MATHEMATICS OF COMPUTATION
Volume 86, Number 304, March 2017, Pages 881–898
http://dx.doi.org/10.1090/mcom/3123
Article electronically published on July 15, 2016

ERROR BOUNDS ON COMPLEX FLOATING-POINT MULTIPLICATION WITH AN FMA

CLAUDE-PIERRE JEANNEROD, PETER KORNERUP, NICOLAS LOUVET, AND JEAN-MICHEL MULLER

Abstract. The accuracy analysis of complex floating-point multiplication done by Brent, Percival, and Zimmermann [*Math. Comp.*, 76:1469–1481, 2007] is extended to the case where a fused multiply-add (FMA) operation is available. Considering floating-point arithmetic with rounding to nearest and unit roundoff u, we show that their bound $\sqrt{5}\,u$ on the normwise relative error $|\widehat{z}/z - 1|$ of a complex product z can be decreased further to $2u$ when using the FMA in the most naive way. Furthermore, we prove that the term $2u$ is asymptotically optimal not only for this naive FMA-based algorithm but also for two other algorithms, which use the FMA operation as an efficient way of implementing rounding error compensation. Thus, although highly accurate in the componentwise sense, these two compensated algorithms bring no improvement to the normwise accuracy $2u$ already achieved using the FMA naively. Asymptotic optimality is established for each algorithm thanks to the explicit construction of floating-point inputs for which we prove that the normwise relative error then generated satisfies $|\widehat{z}/z - 1| \to 2u$ as $u \to 0$. All our results hold for IEEE floating-point arithmetic, with radix β, precision p, and rounding to nearest; it is only assumed that underflows and overflows do not occur and that $\beta^{p-1} \geqslant 24$.

1. Introduction

Given complex numbers $x = a + ib$ and $y = c + id$, let their product $z = xy$ be expressed as

$$z = ac - bd + i(ad + bc).$$

Assuming that a, b, c, d are floating-point numbers and that the operations $+, -, \times$ on such numbers are performed with rounding to nearest (RN), the conventional way of evaluating the expression above can be described as follows:

$$\mathcal{A}_0 : (a + ib, c + id) \mapsto \mathrm{RN}\big(\mathrm{RN}(ac) - \mathrm{RN}(bd)\big) + i \cdot \mathrm{RN}\big(\mathrm{RN}(ad) + \mathrm{RN}(bc)\big).$$

The normwise accuracy of algorithm $\mathcal{A}_0$ was studied by Brent, Percival, and Zimmermann [3] for standard floating-point arithmetic (with radix β and precision p such that $\beta^{p-1} \geqslant 16$) and assuming that underflows and overflows do not occur. They showed that the computed value has the form

$$\widehat{z}_0 = z(1 + \epsilon), \qquad |\epsilon| < \sqrt{5}\,u, \qquad u = \tfrac{1}{2}\beta^{1-p},$$

which for z nonzero implies that the normwise relative error $|\widehat{z}_0/z - 1|$ is always less than $\sqrt{5} = 2.236\ldots$ times the unit roundoff. For $\beta = 2$ and rounding 'to nearest

Received by the editor September 26, 2013 and, in revised form, July 25, 2014, May 15, 2015, and September 28, 2015.

2010 *Mathematics Subject Classification.* Primary 65G50.

even', they also showed by constructing specific inputs for $\mathcal{A}_0$ that the upper bound $\sqrt{5}\,u$ should be considered sharp: in the cases $p = 24$ and $p = 53$ the largest possible errors have the form $\sqrt{4.9999899864\ldots}\,u$ and $\sqrt{4.9999999999999893\ldots}\,u$, respectively; more generally, when $p \geqslant 2$ they provide floating-point numbers a, b, c, d for which $|\widehat{z}_0/z - 1| = \sqrt{5}\,u - O(u^2)$ as $u \to 0$, so that the relative error bound $\sqrt{5}\,u$ is *asymptotically optimal* for algorithm $\mathcal{A}_0$.

The goal of our paper is to extend this study of the normwise accuracy of complex floating-point multiplication by allowing not only floating-point $+$, $-$, $\times$ but also the fused multiply-add (FMA) operation. Given three floating-point numbers a, b, c the FMA evaluates expressions of the form $ab+c$ with one rounding error instead of two, so that with round to nearest, the result is $\mathrm{RN}(ab+c)$ rather than $\mathrm{RN}(\mathrm{RN}(ab)+c)$. This operation has been required since the 2008 revision of the IEEE 754 standard for floating-point arithmetic [7] and is therefore being supported by an increasing number of processors.

With an FMA, the conventional way of evaluating $ac - bd + i(ad + bc)$ becomes

$$\mathcal{A}_1 : (a + ib, c + id) \mapsto \mathrm{RN}\big(ac - \mathrm{RN}(bd)\big) + i \cdot \mathrm{RN}\big(ad + \mathrm{RN}(bc)\big).$$

Algorithm $\mathcal{A}_1$ is of course just one of four variants that differ only in the choice of the products to which the FMA operations apply. Our first contribution is to prove that for any of these four conventional FMA-based algorithms the computed complex product $\widehat{z}_1$ satisfies

$$|\widehat{z}_1 - z| \leqslant 2u|z| \tag{1.1}$$

and, by constructing inputs a, b, c, d for which $|\widehat{z}_1/z - 1| = 2u - O(u^{1.5})$ as $u \to 0$, that the relative error bound (1.1) is asymptotically optimal.

Another classical way of exploiting the FMA is for efficiently computing the error in a floating-point product [6, §2.6]: given two floating-point numbers a and b and barring underflow and overflow, the error $e = ab - \mathrm{RN}(ab)$ can be produced as

$$e = \mathrm{RN}\big(ab - \mathrm{RN}(ab)\big)$$

in one multiplication and one FMA. (In contrast, without an FMA and using only $+$, $-$, $\times$, the cheapest algorithm we are aware of is due to Dekker and Veltkamp and uses 17 operations; see [2, 5] and [16, p. 135].) Once such rounding errors e are available, they can be used to construct a correction term aimed at improving the overall accuracy of the computed result. This approach, called *compensation*, can be traced back to the works of Møller [13,14], Kahan [10], Dekker [5], Pichat [17,18], and Linnainmaa [11,12]; Cornea, Harrison, and Tang [4, p. 273] use it explicitly in the following algorithm to evaluate

$$r = ab + cd$$

accurately in 7 floating-point operations:

algorithm $\mathrm{CHT}(a, b, c, d)$
$\widehat{w}_1 := \mathrm{RN}(ab)$; $\widehat{w}_2 := \mathrm{RN}(cd)$;
$e_1 := \mathrm{RN}(ab - \widehat{w}_1)$; $e_2 := \mathrm{RN}(cd - \widehat{w}_2)$; // these two operations are exact.
$\widehat{f} := \mathrm{RN}(\widehat{w}_1 + \widehat{w}_2)$;
$\widehat{e} := \mathrm{RN}(e_1 + e_2)$;
$\widehat{r} := \mathrm{RN}\big(\widehat{f} + \widehat{e}\big)$;
return $\widehat{r}$;

In algorithm CHT an approximation $\widehat{f}$ to r is computed by simply evaluating and adding the products ab and cd. Simultaneously, the rounding errors e_1 and e_2 due to the evaluation of these two products are computed with two FMA operations and then added together into a correction term $\widehat{e}$. The corrected solution $\widehat{r}$ is finally produced as the rounded sum of $\widehat{f}$ and $\widehat{e}$. While $\widehat{f}$ can be inaccurate due to cancellation, $\widehat{r}$ turns out to be always highly accurate. In radix 2 floating-point arithmetic, Cornea, Harrison, and Tang [4] show that in the absence of underflow and overflow $|\widehat{r} - r| \leqslant O(u)|r|$ and, analyzing their algorithm further, Muller [15] shows that $|\widehat{r} - r| \leqslant (2u + 7u^2 + 6u^3)|r|$ and that this bound is asymptotically optimal. Recently, Jeannerod [8] improved the analysis of the algorithm by providing asymptotically optimal error bounds valid for any radix β: from [8, Theorem 1.1] it follows that when $\beta^{p-1} \geqslant 24$,

$$|\widehat{r} - r| \leqslant \begin{cases} 2u|r| & \text{if } \beta \text{ is odd or RN is 'to nearest even',} \\ \frac{2\beta u + 2u^2}{\beta - 2u^2}|r| & \text{otherwise.} \end{cases} \tag{1.2a}$$

Thus, for rounding 'to nearest even' (which is the default rounding direction attribute, called roundTiesToEven in the IEEE 754-2008 standard [7]), the relative error of the CHT algorithm is at most $2u$. However, for other tie-breaking strategies it may be impossible to remove the term $O(u^2)$ in the bound $\frac{2\beta u + 2u^2}{\beta - 2u^2} = 2u + O(u^2)$, and [8, Theorem 1.2] shows the existence of input values a, b, c, d for which rounding 'to nearest away' (roundTiesToAway in [7]) yields an error larger than $2u + \frac{2}{\beta}u^2 - 4u^3$.

