Modeling the x87 Transcendental Instructions with Elementary Polynomial Approximations

David M. Russinoff

December 16, 2007

1 Introduction

The x87 floating-point instruction set includes eight instructions that compute a variety of trigonometric, logarithmic, and exponential functions: FSIN ($\sin x$), FCOS ($\cos x$), FSINCOS ($\sin x$ and $\cos x$), FPTAN ($\tan x$), FPATAN ($\arctan x$), F2XM1 ($2^x - 1$), FYL2X ($y \cdot \log_2 x$), and FYL2XP1 ($y \cdot \log_2(x+1)$). The inherent difficulty of computing these functions accurately presents a challenge in their implementation, verification, and documentation. Consequently, although this issue is not adequately addressed by the standard programming manuals [1, 3], their specifications cannot realistically be as rigid as those of other floating-point instructions.

In contrast, the elementary arithmetic operations are fully specified by the IEEE standard [2], which requires the returned value to be the result of rounding the precise mathematical value of the modeled function according to a rounding mode determined by the processor state. For the operations of division and square root extraction, this is normally achieved by first computing an approximation of the true value and then determining whether it is an overestimate or underestimate by applying the inverse operation. For example, given an approximation z to the square root of an operand x, the proper direction of rounding may be determined by comparing z^2 to x.

For a transcendental function, while there are efficient algorithms for computing accurate approximations, the problem of correct rounding is intractable because the inverse function cannot be evaluated precisely any more easily than the function itself. Consequently, a typical implementation simply computes an approximation and rounds it. Regardless of the accuracy of this approximation, if it happens to be close to a rounding boundary, it may not round to the same result as the true mathematical value. In such cases, the specification of a transcendental instruction must provide two admissible rounded results.

Error analysis of floating-point operations is commonly based on a measurement that we shall call *exponent-relative error*, by which the error of an approximation v with respect to a true value v_0 is computed as

$$\left|\frac{v - v_0}{2^k}\right|,$$

where k is the integer satisfying $2^k \leq |v_0| < 2^{k+1}$. A related quantity is the *ulp* (unit in the last place), which depends on the precision of the target format: for a format with n bits of precision, an ulp (of v_0) is the absolute error corresponding to an exponent-relative error of $1/2^{n-1}$.

In the case of the x87 transcendental instructions, the results of which are encoded in the double extended precision format, an ulp corresponds to an exponent-relative error of 2^{-63}. Intel makes the following claim regarding its products [3]:

> With the Pentium processor and later IA-32 processors, the worst case error on transcendental instructions is less than 1 ulp when rounding to the nearest (even) and less than 1.5 ulps when rounding in other modes.

This approach to specifying the accuracy of floating-point instructions suffers from two deficiencies.

The first is the conflation of approximation error and rounding error, which accounts for the consideration of rounding mode in the specification quoted above. In both cases listed, the error allowed in the rounded result is apparently intended to accommodate an error of .5 ulp in the approximation from which it is derived. A more direct approach would be to impose an error bound of .5 ulp on the unrounded approximation and to require that it be rounded correctly with respect to the indicated mode.

The second is in the notion of exponent-relative error itself. Although it has the advantage of being simply related to absolute error, it is less suitable for measuring approximation error than rounding error because of its arbitrary dependence on the proximity of v_0 to the nearest power of 2. The most meaningful measure of the accuracy of a numerical algorithm, from a design or verification perspective, is the standard notion of *relative error*, defined simply as

$$\left|\frac{v - v_0}{v_0}\right|.$$

As a consequence of the definitions, if the relative error of an approximation is ϵ, then its exponent-relative error may lie anywhere in the interval $[\epsilon, 2\epsilon)$. It follows that the weakest bound on the relative error of approximation that ensures an exponent-relative error bound of 2^{-64}, or .5 ulp, and therefore guarantees Intel's criterion of accuracy for the transcendental intructions, is 2^{-65}. Thus, we propose the following as a specification of correctness: *The result returned by a transcendental intruction must be derived by correctly rounding an approximation v of the precise value v_0 that satisfies*

$$\left|\frac{v - v_0}{v_0}\right| < 2^{-65}.$$

Implementations that meet the Intel criterion generally satisfy this specification as well, even though it is somewhat stricter, allowing only one or two admissible rounded results depending on the proximity of v_0 to a rounding boundary, whereas Intel allows at least two admissible results and as many as four in some cases. In fact, today's commercial x86 processors are designed to generate approximations of the transcendental functions that are well within this range of error. However, because of the usual emphasis on efficiency, their designs are based on sophisticated algorithms and optimizations that are difficult to analyze. Consequently, confidence in their correctness cannot be achieved without extensive testing, typically through co-simulation with a trusted software model.

