SCIS 2002 The 2002 Symposium on Cryptography and Information Security
Shirahama,Japan, Jan.29-Feb.1,2002
The Institute of Electronics, Information and Communication Engineers

種数2の超楕円曲線上の因子類群の高速演算法に関する考察
A Fast Addition Algorithm of Genus Two Hyperelliptic Curves

宮本 洋輔* 土井 洋† 松尾 和人† 趙 晋輝‡
Yosuke MIYAMOTO Hiroshi DOI Kazuto MATSUO Jinhui CHAO
辻井 重男*
Shigeo TSUJII

あらまし 高速な超楕円曲線暗号系を構成するためには超楕円曲線上の因子類群の加算の高速化が不可欠である．最近，Harleyによって因子類群の高速加算法が提案され，また，松尾，趙，辻井によりその改良が行われた．これらの加算法を用いることで楕円曲線暗号系と同等の速度で超楕円曲線暗号系を構成可能になった．しかし，これらのアルゴリズムでは基礎体上の逆元計算が2回必要であった．そこで本論文では計算コストの高い逆元計算の回数削減を中心にHarleyアルゴリズムの改良を行い，逆元計算を1回のみ必要とするアルゴリズムと，変形Mumford表現を利用した逆元計算を必要としないアルゴリズムを示す．

キーワード 超楕円曲線暗号，高速加算法，Cantorアルゴリズム，Harleyアルゴリズム，Mumford表現，Montgomery逆元計算

1 はじめに

超楕円曲線のJacobi多様体を用いた暗号系は，Cantorによって提案された因子類群の高速加算アルゴリズム[Can87]によって実現可能になり，以後，超楕円曲線暗号系に関する研究が数多く行われてきた．特に，Cantorアルゴリズムについての多くの改良が提案されてきた[Kob89, PS98, SSI98, SS98, Sma99, Nag00]．

近年，GaudryとHarleyにより，種数2に限った高速加算法(以下Harleyアルゴリズム)が提案された[GH00, Har00a, Har00b]．Harleyアルゴリズムでは，Newton反復法，中国人剰余定理，更にKaratsuba乗算法を利用することにより，Cantorアルゴリズムの1/2程度の計算コストで因子類群の加算を実現した．更に，松尾，趙，辻井はHarleyアルゴリズムに対する改良を行い，厳密な速度評価を行った．その結果，種数2の場合は超楕円曲線暗号系は楕円曲線暗号系と同等の演算速度を達成し得ることが示された[MCT01]．

本研究では，計算コストの高い逆元計算の回数削減を中心に，Harleyアルゴリズムの改良を行い，Montgomery逆元計算を用いる逆元計算を1回だけ必要とするアルゴリズム，および，変形Mumford表現を利用する逆元計算を必要としないアルゴリズムを示し，計算コストを評価する．

* 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, Department of Information System Engineering, Chuo University 1-13-27, Kasuga Bunkyo-ku, Tokyo 112-8551 JAPAN
† 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, Research and Development Initiative, Chuo University, 42-8 Ichigaya Honmura-cho, Shinjuku-ku, Tokyo, 162-8473 JAPAN
‡ 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, Department of Electrical,Electronic, and Communication Engineering, Chuo University, 1-13-27 Kasuga, Bunkyo-ku, Tokyo, 112-8551 JAPAN

2 準備

定義 2.1. **(種数2の超楕円曲線)**
p を奇素数，n を正整数とし，$q = p^n$ とする．このとき $\mathbb{F}_q$ 上定義された種数2の超楕円曲線 $C/\mathbb{F}_q$ を

$$\begin{aligned} C : Y^2 &= F(X) \\ F(X) &= X^5 + f_4X^4 + f_3X^3 + f_2X^2 + f_1X + f_0 \end{aligned}$$

で定義する．ただし $F(X) \in \mathbb{F}_q[X]$ であり，$F(X)$ は重根を持たないものとする．また，P_∞ で唯一の無限遠点を表す． □

$P = (x, y) \in C$ に対し，$-P = (x, -y)$ と定義する．また $-P_\infty = P_\infty$ とする．

超楕円曲線 C 上の有限個の点 $P_1, ..., P_r$ の形式的有限和

$$\mathcal{D} = \sum_{P_i \in C} \mathrm{ord}_{P_i}(\mathcal{D}) P_i, \quad \mathrm{ord}_{P_i}(\mathcal{D}) \in \mathbb{Z}$$

を因子 (divisor) とよび，その次数を $\sum_{P_i \in C} \mathrm{ord}_{P_i}(\mathcal{D})$ で定義する．次数0の因子全体を $\mathcal{D}^0$ で表し，主因子全体を $\mathcal{D}^l$ で表す．次数0の因子類群 $\mathcal{D}^0/\mathcal{D}^l$ を曲線 C のJacobi多様体 $\mathcal{J}_C$ と呼ぶ．$\mathcal{J}_C$ の $\mathbb{F}_q$-有理点集合を $\mathcal{J}_C(\mathbb{F}_q)$ と書く．$\mathcal{J}_C(\mathbb{F}_q)$ は有限アーベル群であり，この上で離散対数問題に基づく暗号系を構成可能である．

