Advances in Mathematics of Communications
Volume , No. 0, 2010, 1–46
Web site: http://www.aimSciences.org

EXPLICIT FORMULAS FOR REAL HYPERELLIPTIC CURVES OF GENUS 2 IN AFFINE REPRESENTATION

S. Erickson

Department of Mathematics and Computer Science
Colorado College
14 E. Cache La Poudre
Colorado Spgs., CO 80903, USA

M. J. Jacobson, Jr.

Department of Computer Science
University of Calgary
2500 University Drive NW
Calgary, Alberta, Canada T2N 1N4

A. Stein

Institut für Mathematik
Carl-von-Ossietzky Universität Oldenburg
D-26111 Oldenburg, Germany

(Communicated by ???)

Abstract. We present a complete set of efficient explicit formulas for arithmetic in the degree 0 divisor class group of a genus two real hyperelliptic curve givenin affine coordinates. In addition to formulas suitable for curvesdefined over an arbitrary finite field, we give simplified versions for both the odd and the even characteristic cases. Formulas for baby steps, inverse baby steps, divisor addition, doubling, and special cases such as adding a degenerate divisor are provided, with variations for divisors given in reduced and adapted basis. We describe the improvements and the correctness together with a comprehensive analysis of the number of field operations for each operation. Finally, we perform a direct comparison of cryptographic protocols using explicit formulas for real hyperelliptic curves with the corresponding protocols presented in the imaginary model.

1. Introduction and Motivation for explicit formulas in the real case

In 1989, Koblitz [Kob88] first proposed the Jacobian of an imaginary hyperelliptic curve for use in public-key cryptographic protocols. Hyperelliptic curves are in a sense generalizations of elliptic curves and can be used with the same key-per-bit strength as long as the genus is very small. More precisely, because of recent attacks [Gau00, DGTT07], only genus 2 and possibly genus 3 hyperelliptic curves might offer the same advantage as elliptic curves.

The Jacobian of a hyperelliptic curve defined over a finite field is a finite abelian group which, like elliptic curve groups, has unique representatives of group elements

2000 *Mathematics Subject Classification:* Primary: 94A60, 14H45; Secondary: 14Q05.
Key words and phrases: hyperelliptic curve, reduced divisor, infrastructure and distance, Cantor's algorithm, explicit formulas, efficient implementation, cryptographic key exchange.
The second author is supported in part by NSERC of Canada.

and efficient arithmetic (divisor addition and reduction). Although the arithmetic appears more complicated than that of elliptic curves [Lan05, PWP03, WPP05, Ava04, JMS04], there are some indications that it can be more efficient in some cases. Those results are based on optimized explicit formulas and very efficient implementations for genus 2 and 3 imaginary hyperelliptic curves.

Several years later, a key exchange protocol was presented for the real model of a hyperelliptic curve [SSW96]. Its underlying key space was the set of reduced principal ideals in the ring of regular functions of the curve, together with its group-like infrastructure. Although the main operation of divisor addition and reduction is comparable in efficiency to that of the imaginary model [Ste01], the protocol in [SSW96] was significantly slower and more complicated than its imaginary cousin [Kob88], while offering no additional security; the same was true for subsequent modifications presented in [Sch01].

Despite the apparent short-comings of the real model, recent work [JSS06] shows that it may admit protocols that are comparable in efficiency to those based on the imaginary model. The main idea is that, in addition to the divisor addition operation, the real model has a second operation called a *baby step* that is significantly more efficient. By exploiting this operation and relying on some reasonable heuristics, new public-key protocols for key exchange, digital signatures, and encryption have been devised that are significantly faster than all previous protocols in real hyperelliptic curves and might even be comparable in efficiency with analogous protocols in the imaginary setting. However, the numerical results in [JSS06] were based on a generic implementation and did not incorporate explicit formulas. In order to examine the efficiency of these new protocols completely, it is necessary to devise explicit formulas for divisor arithmetic in the real model of cryptographically-relevant low genus curves.

From [SSW96, PR99, JSS06], we know that the underlying computationally hard problem of these protocols is the infrastructure discrete logarithm problem, which has the same complexity as the discrete logarithm problem in the Jacobian of a hyperelliptic curve. In the case of a real hyperelliptic curve, the infrastructure operation plus some information on the distance can be interpreted as a group operation in the Jacobian of the curve. A more general interpretation of the infrastructure including a precise definition of baby steps, giant steps, and distances has been provided in [Fon08a, Fon08b]. As a special case, the author gives a complete description of the one-dimensional infrastructure, as is the case for real hyperelliptic curves, and thus provides an interpretation of the protocols for real hyperelliptic curves [SSW96, JSS06] in a group setting. A similar result is described in [Mir08], in which the infrastructure is embedded explicitly in the subgroup of the Jacobian generated by the divisor $\infty_1 - \infty_2$, allowing infrastructure-based algorithms to be described in the Jacobian in a more efficient way than those found in [PR99].

The case of the general one-dimensional infrastructure gives rise to various realizations of cryptographic protocols, which are in concept similar to the ones for imaginary hyperelliptic curves. In an explicit situation, the correct choice of the base divisor and the interpretation of the points at infinity appears to be very flexible. This line of research appears to still be very promising and should allow several improvements. For low genus hyperelliptic curves, especially for genus 2, it is therefore important to develop the fastest explicit formulas that can be used in any realization.

The contribution of this paper is to present a complete set of efficient explicit formulas for divisor arithmetic on real hyperelliptic curves. We concentrate on genus 2 real hyperelliptic curves in affine coordinates. We thus provide explicit formulas for the protocols in [JSS06], thereby enabling a direct comparison with the corresponding protocols presented in the imaginary model. The formulas for the case where the underlying finite field has characteristic greater than 3 first appeared in [EJS+07]. This paper completes the picture for genus two by including formulas for arbitrary characteristic, divisor arithmetic in both reduced and adapted basis, and formulas for various special cases that can arise. As indicated in [EJS+07], our formulas also lead to a savings of one field squaring in the general addition formula in the imaginary case.

Although there exist easy transformations from the imaginary model to the real model of a hyperelliptic curve, the converse is only possible if the curve contains an $\mathbb{F}_q$-rational point defined over $\mathbb{F}_q$. If q is odd and one uses an irreducible polynomial for the generation of the real hyperelliptic curve, then in the worst case one has to extend the field of constants to $\mathbb{F}_{q^{2g+2}}$ in order to be able to perform this transformation, which is unrealistic for efficient implementations. Furthermore, complex multiplication methods for generating hyperelliptic curves of small genus often produce real hyperelliptic curves. With an efficient arithmetic, those curves can be readily used in cryptographic protocols. Another important motivation is that explicit formulas will enable a real-world comparison of index-calculus attacks on hyperelliptic curve cryptosystems in both the real and imaginary setting. Finally, real hyperelliptic curves have become very popular in recent developments in elliptic and hyperelliptic curve cryptography. Real models of elliptic and hyperelliptic curves appear to be at least comparable to imaginary models in various applications. A novel approach was given in [GLM08], where the authors showed how real hyperelliptic curves can be used in pairing-based cryptography. The authors make use of an interesting approach using so-called balanced divisors [GHM08] in order to get rid off unnecessary baby steps in certain protocols. This representation of divisors, when used for key exchange, likely yields comparable performance to the protocols in [JSS07]. Both the balanced divisor approach and infrastructure computations require the same underlying arithmetic operations, so the explicit formulas presented in this paper will be useful in either model.

Analysis of our formulas shows that they require a few more finite field multiplications than their imaginary counterparts. However, the baby step operation in its explicit form is significantly more efficient than divisor addition in either setting, and as a result, the cryptographic protocols in the real setting perform almost as well as those in the imaginary case. [1] In addition, even though the formulas are not as fast as those in the imaginary case, they are certainly more efficient than using generic algorithms in the real setting. Thus, using our formulas will significantly speed other computations in the divisor class group or infrastructure of a real hyperelliptic curve, such as computing the regulator or class number.

The paper is organized as follows. We first provide the necessary background on real hyperelliptic curves and introduce the notation. We also present the essential, generic algorithms for real hyperelliptic curves and explain how to perform arithmetic in the degree 0 divisor class group via ideal arithmetic. In Section 3, we present the explicit formulas for the basic algorithms in even and odd characteristic,

[1] Though, it should be mentioned that in the imaginary case, one could also enforce a baby step-like operation by using degenerate divisors.

as well as descriptions of their derivation, correctness, and a comprehensive analysis of the number of field operations required. Section 9 contains numerical data comparing cryptographic protocols based on real hyperelliptic curves with those using imaginary hyperelliptic curves, where divisor arithmetic is implemented using explicit formulas in both cases.

2. Background and Notation

Throughout this paper, let $\mathbb{F}_q$ be a finite field with $q = p^l$ elements, where p is a prime, and let $\overline{\mathbb{F}}_q = \bigcup_{n\geq 1} \mathbb{F}_{q^n}$ be its algebraic closure. For details on the arithmetic of hyperelliptic curves we refer to [MWZ96, JMS04, CF05, JSS06], and specifically for real hyperelliptic curves we refer to [PR99, Ste01, Eng01, JSS06, JSS07].

Definition 2.1. *A hyperelliptic curve C of genus g defined over $\mathbb{F}_q$ is an absolutely irreducible non-singular curve defined by an equation of the form*

$$C : y^2 + h(x)y = f(x), \tag{2.1}$$

where $f, h \in \mathbb{F}_q[x]$ are such that $y^2 + h(x)y - f(x)$ is absolutely irreducible, i.e. irreducible over $\overline{\mathbb{F}}_q$, and if $b^2 + h(a)b = f(a)$ for $(a, b) \in \overline{\mathbb{F}}_q \times \overline{\mathbb{F}}_q$, then $2b + h(a) \neq 0$ or $h'(a)b - f'(a) \neq 0$. A hyperelliptic curve C is called

1. *an* imaginary hyperelliptic curve *if the following hold: If q is odd, then f is monic,* $\deg f = 2g + 1$, *and $h = 0$. If q is even, then h and f are monic,* $\deg f = 2g + 1$, *and* $\deg h \leq g$.
2. *a* real hyperelliptic curve *if the following hold: If q is odd, then f is monic,* $\deg f = 2g + 2$, *and $h = 0$. If q is even, then h is monic,* $\deg h = g + 1$, *and either (a)* $\deg f \leq 2g + 1$ *or (b)* $\deg f = 2g + 2$ *and the leading coefficient of f is of the form $\beta^2 + \beta$ for some $\beta \in \mathbb{F}_q^*$.*

The function field $K = \mathbb{F}_q(C)$ of a hyperelliptic curve C is a quadratic, separable extension of $\mathbb{F}_q(x)$, and the integral closure of $\mathbb{F}_q(x)$ in K is given by $\mathbb{F}_q[C] = \mathbb{F}_q[x, y]/(y^2 + h(x)y - f(x))$. Let S_∞ denote the set of points at infinity. Then the set $C(\overline{\mathbb{F}}_q) = \{(a, b) \in \overline{\mathbb{F}}_q \times \overline{\mathbb{F}}_q : b^2 + h(a)b = f(a)\} \cup S_\infty$ is called the set of ($\overline{\mathbb{F}}_q$-rational) points on C. For a point $P = (a, b) \in C(\overline{\mathbb{F}}_q)$, the hyperelliptic involution is given by $\iota(a, b) = (a, -b - h(a)) \in C(\overline{\mathbb{F}}_q)$.

Notice that in both cases we can assume $h = 0$ if q is odd. The imaginary model[2] corresponds to the case where $S_\infty = \{\infty_1\}$. In the real model[3], there exist two points at infinity so that $S_\infty = \{\infty_1, \infty_2\}$. Let v_1 and v_2 be the corresponding normalized valuations of K. From now on, we only consider the real case[4].

Let C be a real hyperelliptic curve given as in Definition 2.1. A *divisor* of C is a finite formal sum $D = \sum_{P \in C} m_P P$ of points $P \in C(\overline{\mathbb{F}}_q)$, where $m_P \in \mathbb{Z}$ and $m_P = 0$ for all but finitely many P. The *degree* of D is defined by $\deg D = \sum_P m_P$. A divisor D of C is *effective* if $m_P \geq 0$ for all P, and a divisor D is *defined over* $\mathbb{F}_q$ if $D^\sigma = \sum_P m_P P^\sigma = D$ for all automorphisms σ of $\overline{\mathbb{F}}_q$ over $\mathbb{F}_q$. The set $Div(K)$ of divisors of C defined over $\mathbb{F}_q$ forms an additive abelian group under formal addition with the set $Div_0(K)$ of all degree zero divisors of C defined

[2]In function field terms, the pole divisor ∞ of x in $\mathbb{F}_q(x)$ is totally ramified in K so that $\mathrm{Con}(\infty) = 2\infty_1$.

[3]In function field terms, the pole divisor ∞ of x in $\mathbb{F}_q(x)$ splits completely in K so that $\mathrm{Con}(\infty) = \infty_1 + \infty_2$.

[4]There is one additional model which corresponds to the case where the infinite prime of $\mathbb{F}_q(x)$ is inert in K. Those hyperelliptic curves are called *unusual*.

over $\mathbb{F}_q$ being a subgroup. For $G \in \mathbb{F}_q[C]$, we can associate a principal divisor $\mathrm{div}(G) = \sum_P v_P(G)P$, where $v_P(G)$ is the valuation of G at P. The group of principal divisors $P(K) = \{\mathrm{div}(G) : G \in K\}$ of C forms a subgroup of $Div_0(K)$. The factor group $J(K) = Div_0(K)/P(K)$ is called the *divisor class group* of K. We denote by $\overline{D} \in J(K)$ the class of $D \in Div_0(K)$.

As C is a real hyperelliptic curve, we have $S_\infty = \{\infty_1, \infty_2\}$. From [PR99], every degree 0 divisor class can be represented by $\overline{D}$ such that $D = \sum_{i=1}^{r} P_i - r\infty_2$, where $P_i \in C(\overline{\mathbb{F}}_q)$, $P_i \neq \infty_2$, and $P_i \neq \iota P_j$ if $i \neq j$. The representative D of $\overline{D}$ is then called semi-reduced. In addition, there exists a representative D such that $r \leq g$; such a representative D is called reduced. Notice that $P_i = \infty_1$ is allowed for some i. It follows that every degree 0 divisor class contains a unique representative

$$D = \sum_{i=1}^{l(D)} Q_i - l(d)\infty_2 + v_1(D)(\infty_1 - \infty_2) \ ,$$

where $Q_i \in C(\overline{\mathbb{F}}_q)$, $Q_i \neq \infty_1, \infty_2$, $Q_i \neq \iota Q_j$ if $i \neq j$, and $0 \leq l(D) + v_1(D) \leq g$. The regulator R of $\mathbb{F}_q(C)$ in $\mathbb{F}_q[C]$ is defined to be the order of the degree 0 divisor class containing $\infty_1 - \infty_2$.

In this paper, we follow the basic setting of hyperelliptic curves described above. One interesting alternative is given in [GHM08, GLM08], where the authors use balanced divisors in order to produce a more efficient representation. Their results and possible alternatives, such as those that can be derived from [Fon08a, Fon08b] or [Mir08], warrant further investigation.

It is well-known that we can identify a (real) hyperelliptic function field as a subfield $\mathbb{F}_q(C) \subseteq \mathbb{F}_q\langle t^{-1}\rangle$ of the field of Puiseux series in t^{-1}. Any non-zero $\alpha \in \mathbb{F}_q\langle t^{-1}\rangle$ is a power series $\alpha = \sum_{i=-\infty}^{m} a_i t^i$, where $m \in \mathbb{Z}$, $a_i \in \mathbb{F}_q$ for $-\infty \leq i \leq m$, and $a_m \neq 0$. Then $\lfloor \alpha \rfloor = \sum_{i=0}^{m} a_i t^i$, $\mathrm{sgn}(\alpha) = a_m$, and $\deg(\alpha) = m$. For $\alpha = 0$, we put $\lfloor 0 \rfloor = 0$ and $\deg(0) = -\infty$.

We know that $\mathbb{F}_q[C]$ is a Dedekind domain and the ideal class group $\mathrm{Cl}(K)$ of $K = \mathbb{F}_q(C)$ is the factor group of fractional $\mathbb{F}_q[C]$-ideals modulo principal fractional ideals. A non-zero integral ideal $\mathfrak{a}$ in $\mathbb{F}_q[C]$ is a fractional ideal such that $\mathfrak{a} \subseteq \mathbb{F}_q[C]$. It can be represented as $\mathfrak{a} = k[x]\, d(x)u(x) + k[x]\, d(x)(v(x) + y)$, where $u, v \in k[x]$ and $u \mid f + hv - v^2$. Note that d and u are unique up to factors in $\mathbb{F}_q^*$ and v is unique modulo u. The ideal $\mathfrak{a}$ is said to be *primitive* if we can take $d(x) = 1$, in which case we simply write $\mathfrak{a} = [u(x), v(x)+y]$. A primitive ideal $\mathfrak{a} = [u(x), v(x)+y]$ is *reduced* if $\deg u \leq g$. A basis $\{u(x), v(x)+y\}$ of a primitive ideal is called *adapted* or *standard* if $\deg(v) < \deg(u)$ and u is monic. For instance, $\mathbb{F}_q[C]$ is represented as $\mathbb{F}_q[C] = [1, y]$. The *degree* of a primitive ideal is $\deg(\mathfrak{a}) = \deg u$. We call a basis $\{u(x), v(x) + y\}$ of a primitive ideal *reduced* if $-v_1(v - h - y) < -v_1(u) = \deg(u) < -v_1(v + y)$ and u is monic. In practice, it is common to have reduced ideals given in adapted form for imaginary and unusual curves and in reduced (or possibly adapted) form for real curves.

For any two ideals $\mathfrak{a}$ and $\mathfrak{b}$ in the same ideal class, there exists $\alpha \in \mathbb{F}_q(C)^*$ with $\mathfrak{b} = (\alpha)\mathfrak{a}$. We then define the *distance of* $\mathfrak{b}$ *with respect to* $\mathfrak{a}$ as $\delta(\mathfrak{b}, \mathfrak{a}) = -v_1(\alpha) \pmod R$ where R is the regulator. Note that the distance is only well-defined and unique modulo R. In each ideal class, we expect up to R many reduced ideals. If we restrict to the principal ideal class, then we may assume that $\mathfrak{a} = \mathfrak{a}_1 = \mathbb{F}_q[C] = (1)$. Then, for any principal ideal $\mathfrak{b} = (\alpha)$, we let $\delta(\mathfrak{b}) = \delta(\mathfrak{b}, \mathfrak{a}_1) = -v_1(\alpha) \pmod R$. The distance defines an order on all reduced principal ideals,

i.e., the set of reduced principal ideals is $\mathcal{R} = \{\mathfrak{a}_1, \mathfrak{a}_2, \ldots, \mathfrak{a}_m\}$ where $\delta(\mathfrak{a}_1) = 0 < \delta(\mathfrak{a}_2) < \ldots < \delta(\mathfrak{a}_m) < R$ and $\delta(\mathfrak{a}_i)$ is the unique nonnegative integer between 0 and $R-1$ modulo R.

The following theorem gives a representation of degree 0 divisor classes in terms of reduced ideals and corresponds to the Mumford representation [Mum84, page 317] in the imaginary model.

