精度保証付き数値計算の基礎
1章：浮動小数点演算と区間演算

尾崎 克久

芝浦工業大学 システム理工学部 数理科学科
共同執筆者：荻田 武史（東京女子大学）

チュートリアル，2018年9月10日，早稲田大学

はじめに

精度保証付き数値計算には，数値計算の理解が必須

現代の数値計算では，IEEE 754規格が定める浮動小数点数とその演算を使用.

ANSI/IEEE Std 754-2008: IEEE Standard for Floating-Point Arithmetic. New-York, 2008.

本日の内容

- 浮動小数点数
- 浮動小数点演算
- 区間演算

内容を概観するため，証明は省きます．

浮動小数点数

はじめに

IEEE 754の2進浮動小数点数の例

- 単精度(binary32)
- 倍精度(binary64)
- 四倍精度(binary128)

binaryの後の数値は，その浮動小数点数の表現に必要なビットの総数を意味する．

浮動小数点数：

- 正規化数
- 非正規化数
- 零
- 正負の無限大に対応する±Inf

がある（その他，非数NaN (Not a Number)がある）.

$\mathbb{F}$をある固定された精度における正規化数の集合，非正規化数の集合，零の和集合とする$(\mathbb{F} \subset \mathbb{R})$.

正規化数または非正規化数である浮動小数点数$a \in \mathbb{F}$は

$$a = s \cdot f \cdot 2^e, \quad s = \pm 1, \quad f = \sum_{i=0}^{p-1} \frac{d_i}{2^i}, \quad d_i \in \{0,\ 1\} \qquad (1)$$

と表現される.

sを符号部，fを仮数部，eを指数部と呼ぶ.

pは精度を表す．単精度：$p = 24$，倍精度：$p = 53$

正規化数または非正規化数である浮動小数点数$a \in \mathbb{F}$は

$$a = s \cdot f \cdot 2^e, \quad s = \pm 1, \quad f = \sum_{i=0}^{p-1} \frac{d_i}{2^i}, \quad d_i \in \{0,\ 1\} \qquad (2)$$

と表現される．

$e \in \mathbb{Z}$は$e_{\min} \le e \le e_{\max}$である．

binary32では$(e_{\min}, e_{\max}) = (-126, 127)$

binary64では$(e_{\min}, e_{\max}) = (-1022, 1023)$

正規化数または非正規化数である浮動小数点数$a \in \mathbb{F}$は

$$a = s \cdot f \cdot 2^e, \quad s = \pm 1, \quad f = \sum_{i=0}^{p-1} \frac{d_i}{2^i}, \quad d_i \in \{0,\ 1\} \qquad (3)$$

と表現される.

- 正規化数では必ず$d_0 = 1$
- 非正規化数では$d_0 = 0$かつ$e = e_{\min}$

ufpの導入

実数$a \in \mathbb{R}$に対するunit in the first place (ufp)を表す関数$\mathtt{ufp}(a)$の定義：

$$\mathtt{ufp}(a) := \begin{cases} 2^{\lfloor \log_2 |a| \rfloor} \ (a \neq 0) \\ 0 \ (a = 0) \end{cases}, \quad a \in \mathbb{R}. \tag{4}$$

$\mathtt{ufp}(a)$は，実数aを2進展開した際の先頭ビットの位．

例：

$$\begin{aligned} \mathtt{ufp}(3.5) &= \mathtt{ufp}(\underline{2^1} + 2^0 + 2^{-1}) = 2 \\ \mathtt{ufp}(0.625) &= \mathtt{ufp}(\underline{2^{-1}} + 2^{-8}) = 2^{-1} \end{aligned}$$

$a \neq 0$ならば

$$\mathtt{ufp}(a) \leq |a| < 2\mathtt{ufp}(a), \quad \mathtt{ufp}(a) = 2^k, \quad \exists k \in \mathbb{Z}$$

が成立する．

定数の説明

- 単位相対丸めを$\mathbf{u}$とする（$\mathbf{u} = 2^{-p}$）.
  - 単精度では$\mathbf{u} = 2^{-24}$，倍精度では$\mathbf{u} = 2^{-53}$
- 正規化数である浮動小数点数の正の最小値を$\mathbf{u}_{\mathrm{N}}$とする.
  - 単精度：$\mathbf{u}_{\mathrm{N}} = 2^{-126}$，倍精度：$\mathbf{u}_{\mathrm{N}} = 2^{-1022}$
- $\mathbb{F}$に属する正の最小数を$\underline{\mathbf{u}}$とする.
  - 単精度：$\underline{\mathbf{u}} = 2^{-149}$，倍精度：$\underline{\mathbf{u}} = 2^{-1074}$

$$a = s \cdot f \cdot 2^e, \quad s = \pm 1, \quad f = \sum_{i=0}^{p-1} \frac{d_i}{2^i}, \quad d_i \in \{0,\ 1\}$$

