社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.
ISEC2002-9 (2002-05)

標数2の有限体上の超楕円曲線に対するHarley加算アルゴリズムの拡張

杉崎　大樹[†]　松尾　和人[††]　趙　晋輝[†††]　辻井　重男[†]

[†] 中央大学 理工学部 情報工学科
〒 112-8551 東京都文京区春日 1-13-27
[††] 中央大学 研究開発機構
〒 162-8473 東京都新宿区市谷本村町 42-8
[†††] 中央大学 理工学部 電気電子情報通信工学科
〒 112-8551 東京都文京区春日 1-13-27

あらまし　超楕円曲線を用いた暗号系の構成において，超楕円曲線の因子類の高速加算法は必要不可欠である．最近，Harleyによって2でない標数の有限体上の種数2の超楕円曲線に対する高速加算法(Harley algorithm)が提案された．本論文では標数2の有限体上で定義された種数2の超楕円曲線に対するHarley algorithmの拡張を示す．提案アルゴリズムは，因子類の加算を$I+25M$のコスト，2倍算を$I+27M$のコストで実現可能である.（ここで，IおよびMは定義体上の逆元および乗算のコストを表す.）

キーワード　超楕円暗号，種数2，標数2，加算アルゴリズム，Cantor algorithm，Harley algorithm

An Extension of Harley Addition Algorithm for Hyperelliptic Curves over Finite Fields of Characteristic Two.

Hiroki SUGIZAKI[†], Kazuto MATSUO[††], Jinhui CHAO[†††], and Shigeo TSUJII[†]

[†] Dept. of Information and System Engineering, Chuo University, 1-13-27 Kasuga, Bunkyo-ku, Tokyo, 112-8551 Japan
[††] Research and Development Initiative, Chuo University, 42-8 Ichigayahonmuracho, Shinjuku-ku, Tokyo, 162-8473 Japan
[†††] Dept. of Electrical, Electronic, and Communication Engineering, Chuo University, 1-13-27 Kasuga, Bunkyo-ku, Tokyo, 112-8551 Japan

Abstract　Fast addition algorithm for divisor classes of hyperelliptic curves is of essentially importance for construction of hyperelliptic curve cryptosystems. Recently, a fast addition algorithm for divisor class groups of genus two hyperelliptic curves was proposed by Harley. The algorithm is designed only for curves over finite fields of odd characteristic.

In this paper, we present an extension of Harley's algorithm to curves over finite fields of even characteristic. The proposed algorithm takes $I+25M$ for an addition and $I+27M$ for a doubling, where I and M denote costs for an inversion and a multiplication over the definition finite field respectively.

Key words　Hyperelliptic curve cryptosystems, Genus two, Characteristic two, Addition algorithms, Cantor algorithm, Harley algorithm

1. Introduction

It is known that implementation of practical hyperelliptic curve cryptosystems became possible only when the fast addition algorithm for divisor class groups of hyperelliptic curves by Cantor in [1] was available [7]. Since then improvement of the Cantor algorithm has been a central theme in researches of hyperelliptic curve cryptosystems [7], [12]~[15], aiming to build faster hyperelliptic cryptosystems, e.g. comparable with the elliptic curve cryptosystems. Besides, the original Cantor algorithm was proposed for curves defined over finite fields of odd characteristic. It is extended by Koblitz to curves over finite fields of even characteristic or over $\mathbb{F}_{2^n}$ [7].

Recently, a novel addition algorithm based on a totally different strategy from the Cantor algorithm was proposed by Harley [4]~[6] for divisor class groups of

$$Y^2 = X^5 + f_4X^4 + f_3X^3 + \cdots + f_0 \tag{1}$$

over $\mathbb{F}_q$ of odd characteristic. Hereafter, we will call it the Harley algorithm.

The Harley algorithm is a generalization of the chord-tangent law of elliptic curves to divisor addition of hyperelliptic curves. The other features of the algorithm include adoption of Mumford's representation for representation of divisors, usage of the Chinese remainder theorem, Newton's iteration and the Karatsuba multiplication over finite fields. As a result, the Harley algorithm reduces the computation cost significantly comparing with the Cantor algorithm and its improvements as well. In fact, the improved versions of the Harley algorithm shown in [9], [10], [16] take $I + 25M$ for an addition and $I + 29M$ (or $I + 27M$, if $f_4 = 0$) for a doubling, where I, M denote the costs of an inversion and a multiplication over definition fields respectively. Moreover, as reported in [9] that with the improved Harley algorithm, it is possible to implement hyperelliptic curve cryptosystems with the same encryption rate as the elliptic curve cryptosystems.

