

令和 6 年度

卒業論文

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

2024 年 12 月 25 日

指導教員 関根 晃太 准教授

千葉工業大学 情報科学部 情報工学科

学生番号 2131701

齋藤悠希

目次

第 1 章	はじめに	3
第 2 章	準備	4
2.1	バナッハ空間	4
2.2	作用素	7
2.3	Banach の不動点定理	18
2.4	簡易ニュートン写像	20
2.5	フーリエ級数	21
2.6	離散フーリエ変換	26
2.7	畳み込みの FFT アルゴリズム	26
第 3 章	Newton Kantorovich の定理を用いた精度保証付き数値計算	31
3.1	radii polynomial approach [2]	31
3.2	Newton-Kantorovich の定理の特徴 [3]	32
第 4 章	既存の van der Pol 方程式の精度保証付き数値計算 [2]	36
4.1	van der Pol 方程式	36
4.2	フーリエ・スペクトル法	36
4.3	重み付き空間と作用素の決定	39
4.4	Y_0, Z_0, Z_1, Z_2 の評価	42
第 5 章	無限次元ガウスの消去法 [4]	48
5.1	射影	48
5.2	射影を用いた Banach 空間上の無限次元ガウスの消去法	50
第 6 章	提案手法	55
6.1	定義	55
6.2	計算手順	56
第 7 章	実験結果	58
7.1	実験環境	58

7.2	実験結果	58
第 8 章	おわりに	60
	参考文献	62

第 1 章

はじめに

今日では，コンピュータを用いることで，微分方程式など手計算では困難な数式問題を解析することができる．しかし，コンピュータは有限の数値のみを扱うことしかできないため，計算結果が正確にならず，出力した解には誤差が含まれる．そこで，問題と解とその誤差範囲を評価し保証するために，精度保証付き数値計算を用いる [1]．

精度保証付き数値計算に関する定理の一つに，Newton-Kantorovich 型の定理を利用した radii polynomial approach がある．この定理は有限次元や無限次元を問わず，非線形方程式や偏微分方程式など殆どの微分方程式に用いることができる．

従来の radii polynomial approach では，Banach 空間は重み付き l_1 空間より定義している． l_1 空間を用いることで，従来手法よりも精度の向上を行うことができる．そこで，本研究では，radii polynomial approach の評価値計算の際に，無限次元ガウスの消去法を用いていることで，適用可能な問題を増やすことを目的とする．

本論文の構成は以下の通りである．2 章では準備として関数解析の基本知識をまとめる．3 章では既存手法である radii polynomial approach の定理と証明，4 章では van der Pol 方程式を対象として既存の radii polynomial approach を用いた計算機での実装法を提示する．5 章では，無限次元ガウスの消去法に関する知識をまとめる．6 章では提案手法として，無限次元ガウスの消去法を radii polynomial approach に適用させる手法を示す．7 章では，6 章で述べた手法の計算機を用いた検証の結果を記載する．8 章では，本論文のまとめを行う．

第 2 章

準備

2.1 バナッハ空間

定義 1 (線形空間の公理). 空でない集合 X が, 係数体 \mathbb{K} 上の線形空間であるとは, 任意の $u + v \in X$ とスカラー $\alpha \in \mathbb{K}$ に対して, 加法 $u + v \in X$ とスカラー乗法 $\alpha u \in X$ が定義されていて, 任意の $u, v, w \in X$ とスカラー $\alpha, \beta \in \mathbb{K}$ に対して次のことが成り立つことである.

1. $(u + v) + w = u + (v + w)$
2. $u + v = v + u$
3. $u + 0 = u$ となる $0 \in X$ が一意に存在
4. $u + (-u) = 0$ となる $-u \in X$ が一意に存在
5. $\alpha(u + v) = \alpha u + \alpha v$
6. $(\alpha + \beta)u = \alpha u + \beta u$
7. $(\alpha\beta)u = \alpha(\beta u)$
8. $1u = u, 1 \in \mathbb{K}$

定義 2 (ノルムとノルム空間の定義). X を係数体 \mathbb{K} 上の線形空間とする. X で定義された関数 $\|\cdot\| : X \rightarrow \mathbb{K}$ 上で定義された関数が X のノルムであるとは

1. $\|u\| \geq 0, \quad u \in X$
2. $\|u\| = 0 \Leftrightarrow u = 0$
3. $\|\alpha u\| = |\alpha| \|u\|, \quad (\alpha \in \mathbb{K}, u \in X)$
4. $\|u + v\| \leq \|u\| + \|v\|$

が成立することである. さらに X に 1 つのノルムが指定されているとき, X はノルム空間という.

定義 3 (ノルム空間の収束と極限). X をノルム空間とする. X の点列 $(u_n) \subset X$ は

$$\forall \varepsilon > 0, \exists N \in \mathbb{N}, \forall n \geq N \text{ に対して } \|u_n - u\| < \varepsilon$$

のとき、点 $u \in X$ に収束するといい、

$$\|u_n - u\| \rightarrow 0, (n \rightarrow \infty)$$

と表す。このとき、 u を u_n の極限といい、

$$u_n \rightarrow u, (n \rightarrow \infty)$$

と表す。

定義 4 (Cauchy 列). X をノルム空間とする。そのとき X が Cauchy 列であるとは

$$u_n - u_m \rightarrow 0, (n, m \rightarrow \infty)$$

が成立することである。即ち

$$\|u_n - u_m\| \rightarrow 0, (n, m \rightarrow \infty)$$

が成立することである。

定義 5 (完備). X をノルム空間とする。 X が完備であるとは、任意の Cauchy 列 (u_n) が X の中で極限をもつことである。すなわち、任意の Cauchy 列 $(u_n \subset X)$ が

$$\|u_n - u\| \rightarrow 0, (n \rightarrow \infty)$$

となる極限 u を X 内に持つことである。

定義 6 (Banach 空間). ノルム空間 X が Banach 空間であるとは、 X が完備であることである。

定理 1 (逆三角不等式). X をノルム空間とする。任意の $u, v \in X$ について次の不等式を満たす。

$$|||u| - |v|| \leq \|u - v\|$$

証明. 任意の $u, v \in X$ について

$$\begin{aligned} \|u\| &= \|u - v + v\| \leq \|u - v\| + \|v\| \\ \|v\| &= \|v - u + u\| \leq \|v - u\| + \|u\| = \|u - v\| + \|u\| \end{aligned}$$

となる。よって

$$\|u\| - \|v\| \leq \|u - v\| \quad \|v\| - \|u\| \leq \|u - v\|$$

となるため、

$$|||u| - |v|| \leq \|u - v\|$$

を持つ。 ■

定義 7 (有界列). X をノルム空間とする。そのとき X の点列 (u_n) が有界列とは任意の $n \in \mathbb{N}$ に対して

$$\|u_n\| \leq M$$

となる定数 $M > 0$ が存在することである。

定理 2 (Cauchy 列ならば有界列). X をノルム空間とする. そのとき X の点列 (u_n) が Cauchy 列ならば有界列でもある.

証明. X の点列 (u_n) が Cauchy 列であるために, $\varepsilon - N$ 論法を用いた表記で

$$\forall \varepsilon > 0, \exists N \in \mathbb{N}, \forall n, m \geq N \text{ に対して } \|u_n - u_m\| < \varepsilon$$

を満たす. $\varepsilon = 1$ としても, それに対応した N が存在し, 任意の $n \geq N$ に対して

$$\|u_n - u_N\| < 1$$

を満たす.

任意の $n \geq N$ に対して $\|u_n\|$ が $\|u_N\|$ で評価できることを示す. 逆三角不等式である定理 1 を用いると

$$|\|u_n\| - \|u_N\|| \leq \|u_n - u_N\| < 1$$

となる. 絶対値の性質より $|\|u_n - u_N\|| < 1$ は

$$\|u_N\| - 1 \leq \|u_n\| < \|u_N\| + 1$$

となる. よって

$$M = \max\{\|u_1\|, \|u_2\|, \dots, \|u_{N-1}\|, \|u_N\| + 1\}$$

とすると, 任意の $n \in \mathbb{N}$ について

$$\|u_n\| \leq M$$

が成り立つため, 点列 (u_n) は有界列である. ■

定義 8 (線形部分空間). 線形空間 X の空でない集合 M が任意の元 $u, v \in M$ と任意の係数体 $\alpha \in \mathbb{K}$ に対して

$$\begin{aligned} u + v &\in M \\ \alpha u &\in M \end{aligned}$$

を満たすとき, M は線形空間 X の線形部分空間と呼ぶ.

定義 9 (ノルム空間の開球). X をノルム空間とする. $x \in X$ とし, $r > 0$ を正実数とする. そのとき, 集合

$$B_X(x, r) := \{y \in X \mid \|x - y\|_X < r\}$$

を中心 x , 半径 r の開球という. X が明らかな場合は $B_X(x, r)$ を省略して $B(x, r)$ と表記する.

定義 10 (ノルム空間の開集合). X をノルム空間とし, M を X の部分集合とする. 任意の $x \in M$ に対して, $B_X(x, r) \subset M$ となる $r > 0$ が存在する場合, M が開集合であるという.

定義 11 (ノルム空間の閉集合). X をノルム空間とし, M を X の部分集合とする. M が閉集合であるとは, M の任意の点列 (u_n) の極限 $u \in X$ が M にも属することである. すなわち, 点列 $(u_n) \subset M$ について

$$u_n \rightarrow u, \quad (n \rightarrow \infty) \Rightarrow u \in M$$

であるとき, M は閉集合であるという.

定義 12 (閉部分空間). X をノルム空間とし, M を X の線形部分空間が閉集合であるとき, M を閉部分空間である.

2.2 作用素

定義 13 (作用素). ある線形空間 X から別の線形空間 Y への作用素 A とは,

$$\mathcal{D}(A) := \{u \in X \mid Au \in Y\}$$

としたとき, $\mathcal{D}(A)$ のどんな元に対しても, それぞれ集合 Y のただ一つの元を指定する規則のことである. また, $\mathcal{D}(A)$ は A の定義域と呼ばれ

$$\mathcal{R}(A) := \{Au \in Y \mid u \in \mathcal{D}(A)\}$$

を値域と呼ぶ

定義 14 (単射). 線形空間 X から線形空間 Y への作用素 A が

$$u_1 \neq u_2, \quad \forall u_1, u_2 \in \mathcal{D}(A) \Rightarrow A(u_1) \neq A(u_2)$$

定義 15 (全射). 線形空間 X から線形空間 Y への作用素 A が

$$Y = \mathcal{R}(A)$$

を満たすときに作用素 A は全単射であるという.

定義 16 (逆作用素). 線形空間 X から線形空間 Y への作用素 A とし, その定義域を $\mathcal{D}(A) \subset X$, 値域を $\mathcal{R}(A) \subset Y$ とする. そのとき,

$$A^{-1}(A(u)) = u, \quad u \in \mathcal{D}(A)$$

$$A(A^{-1}(v)) = v, \quad v \in \mathcal{R}(A)$$

かつ

$$\mathcal{D}(A^{-1}) = \mathcal{R}(A)$$

$$\mathcal{R}(A^{-1}) = \mathcal{D}(A)$$

となる Y から X への作用素 A^{-1} を A の逆作用素と呼ぶ.

定理 3 (単射と逆作用素の関係). 線形空間 X から線形空間 Y への作用素 A とすると,

$$A \text{ が逆作用素を持つ} \Leftrightarrow A \text{ が単射である}$$

証明. 「 A が逆作用素を持つ $\Rightarrow A$ が単射である」の証明

単射の定義 14 の対偶「任意の $u_1, u_2 \in \mathcal{D}(A)$ に対し $A(u_1) = A(u_2) \Rightarrow u_1 = u_2$ 」を満たすことを確かめる. A の逆作用素を A^{-1} とすると, 任意の $u_1, u_2 \in \mathcal{D}(A)$ に対し

$$\begin{aligned} A(u_1) &= A(u_2) \\ \Rightarrow A^{-1}(A(u_1)) &= A^{-1}(A(u_2)) \\ \Rightarrow u_1 &= u_2 \end{aligned}$$

となる.

「 A が単射である $\Rightarrow A$ が逆作用素 A^{-1} をもつ」の証明

A の値域の定義 $\mathcal{R}(A) = \{A(u) \in Y \mid u \in \mathcal{D}(A)\}$ より, 任意の $v \in \mathcal{R}(A)$ に対し,

$$A(u) = v$$

となる $u \in \mathcal{D}(A)$ が存在する. その上, A が単射であるため, 単射の定義の対偶より $u \in \mathcal{D}(A)$ はどんな $u \in \mathcal{R}(A)$ に対してもただ一つの元である. そのため, 作用素の定義より, 上記の $u \in \mathcal{R}(A)$ に対してただ一つの元 $u \in \mathcal{D}(A)$ を指定する規則として

$$B(v) = u$$

となる定義域 $\mathcal{D}(B) = \mathcal{R}(A)$ と値域 $\mathcal{R}(B) = \mathcal{D}(A)$ となる Y から X への作用素 B が定義できる. その上, $B(v) = u$ の $v = A(u)$ を代入すると

$$B(A(u)) = u$$

となる. 同様に, $A(u) = v$ の u に $u = B(v)$ を代入すると

$$A(B(v)) = v$$

となる. よって, 定義域 $\mathcal{D}(B) = \mathcal{R}(A)$ と値域 $\mathcal{R}(B) = \mathcal{D}(A)$ となる Y から X への作用素 B は A の逆作用素であるため, A は逆作用素を持つ. ■

定義 17 (作用素の等号). 線形空間 X から線形空間 Y への作用素 A と B が等しいとは

$$\mathcal{D}(A) = \mathcal{D}(B)$$

かつ

$$Au = Bu, \forall u \in \mathcal{D}(A) = \mathcal{D}(B)$$

が成立することであり,

$$A = B$$

と表記する.

定義 18 (作用素の連続). ノルム空間 X からノルム空間 Y への作用素 A が $u \in \mathcal{D}(A)$ で連続であるとは

$$u_n \rightarrow u, (n \rightarrow \infty)$$

となる任意の $u_n \in \mathcal{D}(A) \subset X$ に対して

$$Au_n \rightarrow Au, (n \rightarrow \infty)$$

を満たすときである. さらに, A が任意の $u \in \mathcal{D}(A)$ において連続であるとき, A は連続であるという.

定義 19 (線形作用素). 線形空間 X から線形空間 Y への作用素 A が, 任意の $u, v \in \mathcal{D}(A) \subset X$ と $\alpha \in \mathbb{K}$ に対し,

$$\mathcal{D}(A) \text{ が } X \text{ の線形部分空間}$$

$$A(u + v) = Au + Av$$

$$A(\alpha u) = \alpha Au$$

を満たすとき, A を作用素と呼ぶ.

定義 20 (線形作用素の加法). 線形空間 X から線形空間 Y への線形作用素 A と B の和を

$$(A + B)u := Au + Bu, u \in \mathcal{D}(A) \cap \mathcal{D}(B)$$

と定義する. このとき, X から Y への作用素 $A + B$ の定義域は

$$\mathcal{D}(A + B) = \mathcal{D}(A) \cap \mathcal{D}(B)$$

とする.

定義 21 (線形作用素のスカラー乗法). 線形空間 X から線形空間 Y への線形作用素 A の $\alpha \in \mathbb{K}$ によるスカラー倍を

$$(\alpha A)u := \alpha Au, u \in \mathcal{D}(A)$$

と定義する. このとき, X から Y への作用素 αA の定義域は

$$\mathcal{D}(\alpha A) := \mathcal{D}(A)$$

とする.

定義 22 (合成作用素). X, Y, Z を線形空間とする. A を Y から Z への線形作用素とし, B を X から Y への線形作用素とする. そのとき, A と B の合成作用素 AB は

$$(AB)u := A(Bu), u \in \{v \in \mathcal{D}(B) \mid Bv \in \mathcal{D}(A)\}$$

と定義する. このとき, X から Z への合成作用素 AB の定義域は

$$\mathcal{D}(AB) := \{v \in \mathcal{D}(B) \mid Bv \in \mathcal{D}(A)\}$$

とする.

定理 4 (線形作用素に対する単射性 (1)). 線形空間 X から線形空間 Y への線形作用素 A において以下は同値である.

1. 線形作用素が A の単射である.
2. $Au = 0, u \in \mathcal{D}(A) \Rightarrow u = 0$

証明. 単射の定義の対偶は

$$Au_1 = Au_2, \forall u_1, u_2 \in \mathcal{D}(A) \Rightarrow u_1 = u_2$$

となる. その上, A は線形作用素であるため,

$$Au_1 = Au_2 \Leftrightarrow A(u_1 - u_2) = 0$$

となる. $u_1 - u_2 \in \mathcal{D}(A)$ を $u \in \mathcal{D}(A)$ とおきなおせば, $1 \Rightarrow 2$ は証明された. また, 証明を逆に追うことで $2 \Rightarrow 1$ も示せる. ■

定理 5 (線形作用素に対する単射性 (2)). ノルム空間 X からノルム空間 Y への線形作用素 A とする. 不等式

$$\|u\|_X \leq K\|Au\|_Y, u \in \mathcal{D}(A)$$

を満たす定数 $K > 0$ が存在するならば, 線形作用素 A は単射である.

証明. A が線形作用素であるため, $Au = 0, u \in \mathcal{D}(A) \Rightarrow u = 0$ を使って証明する. ノルムの定義より

$$Au = 0, \forall u \in \mathcal{D}(A) \Leftrightarrow \|Au\|_Y = 0$$

となる. さらに, $Au = 0$ ならば,

$$\|u\|_X \leq K\|Au\|_Y = 0, u \in \mathcal{D}(A)$$

より $\|u\|_X = 0$ となる. よって, 再びノルムの定義より

$$\|u\|_X = 0, \forall u \in \mathcal{D}(A) \Leftrightarrow u = 0$$

より, $Au = 0$ ならば $u = 0$ となる. ■

定義 23 (有界な線形作用素). ノルム空間 X からノルム空間 Y への作用素 A に対し,

$$\|Au\|_Y \leq K\|u\|_X, u \in \mathcal{D}(A)$$

を満たす正の定数 K が存在する時, 線形作用素 A を有界な作用素と呼ぶ.

定理 6 (有界な線形作用素と連続な線形作用素). ノルム空間 X からノルム空間 Y への作用素 A に対し,

$$A \text{ が有界} \Leftrightarrow A \text{ が連続}$$

証明. 「 A が有界 $\Rightarrow A$ が連続」の証明

連続性の定義より, $u_n \rightarrow u$ となる任意の $u_n \in \mathcal{D}(A)$ に対して $Au_n \rightarrow Au$ となることを確かめる. $u_n \rightarrow u$ となる任意の $u_n \in \mathcal{D}(A)$ から $\|u_n - u\|_X \rightarrow 0, (n \rightarrow \infty)$ を持つ. その上, A は有界であることから

$$\|Au_n - Au\|_Y \leq M\|u_n - u\|_X \rightarrow 0, (n \rightarrow \infty)$$

よって, $u_n \rightarrow u, (n \rightarrow \infty)$ ならば, $Au_n \rightarrow Au$ であるため, A は連続である.

「 A が連続 $\Rightarrow A$ が有界」の証明

背理法によって証明する. すなわち, A が連続ならば, 任意の $M_2 > 0$ に対して

$$\|Au\|_Y > M_2\|u\|_X$$

を満たす $u \in \mathcal{D}$ が存在すると仮定して矛盾を見つける. この仮定より自然数 n に対して,

$$\|Au_n\|_Y > n\|u_n\|_X$$

を満たす $u_n \in \mathcal{D}(A)$ が存在する. このとき, $\|u_n\|_X \neq 0$ であることに注意する. ノルム空間 X はノルム空間全体の定義より線形空間であるため, ゼロ元 $0 \in X$ を持つ. その上, 線形作用素の定義より $\mathcal{D}(A)$ は X の部分空間であるため, ゼロ元 $0 \in \mathcal{D}(A) \subset X$ を持つ. その上, A が連続であるため, A は $0 \in \mathcal{D}(A)$ でも連続である. $\varepsilon - \sigma$ 論法による A の $0 \in \mathcal{D}(A) \subset X$ における連続の定義を記述すると

$$\forall \varepsilon > 0, \exists \delta > 0, \|u_n\|_X < \delta \text{ となる } \forall u_n \in X \text{ に対して } \|Au_n\|_Y < \varepsilon$$

となる. その上, ε を $n\|u_n\|_X$ とすると, $\delta_n > 0$ が存在し, $\|u_n\|_X < \delta$ となる任意の $u_n \in \mathcal{D}(A)$ に対して,

$$\|Au_n\|_Y < n\|u_n\|_X$$

となる. 有界ではないという仮定と組み合わせると

$$n\|u_n\|_X < \|Au_n\|_Y < n\|u_n\|_X$$

となるため矛盾する. ■

定義 24 (定義域が X の全体となる有界な線形作用素全体の集合 $\mathcal{L}(X, Y)$). 定義域が Banach 空間 X 全体となる X から Y への有界線形作用素全体を

$$\mathcal{L}(X, Y)$$

とする.

定理 7 ($\mathcal{L}(X, Y)$ は Banach 空間). X をノルム空間とし, Y を Banach 空間とする. 定義域が X 全体となる X から Y への有界な線形作用素全体の集合 $\mathcal{L}(X, Y)$ のノルムを

$$\|A\|_{\mathcal{L}(X, Y)} := \sup_{u \in X \setminus \{0\}} \frac{\|Au\|_Y}{\|u\|_X}, A \in \mathcal{L}(X, Y)$$

とすると, $\mathcal{L}(X, Y)$ は Banach 空間となる.

証明. 作用素の加法 (定義 20) と作用素のスカラー乗法 (定義 21) の定義をもとに線形空間の公理 (1) が満たされていることが導かれる. ただし, $\mathcal{L}(X, Y)$ のゼロ元は任意の $u \in X$ を $0 \in Y$ へ写す作用素であることに注意が必要である.

「ノルム空間」

$\|A\|_{\mathcal{L}(X, Y)}$ がノルムの定義を満たすことを示せばよい. ノルム空間 X と Banach 空間 Y であるため $\|\cdot\|_X \geq 0$ と $\|\cdot\|_Y \geq 0$ であることから

$$\frac{\|Au\|_Y}{\|u\|_X} \geq 0$$

となるため, $\|A\|_{\mathcal{L}(X, Y)} \geq 0$ となり, ノルムの定義 (1) はいえる.

次に, $A = 0$ ならば $\|Au\|_Y = 0$ であるため,

$$\|A\|_{\mathcal{L}(X, Y)} = \sup_{u \in X \setminus \{0\}} \frac{\|Au\|_Y}{\|u\|_X} = \sup_{u \in X \setminus \{0\}} \frac{0}{\|u\|_X} = 0$$

である. さらに, 任意の $u \in X \setminus \{0\}$ について

$$\frac{\|Au\|_Y}{\|u\|_X} = 0 \Leftrightarrow \|Au\|_Y = 0 \Leftrightarrow Au = 0$$

任意の $u \in X \setminus \{0\}$ を $0 \in Y$ へ写す作用素は $\mathcal{L}(X, Y)$ が線形空間により一意に存在し, $A = 0$ である. よって, ノルムの定義 (2) も示された.

続いて $\alpha \in K$ としたとき, Y は Banach 空間であるため, $\|\cdot\|_Y$ はノルムの定義を満たすため,

$$\|\alpha A\|_{\mathcal{L}(X, Y)} = \sup_{u \in X \setminus \{0\}} \frac{\|\alpha Au\|_Y}{\|u\|_X} = |\alpha| \sup_{u \in X \setminus \{0\}} \frac{\|Au\|_Y}{\|u\|_X} = |\alpha| \|A\|_{\mathcal{L}(X, Y)}$$

となるため, ノルムの定義 (3) も示された.

最後に, 任意の $A, B \in \mathcal{L}(X, Y)$ について

$$\begin{aligned} \|A + B\|_{\mathcal{L}(X, Y)} &= \sup_{u \in X \setminus \{0\}} \frac{\|(A + B)u\|_Y}{\|u\|_X} \\ &= \sup_{u \in X \setminus \{0\}} \frac{\|Au + Bu\|_Y}{\|u\|_X} \\ &\leq \sup_{u \in X \setminus \{0\}} \frac{\|Au\|_Y + \|Bu\|_Y}{\|u\|_X} \\ &\leq \sup_{u \in X \setminus \{0\}} \frac{\|Au\|_Y}{\|u\|_X} + \sup_{u \in X \setminus \{0\}} \frac{\|Bu\|_Y}{\|u\|_X} \\ &= \|A\|_{\mathcal{L}(X, Y)} + \|B\|_{\mathcal{L}(X, Y)} \end{aligned}$$

となり, ノルムの定義 (4) も示されたため, $\mathcal{L}(X, Y)$ はノルム空間である.