Theorem 2.1. *(Paulus-Rück, 1999) There is a canonical bijection between the divisor class group $J(K)$ and the set of pairs $\{(\mathfrak{a}, n)\}$, where $\mathfrak{a}$ is a reduced ideal of $\mathbb{F}_q[C]$ and n is an integer with $0 \leq \deg(\mathfrak{a}) + n \leq g$.*

The bijection is such that the unique reduced divisor D in a degree 0 divisor class $\overline{D}$ corresponds to such a pair $\{(\mathfrak{a}, n)\}$. It follows that arithmetic in $J(K)$ can be performed via arithmetic of reduced ideals. An algorithm for computing the group law in $J(K)$ based on this theorem is presented in [SSW96, PR99, Ste01]. It consists of three steps, namely (a) multiplication of reduced ideals, (b) reduction of the primitive part of the product, and (c) baby steps, i.e. adjusting the output of the reduction so that the degree condition of the theorem is satisfied. Step (a) and (b) together are called a giant step. A giant step is the analogue of the group operation in the imaginary case. Elements in $J(K)$ can be represented as triples $[u, v, n]$ where $[u(x), v(x) + y]$ is a reduced ideal and $0 \leq \deg(\mathfrak{a}) + n \leq g$. It can be easily seen that the arithmetic can be restricted to the special subset $\mathcal{R} = \{(\mathfrak{a}, 0) : \mathfrak{a} \text{ reduced and principal}\}$ called the *infrastructure*, which is not a group. Those elements can be represented as $[u, v, 0]$ or simply as pairs $[u, v]$. We therefore assume that we only perform operations on elements of $J(K)$ which are given by a pair $\overline{D} = [u, v]$, where $u, v \in \mathbb{F}_q[x]$ such that

1. u is monic,
2. $\deg(u) \leq g$,
3. $u \mid f + hv - v^2$,
4. one of the following degree conditions is satisfied, namely
   (a) for the reduced basis: $-v_1(v - h - y) < -v_1(u) = \deg(u) < -v_1(v + y)$, or
   (b) for the adapted (standard) basis: $\deg(v) < \deg(u)$.

If only 1, 3, and 4 are satisfied, the ideal $[u(x), v(x) + y]$ is only primitive and the corresponding representative $D \in \overline{D}$ is semi-reduced. We also denote this element by $[u, v]$.

In [JSS06], several optimized key-exchange protocols were presented that use arithmetic in $\mathcal{R}$. In fact, under reasonable assumptions, one can avoid the additional adjusting steps and replace some giant steps by baby steps. Furthermore, in each giant step, it is easy to keep track of the distances of the corresponding reduced ideals. In fact, assuming certain heuristics, one can even avoid computing distances when $|\mathbb{F}_q|$ is sufficiently large. We will therefore ignore the computation of distances. Even in those cases, where distances are needed, the running time for the computation of the distance is negligible. The protocols for real hyperelliptic curves are analogous to the ones in the imaginary setting, but they also make use of the additional baby step operation in order to improve their efficiency significantly.

We now give all three relevant generic algorithms. For details on how to produce key exchange protocols with these algorithms, we refer to [SSW96, JSS06]. We will use additive notation in order to express the group operation in $J(K)$ even

though ideal arithmetic is usually denoted multiplicatively. Note that, by using these algorithms, arithmetic in $J(K)$ is reduced to polynomial arithmetic in $\mathbb{F}_q[x]$.

Algorithm 2.1 (Multiplication)**.**

Input: $\overline{D}_1 = [u_1, v_1]$, $\overline{D}_2 = [u_2, v_2]$, and $h(x), f(x)$ as in (2.1).
Output: $\overline{D} = [u, v]$ such that D is semi-reduced and $\overline{D} = \overline{D}_1 + \overline{D}_2$.

1. Compute $d, x_1, x_2, x_3 \in \mathbb{F}_q[x]$ such that
$$d = \gcd(u_1, u_2, v_1 + v_2 + h) = x_1 u_1 + x_2 u_2 + x_3(v_1 + v_2 + h) \ .$$
2. Put $u = u_1 u_2 / d^2$ and $v = (x_1 u_1 v_2 + x_2 u_2 v_1 + x_3(v_1 v_2 + f))/d \pmod{u}$.

For the group operation, we assume that the representatives of the divisor classes D_1 and D_2 are reduced so that the ideals $[u_1(x), v_1(x) + y]$ and $[u_2(x), v_2(x) + y]$ are reduced, i.e. $\deg(u_1), \deg(u_2) \leq g$. However, the algorithm also allows semi-reduced representatives D_1 and D_2 as an input. Notice that the output of this algorithm $\overline{D} = [u, v]$ corresponds to a semi-reduced divisor so that $(u, v + y)$ is a primitive ideal which is not necessarily reduced.

For the second step, we need to precompute the principal part $H(y) = \lfloor y \rfloor$ of a root y of $y^2 + h(x)y - f(x) = 0$. The other root is $-y - h$. If $y = \sum_{i=-\infty}^{m} y_i x^i \in \mathbb{F}_q\langle x^{-1} \rangle$, then $H(y) = \sum_{i=0}^{m} y_i x^i$.

Algorithm 2.2 (Reduction)**.**

Input: $\overline{D} = [u, v]$, where D is semi-reduced, and $h(x), f(x)$ as in (2.1).
Output: $\overline{D}' = [u', v']$ such that D' is reduced and $\overline{D}' = \overline{D}$.

1. Compute $a = (v + H(y)) \text{ div } u$.
2. Let $v' = au - v + h$, $u' = (f + hv' - v'^2)/u$.
3. If $\deg(u') > g$, put $u = u'$, $v = v'$, and goto 1.
4. Make u' monic. If an adapted basis is required, reduce $v' \bmod u'$.

If we allow the input of Algorithm 2.2 to be reduced and only perform Steps 1, 2, 4, then the output will be another reduced divisor. In this case, we call this operation a *baby step*, and denote it by $\rho([u, v])$.

3. Explicit formulas

In the following sections, we present explicit formulas for giant steps and baby steps on genus 2 real hyperelliptic curves given in affine representation. Let $[u, v]$ be a reduced representative of a divisor class. We present formulas assuming both adapted and reduced bases for the divisors. Recall that for the adapted basis, v is the unique polynomial which has degree strictly less than $\deg(u)$, so for genus 2 and $\deg(u) = 2$, we have $v = v_1 x + v_0$. For the reduced basis, v is the unique degree 3 polynomial of the form $(y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_1 x + v_0$. Note that the degree two and three terms depend only on the equation of the curve, so, as with the adapted basis, only two coefficients are required to represent v in an implementation.

For the operation counts, some generic assumptions on the coefficients can be made after applying certain isomorphic transformations. If the underlying finite field has odd characteristic, then the equation of a genus 2 hyperelliptic curve in the real model can be written as $y^2 = f(x)$, where $f(x) = x^6 + f_5 x^5 + f_4 x^4 + f_3 x^3 + f_2 x^2 + f_1 x + f_0$. If the characteristic is not 3, then the linear transformation $x \mapsto x - f_5/6$ eliminates the x^5 term in $f(x)$. In even characteristic, the curve

can be written as $y^2 + h(x)y = f(x)$, where $h(x) = x^3 + h_2x^2 + h_1x + h_0$ and $f(x) = f_6x^6 + f_5x^5 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0$. The isomorphic transformation

$$x \mapsto x + h_2, y \mapsto y + f_5x^2 + (f_4 + h_2f_5 + f_5^2 + h_2^2f_6)x + (f_3 + f_5(h_1 + h_2^2))$$

makes the x^2 term in $h(x)$ and the x^5, x^4, x^3 terms in $f(x)$ vanish. Thus we can assume the hyperelliptic curve is of the form

$$y^2 + (x^3 + h_1x + h_0)y = f_6x^6 + f_2x^2 + f_1x + f_0 \ .$$

Given $f(x)$ and $h(x)$, it is a straightforward matter to compute the coefficients of $H(y)$ by computing $y^2 + h(x)y$ symbolically and equating coefficients with f. If $f(x)$ and $h(x)$ have no restrictions on their coefficients, then y_3 is set to be a solution of the quadratic equation $y_3^2 + h_3y_3 = f_6$, while the remaining coefficients are computed as follows:

$$\begin{aligned} y_2 &= (f_5 - y_3h_2)/(2y_3 + h_3) \\ y_1 &= (f_4 - y_3h_1 - y_2(y_2 + h_2))/(2y_3 + h_3) \\ y_0 &= (f_3 - y_3h_0 - y_2(2y_1 + h_1) - y_1h_2)/(2y_3 + h_3) \ . \end{aligned}$$

In summary, we will assume the following simplified forms for the curve equation. Here $H(y) = y_3x^3 + y_2x^2 + y_1x + y_0$.

- Odd Characteristic: $h(x) = 0$, $f_6 = 1$, $y_3 = 1$. If the field characteristic is not 3, then $f_5 = 0$ and $y_2 = 0$ (assumed in the operations counts). The formulas above for the coefficients of $H(y)$ reduce to $y_1 = f_4/2$ and $y_0 = f_3/2$.
- Even Characteristic: $h_3 = 1$, $h_2 = y_2 = f_5 = f_4 = f_3 = 0$, and $y_3 = \beta$, where $f_6 = \beta^2 + \beta$ comes from Definition 2.1. The formulas for the coefficients of $H(y)$ reduce to $y_1 = h_1y_3$ and $y_0 = h_0y_3$.

Although the formulas in the tables below are completely general, we make these assumptions when counting the number of field operations and giving simplified formulas for both odd and even characteristic.

We only count inversions, squarings and multiplications of finite field elements, which consist of the bulk of the computation when compared with additions and subtractions. In the tables below, we let I, S and M denote "inversion," "squaring," and "multiplication," respectively. QR denotes "quadratic root," which only occurs in the precomputation when calculating y_3 given generic coefficients.

Most of the cases below use several precomputed constants that follow directly from the curve coefficients. We present these precomputations in Table 1 and do not include them in the operation counts. Under the isomorphic transformations described above, most of these precomputed constants are very simple (0, 1, or 2) and are not necessary.

It may be necessary to change from the adapted basis into the reduced basis or vice versa. To change a divisor $[u, v]$ from reduced basis into adapted basis,

$$\begin{aligned} u' &= u \\ v' &= v \bmod u \ . \end{aligned}$$

To change a divisor $[u, v]$ from adapted basis into reduced basis,

$$\begin{aligned} u' &= u \\ v' &= H(y) + h - [(H(y) + h - v) \bmod u] \ . \end{aligned}$$

TABLE 1. Table of precomputed constants.

Precomputation		
Step	Expression	Operations
Generic Coefficients		
1	y_3 is a root of $y_3^2 + h_3 y_3 - f_6 = 0$. $c_3 = 2y_3 + h_3$, $d_3 = c_3^{-1}$, $y_2 = d_3(f_5 - y_3 h_2)$, $y_1 = d_3(f_4 - y_3 h_1 - y_2(y_2 + h_2))$, $y_0 = d_3(f_3 - y_3 h_0 - y_2(2y_1 + h_1) - y_1 h_2)$	1QR, 1I, 9C
2	$c_2 = 2y_2 + h_2$, $c_1 = y_1 + h_1$, $c_0 = y_0 + h_0$	
3	$d_2 = f_2 - h_0 y_2 - c_2 y_0$, $d_1 = f_2 - h_0 y_2 - c_1 y_1$, $d_0 = f_1 + c_1 h_0$	5C
Odd Characteristic		
1	$y_3 = 1$, $y_2 = 0$, $y_1 = f_4/2$, $y_0 = f_3/2$	
2	$c_3 = 2$, $c_2 = 0$, $c_1 = f_4/2$, $c_0 = f_3/2$	
3	$d_3 = 1/2$, $d_2 = f_2$, $d_1 = f_2 - y_1^2$, $d_0 = f_1$	1C
Even Characteristic		
1	y_3 is a root of $y_3^2 + y_3 + f_6 = 0$, $y_2 = 0$, $y_1 = y_3 h_1$, $y_0 = y_3 h_0$	1QR, 2C
2	$c_3 = 1$, $c_2 = 0$, $c_1 = (y_3 + 1)h_1$, $c_0 = (y_3 + 1)h_0$	
3	$d_3 = 1$, $d_2 = f_2$, $d_1 = f_2 + y_1(y_1 + h_1)$, $d_0 = f_1 + h_0(y_1 + h_1)$	2C

One disadvantage of the reduced basis is that hyperelliptic involution is not necessarily a trivial operation. In the adapted basis, one must perform the following operations:

$$u' = u$$
$$v' = h - v \bmod u$$

In the reduced basis, it is necessary to change the result of hyperelliptic involution into the reduced basis. This turns out to be equivalent to Step 1 in the Baby Step formulas in the reduced basis:

$$u' = u$$
$$v' = H(y) + h - [(H(y) + v) \bmod u]$$

In both even and odd characteristic, it is possible to reduce the operation count to 1S, 1M (see the Simplifications for Odd and Even Characteristic after the Baby Step formulas).

4. Baby Step

Let $[u, v]$ be a reduced representative of a divisor class. To compute the baby step $\rho[u, v] = [u', v']$, we apply the following formulas (Steps 1, 2, and 4 of Algorithm 2.2):

$$v' = H(y) + h - [(H(y) + v) \bmod u],$$
$$u' = \text{Monic}\left(\frac{f + hv' - (v')^2}{u}\right)$$

where, as mentioned above, $H(y) = y_3x^3 + y_2x^2 + y_1x + y_0$ is the principal part of a root y of $y^2 + h(x)y - f(x) = 0$.

Explicit formulas are derived by simply expanding the operations and using the formula for reducing a degree three polynomial $(H(y) + v)$ modulo a monic polynomial of degree two (u) described in [Lan05]. In Step 2, we equate the x^4 and x^3 coefficients of f and $H(y)^2 + H(y)h$ to simplify expressions involving coefficients of f, $H(y)$, and h. The resulting formulas are presented in the following tables.

4.1. Baby Step, Reduced Basis.

Baby Step, Reduced Basis, $\deg u = 2$		
Input	$u = x^2 + u_1x + u_0$, $v = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_1x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho[u, v]$	
Step	Expression	Operations
1	$\underline{v' = H(y) + h - [(H(y) + v) \bmod u]}$ $t_1 = c_3u_1$, $t_2 = c_2 - t_1$, $t_3 = t_2 \cdot u_0$ $v'_1 = h_1 - v_1 + (c_3 + t_2) \cdot (u_0 + u_1) - t_1 - t_3$ $v'_0 = h_0 - v_0 + t_3$	2M, 1C
2	$w_0 = c_0 - v'_0$, $w_1 = c_1 - v'_1$	
General case: $w_1 \neq 0$ $(\deg(u') = 2)$		
3	$\underline{u' = \text{Monic}((f + hv' - (v')^2)/u)}$ $w_2 = c_2w_0$, $w_3 = c_3w_1$, $I = w_3^{-1}$ $u'_1 = I \cdot ((c_2 + c_3)(w_0 + w_1) - w_2 - w_3) - u_1$ $u'_0 = I \cdot (d_2 + v'_1 \cdot (h_1 - v'_1) + w_2) - u_0 - u_1 \cdot u'_1$	1I, 4M, 3C
Special case 1: $w_1 = 0$ and $w_0 \neq 0$ $(u' = x + u'_0)$		
$3'$	$\underline{u' = \text{Monic}((f + hv' - (v')^2)/u)}$ $w_2 = c_2w_0$, $w_3 = c_3w_0$, $I = w_3^{-1}$ $u'_0 = I \cdot (d_2 + v'_1 \cdot (h_1 - v'_1) + w_2) - u_1$	1I, 2M, 2C
Special case 2: $w_1 = w_0 = 0$ $(u' = 1, v' = y + h)$		
$3''$	$u'_0 = 1$	
Total	General case	1I, 6M, 4C
	Special case 1	1I, 4M, 3C
	Special case 2	2M, 1C

Simplifications for Odd and Even Characteristic.

1. In Step 1, we skip the calculation of t_1, t_2, and t_3. Let

$$v'_1 = -v_1 - 2(u_1^2 - u_0) \ ,$$
$$v'_0 = -v_0 - 2u_0 \cdot u_1$$

in odd characteristic. In even characteristic, let

$$v'_1 = h_1 + v_1 + u_1^2 + u_0 \ ,$$
$$v'_0 = h_0 + v_0 + u_0 \cdot u_1 \ .$$

This reduces the operation count of Step 2 to 1S, 1M in both odd and even characteristic.

2. In Step 3, the general case simplifies to

$$\begin{aligned} w_3 &= f_4 - 2v_1' \ , \\ u_1' &= I \cdot (f_3 - 2v_0') - u_1 \ , \\ u_0' &= I \cdot \left(f_2 - (v_1')^2\right) - u_0 - u_1 \cdot u_1' \end{aligned}$$

in odd characteristic, and in even characteristic

$$\begin{aligned} w_3 &= y_1 + h_1 + v_1' \ , \\ u_1' &= I \cdot (y_0 + h_0 + v_0') + u_1 \ , \\ u_0' &= I \cdot (f_2 + v_1' \cdot (h_1 + v_1')) + u_0 + u_1 \cdot u_1' \ . \end{aligned}$$

This reduces the operation count of Step 3 to 1I, 1S, 3M in odd characteristic and 1I, 4M in even characteristic. For special values of h_1 (such as 0 or 1), one multiplication can be changed into a squaring for an operation count of 1I, 1S, 3M in even characteristic.

If $w_1 = 0$ and $w_0 \neq 0$, then Step 3′ in Special case 1 simplifies to

$$\begin{aligned} w_3 &= f_3 - 2v_0' \ , \\ u_0' &= I \cdot \left(f_2 - (v_1')^2\right) - u_1 \end{aligned}$$

in odd characteristic, and in even characteristic

$$\begin{aligned} w_3 &= y_0 + h_0 + v_0' \ , \\ u_0' &= I \cdot (f_2 + v_1' \cdot (h_1 + v_1')) + u_1 \ . \end{aligned}$$

If $w_0 = 0$, then Special case 2 applies and the output can be computed directly.

In summary, the number of operations required to compute a baby step in the general case ($\deg(u) = 2$) in odd characteristic is 1I, 2S, and 4M. In even characteristic, the operation count is 1I, 1S, 5M.

Baby Step, Reduced Basis, Special Case ($\deg(u) = 1$). One special case of the baby step operation is when the input divisor has $\deg(u) = 1$. Explicit formulas for the operation in this case follow.

Baby Step, Reduced Basis, $\deg u = 1$		
Input	$u = x + u_0$, $v = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho[u, v]$	
Step	Expression	Operations
	Baby Step	
1	$\underline{v' = H(y) + h - [(H(y) + v) \bmod u]}$ $v'_1 = c_1$, $v'_0 = h_0 - v_0 + u_0 \cdot (c_1 + y_1 - u_0 \cdot (c_2 - c_3 u_0))$	2M, 1C
2	$w_0 = d_1 + c_2(h_0 - v'_0)$, $w_1 = c_0 - v'_0$	1C
	General case: $w_1 \neq 0$ ($\deg(u') = 2$)	
3	$\underline{u' = \text{Monic}((f + hv' - (v')^2)/u)}$ $w_3 = c_3 w_1$, $I = w_3^{-1}$ $u'_1 = I \cdot w_0 - u_0$, $u'_0 = I \cdot (d_0 - (c_1 + y_1)v'_0) - u_0 \cdot u'_1$	1I, 3M, 2C
	Special case 1: $w_1 = 0$ and $w_0 \neq 0$ ($u' = x + u'_0$)	
$3'$	$\underline{u' = \text{Monic}((f + hv' - (v')^2)/u)}$ $I = w_0^{-1}$, $u'_0 = I \cdot (d_0 - (c_1 + y_1)v'_0) - u_0$	1I, 1M, 1C
	Special case 2: $w_1 = w_0 = 0$ ($u' = 1$, $v' = y + h$)	
$3''$	$u'_0 = 1$	
Total	General case	1I, 5M, 4C
	Special case 1	1I, 3M, 3C
	Special case 2	2M, 2C

Simplifications for Odd and Even Characteristic.

1. In Step 1, we let

$$\begin{aligned} v'_1 &= y_1 \ , \\ v'_0 &= 2(y_1 + u_0^2) \cdot u_0 - v_0 \end{aligned}$$

in odd characteristic. In even characteristic, let

$$\begin{aligned} v'_1 &= y_1 + h_1 \ , \\ v'_0 &= h_0 + v_0 + (h_1 + u_0^2) \cdot u_0 \ . \end{aligned}$$

This reduces the operation count of Step 2 to 1S, 1M in both odd and even characteristic.

2. In Step 3, the general case simplifies to

$$\begin{aligned} w_3 &= f_3 - 2v'_0 \ , \\ u'_1 &= d_1 I - u_0 \ , \\ u'_0 &= I \cdot (f_1 - 2y_1 v'_0) - u_0 \cdot u'_1 \end{aligned}$$

in odd characteristic and

$$\begin{aligned} w_3 &= y_0 + h_0 + v'_0 \ , \\ u'_1 &= d_1 I + u_0 \ , \\ u'_0 &= I \cdot (d_0 + h_1 v'_0) + u_0 \cdot u'_1 \end{aligned}$$

in even characteristic. This reduces the operation count of Step 3 to 1I, 2M, 2C in both cases.

