

無限次元ガウスの消去法を用いた radii-polynomial approach の改良

関根研究室 2131701 齋藤 悠希

1. 背景と目的

非線形微分方程式は、さまざまな現象を数学モデル化することができる。この非線形微分方程式を解くことで、数学モデルの現象の解析を行うことができる。

radii-polynomial approach は、非線形方程式の解の精度保証付き数値計算に関する定理である。従来の radii-polynomial approach は、難しい計算部分を簡略化している。そのため、精度保証の性能が低くなる。そこで、この簡略化をせずに、無限次元ガウスの消去法を用いて、精度保証の性能を向上させる。

2. radii-polynomial approach

- X, Y は Banach 空間
- $\mathcal{L}(X, Y)$ は X から Y への有界線形作用素集合
- 有界線形作用素 $A^\dagger \in \mathcal{L}(X, Y), A \in \mathcal{L}(Y, X)$
- 作用素 $F : X \rightarrow Y$ が C^1 -Fréchet 微分可能

$\tilde{x} \in X$ に対して、正定数 Y_0, Z_0, Z_1 および非減少関数 $Z_2(r) (r > 0)$ が存在して、次の不等式を満たすとする。

$$\|AF(\tilde{x})\|_X \leq Y_0 \quad (1)$$

$$\|I - AA^\dagger\|_{\mathcal{L}(X)} \leq Z_0 \quad (2)$$

$$\|A(DF(\tilde{x}) - A^\dagger)\|_{\mathcal{L}(X)} \leq Z_1 \quad (3)$$

$$\|A(DF(b) - DF(\tilde{x}))\|_{\mathcal{L}(X)} \leq Z_2(r) \quad (4)$$
$$\forall b \in \overline{B(\tilde{x}, r)}$$

このとき、radii polynomial を以下で定義する。

$$p(r) := Z_2(r)r^2 - (1 - Z_1 - Z_0)r + Y_0 \quad (5)$$

これに対し、 $p(r_0) < 0$ となる $r_0 > 0$ が存在するならば、 $F(x) = 0$ を満たす解 \tilde{x} が $\overline{B(x, r)}$ 内に一意に存在する。

ここで、 $DF(\bar{x})$ を F の \bar{x} における Fréchet 微分、 A^\dagger を $DF(\bar{x})$ の近似、 A を A^\dagger の近似左逆作用素 ($AA^\dagger \approx I$) とする。

3. 提案手法

従来の radii-polynomial approach では、 A^\dagger, A は、簡略化のために近似した作用素としていた。

提案手法では、 A^\dagger, A を以下のように、正確なニュートン法である作用素として計算をする。

$$A^\dagger = DF(\bar{x}), \quad A = DF(\bar{x})^{-1} \quad (6)$$

この式(6)を、radii-polynomial approach の式(1)に代入する。

$$\|DF(\bar{x})^{-1}F(\tilde{x})\|_X \leq Y_0 \quad (7)$$

ここで、 $\phi := DF(\bar{x})^{-1}F(\tilde{x})$ とする。この式に、無限次元ガウス消去法を用いて変形する。ただし、 Π_N は射影作用素とする。

$$DF(\bar{x})\phi = F(\tilde{x}) \quad (8)$$

$$\begin{pmatrix} \Pi_N DF \Pi_N & \Pi_N DF(I - \Pi_N)\phi \\ (I - \Pi_N)DF \Pi_N & (I - \Pi_N)DF \Pi_N \end{pmatrix} \begin{pmatrix} \Pi_N \phi \\ (I - \Pi_N)\phi \end{pmatrix} \quad (9)$$
$$= \begin{pmatrix} \Pi_N F(\tilde{x}) \\ (I - \Pi_N)F(\tilde{x}) \end{pmatrix}$$

4. 今後の課題

- 提案手法で提示した式(9)のガウスの消去法による展開
- Julia を用いたプログラムの実証

参考文献

- [1] 高安亮紀, Julia 言語を使った精度保証付き数値計算のチュートリアル