A straightforward application of the CHT algorithm is to evaluate accurately the real and imaginary parts of the complex product $z = ac - bd + i(ad + bc)$. This is shown in algorithm $\mathcal{A}_2$ below, which uses 14 floating-point operations:

$$\mathcal{A}_2 : (a + ib, c + id) \mapsto \mathrm{CHT}(a, c, -b, d) + i \cdot \mathrm{CHT}(a, d, b, c).$$

By applying (1.2a) twice (first with $r = ac - bd$, and then with $r = ad + bc$), we deduce immediately that if $\beta^{p-1} \geqslant 24$, then the approximate product $\widehat{z}_2$ computed by $\mathcal{A}_2$ satisfies

$$|\widehat{z}_2 - z| \leqslant \begin{cases} 2u|z| & \text{if } \beta \text{ is odd or RN is 'to nearest even',} \\ \frac{2\beta u + 2u^2}{\beta - 2u^2}|z| & \text{otherwise.} \end{cases} \tag{1.2b}$$

Our second contribution is to show that the bound $2u + O(u^2)$ implied by (1.2b) is asymptotically optimal for algorithm $\mathcal{A}_2$; that is, its term $2u$ cannot be replaced by λu for some constant $\lambda < 2$. In particular, this asymptotic optimality result says that the compensation for the errors in ac, bd, ad, bc performed by algorithm CHT brings no improvement to the normwise relative accuracy $2u$ of the non-compensated, conventional algorithm $\mathcal{A}_1$. Note that the simple bound $|\widehat{z}_2 - z| \leqslant 2u|z|$ holds in particular when $p \geqslant 6$ and ties are rounded to even and can thus be used in the vast majority of practical scientific computations.

Algorithm CHT makes no use of the FMA to produce the initial approximation $\widehat{f}$ to $r = ab + cd$. However, if we employ the FMA already at this stage, then the rounding error of only one product (say, cd) needs to be recovered to eventually

ensure high relative accuracy. This is the basis of the algorithm below, attributed to Kahan in [6, p. 65], which evaluates r accurately in 4 floating-point operations:

algorithm Kahan(a, b, c, d)
  $\widehat{w} := \mathrm{RN}(cd)$;
  $e := \mathrm{RN}(cd - \widehat{w})$; // this operation is exact: $e = cd - \widehat{w}$.
  $\widehat{f} := \mathrm{RN}(ab + \widehat{w})$;
  $\widehat{r} := \mathrm{RN}\big(\widehat{f} + e\big)$;
  return $\widehat{r}$;

It was shown in [9, Theorem 1.2] that for $\beta, p \geqslant 2$ and in the absence of underflow and overflow, the result $\widehat{r}$ produced by Kahan's algorithm satisfies

$$|\widehat{r} - r| \leqslant 2u|r| \tag{1.3a}$$

and for β even and rounding 'to nearest even', that this relative error bound is asymptotically optimal. In the same way as for the CHT algorithm, we can apply Kahan's algorithm to the accurate evaluation of the real and imaginary parts of the complex product z, but now using 8 floating-point operations instead of 14:

$$\mathcal{A}_3 : (a + ib, c + id) \mapsto \mathrm{Kahan}(a, c, -b, d) + i \cdot \mathrm{Kahan}(a, d, b, c).$$

Since the expression $r = ab + cd$ can be evaluated either as Kahan(a, b, c, d) or Kahan(c, d, a, b), algorithm $\mathcal{A}_3$ comes in fact with three other variants. Using (1.3a), we see immediately that for any of these four algorithms the computed complex product $\widehat{z}_3$ satisfies

$$|\widehat{z}_3 - z| \leqslant 2u|z|. \tag{1.3b}$$

Our third contribution in this paper is to show that the normwise relative error bound (1.3b) is asymptotically optimal, which proves that $\widehat{z}_3$ still does not improve on the accuracy achieved by the straightforward FMA-based solution $\widehat{z}_1$.

To summarize, the two main conclusions in this paper are as follows:

- The availability of an FMA makes it possible to replace the classical accuracy bound $\sqrt{5}u$ by $2u$, and this new bound is sharp when the FMA is used in the conventional way, as in algorithm $\mathcal{A}_1$.
- The term $2u$ cannot be reduced further by FMA-based, compensated schemes like algorithms $\mathcal{A}_2$ and $\mathcal{A}_3$.

These conclusions hold for *normwise* relative accuracy only. For *componentwise* relative accuracy, where we bound $\max\big(|\mathrm{Re}\,\widehat{z}/\mathrm{Re}\,z - 1|, |\mathrm{Im}\,\widehat{z}/\mathrm{Im}\,z - 1|\big)$ instead of $|\widehat{z}/z - 1|$, the benefit of using FMA-based compensation via the CHT algorithm or Kahan's cheaper variant is clear: algorithms $\mathcal{A}_2$ and $\mathcal{A}_3$ guarantee a tiny componentwise relative error, while algorithms $\mathcal{A}_0$ and $\mathcal{A}_1$ can both be highly inaccurate due to possible catastrophic cancellations in the real or imaginary part of the computed product.

This paper is organized as follows. After some preliminaries in Section 2, we establish the normwise relative error bound (1.1) in Section 3. Section 4 then gathers our asymptotic optimality results: we begin, in Section 4.1, by constructing some input for which both algorithm $\mathcal{A}_1$ and algorithm $\mathcal{A}_3$ have their normwise relative error lower bounded by $2u - O(u^{1.5})$; in Section 4.2, this lower bound is achieved for both algorithm $\mathcal{A}_2$ and algorithm CHT via the construction of another input. Concluding remarks are given in Section 5.

Finally, let us emphasize that all the error bounds presented in this paper are valid assuming the absence of underflow/overflow. Some mild assumptions on the radix β and precision p are also used: our upper bounds are valid for any $\beta, p \geqslant 2$ such that $\beta^{p-1} \geqslant 24$, and all our lower bounds assume $\beta^{p-1} \geqslant 11$. Therefore, *the results of this paper hold for all IEEE 754-2008 floating-point formats and as long as underflows and overflows do not occur.*

2. Preliminaries

This section provides the main definitions and assumptions used in the paper. We also show that when analyzing the normwise accuracy of any of the complex multiplication algorithms studied here, we can assume with no loss of generality that the operands $a+ib$ and $c+id$ satisfy $abcd \geqslant 0$. Then, we consider the possible variants of algorithms $\mathcal{A}_1$ and $\mathcal{A}_3$ and show that we can restrict the accuracy analyses to these two algorithms only. Finally, we recall that (1.2b) and (1.3b) are simply consequences of [8, Theorem 1.1] and [9, Theorem 1.2], respectively.

2.1. **Definitions and assumptions.** Throughout this paper β and p are integers such that

$$\beta, p \geqslant 2,$$

and $\mathbb{F}$ is the set of floating-point numbers with radix β and precision p, assuming an unbounded exponent range:

$$\mathbb{F} = \{0\} \cup \left\{ M \cdot \beta^{e-p+1} \ : \ M, e \in \mathbb{Z},\ \beta^{p-1} \leqslant |M| < \beta^p \right\}.$$

Associated with this set are the *unit roundoff* $u = \frac{1}{2}\beta^{1-p}$ as well as a round-to-nearest function RN, which maps any real number t to a nearest element in $\mathbb{F}$, denoted by $\mathrm{RN}(t)$. This rounding function is assumed to satisfy $\mathrm{RN}(-t) = -\mathrm{RN}(t)$ and $\mathrm{RN}(\beta^k t) = \beta^k \mathrm{RN}(t)$ for all $t \in \mathbb{R}$ and $k \in \mathbb{Z}$. Note that since all the results in this paper are proved using $\mathbb{F}$, whose range is unbounded, they remain valid for IEEE floating-point arithmetic as long as neither underflow nor overflow occurs.

We write ulp to denote the *unit in the last place* function: $\mathrm{ulp}(0) = 0$ and for any nonzero real number t, $\mathrm{ulp}(t)$ is the unique integer power of β such that $\beta^{p-1} \leqslant |t|/\mathrm{ulp}(t) < \beta^p$. Combining the definitions of RN, ulp, and u leads to

$$|\mathrm{RN}(t) - t| \leqslant \tfrac{1}{2}\mathrm{ulp}(t) \leqslant u|t| \quad \text{for any real number } t. \tag{2.1}$$

In particular, it follows that the exact result t of a floating-point operation is related to its correctly rounded value $\widehat{t} = \mathrm{RN}(t)$ by the identity below, referred to as the *standard model* of floating-point arithmetic [6, p. 40]:

$$\widehat{t} = t\,(1+\delta), \qquad |\delta| \leqslant u. \tag{2.2}$$

Here and hereafter, z denotes the exact product of two complex numbers $a+ib$ and $c+id$ having their real and imaginary parts in $\mathbb{F}$, that is,

$$z = ac - bd + i(ad+bc), \qquad a, b, c, d \in \mathbb{F}.$$

For each of the complex multiplication algorithms introduced in Section 1, we define

$$\widehat{z}_h = \text{the approximation to } z \text{ produced by algorithm } \mathcal{A}_h \text{ for } h \in \{0,1,2,3\},$$

and, when dealing with the real and imaginary parts of z and $\widehat{z}_h$ explicitly, we shall use the notation

$$z = R + iI, \qquad \widehat{z}_h = \widehat{R}_h + i\widehat{I}_h.$$

Note in particular that replacing (c, d) by $(-d, c)$ changes the sign of the product $abcd$ but has no effect on the rounding errors committed: the exact and approximate products z and $\widehat{z}_h$ become $\zeta = -I + iR$ and $\widehat{\zeta}_h = -\widehat{I}_h + i\widehat{R}_h$, respectively, so that $|\zeta| = |z|$ and $|\widehat{\zeta}_h - \zeta| = |\widehat{z}_h - z|$. Therefore, when establishing our error bounds it will always be possible to assume without loss of generality that

$$abcd \geqslant 0.$$

2.2. **Variants of algorithms $\mathcal{A}_1$ and $\mathcal{A}_3$.** As already mentioned in the introduction, algorithms $\mathcal{A}_1$ and $\mathcal{A}_3$ have three other variants each. This is due to the fact that with an FMA the expression $ab+cd$ can be evaluated either as $\mathrm{RN}(ab+\mathrm{RN}(cd))$ or as $\mathrm{RN}(\mathrm{RN}(ab)+cd)$, each of these two ways possibly producing a different result.

