On the relative error of computing complex square roots in floating-point arithmetic

Claude-Pierre Jeannerod, Jean-Michel Muller

On the relative error of computing complex square roots in floating-point arithmetic

Claude-Pierre Jeannerod* and Jean-Michel Muller†

*Univ Lyon, Inria, CNRS, ENS de Lyon, Université Claude Bernard Lyon 1, LIP UMR 5668, F-69007 LYON, France,
†Univ Lyon, CNRS, ENS de Lyon, Inria, Université Claude Bernard Lyon 1, LIP UMR 5668, F-69007 LYON, France

***Abstract*—We study the accuracy of a classical approach to computing complex square-roots in floating-point arithmetic. Our analyses are done in binary floating-point arithmetic in precision p, and we assume that the (real) arithmetic operations $+$, $-$, $\times$, $\div$, $\sqrt{}$ are rounded to nearest, so the unit roundoff is $u = 2^{-p}$. We show that in the absence of underflow and overflow, the componentwise and normwise relative errors of this approach are at most $\frac{7}{2}u$ and $\frac{\sqrt{37}}{2}u$, respectively, and this without having to neglect terms of higher order in u. We then provide some input examples showing that these bounds are reasonably sharp for the three basic binary interchange formats (binary32, binary64, and binary128) of the IEEE 754 standard for floating-point arithmetic.**

***Index Terms*—binary floating-point arithmetic; rounding error analysis; relative error; complex square root**

I. Introduction

We consider the problem of computing a square root of a complex number $a + ib$ accurately in floating-point arithmetic: given two floating-point numbers a and b, we want to deduce very good floating-point approximations to some reals x and y such that

$$(x + iy)^2 = a + ib. \tag{1}$$

In exact arithmetic, explicit formulas for x and y are easy to derive: first, by rewriting (1) as

$$x^2 - y^2 = a \qquad \text{and} \qquad 2xy = b,$$

and solving quadratic equations in $x^2 \geqslant 0$ or $y^2 \geqslant 0$, we obtain

$$x = \pm\sqrt{\frac{h+a}{2}}, \qquad h := \sqrt{a^2+b^2}, \tag{2}$$

and

$$y = \pm\sqrt{\frac{h-a}{2}}. \tag{3}$$

Then it suffices to adjust the signs of x and y in order to ensure that $2xy = b$ holds and to make the complex square root a single-valued function. For example, one can take $x \geqslant 0$ and $\operatorname{sign}(y) = \operatorname{sign}(b)$ with $\operatorname{sign}(0) = +1$; see [2, §4.2]. (See also [4, p. 201] for a sign function supporting signed zeros.)

In floating-point arithmetic, however, it is in general not recommended to use the above formulas for x and y simultaneously when $b^2 \ll a^2$, since then cancellation can occur either when evaluating $h + a$ if $a < 0$, or when evaluating $h - a$ if $a > 0$.

To avoid such a possible loss of accuracy, Friedland [1] proposed the following approach (which is now classical and can also be seen in [4] and [2]):

- if $a \geqslant 0$, then compute x using (2) and deduce y using
$$y = \frac{b}{2x};$$
- if $a < 0$, then compute y using (3) and deduce x using
$$x = \frac{b}{2y}.$$

Note that in the above expressions division by zero can be avoided by assuming that $(a, b) \neq (0, 0)$ and by handling the situation where $a = b = 0$ separately.

Since $h - a = h + |a|$ when $a < 0$, we see that the two cases in Friedland's approach eventually rely on a single core computation, which can be summarized as follows: given two floating-point numbers a and b such that

$$(a, b) \neq (0, 0) \qquad \text{and} \qquad a \geqslant 0,$$

evaluate

$$h = \sqrt{a^2+b^2}, \qquad x = \sqrt{\frac{h+a}{2}}, \qquad y = \frac{b}{2x}.$$

In radix-2, precision-p floating-point arithmetic with rounding to nearest (RN), this corresponds to Algorithm 1 below.

Algorithm 1 Core computation of $\sqrt{a+ib}$, assuming $(a, b) \neq (0, 0)$ and $a \geqslant 0$.

