On complex multiplication and division with an FMA

Jean-Michel Muller

Joint work with C.-P. Jeannerod, P. Kornerup and N. Louvet
INVA 2014
Thank you!

ありがとうございます
Complex multiplication and division

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

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

and their quotient $x/y$ can be expressed as

$$q = \frac{ac + bd}{c^2 + d^2} + i \frac{bc - ad}{c^2 + d^2}.$$

In floating-point arithmetic, several issues:

- accuracy,
- spurious overflow/underflow (e.g., $c^2 + d^2$ overflows, whereas the real and imaginary parts of $q$ are representable);

Here: focus on accuracy problems. Scaling techniques to avoid spurious overflow/underflow dealt with in further work.

We assume that an FMA instruction is available.
Floating-Point numbers, roundings

**Precision-\( p \) binary FP number:** either 0 or

\[
x = X \cdot 2^{e_x - p + 1},
\]

where \( X \) and \( e_x \in \mathbb{Z} \), with \( 2^{p-1} \leq |X| \leq 2^p - 1 \). We denote this set by \( \mathbb{F}_p \).

- unlimited exponent range → results valid in usual FP arithmetic unless underflow/overflow occurs;
- \( X \): integral significand of \( x \);
- \( 2^{1-p} \cdot X \): significand of \( x \);
- \( e_x \): exponent of \( x \).

This presentation: binary arithmetic only.
In general, the sum, product, quotient, etc., of two FP numbers is not an FP number: it must be rounded;

**correct rounding**: *Rounding function* \(\circ\), and when \((a \top b)\) is performed, the returned value is \(\circ(a \top b)\);

default rounding function: round to nearest even: RN;

for any real number \(t\), \(|RN(t) - t| \leq u \cdot |t|\), where \(u = 2^{-p}\);

\(\rightarrow\) all arithmetic operations performed with relative error \(\leq u\);

we assume that an FMA instruction is available: computes RN\((ab + c)\).

**Here**: we try to analyze several simple algorithms.
Componentwise and normwise relative errors

When $\hat{z}$ approximates $z$:

- **componentwise error:**

  $$\max \left\{ \left| \frac{\text{Re}(z) - \text{Re}(\hat{z})}{\text{Re}(z)} \right|, \left| \frac{\text{Im}(z) - \text{Im}(\hat{z})}{\text{Im}(z)} \right| \right\};$$

- **normwise error:**

  $$\left| \frac{z - \hat{z}}{z} \right|.$$

Choosing between both kinds of error depends on the application.

- **componentwise error** $\leq \epsilon \Rightarrow$ normwise error $\leq \epsilon$;
- the converse is not true.
Naive multiplication algorithm without an FMA

$$A_0 : (a + ib, c + id) \mapsto \text{RN}(\text{RN}(ac) - \text{RN}(bd)) + i \cdot \text{RN}(\text{RN}(ad) + \text{RN}(bc))$$

- Componentwise error: can be huge (yet finite);
- Normwise accuracy: studied by Brent, Percival, and Zimmermann (2007). The computed value has the form

$$\hat{z}_0 = z(1 + \epsilon), \quad |\epsilon| < \sqrt{5}u,$$

→ the normwise relative error $|\hat{z}_0 / z - 1|$ is always $\leq \sqrt{5} \cdot u$.

For any $p \geq 2$ they provide FP numbers $a, b, c, d$ for which

$$|\hat{z}_0 / z - 1| = \sqrt{5}u - O(u^2) \rightarrow \text{the relative error bound } \sqrt{5}u \text{ is asymptotically optimal as } u \rightarrow 0 \text{ (or, equivalently, as } p \rightarrow +\infty).$$

Can we do better if an FMA instruction is available?
Naive multiplication algorithm with an FMA

With an FMA, the simple way of evaluating \( ac - bd + i(ad + bc) \) becomes:

\[
\mathcal{A}_1 : (a + ib, c + id) \mapsto \text{RN}(ac - \text{RN}(bd)) + i \cdot \text{RN}(ad + \text{RN}(bc))
\]

Algorithm \( \mathcal{A}_1 \) is just one of 4 variants that differ only in the choice of the products to which the FMA operations apply.

