An Implementation of Sin and Cos Using Gal's Accurate Tables

Pascal Leroy (phl)

2025-02-02

This document describes the implementation of functions `Sin` and `Cos` in Principia. The goals of that implementation are to be portable (including to machines that do not have a fused multiply-add instruction), achieve good performance, and ensure correct rounding.

Overview

The implementation follows the ideas described by [GB91] and uses accurate tables produced by the method presented in [SZ05]. It guarantees correct rounding with a high probability. In circumstances where it cannot guarantee correct rounding, it falls back to the (slower but correct) implementation provided by the CORE-MATH project [SZG22] [ZSG+24]. More precisely, the algorithm proceeds through the following steps:

- perform argument reduction using Cody and Waite's algorithm in double precision (see [Mul+10, p. 379]);
- if argument reduction loses too many bits (i.e., the argument is close to a multiple of $\frac{\pi}{2}$), fall back to `cr_sin` or `cr_cos`;
- otherwise, uses accurate tables and a polynomial approximation to compute `Sin` or `Cos` with extra accuracy;
- if the result has a "dangerous rounding configuration" (as defined by [GB91]), fall back to `cr_sin` or `cr_cos`;
- otherwise return the rounded result of the preceding computation.

Notation and Accuracy Model

In this document we assume a base-2 floating-point number system with M significand bits[1] similar to the IEEE formats. We define a real function $\mathfrak{m}$ and an integer function $\mathfrak{e}$ denoting the *significand* and *exponent* of a real number, respectively:

$$x = \pm\,\mathfrak{m}(x) \times 2^{\mathfrak{e}(x)} \qquad \text{with} \qquad 2^{M-1} \leq \mathfrak{m}(x) \leq 2^{M} - 1$$

Note that this representation is unique. Furthermore, if x is a floating-point number, $\mathfrak{m}(x)$ is an integer.

The *unit of the last place* of x is defined as:

$$\mathfrak{u}(x) \coloneqq 2^{\mathfrak{e}(x)}$$

In particular, $\mathfrak{u}(1) = 2^{1-M}$ and:

$$\frac{|x|}{2^M} < \frac{|x|}{2^M - 1} \leq \mathfrak{u}(x) \leq \frac{|x|}{2^{M-1}} \tag{1}$$

We ignore the exponent bias, overflow and underflow as they play no role in this discussion.

Finally, for error analysis we use the accuracy model of [Hig02], equation (2.4): everywhere they appear, the quantities δ_i represent a roundoff factor such that $|\delta_i| < u = 2^{-M}$ (see pages 37 and 38). We also use θ_n and γ_n with the same meaning as in [Hig02], lemma 3.1.

[1] In `binary64`, $M = 53$.

Approximation of $\frac{\pi}{2}$

To perform argument reduction, we need to build approximations of $\frac{\pi}{2}$ with extra accuracy and analyse the circumstances under which they may be used and the errors that they entail on the reduced argument.

Let $z \geq 0$. We start by defining the truncation function $\mathrm{Tr}(\kappa, z)$ which clears the last κ bits of the significand of z:

$$\mathrm{Tr}(\kappa, z) \coloneqq \lfloor 2^{-\kappa}\, \mathfrak{m}(z) \rfloor\; 2^{\kappa}\, \mathfrak{u}(z)$$

We have:

$$z - \mathrm{Tr}(\kappa, z) = (2^{-\kappa}\, \mathfrak{m}(z) - \lfloor 2^{-\kappa}\, \mathfrak{m}(z) \rfloor)\; 2^{\kappa}\, \mathfrak{u}(z)$$

The definition of the floor function implies that the quantity in parentheses is in $[0, 1[$ and therefore:

$$0 \leq z - \mathrm{Tr}(\kappa, z) < 2^{\kappa}\, \mathfrak{u}(z)$$

Furthermore if the bits that are being truncated start with exactly k zeros we have the stricter inequality:

$$2^{\kappa' - 1}\, \mathfrak{u}(z) \leq z - \mathrm{Tr}(\kappa, z) < 2^{\kappa'}\, \mathfrak{u}(z) \quad \text{with} \quad \kappa' = \kappa - k \tag{2}$$