The main difference between the design of such a model and that of a hardware implementation is that since execution efficiency of the model is not an overriding concern, it may be based on a simpler algorithm, one that is more susceptible to formal analysis. On the other hand, the model faces the same issues with respect to accuracy as the implementation. Thus, not only might a compliant implementation produce a

result that differs from that of the model, but the model cannot be expected in all cases to validate that result with respect to the specification stated above, since this would require an absolutely precise computation.

However, if the approximations computed by the software model and the hardware implementation are both sufficiently accurate, then it is possible for the model to identify a range of values such that the following conditions hold:

(a) Every value in this range satisfies the specified relative error bound of 2^{-65}.

(b) The approximation computed by the implementation may be expected to lie within this range.

It follows from (a) that for any IEEE rounding mode, the two endpoints of the range either round to the same 64-bit value or round to two consecutive 64-bit values, and from (b) that the final result returned by the implementation coincides with (at least) one of these rounded values. This provides a test that the implementation will fail if it returns an incorrect result, and will pass if its approximation is as accurate as it is supposed to be.

In order to make this strategy concrete, let v_s and v_h be the approximations computed by the software model and the hardware implementation, respectively, and suppose that we have a known relative error bound for v_s of 2^{-68} and a conjectured bound for v_h of 2^{-66}, neither of which is unrealistic. Thus, if the precise targeted value is v_0, then

$$v_0(1 - 2^{-68}) < v_s < v_0(1 + 2^{-68})$$

and

$$v_0(1 - 2^{-66}) < v_h < v_0(1 + 2^{-66}).$$

It follows that

$$v_h < v_0(1 + 2^{-66}) < v_s \frac{1 + 2^{-66}}{1 - 2^{-68}} < v_0 \frac{(1 + 2^{-68})(1 + 2^{-66})}{1 - 2^{-68}} < v_0(1 + 2^{-65})$$

and

$$v_h > v_0(1 - 2^{-66}) > v_s \frac{1 - 2^{-66}}{1 + 2^{-68}} > v_0 \frac{(1 - 2^{-68})(1 - 2^{-66})}{1 + 2^{-68}} > v_0(1 - 2^{-65}).$$

Thus, if the implementation is as accurate as advertised, then its approximation must be verifiably within the range

$$v_s \frac{1 - 2^{-66}}{1 + 2^{-68}} < v_h < v_s \frac{1 + 2^{-66}}{1 - 2^{-68}}, \tag{1}$$

and from this it follows that

$$v_0(1 - 2^{-65}) < v_h < v_0(1 + 2^{-65}). \tag{2}$$

Consequently, correctness of the implementation as expressed by (2) may be verified by comparing the returned rounded value to the two rounded results produced by the extreme values of the range given by (1).

Our objective, then, is to design an algorithm that may be easily understood and computes a rational approximation that may be rigorously verified to satisfy a strict relative error bound of 2^{-68}, for each of the functions of interest:

- $\sin x$, $\cos x$, and $\tan x$, for $-\frac{\pi}{4} \leq x \leq \frac{\pi}{4}$;
- $\log_2 x$, for $x > 0$;
- $2^x - 1$, for $-1 \leq x \leq 1$;
- $\arctan x$, for $-1 \leq x \leq 1$.