- componentwise error: can be huge (even infinite);
- normwise error:
  - for any of these 4 variants the computed complex product \( \hat{z}_1 \) satisfies
    \[
    \left| \hat{z}_1 - z \right| \leq 2u|z| \quad (1)
    \]
  - we build inputs \( a, b, c, d \) for which \( |\hat{z}_1/z - 1| = 2u - O(u^{1.5}) \) as \( u \to 0 \Rightarrow \) the relative error bound (1) is asymptotically optimal (given later on).

→ the FMA improves the situation from a normwise point of view.
The CHT algorithm

Given FP numbers $a$ and $b$, the error $e = ab - \text{RN}(ab)$ satisfies

$$e = \text{RN}(ab - \text{RN}(ab))$$

$\rightarrow$ it is computed exactly with an FMA;

$\rightarrow$ compensated algorithms: we "re-inject" that error later on in the calculation.

(without an FMA and using only $+$, $-$, $\times$, the cheapest algorithm we are aware of for computing $e$ uses 17 operations)

Cornea, Harrison, and Tang use this property in the following algorithm to evaluate

$$r = ab + cd$$

accurately in 7 floating-point operations.
The CHT algorithm

We approximate

\[ r = ab + cd \]

by \( \hat{r} \) obtained as follows

```
algorithm CHT(a, b, c, d)
    \( \hat{w}_1 := \text{RN}(ab); \) \( \hat{w}_2 := \text{RN}(cd); \)
    \( e_1 := \text{RN}(ab - \hat{w}_1); \) \( e_2 := \text{RN}(cd - \hat{w}_2); \) // exact operations
    \( \hat{f} := \text{RN}(\hat{w}_1 + \hat{w}_2); \)
    \( \hat{e} := \text{RN}(e_1 + e_2); \)
    \( \hat{r} := \text{RN}(\hat{f} + \hat{e}); \)
    \text{return} \( \hat{r}; \)
```

Cornea, Harrison, and Tang show that the error is \( \mathcal{O}(u) \).
Properties of the CHT algorithm

- we have shown that
  \[ |\hat{r} - r| \leq 2u \cdot |r| \]  
  (2)

- we build a “generic example” parameterized by \( p \), that shows that the bound (2) is asymptotically optimal (as \( u \to 0 \));

- for instance, in double precision arithmetic, with our generic example, error
  \[ u \times 1.99999999999999922284 \cdots \]

  is attained.
Application of CHT to the complex product

- Evaluate separately the real and imaginary parts of
  \[ z = ac - bd + i(ad + bc) \] using CHT;
- uses 14 floating-point operations.

\[ \mathcal{A}_2 : (a + ib, c + id) \mapsto \text{CHT}(a, c, -b, d) + i \cdot \text{CHT}(a, d, b, c) \]

- componentwise error \( \leq 2u \) (asymptotically optimal);
- consequence: normwise error \( \leq 2u \).

The normwise bound is also asymptotically optimal.
Application of CHT to the complex product

**Theorem 1**

Let \(a, b \in \mathbb{F}_p\) be given by

\[
a = RD\left(\left(1 - 2^{-p}\right)\sqrt{2^{p-2}}\right), \quad b = 2^{p-1} + \left\lfloor \sqrt{2^{p-2}} \right\rfloor + 1,
\]

where, for \(t \in \mathbb{R}\), \(RD(t) = \max\{f \in \mathbb{F}_p : f \leq t\}\) denotes rounding down in \(\mathbb{F}_p\). Let also \(\hat{z}_2\) be the approximation to \(z = (a + ib)^2\) computed by algorithm \(A_2\). If \(p \geq 5\) then, barring underflow and overflow,

\[
|\hat{z}_2/z - 1| > 2u - 8u^{1.5} - 6u^2.
\]

To be compared with the upper bound \(2u\).
Kahan’s algorithm for $ab + cd$

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

- 4 operations (CHT needed 7);
- \( e = cd - \text{RN}(cd) \) computed exactly thanks to the FMA instruction;
- it is added to \( \hat{f} \) in order to yield the approximation \( \hat{r} \) to \( r = ab + cd \).

