(Version: 2018/9/5)

シンプルな区間数学関数の実装

柏木 雅英 (kashi@waseda.jp)

1 はじめに

区間演算は、精度保証付き数値計算を行うための基礎となる重要な演算である。浮動小数点演算の規格である IEEE Standard for Binary Floating-Point Arithmetic (IEEE Std 754、以後単に IEEE 754 と書く) には丸めの向きを変更する機能があるが、これは区間演算の実装を念頭に置いて要請されたものである。

IEEE 754 において、加減乗除と平方根に関しては、

- 最近点丸め (無誤差の計算値をそれに最も近い浮動小数点数に丸める) を行える。
- 丸めの向きを、
  - 最近点丸め (nearest)
  - $+\infty$ 方向への丸め (up)
  - $-\infty$ 方向への丸め (down)
  - 0 方向への丸め (chop)

  の 4 通りに切り替えることが出来る。

ことが要請されている。しかし、それ以外の数学関数に関しては、誤差に関する規定は無い。また、丸めの向きを変えることも出来ない。よって、IEEE 754 を用いて区間演算を実装しようとすると、加減乗除と平方根以外の演算をどう実装するかという問題に突き当たる。

現状、ある程度信頼できると思われる数学関数の実装は、

- Rump による INTLAB (http://www.ti3.tu-harburg.de/rump/intlab/) の中の数学関数の実装
- crlibm (http://lipforge.ens-lyon.fr/www/crlibm/)
- mpfr (http://www.mpfr.org/)

が挙げられる。以下、それぞれの特徴を述べる。

INTLAB は、matlab 上で書かれた精度保証付き数値計算用のソフトウェア群である。表引きを用いて (matlab 上としては) ある程度の高速性も追求された、両端に double を持つ区間に対する精度保証付き数学関数を持っている。しかし、正しさの完全な証明はなされていない。また、matlab 以外の環境に組み込むのは難しい。

crlibm は、最近点丸めと上向き、下向き丸めを備えた double に対する数学関数群である。上向き、下向き丸めを持つので、これを使って区間演算を実装するのは難しくない。実際、boost C++ libraries (http://www.boost.org/) に含まれる区間演算と crlibm を組み合わせて区間数学関数を得ることが出来る。crlibm は部分的に double double や triple double を用いながら高速性も追求しており、また全ての関数についてそのアルゴリズムの正当性を解説した論文も出されている。大

変な力作であるが、論文及びプログラムは長大で、その正しさを検証するには大変な労力を要する。windows で make するのは難しい。

(crlibm に関する追記) crlibm は、2010 年 3 月 4 日の crlibm-1.0beta4.tar.gz を最後に更新を停止し、現在はプロジェクトのページにもアクセスできなくなっている。証明が長すぎて (例えば C 言語のわずか 3 行の部分の証明が 4 ページにも及ぶ)、このような証明は誰も信じないし誰も読まない、という作者自身の反省の言葉がある。そこで、形式的証明とコードの自動生成を目指して、MetaLibm (`http://www.metalibm.org/`) という新たなプロジェクトが立ち上がり、進行している。

mpfr は、内部に gmp (`http://gmplib.org/`) を用いた、任意精度浮動小数点計算のためのライブラリである。任意精度の最近点丸め、上向き、下向き丸めの可能な数学関数を持つ。広く使われており信頼性は高いと思われるが、内部計算は多倍長整数に頼っているので計算速度は速くない。

なお、区間演算ライブラリとして比較的有名な PROFIL/BIAS (`http://www.ti3.tuhh.de/keil/profil/index_e.html`) は、C 言語の組み込みの数学関数を呼んで上下に丸めるだけという大変雑な実装であり、全く精度保証は出来ていないことを注意しておく。