浮動小数点数の最大数$f_{\max}$は，以下に設定した結果である．

$$s = 1,\ d_i = 1\ (0 \le i < p),\ e = e_{\max},\ (f_{\max} = 2^{e_{\max}+1}(1 - \mathbf{u}))$$

単精度：$f_{\max} = 2^{128}(1 - 2^{-24}) \approx 3.40 \times 10^{38}$,

倍精度：$f_{\max} = 2^{1024}(1 - 2^{-53}) \approx 1.79 \times 10^{308}$

$\mathbb{F}$の要素かどうか

定理 1 $a \in \mathbb{R}$を与える．$a \in \mathbb{F}$であるための必要十分条件は

$$|a| \le f_{\max}, \quad a \in 2\mathbf{u} \cdot \mathtt{ufp}(a)\mathbb{Z}, \quad a \in \underline{\mathbf{u}}\mathbb{Z} \tag{5}$$

がすべて成立することである．

ある2のべき乗数$c = \mathtt{ufp}(c)$を用いて

$$|a| \le f_{\max}, \quad |a| \le c, \quad a \in \mathbf{u}c\mathbb{Z}, \quad a \in \underline{\mathbf{u}}\mathbb{Z}$$

と表現しても良い．

関係

ここで$2\mathbf{u}\cdot\mathtt{ufp}(a)$については，$a$に対するUnit in the Last Place (ulp)としてよく使用されている．これらの関係をFig. 1に示す．

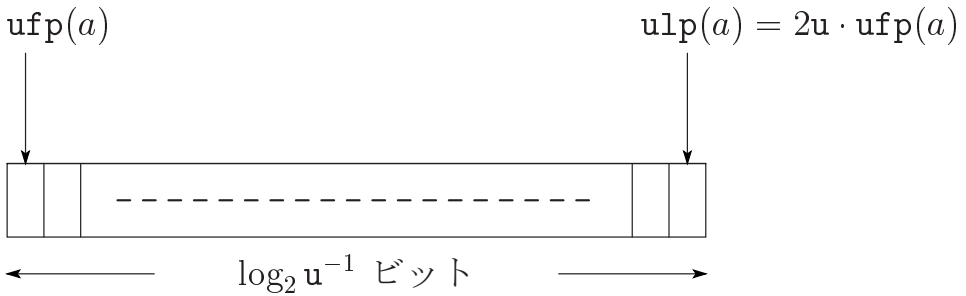

Figure 1: 浮動小数点数に対するufpとulp

関係

$2^{1023}, 2^{1022}$ ... $2^{1}, 2^{0}$ ... $2^{-1073}, 2^{-1074}$

Figure 2: 使用可能範囲（倍精度）

丸め

浮動小数点数で表現できない実数がある．

$a \in \mathbb{R}$を$b \in \mathbb{F} \cup \{\pm\mathtt{Inf}\}$に対応づける関数$b = \mathtt{RN}(a)$を以下のように定義する．

$$\begin{cases} b \in \mathbb{F} \text{ s.t. } |b-a| = \min\limits_{f\in\mathbb{F}} |f-a|, & |a| < f_{\max} + \mathbf{u}\cdot\mathtt{ufp}(f_{\max}) \\ \mathtt{Inf}, & a \geq f_{\max} + \mathbf{u}\cdot\mathtt{ufp}(f_{\max}) \\ -\mathtt{Inf}, & a \leq -(f_{\max} + \mathbf{u}\cdot\mathtt{ufp}(f_{\max})) \end{cases}$$

偶数丸め

ただし，aが隣接する2つの浮動小数点の中点であり，最近点が2つある場合には，偶数丸め方式を採用する．

これは仮数部の最終ビットd_{p-1}が0になるように結果を丸める方式である．

偶数丸めの例：$\mathtt{RN}(1+\mathbf{u}) = 1, \mathtt{RN}(1+3\mathbf{u}) = 1+4\mathbf{u}$