This leads to the following upper bound for the unit of the last place of the truncation error:

$$\mathfrak{u}(z - \mathrm{Tr}(\kappa, z)) < 2^{\kappa' - M + 1}\, \mathfrak{u}(z)$$

which can be made more precise by noting that the function $\mathfrak{u}$ is always a power of 2:

$$\mathfrak{u}(z - \mathrm{Tr}(\kappa, z)) = 2^{\kappa' - M}\, \mathfrak{u}(z) \tag{3}$$

Two-Term Approximation

In this scheme we approximate $\frac{\pi}{2}$ as the sum of two floating-point numbers:

$$\frac{\pi}{2} \simeq C_1 + \delta C_1$$

which are defined as:

$$\begin{cases} C_1 & \coloneqq \mathrm{Tr}\left(\kappa_1, \dfrac{\pi}{2}\right) \\ \delta C_1 & \coloneqq \left[\!\left[\dfrac{\pi}{2} - C_1 \right]\!\right] \end{cases}$$

Equation (??) applied to the definition of C_1 yields:

$$2^{\kappa_1' - 1}\, \mathfrak{u}\left(\frac{\pi}{2}\right) \leq \frac{\pi}{2} - C_1 < 2^{\kappa_1'}\, \mathfrak{u}\left(\frac{\pi}{2}\right)$$

where $\kappa_1' \leq \kappa_1$ accounts for any leading zeroes in the bits of $\frac{\pi}{2}$ that are being truncated. Accordingly equation (??) yields, for the unit of the last place:

$$\mathfrak{u}\left(\frac{\pi}{2} - C_1\right) = 2^{\kappa_1' - M}\, \mathfrak{u}\left(\frac{\pi}{2}\right)$$

Noting that the absolute error on the rounding that appears in the definition of δC_1 is bounded by $\frac{1}{2}\, \mathfrak{u}\left(\frac{\pi}{2} - C_1\right)$, we obtain the absolute error on the two-term approximation:

$$\left|\frac{\pi}{2} - C_1 - \delta C_1\right| \leq \frac{1}{2}\, \mathfrak{u}\left(\frac{\pi}{2} - C_1\right) = 2^{\kappa_1' - M - 1}\, \mathfrak{u}\left(\frac{\pi}{2}\right) \tag{4}$$

and the following upper bound for δC_1:

$$\begin{aligned} |\delta C_1| &< \frac{\pi}{2} - C_1 + \frac{1}{2}\, \mathfrak{u}\left(\frac{\pi}{2} - C_1\right) \\ &< 2^{\kappa_1'}\, \mathfrak{u}\left(\frac{\pi}{2}\right) + 2^{\kappa_1' - M - 1}\, \mathfrak{u}\left(\frac{\pi}{2}\right) = 2^{\kappa_1'}\, (1 + 2^{-M-1})\, \mathfrak{u}\left(\frac{\pi}{2}\right) \end{aligned} \tag{5}$$

This scheme gives a representation with a significand that has effectively $2M - \kappa_1'$ bits and is such that multiplying C_1 by an integer less than or equal to 2^{κ_1} is exact.

Three-Term Approximation

In this scheme we approximate $\frac{\pi}{2}$ as the sum of three floating-point numbers:

$$\frac{\pi}{2} \simeq C_2 + C_2' + \delta C_2$$

which are defined as:

$$\begin{cases} C_2 & := \mathrm{Tr}\left(\kappa_2, \frac{\pi}{2}\right) \\ C_2' & := \mathrm{Tr}\left(\kappa_2, \frac{\pi}{2} - C_2\right) \\ \delta C_2 & := \left[\!\left[\frac{\pi}{2} - C_2 - C_2'\right]\!\right] \end{cases}$$

