac{27}{52}, \frac{9}{208}, \frac{27}{208}, \frac{3}{52}, \frac{9}{52}, \frac{3}{208}, \frac{9}{208}, \frac{1}{52}] . \quad (5)$$

Thus, the probability of the zero digit and nonzero digits are $\pi_0 + \pi_3 + \pi_4 + \pi_7 = \frac{10}{13}$ and $\pi_1 + \pi_2 + \pi_5 + \pi_6 = \frac{3}{13}$ respectively. Therefore, the average Hamming weight of the proposed representation is $\frac{3n}{13}$ for n-bit integers.

4.2. The Proposed Scalar Multiplication Algorithm

Since Hamming weight of integers plays an important role in the scalar multiplication, the proposed recoding technique is a suitable technique for applying to the binary scalar multiplication. In this section, the L2R binary scalar multiplication is modified to utilize the proposed recoding technique. The proposed L2R scalar multiplication is shown in algorithm 4.

Algorithm 4: The proposed L2R scalar multiplication algorithm
Input: $k = (k_{n-1}, k_{n-2}, \ldots, k_1, k_0)_2$, P=(x, y);
Output:Q=(x', y')=kP;
1. Q=0;
2. Compute B by applying algorithm 3 to k;
3. For i=n Downto 0
4. Q=2Q;
5. If (b_i>0) Then Q=Q+P;
6. Else If (b_i<0) Then Q=Q-P;
7. Return Q;

The inputs of this algorithm are scalar $k = (k_{n-1}, k_{n-2}, \ldots, k_1, k_0)_2$ and elliptic curve point P=(x, y), the output is elliptic curve point Q =(x', y')= kP. In this algorithm, B is computed by applying the proposed recoding technique to the scalar k. The point doubling operation is

performed per iteration, while the point addition/subtraction operation is only performed where $b_i \neq 0$.

5. Comparison

As described in the previous sections, the Hamming weight of the scalar denotes the number of required point addition/subtraction operation in the scalar multiplication algorithm. Thus, reducing the Hamming weight can enhance the performance of the scalar multiplication algorithm [7, 9, 11, 17]. So, the effect of the proposed recoding technique on the scalar multiplication is investigated in this section.

The computation cost (the number of required point addition/subtraction operation) of the proposed scalar multiplication algorithm and two recent modifications of the L2R binary scalar multiplication algorithm for various operand size are computed and summarized in Table 2 and Figure 2.

Table 2. The Comparative Table for Computation Cost

Reference	Average Hamming weight	Computation cost				
		n=192	n=224	n=256	n=384	n=521
[11]	$\frac{n}{3}$	64	75	86	128	174
[12]	$\frac{n}{4}$	48	56	64	96	130
This paper	$\frac{3n}{13}$	45	52	60	89	121

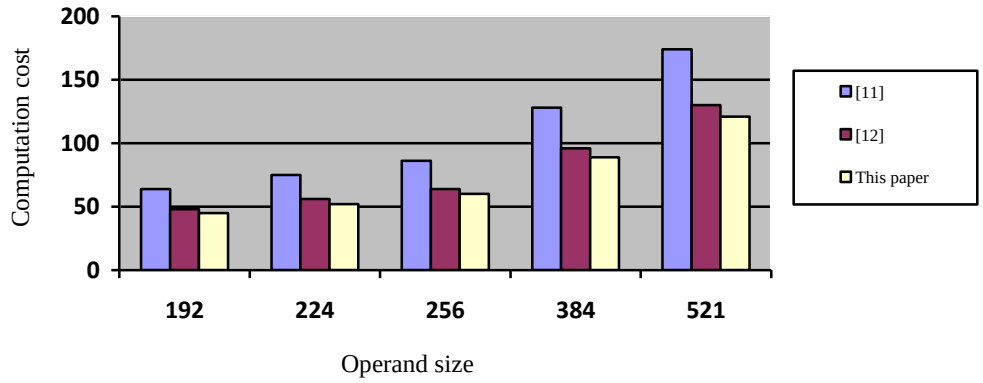

Figure 2. The Comparison of the Computation Cost

Based on our analysis which is shown in Table 2 and Figure 2, the computation cost (the number of required point addition/subtraction operation) in the proposed L2R scalar multiplication algorithm is reduced in comparison with recent modifications of the L2R binary scalar multiplication algorithm [11, 12].

6. Conclusion

An important operation in the pairing based cryptosystems and ECCs is scalar multiplication. In this paper, a novel and efficient scalar multiplication algorithm based on a new recoding technique is presented. In the proposed scalar multiplication algorithm, the canonical recoding technique is applied to the complementary recoded scalar. The Markov

chain is used to prove that the average Hamming weight of the proposed integer representation is $\frac{3n}{13}$. Our analysis shows that using the proposed scalar multiplication algorithm, the computation cost (the number of required point addition/subtraction operation) is effectively reduced in comparison with other modifications of the L2R binary scalar multiplication algorithm [11, 12].