関係

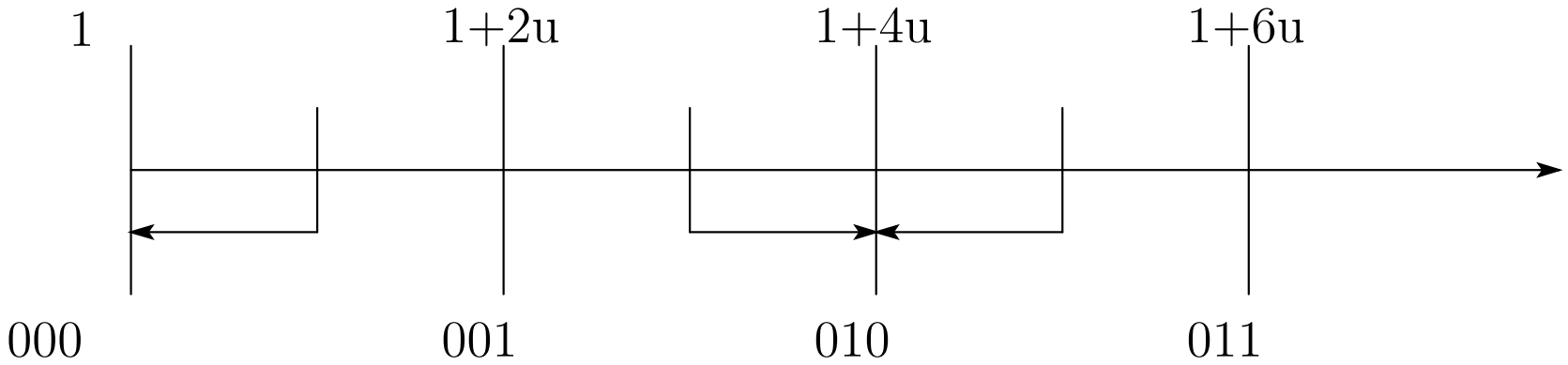

Figure 3: 偶数丸めのイメージ

Infの発生

Infは，丸める前の値が$f_{\max} + \mathbf{u} \cdot \mathtt{ufp}(f_{\max})$以上のときに現れ，負の場合も同様である．$\pm$Infが返ってくることをオーバーフローが起きたという．

丸める前の実数が，浮動小数点数の最大数を超えても，Infになるとは限らない．

丸め誤差

定理 2 $a \in \mathbb{R}$に対して，$\mathbf{u}_{\mathrm{N}} \leq |a| \leq f_{\max}$とする．実数を浮動小数点数に丸めた場合

$$|a - \mathtt{RN}(a)| \leq \mathbf{u} \cdot \mathtt{ufp}(a) \leq \mathbf{u}|a| \tag{6}$$

が成立する．

$a = 1 + \mathbf{u}$のとき$\mathtt{RN}(a) = 1$となり，等号が成立する．

丸め誤差

定理 3 $a \in \mathbb{R}$に対して，$\mathbf{u}_{\mathrm{N}} \leq |a| \leq f_{\max}$とする．実数を浮動小数点数に丸めた場合

$$\mathtt{RN}(a) = a(1+\delta), \quad |\delta| \leq \frac{\mathbf{u}}{1+\mathbf{u}} \leq \mathbf{u} \tag{7}$$

が成立する．

$a = 1 + \mathbf{u}$のとき$\mathtt{RN}(a) = 1$となり，等号が成立する．

丸め誤差

定理 4 $a \in \mathbb{R}$に対して，$\mathbf{u}_{\mathrm{N}} \leq |a| \leq f_{\max}$とする．このとき

$$a = \mathrm{RN}(a)(1+\delta), \quad |\delta| \leq \mathbf{u} \tag{8}$$

が成立する．

$a = 1 + \mathbf{u}$のとき$\mathrm{RN}(a) = 1$となり，等号が成立する．

以上は最適な評価であるが，アンダーフローが考慮されていない．

アンダーフロー

$|\mathrm{RN}(a)| < \mathbf{u}_{\mathrm{N}}$ または $|a| < \mathbf{u}_{\mathrm{N}}$ のときのこと．

アンダーフローが発生する場合は，最終ビット $\underline{\mathbf{u}}$ 以降で誤差を考慮するため，以下の定理が直ちに導出される．

定理 5 $|a| \leq \mathbf{u}_{\mathrm{N}}$ となる $a \in \mathbb{R}$ に対して

$$a = \mathrm{RN}(a) + \eta, \quad |\eta| \leq \underline{\mathbf{u}}/2 \tag{9}$$

が成立する．

丸め

よって，定理2，定理3に対して定理5をそれぞれ合わせて以下の定理を得る．

定理 6 $a \in \mathbb{R}$に対して，$|a| \leq f_{\max}$とする．実数を浮動小数点数に丸めた場合

$$|a - \mathtt{RN}(a)| \leq \mathbf{u} \cdot \mathtt{ufp}(a) + \eta, \quad \mathtt{RN}(a) = a(1+\delta) + \eta \quad (10)$$

が成立する．ここで$|\eta| \leq \mathbf{u}/2$，$|\delta| \leq \frac{\mathbf{u}}{1+\mathbf{u}}$，$\delta \cdot \eta = 0$である．

浮動小数点演算

浮動小数点演算

浮動小数点数どうしの演算の結果を浮動小数点数で表す

$\mathrm{fl}(\cdot)$ は括弧内のすべての二項演算を浮動小数点演算で評価する表記

例えば，$a, b, c \in \mathbb{F}$ に対して $\mathrm{fl}(\mathrm{fl}(a+b)+c)$ は，表記の簡略化のため，$\mathrm{fl}((a+b)+c)$ と記載しても同じ意味

