Computing the Frobenius norm using cr_hypot

Vedran Novaković

August 9, 2025

Consider a well-defined hypot function, i.e.,

$$\operatorname{hypot}(x,y)=\sqrt{x^2+y^2}(1+\epsilon'),$$

where $|\epsilon'|\ll 1$. If it is correctly rounded for all inputs, denote it by cr_hypot.

Theorem 1. *Let* $\mathbf{x}=\begin{bmatrix}x_1 & \cdots & x_n\end{bmatrix}^T$ *be a vector of finite floating-point values, and* $\|\mathbf{x}\|_F$ *its Frobenius norm. If its approximation* $\underline{\|\mathbf{x}\|_F}$ *is computed as* $\underline{\|\mathbf{x}\|_F}=\underline{f_n}$*, where* $\underline{f_0}=f_0=0$*,* $\underline{f_1}=f_1=|x_1|$*, and*

$$2\le i\le n \implies \underline{f_i}=\operatorname{hypot}(\underline{f_{i-1}},x_i), \tag{1}$$

then, barring any overflow and inexact underflow, when $x_i\ne 0$ *it holds*

$$\underline{f_i}=f_i(1+\epsilon_i),\qquad 1+\epsilon_i=\sqrt{1+\epsilon_{i-1}(2+\epsilon_{i-1})\frac{f_{i-1}^2}{f_i^2}}(1+\epsilon_i'). \tag{2}$$

Assume that hypot *is* cr_hypot*. Then,* $|\epsilon_i'|\le\varepsilon$*, with* ε *being the machine precision. If* $x_i=0$*, then* $\underline{f_i}=\underline{f_{i-1}}$*. If a lower bound of* ϵ_{i-1} *is denoted by* ϵ_{i-1}^- *and an upper bound by* ϵ_{i-1}^+*, where* $\epsilon_1^-=\epsilon_1^+=0$*, then, when* $0\ge\epsilon_{i-1}^-\ge -1$*, the relative error factor* $1+\epsilon_i$ *from* (2) *can be bounded irrespectively of* $x_i\ne 0$ *and* f_i *as*

$$\begin{aligned}1+\epsilon_i^- &= \sqrt{1+\epsilon_{i-1}^-(2+\epsilon_{i-1}^-)}(1-\varepsilon)\le 1+\epsilon_i\\ &\le \sqrt{1+\epsilon_{i-1}^+(2+\epsilon_{i-1}^+)}(1+\varepsilon)=1+\epsilon_i^+.\end{aligned} \tag{3}$$

Proof. Using any well-defined hypot function, assuming that (2) holds for all j such that $0\le j<i$, where $2\le i\le n$, and that $x_i\ne 0$, it follows

$$\underline{f_i}=\sqrt{\underline{f_{i-1}^2}+x_i^2}(1+\epsilon_i')=\sqrt{f_{i-1}^2(1+\epsilon_{i-1})^2+x_i^2}(1+\epsilon_i'). \tag{4}$$

If the term under the square root on the right hand side of (4) is written as

$$f_{i-1}^2(1+\epsilon_{i-1})^2 + x_i^2 = (f_{i-1}^2 + x_i^2)(1+y), \tag{5}$$

then an easy algebraic manipulation gives

$$y = \epsilon_{i-1}(2+\epsilon_{i-1})\frac{f_{i-1}^2}{f_{i-1}^2 + x_i^2} = \epsilon_{i-1}(2+\epsilon_{i-1})\frac{f_{i-1}^2}{f_i^2}, \tag{6}$$

what, after taking the absolute value of both sides of (6), leads to

$$|y| \le |\epsilon_{i-1}|(2+|\epsilon_{i-1}|)\frac{f_{i-1}^2}{f_i^2} \le |\epsilon_{i-1}|(2+|\epsilon_{i-1}|). \tag{7}$$

Substituting (5) into (4) yields

$$\underline{f_i} = \sqrt{f_{i-1}^2 + x_i^2}\sqrt{1+y}(1+\epsilon_i') = f_i\sqrt{1+y}(1+\epsilon_i') = f_i(1+\epsilon_i),$$

where $(1+\epsilon_i) = \sqrt{1+y}(1+\epsilon_i')$, as claimed in (2). The recurrence (3) for bounds on $1+\epsilon_i$ when hypot is cr_hypot follows from (7) and the fact that the function $x \mapsto x(2+x)$ is monotonically increasing for $x \ge -1$. $\square$

More practically, (3) can be further simplified as $-\epsilon_i^- < \epsilon_i^+ < i\varepsilon$ when

$$((\varepsilon = 2^{-24}) \wedge (3 \le i \le 5793)) \quad \vee \quad ((\varepsilon = 2^{-53}) \wedge (3 \le i \le 134217729)),$$

what follows from evaluating (3) iteratively over i using the MPFR library [1] with 2048 bits of precision.

A result similar to Theorem 1 can be obtained in the case of combining two partial norms as

$$\underline{f_{[i,j]}} = \mathrm{hypot}(\underline{f_{[i]}}, \underline{f_{[j]}}),$$

e.g., when OpenMP-reducing the partial, per-thread results. Bear in mind that the OpenMP standard [2, §7.6.7] leaves the reduction order unspecified.