Given

$$x = a + ib, \qquad y = c + id,$$

the complex floating-point numbers returned by algorithm $\mathcal{A}_1$ and its variants are displayed in the following table:

ℓ	$\widehat{z}_1^{(\ell)}(x, y)$
1	$\mathrm{RN}(ac - \mathrm{RN}(bd)) + i \cdot \mathrm{RN}(ad + \mathrm{RN}(bc))$
2	$\mathrm{RN}(ac - \mathrm{RN}(bd)) + i \cdot \mathrm{RN}(\mathrm{RN}(ad) + bc)$
3	$\mathrm{RN}(\mathrm{RN}(ac) - bd) + i \cdot \mathrm{RN}(\mathrm{RN}(ad) + bc)$
4	$\mathrm{RN}(\mathrm{RN}(ac) - bd) + i \cdot \mathrm{RN}(ad + \mathrm{RN}(bc))$

Here, $\widehat{z}_1^{(1)}(x, y)$ is the output of algorithm $\mathcal{A}_1$. Although the $\widehat{z}_1^{(\ell)}(x, y)$ can differ for some (x, y), they share the same normwise error bound. To see this, let

$$(x^{(2)}, y^{(2)}) = (y, x), \quad (x^{(3)}, y^{(3)}) = (-ix, iy), \quad (x^{(4)}, y^{(4)}) = (iy, -ix). \tag{2.3}$$

For $\ell = 2, 3, 4$ we see that $x^{(\ell)}y^{(\ell)} = xy$, and it is easily checked that $\widehat{z}_1^{(\ell)}(x, y) = \widehat{z}_1^{(1)}(x^{(\ell)}, y^{(\ell)})$, thus leading to

$$\left|\widehat{z}_1^{(\ell)}(x, y) - xy\right| = \left|\widehat{z}_1^{(1)}(x^{(\ell)}, y^{(\ell)}) - x^{(\ell)}y^{(\ell)}\right|.$$

In other words, for $\ell = 2, 3, 4$, the normwise (absolute or relative) error committed when approximating the product xy by $\widehat{z}_1^{(\ell)}(x, y)$ is the same as the one committed by algorithm $\mathcal{A}_1$ for some input $(x^{(\ell)}, y^{(\ell)})$ that can be deduced from (x, y) via the error-free transformations defined in (2.3). Consequently, the four conventional algorithms with FMA enjoy the same normwise error bounds, and we shall focus on the analysis of $\mathcal{A}_1$ only.

Similarly, four complex multiplication algorithms based on Kahan's algorithm are also possible:

ℓ	$\widehat{z}_3^{(\ell)}(x, y)$
1	$\mathrm{Kahan}(a, c, -b, d) + i \cdot \mathrm{Kahan}(a, d, b, c)$
2	$\mathrm{Kahan}(a, c, -b, d) + i \cdot \mathrm{Kahan}(b, c, a, d)$
3	$\mathrm{Kahan}(-b, d, a, c) + i \cdot \mathrm{Kahan}(b, c, a, d)$
4	$\mathrm{Kahan}(-b, d, a, c) + i \cdot \mathrm{Kahan}(a, d, b, c)$

The value returned by algorithm $\mathcal{A}_3$ is $\widehat{z}_3^{(1)}(x, y)$ and, using (2.3) and the same reasoning as before, we can check that it shares the same normwise error bounds

as any of $\widehat{z}_3^{(\ell)}(x,y)$, $\ell = 2,3,4$. Hence it suffices to analyze algorithm $\mathcal{A}_3$ to deduce the error behavior of any of its three other variants.

2.3. **Error bounds for algorithms $\mathcal{A}_2$ and $\mathcal{A}_3$.** We remarked in the introduction that the error bound given in (1.2b) for algorithm $\mathcal{A}_2$ follows immediately from the error bound (1.2a) obtained in [8] for algorithm CHT. For the sake of completeness, we summarize this result in the following theorem:

Theorem 2.1. *For any β, p such that $\beta^{p-1} \geqslant 24$ and in the absence of underflow and overflow, algorithm $\mathcal{A}_2$ computes $\widehat{z}_2$ such that*

$$|\widehat{z}_2 - z| \leqslant \begin{cases} 2u|z| & \text{if } \beta \text{ is odd or RN is 'to nearest even',} \\ \frac{2\beta u + 2u^2}{\beta - 2u^2}|z| & \text{otherwise.} \end{cases}$$

Proof. By definition, algorithm $\mathcal{A}_2$ approximates $z = R + iI$ by $\widehat{z}_2 = \widehat{R}_2 + i\widehat{I}_2$, where, using [8, Theorem 1.1], $|\widehat{R}_2 - R| \leqslant \alpha|R|$ and $|\widehat{I}_2 - I| \leqslant \alpha|I|$ with $\alpha = 2u$ if β is odd or RN is 'to nearest even', and $\alpha = \frac{2\beta u + 2u^2}{\beta - 2u^2}$ otherwise. Hence $|\widehat{z}_2 - z| = \sqrt{\big(\widehat{R}_2 - R\big)^2 + \big(\widehat{I}_2 - I\big)^2} \leqslant \alpha\sqrt{R^2 + I^2} = \alpha|z|$. □

A similar error bound follows directly from [9, Theorem 1.2] for algorithm $\mathcal{A}_3$:

Theorem 2.2. *For any $\beta, p \geqslant 2$ and in the absence of underflow and overflow, algorithm $\mathcal{A}_3$ computes $\widehat{z}_3$ such that*

$$|\widehat{z}_3 - z| \leqslant 2u|z|.$$

3. Error bound for algorithm $\mathcal{A}_1$

In this section, we show that the relative error of the conventional algorithm with FMA is upper bounded by $2u$, for any radix and precision $\beta, p \geqslant 2$ and as long as underflows and overflows do not occur. We establish the following result:

Theorem 3.1. *For $\beta, p \geqslant 2$ and in the absence of underflow and overflow, algorithm $\mathcal{A}_1$ computes $\widehat{z}_1$ such that*

$$|\widehat{z}_1 - z| \leqslant 2u|z|.$$

To prove this theorem we rely on three lemmas that provide suitable bounds on the absolute errors $|\widehat{R}_1 - R|$ and $|\widehat{I}_1 - I|$.

Lemma 3.1. $|\widehat{R}_1 - R| \leqslant u|R| + u|bd| + u^2|bd|$.

Proof. Applying the standard model (2.2) to the real part $\widehat{R}_1$ of the result of algorithm $\mathcal{A}_1$ gives $\widehat{R}_1 = \big(ac - bd(1+\delta_1)\big)(1+\delta_2)$ with $|\delta_1|, |\delta_2| \leqslant u$. Since $R = ac - bd$, we deduce that $\widehat{R}_1 - R = R\delta_2 - bd\delta_1 - bd\delta_1\delta_2$. The result follows from the triangle inequality and the bounds on $|\delta_1|$ and $|\delta_2|$. □

Similarly, the imaginary part of $\widehat{z}_1$ satisfies $|\widehat{I}_1 - I| \leqslant u|I| + u|bc| + u^2|bc|$. The next two lemmas aim at removing the $O(u^2)$ terms in each of these bounds.

Lemma 3.2. *If $abcd \geqslant 0$, then $|\widehat{I}_1 - I| \leqslant u|I| + u|bc|$.*

Proof. Recalling that $I = ad + bc$ and $\widehat{I}_1 = \mathrm{RN}(g)$ with $g = ad + \mathrm{RN}(bc)$, we have

$$\begin{aligned}|\widehat{I}_1 - I| &\leqslant |\mathrm{RN}(g) - g| + |\mathrm{RN}(bc) - bc| \\ &\leqslant \tfrac{1}{2}\mathrm{ulp}(g) + u|bc|.\end{aligned}$$

Since $\frac{1}{2}\mathrm{ulp}(I) \leqslant u|I|$, the conclusion follows immediately when $\mathrm{ulp}(g) \leqslant \mathrm{ulp}(I)$. Assume now that $\mathrm{ulp}(I) < \mathrm{ulp}(g)$. In this case, $|I| < \beta^k \leqslant |g|$ for some integer k and, since $\beta^k \in \mathbb{F}$ and by definition of rounding to nearest, $|\mathrm{RN}(g) - g| \leqslant |g| - \beta^k < |g| - |I| \leqslant |g - I| = |\mathrm{RN}(bc) - bc| \leqslant u|bc|$. Therefore, $|\widehat{I}_1 - I| \leqslant 2u|bc|$. By assumption, the products ad and bc have the same sign, so that $|bc| \leqslant |I|$ and the conclusion follows. □

Lemma 3.3. *If* $|ac| \leqslant \frac{1}{2}|bd|$, *then* $|\widehat{R}_1 - R| \leqslant u|R| + u|bd|$.