本稿では、次のようなポリシーで、区間数学関数の実装法を示す。

- 加減乗除と平方根の double を両端に持つ区間演算が可能な環境 (つまり普通の IEEE 754 の演算が出来る環境) を想定し、それを用いて区間数学関数を実装する。
- アルゴリズムは、Taylor 展開など平易な数学のみを用いて、「誰が見てもこれは正しいだろう」と思えるようなものとする。簡単に実装可能であるようなアルゴリズムを目指す。数学の証明に精度保証付き数値計算を使う場合、数学的な正しさを検証しやすいことが重要である。
- 計算結果の区間幅は、1ulp(下限と上限が隣同士の浮動小数点) であることを目指さない。大きめの区間が帰ってくるが、その中に真値が高い信頼性を持って含まれていることを重視する。
- 高速性よりも信頼性を重視する。
- 幅の広い区間の入力に対してもきちんと精度保証された結果を返す。例えば、$\cos([2,4])$ の計算結果は $[-1, \max(\cos(2), \cos(4))]$ を含むなるべく小さい区間となる。
- double より高い精度の数値を用いた区間演算が利用可能ならば、その精度に応じた精度保証付き区間数学関数として使えるように設計する。
- 区間の端点に $-\infty, \infty$ を許す (但し、$[-\infty, -\infty], [\infty, \infty]$ は許さないものとする)。これを利用し、例えば IEEE 754 double 環境下で e^{710} の計算結果は $[\mathrm{DBL\_MAX}(= 2^{1024} - 2^{971}), \infty]$ のように真値を含む区間として計算するようにする。

以下のように記号を定義する。

- $\mathrm{F}_{\max}$: 最大の浮動小数点数。IEEE 754 double なら $2^{1024} - 2^{971}$。
- $\mathrm{F}_{\min}$: 正の最小のの浮動小数点数。IEEE 754 double なら 2^{-1074}。
- $\mathrm{hull}(a, b, c, \ldots)$: $a, b, c, \ldots$ を含む最小の閉区間。但し $a, b, c, \ldots$ はそれぞれ実数または実数を両端にもつ閉区間とする。

2 指数関数

2.1 指数関数

幅の広い入力区間 $I = [x, y]$ に対する exp の計算は、exp は単調増加なので

$$[\exp(x) \text{ の下限}, \exp(y) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\exp(x)$ の計算を示す。まず、$x = \infty$ なら $[\mathrm{F}_{\max}, \infty]$ を、$x = -\infty$ なら $[0, \mathrm{F}_{\min}]$ を返す。そうでなければ、x を

$$x = a + b, \quad a \in \mathbb{N}, \quad -\frac{1}{2} \leq b \leq \frac{1}{2}$$

のように分解し、

$$\exp(x) = \mathrm{e}^a \exp(b)$$

のように行うことにする。$\exp(b)$ を 0 を中心に Taylor 展開すると、

$$\exp(b) = 1 + b + \frac{1}{2!}b^2 + \frac{1}{3!}b^3 + \cdots + \frac{1}{n!}\exp(\theta)b^n$$
$$\text{ただし、} \theta \in \mathrm{hull}(0, b)$$

となる。b の値の範囲を考えると、

$$\exp(\mathrm{hull}(0, b)) \subset \exp(\left[-\frac{1}{2}, \frac{1}{2}\right]) = \left[\mathrm{e}^{-\frac{1}{2}}, \mathrm{e}^{\frac{1}{2}}\right]$$

なので、$\exp(b)$ は

$$\exp(b) \in 1 + b + \frac{1}{2!}b^2 + \frac{1}{3!}b^3 + \cdots + \frac{1}{n!}\left[\mathrm{e}^{-\frac{1}{2}}, \mathrm{e}^{\frac{1}{2}}\right] b^n$$

を区間演算で計算することにより得る。

なお、n はいくつであっても精度保証はされているが、

$$\left|\frac{1}{n!}\left[\mathrm{e}^{-\frac{1}{2}}, \mathrm{e}^{\frac{1}{2}}\right] b^n\right| \leq \frac{1}{n!}\mathrm{e}^{\frac{1}{2}}\left(\frac{1}{2}\right)^n$$

は、$n \geq 15$ で

$$\frac{1}{n!}\mathrm{e}^{\frac{1}{2}}\left(\frac{1}{2}\right)^n \leq 2^{-53}$$

を満たすので、double の場合は n は 15 程度で十分であることが分かる。

2.2 `expm1`

特に $x = 0$ 付近の精度のため、

$$\texttt{expm1}(x) = \exp(x) - 1$$

を持つプログラミング言語がある。これは、$-0.5 \leq x \leq 0.5$ の場合に Taylor 展開

$$\texttt{exmp1}(x) \in 0 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \cdots + \frac{1}{n!}\left[\mathrm{e}^{-\frac{1}{2}}, \mathrm{e}^{\frac{1}{2}}\right] x^n$$