In this paper, we show an extension of the Harley algorithm to curves over finite fields of characteristic two. The proposed algorithm takes $I + 25M$ for an addition and $I + 27M$ for a doubling.

2. Preliminaries

2.1 Hyperelliptic curves and their Jacobian varieties

Let n be a positive integer. A genus 2 hyperelliptic curve C over $\mathbb{F}_{2^n}$ is defined as follows:

$$\begin{aligned} C : & Y^2 + H(X)Y = F(X), \\ & H(X) = h_2X^2 + h_1X + h_0, \\ & F(X) = X^5 + f_4X^4 + \cdots + f_0, \end{aligned} \tag{2}$$

where $h_i, f_i \in \mathbb{F}_{2^n}$, and

$$\begin{aligned} \{(x,y) \in \overline{\mathbb{F}}_{2^n}^2 \,|y^2 + H(x)y + F(x) = \\ H(x) = H'(X)y + F'(x) = 0\} = \phi. \end{aligned} \tag{3}$$

Let $P = (x, y)$ be a point on C. The opposite $-P$ of P is defined as $-P = (x, y + H(x))$. Let P_∞ be the point at infinity on C and $-P_\infty = P_\infty$. We call a P such that $P = -P$ a ramification point and a P such that $P \neq -P$ a generic point. A point P is a ramification point, if and only if its X-coordinate is a root of $H(x) = 0$ or $P = P_\infty$.

A divisor $\mathcal{D}$ on C is defined as a finite formal sum

$$\mathcal{D} = \sum_{P_i \in C} m_iP_i, m_i \in \mathbb{Z}. \tag{4}$$

The divisors form an Abelian group $\mathfrak{D}$.

The degree $\deg \mathcal{D}$ of $\mathcal{D}$ is defined as

$$\deg \mathcal{D} = \sum_i m_i. \tag{5}$$

The divisors whose degree are zero form a subgroup $\mathfrak{D}^0$ of $\mathfrak{D}$.

For a rational function f on C, a divisor (f) is defined as

$$(f) = \sum m_zP - \sum m_pQ, \tag{6}$$

where P are zeros of f with multiplicities m_z and Q are poles of f with multiplicities m_p on C. A divisor of such form is called a principal divisor. The set of principal divisors is a subgroup $\mathfrak{D}^l$ of $\mathfrak{D}^0$.

The Jacobian variety of C is defined as

$$\mathcal{J}_C = \mathfrak{D}^0/\mathfrak{D}^l. \tag{7}$$

The divisor classes in $\mathcal{J}_C$ fixed by the 2^nth-power

Frobenius map form a subgroup $\mathcal{J}_C(\mathbb{F}_{2^n})$ of $\mathcal{J}_C$. It is a finite Abelian group, therefore can be used to define discrete logarithm problems.

This paper will consider addition of $\mathcal{D} \in \mathcal{J}_C(\mathbb{F}_{2^n})$.

2.2 Ordinary Jacobian varieties and isomorphic curves

For a positive integer r,

$$\mathcal{J}_C[r] = \{\mathcal{D} \in \mathcal{J}_C \mid r\mathcal{D} = 0\} \tag{8}$$

is called the r–torsion group of $\mathcal{J}_C$.

$\mathcal{J}_C$ is called ordinary, if and only if

$$\mathcal{J}_C[2] \cong (\mathbb{Z}/2\mathbb{Z})^2. \tag{9}$$

i.e., $\mathcal{J}_C$ is ordinary, if and only if $\deg H = 2$ in (2).

We assume $\mathcal{J}_C$ is ordinary hereafter.

If $\mathcal{J}_C$ is ordinary, the definition of C can be transformed into the form

$$\begin{aligned} C/\mathbb{F}_{2^n} :& Y^2 + H(X)Y = F(X), \\ & H(X) = X^2 + h_1X + h_0, \\ & F(X) = X^5 + f_3X^3 + f_1X + f_0 \end{aligned} \tag{10}$$

by an isomorphism.