If $w_1 = 0$ and $w_0 \neq 0$, then Step 3′ of Special case 1 simplifies to

$$\begin{aligned} I &= d_1^{-1} \ , \\ u_0' &= I\left(d_0 - 2v_0' y_1\right)) - u_0 \end{aligned}$$

in odd characteristic and

$$\begin{aligned} I &= d_1^{-1} \ , \\ u_0' &= I\left(d_0 + h_1 v_0'\right) + u_0 \end{aligned}$$

in even characteristic. Note that d_1 is a precomputed constant, so finding $I = d^{-1}$ can be treated as a constant multiplication. If $w_1 = 0$ and $w_0 = 0$, then Special case 2 applies and the output can be computed directly.

When $\deg(u) = 1$, the number of operations for the general case ($\deg(u') = 2$) is 1I, 1S, 3M, and 2C in both odd and even characteristic.

Baby Step, Reduced Basis, Special Case ($\deg(u) = 0$). The last special case to consider is when the input divisor is the identity, which in reduced basis means $u = 1$ and $v = H(y) + h$. In this case, we have $v' = H(y) + h$ and $u' = \text{Monic}(f - hH(y) - H(y)^2)$, so the formulas for u' and v' only involve on constants from the defining equations of the function field. Thus, the output for this special case may be completely precomputed when fixing the field parameters.

4.2. Baby Step, Adapted Basis. The same formulas are used to compute baby steps in adapted basis. There are two differences from the reduced basis formulas:

1. The input divisor has $\deg(v) < \deg(u)$ (i.e., it is in adapted basis).
2. The output v' is reduced modulo u' (to ensure that the result is again in adapted basis).

The resulting explicit formulas are given below.

Baby Step, Adapted Basis, $\deg u = 2$		
Input	$u = x^2 + u_1 x + u_0$, $v = v_1 x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho[u, v]$	
Step	Expression	Operations
1	$\underline{\widetilde{v} = H(y) + h - [(H(y) + v) \bmod u]}$ $t_1 = y_3 u_1$, $t_2 = y_2 - t_1$, $t_3 = t_2 \cdot u_0$ $\widetilde{v}_1 = h_1 - v_1 + (y_3 + t_2) \cdot (u_0 + u_1) - t_1 - t_3$ $\widetilde{v}_0 = h_0 - v_0 + t_3$	2M, 1C
2	$w_0 = c_0 - \widetilde{v}_0$, $w_1 = c_1 - \widetilde{v}_1$	
General case: $w_1 \neq 0$ ($\deg(u') = 2$)		
3	$\underline{u' = \text{Monic}((f + h\widetilde{v} - (\widetilde{v})^2)/u)}$ $w_2 = c_2 w_0$, $w_3 = c_3 w_1$, $I = w_3^{-1}$ $u'_1 = I \cdot ((c_2 + c_3)(w_0 + w_1) - w_2 - w_3) - u_1$ $u'_0 = I \cdot (d_2 + \widetilde{v}_1 \cdot (h_1 - \widetilde{v}_1) + w_2) - u_0 - u_1 \cdot u'_1$	1I, 4M, 3C
4	$\underline{v' = \widetilde{v} \bmod u'}$ $t_1 = (y_3 + h_3) u'_1$, $t_2 = y_2 + h_2 - t_1$, $t_3 = t_2 \cdot u'_0$ $v'_1 = \widetilde{v}_1 - (c_3 - y_3 + t_2) \cdot (u'_0 + u'_1) + t_1 + t_3$, $v'_0 = \widetilde{v}_0 - t_3$	2M, 1C
Special case 1: $w_1 = 0$ and $w_0 \neq 0$ ($u' = x + u'_0$)		
3′	$\underline{u' = \text{Monic}((f + h\widetilde{v} - (\widetilde{v})^2)/u)}$ $w_2 = c_2 w_0$, $w_3 = c_3 w_0$, $I = w_3^{-1}$ $u'_0 = I \cdot (d_2 + \widetilde{v}_1 \cdot (h_1 - \widetilde{v}_1) + w_2) - u_1$	1I, 2M, 2C
4′	$\underline{v' = \widetilde{v} \bmod u'}$ $v'_1 = 0$, $v'_0 = \widetilde{v}_0 - u'_0 \cdot (\widetilde{v}_1 - u'_0 \cdot (c_2 - y_2 - (c_3 - y_3) u'_0))$	2M, 1C
Special case 2: $w_1 = w_0 = 0$ ($u' = 1$, $v' = 0$)		
3″	$u'_0 = 1$, $v'_1 = 0$, $v'_0 = 0$	
Total	General case	1I, 8M, 5C
	Special case 1	1I, 6M, 4C
	Special case 2	2M, 1C

Simplifications for Odd and Even Characteristic.

1. In Step 1, we skip the calculation of t_1, t_2, and t_3. In odd characteristic, let

$$\widetilde{v}_1 = u_0 - u_1^2 - v_1 \ ,$$
$$\widetilde{v}_0 = -v_0 - u_0 \cdot u_1$$

   and in even characteristic, let

$$\widetilde{v}_1 = h_1 + v_1 + y_3(u_1^2 + u_0) \ ,$$
$$\widetilde{v}_0 = h_0 + v_0 + y_3 u_0 \cdot u_1 \ .$$

   This reduces the operation count of Step 2 to 1S, 1M in odd characteristic and 1S, 1M, 2C in even characteristic.
2. In Step 3, the same simplifications can be made as for the reduced basis for the general and special cases (with v'_1 and v'_0 replaced with $\widetilde{v}_1$ and $\widetilde{v}_0$, respectively). In the general case, this results in operation counts of to 1I, 1S, 3M in odd characteristic and 1I, 4M in even characteristic. For special values of h_1 (such as 0 or 1), one multiplication can be changed into a squaring for an operation count of 1I, 1S, 3M in even characteristic.

3. In Step 4, assuming the general case with $\deg(u') = 2$, let

$$v'_1 = \widetilde{v}_1 - u'_0 + (u'_1)^2 \ ,$$
$$v'_0 = \widetilde{v}_0 + u'_0 \cdot u'_1$$

in odd characteristic and

$$v'_1 = \widetilde{v}_1 + (y_3 + 1)((u'_1)^2 + u'_0) \ ,$$
$$v'_0 = \widetilde{v}_0 + (y_3 + 1)u'_0 \cdot u'_1$$

in even characteristic. This costs 1S, 1M in odd characteristic, and 1S, 1M, 2C in even characteristic.

For Step $4'$ of Special case 1, we have $v'_0 = \widetilde{v}_0 - u'_0 \cdot (\widetilde{v}_1 + (u'_0)^2)$ in odd characteristic and $v'_0 = \widetilde{v}_0 + u'_0 \cdot (\widetilde{v}_1 + (y_3 + 1)(u'_0)^2)$ in even characteristic.

When $\deg(u) = 2$, the total number of operations for the general case ($\deg(u') = 2$) in odd characteristic is 1I, 3S, and 5M. In even characteristic it is 1I, 2S, 6M, and 4C.

Baby Step, Adapted Basis, Special Case ($\deg(u) = 1$*).*

Baby Step, Adapted Basis, $\deg u = 1$		
Input	$u = x + u_0$, $v = v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho[u, v]$	
Step	Expression	Operations
1	$\underline{\widetilde{v} = H(y) + h - [(H(y) + v) \bmod u]}$ $\widetilde{v}_1 = c_1$, $\widetilde{v}_0 = h_0 - v_0 + u_0 \cdot (y_1 - u_0 \cdot (y_2 - y_3 u_0))$	2M, 1C
2	$w_0 = d_1 + c_2(h_0 - \widetilde{v}_0)$, $w_1 = c_0 - \widetilde{v}_0$	1C
General case: $w_1 \neq 0$ ($\deg(u') = 2$)		
3	$\underline{u' = \text{Monic}((f + h\widetilde{v} - (\widetilde{v})^2)/u)}$ $w_3 = c_3 w_1$, $I = w_3^{-1}$, $u'_1 = I \cdot w_0 - u_0$, $u'_0 = I \cdot (d_0 - (c_1 + y_1)\widetilde{v}_0) - u_0 \cdot u'_1$	1I, 3M, 2C
4	$\underline{v' = \widetilde{v} \bmod u'}$ $t_1 = (y_3 + h_3)u'_1$, $t_2 = y_2 + h_2 - t_1$, $t_3 = t_2 \cdot u'_0$ $v'_1 = \widetilde{v}_1 - (c_3 + t_2) \cdot (u'_0 + u'_1) + t_1 + t_3$, $v'_0 = \widetilde{v}_0 - t_3$	2M, 1C
Special case 1: $w_1 = 0$ and $w_0 \neq 0$ ($u' = x + u'_0$)		
$3'$	$\underline{u' = \text{Monic}((f + h\widetilde{v} - (\widetilde{v})^2)/u)}$ $I = w_0^{-1}$, $u'_0 = I \cdot (d_0 - (c_1 + y_1)\widetilde{v}_0) - u_0$	1I, 1M, 1C
$4'$	$\underline{v' = \widetilde{v} \bmod u'}$ $v'_1 = 0$, $v'_0 = \widetilde{v}_0 - u'_0 \cdot (c_1 - u'_0 \cdot (y_2 + h_2 - (y_3 + h_3)u'_0))$	2M, 1C
Special case 2: $w_1 = w_0 = 0$ ($u' = 1$, $v' = 0$)		
$3''$	$u'_0 = 1$, $v'_1 = 0$, $v'_0 = 0$	
Total	General case	1I, 7M, 5C
	Special case 1	1I, 5M, 4C
	Special case 2	2M, 2C

Simplifications for Odd and Even Characteristic.

1. In Step 1, let

$$\widetilde{v}_1 = y_1 \ ,$$
$$\widetilde{v}_0 = (y_1 + u_0^2) \cdot u_0 - v_0$$

in odd characteristic. In even characteristic, let

$$\widetilde{v}_1 = y_1 + h_1 \ ,$$
$$\widetilde{v}_0 = h_0 + v_0 + u_0 \cdot (y_1 + y_3 u_0^2) \ .$$

This reduces the operation count of Step 2 to 1S, 1M in odd characteristic and 1S, 1M, 1C in even characteristic.

2. In Step 3, the same simplifications can be made as for the reduced basis for the general and special cases (with v_0' replaced with $\widetilde{v}_0$). In the general case, this results in operation counts of 1I, 2M, and 2C in both odd and even characteristic.
3. In Step 4, assuming the general case with $\deg(u') = 2$, let

$$v_1' = \widetilde{v}_1 - u_0' + (u_1')^2 \ ,$$
$$v_0' = \widetilde{v}_0 + u_0' \cdot u_1'$$

in odd characteristic and

$$v_1' = \widetilde{v}_1 + (y_3 + 1)((u_1')^2 + u_0') \ ,$$
$$v_0' = \widetilde{v}_0 + (y_3 + 1)u_0' \cdot u_1'$$

in even characteristic. This costs 1S, 1M in odd characteristic, and 1S, 1M, 2C in even characteristic.

For Step 4′ in Special case 1, we have $v_0' = \widetilde{v}_0 - u_0'(y_1 + (u_0')^2)$ in odd characteristic and $v_0' = \widetilde{v}_0 + u_0'(c_1 + (y_3 + 1)(u_0')^2)$ in even characteristic.

When $\deg(u) = 1$, the total number of operations for the general case of $\deg(u') = 2$ in odd characteristic is 1I, 2S, 4M, and 2C. In even characteristic it is 1I, 2S, 4M, and 5C.

Baby Step, Adapted Basis, Special Case ($\deg(u) = 0$*).* As with the reduced basis description, the last special case to consider is when the input divisor is the identity, which means $u = 1$ and $v = 0$ in adapted basis. In this case, we have $u' = \text{Monic}(f - hH(y) - H(y)^2)$ and $v' = (H(y) + h) \bmod u'$. Again, the formulas for u' and v' only involve constants from the defining equations of the function field and the output for this special case may be completely precomputed when fixing the field parameters.

5. Inverse Baby Step

Let $[u, v]$ be a reduced representative of a divisor class. To compute $\rho^{-1}[u, v] = [u', v']$, we apply the following formulas:

$$u' = \text{Monic}\left(\frac{f + hv - v^2}{u}\right)$$
$$v' = H(y) + h - [(H(y) + v) \bmod u'] \ .$$

Explicit formulas are derived in a similar manner to those for computing $\rho[u, v]$, and are presented in the following tables.

5.1. INVERSE BABY STEP, REDUCED BASIS.

Inverse Baby Step, Reduced Basis, $\deg u = 2$		
Input	$u = x^2 + u_1x + u_0$, $v = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_1x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho^{-1}[u, v]$	
Step	Expression	Operations
1	$w_0 = c_0 - v_0$, $w_1 = c_1 - v_1$	
General case: $w_1 \neq 0$ ($\deg(u') = 2$)		
2	$\underline{u' = \text{Monic}((f + hv - v^2)/u)}$ $w_2 = c_2w_0$, $w_3 = c_3w_1$, $I = w_3^{-1}$ $u'_1 = I \cdot ((c_2 + c_3)(w_0 + w_1) - w_2 - w_3) - u_1$ $u'_0 = I \cdot (d_2 + v_1 \cdot (h_1 - v_1) + w_2) - u_0 - u_1 \cdot u'_1$	1I, 4M, 3C
3	$\underline{v' = H(y) + h - [(H(y) + v) \bmod u']}$ $t_1 = c_3u'_1$, $t_2 = c_2 - t_1$, $t_3 = t_2 \cdot u'_0$ $v'_1 = h_1 - v_1 + (c_3 + t_2) \cdot (u'_0 + u'_1) - t_1 - t_3$ $v'_0 = h_0 - v_0 + t_3$	2M, 1C
Special case 1: $w_1 = 0$ and $w_0 \neq 0$ ($u' = x + u'_0$)		
$2'$	$\underline{u' = \text{Monic}((f + hv - v^2)/u)}$ $w_2 = c_2w_0$, $w_3 = c_3w_0$, $I = w_3^{-1}$ $u'_0 = I \cdot (d_2 + v_1 \cdot (h_1 - v_1) + w_2) - u_1$	1I, 2M, 2C
$3'$	$\underline{v' = H(y) + h - [(H(y) + v) \bmod u']}$ $v'_1 = c_1$, $v'_0 = h_0 - v_0 + u'_0 \cdot (c_1 + y_1 - u'_0 \cdot (c_2 - c_3u'_0))$	2M, 1C
Special case 2: $w_1 = w_0 = 0$ ($u' = 1$, $v' = y + h$)		
$2''$	$u'_0 = 1$, $v'_1 = c_1$, $v'_0 = c_0$	
Total	General case	1I, 6M, 4C
	Special case 1	1I, 4M, 3C
	Special case 2	

Simplifications for Odd and Even Characteristic.

1. In Step 2, let

$$\begin{aligned} w_3 &= f_4 - 2v_1 \ , \\ u'_1 &= I \cdot (f_3 - 2v_0) - u_1 \ , \\ u'_0 &= I \cdot \left(f_2 - v_1^2\right) - u_0 - u_1 \cdot u'_1 \end{aligned}$$

in odd characteristic. In even characteristic, let

$$\begin{aligned} w_3 &= y_1 + h_1 + v_1 \ , \\ u'_1 &= I \cdot (y_0 + h_0 + v'_0) + u_1 \ , \\ u'_0 &= I \cdot (f_2 + v_1 \cdot (h_1 + v_1)) + u_0 + u_1 \cdot u'_1 \ . \end{aligned}$$

This reduces the operation count of Step 2 to 1I, 1S, 3M in odd characteristic and 1I, 4M in even characteristic. For special values of h_1 (such as 0 or 1), one multiplication can be changed into a squaring for an operation count of 1I, 1S, 3M in even characteristic.

If $w_1 = 0$ and $w_0 \neq 0$, then Step $2'$ of Special case 1 simplifies to

$$\begin{aligned} w_0 &= f_3 - 2v_0 \ , \\ u'_0 &= I \cdot \left(f_2 - v_1^2\right) - u_1 \end{aligned}$$

in odd characteristic, and in even characteristic

$$w_0 = y_0 + h_0 + v_0 \ ,$$
$$u_0' = I \cdot (f_2 + v_1 \cdot (h_1 + v_1)) + u_1 \ .$$

If $w_1 = 0$ and $w_0 = 0$, then Special case 2 applies and the output can be computed directly.

2. In Step 3, we skip the calculation of t_1, t_2, and t_3. Let

$$v_1' = -v_1 + 2(u_0' - (u_1')^2) \ ,$$
$$v_0' = -v_0 - 2u_0' \cdot u_1'$$

in odd characteristic. In even characteristic, let

$$v_1' = h_1 + v_1 + u_0' + (u_1')^2 \ ,$$
$$v_0' = h_0 + v_0 + u_0' \cdot u_1' \ .$$

This reduces the operation count of Step 3 to 1S, 1M in both odd and even characteristic.

In Step 3′ of Special case 1, we have $v_0' = 2u_0' \cdot (y_1 + (u_0')^2) - v_0$ in odd characteristic and $v_0' = h_0 + v_0 + u_0' \cdot (h_1 + (u_0')^2)$ in even characteristic.

In the general case where the input divisor has $\deg(u) = 2$, the total number of operations for an inverse baby step in odd characteristic is 1I, 2S, 4M, and in even characteristic is 1I, 1S, 5M.

Inverse Baby Step, Reduced Basis, Special Case ($\deg(u) = 1$*).*

Inverse Baby Step, Reduced Basis, $\deg u = 1$		
Input	$u = x + u_0$, $v = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho^{-1}[u, v]$	
Step	Expression	Operations
1	$w_1 = c_0 - v_0$, $w_0 = d_1 + c_2(h_0 - v_0)$	1C
General case: $w_1 \neq 0$ ($\deg(u') = 2$)		
2	$\underline{u' = \text{Monic}((f + hv - v^2)/u)}$ $w_3 = c_3 w_1$, $I = w_3^{-1}$, $u_1' = I \cdot w_0 - u_0$ $u_0' = I \cdot (d_0 - (c_1 + y_1)v_0) - u_0 \cdot u_1'$	1I, 3M, 2C
3	$\underline{v' = H(y) + h - [(H(y) + v) \bmod u']}$ $t_1 = c_3 u_1'$, $t_2 = c_2 - t_1$, $t_3 = t_2 \cdot u_0'$ $v_1' = h_1 - v_1 + (c_3 + t_2) \cdot (u_0' + u_1') - t_1 - t_3$ $v_0' = h_0 - v_0 + t_3$	2M, 1C
Special case 1: $w_1 = 0$ and $w_0 \neq 0$ ($u' = x + u_0'$)		
2′	$\underline{u' = \text{Monic}((f + hv - v^2)/u)}$ $I = w_0^{-1}$, $u_0' = I \cdot (d_0 - (c_1 + y_1)v_0) - u_0$	1I, 1M, 1C
3′	$\underline{v' = H(y) + h - [(H(y) + v) \bmod u']}$ $v_1' = c_1$, $v_0' = h_0 - v_0 + u_0' \cdot (c_1 + y_1 - u_0' \cdot (c_2 - c_3 u_0'))$	2M, 1C
Special case 2: $w_1 = w_0 = 0$ ($u' = 1$, $v' = y + h$)		
2″	$u_0' = 1$, $v_1' = c_1$, $v_0' = c_0$	
Total	General case	1I, 5M, 4C
	Special case 1	1I, 3M, 3C
	Special case 2	1C

Simplifications for Odd and Even Characteristic.

1. In Step 2, the general case simplifies to

$$\begin{aligned} w_3 &= f_3 - 2v_0 \ , \\ u_1' &= d_1 I - u_0 \ , \\ u_0' &= I \cdot (f_1 - 2y_1 v_0) - u_0 \cdot u_1' \end{aligned}$$

in odd characteristic and

$$\begin{aligned} w_3 &= y_0 + h_0 + v_0 \ , \\ u_1' &= d_1 I + u_0 \ , \\ u_0' &= I \cdot (d_0 + h_1 v_0) + u_0 \cdot u_1' \end{aligned}$$

in even characteristic. This reduces the operation count of Step 3 to 1I, 2M, 2C in both cases.