「Banach 空間」

Banach 空間であることを証明するには $\mathcal{L}(X, Y)$ の任意の Cauchy 列 $(A_n) \subset \mathcal{L}(X, Y)$ が極限 T を $\mathcal{L}(X, Y)$ 内に持つことを示せばよい。

まず、極限の候補 \tilde{A} が定義できるか確認する。任意の Cauchy 列 $(A_n) \subset \mathcal{L}(X, Y)$ は Cauchy 列の定義 (4) より

$$\|A_n - A_m\|_{\mathcal{L}(X, Y)} \rightarrow 0, (n, m \rightarrow \infty)$$

となる。任意の $u \in X \setminus \{0\}$ に対して、点列 $(A_n u) \subset Y$ は

$$\begin{aligned} \|A_n u - A_m u\|_Y &= \frac{\|(A_n - A_m)u\|_Y}{\|u\|_X} \|u\|_X \\ &\leq \sup_{\phi \in X \setminus \{0\}} \frac{\|(A_n - A_m)\phi\|_Y}{\|\phi\|_X} \|\phi\|_X \\ &= \|A_n - A_m\|_{\mathcal{L}(X, Y)} \|u\|_X \rightarrow 0, (n, m \rightarrow \infty) \end{aligned}$$

を持つため、点列 $(A_n u) \subset Y$ は Cauchy 列になる。その上、 Y は Banach 空間であるため、 Y の任意の Cauchy 列は収束し、 Y 内に極限 $\tilde{A}u$ となるような X から Y への作用素 \tilde{A} が存在する。ここで、任意の $u \in X$ に対して極限 $\tilde{A}u$ が定義されることから、 \tilde{A} の定義域は $\mathcal{D}(\tilde{A}) = X$ である。これにより、 $\mathcal{L}(X, Y)$ の任意の Cauchy 列 (A_n) の極限の候補 \tilde{A} が定義できた。

続いて、定義した極限の候補 \tilde{A} が $\mathcal{L}(X, Y)$ に属しているか確認する。 \tilde{A} が有界な線形作用素であり、かつ $\mathcal{D}(\tilde{A}) = X$ であることを示せばよい。 $\mathcal{L}(X, Y)$ の任意の Cauchy 列 (A_n) の元 A_n は線形作用素であるため、線形作用素の定義より任意の $\alpha, \beta \in \mathbb{K}$ と $u, v \in X$ について

$$A_n(\alpha u + \beta v) = \alpha A_n u + \beta A_n v$$

を持つ。よって $n \rightarrow \infty$ とすると

$$\tilde{A}(\alpha u + \beta v) = \alpha \tilde{A}u + \beta \tilde{A}v$$

となり、極限の候補 \tilde{A} は線形作用素である。次に極限の候補 \tilde{A} が有界作用素であることを示す。点列 (A_n) は Cauchy 列であるため定理 (2) より有界列でもある。すなわち、どんな $n \in \mathbb{N}$ に対しても

$$\|A_n\|_{\mathcal{L}(X, Y)} \leq M$$

となる $n \in \mathbb{N}$ に依存しない定数 M が存在する。この $n \in \mathbb{N}$ に依存しない定数 M は、任意の $n \in X$ について

$$\|A_n u\|_Y \leq M \|u\|_X$$

も満たす。 $\|A_n u\| \rightarrow \tilde{A}u$, $(n \rightarrow \infty)$ であるため、上の不等式に対して $n \rightarrow \infty$ とすると M が n に依存しないため

$$\|\tilde{A}u\|_Y \leq M \|u\|_X$$

を得る。よって、点列 (A_n) の極限の候補 \tilde{A} は $\mathcal{L}(X, Y)$ に属する。

最後に、点列 (A_n) の極限が \tilde{A} であることを示す。任意の $u \in X$ に対して、点列 $(A_n u) \subset Y$ は Y 内に極限 $\tilde{A}u$ を持つこと、すなわち

$$A_n u \rightarrow \tilde{A}u, \quad (n \rightarrow \infty)$$

を持つことから

$$\|A_n u - A_m u\|_Y \rightarrow \|A_n u - \tilde{A}u\|_Y, \quad (m \rightarrow \infty)$$

となる。その上、 $\mathcal{L}(X, Y)$ のノルムの定義と $\tilde{A} \in \mathcal{L}(X, Y)$ から

$$\|A_n - A_m\|_{\mathcal{L}(X, Y)} \rightarrow \|A_n - \tilde{A}\|_{\mathcal{L}(X, Y)}, \quad (m \rightarrow \infty)$$

を得る。点列 (A_n) が Cauchy 列であるため

$$\forall \varepsilon > 0, \exists N \in \mathbb{N}, \forall n, m \geq N \text{ に対して } \|A_n - A_m\|_{\mathcal{L}(X, Y)} < \varepsilon$$

を満たす。その上、 $m \rightarrow \infty$ とすると

$$\forall \varepsilon > 0, \exists N \in \mathbb{N}, \forall n \geq N \text{ に対して } \|A_n - \tilde{A}\|_{\mathcal{L}(X, Y)} < \varepsilon$$

となり、 $\tilde{A} \in \mathcal{L}(X, Y)$ は Cauchy 列 (A_n) の極限である。よって、任意の Cauchy 列は $\mathcal{L}(X, Y)$ 内に極限を持つため、ノルム空間 $\mathcal{L}(X, Y)$ は Banach 空間である。 ■

定理 8 ($\mathcal{L}(X, Y)$ ノルムの性質 (1)). X をノルム空間とし、 Y を Banach 空間とする。そのとき、任意の $u \in X$ と任意の $A \in \mathcal{L}(X, Y)$ について以下の不等式が成り立つ。

$$\|Au\|_Y \leq \|A\|_{\mathcal{L}(X, Y)} \|u\|_X$$

証明. $u = 0$ の時は明らかに成り立つため、 $u \in X \setminus \{0\}$ について考える。 $u \in X \setminus \{0\}$ について

$$\|Au\|_Y = \frac{\|Au\|_Y}{\|u\|_X} \|u\|_X \leq \sup_{\phi \in X \setminus \{0\}} \frac{\|A\phi\|_Y}{\|\phi\|_X} \|u\|_X = \|A\|_{\mathcal{L}(X, Y)} \|u\|_X$$

となるため、題意は示された。 ■

定理 9 ($\mathcal{L}(X, Y)$ ノルムの性質 (2)). X をノルム空間とし、 Y と Z を Banach 空間とする。そのとき、任意の $B \in \mathcal{L}(X, Y)$ と $A \in \mathcal{L}(X, Y)$ の合成作用素 AB は $\mathcal{L}(X, Y)$ に属する。その上、

$$\|AB\|_{\mathcal{L}(X, Z)} \leq \|A\|_{\mathcal{L}(Y, Z)} \|B\|_{\mathcal{L}(X, Y)}$$

証明. 合成作用素の定義 22 から

$$\mathcal{D}(AB) = \{v \in \mathcal{D}(B) = X \mid Bv \in \mathcal{D}(A) = Y\}$$

となるが、 $B \in \mathcal{L}(X, Y)$ であるため、任意の $v \in X$ に対して Bv は Y に属する。よって、

$$\mathcal{D}(AB) = \mathcal{D}(B) = X$$

となる。その上、 A も B も線形作用素であることから、任意の $u, v \in X$ と任意の $\alpha, \beta \in \mathbb{K}$ に対して

$$AB(\alpha u + \beta v) = A(B\alpha u + B\beta v) = A(\alpha Bu + \beta Bv) = A\alpha Bu + A\beta Bv = \alpha ABu + \beta ABv$$

となるため、合成作用素 AB は定義域が X 全体となる線形作用素である。よって AB は $\mathcal{L}(X, Y)$ に属する。その上、

$$\begin{aligned} \|AB\|_{\mathcal{L}(X, Z)} &= \sup_{u \in X \setminus \{0\}} \frac{\|ABu\|_Z}{\|u\|_X} \\ &\leq \sup_{u \in X \setminus \{0\}} \frac{\|A\|_{\mathcal{L}(Y, Z)} \|B\|_{\mathcal{L}(X, Y)} \|u\|_X}{\|u\|_X} \|u\|_X \\ &= \|A\|_{\mathcal{L}(Y, Z)} \|B\|_{\mathcal{L}(X, Y)} \end{aligned}$$

■

定義 25 (X 上の恒等作用素). X を Banach 空間とする。任意の $u \in X$ に対して

$$Iu = u$$

となる $I \in \mathcal{L}(X)$ を X 上の恒等作用素と呼ぶ。

定理 10 (Neumann 級数). X を Banach 空間とする。 $B \in \mathcal{L}(X)$ とし、 $I \in \mathcal{L}(X)$ を X 上の恒等作用素とする。もし

$$\|I - B\|_{\mathcal{L}(X)} < 1$$

ならば逆作用素をもち $B^{-1} \in \mathcal{L}(X)$ となる。そのうえ、

$$B^{-1} = I + (I - B) + (I - B)^2 + \cdots = \sum_{i=0}^{\infty} (I - B)^i$$

で、かつ

$$\|B^{-1}\|_{\mathcal{L}(X)} \leq \frac{1}{1 - \|I - B\|_{\mathcal{L}(X)}}$$

証明.

$$S_n = I + (I - B) + (I - B)^2 + \cdots + (I - B)^n$$

とすると、 B と I ともに $\mathcal{L}(X)$ に属するため、加法 $I - B$ や合成作用素 $(I - B)(I - B)$ などとも $\mathcal{L}(X)$ に属する。よって S_n も $\mathcal{L}(X)$ に属する。

続いて点列 $(S_n) \subset \mathcal{L}(X)$ が極限 S を $\mathcal{L}(X)$ 内に持つか確認する。定理 (9) より

$$\|(I - B)^i\|_{\mathcal{L}(X)} \leq \|(I - B)\|_{\mathcal{L}(X)}^i, \quad i = 0, 1, \dots$$

となるため、 $n > m > 0$ となる整数に対して

$$\|S_n - S_m\|_{\mathcal{L}(X)} = \left\| \sum_{i=m+1}^n (I - B)^i \right\|_{\mathcal{L}(X)} \leq \sum_{i=m+1}^n \|(I - B)\|_{\mathcal{L}(X)}^i$$

となる．定理の仮定より $\|(I - B)\|_{\mathcal{L}(X)} < 1$ であるため，

$$\sum_{i=m+1}^n \|(I - B)\|_{\mathcal{L}(X)}^i \rightarrow 0, \quad (n, m \rightarrow \infty)$$

となる．よって

$$\|S_n - S_m\|_{\mathcal{L}(X)} \rightarrow 0, \quad (n, m \rightarrow \infty)$$

となるため，点列 (S_n) は Cauchy 列である．その上， $\mathcal{L}(X)$ は Banach 空間であるため，任意の Cauchy 列は極限 $\mathcal{L}(X)$ に持つため，点列 (S_n) は

$$\|S_n - S\|_{\mathcal{L}(X)} \rightarrow 0, \quad (n \rightarrow \infty)$$

となる極限 $S \in \mathcal{L}(X)$ を持つ．

次に S が B^{-1} になることを示す．合成作用素の定義 22 にしたがって合成作用素 BS_n を考える． X は Banach 空間であり， $S, B_n \in \mathcal{L}(X)$ であるため，定理 9 より合成作用素 BS_n は $\mathcal{L}(X)$ に属する．その上，点列 $(BS_n) \subset \mathcal{L}(X)$ は

$$\|BS_n - BS\|_{\mathcal{L}(X)} \leq \|B\|_{\mathcal{L}(X)} \|S_n - S\|_{\mathcal{L}(X)} \rightarrow 0, \quad (n \rightarrow \infty)$$

となるため，極限 BS を $\mathcal{L}(X)$ 内にもつ．一方で，

$$\begin{aligned} BS_n &= (I - (I - B))S_n \\ &= S_n - (I - B)S_n \\ &= \sum_{i=0}^n (I - B)^i - \sum_{i=1}^{n+1} (I - B)^i \\ &= I - (I - B)^{n+1} \end{aligned}$$

となり，定理の仮定より $\|I - B\|_{\mathcal{L}(X)} < 1$ を持つため

$$\|BS_n - I\|_{\mathcal{L}(X)} = \|(I - B)^{n+1}\|_{\mathcal{L}(X)} \leq \|(I - B)\|_{\mathcal{L}(X)}^{n+1} \rightarrow 0, \quad (n \rightarrow \infty)$$

となるため，点列 (BS_n) は極限 I も $\mathcal{L}(X)$ 内に持つ．よって，極限の一意性より

$$BS = I$$

を得る． $\mathcal{R}(I) = X$ であるため， $\mathcal{R}(BS) = X$ である．その上， $X = \mathcal{R}(BS) \subset \mathcal{R}(B)$ と $\mathcal{R}(B) \subset X$ となるため，

$$\mathcal{D}(S) = \mathcal{R}(B) = X$$

となる．

同様の議論を $S_n B \in \mathcal{L}(X)$ について行くと

$$SB = I$$

と

$$\mathcal{D}(B) = \mathcal{R}(S) = X$$

が得られる．そのため， B は逆作用素を持ち，逆作用素 $B^{-1} = S \in \mathcal{L}(X)$ である．

また，

$$S_n = I + (I - B) + (I - B)^2 + \cdots + (I - B)^n \rightarrow B^{-1}, \quad (n \rightarrow \infty)$$

より

$$B^{-1} = I + (I - B) + (I - B)^2 + \cdots = \sum_{i=0}^{\infty} (I - B)^i$$

となる．

最後に

$$\|B^{-1}\|_{\mathcal{L}(X)} = \left\| \sum_{i=0}^{\infty} (I - B)^i \right\|_{\mathcal{L}(X)} \leq \sum_{i=0}^{\infty} \|(I - B)\|^i$$

となり，初項 1，公比 $\|I - B\|_{\mathcal{L}(X)} < 1$ の総和より

$$\|B^{-1}\|_{\mathcal{L}(X)} \leq \frac{1}{1 - \|I - B\|_{\mathcal{L}(X)}}$$

■

定理 11. X と Y を Banach 空間とする． $A \in \mathcal{L}(X, Y), R \in \mathcal{L}(Y, X)$ とする．もし RA が全単射ならば， A は単射であり， R は全射である．

証明. 「 A が単射」の証明

定理 4(線形作用素に対する単射性 (1)) の (2) を用いて証明する． $u \in X$ とし， RA が単射であることに注意すると

$$Au = 0 \Rightarrow RAu = 0 \Rightarrow u = 0$$

よって， A が全単射である．

「 R が全射」の証明

RA が全射であるため任意の $g \in X$ に対して，

$$RAu = g$$

となる $u \in X$ が存在する．その上， $v = Au$ とすると任意の $g \in X$ に対して

$$Rv = g$$

となる $v \in Y$ が存在するため， R は全射である．

■

定義 26 (Fréchet 微分)．作用素 $F : X \rightarrow Y$ が $x_0 \in X$ で Fréchet 微分可能であるとは，ある有界線形作用素 $E : X \rightarrow Y$ が存在して

$$\lim_{\|h\|_X \rightarrow 0} \frac{\|F(x_0 + h) - F(x_0) - Eh\|_Y}{\|h\|_X} = 0$$

が成り立つことをいう。このとき E は作用素 F の x_0 における Fréchet 微分といい、 $E = DF(x_0)$ とも書く。もしも作用素 $F : X \rightarrow Y$ がすべての $x \in X$ に対して Fréchet 微分可能ならば、 F は X において C^1 -Fréchet 微分可能という。

2.3 Banach の不動点定理

定義 27 (不動点). X を係数体が \mathbb{R} の Banach 空間とする。 M は空でない閉集合で $M \subset X$ を満たすとする。 A を M から M への写像とする。 $x \in M$ が A の不動点であるとは、 x が

$$x = Ax$$

を満たすことである。

定義 28 (距離空間). X をノルム空間とし、 $x, y \in X$ に対して実数値を対応させる関数 $d(\cdot, \cdot) : X \times X \rightarrow \mathbb{R}$ が定義され、

1. $d(x, y) \geq 0$ かつ $d(x, y) = 0 \Leftrightarrow x = y, x, y \in X$
2. $d(x, y) = d(y, x), x, y \in X$
3. $d(x, y) \leq d(x, z) + d(z, y), x, y, z \in X$

を満たすとき、 d を距離空間という。距離の備わった集合を距離空間という。

定義 29 (縮小写像). X を係数体が \mathbb{R} の Banach 空間とする。 M は空でない閉集合で $M \subset X$ を満たすとする。写像 $A : M \rightarrow M$ が k 次の縮小写像であるとは、 $0 \leq k < 1$ を満たす定数 k が存在し、 $\forall x, y \in M$ について

$$\|Ax - Ay\| \leq k\|x - y\|$$

を満たすことである。

定理 12 (Banach の不動点定理). X を係数体が \mathbb{R} の Banach 空間とする。 M は空でない閉集合で $M \subset X$ を満たすとする。 A は M から M への k 次の縮小写像とする。そのとき、問題

$$\text{Find } u \in M \text{ s.t. } u = Au \tag{2.1}$$

は真の解 u^* を M 内にただ一つ持つ。即ち、写像 A は M 上にただ一つ不動点 u^* を持つ。

証明. u_0 を閉集合 M の元として与えられていると仮定する。点列 (u_n) は反復法

$$u_{n+1} = Au_n, n = 0, 1, \dots \tag{2.2}$$

によって得られる。そのとき、証明のプロセスは次のように考える。

1. (u_n) が Cauchy 列になること、さらに Banach 空間の完備性を使うことで、 $u_n \rightarrow u, n \rightarrow \infty$ となる u が X 内に存在することを示す。

2. u が (2.1) を満たす真の解 u^* と一致することを示す (解の存在性).
3. 真の解 u^* が M 内で一意であることを示す.

1

(2.2) より,

$$\|u_n - u_{n+1}\| = \|Au_{n-1} - Au_n\|$$

となる. 定理の仮定より A は k 次の縮小写像であるため,

$$\|Au_{n-1} - Au_n\| \leq k\|u_{n-1} - u_n\|$$

となる定数 k が存在する. 同様に $\|u_{n-1} - u_n\|$ に (2.2) と A の縮小写像の性質を使うと最終的に

$$\|u_n - u_{n+1}\| \leq k^n \|u_0 - u_1\| \quad (2.3)$$

を得る.

次に三角不等式より, $n = 0, 1, 2, \dots, m > n$ について

$$\begin{aligned} \|u_n - u_m\| &= \|(u_n - u_{n+1}) + (u_{n+1} - u_{n+2}) + \dots + (u_{m-1} - u_m)\| \\ &\leq \|u_n - u_{n+1}\| + \|u_{n+1} - u_{n+2}\| + \dots + \|u_{m-1} - u_m\| \end{aligned}$$

となる. 上の式に (2.3) を適用すると

$$\begin{aligned} \|u_n - u_m\| &\leq \|u_n - u_{n+1}\| + \|u_{n+1} - u_{n+2}\| + \dots + \|u_{m-1} - u_m\| \\ &\leq k^n (\|u_n - u_{n+1}\| + k^{n+1} \|u_{n+1} - u_{n+2}\| + \dots + k^{m-1} \|u_{m-1} - u_m\|) \\ &= k^n (1 + k + \dots + k^{m-n-1}) \|u_0 - u_1\| \end{aligned} \quad (2.4)$$

となる. ここで, k は $0 < k < 1$ であるため, $1 + k + \dots + k^{m-n-1} \leq 1 + k + \dots + k^{m-1}$ となる. さらに, 等比級数から

$$1 + k + \dots + k^{m-1} = \frac{1 - k^m}{1 - k}$$

であるため, (2.4) は

$$\|u_n - u_m\| \leq \frac{k^n (1 - k^m)}{1 - k} \|u_0 - u_1\|$$

となる. よって, k は $0 < k < 1$ から $k^n \rightarrow 0, n \rightarrow \infty$ と $k^m \rightarrow 0, m \rightarrow \infty$ となる. すなわち,

$$\|u_n - u_m\| \rightarrow 0, (n, m \rightarrow \infty)$$

となるため, 点列 (u_n) は Cauchy 列である. さらに X は Banach 空間であるため, X は完備である. すなわち, 任意の Cauchy 列が X の中で極限を持つ. よって, 点列 (u_n) は

$$u_n \rightarrow u, n \rightarrow \infty$$

となる $u \in X$ が存在する.

2

u_0 を M の元とする．仮定より, A は M から M への写像であるため, $A(M) \subseteq M$ となる．すなわち, $u_1 = Au_0$ が成立する $u_1 \in M$ が存在する．同様に, $\forall n \in \mathbb{N}$ について $u_n \in M$ が存在する．さらに, M は閉集合であるため, **1** で存在を証明した u は M に属する．そのうえで仮定より A は k 次の縮小写像であるため

$$\|Au_n - Au\| \leq k\|u_n - u\|$$

を得る．**1** より点列 (u_n) は極限をもつため, $\|u_n - u\| \rightarrow 0, n \rightarrow \infty$ となる．すなわち

$$\|Au_n - Au\| \rightarrow 0, n \rightarrow \infty$$

となるため, Au は Au_n の極限である．よって (2.2) について $n \rightarrow \infty$

$$u = Au$$

が成立する．とって u は (2.1) を満たす真の解 u^* となる．

3

$u^*, v^* \in M$ をそれぞれ $u^* = Au^*$ と $v^* = Av^*$ を満たすとする．その時, A は k 次の縮小写像であるため

$$\|u^* - v^*\| \leq \|Au^* - Av^*\| \leq k\|u^* - v^*\|$$

を得る．ここで $0 \leq k < 1$ であるため, 不等式を満たすものは $u^* = v^*$ の場合のみである．すなわち (2.1) を満たす真の解は一意である．

■

2.4 簡易ニュートン写像

本節では, X, Y を Banach 空間とし, 写像 $F : X \rightarrow Y$ に対して

$$F(x) = 0 \in Y$$

という (非線形) 作用素方程式を考える．

このとき, 写像 $T : X \rightarrow X$ を

$$T(x) := x - AF(x)$$

で定義する．これを簡易ニュートン写像という．ここで $A : Y \rightarrow X$ はある全単射な線形作用素である．いま \bar{x} を $F(\bar{x}) \approx 0$ となる近似解とし, \bar{x} の近傍を

$$B(\bar{x}, r) := \{x \in X \mid \|x - \bar{x}\| < r\} \text{ (開球)}$$

$$\overline{B(\bar{x}, r)} := \{x \in X \mid \|x - \bar{x}\| \leq r\} \text{ (閉球)}$$

で定義する．このとき, もし, $B(\bar{x}, r)$ 上で写像 T が縮小写像となれば, Banach の不動点定理から $F(\tilde{x}) = 0$ をみたす解 $\tilde{x} \in B(\bar{x}, r)$ がただ一つ存在することになる．このように解の存在を仮定せずに近似解近傍での収束という定理を Newton-Kantorovich の定理という．

2.5 フーリエ級数

2.5.1 フーリエ級数の導出

ある関数 $f(x)$ が有限の閉空間 $[a, b]$ で定義されている場合に、次の性質をもつとき区分的に連続であるという。

1. $f(x)$ は有限個の不連続点を除いて連続である。
2. $f(x)$ の不連続点 c では $f(c+0)$ と $f(c-0)$ が存在する。

また、 $f(x)$ が無限区間で定義されている場合には、 $f(x)$ が任意の有限区間で区分的に連続であるときに、 $f(x)$ は区分的に連続であるという。以降は、特に断らないかぎり区分的に連続な関数のみを考えることにする。

関数 $f(x)$, ($x \in [-\pi, \pi]$) を周期 2π の周期関数とする。このような $f(x)$ が以下の三角関数で表されるとして、係数 $(a_0, a_1, a_2, \dots, b_0, b_1, b_2, \dots)$ を求めることを考える。

$$\begin{aligned} f(x) &= \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \\ &= \frac{a_0}{2} + (a_1 \cos x + b_1 \sin x) + (a_2 \cos 2x + b_2 \sin 2x) + \dots \end{aligned} \quad (2.5)$$