Equation (??) applied to the definition of C_2 yields:

$$2^{\kappa_2'-1}\,\mathfrak{u}\left(\frac{\pi}{2}\right) \leq \frac{\pi}{2} - C_2 < 2^{\kappa_2'}\,\mathfrak{u}\left(\frac{\pi}{2}\right) \tag{6}$$

where $\kappa_2' \leq \kappa_2$ accounts for any leading zeroes in the bits of $\frac{\pi}{2}$ that are being truncated. Accordingly equation (??) yields, for the unit of the last place:

$$\mathfrak{u}\left(\frac{\pi}{2} - C_2\right) = 2^{\kappa_2'-M}\,\mathfrak{u}\left(\frac{\pi}{2}\right)$$

Similarly, equation (??) applied to the definition of C_2' yields:

$$\begin{aligned} 2^{\kappa_2''-1}\,\mathfrak{u}\left(\frac{\pi}{2} - C_2\right) \leq {} & \frac{\pi}{2} - C_2 - C_2' < 2^{\kappa_2''}\,\mathfrak{u}\left(\frac{\pi}{2} - C_2\right) \\ 2^{\kappa_2'+\kappa_2''-M-1}\,\mathfrak{u}\left(\frac{\pi}{2}\right) \leq {} & \qquad\qquad\quad < 2^{\kappa_2'+\kappa_2''-M}\,\mathfrak{u}\left(\frac{\pi}{2}\right) \end{aligned}$$

where $\kappa_2'' \leq \kappa_2$ accounts for any leading zeroes in the bits of $\frac{\pi}{2} - C_2$ that are being truncated. Note that normalization of the significand of $\frac{\pi}{2} - C_2$ effectively drops the zeroes at positions κ_2 to κ_2' and therefore the computation of C_2' applies to a significand aligned on position κ_2'.

It is straightforward to transform these inequalities using (??) to obtain bounds on C_2':

$$2^{\kappa_2'}\left(\frac{1}{2} - 2^{\kappa_2''-M}\right)\mathfrak{u}\left(\frac{\pi}{2}\right) < C_2' < 2^{\kappa_2'}(1 - 2^{\kappa_2''-M-1})\,\mathfrak{u}\left(\frac{\pi}{2}\right)$$

Equation (??) applied to the definition of C_2' yields, for the unit of the last place:

$$\begin{aligned} \mathfrak{u}\left(\frac{\pi}{2} - C_2 - C_2'\right) &= 2^{\kappa_2''-M}\,\mathfrak{u}\left(\frac{\pi}{2} - C_2\right) \\ &= 2^{\kappa_2'+\kappa_2''-2M}\,\mathfrak{u}\left(\frac{\pi}{2}\right) \end{aligned}$$

Noting that the absolute error on the rounding that appears in the definition of δC_2 is bounded by $\frac{1}{2}\,\mathfrak{u}\left(\frac{\pi}{2} - C_2 - C_2'\right)$, we obtain the absolute error on the three-term approximation:

$$\left|\frac{\pi}{2} - C_2 - C_2' - \delta C_2\right| \leq \frac{1}{2}\,\mathfrak{u}\left(\frac{\pi}{2} - C_2 - C_2'\right) = 2^{\kappa_2'+\kappa_2''-2M-1}\,\mathfrak{u}\left(\frac{\pi}{2}\right) \tag{7}$$

and the following upper bound for δC_2:

$$|\delta C_2| < 2^{\kappa_2'+\kappa_2''-M}(1 + 2^{-M-1})\,\mathfrak{u}\left(\frac{\pi}{2}\right) \tag{8}$$

This scheme gives a representation with a significand that has effectively $3M - \kappa_2' - \kappa_2''$ bits and is such that multiplying C_2 and C_2' by an integer less than or equal to 2^{κ_2} is exact.

Argument Reduction

Given an argument x, the purpose of argument reduction is to compute a pair of floating-point numbers $(\hat{x}, \delta\hat{x})$ such that:

$$\begin{cases} \hat{x} + \delta\hat{x} \cong x \quad (\text{mod } \dfrac{\pi}{2}) \\ \hat{x} \text{ is approximately in } \left[-\dfrac{\pi}{4}, \dfrac{\pi}{4}\right] \\ |\delta\hat{x}| \leq \dfrac{1}{2}\,\mathrm{u}(\hat{x}) \end{cases}$$

Argument Reduction for Small Angles

If $|x| < \left\llbracket \frac{\pi}{4} \right\rrbracket$ then $\hat{x} = x$ and $\delta\hat{x} = 0$.