If $w_1 = 0$ and $w_0 \neq 0$, then Step 2′ of Special case 1 simplifies to

$$\begin{aligned} I &= d_1^{-1} \ , \\ u_0' &= I\,(d_0 - 2y_1 v_0) - u_0 \end{aligned}$$

in odd characteristic and

$$\begin{aligned} I &= d_1^{-1} \ , \\ u_0' &= I\,(d_0 + h_1 v_0) + u_0 \end{aligned}$$

in even characteristic. If $w_1 = 0$ and $w_0 = 0$, then Special case 2 applies and the output can be computed directly.

2. The simplifications for Step 3 are the same as in the $\deg(u) = 2$ version of the algorithm, resulting in operation counts of 1S, 1M for the general case in both odd and even characteristic.

When $\deg(u) = 1$, the number of operations for the general case ($\deg(u') = 2$) is 1I, 1S, 3M, and 2C in both odd and even characteristic.

Inverse Baby Step, Reduced Basis, Special Case ($\deg(u) = 0$*).* The last special case to consider is when the input divisor is the identity, which in reduced basis means $u = 1$ and $v = y + h$. In this case, we have $u' = \text{Monic}(f - hH(y) - H(y)^2)$ and $v' = H(y) + h - [(2H(y) + h) \bmod u']$, so as before the formulas for u' and v' only involve constants from the defining equations of the function field.

5.2. Inverse Baby Step, Adapted Basis. As with the baby step, almost the same formulas are used to compute the inverse baby step using adapted basis as in the reduced basis case. Because the input divisor has $\deg(v) < \deg(u)$, it first has to be put into reduced basis before applying the formulas. In addition, the output should once again be in adapted basis, so the resulting v' must be reduced modulo u'. The resulting formulas are

$$\begin{aligned} \widetilde{v} &= H(y) + h - [(H(y) + h - v) \bmod u] \\ u' &= \text{Monic}\left(\frac{f + h\widetilde{v} - (\widetilde{v})^2}{u}\right) \\ v' &= (H(y) + h - [(H(y) + \widetilde{v}) \bmod u']) \bmod u' = (h - \widetilde{v}) \bmod u' \ . \end{aligned}$$

Explicit formulas are presented below.

Inverse Baby Step, Adapted Basis, $\deg u = 2$		
Input	$u = x^2 + u_1x + u_0$, $v = v_1x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho^{-1}[u, v]$	
Step	Expression	Operations
1	$\underline{\widetilde{v} = H(y) + h - [(H(y) + h - v) \bmod u]}$ $t_1 = (c_3 - y_3)u_1$, $t_2 = c_2 - y_2 - t_1$, $t_3 = t_2 \cdot u_0$ $\widetilde{v}_1 = v_1 + (c_3 - y_3 + t_2) \cdot (u_0 + u_1) - t_1 - t_3$, $\widetilde{v}_0 = v_0 + t_3$	2M, 1C
2	$w_0 = c_0 - \widetilde{v}_0$, $w_1 = c_1 - \widetilde{v}_1$	
General case: $w_1 \neq 0$ ($\deg(u') = 2$)		
3	$\underline{u' = \text{Monic}((f + h\widetilde{v} - (\widetilde{v})^2)/u)}$ $w_2 = c_2w_0$, $w_3 = c_3w_1$, $I = w_3^{-1}$ $u'_1 = I \cdot ((c_2 + c_3)(w_0 + w_1) - w_2 - w_3) - u_1$ $u'_0 = I \cdot (d_2 + \widetilde{v}_1 \cdot (h_1 - \widetilde{v}_1) + w_2) - u_0 - u_1 \cdot u'_1$	1I, 4M, 3C
4	$\underline{v' = (h - \widetilde{v}) \bmod u'}$ $t_1 = y_3u'_1$, $t_3 = y_2 - t_1$, $t_3 = t_2 \cdot u'_0$ $v'_1 = h_1 - \widetilde{v}_1 + (y_3 + t_2) \cdot (u'_0 + u'_1) - t_1 - t_3$ $v'_0 = h_0 - \widetilde{v}_0 + t_3$	2M, 1C
Special case 1: $w_1 = 0$ and $w_0 \neq 0$ ($u' = x + u'_0$)		
$3'$	$\underline{u' = \text{Monic}((f + h\widetilde{v} - (\widetilde{v})^2)/u)}$ $w_2 = c_2w_0$, $w_3 = c_3w_0$, $I = w_3^{-1}$ $u'_0 = I \cdot (d_2 + \widetilde{v}_1 \cdot (h_1 - \widetilde{v}_1) + w_2) - u_1$	1I, 2M, 2C
$4'$	$\underline{v' = (h - \widetilde{v}) \bmod u'}$ $\underline{v'_0 = h_0 - \widetilde{v}_0 - u'_0 \cdot (h_1 - \widetilde{v}_1 + u'_0 \cdot (y_2 - y_3u'_0))}$	2M, 1C
Special case 2: $w_1 = w_0 = 0$ ($u' = 1$, $v' = 0$)		
$3''$	$u'_0 = 1$, $v'_1 = 0$, $v'_0 = 0$	
Total	General case	1I, 8M, 5C
	Special case 1	1I, 6M, 4C
	Special case 2	2M, 1C

Simplifications for Odd and Even Characteristic.

1. In Step 1, we skip the calculation of t_1, t_2, and t_3. In odd characteristic, let

$$\widetilde{v}_1 = u_0 - u_1^2 - v_1 \ ,$$
$$\widetilde{v}_0 = -v_0 - u_0 \cdot u_1$$

   and in even characteristic, let

$$\widetilde{v}_1 = h_1 + v_1 + y_3(u_1^2 + u_0) \ ,$$
$$\widetilde{v}_0 = h_0 + v_0 + y_3u_0 \cdot u_1 \ .$$

   This reduces the operation count of Step 2 to 1S, 1M in odd characteristic and 1S, 1M, 2C in even characteristic.
2. Step 3 can be simplified using the same formulas as for Step 2 of the reduced basis case (with v_1 and v_0 replaced with $\widetilde{v}_1$ and $\widetilde{v}_0$, respectively). This reduces the operation count in the general case ($\deg(u') = 2$) to 1I, 1S, 3M in odd characteristic and 1I, 4M in even characteristic. For special values of h_1 (such as 0 or 1), one multiplication can be changed into a squaring for an operation count of 1I, 1S, 3M in even characteristic.

3. In Step 4, we skip the calculation of t_1, t_2, and t_3. Assuming the general case with $\deg(u') = 2$, let

$$v'_1 = u'_0 - (u'_1)^2 - \widetilde{v}_1 \ ,$$
$$v'_0 = -\widetilde{v}_0 - u'_0 \cdot u'_1$$

in odd characteristic and

$$v'_1 = \widetilde{v}_1 + y_3(u'_0 + (u'_1)^2) \ ,$$
$$v'_0 = \widetilde{v}_0 + y_3 u'_0 \cdot u'_1$$

in even characteristic. This costs 1S, 1M in odd characteristic, and 1S, 1M, 2C in even characteristic.

For Step $4'$ of Special case 1, we have $v'_0 = u'_0 \cdot (y_1 + (u'_0)^2) - \widetilde{v}_0$ in odd characteristic and $v'_0 = h_0 + \widetilde{v}_0 + u'_0 \cdot (y_1 + y_3(u'_0)^2)$ in even characteristic.

When $\deg(u) = 2$, the total number of operations for the general case ($\deg(u') = 2$) in odd characteristic is 1I, 3S, and 5M. In even characteristic, the operation count is 1I, 2S, 6M, and 4C.

Inverse Baby Step, Adapted Basis, Special Case ($\deg(u) = 1$*).*

Inverse Baby Step, Adapted Basis, $\deg u = 1$		
Input	$u = x + u_0$, $v = v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = \rho^{-1}[u, v]$	
Step	Expression	Operations
1	$\underline{\widetilde{v} = H(y) + h - [(H(y) + h - v) \bmod u]}$ $\widetilde{v}_1 = c_1$, $\widetilde{v}_0 = v_0 + u_0 \cdot (c_1 - v_1 - u_0 \cdot (c_2 - y_2 - (c_3 - y_3)u_0))$	2M, 1C
2	$w_1 = c_0 - \widetilde{v}_0$, $w_0 = d_1 + c_2(h_0 - \widetilde{v}_0)$	1C
General case: $w_1 \neq 0$ ($\deg(u') = 2$)		
3	$\underline{u' = \text{Monic}((f + h\widetilde{v} - \widetilde{v}^2)/u)}$ $w_3 = c_3 w_1$, $I = w_3^{-1}$, $u'_1 = I \cdot w_0 - u_0$ $u'_0 = I \cdot (d_0 - (c_1 + y_1)\widetilde{v}_0) - u_0 \cdot u'_1$	1I, 3M, 2C
4	$\underline{v' = (h - \widetilde{v}) \bmod u'}$ $t_1 = y_3 u'_1$, $t_2 = y_2 - t_1$, $t_3 = t_2 \cdot u'_0$ $v'_1 = h_1 - \widetilde{v}_1 + (y_3 + t_2) \cdot (u'_0 + u'_1) - t_1 - t_3$ $v'_0 = h_0 - \widetilde{v}_0 + t_3$	2M, 1C
Special case 1: $w_1 = 0$ and $w_0 \neq 0$ ($u' = x + u'_0$)		
$3'$	$\underline{u' = \text{Monic}((f + hv - v^2)/u)}$ $I = w_0^{-1}$, $u'_0 = I \cdot (d_0 - (c_1 + y_1)\widetilde{v}_0) - u_0$	1I, 1M, 1C
$4'$	$\underline{v' = (h - v'') \bmod u'}$ $v'_0 = h_0 - \widetilde{v}_0 - u'_0 \cdot (h_1 - \widetilde{v}_1 + u'_0 \cdot (y_2 - y_3 u'_0))$	2M, 1C
Special case 2: $w_1 = w_0 = 0$ ($u' = 1$, $v' = 0$)		
$3''$	$u'_0 = 1$, $v'_1 = 0$, $v'_0 = 0$	
Total	General case	1I, 7M, 5C
	Special case 1	1I, 5M, 4C
	Special case 2	2M, 2C

Simplifications for Odd and Even Characteristic.

1. In Step 1, in odd characteristic let
$$\widetilde{v}_1 = y_1 \ ,$$
$$\widetilde{v}_0 = v_0 + u_0 \cdot (y_1 - v_1 + u_0^2)$$
and in even characteristic, let
$$\widetilde{v}_1 = y_1 + h_1 \ ,$$
$$\widetilde{v}_0 = h_0 + v_0 + u_0 \cdot (y_1 + y_3 u_0^2) \ .$$
This reduces the operation count of Step 2 to 1S, 1M in odd characteristic and 1S, 1M, 2C in even characteristic.
2. Step 3 can be simplified using the same formulas as for Step 3 of the reduced basis case (with v_0 replaced with $\widetilde{v}_0$). This reduces the operation count in the general case ($\deg(u') = 2$) to 1I, 2M, 2C both odd and even characteristic.
3. In Step 4, we skip the calculation of t_1, t_2, and t_3. Assuming the general case with $\deg(u') = 2$, let
$$v_1' = u_0' - (u_1')^2 - \widetilde{v}_1 \ ,$$
$$v_0' = -\widetilde{v}_0 - u_0' \cdot u_1'$$
in odd characteristic and
$$v_1' = \widetilde{v}_1 + (y_3 + 1)(u_0' + (u_1')^2) \ ,$$
$$v_0' = \widetilde{v}_0 + (y_3 + 1) u_0' \cdot u_1'$$
in even characteristic. This costs 1S, 1M in odd characteristic, and 1S, 1M, 2C in even characteristic.

   For Step 4′ of Special case 1, we have $v_0' = u_0' \cdot (y_1 + (u_0')^2) - \widetilde{v}_0$ in odd characteristic and $v_0' = h_0 + \widetilde{v}_0 + u_0' \cdot (y_1 + y_3 (u_0')^2)$ in even characteristic.

When $\deg(u) = 1$, the total number of operations for the general case of $\deg(u') = 2$ in odd characteristic is 1I, 2S, 4M, and 2C. In even characteristic, the operation count is 1I, 2S, 4M, and 6C.

Inverse Baby Step, Adapted Basis, Special Case ($\deg(u) = 0$*).* The last special case to consider is when the input divisor is the identity. In this case, we have $u' = \text{Monic}(f - hH(y) - H(y)^2)$ and $v' = -H(y) \bmod u'$, so as before the formulas for u' and v' only involve constants from the defining equations of the function field.

6. Divisor Addition

Let $[u_1, v_1]$ and $[u_2, v_2]$ be reduced representatives of two divisor classes. The main case of divisor addition occurs when the two divisor classes consist of four points on the curve which are different from each other and their opposites. This situation occurs precisely when $\deg(u_1) = \deg(u_2) = 2$ and u_1, u_2 are relatively prime. In the rare cases when u_1 or u_2 has degree less than 2, or when u_1 and u_2 are not relatively prime, the costs are considerably less than the main case.

In the table below, we do not assume that u_1 and u_2 are relatively prime before performing the operations. Since u_1 and u_2 are not relatively prime exactly when the computed value r is equal to zero, we treat this special case in the Comments section after the tables. We also skip the even and odd simplifications for all the special cases, since they rarely occur.

We describe the main case first. To optimize the computations, as described in [WPP05], we do not follow Algorithm 2.1 literally. Instead, we compute the following expressions, then show that these formulas give the desired result:

$$\begin{aligned} r &= \text{resultant of } u_1,\, u_2, & inv &\equiv r(u_2)^{-1} \pmod{u_1},\\ s' &= (v_1 - v_2)\cdot inv \pmod{u_1}, & s &= \tfrac{1}{r}s,\\ l &= s\cdot u_2, & k &= \tfrac{f+h\cdot v_2 - v_2^2}{u_2},\\ m &= k + s\cdot(h - 2v_2 - l), & m' &= m/m_4 = m \text{ made monic},\end{aligned}$$

$$\begin{aligned} u' &= m'/u_1,\\ \text{Adapted: } v' &\equiv h - v_2 - l \pmod{u'}, \text{or}\\ \text{Reduced: } v' &= H(y) + h - [(H(y) + v_2 + l) \bmod u'] \ .\end{aligned}$$

Assume u_1 and u_2 are both degree 2 and are relatively prime. To see that the above formulas are correct, note that Algorithm 2.1 implies

$$\begin{aligned} U_0 &= u_1 u_2\\ V_0 &\equiv v_2 + s u_2 = v_2 + l \pmod{U_0}\end{aligned}$$

where $s \equiv u_2^{-1}(v_1 - v_2) \pmod{u_1}$ and $l = s u_2$. The reduction step can be expressed as

$$\begin{aligned} V_1 &= h - V_0 + \left\lfloor \tfrac{V_0 + H(y)}{U_0} \right\rfloor \cdot U_0\\ U_1 &= \tfrac{f + h\cdot V_1 - V_1^2}{U_0} \ .\end{aligned}$$

Since V_0 and $H(y)$ both have degree 3 and U_0 has degree 4, $\left\lfloor \frac{V_0+H(y)}{U_0} \right\rfloor = 0$, and so

$$V_1 = h - V_0 = h - (v_2 + l)$$

Plugging this into the formula U_1 yields

$$\begin{aligned} U_1 &= \tfrac{f + h\cdot(h - (v_2+l)) - (h-(v_2+l))^2}{u_1 u_2}\\ &= \tfrac{f + h v_2 + h l - v_2^2 - 2 v_2 l - l^2}{u_1 u_2}\\ &= \tfrac{1}{u_1}\left(\tfrac{f + h v_2 - v_2^2}{u_2} + \tfrac{l(h - 2v_2 - l)}{u_2}\right)\\ &= \tfrac{k + s(h - 2v_2 - l)}{u_1}\end{aligned}$$

where $k = \frac{f + h v_2 - v_2^2}{u_2}$.

The final step is to reduce $[U_1, V_1]$ into either adapted or reduced basis. For the adapted basis, $[u', v']$ is found by setting u' to U_1 made monic, and $v' \equiv V_1 \pmod{u'}$. For the reduced basis, $[u', v']$ is found by setting $u' = U_1$ made monic, and $v' = H(y) + h - [(H(y) + h - V_1) \bmod u'] = H(y) + h - [(H(y) + v_2 + l) \bmod u']$. In the formulas, we first find the leading coefficient of $m = k + s(h - 2v_2 - l)$, compute its inverse, then compute $m' = m$ made monic.

The major advantage of using the reduced basis over the adapted basis is that the first two coefficients of v_2 are chosen to cancel the coefficients of the two highest degree coefficients in k regardless of characteristic or assumptions on the curve coefficients, reducing the number of operations to calculate k to at most one constant multiplication. In contrast, calculating k in the adapted basis requires at least one squaring and in general four constant multiplications.

We also note that the addition of divisors is exactly the same for both real and imaginary curves. These two steps can replace the first three steps of imaginary divisor addition found in [Lan05] for a savings of one squaring. For reference, this improvement makes 1I, 2S, 22M the least known number of field operations needed for divisor addition in the imaginary case.

Addition, Reduced Basis, $\deg u_1 = \deg u_2 = 2$		
Input	$u_1 = x^2 + u_{11}x + u_{10}$, $v_1 = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_{11}x + v_{10}$ $u_2 = x^2 + u_{21}x + u_{20}$, $v_2 = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_{21}x + v_{20}$ Precomputed Constants in Table 1	
Output	$[u', v'] = [u_1, v_1] + [u_2, v_2]$	
Step	Expression	Operations
	Addition	
1	$\underline{inv = z_1x + z_2}$ $z_0 = u_{10} - u_{20}$, $z_1 = u_{11} - u_{21}$, $z_2 = u_{11} \cdot z_1 - z_0$ $z_3 = u_{10} \cdot z_1$, $r = z_1 \cdot z_3 - z_0 \cdot z_2$	4M
2	$\underline{s' = s'_1x + s'_0}$ $w_0 = v_{10} - v_{20}$, $w_1 = v_{11} - v_{21}$, $s'_1 = w_0 \cdot z_1 - w_1 \cdot z_0$, $s'_0 = w_0 \cdot z_2 - w_1 \cdot z_3$ If $r = 0$, see Comments below.	4M
	Reduction	
3	$\underline{k = k_4x^4 + k_3x^3 + k_2x^2 + k_1x + k_0}$ $k_2 = c_3(c_1 - v_{21})$	1C
4	$\underline{\widehat{w}_0 = m'_4}$ $r_2 = r^2$, $\widehat{w}_0 = (s'_1 + c_3r) \cdot s'_1$ If $\widehat{w}_0 = 0$, see Special case.	1S, 1M, 1C
5	$\underline{s = \frac{1}{r}s' = s_1x + s_0}$, $\underline{\widehat{w}_3 = m_4^{-1}}$ $\widehat{w}_1 = (r \cdot \widehat{w}_0)^{-1}$, $\widehat{w}_2 = \widehat{w}_0 \cdot \widehat{w}_1$, $s_1 = s'_1 \cdot \widehat{w}_2$, $s_0 = s'_0 \cdot \widehat{w}_2$, $\widehat{w}_3 = r \cdot r_2 \cdot \widehat{w}_1$	1I, 6M
6	$\underline{l = l_3x^3 + l_2x^2 + l_1x + l_0}$ $\widetilde{w}_0 = s_0 \cdot u_{20}$, $\widetilde{w}_1 = s_1 \cdot u_{21}$ $l_2 = s_0 + \widetilde{w}_1$, $\widetilde{w}_2 = c_2 + l_2$ $l_1 = (s_0 + s_1) \cdot (u_{20} + u_{21}) - \widetilde{w}_0 - \widetilde{w}_1$, $l_0 = \widetilde{w}_0$	3M
7	$\underline{u' = x^2 + u'_1x + u'_0}$ $u'_1 = \widehat{w}_3 \cdot ((s_0 + \widetilde{w}_2) \cdot s_1 + c_3s_0) - u_{11}$ $u'_0 = \widehat{w}_3 \cdot ((2v_{21} - h_1 + l_1) \cdot s_1 + \widetilde{w}_2 \cdot s_0 - k_2) - u_{10} - u_{11} \cdot u'_1$	6M, 1C
8	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v'_1x + v'_0}$ $\overline{w}_0 = c_3 + s_1$, $\overline{w}_1 = u'_1 \cdot \overline{w}_0$, $\overline{w}_2 = \widetilde{w}_2 - \overline{w}_1$, $\overline{w}_3 = u'_0 \cdot \overline{w}_2$ $v'_1 = h_1 - v_{21} - l_1 + (u'_0 + u'_1) \cdot (\overline{w}_0 + \overline{w}_2) - \overline{w}_1 - \overline{w}_3$, $v'_0 = h_0 - v_{20} - l_0 + \overline{w}_3$	3M
Total	General case	1I, 1S, 27M, 3C

Special case: $\widehat{w}_0 = 0$ $(\deg(u') = 1)$		
$5'$	$\underline{\widehat{w}_1 = m_3'}$ If $s_1' = 0$, then $s_1 = 0$. Otherwise, $s_1 = -c_3$. $t_1 = c_2 + s_1 u_{21}$, $\widehat{w}_1 = s_1(s_0' + r \cdot t_1) + (s_1 + c_3)s_0'$	1M, 3C
$6'$	$\underline{s = \frac{1}{r}s' = s_1 x + s_0,\ \widehat{w}_3 = m_3^{-1}}$ $\widehat{w}_2 = (r \cdot \widehat{w}_1)^{-1}$, $s_0 = s_0' \cdot \widehat{w}_1 \cdot \widehat{w}_2$, $\widehat{w}_3 = r_2 \cdot \widehat{w}_2$	1I, 4M
$7'$	$\underline{u' = x + u_0'}$ $\widetilde{w}_0 = s_0 \cdot u_{21}$, $\widetilde{w}_1 = s_1 u_{20}$ $u_0' = \widehat{w}_3 \cdot (s_0^2 + c_2 s_0 + s_1(2(v_{21} + \widetilde{w}_0) + \widetilde{w}_1 - h_1) - k_2) - u_{11}$	1S, 2M, 2C
$8'$	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_1' x + v_0'}$ $v_1' = c_1$ $v_0' = h_0 - v_{20} - s_0 \cdot u_{20} + u_0' \cdot (y_1 + v_{21} + \widetilde{w}_0 + \widetilde{w}_1 - u_0' \cdot (s_0 + t_1 - (c_3 + s_1)u_0'))$	3M, 1C
Total	Special case	1I, 2S, 19M, 8C

Simplifications for Odd and Even Characteristic.