Therefore, (10) will be used as the definition equation of C hereafter.

2.3 Representation of divisor classes

For $\mathcal{D}_1, \mathcal{D}_2 \in \mathfrak{D}^0$, we use $\mathcal{D}_1 \sim \mathcal{D}_2$ to mean that $\mathcal{D}_1 - \mathcal{D}_2 \in \mathfrak{D}^l$.

A divisor class of $\mathcal{J}_C$ can be represented by a divisor in the following form:

$$\mathcal{D} = \sum_i m_iP_i - \sum_i m_iP_\infty, \ m_i \geqq 0 \tag{11}$$

where, $P_i \neq -P_j$ for $\forall i \neq j$.

Such a divisor $\mathcal{D}$ in (11) is called semi–reduced. $\sum_i m_i$ is defined as the weight of $\mathcal{D}$[4]. A semi-reduced divisor with weight less than the genus is called a reduced divisor. Then any divisor class $\mathcal{D} \in \mathcal{J}_C$ can be uniquely represented by a reduced divisor.

Furthermore, a semi–reduced divisor $\mathcal{D}$ can be represented by a pair of polynomials

$$\mathcal{D} = (U, V), \tag{12}$$

where $U, V \in \overline{\mathbb{F}}_{2^n}[X]$.

Denote $P_i = (x_i, y_i)$,

$$U = \prod (X + x_i)^{m_i} \tag{13}$$

and V is the unique polynomial satisfies

$$F + HV + V^2 \equiv 0 \bmod U, \deg V < \deg U. \tag{14}$$

Moreover,

$$y_i = V(x_i) \tag{15}$$

for $P_i = (x_i, y_i)$ in (11).

We call such a representation of $\mathcal{D}$ given by (12) as Mumford's representation.

Notice that $\mathcal{D} = (U, V)$, $U, V \in \mathbb{F}_{2^n}[X]$ is equivalent to $\mathcal{D} \in \mathcal{J}_C(\mathbb{F}_{2^n})$. Therefore we assume hereafter $U, V \in \mathbb{F}_{2^n}[X]$.

For a weight two divisor $\mathcal{D} = (U, V)$,

$$-\mathcal{D} = (U, U + V + H). \tag{16}$$

For a weight one divisor $\mathcal{D} = (X + u_0, v_0)$,

$$-\mathcal{D} = (X + u_0, v_0 + H(u_0)). \tag{17}$$

3. The Harley algorithm for curves over odd characteristic fields [5]

This section outlines the Harley algorithm for a hyperelliptic curve (1) according to [4], [5].

For the curve (1), Mumford's representation of its semi–reduced divisors can also be defined [1], if one replaces (13) with

$$U = \prod (X - x_i)^{m_i}, \tag{18}$$

(14) with

$$F - V^2 \equiv 0 \bmod U, \deg V < \deg U, \tag{19}$$

and (16) with

$$-\mathcal{D} = (U, -V). \tag{20}$$

The Harley algorithm consists of two different computation procedures, one for addition $\mathcal{D}_3 = \mathcal{D}_1 + \mathcal{D}_2$, and the other for doubling $\mathcal{D}_2 = 2\mathcal{D}_1$.

In the addition procedure, all I/O divisors $\mathcal{D}_1 = (U_1, V_1), \mathcal{D}_2 = (U_2, V_2), \mathcal{D}_3 = (U_3, V_3)$ are assumed to be reduced.

If the size of the definition field $\mathbb{F}_q$ is large enough, the weights of both $\mathcal{D}_1, \mathcal{D}_2$ almost always equal two. Besides, $\mathcal{D}_1, \mathcal{D}_2$ do not contain the same point or points opposite to each others. In other words, $\deg U_1 = \deg U_2 = 2$ and $\gcd(U_1, U_2) = 1$. We will call such case the most frequent case in this paper.

Below, we sketch roughly the addition procedure of Harley algorithm for the most frequent case, which consists of composition and reduction.

In the composition part, one computes the semi-reduced divisor $\mathcal{D} = (U, V)$ such that $\mathcal{D} \sim -\mathcal{D}_3$ and $U = U_1U_2$. V are obtained by the Chinese remainder theorem. Then in the reduction part, one computes a reduced divisor such that $\mathcal{D}_3 \sim -\mathcal{D}$.