一般に $a, b \in \mathbb{F} \Rightarrow a+b \in \mathbb{F}$ が成立しないため，$a+b$ と $\mathrm{fl}(a+b)$ の差（丸め誤差）を考える．

浮動小数点演算

IEEE 754規格による四則演算，平方根の計算は，無限精度で計算した後に最も近い浮動小数点数に丸めた結果を返すことが定められている．

定理 7 $x, y \in \mathbb{F}$がある．$|\mathrm{fl}(x \pm y)| < 2\mathbf{u}_{\mathrm{N}}$のとき，$\mathrm{fl}(x \pm y) = x \pm y$が成立する．また，$x$と$y$がともに非正規化数のとき，$\mathrm{fl}(x \pm y) = x \pm y$が成立する．

浮動小数点演算

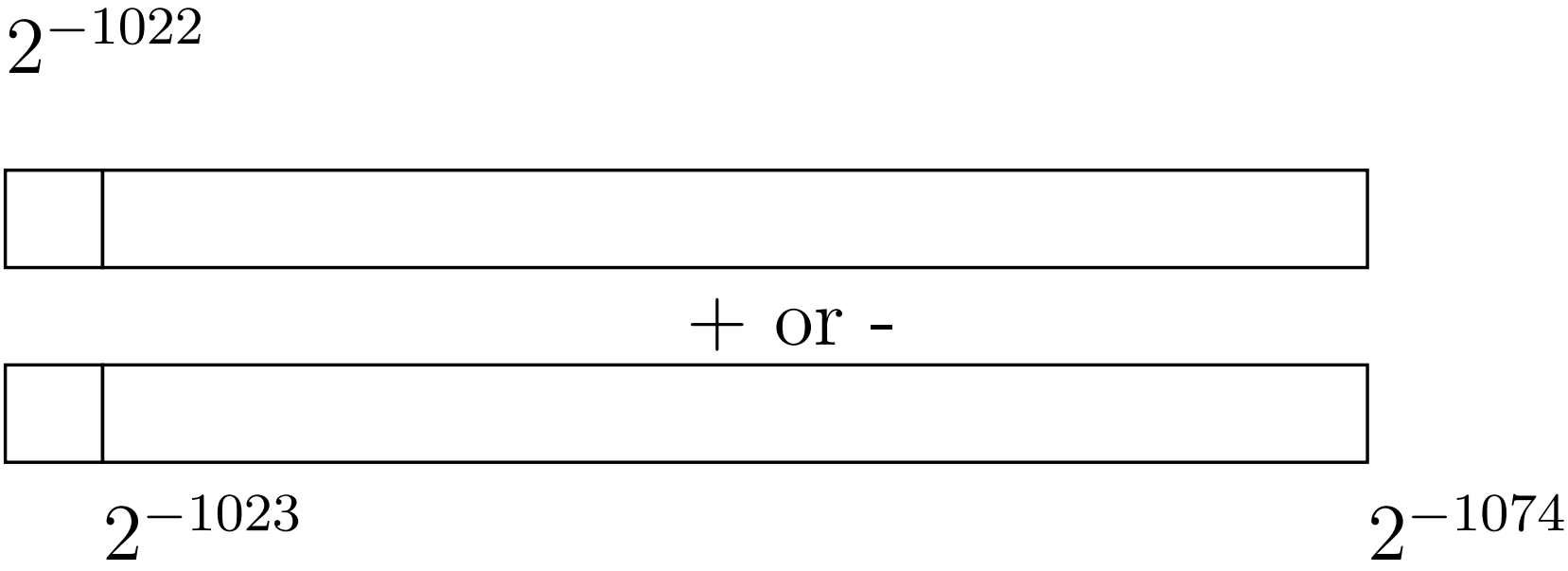

Figure 4: 非正規化数どうしの計算

浮動小数点演算（和・差）

定理 8 $a, b \in \mathbb{F}$に対して，

$$\mathrm{fl}(a \pm b) = (a \pm b)(1 + \delta_1), \quad |\delta_1| \le \frac{\mathbf{u}}{1 + \mathbf{u}}$$

$$(a \pm b) = \mathrm{fl}(a \pm b)(1 + \delta_2), \quad |\delta_2| \le \mathbf{u} \tag{11}$$

$$\mathrm{fl}(a \pm b) = a \pm b + \delta_3, \quad |\delta_3| \le \mathbf{u} \cdot \mathtt{ufp}(a \pm b) \tag{12}$$

が成り立つ．ただし，オーバーフローが発生しないことを仮定する．

浮動小数点演算（積）

定理 9 $a, b \in \mathbb{F}$に対して，

$$\mathrm{fl}(a \cdot b) = a \cdot b + \delta + \eta,$$

$$|\delta| \leq \mathbf{u} \cdot \mathrm{ufp}(a \cdot b) \quad , \qquad |\eta| \leq \underline{\mathbf{u}}/2, \quad \delta \cdot \eta = 0$$