1. In Step 3, let $k_2 = f_4 - 2v_{21}$ in odd characteristic. In even characteristic, let $k_2 = c_1 + v_{21}$. This eliminates the constant multiplication in Step 3.
2. In Step 4, let $\widehat{w}_0 = (s_1' + r)^2 - r_2$ in odd characteristic. In even characteristic, let $\widehat{w}_0 = (s_1' + r) \cdot s_1'$. This reduces the operation count of Step 4 to 2S in odd characteristic and 1S, 1M in even characteristic.
3. In Step 6, we skip the calculation of $\widetilde{w}_2$ (since $c_2 = 0$). In Step 7, let

$$\begin{aligned} u_1' &= \widehat{w}_3 \cdot ((s_0 + l_2) \cdot s_1 + 2s_0) - u_{11} \ , \\ u_0' &= \widehat{w}_3 \cdot ((2v_{21} + l_1) \cdot s_1 + l_2 \cdot s_0 - k_2) - u_{10} - u_{11} \cdot u_1' \end{aligned}$$

   in odd characteristic. In even characteristic, let

$$\begin{aligned} u_1' &= \widehat{w}_3 \cdot ((s_0 + l_2) \cdot s_1 + s_0) + u_{11} \ , \\ u_0' &= \widehat{w}_3 \cdot ((h_1 + l_1) \cdot s_1 + l_2 \cdot s_0 + k_2) + u_{10} + u_{11} \cdot u_1' \ . \end{aligned}$$

   This reduces the operation count of Step 7 to 6M in both odd and even characteristic.
4. In Step 8, let

$$\begin{aligned} \overline{w}_0 &= s_1 + 2, \quad \overline{w}_2 = l_2 - \overline{w}_1 \ , \\ v_1' &= -v_{21} - l_1 + (u_0' + u_1') \cdot (\overline{w}_0 + \overline{w}_2) - \overline{w}_1 - \overline{w}_3 \ , \\ v_0' &= -v_{20} - l_0 + \overline{w}_3 \end{aligned}$$

   in odd characteristic. In even characteristic, let

$$\begin{aligned} \overline{w}_0 &= s_1 + 1, \quad \overline{w}_2 = l_2 + \overline{w}_1 \ , \\ v_1' &= h_1 + v_{21} + l_1 + (u_0' + u_1') \cdot (\overline{w}_0 + \overline{w}_2) + \overline{w}_1 + \overline{w}_3 \ , \\ v_0' &= h_0 + v_{20} + l_0 + \overline{w}_3 \ . \end{aligned}$$

When $\deg(u_1) = \deg(u_2) = 2$, the total operation count for addition in the reduced basis for the general case of $\deg(u') = 2$ is 1I, 2S, 26M in odd characteristic and 1I, 1S, 27M in even characteristic.

Addition, Adapted Basis, $\deg u_1 = \deg u_2 = 2$	
Input	$u_1 = x^2 + u_{11}x + u_{10}$, $v_1 = v_{11}x + v_{10}$ $u_2 = x^2 + u_{21}x + u_{20}$, $v_2 = v_{21}x + v_{20}$ Precomputed Constants in Table 1
Output	$[u', v'] = [u_1, v_1] + [u_2, v_2]$

Step	Expression	Operations
	Addition	
1	$\underline{inv = z_1 x + z_2}$ $z_0 = u_{10} - u_{20}$, $z_1 = u_{11} - u_{21}$, $z_2 = u_{11} \cdot z_1 - z_0$ $z_3 = u_{10} \cdot z_1$, $r = z_1 \cdot z_3 - z_0 \cdot z_2$	4M
2	$\underline{s' = s'_1 x + s'_0}$ $w_0 = v_{10} - v_{20}$, $w_1 = v_{11} - v_{21}$, $s'_1 = w_0 \cdot z_1 - w_1 \cdot z_0$, $s'_0 = w_0 \cdot z_2 - w_1 \cdot z_3$ If $r = 0$, see Comments below.	4M
	Reduction	
3	$\underline{k = k_4 x^4 + k_3 x^3 + k_2 x^2 + k_1 x + k_0}$ $k_4 = f_6$, $k_3 = f_5 - f_6 u_{21}$ $k_2 = f_4 + h_3 v_{21} - f_5 u_{21} + f_6(u_{21}^2 - u_{20})$	1S, 4C
4	$\underline{\widehat{w}_0 = m'_4}$ $r_2 = r^2$, $\widehat{w}_0 = s'_1 \cdot (s'_1 - h_3 r) - k_4 r_2$ If $\widehat{w}_0 = 0$, see Special case.	1S, 1M, 2C
5	$\underline{s = \frac{1}{r} s' = s_1 x + s_0}$, $\widehat{w}_3 = m_4^{-1}$ $\widehat{w}_1 = (r \cdot \widehat{w}_0)^{-1}$, $\widehat{w}_2 = \widehat{w}_0 \cdot \widehat{w}_1$, $s_1 = s'_1 \cdot \widehat{w}_2$, $s_0 = s'_0 \cdot \widehat{w}_2$, $\widehat{w}_3 = r \cdot r_2 \cdot \widehat{w}_1$	1I, 6M
6	$\underline{l = l_3 x^3 + l_2 x^2 + l_1 x + l_0}$ $\widetilde{w}_0 = s_0 \cdot u_{20}$, $\widetilde{w}_1 = s_1 \cdot u_{21}$, $l_2 = s_0 + \widetilde{w}_1$, $\widetilde{w}_2 = h_2 - l_2$ $l_1 = (s_0 + s_1) \cdot (u_{20} + u_{21}) - \widetilde{w}_0 - \widetilde{w}_1$, $l_0 = \widetilde{w}_0$	3M
7	$\underline{u' = x^2 + u'_1 x + u'_0}$ $u'_1 = \widehat{w}_3 \cdot ((s_0 - \widetilde{w}_2) \cdot s_1 - h_3 s_0 - k_3) - u_{11}$, $u'_0 = \widehat{w}_3 \cdot ((2v_{21} - h_1 + l_1) \cdot s_1 + \widetilde{w}_2 \cdot s_0 - k_2) - u_{10} -$ $u_{11} \cdot u'_1$	6M, 1C
8	$\underline{v' = v'_1 x + v'_0}$ $\overline{w}_0 = h_3 - s_1$, $\overline{w}_1 = u'_1 \cdot \overline{w}_0$, $\overline{w}_2 = \widetilde{w}_2 - \overline{w}_1$ $\overline{w}_3 = u'_0 \cdot \overline{w}_2$, $v'_1 = h_1 - v_{21} - l_1 - (u'_0 + u'_1) \cdot (\overline{w}_0 + \overline{w}_2) + \overline{w}_1 + \overline{w}_3$, $v'_0 = h_0 - v_{20} - l_0 - \overline{w}_3$	3M
Total		1I, 2S, 27M, 7C

Special case: $\widehat{w}_0 = 0$ $(\deg(u') = 1)$		
$5'$	$\underline{\widehat{w}_1 = m_3'}$ If $s_1' = -y_3 r$, let $s_1 = -y_3$. Otherwise, let $s_1 = y_3 + h_3$. $t_1 = s_1 u_{21} - h_2$, $\widehat{w}_1 = (2s_1 - h_3)s_0' + (s_1 t_1 - k_3) \cdot r$	1M, 3C
$6'$	$\underline{s = \frac{1}{r} s' = s_1 x + s_0,\ \widehat{w}_3 = m_3^{-1}}$ $\widehat{w}_2 = (r \cdot \widehat{w}_1)^{-1}$, $s_0 = s_0' \cdot \widehat{w}_1 \cdot \widehat{w}_2$, $\widehat{w}_3 = r_2 \cdot \widehat{w}_2$	1I, 4M
$7'$	$\underline{u' = x + u_0'}$ $\widetilde{w}_0 = s_0 \cdot u_{21}$, $\widetilde{w}_1 = s_1 u_{20}$ $u_0' = \widehat{w}_3 \cdot (s_0^2 - h_2 s_0 + s_1(2(v_{21} + \widetilde{w}_0) + \widetilde{w}_1 - h_1) - k_2) - u_{11}$	1S, 2M, 2C
$8'$	$\underline{v' = v_0'}$ $v_0' = h_0 - v_{20} - s_0 \cdot u_{20} + u_0' \cdot (v_{21} + \widetilde{w}_0 + \widetilde{w}_1 - h_1 + u_0' \cdot (s_0 + t_1 + (s_1 - h_3)u_0'))$	3M, 1C
Total	Special case	1I, 3S, 19M, 12C

Simplifications for Odd and Even Characteristic.

1. In Step 3, let $k_4 = 1$, $k_3 = -u_{21}$, $k_2 = f_4 + u_{21}^2 - u_{20}$ in odd characteristic. In even characteristic, let $k_4 = f_6$, $k_3 = f_6 u_{21}$, $k_2 = v_{21} + f_6(u_{21}^2 + u_{20})$. This reduces the operation count of Step 3 to 1S in odd characteristic and 1S, 2C in even characteristic.
2. In Step 4, let $\widehat{w}_0 = (s_1')^2 - r_2$ in odd characteristic. In even characteristic, let $\widehat{w}_0 = s_1' \cdot (s_1' + r) + f_6 r_2$. This reduces the operation count of Step 4 to 2S in odd characteristic and 1S, 1M, 1C in even characteristic.
3. In Step 6, we skip the calculation of $\widetilde{w}_2$ (since $h_2 = 0$). In Step 7, let

$$u_1' = \widehat{w}_3 \cdot ((s_0 + l_2) \cdot s_1 + u_{21}) - u_{11} \ ,$$
$$u_0' = \widehat{w}_3 \cdot ((2v_{21} + l_1) \cdot s_1 + l_2 \cdot s_0 - k_2) - u_{10} - u_{11} \cdot u_1'$$

   in odd characteristic. In even characteristic, let

$$u_1' = \widehat{w}_3 \cdot ((s_0 + l_2) \cdot s_1 + s_0 + k_3) + u_{11} \ ,$$
$$u_0' = \widehat{w}_3 \cdot ((h_1 + l_1) \cdot s_1 + l_2 \cdot s_0 + k_2) + u_{10} + u_{11} \cdot u_1' \ .$$

   This reduces the operation count of Step 7 to 6M in both characteristics.
4. In Step 8, let

$$\overline{w}_0 = -s_1, \quad \overline{w}_1 = u_1' \cdot \overline{w}_0 \ , \quad \overline{w}_2 = -l_2 - \overline{w}_1 \ , \quad \overline{w}_3 = u_0' \cdot \overline{w}_2 \ ,$$
$$v_1' = -v_{21} - l_1 - (u_0' + u_1') \cdot (\overline{w}_0 + \overline{w}_2) + \overline{w}_1 + \overline{w}_3 \ ,$$
$$v_0' = -v_{20} - l_0 - \overline{w}_3 \ .$$

   in odd characteristic. In even characteristic, let

$$\overline{w}_0 = s_1 + 1 \ , \quad \overline{w}_1 = u_1' \cdot \overline{w}_0 \ , \quad \overline{w}_2 = l_2 + \overline{w}_1 \ , \quad \overline{w}_3 = u_0' \cdot \overline{w}_2 \ ,$$
$$v_1' = h_1 + v_{21} + l_1 + (u_0' + u_1') \cdot (\overline{w}_0 + \overline{w}_2) + \overline{w}_1 + \overline{w}_3 \ ,$$
$$v_0' = h_0 + v_{20} + l_0 + \overline{w}_3 \ .$$

When $\deg(u_1) = \deg(u_2) = 2$, the total operation count for addition in the adapted basis for the general case of $\deg(u') = 2$ is 1I, 3S, 26M in odd characteristic and 1I, 2S, 27M, 3C in even characteristic.

Comments when $r = 0$. If $r = 0$ and $u_1 = u_2$ (which is equivalent to $z_0 = 0$ and $z_1 = 0$), then one of the following must be true:

1. $v_2 = v_1$ (which is equivalent to $w_0 = 0$ and $w_1 = 0$). In this case, the doubling formulas should be applied.
2. $v_2 = y + h - [y + v_1 \bmod u_1]$ in the reduced basis, or $v_2 = h - v_1$ in the adapted basis. In this case, D_1 and D_2 are involutions of each other. So adding the divisors results in the identity divisor ($u = 1$, $v = y + h$ in reduced basis or $u = 1$, $v = 0$ in adapted basis).
3. Otherwise, D_1 and D_2 contain one point in common, while the other point of the divisor are opposites of each other. In this case, we can determine x-coordinate of the common point by find the root of the difference of v_1 and v_2. Since $v_1 - v_2 = w_1 x + w_0$ (which are calculated in Step 2), the x-coordinate is $x_1 = -\frac{w_0}{w_1}$. The final result is found by first calculating $x_1 = -\frac{w_0}{w_1}$, then doubling the degree 1 divisor $[x - x_1, v_1(x_1)]$ in adapted basis, then changing the result into reduced basis if desired.

If $r = 0$ and $u_1 \neq u_2$ (which is equivalent to $z_1 \neq 0$), then u_1 and u_2 must have a common linear factor, which must be $u_1 - u_2 = z_1 x + z_0$ since u_1 and u_2 are both monic quadratics. Thus, the common root is $x_0 = -\frac{z_0}{z_1}$. From here, it depends on whether $v_1(x_0) = v_2(x_0)$, or equivalently if $(v_1 - v_2)(x_0) = w_1 \cdot (-\frac{z_0}{z_1}) + w_0 = 0$. Multiplying through by z_1, we have $s_1' = w_0 \cdot z_1 - w_1 \cdot z_0 = 0$. Hence, we have the following cases:

1. $r = 0$ and $s_1' = 0$. In this case, $D_1 = P_1 + P_2 - 2\infty$ and $D_2 = P_1 + P_3 - 2\infty$ contain copies of the same point. Let $x_1 = -\frac{z_0}{z_1}$, $x_2 = -u_{11} - x_1$, and $x_3 = -u_{21} - x_1$. Set $D_1' = 2[x - x_1, v_1(x_1)]$ using the (adapted) doubling formula for degree 1 divisors, and set $D_2' = [x - x_2, v_1(x_2)] + [x - x_3, v_2(x_3)]$ using the (adapted) addition formula for two degree 1 divisors. Finally, apply the addition formulas to $D' = D_1' + D_2'$ to obtain the final result, then changing to reduced basis if desired.
2. $r = 0$ and $s_1' \neq 0$. In this case, $D_1 = P_1 + P_2 - 2\infty$ and $D_2 = (-P_1) + P_3 - 2\infty$ contain negatives of a common point. As above, let $x_1 = -\frac{z_0}{z_1}$, $x_2 = -u_{11} - x_1$, and $x_3 = -u_{21} - x_1$. The final result is $D' = [x - x_2, v_1(x_2)] + [x - x_3, v_2(x_3)]$ using the (adapted) addition formula for two degree 1 divisors, then changing to reduced basis if desired.