で直接計算し、それ以外の場合は単に exp を呼び出して $\exp(x) - 1$ を計算する。

3 対数関数

3.1 簡単な方法

幅の広い入力区間 $I = [x, y]$ に対する log の計算は、log は単調増加なので

$$[\log_{\mathrm{e}} x \text{ の下限}, \log_{\mathrm{e}} y \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\log_{\mathrm{e}} x$ の計算方法を示す。まず、$x = 0$ なら $[-\infty, -\mathrm{F}_{\max}]$ を、$x = \infty$ なら $[\mathrm{F}_{\max}, \infty]$ を返す。そうでなければ、

$$x = 2^a \cdot b$$

と分解し ($a \in \mathbb{N}$)、

$$\log_{\mathrm{e}} x = a \log_{\mathrm{e}} 2 + \log_{\mathrm{e}} b$$

で計算する。

$\log_{\mathrm{e}} b$ の値は 1 を中心の Taylor 展開で計算する。b の値は、ある値 c に対して

$$c \le b \le 2c$$

となるように正規化できるが、c と $2c$ を中心を 1 に、すなわち $c = \frac{2}{3}$ とする。つまり、

$$\frac{2}{3} \le b \le \frac{4}{3}$$

となるように正規化する。

$\log_{\mathrm{e}} b$ を 1 を中心に Taylor 展開すると、

$$\log_{\mathrm{e}} b = 0 + (b-1) - \frac{1}{2}(b-1)^2 + \frac{1}{3}(b-1)^3 - \frac{1}{4}(b-1)^4 + \cdots + (-1)^{n-1}\frac{1}{n}\theta^{-n}(b-1)^n$$
ただし、$\theta \in \mathrm{hull}(1, b)$

となり、θ を $\mathrm{hull}(1, b)$ で置き換えてこれを区間演算で計算する。これを実装するときは、$b - 1$ は b の値の範囲から無誤差で計算できることに注意する。

なお、n はいくつであっても精度保証はされているが、

$$\left|(-1)^{n-1}\frac{1}{n}\theta^{-n}(b-1)^n\right| \le \left|\frac{1}{n}\left(\frac{1}{[\frac{2}{3}, \frac{4}{3}]}\right)^n (\frac{1}{3})^n\right| = \frac{1}{n}\left(\frac{3}{2}\right)^n \left(\frac{1}{3}\right)^n = \frac{1}{n2^n}$$

は、$n \ge 48$ で

$$\frac{1}{n2^n} \le 2^{-53}$$

を満たすので、double の場合は n は 48 程度必要であることが分かる。

3.2 改良案 1

$n = 48$ では少し計算量が多く、また丸め誤差による区間の膨張も大きいと思われるので、多少改善することを考える。

$\log_e x$ を、

$$x = 2^a \cdot b$$
$$2a \in \mathbb{N} \quad (\text{すなわち、} a \in \ldots, -1.5, -1, -0.5, 0, 0.5, 1, 1.5, \ldots)$$

と分解することにする。a が奇数になるときには無誤差で分解できないので、b は区間で扱う。

このとき、b の変動範囲が

$$2(\sqrt{2}-1) \simeq 0.83 \leq b \leq 4 - 2\sqrt{2} \simeq 1.17$$

となるように正規化する。剰余項の大きさは

$$\left| \frac{1}{n} \left(\frac{1}{[2(\sqrt{2}-1), 4-2\sqrt{2}]} \right)^n (3-2\sqrt{2})^n \right| = \frac{1}{n} \left(\frac{\sqrt{2}-1}{2} \right)^n$$

となり、$n \geq 22$ で

$$\frac{1}{n} \left(\frac{\sqrt{2}-1}{2} \right)^n \leq 2^{-53}$$

を満たすので、double の場合は n は 22 程度で十分であることが分かる。

3.3 改良案 2

よく数値解析の教科書に載っている方法である。

$$b' = \frac{b-1}{b+1}$$

とおき、b について解くと、

$$b = \frac{1+b'}{1-b'}$$

となる。このとき、

$$\log b = \log \frac{1+b'}{1-b'} = \log(1+b') - \log(1-b')$$

となり、これを b' についての Taylor 展開で計算する。b の変動範囲を

$$\frac{\sqrt{2}}{2} \leq b \leq \sqrt{2}$$

と正規化すると、b' の変動範囲は $\pm(3-2\sqrt{2}) \simeq 0.17$ となり、結局 double の場合の必要な項数は $n = 22$ で改良案 1 と変わらない。