そのために、次の公式を用意する。ここで、 $m, n \in \mathbb{N}$ である。

$$\begin{aligned} \int_{-\pi}^{\pi} \cos nx \cos mx dx &= \begin{cases} 0 & (n \neq m) \\ \pi & (n = m) \end{cases} \\ \int_{-\pi}^{\pi} \sin nx \sin mx dx &= \begin{cases} 0 & (n \neq m) \\ \pi & (n = m) \end{cases} \\ \int_{-\pi}^{\pi} \sin nx \cos mx dx &= 0 \end{aligned} \quad (2.6)$$

$$\int_{-\pi}^{\pi} \sin nx dx = \int_{-\pi}^{\pi} \cos mx dx = 0 \quad (2.7)$$

ここで、次のような形式的計算が許されると仮定する。まず、式 (2.5) の両辺について $-\pi$ から π まで積分し、式 (2.7) を用いれば

$$\begin{aligned} \int_{-\pi}^{\pi} f(x) dx &= \frac{a_0}{2} \int_{-\pi}^{\pi} dx + \sum_{n=1}^{\infty} (a_n \int_{-\pi}^{\pi} \cos nx dx + \int_{-\pi}^{\pi} b_n \sin nx dx) \\ &= \pi a_0 \\ \therefore a_0 &= \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) dx \end{aligned} \quad (2.8)$$

次に、式 (2.5) の両辺に $\cos mx$ をかけて、 $-\pi$ から π まで積分し、式 (2.6),(2.7) を用いれば

$$\begin{aligned}\int_{-\pi}^{\pi} f(x) \cos mx dx &= \frac{a_0}{2} \int_{-\pi}^{\pi} \cos mx dx \\ &\quad + \sum_{n=1}^{\infty} (a_n \int_{-\pi}^{\pi} \cos nx \cos mx dx + \int_{-\pi}^{\pi} b_n \sin nx \cos mx dx) \\ &= \pi a_0 \\ \therefore a_m &= \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos mx dx\end{aligned}\tag{2.9}$$

同様に、式 (2.5) の両辺に $\sin mx$ をかけて、 $-\pi$ から π まで積分することにより次の式が得られる。

$$b_m = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin mx dx\tag{2.10}$$

式 (2.8), 式 (2.9), 式 (2.10) をまとめて次のようになる。

定義 30 (フーリエ級数).

$$f(x) \sim \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos nx + \sum_{n=1}^{\infty} b_n \sin nx$$

ただし

$$\begin{aligned}a_n &= \frac{1}{\pi} \int_0^{2\pi} f(x) \cos nx dx, \quad (n \geq 0) \\ a_n &= \frac{1}{\pi} \int_0^{2\pi} f(x) \cos nx dx, \quad (n \geq 1)\end{aligned}$$

この無限級数を f のフーリエ級数といい、 a_n, b_n をフーリエ係数という。

定義 31 (複素フーリエ変換). また,

$$\cos nx = \frac{\exp(inx) + \exp(-inx)}{2}, \quad \sin nx = \frac{\exp(inx) - \exp(-inx)}{2i} \quad (i \text{ は虚数単位})$$

という関係を用いて

$$\begin{aligned}f(x) &= \sum_{k=-\infty}^{\infty} c_k \exp(ikx) \\ c_k &= \frac{1}{2\pi} \int_0^{2\pi} f(x) \exp(-ikx) dx\end{aligned}$$

と複素数を用いた形式も考えられる。これを f の複素フーリエ級数、 c_k を複素フーリエ係数という。

これらには関係式

$$c_k = \begin{cases} \frac{a_k}{2} & , k = 0 \\ \frac{a_k - ib_k}{2} & , k > 0 \\ \frac{a_{-k} - ib_{-k}}{2} & , k < 0 \end{cases}$$

があり，変換可能である．

2.5.2 フーリエ級数の性質

対称性

周期関数 $f(x)$ が偶関数であるとき，偶関数のフーリエ級数は

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos nx$$

と表すことができる．上式のように \cos 関数のみで表されるフーリエ級数をフーリエ・コサイン級数という．このとき，複素フーリエ係数は $c_{-k} = -c_k$ が成り立つ．

一方で， $f(x)$ が奇関数であるとき，奇関数のフーリエ級数は

$$f(x) = \sum_{n=1}^{\infty} b_n \sin nx$$

と表すことができる．上式のように \sin 関数のみで表されるフーリエ級数をフーリエ・サイン級数という．このときも，複素フーリエ係数は $c_{-k} = -c_k$ が成り立つ．

実数値関数

$f(x)$ が実数値関数 $f(x) \in \mathbb{R}$ であるとき，各フーリエ係数 a_n, b_n は

$$a_n, b_n \in \mathbb{R}$$

となる．このとき，複素フーリエ級数 c_k は

$$c_{-k} = \overline{c_k}$$

を満たす．これは， $f(x) = \overline{f(x)}$ という実数値関数の性質を使うことで確認できる．

係数の収束

ある周期関数 $f(x)$ のフーリエ係数を a_n として， $n \rightarrow \infty$ での収束オーダーを考えると，

$$a_n = \begin{cases} \mathcal{O}(n^{-K}) & k \text{ 次オーダーの収束} \\ \mathcal{O}(e^{-qn^r}) & k \text{ 次オーダーの収束} \\ \mathcal{O}(e^{-qb \log(n)}) & \text{超幾何収束} \end{cases}$$

などのパターンがあり，それぞれ $f(x)$ の滑らかさによって決まる．例えば， $f(x)$ が k 次オーダーの収束をする場合は，関数 f は C^k - 級の関数である．指数オーダーの収束の場

合は実解析関数（曲や分岐点を持つ一般的な有限区間・無限区間上の関数），超幾何収束の場合は複素平面上で ∞ 以外で特異点を持たない関数（整関数）とそれぞれ知られている。

ランダウの記号 \mathcal{O} は，例えば， $a_n = \mathcal{O}(n^{-2})$ ($n \rightarrow \infty$) は，左辺の絶対値が右辺の絶対値の定数倍以下であることを表す．すなわち，ある定数 C があって， n が十分大きいすべての $n \geq 0$ に対して， $|a_n| \leq Cn^{-2}$ が成立することを意味する．

微分とシフト

フーリエ変換の便利な性質として微分およびシフト（平行移動）がある．

いま， $f(x)$ を複素フーリエ級数とする微分操作は

$$\frac{d}{dx}f(x) \sim \sum_{k=-\infty}^{\infty} (ik)c_k \exp(ikx)$$

と表すことができる．

同様に，シフト操作も

$$\frac{d}{dx}f(x) \sim \sum_{k=-\infty}^{\infty} (ik)c_k \exp(ikx)$$

することでシフト d だけ x 方向に平行移動できる．以降，簡単のために特に混合がなければ，複素フーリエ級数も単にフーリエ級数と呼ぶ．

2.5.3 フーリエ変換の数値計算

与えられた関数のフーリエ級数を計算する

周期関数 $f(x)$ のフーリエ係数 c_k を数値計算で求める．無限級数を計算機で表現することは難しいため，係数を打ち切る操作が必要である．この時のフーリエ係数のサイズ N を $|k| < N$ となるように定める（ $N-1$ を最大端数という）．

いま， $0 = x_0 \leq x_1 \leq \dots \leq x_{2N-1} = 2\pi$ と区間 $[0, 2\pi]$ を等間隔に分割した点 $x_j = jh$ ($j = 0, \dots, 2N-1$)， $h = 2\pi/(2N-1)$ を標本点といい，標本点上での関数値を用いてフーリエ係数の近似 $\overline{c_k}$ を得る．

$$c_k = \frac{1}{2\pi} \int_0^{2\pi} f(x) \exp(-kx) dx \approx \frac{1}{2N-1} \sum_{j=0}^{2N-2} f(x_j) \exp\left(-2\pi i \frac{kj}{2N-1}\right) = \overline{c_k}, (|k| < N)$$

この $\overline{c_k}$ の式は，離散フーリエ変換 $a_k = \mathcal{F}(b) = \sum_{b_j}^{2N-2} f(x_j) \exp\left(-2\pi i \frac{kj}{2N-1}\right)$ そのものである．したがって，高速フーリエ変換（FFT）を用いて近似フーリエ係数 $\overline{c_k}$ を計算できる．そして， $\overline{c_k}$ を使って，元の関数 $f(x)$ の近似 $f^{(N)}(x)$ が

$$f(x) \approx f^{(N)}(x) = \sum_{|k| < N} \overline{c_k} \exp(ikx)$$

と得られる．

フーリエ係数から元の関数の概形を求める

近似関数 $f^{(N)}(x)$ の係数 c_k から関数の概形を求める.

いま, 標本点上の関数値は

$$f^{(N)}(x_j) = \sum_{|k| < N} \bar{c}_k \exp(ikx_j) = \sum_{|k| < N} \bar{c}_k \exp\left(2\pi i \frac{kj}{2N-1}\right)$$

これは逆フーリエ変換 (IFFT) に相当する. そこで, IFFT を用いて元の関数の外形を求める. しかし, このまま IFFT を用いると標本点と同数の関数値しか得られないため, グラフに描画すると粗い概形となる. そこで, フーリエ係数 \bar{c}_k に 0 を余分に張り合わせて, なめらかなグラフのプロットを得る.

周期が 2π 以外の場合の計算

周期が 2π 以外の場合も, ほぼ同様な考え方でフーリエ級数を考えられる.

いま, $f(x)$ を周期 L の周期関数とする. このとき, 変数 $t: a \rightarrow b$, ($L = b - a$) に対して, 変数 x を $x = \omega(t - a)$, ($\omega = 2\pi/L$) と定めると, $x: 0 \rightarrow 2\pi$ となり, 関数 $g(x) = f(a + \omega^{-1}x)$ は周期 2π の周期関数である. そこで $g(x)$ がフーリエ級数 $g(x)$ がフーリエ級数

$$g(x) = \sum_{k \in \mathbb{Z}} c_k \exp(ikx)$$

で表されたとすると, $f(x)$ のフーリエ級数は

$$f(t) = g(\omega(t - a)) = \sum_{k \in \mathbb{Z}} c_k \exp(ikx)$$

が成り立つ. フーリエ係数は $dx = \omega dt$ より

$$\begin{aligned} c_k &= \frac{1}{2\pi} \int_0^{2\pi} g(x) \exp(-ikx) dx \\ &= \frac{1}{2\pi} \int_0^{2\pi} g(\omega(t - a)) \exp(-ik\omega(t - a)) \omega dx \\ &= \frac{1}{L} \int_0^{2\pi} f(t) \exp(-ik\omega(t - a)) dt \quad (k \in \mathbb{Z}) \end{aligned}$$

と f の離散フーリエ変換で求められる. したがって, サイズ N のフーリエ係数は標本点を $a = t_0 \leq t_1 \leq \dots \leq t_{2N-1} = b$ と区間 $[a, b]$ を等間隔に分割した点 $t_j = a + jh$ ($j = 0, \dots, 2N-1, h = L/(2N-1)$) でとり, この標本点上での関数値を用いてフーリエ係数の近似 \bar{c}_k を得る.

$$\begin{aligned} c_k &= \frac{1}{L} \int_a^b f(t) \exp(-ik\omega(t - a)) dt \\ &\approx \frac{1}{2N-1} \sum_{j=0}^{2N-2} f(t_j) \exp\left(-2\pi i \frac{kj}{2N-1}\right) = \bar{c}_k, \quad (|k| < N) \end{aligned}$$

ここで, $t_j = a + \frac{jL}{2N-1} (j = 0, 1, \dots, 2N-1)$ から, 周期が 2π の周期関数のフーリエ係数の近似が同じ式になるため, フーリエ係数の計算方法は変わらない.

2.6 離散フーリエ変換

定義 32 (離散フーリエ変換). $b = (b_0, b_1, \dots, b_{2M-2}) \in \mathbb{C}^{2M-1}$ に対して, $a = \mathcal{F}(b) \in \mathbb{C}^{2M-1}$ を

$$a_k = \mathcal{F}_k(b) := \sum_{j=0}^{2M-2} b_j \exp\left(2\pi i \frac{kj}{2M-1}\right), \quad (k = -M+1, \dots, M-1)$$

とし, 離散フーリエ変換 (DFT) と呼ぶ.

定義 33 (逆離散フーリエ変換). $a = (a_k)_{|k|<M} = (a_{-M+1}, \dots, a_{M-1}) \in \mathbb{C}^{2M-1}$ に対して, $b = \mathcal{F}^{-1}(a) \in \mathbb{C}^{2M-1}$ を

$$b_j = \mathcal{F}_j^{-1}(a) := \sum_{k=-M+1}^{M-1} a_k \exp\left(2\pi i \frac{kj}{2M-1}\right), \quad (j = 0, \dots, 2M-2)$$

とし, 逆離散フーリエ変換 (IDFT) と呼ぶ.

2.7 畳み込みの FFT アルゴリズム

2.7.1 畳み込み定理

u_1, u_2 : 周期 L の周期関数とする.

$$u_1(t) = \sum_{k \in \mathbb{Z}} a_k^{(1)} \exp(ik\omega t), \quad a^{(1)} = (a_k^{(1)})_{k \in \mathbb{Z}}$$

$$u_2(t) = \sum_{k \in \mathbb{Z}} a_k^{(2)} \exp(ik\omega t), \quad a^{(2)} = (a_k^{(2)})_{k \in \mathbb{Z}}$$

とする. ただし, $\omega = 2\pi/L$ としたとする. このとき, 関数の積 $u_1(t)u_2(t)$ は次のようにフーリエ級数で表せる.

$$u_1(t)u_2(t) = \sum_{k \in \mathbb{Z}} \left(a^{(1)} * a^{(2)}\right)_k \exp(ik\omega t)$$

ここで $a = (a_k)_{k \in \mathbb{Z}}, b = (b_k)_{k \in \mathbb{Z}}$ に対して,

$$(a * b)_k = \sum_{\substack{k_1+k_2=k \\ k_1, k_2 \in \mathbb{Z}}} a_{k_1} b_{k_2}, \quad k \in \mathbb{Z}$$

を離散畳み込みという. スペクトル法の非線形項の計算, i.e., u_1, u_2 のような (有限モードのフーリエ級数で表される) 周期関数が p 個 ($p \in \mathbb{N}$) あったとき,

$$u_l(t) = \sum_{|k|<M} a_k^{(l)} \exp(ik\omega t), \quad (l = 1, 2, \dots, p)$$

とすると、離散畳み込みはこれらの周期関数の積

$$\begin{aligned} u_1(t)u_2(t)\cdots u_p(t) &= \sum_{|k|<M} \left(a^{(1)} * a^{(2)} * \cdots * a^{(p)} \right)_k \exp(ik\omega t) \\ &= \sum_{|k|\leq p(M-1)} \left(a^{(1)} * a^{(2)} * \cdots * a^{(p)} \right)_k \exp(ik\omega t) \end{aligned}$$

を表すことになる。ここで、

$$\left(a^{(1)} * \cdots * a^{(p)} \right)_k = \sum_{\substack{k_1+\cdots+k_p=k \\ |k|\leq p(M-1) \\ |k_1|,\dots,|k_p|<M}} a_{k_1}^{(1)} \cdots a_{k_p}^{(p)}$$

と表される。

畳み込み定理

畳み込みをフーリエ変換下もには、それぞれのフーリエ変換の積になる。これを畳み込み定理という。

$$\begin{aligned} \mathcal{F} \left(a^{(1)} * a^{(2)} * \cdots * a^{(p)} \right) &= \mathcal{F} \left(a^{(1)} \right) \mathcal{F} \left(a^{(2)} \right) \cdots \mathcal{F} \left(a^{(p)} \right) \\ &= b^{(1)} b^{(2)} \cdots b^{(p)} \end{aligned}$$

ただし、ベクトルの積は要素ごとの積を表す。

2.7.2 畳み込みの FFT アルゴリズム

p 個の有限点列 $\left\{ a_k^{(l)} \right\}_{|k|<M}$ に対して、FFT を使って

$$\left(a^{(1)} * a^{(2)} * \cdots * a^{(p)} \right)_k = \sum_{\substack{k_1+\cdots+k_p=k \\ |k_l|<M}} a_{k_1}^{(1)} \cdots a_{k_p}^{(p)}, \quad |k| \leq p(M-1)$$

を計算する。

入力：

$$a^{(l)} = \left(a_k^{(l)} \right)_{|k|<M} \in \mathbb{C}^{2M-1}, \quad (l = 1, \dots, p)$$

Step1：

エイリアシングエラーを防ぐために、入力された値 $a^{(l)}$ の両脇に $(p-1)M$ 個の 0 を追加する。これを $\tilde{a}^{(l)}$ とする。

$$\tilde{a}^{(l)} = \begin{cases} a_j^{(l)}, & |j| < M \\ 0, & M \leq |j| \leq pM-1 \end{cases}$$

という条件で $\tilde{a} \in \mathbb{C}^{2pM-1}$ を作る.

$$\tilde{a}_j^{(l)} = (\underbrace{0, \dots, 0}_{p(M-1)}, \underbrace{a_{-M+1}^{(l)}, \dots, a_{M-1}^{(l)}}_{2M-1}, \underbrace{0, \dots, 0}_{p(M-1)}) \in \mathbb{C}^{2pM-1}$$

Step2:

$\tilde{b}^{(l)} = (\tilde{b}_j^{(l)})_{j=0, \dots, 2pM-2}$ を $\tilde{a}^{(l)}$ の逆離散フーリエ変換を行い,

$$\tilde{b}_j^{(l)} := \mathcal{F}_j^{-1}(\tilde{a}^{(l)}) = \sum_{k=-pM+1}^{pM-1} \tilde{a}_k^{(l)} \exp\left(2\pi i \frac{kj}{2pM-1}\right)$$

で定める.

Step3:

ベクトルの要素毎の積を計算する.

$$(\tilde{b}^{(1)} \hat{*} \dots \hat{*} \tilde{b}^{(p)})_j := \tilde{b}_j^{(1)} \dots \tilde{b}_j^{(p)}, \quad j = 0, \dots, 2pM-2$$

Step4:

Step3 で計算した $(\tilde{b}^{(1)} \hat{*} \dots \hat{*} \tilde{b}^{(p)})$ の逆離散フーリエ変換を行うと, $k = -pM+1, \dots, pM-1$ で,

$$\begin{aligned} & \mathcal{F}_k(\tilde{b}^{(1)} \hat{*} \dots \hat{*} \tilde{b}^{(p)}) \\ &= \sum_{j=0}^{2pM-2} \tilde{b}_j^{(1)} \hat{*} \dots \hat{*} \tilde{b}_j^{(p)} \exp\left(-2\pi i \frac{kj}{2pM-1}\right) \\ &= \sum_{j=0}^{2pM-2} \prod_{l=1}^p \left(\sum_{k_l=-pM+1}^{pM-1} \tilde{a}_{k_l}^{(l)} \exp\left(2\pi i \frac{k_l j}{2pM-1}\right) \right) \exp\left(2\pi i \frac{kj}{2pM-1}\right) \\ & \qquad \qquad \qquad \underbrace{\hspace{10em}}_{:= S_k(j)} \end{aligned}$$

いま

$$\begin{aligned} S_k(j) &:= \prod_{l=1}^p \left(\sum_{k_l=-pM+1}^{pM-1} \tilde{a}_{k_l}^{(l)} \exp\left(2\pi i \frac{k_l j}{2pM-1}\right) \right) \exp\left(2\pi i \frac{kj}{2pM-1}\right) \\ &= \underbrace{\sum_{\substack{k_1+\dots+k_p=k \\ |k_l|<M}} a_{k_1}^{(1)} \dots a_{k_p}^{(p)}}_{k_1+\dots+k_p-k \equiv 0 \pmod{pM-1}, \exp(\pi i j) \equiv 1} + \sum_{m=1}^p \left(\sum_{\substack{k_1+\dots+k_p=k \pm m(2pM-1) \\ |k_l|<M}} a_{k_1}^{(1)} \dots a_{k_p}^{(p)} \right) \\ &+ \underbrace{\sum_{\substack{k_1+\dots+k_p \notin \{k \pm m(2pM-1)m=0, \dots, p\} \\ |k_l|<M}} a_{k_1}^{(1)} \dots a_{k_p}^{(p)} \exp\left(2\pi i \frac{k_1+\dots+k_p-k}{2pM-1} j\right)}_{k_1+\dots+k_p-k \not\equiv 0 \pmod{2pM-1}} \end{aligned}$$

として,

$$\begin{aligned}
\mathcal{F}\left(\tilde{b}^{(1)} \hat{*} \cdots \hat{*} \tilde{b}^{(p)}\right) &= \sum_{j=0}^{2pM-2} S_k(j) \\
&= (2pM-1) \sum_{\substack{k_1+\cdots+k_p=k \\ |k_l|<M}} a_{k_1}^{(1)} \cdots a_{k_p}^{(p)} \\
&\quad + (2pM-1) \sum_{m=1}^p \left(\sum_{\substack{k_1+\cdots+k_p=k \pm m(2pM-1) \\ |k_l|<M}} a_{k_1}^{(1)} \cdots a_{k_p}^{(p)} \right) \\
&\quad + \sum_{\substack{(k_1+\cdots+k_p) \\ \notin \{k \pm m(2pM-1): m=0, \dots, p\} \\ |k_l|<M}} a_{k_1}^{(1)} \cdots a_{k_p}^{(p)} \left(\sum_{j=0}^{2pM-2} \exp\left(2\pi i \frac{k_1+\cdots+k_p-k}{2pM-1} j\right) \right)
\end{aligned}$$

このとき, オイラーの公式より, $k_1 + \cdots + k_p - k \equiv 0 \pmod{2pM-1}$ のとき

$$\sum_{j=0}^{2pM-2} \exp(2\pi i \frac{k_1 + \cdots + k_p - k}{2pM-1} j) = 0$$

そして, $|k_1|, \dots, |k_p| < M$ と $|k| \leq p(M-1)$ (最高次) から

$$k_1 + \cdots + k_p - k \in \{-2pM-1, \dots, 2p(M-1)\}$$

となり, $k_1 + \cdots + k_p = k \pm m(2pM-1)$, ($m = 1, \dots, p$) から

$$m(2pM-1) = k_1 + \cdots + k_p - k \in \{-2p(M-1), \dots, 2p(M-1)\}$$

となる. しかし, これはありえない. なぜなら

$$2pM-1 > 2p(M-1)$$

なので, $M = 1$ としても, 上の式は偽となる. すなわち, 0 となる. 最終的に,

$$\sum_{\substack{k_1+\cdots+k_p=k \\ |k_m|<M}} a_{k_1}^{(1)} \cdots a_{k_p}^{(p)} = \frac{1}{2pM-1} \mathcal{F}_k\left(\tilde{b}^{(1)} \hat{*} \cdots \hat{*} \tilde{b}^{(p)}\right), \quad |k| \leq p(M-1)$$

を得る.

まとめると, FFT アルゴリズムは以下になる.

入力： $a^{(m)} = (a^{(m)_k})_{|k| < M} \quad (m = 1, \dots, p)$ (フーリエ・チェビシェフ級数)

$$1. \quad \tilde{a}^{(m)_j} = \left(\underbrace{0, \dots, 0}_{(p-1)M}, \underbrace{a_{-M+1}^{(m)}, \dots, a_{M-1}^{(m)}}_{2M-1}, \underbrace{0, \dots, 0}_{(p-1)M} \right) \in \mathbb{C}^{2pM-1}$$

$$2. \quad \tilde{b}^{(m)} = \mathcal{F}^{-1}(\tilde{a}^{(m)}) \in \mathbb{C}^{2pM-1} \quad (\text{IFFT を用いる})$$

$$3. \quad \left(\tilde{b}^{(1)} \hat{*} \dots \hat{*} \tilde{b}^{(p)} \right) = \tilde{b}^{(1)} \dots \tilde{b}^{(p)} \quad (j = 0, \dots, 2pM - 2)$$

$$4. \quad c_k = \frac{1}{2pM-1} \mathcal{F}_k \left(\tilde{b}^{(1)} \hat{*} \dots \hat{*} \tilde{b}^{(p)} \right), \quad (|k| \leq p(M-1))$$

出力： $c = (c_k)_{|k| \leq p(M-1)} \in \mathbb{C}^{2p(M-1)+1}$

第 3 章

Newton Kantorovich の定理を用いた精度保証付き数値計算

X, Y を Banach 空間とし, 非線形方程式

$$F(u) = 0, u \in X$$

を求める問題を考える. この問題に対して近似解の品質保証を行う, 定理の 1 つを以下に説明する.

