

有理関数型 Brent 法の提案

日大生産工 (院)
日大生産工

○ 茂木 渉
篠原 正明

1. はじめに

非線形方程式の数値解法は数多く存在しているが、根の存在する保証がない、存在するとしても根の個数がわからない、どんな問題にも適用できる万能な解法が存在しないといった理由から、非線形方程式の根を求めることは困難であるとされている。従って、手法の改良、組み合わせ、使い分けなど、問題に応じて試行錯誤していかねばならない。

例えば、一般的に非線形方程式を解くには Newton-Raphson 法を使用するのが良いとされるが、大域的収束性がないなどの弱点もある。そこで序盤は二分法などで荒い解を求めておき、最終段階で根の精度に磨きをかけるために Newton-Raphson 法を使う、といった混成アルゴリズムがある。

本論文では、導関数や関数形が利用できない方程式を問題として想定し、根の囲い込みを前提とする中で有効な手法といわれている「Brent の方法」に「有理関数補間」を組み合わせた解法アルゴリズムを提案する。

2. Van Wjingaarden-Dekker-Brent 法

Van Wjingaarden-Dekker-Brent 法 (Brent の方法) は、二分法の確実性と、逆 2 次補間や割線法といった高次の収束をする手法を組み合わせた解法である [1,3,4]。

超 1 次収束するはずの方法を監視しながら使用し、超 1 次収束しないようならば一時的に二分法に切り替え、少なくとも 1 次収束させるというものであり、初期区間が根を含み、その区間内で関数が計算できる限り、必ず収束することが保証されており、確実性と速さを兼ね備える優れた手法であるといわれている。

2.1. 逆 2 次補間

逆 2 次補間 (*inverse quadratic interpolation*) とは、3 点を通る逆 2 次関数 (2 次関数の逆関数) を使い、根の近くに急速に収束させることをねらう手法である。根の存在範囲を超えて発散してしまう危険性や二分法よりも収束が遅い場合もあるが、妥当な場合には 1.839 次収束する。

逆 2 次補間の公式は、式の次数が 2 と低いことなどの理由から、Lagrange 補間 (*Lagrange interpolation*) によって導出するのが簡単である。Lagrange の補間多項式は (1),(2) 式で与えられる。

$$y = \sum_{k=0}^N L_k(x) y_k \quad (1)$$

$$L_k(x) = \prod_{j=0, j \neq k}^N \frac{x - x_j}{x_k - x_j} \quad (2)$$

3 点を $[a, f(a)]$, $[b, f(b)]$, $[c, f(c)]$ として補間式 (1),(2) に代入すると、(3) 式のようにになる。

$$\begin{aligned} y &= \frac{(x-a)(x-b)}{(c-a)(c-b)} f(c) \\ &+ \frac{(x-b)(x-c)}{(a-b)(a-c)} f(a) \\ &+ \frac{(x-c)(x-a)}{(b-c)(b-a)} f(b) \end{aligned} \quad (3)$$

(3) 式の逆関数、つまり x と y を入れ替えた関数を作る。 $[x \leftrightarrow y, a \leftrightarrow f(a), b \leftrightarrow f(b), c \leftrightarrow f(c)]$

$$\begin{aligned} x &= \frac{[y-f(a)][y-f(b)]c}{[f(c)-f(a)][f(c)-f(b)]} \\ &+ \frac{[y-f(b)][y-f(c)]a}{[f(a)-f(b)][f(a)-f(c)]} \\ &+ \frac{[y-f(c)][y-f(a)]b}{[f(b)-f(c)][f(b)-f(a)]} \end{aligned} \quad (4)$$

この逆 2 次関数と x 軸との交点は、 $y = 0$ と置くことにより、(5) 式のように書くことができる。このときの x の値を根の推定値とする。

$$\begin{aligned} x &= \frac{f(a)f(b)c}{[f(c)-f(a)][f(c)-f(b)]} \\ &+ \frac{f(b)f(c)a}{[f(a)-f(b)][f(a)-f(c)]} \\ &+ \frac{f(c)f(a)b}{[f(b)-f(c)][f(b)-f(a)]} \end{aligned} \quad (5)$$

