

浮動小数点計算を用いた代数的数の符号判定

東京大学大学院工学系研究科 日吉 久礎 (Hisamoto Hiyoshi)¹⁾

NTT コミュニケーション科学研究所 関川 浩 (Hiroshi Sekigawa)²⁾

1. はじめに

計算代数や計算幾何の多くの問題は、実係数多項式 $f(x_1, \dots, x_n)$ および実数 $a_1, \dots, a_n$ に対して $f(a_1, \dots, a_n)$ の符号を求める問題に帰着される. 例えば、2 次元凸包を構成する問題は、3 点 $p_i = (x_i, y_i)$, $i = 1, 2, 3$, の向きを表す行列式

$$\Delta(p_1, p_2, p_3) = \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix}$$

の符号を求める問題に帰着される.

近似演算のみでは正確に $f(a_1, \dots, a_n)$ の符号を求めることができないが、別の情報と組合せることによって可能になる. 次のようなアプローチがよく知られている [4]:

1. $f(a_1, \dots, a_n)$ の separation bound q を求める. ここで、 q が α の separation bound であるとは、 $|\alpha| < q$ ならば $\alpha = 0$ であるときをいう.
2. 近似演算を用いて、 $|f(a_1, \dots, a_n)| < q$ であるかどうかを決定する.

関川は、separation bound を得るために Mahler の measure を、また近似計算として区間演算を用いた代数的数の符号判定法を提案した [3]. 本研究では、区間演算の代わりに浮動小数点演算を用いた符号判定法を提案し、区間を用いた符号判定法と実験的に比較する. 以下では、代数的数の集合を $\mathbf{K}$, 実代数的数の集合を $\mathbf{A}$ と書くことにする. 考える問題を以下のように定式化する:

問題. 多項式 $f(x_1, \dots, x_n) \in \mathbf{A}[x_1, \dots, x_n]$ および 実代数的数 $\alpha_1, \dots, \alpha_n$ が与えられたとする. このとき、 $\gamma = f(\alpha_1, \dots, \alpha_n)$ の符号を決定せよ.

2. Mahler の measure

代数的数の Mahler の measure (以下では単に measure と呼ぶことにする. measure についての詳細は、例えば [1] を参照) は、次のように定義される:

¹⁾ hiyoshi@simplex.t.u-tokyo.ac.jp

²⁾ sekigawa@cslab.kecl.ntt.co.jp

定義 1 代数的数 α に対して, $f \in \mathbf{Z}[x]$ を α の原始最小多項式とする. f は,

$$f = a \prod_{i=1}^d (x - \beta_i), \quad a \in \mathbf{Z}, \quad \beta_i \in \mathbf{K}$$

と書くことができる. このとき, α の measure $M(\alpha)$ を

$$M(\alpha) = |a| \prod_{i=1}^d \max\{1, |\beta_i|\}$$

と定義する.

代数的数の measure と絶対値の間に, 次のような定理が成り立つ:

定理 2 任意の 0 でない代数的数 α に対して,

$$1/M(\alpha) \leq |\alpha| \leq M(\alpha)$$

が成り立つ. ■

この定理から, $1/M(\alpha)$ が α の separation bound であることがわかる. さらに, 任意の $M(\alpha)$ の上界 m に対して, $1/m$ もまた separation bound になる. 関川は measure の上界を求める手続きを提案した [3]. measure を用いて separation bound を求めるのは, 他のノルムを用いる場合よりも易しい.

3. 区間を用いた符号判定法

以下では, 浮動小数点体系の仮数部の桁数を精度と呼ぶことにする.

関川の提案した符号判定法は, 以下のようなものである [3]:

符号判定法 1 (区間法)

1. γ の measure の上界 m を求める.
2. $\mu \leftarrow$ (初期精度).
3. 精度 μ の機械区間演算によって, γ を含む区間 I を求める.
4. **if** $I > 0$ **return** +.
5. **if** $I < 0$ **return** -.
6. **if** $m \cdot |I| < 1$ **return** 0.
7. 精度 μ を上げる.
8. **goto** 3.

上のアルゴリズムにおいて, 初期精度および精度の上げ方は任意である.