We acknowledge that the domains cited for the trigonometric instructions represent small subsets of the intervals on which these instructions actually operate. In fact, the claims of accuracy are not valid outside of these restricted domains. In particular, although FSIN, FCOS, FSINCOS, and FPTAN compute numerical results for all operands in the range $-2^{63} < x < 2^{63}$, this is achieved through a reduction procedure that produces grossly inaccurate results for $|x| > \frac{\pi}{4}$. This problem is well known but has been tolerated in the interest of backward compatibility. It remains a mystery, however, why Intel insists, in its published documentation, on maintaining the myth that the reduction is designed "to guarantee no loss of significance in a source operand, provided the operand is within the specified range for the instruction." [3]

In this note, we present a set of algorithms for the functions listed above along with proofs of the required error bounds. All of the proofs are elementary enough to be readily understood by a first year calculus student. The deepest result used is the following restricted form of Taylor's Theorem, which pertains to the Taylor series expansion of $f(x)$ about $x = 0$, known as the Maclaurin series.

Theorem 1 (Taylor) *Let f be a function that is continuous together with its first $n+1$ derivatives on an interval containing 0 and x. Then*

$$f(x) = P_n(x) + R_n(x),$$

where

$$P_n(x) = f(0) + f'(0) \cdot x + \frac{f''(0)}{2!}x^2 + \cdots + \frac{f^{(n)}(0)}{n!}x^n = \sum_{k=0}^{n} \frac{f^{(k)}(0)}{k!}x^k$$

and

$$R_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!}x^{n+1},$$

for some c between 0 and x.

Although execution efficiency is not our first priority, our algorithms must admit implementations that are fast enough to be of practical use in co-simulation. In most cases, a Taylor series approximation of at most sixty-five terms is sufficient for this purpose. The exception is arc tangent, for which the Taylor series converges so slowly that thousands of terms would be required to achieve the required accuracy. For this case, rather than appeal to more advanced methods, we refer to an elementary result of H. Medina [4], which provides a polynomial of degree 55 with the requisite accuracy. The first theorem stated below describes a sequence of recursively defined polynomials $h_m(x)$ of degree $8m-1$, of which we shall make use of $h_7(x)$. The second provides a closed form for the coefficients of $h_m(x)$ that allows them to be computed independently. Neither of the proofs of these results requires any mathematics beyond elementary calculus.

Theorem 2 (Medina) *Let* $p_1(x) = 4 - 4x^2 + 5x^4 - 4x^5 + x^6$ *and for* $m \geq 2$,

$$p_m(x) = x^4(1-x)^4 p_{m-1}(x) + (-4)^{m-1} p_1(x).$$

Let

$$h_m(x) = \int_0^x \frac{(-1)^{m+1}}{4^m} p_m(t)dt.$$

Then for all $x \in [0,1]$,

$$|h_m(x) - \arctan\, x| \leq \left(\frac{1}{4}\right)^{5m}.$$

Theorem 3 (Medina) *For* $m = 1, 2, \ldots$,

$$h_m(x) = \sum_{j=1}^{2m} \frac{(-1)^{j+1}}{2j-1} x^{2j-1} + \sum_{j=0}^{4m-2} \frac{a_j}{(-1)^{m+1} 4^m (4m+j+1)} x^{4m+j+1},$$

where

$$a_{2i} = (-1)^{i+1} \sum_{k=i+1}^{2m} \binom{4m}{2k} (-1)^k$$

and

$$a_{2i-1} = (-1)^{i+1} \sum_{k=i}^{2m-1} \binom{4m}{2k+1} (-1)^k.$$

2 Trigonometric Functions

The Taylor series for $\sin x$ and $\cos x$ are readily derived from the equations $\frac{d}{dx}\sin x = \cos x$ and $\frac{d}{dx}\cos x = -\sin x$; the inequalities $|\sin x| \leq 1$ and $|\cos x| \leq 1$ provide simple estimates of the remainders. Note that our error bound for these approximations is strengthened to 2^{-70} so that they may be used to derive a suitably accurate approximation of $\tan x$.