Argument Reduction Using the Two-Term Approximation

If $|x| \leq 2^{\kappa_1}\left\llbracket \frac{\pi}{2} \right\rrbracket$ we compute:

$$\begin{cases} n & = \left\llbracket \left\llbracket x \left\llbracket \dfrac{2}{\pi} \right\rrbracket \right\rrbracket \right\rrbracket \\ y & = x - n\ C_1 \\ \delta y & = \llbracket n\ \delta C_1 \rrbracket \\ (\hat{x}, \delta\hat{x}) & = \text{TwoDifference}(y, \delta y) \end{cases}$$

The first thing to note is that $|n| \leq 2^{\kappa_1}$. We have:

$$|x| \leq 2^{\kappa_1}\left\llbracket \frac{\pi}{2} \right\rrbracket = 2^{\kappa_1}\frac{\pi}{2}(1 + \delta_1)$$

and:

$$\left\llbracket x \left\llbracket \frac{2}{\pi} \right\rrbracket \right\rrbracket = x\frac{2}{\pi}(1 + \delta_2)(1 + \delta_3) \tag{9}$$

from which we deduce the upper bound:

$$\begin{aligned} |n| &\leq \left\lceil 2^{\kappa_1}\frac{\pi}{2}(1 + \delta_1)\frac{2}{\pi}(1 + \delta_2)(1 + \delta_3) \right\rfloor \\ &\leq \lceil 2^{\kappa_1}(1 + \gamma_3) \rfloor \end{aligned}$$

If $2^{\kappa_1}\gamma_3$ is small enough (less that 1/2), the rounding cannot cause n to exceed 2^{κ_1}. In practice we choose a relatively small value for κ_1, so this condition is met.

Now if x is close to an odd multiple of $\frac{\pi}{4}$ it is possible for misrounding to happen. In the following analysis we assume that $n > 0$. The results are symmetrical if $n < 0$. There are two possible kinds of misrounding, with different bounds.

A misrounding of the first kind occurs if:

$$x < \left(n - \frac{1}{2}\right)\frac{\pi}{2} \quad \text{and} \quad \left\llbracket x \left\llbracket \frac{2}{\pi} \right\rrbracket \right\rrbracket > n - \frac{1}{2}$$

Using equation (??) we find that this misrounding is possible iff:

$$x > \frac{\pi}{2}\left(n - \frac{1}{2}\right)\frac{1}{(1 + \delta_2)(1 + \delta_3)} \geq \frac{\pi}{2}\left(n - \frac{1}{2}\right)\frac{1}{(1 + \gamma_2)}$$

In which case the computation of n results in:

$$n\frac{\pi}{2} - x < \frac{\pi}{4}\left(1 + \frac{\gamma_2}{1 + \gamma_2}(2n - 1)\right)$$

This bound tells us that the absolute value of the reduced angle may exceed $\frac{\pi}{4}$ by as much as:

$$\frac{\pi}{4}\frac{\gamma_2}{1 + \gamma_2}(2^{\kappa_1 + 1} - 1) \tag{10}$$

A misrounding of the second kind occurs if:

$$x > \left(n + \frac{1}{2}\right)\frac{\pi}{2} \quad \text{and} \quad \left\llbracket x \left\llbracket \frac{2}{\pi} \right\rrbracket \right\rrbracket < n + \frac{1}{2}$$

A derivation similar to the one above gives the following condition for this misrounding to be possible. Using equation (??):

$$x < \frac{\pi}{2}\left(n + \frac{1}{2}\right)\frac{1}{(1+\delta_2)(1+\delta_3)} \leq \frac{\pi}{2}\left(n + \frac{1}{2}\right)(1 + \gamma_2)$$

from which we derive the bound:

$$x - n\frac{\pi}{2} < \frac{\pi}{4}(1 + \gamma_2(2n+1))$$

and thus the excess above $\frac{\pi}{4}$:

$$\frac{\pi}{4}\gamma_2(2^{\kappa_1+1} + 1) \tag{11}$$

The bounds (??) and (??) need to be taken into account when building the accurate tables.

