Classical floating-point error bounds revisited

Siegfried M. Rump

Institute for Reliable Computing, Hamburg University of Technology, Schwarzenbergstraße 95, Hamburg 21071, Germany, and Visiting Professor at Waseda University, Faculty of Science and Engineering, 3–4–1 Okubo, Shinjuku-ku, Tokyo 169–8555, Japan

rump@tu-harburg.de

Keywords: Floating-point arithmetic, classical error estimate, IEEE 754

Denote by $\mathbb{F}$ a set of floating-point numbers with operations defined according to the IEEE 754 [1] floating-point standard. Let an arithmetic expression $f : \mathbb{F}^n \to \mathbb{R}$ be given, and let $\tilde{f} : \mathbb{F}^n \to \mathbb{F}$ be the function obtained by replacing each operation in f by the corresponding floating-point operation. For a given vector $x \in \mathbb{F}^n$ we are interested in rigorous estimates for the error $|\tilde{f}(x) - f(x)|$ depending on some computed floating-point approximation and/or on the exact value $|f(x)|$.

Examples are the sum of n floating-point numbers, the dot product of two n-vectors, the factorization of an $n \times n$ matrix, and alike. Typical classical error estimates are

$$x, y \in \mathbb{F}^n : \qquad |\mathrm{float}(x^T y) - x^T y| \le \gamma_n |x^T| \, |y| \tag{1}$$

or

$$T \in \mathbb{F}^{n \times n}, b \in \mathbb{F}^n : \qquad (T + \Delta T)\hat{x} = b \quad \text{with} \quad |\Delta T| \le \gamma_n |T| \tag{2}$$

where $\hat{x}$ denotes the solution of the triangular system $Tx = b$ by substition in floating-point. Here $\gamma_n := \frac{nu}{1-nu}$ provided $u < 1$ is the classical way to bound higher order terms for u denoting the relative rounding error unit.

Note that (1) and (2) inevitably require $nu < 1$, so the dimension is restricted through the relative rounding error unit u. In double precision (binary64) floating-point arithmetic this limits n to about 10^{16} which imposes hardly a practical restriction. However, in single precision (binary32) and application to huge matrices such as in [2] it might be restrictive. Also note that for $n \ge u^{-1}$ no error estimate is known.

We present several new error estimates, both depending on a computed approximation as well as on the exact value. The former yields practically computable error bounds, the latter serve mainly theoretical purposes and are standard in every textbook like Higham's ASNA [3].

Let $x \in \mathbb{F}^n$ be given. Then we show that the classical error estimate for summation

$$\Delta := \left| \mathrm{float}\left(\sum_{i=1}^{n} x_i \right) - \sum_{i=1}^{n} x_i \right| \le \gamma_{n-1} \sum_{i=1}^{n} |x_i| \quad \text{provided} \quad nu \le 1$$

can be improved [1] into the two new estimates

$$\Delta \le (n-1)u \sum_{i=1}^{n} |x_i| \tag{3}$$

and

$$\Delta \le (n-1)u \cdot \mathrm{ufp}(\tilde{S}). \tag{4}$$

Here $\tilde{S}$ denotes the floating-point sum of the absolute values $|x_i|$, so that (3) is a computable bound. Moreover, $\mathrm{ufp}(f)$ for $f \in \mathbb{F}$ is the value of the leading bit in the binary representation of f, so that by $\mathrm{ufp}(f) \le |f| < 2\mathrm{ufp}(f)$ the bound in (4) is potentially sharper by a factor 2 compared to using $\tilde{S}$ instead. Both bounds (3) and (4) are valid without restriction on the vector length n.

Next the classical error bound (1) for a floating-point dot product of two given vectors $x, y \in \mathbb{F}^n$ is improved [4] into

$$|\mathrm{float}(x^T y) - x^T y| \le (n+2)u\, \mathrm{float}(|x^T||y|) + n\mathrm{eta}/2 \tag{5}$$

provided $(n+2)u < 1$. Here eta denotes the smallest positive (unnormalized) floating-point number. The error estimate depending on the true result can be improved and generalized to real input as follows. Let $z \in \mathbb{R}^n$ be given. Then, barring overflow and underflow,

$$\left| \mathrm{float}\left(\sum_{i=1}^{n} \mathrm{fl}(z_i) \right) - \sum_{i=1}^{n} z_i \right| \le nu \sum_{i=1}^{n} |z_i|. \tag{6}$$

So for arbitrary real numbers z_i the error of the floating-point summation of the rounded values $\mathrm{fl}(z_i)$ is bounded by nu times the sum of the real numbers $|z_i|$. Note that the bound is valid without restriction on n. The bound (6) applies in particular to dot products by setting $z_i := x_i y_i$ for $x, y \in \mathbb{F}^n$.