3.1 radii polynomial approach [2]

定理 13 (Newton-Kantorovich type argument). 有界線形作用素 $A^\dagger \in \mathcal{L}(X, Y)$, $A \in \mathcal{L}(Y, X)$ を考え, 作用素 $F : X \rightarrow Y$ が C^1 -Fréchet 微分可能とする. また, A が単射で $AF : X \rightarrow X$ とする. いま, $\bar{x} \in X$ に対して

$$\begin{aligned} \|AF(\bar{x})\|_X &\leq Y_0 \\ \|I - A^\dagger\|_{\mathcal{L}(X)} &\leq Z_0 \\ \|A(DF(\bar{x}) - A^\dagger)\|_{\mathcal{L}(X)} &\leq Z_1 \\ \|A(DF(b) - A^\dagger)\|_{\mathcal{L}(X)} &\leq Z_2(r)r, \forall b \in \overline{B(\bar{x}, r)} \end{aligned}$$

が成り立つとする. このとき

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

を radii polynomial といい, もし $p(r_0) < 0$ となる $r_0 > 0$ が存在すれば, $F(\tilde{x}) = 0$ をみたす \tilde{x} が $\overline{B(\tilde{x}, r_0)}$ 内に一意存在する.

証明. 写像 T が閉球 $\overline{B(\bar{x}, r_0)}$ において縮小写像になることを示す.

まず,

$$DT(x) = I - ADF(x), \quad x \in X$$

から, 任意の $x \in \overline{B(\bar{x}, r_0)}$ に対して, 仮定より

$$\begin{aligned} \|DT(x)\|_{\mathcal{L}(X)} &= \|I - ADF(x)\|_{\mathcal{L}(X)} \\ &\leq \|I - AA^\dagger\|_{\mathcal{L}(X)} + \|A(A^\dagger - DF(\bar{x}))\|_{\mathcal{L}(X)} + \|A(DF(\bar{x}) - DF(x))\|_{\mathcal{L}(X)} \\ &\leq Z_0 + Z_1 + Z_2(r_0)r_0 \end{aligned}$$

■

そして, 任意の $x \in \overline{B(\bar{x}, r_0)}$ に対して, 平均値の定理より,

$$\begin{aligned} \|T(x) - \bar{x}\|_X &= \|T(x) - T(\bar{x})\|_X + \|T(\bar{x}) - \bar{x}\|_X \\ &\leq \sup_{b \in B(\bar{x}, r_0)} \|DT(b)\|_{\mathcal{L}(X)} \|x - \bar{x}\|_X + \|AF(\bar{x})\|_X \\ &\leq (Z_0 + Z_1 + Z_2(r_0)r_0)r_0 + Y_0 \\ &= p(r_0) + r_0 \end{aligned}$$

よって, $p(r_0) < 0$ ならば, $\|T(x) - \bar{x}\|_X < r_0$. よって, $T(x) \in \overline{B(\bar{x}, r_0)}$.

次に, X の距離として, $d(x, y) := \|x - y\|_X$, $x, y \in X$ と定義すると,

$$\begin{aligned} d(T(x), T(y)) &\leq \sup_{b \in B(\bar{x}, r_0)} \|DT(b)\|_{\mathcal{L}(X)} d(x, y) \\ &\leq (Z_0 + Z_1 + Z_2(r_0)r_0)d(x, y), \quad x, y \in \overline{B(\bar{x}, r_0)} \end{aligned}$$

このとき, $p(r_0) < 0$ より,

$$Z_0 + Z_1 + Z_2 r_0 r_0 + \frac{Y_0}{r_0} < 1$$

よって, $k := Z_0 + Z_1 + Z_2(r_0)r_0 < 1$ となるため, T は $\overline{B(\bar{x}, r_0)}$ において縮小写像になる.

したがって, Banach の不動点定理より, ただ 1 つの不動点 \tilde{x} が $\overline{B(\bar{x}, r_0)}$ 内に存在し, A の単射性により, この不動点は写像 F の零点となる, i.e., $F(\tilde{x}) = 0$

3.2 Newton-Kantorovich の定理の特徴 [3]

定理 14 (Newton-Kantorovich の定理の亜種). X と Y を Banach 空間とする. $F : X \rightarrow Y$ を与えられた作用素とし, $\bar{x} \in X$ を与えられているとする. $A \in \mathcal{L}(Y, X)$ とする. F は \bar{x} で Fréchet 微分可能とし, $DF(\bar{x})$ と表記する. $DF(\bar{x})$ は全射であるとする. η, δ を不等式

$$\begin{aligned} \|AF(\bar{x})\|_X &\leq \eta \\ \|I - A(DF(\bar{x}))\|_{\mathcal{L}(X)} &\leq \delta < 1 \end{aligned}$$

を満たす定数とする.

$\overline{B(0, \frac{2\eta}{1-\delta})} = \{z \in X \mid \|z\|_X \leq \frac{2\eta}{1-\delta}\}$ とし、定数 K を不等式

$$\|A(DF(\bar{x}) - DF(\bar{x} + z))\|_{\mathcal{L}(X)} \leq K, \quad z \in \overline{B\left(0, \frac{2\eta}{1-\delta}\right)}$$

を満たす定数とする。もし、 $2K + \delta \leq 1$ ならば、

$$\|\tilde{x} - \bar{x}\|_X \leq \frac{\eta}{1 - (K + \delta)} := p$$

に対し、真の解 \tilde{x} は $\bar{B}(\bar{x}, p)$ 内に存在する。その上、 $\overline{B\left(\bar{x}, \frac{2\eta}{1-\delta}\right)}$ 内で一意である。

証明. まず、 A と $DF(\bar{x})$ が全単射であることを示す。 $DF(\bar{x})$ は F の \bar{x} における Fréchet 微分であるから、 $DF(\bar{x})$ は $\mathcal{L}(X, Y)$ に属する。さらに、Neumann 級数の定理 (定理 10) と仮定 $\|I - ADF(\bar{x})\|_{\mathcal{L}(X)} \leq \delta < 1$ より、 $ADF(\bar{x})$ は全単射である。その上、定理 11 より、 $DF(\bar{x})$ は単射であり、 A は全射である。また、定理の仮定より、 $DF(\bar{x})$ は全単射となるため、逆作用素 $DF(\bar{x})^{-1}$ が存在し、 $\mathcal{L}(Y, X)$ に属する。また A の単射性について、定理 4 と逆作用素 $DF(\bar{x})^{-1}$ を用いて

$$A\phi = 0 \Rightarrow ADF(\bar{x})DF(\bar{x})^{-1}\phi = 0 \Rightarrow \phi = 0$$

となるため、 A も全単射である。

続いて、Banach の不動点定理 (定理 12) を用いて解の存在を示す。まず、作用素方程式 $F(x) = 0$ を不動点方程式に変形する。 $w := \tilde{x} - \bar{x}$ とする。 A が単射であるため、

$$\begin{aligned} F(\tilde{x}) &= 0 \\ \Leftrightarrow w &= w - AF(\tilde{x} + w) \\ \Leftrightarrow w &= -AF(\bar{x}) + w - A(F(\bar{x} + w) - F(\bar{x})) \end{aligned}$$

$\mathcal{T} : V \rightarrow V$ を

$$\mathcal{T}(w) := -AF(\bar{x}) + w - A(F(\bar{x} + w) - F(\bar{x}))$$

となる非線形作用素とし、不動点方程式 $w = \mathcal{T}(w)$ の解の存在を Banach の不動点定理 (定理 12) を用いて示す。

Banach の不動点定理 (定理 12) では M を決めて、 \mathcal{T} が M から M への縮小写像になることを確認しなければならない。この定理では、 $M = \overline{B(0, \rho)}$ 、 $\rho = \frac{\eta}{1 - (K + \delta)}$ と選ぶ。まず、 M として選んだ閉球 $\overline{B(0, \rho)}$ と定理で出てくるもう 1 つの閉球 $B\left(0, \frac{\eta}{1-\delta}\right)$ の関係性を確認する。定理の仮定より $1 - \delta > 0$ と $1 - (\delta + 2K) \geq 0$ を持つため、

$$\begin{aligned} \rho - \frac{2\eta}{1-\delta} &= \frac{\eta(1-\delta)}{(1-(\delta+K))(1-\delta)} - \frac{2\eta(1-(\delta+K))}{(1-(\delta+K))(1-\delta)} \\ &= \frac{\eta((1-\delta) - 2(1-\delta) + 2K)}{(1-(\delta+2K))(1-\delta)} \\ &= \frac{-\eta(1-\delta-2K)}{(1-(\delta+K))(1-\delta)} \leq 0 \end{aligned}$$

となる．よって、 $\rho \leq \frac{2\eta}{1-\delta}$ から $\overline{B(\bar{x}, \rho)} \subset \overline{B(0, \rho)}$ となる．

次に、 $\mathcal{T}(\overline{B(0, \rho)}) \subset \overline{B(0, \rho)}$ を示す．任意の $w \in \overline{B(0, \rho)}$ に対し、Fréchet 微分と Bochner 積分に対する微分積分学の定理を用いることで

$$\begin{aligned}
\|\mathcal{T}(w)\|_X &\leq \|AF(\bar{x})\|_X + \|w - A(F(\bar{x} + w) - F(\bar{x}))\|_X \\
&\leq \eta + \left\| w - R \int_0^1 DF((1-t)\bar{x}) + t(\bar{x} + w)w dt \right\|_X \\
&\leq \eta + \int_0^1 \|w - ADF(\bar{x} + tw))w\|_X dt \\
&\leq \eta + \int_0^1 \|I - ADF(\bar{x} + tw)\|_{\mathcal{L}(V, V)} \|w\|_X dt \\
&\leq \eta + \int_0^1 \|A(DF(\bar{x}) - DF(\bar{x} + tw))\|_{\mathcal{L}(X)} + \|I - ADF(\bar{x})\|_{\mathcal{L}(X)} \|w\|_X dt \\
&\leq \eta + \int_0^1 (\|A(DF(\bar{x}) - DF(\bar{x} + tw))\|_{\mathcal{L}(X)} + m) \|w\|_X dt
\end{aligned}$$

を得る．さらに、 $t \in [0, 1]$ に対し $tw \in \overline{B(0, \rho)} \subset \overline{B\left(0, \frac{2\eta}{1-\delta}\right)}$ となるため、定理の仮定 $\|R(DF(\bar{x}) - DF(\bar{x} + tw))\|_{\mathcal{L}(X)} \leq K$ から、

$$\begin{aligned}
\|\mathcal{T}(w)\|_X &\leq \eta + (K + \delta)\|w\|_X \\
&\leq \eta + (K + \delta)\rho \\
&= \eta + \frac{\eta(K + \delta)}{1 - (K + \delta)} \\
&= \frac{\eta - \eta(\delta + K)}{1 - (\delta + K)} + \frac{\eta(K + \delta)}{1 - (\delta + K)} \\
&= \rho
\end{aligned}$$

となる．よって、任意の $w \in \overline{B(0, \rho)}$ に対して $\|\mathcal{T}(w)\|_X \leq \rho$ となることから、 $\mathcal{T}(\overline{B(0, \rho)}) \subset \overline{B(0, \rho)}$ となる．

次に、 $\mathcal{T} : \overline{B(0, \rho)} \rightarrow \overline{B(0, \rho)}$ が縮小写像になることを確認する．すなわち

$$\|\mathcal{T}(w_1) - \mathcal{T}(w_2)\|_X \leq k\|w_1 - w_2\|_X, \quad \forall w_1, w_2 \in \overline{B(0, \rho)}$$

となる定数 k が 1 未満になることを確認しなければならない．この定理では、一意性の範囲を広げるために、 $\overline{B(0, \rho)} \subset \overline{B\left(0, \frac{2\eta}{1-m}\right)}$ であることを用いて

$$\|\mathcal{T}(w_1) - \mathcal{T}(w_2)\|_X \leq k\|w_1 - w_2\|_X, \quad \forall w_1, w_2 \in \overline{B\left(0, \frac{2\eta}{1-\delta}\right)}$$

となる定数 k が 1 未満になることを確かめる．その上, 任意の $w_1, w_2 \in \overline{B\left(\frac{2\eta}{1-\delta}\right)}$ に対し,

$$\begin{aligned}\|\mathcal{T}(w_1) - \mathcal{T}(w_2)\|_X &= \|w_1 - w_2 - A(D(\bar{x} + w_1) - F(\bar{x} + w_2))\|_X \\ &= \left\| (w_1 - w_2) - A \int_0^1 DF(\bar{x} + (1-t)w_2 + tw_1) dt \right\|_X \\ &\leq \int_0^1 \|I - ADF(\bar{x} + (1-t)w_2 + tw_1)\|_{\mathcal{L}(X)} dt \|w_1 - w_2\|_X \\ &\leq \int_0^1 \|A(DF(\bar{x}) - DF(\bar{x} + (1-t)w_2 + tw_1))\|_{\mathcal{L}(X)} + \delta dt \|w_1 - w_2\|_X\end{aligned}$$

となる．任意の $w_1, w_2 \in \overline{B\left(0, \frac{2\eta}{1-\delta}\right)}$ と $0 \leq t \leq 1$ に対し, $\|(1-t)w_2 + tw_1\|_X \leq (1-t)\|w_2\|_X + t\|w_1\|_X \leq \frac{2\eta}{1-\delta}$ となるため,

$$\|\mathcal{T}(w_1) - \mathcal{T}(w_2)\|_X \leq (K + \delta)\|w_1 - w_2\|_X$$

となる．その上, 仮定 $2K + \delta \leq 1$ かつ $\delta < 1$ より $K + \delta < 1$ も満たすため, \mathcal{T} は $\overline{B(0, \rho)}$ から $\overline{B(0, \rho)}$ への縮小写像となる．よって, Banach の不動点定理 (定理 12) より不動点方程式 $w = \mathcal{T}(w)$ の $F(u) = 0$ を満たす解が $\overline{B(0, \rho)}$ に一意に存在する． $w = \tilde{x} - \bar{x}$ であったことから, 作用素方程式 $F(x) = 0$ を満たす解が $\overline{B(\bar{x}, \rho)}$ 内に一意に存在する．

最後に, $\overline{B\left(\bar{x}, \frac{2\eta}{1-\delta}\right)}$ 内で一意に存在することを示す． $\overline{B\left(0, \frac{2\eta}{1-\delta}\right)}$ 内に不動点方程式 $w = \mathcal{T}(w)$ の解が 2 つあったとする．すなわち, 2 つの解 $\tilde{w}_1, \tilde{w}_2 \in \overline{B\left(0, \frac{2\eta}{1-\delta}\right)}$ とし, $\tilde{x}_1 = \mathcal{T}(\tilde{w}_1)$ と $\tilde{x}_2 = \mathcal{T}(\tilde{w}_2)$ を満たすとする．そのとき,

$$\|\tilde{w}_1 - \tilde{w}_2\|_X = \|\mathcal{T}(\tilde{w}_1) - \mathcal{T}(\tilde{w}_2)\|_X \leq (K + m)\|\tilde{w}_1 - \tilde{w}_2\|_X$$

となる．ここで, $2K + m < 1$ であるため, 不等式を満たすものは $\tilde{w}_1 = \tilde{w}_2$ の場合のみである．すなわち, 不動点方程式 $w = \mathcal{T}(w)$ を満たす解は $\overline{B(\bar{x}, \rho)}$ 内に一意である．よって, 作用素方程式 $F(x) = 0$ を満たす解が $\overline{B(\bar{x}, \frac{2\eta}{1-\delta})}$ 内に一意に存在する．

■

第 4 章

既存の van der Pol 方程式の精度保証付き数値計算 [2]

4.1 van der Pol 方程式

van der Pol 方程式とは、ある発振現象をもつ電気回路の方程式であり、以下のように表す。

$$\frac{d^2x}{dt^2} - \mu(1 - x^2)\frac{dx}{dt} + x = 0$$

未知関数は $x(t)$ で、 $\mu > 0$ は非線形の減衰の強さを表すパラメータである。

4.2 フーリエ・スペクトル法

フーリエ・スペクトル法の計算に必要な参照軌道をまず得るために、van der Pol 方程式の解を数値計算する。まず、van der Pol 方程式を次の連立常微分方程式系にして ODE ソルバーで数値を計算する。

$$\begin{cases} \frac{dx}{dt} = y \\ \frac{dy}{dt} = \mu(1 - x^2)y - x \end{cases}$$

初期値 $x(0) = 0$, $y(0) = 2$ とし、 $\mu = 1$ の時の数値計算を実行する。周期解をフーリエ級数で表し、その係数と周期を求めるため、van der Pol 方程式の周期解の周期を大まかに求める。得た近似周期軌道と近似周期を使って、起動のフーリエ補完を計算する。van der Pol 方程式は、 $\dot{x} = \frac{dx}{dt}$ とおくと、以下のように表すことができる。

$$\ddot{x} - \mu(1 - x^2)\dot{x} - x = 0$$

後の計算のために、式を整理すると、

$$\ddot{x} - \mu\dot{x} + \frac{\mu}{3}(\dot{x}^3) + x = 0$$

周期解 $x(t)$ を周期 L の周期関数とし、 $\omega = \frac{2\pi}{L}$ とおくと、 $x(t)$ とその微分やべき乗はフーリエ級数を使って、

$$\begin{aligned} x(t) &= \sum_{k \in \mathbb{N}} a_k \exp(ik\omega t) \\ \frac{d(x)}{dt} &= \sum_{k \in \mathbb{N}} (ik\omega) a_k \exp(ik\omega t) \\ \frac{d^2 x(t)}{dt^2} &= \sum_{k \in \mathbb{N}} (-k^2 \omega^2) a_k \exp(ik\omega t) \\ x(t)^3 &= \sum_{k \in \mathbb{N}} (a * a * a)_k \exp(ik\omega t) \end{aligned}$$

と書くことができる。ここで

$$(a * a * a)_k = \sum_{\substack{k_1 + k_2 + k_3 = k \\ k_i \in \mathbb{Z}}} a_{k_1} a_{k_2} a_{k_3}, \quad k \in \mathbb{Z}$$

は 3 次の離散畳み込みである。

そしてフーリエ係数に関する式を立てる。係数 $a = (a_k)_{k \in \mathbb{Z}}$ に対して、van der Pol 方程式に求めたフーリエ級数を代入すると、

$$f_k(a) := -k^2 \omega^2 a_k - \mu i k \omega a_k + \frac{\mu}{3} (ik\omega) (a * a * a)_k + a_k$$

となる点列 $(f_k(a))_{k \in \mathbb{Z}}$ が得られる。そして、各 $k \in \mathbb{Z}$ について

$$f_k(a) = 0$$

となる点列 a があれば、van der Pol 方程式の解のフーリエ係数になる。未知数は周期数 ω と点列 a であり、これらを並べて $x = (\omega, a)$ と書くことにする。未知数 x に対して、 $f_k(a) = 0$ という方程式だけでは不遑な方程式になるため、解の形を 1 つに定めることができない。そこで、位相条件

$$\eta(a) = \sum_{|k| \in N} a_k - \eta_0 = 0, \quad \eta_0 \in \mathbb{R}$$

を加える。この条件は、 $x(t)$ の切片 $x(0) = \eta_0$ を表している。最終的に van der Pol 方程式の周期解の球解は次の代数方程式を解くことに帰着される。

$$F(x) := \begin{bmatrix} \eta(a) \\ (f_k(a))_{k \in \mathbb{Z}} \end{bmatrix}$$

以下、この零点探索問題 $F(x) = 0$ について Newton 法で解を得ることを考える。まず、 N をフーリエ係数の打ち切り番号（最大波数： $N - 1$ ）とし、周期解の近似を次のように構成する。

$$x_N(t) = \sum_{|k| < N} \bar{a}_k \exp(ik\omega t)$$

このとき、フーリエ係数と（近似）周期をならべた

$$\bar{x} = (\bar{\omega}, \bar{a}_{-N+1}, \dots, \bar{a}_{N-1}) \in \mathbb{C}^{2N}$$

を近似解と呼ぶ。近似解 \bar{x} の項数は $2N$ 個。そして、 $f_k(a) = 0$ を $|k| < N$ の範囲で打ち切る方程式

$$F^{(N)}(x^{(N)}) = \begin{bmatrix} \eta(a^{(N)}) \\ (f_k(a^{(N)}))_{|k| < N} \end{bmatrix} = 0$$

を考える。ここで $a^{(N)} = (a_k)_{|k| < N}$, $x^{(N)} = (\omega, a^{(N)})$ をそれぞれ表し、 $F^{(N)} : \mathbb{C}^{2N} \rightarrow \mathbb{C}^{2N}$ である。したがって $F^{(N)}(x^{(N)}) = 0$ という有限次元の非線形方程式を解くことで、近似解 \bar{x} を得られる。

実際に Newton 法を用いて、周期化の数値計算を行っていく。Newton 法では、ある適当な初期値 x_0 を最初に定め、以下の反復計算によって計算できる。

$$x_{n+1} = x_n - DF^{(N)}(x_n)^{-1} F^{(N)}(x_n), \quad n = 0, 1, \dots$$

このことから、 $DF^{(N)}(x_n)^{-1}$ と $F^{(N)}(x_n)$ を計算することができれば、近似解を得ることができる。はじめに、離散畳み込みの関数を用意する。

$F^{(N)}(x^{(N)})$ のヤコビ行列は

$$DF^{(N)}(x^{(N)}) = \left[\begin{array}{c|ccc} 0 & 1 & \cdots & 1 \\ \vdots & & \vdots & \\ \partial_\omega f_k & \cdots & \partial_{a_j} f_k & \cdots \\ \vdots & & \vdots & \end{array} \right] \in \mathbb{C}^{2N \times 2N} \quad (|k|, |j| < N)$$

ここで、

$$\begin{cases} \partial_\omega f_k = (-2k^2\omega - \mu ik)a_k + \frac{\mu ik}{\omega}(a * a * a)_k & (|k| < N) \\ \partial_{a_j} f_k = (k^2\omega^2 - \mu ik\omega)\delta_{k_j} + \mu ik\omega(a * a)_{k-j} & (|k| < N) \end{cases}$$

$$\delta_{k_j} = \begin{cases} 1 & (k = j) \\ 0 & (k \neq j) \end{cases}$$

である。ヤコビ行列の各要素との対応は

$$\begin{aligned} & \left(DF^{(N)}(x^{(N)}) \right)_{l,m} \\ &= \begin{cases} 0 & (l = m = 1) \\ 1 & (l = 1, m = 2, \dots, 2N) \\ \partial_\omega f_k & (l = 2, \dots, 2N, m = 1, \text{ i.e., } l = k + N + 1 \text{ for } |k| < N) \\ \partial_{a_j} f_k & (l, m = 2, \dots, 2N, \text{ i.e., } l = k + N + 1 \text{ for } |k| < N, m = j + N + 1 \text{ for } |j| < N) \end{cases} \end{aligned}$$

4.3 重み付き空間と作用素の決定

定義 34 (許容重み). 点列 $\omega = (\omega_k)_{k \in \mathbb{Z}}$ について,

$$\begin{aligned}\omega_k &> 0, \forall k \in \mathbb{Z} \\ \omega_{n+k} &\leq \omega_n \omega_k, \forall n, k \in \mathbb{Z}\end{aligned}$$

が成立するとき, 許容重み (admissible wights) であるという.

定理 15 (重み付き l^1 空間).

$$l_\omega^1 := \left\{ a = (a_k)_{k \in \mathbb{Z}} : a_k \in \mathbb{C}, \|a\|_\omega := \sum_{k \in \mathbb{Z}} |a_k| \omega_k < \infty \right\}$$

点列 ω を許容重みとする. $(l_m^1, *)$ は可換な Banach 環となる. すなわち, 点列 $a, b \in l_m^1$ として, 離散畳み込み $*$ に対して

$$\|a * b\|_\omega \leq \|a\|_\omega \|b\|_\omega$$

が成立する.

証明. $a, b \in l_\omega^1$ として,

$$\begin{aligned}\|a * b\|_\omega &= \sum_{k \in \mathbb{Z}} \|(a * b)\|_{\omega_k} \\ &= \sum_{k \in \mathbb{Z}} \left\| \sum_{\substack{k_1 + k_2 = k \\ k_1, k_2 \in \mathbb{Z}}} a_{k_1} a_{k_2} \right\|_{\omega_k} \\ &\leq \sum_{k \in \mathbb{Z}} \left(\sum_{\substack{k_1 + k_2 = k \\ k_1, k_2 \in \mathbb{Z}}} |a_{k_1}| |a_{k_2}| \omega_k \right) \\ &\leq \sum_{k \in \mathbb{Z}} \left(\sum_{\substack{k_1 + k_2 = k \\ k_1, k_2 \in \mathbb{Z}}} |a_{k_1}| \omega_{k_1} |a_{k_2}| \omega_{k_2} \right) \\ &\leq \left(\sum_{k_1 \in \mathbb{Z}} |a_{k_1}| \omega_{k_1} \right) \left(\sum_{k_2 \in \mathbb{Z}} |a_{k_2}| \omega_{k_2} \right) \\ &= \|a\|_\omega \|b\|_\omega\end{aligned}$$

■

定義 35 (Banach 空間 X). Banach 空間 X を次のように定める. はじめに重み付き l^1 空間を重み $w_k = \nu^{|k|}$ ($\nu = 1.05$) として次のように定める.

$$l_\nu^1 := \left\{ a = (a_k)_{k \in \mathbb{Z}} : a_k \in \mathbb{C}, \|a\|_w := \sum_{k \in \mathbb{Z}} |a_k| \nu^{|k|} < \infty \right\}$$

そして, 検証に用いいる関数空間 X は Banach 空間となる.

$$X := \mathbb{C} \times l_\nu^1, \quad x = (\omega, a), \quad \omega \in \mathbb{C}, \quad a \in l_\nu^1$$

と定め, そのノルムを

$$\|x\|_X := \max\{|\omega|, \|a\|_w\}$$

として, 定義する. このとき, X は Banach 空間となる.