1: $s_a \leftarrow \mathrm{RN}(a^2)$
2: $s_b \leftarrow \mathrm{RN}(b^2)$
3: $s \leftarrow \mathrm{RN}(s_a + s_b)$
4: $\rho \leftarrow \mathrm{RN}(\sqrt{s})$
5: $\nu \leftarrow \mathrm{RN}(\rho + a)$
6: $\widehat{x} \leftarrow \mathrm{RN}\big(\sqrt{\nu/2}\big)$
7: $\widehat{y} \leftarrow \mathrm{RN}\big(b/(2\widehat{x})\big)$

A detailed rounding error analysis of Algorithm 1 is given by Hull, Fairgrieve, and Tang in [2]: assuming that underflows and overflows do not occur and using the fact that for any real number t,

$$\mathrm{RN}(t) = t(1+\delta), \qquad |\delta| \leqslant u := 2^{-p}, \tag{4}$$

they show that the computed floating-point numbers $\widehat{x}$ and $\widehat{y}$ satisfy

$$\frac{|\widehat{x}-x|}{|x|} \leqslant \frac{5}{2}u + \mathcal{O}(u^2)$$

and

$$\frac{|\widehat{y}-y|}{|y|} \leqslant \frac{7}{2}u + \mathcal{O}(u^2);$$

they also show that for $\widehat{z} = \widehat{x} + i\widehat{y}$ and $z = x + iy$, the associated normwise relative error $|\widehat{z}-z|/|z|$ admits a bound smaller than $\frac{7}{2}u + \mathcal{O}(u^2)$, namely,

$$\frac{|\widehat{z}-z|}{|z|} \leqslant \frac{\sqrt{37}}{2}u + \mathcal{O}(u^2), \qquad \frac{\sqrt{37}}{2} = 3.041\ldots.$$

Finally, for the binary32 format ($p = 24$), they provide two numbers a and b for which $|\widehat{z}-z|/|z| \approx 2.980u$.

In this paper, we refine the analysis of [2] in two ways: we show that the terms $\mathcal{O}(u^2)$ in the three bounds above can be removed and, on the other hand, we give examples of inputs in the binary64 and binary128 formats (that is, for $p = 53$ and $p = 113$) for which $|\widehat{z}-z|/|z| > 3u$.

For our analyses it will be useful to exploit the following refinement of (4), which can be found for example in [5, p. 232]:

$$\mathrm{RN}(t) = t(1+\delta), \qquad |\delta| \leqslant \frac{u}{1+u}. \tag{5}$$

We shall apply (5) to floating-point additions and multiplications; for floating-point divisions and square roots, we can use the following smaller bounds, introduced in [3]. Let a and b be two floating-point numbers. If $a \geqslant 0$, then

$$\mathrm{RN}(\sqrt{a}) = \sqrt{a}\,(1+\delta), \qquad |\delta| \leqslant 1 - \frac{1}{\sqrt{1+2u}}; \tag{6}$$

if $b \neq 0$, then

$$\mathrm{RN}\Big(\frac{a}{b}\Big) = \frac{a}{b}\,(1+\delta), \qquad |\delta| \leqslant u - 2u^2. \tag{7}$$

As we shall see in §II, the bounds in (5–7) are enough to show that $|\widehat{x}-x| \leqslant \frac{5}{2}u|x|$. However, our analysis for $\widehat{y}$ will use some variants of (6) and (7), which we detail in §III. We conclude in §IV with the derivation of the normwise bound and three numerical examples.