In the doubling procedure, both the input divisor $\mathcal{D}_1 = (U_1, V_1)$ and the output divisor $\mathcal{D}_2 = (U_2, V_2)$ are also assumed to be reduced.

Once again, if the size of $\mathbb{F}_q$ is large enough, the weight of $\mathcal{D}_1$ almost always equals two. Besides, $\mathcal{D}_1$ does not contain ramification points except P_∞. In other words, $\deg U_1 = 2$ and $\gcd(U_1, V_1) = 1$, which is also called the most frequent case.

Similar to the addition, the doubling procedure for the most frequent case also consists of the composition part and the reduction part, where the Chinese remainder theorem is replaced by Newton's iteration.

Besides, one may notice that when $\gcd(U_1, U_2) \neq 1$, the most frequent case addition procedure fails since the Chinese remainder theorem does not hold. The most frequent case doubling also fails in the case $\gcd(U_1, V_1) \neq 1$, because Newton's iteration can not be applied. In these cases, different procedures should be used.

In fact, the Harley algorithm contains various procedures each corresponding to different weights of input divisors. Therefore, classification of input divisor classes is necessary before composition. See [5], [9] for details of classification.

4. Most Frequent Case Algorithm

In this section we extend the Harley algorithm to a hyperelliptic curve (10) over $\mathbb{F}_{2^n}$ and show procedures in the most frequent case.

Hereafter, small letters are used to denote elements of $\mathbb{F}_{2^n}$ and capital letters denote polynomials in X over $\mathbb{F}_{2^n}$. The coefficient of X^i in $T \in \mathbb{F}_{2^n}[X]$ is as t_i.

4.1 Most Frequent Case Addition Algorithm

Below, we show a procedure for addition $\mathcal{D}_3 = \mathcal{D}_1 + \mathcal{D}_2$, $\mathcal{D}_i = (U_i, V_i) \in \mathcal{J}_C(\mathbb{F}_{2^n})$ in the most frequent case when $\deg U_1 = \deg U_2 = 2$ and $\gcd(U_1, U_2) = 1$.

In the composition part, one computes the semi-reduced divisor $\mathcal{D} = (U, V)$ such that $\mathcal{D} \sim -\mathcal{D}_3$ and $U = U_1U_2$. Here V is obtained as

$$V = SU_1 + V_1, \tag{21}$$

$$S \equiv (V_2 + V_1)U_1^{-1} \bmod U_2, \ \deg S \leqq 1 \tag{22}$$

by applying the Chinese remainder theorem to

$$V \equiv V_1 \bmod U_1, \tag{23}$$

$$V \equiv V_2 \bmod U_2. \tag{24}$$

In the first place of the reduction part, one computes a reduced divisor $\mathcal{D}_3'$ such that $\mathcal{D}_3' \sim \mathcal{D}$. U_3' is computed as

$$U_3' = s_1^{-2}\frac{F + HV + V^2}{U} \tag{25}$$

according to [9], [10]. In fact, when $s_1 = 0$ one needs another procedure, which is omitted here due to the space limitation.

V_3' is obtained as

$$V_3' = S(U_1 + U_3') + s_1(u_{11} + u_{31}')U_3' + V_1 \tag{26}$$

from

$$V_3' \equiv V \bmod U_3' \tag{27}$$

and (21).

Finally, the output divisor $\mathcal{D}_3 = -\mathcal{D}_3'$ is given by

$$\mathcal{D}_3 = (U_3, V_3) = (U_3', U_3' + V_3' + H) \tag{28}$$

according to (16).

In practice, U_3' is computed by substituting (21) for (25) as in [9], [10], rather than using (21) directly. The Karatsuba multiplication is used for multiplication over $\mathbb{F}_{2^n}$ in (22) and (26). Moreover, similar to [10], two inversions required in (22) and (25) can be replaced with one inversion and four multiplications by Montgomery's multiple inversion technique [3].

Consequently, we obtain a most frequent case addition procedure which costs $I + 25M$ (or $2I + 21M$

without the multiple inversion).

Table 1 in Appendix shows further details of the proposed addition algorithm and the cost of each step in the algorithm.