定義 36 (ヤコビ行列 $DF^{(N)}(x^{(N)})$). $F^{(N)}(x^{(N)})$ のヤコビ行列は, 次のような形をしている.

$$DF^{(N)}(x^{(N)}) = \left[\begin{array}{c|ccc} 0 & 1 & \cdots & 1 \\ \vdots & & \vdots & \\ \partial_w f_k & \cdots & \partial_{a_j} f_k & \cdots \\ \vdots & & \vdots & \end{array} \right] \in \mathbb{C}^{2N \times 2N}, \quad (|k|, |j| < N)$$

ここで,

$$\begin{cases} \partial_w f_k = (-2k^2\omega - \mu ik)a_k + \frac{\mu ik}{3}(a * a * a)_k & (|k| < N) \\ \partial_{a_j} f_k = (k^2\omega^2 - \mu ik\omega)\delta_{k_j} + \mu ik\omega(a * a)_{k-j} & (|k| < N) \end{cases}$$

$$\delta_{k_j} = \begin{cases} 1 & (k = j) \\ 0 & (k \neq j) \end{cases}$$

である. ヤコビ行列の各要素の伏せ字とその対応は

$$\begin{aligned} & \left(DF^{(N)}(x^{(N)}) \right)_{l,m} \\ &= \begin{cases} 0 & (l = m = 1) \\ 1 & (l = 1, m = 2, \dots, 2N) \\ \partial_w f_k & (l = 2, \dots, 2N, m = 1, \text{ i.e., } l = k + N + 1 \text{ for } |k| < N) \\ \partial_{a_j} f_k & (l, m = 2, \dots, 2N, \text{ i.e., } l = k + N + 1 \text{ for } |k| < N, m = j + N + 1 \text{ for } |j| < N) \end{cases} \end{aligned}$$

定義 37 (作用素 A^\dagger, A の定義). まず, Banach 空間 $X = \mathbb{C} \times l_\nu^1$ から $Y = \mathbb{C} \times l_{\nu'}^1$, ($\nu' < \nu$) と設定し, A^\dagger を $A^\dagger \in (X, Y)$ として, $b = (b_0, b_1) \in \mathbb{C} \times l_\nu^1 = X$ に対して, $A^\dagger b = ((A^\dagger b)_0, (A^\dagger b)_1)$

と作用素するように定義する．ここで， A^\dagger を形式的に見ると，

$$A^\dagger = \left[\begin{array}{c|cccccc} 0 & 1 & \cdots & 1 & \cdots & 0 & \cdots \\ \vdots & & \vdots & & & & \\ \partial_w f_k & \cdots & \partial_{a_j} f_k & \cdots & & 0 & \\ \vdots & & \vdots & & & & \\ \vdots & & & & \lambda_N & & 0 \\ 0 & & 0 & & & \lambda_{N+1} & \\ \vdots & & & 0 & & & \ddots \end{array} \right] = \left[\begin{array}{c|c} 0 & A_{a,0}^\dagger \\ \hline A_{\omega,1}^\dagger & A_{a,0}^\dagger \end{array} \right]$$

このことから， $A^\dagger b$ は

$$A^\dagger b = \begin{bmatrix} 0 & A_{a,0}^\dagger \\ A_{\omega,1}^\dagger & A_{a,1}^\dagger \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} A_{a,0}^\dagger b_1 \\ A_{\omega,1}^\dagger b_0 + A_{a,1}^\dagger b_1 \end{bmatrix} =: \begin{bmatrix} (A^\dagger b)_0 \\ (A^\dagger b)_1 \end{bmatrix}$$

と表すことができ，

$$(A^\dagger b)_0 = \sum_{|k| < N} (b_1)_k$$

$$((A^\dagger b)_1)_k := \begin{cases} \partial_\omega f_k b_0 + \sum_{|j| < N} \partial_{a_j} f_k (b_1)_j & , |k| < N \\ \lambda_k (b_1)_k & , |k| \geq N, \lambda_k := -k^2 \omega^2 - \mu i k \omega + 1 \end{cases}$$

と書ける．またこのとき， $(A^\dagger b)_0$ と $(A^\dagger b)_1$ はそれぞれ

$$(A^\dagger b)_0 = A_{a,0}^\dagger b_1 = \sum_{|k| < N} (b_1)_k \in \mathbb{C}$$

$$(A^\dagger b)_1 = A_{\omega,1}^\dagger b_0 + A_{a,1}^\dagger b_1 \in l_\nu^1,$$

次に，作用素 A について考える．

$$A^N = \begin{bmatrix} A_{\omega,0}^{(N)} & A_{a,0}^{(N)} \\ A_{\omega,1}^{(N)} & A_{a,1}^{(N)} \end{bmatrix} \approx DF^{(N)}(\bar{x})^{-1} \in \mathbb{C}^{2N \times 2N}$$

をヤコビ行列の近似逆行列とする． $A \in \mathcal{L}(Y, X)$ として， $b = (b_0, b_1) \in X$ に対して， $Ab = ((Ab)_0, (Ab)_1)$ と作用するように定義する．ここで，

$$(Ab)_0 = A_{\omega,0}^{(N)} b_0 + A_{a,0}^{(N)} b_1^{(N)}$$

$$(Ab)_1 = A_{\omega,1}^{(N)} b_0 + A_{a,1}^{(N)} b_1$$

ただし，無限次元の $A_{a,1} b_1$ は以下のようになる．

$$(A_{a,1} b_1) = \begin{cases} (A_{a,1}^{(N)} b_1^{(N)})_k & (|k| < N) \\ \frac{(b_1)_k}{\lambda_k} & (|k| \geq N) \end{cases}$$

この定義を形式的に見ると,

$$A = \left[\begin{array}{c|ccc} A_{\omega,0}^{(N)} & A_{a,0}^{(N)} & 0 & \cdots & 0 \\ \hline A_{\omega,1}^{(N)} & A_{a,1}^{(N)} & & & 0 \\ 0 & & \frac{1}{\lambda_N} & & \\ \vdots & & & \frac{1}{\lambda_{N+1}} & \\ 0 & & 0 & & \ddots \end{array} \right] = \left[\begin{array}{c|c} 0 & A_{a,0} \\ \hline A_{\omega,1} & A_{a,0} \end{array} \right]$$

と表記できる.

4.4 Y_0, Z_0, Z_1, Z_2 の評価

Y_0

$$\|AF(\bar{x})\|_X \leq Y_0$$

Z_0

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

Z_1

$$\|A(DF(\bar{x})A^\dagger)c\|_X \leq Z_1, \quad c \in \overline{B(0,1)}$$

また, ここでの $\overline{B(0,1)}$ とは $\|c\|_X = 1$ ということである.

Z_2

$b \in \overline{B(\bar{x}, r)}, \quad h \in \overline{B(0,1)}$ として,

$$\|A(DF(b) - DF(\bar{x})h)\|_X \leq Z_2(r)r$$

4.4.1 Y_0 を計算する

$$F(\bar{x}) = (\delta_0, \delta_1) \in \mathbb{C} \times l'_\nu$$

とすると, A の定義より,

$$\|AF(\bar{x})\|_X \leq \max \left\{ \left| A_{\omega,0}^{(N)} \delta_0 + A_{a,0}^{(N)} \delta_1^{(N)} \right|, \left\| A_{\omega,1}^{(N)} \delta_0 + A_{a,1}^{(N)} \delta_1^{(N)} \right\|_\omega + \sum_{|k| > N} \left\| \frac{(\delta_1^\infty)_k}{\lambda_k} \right\| \nu^{|k|} \right\}$$

ここで, $\delta_1 = (\delta_1^{(N)}, \delta_1^{(\infty)}) \in \mathbb{C}^{2 \times 3(N-1)+1}$ であり,

$$(\delta_1)_k = \begin{cases} \delta_1^{(N)} & (k < |N|) \\ \delta_1^{(\infty)} & (k \geq |N|) \end{cases}$$

と表す.

4.4.2 Z_0 を計算する

$$B := I - AA^\dagger = \begin{bmatrix} B_{\omega,0} & B_{a,0} \\ B_{\omega,1} & B_{a,1} \end{bmatrix}$$

この B を $c \in \overline{B(0,1)}$, $\|c\|_X \leq 1$ である $c = (c_0, c_1)$ に作用させると,

$$(Bc)_0 = B_{\omega,0}c_0 + B_{a,0}c_1$$

$$(Bc)_1 = B_{\omega,1}c_0 + B_{a,1}c_1$$

Bc_0 はスカラー値なので,

$$\begin{aligned} |B_{a,0}c_1| &\leq \sum_{k \in \mathbb{Z}} |(B_{a,0})_k| |(c_1)_k| \\ &= \sum_{k \in \mathbb{Z}} \frac{|(B_{a,0})_k|}{\omega_k} |(c_1)_k| \omega_k \\ &\leq \max_{k < |N|} \frac{|(B_{a,0})_k|}{\omega_k} \sum_{k \in \mathbb{Z}} |(c_1)_k| \omega_k \\ &\leq \max_{k < |N|} \frac{|(B_{a,0})_k|}{\omega_k}, \left(\sum_{k \in \mathbb{Z}} |(c_1)_k| \omega_k = \|c_1\|_\omega \leq 1 \right) \end{aligned}$$

以上より,

$$|(Bc)_0| \leq |B_{\omega,0}| + \max_{|k| < N} \frac{|(B_{a,0})_k|}{\omega_k} = Z_0^{(0)}$$

またここで, 作用素 $M : l_\nu^1 \Rightarrow l_\nu^1$ の作用素ノルムについて以下の補題を陣日する.

補題 1. 行列 $M^{(N)}$ を $M^{(N)} \in \mathbb{C}^{(2N-1) \times (2N-1)}$, 双方向の複素無限点列を $(\delta_k)_{|k| \geq N}$ と定義する. ここで, $\delta_N > 0$ であり,

$$|\delta_k| \leq \delta_N, \quad \forall |k| \geq N$$

を満たすとする. そして, $a = (a_k)_{k < \mathbb{Z}} \in l_\nu^1$ に対して $a^{(N)} = (a_{-N+1}, \dots, a_{N-1}) \in \mathbb{C}^{2N-1}$ と表し, 作用素 $M : l_\nu^1 \Rightarrow l_\nu^1$ を以下のように定義する.

$$[Ma]_k := \begin{cases} [M^{(N)}a^{(N)}]_k & (|k| < N) \\ \delta_k a_k & (|k| \geq N) \end{cases}$$

このとき, M は有界線形作用素であり,

$$\|M\|_{\mathcal{L}(l_\nu^1)} \leq \max(K, \delta_N), \quad K := \max_{|n| < N} \frac{1}{\nu^{|n|}} \sum_{|k| < N} |M_{n,k}| \nu^{|k|}$$

と評価させる.

上の補題を利用すると,

$$\|(Bc)_1\|_\omega \leq \|B_{\omega,1}\|_\omega + \|B_{a,1}\|_{\mathcal{L}(l_\nu^1)} = Z_0^{(1)}$$

が評価可能となり, 結論としては, 求めたい Z_0 は $Z_0 := \{Z_0^{(0)}, Z_0^{(0)}\}$ となる.

4.4.3 Z_1 を計算する

$$\|A(DF(\bar{x}) - A^\dagger)c\|_X \leq Z_1, \quad c = (c_0, c_1) \in \overline{B(0,1)} \Leftrightarrow \|c\|_X \leq 1$$

点列 z を下記のように定義する.

$$z := (DF(\bar{x}) - A^\dagger)c = \begin{bmatrix} z_0 \\ z_1 \end{bmatrix}$$

ここで, $DF(\bar{x})$ と A^\dagger は

$$DF(\bar{x}) = \left[\begin{array}{c|ccc} 0 & \cdots & 0 & \cdots \\ \hline \partial_\omega f_k^{(N)} & \cdots & 0 & \cdots \\ \vdots & & \vdots & \end{array} \right]$$

$$A^\dagger = \left[\begin{array}{c|ccccccc} 0 & 1 & \cdots & 1 & \cdots & 0 & \cdots \\ \hline \vdots & & \vdots & & & & \\ \partial_w f_k & \cdots & \partial_{a_j} f_k & \cdots & & & \\ \vdots & & \vdots & & & 0 & \\ \vdots & & & & \lambda_N & & \\ 0 & & 0 & & & \lambda_{N+1} & \\ \vdots & & & & & & \ddots \end{array} \right]$$

$$\lambda_k := -k^2\omega^2 - \mu ik\omega + 1$$

と表される. すると z_0 は,

$$z_0 = \sum_{|k| < N} (c_1)_k,$$

$$|z_0| \leq \frac{1}{\omega_N} \sum_{|k| \geq N} |(c_1)_k| \omega_k \leq \frac{1}{\omega_N}$$

次に, z_1 について考える. $DF(\bar{x})c$ 部分は,

$$\begin{aligned} ((DF(\bar{x})c)_1)_k &= \partial_\omega f_k c_0 + \partial_a f_k c_1 \\ &= \frac{\partial \lambda_k}{\partial \omega} c_0 \bar{a}_k + \frac{\mu ik}{3} (\bar{a} * \bar{a} * \bar{a})_k c_0 + \lambda_k (c_1)_k + \mu ik\omega (\bar{a} * \bar{a} * c_1)_k, \quad k \in \mathbb{Z} \end{aligned}$$

と書け, $|k| \geq N$ で $\bar{a}_k = 0$ より, $c_1 = c_1^{(N)} + c_1^{(\infty)}$ として,

$$(z_1)_k = \begin{cases} \mu i k \omega \left(\bar{a} * \bar{a} * c_1^{(\infty)} \right)_k, & |k| < N \\ \frac{\mu i k}{3} (\bar{a} * \bar{a} * \bar{a})_k c_0 + \mu i k \omega (\bar{a} * \bar{a} * c_1), & |k| \geq N \end{cases}$$

と表せる．ここから， z_1 の絶対値をとると， $|k| < N$ で

$$|(z_1)_k| \leq |\mu i k \omega| \max \left\{ \max_{k-N+1 \leq j \leq -N} \frac{|(\bar{a} * \bar{a})_{k-j}|}{\omega_j}, \max_{N \leq j \leq k+N-1} \frac{|(\bar{a} * \bar{a})_{k-j}|}{\omega_j} \right\} := \zeta,$$

$$\zeta = (\zeta_k)_{k < N} \in \mathbb{R}^{2N-1}$$

最後に， Z_1 を評価していく． $Z_1^{(0)}$ の評価は

$$\begin{aligned} |(A(DF(\bar{x}) - A^\dagger)c)_0| &= |(Az)_0| \\ &\leq |A_{\omega,0}^{(N)}| |Z_0| + |A_{a,0}^{(N)}| |Z_1^{(N)}| \\ &\leq \frac{|A_{\omega,0}^{(N)}|}{\omega_N} + |A_{a,0}^{(N)}| \zeta \\ &=: z_1^{(0)} \end{aligned}$$

$Z_1^{(1)}$ の評価は

$$\begin{aligned} \|(A(DF(\bar{x}) - A^\dagger)c)_1\|_\omega &= \|(Az)_1\|_\omega \\ &= \|A_{\omega,1}^{(N)} z_0 + A_{a,1} z_1\|_\omega \\ &\leq \frac{\|A_{\omega,1}^{(N)}\|_\omega}{\omega_N} + \sum_{|k| < N} \left(|A_{a,1}^{(N)}| \zeta \right)_k \omega_k + \sum_{N \leq |k| \leq 3(N-1)} \frac{|\mu i k (\bar{a} * \bar{a} * \bar{a})_k|}{3 |\lambda_k|} \omega_k \\ &\quad + \sum_{|k| \geq N} \frac{|\mu i k \omega (\bar{a} * \bar{a} * \bar{x})_k|}{|\lambda_k|} \omega_k \\ &\leq \frac{\|A_{\omega,1}^{(N)}\|_\omega}{\omega_N} + \| |A_{a,1}^{(N)}| \zeta \|_\omega + \sum_{N \leq |k| \leq 3(N-1)} \frac{|\mu i k (\bar{a} * \bar{a} * \bar{a})_k|}{3 |\lambda_k|} \omega_k \\ &\quad + \frac{1}{N} \frac{\mu \omega \|\bar{a}\|_\omega^2}{\omega^2 - \frac{1}{N^2}} \\ &=: Z_1^{(1)} \end{aligned}$$

よって，

$$Z_1 := \max \left\{ Z_1^{(0)}, Z_1^{(1)} \right\}$$

4.4.4 Z_2 を計算する

$b \in \overline{B(\bar{x}, r)}$, $c = (c_0, c_1) \in \overline{Z(0, 1)}$ について

$$\|A(DF(b) - DF(\bar{x}))c\|_X \leq Z_2(r)r$$

を考える．まず z を

$$z := (DF(b) - DF(\bar{x}))c = \begin{bmatrix} z_0 \\ z_1 \end{bmatrix} = \begin{bmatrix} 0 \\ z_1 \end{bmatrix}$$

と定義する． $z_0 = 0$ となるので， z_1 だけを考えればよく

$$(z_1)_k := (\partial_\omega f_k(b) - \partial_\omega f_k(\bar{x}))c_0 + [(\partial_a f(b) - \partial_a f(\bar{x}))c_1]_k, \quad k \in \mathbb{Z}$$

と書ける． $b = (\omega, (a_k)_{k \in \mathbb{Z}})$, $\bar{x} = (\bar{\omega}, (\bar{a}_k)_{|k| < N})$ として，第 1 項は

$$\begin{aligned} & (\partial_\omega f_k(b) - \partial_\omega f_k(\bar{x}))c_0 \\ &= \left[\left((-2k^2\omega - \mu ik) a_k + \frac{\mu ik}{3} (a * a * a)_k \right) - \left((-2k^2\bar{\omega} - \mu ik) \bar{a}_k + \frac{\mu ik}{3} (\bar{a} * \bar{a} * \bar{a})_k \right) \right] c_0 \\ &= \left[-2k^2\omega (a_k - \bar{a}_k) - 2k^2(\omega - \bar{\omega}) \bar{a}_k - \mu ik (a_k - \bar{a}_k) + \frac{\mu ik}{3} ((a * a * a)_k - (\bar{a} * \bar{a} * \bar{a})_k) \right] c_0 \end{aligned}$$

と書ける．そして，第 2 項は，

$$\begin{aligned} & [(\partial_a f(b) - \partial_a f(\bar{x}))c_1]_k \\ &= (-k^2\omega^2 - \mu ik\omega + 1)(c_1)_k + \mu ik\omega(a * a * c_1)_k - (-k^2\bar{\omega}^2 - \mu ik\bar{\omega} + 1)(c_1)_k \\ & \quad + \mu ik\bar{\omega}(\bar{a} * \bar{a} * c_1)_k \\ &= [-k^2(\omega + \bar{\omega})(\omega - \bar{\omega}) - \mu ik(\omega - \bar{\omega})](c_1)_k + \mu ik\omega((a + \bar{a}) * (a - \bar{a}) * c_1)_k \\ & \quad + \mu ik(\omega - \bar{\omega})(\bar{a} * \bar{a} * c_1)_k \end{aligned}$$

と書ける． $(Az)_0, (Az)_1$ は，

$$\begin{aligned} (Az)_0 &= A_{a,0}^{(N)} + z_1^{(N)} \\ (Az)_1 &= A_{a,1} + z_1 \end{aligned}$$

より，

$$\|Az\|_X = \max \left\{ \left| A_{a,0}^{(N)} z_1^{(N)} \right|, \|A_{a,1} z_1\| \right\}$$

となる．

$\left| A_{a,0}^{(N)} z_1^{(N)} \right|$ を上から評価する．はじめに， $\tilde{A}_{a,0}, \tilde{B}_{a,0}$ を以下のように定義する．

$$\begin{aligned} \tilde{A}_{a,0} &:= \left(|k| \left(A_{a,0}^{(N)} \right)_k \right)_{|k| < N} \\ \tilde{B}_{a,0} &:= \left(k^2 \left(A_{a,0}^{(N)} \right)_k \right)_{|k| < N} \end{aligned}$$

すると，

$$\begin{aligned} \left| A_{a,0}^{(N)} z_1^{(N)} \right| &\leq 2(\tilde{\omega} + r) \left\| \tilde{B}_{a,0} \right\|_\omega r + 2 \left\| \tilde{B}_{a,0} \right\|_\omega \left\| \tilde{a} \right\|_\omega r + \mu \left\| \tilde{A}_{a,0} \right\|_\omega r \\ &\quad + \frac{\mu}{3} \left\| \tilde{A}_{a,0} \right\|_\omega \left(r^2 + 3 \left\| \tilde{a} \right\|_\omega r + 3 \left\| \tilde{a} \right\|_\omega^2 \right) r + \left\| \tilde{B}_{a,0} \right\|_\omega (2\tilde{\omega} + r) r + \mu \left\| \tilde{A}_{a,0} \right\|_\omega r \\ &\quad + \mu(\tilde{\omega} + r) \left\| \tilde{A}_{a,0} \right\|_\omega (2 \left\| \tilde{a} \right\|_\omega + r) r + \mu \left\| \tilde{A}_{a,0} \right\|_\omega \left\| a \right\|_\omega^2 r \\ &= Z_2^{(4)} r^3 + Z_2^{(3)} r^2 + Z_2^{(2)} r \end{aligned}$$

となる．同様に， $\|A_{a,1}z_1\|_\omega$ を上から評価する． $\tilde{A}_{a,1}, \tilde{B}_{a,1}$ を以下のように定義する．

$$\begin{aligned}\tilde{A}_{a,1} &:= \left(|j| (A_{a,1})_{k,j} \right)_{k,j \in \mathbb{Z}} \\ \tilde{B}_{a,1} &:= \left(j^2 (A_{a,1})_{k,j} \right)_{k,j \in \mathbb{Z}}\end{aligned}$$

すると，

$$\begin{aligned}\|A_{a,1}z_1\|_\omega &\leq 2(\tilde{\omega} + r) \left\| \tilde{B}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} r + 2 \left\| \tilde{B}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} \|\tilde{a}\|_\omega r + \mu \left\| \tilde{A}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} r \\ &\quad + \frac{\mu}{3} \left\| \tilde{A}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} \left(r^2 + 3 \|\tilde{a}\|_\omega r + 3 \|\tilde{a}\|_\omega^2 \right) r + \left\| \tilde{B}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} (2\tilde{\omega} + r) r + \mu \left\| \tilde{A}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} r \\ &\quad + \mu (\tilde{\omega} + r) \left\| \tilde{A}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} (2 \|\tilde{a}\|_\omega + r) r + \mu \left\| \tilde{A}_{a,1} \right\|_{\mathcal{L}(l_\nu^1)} \|a\|_\omega^2 r \\ &= Z_2^{(4,1)} r^3 + Z_2^{(3,1)} r^2 + Z_2^{(2,1)} r\end{aligned}$$

と書ける． $Z_2^{(4,1)}, Z_2^{(3,1)}, Z_2^{(2,1)}$ は，先ほどの $\tilde{A}_{a,0}, \tilde{B}_{a,0}$ を $\tilde{A}_{a,1}, \tilde{B}_{a,1}$ に置き換えたものになる． $j = 2, 3, 4$ で

$$Z_2^{(j)} := \max \left\{ Z_2^{(j,0)}, Z_2^{(j,1)} \right\}$$

とすれば，

$$Z_2(r) := Z_2^{(4,1)} r^2 + Z_2^{(3,1)} r + Z_2^{(2,1)}$$

となる．

第 5 章

無限次元ガウスの消去法 [4]

5.1 射影

定義 38 (代数的直和). Banach 空間 X とする. また, X_1, X_2 を X の線形部分空間とする. ただし, X_1 と X_2 のノルムは, X のノルムと同一とする. そのとき, X が X_1 と X_2 の代数的直和であるとは

- $X = X_1 + X_2 := \{x_1 + x_2 \mid \forall x_1 \in X_1, \forall x_2 \in X_2\}$
- $X_1 \cap X_2 = \{0\}$

が成立することをいう.