Proof. Recall that $R = ac - bd$ and let $f = ac - \mathrm{RN}(bd)$. Then $\widehat{R}_1 = \mathrm{RN}(f)$ and

$$\begin{aligned}|\widehat{R}_1 - R| &\leqslant |\mathrm{RN}(f) - f| + |\mathrm{RN}(bd) - bd| \\ &\leqslant \tfrac{1}{2}\mathrm{ulp}(f) + u|bd|,\end{aligned}$$

so that the result is true as soon as $\mathrm{ulp}(f) \leqslant \mathrm{ulp}(R)$.

Assume now that $\mathrm{ulp}(R) < \mathrm{ulp}(f)$. This implies that $f \neq 0$ and that there exists an integer k such that

$$|R| < \beta^k \leqslant |f|. \tag{3.1}$$

Since $\beta^k \in \mathbb{F}$, the definition of RN leads to $|\mathrm{RN}(f) - f| \leqslant |f| - \beta^k$, and since

$$|f| \leqslant |R| + \tfrac{1}{2}\mathrm{ulp}(bd), \tag{3.2}$$

we deduce that $|\mathrm{RN}(f) - f| < |f| - |R| \leqslant \frac{1}{2}\mathrm{ulp}(bd)$. Consequently, the result is true as soon as $\mathrm{ulp}(bd) \leqslant \mathrm{ulp}(R)$ and we are left with dealing with the case where

$$\mathrm{ulp}(R) < \mathrm{ulp}(f) \qquad \text{and} \qquad \mathrm{ulp}(R) < \mathrm{ulp}(bd). \tag{3.3}$$

Using the assumption $|ac| \leqslant \frac{1}{2}|bd|$, we have $|bd| \leqslant |R| + |ac| \leqslant |R| + \frac{1}{2}|bd|$ and thus

$$|bd| \leqslant 2|R|. \tag{3.4}$$

Since $|f| \leqslant |R| + u|bd|$ by (3.2), we deduce that $|f| \leqslant (1+2u)|R| \leqslant \beta|R|$ for $\beta, p \geqslant 2$, and, therefore,

$$\mathrm{ulp}(f) \leqslant \beta\mathrm{ulp}(R). \tag{3.5}$$

On the other hand, $|bd| \leqslant 2|R|$ and $2 \leqslant \beta$ give $|bd| \leqslant \beta|R|$, so that

$$\mathrm{ulp}(bd) \leqslant \beta\mathrm{ulp}(R). \tag{3.6}$$

Hence, combining (3.3), (3.5), and (3.6) we have

$$\mathrm{ulp}(f) = \mathrm{ulp}(bd) = \beta\mathrm{ulp}(R).$$

Now, using $\mathrm{ulp}(f) = \mathrm{ulp}(bd)$ together with (3.1) and (3.2) gives $\beta^k \leqslant |f| < \beta^k + \frac{1}{2}\mathrm{ulp}(f)$, from which we deduce that

$$|\widehat{R}_1| = \beta^k.$$

Combining the latter equality with (3.1) and (3.2) and $\frac{1}{2}\mathrm{ulp}(bd) \leqslant u|bd|$, we obtain

$$|\widehat{R}_1| > |R| \geqslant |\widehat{R}_1| - u|bd|.$$

Furthermore, by Lemma 3.1 and the inequality in (3.4) we have $|\widehat{R}_1 - R| \leqslant u|R| + (u+u^2)|bd| \leqslant (3u+2u^2)|R|$. If $R = 0$, then the result is clearly true, while if $R \neq 0$ the latter inequality implies that

$$|\widehat{R}_1 - R| < |R|,$$

which ensures that $\widehat{R}_1$ and R have the same sign. Thus, overall, we arrive at $|\widehat{R}_1 - R| = |\widehat{R}_1| - |R| \leqslant u|bd|$, which concludes the proof. □

Using Lemmas 3.1, 3.2, and 3.3 we can now establish Theorem 3.1 as follows.

Proof of Theorem 3.1. We can assume without loss of generality that $abcd \geqslant 0$. This implies that $\operatorname{sign}(ac) = \operatorname{sign}(bd)$, from which it follows that

$$|R| = \big||ac| - |bd|\big|.$$

We now consider separately three cases, depending on how $|ac|$ relates to $|bd|$. In each case we show that $|\widehat{z}_1 - z| \leqslant 2u|z|$ by checking the equivalent inequality

$$Q \leqslant 4(R^2 + I^2), \tag{3.7}$$

where $Q = \frac{1}{u^2}\big((\widehat{R}_1 - R)^2 + (\widehat{I}_1 - I)^2\big)$ and $R^2 + I^2 = (ac)^2 + (bd)^2 + (ad)^2 + (bc)^2$.

■ If $|ac| \geqslant |bd|$, then $|R| = |ac| - |bd|$, so that Lemmas 3.1 and 3.2 imply that

$$\begin{aligned} Q &\leqslant (|ac| + u|bd|)^2 + (|I| + |bc|)^2 \\ &\leqslant (|ac| + |bd|)^2 + (|ad| + 2|bc|)^2 \\ &= (ac)^2 + (bd)^2 + (ad)^2 + 4(bc)^2 + 6\pi, \end{aligned}$$

where

$$\pi := abcd.$$

The inequality in (3.7) then follows from $2\pi \leqslant (ac)^2 + (bd)^2$.

■ If $\frac{1}{2}|bd| \leqslant |ac| < |bd|$, then $|R| = |bd| - |ac|$ and $\frac{1}{2}(bd)^2 \leqslant \pi$. Thus, applying Lemma 3.1 and Lemma 3.2, we obtain

$$\begin{aligned} Q &\leqslant (2|bd| - |ac| + u|bd|)^2 + (|ad| + 2|bc|)^2 \\ &= (ac)^2 + 4(bd)^2 + (ad)^2 + 4(bc)^2 + Q' + Q'', \end{aligned}$$

where $Q' = 4u(bd)^2$ and $Q'' = u^2(bd)^2 - 2\pi u$. Using $u^2 \leqslant u$ and the lower bound on π gives $Q'' \leqslant 0$. On the other hand, recalling that $u = \frac{1}{2}\beta^{1-p}$, we see that $u \leqslant \frac{1}{6}$ for all $\beta, p \geqslant 2$ such that $(\beta, p) \neq (2, 2)$. Hence in this case $Q' \leqslant \frac{2}{3}(bd)^2 \leqslant \frac{8}{3}(ac)^2 \leqslant 3(ac)^2$, from which (3.7) follows. In the case where $\beta = p = 2$, one has $u = \frac{1}{4}$, and it can be checked that $|\mathrm{RN}(bd) - bd| \leqslant \frac{1}{4}\mathrm{ulp}(bd) \leqslant \frac{u}{2}|bd|$. This implies that $Q' \leqslant 2(ac)^2$, and thus (3.7) follows as well.

■ If $|ac| < \frac{1}{2}|bd|$, then $|R| = |bd| - |ac|$ and, by Lemma 3.2 and Lemma 3.3,

$$\begin{aligned} Q &\leqslant (2|bd| - |ac|)^2 + (|ad| + 2|bc|)^2 \\ &= (ac)^2 + 4(bd)^2 + (ad)^2 + 4(bc)^2, \end{aligned}$$

which implies the inequality in (3.7). □

4. Asymptotic optimality of the error bounds for algorithms $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{A}_3$

We recalled in Section 2.3 that [8, Theorem 1.1] and [9, Theorem 1.2] imply relative error bounds of the form $2u + O(u^2)$ and $2u$ for algorithms $\mathcal{A}_2$ and $\mathcal{A}_3$, respectively. In the previous section we also obtained the bound $2u$ for algorithm $\mathcal{A}_1$. Here we provide certificates showing that these upper bounds are asymptotically optimal as $u \to 0$. Each certificate consists of a pair $(a, b) \in \mathbb{F} \times \mathbb{F}$ expressed explicitly in terms of β and p, and for which we prove that the relative error when evaluating the square $(a+ib)^2$ by a given multiplication algorithm is lower bounded by $2u - O(u^{1.5})$. Our proofs require only a mild assumption on the radix and precision, namely $\beta^{p-1} \geqslant 11$ (which is satisfied by all IEEE floating-point formats), and assume as before that underflows and overflows do not occur.