4.2 Most Frequent Case Doubling Algorithm

Below, we show a procedure for doubling $\mathcal{D}_2 = 2\mathcal{D}_1$ in the most frequent case, i.e., $\deg U_1 = 2$ and $\gcd(U_1, H) = 1$. Notice that one needs different conditions from those for (1) in Section 3. of odd characteristic.

Firstly, one computes the semi-reduced divisor $\mathcal{D} = (U, V)$ such that $\mathcal{D} \sim -\mathcal{D}_2$ and $U = U_1^2$ in the composition part. Here V is obtained as

$$V = SU_1 + V_1, \tag{29}$$

$$S \equiv \frac{F + HV_1 + V_1^2}{U_1} H^{-1} \bmod U_1, \ \deg S \leqq 1 \tag{30}$$

by applying Newton's iteration to

$$V \equiv V_1 \bmod U_1. \tag{31}$$

The reduction part follows the same steps as for addition. i.e., the output divisor $\mathcal{D}_2 = (U_2, V_2)$ is obtained as

$$U_2 = s_1^{-2} \frac{F + HV + V^2}{U}, \tag{32}$$

$$V_2 = S(U_1 + U_2) + (s_1(u_{11} + u_{21}) + 1) U_2 + V_1 + H. \tag{33}$$

Similar to the addition procedure, the doubling procedure is in practice further tuned up by using techniques such as the Karatsuba multiplication, and the multiple inversion technique.

Consequently, the procedure for the most frequent case doubling costs $I + 27M$ (or $2I + 23M$ without the multiple inversion).

Further details of the proposed doubling algorithm and the cost of each step can be found in Table 2 in Appendix.

Remark 1. The reduction procedures for V described here are applicable to the improved Harley algorithm for (1) shown in [9], [10]. For example, V_3 can be computed as

$$V_3 = S(U_3 - U_1) - s_1(u_{31} - u_{11})U_3 - V_1 \tag{34}$$

in the addition. Using this technique, the improved Harley algorithm needs $I + 25M$ for an addition and $I + 28M$ (or $I + 26M$, if $f_4 = 0$) for a doubling, or $2I + 21M$ for an addition and $2I + 24M$ (or $2I + 22M$, if $f_4 = 0$) for a doubling without the multiple inversion.

5. Algorithms for Other Cases

This section outlines procedures of the proposed algorithm for cases other than the most frequent case. Classification of input divisor classes is also discussed here.

In fact, the classification of input divisor classes for addition is the same as that given in [5]. As shown in 4.1, the other procedures can also be easily obtained by modifying those given in [5]. Therefore, we will only give an outline for doubling, i.e. to compute $\mathcal{D}_2 = (U_2, V_2) = 2\mathcal{D}_1$ for $\mathcal{D}_1 = (U_1, V_1)$.

An input divisor $\mathcal{D}_1$ for doubling can be classified according to its weight, or $\deg U_1$ [5].

Since when the weight of $\mathcal{D}_1$ is zero, $\mathcal{D}_2 = 0$, we will only show the procedures for the cases when $\mathcal{D}_1$ has weight one and two.

5.1 Doubling of weight one divisors

Let

$$\mathcal{D}_1 = (U_1, V_1) = P - P_\infty, \tag{35}$$

then $P = (u_{10}, v_{10})$ obviously.

To double a weight one divisor $\mathcal{D}_1$ requires to detect whether P is a ramification point or not.

If H is irreducible over $\mathbb{F}_{2^n}$ then P is a generic point since

$$P : \text{a ramification point} \Leftrightarrow \gcd(U_1, H) = U_1. \tag{36}$$

If H is reducible over $\mathbb{F}_{2^n}$, one has to precompute the roots of H over $\mathbb{F}_{2^n}$, α_i, $i = 1, 2$. If

$$u_{10} = \alpha_i \tag{37}$$

for $i = 1$ or 2, then P is a ramification point and $\mathcal{D}_2 = 0$.

Now, we assume P to be a generic point.

In this case, U_2 is obtained as

$$U_2 = U_1^2 = X^2 + u_{10}^2, \tag{38}$$

and V_2 as

$$v_{21} = \frac{F'(u_{10}) + H'(u_{10})v_{10}}{H(u_{10})}, \tag{39}$$

$$v_{20} = u_{10}v_{21} + v_{10}, \tag{40}$$

because V_2 is the tangent of C at P and

$$\frac{dY}{dX} = \frac{F' + H'Y}{H}. \tag{41}$$

5.2 Doubling of weight two divisors

Doubling of a weight two divisor $\mathcal{D}_1$, also requires classification of the divisor by the number of ramification points in $\mathcal{D}_1$. This can be done by checking $\gcd(U_1, H_1)$.