II. Refining the bound on $|\widehat{x}-x|/|x|$

First, let us apply (5) to steps 1, 2, 3 of Algorithm 1: we have

$$a^2\left(1-\frac{u}{1+u}\right) \leqslant s_a \leqslant a^2\left(1+\frac{u}{1+u}\right)$$

and similarly for s_b, so that

$$(a^2+b^2)\left(1-\frac{u}{1+u}\right) \leqslant s_a+s_b \leqslant (a^2+b^2)\left(1+\frac{u}{1+u}\right)$$

and then

$$(a^2+b^2)\left(1-\frac{u}{1+u}\right)^2 \leqslant s \leqslant (a^2+b^2)\left(1+\frac{u}{1+u}\right)^2.$$

By taking square roots and with $h = \sqrt{a^2+b^2}$, we find

$$h\left(1-\frac{u}{1+u}\right) \leqslant \sqrt{s} \leqslant h\left(1+\frac{u}{1+u}\right).$$

Using (6), we deduce that the value of $\rho = \mathrm{RN}(\sqrt{s})$ at step 4 of Algorithm 1 satisfies

$$hL \leqslant \rho \leqslant hU,$$

where

$$\begin{aligned} L &:= \left(1-\frac{u}{1+u}\right)\cdot\frac{1}{\sqrt{1+2u}} \\ &= 1 - 2u + \frac{7}{2}u^2 + \mathcal{O}(u^3) \end{aligned}$$

and

$$\begin{aligned} U &:= \left(1+\frac{u}{1+u}\right)\left(2-\frac{1}{\sqrt{1+2u}}\right) \\ &= 1 + 2u - \frac{3}{2}u^2 + \mathcal{O}(u^3). \end{aligned}$$

Since $a \geqslant 0$ and $0 \leqslant L \leqslant 1 \leqslant U$, this leads to

$$(h+a)L \leqslant \rho + a \leqslant (h+a)U.$$

By applying (5), we see that $\nu = \mathrm{RN}(\rho + a)$ at step 5 satisfies

$$\begin{aligned} &(h+a)\left(1-\tfrac{u}{1+u}\right)^2\cdot\tfrac{1}{\sqrt{1+2u}} \\ &\leqslant \nu \\ &\leqslant (h+a)\left(1+\tfrac{u}{1+u}\right)^2\left(2-\tfrac{1}{\sqrt{1+2u}}\right). \end{aligned}$$

Recalling that $x = \sqrt{(h+a)/2}$, it follows that $\sqrt{\nu/2}$ satisfies

$$\begin{aligned} &x\left(1-\tfrac{u}{1+u}\right)\cdot\tfrac{1}{(1+2u)^{1/4}} \\ &\leqslant \sqrt{\nu/2} \\ &\leqslant x\left(1+\tfrac{u}{1+u}\right)\left(2-\tfrac{1}{\sqrt{1+2u}}\right)^{1/2}. \end{aligned}$$

By applying (6) once again, we find that the value $\widehat{x} = \mathrm{RN}\big(\sqrt{\nu/2}\big)$ produced at step 6 satisfies

$$xL' \leqslant \widehat{x} \leqslant xU', \tag{8}$$

where

$$\begin{aligned} L' &:= \left(1-\frac{u}{1+u}\right)\cdot\frac{1}{(1+2u)^{3/4}} \\ &= 1 - \frac{5}{2}u + \frac{41}{8}u^2 + \mathcal{O}(u^3) \end{aligned} \tag{9}$$

and

$$\begin{aligned} U' &:= \left(1+\frac{u}{1+u}\right)\left(2-\frac{1}{\sqrt{1+2u}}\right)^{3/2} \\ &= 1 + \frac{5}{2}u - \frac{11}{8}u^2 + \mathcal{O}(u^3). \end{aligned}$$

Since $L' \geqslant 1 - \frac{5}{2}u$ and $U' \leqslant 1 + \frac{5}{2}u$, we conclude that

$$|\widehat{x}-x| \leqslant \frac{5}{2}u|x|.$$

III. Refining the bound on $|\widehat{y} - y|/|y|$

Let us now analyze the relative accuracy of the value $\widehat{y} = \mathrm{RN}(b/(2\widehat{x}))$ produced by the last step of Algorithm 1.

