Optimal Average Joint Hamming Weight and Digit Set Expansion of Integer Pairs

Vorapong Suppakitpaisarn[1, 2] Masato Edahiro[1, 3]

Graduate School of Information Science and Technology, the University of Tokyo[1]

ERATO-SORST Quantum Computation and Information Project[2]

System IP Core Research Laboratories, NEC Corporation[3]

1 Introduction

The joint hamming weight of integer pairs (r_1, r_2) is defined as

$$JW_E(r_1, r_2) = ||\{c \mid E_C(r_1, r_2) \neq (0,0)\}||,$$

when $E_C(r_1, r_2)$ is (p_1, p_2). p_i is the c^{th} bit when expand r_i in the representation E. The joint weight is known to affect the computation time of many operations such as the multi-scalar point multiplication of the elliptic curve cryptography, $r_1P + r_2Q$. We illustrate how the joint hamming weight relates to the computation time of the operation in Figure 1. In the figure, the number of point additions is almost equal to the joint hamming weight, and the number of point doubles remains constant. Because of this fact, the lower weight expansions can improve the operation, and many works have explored the lower weight expansion on many specific representations. These include the work by Solinas [1], which proposes the minimal joint weight expansion when the digit set is {0, 1, -1}. Also, the work by Heuberger and Muir [2] presents the expansions for the digit set {-*l*, - (*l* – 1), ... ,-1, 0, 1, ... , (*u* – 1), *u*} for any natural number *l*, and positive integer *u*.

However, most of the previous works propose the minimal weight conversion and its analysis from the mathematical construction, and the properties of the expansion. On some digit sets, the constructions of expansions are complex and hard to be found. These include {-3, -1, 0, 1, 3}, that uses the same amount of memory to store the pre-computed points as {-2, -1, 0, 1, 2}, but have lower minimal average joint hamming weight.

2 Our method

We propose the minimal weight conversion that can be applied to any digit set using the dynamic programming without concerning on the construction of the expansion. This results that our algorithm might not be as fast as most of the algorithms in the literatures. But, we believe that the implementer can use our algorithm as a framework, and produce more efficient algorithm for their specific digit set.

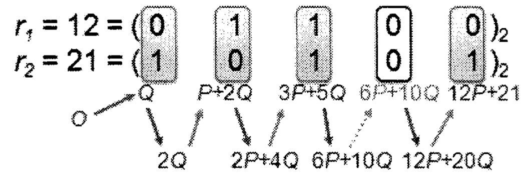
$r_1 = 12 = (0\ 1\ 1\ 0\ 0)_2$
$r_2 = 21 = (1\ 0\ 1\ 0\ 1)_2$
O
Q
P+2Q
3P+5Q
6P+10Q
12P+21Q
2Q
2P+4Q
6P+10Q
12P+20Q

Figure 1: The multi-scalar point multiplication, $12P + 21Q$. It needs 3 point additions (joint hamming weight – 1), and 4 point doubles.

Then, we propose the algorithm to construct the Markov chain for analyzing the average joint hamming weight of any digit sets. As the conversion can be applied for any digit sets, we can find the optimal average joint hamming weight of any digit sets.

3 Results

Our results close many open problems. One of the most interesting point is the expansion when the digit set {-3, -1, 0, 1, 3} mentioned in Section 1. For this digit set, many previous works have proposed the construction, conversion, and analysis. We can improve their results, and prove that our average joint hamming weight is optimal for this digit set. We compare our results with the previous works in Table 1.

Table 1: Comparing our average joint hamming weight with the other preliminary works when expand an integer pairs using {-3, -1, 0, 1, 3}

Research	Average Joint Hamming Weight
[Ava02] [3]	0.3750 = 3/8
[KZZ04] [4]	0.3712 = 121/326
[Mol05] [5]	0.3636 = 4/11
[DOT08] [6]	0.3615 = 239/661
[Our Result]	0.3575 = 281/786 [Optimal]

整数対の桁集合拡張と最適共同ハミング重み
スッパキットパイサーン ウォラポン
(mr_t_dtone@is.s.u-tokyo.ac.jp)
枝廣正人(eda@bp.jp.nec.com)
[1] 東京大学大学院情報理工学系研究科
[2] ERATO-SORST 量子情報システムアーキテクチャ
[3] NEC システム IP コア研究所

We also show the result when the digit set is {-(2h + 1), ... -3, -1, 0, 1, 3, ..., 2h + 1} and *h* is the natural number less than 5. This is also the open problem in [1]. It can be used for showing the relation between the size of digit set and the average joint hamming weight on integer pairs. Note that when h > 2, the number of states of the Markov chain automatically generated by our method is too large to use the exact computation. We can get only the estimated result in this state.