3.4 log1p

特に 1 付近の精度のため、

$$\texttt{log1p}(x) = \log_{\mathrm{e}}(x+1)$$

を持つプログラミング言語がある。(改良案 1 を採用したとして) $-(3-2\sqrt{2}) \leq x \leq 3-2\sqrt{2}$ の場合に Taylor 展開

$$\log_{\mathrm{e}}(x+1) = 0 + x - \frac{1}{2}x^2 + \frac{1}{3}x^3 - \frac{1}{4}x^4 + \cdots + (-1)^{n-1}\frac{1}{n}\mathrm{hull}(1, x+1)^{-n}x^n$$

で直接計算し、それ以外の場合は単に log を呼び出して $\log_{\mathrm{e}}(x+1)$ を計算する。

4 三角関数

4.1 sin, cos

幅の広い区間 $I = [x, y]$ に対して、 $\sin I$ 及び $\cos I$ を計算することを考える。x または y が ∞ または $-\infty$ なら、$[-1, 1]$ を返す。そうでなければ、

$$I' = [x', y'] = I - 2n\pi$$

として、

$$-\pi \leq x' \leq \pi$$

となるように正規化する。ただし、この計算は区間値の π を用いて区間演算で行う必要がある。よって、I' の幅は I の幅より一般に大きくなることに注意。高精度な π を用いるなどして慎重に行えればなお良い。

ここで、$y' - x' \geq 2\pi$ なら、$[-1, 1]$ を計算結果とすれば良い。よって、$-\pi \leq y' \leq 3\pi$ だけ考えれば良い。

このとき、求める区間は、sin の場合、

$$\mathrm{hull}(\sin x', \sin y', -1(I' \text{ が } -\tfrac{\pi}{2},\ \tfrac{3}{2}\pi \text{ を含むときのみ}), 1(I' \text{ が } \tfrac{\pi}{2},\ \tfrac{5}{2}\pi \text{ を含むときのみ})) \cap [-1, 1]$$

cos の場合、

$$\mathrm{hull}(\cos x', \cos y', -1(I' \text{ が } -\pi,\ \pi,\ 3\pi \text{ を含むときのみ}), 1(I' \text{ が } 0,\ 2\pi \text{ を含むときのみ})) \cap [-1, 1]$$

となる。この包含のチェックは数学的に厳密に行う必要がある。$\cap[-1, 1]$ は、念のために行うもので、過大評価によって絶対値が 1 を超えてしまうのを防いでいる。

後は x', y' の sin または cos の計算に帰着する。もし $y' \geq \pi$ ならば y' を $y' - 2\pi$(これは微小な幅を持った区間になる) で置き換える。以下、幅の狭い区間である $-\pi \leq x', y' \leq \pi$ の範囲の x', y' に対して、sin または cos を計算することを考える。この計算は、次の表に従う。

	$\sin x$	$\cos x$
$-\pi \le x \le -\frac{3}{4}\pi$	$-\sin(x+\pi)$	$-\cos(x+\pi)$
$-\frac{3}{4}\pi \le x \le -\frac{\pi}{2}$	$-\cos(-\frac{\pi}{2}-x)$	$-\sin(-\frac{\pi}{2}-x)$
$-\frac{\pi}{2} \le x \le -\frac{\pi}{4}$	$-\cos(x+\frac{\pi}{2})$	$\sin(x+\frac{\pi}{2})$
$-\frac{\pi}{4} \le x \le 0$	$-\sin(-x)$	$\cos(-x)$
$0 \le x \le \frac{\pi}{4}$	$\sin(x)$	$\cos(x)$
$\frac{\pi}{4} \le x \le \frac{\pi}{2}$	$\cos(\frac{\pi}{2}-x)$	$\sin(\frac{\pi}{2}-x)$
$\frac{\pi}{2} \le x \le \frac{3}{4}\pi$	$\cos(x-\frac{\pi}{2})$	$-\sin(x-\frac{\pi}{2})$
$\frac{3}{4}\pi \le x \le \pi$	$\sin(\pi-x)$	$-\cos(\pi-x)$

すなわち、$0 \le x \le \frac{\pi}{4}$ の範囲の sin または cos の計算に帰着させている。なお、この表の区分け (x がどの範囲に属するか) はさほど厳密に判定しなくてもよいが、括弧の中の $\frac{\pi}{2} - x$ などの計算は区間演算で厳密に行う必要がある。

最終的な sin または cos の計算は、Taylor 展開を用いて

$$\sin x \in x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \cdots + \frac{1}{n!}[-1,1]x^n$$

$$\cos x \in 1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \cdots + \frac{1}{n!}[-1,1]x^n$$

で計算する。

剰余項の大きさは、

$$\left|\frac{1}{n!}[-1,1]x^n\right| \le \frac{1}{n!}\left(\frac{\pi}{4}\right)^n$$