Proposition 2.1 *Let* $a \in \mathbb{R}$ *with* $0 < a \leq \frac{\pi}{4}$ *and let*

$$\sigma = \sum_{k=1}^{11} \frac{(-1)^{k-1} a^{2k-1}}{(2k-1)!}.$$

Then

$$\left| \frac{\sigma - \sin\, a}{\sin\, a} \right| < 2^{-70}.$$

PROOF: Applying Taylor's Theorem, we have, for all $n \in \mathbb{N}$ and $x \in \mathbb{R}$,

$$\sin x = P_n(x) + R_n(x),$$

where

$$P_{2n}(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \cdots + \frac{(-1)^{n-1} x^{2n-1}}{(2n-1)!} = \sum_{k=1}^{n} \frac{(-1)^{k-1} x^{2k-1}}{(2k-1)!}$$

and

$$|R_{2n}(x)| < \frac{|x|^{2n+1}}{(2n+1)!}.$$

Thus, $\sigma = P_{22}(a)$ and

$$\sin\, a = \sigma + R_{22}(a),$$

where

$$|R_{22}(a)| \leq \frac{|a|^{23}}{23!},$$

and hence,

$$\left|\frac{R_{22}(a)}{a}\right| \leq \frac{|a|^{22}}{23!} < \frac{1}{23!}.$$

On the other hand, by the Mean Value Theorem, there exists c between 0 and a such that

$$\frac{\sin\, a}{a} = \frac{\sin\, a - \sin\, 0}{a - 0} = \frac{d}{dx}\sin\, x|_{x=c} = \cos\, c \geq \cos\, \frac{\pi}{4} = \frac{\sqrt{2}}{2}.$$

Consequently,

$$\left|\frac{\sigma - \sin\, a}{\sin\, a}\right| = \left|\frac{R_{22}(a)}{\sin\, a}\right| = \left|\frac{R_{22}(a)}{a}\right| \left|\frac{a}{\sin\, a}\right| < \frac{1}{23!} \cdot \sqrt{2} < 2^{-70}. \ \square$$

Proposition 2.2 *Let $a \in \mathbb{R}$ with $0 < a \leq \frac{\pi}{4}$ and let*

$$\kappa = \sum_{k=0}^{11} \frac{(-1)^k a^{2k}}{(2k)!}.$$

Then

$$\left|\frac{\kappa - \cos\, a}{\cos\, a}\right| < 2^{-70}.$$

PROOF: Applying Taylor's Theorem, we have, for all $n \in \mathbb{N}$ and $x \in \mathbb{R}$,

$$\cos\, x = P_n(x) + R_n(x),$$

where

$$P_{2n}(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \cdots + \frac{(-1)^n x^{2n}}{(2n)!} = \sum_{k=0}^{n} \frac{(-1)^k x^{2k}}{(2k)!}$$

and

$$|R_{2n}(x)| < \frac{|x|^{2n+1}}{(2n+1)!}.$$

Thus, $\kappa = P_{22}(a)$ and

$$\cos\, a = \kappa + R_{22}(a),$$

where

$$|R_{22}(a)| \leq \frac{|a|^{23}}{23!} < \frac{1}{23!}.$$

Since $\cos\, a \geq \cos\, \frac{\pi}{4} = \frac{\sqrt{2}}{2}$,

$$\left|\frac{\kappa - \cos\, a}{\cos\, a}\right| = \left|\frac{R_{22}(a)}{\cos\, a}\right| < \frac{\sqrt{2}}{23!} < 2^{-70}. \ \square$$

Proposition 2.3 *Let $a \in \mathbb{R}$ with $0 < a \leq \frac{\pi}{4}$ and let $\tau = \frac{\sigma}{\kappa}$, where σ and κ are defined as in Propositions 2.1 and 2.2. Then*

$$\left| \frac{\tau - \tan a}{\tan a} \right| < 2^{-68}.$$

PROOF: Since

$$\tau - \tan a = \frac{\sigma}{\kappa} - \frac{\sin a}{\cos a} = \frac{\sigma - \sin a}{\kappa} + \frac{\sin a(\cos a - \kappa)}{\kappa \cos a} = \frac{\sin a}{\kappa}\left(\frac{\sigma - \sin a}{\sin a} + \frac{\cos a - \kappa}{\cos a} \right),$$

we have

$$\frac{\tau - \tan a}{\tan a} = \frac{\cos a}{\kappa}\left(\frac{\sigma - \sin a}{\sin a} + \frac{\cos a - \kappa}{\cos a} \right).$$