Table 2: Average joint hamming weight when the digit set is {-(2h + 1), ... -3, -1, 0, 1, 3, ..., 2h + 1} and 0 < *h* < 5.

h	Average Joint Hamming Weight
0	0.5000 = ½
1	0.3575 = 281/786
2	0.3100 = 1496396/4826995
3	0.275 (Estimated)
4	0.258 (Estimated)

The application of our algorithm is not limited to the scalar point multiplication. We also apply our algorithm to complex number representations. This is done by considering a real part and an imaginary part of the complex number as an integer pair. We can find the optimal average hamming weight of RCNS [7]. It is 0.66. This improves the digit set conversion on the paper, which is shown that the average hamming weight is 1.43 [8]. Also, we can improve the average joint hamming weight on the extended octagonal representation [8] from 0.67 to 0.50 by improving its representation method.

4 Hamming Weight Distribution

As we analyze the Markov chain for the average joint hamming weight, we are also able to find the distribution of the minimal joint weight. It is the normal distribution, for all digit sets. Also, we can find the expected value and the variance from the Markov chain automatically.

We use that result to propose another countermeasure of the side channel attack. Our work utilizes randomized expansions to make the power attack harder, similar to the work by Ha and Moon [9]. It has the fixed-hamming weight, and makes the timing attack more difficult, as in the work by Mamiya and Miyaji [10]. And, we can improve the time used for computing the scalar-point multiplication on both papers.

5 Future works

Mentioned in Section 2, our algorithm can be applied to any digit sets, but it is comparatively slow. Then, we need to improve the efficiency of the algorithm. Also, we need to reduce the number of states of the Markov chain constructed automatically from our algorithm, to explore larger digit sets.

Recently, there is a work by Doche, Kohel, and Sica [11]. They propose the lower joint weight conversion and the analysis for double-base number system. In this state, we can present the minimal joint hamming weight conversion, but the analysis is still difficult to be found. However, the proposed minimal weight conversion makes us improve the upper bound of the average joint hamming weight. We can improve it from 0.3945 [9] to 0.3932.

Reference

[1] J.A. Solinas, "Low-weight binary representation for pairs of integers", Centre for Applied Cryptographic Research, University of Waterloo, Combinatorics and Optimization Research Report CORR, 2001.

[2] C. Heuberger, J.A. Muir, "Minimal weight and colexicographically minimal integer representations", Journal of Mathematical Cryptology, vol.1, pp. 297-328, 2007.

[3] R. Avanzi, "On multi-exponentiation in cryptography", Cryptology ePrint Archive, 154, 2002.

[4] B. Kuang, Y. Zhu, and Y. Zhang, "An improved algorithm for *uP* + *vQ* using JSF^1_3", LNCS, vol. 3089, pp. 467-478, 2004.

[5] B. Moller, "Fractional windows revisited: Improved signed-digit representations for efficient exponentiation", LNCS, vol. 3506, pp. 137-153, 2005.

[6] E. Dahmen, K. Okeya, and T. Takagi, "A new upper bound for the minimal density of joint representations in elliptic curve cryptosystems", IEICE Transaction on Fundamentals, E90-A(5), pp. 952-959, 2008.

[7] T. Aoki, Y. Ohki, and T. Higuchi, "Redundant complex number arithmetic for high-speed signal processing", In the Proceeding of IEEE Workshop on VLSI Signal Processing VIII, pp. 523-532, 1995.

[8] V. Suppakitpaisarn and M. Edahiro, "The novel representation for redundant complex number arithmetic: extended octagonal representation", In the Proceeding of 11th WAAC, pp. 7-14, 2008.

[9] J.C. Ha and S.J. Moon, "Randomized signed-scalar multiplication of ECC to resist power attacks", LNCS, vol. 2523, pp. 551-563, 2003.

[10] H. Mamiya and A. Miyaji, "Fixed-hamming-weight representation for indistinguishable addition formulae", IPSJ Journal, vol. 47, pp. 2430-2439, 2006.

[11] C. Doche, D.R. Kohel, and F. Sica, "Double-base number system for multi-scalar multiplication", LNCS, vol. 5479, pp. 502-517, 2009.