All bounds for summation and dot products are based on an individual error analysis, carefully exploring the worst possible case. More involved are bounds for algorithms such as Gaussian elimination or Cholesky decomposition. For computed factors L, U and G, respectively, the classical bounds [3, Sections 9 and 10] are

$$|A - LU| \le \gamma_n |L|\,|U| \qquad \text{if} \quad nu < 1$$

and

$$|A - GG^T| \le \gamma_{n+1} |G|\,|G^T| \qquad \text{if} \quad (n+1)u < 1.$$

For large values of n no error bound is known. We improve the bounds into

$$|A - LU| \le nu|L|\,|U| \tag{7}$$

and

$$|A - GG^T| \le (n+1)u|G|\,|G^T|. \tag{8}$$

The improvement is two-fold: The bounds do not involve higher terms, and they are valid without restriction on the dimension n. Moreover, the new bonds bear a certain mathematical beauty.

The classical bounds for Gaussian elemination, Cholesky decomposition as well as for triangular system solving by forward or backward substitution are based on the famous Lemma 8.4 in ASNA [3]:

Lemma 8.4 [ASNA] *Let $k \in N_{>0}$ and $a_1, \ldots, a_{k-1}, b_1, \ldots, b_{k-1}, b_k, c \in \mathbb{F}$ be given, with b_k nonzero. If $y = \left(c - \sum_{i=1}^{k-1} a_i b_i\right)/b_k$ is evaluated in floating-point arithmetic then, in the absence of underflow and overflow and no matter what the order of evaluation, the computed $\hat{y}$ satisfies*

$$b_k \hat{y}(1 + \Theta_k^{(0)}) = c - \sum_{i=1}^{k-1} a_i b_i (1 + \Theta_k^{(i)}), \qquad |\Theta_k^{(i)}| \le \gamma_k \quad \textit{for all} \quad i.$$

If $b_k = 1$, so that there is no division, then $|\Theta_k^{(i)}| \le \gamma_{k-1}$ for all i.

These bounds involve higher order terms, and they are only valid with a restriction on the number of terms. We improve and generalize [6] this Lemma as follows:

Lemma 1. *Given $n \in N_{>0}$ and $j \in \{1, \ldots, n\}$, let $x_1, \ldots, x_n \in \mathbb{R}$ be such that $x_j \in \mathbb{F}$ and, for all $i \neq j$, $\mathrm{fl}(x_i)$ does not underflow. Let $\tilde{s}$ be a floating-point sum of $\mathrm{fl}(x_1), \ldots, \mathrm{fl}(x_n)$ no matter what the order of evaluation, and let $\varrho \in \mathbb{R}$ be such that*

$$|\varrho - \hat{s}| \le lu|\varrho| \quad \textit{for some} \quad l \in \mathbb{N}.$$

Then, in the absence of overflow,

$$\Delta := \varrho - \sum_{i=1}^{n} x_i \quad \textit{satisfies} \quad |\Delta| \le (n + l - 1)u \left(|\varrho| + \sum_{i=1, i \neq j}^{n} |x_i| \right).$$

This lemma is designed to prove the mentioned improved error bounds for triangular system solving by substitution (2), the bound for Gaussian elimination (7) and the bound for Cholesky factorization (8).

In contrast to the classical error estimates, which are known since more than 50 years, our bounds do not involve higher order terms, and they are valid for any dimension n. As the classical bounds, our new bounds are valid for any order of evaluation. It seems interesting that refined analyses allow such unexpected results.

Finally we stress that all bounds are based on an individual analysis. There is no panacea to generally replace γ_n by nu without restriction on n.

References:

[1] IEEE 754-2008, IEEE Standard for Floating-Point Arithmetic, New York, 2008.

[2] Rebecca S. Wills and Ilse C. F. Ipsen, Ordinal ranking for Google's PageRank, *SIAM J. Matrix Anal. Appl.*, 30(4), pp. 1677–1696, 2008/09.

[3] N.J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM Publications, Philadelphia, 1996.

[4] S. M. Rump, Error estimation of floating-point summation and dot product, *BIT Numerical Mathematics*, 52(1), pp. 201–220, 2012.

[5] C.-P. Jeannerod and S. M. Rump, Improved error bounds for inner products in floating-point arithmetic, *SIAM J. Matrix Anal. Appl.*, 34, pp. 338–344, 2013.

[6] S.M. Rump and C.-P. Jeannerod, Improved error bounds for LU and Cholesky factorizations, submitted for publication, 2014.