Firstly, the case $\gcd(U_1, H) = 1$ is detected by computing the resultant of U_1 and H since

$$\gcd(U_1, H) = 1 \Leftrightarrow \mathrm{res}(U_1, H) \neq 0. \tag{42}$$

If $\gcd(U_1, H) = 1$ then $\mathcal{D}_1$ does not contain ramification points except P_∞, which is the most frequent case. The procedure for this case has been shown in 4.2. See Step 1 in Table 2 for computation of $\mathrm{res}(U_1, H)$.

When $\gcd(U_1, H) \neq 1$, $\gcd(U_1, H)$ is a polynomial of degree one or two.

The case $\deg\gcd(U_1, H) = 2$ can be easily detected from

$$\deg\gcd(U_1, H) = 2 \Leftrightarrow U_1 = H. \tag{43}$$

In this case, $\mathcal{D}_1$ consists of ramification points and $\mathcal{D}_2 = 0$.

Finally, we show the procedure for the case $\deg\gcd(U_1, H) = 1$. If H is irreducible over $\mathbb{F}_{2^n}$, $\deg\gcd(U_1, H) \neq 1$ and the procedure is unnecessary, so we assume H to be reducible over $\mathbb{F}_{2^n}$.

Let

$$\mathcal{D}_1 = P + P_r - 2P_\infty \tag{44}$$

and P_r be a ramification point. Then,

$$\mathcal{D}_2 = 2\mathcal{D}'_1,\ \mathcal{D}'_1 = P - P_\infty \tag{45}$$

because

$$2(P_r - P_\infty) = 0. \tag{46}$$

Therefore, $\mathcal{D}_2$ can be obtained from (45) by the procedure described in 5.1, if the weight one divisor $\mathcal{D}'_1$ is known.

Using the roots α_1, α_2 of H over $\mathbb{F}_{2^n}$ given by precomputation, one can found $\mathcal{D}'_1 = (U'_1, V'_1)$ as follows.

The X-coordinate of P_r equals α_1, if $U_1(\alpha_1) = 0$. Otherwise, it equals α_2 because it is either α_1 or α_2. If α_1 is the X-coordinate of P_r, $\mathcal{D}'_1$ can obtained as

$$U'_1 = X + u_{11} + \alpha_1, \tag{47}$$

$$V'_1 = v_{11}u'_{10} + v_{10}, \tag{48}$$

because $u_{11} + \alpha_1$ is the X-coordinate of P and $v_{11}(u_{11} + \alpha_1) + v_{10}$ is the Y-coordinate.

6. Conclusion

This paper proposed an extension of the Harley algorithm to curves over finite fields of characteristic two. The proposed algorithm takes $I + 25M$ for an addition and $I + 27M$ for a doubling.

Acknowledgement A part of this research was supported by Telecommunications Advancement Organization of Japan (TAO).

References

[1] D. G. Cantor. Computing in the Jacobian of hyperelliptic curve. *Math. Comp.*, Vol. 48, No. 177, pp. 95–101, 1987.

[2] J. W. S. Cassels and E. V. Flynn. *Prolegomena to middlebrow arithmetic of curves of genus 2.* No. 230 in London Mathematical Society Lecture Note Series. Cambridge U. P., 1996.

[3] H. Cohen. *A Course in Computational Algebraic Number Theory.* No. 138 in Graduate Text in Mathematics. Springer-Verlag, 1993.

[4] P. Gaudry and R. Harley. Counting points on hyperelliptic curves over finite fields. In W. Bosma, editor, *ANTS-IV*, No. 1838 in Lecture Notes in Computer Science, pp. 297–312. Springer-Verlag, 2000.

[5] R. Harley. adding.text. `http://cristal.inria.fr/~harley/hyper/`, 2000.

[6] R. Harley. doubling.c. `http://cristal.inria.fr/~harley/hyper/`, 2000.

[7] N. Koblitz. Hyperelliptic curve cryptosystems. *J. Cryptology*, Vol. 1, No. 3, pp. 139–150, 1989.