We have shown that

\[
|\hat{r} - r| \leq 2u|r|,
\]

and that this bound is asymptotically optimal (as \( u \to 0 \)).
Application of Kahan’s algorithm to the complex product

- Evaluate separately the real and imaginary parts of $z = ac - bd + i(ad + bc)$ using Kahan’s algorithm;
- uses **8 floating-point operations** (instead of 14 with CHT);

$$A_3 : (a + ib, c + id) \mapsto \text{Kahan}(a, c, -b, d) + i \cdot \text{Kahan}(a, d, b, c)$$

- componentwise error $\leq 2u$ (asymptotically optimal);
- consequence: normwise error $\leq 2u$.

The normwise bound is asymptotically optimal.
Theorem 2

Let $a, b \in \mathbb{F}_p$ be given by

$$a = \text{pred} \left( \sqrt{2^{p-2}} \right), \quad b = 2^{p-1} + \left\lfloor \sqrt{2^{p-2}} \right\rfloor + 1,$$

where, for $t \in \mathbb{R}_{>0}$, $\text{pred}(t) = \max\{f \in \mathbb{F}_p : f < t\}$ denotes the predecessor of $t$ in $\mathbb{F}_p$. Let also $\hat{z}_1$ and $\hat{z}_3$ be the approximations to $z = (a + ib)^2$ computed by algorithms $A_1$ and $A_3$, respectively. If $p \geq 5$ then, barring underflow and overflow,

$$|\hat{z}_h/z - 1| > 2u - 8u^{1.5} - 4u^2, \quad h \in \{1, 3\}.$$
Conclusion on complex multiplication

- the availability of an FMA makes it possible to replace the classical normwise accuracy bound $\sqrt{5}u$ by $2u$ with simple algorithms,
- this new bound is sharp (asymptotically optimal with Algorithms $A_1$, $A_2$ and $A_3$),
- if normwise error only is at stake, the simplest algorithm (naive multiplication with FMA: Algorithm $A_1$) is juste fine,
- however if we also want to reduce the componentwise error the multiplication based on Kahan’s algorithm (i.e., Algorithm $A_3$) is to be preferred.

More on this: http://perso.ens-lyon.fr/jean-michel.muller/JeKoLoMu13-submission.pdf
A few words on complex division with an FMA

\[ q = \frac{ac + bd}{c^2 + d^2} + i \frac{bc - ad}{c^2 + d^2}. \]

- here: componentwise error only;
- basic idea: separately compute \( ac + bd, bc - ad, \) and \( c^2 + d^2 \) using one of the previously seen methods (naive without FMA, naive with FMA, CHT, Kahan);
- \( c^2 + d^2 \) is a special case: cancellation cannot occur;
- notice that \( ac + bd \) and \( bc - ad \) cannot cancel simultaneously:
  - if \( abcd > 0 \) then \( ac \) and \( bd \) have the same sign;
  - otherwise \( bc \) and \( -ad \) have the same sign (unless one of the inputs is zero: case easily dealt with).
- straight-line algorithms: no tests.
The special case $c^2 + d^2$

- more generally, computation of $ac + bd$ with $ab$ and $cd$ of the same sign;
- the naive method, the naive method with FMA, Kahan’s algorithm have the same relative error bound $2u$;
- the bound is sharp, even if we restrict ourselves to $c^2 + d^2$.