定義 39 (射影). X をノルム空間とする. 定義域を X とした X 上の線形作用素 P が

$$P^2 = P$$

となるとき, 線形作用素, あるいは, 単に射影と呼ぶ.

定理 16. Banach 空間 X がその線形部分空間 X_1, X_2 の代数的直和であるとする. そのとき, $x \in X$ について

$$x = x_1 + x_2, x_1 \in X_1, x_2 \in X_2$$

とし, $\mathcal{D}(P) = X$ となる X 上の線形作用素 P を

$$Px = x_1, (I - P)x = x_2$$

とすると, 線形作用素 P と $I - P$ は射影になる.

証明. まず, $Px_2 = 0$ を証明する. $Px_2 \neq 0$ となる $x_2 \in X_2$ が存在すると仮定し, 矛盾を示す.

定義より,

$$x = x_1 + x_2 = x_1 + Px_2 + (I - P)x_2$$

に対して, P の閾値 $\mathcal{R}(P) = X_1$ であることと, X_1 が線形空間であることに注意すると,

$x_1 + Px_2 \in x_2$ である。しかし、これは代数的直和の定義より x の分解の一意性に矛盾する。よって $Px_2 = 0$ となる。

そのうえで、

$$Px_2 = P(I - P)x = Px - P^2x = 0$$

より

$$Px = P^2x$$

となるため、 P は射影となる。

また、

$$(I - P)(I - P)x = (I - 2P + P^2)x = (I - 2P + P)x = (I - P)x$$

となるため、 $I - P$ も射影となる。 ■

定理 17. Banach 空間 X が X_1 と X_2 の代数的直和であるとする。 $x \in X$ について

$$x = x_1 + x_2, \quad x_1 \in X_1, \quad x_2 \in X_2$$

とし、 $\mathcal{D}(P) = X$ となる X となる X 上の線形作用素 P を

$$Px = x_1, \quad (I - P)x = x_2$$

とすると以下が成立する。

$$P \text{ が連続} \Leftrightarrow X_1, X_2 \text{ がともに Banach 空間}$$

証明. 「 P が連続 $\Rightarrow X_1, X_2$ がともに Banach 空間の証明」まず、 X_1 が Banach 空間であることを示す。 X が X_1, X_2 の代数的直和であることから、 X_1 は X の線形部分空間である。そのため、 X_1 が閉空間であること示せば、定理 5 より、 X_1 が Banach 空間であることを示せる。よって、 X_1 の任意の点列 $(x_n) \subset X_1$ に対し、極限 x^* が X_1 に属することを示せばよい。 x_n は X_1 に属することから $Px_n = x_n$ になる。そのうえで、 P が連続であることから、 $Px_n = x_n$ の極限は $Px^* = x^*$ となる。よって、 $Px^* = x^* \in X_1$ から、 X_1 は閉集合となるため、Banach 空間となる。

次に、 X_2 が Banach 空間であることを示す。そのために、まず、 $I - P$ が連続になることを示す。 P が連続であることから、 $x_n \rightarrow x^*$ となる任意の x_n に対して、 $Px_n \rightarrow Px^*$ となることに注意すると

$$(I - P)x_n = x_n - Px_n \rightarrow x^* - Px^* = (I - P)x^*$$

となるため、 $I - P$ も連続となる。あとは、 P と X_1 のときと同様の議論をすればよい。すると、 $(I - P)x^* = x^* \in X_2$ から、 X_2 は閉集合となるため、Banach 空間となる。

「 P が連続 $\Leftarrow X_1, X_2$ がともに Banach 空間の証明」

Banach 空間 X の $x_n \rightarrow x^*$ となる任意の点列 (x_n) に対し、 $Px_n \rightarrow y$ としたとき、 $y = Px^*$ となることで示せば良い。上記の点列を用いて X_1 の点列 (Px_n) を作成すると、 X_1 が Banach 空間であることから、閉集合であるため点列 (Px_n) の極限 y は X_1 にも属する。よって、

$$y = Py$$

となる。

さらに、同様に X_2 の点列 $((I - P)x_n)$ を作成すると、その極限は

$$(I - P)x_n = x_n - Px_n \rightarrow x^* - y$$

となり、 X_2 も Banach 空間であることがら、閉集合であるため点列 $((I_P)x_n)$ の極限 $x^* - y$ は X^2 にも属する。そのうえで、Banach 空間 X が X_1 と X_2 の代数的直和であることから、 $X_1 \cup X_2 = 0$ に注意すると $P(x^* - y) = 0$ となる。よって、

$$0 = P(x^* - y) = Px^* - Py = Px^* - y$$

となるため、

$$Px^* = y$$

となる。 ■

定義 40 (位相的直和). Banach 空間 X が X_1 と X_2 の代数的直和であるとする。その上、 X_1 と X_2 が共に Banach 空間であるとき、Banach 空間 X が X_1 と X_2 の位相的直和であるという。

5.2 射影を用いた Banach 空間上の無限次元ガウスの消去法

初めに、 $X, X_1, X_2, L, g, P, \phi_1, \phi_2, T, B, C, D$ を定義する。

Banach 空間 X とし、有界な線形作用素 $L \in \mathcal{B}(X)$ と $g \in X$ に対して問題

$$\text{Find } \phi \in X \text{ s.t. } L\phi = g$$

の解を求める方法を考える。

Banach 空間 X が X_1 と X_2 の代数的直和であるとし、 $x \in X$ について

$$x = x_1 + x_2, x_1 \in X_1, x_2 \in X_2$$

とし、 P を $\mathcal{D}(P) = X$

$$Px = x_1, (I - P)x = x_2$$

を満たす射影をする。

上記の式より、

$$L\phi = g \Leftrightarrow \begin{cases} PL\phi = Pg \\ (I - P)L\phi = (I - P)g \end{cases}$$

さらに、解 ϕ も射影を使って分解する。

$$\phi = \phi_1 + \phi_2, \phi_1 := P\phi, \phi_2 := (I - P)\phi$$

この分解した解を利用すると

$$\begin{cases} PL(\phi_1 + \phi_2) = Pg \\ (I - P)L\phi = (I - P)g \end{cases}$$

となる。ここで、それぞれの線形作用素を

$$\begin{aligned} T &:= PL|_{X_1}: X_1 \rightarrow X_1, & B &:= PL|_{X_2}: X_2 \rightarrow X_1 \\ C &:= (I - P)L|_{X_1}: X_1 \rightarrow X_2, & D &:= (I - P)L|_{X_2}: X_2 \rightarrow X_2 \end{aligned}$$

と定義すると、次のように変形できる。

$$\begin{cases} T\phi_1 + B\phi_2 = Pg \\ C\phi_1 + D\phi_2 = (I - P)g \end{cases}$$

ここで、

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix}: X_1 \times X_2 \rightarrow X_1 \times X_2$$

を定義すると

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix}$$

と変形できる。

定理 18. $X, X_1, X_2, L, g, P, \phi_1, \phi_2, T, B, C, D$ は、本節で定義したものとする。

線形作用素 T を全単射であると仮定する、作用素 S を

$$S := D - CT^{-1}B: X_2 \rightarrow X_2$$

とする。もし、 S が全単射ならば、

$$\begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} T^{-1} + T^{-1}BS^{-1}CT^{-1} & -T^{-1}BS^{-1} \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix}$$

となり、有界線形作用素 L は全単射である。

証明. 方程式

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix}$$

に対して、 T が全単射であることから左から

$$\begin{pmatrix} T^{-1} & 0 \\ 0 & I_{X_2} \end{pmatrix}$$

を掛けると

$$\begin{pmatrix} I_{X_1} & T^{-1}B \\ C & D \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} T^{-1} & 0 \\ 0 & I_{X_2} \end{pmatrix} \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix}$$

となる。次に、左から

$$\begin{pmatrix} I_{X_1} & 0 \\ -C & I_{X_2} \end{pmatrix}$$

を掛けると

$$\begin{aligned} \begin{pmatrix} I_{X_1} & T^{-1}B \\ 0 & S \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} &= \begin{pmatrix} I_{X_1} & 0 \\ -C & I_{X_2} \end{pmatrix} \begin{pmatrix} T^{-1} & 0 \\ 0 & I_{X_2} \end{pmatrix} \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix} \\ &= \begin{pmatrix} T^{-1} & 0 \\ -CT^{-1} & I_{X_2} \end{pmatrix} \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix} \end{aligned}$$

となる．次に， S が全単射であることから左から

$$\begin{pmatrix} I_{X_1} & 0 \\ 0 & S^{-1} \end{pmatrix}$$

を掛けると

$$\begin{pmatrix} I_{X_1} & T^{-1}B \\ 0 & S \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} T^{-1} & 0 \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} Pg \\ (I-P)g \end{pmatrix}$$

となる．最後に，

$$\begin{pmatrix} I_{X_1} & -T^{-1}B \\ 0 & I_{X_2} \end{pmatrix}$$

を左から掛けると

$$\begin{aligned} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} &= \begin{pmatrix} I_{X_1} & -T^{-1}B \\ 0 & I_{X_2} \end{pmatrix} \begin{pmatrix} T^{-1} & 0 \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} Pg \\ (I-P)g \end{pmatrix} \\ &= \begin{pmatrix} T^{-1} + T^{-1}BS^{-1}CT^{-1} & -T^{-1}BS^{-1} \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} Pg \\ (I-P)g \end{pmatrix} \end{aligned}$$

が得られる．これは，

$$\begin{pmatrix} T^{-1} + T^{-1}BS^{-1}CT^{-1} & -T^{-1}BS^{-1} \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} T & B \\ C & D \end{pmatrix} = \begin{pmatrix} I_{X_1} & 0 \\ 0 & I_{X_2} \end{pmatrix}$$

を意味する．逆に

$$\begin{aligned} &\begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} T^{-1} + T^{-1}BS^{-1}CT^{-1} & -T^{-1}BS^{-1} \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \\ &= \begin{pmatrix} I_{X_1} & T^{-1}B + T^{-1}BS^{-1}CT^{-1}B - T^{-1}BS^{-1}D \\ 0 & -S^{-1}CT^{-1}B + S^{-1}D \end{pmatrix} \\ &= \begin{pmatrix} I_{X_1} & T^{-1}B - T^{-1}BS^{-1}(D - CT^{-1}B) \\ 0 & -S^{-1}(D - CT^{-1}B) \end{pmatrix} = \begin{pmatrix} I_{X_1} & 0 \\ 0 & I_{X_2} \end{pmatrix} \end{aligned}$$

となるため，定義 16 と定理 3 から

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix}$$

は単射である．また，任意の $g_1 \in X_1$ と $g_2 \in X_2$ に対して，

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

は

$$\begin{aligned} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} &= \begin{pmatrix} I_{X_1} & -T^{-1}B \\ 0 & I_{X_2} \end{pmatrix} \begin{pmatrix} T^{-1} & 0 \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} Pg \\ (I-P)g \end{pmatrix} \\ &= \begin{pmatrix} T^{-1} + T^{-1}BS^{-1}CT^{-1} & -T^{-1}BS^{-1} \\ -S^{-1}CT^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} g_1 \\ g_2 \end{pmatrix} \end{aligned}$$

となる解

$$\begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} \in X_1 \times X_2$$

を持つため,

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix} : X_1 \times X_2 \rightarrow X_1 \times X_2$$

は全射でもある.

続いて, L が単射であることを示す. 定理 14 から $L\phi = 0$ において, 解が $\phi = 0$ だけであることを示せば良い. X が X_1 と X_2 の代数的直和であることから, 解 ϕ は $P\phi$ と $(I - P)\phi$ に一意に分解できる. その上,

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} P\phi \\ (I - P)\phi \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

となる. ここで,

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix}$$

が全単射であることから, 解は $P\phi = 0, (I - P)\phi = 0$ のみである. よって, $L\phi = 0$ において, 解は $\phi = 0$ のみである.

最後に, L が全射であることを示す. 任意の $g \in X$ に対して $L\phi = g$ を満たす解 $\phi \in X$ が存在すれば, 定義 15 から L が全射であることがいえる. X が X_1 と X_2 の代数的直和であることから, 解 ϕ は $P\phi$ と $(I - P)\phi$ に一意に分解できる. その上,

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} P\phi \\ (I - P)\phi \end{pmatrix} = \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix}$$

となる. ここで,

$$\begin{pmatrix} T & B \\ C & D \end{pmatrix}$$

が全単射であることから, 解 $(P\phi, (I - P)\phi) \in X_1 \times X_2$ は常に存在する. よって, $\phi = P\phi + (I - P)\phi \in X$ であるため, 任意の $g \in X$ に対して解 $\phi \in X$ は存在する. ■

定理 19. $X, X_1, X_2, L, g, P, \phi_1, \phi_2, T, B, C, D$ はすべて本節で定義したものとする. 線形作用素 T を全単射であると仮定する. そのとき,

$$S \text{ が全単射} \Leftrightarrow L \text{ が全単射}$$

証明. 「 S が全単射 $\Rightarrow L$ が全単射」定理 18 でいえる.

「 S が全単射 $\Leftarrow L$ が全単射」を示す. L が全単射であることから, $L\phi = 0$ を満たす解は $\phi = 0$ のみである. また, X が X_1 と X_2 の代数的直和であることから, 解 $\phi = 0$ は $P\phi = 0$ と $(I - P)\phi$ に一意に分解できる. その上, T が全単射であることを利用すると式 (5.2) が得られるため, 以下のようになる:

$$\begin{aligned} L\phi = 0 &\Leftrightarrow \begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} P\phi \\ (I - P)\phi \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ &\Leftrightarrow \begin{pmatrix} I_{X_1} & T^{-1}B \\ 0 & S \end{pmatrix} \begin{pmatrix} P\phi \\ (I - P)\phi \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \end{aligned}$$

上記の方程式を満たす解が, $P\phi = 0$ と $(I - P)\phi = 0$ のみのため,

$$\begin{pmatrix} I_{X_1} & T^{-1}B \\ 0 & S \end{pmatrix}$$

は単射である. ここで, S が単射でないと仮定して, 矛盾を示す. S が単射でないことから,

$$S\phi_2 = 0$$

を満たす 0 以外の解 $\phi_2 \in X_2$ が存在する. そのうえで,

$$\phi_1 = -T^{-1}B\phi_2$$

とすると (ϕ_1, ϕ_2) は方程式

$$\begin{pmatrix} I_{X_2} & T^{-1}B \\ 0 & S \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

の解となる. しかし, $L\phi = 0$ を満たす解は $\phi = 0$ のみであり,

$$\begin{pmatrix} I_{X_2} & T^{-1}B \\ 0 & S \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

を満たす解は 0 のみであるため, 矛盾する. よって, S は単射である.

続いて, L が全単射のとき, S が全単射であることを示す. L が全単射であるため, 任意の $g \in X$ に対して

$$\phi = L^{-1}g$$

となる解 $\phi \in X$ が一意に存在する. よって, 代数的直和による分解の一意性から $P\phi \in X_1$ と $(I - P)\phi \in X_2$ が一意に存在する. そのうえで,

$$\begin{aligned} L\phi = 0 &\Leftrightarrow \begin{pmatrix} T & B \\ C & D \end{pmatrix} \begin{pmatrix} P\phi \\ (I - P)\phi \end{pmatrix} = \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix} \\ &\Leftrightarrow \begin{pmatrix} I_{X_1} & T^{-1}B \\ 0 & S \end{pmatrix} \begin{pmatrix} P\phi \\ (I - P)\phi \end{pmatrix} = \begin{pmatrix} T^{-1} & 0 \\ -CT^{-1} & I_{X_2} \end{pmatrix} \begin{pmatrix} Pg \\ (I - P)g \end{pmatrix} \end{aligned}$$

となる. さらに, 第二式

$$S(I - P)\phi = -CT^{-1}Pg + (I - P)g$$

となる. ここで, 任意の $g \in X$ に対して $L\phi = g$ となる解 $\phi \in X$ が存在し, $\phi = P\phi + (I - P)\phi$ のように一意に分解できることに注意すると, 任意の $g_2 \in X_2$ に対しても $L\phi = g_2$ となる解 $\phi \in X$ が存在し, $\phi = P\phi + (I - P)\phi$ となる $(I - P)\phi \in X_2$ が存在する. そのうえで,

$$S(I - P)\phi = -CT^{-1}Pg_2 + (I - P)g_2 = g_2$$

となる解 $(I - P)\phi \in X_2$ が存在するため, S は全射である. ■

第 6 章

提案手法

6.1 定義

定義 41 (許容重みなし Banach 空間 X). Banach 空間 X を次のように定める. はじめに l^1 空間を次のように定める.

$$l^1 := \left\{ a = (a_k)_{k \in \mathbb{Z}} : a_k \in \mathbb{C}, \|a\| := \sum_{k \in \mathbb{Z}} |a_k| < \infty \right\}$$

そして, 検証に用いる関数空間 X は

$$X := \mathbb{C} \times l^1, x = (\omega, a), \omega \in \mathbb{C}, a \in l^1$$

と定め, そのノルムを

$$\|x\|_X := \max\{|\omega|, \|a\|\}$$

として定義する. このとき, X は Banach 空間となる.

定義 42 (拡張したヤコビ行列 $DF(x)$). まず, 近似解 x について

$$x = (\omega, \underbrace{0, \dots, 0}_M, a, \underbrace{0, \dots, 0}_M) \in \mathbb{C}^{2N+2M}$$

と定めると, $F(x)$ のヤコビ行列は

$$DF(x) = \left[\begin{array}{c|ccc} 0 & \cdots & 1 & \cdots \\ \vdots & & \vdots & \\ \partial_w f_k & \cdots & \partial_{a_j} f_k & \cdots \\ \vdots & & \vdots & \end{array} \right]$$

と定義できる.

定義 43 (作用素 A_M の定義). はじめに,

$$\bar{x} = (\omega, \underbrace{0, \dots, 0}_M, a, \underbrace{0, \dots, 0}_M) \in \mathbb{C}^{2N+2M}$$

と定めると, A_M の定義は, 以下のように定義できる.

$$A_M = \left(\begin{array}{c|ccc} DF(\bar{x})^{-1} & 0 & \cdots & \cdots \\ \hline 0 & \lambda_N^{-1} & & 0 \\ \vdots & & \lambda_{N+1}^{-1} & \\ \vdots & 0 & & \ddots \end{array} \right)$$

6.2 計算手順

$A = DF(\bar{x})^{-1}$ とおき, $\|AF(\bar{x})\| \leq Y_0$ に代入する. 節 5.2 より,

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

となる. ここで,

$$\phi := DF(\bar{x})^{-1}F(\bar{x})$$

とおくと,

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

と変形できる. Π_N を射影演算子とすると,

$$\begin{aligned} \begin{pmatrix} \Pi_N DF(\bar{x})\Pi_N & \Pi_N DF(\bar{x})(I - \Pi_N) \\ (I - \Pi_N)DF(\bar{x})\Pi_N & (I - \Pi_N)DF(\bar{x})(I - \Pi_N) \end{pmatrix} \begin{pmatrix} \Pi_N \phi \\ (I - \Pi_N)\phi \end{pmatrix} \\ = \begin{pmatrix} \Pi_N F(\bar{x}) \\ (I - \Pi_N)F(\bar{x}) \end{pmatrix} \end{aligned}$$

となり, 両辺に A_M を掛けて

$$\begin{aligned} A_M \begin{pmatrix} \Pi_N DF(\bar{x})\Pi_N & \Pi_N DF(\bar{x})(I - \Pi_N) \\ (I - \Pi_N)DF(\bar{x})\Pi_N & (I - \Pi_N)DF(\bar{x})(I - \Pi_N) \end{pmatrix} \begin{pmatrix} \Pi_N \phi \\ (I - \Pi_N)\phi \end{pmatrix} \\ = A_M \begin{pmatrix} \Pi_N F(\bar{x}) \\ (I - \Pi_N)F(\bar{x}) \end{pmatrix} \end{aligned}$$

となる. ここで, 線形作用素 T, B, C, E それぞれに対し,

$$\begin{aligned} T &:= \Pi_N ADF(\bar{x})|_{X_1}: X_1 \rightarrow X_1, & B &:= \Pi_N ADF(\bar{x})|_{X_2}: X_2 \rightarrow X_1, \\ C &:= (I - \Pi_N)ADF(\bar{x})|_{X_1}: X_1 \rightarrow X_2, & E &:= (I - \Pi_N)ADF(\bar{x})|_{X_2}: X_2 \rightarrow X_2 \end{aligned}$$

と定義でき, 式 (6.2) を次のように変形できる.

$$\begin{pmatrix} T & B \\ C & E \end{pmatrix} \begin{pmatrix} \Pi_N \phi \\ (I - \Pi_N)\phi \end{pmatrix} = \begin{pmatrix} \Pi_N AF(\bar{x}) \\ (I - \Pi_N)AF(\bar{x}) \end{pmatrix}$$

線形作用素 S を, 定理 18 より, 線形作用素 S を

$$S := E - CT^{-1}B: X_2 \rightarrow X_2$$

とする．もし， S が全単射ならば，

$$\begin{pmatrix} \Pi_N \phi \\ (I - \Pi_N) \phi \end{pmatrix} = \begin{pmatrix} T^{-1} + T^{-1} B S T^{-1} & -T^{-1} B S^{-1} \\ -S^{-1} C T^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} P g \\ (I - P) g \end{pmatrix}$$

となり，有界線形作用素 $ADF(\bar{x})$ は全単射となる．

S について計算する．式 (6.2) より，

$$\begin{aligned} S &= E - C T^{-1} B \\ &= ((I - \Pi_N) ADF(\bar{x})) - ((I - \Pi_N) ADF(\bar{x})) (\Pi_N ADF(\bar{x}))^{-1} ((I - \Pi_N) ADF(\bar{x})) \end{aligned}$$

となる．ここで，

$$\|I_{X_2} - S\| < 1$$

であれば， S は全単射となる．

第 7 章

実験結果

7.1 実験環境

本研究を行った実験環境を表 7.1, 使用したパッケージを表 7.2 を示す.

表 7.1 実験環境

環境	詳細
CPU	12th Gen Intel(R) CORE(TM) i7-12700
OS	Linux(x86_64-linyc-gnu)
Julia	Version 1.11.2

表 7.2 使用パッケージ

パッケージ	バージョン
DifferentialEquations	v7.10.0
FFMPEG	v0.4.1
FFTW	v1.7.1
ForwardDiff	v0.10.36
GenericFFT	v0.1.6
IntervalArithmetic	v0.20.9
Plots	v1.39.0

7.2 実験結果

提案手法に対して, フーリエ・スペクトル法におけるフーリエ係数のサイズを変更し, 次のノルム値

$$\|I_{X-2} - S\| < 1$$

を満たすかを検証した.

実験結果の数値を表 7.3 に, グラフにしたものを図 7.1 に示す.

結果より, フーリエ係数のサイズが大きくなるにつれてノルム値が小さくなることがわかる. また, いかなるフーリエ係数のサイズであっても, ノルム値が 1 を超えることはなかったことがわかる. これは, フーリエ係数のサイズが大きくなるにつれて, 計算における λ^{-1} が小さくなるため, ノルム値が小さくなるためであると考えられる.

表 7.3 フーリエサイズとノルム値

フーリエサイズ	$\ I_{X_2} - S\ $
50	0.22815114629236252
100	0.11455533660051737
150	0.07655718822651922
200	0.05749210273025131

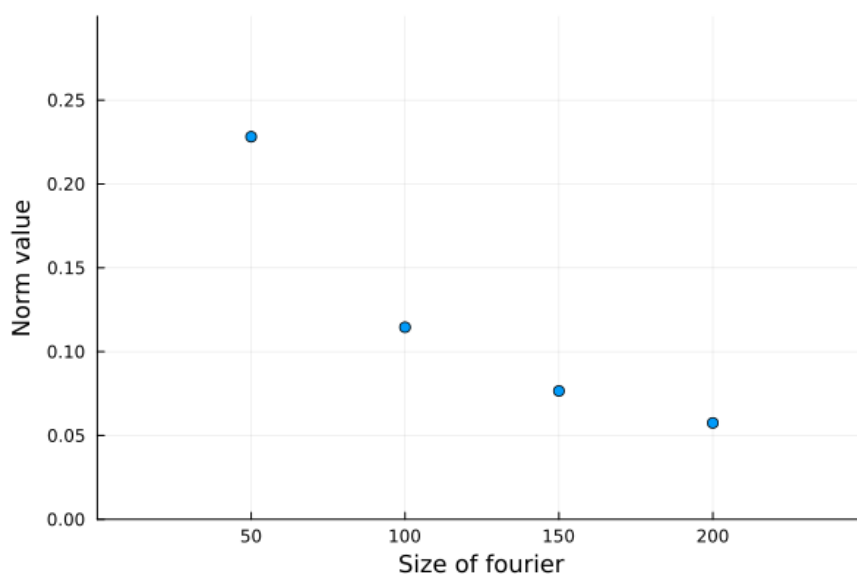


図 7.1 フーリエサイズとノルム値のグラフ

第 8 章

おわりに

本研究では, radii polynomial approach を用いた精度保証付き数値計算の問題適用範囲の拡大を目的とした. radii polynomial approach における作用素の計算に, 無限次元ガウスの消去法を用いることで, l_ν^1 を l_1 と定義し, 許容重みを定義に付加せずに解を導出できるかを検証した. プログラムによる検証により, 無限次元ガウスの消去法を用いることで radii polynomial approach の精度の向上が可能であることがわかった.