As noted in the proof of Proposition 2.2, $|\cos a - \kappa| < \frac{1}{23!}$, and hence

$$\frac{\kappa}{\cos a} = \frac{\cos a - (\cos a - \kappa)}{\cos a} \geq 1 - \frac{|\cos a - \kappa|}{\cos a} > 1 - \frac{\sqrt{2}}{23!} > \frac{1}{2},$$

which yields

$$\begin{aligned} \frac{\tau - \tan a}{\tan a} &= \left| \frac{\cos a}{\kappa}\left(\frac{\sigma - \sin a}{\sin a} + \frac{\cos a - \kappa}{\cos a} \right) \right| \\ &\leq \left| \frac{\cos a}{\kappa} \right| \left(\left| \frac{\sigma - \sin a}{\sin a} \right| + \left| \frac{\cos a - \kappa}{\cos a} \right| \right) \\ &< 2(2^{-70} + 2^{-70}) \\ &= 2^{-68}. \ \square \end{aligned}$$

3 The Logarithmic Function

Our approximation of $\log_2 x$ is based on the identity

$$\log_2 x = \frac{\ln x}{\ln 2}$$

and the Maclauring series expansion of $\ln(1 + x)$. Note that the estimate of $\ln 2$ derived in the proof is used again in the approximation of 2^x given in the next section.

Proposition 3.1 *For all $n \in \mathbb{N}$ and $x \in \mathbb{R}$, let*

$$P_n(x) = \sum_{k=1}^{n} \frac{(-1)^{k+1} x^k}{k}.$$

Let

$$\chi = -P_{66}\left(-\frac{1}{2} \right) = \sum_{k=1}^{66} \frac{1}{2^k k}.$$

Given $a \in \mathbb{R}$, $a > 0$, let s and b be defined by $a = 2^b s$, with $b \in \mathbb{Z}$ and $\frac{\sqrt{2}}{2} < s < \sqrt{2}$, and let

$$\lambda = b + \frac{1}{\chi} P_{65}(s - 1).$$

Then $0 < \ln 2 - \chi \leq 2^{-72}$ *and*

$$\left|\frac{\log_2 a - \lambda}{\log_2 a}\right| < 2^{-68}.$$

PROOF: By Taylor's Theorem, for $x > -1$,

$$\ln(1+x) = P_n(x) + R_n(x),$$

where

$$|R_n(x)| = \left|\frac{(-1)^n n!}{(1+c)^{n+1}} \cdot \frac{x^{n+1}}{(n+1)!}\right| = \frac{1}{n+1}\left(\frac{|x|}{1+c}\right)^{n+1}$$

for some c with $|c| \leq |x|$. Thus, if $|x| \leq \frac{1}{2}$, then $|R_n(x)| \to 0$ as $n \to \infty$ and

$$\ln(1+x) = \sum_{k=1}^{\infty} \frac{(-1)^{k+1} x^k}{k}.$$

A stricter bound on the remainder may be achieved by observing that

$$|R_n(x)| = \left|\sum_{k=n+1}^{\infty} \frac{(-1)^{k+1} x^k}{k}\right| \leq \sum_{k=n+1}^{\infty} \frac{|x|^k}{k} \leq \frac{|x|^n}{n+1} \sum_{k=1}^{\infty} |x|^k \leq \frac{|x|^n}{n+1}.$$

In particular,

$$0 < \ln 2 - \chi = -\ln\left(1 - \frac{1}{2}\right) + P_{66}\left(-\frac{1}{2}\right) = -R_{66}\left(-\frac{1}{2}\right) \leq \frac{1}{2^{66} \cdot 67} < 2^{-72},$$

and

$$\frac{1}{\chi} - \frac{1}{\ln 2} = \frac{\ln 2 - \chi}{\chi \ln 2} < 4 \cdot 2^{-72} = 2^{-70}.$$