Recalling that $y = b/(2x)$, we deduce from the bounds on $\widehat{x}$ in (8) that

$$\frac{y}{U'} \leqslant \frac{b}{2\widehat{x}} \leqslant \frac{y}{L'}. \tag{10}$$

Applying (7) then shows that $\widehat{y}$ satisfies

$$y \cdot \frac{1 - u + 2u^2}{U'} \leqslant \widehat{y} \leqslant y \cdot \frac{1 + u - 2u^2}{L'}. \tag{11}$$

One has

$$\frac{1 - u + 2u^2}{U'} = 1 - \frac{7}{2}u + \frac{97}{8}u^2 + \mathcal{O}(u^3)$$

and one can check that this is larger than $1 - \frac{7}{2}u$. However, the upper bound has the form

$$\frac{1 + u - 2u^2}{L'} = 1 + \frac{7}{2}u + \frac{13}{8}u^2 + \mathcal{O}(u^3)$$

and is *not* smaller than $1 + \frac{7}{2}u$. Thus, at this stage, all we have is

$$y\left(1 - \frac{7}{2}u\right) \leqslant \widehat{y} \leqslant y\left(1 + \frac{7}{2}u + \mathcal{O}(u^2)\right). \tag{12}$$

To remove the term $\mathcal{O}(u^2)$, we introduce the following two lemmas, which show that the bounds in (6) and (7) can be reduced slightly under suitable assumptions.

Lemma III.1. *Let a be a nonnegative floating-point number. If a is not an integral power of 2, then*

$$\mathrm{RN}\big(\sqrt{a}\big) = \sqrt{a}\,(1 + \delta), \qquad |\delta| \leqslant \frac{u}{\sqrt{1 + 6u}}.$$

Proof. The result is clear for $a = 0$, so we assume that $a > 0$. Then one can write $a = m \cdot 2^k$, where k is an even integer and m is an integral multiple of $2u = 2^{1-p}$ such that $1 \leqslant m < 4$. We now consider the following three cases:

- if $m = 1$ or $m = 1 + 2u$, then $\mathrm{RN}(\sqrt{a}) = 2^{k/2}$ is an integral power of two;
- if $m = 1 + 4u$, then $\mathrm{RN}(\sqrt{a}) = (1 + 2u) \cdot 2^{k/2}$ and the relative error is less than $2u^2$, and thus less than $u/\sqrt{1 + 6u}$ (since in this case we necessarily have $p \geqslant 2$);
- if $m \geqslant 1 + 6u$, then, since $\sqrt{a} \in [2^{k/2}, 2^{k/2+1})$,
$$\frac{|\mathrm{RN}(\sqrt{a}) - \sqrt{a}|}{\sqrt{a}} \leqslant \frac{u \cdot 2^{k/2}}{\sqrt{a}} = \frac{u}{\sqrt{m}} \leqslant \frac{u}{\sqrt{1 + 6u}}.$$

□

If we compare with (6), we see that the above lemma gives a slightly smaller bound, since

$$\frac{u}{\sqrt{1 + 6u}} = u - 3u^2 + \mathcal{O}(u^3),$$

whereas

$$1 - \frac{1}{\sqrt{1 + 2u}} = u - \frac{3}{2}u^2 + \mathcal{O}(u^3).$$

Lemma III.2. *Let a and b be two floating-point numbers, with b nonzero. If b is not equal to $2 - 2u$ times an integral power of 2, then*

$$\mathrm{RN}\Big(\frac{a}{b}\Big) = \frac{a}{b}\,(1 + \delta), \qquad |\delta| \leqslant \frac{u}{1 + 3u}.$$

Proof. Up to scaling by suitable powers of two, we can assume that $1 \leqslant b < 2$ and $1 \leqslant a/b < 2$, so the assumption on b becomes $b \leqslant 2 - 4u$. If $a = b$ then the division is exact, so it remains to consider the case where $a > b$, that is, $a \geqslant b + 2u$. Consequently,

$$\frac{a}{b} \geqslant 1 + \frac{2u}{b} \geqslant 1 + \frac{u}{1 - 2u} > 1 + u,$$

and three cases can occur:

- if $a/b \leqslant 1 + 2u$, then $\mathrm{RN}(a/b) = 1 + 2u$ and the relative error satisfies
$$\left|\frac{\mathrm{RN}(a/b) - a/b}{a/b}\right| \leqslant \frac{1 + 2u}{1 + \frac{u}{1-2u}} - 1 = \frac{u(1 - 4u)}{1 - u},$$
with the latter quantity being less than $u/(1 + 3u)$ for $u > 0$;
- if $1 + 2u < a/b < 1 + 3u$, then $\mathrm{RN}(a/b) = 1 + 2u$ and
$$\left|\frac{\mathrm{RN}(a/b) - a/b}{a/b}\right| < 1 - \frac{1 + 2u}{1 + 3u} = \frac{u}{1 + 3u};$$
- if $a/b \geqslant 1 + 3u$, then, using the fact that $a/b < 2$,
$$\left|\frac{\mathrm{RN}(a/b) - a/b}{a/b}\right| \leqslant \frac{u}{|a/b|} \leqslant \frac{u}{1 + 3u}.$$

□

Note that $u/(1+3u) = u - 3u^2 + \mathcal{O}(u^3)$, which is slightly smaller than the expression $u - 2u^2$ in (7).

We can now exploit these two lemmas as follows, by considering three different cases depending on the shape of the floating-point number

$$\widehat{x} = \mathrm{RN}\big(\sqrt{\nu/2}\big)$$

produced at step 6 of Algorithm 1:

1) If $\widehat{x}$ is an integral power of 2, then the floating-point division at step 7 is exact. Hence $\widehat{y} = b/(2\widehat{x})$ and it follows from (10) that
$$\widehat{y} \leqslant y \cdot \frac{1}{L'},$$
where $1/L'$ has the form
$$1 + \frac{5}{2}u + \mathcal{O}(u^2)$$
and is less than $1 + \frac{7}{2}u$ for $u \leqslant 1/2$.
2) If $\widehat{x} = (2 - 2u) \cdot 2^k$ for some integer k, then $\sqrt{\nu/2} \geqslant (2 - 3u) \cdot 2^k$ and the relative error due to rounding is at most $u/(2 - 3u) = u/2 + \mathcal{O}(u^2)$. This means that instead of L' as in (9), one can take
$$\begin{aligned} L'' &:= \left(1 - \frac{u}{1 + u}\right) \cdot \frac{1}{(1 + 2u)^{1/4}} \cdot \left(1 - \frac{u}{2 - 3u}\right) \\ &= 1 - 2u + \mathcal{O}(u^2) \end{aligned}$$

and replace the upper bound in (11) by

$$\widehat{y} \leqslant y \cdot \frac{1+u-2u^2}{L''}.$$

Here $(1+u-2u^2)/L''$ has the form

$$1+3u+\frac{15}{8}u^2+\mathcal{O}(u^3)$$

and is less than $1+\frac{7}{2}u$ for $u \leqslant 1/8$.

3) In all the other cases, Lemmas III.1 and III.2 imply that

$$\widehat{x} = \sqrt{\frac{\nu}{2}} \cdot (1+\delta), \qquad |\delta| \leqslant \frac{u}{\sqrt{1+6u}}$$

and

$$\widehat{y} = \frac{b}{2\widehat{x}} \cdot (1+\delta'), \qquad |\delta'| \leqslant \frac{u}{1+3u}.$$

Therefore, the upper bound in (11) can be replaced by

$$\widehat{y} \leqslant y \cdot \frac{1+\frac{u}{1+3u}}{L'''},$$

where

$$L''' := \left(1-\frac{u}{1+u}\right) \cdot \frac{1}{(1+2u)^{1/4}} \cdot \left(1-\frac{u}{\sqrt{1+6u}}\right).$$

It can then be checked that $(1+u/(1+3u))/L'''$ has the form

$$1+\frac{7}{2}u-\frac{7}{8}u^2+\mathcal{O}(u^3)$$

and is less than $1+\frac{7}{2}u$ for $u \leqslant 1/8$.

The three cases above thus show that $\widehat{y} \leqslant y\,(1+\frac{7}{2}u)$ if $p \geqslant 3$. By combining this upper bound with the lower bound in (12), we conclude that

$$|\widehat{y}-y| \leqslant \frac{7}{2}u|y| \qquad \text{if } p \geqslant 3.$$

IV. Conclusion

The refined bounds $\frac{5}{2}u$ and $\frac{7}{2}u$ we have obtained on the relative errors of $\widehat{x}$ and $\widehat{y}$ can also be used to deduce the refined bound $\frac{\sqrt{37}}{2}u$ on the normwise relative error $|\widehat{z}-z|/|z|$.