が成り立つ．

この定理では，$\mathrm{fl}(a \cdot b)$においてアンダーフローが発生する場合には$\delta = 0$と，アンダーフローが発生しない場合には$\eta = 0$としてよい．

浮動小数点演算

定理8の結果は以下のように拡張できる．

定理 10 $a, b \in \mathbb{F}$に対して

$$|\mathrm{fl}(a+b) - (a+b)| \leq \min(|a|, |b|, \mathbf{u} \cdot \mathtt{ufp}(a+b))$$

が成立する．

aやbの絶対値が他方に比べて極端に小さいときに有用

浮動小数点演算

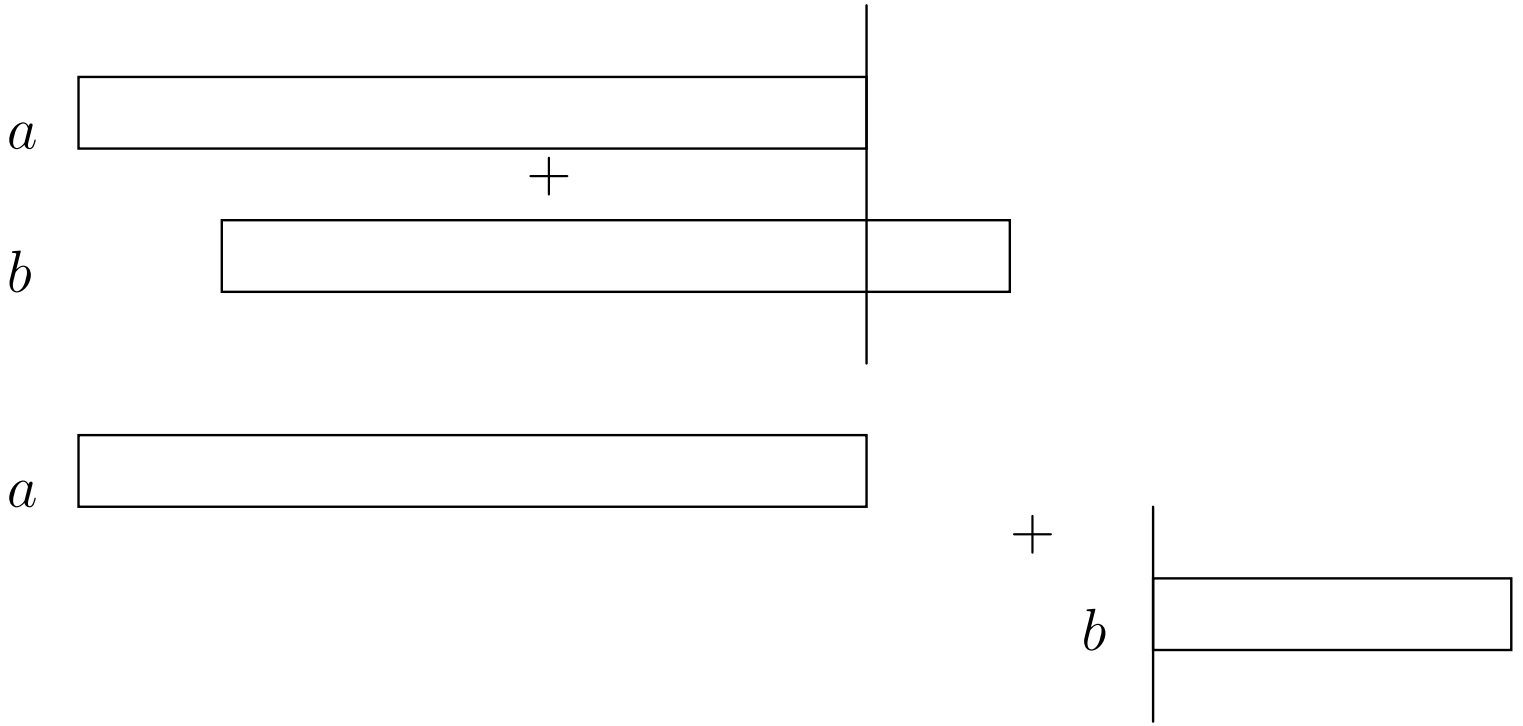

Figure 5: 特別なケース

区間演算

はじめに（区間演算）

区間演算は精度保証付き数値計算において重要な役割を担うため，ここで解説を行う．

区間演算は1950年代に須永照夫氏により提唱され，その後R. E. Moore氏らの貢献により発展し，現在も研究が進んでいる．

T. Sunaga. ”Theory of an interval algebra and its application to numerical analysis.” Japan Journal of Industrial and Applied Mathematics 26.2-3 (2009): 125-143.