$\mathcal{J}_C$ の元 $\mathcal{D}$ は以下の形式で表現可能である．

$$\mathcal{D} = \sum_i \mathrm{ord}_{P_i}(\mathcal{D}) P_i - \Big(\sum_i \mathrm{ord}_{P_i}(\mathcal{D})\Big) P_\infty \ , \mathrm{ord}_{P_i}(\mathcal{D}) \geq 0 \tag{1}$$

ただし $\mathrm{ord}_{P_i}(\mathcal{D}) > 0$ なる P_i に対し $\mathrm{ord}_{-P_i}(\mathcal{D}) = 0$ である．上記の因子を semi-reduced divisorといい $\sum_i \mathrm{ord}_{P_i}(\mathcal{D})$

を因子 $\mathcal{D}$ の weight と呼ぶ [GH00]. また，特に weight が種数以下である semi-reduced divisor を reduced divisor と呼ぶ．Reduced divisor によって $\mathcal{J}_C$ の元を一意に表現可能である．そこで実際には $\mathcal{J}_C$ 上の加算は reduced divisor を用いた演算により実現される．

Semi-reduced divisor は以下で定義される多項式の対で表現可能である [Mum84].

定義 2.2. 種数 g の超楕円曲線 C が与えられた時，式 (1) で与えられる semi-reduced divisor $\mathcal{D}$ の Mumford 表現とは，多項式の対 $\mathcal{D} = (U, V)$ である．ただし，点 P_i の x 座標と y 座標を各々 x_i, y_i とすると ,$U, V \in \mathbb{F}_q[X]$ は，

$$U = \prod (X - x_i)^{\mathrm{ord}_{P_i}(\mathcal{D})} \quad (2)$$

$$V : F - V^2 \equiv 0 \bmod U \ , \deg V < \deg U \quad (3)$$

を満たす． □

U の次数が g 以下のとき，$\mathcal{D}$ は reduced divisor である．

3 Harley アルゴリズム

本章では，Harley アルゴリズム [GH00] と，[MCT01] に示された Harley アルゴリズムの改良について概略を説明する．

Harley アルゴリズムでは，[Mum84] に見られる理論を利用し，入力因子の詳細な場合分けを行うことで，重複する処理を省き無駄のない処理を行っている．また多項式の演算について係数レベルで最適化を行い，定義体上の演算回数 (乗算, 除算) を削減させることにより高速化を達成した．

q が十分に大きいとき，ほとんどの加算演算において，その入力 $\mathcal{D}_1 = (U_1, V_1)$, $\mathcal{D}_2 = (U_2, V_2)$ は， weight が 2 で，$\gcd(U_1, U_2) = 1$ となる．この場合，Harley アルゴリズムは $\mathcal{D}_3 = (U_3, V_3) = \mathcal{D}_1 + \mathcal{D}_2$ を次の手順で計算する．まず $U = U_1 U_2$ を求め，V は，

$$V \equiv V_1 \bmod U_1, \quad (4)$$

$$V \equiv SU_1 + V_1 \bmod U, \quad S \in \mathbb{F}_q[X] \quad (5)$$

を満たすように求める．S は中国人剰余定理を利用し，

$$S \equiv \frac{V_2 - V_1}{U_1} \bmod U_2 \quad (6)$$

として求めることができる．

このようにして求めた $\mathcal{D}' = (U, V)$ は $-\mathcal{D}_3$ と同値な semi-reduced divisor になる．そこで $-\mathcal{D}'$ を還元し，最終的に reduced divisor $\mathcal{D}_3$ を得る．

2 倍算は中国人剰余定理に代えて Newton 反復法を用いることで実現される．

[MCT01] では，Harley アルゴリズムの詳細な最適化を図り，その高速化を実現した．この改良 Harley アルゴリズムは加算を $2I + 23M$，2 倍算を $2I + 25M$ のコストで実現可能であり，群位数が同程度の楕円曲線の加算アルゴリズムと同等の速度で加算演算を実現可能である．ここで，M, I は基礎体上の乗算，逆元計算のコストを各々表す．

4 逆元計算を 1 回のみ必要とするアルゴリズム

Harley アルゴリズムと [MCT01] におけるその改良では加算演算に有限体上の逆元計算を 2 回必要とする．一般に逆元計算は多くのコストを必要とし，この回数を削減することはアルゴリズムの実際的な高速化に大きく寄与する．

そこで，本章では Harley アルゴリズムの逆元計算回数を 1 回に削減する．また，超楕円曲線 C の同型変換を用いて曲線のパラメータ数を減らすことで，加算コストを削減可能であることを述べる．