となり、$n \ge 17$ で

$$\frac{1}{n!}\left(\frac{\pi}{4}\right)^n \le 2^{-53}$$

を満たすので、double の場合は n は 17 程度で十分であることが分かる。

4.2 tan

区間 $I = [x, y]$ に対して、x または y が ∞ または $-\infty$ なら、$[-\infty, \infty]$ を返す。そうでなければ、

$$I' = [x', y'] = I - n\pi$$

として、

$$-\frac{\pi}{2} < x' < \frac{\pi}{2}$$

となるように正規化する。ただし、この計算は区間値の π を用いて区間演算で行う必要がある。よって、I' の幅は I の幅より一般に大きくなることに注意。

ここで、$y' > \frac{\pi}{2}$ ならば、$[-\infty, \infty]$ が解である。

そうでないならば、区間 $(-\frac{\pi}{2}, \frac{\pi}{2})$ において tan は単調増加なので、

$$[\frac{\sin x'}{\cos x'}\text{の下限}, \frac{\sin y'}{\cos y'}\text{の上限}]$$

を結果とすればよい。

5 逆三角関数

5.1 arctan

幅の広い入力区間 $I = [x, y]$ に対する arctan の計算は、arctan は単調増加なので

$$[\arctan(x) \text{ の下限}, \arctan(y) \text{ の上限}]$$

を結果とすればよい。

$\arctan(x)$ の計算は、次の表に従って行う。

$x \geq \sqrt{2}+1$	$\frac{\pi}{2} - \arctan(\frac{1}{x})$
$\sqrt{2}-1 \leq x \leq \sqrt{2}+1$	$\frac{\pi}{4} + \arctan(\frac{x-1}{x+1})$
$-(\sqrt{2}-1) \leq x \leq \sqrt{2}-1$	$\arctan(x)$
$-(\sqrt{2}+1) \leq x \leq -(\sqrt{2}-1)$	$-\frac{\pi}{4} + \arctan(\frac{1+x}{1-x})$
$x \leq -(\sqrt{2}+1)$	$-\frac{\pi}{2} - \arctan(\frac{1}{x})$

これにより、x の変域を $|x| \leq \sqrt{2}-1 \simeq 0.41$ に限定することが出来る。

最終的な arctan の計算は、0 を中心とした Taylor 展開で行う。arctan の n 回微分は、

$$(\arctan(x))^{(n)} = (n-1)! \cos^n(\arctan(x)) \sin(n(\arctan(x) + \frac{\pi}{2}))$$

なので、Taylor 展開は、

$$\arctan(x) \in x - \frac{1}{3}x^3 + \frac{1}{5}x^5 - \frac{1}{7}x^7 + \cdots + \frac{1}{n}[-1,1]x^n$$

となり、これを区間演算で計算する。剰余項の大きさは、

$$\left|\frac{1}{n}[-1,1]x^n\right| \leq \frac{1}{n}(\sqrt{2}-1)^n$$

となり、$n \geq 38$ で

$$\frac{1}{n}(\sqrt{2}-1)^n \leq 2^{-53}$$

を満たすので、double の場合は n は 38 程度で十分であることが分かる。

5.2 arcsin

幅の広い入力区間 $I = [x, y]$ に対する arcsin の計算は、arcsin は単調増加なので

$$[\arcsin(x) \text{ の下限}, \arcsin(y) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\arcsin(x)$ の計算は次のように行う。$x < -1$ または $x > 1$ ならエラー、$x = \pm 1$ のときは直接 $\pm\frac{\pi}{2}$ を返す。$-1 < x < 1$ のときは、

$$\arcsin(x) = \arctan\left(\frac{x}{\sqrt{1-x^2}}\right)$$

で行うが、$|x| \simeq 1$ のとき、具体的には $\frac{\sqrt{6}}{3} \simeq 0.816 \leq |x| < 1$ のときは、

$$\arcsin(x) = \arctan\left(\frac{x}{\sqrt{(1+x)(1-x)}}\right)$$

のようにした方が精度が良い。

5.3 arccos

幅の広い入力区間 $I = [x, y]$ に対する arccos の計算は、arccos は単調減少なので

$$[\arccos(y) \text{ の下限}, \arccos(x) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\arccos(x)$ の計算は次のように行う。$x < -1$ または $x > 1$ ならエラー、$x = -1, 1$ のときは 直接 $\pi, 0$ を返す。$-1 < x < 1$ のときは、

$$\begin{aligned} \arccos(x) &= \frac{\pi}{2} - \arcsin(x) \\ &= \frac{\pi}{2} - \arctan\left(\frac{x}{\sqrt{1-x^2}}\right) \end{aligned}$$

