

Rounding proofs for decimal_gda

Supplement to the decimal_gda design page of Luna-Flow/floating

This note proves the four facts the design page relies on: the error bound of rounding to precision, when two successive roundings equal one, why the certified elementary functions return the correctly rounded result, and the error bound of integer powers. Notation follows the design page: a finite value is $(-1)^s c \cdot 10^e$ with $c \in \mathbb{N}$, $d(c)$ is the number of decimal digits of $c > 0$, and the context precision is $p \geq 1$.

1. Rounding to precision

Let $v = (-1)^s c \cdot 10^e$ be exact with $d(c) = p + k$, $k \geq 1$. Write $c = q \cdot 10^k + r$ with $0 \leq r < 10^k$. Each mode returns $(-1)^s (q + \delta) \cdot 10^{e+k}$ with an increment $\delta \in \{0, 1\}$:

$$\delta_{\text{Down}} = 0, \quad \delta_{\text{Up}} = [r > 0], \quad \delta_{\text{Ceiling}} = [r > 0 \wedge s = 0], \quad \delta_{\text{Floor}} = [r > 0 \wedge s = 1], \quad (1)$$

$$\delta_{\text{HalfUp}} = [2r \geq 10^k], \quad \delta_{\text{HalfDown}} = [2r > 10^k], \quad \delta_{\text{HalfEven}} = [2r > 10^k \vee (2r = 10^k \wedge q \text{ odd})], \quad (2)$$

$$\delta_{\text{ZeroFiveUp}} = [r > 0 \wedge q \equiv 0 \pmod{5}]. \quad (3)$$

The last condition is “the last digit of q is 0 or 5”, because $q \bmod 10 \in \{0, 5\} \Leftrightarrow q \equiv 0 \pmod{5}$.

Lemma 1 (error bound). Let $u = 10^{e+k}$ be the unit in the last place of the result $\hat{v}$. Then $|\hat{v} - v| \leq u/2$ for the three half modes and $|\hat{v} - v| < u$ for the other five. Relative to v ,

$$|\hat{v} - v| / |v| \leq \frac{1}{2} \cdot 10^{1-p} \quad (\text{half modes}), \quad |\hat{v} - v| / |v| < 10^{1-p} \quad (\text{others}). \quad (4)$$

Proof. $|\hat{v} - v| = |\delta \cdot 10^k - r| \cdot 10^e$. For the half modes $\delta = 1$ only if $r \geq 10^k/2$, giving $10^k - r \leq 10^k/2$, and $\delta = 0$ only if $r \leq 10^k/2$. For the other modes $|\delta 10^k - r| < 10^k$ because $0 \leq r < 10^k$. For the relative bound, $c \geq 10^{p+k-1}$, so $|v| \geq 10^{p-1} \cdot u$; divide. If $q + \delta = 10^p$ the result is renormalized to $10^{p-1} \cdot 10^{e+k+1}$, which is the same number, so the bounds are unchanged. $\square$

2. Two roundings and one

Rounding to a finer unit 10^j and then to a coarser unit 10^{j+m} ($m \geq 1$) is called *double rounding*.

Lemma 2. For Down, Up, Ceiling and Floor, double rounding equals a single rounding to 10^{j+m} . For the half modes it does not in general.

Proof. Take Down on magnitudes: truncating to a multiple of 10^j and then to a multiple of 10^{j+m} gives $10^{j+m} \lfloor x/10^{j+m} \rfloor$ because $\lfloor \lfloor x/10^j \rfloor / 10^m \rfloor = \lfloor x/10^{j+m} \rfloor$. Up is the same identity with ceilings; Ceiling and Floor are Up or Down depending on the sign, and the sign is unchanged by the first rounding. For the half modes take $p = 1$ and $x = 1/2222 = 0.000450045\dots$. To the unit 10^{-7} (the first rounding used by divide), half-even gives $4500 \cdot 10^{-7}$, an exact tie for the second rounding to 10^{-4} , which half-even resolves downwards to $4 \cdot 10^{-4}$. A single rounding of x gives $5 \cdot 10^{-4}$ because $x > 4.5 \cdot 10^{-4}$. $\square$

Corollary (division). Let divide form $Q = c_1 10^t / c_2$ with $t = p + d(c_2) + 2$, round Q to an integer q with the context mode, and then round q to p digits. The quotient is correctly rounded in the four directed modes. In a half mode it is wrong exactly when the first rounding manufactures a tie: the digits of q discarded by the second rounding are $50\dots0$ while $Q \neq q$ and the exact discarded part is not one half.

Proof. The directed modes follow from Lemma 2. In a half mode the second rounding compares the discarded part of q with the integer threshold $h = 5 \cdot 10^{m-1}$ (m discarded digits) after removing the retained multiple of 10^m . Since q is the nearest integer to Q and h is an integer, $Q < h$ implies $q \leq h$ and $Q > h$ implies $q \geq h$. Hence the comparison of q with h agrees with that of Q unless $q = h \neq Q$, the manufactured tie. $\square$

The defect is avoided by keeping a sticky bit ($Q \notin \mathbb{Z}$) with q and letting the second rounding treat “50...0 plus sticky” as above half, as `normalize` already does for the other operations.