Since

$$\frac{\sqrt{2}}{2} - 1 < s - 1 \leq \sqrt{2} - 1,$$

$|s-1| < \frac{1}{2}$. Also note that

$$\log_2 a = b + \log_2 s,$$

where

$$-\frac{1}{2} < \log_2 s \leq \frac{1}{2},$$

and hence

$$b - \frac{1}{2} < \log_2 a \leq b + \frac{1}{2}.$$

It follows that if $b \geq 1$, then $\log_2 a > \frac{1}{2}$, and if $b \leq 1$, then $\log_2 a \leq \frac{1}{2}$. Thus, whwnever $b \neq 0$, $|\log_2 a| \geq \frac{1}{2} > |s-1|$. Suppose $b = 0$. Then $\log_2 a = \log_2 s$. If $s \leq 1$, then

$$|\log_2 s| = \left|\frac{\ln s}{\ln 2}\right| = \left|\frac{1}{\ln 2}\int_1^s \frac{dt}{t}\right| = \frac{1}{\ln 2}\int_s^1 \frac{dt}{t} \geq \frac{1}{\ln 2}\int_s^1 \frac{dt}{1} = \frac{|s-1|}{\ln 2} > |s-1|.$$

Similarly, if $s > 1$, then

$$\log_2 s = \frac{1}{\ln 2}\int_1^s \frac{dt}{t} \geq \frac{1}{\ln 2}\int_1^s \frac{dt}{s} = \frac{s-1}{s \ln 2} \geq \frac{s-1}{\sqrt{2}\ln 2} > s - 1.$$

Thus, in all cases, we have

$$|\log_2 a| > s - 1.$$

Therefore,

$$\left|\frac{R_n(s-1)}{\log_2 s}\right| < \frac{|s-1|^{n-1}}{n+1} < \frac{1}{2^{n-1}(n+1)}$$

and

$$\left|\frac{P_n(s-1)}{\log_2 s}\right| \leq \sum_{k=1}^{n} |s-1|^{k-1} \leq \sum_{k=1}^{n} \frac{1}{2^{k-1}} < 2.$$

Now

$$\ln s = \ln (1 + s - 1) = P_n(s-1) + R_n(s-1),$$

and hence

$$\log_2 a = b + \log_2 s = b + \frac{\ln s}{\ln 2} = b + \frac{1}{\ln 2}(P_n(s-1) + R_n(s-1)).$$

In particular,

$$\begin{aligned} |\log_2 a - \lambda| &= \left|\frac{1}{\ln 2}(P_n(s-1) + R_n(s-1)) - \frac{1}{\chi}P_{65}(s-1)\right| \\ &\leq \left|\frac{1}{\ln 2} - \frac{1}{\chi}\right| |P_{65}(s-1)| + \frac{1}{\ln 2}|R_{65}(s-1)| \\ &< 2^{-70}|P_{65}(s-1)| + 2|R_{65}(s-1)| \end{aligned}$$

and

$$\left|\frac{\log_2 a - \lambda}{\log_2 a}\right| < 2^{-70} \cdot 2 + 2 \cdot \frac{1}{2^{64}66} < 2^{-68}. \ \square$$

4 The Exponential Function

In the proposition below, we derive an approximation ρ of 2^a based on the definition $2^x = e^{x \ln 2}$ and the Taylor series expansion of e^x. Since our concern is the relative error of $\rho - 1$ as an approximation of $2^a - 1$, we establish a bound on

$$\left|\frac{(\rho - 1) - (2^a - 1)}{2^a - 1}\right| = \left|\frac{2^a - \rho}{2^a - 1}\right|.$$

Proposition 4.1 *Let $a \in \mathbb{R}$ with $0 \leq |a| \leq 1$ and let*

$$\rho = \sum_{k=0}^{22} \frac{(\chi a)^k}{k!},$$

where $\chi = \sum_{k=1}^{66} \frac{1}{2^k k}$. Then

$$\left|\frac{2^a - \rho}{2^a - 1}\right| < 2^{-68}.$$

PROOF: By Taylor's Theorem, since $\frac{d^n}{dx^n}e^x|_{x=0} = e^x|_{x=0} = 1$ for all n, we have, for all $n \in \mathbb{N}$ and $x \in \mathbb{R}$,

$$e^x = P_n(x) + R_n(x),$$

where

$$P_n(x) = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots + \frac{x^n}{n!} = \sum_{k=0}^{n} \frac{x^k}{k!}$$

and

$$|R_n(x)| \leq \frac{\max(e^x, 1)|x|^{n+1}}{(n+1)!}.$$

Note that $\rho = P_{22}(\chi a)$. Thus,

$$2^a - \rho = 2^a - e^{\chi a} + R_{22}(\chi a)$$

and

$$\left|\frac{2^a - \rho}{a}\right| \leq \left|\frac{2^a - e^{\chi a}}{a}\right| + \left|\frac{R_{22}(\chi a)}{a}\right|.$$