[8] N. Koblitz. *Algebraic Aspects of Cryptography.* No. 3 in Algorithms and Computation in Mathematics. Springer-Verlag, 1998.

[9] K. Matsuo, J. Chao, and S. Tsujii. Fast genus two hyperelliptic curve cryptosystems. Technical Report ISEC2001-31, IEICE Japan, 2001.

[10] Y. Miyamoto, H. Doi, K. Matsuo, J. Chao, and S. Tsujii. Fast addition algorithm for genus two hyperelliptic curves. In *Proc. of SCIS2002*, pp. 497–502, 2002 (in Japanese).

[11] D. Mumford. *Tata Lectures on Theta II.* No. 43 in Progress in Mathematics. Birkhäuser, 1984.

[12] K. Nagao. Improving group law algorithms for Jacobians of hyperelliptic curves. In W. Bosma, editor, *ANTS-IV*, No. 1838 in Lecture Notes in Computer Science, pp. 439–448. Springer-Verlag, 2000.

[13] S. Paulus and A. Stein. Comparing real and imaginary arithmetics for divisor class groups of hyperelliptic curves. In *ANTS-III*, No. 1423 in Lecture Notes in Computer Science, pp. 576–591. Springer-Verlag, 1998.

[14] Y. Sakai and K. Sakurai. Design of hyperelliptic cryptosystems in small characteristic and a software implementation over $\mathbf{F}_{2^n}$. In K. Ohta and D. Pei, editors, *Advances in Cryptology - ASIACRYPT'98*, No. 1514 in Lecture Notes in Computer Science, pp. 80–94. Springer-Verlag, 1998.

[15] Y. Sakai, K. Sakurai, and H. Ishizuka. Secure hyperelliptic cryptosystems and their performance. In H. Imai and Y. Zheng, editors, *Public Key Cryptography*, No. 1431 in Lecture Notes in Computer Science, pp. 164–181. Springer-Verlag, 1998.

[16] M. Takahashi. Improving Harley algorithms for Jacobians of genus 2 hyperelliptic curves. In *Proc. of SCIS2002*, pp. 155–160, 2002 (in Japanese).

Appendix: Details of the most frequent case algorithms

Input	Weight two reduced divisors : $\mathcal{D}_1 = (U_1, V_1)$ and $\mathcal{D}_2 = (U_2, V_2)$, $C : Y^2 + H(X)Y = F(X)$	
Output	A weight two reduced divisor $\mathcal{D}_3 = (U_3, V_3) = \mathcal{D}_1 + \mathcal{D}_2$	
Step	Procedure	Cost
1	**Compute the resultant r of U_1 and U_2.** $w_1 \leftarrow u_{11} + u_{21};\ w_0 \leftarrow u_{21}w_1 + u_{10} + u_{20};\ r \leftarrow (u_{10} + u_{20})w_0 + u_{20}w_1^2.$	$4M$
2	**If $r = 0$ then call another procedure.**	—
3	**Compute $I = i_1X + i_0 \equiv rU_1^{-1} \bmod U_2$.** $i_1 \leftarrow w_1; i_0 \leftarrow w_0.$	—
4	**Compute $T = t_1X + t_0 \equiv (V_1 + V_2)I \bmod U_2$.** $t_2 \leftarrow (v_{11} + v_{21})w_1;\ t_0 \leftarrow (v_{10} + v_{20})w_0;\ t_1 \leftarrow (v_{11} + v_{21} + v_{10} + v_{20})(w_1 + w_0) + t_2 + t_0;$ $t_1 \leftarrow t_1 + t_2u_{21};\ t_0 \leftarrow t_0 + t_2u_{20}.$	$5M$
5	**If $t_1 = 0$ then call the sub-procedure.**	—
6	**Compute $S = s_1X + s_0$. (Multiple inversion technique)** $w_2 \leftarrow (rt_1)^{-1};\ w_3 \leftarrow w_2r;\ w_4 \leftarrow w_2t_1;\ w_5 \leftarrow w_3r; s_1 \leftarrow w_4t_1;\ s_0 \leftarrow w_4t_0.$	$I + 6M$
7	**Compute $U_3 = X^2 + u_{31}X + u_{30} = s_1^{-2}(F + H(SU_1 + V_1) + (SU_1 + V_1)^2)/(U_1U_2)$.** $u_{31} \leftarrow w_1 + w_5(1 + w_5);\ u_{30} \leftarrow u_{21}w_1 + u_{10} + u_{20} + w_5(w_1 + w_5(s_0 + s_0^2 + w_1)).$	$5M$
8	**Compute $V_3 = v_{31}X + v_{30} \equiv SU_1 + V_1 + H \bmod U_3$.** $w_1 \leftarrow u_{11} + u_{31};\ w_0 \leftarrow u_{10} + u_{30};\ w_2 \leftarrow s_1w_1;\ w_3 \leftarrow s_0w_0;$ $w_4 \leftarrow (s_1 + s_0)(w_1 + w_0) + w_2 + w_3;\ w_2 \leftarrow w_2 + 1;\ w_1 \leftarrow w_4 + w_2u_{31};\ w_0 \leftarrow w_3 + w_2u_{30};$ $v_{31} \leftarrow w_1 + v_{11} + h_1;\ v_{30} \leftarrow w_0 + v_{10} + h_0.$	$5M$
Total		$I + 25M$