4.1 Montgomery 逆元計算の利用

有限体上の独立な k 個の元の逆元計算を 1 回の逆元計算と $3k - 3$ 回の乗算で実現する方法が知られている [Coh93]. 本論文では，この方法を Montgomery 逆元計算と呼ぶ．

Harley アルゴリズムに現われる逆元計算は独立に計算できるものではないが，若干の修正を加えることで Montgomery 逆元計算を適用可能である．実際，加算の場合は与えられた $rs, r \in \mathbb{F}_q$ に対し，r^{-1}, s^{-1} を計算する必要があり，この計算は 1 回の逆元計算と 4 回の乗算によって実現される．したがって，乗算と逆元計算のコスト差が $I > 4M$ の場合は，Montgomery 逆元計算を利用することにより従来アルゴリズムより高速な加算を実現可能である．この逆元計算を以下に示す．

Algorithm 4.1. (Montgomery)

Input $s_1'(= rs_1)$, $r \in \mathbb{F}_q$

Output $s_1^{-1}, r^{-1} \in \mathbb{F}_q$

1	$w_1 \leftarrow rs_1'$	(Multiplication)
2	$w_1 \leftarrow w_1^{-1}$	(Inversion)
3	$r^{-1} \leftarrow w_1 s_1'$	(Multiplication)
4	$w_2 \leftarrow r^2$	(Multiplication)
5	$s_1^{-1} \leftarrow w_1 w_2$	(Multiplication)

□

同様に，2 倍算の場合も 2 回の逆元計算を，1 回の逆元計算と 4 回の乗算で実現できる．

4.2 曲線の制限

従来，超楕円曲線 C の定義方程式として，定義 2.1 で与えられる形を考えてきた．しかし，$p \neq 5$ を仮定とすると，変換 $(X, Y) \mapsto \left(X + \frac{f_4}{5}, Y\right)$ により C と $\mathbb{F}_q$ 上同型な超楕円曲線

$$C' : Y^2 = F(X),$$
$$F(X) = X^5 + f_3 X^3 + f_2 X^2 + f_1 X + f_0$$

を得る．本論文では，これを標準形と呼び，以下ではこの形の超楕円曲線のみを考察することとする．また，記号 C を C' で再定義する．

$f_4 = 0$ を利用することで，2 倍算において [MCT01] に示されたアルゴリズムと比較し有限体上の乗算を 2 回削減可能である．

4.3 評価

標準形の超楕円曲線に対して Montgomery 逆元計算を用いて [MCT01] に示されたアルゴリズムの最適化を行った．加算アルゴリズムを表 4 に，2 倍算アルゴリズ

ムを表 5 に示す．また，weight が 2 で因子が互いに素な場合の計算コストを表 1 に示す．

表 1: 計算コストの対比

アルゴリズム	addition	doubling
Harley	$2I + 27M$	$2I + 30M$
[MCT01]	$2I + 23M$	$2I + 25M$
逆元計算 1 回	$I + 26M$	$I + 27M$

M : 基礎体上の乗算　　I : 基礎体上の逆元計算

更に，提案アルゴリズムの実装を行った．

実装の際，基礎体に [MCT01] に示された実装と同一の 186bitOEF [BP98] を用いた．また，OEF の演算，スカラー倍算には [MCT01] と同一のコードを用いた．

Pentium III 886MHz 上での実装結果を表 2 に示す．

表 2: 実装による性能比較

アルゴリズム	addition	doubling	scalar mul.
[MCT01]	8.32μs.	8.74μs.	1.98ms.
逆元計算 1 回	7.22μs.	7.50μs.	1.69ms.

本実装により提案アルゴリズムが [MCT01] と比較し 10% 以上高速であることが確認された．

本実装に用いた OEF 演算では $I \approx 6.4M$ であるが，素体等を用いた一般の実装では $I > 20M$ である場合が多く，そのような実装では提案アルゴリズムはより高い効果があると考えられる．

5 逆元計算を必要としないアルゴリズム

前章で, 逆元計算を 1 回のみ必要とするアルゴリズムを示した．本章では，逆元計算を必要としないアルゴリズムの実現について考察する．

5.1 変形 Mumford 表現

Mumford 表現では多項式 U をモニック多項式と定義している．例えば，種数 2 の場合，最も一般的な因子 $\mathcal{D} = (U, V)$ の表現は

$$\begin{aligned} U &= X^2 + u_1 X + u_0, \\ V &= v_1 X + v_0 \end{aligned}$$

となる．一方， Harley アルゴリズムの計算途中で得られる U はモニックとは限らず，最終的な出力結果をモニック化するために，有限体上の逆元計算が必ず必要である．もし Mumford 表現がモニックでない U を許せば，前章の結果と合わせ逆元計算を必要としない Harley アルゴリズムを構成できる可能性がある．