To see this, one can proceed exactly as Hull, Fairgrieve, and Tang in [2, p. 230]. The normwise error satisfies

$$\frac{|\widehat{z}-z|}{|z|} = \frac{\sqrt{(\widehat{x}-x)^2+(\widehat{y}-y)^2}}{\sqrt{x^2+y^2}}$$
$$\leqslant \frac{\sqrt{\frac{25}{4}u^2x^2+\frac{49}{4}u^2y^2}}{\sqrt{x^2+y^2}} =: f(x,y) \qquad \text{for } p \geqslant 3.$$

Since $(a,b) \neq (0,0)$ and $a \geqslant 0$ by assumption, we have $x > 0$ and $0 \leqslant y \leqslant x$. On this domain, $f(x,y)$ is largest when $x = y$, and its maximum equals

$$f(x,x) = \frac{\sqrt{\frac{25}{4}+\frac{49}{4}}}{\sqrt{2}}\,u = \frac{\sqrt{37}}{2}\,u.$$

To summarize, we have shown the following:

Theorem IV.1. *Assume binary floating-point arithmetic with precision $p \geqslant 3$ and rounding to nearest. Then, in the absence of underflow and overflow, the floating-point values $\widehat{x}$ and $\widehat{y}$ computed by Algorithm 1 satisfy*

$$|\widehat{x}-x| \leqslant \frac{5}{2}u|x|, \qquad |\widehat{y}-y| \leqslant \frac{7}{2}u|y|, \qquad |\widehat{z}-z| \leqslant \frac{\sqrt{37}}{2}u|z|,$$

where $\widehat{z} = \widehat{x}+i\widehat{y}$, $z = x+iy$, and $\sqrt{37}/2 = 3.041\ldots$.

We also note that these bounds are reasonably sharp. For example,

- for $p = 24$ (binary32/single-precision format) and with $a = 53877/2^{23}$ and $b = 8433897/2^{22}$, the values $\widehat{x}$ and $\widehat{y}$ computed by Algorithm 1 satisfy $|\widehat{x}-x|/|x| > 2.459u$, $|\widehat{y}-y|/|y| > 3.446u$, and $|\widehat{z}-z|/|z| > 2.992u$;
- for $p = 53$ (binary64/double-precision format) and with
$$a = 650824205667/2^{52}$$
and
$$b = 4507997673885435/2^{51},$$
these errors are larger than $2.482u$, $3.481u$, and $3.023u$, respectively;
- for $p = 113$ (binary128/quad-precision format) and with
$$a = 5964355165421358811162724754522111/2^{150}$$
and
$$b = 5192298808565739300701174676465595/2^{111},$$
these errors are larger than $2.483u$, $3.471u$, and $3.018u$, respectively.

Acknowledgment

This research was supported in part by the French National Research Agency under grant ANR-13-INSE-0007 (MetaLibm project).

References

[1] P. Friedland. Algorithm 312: Absolute value and square root of a complex number. *Communications of the ACM*, 10(10):665, 1967.

[2] T. E. Hull, T. F. Fairgrieve, and P. T. P. Tang. Implementing complex elementary functions using exception handling. *ACM Transactions on Mathematical Software*, 20(2):215–244, 1994.

[3] C.-P. Jeannerod and S. M. Rump. On relative errors of floating-point operations: optimal bounds and applications, 2014. Manuscript available at https://hal.inria.fr/hal-00934443. To appear in *Mathematics of Computation*.

[4] W. Kahan. Branch cuts for complex elementary functions or much ado about nothing's sign bit. In A. Iserles and M. J. D. Powell, editors, *The State of the Art in Numerical Analysis*, pages 165–211. Oxford University Press, 1987.

[5] D. E. Knuth. *The Art of Computer Programming, Volume 2, Seminumerical Algorithms*. Addison-Wesley, Reading, MA, USA, third edition, 1998.