4.1. **Certificate for algorithms $\mathcal{A}_1$ and $\mathcal{A}_3$.** We provide in Theorem 4.1 a single certificate that applies to both algorithms $\mathcal{A}_1$ and $\mathcal{A}_3$, thus proving the asymptotic optimality of their common error bound $2u$. For this, we first establish the following technical lemma. Here and hereafter, $\lfloor \cdot \rfloor$ denotes the usual floor function.

Lemma 4.1. *For $\beta \geqslant 2$ and $p \geqslant 2$, let $n = \left\lfloor \sqrt{\frac{1}{2}\beta^{p-1}} \right\rfloor + 1$. Then*

(i) $\beta^{p-1} + n \leqslant \beta^p$;
(ii) $\beta^{p-1} + 2n < \beta^p$ *if* $\beta^{p-1} \geqslant 5$;
(iii) $\frac{1}{2}\beta^{p-1} < n^2 \leqslant \beta^{p-1}$ *if* $\beta^{p-1} \geqslant 9$.

Proof. Defining $N = \beta^{p-1}$, we have $\sqrt{N/2} < n \leqslant \sqrt{N/2} + 1$ and $N \geqslant 2$.

(i) Since $\beta \geqslant 2$, the announced inequality is implied by $N + \sqrt{N/2} + 1 \leqslant 2N$; that is, $\left(2\sqrt{N/2} + 1\right)\left(1 - \sqrt{N/2}\right) \leqslant 0$, which is true since $N \geqslant 2$.

(ii) The claimed inequality is implied by $N + 2\sqrt{N/2} + 2 < 2N$ or, equivalently, $\left(1 - \sqrt{N/2}\right)\sqrt{N/2} + 1 < 0$, which is easily seen to be true for all integers $N \geqslant 6$. When $N = 5$, we have $n = 2$ and thus $N + 2n < 2N \leqslant \beta N$, as wanted.

(iii) The lower bound follows from $\sqrt{N/2} < n$. The upper bound is implied by $-N/2 + 2\sqrt{N/2} + 1 \leqslant 0$, which holds for all integers $N \geqslant 12$. When $9 \leqslant N \leqslant 11$, we have $n = 3$, and it follows immediately that $n^2 \leqslant N$. □

Theorem 4.1. *Let $a, b \in \mathbb{F}$ be given by*

$$a = \operatorname{pred}\left(\sqrt{\tfrac{1}{2}\beta^{p-1}}\right), \qquad b = \beta^{p-1} + \left\lfloor \sqrt{\tfrac{1}{2}\beta^{p-1}} \right\rfloor + 1, \tag{4.1}$$

where, for $t \in \mathbb{R}_{>0}$, $\operatorname{pred}(t) = \max\{f \in \mathbb{F} : f < t\}$ denotes the predecessor of t in $\mathbb{F}$. Let also $\widehat{z}_1$ and $\widehat{z}_3$ be the approximations to $z = (a + ib)^2$ computed by algorithms $\mathcal{A}_1$ and $\mathcal{A}_3$, respectively. If $\beta^{p-1} \geqslant 10$, then, barring underflow and overflow,

$$|\widehat{z}_h/z - 1| > 2u - 8u^{1.5} - 4u^2, \qquad h \in \{1, 3\}.$$

Proof. Note first that both a and b are indeed in $\mathbb{F}$: for a this is true by definition, while for b this is a direct consequence of Lemma 4.1 (i).

Let us first prove the lower bound for algorithm $\mathcal{A}_1$. Writing R for the real part of z and $\widehat{R}_1$ for the real part of $\widehat{z}_1$, we have

$$|\widehat{z}_1/z - 1| \geqslant |\widehat{R}_1 - R|/|z|. \tag{4.2}$$

The rest of the proof consists of deriving suitable upper and lower bounds on, respectively, $|z|$ and $|\widehat{R}_1 - R|$. These bounds will be expressed in terms of $u = \frac{1}{2}\beta^{1-p}$.

By definition of a we have the strict inequality

$$a^2 < \tfrac{1}{2}\beta^{p-1} = \tfrac{1}{4}u^{-1} \tag{4.3}$$

and, by definition of b,

$$b^2 \leqslant \left(\tfrac{1}{2}u^{-1} + \tfrac{1}{2}u^{-1/2} + 1\right)^2 = \tfrac{1}{4}u^{-2} + \tfrac{1}{2}u^{-3/2} + \tfrac{5}{4}u^{-1} + u^{-1/2} + 1. \tag{4.4}$$

Applying (4.3) and (4.4) to $|z| = a^2 + b^2$ thus gives the upper bound

$$|z| < \tfrac{1}{4}u^{-2} + \tfrac{1}{2}u^{-3/2} + \tfrac{3}{2}u^{-1} + u^{-1/2} + 1. \tag{4.5}$$

Let us now derive a lower bound on $|\widehat{R}_1 - R|$. Defining $s = b^2$ and $\widehat{s} = \mathrm{RN}(s)$ we have

$$\widehat{R}_1 = \mathrm{RN}(a^2 - \widehat{s}) \qquad \text{and} \qquad R = a^2 - s.$$

Furthermore, with $n = \left\lfloor \sqrt{\frac{1}{2}\beta^{p-1}} \right\rfloor + 1$, we can write $b = \beta^{p-1} + n$ and then

$$s = f + g, \qquad f = (\beta^{p-1} + 2n)\beta^{p-1}, \qquad g = n^2.$$

Lemma 4.1 (ii) shows that f is in $\mathbb{F}$ and satisfies $\mathrm{ulp}(f) = \beta^{p-1}$. Then, applying Lemma 4.1 (iii) gives $\frac{1}{2}\mathrm{ulp}(f) < g \leqslant \mathrm{ulp}(f)$, so that rounding s to nearest produces

$$\widehat{s} = f + \mathrm{ulp}(f). \tag{4.6}$$

A first consequence of (4.6) is $\beta^{2p-2} < \widehat{s} \leqslant \beta^{2p-1}$, which together with (4.3) yields

$$\widehat{R}_1 = -\widehat{s}, \tag{4.7}$$

and then $|\widehat{R}_1 - R| = |\widehat{s} - s + a^2|$. Another consequence is that $\widehat{s} - s = \mathrm{ulp}(f) - g$ is nonnegative, which further implies

$$|\widehat{R}_1 - R| = \widehat{s} - s + a^2. \tag{4.8}$$

Third, it turns out that $\widehat{s} - s$ and a^2 are large enough to provide a useful lower bound on the relative error. More precisely,

$$\begin{aligned} \widehat{s} - s &= \beta^{p-1} - n^2 \\ &\geqslant \beta^{p-1} - \left(\sqrt{\tfrac{1}{2}\beta^{p-1}} + 1\right)^2 = \tfrac{1}{4}u^{-1} - u^{-1/2} - 1 \end{aligned} \tag{4.9}$$

and, since the definition of predecessor implies that $\mathrm{pred}(t) \geqslant t - \mathrm{ulp}(t)$ for $t > 0$,

$$\begin{aligned} a^2 &\geqslant \left(\sqrt{\tfrac{1}{2}\beta^{p-1}} - \mathrm{ulp}\left(\sqrt{\tfrac{1}{2}\beta^{p-1}}\right)\right)^2 \\ &\geqslant \tfrac{1}{2}\beta^{p-1}(1-2u)^2 = \tfrac{1}{4}u^{-1} - 1 + u. \end{aligned} \tag{4.10}$$

From (4.8), (4.9), and (4.10) we deduce that

$$|\widehat{R}_1 - R| \geqslant \tfrac{1}{2}u^{-1} - u^{-1/2} - 2 + u \qquad \text{for } \beta^{p-1} \geqslant 9. \tag{4.11}$$

Combining (4.2), (4.5), and (4.11) we have, overall,

$$|\widehat{z}_1/z - 1| > \tfrac{2u - 4u^{3/2} - 8u^2 + 4u^3}{1 + 2u^{1/2} + 6u + 4u^{3/2} + 4u^2} =: \varphi(u) \qquad \text{for } \beta^{p-1} \geqslant 9.$$

Furthermore, it is easily checked that $\varphi(u) \geqslant 2u - 8u^{3/2} - 4u^2$ and that the latter quantity is positive as soon as $\beta^{p-1} \geqslant 10$, which concludes the proof for $\mathcal{A}_1$.

Let us now show that the same lower bound on the normwise relative error also holds for the result $\widehat{z}_3$ computed by algorithm $\mathcal{A}_3$. The reason for this is that $\mathcal{A}_3$ produces in this example the same real part as $\mathcal{A}_1$: since the real part $\widehat{R}_3$ of $\widehat{z}_3$ is computed using Kahan's algorithm, we have

$$\widehat{R}_3 = \mathrm{RN}\big(\widehat{R}_1 + e\big)$$

with $e = \widehat{s} - s$. Hence using (4.7) we conclude that $\widehat{R}_3 = -\mathrm{RN}(s) = \widehat{R}_1$. □

The following corollary of Theorem 4.1 provides a certificate of asymptotic optimality for the relative error bound $2u$ of Kahan's algorithm, which holds independently of the parity of β and the tie-breaking strategy for RN. This is in contrast to the certificate in [9, Example 4.6], which assumes that β is even and that RN breaks ties to even.