そこで本節ではモニック多項式でない U を許すように，Mumford 表現の一般化を与える．本論文ではこの一般化された Mumford 表現を「 変形 Mumford 表現」と呼ぶ．

定義 5.1. (変形 Mumford 表現)
Reduced divisor $\mathcal{D} = (U, V)$ が

$$\begin{aligned} U &= X^2 + u_1 X + u_0 \\ V &= v_1 X + v_0 \end{aligned}$$

として与えられたとき，その定数倍

$$\begin{aligned} U &= a(X^2 + u_1 X + u_0) \\ V &= a(v_1 X + v_0) \end{aligned}$$

を変形 Mumford 表現と呼ぶ．ただし，$a \in \mathbb{F}_q$ かつ $a \neq 0$ である． □

明らかに，変形 Mumford 表現から Mumford 表現に一意的な変形が可能である．

例えば，$2(2(\mathcal{D}) + \mathcal{D})$ の様に加算や 2 倍算を繰り返し使用する場合は，内部表現として変形 Mumford 表現を用いて計算すればよい．Mumford 表現が必要な場合は，出力結果 U の最高次係数で U, V を割れば求めることができる．

5.2 変形 Mumford 表現を用いた Harley アルゴリズム

4 章で提案したアルゴリズムの出力として変形 Mumford 表現を許せば，アルゴリズムに逆元計算は不要になる．しかしこの場合，同時にアルゴリズムの入力に変形 Mumford 表現を許す必要があり，アルゴリズムの全面的な修正が必要となる．

そこで本節では 4 章で提案したアルゴリズムを変形 Mumford 表現を利用できるように修正する方式を示す．

Weight 2 で互いに素な因子 $\mathcal{D}_1 = (U_1, V_1), \mathcal{D}_2 = (U_2, V_2)$ の加算，すなわち $\mathcal{D}_1 + \mathcal{D}_2$ では，最初に resultant を計算する．スペースの都合上，この計算を例にとり修正手法を説明する．

Algorithm 5.1. (変形 Mumford 表現に対する Harley アルゴリズム中の resultant の計算)
表 4 に示したアルゴリズムでは，

$$\begin{aligned} \text{Input} \quad & U_1 = X^2 + u_{11} X + u_{10} \\ & U_2 = X^2 + u_{21} X + u_{20} \\ \text{Output} \quad & r = u_{10}((u_{21}(u_{21} - u_{11}) + u_{10} - u_{20}) - u_{20}) \\ & \quad + u_{20}(u_{20} - u_{11}(u_{21} - u_{11})) \end{aligned}$$

である．この入力に変形 Mumford 表現を許すとその計算は

$$\begin{aligned} \text{Input} \quad & U_1 = u'_{12} X^2 + u'_{11} X + u'_{10} (= u'_{12} \cdot U_1) \\ & U_2 = u'_{22} X^2 + u'_{21} X + u'_{20} (= u'_{22} \cdot U_2) \\ \text{事前計算} \quad & c = u'_{12} u'_{22};\ U_{1i} = u'_{22} u'_{1i};\ U_{2i} = u'_{12} u'_{2i} \\ \text{Output} \quad & R = U_{10}(U_{21}(U_{21} - U_{11}) + c(U_{10} - U_{20}) \\ & \quad - cU_{20}) + U_{20}(cU_{20} - U_{11}(U_{21} - U_{11})) \end{aligned}$$

と実現される．なお，$R = c^3 r$ である．計算を最適化すると，乗算回数は前者が 4 回，後者は事前計算を除き 6 回となる(詳細は表 6 を参照)． □

上記計算で得られた R は，表 4 に示されたアルゴリズムで求まる r の $(u'_{12} u'_{22})^3$ 倍である．

以下，同様に表 4 に示されたアルゴリズムを修正することで，加算結果の変形 Mumford 表現を，(乗算回数は増えるが) 逆元計算を行うことなく得られる．ただし，実際には自然な修正では最終結果が変形 Mumford 表現にはならないため，最終結果の変形 Mumford 表現への変換が必要となる．

以上により得られた加算アルゴリズムを表 6 に，2 倍算アルゴリズムを表 7 に示す．

5.3 結果

前節で得られた改良 Harley アルゴリズムの $\mathcal{D}_3 = \mathcal{D}_1 + \mathcal{D}_2$ に対する計算コストを表 3 に示す．表 3 の第一行は $\mathcal{D}_1$ の semi-reduced divisor としての形式を P_∞ の項を除いて示したものであり，同様に第一列は $\mathcal{D}_2$ の semi-reduced divisor としての形式を P_∞ の項を除いて示したものである．