Using the bound on $|n|$ and the fact that C_1 has κ_1 trailing zeroes, we see that the product $n\ C_1$ is exact. The subtraction $x - n\ C_1$ is exact by Sterbenz's Lemma. Finally, the last step performs an exact addition[2] using algorithm 4 of [HLB08].

To compute the overall error on argument reduction[3], first remember that, from equation (??), we have:

$$C_1 + \delta C_1 = \frac{\pi}{2} + \zeta \quad \text{with} \quad |\zeta| \leq 2^{\kappa_1' - M - 1}\,\mathrm{u}\left(\frac{\pi}{2}\right)$$

The error computation proceeds as follows:

$$\begin{aligned} y - \delta y &= x - n\ C_1 - n\ \delta C_1(1 + \delta_4) \\ &= x - n(C_1 + \delta C_1) - n\ \delta C_1\ \delta_4 \\ &= x - n\frac{\pi}{2} - n(\zeta + \delta C_1\ \delta_4) \end{aligned}$$

from which we deduce an upper bound on the absolute error of the reduction:

$$\begin{aligned} \left| y - \delta y - \left(x - n\frac{\pi}{2}\right)\right| &\leq 2^{\kappa_1} 2^{\kappa_1'}(2^{-M-1} + 2^{-M} + 2^{-2M-1})\,\mathrm{u}\left(\frac{\pi}{2}\right) \\ &= 2^{\kappa_1 + \kappa_1' - M}\left(\frac{3}{2} + 2^{-M-1}\right)\mathrm{u}\left(\frac{\pi}{2}\right) \\ &< 2^{\kappa_1 + \kappa_1' - M + 1}\,\mathrm{u}\left(\frac{\pi}{2}\right) \end{aligned}$$

where we have used the upper bound for δC_1 given by equation (??).

In the computation of the trigonometric functions, we need $\hat{x} + \delta\hat{x}$ to provide enough accuracy that the final result is correctly rounded most of the time, and that

[2] The more efficient QuickTwoDifference is not usable here. First, note that $|y|$ is equal to $\mathrm{u}(x)$ if we take x to be the successor or the predecessor of nC_1 for any n. Ignoring rounding errors we have:

$$|\delta y| \geq n\ 2^{\kappa_1' - 1}\,\mathrm{u}\left(\frac{\pi}{2}\right) \geq 2^{\kappa_1' + M - 2}\,\mathrm{u}\left(\frac{\pi}{2}\right)\mathrm{u}(n)$$

where we used the bound given by equation (??). Now the computation of n can result in a value that is either in the same binade or in the binade below that of x. Therefore $\mathrm{u}(n) \geq \frac{1}{2}\,\mathrm{u}(x)$ and the above inequality becomes:

$$|\delta y| \geq 2^{\kappa_1' + M - 3}\,\mathrm{u}\left(\frac{\pi}{2}\right)\mathrm{u}(x)$$

plugging $\mathrm{u}\left(\frac{\pi}{2}\right) = 2^{1-M}$ we find:

$$|\delta y| \geq 2^{\kappa_1' - 2}\,\mathrm{u}(x)$$

Therefore, as long as $\kappa_1' > 2$, there exist arguments x for which $|\delta y| > |y|$.

[3] Note that this error analysis is correct even in the face of misrounding. Misrounding can combine with the argument reduction error, though, to cause $|y - \delta y|$ to move farther above $\frac{\pi}{4}$

any case of incorrect rounding may be detected. The above error bound shows that, if $\hat{x}$ is very small (i.e., if x is very close to a multiple of $\frac{\pi}{2}$), the two-term approximation may not provide enough correct bits. Formally, say that we want to have $M + \kappa_3$ correct bits in the mantissa of $\hat{x} + \delta\hat{x}$. The error must be less than $2^{-\kappa_3}$ half-units of the last place of the result:

$$2^{\kappa_1+\kappa_1'-M+1}\,\mathrm{u}\left(\frac{\pi}{2}\right) \le 2^{-\kappa_3-1}|\mathrm{u}(\hat{x})| \le 2^{-\kappa_3-M}|\hat{x}|$$

which leads to the following condition on the reduced angle:

$$|\hat{x}| \ge 2^{\kappa_1+\kappa_1'+\kappa_3+1}\,\mathrm{u}\left(\frac{\pi}{2}\right) = 2^{\kappa_1+\kappa_1'+\kappa_3-M+2}$$