4. 浮動小数点演算を用いた符号判定法

4.1. 誤差解析

この節では、実係数多項式 $f(x_1, \dots, x_n)$ および実数 $a_1, \dots, a_n$ が与えられたとき、 $c = f(a_1, \dots, a_n)$ を精度 μ の浮動小数点体系で近似計算したときの誤差を求める方法を述べる。この誤差推定法では、誤差が精度 μ の関数として求められる。したがって、一度誤差を評価しておけば、誤差評価式に精度の値を代入することで誤差を容易に求めることができる。一方、区間演算では精度を変えるたびに誤差評価を行わなければならない。また、この誤差推定法は、符号判定の可能な精度の推定に応用することができる (4.3. 節参照)。

用いる浮動小数点体系の基数を 2 とし、精度 μ の浮動小数点体系を M_μ と表すことにする。用いる浮動小数点体系の丸めの向きは以下の議論に無関係だが、計算の途中でオーバーフローやアンダーフローが発生しないと仮定する。精度 μ の浮動小数点体系における加減乗算をそれぞれ $+_\mu, -_\mu, \cdot_\mu$ と書くことにする。

集合 E を $E = \mathbf{Z} \cup \{-\infty\}$ と定義する。

定義 3 実数 $a \in \mathbf{R}$ に対して、 $\varepsilon: \mathbf{R} \rightarrow E$ を

$$\varepsilon(a) = \begin{cases} -\infty, & \text{if } a = 0, \\ \lfloor \log_2 |a| \rfloor, & \text{otherwise} \end{cases}$$

と定義する。

定義 4 E 上の演算 $\tilde{+}, \tilde{\cdot}$ を

$$\xi \tilde{+} \eta = \begin{cases} \eta, & \text{if } \xi = -\infty, \\ \xi, & \text{if } \xi \neq -\infty \wedge \eta = -\infty, \\ \max\{\xi, \eta\} + 2, & \text{otherwise,} \end{cases}$$

$$\xi \tilde{\cdot} \eta = \begin{cases} -\infty, & \text{if } \xi = -\infty \vee \eta = -\infty, \\ \xi + \eta + 2, & \text{otherwise} \end{cases}$$

と定義する。

$(E, \tilde{+}, \tilde{\cdot})$ を用いると、精度 μ で c の近似値 c_μ を計算したときの誤差の評価法が以下のように与えられる。 $\{b_1, \dots, b_m\} = \{a_1, \dots, a_n\} \cup \{f \text{ の係数}\}$ とする。 c の値は、次のような計算過程によって計算される:

$$\begin{aligned} w_1 &:= s_1 \odot_1 t_1, \\ w_2 &:= s_2 \odot_2 t_2, \\ &\vdots \\ w_l &:= s_l \odot_l t_l. \end{aligned}$$

ここで、 $\odot_i$ は $+, -, \cdot$ のいずれか、 s_i, t_i は $b_1, \dots, b_m, w_1, \dots, w_{i-1}$ のいずれかである。上の計算過程において、 $b_1, \dots, b_m$ を $\varepsilon(b_1), \dots, \varepsilon(b_m)$ に、 $+, -, \cdot$ を $\tilde{+}, \tilde{-}, \tilde{\cdot}$ にそれぞれ置き換え

る. このようにして得られた w_l によって, c_μ の誤差は

$$\begin{cases} |c_\mu - c| = 0, & \text{if } w_l = -\infty, \\ |c_\mu - c| < 2^{w_l - \mu}, & \text{otherwise} \end{cases}$$

のように与えられる. 以下の補題でこれを証明する.

近似計算された値の絶対値について, 次の 2 つの補題が成り立つ:

補題 5 実数 a に対して, 精度 μ の浮動小数点体系において丸めた結果を a_μ とする. このとき,

$$|a_\mu| \leq 2^{\varepsilon(a)}$$

が成り立つ.

証明 浮動小数点体系の構造から明らか. ■

補題 6 浮動小数点数 $a_\mu, b_\mu \in \mathbf{M}_\mu$ に対して,

$$|a_\mu| \leq 2^\xi, \quad |b_\mu| \leq 2^\eta$$

が成り立っていると仮定する. このとき,

$$\begin{aligned} |a_\mu \pm_\mu b_\mu| &< 2^{\xi+\eta}, \\ |a_\mu \cdot_\mu b_\mu| &< 2^{\xi+\eta}. \end{aligned}$$

証明 加減算については,

$$|a_\mu \pm_\mu b_\mu| \leq 2^{\max\{\xi, \eta\}+1} < 2^{\max\{\xi, \eta\}+2}.$$

乗算については,

$$|a_\mu \cdot_\mu b_\mu| \leq 2^{a+b} < 2^{a+b+2}.$$

近似計算された値の誤差について, 次の 2 つの補題が成り立つ:

補題 7 任意の実数 a に対して, 精度 μ の浮動小数点体系において丸めた結果を a_μ とする. このとき,

$$|a_\mu - a| < 2^{\varepsilon(a) - \mu}$$

が成り立つ.