6.1. Degenerate Divisor Addition.

Divisor Addition, Reduced Basis, Special Case ($\deg(u_1) = 1$, $\deg(u_2) = 2$). Let $[u_1, v_1]$ and $[u_2, v_2]$ be two reduced divisors. Assume u_1 is degree 1 and u_2 is degree 2. The addition and reduction step of Cantor's Algorithm is then given by

$$\begin{aligned} U_0 &= u_1 u_2 \\ V_0 &\equiv v_2 + s u_2 \\ V_1 &= h - V_0 + \left\lfloor \tfrac{V_0 + H(y)}{U_0} \right\rfloor \cdot U_0 \\ U_1 &= \tfrac{f + h \cdot V_1 - V_1^2}{U_0} \quad . \end{aligned}$$

In this case,

$$\left\lfloor \frac{V_0 + H(y)}{U_0} \right\rfloor = \left\lfloor \frac{(v_{23} x^3 + \dots) + (y_3 x^3 + \dots)}{(x^3 + \dots)} \right\rfloor = v_{23} + y_3$$

Hence,

$$V_1 = h - V_0 + (v_{23} + y_3)U_0$$

and, finally,

$$\begin{aligned} u' &= U_1 \text{ made monic } , \\ \text{Reduced: } v' &= H(y) + h - [(H(y) + h - V_1) \bmod u'] \text{ or} \\ \text{Adapted: } v' &\equiv V_1 \pmod{u'} \ . \end{aligned}$$

Addition, Reduced Basis, $\deg u_1 = 1$, $\deg u_2 = 2$		
Input	$u_1 = x + u_{10}$, $v_1 = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + v_{10}$ $u_2 = x^2 + u_{21}x + u_{20}$, $v_2 = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_{21}x + v_{20}$ Precomputed Constants in Table 1	
Output	$[u', v'] = [u_1, v_1] + [u_2, v_2]$	
Step	Expression	Operations
	Addition	
1	$\underline{r = u_2 \bmod u_1,\ s'_0}$ $t_0 = u_{10} \cdot u_{21}$, $r = u_{20} - t_0 + u_{10}^2$ $z_2 = u_{10} + u_{21}$, $z_1 = t_0 + u_{20}$, $z_0 = u_{10} \cdot u_{20}$ $s'_0 = v_{10} - v_{20} - u_{10} \cdot (c_1 - v_{21})$. If $r = 0$, see Comments below.	1S, 3M
	Reduction	
2	$\underline{\widehat{w}_1 = U'_{12}}$ $\widehat{w}_0 = c_2 - c_3 z_2$, $\widehat{w}_1 = s'_0 + r \cdot \widehat{w}_0$ If $\widehat{w}_1 = 0$, see Special case.	1M, 1C
3	$\underline{s_0,\ \widehat{w}_3 = U_{12}^{-1}}$ $\widehat{w}_2 = (r \cdot \widehat{w}_1)^{-1}$, $s_0 = \widehat{w}_1 \cdot \widehat{w}_2 \cdot s'_0$, $\widehat{w}_3 = r^2 \cdot \widehat{w}_2$	1I, 1S, 4M
4	$\underline{V_1 = V_{13}x^3 + V_{12}x^2 + V_{11}x + V_{10}}$ $\widetilde{w}_2 = s_0 + \widehat{w}_0$, $\widetilde{w}_1 = y_1 + v_{21} + s_0 \cdot u_{21} - c_3 z_1$ $\widetilde{w}_0 = y_0 + v_{20} + s_0 \cdot u_{20} - c_3 z_0$	2M, 2C
5	$\underline{u' = x^2 + u'_1 x + u'_0}$ $u'_1 = \widehat{w}_3 \cdot (d_3((c_2 - \widetilde{w}_2) \cdot \widetilde{w}_2) + \widetilde{w}_1) - z_2$ $u'_0 = \widehat{w}_3 \cdot (d_3((c_1 + y_1 - 2\widetilde{w}_1) \cdot \widetilde{w}_2 + c_2\widetilde{w}_1) + \widetilde{w}_0) -$ $z_1 - z_2 \cdot u'_1$	5M, 3C
6	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v'_1 x + v'_0}$ $v'_1 = c_1 - \widetilde{w}_1 + \widetilde{w}_2 \cdot u'_1$, $v'_0 = c_0 - \widetilde{w}_0 + \widetilde{w}_2 \cdot u'_0$	2M
Total		1I, 2S, 17M, 6C

Special case: $\widehat{w}_1 = 0$ $(\deg(u') = 1)$		
$3'$	$\underline{s_0,\ \widehat{w}_3 = z_1^{-1}}$ $\widehat{w}_0 = v_{21} + y_1 - c_3 z_1$, $\widehat{w}_1 = s'_0 \cdot u_{21} + r \cdot \widehat{w}_0$ $\widehat{w}_2 = (r \cdot \widehat{w}_1)^{-1}$, $s_0 = \widehat{w}_1 \cdot \widehat{w}_2 \cdot s'_0$, $\widehat{w}_3 = r^2 \cdot \widehat{w}_2$	1I, 1S, 6M, 1C
$4'$	$\underline{u' = x + u'_0}$ $\widetilde{w}_1 = \widehat{w}_0 + s_0 \cdot u_{21}$ $\widetilde{w}_0 = y_0 + v_{20} + s_0 \cdot u_{20} - c_3 z_0$ $u'_0 = \widehat{w}_3 \cdot (d_3 c_2 \widetilde{w}_1 + \widetilde{w}_0) - z_2$	3M, 3C
$5'$	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v'_1 x + v'_0}$ $v'_1 = c_1$, $v'_0 = c_0 - \widetilde{w}_0 + \widetilde{w}_1 \cdot u'_0$	1M
Total	Special Case	1I, 2S, 14M, 5C

Simplifications for Odd and Even Characteristic.

1. In Step 2, skip the calculation of $\widehat{w}_0$. Let $\widehat{w}_1 = s_0' - 2r \cdot z_2$ in odd characteristic. In even characteristic, let $\widehat{w}_1 = s_0' + r \cdot z_2$.
2. In Step 4, let
$$\begin{aligned} \widetilde{w}_2 &= s_0 - 2z_2 \ , \\ \widetilde{w}_1 &= y_1 + v_{21} + s_0 \cdot u_{21} - 2z_1 \ , \\ \widetilde{w}_0 &= y_0 + v_{20} + s_0 \cdot u_{20} - 2z_0 \end{aligned}$$
in odd characteristic. In even characteristic, let
$$\begin{aligned} \widetilde{w}_2 &= s_0 + z_2 \ , \\ \widetilde{w}_1 &= y_1 + v_{21} + s_0 \cdot u_{21} + z_1 \ , \\ \widetilde{w}_0 &= y_0 + v_{20} + s_0 \cdot u_{20} + z_0 \ . \end{aligned}$$
3. In Step 5, let
$$\begin{aligned} u_1' &= \widehat{w}_3 \cdot (-\tfrac{1}{2}\widetilde{w}_2^2 + \widetilde{w}_1) - z_2 \\ u_0' &= \widehat{w}_3 \cdot ((f_4 - \widetilde{w}_1) \cdot \widetilde{w}_2 + \widetilde{w}_0) - z_1 - z_2 \cdot u_1' \end{aligned}$$
in odd characteristic. In even characteristic, let
$$\begin{aligned} u_1' &= \widehat{w}_3 \cdot (\widetilde{w}_2^2 + \widetilde{w}_1) + z_2 \\ u_0' &= \widehat{w}_3 \cdot (h_1\widetilde{w}_2 + \widetilde{w}_0) + z_1 + z_2 \cdot u_1' \ . \end{aligned}$$
This reduces the operation count of Step 6 to 1S, 3M in odd characteristic and 1S, 2M, 1C in even characteristic. For special values of h_1 (such as 0 or 1), the constant multiplication can be removed for an operation count of 1S, 2M in even characteristic.

When $\deg(u_1) = 1$, $\deg(u_2) = 2$, the total operation count for addition in the reduced basis for the general case of $\deg(u') = 2$ is 1I, 3S, 16M in odd characteristic and 1I, 3S, 15M, 1C in even characteristic.

Divisor Addition, Adapted Basis, Special Case ($\deg(u_1) = 1$, $\deg(u_2) = 2$*).*

Addition, Adapted Basis, $\deg u_1 = 1$, $\deg u_2 = 2$		
Input	$u_1 = x + u_{10}$, $v_1 = v_{10}$ $u_2 = x^2 + u_{21}x + u_{20}$, $v_2 = v_{21}x + v_{20}$ Precomputed Constants in Table 1	
Output	$[u', v'] = [u_1, v_1] + [u_2, v_2]$	
Step	Expression	Operations
	Addition	
1	$\underline{r = u_2 \bmod u_1,\ s'_0}$ $t_0 = u_{10} \cdot u_{21}$, $r = u_{20} - w_0 + u_{10}^2$ $z_2 = u_{10} + u_{21}$, $z_1 = t_0 + u_{20}$, $z_0 = u_{10} \cdot u_{20}$ $s'_0 = v_{10} - v_{20} + u_{10} \cdot v_{21}$. If $r = 0$, see Comments below.	1S, 3M
	Reduction	
2	$\underline{\widehat{w}_1 = U'_{12}}$ $\widehat{w}_0 = y_2 - y_3 z_2$, $\widehat{w}_1 = s'_0 + r \cdot \widehat{w}_0$ If $\widehat{w}_1 = 0$, see Special case.	1M, 1C
3	$\underline{s_0,\ \widehat{w}_3 = z^{-1}}$ $\widehat{w}_2 = (r \cdot \widehat{w}_1)^{-1}$, $s_0 = \widehat{w}_1 \cdot \widehat{w}_2 \cdot s'_0$, $\widehat{w}_3 = r^2 \cdot \widehat{w}_2$	1I, 1S, 4M
4	$\underline{V_1 = V_{13}x^3 + V_{12}x^2 + V_{11}x + V_{10}}$ $\widetilde{w}_2 = s_0 + \widehat{w}_0$, $\widetilde{w}_1 = y_1 + v_{21} + s_0 \cdot u_{21} - y_3 z_1$ $\widetilde{w}_0 = y_0 + v_{20} + s_0 \cdot u_{20} - y_3 z_2$	2M, 2C
5	$\underline{u' = x^2 + u'_1 x + u'_0}$ $u'_1 = \widehat{w}_3 \cdot (d_3((c_2 - \widetilde{w}_2) \cdot \widetilde{w}_2) + \widetilde{w}_1) - z_2$ $u'_0 = \widehat{w}_3 \cdot (d_3((h_1 - 2\widetilde{w}_1) \cdot \widetilde{w}_2 + c_2 \widetilde{w}_1) + \widetilde{w}_0) - z_1 - z_2 \cdot u'_1$	5M, 3C
6	$\underline{v' = v'_1 x + v'_0}$ $\overline{w}_4 = (y_3 + h_3) u'_1$, $\overline{w}_3 = y_2 + h_2 - \widetilde{w}_2 - \overline{w}_4$ $\overline{w}_2 = \overline{w}_3 \cdot u'_0$ $v'_1 = c_1 - \widetilde{w}_1 - (y_3 + h_3 + \overline{w}_3) \cdot (u'_0 + u'_1) + \overline{w}_2 + \overline{w}_4$, $v'_0 = c_0 - \widetilde{w}_0 - \overline{w}_2$	2M, 1C
Total		1I, 2S, 17M, 7C

Special case: $\widehat{w}_1 = 0$ $(\deg(u') = 1)$		
$3'$	$\underline{\widehat{w}_3 = U_{11}^{-1}}$ $\widehat{w}_0 = y_1 + v_{21} - y_3 z_1$, $\widehat{w}_1 = s'_0 \cdot u_{21} + r \cdot \widehat{w}_0$ $\widehat{w}_2 = (r \cdot \widehat{w}_1)^{-1}$, $s_0 = \widehat{w}_1 \cdot \widehat{w}_2 \cdot s'_0$, $\widehat{w}_3 = r^2 \cdot \widehat{w}_2$	1I, 1S, 6M, 1C
$4'$	$\underline{u' = x + u'_0}$ $\widetilde{w}_1 = \widehat{w}_0 + s_0 \cdot u_{21}$, $\widetilde{w}_0 = y_0 + v_{20} + s_0 \cdot u_{20} - y_3 z_0$ $u'_0 = \widehat{w}_3 \cdot (d_3 c_2 \widetilde{w}_1 + \widetilde{w}_0) - z_2$	4M, 3C
$5'$	$\underline{v' = v'_0}$ $v'_0 = c_0 - \widetilde{w}_0 - u'_0 \cdot (c_1 - \widetilde{w}_1 - (y_2 + h_2) u'_0 + (y_3 + h_3) u'^2_0)$	1S, 1M, 2C
Total	Special case	1I, 2S, 15M, 7C

Simplifications for Odd and Even Characteristic.

1. In Step 2, skip the calculation of $\widehat{w}_0$. Let $\widehat{w}_1 = s'_0 - r \cdot (u_{10} + u_{21})$ in odd characteristic. In even characteristic, let $\widehat{w}_1 = s'_0 + y_3 r \cdot (u_{10} + u_{21})$.

2. In Step 4, let
$$\begin{aligned}\widetilde{w}_2 &= s_0 - 2(u_{10} + u_{21}) \ , \\ \widetilde{w}_1 &= \tfrac{1}{2} f_4 + v_{21} + s_0 \cdot u_{21} - (w_0 + u_{20}) \ , \\ \widetilde{w}_0 &= \tfrac{1}{2} f_3 + v_{20} + s_0 \cdot u_{20} - (u_{10} \cdot u_{20})\end{aligned}$$
in odd characteristic. In even characteristic, let
$$\begin{aligned}\widetilde{w}_2 &= s_0 + (u_{10} + u_{21}) \ , \\ \widetilde{w}_1 &= y_1 + v_{21} + s_0 \cdot u_{21} + y_3(w_0 + u_{20}) \ , \\ \widetilde{w}_0 &= y_0 + v_{20} + s_0 \cdot u_{20} + y_3(u_{10} \cdot u_{20}) \ .\end{aligned}$$
3. In Step 5, let
$$\begin{aligned}u_1' &= \widehat{w}_3 \cdot (-\tfrac{1}{2}\widetilde{w}_2^2 + \widetilde{w}_1) - z_2 \\ u_0' &= \widehat{w}_3 \cdot ((f_4 - \widetilde{w}_1) \cdot \widetilde{w}_2 + \widetilde{w}_0) - z_1 - z_2 \cdot u_1'\end{aligned}$$
in odd characteristic. In even characteristic, let
$$\begin{aligned}u_1' &= \widehat{w}_3 \cdot (\widetilde{w}_2^2 + \widetilde{w}_1) + z_2 \\ u_0' &= \widehat{w}_3 \cdot (h_1\widetilde{w}_2 + \widetilde{w}_0) + z_1 + z_2 \cdot u_1'\end{aligned}$$
This reduces the operation count of Step 6 to 1S, 4M in odd characteristic and 1S, 3M, 1C in even characteristic. For special values of h_1 (such as 0 or 1), the constant multiplication can be removed for an operation count of 1S, 3M in even characteristic.
4. In Step 6, replace the calculation of $\overline{w}_4$, $\overline{w}_3$, and $\overline{w}_2$ with
$$\begin{aligned}\overline{w}_3 &= \widetilde{w}_2 + u_1' \ , \\ v_1' &= c_1 - \widetilde{w}_1 - u_0' + \overline{w}_3 \cdot u_1' \\ v_0' &= c_0 - \widetilde{w}_0 + \overline{w}_3 \cdot u_0'\end{aligned}$$
in odd characteristic. In even characteristic, let
$$\begin{aligned}\overline{w}_3 &= \widetilde{w}_2 + y_3 u_1' \ , \\ v_1' &= c_1 + \widetilde{w}_1 + y_3 u_0' + \overline{w}_3 \cdot u_1' \ , \\ v_0' &= c_0 + \widetilde{w}_0 + \overline{w}_3 \cdot u_0' \ .\end{aligned}$$

When $\deg(u_1) = 1$, $\deg(u_2) = 2$, the total operation count for addition in the adapted basis for the general case of $\deg(u') = 2$ is 1I, 3S, 17M in odd characteristic and 1I, 3S, 16M, 3C in even characteristic.

Comments when $r = 0$. Since $r = u_{20} - u_{10} \cdot u_{21} + u_{10}^2$ represents plugging in the one root of $u_1 = x + u_{10}$ into u_2, $r = 0$ if and only if u_1 divides u_2. In terms of divisors, $D_1 = P_1 - \infty$ and either $D_2 = P_1 + P_2 - 2\infty$ or $D_2 = -P_1 + P_2 - 2\infty$. These two cases can be distinguished by $v_1(-u_{10}) = v_2(-u_{10})$, or equivalently $s_0' = (v_1 - v_2)(-u_{10}) = 0$. In either case, $u_2 = x^2 + u_{21}x + u_{20} = (x - x_1)(x - x_2)$, so the x-coordinate of the second point of D_2 can be expressed as $x_2 = u_{10} - u_{21}$.

1. If $s_0' \neq 0$, then $D_1 = P_1 - \infty$ and $D_2 = -P_1 + P_2 - 2\infty$, so $D' = P_2 - \infty$ can be expressed as $D' = [x + (u_{21} - u_{10}), (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + (v_{20} + (u_{21} - u_{10}) \cdot (c_1 - v_{21}))]$ in the reduced basis, and $D' = [x + (u_{21} - u_{10}), v_{20} - (u_{21} - u_{10}) \cdot v_{21}]$ in the adapted basis.
2. If $s_0' = 0$, then $D_1 = P_1 - \infty$ and $D_2 = P_1 + P_2 - 2\infty$. There are two subcases, depending on whether $P_2 = P_1$, or equivalently $u_{21} = 2u_{10}$.

(a) If $u_{21} \neq 2u_{10}$, then compute $D_1' = 2D_1$ using the degree 1 doubling formula, and let $D_2' = [x + (u_{21} - u_{10}), (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + (v_{20} + (u_{21} - u_{10}) \cdot (c_1 - v_{21}))]$ in the reduced basis, and $D_2' = [x + (u_{21} - u_{10}), v_{20} - (u_{21} - u_{10}) \cdot v_{21}]$ in adapted basis. The end result will be $D' = D_1' + D_2'$.
(b) If $u_{21} = 2u_{10}$, then compute $D_1' = -D_1$, $D_2' = 2D_2$, and the final result is $D' = D_1' + D_2'$.

Divisor Addition, Special Case ($\deg(u_1) = 1$, $\deg(u_2) = 1$*).* If $u_1 = u_2$, there are two cases. If $v_1 \neq v_2$, then the two divisors are involutions of each other, and the output is the identity divisor ($u = 1$, $v = y + h$ in reduced basis or $u = 1$, $v = 0$ in adapted basis). If $v_1 = v_2$, then we are in the doubling case, which will be provided later.

If $u_1 \neq u_2$, then Cantor's Algorithm can be performed without the reduction step:

$$u' = u_1 \cdot u_2$$
$$v' = v_2 + u_2 \cdot \left[(v_1 - v_2) \cdot u_2^{-1} \bmod u_1\right]$$

Performing this explicitly provides the following formulas:

Addition, Reduced Basis, $\deg u_1 = 1$, $\deg u_2 = 1$		
Input	$u_1 = x + u_{10}$, $v_1 = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + v_{10}$ $u_2 = x + u_{20}$, $v_2 = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + v_{20}$	
Output	$[u', v'] = [u_1, v_1] + [u_2, v_2]$	
Step	Expression	Operations
1	$\underline{u' = x^2 + u_1'x + u_0'}$ If $u_{10} = u_{20}$, see Comments below. $u_1' = u_{10} + u_{20}$, $u_0' = u_{10} \cdot u_{20}$	1M
2	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_1'x + v_0'}$ $w_0 = (v_{10} - v_{20}) \cdot (u_{20} - u_{10})^{-1}$ $v_1' = c_1 + w_0$, $v_0' = v_{20} + u_{20} \cdot w_1$	1I, 2M
Total		1I, 3M

Addition, Adapted Basis, $\deg u_1 = 1$, $\deg u_2 = 1$		
Input	$u_1 = x + u_{10}$, $v_1 = v_{10}$ $u_2 = x + u_{20}$, $v_2 = v_{20}$	
Output	$[u', v'] = [u_1, v_1] + [u_2, v_2]$	
Step	Expression	Operations
1	$\underline{u' = x^2 + u_1'x + u_0'}$ If $u_{10} = u_{20}$, see Comments below. $u_1' = u_{10} + u_{20}$, $u_0' = u_{10} \cdot u_{20}$	1M
2	$\underline{v' = v_1'x + v_0'}$ $v_1' = (v_{10} - v_{20}) \cdot (u_{20} - u_{10})^{-1}$, $v_0' = v_{20} + u_{20} \cdot v_1'$	1I, 2M
Total		1I, 3M

7. Doubling Formulas

Let $[u, v]$ be a reduced representative of a divisor class with $\deg u = 2$. The divisor $[u', v'] = 2[u, v]$ can be computed using the following formulas:

$$\begin{aligned}
&\widetilde{v} \equiv 2v - h \pmod{u}, && inv = r(\widetilde{v})^{-1} \pmod{u},\\
&k = \tfrac{f+hv-v^2}{u}, && k' \equiv k \pmod{u},\\
&s' \equiv k' \cdot inv \pmod{u}, && s = \tfrac{1}{r} \cdot s',\\
&\widetilde{u} = s^2 + \tfrac{(2v-h)\cdot s - k}{u}, && u' = \widetilde{u} \text{ made monic},
\end{aligned}$$

$$\begin{aligned}
&\text{Adapted: } v' \equiv h - v - su \pmod{u'} \text{ or}\\
&\text{Reduced: } v' = H(y) + h - [(H(y) + v + su) \bmod u'] \ .
\end{aligned}$$

To see that these formulas are correct, observe that Algorithm 2.1 results in $[U_1, V_1]$ such that

$$\begin{aligned}
&U_0 = u^2,\\
&V_0 \equiv v \pmod{u} \quad (V_0 = v + su \text{ for some } s) \ ,\\
&V_1 = h - V_0 + \left\lfloor \tfrac{V_0 + H(y)}{U_0} \right\rfloor U_0,\\
&U_1 = \tfrac{f + hV_1 - V_1^2}{U_0} \ .
\end{aligned}$$

Here, s is chosen such that U_0 divides $f + hV_0 - V_0^2$. Again, $\left\lfloor \frac{V_0+H(y)}{U_0} \right\rfloor$ is zero since U_0 has degree 4 and $V_0 + H(y)$ has degree 3. Hence, $V_1 = h - V_0 = h - v - su$ and

$$\begin{aligned}
U_1 &= \tfrac{f + h(h-v-su) - (h-v-su)^2}{u^2}\\
&= \tfrac{f + hv - v^2 + (h-2v)su - s^2u^2}{u^2}\\
&= \tfrac{1}{u}\left(\tfrac{f+hv-v^2}{u} + (h-2v)s\right) - s^2\\
&= \tfrac{k + (h-2v)s}{u} - s^2
\end{aligned}$$

where the division in $k = \frac{f+hv-v^2}{u}$ is exact. By choosing $s \equiv -k \cdot (h-2v)^{-1} \pmod{u}$, we have

$$k + (h-2v)s \equiv k - (h-2v) \cdot k \cdot (h-2v)^{-1} \equiv 0 \pmod{u}$$

so that the division of $k + (h-2v)s$ by u is exact.