で行うが、$|x| \simeq 1$ のとき、具体的には $\frac{\sqrt{6}}{3} \simeq 0.816 \leq |x| < 1$ のときは、

$$\arccos(x) = \frac{\pi}{2} - \arctan\left(\frac{x}{\sqrt{(1+x)(1-x)}}\right)$$

のようにした方が精度が良い。また、$\frac{\pi}{2} - \arctan(\alpha)$ は $\alpha \simeq \infty$ のとき桁落ちで精度が出ない。そこで、arctan の計算式を直接修正した次の表で計算する。

$\alpha \geq \sqrt{2}+1$	$\arctan(\frac{1}{\alpha})$
$\sqrt{2}-1 \leq \alpha \leq \sqrt{2}+1$	$\frac{\pi}{4} - \arctan(\frac{\alpha-1}{\alpha+1})$
$-(\sqrt{2}-1) \leq \alpha \leq \sqrt{2}-1$	$\frac{\pi}{2} - \arctan(\alpha)$
$-(\sqrt{2}+1) \leq \alpha \leq -(\sqrt{2}-1)$	$\frac{3\pi}{4} - \arctan(\frac{1+\alpha}{1-\alpha})$
$\alpha \leq -(\sqrt{2}+1)$	$\pi + \arctan(\frac{1}{\alpha})$

5.4 atan2

いくつかのプログラミング言語は、2 引数を取る `atan2`(y, x) を持つ。`atan` が $-\frac{\pi}{2} \sim \frac{\pi}{2}$ の範囲の値を返すのに対して、`atan2` は点 (x, y) を 2 次元平面の座標と見なしてその偏角を $-\pi \sim \pi$ の範囲で返す。y と x の符号を用いて返却値の象限を決定する。

まず、幅の狭い区間 y, x を入力とする `atan2n`(y, x) を作成する。これは、次の表に従う。

$y \leq x, \quad y > -x$	$\arctan(y/x)$
$y > x, \quad y > -x$	$\frac{\pi}{2} - \arctan(x/y)$
$y > x, \quad y \leq -x, \quad y \geq 0$	$\pi + \arctan(y/x)$
$y > x, \quad y \leq -x, \quad y < 0$	$-\pi + \arctan(y/x)$
$y \leq x, \quad y \leq -x$	$-\frac{\pi}{2} - \arctan(x/y)$

次に、区間入力に対する `atan2`(I_y, I_x) を作成する。これは、次の表に従う。

$I_x \ni 0, \quad I_y \ni 0$	$[-\pi, \pi]$	
$I_x \ni 0, \quad I_y \not\ni 0, \quad I_y > 0$	$\boxed{\underline{\texttt{atan2n}(I_y, \overline{I_x})}, \overline{\texttt{atan2n}(I_y, \underline{I_x})}}$	
$I_x \ni 0, \quad I_y \not\ni 0, \quad I_y < 0$	$\boxed{\underline{\texttt{atan2n}(\overline{I_y}, I_x)}, \overline{\texttt{atan2n}(\overline{I_y}, \overline{I_x})}}$	
$I_x \not\ni 0, \quad I_y \ni 0, \quad I_x > 0$	$\boxed{\underline{\texttt{atan2n}(I_y, \underline{I_x})}, \overline{\texttt{atan2n}(\overline{I_y}, I_x)}}$	
$I_x \not\ni 0, \quad I_y \ni 0, \quad I_x < 0$	$\boxed{\underline{\texttt{atan2n}(\overline{I_y}, \overline{I_x})}, \overline{\texttt{atan2n}(I_y, \overline{I_x})(+2\pi)}}$	$(*)$
$I_x \not\ni 0, \quad I_y \not\ni 0, \quad I_x > 0, \quad I_y > 0$	$\boxed{\underline{\texttt{atan2n}(I_y, \overline{I_x})}, \overline{\texttt{atan2n}(\overline{I_y}, I_x)}}$	
$I_x \not\ni 0, \quad I_y \not\ni 0, \quad I_x > 0, \quad I_y < 0$	$\boxed{\underline{\texttt{atan2n}(I_y, \underline{I_x})}, \overline{\texttt{atan2n}(\overline{I_y}, \overline{I_x})}}$	
$I_x \not\ni 0, \quad I_y \not\ni 0, \quad I_x < 0, \quad I_y > 0$	$\boxed{\underline{\texttt{atan2n}(\overline{I_y}, \overline{I_x})}, \overline{\texttt{atan2n}(I_y, I_x)}}$	
$I_x \not\ni 0, \quad I_y \not\ni 0, \quad I_x < 0, \quad I_y < 0$	$\boxed{\underline{\texttt{atan2n}(\overline{I_y}, I_x)}, \overline{\texttt{atan2n}(I_y, \overline{I_x})}}$	