超楕円曲線の因子類群の加算を行う際，ほとんどの場合は入力因子 $\mathcal{D}_1, \mathcal{D}_2$ は weight が 2 で，$\gcd(U_1, U_2) = 1$ となる．また，2 倍算においてはほとんどの場合は入力因子 $\mathcal{D}_1$ は weight が 2 で，$\gcd(U_1, V_1) = 1$ となる．この場合，本章で示したアルゴリズムのコストは，加算が $54M$，2 倍算が $53M$ となった．

表 3: 除算を必要としないアルゴリズムの計算コスト

	P_1	$2P_1$	$P_1 + P_2$
P_1	$16M$	$42M$	$67M$
$-P_1$	$4M$	$15M$	$15M$
P_2	$12M$	$37M$	$67M$
$2P_1$	$42M$	$53M$	$108M$
$P_1 + P_2$	$67M$	$108M$	$53M$
$-P_1 + P_2$	$15M$	$45M$	$28M$
$P_1 + P_3$	$67M$	$108M$	$108M$
$-P_1 + P_3$	$15M$	$45M$	$45M$
$P_3 + P_4$	$37M$	$54M$	$54M$

M : 基礎体上の乗算

6 結論

本論文では，主に有限体上の逆元計算を削減することで，Harley アルゴリズムの高速化を行った．結果として，逆元計算 1 回，および逆元計算を必要としない加算と 2 倍算アルゴリズムを構成した．これらのアルゴリズムは多くの実装において従来の Harley アルゴリズムより高速な加算を可能とするものである．

逆元計算を必要としないアルゴリズムに関しては，様々な性能向上の可能性が残っており，広範囲からの考察，評価を継続したい．

謝辞

本研究に関連し有益な御助言を頂いた青木和麻呂氏に感謝致します．なお，本研究の一部は通信・放送機構「情報セキュリティ高度化のための第 3 世代暗号技術の研究開発」プロジェクトの一環として行なわれました．

参考文献

[BP98] D. V. Bailey and C. Paar, *Optimal extension fields for fast arithmetic in publickey algorithms*, Advances in Cryptology - CRYPTO'98 (H. Krawczyk, ed.), Lecture Notes in Computer Science, no. 1462, Springer-Verlag, 1998, pp.472–485.

[Can87] D.G. Cantor, *Computing in the Jacobian of hyperelliptic curve*, Math. Comp. **48** (1987), no.177, 95–101.

[GH00] P.Gaudry and R.Harley, *Counting points on hyperelliptic curves over finite fields*, ANTS-IV (W.Bosma, ed.), Lecture Notes in Computer Science, no. 1838, Springer-Verlag, 2000, pp.297–312.

[Har00a] R.Harley, *adding.text*, `http://cristal.inria.fr/~harley/hyper/`, 2000.

[Har00b] R.Harley, *doubling.c*, `http://cristal.inria.fr/~harley/hyper/`, 2000.

[Kob89] N.Koblitz, *Hyperelliptic curve cryptosystems*, J. Cryptology **1** (1989), no.3, 139–150.

[Mum84] D.Mumford, *Tata lectures on theta II*, Progress in Mathematics, no.43, Birkhäuser, 1984.

[Nag00] K.Nagao, *Improving group law algorithms for Jacobians of hyperelliptic curves*, ANTS-IV (W.Bosma, ed.), Lecture Notes in Computer Science, no. 1838, Springer-Verlag, 2000, pp.439–448.

[PS98] S.Paulus and A.Stein, *Comparing real and imaginary arithmetics for divisor class groups of hyperelliptic curves*, ANTS-III (J.P.Buhler, ed.), Lecture Notes in Computer Science, no. 1423, Springer-Verlag, 1998, pp.576–591.

[Sma99] N.Smart, *On the performance of hyperelliptic cryptosystems*, Advances in Cryptology - EUROCRYPT'99 (J. Stern, ed.), Lecture Notes in Computer Science, no. 1592, Springer-Verlag, 1999, pp.165–175.

[SS98] Y.Sakai and K.Sakurai, *Design of hyperelliptic cryptosystems in small characteristic and a software impremention over F_{2^n}*, Advances in Cryptology - ASIACRYPT'98 (K.Ohta and D.Pei, eds.), Lecture Notes in Computer Science, no. 1514, Springer-Verlag, 1998, pp.80–94.

[SSI98] Y.Sakai, K.Sakurai, and H.Ishizuka, *Secure hyperelliptic cryptosystems and their performance*, Public Key Cryptography (H.Imai and Y.Zheng, eds.), Lecture Notes in Computer Science, no. 1431, Springer-Verlag, 1998, pp.164–181.

[MCT01] K.Matsuo,J.Chao,S.Tsujii, *Fast Genus Two Hyperelliptic Curve Cryptosystems*, ISEC2001-23, Technical Report of IEICE, 2001.

[CF96] J. W. S. Cassels, E. V. Flynn, *Proglegomena to a middlebrow arithmetic of c urves of genus 2*, London Math. Soc. LNS230, Cambridge, (1996).