The final step is to reduce $[U_1, V_1]$ into either adapted or reduced basis. For the adapted basis, $[u', v']$ is found by setting u' to U_1 made monic, and $v' \equiv V_1 \pmod{u'}$. For the reduced basis, $[u', v']$ is found by setting $u' = U_1$ made monic, and $v' = H(y) + h - [(H(y) + h - V_1) \bmod u'] = H(y) + h - [(H(y) + v_2 + l) \bmod u']$. For the doubling formulas, we compute the leading coefficients $\widetilde{u} = -U_1 = s^2 + \frac{(2v-h)s-k}{u}$ compute the inverse of the first nonzero coefficient, then compute $u' = \widetilde{u}$ made monic. Note that since we have already computed $k' \equiv k \pmod{u}$ and $\widetilde{v} = (2v - h) \pmod{u}$, several of the coefficients of k' and $\widetilde{v}$ appear in the coefficients of $\widetilde{u}$. This observation considerably simplifies the formulas.

Doubling, Reduced Basis, $\deg u = 2$		
Input	$u = x^2 + u_1x + u_0, v = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v_1x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = 2[u, v]$	
Step	Expression	Operations
	Addition	
1	$\underline{\widetilde{v} = \widetilde{v}_1x + \widetilde{v}_0}$ $t_1 = c_2 - c_3u_1$, $t_2 = u_1^2$, $\widetilde{v}_1 = 2v_1 - h_1 - c_2u_1 + c_3(t_2 - u_0)$, $\widetilde{v}_0 = 2v_0 - h_0 - u_0 \cdot t_1$	1S, 1M, 3C
2	$\underline{inv = z_1x + z_2}$ $z_1 = \widetilde{v}_1$, $z_2 = u_1 \cdot \widetilde{v}_1 - \widetilde{v}_0$, $z_3 = u_0 \cdot \widetilde{v}_1$, $r = \widetilde{v}_0 \cdot z_2 - \widetilde{v}_1 \cdot z_3$ If $r = 0$, see Comments below.	4M
3	$\underline{k' = k'_1x + k'_0}$ $w_0 = c_0 - v_0$, $w_1 = c_1 - v_1$, $w_2 = c_2w_0$, $k'_2 = c_3w_1$ $k'_1 = (c_2 + c_3)(w_0 + w_1) - w_2 - k'_2 - 2k'_2 \cdot u_1$ $k'_0 = d_2 + w_2 + (h_1 - v_1) \cdot v_1 - k'_1 \cdot u_1 - k'_2 \cdot (t_2 + 2u_0)$	4M, 3C
4	$\underline{s' = s'_1x + s'_0}$ $s'_1 = \widetilde{v}_1 \cdot k'_0 - \widetilde{v}_0 \cdot k'_1$, $s'_0 = z_2 \cdot k'_0 - z_3 \cdot k'_1$.	4M
	Reduction	
5	$\underline{\widehat{w}_0 = \widetilde{u}_2}$ $r_2 = r^2$, $\widehat{w}_0 = (s'_1 + c_3r) \cdot s'_1$. If $\widehat{w}_0 = 0$, see Special case.	1S, 1M, 1C
6	$\underline{s = \frac{1}{r}s' = s_1x + s_0, \widetilde{u}_2^{-1}}$ $\widehat{w}_1 = (r \cdot \widehat{w}_0)^{-1}$, $\widehat{w}_2 = \widehat{w}_0 \cdot \widehat{w}_1$, $s_1 = \widehat{w}_2 \cdot s'_1$, $s_0 = \widehat{w}_2 \cdot s'_0$, $\widehat{w}_3 = r \cdot r_2 \cdot \widehat{w}_1$	1I, 6M
7	$\underline{u' = x^2 + u'_1x + u'_0}$ $u'_1 = \widehat{w}_3 \cdot ((2s_0 + t_1) \cdot s_1 + c_3s_0)$ $u'_0 = \widehat{w}_3 \cdot ((s_0 + t_1) \cdot s_0 + \widetilde{v}_1 \cdot s_1 - k'_2)$	5M, 1C
8	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v'_1x + v'_0}$ $\overline{z}_0 = u'_0 - u_0$, $\overline{z}_1 = u'_1 - u_1$, $\overline{w}_0 = s_0 \cdot \overline{z}_0$, $\overline{w}_1 = s_1 \cdot \overline{z}_1$, $\overline{w}_2 = -\overline{w}_1 + c_2 - c_3u'_1$ $v'_1 = h_1 - v_1 + (s_0 + s_1) \cdot (\overline{z}_0 + \overline{z}_1) - \overline{w}_0 - \overline{w}_1 +$ $c_3u'_0 + \overline{w}_2 \cdot u'_1$ $v'_0 = h_0 - v_0 + \overline{w}_0 + \overline{w}_2 \cdot u'_0$	5M, 2C
Total		1I, 2S, 30M, 10C

Special case: $\widehat{w}_0 = 0$ $(\deg(u') = 1)$		
$6'$	$\underline{\widehat{w}_1 = \widetilde{u}_1}$ If $s'_1 = 0$, then $s_1 = 0$. Otherwise, $s_1 = -c_3$. $\widehat{w}_1 = (2s_1 + c_3)s'_0 + s'_1 \cdot t_1$	1M, 1C
$7'$	$\underline{s = \frac{1}{r}s' = s_1 x + s_0,\ \widehat{w}_3 = \widetilde{u}_1^{-1}}$ $\widehat{w}_2 = (r \cdot \widehat{w}_1)^{-1},\ s_0 = s'_0 \cdot \widehat{w}_1 \cdot \widehat{w}_2,\ \widehat{w}_3 = r_2 \cdot \widehat{w}_2$	1I, 4M
$8'$	$\underline{u' = x + u'_0}$ $u'_0 = \widehat{w}_3 \cdot ((s_0 + t_1) \cdot s_0 + s_1\widetilde{v}_1 - k_2)$	2M, 1C
$9'$	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v'_1 x + v'_0}$ $\widetilde{w}_0 = s_0 \cdot u_0,\ \widetilde{w}_1 = s_1 u_1,\ v'_1 = c_1$ $v'_0 = h_0 - v_0 - \widetilde{w}_0 + u'_0 \cdot (y_1 + v_1 + (s_0 + s_1) \cdot (u_0 +$ $u_1) - \widetilde{w}_0 - \widetilde{w}_1 - u'_0 \cdot (c_2 + s_0 + \widetilde{w}_1 - (c_3 + s_1)u'_0))$	4M, 2C
Total	Special case	1I, 2S, 25M, 11C

Simplifications for Odd and Even Characteristic.

1. In Step 1, we skip the computation of t_1. Let
$$\widetilde{v}_1 = 2(v_1 + t_2 - u_0)\ ,$$
$$\widetilde{v}_0 = 2(v_0 + u_0 \cdot u_1)$$
in odd characteristic. Let
$$\widetilde{v}_1 = h_1 + t_2 + u_0\ ,$$
$$\widetilde{v}_0 = h_0 + u_0 \cdot u_1$$
in even characteristic. This reduces the operation count of Step 1 to 1S, 1M in both odd and even characteristic.
2. In Step 3, let
$$k'_2 = f_4 - 2v_1\ ,$$
$$k'_1 = f_3 - 2v_0 - 2k'_2 \cdot u_1\ ,$$
$$k'_0 = f_2 - v_1^2 - k'_1 \cdot u_1 - k'_2 \cdot (t_2 + 2u_0)$$
in odd characteristic. Let
$$k'_2 = c_1 + v_1\ ,$$
$$k'_1 = c_0 + v_0\ ,$$
$$k'_0 = f_2 + (h_1 + v_1) \cdot v_1 + k'_1 \cdot u_1 + k'_2 \cdot t_2$$
in even characteristic. This reduces the operation count of Step 3 to 1S, 3M in odd characteristic and 3M in even characteristic. For special values of h_1 (such as 0 or 1), one multiplication can be changed into a squaring for an operation count of 1S, 2M in even characteristic.
3. In Step 5, let $\widehat{w}_0 = (s'_1 + r)^2 - r_2$ in odd characteristic. In even characteristic, let $\widehat{w}_0 = (s'_1 + r) \cdot s'_1$. This reduces the operation count of Step 5 to 2S in odd characteristic and 1S, 1M in even characteristic.
4. In Step 7, let
$$u'_1 = 2\widehat{w}_3 \cdot ((s_0 - u_1) \cdot s_1 + s_0)\ ,$$
$$u'_0 = \widehat{w}_3 \cdot (-f_4 + 2v_1 + (s_0 - 2u_1) \cdot s_0 + \widetilde{v}_1 \cdot s_1)$$

in odd characteristic, and

$$u'_1 = \widehat{w}_3 \cdot (s_0 + u_1 \cdot s_1) \ ,$$
$$u'_0 = \widehat{w}_3 \cdot (v_1 + h_1 + \widetilde{v}_1 \cdot s_1 + (u_1 + s_0) \cdot s_0)$$

in even characteristic. This reduces the operation count of Step 7 to 5M in both odd and even characteristic.

5. In Step 8, skip the calculation of $\overline{w}_2$. Let

$$v'_1 = 2u'_0 - v_1 + (s_0 + s_1) \cdot (\overline{z}_0 + \overline{z}_1) - \overline{w}_0 - \overline{w}_1 - u'_1 \cdot (2u'_1 + \overline{w}_1) \ ,$$
$$v'_0 = \overline{w}_0 - v_0 - u'_0 \cdot (2u'_1 + \overline{w}_1)$$

in odd characteristic, and

$$v'_1 = h_1 + v_1 + u'_0 + (s_0 + s_1) \cdot (z_0 + z_1) + \overline{w}_0 + \overline{w}_1 + u'_1 \cdot (u'_1 + \overline{w}_1) \ ,$$
$$v'_0 = h_0 + v_0 + \overline{w}_0 + (u'_1 + \overline{w}_1) \cdot u'_0$$

in even characteristic. This reduces the operation count of Step 8 to 5M in both odd and even characteristic.

When $\deg(u) = 2$, the total operation count for doubling in the reduced basis for the general case of $\deg(u') = 2$ is 1I, 4S, 28M in odd characteristic and 1I, 2S, 29M in even characteristic.

Doubling, Adapted Basis, $\deg u = 2$		
Input	$u = x^2 + u_1x + u_0, v = v_1x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = 2[u, v]$	
Step	Expression	Operations
	Addition	
1	$\underline{\widetilde{v} = \widetilde{v}_1x + \widetilde{v}_0}$ $t_1 = h_2 - h_3u_1$, $t_2 = u_1^2$ $\widetilde{v}_1 = 2v_1 - h_1 + h_2u_1 - h_3(t_2 - u_0)$, $\widetilde{v}_0 = 2v_0 - h_0 + u_0 \cdot t_1$	1S, 1M, 3C
2	$\underline{inv = z_1x + z_2}$ $z_1 = \widetilde{v}_1$, $z_2 = u_1 \cdot \widetilde{v}_1 - \widetilde{v}_0$, $z_3 = u_0 \cdot \widetilde{v}_1$, $r = \widetilde{v}_0 \cdot z_2 - \widetilde{v}_1 \cdot z_3$ If $r = 0$, see Comments below.	4M
3	$\underline{k' = k'_1x + k'_0}$: $w_0 = h_2v_0$, $w_1 = h_3v_1$, $k'_3 = f_5 - 2f_6u_1$ $k'_2 = f_4 + w_1 - 2f_5u_1 + f_6(3t_2 - 2u_0)$ $k'_1 = f_3 + (h_2 + h_3)(v_0 + v_1) - w_0 - w_1 - f_5(t_2 + 2u_0) + 2(f_6(t_2 + u_0) - k'_2) \cdot u_1$ $k'_0 = f_2 + w_0 + (h_1 - v_1) \cdot v_1 - (k'_1 + 2f_5u_0) \cdot u_1 - (k'_2 + 4f_6u_0) \cdot (t_2 + 2u_0) - 9f_6u_0^2$	1S, 4M, 11C
4	$\underline{s' = s'_1x + s'_0}$ $s'_1 = \widetilde{v}_1 \cdot k'_0 - \widetilde{v}_0 \cdot k'_1$, $s'_0 = z_2 \cdot k'_0 - z_3 \cdot k'_1$.	4M
	Reduction	
5	$\underline{\widehat{w}_0 = \widetilde{u}_2}$ $r_2 = r^2$, $\widehat{w}_0 = (s'_1 - h_3r) \cdot s'_1 - f_6r_2$. If $\widehat{w}_0 = 0$, see Special case.	1S, 1M, 2C
6	$\underline{s = \frac{1}{r}s' = s_1x + s_0, \widetilde{u}_2^{-1}}$ $\widehat{w}_1 = (r \cdot \widehat{w}_0)^{-1}$, $\widehat{w}_2 = \widehat{w}_0 \cdot \widehat{w}_1$, $s_1 = \widehat{w}_2 \cdot s'_1$, $s_0 = \widehat{w}_2 \cdot s'_0$, $\widehat{w}_3 = r \cdot r_2 \cdot \widehat{w}_1$	1I, 6M
7	$\underline{u' = x^2 + u'_1x + u'_0}$ $u'_1 = \widehat{w}_3 \cdot ((2s_0 - t_1) \cdot s_1 - h_3s_0 - k'_3)$ $u'_0 = \widehat{w}_3 \cdot ((s_0 - t_1) \cdot s_0 + \widetilde{v}_1 \cdot s_1 - k'_2)$	5M, 1C
8	$\underline{v' = v'_1x + v'_0}$ $\overline{z}_0 = u'_0 - u_0$, $\overline{z}_1 = u'_1 - u_1$, $\overline{w}_0 = \overline{z}_0 \cdot s_0$, $\overline{w}_1 = \overline{z}_1 \cdot s_1$, $\overline{w}_2 = -\overline{w}_1 - h_2 + h_3u'_1$ $v'_1 = h_1 - v_1 + (s_0 + s_1) \cdot (\overline{z}_0 + \overline{z}_1) - \overline{w}_0 - \overline{w}_1 - h_3u'_0 + \overline{w}_2 \cdot u'_1$ $v'_0 = h_0 - v_0 + \overline{w}_0 + \overline{w}_2 \cdot u'_0$	5M, 2C
Total		1I, 3S, 30M, 19C

Special case: $\widehat{w}_0 = 0$ $(\deg(u') = 1)$		
$6'$	$\underline{\widehat{w}_1 = \widetilde{u}_1}$ If $s'_1 = -y_3 r$, then $s_1 = -y_3$. Otherwise, $s_1 = y_3 + h_3$. $\widehat{w}_1 = (2s_1 - h_3)s'_0 - (s_1 t_1 + k'_3) \cdot r$	1M, 3C
$7'$	$\underline{s = \frac{1}{r}s' = s_1 x + s_0,\ \widehat{w}_2 = \widetilde{u}_1^{-1}}$ $\widehat{w}_1 = (r \cdot \widehat{w}_0)^{-1},\ s_0 = s'_0 \cdot \widehat{w}_0 \cdot \widehat{w}_1,\ \widehat{w}_2 = r_2 \cdot \widehat{w}_1$	1I, 4M
$8'$	$\underline{u' = x + u'_0}$ $u'_0 = \widehat{w}_2 \cdot ((s_0 - t_1) \cdot s_0 + s_1 \widetilde{v}_1 - k'_2)$	2M, 1C
$9'$	$\underline{v' = v'_1 x + v'_0}$ $\widetilde{w}_0 = s_0 \cdot u_0,\ \widetilde{w}_1 = s_1 u_1,\ v'_1 = 0$ $v'_0 = h_0 - v_0 - \widetilde{w}_0 + u'_0 \cdot (v_1 - h_1 + (s_0 + s_1) \cdot (u_0 +$ $u_1) - \widetilde{w}_0 - \widetilde{w}_1 + u'_0 \cdot (s_0 + \widetilde{w}_1 - h_2 + (s_1 - h_3)u'_0))$	4M, 2C
Total	Special case	1I, 3S, 25M, 22C

Simplifications for Odd and Even Characteristic.

1. In Step 1, let $\widetilde{v}_1 = 2v_1$, $\widetilde{v}_0 = 2v_0$ in odd characteristic, and $\widetilde{v}_1 = h_1 + u_0 + t_2$, $\widetilde{v}_0 = h_0 + u_0 \cdot u_1$ in even characteristic. This reduces the operation count of Step 1 to 1S in odd characteristic and 1S, 1M in even characteristic.
2. In Step 3, let

$$\begin{aligned} k'_3 &= -2u_1 \ , \\ k'_2 &= f_4 + 3t_2 - 2u_0 \ , \\ k'_1 &= f_3 + 2(t_2 + u_0 - k'_2) \cdot u_1 \ , \\ k'_0 &= f_2 - v_1^2 - k'_1 \cdot u_1 - (k'_2 + 4u_0) \cdot (t_2 + 2u_0) - 9u_0^2 \end{aligned}$$

   in odd characteristic. Let

$$\begin{aligned} k'_3 &= 0 \ , \\ k'_2 &= v_1 + f_6 t_2 \ , \\ k'_1 &= v_0 \ , \\ k'_0 &= f_2 + (h_1 + v_1 + t_2) \cdot v_1 + u_1 \cdot v_0 + f_6(t_2^2 + u_0^2) \end{aligned}$$

   in even characteristic. This reduces the operation count of Step 3 to 2S, 3M in odd characteristic and 2S, 2M, 2C in even characteristic.
3. In Step 5, let $\widehat{w}_0 = s'^2_1 - r^2$ in odd characteristic, and let $\widehat{w}_0 = (s'_1 + r) \cdot s'_1 + f_6 r^2$ in even characteristic. This reduces the operation count of Step 5 to 2S in odd characteristic and 1S, 1M, 1C in even characteristic.
4. In Step 7, let

$$\begin{aligned} u'_1 &= 2\widehat{w}_3 \cdot (u_1 + s_0 \cdot s_1) \ , \\ u'_0 &= \widehat{w}_3 \cdot (s_0^2 + \widetilde{v}_1 \cdot s_1 - k'_2) \end{aligned}$$

   in odd characteristic, and

$$\begin{aligned} u'_1 &= \widehat{w}_3 \cdot (s_0 + u_1 \cdot s_1) \ , \\ u'_0 &= \widehat{w}_3 \cdot ((s_0 + u_1) \cdot s_0 + \widetilde{v}_1 \cdot s_1 + k'_2) \end{aligned}$$

   in even characteristic. This reduces the operation count of Step 7 to 1S, 4M in odd characteristic and 5M in even characteristic.

5. In Step 8, we skip the calculation of $\overline{w}_2$. Let

$$v_1' = -v_1 + (s_0 + s_1) \cdot (\overline{z}_0 + \overline{z}_1) - \overline{w}_0 - (1 + u_1') \cdot \overline{w}_1 \ ,$$
$$v_0' = -v_0 + \overline{w}_0 - u_0' \cdot \overline{w}_1$$

in odd characteristic, and

$$v_1' = h_1 + v_1 + u_0' + (s_0 + s_1) \cdot (\overline{z}_0 + \overline{z}_1) + \overline{w}_0 + \overline{w}_1 + u_1' \cdot (u_1' + \overline{w}_1) \ ,$$
$$v_0' = h_0 + v_0 + \overline{w}_0 + u_0' \cdot (u_1' + \overline{w}_1)$$

in even characteristic. This reduces the operation count of Step 8 to 5M in both odd and even characteristic.

When $\deg(u) = 2$, the total operation count for doubling in the adapted basis for the general case of $\deg(u') = 2$ is 1I, 6S, 26M in odd characteristic and 1I, 4S, 28M, 3C in even characteristic.

Comments when $r = 0$. In the doubling formulas, r is the resultant of u and $2v - h$. Here, $r = 0$ implies that either one or both of the points of $D = P_1 + P_2 - 2\infty$ equals its own opposite.