証明 浮動小数点体系の構造から明らか. ■

補題 8 実数 $a, b \in \mathbf{R}$ およびその近似 $a_\mu, b_\mu \in \mathbf{M}_\mu$ に対して,

$$\begin{aligned} |a_\mu| &\leq 2^\xi, & |a_\mu - a| &< 2^{\xi - \mu}, \\ |b_\mu| &\leq 2^\eta, & |b_\mu - b| &< 2^{\eta - \mu} \end{aligned}$$

が成り立っているとす. このとき,

$$\begin{aligned} |(a_\mu \pm_\mu b_\mu) - (a \pm b)| &< 2^{\xi+\eta-\mu}, \\ |a_\mu \cdot_\mu b_\mu - a \cdot b| &< 2^{\xi+\eta-\mu} \end{aligned}$$

が成り立つ.

証明 加減算については,

$$\begin{aligned}
 & |(a_\mu \pm_\mu b_\mu) - (a \pm b)| \\
 & \leq |(a_\mu \pm_\mu b_\mu) - (a_\mu \pm b_\mu)| + |a_\mu - a| + |b_\mu - b| \\
 & < 2^{\max\{\xi, \eta\}+1-\mu} + 2^{\xi-\mu} + 2^{\eta-\mu} \\
 & \leq 2^{\max\{\xi, \eta\}+2-\mu}.
 \end{aligned}$$

乗算については,

$$\begin{aligned}
 & |a_\mu \cdot_\mu b_\mu - a \cdot b| \\
 & \leq |a_\mu \cdot_\mu b_\mu - a_\mu \cdot b_\mu| + |a_\mu \cdot b_\mu - a \cdot b| \\
 & \leq |a_\mu \cdot_\mu b_\mu - a_\mu \cdot b_\mu| + |a_\mu - a||b_\mu| + |b_\mu - b||a_\mu| + |a_\mu - a||b_\mu - b| \\
 & < 2^{\xi+\eta-\mu} + 2^{\xi+\eta-\mu} + 2^{\xi+\eta-\mu} + 2^{\xi+\eta-2\mu} \\
 & < 2^{\xi+\eta-\mu} + 2^{\xi+\eta-\mu} + 2^{\xi+\eta-\mu} + 2^{\xi+\eta-\mu} \\
 & = 2^{\xi+\eta+2-\mu}.
 \end{aligned}$$

■

4.2. 提案する符号判定法 — その 1

4.1. 節で提案した誤差評価法によって, 区間法の区間演算を浮動小数点演算に置き換える:

符号判定法 2 (浮動小数点法 1)

1. γ の measure の上界 m を求める.
2. 4.1. 節の方法によって, γ を近似計算したときの誤差評価をする. 結果を

$$|\gamma_\mu - \gamma| < 2^{\xi-\mu}$$

とする.

3. $\mu \leftarrow$ (初期精度).
4. 精度 μ の浮動小数点演算で γ の値を計算する. 結果を γ_μ とする.
5. if $\gamma_\mu \geq 2^{\xi-\mu}$ return +.
6. if $\gamma_\mu \leq -2^{\xi-\mu}$ return -.
7. if $m \cdot (|\gamma_\mu| + 2^{\xi-\mu}) < 1$ return 0.
8. 精度 μ を上げる.
9. goto 4.

符号判定法 2 において, 初期精度および精度の上げ方は任意である.

4.3. 提案する浮動判定法 — その 2

区間法および浮動小数点法 1 では, γ の符号が確定するまで, 精度を上げていった. 一方,

4.1. 節の誤差評価法を応用すると, γ の符号判定の可能な精度を事前に知ることができる:

定理 9 実代数的数 $\alpha \in \mathbf{A}$ および浮動小数点数 $\alpha_\mu \in \mathbf{M}_\mu$ に対して,

$$|\alpha_\mu - \alpha| < 2^{\xi-\mu}, \quad M(\alpha) < 2^m$$

を仮定する. このとき,

$$\mu > \xi + m + 1$$

ならば, α の符号が確定する. すなわち,

- $\alpha > 0$ if $\alpha_\mu > 2^{-m-1}$.
- $\alpha < 0$ if $\alpha_\mu < -2^{-m-1}$.
- $\alpha = 0$ otherwise.