Proposal of Rational Function Type Brent's Method

Wataru MOGI[†] and Masaaki SHINOHARA

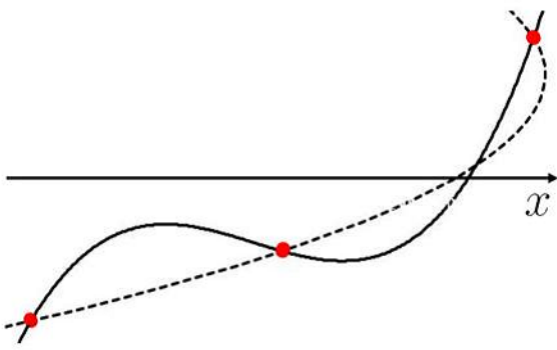


図 1. 逆 2 次補間

2.2. 割線法

割線法 (*secant method*) は 2 点を通る直線を利用して根を推定する手法であり、線形関数で補間することから、線形逆補間法とも呼ばれる。また、割線法の収束次数は、黄金分割比 ($=1.618$) であることが知られている。但し、確実に根を囲い込む保証はなく、状況によっては発散する可能性もある。

2 点を $[a, f(a)], [b, f(b)]$ とすると、補間式は (6) 式のようになる。

$$x = b - f(b) \frac{b - a}{f(b) - f(a)} \quad (6)$$

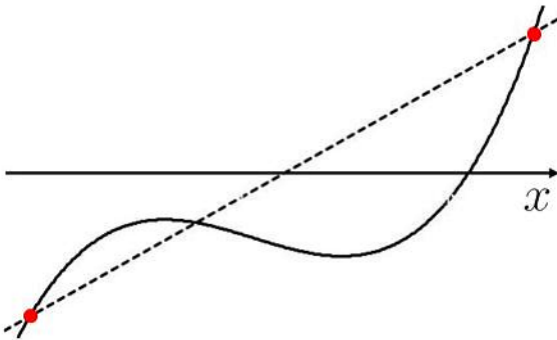


図 2. 割線法

2.3. 二分法

二分法 (*bisection method*) は古典的で単純であるが、失敗のない頑健 (*robust*) な方法である。

ある区間 $[x_l, x_u]$ において、 $f(x_l)$ と $f(x_u)$ の符号が異なれば、区間内に根が存在することが中間値の定理によって保証される。ここで、区間の中点 $x_m = (x_l + x_u)/2$ における関数の値 $f(x_m)$ の符号を調べれば、根が中点のどちらにあるか絞り込むことができる。

3. 有理関数逆補間

補間に多項式ではなく、有理関数を用いる方法が有理関数補間 (*rational function interpolation*) である。

多項式補間ではうまく近似できない場合も、有理関数補間では近似できることがあり、収束次数の面でも有利であることが示されている [2]。

3 点 $(x_k, y_k), (x_{k-1}, y_{k-1}), (x_{k-2}, y_{k-2})$ を基にして有理関数補間をするには、最も簡単な有理 1 次式

$$y = \frac{\alpha x + \beta}{\gamma x + \delta} \quad (7)$$

を使用する。ここで、 δ は任意で決められるため、 $\delta = 1$ とするのが一般的である。

3 点を (7) 式に代入し、それらを「 α, β, γ を未知数とする 3 元連立 1 次方程式」とみなして解き、(7) 式において $y = 0$ を満たす x の値を根の推定値とする。

但し、連立方程式をそのまま解くのは手間がかかるので、以下のような工夫をする。

[原点移動] 変数変換

$$\begin{cases} \xi &= x - x_k \\ \xi_{k-1} &= x_{k-1} - x_k \\ \xi_{k-2} &= x_{k-2} - x_k \end{cases} \quad (8)$$

と置き、有理 1 次式を

$$\alpha \xi + \beta - \gamma \xi y = y \quad (9)$$

とすれば「 $x = x_k \rightarrow y = y_k$ 」の条件から、ただちに

$$\beta = y_k \quad (10)$$

が得られるので、あとは残りの 2 元連立 1 次方程式

$$\begin{cases} \alpha \xi_{k-1} - \gamma \xi_{k-1} y_{k-1} = y_{k-1} - y_k \\ \alpha \xi_{k-2} - \gamma \xi_{k-2} y_{k-2} = y_{k-2} - y_k \end{cases} \quad (11)$$