今後の課題として, 本研究では触れなかった評価値 $Z_0, Z_1, Z_2(r)r$ に対して, 無限次元ガウスの消去法を用いて計算ができるか, また無限次元ガウスの消去法を用いた計算手法を導出することについて, 検討をする.

謝辞

本研究を進めるに際して、千葉工業大学の関根晃太准教授には多くのご指導，ご助言を頂きましたこと深く感謝いたします。そして最後に、本研究に対しご助力頂いたすべての皆様に感謝し、お礼申し上げることで謝辞とさせていただきます。

参考文献

- [1] 大石進一, 精度保証付き数値計算, コロナ社 (2018)
- [2] 船越康太, 井藤佳奈子, 大谷俊輔, 近藤慎佑, 高橋和暉, 瀬戸翔太, 二平泰知, 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”
https://github.com/tak-lab/rigorous_numerics_tutorial_julia, (2024/11/29)
- [3] 関根晃太, 中尾充宏, 大石進一, Numerical verification methods for a system of elliptic PDEs, and their software library
- [4] Kouta Sekine, Mitsuhiro T. Nakao, and Shin’ichi Oishi: “A new formulation using the Schur complement for the numerical existence proof of solutions to elliptic problems: without direct estimation for an inverse of the linearized operator” , Numer. Math., 146, 907–926, Oct., 2020.

ソースコード 1: calc_solve.jl

```

1  using LinearAlgebra, DifferentialEquations, FFTW
2  include("FourierChebyshev.jl")
3
4  # van der 方程式Pol
5  function vanderpol(du, u,  $\mu$ , t)
6      x, y = u
7      du[1] = y
8      du[2] =  $\mu * (1 - x^2) * y - x$ 
9  end
10
11 #  $F^{(N)}(x_n)$ 
12 function F_fourier(x,  $\mu$ ,  $\eta^{e2^{82^{80}}}$ )
13     N = length(x) / 2
14      $\omega$  = x[1]
15     a = x[2:end]
16     ( $a^{c2^{b3}}$ , ~) = powerconvfourier(a, 3)
17     eta = sum(a) -  $\eta^{e2^{82^{80}}}$ 
18
19     k = -(N - 1):(N-1)
20     f = (-k.^2 *  $\omega^2$  -  $\mu * im * k * \omega$  .+ 1) .* a +  $\mu * im * k * \omega$ 
        .*  $a^{c2^{b3}} / 3$ 
21
22     return [eta; f]
23 end
24
25 function DF_fourier(x,  $\mu$ )
26     N = Int((length(x)) / 2)
27      $\omega$  = x[1]
28     a = x[2:end]
29     k = (-N+1):(N-1)
30     ( $a^{c2^{b3}}$ , ~) = powerconvfourier(a, 3)
31
32     DF = zeros(ComplexF64, 2N, 2N)
33
34     DF[1, 2:end] .= 1
35     DF[2:end, 1] = (-2 *  $\omega * k$  .^ 2 -  $\mu * im * k$ ) .* a +  $\mu$ 
        * im * k .*  $a^{c2^{b3}} / 3$ 
36
37     (~, a2) = powerconvfourier(a, 2)
38
39     M = zeros(ComplexF64, 2 * N - 1, 2 * N - 1)
40
41     for j = (-N+1):(N-1)
42         M[k.+N, j+N] =  $\mu * im * k * \omega$  .* a2[k.-j.+(2*N-1)]
43     end
44
45     L = diagm(-k.^2 *  $\omega^2$  -  $\mu * im * k * \omega$  .+ 1)
46
47     DF[2:end, 2:end] = L + M
48     return DF
49 end
50

```



```

51 # a function of fourier coeffs
52 function odefouriercoeffs(f, N, I, n=1)
53     a = I[1]
54     b = I[2]
55     # x_j: equidistance node points
56     h = (b - a) / (2N - 1)
57     j = 0:2N-2
58     x2j = a + j * h
59     # f_j: function values on node points
60     f2j = f(x2j)[n, :]
61     return (fftshift(fft(f2j))) / (2 * N - 1)
62 end
63
64 # たぶん, 初期値設定
65 u_0 = [0.0; 2.0]
66 tspan = (0.0, 300)μ
67 μ = 1.0
68 prob = ODEProblem(vanderpol, u_0, tspan, μ)
69 sol = solve(prob, Tsit5(), reltol=1e-8, abstol=1e-8)
70 u = hcat(sol.u...)
71 ind = floor(Int, length(sol.t) / 2)
72
73 # おおよその周期
74 #a = 30
75 #b = 36.55
76 a = 30
77 app_period = 6.55
78 timestep = 0.1
79
80 f_tmp = sol(a+app_period/2:timestep:a+3*app_period/2)
81 find_period = abs.(f_tmp .- sol(a))
82 (~, ind) = findmin(find_period[1, :])
83 b = a + app_period / 2 + timestep * (ind - 1)
84 #calc fouriercoeffs
85
86 N = 200 # size of Fourier
87
88 println("size of Fourier = $N")
89 a_0 = odefouriercoeffs(sol, N, [a, b])
90
91 include("IntervalFunctions.jl")
92 # Initial value of Newton method η
93 _0 = 0.0
94 x = [2 * pi / (b - a); a_0]
95
96 # Newton iteration
97 tol = 5e-12
98 F = F_fourier(x, μ, η_0)
99 println("Before step #1, ||F||_1 = $(norm(F,1))")
100 num_itr = 0
101
102 while num_itr < 100
103     global x = x - DF_fourier(x, μ) \ F

```

```

104 global num_itr += 1
105 global F = F_fourier(x,  $\mu$ ,  $\eta_0$ )
106 # println("After step #$(num_itr), ||F||_1 = $(norm(F,1))")
107 if norm(F, 1) < tol
108     break
109 end
110 end
111
112 # A^(N)
113 #ix = map(interval, x)
114 # $\omega i^{cc^{84}}$  = map(interval, real(x[1]))
115 # $i^{c4^{81}}$  = map(interval, x[2:end])
116
117 ix = x $\omega$ 
118  $i^{cc^{84}}$  = real(x[1])
119  $i^{c4^{81}}$  = x[2:end]
120
121 function DF_fourier(x::Vector{Complex{Interval{T}}},  $\mu$ ) where {T}
122     N = Int((length(x)) / 2)
123      $\omega$  = x[1]
124     a = x[2:end]
125     k = (-N+1):(N-1)
126
127     println("pcf Input: ", a, size(a), typeof(a))
128
129     ( $a^{c2^{b3}}$ , ~) = powerconvfourier(a, 3)
130     DF = zeros(Complex{Interval{T}}, 2N, 2N)
131     DF[1, 2:end] .= 1
132     DF[2:end, 1] = (-2 *  $\omega$  * k.^2 -  $\mu$  * im * k) .* a +  $\mu$ 
133         * im * k .*  $a^{c2^{b3}}$  / 3
134     (~, a2) = powerconvfourier(a, 2)
135     M = zeros(Complex{Interval{T}}, 2 * N - 1, 2 * N - 1)
136     for j = (-N+1):(N-1)
137         M[k.+N, j+N] =  $\mu$  * im * k *  $\omega$  .* a2[k.-j.+(2*N-1)]
138     end
139     L = diagm(-k.^2 *  $\omega^2$  -  $\mu$  * im * k *  $\omega$  .+ 1)
140     DF[2:end, 2:end] = L + M
141     return DF
142 end
143
144 iDF = DF_fourier(ix,  $\mu$ );
145 iA = inv(iDF) # map(Interval, inv(mid.(iDF)))
146
147 ## =====
148 ## I get a and omega by x.
149 ## =====
150 ""
151 println("x = $x")
152 println( $\mu$  =  $\mu$ $")
153 println("N = $N") $\mu$ 
154
155 ""

```

```

156
157 omega = x[1]
158 a = x[2:end]
159 mu =  $\mu$ 
160
161 extend_size1 = (3 * N + 1)
162 extend_size2 = extend_size1 * 2 + 1
163 topleft_size = extend_size1
164
165 # define DF[]
166 zero_padding = zeros(ComplexF64, Int(extend_size2))
167 extend_x = vcat(omega, zero_padding, a, zero_padding)
168 extend_DF2 = DF_fourier(extend_x, mu)
169
170
171 # define A
172 zero_padding = zeros(ComplexF64, Int(extend_size1))
173 extend_x = vcat(omega, zero_padding, a, zero_padding)
174 extend_DF1 = DF_fourier(extend_x, mu)
175 T_inv = inv(extend_DF1)
176
177 ## lambda 行列
178 topleft_size = size(T_inv)[1]
179 DF_bottomright = extend_DF2[topleft_size+1:end, topleft_size+1:end]
180 inv_lambda = inv(diagm(diag(DF_bottomright)))
181
182 # norm_D
183 D = inv_lambda * DF_bottomright
184 I_minus_D = 1.0I(size(D)[1]) - D
185 normI_minus_D = maximum(sum(abs.(I_minus_D), dims=1))
186
187 # norm_C
188 DF_bottomleft = extend_DF2[topleft_size+1:end, 1:topleft_size]
189 C = inv_lambda * DF_bottomleft
190 normC = maximum(sum(abs.(C), dims=1))
191
192 # norm_B
193 DF_topright = extend_DF2[1:topleft_size, topleft_size+1:end]
194 B = T_inv * DF_topright
195 normB = maximum(sum(abs.(B), dims=1))
196 normDF_topright = maximum(sum(abs.(DF_topright), dims=1))
197 normT_inv = maximum(sum(abs.(T_inv), dims=1))
198
199 println("||I - D|| = ", normI_minus_D)
200 println("||C|| = ", normC)
201 println("||T^-1|| = ", normT_inv)
202 println("||B|| = ", normB)
203
204 norm_result = normI_minus_D + normC * normT_inv * normB
205
206 println("||I - D|| + ||C|| ||T^-1|| ||B|| = ", norm_result)

```

ソースコード 2: FourierChebyshev.jl

```

1  ### Fourier functions
2  using FFTW, Plots
3  function fouriercoeffs(f, N, I=[0,  $\pi$ 2])
4      # f: any periodic function on I
5      # N: size of Fourier series
6      a = I[1]
7      b = I[2]
8      h = (b - a) / (2N - 1)
9      j = 0:2N-2
10     x2b1bc = a .+ j * h
11     f2b1bc = f.(x2b1bc)
12     return fftshift(fft(f2b1bc)) / (2N - 1)
13 end
14
15 function odefouriercoeffs(f, N, I, n=1)
16     a = I[1]
17     b = I[2]
18     h = (b - a) / (2N - 1)
19     j = 0:2N-2
20     x2b1bc = a .+ j * h
21     f2b1bc = f(x2b1bc)[n, :]
22     return fftshift(fft(f2b1bc)) / (2N - 1)
23 end
24
25 function plot_fourier(bc, I=[0,  $\pi$ 2])
26     # bc: Fourier coefficients
27     a = I[1]
28     b = I[2]
29     N = (length(bc) + 1) / 2 # 2N-1
30     n_pad = 500
31     bc_pad = [zeros(n_pad); bc; zeros(n_pad)]
32     N_pad = N + n_pad
33     h_pad = (b - a) / (2N_pad - 1)
34     xj_pad = a .+ h_pad * (0:2N_pad-2)
35     fNj_pad = real((2N_pad - 1) * ifft(ifftshift(bc_pad)))
36     plot(xj_pad, fNj_pad, legend=false, xlabel="\$x\$", ylabel="\$f(x)\$")
37 end
38
39 function plot_fourier!(bc, I=[0,  $\pi$ 2]; label="")
40     # bc: Fourier coefficients
41     a = I[1]
42     b = I[2]
43     N = (length(bc) + 1) / 2 # 2N-1
44     n_pad = 500
45     bc_pad = [zeros(n_pad); bc; zeros(n_pad)]
46     N_pad = N + n_pad
47     h_pad = (b - a) / (2N_pad - 1)
48     xj_pad = a .+ h_pad * (0:2N_pad-2)
49     fNj_pad = real((2N_pad - 1) * ifft(ifftshift(bc_pad)))
50     plot!(xj_pad, fNj_pad, label=label)
51 end
52

```

```

53 function plot_fouriercoeffs(bc)
54     N = (length(bc) + 1) / 2 # 2N-1
55     plot(-N+1:N-1, abs.(bc), ylabel=:log10,
56         legend=false,
57         xlabel="\$k\$",
58         ylabel="\$|\\bar{c}_k\\$,|\\$",
59         title="Absolute values of Fourier coefficients"
60     )
61 end
62
63 function plot_solution(u, index) # u =  $\omega$ 
64     [, a_{-N+1}, ..., a_0, ..., a_{N-1}], length(u) = 2N
65     # index = 1: profile of solution
66     #           2: Fourier mode
67     #           3: phase profile
68      $\omega$  = real(u[1])
69     L =  $\pi^2$  /  $\omega$ 
70     a = u[2:end]
71     N = length(u) / 2 # N: size of Fourier
72     n_pad = 500
73     a_pad = [zeros(n_pad); a; zeros(n_pad)]
74     N_pad = N + n_pad
75     dx = L / (2 * N_pad - 1)
76     x = dx * (0:2*N_pad-2)
77     if index == 1
78         # Plot profile:
79         plot(plot_fourier(a, [0, L]),
80             # plot(x,real((2N_pad-1)*ifft(ifftshift(a_pad))),
81             xlabel="\$t\$",
82             ylabel="\$x\\$, (t)\$",
83             line=1.6,
84             title="Profile of solution",
85             size=(720, 400),
86             legend=false,
87         )
88     elseif index == 2
89         # Plot Fourier coefficients:
90         plot(plot_fouriercoeffs(a),
91             # plot((-N+1):(N-1),abs.(a),ylabel=:log10,
92             xlabel="\$k\$",
93             ylabel="\$|a_k\\$,|\\$",
94             line=1.6,
95             title="Absolute values of Fourier coefficients",
96             size=(720, 400),
97             legend=false,
98         )
99     elseif index == 3
100         # Plot phase:
101         k = (-N_pad+1):(N_pad-1)
102         plot(real((2N_pad - 1) * ifft(ifftshift(a_pad))), real((2N_pad - 1) * ifft(
103             ifftshift(a_pad .* (im * k *  $\omega$ 
104             )))),
105             xlabel="\$x(t)\$",
106             ylabel="\$\\dot{x}\\$, (t)\$",

```

```

104     line=1.6,
105     title="Phase plot of a numerical solution",
106     size=(720, 400),
107     legend=false,
108 )
109 end
110 end
111
112 function plot_solution!(u)
113     L =  $\pi^2$  / real(u[1])
114     a = u[2:end]
115     N = length(u) / 2
116     n_pad = 1000
117     a_pad = [zeros(n_pad); a; zeros(n_pad)]
118     N_pad = N + n_pad
119     k = (-N_pad+1):(N_pad-1)
120     dx = L / (2 * N_pad - 1)
121     x = dx * (0:2*N_pad-2)
122     plot!(real((2N_pad - 1) * ifft(ifftshift(a_pad))), real((2N_pad - 1) * ifft(
        ifftshift(a_pad .* (im * k)))), line=1.6,)
123 end
124
125 function powerconvfourier(a::Vector{Complex{T}}, p) where {T}
126     M = Int((length(a) + 1) / 2)
127     N = (p - 1) * M
128     ta = [zeros(N, 1); a; zeros(N, 1)] # 1. Padding zeros: size(ta) = 2pM-1
129     tb = ifft(ifftshift(ta)) # 2. IFFT of ta
130     tb^=e1^b5^96 = tb .^ p # 3. tb*^tb
131     c^=e1^b5^96 = fftshift(fft(tb^=e1^b5^96)) * (2.0 * p * M - 1)^(p - 1)
132     return c^=e1^b5^96[N+1:end-N], c^=e1^b5^96[p:end-(p-1)]# return (
        truncated, full) version
133 end
134
135
136 ### Chebyshev functions
137 function chebpts(n, a=-1, b=1) # n: order of Chebyshev polynomials
138     m = -n:2:n
139     x = sinpi.(m / (2 * n))
140     return (1.0 .- x) .* a / 2 + (1.0 .+ x) .* b / 2
141 end
142
143 function chebcoeffs(f, M, I=[-1, 1])
144     a = I[1]
145     b = I[2]
146     n = M - 1
147     cpts = chebpts(n, a, b)
148     fvals = f.(cpts)
149     FourierCoeffs = real(fft([reverse(fvals); fvals[2:end-1]]))
150     ChebCoeffs = FourierCoeffs[1:n+1] / n
151     ChebCoeffs[1] = ChebCoeffs[1] / 2
152     ChebCoeffs[end] = ChebCoeffs[end] / 2
153     return ChebCoeffs # return Two-sided Chebyshev
154 end

```

```

155
156 function chebcoeffs_complex(f, M, I=[-1, 1])
157     a = I[1]
158     b = I[2]
159     n = M - 1
160     cpts = chebpts(n, a, b)
161     fvals = f.(cpts)
162     FourierCoeffs = fft([reverse(fvals); fvals[2:end-1]])
163     ChebCoeffs = FourierCoeffs[1:n+1] / n
164     ChebCoeffs[1] = ChebCoeffs[1] / 2
165     ChebCoeffs[end] = ChebCoeffs[end] / 2
166     return ChebCoeffs # return Two-sided Chebyshev
167 end
168
169 function cheb(f, I=[-1, 1]; tol=5e-15, Nmax=10000)
170     a = I[1]
171     b = I[2]
172     m = 0.5 * (a + b)
173     r = 0.5 * (b - a)
174     x = rand(5)
175     x1 = m .+ x * r
176     x2 = m .- x * r
177     if f.(x1) ^e2^89^88 f.(x2)
178         odd_even = 1 # even function: 1
179     elseif f.(x1) ^e2^89^88 -f.(x2)
180         odd_even = -1 # odd function: -1
181     else
182         odd_even = 0 # otherwise: 0
183     end
184     i = 3
185     schbc = 0 # sampling chebyshev coefficients
186     while true
187         schbc = chebcoeffs(f, 2^i + 1, I)
188         if all(abs.(schbc[end-2:end]) .< tol) || (2^i + 1 > Nmax)
189             break
190         end
191         i += 1
192     end
193     M = findlast(abs.(schbc) .> tol)
194     cc = schbc[1:M]
195     # cc = chebcoeffs(f,M,I)
196     if odd_even == 1 # even function
197         cc[2:2:end] .= 0
198     elseif odd_even == -1 # odd function
199         cc[1:2:end] .= 0
200     end
201     return cc # return Two-sided Chebyshev
202 end
203
204 using GenericFFT
205 function bigcheb(f, I=[-1, 1]; tol=5e-15, Nmax=10000)
206     a = I[1]
207     b = I[2]

```

```

208 m = 0.5 * (a + b)
209 r = 0.5 * (b - a)
210 x = rand(5)
211 x1 = m .+ x * r
212 x2 = m .- x * r
213 if f.(x1) ^e2^89^88 f.(x2)
214     odd_even = 1 # even function: 1
215 elseif f.(x1) ^e2^89^88 -f.(x2)
216     odd_even = -1 # odd function: -1
217 else
218     odd_even = 0 # otherwise: 0
219 end
220 i = 3
221 schbc = 0 # sampling chebyshev coefficients
222 while true
223     schbc = chebcoeffs(f, big(2^i + 1), I)
224     if all(abs.(schbc[end-2:end]) .< tol) || (2^i + 1 > Nmax)
225         break
226     end
227     i += 1
228 end
229 M = findlast(abs.(schbc) .> tol)
230 cc = schbc[1:M]
231 # cc = chebcoeffs(f,M,I)
232 if odd_even == 1 # even function
233     cc[2:2:end] .= 0
234 elseif odd_even == -1 # odd function
235     cc[1:2:end] .= 0
236 end
237 return cc # return Two-sided Chebyshev
238 end
239
240 function cheb_complex(f, I=[-1; 1]; tol=5e-15, Nmax=10000)
241     i = 3
242     schbc = 0 # sampling chebyshev coefficients
243     while true
244         schbc = chebcoeffs_complex(f, 2^i + 1, I)
245         if all(abs.(schbc[end-2:end]) .< tol) || (2^i + 1 > Nmax)
246             break
247         end
248         i += 1
249     end
250     M = findlast(abs.(schbc) .> tol)
251     return schbc[1:M]
252 # return cc # return Two-sided Chebyshev
253 end
254
255 function plot_chebcoeffs(f)
256     zero_ind = findall(x -> x == 0, f)
257     f[zero_ind] .= f[zero_ind.+1]
258     plot(0:length(f)-1, abs.(f),
259         yscale=:log10,
260         title="Chebyshev coefficients",

```



```

261     xlabel="Degree of Chebyshev polynomial",
262     ylabel="Magnitude of coefficient",
263     size=(800, 400),
264     legend=false,
265 )
266 end
267
268 function clenshaw(a, x) # Clenshaw's algorithm
269     # a: (Two-sided) Chbyshev coefficients
270     # x: evaluating points in [-1,1]
271     n = length(a) - 1
272     bk1 = 0.0
273     bk2 = 0.0
274     x = 2x
275     for r = (n+1):-2:3
276         bk2 = x .* bk1 .- bk2 .+ a[r]
277         bk1 = x .* bk2 .- bk1 .+ a[r-1]
278     end
279     if isodd(n)
280         b2 = x .* bk1 .- bk2 .+ a[2]
281         bk2 = bk1 # b3
282         bk1 = b2
283     end
284     return -bk2 .+ 0.5x .* bk1 .+ a[1] # y = c(1) + .5*x.*bk1 - bk2;
285 end
286
287 function clenshaw_secondkind(a, x) # Clenshaw's algorithm
288     # a: (Two-sided) Chbyshev coefficients
289     # x: evaluating points in [-1,1]
290     n = length(a) - 1
291     bk0 = 0
292     bk1 = 0
293     for r = (n+1):-1:1
294         tmp = 2x .* bk0 .- bk1 .+ a[r]
295         bk1 = bk0
296         bk0 = tmp
297     end
298     return bk0 #.+ bk1*(-x)
299 end
300
301 function eval_cheb(a, x; I=[-1, 1])
302     # a: (Two-sided) Chbyshev coefficients
303     # x: evaluating points in domain
304      $\xi = 2 * (x .- I[1]) / (I[2] - I[1]) .- 1$ 
305     return clenshaw(a,  $\xi$ )
306 end
307
308 function eval_cheb_naive(ChebCoeffs_twosided, x; I=[-1, 1])
309     M = length(ChebCoeffs_twosided) # M: size of chebyshev
310     a = I[1]
311     b = I[2]
312     k = 0:M-1
313      $\xi = 2 * (x .- a) / (b - a) .- 1$ 