$(*)$ の場合は注意が必要である。$\underline{I_y} = 0$ の場合はそのままでいいが、$\underline{I_y} < 0$ の場合は返却値が 2 つの集合に分かれてしまうのを防ぐため、上限に 2π を加算する。この場合、π を超える値が返ってくることになる。

6 双曲線関数

6.1 sinh

幅の広い入力区間 $I = [x, y]$ に対する sinh の計算は、sinh は単調増加なので

$$[\sinh(x) \text{ の下限}, \sinh(y) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\sinh(x)$ の計算は、基本的に

$$\sinh(x) = \frac{\exp(x) - \exp(-x)}{2}$$

のように定義式通り計算する。exp の計算を一回で済ます工夫をする場合は、

$$\begin{aligned} \sinh(x) &= \frac{\exp(x) - \frac{1}{\exp(x)}}{2} \quad (x \geq 0) \\ &= \frac{\frac{1}{\exp(-x)} - \exp(-x)}{2} \quad (x < 0) \end{aligned}$$

のように場合分けするとゼロ除算を避けられる。また、$x = 0$ 付近で $1 - 1$ の形になり、桁落ちによって精度が出ない。`expm1` を使えば解決するが、ここでは、$-0.5 \leq x \leq 0.5$ のときは Taylor 展開で直接計算することにする。Taylor 展開は、

$$\begin{aligned} &\sinh(x) = x + \frac{1}{3!}x^3 + \frac{1}{5!}x^5 + \cdots + \frac{1}{n!}\frac{\exp(\theta) - (-1)^n \exp(-\theta)}{2}x^n \\ &\text{ただし、} \theta \in \mathrm{hull}(0, x) \end{aligned}$$

となる。x の変域を考えると、

$$\begin{aligned} \left\{ \frac{\exp(\theta) - (-1)^n \exp(-\theta)}{2} \middle| \theta \in \mathrm{hull}(0, x) \right\} &\subset \left[-\frac{\exp(\frac{1}{2}) + \exp(-\frac{1}{2})}{2}, \frac{\exp(\frac{1}{2}) + \exp(-\frac{1}{2})}{2} \right] \\ &= \left[-\cosh\left(\frac{1}{2}\right), \cosh\left(\frac{1}{2}\right) \right] \end{aligned}$$

となるので、

$$\sinh(x) = x + \frac{1}{3!}x^3 + \frac{1}{5!}x^5 + \cdots + \frac{1}{n!}\left[-\cosh\left(\frac{1}{2}\right), \cosh\left(\frac{1}{2}\right)\right] x^n$$

を区間演算で計算する。

剰余項の大きさは、

$$\left|\frac{1}{n!}\left[-\cosh\left(\frac{1}{2}\right), \cosh\left(\frac{1}{2}\right)\right] x^n\right| \leq \frac{1}{n!}\cosh\left(\frac{1}{2}\right)\left(\frac{1}{2}\right)^n$$

となり、$n \geq 15$ で

$$\frac{1}{n!}\cosh\left(\frac{1}{2}\right)\left(\frac{1}{2}\right)^n \leq 2^{-53}$$

を満たすので、double の場合は n は 15 程度で十分であることが分かる。

6.2 cosh

幅の広い区間 $I = [x, y]$ に対する $\cosh(I)$ は、

$$\mathrm{hull}(\cosh(x), \cosh(y), 1(I \text{ が } 0 \text{ を含むとき}))$$

で計算する。

点入力 (幅 0 の区間) に対する $\cosh(x)$ は、単純に

$$\cosh(x) = \frac{\exp(x) + \exp(-x)}{2}$$

のように定義式通り計算する。exp の計算を一回で済ます工夫をする場合は、

$$\begin{aligned} \cosh(x) &= \frac{\exp(x) + \frac{1}{\exp(x)}}{2} \quad (x \geq 0) \\ &= \frac{\frac{1}{\exp(-x)} + \exp(-x)}{2} \quad (x < 0) \end{aligned}$$