Corollary 4.1. *Given $a, b \in \mathbb{F}$ as in (4.1), let $r = a^2 - b^2$, and let $\widehat{r}$ be the approximation to r computed by Kahan's algorithm. Barring underflow and overflow, if $\beta^{p-1} \geqslant 10$, then $|\widehat{r}/r - 1| > 2u - 8u^{1.5} - 4u^2$.*

Proof. Since r is the real part of $z = (a+ib)^2$ we have $|z| \geqslant |r|$; hence $|\widehat{R}_3/r - 1| \geqslant |\widehat{R}_3 - r|/|z|$. Since $\widehat{R}_1 = \widehat{R}_3$, the conclusion then follows from (4.5) and (4.11). □

4.2. **Certificate for algorithm $\mathcal{A}_2$.** We provide here a certificate showing that the relative error bound $2u + O(u^2)$ for algorithm $\mathcal{A}_2$ is asymptotically optimal. To prove this result, the following lemma will be used.

Lemma 4.2. *Let $x \in \mathbb{R}_{\geqslant 0}$ and $y \in \mathbb{R}_{>0}$. If $x < y - \frac{1}{2}\mathrm{ulp}(y)$, then $RN(x) < y$.*

Proof. From (2.1) we deduce that $\mathrm{RN}(x) \leqslant x + \frac{1}{2}\mathrm{ulp}(x)$. On the other hand, $0 \leqslant x < y$ and thus $\mathrm{ulp}(x) \leqslant \mathrm{ulp}(y)$. Hence $\mathrm{RN}(x) \leqslant x + \frac{1}{2}\mathrm{ulp}(y) < y$. □

Theorem 4.2. *Let $a, b \in \mathbb{F}$ be given by*

$$a = RD\Big(\big(1 - \tfrac{1}{2}\beta^{1-p}\big)\sqrt{\tfrac{1}{2}\beta^{p-1}}\Big), \qquad b = \beta^{p-1} + \left\lfloor \sqrt{\tfrac{1}{2}\beta^{p-1}} \right\rfloor + 1, \tag{4.12}$$

where, for $t \in \mathbb{R}$, $RD(t) = \max\{f \in \mathbb{F} : f \leqslant t\}$ denotes rounding down in $\mathbb{F}$. Let also $\widehat{z}_2$ be the approximation to $z = (a + ib)^2$ computed by algorithm $\mathcal{A}_2$. If $\beta^{p-1} \geqslant 11$, then, barring underflow and overflow,

$$|\widehat{z}_2/z - 1| > 2u - 8u^{1.5} - 6u^2.$$

Proof. The proof is organized in the same way as for Theorem 4.1: defining $\widehat{R}_2$ as the real part of $\widehat{z}_2$ and R as the real part of z, we combine the inequality $|\widehat{z}_2/z - 1| \geqslant |\widehat{R}_2 - R|/|z|$ with a lower bound on $|\widehat{R}_2 - R|$ and an upper bound on $|z|$.

For $|z| = a^2 + b^2$, an upper bound is again easily derived: from $u = \frac{1}{2}\beta^{1-p}$ and the definition of a we have

$$a^2 \leqslant \tfrac{(1-u)^2}{4} u^{-1}, \tag{4.13}$$

which together with the upper bound on b^2 already obtained in (4.4) leads to

$$|z| \leqslant \tfrac{1}{4}u^{-2} + \tfrac{1}{2}u^{-3/2} + \tfrac{3}{2}u^{-1} + u^{-1/2} + \tfrac{1}{2} + \tfrac{1}{4}u. \tag{4.14}$$

(Here, only the two rightmost summands differ from those in (4.5).)

For $|\widehat{R}_2 - R|$, however, obtaining a suitable lower bound is now more involved than in Theorem 4.1, essentially because of a more complicated expression for $\widehat{R}_2$. Let

$$s = b^2, \qquad \widehat{s} = \mathrm{RN}(b^2), \qquad e_1 = a^2 - \mathrm{RN}(a^2), \qquad e_2 = \widehat{s} - s.$$

Then $R = a^2 - s$ and by definition of algorithm $\mathcal{A}_2$ the real part of $\widehat{z}_2$ satisfies

$$\widehat{R}_2 = \mathrm{RN}\Big(\mathrm{RN}\big(\mathrm{RN}(a^2) - \widehat{s}\big) + \widehat{e}\Big), \qquad \widehat{e} = \mathrm{RN}(e_1 + e_2). \tag{4.15}$$

As already shown in the proof of Theorem 4.1 (where b is the same as here), if $\beta^{p-1} \geqslant 9$, then the floating-point number $\widehat{s}$ satisfies the following:

$$\widehat{s} \geqslant 0, \qquad \mathrm{ulp}(\widehat{s}) \geqslant \beta^{p-1}, \qquad \widehat{s} \neq \beta^{2p-2}, \tag{4.16a}$$

$$\widehat{s} - s \text{ is a positive integer such that } \widehat{s} - s < \tfrac{1}{2}\beta^{p-1}. \tag{4.16b}$$

We will now see that the quantities $\mathrm{RN}(a^2)$ and $|\widehat{e}|$ are smaller than $\frac{1}{2}\mathrm{ulp}(\widehat{s})$, thus implying that the first identity in (4.15) simplifies to $\widehat{R}_2 = -\widehat{s}$.

■ *Bounding* $\mathrm{RN}(a^2)$. From (4.13) we have $a^2 \leqslant x$ with $x = \frac{1}{2}\beta^{p-1} - \frac{1}{2} + \frac{u}{4}$ and thus, by rounding to nearest, $\mathrm{RN}(a^2) \leqslant \mathrm{RN}(x)$. On the other hand, setting $y = \frac{1}{2}\beta^{p-1}$, we deduce from $\beta \geqslant 2$ that y belongs to $[\beta^{p-2}, \beta^{p-1})$ and that $\mathrm{ulp}(y) = \beta^{-1} \leqslant 1/2$. Consequently, $x + \frac{1}{2}\mathrm{ulp}(y) \leqslant \frac{1}{2}\beta^{p-1} - \frac{1}{4} + \frac{u}{4} < y$ for $\beta \geqslant 2$ and $p \geqslant 2$. Applying Lemma 4.2 then gives $\mathrm{RN}(x) < y$, and we conclude that

$$\mathrm{RN}(a^2) < \tfrac{1}{2}\beta^{p-1}. \tag{4.17}$$

■ *Bounding* $|\widehat{e}|$. First, by using the properties of the RN function we see that $|\widehat{e}| = \mathrm{RN}(|e_1 + e_2|) \leqslant \mathrm{RN}(|e_1| + |e_2|)$. Then, recalling (4.13) and by definition of e_1, we have $|e_1| \leqslant ua^2 \leqslant \frac{1}{4}(1-u)^2$. Third, it follows from (4.16b) that $|e_2|$ is an integer such that $|e_2| < \frac{1}{2}\beta^{p-1}$; since β^{p-1} is also an integer, this strict inequality implies that $|e_2| \leqslant \frac{1}{2}\beta^{p-1} - \frac{1}{2}$. Therefore, by adding these bounds on $|e_1|$ and $|e_2|$ and by rounding to nearest, we obtain $|\widehat{e}| \leqslant \mathrm{RN}(x')$ with $x' = \frac{1}{2}\beta^{p-1} - \frac{1}{4} - \frac{u}{2} + \frac{u^2}{4}$. Taking $y = \frac{1}{2}\beta^{p-1}$ as in the previous paragraph, we can check that $x' < y - \frac{1}{2}\mathrm{ulp}(y)$, which by Lemma 4.2 implies that $\mathrm{RN}(x') < y$ and then

$$|\widehat{e}| < \tfrac{1}{2}\beta^{p-1}. \tag{4.18}$$

From (4.16a) and (4.17) it follows that $\mathrm{RN}\big(\mathrm{RN}(a^2) - \widehat{s}\big)$ equals $-\widehat{s}$, so that $\widehat{R}_2 = \mathrm{RN}(-\widehat{s} + \widehat{e})$. Applying (4.16a) and (4.18) to the latter identity gives further that

$$\widehat{R}_2 = -\widehat{s}.$$

Since $\widehat{s} - s$ is nonnegative by (4.16b), we deduce that $|\widehat{R}_2 - R| = \widehat{s} - s + a^2$. Furthermore, the definition of a yields

$$\begin{aligned} a^2 &\geqslant (1-2u)^2\big(1 - \tfrac{1}{2}\beta^{1-p}\big)^2 \tfrac{1}{2}\beta^{p-1} \\ &= \tfrac{1}{4}u^{-1} - \tfrac{3}{2} + \tfrac{13}{4}u - 3u^2 + u^3, \end{aligned}$$

which together with (4.9) leads to

$$|\widehat{R}_2 - R| \geqslant \tfrac{1}{2}u^{-1} - u^{-1/2} - \tfrac{5}{2} + \tfrac{13}{4}u - 3u^2 + u^3 \qquad \text{for } \beta^{p-1} \geqslant 9. \tag{4.19}$$

From (4.14) and (4.19) it follows that if $\beta^{p-1} \geqslant 9$, then the relative error $|\widehat{z}_2/z - 1|$ is lower bounded by a rational function $\psi(u)$ which is easily seen to be larger than $2u - 8u^{3/2} - 6u^2$. The latter quantity is positive for $\beta^{p-1} \geqslant 11$, which concludes the proof. □

It turns out that the certificate introduced in Theorem 4.2 for algorithm $\mathcal{A}_2$ can also be used to show that when evaluating $a^2 - b^2$ with CHT or $(a+ib)^2$ with $\mathcal{A}_0$, the relative error can have the form $2u - O(u^{1.5})$.