表記

- 行列・ベクトルに対する絶対値記号は，すべての成分に絶対値をつける
- 行列やベクトルに対する不等式は成分毎にすべて成立することを意味する．

例えば$x = (-1, 2, -3)^T$に対しては$|x| = (1, 2, 3)^T$である．

下端・上端型

まず実区間の集合を$\mathbb{IR}$とし，区間型の変数を$\boldsymbol{a}$のように太字を用いて表す.

下端・上端型の実区間は,

$$[\underline{a}, \overline{a}] = \{x \in \mathbb{R} \mid \underline{a} \leq x \leq \overline{a}\}, \quad \underline{a}, \overline{a} \in \mathbb{R},\ \underline{a} \leq \overline{a}$$

と表す.

中心・半径型

また，中心・半径型の実区間は，$c, r \in \mathbb{R}, r \geq 0$に対して

$$\langle c, r \rangle = \{x \in \mathbb{R} \mid c - r \leq x \leq c + r\},$$

と表現する．

- 区間$\boldsymbol{a}$の中心を$\mathtt{mid}(\boldsymbol{a})$と表記
- 区間$\boldsymbol{a}$の半径を$\mathtt{rad}(\boldsymbol{a})$と表記

和・差・積

$\boldsymbol{a}, \boldsymbol{b} \in \mathbb{IR}$, $\boldsymbol{a} \circ \boldsymbol{b}$, $\circ \in \{+, -, *\}$は

$$\boldsymbol{a} \circ \boldsymbol{b} := \{a \circ b \in \mathbb{R} \mid \forall a \in \boldsymbol{a},\ \forall b \in \boldsymbol{b}\}$$

を表す．ここで，具体的には

$$\boldsymbol{a} + \boldsymbol{b} = [\underline{a} + \underline{b}, \overline{a} + \overline{b}], \quad \boldsymbol{a} - \boldsymbol{b} = [\underline{a} - \overline{b}, \overline{a} - \underline{b}], \tag{13}$$

$$\boldsymbol{ab} = [\min(\underline{ab}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b}), \max(\underline{ab}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b})] \tag{14}$$

となる．

除算

除算に関しては，$0 \notin \boldsymbol{b}$ とすれば

$$\boldsymbol{a}/\boldsymbol{b} = \boldsymbol{a} \times [1/\overline{b}, 1/\underline{b}]$$

である．

定理 11 *2*つの中心・半径型の区間$\langle c_1, r_1\rangle, \langle c_2, r_2\rangle$に対する演算は

$$\langle c_1, r_1\rangle \pm \langle c_2, r_2\rangle = \langle c_1 \pm c_2, r_1 + r_2\rangle \tag{15}$$

$$\langle c_1, r_1\rangle \cdot \langle c_2, r_2\rangle = \langle c_1 c_2 + \delta_1, \delta_2\rangle \tag{16}$$

となる．ここで

$$\begin{aligned} \delta_1 &= \mathrm{sgn}(c_1 c_2)\min(r_1|c_2|, |c_1|r_2, r_1 r_2) \\ \delta_2 &= \max(r_1(|c_2| + r_2), (|c_1| + r_1)r_2, r_1|c_2| + |c_1|r_2) \end{aligned}$$

である．sgnは括弧内の符号をかえす関数とする．

高速な計算

定理11における区間の積を計算する際には，最大・最小の比較が必要である．これを避けるために，

$$\langle c_1, r_1\rangle \cdot \langle c_2, r_2\rangle \subseteq \langle c_1c_2, |c_2|r_1 + |c_1|r_2 + r_1r_2\rangle \qquad (17)$$

という関係もよく利用される．

これは，区間の積の包含を意味するが，(14)や(16)と比べて区間幅の拡大は最大で1.5倍までということが知られている．

区間拡張

関数$f(x)$が区間$\boldsymbol{x}$に対して取り得る範囲を

$$f(\boldsymbol{x}) := \{f(x) \mid x \in \boldsymbol{x}\}$$

と表す．これは関数の区間$\boldsymbol{x}$における最小値から最大値までの区間を意味する．

また，関数$f(x)$に区間$\boldsymbol{x}$を代入し，以後関数の評価を数式の形のまま区間演算で評価することを区間拡張と呼び，$f_{[\ ]}(\boldsymbol{x})$と表す．

区間拡張

$f(x) = x^2 + 3x + 2$とする．

xの代わりに$\boldsymbol{x} = [-1, 1]$を代入し，区間演算を行うと

$$\begin{aligned} f_{[\,]}(\boldsymbol{x}) &= \boldsymbol{x}^2 + 3\boldsymbol{x} + 2 \\ &= [-1, 1] * [-1, 1] + 3 * [-1, 1] + 2 \\ &= [-1, 1] + [-3, 3] + 2 = [-2, 6] \end{aligned} \tag{18}$$