By Proposition 3.1, $\chi < \ln 2 < 1$ and $\ln 2 - \chi \leq 2^{-72}$. Consequently,

$$\left|\frac{R_{22}(\chi a)}{a}\right| \leq \left|\frac{\max(e^{\chi a}, 1)(\chi a)^{23}}{23! \cdot a}\right| < \left|\frac{2\chi^{23}a^{22}}{23!}\right| < \frac{2}{23!} < 2^{-70}.$$

By the Mean Value Theorem,

$$\frac{e^{a \ln 2} - e^{\chi a}}{a \ln 2 - \chi a} = \frac{d}{dx}e^x|_{x=c} = e^c$$

for some c between $a \ln 2$ and χa. But then $e^c < e^{a \ln 2} \leq e^{\ln 2} = 2$, and hence

$$|2^a - e^{\chi a}| = |e^{a \ln 2} - e^{\chi a}| \leq 2|a(\ln 2 - \chi)| \leq 2^{-71}|a|$$

and

$$\left|\frac{2^a - e^{\chi a}}{a}\right| \leq 2^{-71}.$$

Thus,

$$\left|\frac{2^a - \rho}{a}\right| \leq 2^{-71} + 2^{-70} < 2^{-69}.$$

Next, we shall derive an estimate of $|(2^a - 1)/a|$. First suppose $a > 0$. Applying the Mean Value Theorem again, we have

$$\frac{2^a - 1}{a} = \frac{2^a - 2^0}{a - 0} = \frac{d}{dx}2^x|_{x=c} = 2^c \ln 2,$$

where $0 \leq c \leq a$, which implies

$$\frac{2^a - 1}{a} \geq \ln 2.$$

Now suppose $-1 \leq a \leq 0$. Let $f(x) = 2^x - x/2$. Since $f''(x) = 2^x(\ln 2)^2 > 0$ for all x, the maximum value of $f(x)$ for $-1 \leq x \leq 0$ must occur at either $x = -1$ or $x = 0$. But then since $f(-1) = 2^{-1} - (-1)/2 = 1$ and $f(0) = 2^0 - 0/2 = 1$, that maximum is 1. In

particular, $f(a) = 2^a - a/2 \leq 1$, which implies $|2^a - 1| = 1 - 2^a \geq -a/2 = |a/2|$ and $|(2^a - 1)/a| \geq \frac{1}{2}$.

Thus, in all cases, $|(2^a - 1)/a| \geq \frac{1}{2}$, and hence

$$\left|\frac{2^a - \rho}{2^a - 1}\right| = \left|\frac{2^a - \rho}{a}\right| \left|\frac{a}{2^a - 1}\right| < 2^{-69} \cdot 2 = 2^{-68}. \ \square$$

5 Arc Tangent

We employ two distinct methods for approximating $\arctan x$, neither of which is sufficient alone for the entire domain $0 < |x| < 1$. For $|x| \leq \frac{1}{2}$, we use the Taylor series, which converges too slowly for arguments close to 1. For $|x| > \frac{1}{2}$, we use Medina's result, which provides a bound on absolute error from which we may derive the required estimate of relative error only if x is bounded away from 0.

Since the computation given by Theorem 1 is unwieldy in this case, we apply a more direct computation in our derivation of the Taylor series. The two methods may be shown to produce the same result, but this is irrelevant to our objective.

Proposition 5.1 *Let $a \in \mathbb{R}$ with $0 \leq |a| \leq 1$ and let*

$$\alpha = \begin{cases} \sum_{k=1}^{32} \frac{(-1)^{k-1} a^{2k-1}}{2k-1} & \text{if } |a| \leq \frac{1}{2} \\ h_7(a) & \text{if } \frac{1}{2} \leq a \leq 1 \\ -h_7(-a) & \text{if } -1 \leq a \leq -\frac{1}{2}. \end{cases}$$