[Coh93] H.Cohen, *A Course in Computational Algebraic Number Theory*, GTM 138, Springer, 1993, pp.481

Appendix A. 逆元計算を 1 回のみ必要とする方式

表 4: 逆元計算を 1 回のみ必要とする加法アルゴリズム

Input	Weight two coprime reduced divisors $\mathcal{D}_1 = (U_1, V_1)$ and $\mathcal{D}_2 = (U_2, V_2)$	
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.	$4M$
	$z_1 \leftarrow u_{21} - u_{11}$; $z_2 \leftarrow u_{21}z_1 + u_{10} - u_{20}$;	
	$r \leftarrow u_{10}(z_2 - u_{20}) + u_{20}(u_{20} - u_{11}z_1)$;	
2	If $r = 0$ then $\mathcal{D}_1$ and $\mathcal{D}_2$ have a linear factor in common, and call the exclusive procedure.	—
3	Compute $I_1'(= rI_1) \equiv r/U_1 \bmod U_2$.	—
	$i'_{11} \leftarrow z_1$; $i'_{10} \leftarrow z_2$;	
4	Compute $S'(= rS) \equiv (V_2 - V_1)I_1' \bmod U_2$. (Karatsuba)	$5M$
	$w_1 \leftarrow v_{20} - v_{10}$; $w_2 \leftarrow v_{21} - v_{11}$; $w_3 \leftarrow i'_{10}w_1$; $w_4 \leftarrow i'_{11}w_2$;	
	$s'_1 \leftarrow (i'_{10} + i'_{11})(w_1 + w_2) - w_3 - w_4(1 + u_{21})$; $s'_0 \leftarrow w_3 - u_{20}w_4$;	
5	If $s'_1 = 0$ then $\mathcal{D}_3$ should be weight one, and call the exclusive procedure.	—
6	Compute $S = r^{-1}S'$ and s_1^{-1}. (Montgomery Inversion)	$I + 6M$
	$w_1 \leftarrow rs'_1$; $w_1 \leftarrow w_1^{-1}$; $w_2 \leftarrow w_1s'_1$; $w_3 \leftarrow r^2$; $z_3 \leftarrow w_1w_3$;	
	$s_0 \leftarrow w_2s'_0$; $s_1 \leftarrow w_2s'_1$;	
7	Compute $U_3 = s_1^{-2}((S^2U_1 + 2SV_1)/U_2 - (F - V_1^2)/(U_1U_2))$.	$5M$
	$u_{30} \leftarrow z_3(z_3(s_0^2 + u_{11} + u_{21}) + 2(v_{11} - s_0z_1)) + z_2$;	
	$u_{31} \leftarrow z_3(2s_0 - z_3) - z_1$; $u_{32} \leftarrow 1$;	
8	Compute $V_3 \equiv -(SU_1 + V_1) \bmod U_3$.	$6M$
	$w_1 \leftarrow u_{30} - u_{10}$; $w_2 \leftarrow u_{11} - u_{31}$;	
	$v_{30} \leftarrow s_1u_{30}w_2 + s_0w_1 - v_{10}$; $v_{31} \leftarrow s_1(u_{31}w_2 + w_1) - s_0w_2 - v_{11}$;	
Total		$I + 26M$

表 5: 逆元計算を 1 回のみ必要とする 2 倍算アルゴリズム

Input	A weight two reduced divisor $\mathcal{D}_1 = (U_1, V_1)$ without ramification points	
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 V_1.	$4M$
	$z_1 \leftarrow v_{11}^2$; $z_2 \leftarrow u_{11}v_{11}$; $r \leftarrow u_{10}z_1 + v_{10}(v_{10} - z_2)$;	
2	If $r = 0$ then $\mathcal{D}_1$ is with a ramification point, and call the exclusive procedure.	—
3	Compute $I_1'(= 2rI_1) \equiv r/V_1 \bmod U_1$.	—
	$i'_{11} \leftarrow -v_{11}$; $i'_{10} \leftarrow v_{10} - z_2$;	
4	Compute $T_1 \equiv (F - V_1^2)/U_1 \bmod U_1$	$2M$
	$w_1 \leftarrow u_{11}^2$; $w_2 \leftarrow w_1 + f_3$; $w_3 \leftarrow 2u_{10}$;	
	$t_{10} \leftarrow u_{11}(2w_3 - w_2) + f_2 - z_1$; $t_{11} \leftarrow 2w_1 + w_2 - w_3$;	
5	Compute $S'(= 2rS) \equiv I_1'T_1 \bmod U_1$. (Karatsuba)	$5M$
	$w_1 \leftarrow i'_{10}t_{10}$; $w_2 \leftarrow i'_{11}t_{11}$;	
	$s'_1 \leftarrow (i'_{10} + i'_{11})(t_{10} + t_{11}) - w_1 - w_2(1 + u_{11})$; $s'_0 \leftarrow w_1 - u_{10}w_2$;	
6	If $s'_1 = 0$ then $\mathcal{D}_3$ should be weight one, and call the exclusive procedure.	—
7	Compute $S = (2r)^{-1}S'$ and s_1^{-1}. (Montgomery Inversion)	$I + 6M$
	$w_1 \leftarrow 2r$; $w_2 \leftarrow w_1s'_1$; $w_2 \leftarrow w_2^{-1}$; $w_3 \leftarrow w_2s'_1$; $w_4 \leftarrow w_1^2$; $z_3 \leftarrow w_2w_4$;	
	$s_0 \leftarrow w_3s'_0$; $s_1 \leftarrow w_3s'_1$;	
8	Compute $U_2 = s_1^{-2}(((SU_1 + V_1)^2 - F)/U_1^2)$.	$4M$
	$u_{20} \leftarrow z_3(z_3(s_0^2 + 2u_{11}) + 2v_{11})$; $u_{21} \leftarrow z_3(2s_0 - z_3)$; $u_{22} \leftarrow 1$;	
9	Compute $V_2 \equiv -(SU_1 + V_1) \bmod U_3$.	$6M$
	$w_1 \leftarrow u_{11} - u_{21}$; $v_{20} \leftarrow u_{20}(s_1w_1 + s_0) - s_0u_{10} - v_{10}$;	
	$v_{21} \leftarrow s_1(u_{21}w_1 + u_{20} - u_{10}) - s_0w_1 - v_{11}$;	
Total		$I + 27M$