```

```

314 return cos.(Vector(k)' .* acosξ.()) * ChebCoeffs_twosided
315 end
316
317 function eval_cheb_bc(ChebCoeffs_twosided, x, n=200; I=[-1, 1]) # Barycentric
    interpolation formula
318 M = length(ChebCoeffs_twosided) # M: size of chebyshev
319 a = I[1]
320 b = I[2]
321 k = 0:M-1
322 ξ = 2 * (x - a) / (b - a) - 1
323 xc = (1.0 - ξ) * a / 2 + (1.0 + ξ) * b / 2 # Chebyshev points in [a,b]
324 fxc = cos.(Vector(k)' .* acosξ.()) * ChebCoeffs_twosided
325 valnum = length(x)
326 ξ = 2 * (x - a) / (b - a) - 1
327 # ξ = range(-1, stop=1, length=valnum)
328 # x = (1.0 - ξ) * a / 2 + (1.0 + ξ) * b / 2
329 λ = [1 / 2; ones(n - 1); 1 / 2] .* (-1) .^ (0:n)
330
331 numer = zeros(valnum)
332 denom = zeros(valnum)
333 exact = zeros(Bool, valnum)
334
335 for j = 1:n+1
336     xdiff = x - xc[j]
337     temp = λ[j] ./ xdiff
338     numer += temp * fxc[j]
339     denom += temp
340     exact[xdiff.==0] .= true
341 end
342
343 fx = numer ./ denom
344 jj = findall(exact)
345 fx[jj] = f.(x[jj])
346 fx[jj] = cos.(Vector(k)' .* acosξ.([jj])) * ChebCoeffs_twosided
347 return fx
348 end
349
350 function plot_cheb(ChebCoeffs_twosided; I=[-1, 1], title="", label="", legend=
    true) # Input: Two-sided Chebyshev
351 # M = length(ChebCoeffs_twosided) # M: size of chebyshev
352 # a = I[1]; b = I[2];
353 x = range(I[1], stop=I[2], length=5000)
354 # ξ = 2*(x-a)/(b-a) - 1
355 fx = eval_cheb(ChebCoeffs_twosided, x, I=I)
356 plot(x, fx, legend=legend, label=label, title=title, xlabel="\$x\$", ylabel=
    "\$f(x)\$")
357 end
358
359 function plot_cheb!(ChebCoeffs_twosided; I=[-1, 1], title="", label="", legend=
    true) # Input: Two-sided Chebyshev
360 # M = length(ChebCoeffs_twosided) # M: size of chebyshev
361 # a = I[1]; b = I[2];

```

```

362 x = range(I[1], stop=I[2], length=5000)
363 #  $\xi = 2*(x.-a)/(b-a) .- 1$ 
364 fx = eval_cheb(ChebCoeffs_twosided, x, I=I)
365 plot!(x, fx, legend=legend, label=label, title=title, xlabel="\$x\$", ylabel
      ="\$f(x)\$")
366 end
367
368 function chebdiff(a; I=[-1, 1])# Input is Two-sided
369     M = length(a)
370     b = zeros(M + 1)
371     for r = M-1:-1:1
372         b[r] = b[r+2] + 2 * r * a[r+1]
373     end
374     b[1] /= 2.0
375     return b[setdiff(1:end, end)] * (2 / (I[2] - I[1])) # Output is Two-sided
376 end
377
378 function chebdiff_oneside(a; I=[-1, 1])# Input is One-sided
379     M = length(a)
380     b = zeros(M + 1)
381     for r = M-1:-1:1
382         b[r] = b[r+2] + 2 * r * a[r+1]
383     end
384     return b[setdiff(1:end, end)] * (2 / (I[2] - I[1])) # Output is One-sided
385 end
386
387 function chebdiff_secondkind(a; I=[-1, 1])# Input is Two-sided
388     M = length(a)
389     b = zeros(M - 1)
390     for n = 0:M-2
391         b[n+1] = (n + 1) * a[n+2]
392     end
393     return b * (2 / (I[2] - I[1]))# Output is second kind (Two-sided)
394 end
395
396 function chebindefint(a; I=[-1, 1])# Input is Two-sided
397     M = length(a)
398     a_ext = zeros(M + 2)
399     a_ext[1] = 2 * a[1]
400     a_ext[2:M] = a[2:M]
401     A = zeros(M + 1)
402     for n = 1:M
403         A[n+1] = (a_ext[n] - a_ext[n+2]) / (2n)
404     end
405     # A[1] = sum(A[2:2:end]) - sum(A[3:2:end]) # takes the value 0 at the left
      endpoint
406     return A * (I[2] - I[1]) / 2
407 end
408
409 function chebint(a; I=[-1, 1])# Input is Two-sided
410     M = length(a)
411     n = 0:2:M-1
412     return sum(2a[1:2:end] ./ (1.0 .- n.^ 2)) * ((I[2] - I[1]) / 2)

```

```

413 end
414
415 function chebroots(a, I=[-1, 1]) # Input is two-sided Chebyshev
416     I_lo = I[1]
417     I_up = I[2]
418     n = length(a)
419     # create colleague matrix
420     du = [1; ones(n - 3) * 0.5]
421     dl = ones(n - 2) * 0.5
422     d = zeros(n - 1)
423     A = Tridiagonal(dl, d, du)
424     B = zeros(n - 1, n - 1)
425     B[end, :] = a[1:end-1]
426     C = A - (1 / (2 * a[n])) * B
427     x = eigvals(C)
428     ε = 100 * eps() * (I_up - I_lo) * 0.5
429     if I_lo == -1.0 && I_up == 1.0
430         return real(xε[(-1-.^^e2^^89^^a4real(x).^^e2^^89^^a41ε
431             +).&(imag(x).^^e2^^89^^880)])
432     else
433         x = real(xε[(-1-.^^e2^^89^^a4real(x).^^e2^^89^^a41ε
434             +).&(imag(x).^^e2^^89^^880)])
435         return (1.0 .- x) .* I_lo / 2 + (1.0 .+ x) .* I_up / 2
436     end
437 end
438
439 function endpoints_of_chheb(a) # Input is two-sided Chebyshev
440     n = length(a)
441     atm1 = dot((-1) .^ (0:n-1), a) # endpoint at -1
442     at1 = sum(a) # endpoint at 1
443     return atm1, at1
444 end
445
446 function chebmax(a, I=[-1, 1]) # Input is two-sided Chebyshev
447     M = length(a)
448     b = chebdiff(a)
449     x = chebroots(b)
450     fxc = eval_chheb(a, x)
451     # k = 0:M-1
452     # fxc = cos.((Vector(k))' .* acos.(x)) * a
453     ep = endpoints_of_chheb(a)
454     fvals = [ep[1]; fxc[1:end]; ep[2]]
455     x = [interval(-1); x; interval(1)]
456     ind = argmax(fvals)
457     return x[ind], fvals[ind]
458 end
459
460 function chebmin(a, I=[-1, 1]) # Input is two-sided Chebyshev
461     M = length(a)
462     b = chebdiff(a)
463     x = chebroots(b)
464     fxc = eval_chheb(a, x)
465     # k = 0:M-1
466     # fxc = cos.((Vector(k))' .* acos.(x)) * a

```

```

465 ep = endpoints_of_cheb(a)
466 fvals = [ep[1]; fxc[1:end]; ep[2]]
467 x = [interval(-1); x; interval(1)]
468 ind = argmin(fvals)
469 return x[ind], fvals[ind]
470 end

```

ソースコード 3: IntervalFunctions.jl

```

1 ### Interval Matrix Multiplication
2 function ufp(P)
3     u = 2.0^(-53)
4     ^^cf^^95 = 2.0^52 + 1
5     q = ^^cf^^95 * P
6     T = (1 - u) * q
7     return abs(q - T)
8 end
9
10 function succ(c)
11     s_min = 2.0^-1074
12     u = 2.0^-53
13     ^^cf^^95 = u * (1.0 + 2.0 * u)
14     if abs(c) >= (1.0 / 2.0) * u^(-2) * s_min # 2^(-969)(Float64)
15         e = ^^cf^^95 * abs(c)
16         succ = c + e
17     elseif abs(c) < (1.0 / 2.0) * u^(-1) * s_min # 2^(-1022)(Float64)
18         succ = c + s_min
19     else
20         C = u^(-1) * c
21         e = ^^cf^^95 * abs(C)
22         succ = (C + e) * u
23     end
24     return succ
25 end
26
27 function pred(c)
28     s_min = 2.0^-1074
29     u = 2.0^-53
30     ^^cf^^95 = u * (1.0 + 2.0 * u)
31     if abs(c) >= (1.0 / 2.0) * u^(-2) * s_min # 2^(-969)(Float64)
32         e = ^^cf^^95 * abs(c)
33         pred = c - e
34     elseif abs(c) < (1.0 / 2.0) * u^(-1) * s_min # 2^(-1022)(Float64)
35         pred = c - s_min
36     else
37         C = u^(-1) * c
38         e = ^^cf^^95 * abs(C)
39         pred = (C - e) * u
40     end
41     return pred
42 end
43
44 function mm_ufp(A_mid, B_mid) # A_mid, B_mid: Point matrix

```

```

45  u = 2.0^(-53)
46  realmin = 2.0^(-1022)
47  n = size(A_mid, 2)
48
49  if (2 * (n + 2) * u >= 1)
50      error("mm_ufp is failed!(2(n+2)u>=1)")
51  end
52  # C_mid = A_mid * B_mid;
53  # C_rad = (n+2) * u * ufp.(abs.(A_mid)*abs.(B_mid)) .+ realmin;
54  # return C_mid, C_rad;
55  return A_mid * B_mid, (n + 2) * u * ufp.(abs.(A_mid) * abs.(B_mid)) .+
      realmin
56 end
57
58 function imm_ufp(A_mid, A_rad, B_mid, B_rad) # A = <A_mid, A_rad>, B = <B_mid,
      B_rad>: Interval matrix
59  u = 2.0^(-53)
60  realmin = 2.0^(-1022)
61  n = size(A_mid, 2)
62
63  if (2 * (n + 2) * u >= 1)
64      error("mm_ufp is failed!(2(n+2)u>=1)")
65  end
66  # C, R = mm_ufp(A_mid, B_mid);
67  # C_mid = A_mid * B_mid;
68  R = (n + 2) * u * ufp.(abs.(A_mid) * abs.(B_mid)) .+ realmin
69
70  # T_1, T_2 = mm_ufp(abs.(A_mid), B_rad);
71  T1 = abs.(A_mid) * B_rad
72  T2 = (n + 2) * u * ufp.(T1) .+ realmin
73
74  # T_3 = succ.(abs.(B_mid)+B_rad);
75  T3 = succ.(abs.(B_mid) + B_rad)
76
77  # T_4, T_5 = mm_ufp(A_r, T_3);
78  T4 = A_rad * T3
79  T5 = (n + 2) * u * ufp.(T4) .+ realmin
80
81  rad_sum = R + T1 + T2 + T4 + T5
82
83  # C_rad = succ.(rad_sum + 4*u*ufp.(rad_sum));
84
85  # return C_mid, C_rad;
86  return A_mid * B_mid, succ.(rad_sum + 4 * u * ufp.(rad_sum))
87 end
88
89 # USE IntervalArithmetic.jl
90 using IntervalArithmetic
91 function int_mul(A::Matrix{T}, B::Matrix{T}) where {T}
92     Cmid, Crad = mm_ufp(A, B)
93     return interval(Cmid, Crad; format=:midpoint)
94     # return Cmid ± . Crad
95 end

```

```

96
97 function int_mul(A::Matrix{Interval{T}}, B::Matrix{T}) where {T}
98     Cmid, Crad = imm_uvp(mid.(A), radius.(A), B, zeros(size(B)))
99     return interval(Cmid, Crad; format=:midpoint)
100     # return Cmid ± . Crad
101 end
102
103 function int_mul(A::Matrix{T}, B::Matrix{Interval{T}}) where {T}
104     Cmid, Crad = imm_uvp(A, zeros(size(A)), mid.(B), radius.(B))
105     return interval(Cmid, Crad; format=:midpoint)
106     # return Cmid ± . Crad
107 end
108
109 function int_mul(A::Matrix{Interval{T}}, B::Matrix{Interval{T}}) where {T}
110     Cmid, Crad = imm_uvp(mid.(A), radius.(A), mid.(B), radius.(B))
111     return interval(Cmid, Crad; format=:midpoint)
112     # return Cmid ± . Crad
113 end
114
115 function int_mul(A::Matrix{Complex{T}}, B::Matrix{T}) where {T}
116     Ar = real.(A)
117     Ai = imag.(A) # (Ar + im*Ai)*B = Ar*B + im*(Ai*B)
118     return complex.(int_mul(Ar, B), int_mul(Ai, B))
119 end
120
121 function int_mul(A::Matrix{T}, B::Matrix{Complex{T}}) where {T}
122     Br = real.(B)
123     Bi = imag.(B) # A*(Br + im*Bi) = A*Br + im*(A*Bi)
124     return complex.(int_mul(A, Br), int_mul(A, Bi))
125 end
126
127 function int_mul(A::Matrix{Complex{T}}, B::Matrix{Complex{T}}) where {T}
128     Ar = real.(A)
129     Ai = imag.(A)
130     Br = real.(B)
131     Bi = imag.(B)
132     # (Ar + im*Ai)*(Br + im*Bi) = (Ar*Br - Ai*Bi) + im*(Ar*Bi + Ai*Br)
133     return complex.((int_mul(Ar, Br) - int_mul(Ai, Bi)), (int_mul(Ar, Bi) +
        int_mul(Ai, Br)))
134 end
135
136
137 ### Interval Linear system solver
138 import LinearAlgebra: opnorm
139 function LinearAlgebra.opnorm(a::Matrix{Complex{Interval{Float64}}}, p::String)
140     if p == "1"
141         suma = sum(abs.(a), dims=1)
142         return interval(maximum(inf, suma), maximum(sup, suma))
143     elseif p == "Inf"
144         suma = sum(abs.(a), dims=2)
145         return interval(maximum(inf, suma), maximum(sup, suma))
146     end
147     return NaN

```

```

148 end
149
150 function LinearAlgebra.opnorm(a::Matrix{Interval{Float64}}, p::String)
151     if p == "1"
152         suma = sum(abs.(a), dims=1)
153         return interval(maximum(inf, suma), maximum(sup, suma))
154     elseif p == "Inf"
155         suma = sum(abs.(a), dims=2)
156         return interval(maximum(inf, suma), maximum(sup, suma))
157     end
158     return NaN
159 end
160
161 function verifylss_iB(iA, iB) # verify the solution element-wisely
162     A = mid.(iA)
163     B = mid.(iB)
164     X^cc^84 = A \ B
165     n = size(X^cc^84, 2)
166     R = inv(A)
167     #####
168     iG = interval(Matrix{Float64}(I, n, n)) - interval(R) * iA
169      $\alpha$  = opnorm(iG, "Inf") # Interval arithmetic
170     #####
171     if sup $\alpha$ () < 1
172          $\eta$  = (abs.(iG) * interval(ones(n))) / (interval(1) -  $\alpha$ )
173         Err = interval(zeros(n, n))
174         X^cc^84 = interval(X^cc^84)
175         iR = iA * X^cc^84 - iB # Interval arithmetic
176         iRr = interval(R) * iR
177         for i = 1:n
178             Err[:, i] = abs.(iRr[:, i]) + interval(sup(norm(iRr[:, i], Inf))) *  $\eta$ 
179             # Interval arithmetic
180         end
181         return interval(X^cc^84, Err, format=:midpoint)
182     else
183         println("Oh my way, verification is failed...")
184         return nan
185     end
186 end
187
188 ### Interval all eigenvalues solver
189
190 function verifyalleig(iA, X)
191     n = size(iA, 2)
192     # iD = diagm(interval $\lambda$ ())
193     iX = interval(X)
194     iB = int_mul(iA, iX)
195     iG = verifylss_iB(iX, iB)
196     iR = interval(zeros(n))
197     iC = diag(iG)
198     for i = 1:n
199         for j = 1:n

```



```

200         if i != j
201             ir[i] += interval(mag(iG[i, j]))
202         end
203     end
204     ir[i] += interval(radius(ic[i]))
205 end
206 return interval(ic, ir, format=:midpoint)
207 end
208
209 ### Interval eigen solver (eigpair)
210
211 function verifyeig(iA::Matrix{Interval{T}}, lam, x, B=Matrix{T}(I, size(iA)))
212     where {T<:Real}
213     if typeof(lam) <: Complex || eltype(x) <: Complex
214         lam = convert(Complex{T}, lam)
215         x = convert.(Complex{T}, Vector(x))
216     else
217         lam = convert(T, lam)
218         x = convert.(T, Vector(x))
219     end
220     x = x ./ norm(x)
221     ysize = length(x)
222     ilam = interval(lam)
223     ix = interval(x)
224     iB = interval(B)
225
226     function iDF(w)
227         mat = (zeros(typeof(ilam), length(w), length(w)))
228         mat[1, 2:end] = transpose(interval(2) * (ix + w[2:end]))
229         mat[2:end, 1] = -iB * (ix + w[2:end])
230         mat[2:end, 2:end] = iA - (ilam + w[1]) * iB
231         return mat
232     end
233
234     zero = zeros(T, ysize + 1)
235     R = inv(mid.(iDF(zero)))
236     iR = interval(R)
237     iz = -iR * [dot(ix, ix) - interval(1.0); iA * ix - ilam * iB * ix]
238     ^^cf^^b5 = 2 * sup(norm(iz, 1))
239     if isreal(lam) && isreal(x)
240         lam = real(lam)
241         x = real(x)
242         id = interval(0, ^^cf^^b5; format=:midpoint)
243         iy = interval.(zeros(ysize), ^^cf^^b5; format=:midpoint)
244         iI = interval(Matrix{T}(I, ysize + 1, ysize + 1))
245     else
246         id = Complex(interval(0, ^^cf^^b5; format=:midpoint), interval(0, ^^cf^^b5;
247             format=:midpoint))
248         iy = Complex.(interval.(zeros(ysize), ^^cf^^b5; format=:midpoint), interval
249             .(zeros(ysize), ^^cf^^b5; format=:midpoint))
250         iI = interval(Matrix{Complex{T}}(I, ysize + 1, ysize + 1))
251     end
252     iw = [id; iy]
253     g(w) = iz + (iI - iR * iDF(w)) * w

```

```

250 gw = g(iw)
251 if all(issubset_interval.(gw, iw)) #gw ⊂ iw
252     # while maximum(radius, gw) / norm([lam; x], 1) >= 1e3 * eps(T)
253     #     gw = g(gw)
254     # end
255     return ilam + gw[1] #, ix .+ gw[2:end] 固有ベクトルも返すようにする#
256 else
257     return NaN
258 end
259 end
260
261 ### Verify FFT using Interval Arithmetic
262 function verifyfft(z::Vector{Interval{T}}, sign=1) where {T}
263     n = length(z)
264     col = 1
265     array1 = true
266     if n == 1
267         Z = map(T, z)
268         return Z
269     else
270         isrow_ = false
271     end
272     log2n = Int(round(log2(n))) #check dimension
273     if 2^log2n ≠ n # return error if n is not the powers of 2
274         error("length must be power of 2")
275     end
276     #bit-reversal
277     f = 2^(log2n - 1)
278     v = [0; f]
279     for k = 1:log2n-1
280         f = f >> 1
281         v = append!(v, f .+ v)
282     end
283     z2 = zeros(n, col)
284     if isa(real(z[1]), Interval)
285         z2 = map(T, z2)
286     end
287     # replace z
288     for j = 1:n
289         z2[j, :] = z[v[j]+1, :]
290     end
291     #Danielson-Lanczos algorithm
292     Z = complex(map(Interval, z2))
293     Index = reshape([1:n*col;], n, col)
294
295     theta = map(Interval, sign * (0:n-1) / n) # division exact because n is power
        of 2
296     itheta = map(interval, theta)
297     Phi = cospi.(itheta) + im * sinpi.(itheta) # SLOW?
298     # Phi = cospi.(theta) + im*sinpi.(theta)
299
300     v = [1:2:n;]
301     w = [2:2:n;]

```

```

302 t = Z[w, :]
303 Z[w, :] = Z[v, :] - t
304 Z[v, :] = Z[v, :] + t
305 for index in 1:(log2n-1)
306     m = 2^index
307     m2 = 2 * m
308     vw = reshape([1:n;], m2, Int(n / m2))
309     v = vw[1:m, :]
310     w = vw[m+1:m2, :]
311     indexv = reshape(Index[v[:], :], m, Int(col * n / m2))
312     indexw = reshape(Index[w[:], :], m, Int(col * n / m2))
313     Phi1 = repeat(Phi[1:Int(n / m):end], outer=[1, Int(col * n / m2)])
314     t = Phi1 .* Z[indexw]
315     Z[indexw] = Z[indexv] - t
316     Z[indexv] = Z[indexv] + t
317 end
318 reverse(Z[2:end, :], dims=2)
319 if sign == -1
320     Z = Z / n
321 end
322 if isrow_
323     Z = transpose(Z)#transpose of Z
324 end
325 if array1
326     Z = Z[:, 1]
327 end
328 return Z
329 end
330
331 ### Rigorous convolution algorithm via FFT
332 function powerconvfourier(ia::Vector{Complex{Interval{T}}}, p) where {T}
333     M = Int((length(ia) + 1) / 2) # length(a) = 2M-1
334     N = (p - 1) * M
335
336     length_ia = 2 * p * M - 1
337     length_ia_ext = nextpow(2, length_ia)# 2pM-2+2L
338
339     L = Int((length_ia_ext - length_ia + 1) / 2)
340
341     # step.1 : padding (p-1)M + L zeros for each sides
342     ia_ext = map(Complex{Interval}, zeros(length_ia_ext))
343     ia_ext[L+N+1:end-L-N+1] = ia #\tilde{a}
344
345     # step.2 : inverse fft
346     ib_ext = verifyfft(ifftshift(ia_ext), -1) #sign = -1 : ifft
347
348     # step.3 : power p elementwisely
349     ib_ext^^e1^^b5^^96 = ib_ext .^ p
350
351     # step.4 : fft with rescaling
352     ic_ext^^e1^^b5^^96 = fftshift(verifyfft(ib_ext^^e1^^b5^^96, 1)) *
        length_ia_ext^(p - 1) #sign = 1 : fft
353

```

```

354 # return ic_ext^^e1^^b5^^96,ic_ext^^e1^^b5^^96
355 return ic_ext^^e1^^b5^^96[L+N+1:end-N-L+1], ic_ext^^e1^^b5^^96[L+p:end-(L+p
    -2)] # return (truncated, full) version
356 end
357
358 function convfourier(ia...)
359     p = length(ia)
360     M = Int((length(ia[1]) + 1) / 2) # length(a) = 2M-1
361     N = (p - 1) * M
362
363     length_ia = 2 * p * M - 1
364     length_ia_ext = nextpow(2, length_ia)# 2pM-2+2L
365
366     # itb^^e1^^b5^^96 = ones(Interval,N+length(ia[1])+N)
367     # ibp_ext = map(Complex{Interval},ones(length_ia_ext))
368     ibp_ext = interval(ones(ComplexF64, length_ia_ext))
369
370     L = Int((length_ia_ext - length_ia + 1) / 2)
371
372     for i = 1:p
373         # step.1 : padding (p-1)M + L zeros for each sides
374         # ia_ext = map(Complex{Interval},zeros(length_ia_ext))
375         ia_ext = interval(zeros(ComplexF64, length_ia_ext))
376         @show length(ia[i])
377         @show length(ia_ext[L+N+1:end-L-N+1])
378         @show length(ia[i])
379         ia_ext[L+N+1:end-L-N+1] = ia[i] #\tilde{a}
380         # step.2 : inverse fft
381         ib_ext = verifyfft(fftshift(ia_ext), -1) #sign = -1 : ifft
382         # step.3 : power p elementwisely
383         ibp_ext = ibp_ext .* ib_ext
384         # ib_ext^^e1^^b5^^96 = ibp_ext.^p
385     end
386     # step.4 : fft with rescaling
387     ic_ext^^e1^^b5^^96 = fftshift(verifyfft(ibp_ext, 1)) * length_ia_ext^(p - 1)
388     #sign = 1 : fft
389     return ic_ext^^e1^^b5^^96[L+N+1:end-N-L+1], ic_ext^^e1^^b5^^96[L+p:end-(L+p
        -2)] # return (truncated, full) version
389 end
390
391 function convcos(ia...) # Input: Two-sided (real)
392     M = length(ia[1])
393     FourierCoeffs = []
394     for i = 1:length(ia)
395         ia[i][1] = ia[i][1]
396         ia[i][2:end] = 0.5 * ia[i][2:end] # Two-sided -> One-sided (real)
397         FC_local = map(Complex, [reverse(ia[i][2:end]); ia[i]])
398         if i == 1
399             FourierCoeffs = tuple(FC_local)
400         else
401             FourierCoeffs = tuple(FourierCoeffs..., tuple(FC_local)...)
402         end
403     end

```

```

404 icp, icp_full = convfourier(FourierCoeffs...) # real -> complex (input)
405 iap = interval(zeros(M))
406 iap[1] = real(icp[M])
407 iap[2:end] = 2 * real(icp[M+1:end]) # One-sided (complex) -> Two-sided (real)
408 N = Int((length(icp_full) + 1) / 2) #2N-1
409 iap_full = interval(zeros(N))
410 iap_full[1] = real(icp_full[N])
411 iap_full[2:end] = 2 * real(icp_full[N+1:end]) # One-sided (complex) -> Two-
    sided (real)
412 return iap, iap_full
413 end

```