から α を求める。連立 1 次方程式を Cramer の公式 (*Cramer's rule*) によって解くと、

$$\begin{aligned} \alpha &= \frac{\begin{vmatrix} y_{k-1} - y_k & -\xi_{k-1} y_{k-1} \\ y_{k-2} - y_k & -\xi_{k-2} y_{k-2} \end{vmatrix}}{\begin{vmatrix} \xi_{k-1} & -\xi_{k-1} y_{k-1} \\ \xi_{k-2} & -\xi_{k-2} y_{k-2} \end{vmatrix}} \\ &= \frac{\xi_{k-2} y_{k-2} (y_k - y_{k-1}) - \xi_{k-1} y_{k-1} (y_k - y_{k-2})}{\xi_{k-1} \xi_{k-2} (y_{k-1} - y_{k-2})} \end{aligned} \quad (12)$$

となる。これより (9) 式の有理関数補間式を $y = 0$ と置いた 1 次方程式

$$0 = \alpha \xi + \beta \quad (13)$$

を解き, ξ を求め,

$$x = x_k + \xi \quad (14)$$

を根の近似値とする.

4. 有理関数型 Brent 法

ここで, Brent の方法における逆 2 次補間の代わりに, 有理関数逆補間を用いれば, 確実かつ高速なアルゴリズムができるだろうと考えた.

Brent の方法の基礎アルゴリズム [4] を基に作成したアルゴリズムを以下に示す. 本アルゴリズム中に a, b, c, d, s 及び $mflag$ という変数を使用しているが, これらはそれぞれ次のような意味を持つ.

- (i) a は $f(a)$ と $f(b)$ の符号が反対になるような前反復時もしくはそれ以前の値.
- (ii) b は最新の値で, $f(x) = 0$ に最も近い値.
- (iii) c は前反復時の値.
- (iv) d は前々反復時の値.
- (v) s は補間または二分法によって得られた根の推定値.
- (vi) $mflag$ は二分法を使用したか否かを示すフラグ.

[STEP1]

$f(a) \cdot f(b) < 0$ となるような初期区間 $[a, b]$ を決定し, STEP2 へ.

[STEP2]

もし $|f(a)| < |f(b)|$ ならば STEP3 へ, そうでなければ STEP4 へ.

[STEP3]

a と b を入れ替え, STEP4 へ.

[STEP4]

$c := a$ とし, STEP5 へ.

[STEP5]

$mflag = 1$ とし, STEP6 へ.

[STEP6]

もし, 3 点 a, b, c が互いに異なる, つまり $f(a) \neq f(b)$ かつ $f(b) \neq f(c)$ ならば STEP7(有理関数逆補間) へ. そうでなければ STEP8(割線法) へ.

[STEP7]

有理関数逆補間で根の推定を試み STEP9 へ.

$$s := b - \frac{f(b)[c-b][a-b][f(c)-f(a)]}{f(a)[a-b][f(b)-f(c)] - f(c)[c-b][f(b)-f(a)]} \quad (15)$$

[STEP8]

割線法で根の推定を試み, STEP9 へ.

$$s := b - f(b) \frac{b-a}{f(b)-f(a)} \quad (16)$$

[STEP9]

Brent の追加条件 1,2,3 のいずれかを満たしてしまうならば, 補間は失敗であるため STEP10(二分法) へ. 満たさないのならば補間値を採用するので STEP12 へ.

[Brent の追加条件]

1. s が $(3a+b)/4$ と b の間にない
2. $mflag = 1$ であり, $b \neq c$ かつ $|s-b| \geq |b-c|/2$ である
3. $mflag = 0$ であり, $c \neq d$ かつ $|s-b| \geq |c-d|/2$ である

[STEP10]

補間失敗. 二分法で根を推定し, STEP11 へ.

$$s := \frac{a+b}{2} \quad (17)$$

[STEP11]

$mflag = 1$ として, STEP13 へ.

[STEP12]

$mflag = 0$ として, STEP13 へ.

[STEP13]

$d := c$ として, STEP14 へ.

[STEP14]

$c := b$ として, STEP15 へ.

[STEP15]

もし $f(a) \cdot f(b) < 0$ ならば STEP16 へ. そうでなければ STEP17 へ.

[STEP16]

$b := s$ と置き, STEP18 へ.

[STEP17]

$a := s$ と置き, STEP18 へ.

[STEP18]

もし $|f(a)| < |f(b)|$ ならば STEP19 へ. そうでなければ STEP20 へ.

[STEP19]

a と b を入れ替え, STEP20 へ.

[STEP20]

停止条件を満たしていれば b を返して終了する. 満たしていなければ STEP6 へ.