Corollary 4.2. *Given $a, b \in \mathbb{F}$ as in (4.12), let $r = a^2 - b^2$ and $z = (a+ib)^2$. Let also $\widehat{r}$ and $\widehat{z}_0$ be the approximations to r and z computed by algorithms CHT and $\mathcal{A}_0$. If $\beta^{p-1} \geqslant 11$, then, barring underflow and overflow, both $|\widehat{r}/r - 1|$ and $|\widehat{z}_0/z - 1|$ are larger than $2u - 8u^{1.5} - 6u^2$.*

Proof. For such a and b, we have seen in the proof of Theorem 4.2 that the correction term $\widehat{e}$ in (4.15) has no effect on the initial approximation $\widehat{R}_0 = \mathrm{RN}\big(\mathrm{RN}(a^2) - \mathrm{RN}(b^2)\big)$, so that $\widehat{R}_2 = \widehat{R}_0$. Note also that $\widehat{R}_0 = \mathrm{Re}\,\widehat{z}_0$, $\widehat{r} = \widehat{R}_2$, and $r = R = \mathrm{Re}\,z$. Hence $|\widehat{r}/r - 1| \geqslant |\widehat{r} - r|/|z| = |\widehat{R}_2 - R|/|z|$ and, on the other hand, $|\widehat{z}_0/z - 1| \geqslant |\widehat{R}_0 - R|/|z| = |\widehat{R}_2 - R|/|z|$. In each case the conclusion follows from reusing the fact that (4.14) and (4.19) imply that $|\widehat{R}_2 - R|/|z| > 2u - 8u^{1.5} - 6u^2$. □

This corollary shows that the relative error bound $2u + O(u^2)$ is asymptotically optimal for algorithm CHT. When $\beta = 2$, it thus provides an alternative to the proof given in [15, §4], which takes $a = c = 2^p - 1$, $b = 2^{p-3} + 1/2$, and $d = 2^{p-3} + 1/4$ (and breaks ties to an even last bit).

5. Conclusion

It has been shown that the availability of an FMA instruction makes it possible to improve the normwise relative error in computing a complex floating-point product from a bound of $\sqrt{5}u$ to a bound of $2u$. We have also shown that the term $2u$ is best possible not only for the basic algorithm $\mathcal{A}_1$ but also for the compensated versions $\mathcal{A}_2$ and $\mathcal{A}_3$, and even in the particular case of squaring.

The table below summarizes the bounds now available for these three algorithms as well as those given in [3] for algorithm $\mathcal{A}_0$, which makes no use of the FMA. Recall that here the lower bounds and the upper bounds hold under the respective conditions $\beta^{p-1} \geqslant 11$ and $\beta^{p-1} \geqslant 24$ (which are satisfied in all practical cases). Let us also recall that the upper bound $\frac{2\beta u + 2u^2}{\beta - 2u^2} = 2u + O(u^2)$ for algorithm $\mathcal{A}_2$ can be replaced by $2u$ when the default rounding direction attribute 'to nearest even' is used.

	lower bound on largest normwise error	upper bound on largest normwise error	flop count	properties: P1	P2
$\mathcal{A}_0$	$\sqrt{5}\,u - O(u^2)$ if $\beta = 2$	$\sqrt{5}\,u$	6	yes	yes
$\mathcal{A}_1$	$2u - 8u^{1.5} - 4u^2$	$2u$	4	no	no
$\mathcal{A}_2$	$2u - 8u^{1.5} - 6u^2$	$\frac{2\beta u + 2u^2}{\beta - 2u^2}$	14	yes	yes
$\mathcal{A}_3$	$2u - 8u^{1.5} - 4u^2$	$2u$	8	no	yes

This table also displays in each case the number of floating-point operations (flops) used, and whether the following two basic properties of complex multiplication are preserved or not:

(P1) $x, y \in \mathbb{C} \quad \Rightarrow \quad x\,y = y\,x$ (commutativity);

(P2) $x \in \mathbb{C} \quad \Rightarrow \quad x\,\overline{x} \in \mathbb{R}$,

where $\overline{x}$ denotes the conjugate of x. (Further details showing why these properties are indeed preserved or not can be found in Appendix A.)

If we are only interested in reducing the *normwise* relative error, just using the conventional algorithm $\mathcal{A}_1$ suffices to achieve the error bound $2u$. When a small *componentwise* error is needed, then either algorithm $\mathcal{A}_2$ or algorithm $\mathcal{A}_3$ must be used, since unlike $\mathcal{A}_0$ and $\mathcal{A}_1$ they ensure high relative accuracy for both the real and imaginary parts of the computed product. If in addition properties P1 and P2 are essential, then algorithm $\mathcal{A}_2$ is the only choice; otherwise, one might prefer algorithm $\mathcal{A}_3$, which is cheaper in terms of flops.

Finally, two further remarks can be made, namely about squaring and division:

Algorithm $\mathcal{A}_0$ and complex squaring. We have seen in Section 4 that the error term $2u$ associated with algorithms $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$ is already sharp when *squaring* a complex number, that is, when evaluating $(a + ib)^2$ instead of a general product $(a + ib)(c + id)$. A natural question is whether this is also the case for the bound $\sqrt{5}u$ associated with algorithm $\mathcal{A}_0$. The answer is 'no', at least when $\beta = 2$: when squaring a complex number with algorithm $\mathcal{A}_0$, the normwise error bound $\sqrt{5}u$ of Brent, Percival, and Zimmermann [3] can be reduced further to $2u$, and this new bound turns out to be asymptotically optimal; see Appendix B for a detailed proof.

Application to complex division. A direct application of our error bounds is to complex *division*. As noted by Baudin in [1, p. 25], if the quotient x/y is evaluated using the conventional formula $x/y = (x\,\overline{y})/(y\,\overline{y})$, and if the multiplication algorithm used to evaluate the numerator $x\,\overline{y}$ has its normwise relative error bounded by $\lambda u + O(u^2)$, then the normwise relative error of division is bounded by $B = (3 + \lambda)u + O(u^2)$. Without an FMA, using algorithm $\mathcal{A}_0$ gives $\lambda = \sqrt{5}$ and, therefore, $B < 5.237\,u + O(u^2)$. In contrast, if we use any of the FMA-based algorithms $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, then $B = 5u + O(u^2)$.

Appendix A. Properties of complex multiplication

Complex multiplication satisfies the following two basic properties:

(P1) $x, y \in \mathbb{C} \quad \Rightarrow \quad x\,y = y\,x$ (commutativity);

(P2) $x \in \mathbb{C} \quad \Rightarrow \quad x\,\overline{x} \in \mathbb{R}$,

where $\overline{x}$ denotes the conjugate of x. The following table indicates whether these properties are preserved or not by the four complex multiplication algorithms considered in this paper:

	P1	P2
$\mathcal{A}_0$	yes	yes
$\mathcal{A}_1$	no	no
$\mathcal{A}_2$	yes	yes
$\mathcal{A}_3$	no	yes

Note first that both properties are clearly preserved by algorithm $\mathcal{A}_0$: for all $a, b, c, d \in \mathbb{F}$, we have $\mathcal{A}_0(a+ib, c+id) = \mathcal{A}_0(c+id, a+ib)$ simply because addition and multiplication over $\mathbb{R}$ are commutative operations; also, $\mathcal{A}_0(a+ib, a-ib) \in \mathbb{F}$ because $\mathrm{RN}(-t) = -\mathrm{RN}(t)$ for all real t. Algorithm $\mathcal{A}_2$ satisfies P1 and P2 for the same reasons.

Property P2 is lost for algorithm $\mathcal{A}_1$, since $\mathcal{A}_1(a+ib, a-ib)$ has its imaginary part equal to $\mathrm{RN}\big(\mathrm{RN}(ab) - ab\big)$, which is in general nonzero. On the other hand, we know from [9, Theorem 1.2] that $\mathrm{Kahan}(a, -b, b, a)$ returns zero, which implies that P2 is preserved by algorithm $\mathcal{A}_3$.

Finally, property P1 is lost for both algorithms $\mathcal{A}_1$ and $\mathcal{A}_3$. Indeed, there exist floating-point numbers a, b, c, d such that $\mathrm{RN}\big(ab + \mathrm{RN}(cd)\big) \neq \mathrm{RN}\big(\mathrm{RN}(ab) + cd\big)$ and $\mathrm{Kahan}(a, b, c, d) \neq \mathrm{Kahan}(c, d, a, b)$, as illustrated by the following example.