となる．

区間拡張

関数内の数式が等しい場合でも，その表記が異なる場合には $f_{[\ ]}(\cdot)$ により出力される区間が異なることに注意する．

前述の関数を $g(x) = (x+1)(x+2)$ とすれば

$$g_{[\ ]}([-1,1]) = [0,2] \times [1,3] = [0,6] \tag{19}$$

からわかる．

区間拡張

一般に，

$$f(\boldsymbol{x}) \subseteq f_{[\,]}(\boldsymbol{x})$$

となることが知られている．

ある区間における関数fの最大・最小値を求めることは，関数fが複雑な場合に容易ではないが，$f_{[\,]}$により最大・最小を含む区間を得ることは比較的容易である．

例

区間$[-2,2]$について，$f(x)=x^2+x+7$を考える．

$$\begin{aligned} f_{[\ ]}([-2,2]) &= [-2,2]*[-2,2]+[-2,2]+7 \\ &= [-4,4]+[-2,2]+7=[1,13] \end{aligned}$$

であるため，$f(x)=0$の解はこの区間にない．

もっと区間を細かくとって計算できる．

分配則について

区間$\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \in \mathbb{IR}$に関しては，一般に分配法則$\boldsymbol{a}(\boldsymbol{b}+\boldsymbol{c}) = \boldsymbol{ab}+\boldsymbol{ac}$が成立しない．

例として，$\boldsymbol{a} = [-1, 1], \boldsymbol{b} = [1, 2], \boldsymbol{c} = [-2, 1]$を考えれば

$$\boldsymbol{a}(\boldsymbol{b}+\boldsymbol{c}) = \boldsymbol{a} \times [-1, 3] = [-3, 3]$$
$$\boldsymbol{ab}+\boldsymbol{ac} = [-2, 2] + [-2, 2] = [-4, 4]$$

となることから，分配法則が成立しないことがわかる．

劣分配則

ただし，

$$\boldsymbol{a}(\boldsymbol{b}+\boldsymbol{c}) \subseteq \boldsymbol{ab}+\boldsymbol{ac} \tag{20}$$

が成立する．

(18) と (19) の違い，また (20) の成立から，因数分解された式に区間を代入するほうが，より狭い区間が得られると誘導してしまうかもしれないが，それは間違いである．

注意

次の式

$$x(x-1)(x+1), \quad x^3 - x$$

に区間$[-1, 1]$を代入して区間演算を行うと，結果はそれぞれ$[-4, 4], [-2, 2]$となる．

因数分解された式を利用して区間演算を行った結果のほうが過大評価となる例もある．

丸めモード

$\mathtt{fl}_{\nabla}(\cdot)$ は括弧内の演算を下向きの丸めのモードにより計算することを意味する．

演算結果に誤差がある場合には，真の値以下の最も最大の浮動小数点数に丸めることを意味する．

$\mathtt{fl}_{\Delta}(\cdot)$ は括弧内の演算を上向きの丸めのモードで計算することを意味する．

演算結果に誤差がある場合には，真の値以上の最小の浮動小数点数に丸めることを意味する．

丸めモード

$a, b \in \mathbb{F}$に対して，$\circ \in \{+, -, \times, /\}$として

$$\mathtt{fl}_{\triangledown}(a \circ b) = \max\{x \in \mathbb{F} \mid x \leq a \circ b\}$$

$$\mathtt{fl}_{\triangle}(a \circ b) = \min\{x \in \mathbb{F} \mid x \geq a \circ b\}$$

であり，

$$\mathtt{fl}_{\triangledown}(a \circ b) \leq a \circ b \leq \mathtt{fl}_{\triangle}(a \circ b) \qquad (21)$$

が成立する．これらの方向丸めを実装することはIEEE 754規格により要請されている．

機械区間演算へ

区間演算は，正確な実数演算が使用可能であれば実装できる．

ただし，$a, b \in \mathbb{F}$に対して，$\mathrm{fl}a + b = a + b$がいつでも成り立つとは限らないため，浮動小数点演算では厳密な実装が難しい．

すなわち，$[\underline{a}, \overline{a}] + [\underline{b}, \overline{b}] := [\mathrm{fl}(\underline{a} + \underline{b}), \mathrm{fl}(\overline{a} + \overline{b})]$ と実装すれば，これは誤りである．

機械区間

機械区間とは，浮動小数点数を用いて表現される区間.

区間の下端と上端が浮動小数点数であるような下端・上端型の機械区間の集合を$\mathbb{IF}_{\text{infsup}}$とすると，$[\underline{a}, \overline{a}] \in \mathbb{IF}_{\text{infsup}}$は

$$[\underline{a}, \overline{a}] := \{x \in \mathbb{R} \mid \underline{a} \leq x \leq \overline{a},\ \underline{a}, \overline{a} \in \mathbb{F}\}$$

を表す.