のように場合分けするとゼロ除算を避けられる。

6.3 tanh

幅の広い入力区間 $I = [x, y]$ に対する tanh の計算は、tanh は単調増加なので

$$[\tanh(x) \text{ の下限}, \tanh(y) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\tanh(x)$ は、単純に $\dfrac{\sinh(x)}{\cosh(x)}$ で計算すると $|x|$ が大きい領域でオーバーフローを起こしてしまう。それを防ぐため、$-0.5 \leq x \leq 0.5$ のときは

$$\frac{\sinh(x)}{\cosh(x)}$$

で、$x > 0.5$ のときは

$$1 - \frac{2}{1 + \exp(2x)}$$

で、$x < -0.5$ のときは

$$\frac{2}{1 + \exp(-2x)} - 1$$

で計算することにする。

7 逆双曲線関数

7.1 $\sinh^{-1}$

幅の広い入力区間 $I = [x, y]$ に対する $\sinh^{-1}$ の計算は、単調増加なので、

$$[\sinh^{-1}(x) \text{ の下限}, \sinh^{-1}(y) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\sinh^{-1}(x)$ の計算は、

$$\sinh^{-1}(x) = \log(x + \sqrt{1 + x^2})$$

で計算できる。しかし、$x \simeq 0$ のとき log の中がおよそ 1 になり、このままだと精度が落ちてしまう。これを防ぐため、

$$\begin{aligned}
\sinh^{-1}(x) &= \log(x + \sqrt{1 + x^2}) \\
&= \log(1 + x + \sqrt{1 + x^2} - 1) \\
&= \log(1 + x + \frac{x^2}{\sqrt{1 + x^2} + 1}) \\
&= \log(1 + x(1 + \frac{x}{\sqrt{1 + x^2} + 1})) \\
&= \texttt{log1p}(x(1 + \frac{x}{\sqrt{1 + x^2} + 1}))
\end{aligned}$$

と変形する。また、$x < 0$ の場合は、

$$\begin{aligned}
\sinh^{-1}(x) &= \log(x + \sqrt{1 + x^2}) \\
&= -\log(\frac{1}{x + \sqrt{1 + x^2}}) \\
&= -\log(-x + \sqrt{1 + x^2})
\end{aligned}$$

を使う。

7.2 $\cosh^{-1}$

幅の広い入力区間 $I = [x, y]$ に対する $\cosh^{-1}$ の計算は、定義域 $[1, \infty]$ で単調増加なので、

$$[\cosh^{-1}(x) \text{ の下限}, \cosh^{-1}(y) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\cosh^{-1}(x)$ の計算は、次のように行う。まず、$x < 1$ はエラーとし、$x = 1$ なら 0 とする。$x > 1$ に対しては、

$$\cosh^{-1}(x) = \log(x + \sqrt{x^2 - 1})$$

で計算できる。しかし、$x \simeq 1$ のとき log の中がおよそ 1 になり、この対策のため、$x' = x - 1$ として (これは x が 1 に近いなら正確に計算出来る)、

$$\cosh^{-1}(x) = \texttt{log1p}(x' + \sqrt{x'(x + 1)})$$

で計算する (あまり意味ないかも)。

7.3 $\tanh^{-1}$

幅の広い入力区間 $I = [x, y]$ に対する $\tanh^{-1}$ の計算は、定義域 $(-1, 1)$ で単調増加なので、

$$[\tanh^{-1}(x) \text{ の下限}, \tanh^{-1}(y) \text{ の上限}]$$

を結果とすればよい。

点入力 (幅 0 の区間) に対する $\tanh^{-1}(x)$ の計算は、$x < -1$ または $x > 1$ ならばエラー、$x = -1$ なら $[-\infty, -\mathrm{F}_{\max}]$、$x = 1$ なら $[\mathrm{F}_{\max}, \infty]$ とする。$-1 < x < 1$ に対しては、

$$\tanh^{-1}(x) = \frac{1}{2}\log(\frac{1+x}{1-x})$$

で計算できる。しかし、$x \simeq 0$ のとき log の中がおよそ 1 になり、この対策のため、

$$\tanh^{-1}(x) = \frac{1}{2}\texttt{log1p}(\frac{2x}{1-x})$$

で計算する。

参考文献

[1] 一松 信: “初等関数の数値計算”, 教育出版 (1974) .

[2] 森口 繁一: “数値計算工学”, 岩波書店 (1989) .