Appendix B. 逆元計算を必要としない方式

表 6: 逆元計算を必要としない加算アルゴリズム

Input	Weight two coprime reduced divisors $\mathcal{D}_1 = (U_1, V_1)$ and $\mathcal{D}_2 = (U_2, V_2)$ represented by Modified Mumford.	
Output	A weight two reduced divisor $\mathcal{D}_3 = (U_3, V_3) = \mathcal{D}_1 + \mathcal{D}_2$ represented by Modified Mumford.	
Step	Procedure	Cost
1	Compute $u_{12}u_{22}$ and $u_{22}u_{1i}, u_{22}v_{1i}, u_{12}u_{2i}, u_{12}v_{2i}$ for $i \in \{1, 0\}$. $c \leftarrow u_{12}u_{22}$; $U_{11} \leftarrow u_{22}u_{11}$; $U_{10} \leftarrow u_{22}u_{10}$; $V_{11} \leftarrow u_{22}v_{11}$; $V_{10} \leftarrow u_{22}v_{10}$; $U_{21} \leftarrow u_{12}u_{21}$; $U_{20} \leftarrow u_{12}u_{20}$; $V_{21} \leftarrow u_{12}v_{21}$; $V_{20} \leftarrow u_{12}v_{20}$;	$9M$
2	Compute the resultant r of U_1 and U_2. (Difference is c^3) $z_1 \leftarrow U_{21} - U_{11}$; $z_2 \leftarrow U_{21}z_1 + c(U_{10} - U_{20})$; $w_1 \leftarrow cU_{20}$; $r \leftarrow U_{10}(z_2 - w_1) + U_{20}(w_1 - U_{11}z_1)$;	$6M$
3	If $r = 0$ then $\mathcal{D}_1$ and $\mathcal{D}_2$ have a linear factor in common, and call the exclusive procedure.	—
4	Compute $I_1 \equiv 1/U_1 \bmod U_2$. (Difference is $(c^{-2}r, c^{-1}r)$) $i_{11} \leftarrow z_1$; $i_{10} \leftarrow z_2$;	—
5	Compute $S \equiv (V_1 - V_2)I_1 \bmod U_2$ (Karatsuba). (Difference is (r, r)) $w_1 \leftarrow V_{20} - V_{10}$; $w_2 \leftarrow V_{21} - V_{11}$; $w_3 \leftarrow i_{10}w_1$; $w_4 \leftarrow i_{11}w_2$; $s_1 \leftarrow (ci_{11} + i_{10})(w_1 + w_2) - w_3 - (c + U_{21})w_4$; $s_0 = -U_{20}w_4 + w_3$;	$6M$
6	If $s_1 = 0$ then $\mathcal{D}_3$ should be weight one, and call the exclusive procedure.	—
7	Compute $U_3 = s_1^{-2}((S^2U_1 + 2SV_1)/U_2 - (F - V_1^2)/(U_1U_2))$. (Difference is $(c^2s_1^2, c^2s_1^2, c^2s_1^2)$) $w_1 \leftarrow s_1^2$; $z_3 \leftarrow cw_1$; $w_2 \leftarrow r^2$; $u_{30} \leftarrow c(cs_0^2 + w_2(U_{11} + U_{21}) + 2s_1(rV_{11} - s_0z_1)) + w_1z_2$; $u_{31} \leftarrow c(c(2s_1s_0 - w_2) - w_1z_1)$; $u_{32} \leftarrow cz_3$;	$16M$
8	Compute $V_3 = -(SU_1 + V_1) \bmod U_3$. (Difference is $(c^4rs_1^4, c^4rs_1^4)$) $z_4 \leftarrow rz_3$; $w_1 \leftarrow u_{30} - z_3U_{10}$; $w_2 \leftarrow z_3U_{11} - u_{31}$; $w_3 \leftarrow s_1w_2$; $v_{30} \leftarrow w_3u_{30} + u_{32}(s_0w_1 - z_4V_{10})$; $v_{31} \leftarrow w_3u_{31} + u_{32}(s_1w_1 - s_0w_2 - z_4V_{11})$;	$13M$
9	Adjust coefficients. $w_1 \leftarrow cz_4$; $u_{30} \leftarrow u_{30}w_1$; $u_{31} \leftarrow u_{31}w_1$; $u_{32} \leftarrow u_{32}w_1$;	$4M$
Total		$54M$