機械区間

また，区間の中心と半径が浮動小数点数であるような中心･半径型の機械区間の集合を$\mathbb{IF}_{\text{midrad}}$とすると，$\langle a_c, a_r \rangle \in \mathbb{IF}_{\text{midrad}}$は

$$\langle a_c, a_r \rangle := \{x \in \mathbb{R} \mid a_c - a_r \leq x \leq a_c + a_r,\ a_c, a_r \in \mathbb{F},\ a_r \geq 0\}$$

を意味する．

機械区間

$\mathbb{IF}_{\text{infsup}}$ と $\mathbb{IF}_{\text{midrad}}$ は集合としては異なることに留意.

例：

$$\mathbb{IF}_{\text{infsup}} \not\ni \langle 1, \mathbf{u}^2 \rangle \in \mathbb{IF}_{\text{midrad}}$$

$$\mathbb{IF}_{\text{infsup}} \ni [1, 1 + 2\mathbf{u}] \notin \mathbb{IF}_{\text{midrad}}$$

下端・上端型の機械区間に対する機械区間演算は

$$[a] + [b] \subseteq [\mathtt{fl}_{\nabla}(\underline{a} + \underline{b}), \mathtt{fl}_{\triangle}(\overline{a} + \overline{b})], \quad (2$$

$$[a] - [b] \subseteq [\mathtt{fl}_{\nabla}(\underline{a} - \overline{b}), \mathtt{fl}_{\triangle}(\overline{a} - \underline{b})], \quad (2$$

$$[a][b] \subseteq [\mathtt{fl}_{\nabla}(\min(\underline{ab}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b})), \mathtt{fl}_{\triangle}(\max(\underline{ab}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b}$$

となる．

このようにして浮動小数点演算を用いながらも区間の包含が達成される結果を求めることが可能である．

まとめ

本チュートリアルでは

- 浮動小数点数
- 浮動小数点演算
- 区間演算

の基礎を解説した.

詳細は本を御覧ください

区間型の変換

ここで，下端・上端型の機械区間$[\underline{a}, \overline{a}] \in \mathbb{IF}_{\mathrm{infsup}}$を中心・半径型の機械区間$\langle c, r \rangle \in \mathbb{IF}_{\mathrm{midrad}}$で表現したいとする．

$[\underline{a}, \overline{a}]$の中心は$\frac{\underline{a}+\overline{a}}{2}$，半径は$\frac{\overline{a}-\underline{a}}{2}$であるが，$\frac{\underline{a}+\overline{a}}{2} \notin \mathbb{F}$である可能性もある．

区間型の変換

区間の過大評価を少々許した以下のような変換が知られている．

定理 12 下端・上端型の機械区間 $[\underline{a}, \overline{a}] \in \mathbb{IF}_{\mathrm{infsup}}$ について

$$c = \mathtt{fl}_{\triangle}\left(\frac{\overline{a} + \underline{a}}{2}\right), \quad r = \mathtt{fl}_{\triangle}\left(c - \underline{a}\right)$$

とすれば，$[\underline{a}, \overline{a}] \subseteq \langle c, r \rangle \in \mathbb{IF}_{\mathrm{midrad}}$ となる．

行列とベクトル

ここで表した区間は，行列やベクトルにも拡張できる．

区間行列を $\boldsymbol{C} = [\underline{C}, \overline{C}]$, $\boldsymbol{D} = \langle D_c, D_r \rangle \in \mathbb{IR}^{m \times n}$ は

$$\boldsymbol{C} = [\underline{C}, \overline{C}] := \{C \in \mathbb{R}^{m \times n} \mid \underline{C} \leq C \leq \overline{C}\}$$

$$\boldsymbol{D} = \langle D_c, D_r \rangle := \{D \in \mathbb{R}^{m \times n} \mid D_c - D_r \leq D \leq D_c + D_r\}$$

である．

複素区間

$\underline{C}, \overline{C}, D_c, D_r \in \mathbb{R}^{m\times n}$，$\underline{C} \le \overline{C}$，$D_r \ge O$ (零行列) とする．

$\mathbb{IC}$は複素区間全体の集合を表す．

ここでは複素区間$\boldsymbol{a} \in \mathbb{IC}$は，中心・半径型の区間を用いる．

すなわち，$a_c \in \mathbb{C}, a_r \ge 0$に対して

$$\boldsymbol{a} := \langle a_c, a_r \rangle = \{a \in \mathbb{C} \mid |a - a_c| \le a_r\}$$

表記

このとき，$\boldsymbol{a}$内で絶対値最大の値を

$$\mathrm{mag}(\boldsymbol{a}) := \max_{a \in \boldsymbol{a}} |a| = |a_c| + a_r$$

と表記する．

- n次元複素区間ベクトル全体の集合は$\mathbb{IC}^n$
- $m \times n$複素区間行列全体の集合は$\mathbb{IC}^{m \times n}$