The rest of the implementation assumes that $\kappa_3 = 18$ to achieve correct rounding most of the time and detect cases of dangerous rounding. If we choose $\kappa_1 = 8$ we find that $\kappa_1' = 5$ (because there are three consecutive zeroes at this location in the significand of $\frac{\pi}{2}$) and the desired accuracy is obtained as long as $|\hat{x}| \ge 2^{-20} \simeq 9.5 \times 10^{-7}$.

Argument Reduction Using the Three-Term Approximation

If $|x| \le 2^{\kappa_2}\left\llbracket\frac{\pi}{2}\right\rrbracket$ we compute:

$$\begin{cases} n & = \left\llbracket\left\llbracket x \left\llbracket\frac{2}{\pi}\right\rrbracket\right\rrbracket\right\rrbracket \\ y & = x - n\ C_2 \\ y' & = n\ C_2' \\ \delta y & = \llbracket n\ \delta C_2 \rrbracket \\ (z, \delta z) & = \mathrm{QuickTwoSum}(y', \delta y) \\ (\hat{x}, \delta\hat{x}) & = \mathrm{LongSub}(y, (z, \delta z)) \end{cases}$$

The products $n\ C_2$ and $n\ C_2'$ are exact thanks to the κ_2 trailing zeroes of C_2 and C_2'. The subtraction $x - n\ C_2$ is exact by Sterbenz's Lemma. QuickTwoSum performs an exact addition using algorithm 3 of [HLB08]; it is usable in this case because clearly $|\delta y| < |y'|$. LongSub is the obvious adaptation of the algorithm LongAdd presented in section 5 of [Lin81], which implements precise (but not exact) double-precision arithmetic.

It is straightforward to show, like we did in the preceding section, that:

$$|n| \le \lceil 2^{\kappa_2}(1+\gamma_3) \rfloor$$

and therefore that $|n| \le 2^{\kappa_2}$ as long as $2^{\kappa_2}\gamma_3 < 1/2$. Similarly, the misrounding bounds (??) and (??) are applicable with κ_2 replacing κ_1.

To compute the overall error on argument reduction, first remember that, from equation (??), we have:

$$C_2 + C_2' + \delta C_2 = \frac{\pi}{2} + \zeta_1 \quad \text{with} \quad |\zeta_1| \le 2^{\kappa_2'+\kappa_2''-2M-1}\,\mathrm{u}\left(\frac{\pi}{2}\right)$$

Let ζ_2 be the relative error introduced by LongAdd. Table 1 of [Lin81] indicates that $|\zeta_2| < 2^{2-2M}$. The error computation proceeds as follows:

$$\begin{aligned} y - y' - \delta y &= (x - n\ C_2 - n\ C_2' - n\ \delta C_2(1+\delta_4))(1+\zeta_2) \\ &= \left(x - n\frac{\pi}{2} - n(\zeta_1 + \delta C_2\ \delta_4)\right)(1+\zeta_2) \\ &= x - n\frac{\pi}{2} - n(\zeta_1 + \delta C_2\ \delta_4)(1+\zeta_2) + \left(x - n\frac{\pi}{2}\right)\zeta_2 \end{aligned}$$

from which we deduce an upper bound on the absolute error of the reduction, noting that $\left|x - n\frac{\pi}{2}\right| \le \frac{\pi}{4}$:

$$\begin{aligned}
&\left|y - y' - \delta y - \left(x - n\frac{\pi}{2}\right)\right| \\
&\quad \le 2^{\kappa_2+\kappa_2'+\kappa_2''}\left(2^{-2M-1} + 2^{-2M} + 2^{-3M-1}\right)\left(1 + 2^{2-2M}\right)\mathfrak{u}\left(\frac{\pi}{2}\right) + 2^{2-2M}\frac{\pi}{4} \\
&\quad = 2^{\kappa_2+\kappa_2'+\kappa_2''-2M}\left(\frac{3}{2} + 2^{-M-1}\right)\left(1 + 2^{2-2M}\right)\mathfrak{u}\left(\frac{\pi}{2}\right) + 2^{-2M}\,\pi \\
&\quad < 2^{\kappa_2+\kappa_2'+\kappa_2''-2M+1}\,\mathfrak{u}\left(\frac{\pi}{2}\right) + 2^{-2M}\,\pi
\end{aligned}$$