表 7: 逆元計算を必要としない 2 倍算アルゴリズム

Input	A weight two reduced divisor $\mathcal{D}_1 = (U_1, V_1)$ without ramification points represented by Modified Mumford.	
Output	A weight two reduced divisor $\mathcal{D}_2 = (U_2, V_2) = 2\mathcal{D}_1$ represented by Modified Mumford.	
Step	Procedure	Cost
1	Compute the resultant r of U_1 and V_1. (Difference is u_{12}^3) $z_1 \leftarrow v_{11}^2$; $z_2 \leftarrow u_{12}v_{10} - u_{11}v_{11}$; $r \leftarrow u_{10}z_1 + v_{10}z_2$;	$5M$
2	If $r = 0$ then $\mathcal{D}_1$ is with a ramification point, and call the exclusive procedure.	—
3	Compute $I_1 \equiv 1/(2V_1) \bmod U_1$. (Difference is $(2u_{12}^{-2}r, 2u_{12}^{-1}r)$) $i_{11} \leftarrow -v_{11}$; $i_{10} \leftarrow z_2$;	—
4	Compute $T_1 \equiv (F - V_1^2)/U_1 \bmod U_1$. (Difference is (u_{12}^2, u_{12}^3)) $z_3 \leftarrow u_{12}^2$; $w_1 \leftarrow u_{12}f_3$; $w_2 \leftarrow u_{11}^2$; $t_{11} \leftarrow u_{12}(w_1 - 2u_{10}) + 3w_2$; $t_{10} \leftarrow u_{12}(u_{11}(4u_{10} - w_1) + z_3f_2 - z_1) - u_{11}w_2$;	$8M$
5	Compute $S \equiv I_1T_1 \bmod U_1$. (Difference is $(2u_{12}^2r, 2u_{12}^2r)$) $w_1 \leftarrow i_{11}t_{11}$; $z_4 \leftarrow i_{10}t_{11} + i_{11}t_{10} - u_{11}w_1$; $s_1 \leftarrow u_{12}z_4$; $s_0 \leftarrow i_{10}t_{10} - u_{12}u_{10}w_1$;	$8M$
6	If $s_1 = 0$ then $\mathcal{D}_3$ should be weight one. and call the exclusive procedure.	—
7	Compute $U_2 = s_1^{-2}((SU_1 + V_1)^2 - F)/U_1^2$. (Difference is (s_1^2, s_1^2, s_1^2)) $z_5 \leftarrow ru_{12}$; $w_1 \leftarrow u_{12}z_5$; $u_{20} \leftarrow s_0^2 + 4z_5(2w_1u_{11} + v_{11}s_1)$; $z_6 \leftarrow 2(z_4s_0 - 2w_1z_5)$; $u_{21} \leftarrow u_{12}z_6$; $z_7 \leftarrow z_4^2$; $u_{22} \leftarrow z_3z_7$;	$11M$
8	Compute $V_2 = -(SU_1 + V_1) \bmod U_3$. (Difference is $(2ru_{12}s_1^4, 2ru_{12}s_1^4)$) $w_1 \leftarrow z_7u_{11} - z_6$; $w_2 \leftarrow 2ru_{22}^2$; $w_3 \leftarrow u_{22}u_{10}z_7$; $v_{20} \leftarrow u_{20}s_1(w_1 + z_4s_0) - s_0w_3 - w_2v_{10}$; $v_{21} \leftarrow s_1(u_{21}w_1 + u_{12}u_{20}z_7 - w_3) - s_0w_1u_{22} - w_2v_{11}$;	$17M$
9	Adjust coefficients $w_1 \leftarrow 2z_5u_{22}$; $u_{20} \leftarrow u_{20}w_1$; $u_{21} \leftarrow u_{21}w_1$; $u_{22} \leftarrow u_{22}w_1$;	$4M$
Total		$53M$