3. Certified rounding of elementary functions

Fix a mode m and a representable result r of the context. Its *rounding cell* is $C_m(r) = \{y \in \mathbb{R} : \text{round}_m(y) = r\}$. Let r^- and r^+ be the representable neighbours of r and let $\mu(a, b) = (a + b)/2$. The cell is an interval. The package does not need all of it: it uses an open interval $I_m(r) \subseteq C_m(r)$, namely

$$I_m(r) = \begin{cases} (\mu(r^-, r), \mu(r, r^+)) & \text{half modes} \\ (r, r^+) & \text{Floor} \\ (r^-, r) & \text{Ceiling} \end{cases} \quad (5)$$

For Down the interval lies on the side of r away from zero, for Up on the side towards zero, and for ZeroFiveUp on the side of r that contains the approximation being tested; each of these open intervals rounds to r under its mode. All endpoints are computed exactly as decimals (`decimal_power_rounding_cell`).

Lemma 3 (cell certificate). Let $[L, U]$ be a certified enclosure of $f(x)$, let $(a, b) = I_m(r)$, and let $[a^-, a^+] \ni a$ and $[b^-, b^+] \ni b$ be binary enclosures of the endpoints. If $L > a^+$ and $U < b^-$ then $\text{round}_m(f(x)) = r$.

Proof. $a \leq a^+ < L \leq f(x) \leq U < b^- \leq b$, so $f(x) \in (a, b) = I_m(r) \subseteq C_m(r)$. $\square$

Lemma 4 (endpoint certificate). If $\text{round}_m(L) = \text{round}_m(U) = r$ then $\text{round}_m(f(x)) = r$.

Proof. Every rounding mode is a monotone (non-decreasing) map from $\mathbb{R}$ to the representable set. From $L \leq f(x) \leq U$ follows $r = \text{round}_m(L) \leq \text{round}_m(f(x)) \leq \text{round}_m(U) = r$. $\square$

Lemma 5 (termination). If $f(x)$ is neither representable nor a midpoint of two adjacent representable numbers, and the enclosure width tends to 0 as the working precision grows, then the refinement loop certifies after finitely many steps.

Proof. Under the hypothesis $f(x)$ lies in the open interval $I_m(r)$ for $r = \text{round}_m(f(x))$ (for ZeroFiveUp once the approximation is on the same side of r as $f(x)$). Let $\eta > 0$ be its distance to the endpoints. Once both the enclosure width and the endpoint enclosure widths are below $\eta/2$, the hypotheses of Lemma 3 hold. $\square$

For `exp`, `ln` and `log10` the hypothesis holds for every input that is not special-cased: by the Lindemann–Weierstrass theorem e^x is transcendental for rational $x \neq 0$, so $\ln x$ is irrational for rational $x > 0$, $x \neq 1$; and $\log_{10} x = a/b$ with x rational forces $x^b = 10^a$, so x is an integral power of ten. Representable numbers and midpoints are rational, so none of the remaining values can be one. The budget of twelve refinements, each raising the working precision w to $w + \max(32, \lfloor w/2 \rfloor)$, is a resource bound, not part of the argument.

4. Integer powers

power with an integer exponent n , $|n| < 10^D$, multiplies with $w = p + D + 2$ working digits in half-even (one fewer digit in subset contexts) and rounds the final product once more with the context mode.

Lemma 6. In an extended context, the result of an inexact integer power differs from x^n by at most 0.55 units in the last place in the half modes and by less than 1.05 units in the other modes.

Proof. Any multiplication chain that forms $x^{|n|}$ from x is a product of $|n|$ copies of x in which every copy passes through at most $|n| - 1$ roundings, so the computed value is $x^{|n|}(1 + \theta)$ with $|\theta| \leq (1 + \varepsilon)^{|n|-1} - 1 \leq (|n| - 1)\varepsilon / (1 - (|n| - 1)\varepsilon)$, where $\varepsilon = \frac{1}{2} \cdot 10^{1-w}$ is the bound of Lemma 1 (Higham, *Accuracy and Stability of Numerical Algorithms*, Lemma 3.1). For $n < 0$ the initial reciprocal adds one rounding, covered by replacing $|n| - 1$ by $|n|$. Since $|n| < 10^D$,

$$|\theta| \lesssim 10^D \cdot \frac{1}{2} \cdot 10^{1-p-D-2} = \frac{1}{2} \cdot 10^{-1-p}. \quad (6)$$

The final result $\hat{v}$ has unit u with $|x^n| < 10^p u$, so the working error is below $\frac{1}{2} \cdot 10^{-1-p} \cdot 10^p u = 0.05u$. The final rounding adds at most $u/2$ (half modes) or less than u (others). $\square$

The bound is below one unit but above one half, which is why GDA (and this package) does not promise correct rounding for integer powers.