A sufficient condition for the reduction to guarantee κ_3 extra bits of accuracy is for this error to be less than $2^{-\kappa_3-1}|\mathfrak{u}(\hat{x})|$ which itself is less than $2^{-\kappa_3-M}|\hat{x}|$. Therefore we want:

$$\begin{aligned}
|\hat{x}| &\ge 2^{\kappa_3-M}\left(2^{\kappa_2+\kappa_2'+\kappa_2''+1}\,\mathfrak{u}\left(\frac{\pi}{2}\right) + \pi\right) \\
&= 2^{\kappa_3-M}\left(2^{\kappa_2+\kappa_2'+\kappa_2''-M+2} + \pi\right)
\end{aligned}$$

and it is therefore sufficient to have:

$$|\hat{x}| \ge 2^{\kappa_3-M}\left(2^{\kappa_2+\kappa_2'+\kappa_2''-M+2} + 4\right)$$

If we choose $\kappa_3 = 18$ as above, and $\kappa_2 = 18$ we find that $\kappa_2' = 14$ and $\kappa_2'' = 15$. Therefore, the desired accuracy is obtained as long as $|\hat{x}| \ge 65 \times 2^{-39} \simeq 1.2 \times 10^{-10}$.

Fallback

If any of the conditions above is not met, we fall back on the CORE-MATH implementation.

Accurate Tables and Their Generation

Polynomial Approximations

The *Mathematica* function `GeneralMiniMaxApproximation` produces a minimax polynomial $p(x)$ approximating a function $f(x)$ by minizing the quantity $\frac{f(x)-p(x)}{g(x)}$. By choosing $g(x)$ appropriately, we can obtain an approximation that minimizes either the absolute or relative error on the result.

Sin Near Zero

References

[GB91] S. Gal and B. Bachelis. “An Accurate Elementary Mathematical Library for the IEEE Floating Point Standard”. In: *ACM Transactions on Mathematical Software* 17.1 (Mar. 1991), pp. 26–45.

[Hig02] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, 2002.

[HLB08] Y. Hida, X. S. Li and D. H. Bailey. “Library for Double-Double and Quad-Double Arithmetic”. Preprint at `https://www.davidhbailey.com/dhbpapers/qd.pdf`. 8th May 2008.

[Lin81] S. Linnainmaa. “Software for Doubled-Precision Floating-Point Computations”. In: *ACM Transactions on Mathematical Software* 7.3 (Sept. 1981), pp. 272–283.
DOI: `10.1145/355958.355960`.

[Mul+10] 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, 2010.

[SZ05] D. Stehlé and P. Zimmermann. "Gal's accurate tables method revisited". In: *17th IEEE Symposium on Computer Arithmetic (ARITH'05)* (Cape Cod, MA, USA, 27th–29th June 2005). Ed. by P. Montuschi and E. Schwarz. IEEE Computer Society, June 2005, pp. 257–264.
DOI: 10.1109/ARITH.2005.24.

[SZG22] A. Sibidanov, P. Zimmermann and S. Glondu. "The CORE-MATH Project". In: *2022 IEEE 29th Symposium on Computer Arithmetic (ARITH).* IEEE, Sept. 2022, pp. 26–34.
DOI: 10.1109/ARITH54963.2022.00014.
eprint: https://inria.hal.science/hal-03721525v3/file/core-math-final.pdf.

[ZSG+24] P. Zimmermann, A. Sibidanov, S. Glondu et al. *The CORE-MATH Project.* Software. Apr. 2024.