証明 仮定から,

$$|\alpha_\mu - \alpha| < 2^{-m-1}$$

が導かれる. 今, $|\alpha_\mu| \leq 2^{-m-1}$ と仮定すると,

$$|\alpha| \leq |\alpha_\mu| + |\alpha_\mu - \alpha| < 2^{-m} < 1/M(\alpha)$$

であるから, 定理 2 より $\alpha = 0$. 残りの場合についても同様に示せる. ■

上の定理を符号判定法に応用する:

符号判定法 3 (浮動小数点法 2)

1. γ の measure の上界 M を求める.
2. 4.1. 節の方法によって, γ を近似計算したときの誤差評価をする. 結果を

$$|\gamma_\mu - \gamma| < 2^{\xi-\mu}$$

とする.

3. 小さい精度 μ_0 の浮動小数点演算で γ の値を計算する. 結果を γ_{μ_0} とする.
4. **if** $\gamma_{\mu_0} \geq 2^{\xi-\mu_0}$ **return** +.
5. **if** $\gamma_{\mu_0} \leq -2^{\xi-\mu_0}$ **return** -.
6. $m \leftarrow \lfloor \log_2 M \rfloor$.
7. 精度 $\mu = m + \xi + 2$ の浮動小数点演算で $f(\alpha_1, \dots, \alpha_n)$ の値を計算する. 結果を γ_μ とする.
8. **if** $\gamma_\mu > 2^{-m-1}$ **return** +.
9. **if** $\gamma_\mu < -2^{-m-1}$ **return** -.
10. **return** 0.

3. から 5. で, 小さい精度 μ_0 で符号判定を試みるのは, γ が 0 でなければ, ほとんどの場合 γ の符号判定は小さい精度でも可能だからである.

5. 計算実験

区間法, 浮動小数点法 1 および 浮動小数点法 2 を Graham の二次元凸包構成アルゴリズム (例えば [2] を参照) に応用し, 計算機 (DEC alpha server 4100/5/400, 使用言語 C++) に実装した. 区間法および浮動小数点法 1 において, 初期精度および精度の上げ方は任意であったが, ここでは, 初期精度を 32 とし, 以後 2 倍ずつ上げていった. また, 浮動小数点法 2 においても初期精度が任意であったが, 同様に 32 とした. さまざまな n の値に対して, 以下のような点集合の n 点からなる部分集合をそれぞれ 5 例ずつ作成し, 凸包を構成した. 構成された凸包の例および平均 CPU time を図 1-3 に示す.

実験 1

$$\left(\sqrt{\frac{p}{q}}, \sqrt{\frac{r}{s}}\right), \quad \begin{cases} q = 2, \dots, 255, & p = 1, \dots, q-1, \\ s = 2, \dots, 255, & r = 1, \dots, s-1. \end{cases}$$

実験 2

$$\left(\sqrt{\frac{p}{q+\delta}}, \sqrt{\frac{p}{q}}\right), \quad q = 129, \dots, 254, \quad p = 127, \dots, q-2, \quad \delta = -1, 0, 1.$$

実験 3

$$\left(\sqrt{\frac{p}{q}}, \sqrt{\frac{p}{q}}\right), \quad q = 129, \dots, 255, \quad p = 128, \dots, q-1.$$

3 つの符号判定法によって, 実代数的数の符号が正しく判定されることを確認した. 以上の 3 つの実験のすべてにおいて, 浮動小数点法 2, 浮動小数点法 1, 区間法の順に高速であることがわかった. また, 式の値が 0 となる場合が多ければ多いほど, 各符号判定法の計算時間の差が広がった.

6. 今後の発展

- 本研究で提案した符号判定法の計算量を理論的・実験的に解析する.
- 実数とは限らない代数的数の零判定問題へと問題を拡張する.
- 提案した符号判定法を実用的な問題に応用する.

参 考 文 献

- [1] M. Mignotte, *Mathematics for Computer Algebra*, Springer-Verlag, New York, 1992.
- [2] F. P. Preparata and M. I. Shamos, *Computational Geometry*, Springer-Verlag, New York, 1985.
- [3] 関川 浩, 近似計算による代数的数の符号判定について, 京都大学数理解析研究所講究録 941 (1995), pp. 185-193.
- [4] C. Yap and T. Dubé, The exact computation paradigm, D.-Z. Du and F. Hwang (eds.), *Computing in Euclidean Geometry*, 2nd ed., World Scientific, Singapore, 1995, pp. 452-492.