1. Both points equal their own opposite if and only if $2v - h \equiv 0 \pmod{u}$, or equivalently, $\widetilde{v}_1 = \widetilde{v}_0 = 0$. In this case, $D' = 2D$ is the identity divisor ($u = 1$, $v = y + h$ in reduced basis or $u = 1$, $v = 0$ in adapted basis).
2. If $\widetilde{v}_1 \neq 0$, then exactly one point of D is equal to its own opposite. In this case, the common factor of u and $2v - h$ is $\widetilde{v} = \widetilde{v}_1 x + \widetilde{v}_0$. The root of $\widetilde{v}$ is the x-coordinate of the point equal to its own opposite, so $x_1 = -\frac{\widetilde{v}_0}{\widetilde{v}_1}$. The x-coordinate of the other point is $x_2 = -u_1 - x_1 = -u_1 + \frac{\widetilde{v}_0}{\widetilde{v}_1}$. Subtracting this point from D and doubling gives the desired result $D' = 2[x - x_2, v(x_2)]$, which can be computed using the (adapted) degree 1 doubling formulas, then changed into reduced basis if desired.

Divisor Doubling, Special Case ($\deg(u) = 1$). For doubling a degree 1 divisor, the derivative of the curve equations is required. These are essentially the same formulas as the imaginary model, with the only difference being to change the result into the reduced basis if necessary:

$$u' = u^2$$
$$v' = v + u \cdot (f'(-u_0) - v(-u_0) \cdot h'(-u_0)) \cdot (2v(-u_0) + h(-u_0)))^{-1}$$

Doubling, Reduced Basis, $\deg u = 1$		
Input	$u = x + u_0$, $v = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + (y_1 + h_1)x + v_0$ Precomputed Constants in Table 1	
Output	$[u', v'] = 2[u, v]$	
Step	Expression	Operations
1	$\underline{u' = x^2 + u'_1 x + u'_0}$ $u'_1 = 2u_0$, $u'_0 = u_0^2$	1S
2	$\underline{v' = (y_3 + h_3)x^3 + (y_2 + h_2)x^2 + v'_1 x + v'_0}$ $w_0 = 2v_0 + h_0 - u_0 \cdot (c_1 + y_1 - u_0 \cdot (c_2 - c_3 u_0))$ If $w_0 = 0$, then $u' = 1$, $v' = y + h$. $w_1 = f'(-u_0) + v(-u_0) \cdot h'(-u_0)$ $w_2 = w_1 \cdot (w_0)^{-1}$ $w_3 = y_2 + h_2 - (y_3 + h_3)u'_1$ $v'_1 = w_2 + (y_3 + h_3)u'_0 + w_3 \cdot u'_1$ $v'_0 = \widetilde{v}_0 + w_2 \cdot u_0 + w_3 \cdot u'_0$	1I, 10M, 5C
Total		1I, 1S, 10M, 5C

Doubling, Adapted Basis, $\deg u = 1$		
Input	$u = x + u_0$, $v = v_0$	
Output	$[u', v'] = 2[u, v]$	
Step	Expression	Operations
1	$\underline{u' = x^2 + u'_1 x + u'_0}$ $u'_1 = 2u_0$, $u'_0 = u_0^2$	1S
2	$\underline{v' = v'_1 x + v'_0}$ $w_0 = 2v_0 + h_0 - u_0 \cdot (h_1 - u_0 \cdot (h_2 - h_3 u_0))$ If $w_0 = 0$, then $u' = 1$, $v' = 0$. $w_1 = f'(-u_0) + v_0 \cdot h'(-u_0)$ $w_2 = w_1 \cdot (w_0)^{-1}$ $v'_1 = w_1$, $v'_0 = v_0 + w_1 \cdot u_0$	1I, 8M, 5C
Total		1I, 1S, 8M, 5C

Comments. The only special case occurs when the divisor has order 2, which occurs exactly when the term $w_0 = 2\widetilde{v} + h(-u_0)$ in reduced basis and $w_0 = 2\widetilde{v} + h(-u_0)$ in adapted basis equals 0. When this is the case, doubling the divisor results in the identity divisor ($u = 1$, $v = y + h$ in reduced basis or $u = 1$, $v = 0$ in adapted basis).

8. Summary of Results

The best known results for the imaginary case are found in [Lan05]. As noted after the divisor addition table, an improvement of one less squaring has been found which applies to the addition formula in the imaginary case (though not in the doubling case). Compared to the imaginary model, the addition formulas in the main case for the real model requires four more multiplications in odd characteristic, and five more multiplications but one less squaring in even characteristic. The doubling formulas require six more multiplications but one less squaring in odd characteristic than the imaginary model. It is worth noting that the baby step operation is the cheapest of all, including adding a degenerate divisor in the imaginary case, the nearest imaginary analogue to the baby step.

The table below summarizes the comparison. The operation counts listed assume that simplified isomorphic models of the imaginary hyperelliptic curves are used, as in [Lan05]. The operation counts for all operations in the real case include those for the generic (non-special) cases as well as the simplified isomorphic models in even and odd characteristic from Section 2 (see Table 1).

Imaginary	
Addition (Deg 1 + Deg 2)	1I, 1S, 10M
Addition	1I, 2S, 22M
Doubling	1I, 5S, 22M

Real, Reduced Basis			
	General	Odd	Even
(Inverse) Baby Step	1I, 6M, 4C	1I, 2S, 4M	1I, 1S, 5M
Addition (Deg 1 + Deg 2)	1I, 2S, 14M, 6C	1I, 3S, 13M	1I, 3S, 12M, 1C
Addition	1I, 1S, 27M, 4C	1I, 2S, 26M	1I, 1S, 27M
Doubling	1I, 2S, 30M, 10C	1I, 4S, 28M	1I, 2S, 29M

Real, Adapted Basis			
	General	Odd	Even
(Inverse) Baby Step	1I, 8M, 5C	1I, 3S, 5M	1I, 2S, 6M, 4C
Addition (Deg 1 + Deg 2)	1I, 2S, 16M, 7C	1I, 3S, 15M	1I, 3S, 14M, 3C
Addition	1I, 2S, 27M, 7C	1I, 3S, 26M	1I, 2S, 27M, 3C
Doubling	1I, 3S, 30M, 19C	1I, 6S, 26M	1I, 4S, 28M, 3C

One immediate conclusion based on the tables is that the reduced basis is more amendable for computations in the real case than the more standard adapted basis. All formulas are cheaper or the same using reduced basis. Both bases require the same amount of storage. Although the reduced basis has $\deg(v) = 3$, the degree 3 and 2 terms are fixed by the curve equation, so only the linear and constant terms have to be stored in an implementation.

The main obstruction from getting more competitive formulas in the real case is the extra coefficient interfering with the inversion step. In the imaginary case, the leading coefficient of the new u is simply s_1^2, which allows one to simplify both addition and doubling formulas. In the real case, we found that computing s_0 and s_1 explicitly was the most efficient way to compute addition and doubling of divisors.

9. Numerical results

As cryptographic applications were one of our motivations for developing explicit formulas for divisor arithmetic on genus 2 real hyperelliptic curves, we have implemented key exchange protocols in the imaginary and real models in order to determine whether the real model can be competitive with the imaginary model in terms of efficiency. In the imaginary case, the main operation is scalar multiplication using non-adjacent form, which we will refer to as SCALAR-MULT. In the fixed base scenario, we assume that the base divisor is degenerate with $\deg(u) = 1$ in order to take advantage of the special case multiplication formulas, thereby using the closest analogue to a baby step. In the real case, there are two variations of scalar multiplication described in [JSS06] that comprise the key exchange protocol. Algorithm VAR-DIST2 is a variation of NAF-based scalar multiplication using doubling and baby steps, whereas Algorithm FIXED-DIST2 generalizes the usual

NAF-based scalar multiplication algorithm. All three of these algorithms were implemented, using the explicit formulas from [Lan05] for the imaginary case and the formulas in this paper for the real case.

We used the computer algebra library NTL [Sho01] for finite field and polynomial arithmetic and the GNU `C++` compiler version 4.1.2. The computations described below were performed on a Intel Core Duo 2.66 GHz computer running Linux. Although faster absolute times could be obtained using customized implementations of finite field arithmetic, our goal was to compare the relative performance of algorithms in the imaginary and real settings using exactly the same finite fields as opposed to producing the fastest times possible.

All three algorithms were implemented using curves defined over $\mathbb{F}_p$ and $\mathbb{F}_{2^n}$. We ran numerous examples of the three scalar multiplication algorithms using curves with genus 2 where the underlying finite field was chosen so that the size of the set $\mathcal{R}$ (approximately q^g where the finite field has q elements) was roughly 2^{160}, 2^{224}, 2^{256}, 2^{384}, and 2^{512}. These curves offer 80, 112, 128, 192, and 256 bits of security for cryptographic protocols based on the corresponding DLP. NIST [oSN03] currently recommends these five levels of security for key establishment in U.S. Government applications.

For the finite field $\mathbb{F}_p$, we chose a random prime p of appropriate length such that p^2 had the required bit length. For the finite fields $\mathbb{F}_{2^n}$, we used $n \in \{80, 112, 128, 192, 256\}$. For each finite field, we randomly selected 5000 curves and executed Diffie-Hellman key exchange once for each curve. Thus, we ran 10000 instances of Algorithm SCALAR-MULT (two instances for each participant using each curve) and 5000 instances each of Algorithm FIXED-DIST2 and VAR-DIST2 (one instance of each algorithm per participant using each curve). The random exponents used had 160, 224, 256, 384, and 512 bits, respectively, ensuring that the number of bits of security provided corresponds to the five levels recommended by NIST (again, considering only generic attacks). In order to provide a fair comparison between the three algorithms, the same sequence of random exponents was used for each run of the key exchange protocol.

Tables 2 and 3 contain the average CPU time in milliseconds for each of the three algorithms. The times required to generate domain parameters, including the divisors D_{d+3} and D^* required for our real hyperelliptic curve protocols (see [JSS06]), are not included in these timings, as domain parameter generation is a one-time computation. In the imaginary case, "Fixed" denotes the time for algorithm SCALAR-MULT when using a fixed, degenerate degree 1 divisor as the base (round 1 of Diffie-Hellamn key exchange) and "Var" denotes the time for SCALAR-MULT using an arbitrary base (round 2). In the real case, the time for Algorithm FIXED-DIST2 by "Fixed" and that for Algorithm VAR-DIST2 by "Var." We also list the times required to execute Diffie-Hellman key exchange using both real and imaginary models, denoted by "DH Total." The runtimes achieved using the real model are slower than those using the imaginary model, but they are certainly close, within 6 milliseconds.

10. Conclusions

The formulas presented in this paper are the first complete explicit formulas for divisor arithmetic on a real hyperelliptic curve. Although they are a few field multiplications slower than their imaginary counterparts, they will certainly out-perform

Table 2. Scalar multiplication and key exchange timings over $\mathbb{F}_p$ (in milliseconds).

Security Level	Imaginary			Real		
(in bits)	Fixed	Var	DH Total	Fixed	Var	DH Total
80	2.137	2.304	4.440	2.307	2.618	4.925
112	3.545	3.942	7.487	3.809	4.469	8.278
128	4.702	5.149	9.851	5.003	5.869	10.872
192	10.526	11.562	22.088	11.192	13.048	24.240
256	15.560	17.077	32.636	16.492	19.168	35.660

Table 3. Scalar multiplication and key exchange timings over $\mathbb{F}_{2^n}$ (in milliseconds).

Security Level	Imaginary			Real		
(in bits)	Fixed	Var	DH Total	Fixed	Var	DH Total
80	4.721	5.331	10.052	5.112	6.139	11.250
112	4.096	4.475	8.571	4.425	5.076	9.500
128	4.814	5.304	10.118	5.138	5.920	11.057
192	11.700	12.942	24.641	12.715	14.721	27.436
256	22.255	24.572	46.827	24.525	28.326	52.851

a generic implementation of Cantor's algorithm and will be useful for any computational tasks in the class group or infrastructure. Unfortunately, cryptographic protocols using our formulas in the real model are also slower than those using the imaginary case, even with the improved protocols described in [JSS06] in which many divisor additions are traded for significantly faster baby steps. Nevertheless, we hope the fact that we can achieve run times close to those in the imaginary case will increase interest in cryptographic protocols in this setting.

There is still much work to be done on this topic. It is certainly possible that further improvements can be found to our formulas. Reducing the number of field multiplications required for addition and doubling by only two or three may result in the cryptographic protocols in the real setting being slightly faster than the imaginary case. Another possible improvement is to investigate compound operations. In particular, compounding the doubling and baby step operations will almost certainly save a few multiplications as compared to performing them separately, and this would also improve the speed of the VAR-DIST2 scalar multiplication algorithm (doubling and baby steps) from [JSS06]. Another possibility is to examine the applicability of the NUCOMP algorithm [JSS07] in the explicit formulas settings. NUCOMP has proven to offer significant improvement over Cantor's algorithm even for quite small genera, and it is possible that the ideas of NUCOMP will lead to improvements for the low genus case once the formulas are made explicit.

Finally, a great deal of work has been done on explicit formulas in the imaginary setting including using projective coordinates to obtain inversion-free formulas, formulas for genus 3 and 4, and explicit formulas via theta functions. Formulas using projective coordinates in the real model of a genus 2 hyperelliptic curves have been developed recently by Erickson, Ho, and Zemedkun [EHZ10]. All of the other topics are work in progress.

10.0.1. *Acknowledgments.* This paper, the full version of our preliminary results in [EJS+07], is an outcome of a research project proposed at the RMMC Summer School in Computational Number Theory and Cryptography which was held at the University of Wyoming in 2006. We would like to thank the following sponsors for their support: the University of Wyoming, the National Science Foundation (grant DMS-0612103), the Rocky Mountain Mathematics Consortium, The Number Theory Foundation, The Institute for Mathematics and its Applications (IMA), The Fields Institute, The Centre for Information Security and Cryptography (CISaC) and iCORE of Canada.

References

[Ava04] R. M. Avanzi. Aspects of hyperelliptic curves over large prime fields in software implementations. In *Cryptographic Hardware and Embedded Systems—CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 148–162, Berlin, 2004. Springer.

[CF05] H. Cohen and G. Frey, editors. *Handbook of Elliptic and Hyperelliptic Curve Cryptography*. Number 34 in Discrete Mathematics and Its Applications. Chapman& Hall/CRC, 2005.

[DGTT07] P. Gaudry, E. Thom'e, N. Thériault, and C. Diem. A double large prime variation for small genus hyperelliptic index calculus. *Math. Comp.*, 76(257):475–492, 2007.

[Eng01] A. Enge. How to distinguish hyperelliptic curves in even characteristic. In K. Alster, J. Urbanowicz, and H. C. Williams, editors, *Public-Key Cryptography and Computational Number Theory*, pages 49–58, Berlin, 2001. De Gruyter.

[EHZ10] S. Erickson, T. Ho, S. Zemedkun, *Explicit projective formulas for real hyperelliptic curves of genus* 2, Preprint.

[EJS+07] S. Erickson, M. J. Jacobson, Jr., N. Shang, S. Shen, and A. Stein, *Explicit formulas for real hyperelliptic curves of genus* 2 *in affine representation*, WAIFI 2007 (C. Carlet and B. Sunar, eds.), Lecture Notes in Computer Science, vol. 4547, Springer Verlag, 2007, pp. 202–218.

[Fon08a] F. Fontein. Groups from cyclic infrastructures and Pohlig-Hellman in certain infrastructures. *Advances in Mathematics of Communications*, 2(3), 2008.

[Fon08b] F. Fontein. *The Infrastructure of a Global Field and Baby Step-Giant Step Algorithms*. PhD thesis, University of Zurich, Zurich, Switzerland, 2008.

[GHM08] S. D. Galbraith, M. Harrison, and D. J. Mireles Morales. Efficient hyperelliptic arithmetic using balanced representation for divisors. In Alfred J. van der Poorten and Andreas Stein, editors, *ANTS*, volume 5011 of *Lecture Notes in Computer Science*, pages 342–356. Springer, 2008.

[GLM08] S. D. Galbraith, X Lin, and D. J. Mireles Morales. Pairings on hyperelliptic curves with a real model. In *Pairing*, pages 265–281, 2008.

[Gau00] P. Gaudry. On breaking the discrete log on hyperelliptic curves. In *Advances in Cryptology - Eurocrypt'2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 19–34. Springer-Verlag, 2000.

[JMS04] M. J. Jacobson, Jr., A. J.Menezes, and A. Stein. Hyperelliptic curves and cryptography. In *High Primes and Misdemeanours: lectures in honour of the 60th birthday of Hugh Cowie Williams*, volume 41 of *Fields Institute Communications Series*, pages 255–282. Amer. Math. Soc., 2004.

[JSS06] M. J. Jacobson, Jr., R. Scheidler, and A. Stein. Cryptographic protocols on real and imaginary hyperelliptic curves. *Advances in Mathematics of Communications*, 1(2):197–221, 2007.

[JSS07] M. J. Jacobson, Jr., R. Scheidler, and A. Stein. Fast arithmetic on hyperelliptic curves via continued fraction expansions. In T. Shaska, W.C. Huffman, D. Joyner, and V. Ustimenko, editors, *Advances in Coding Theory and Cryptology*, volume 3 of *Series on Coding Theory and Cryptology*, pages 201–244. World Scientific Publishing, 2007.

[Kob88] N. Koblitz. Hyperelliptic cryptosystems. *Journal of Cryptology*, 1:139–150, 1988.

[Lan05] T. Lange. Formulae for arithmetic on genus 2 hyperelliptic curves. *Appl. Algebra Engrg. Comm. Comput.*, 15:295–328, 2005.

[Mir08] D. J. Mireles Morales. An analysis of the infrastructure in real function fields. eprint archive, No. 2008/299, 2008.

[MST99] V. Müller, A. Stein, and C. Thiel. Computing discrete logarithms in real quadratic congruence function fields of large genus. *Mathematics of Computation*, 68:807–822, 1999.

[Mum84] D. Mumford. *Tata Lectures on Theta I, II*. Birkhäuser, Boston, 1983/84.

[MWZ96] A. J. Menezes, Y. Wu, and R. J. Zuccherato. An elementary introduction to hyperelliptic curves. Technical Report CORR 96-19, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, 1996. In: Koblitz, N.: Algebraic Aspects of Cryptography. Springer-Verlag, Berlin Heidelberg New York (1998).

[oSN03] National Institute of Standards and Technology (NIST). Recommendation on key establishment schemes. NIST Special Publication 800-56, January 2003.

[Sho01] V Shoup. NTL: A library for doing number theory. Software, 2001. See `http://www.shoup.net/ntl`.

[PR99] S. Paulus and H.-G. Rück. Real and imaginary quadratic representations of hyperelliptic function fields. *Mathematics of Computation*, 68:1233–1241, 1999.

[PWP03] J. Pelzl, T. Wollinger, and C. Paar. Low cost security: explicit formulae for genus-4 hyperelliptic curves. In *Selected Areas in Cryptography — SAC 2003*, volume 3006 of *Lect. Notes Comput. Sci*, pages 1–16, Berlin, 2003. Spinger-Verlag.

[Sch01] R. Scheidler. Cryptography in quadratic function fields. *Des. Codes Cryptography*, 22:239–264, 2001.

[SSW96] R. Scheidler, A. Stein, and H. C. Williams. Key-exchange in real quadratic congruence function fields. *Designs, Codes and Cryptography*, 7:153–174, 1996.

[Ste01] A. Stein. Sharp upper bounds for arithmetics in hyperelliptic function fields. *Journal of the Ramanujan Mathematical Society*, 9-16(2):1–86, 2001.

[WPP05] T. Wollinger, J. Pelzl, and C. Paar. Cantor versus Harley: optimization and analysis of explicit formulae for hyperelliptic curve cryptosystems. *IEEE Trans. Computers*, 54:861–872, 2005.

Received August 2010; revised ???.

E-mail address: Stefan.Erickson@ColoradoCollege.edu
E-mail address: jacobs@cpsc.ucalgary.ca
E-mail address: andreas.stein1@uni-oldenburg.de