5. 数値計算例

提案した手法と Brent の方法とで、以下の非線形方程式の問題を解き、両手法の比較を行った。各例題に記された括弧内の値は初期区間を意味している。

また、計算の停止条件には、一般的によく使用される ε 法と δ 法を用い、2 つを併用 (OR の形で使用) した場合で、停止条件を満たすまでのそれぞれの反復回数及びそのときの関数値と根の存在区間を比較した。

[ε 法]

関数の絶対値で収束判定をする方法。

$$|f(b)| < \varepsilon \quad (18)$$

[δ 法]

根の存在区間の幅で収束判定をする方法。

$$|b - a| < \delta \quad (19)$$

但し、 ε, δ は収束判定用定数であり、本論文では共に 10^{-8} としている。

例 1. $2e^{x-1} - 1 = 0$ $[-3, 3]$ (20)

例 2. $\tanh x + 0.2x + 0.3 = 0$ $[-3, 3]$ (21)

例 3. $x - \sin x - \cos x = 0$ $[0, 2]$ (22)

例 4. $\ln x - x + 2 = 0$ $[2, 4]$ (23)

例 5. $(x + 3)(x - 1)^2 = 0$ $[-4, 4/3]$ (24)

例 6. $\tan x - 3x + 1 = 0$ $[0, 1]$ (25)

例 7. $x^3 - 6x^2 + 12x - 11 = 0$ $[3, 4]$ (26)

6. おわりに

関数の値のみ利用可能な非線形方程式の解法として高い評価を得ている Brent の方法の改良案として、有理関数補間を活用した有理関数型 Brent 法を提案した。

数値計算の比較により、反復回数は Brent の方法より 1 回少なくなるか同等程度であり、多くなるということはほぼない。反復回数が減る理由は、逆 2 次補間では Brent の追加条件を満たしてしまうが、有理関数補間では満たさず、補間がうまくいくことによる。

また、同じ条件で逆 2 次補間と有理関数補間を使用した場合、有理関数補間の方が根に近い値に推定されることが多く、同じ反復回数での最終的な数値解を比較しても、(関数の絶対値や根の存在区間が 0 に近いという意味で) 厳密解に近いことが多い。

表 1. 数値実験結果 (Brent の方法)

	回数	$ f(b) $	$ b - a $
例 1	7 回	2.55×10^{-11}	9.28×10^{-5}
例 2	6 回	5.83×10^{-10}	7.49×10^{-4}
例 3	6 回	1.95×10^{-9}	2.07×10^{-6}
例 4	4 回	1.54×10^{-9}	6.83×10^{-6}
例 5	10 回	4.97×10^{-14}	6.04×10^{-8}
例 6	6 回	5.50×10^{-10}	3.43×10^{-4}
例 7	6 回	2.82×10^{-12}	7.88×10^{-9}

表 2. 数値実験結果 (有理関数型 Brent 法)

	回数	$ f(b) $	$ b - a $
例 1	6 回	2.21×10^{-9}	1.14×10^{-4}
例 2	6 回	3.39×10^{-10}	5.82×10^{-4}
例 3	6 回	2.43×10^{-9}	2.04×10^{-6}
例 4	4 回	1.27×10^{-9}	5.62×10^{-6}
例 5	10 回	7.10×10^{-15}	9.04×10^{-7}
例 6	6 回	2.82×10^{-11}	1.29×10^{-7}
例 7	5 回	5.06×10^{-10}	1.32×10^{-5}

さらに、補間式内の演算回数を比較しても、(5) 式は 20 回、(15) 式は 17 回であり、これらのことから有理関数補間の有効性や Brent の方法への組み込みの有用性などは十分に示せたのではないかと考えられる。

具体的に実装をする上では、手法の切り替え時の無駄な計算量の減少、丸め誤差の全体の戦略への影響を考慮する必要があるため、それらを念頭に入れた有理関数逆補間公式の導出やアルゴリズムの修正などは今後の課題とする。

参考文献

- [1] 玄光男・井田憲一：「パソコン数値計算ライブラリ」，株式会社 HBJ(1986)
- [2] 戸川隼人：「UNIX ワークステーションによる科学技術計算ハンドブック [基礎篇 C 言語版]」，サイエンス社 (1992)
- [3] William H.Press・Brian P.Flannery・Saul A.Teukolsky・William T.Veterling(著)，丹慶勝市・佐藤俊郎・奥村晴彦・小林誠 (訳)：「NUMERICAL RECIPES in C[日本語版]」，技術評論社 (1993)
- [4] Wikipedia-the free encyclopedia-：「Brent's method」，
http://en.wikipedia.org/wiki/Brent's_method，
最終閲覧日 (2008/6/14)