Theorem 2. *Assume that* $\underline{f_{[i]}} = f_{[i]}(1+\epsilon_{[i]})$ *and* $\underline{f_{[j]}} = f_{[j]}(1+\epsilon_{[j]})$ *approximate the Frobenius norms of some vectors of length* i *and* j*, respectively, and let*

$$\underline{f_{[i,j]}} = \mathrm{hypot}(\underline{f_{[i]}}, \underline{f_{[j]}})$$

be an approximation of the Frobenius norm of the concatenation (of length $i+j$*) of those two vectors. Then, barring any overflow and inexact underflow,*

$$\underline{f_{[i,j]}} = f_{[i,j]}(1+\epsilon_{[i,j]}), \quad 1+\epsilon_{[i,j]} = \sqrt{1+\epsilon_{[\ell]}(2+\epsilon_{[\ell]})\frac{f_{[l]}^2}{f_{[i,j]}^2}}(1+\epsilon_{[k]})(1+\epsilon_{[i,j]}'), \tag{8}$$

where $1+\epsilon_{[l]} = \min\{1+\epsilon_{[i]}, 1+\epsilon_{[j]}\}$, $1+\epsilon_{[k]} = \max\{1+\epsilon_{[i]}, 1+\epsilon_{[j]}\}$, *and* $1+\epsilon_{[\ell]} = (1+\epsilon_{[l]})/(1+\epsilon_{[k]})$, *i.e.,* $l=i$ *and* $k=j$ *or* $l=j$ *and* $k=i$, *while* $|\epsilon'_{[i,j]}| \le \varepsilon$ *if* hypot *is* cr_hypot.

Proof. Expanding $\underline{f_{[i]}^2} + \underline{f_{[j]}^2}$ gives

$$\underline{f_{[i]}^2} + \underline{f_{[j]}^2} = f_{[i]}^2(1+\epsilon_{[i]})^2 + f_{[j]}^2(1+\epsilon_{[j]})^2 = (f_{[l]}^2(1+\epsilon_{[\ell]})^2 + f_{[k]}^2)(1+\epsilon_{[k]})^2. \quad (9)$$

Similarly to (5), expressing the first factor on the right hand side of (9) as

$$f_{[l]}^2(1+\epsilon_{[\ell]})^2 + f_{[k]}^2 = (f_{[l]}^2 + f_{[k]}^2)(1+z) \quad (10)$$

leads to

$$z = \epsilon_{[\ell]}(2+\epsilon_{[\ell]})\frac{f_{[l]}^2}{f_{[l]}^2 + f_{[k]}^2} = \epsilon_{[\ell]}(2+\epsilon_{[\ell]})\frac{f_{[l]}^2}{f_{[i,j]}^2},$$

and therefore, by substituting (10) into (9),

$$\underline{f_{[i,j]}} = \sqrt{\underline{f_{[i]}^2} + \underline{f_{[j]}^2}}(1+\epsilon'_{[i,j]}) = \sqrt{f_{[i]}^2 + f_{[j]}^2}\sqrt{1+z}(1+\epsilon_{[k]})(1+\epsilon'_{[i,j]}),$$

what is equivalent to (8). □

Compare this to computing the sum of squares, as in `xNRM2` from BLAS (see, e.g., `https://github.com/Reference-LAPACK/lapack/blob/master/BLAS/SRC/dnrm2.f90`).

Theorem 3. *Let* $\mathbf{x} = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T$ *be a vector of finite floating-point values, and* $\|\mathbf{x}\|_F$ *its Frobenius norm. If its approximation* $\underline{\|\mathbf{x}\|_F}$ *is computed as* $\underline{\|\mathbf{x}\|_F} = \mathrm{sqrt}(\underline{g_n})$, *where* $\underline{g_0} = g_0 = 0$ *and*

$$1 \le i \le n \implies \underline{g_i} = \mathrm{fma}(x_i, x_i, \underline{g_{i-1}}),$$

then, barring any overflow and inexact underflow, when $x_i \ne 0$ *it holds*

$$\underline{g_i} = g_i(1+\epsilon''_i), \qquad 1+\epsilon''_i = \left(1+\epsilon''_{i-1}\frac{g_{i-1}}{g_i}\right)(1+\epsilon'''_i), \quad (11)$$

where ϵ'''_i, $|\epsilon'''_i| \le \varepsilon$, *is the rounding error of the* fma. *Also, with* $|\epsilon_{\surd}| \le \varepsilon$,

$$\underline{\|\mathbf{x}\|_F} = \mathrm{sqrt}(\underline{g_n}) = \|\mathbf{x}\|_F\sqrt{1+\epsilon''_n}(1+\epsilon_{\surd}).$$

Proof. From

$$x_i^2 + \underline{g_{i-1}} = x_i^2 + g_{i-1}(1+\epsilon''_{i-1}) = (x_i^2 + g_{i-1})(1+w)$$

it follows

$$w = \epsilon''_{i-1}\frac{g_{i-1}}{g_{i-1}+x_i^2} = \epsilon''_{i-1}\frac{g_{i-1}}{g_i},$$

what had to be proven. □

References

[1] Laurent Fousse, Guillaume Hanrot, Vincent Lefèvre, Patrick Pélissier, and Paul Zimmermann. MPFR: A multiple-precision binary floating-point library with correct rounding. *ACM Trans. Math. Softw.*, 33(2):13, 2007.

[2] OpenMP Architecture Review Board. OpenMP API 6.0 Specification. online: `https://www.openmp.org/wp-content/uploads/OpenMP-API-Specification-6-0.pdf`, Nov 2024.