Consequence: for $c^2 + d^2$, the naive algorithm (3 operations) or the naive-with-FMA algorithm (2 operations) suffice.
A $5u + O(u^2)$ componentwise error bound

- we use Kahan’s algorithm or CHT for $ac + bd$ and $bc − ad$
- we use naive-with-fma for $c^2 + d^2$;
- consider the real part $(ac + bd)/(c^2 + d^2)$. We have:

\[
\text{Re}(\hat{q}) = \frac{\text{Re}(q)(1 + \epsilon)}{(c^2 + d^2)(1 + \epsilon')(1 + \epsilon'')},
\]

where $|\epsilon|, |\epsilon'| \leq 2u$, and where the relative error $|\epsilon''|$ of FP division is bounded by $u$.

$\rightarrow$ the real part of $\hat{q}$ has the form $(1 + \theta) \cdot \text{Re}(q)$, with

\[
|\theta| \leq \frac{1 + 2u}{1 - 2u}(1 + u) - 1,
\]

- the same holds for the imaginary part.

Consequence: componentwise error $\leq 5u + 13u^2$. 
A $5u + \mathcal{O}(u^2)$ componentwise error bound

The obtained algorithm is:

\begin{algorithm}
\textbf{CompDivS}(a + ib, c + id) \\
\hat{\delta} := \text{RN}(c^2 + \text{RN}(d^2)); \\
\hat{g}_{\text{re}} := \text{Kahan}(a, b, c, d); \quad \text{// evaluates } ac + bd \\
\hat{g}_{\text{im}} := \text{Kahan}(b, -a, c, d); \quad \text{// evaluates } bc - ad \\
\hat{q}_{\text{re}} := \text{RN}(\hat{g}_{\text{re}}/\hat{\delta}); \quad \hat{q}_{\text{im}} := \text{RN}(\hat{g}_{\text{im}}/\hat{\delta}); \\
\text{return } \hat{q}_{\text{re}} + i\hat{q}_{\text{im}};
\end{algorithm}
When \( p \) is even the bound is asymptotically optimal

If we choose:

\[
\begin{align*}
a &= 2^p - 5 \cdot 2^{\frac{p}{2} - 1}, \\
b &= -2^{-\frac{p}{2}} \cdot (2^p - 5 \cdot 2^{\frac{p}{2} - 1} + 3), \\
c &= 2^p - 2, \\
d &= 2^{\frac{p}{2} + 1} \cdot (2^{p-1} + 2^{\frac{p}{2} - 1}),
\end{align*}
\]

then the quotient \( \hat{q} \) computed by CompDivS satisfies

\[
\frac{|\text{Re} (\hat{q}) - \text{Re} (q)|}{|\text{Re} (q)|} = 5u - O(u^{3/2}).
\]
When \( p \) is odd...

We have no proof of asymptotic optimality, however:

<table>
<thead>
<tr>
<th>( p )</th>
<th>example</th>
</tr>
</thead>
</table>
| 53    | \( a = 2^{52} + 1 \)  
      | \( b = -142398041 \)  
      | \( c = 2^{52} \)  
      | \( d = 94906267 \cdot 2^{52} = (1 + \lceil 2^{53}/2 \rceil) \cdot 2^{52} \)  
      | \( |\hat{z}_{\text{re}} - \text{Re } z|/|\text{Re } z| = 4.9987 \ldots \times u \) |
| 113   | \( a = 2^{112} + 1 \)  
      | \( b = -152857240142482713 \)  
      | \( c = 2^{112} \)  
      | \( d = 101904826760412363 \cdot 2^{112} \)  
      | \( |\hat{z}_{\text{re}} - \text{Re } z|/|\text{Re } z| = 4.9999 \ldots \times u \) |

**Table 1**: Relative error in \( \hat{z}_{\text{re}} \) computed using CompDivS close to the upper bound \( 5u + 13u^2 \).
accurate complex division is feasible with simple algorithms: componentwise error bound \( 5u + 13u^2 \);
that bound is asymptotically optimal (at least for even \( p \));
to be done: use *scaling techniques* to avoid spurious overflow/underflow.

Thank you for your attention.