Then

$$\left|\frac{\alpha - \arctan a}{\arctan a}\right| < 2^{-68}.$$

PROOF: First consider the case $a \leq \frac{1}{2}$. By the Mean Value Theorem, for some c, $0 \leq |c| \leq |a|$,

$$\frac{\arctan a}{a} = \frac{d}{dx} \arctan x|_{x=c} = \frac{1}{1 + c^2} \geq \frac{1}{1 + \frac{1}{4}} = \frac{4}{5}.$$

For $n \in \mathbb{N}$, let

$$Q_n(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \cdots + \frac{(-1)^{n-1} x^{2n-1}}{2n-1} = \sum_{k=1}^{n} \frac{(-1)^{k-1} x^{2k-1}}{2k-1}.$$

Beginning with the algebraic identity

$$\frac{1}{1-z} = 1 + z + z^2 + \cdots + z^{n-1} + \frac{z^n}{1-z}$$

and substituting $-x^2$ for z, we hace

$$\frac{1}{1+x^2} = 1 - x^2 + x^4 - \cdots + (-1)^{n-1} x^{2n-2} + \frac{(-1)^n x^{2n}}{1+x^2}.$$

Consequently,

$$\arctan a = \int_0^a \frac{dx}{1+x^2} = Q_n(a) + \int_0^a \frac{(-1)^n x^{2n} dx}{1+x^2},$$

and hence

$$|Q_n(a) - \arctan a| = \left| \int_0^a \frac{(-1)^n x^{2n} dx}{1+x^2} \right| \leq \int_0^{|a|} \frac{x^{2n} dx}{1+x^2} \leq \int_0^{|a|} x^{2n} dx = \frac{|a|^{2n+1}}{2n+1}.$$

In particular, since $\alpha = Q_{32}(a)$,

$$\left| \frac{\alpha - \arctan a}{a} \right| \leq \frac{|a|^{64}}{65} \leq \frac{1}{2^{64}65}.$$

Thus,

$$\left| \frac{\alpha - \arctan a}{\arctan a} \right| = \left| \frac{\alpha - \arctan a}{a} \right| \left| \frac{a}{\arctan a} \right| \leq \frac{1}{2^{64}65} \cdot \frac{5}{4} < 2^{-68}.$$

Now consider the case $|a| > \frac{1}{2}$. If $a > \frac{1}{2}$, then by Theorem 2,

$$|h_7(a) - \arctan a| < \left(\frac{1}{4}\right)^{35} = 2^{-70}.$$

In order to establish a lower bound for arctan a, we apply the identity

$$\tan^2\theta = \frac{1 - \cos 2\theta}{1 + \cos 2\theta},$$

which yields

$$\tan^2 \frac{\pi}{8} = \frac{1 - \frac{\sqrt{2}}{2}}{1 + \frac{\sqrt{2}}{2}} = \frac{2 - \sqrt{2}}{2 + \sqrt{2}} = \frac{(2 - \sqrt{2})^2}{2} < \frac{1}{4},$$

i.e, $\tan \frac{\pi}{8} < \frac{1}{2}$, and hence

$$\arctan a > \arctan \frac{1}{2} > \frac{\pi}{8}.$$

Thus,

$$\left| \frac{\alpha - \arctan a}{\arctan a} \right| = \left| \frac{h_7(a) - \arctan a}{\arctan a} \right| < 2^{-70} \cdot \frac{8}{\pi} < 2^{-68}.$$

On the other hand, if $a < -\frac{1}{2}$, then

$$\left| \frac{\alpha - \arctan a}{\arctan a} \right| = \left| \frac{-h_7(-a) - \arctan a}{\arctan a} \right| = \left| \frac{-h_7(-a) - \arctan(-a)}{\arctan(-a)} \right| < 2^{-68}. \ \square$$

References

[1] Advanced Micro Devices, Inc., *AMD64 Architecture Programmer's Manual*, Rev. 3.11, December 2005.

[2] Institute of Electrical and Electronic Engineers, *IEEE Standard for Binary Floating Point Arithmetic*, Std. 754-1985, New York, N.Y., 1985.

[3] Intel Corporation, *Intel 64 and IA-32 Architectures Software Developer's Manual*, April 2008.

[4] Medina, Herbert A., "A Sequence of Polynomials for Approximating Arctangent", *American Mathematical Monthly* (113), pp.156-161, February 2006.