Example A.1. Let $a, b, c, d \in \mathbb{F}$ be defined by

$$a = \beta^{p-1}, \qquad b = c = a + 1, \qquad d = \begin{cases} \beta^p - \frac{\beta}{2} & \text{if } \beta \text{ is even,} \\ \beta^p - \frac{\beta+1}{2} & \text{if } \beta \text{ is odd.} \end{cases}$$

Assuming $p \geqslant 3$, it can be checked that $\mathrm{RN}(ab + \mathrm{RN}(cd))$ and $\mathrm{Kahan}(a, b, c, d)$ are both equal to $\beta^{2p-1} + \beta^{2p-2}$, while $\mathrm{RN}(\mathrm{RN}(ab) + cd)$ and $\mathrm{Kahan}(c, d, a, b)$ are both equal to $\beta^{2p-1} + \beta^{2p-2} + \beta^p$. In this example, a tie occurs when rounding $ab + \mathrm{RN}(cd)$ to nearest, and the standard 'round to the nearest even' tie-breaking rule (roundTiesToEven rounding attribute in [7]) is then used.

Appendix B. Accuracy of squaring with algorithm $\mathcal{A}_0$ in radix 2

When evaluating the complex square $z = (a+ib)^2$ with a, b in $\mathbb{F}$, algorithm $\mathcal{A}_0$ returns $\widehat{z}_0 = \widehat{R}_0 + i\,\widehat{I}_0$ with $\widehat{R}_0 = \mathrm{RN}\big(\mathrm{RN}(a^2) - \mathrm{RN}(b^2)\big)$ and $\widehat{I}_0 = \mathrm{RN}(2\mathrm{RN}(ab))$. If $\beta = 2$, then $\widehat{I}_0 = 2\mathrm{RN}(ab)$, so that only one rounding error is committed when evaluating the imaginary part. The theorem below shows that in this special case the bound $\sqrt{5}\,u$ of Brent, Percival, and Zimmermann [3] can be reduced to $2u$.

Theorem B.1. *Assume $\beta = 2$ and, given a and b in $\mathbb{F}$, let $z = (a+ib)^2$. Then, for $p \geqslant 2$ and in the absence of underflow and overflow, algorithm $\mathcal{A}_0$ computes $\widehat{z}_0$ such that*

$$|\widehat{z}_0 - z| \leqslant 2u|z|,$$

and this relative error bound is asymptotically optimal.

Proof. Note first that if a and b are swapped, then only the signs of the real (exact and computed) parts change, so the error $|\widehat{z}_0 - z|$ remains the same. Hence we can assume $a^2 \geqslant b^2$. Defining $s = \mathrm{RN}(a^2) - \mathrm{RN}(b^2)$, we have

$$|\widehat{R}_0 - R| \leqslant |\mathrm{RN}(s) - s| + |\mathrm{RN}(a^2) - a^2| + |\mathrm{RN}(b^2) - b^2|. \tag{B.1}$$

If $s \in \mathbb{F}$, then (B.1) gives $|\widehat{R}_0 - R| \leqslant u(a^2 + b^2)$. Since $|\widehat{I}_0 - I| \leqslant 2u|ab|$ and $|z|^2 = (a^2+b^2)^2$, it follows that

$$\begin{aligned} |\widehat{z}_0 - z|^2 &\leqslant u^2(a^4 + 6a^2b^2 + b^4) \\ &\leqslant 2u^2|z|^2. \end{aligned}$$

If $s \notin \mathbb{F}$, then, since $a^2 \geqslant b^2$ leads to $\mathrm{RN}(a^2) \geqslant \mathrm{RN}(b^2) \geqslant 0$ and $0 \leqslant s \leqslant \mathrm{RN}(a^2)$, we have $s < \mathrm{RN}(a^2)$. This strict inequality implies $\mathrm{ulp}(s) \leqslant \mathrm{ulp}(a^2)$. Hence (B.1)

now gives $|\widehat{R}_0 - R| \leqslant u(2a^2 + b^2)$, so that

$$\begin{aligned} |\widehat{z}_0 - z|^2 &\leqslant u^2\big(4a^4 + 8a^2b^2 + b^4\big) \\ &\leqslant 4u^2|z|^2. \end{aligned}$$

Thus, in both cases we have $|\widehat{z}_0 - z| \leqslant 2u|z|$.

The asymptotic optimality of this bound follows from Corollary 4.2, which says that for $\beta = 2$ and $p \geqslant 5$, there exist a, b in $\mathbb{F}$ such that $|\widehat{z}_0/z - 1|$ is lower bounded by $2u - 8u^{1.5} - 6u^2$. □

Acknowledgments

This research was supported in part by the French National Research Agency under grants ANR-11-BS02-013 (HPAC project), ANR-2010-BLAN-0203-01 (TaMaDi project), and ANR-13-INSE-0007 (MetaLibm project).

References

[1] M. Baudin, *Error bounds of complex arithmetic*, June 2011, available at `http://forge.scilab.org/upload/compdiv/files/complexerrorbounds_v0.2.pdf`.

[2] S. Boldo, *Pitfalls of a full floating-point proof: example on the formal proof of the Veltkamp/Dekker algorithms*, Automated Reasoning, Lecture Notes in Comput. Sci., vol. 4130, Springer, Berlin, 2006, pp. 52–66, DOI 10.1007/11814771_6. MR2354672

[3] R. Brent, C. Percival, and P. Zimmermann, *Error bounds on complex floating-point multiplication*, Math. Comp. **76** (2007), no. 259, 1469–1481 (electronic), DOI 10.1090/S0025-5718-07-01931-X. MR2299783 (2008b:65062)

[4] M. Cornea, J. Harrison, and P. T. P. Tang, *Scientific Computing on Itanium®-based Systems*, Intel Press, Hillsboro, OR, USA, 2002.

[5] T. J. Dekker, *A floating-point technique for extending the available precision*, Numer. Math. **18** (1971/72), 224–242. MR0299007 (45 #8056)

[6] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd ed., Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002. MR1927606 (2003g:65064)

[7] IEEE Computer Society, *IEEE Standard for Floating-Point Arithmetic*, IEEE Standard 754-2008, August 2008, available at `http://ieeexplore.ieee.org/servlet/opac?punumber=4610933`.

[8] C.-P. Jeannerod, *A radix-independent error analysis of the Cornea-Harrison-Tang method*, ACM Trans. Math. Software **42** (2016), no. 3, Art. 19, 20 pp.

[9] C.-P. Jeannerod, N. Louvet, and J.-M. Muller, *Further analysis of Kahan's algorithm for the accurate computation of* 2×2 *determinants*, Math. Comp. **82** (2013), no. 284, 2245–2264, DOI 10.1090/S0025-5718-2013-02679-8. MR3073198

[10] W. Kahan, *Further remarks on reducing truncation errors*, Communications of the ACM **8** (1965), no. 1, 40.

[11] S. Linnainmaa, *Analysis of some known methods of improving the accuracy of floating-point sums*, Nordisk Tidskr. Informationsbehandling (BIT) **14** (1974), 167–202. MR0483373 (58 #3381)

[12] S. Linnainmaa, *Software for doubled-precision floating-point computations*, ACM Trans. Math. Software **7** (1981), no. 3, 272–283, DOI 10.1145/355958.355960. MR630437 (82h:68041)

[13] O. Møller, *Quasi double-precision in floating point addition*, Nordisk Tidskr. Informationsbehandling (BIT) **5** (1965), 37–50. MR0181130 (31 #5359)

[14] O. Møller, *Note on quasi double-precision*, Nordisk Tidskr. Informationsbehandling (BIT) **5** (1965), 251–255.

[15] J.-M. Muller, *On the error of computing* $ab + cd$ *using Cornea, Harrison and Tang's method*, ACM Trans. Math. Software **41** (2015), no. 2, Art. 7, 8, DOI 10.1145/2629615. MR3318079

[16] J.-M. Muller, N. Brisebarre, F. de Dinechin, C.-P. Jeannerod, V. Lefèvre, G. Melquiond, N. Revol, D. Stehlé, and S. Torres, *Handbook of Floating-Point Arithmetic*, Birkhäuser Boston, Inc., Boston, MA, 2010. MR2568265

[17] M. Pichat, *Correction d'une somme en arithmétique à virgule flottante* (French, with English summary), Numer. Math. **19** (1972), 400–406. MR0324892 (48 #3241)
[18] M. Pichat, *Contributions à l'étude des erreurs d'arrondi en arithmétique à virgule flottante*, Ph.D. thesis, Université Scientifique et Médicale de Grenoble, Grenoble, France, 1976.

Inria, Laboratoire LIP (CNRS, ENS de Lyon, Inria, UCBL), Université de Lyon, 46, allée d'Italie, 69364 Lyon cedex 07, France
E-mail address: claude-pierre.jeannerod@inria.fr

Department of Mathematics and Computer Science, University of Southern Denmark, Campusvej 55, DK-5230 Odense M, Denmark
E-mail address: kornerup@imada.sdu.dk

UCBL, Laboratoire LIP (CNRS, ENS de Lyon, Inria, UCBL), Université de Lyon, 46, allée d'Italie, 69364 Lyon cedex 07, France
E-mail address: nicolas.louvet@ens-lyon.fr

CNRS, Laboratoire LIP (CNRS, ENS de Lyon, Inria, UCBL), Université de Lyon, 46, allée d'Italie, 69364 Lyon cedex 07, France
E-mail address: jean-michel.muller@ens-lyon.fr