Table 1 Addition for weight two coprime divisors (Most frequent case addition algorithm)

Input	A weight two reduced divisor : $\mathcal{D}_1 = (U_1, V_1)$ $C : Y^2 + H(X)Y = F(X)$	
Output	A weight two reduced divisor $\mathcal{D}_2 = (U_2, V_2) = 2\mathcal{D}_1$	
Step	Procedure	Cost
1	**Compute the resultant r of U_1 and H.** $w_1 \leftarrow h_1 + u_{11};\ w_0 \leftarrow h_0 + u_{10} + u_{11}w_1;\ r \leftarrow u_{10}(u_{10} + h_0 + h_1w_1) + h_0w_0.$	$4M$
2	**If $r = 0$ then call another procedure.**	—
3	**Compute $I = i_1X + i_0 \equiv rH^{-1} \bmod U_1$.** $i_1 \leftarrow w_1;\ i_0 \leftarrow w_0.$	—
4	**Compute $T = t_1X + t_0 \equiv I(F + HV_1 + V_1^2)/U_1 \bmod U_1$.** $w_2 \leftarrow f_3 + v_{11} + u_{11}^2;\ w_3 \leftarrow v_{10} + v_{11}(v_{11} + h_1);$ $t_1 \leftarrow w_0w_2 + w_1w_3;\ t_0 \leftarrow (u_{11}w_0 + u_{10}w_1)w_2 + w_0w_3.$	$8M$
5	**If $t_1 = 0$ then call the sub-procedure.**	—
6	**Compute $S = s_1X + s_0$. (Multiple inversion technique)** $w_0 \leftarrow (rt_1)^{-1};\ w_2 \leftarrow w_0r;\ w_3 \leftarrow w_0t_1;\ w_4 \leftarrow w_2r;\ s_1 \leftarrow w_3t_1;\ s_0 \leftarrow w_3t_0.$	$I + 6M$
7	**Compute $U_2 = X^2 + u_{21}X + u_{20} = s_1^{-2}(F + H(SU_1 + V_1) + (SU_1 + V_1)^2)/U_1^2$.** $u_{21} \leftarrow w_4(1 + w_4);\ u_{20} \leftarrow w_4(w_4(s_0(1 + s_0)) + w_1).$	$4M$
8	**Compute $V_2 = v_{21}X + v_{20} \equiv SU_1 + V_1 + H \bmod U_2$.** $w_1 \leftarrow u_{11} + u_{21};\ w_0 \leftarrow u_{10} + u_{20};\ w_2 \leftarrow s_1w_1;\ w_3 \leftarrow s_0w_0;$ $w_4 \leftarrow (s_1 + s_0)(w_1 + w_0) + w_2 + w_3;\ w_2 \leftarrow w_2 + 1;\ w_1 \leftarrow w_4 + w_2u_{21};\ w_0 \leftarrow w_3 + w_2u_{20};$ $v_{21} \leftarrow w_1 + v_{11} + h_1;\ v_{20} \leftarrow w_0 + v_{10} + h_0.$	$5M$
Total		$I + 27M$

Table 2 Doubling for weight two coprime divisors (Most frequent case doubling algorithm)