References

[1] J., "The Weil and Tate pairings as building blocks for public key cryptosystems (survey)", ANTS V, LNCS, 2369. Springer, **(2002)**, pp. 20-32.
[2] N. Koblitz, "Elliptic curve cryptosystem", Math. Comput., vol. 48, **(1987)**, pp. 203-209.
[3] V. Miller, "Use of elliptic curves in cryptography", Proc. of advances in cryptology (CRYPTO85), LNCS, 218, Springer, Heidelberg, **(1985)**, pp. 417-428.
[4] A. Rezai, P. Keshavarzi and Z. Moravej, "Secure SCADA communication by using a modified key management scheme", ISA Trans., vol. 52, no. 4, **(2013)**, pp. 517-524.
[5] H. R. Ahmadi and A. Afzali-kusha, "A low-power and low-energy flexible GF(p) elliptic-curve cryptography processor", J. Zhejiang Uni. sci. C, vol. 11, no. 9, **(2010)**, pp. 724-736.
[6] W. Stallings, "Cryptography and Network Security Principles and Practices", 4th Eds, Prentice Hall, USA, **(2005)**.
[7] B. Qin, M. Li, F. Kong and D. Li, "New left-to-right minimal weight signed-digit radix-r representation", Comput. Elec. Eng, vol. 35, no. 1, **(2009)**, pp. 150-158.
[8] H. Mahdizadeh and M. Masoumi, "Novel architecture for efficient FPGA implementation of elliptic curve cryptographic processor over GF(2^{163})", IEEE Trans. VLSI Syst. DOI: 10.1109/TVLSI.2012.2230410, **(2013)**.
[9] A. Rezai and P. Keshavarzi, "CCS Representation: A new non-adjacent form and its application in ECC", J. Basic Appl. Sci. Res., vol. 2, no. 5, **(2012)**, pp. 4577-4586.
[10] C. Chang, Y. Kuo and C. Lin, "Fast algorithms for common multiplicand multiplication and exponentiation by performing complements", Proc. 17th IEEE int. conf. advanced inf. Netw. Appl. (AINA 2003), IEEE Press, **(2003)**, pp. 807-811.
[11] P. Balasubramaniam and E. Karthikeyan, "Elliptic curve scalar multiplication algorithm using complementary recoding", Appl. Math. comput., vol. 190, no. 1, **(2007)**, pp. 51-56.
[12] X. Huang, P. Shahand and D. Sharma, "Minimizing Hamming weight based on 1's complement of binary numbers over GF(2^m)", Proc. IEEE. 12th int. conf. advanced commun. Tech. (ICACT 2010), IEEE Press, **(2010)**, pp. 1226-1230.
[13] D. Hankerson, A. Menezes and S. Vanstone, "Guide to elliptic curve cryptography", Springer, Heidelberg, **(2004)**.
[14] A. Rezai and P. Keshavarzi, "A new finite field multiplication algorithm to improve elliptic curve cryptosystem implementations", J. Inf. Syst. Telecommun., vol. 1, no. 2, **(2013)**, pp. 47-57.
[15] A. Rezai and P. Keshavarzi, "High-performance implementation approach of elliptic curve cryptosystem for wireless network applications", Proc. IEEE Int. Conf. Consum. Electron. Commun. Netw., IEEE Press, **(2011)**, pp. 1323-1327.
[16] R. Avanzi, "A note on the signed sliding window integer recoding and its left-to-right analogue", Selected areas cryptography, LNCS, Springer, Heidelberg, vol. 3357, **(2005)**, pp. 130-143.
[17] M. Joye and S. Yen, "Optimal left-to-right binary signed-digit recoding", IEEE Trans. comput., vol. 49, no. 7, **(2000)**, pp. 740-748.
[18] G. Reitwiesner, "Binary arithmetic", Adv. Comput., vol. 1, **(1960)**, pp. 231-308.
[19] O. Egeciglu and C. Koc, "Exponentiation using canonical recoding", Theor. Comput. Sci., vol. 129, no. 2, **(1994)**, pp. 407-417.
[20] A. Rezai, and P. Keshavarzi, "A new CMM-NAF modular exponentiation algorithm by using a new modular multiplication algorithm", Trends applied sci. res., vol. 7, no. 3, **(2012)**, pp. 240-247.
[21] A. Rezai and P. Keshavarzi, "High-performance modular exponentiation algorithm by using a new modified modular multiplication algorithm and common-multiplicand-multiplication method", Proc. of the IEEE. World congress on internet security, IEEE Press, pp. 192-197, **(2011)**.

Authors

Abdalhossein Rezai, is an assistant professor in Academic Center for Education, Culture and Research (ACECR), Isfahan University of Technology (IUT) branch, Isfahan, Iran. He received Ph.D. degree in electrical engineering from Semnan University, Semnan, Iran in 2013, M.S. and B.S. degree in electrical engineering from Isfahan University of Technologhy (IUT), Isfahan, Iran in 1999, and 2003, respectively. His research interests include network security, cryptography algorithm and its application, and neural network implementation in nanoelectronics.

Parviz Keshavarzi, is an associate professor in Semnan Univercity, Iran. He received Ph.D. degree in electrical engineering from Manchester University, UK in 1999, and M.S. degree in electrical engineering from Tehran University, Tehran, Iran in 1988. His research interests include network security, cryptography algorithm and its application, and nanoelectronics.