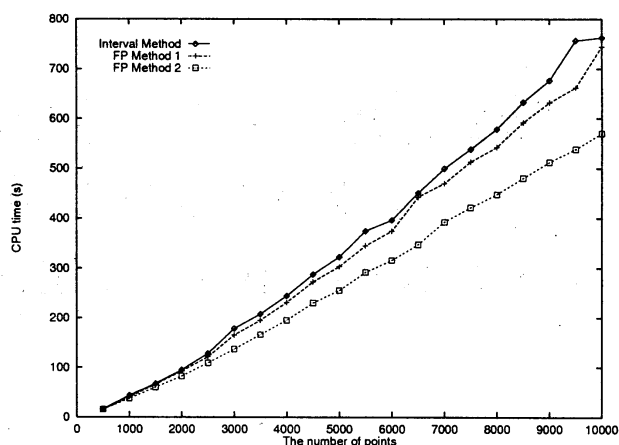
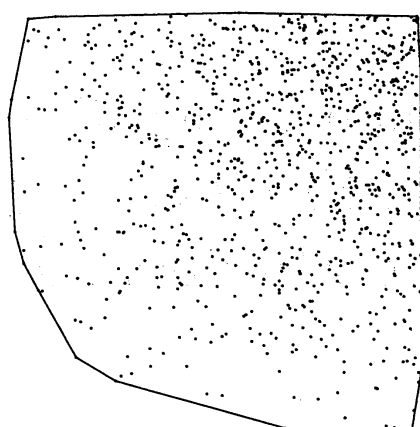


Fig. 1 実験 1 で構成された凸包の例と実験結果.

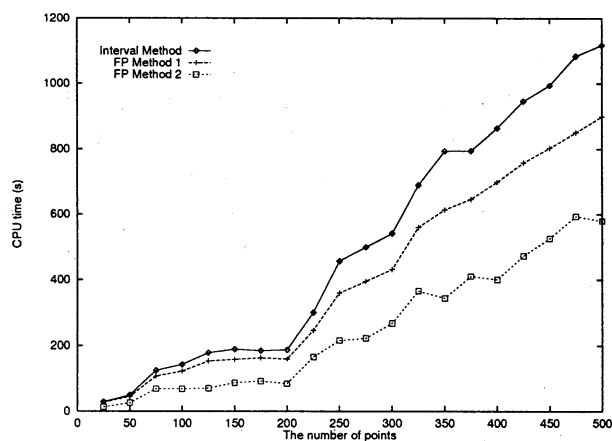
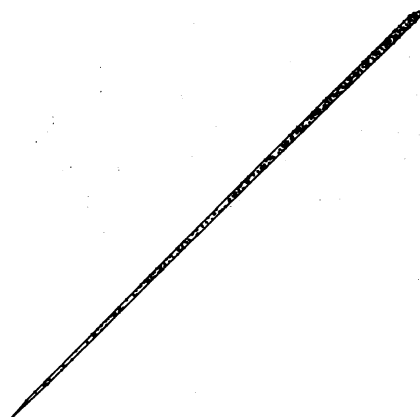


Fig. 2 実験 2 で構成された凸包の例と実験結果.

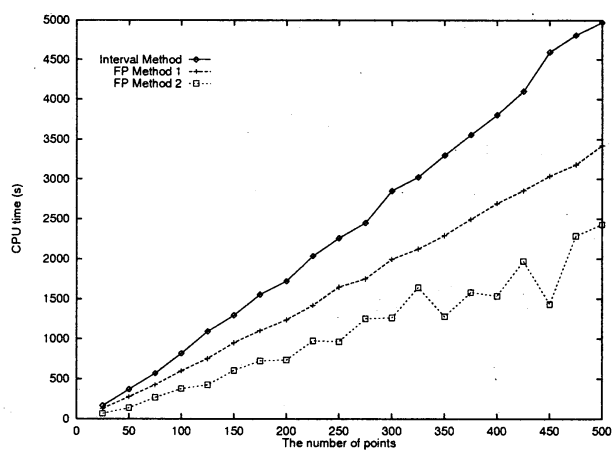
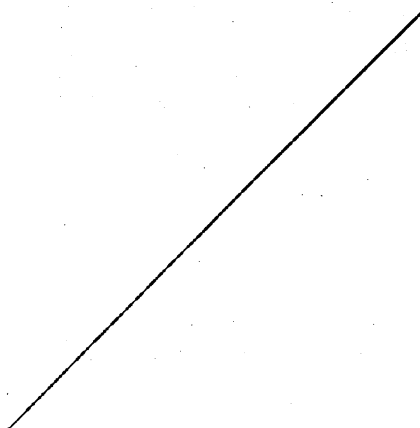


Fig. 3 実験 3 で構成された凸包の例と実験結果.