

平成 19 年度

修士論文

楕円型偏微分方程式の解に対する精度保証について

平成 20 年 2 月 4 日

指導教授： 大石 進一 教授

早稲田大学院 理工学研究科 情報・ネットワーク専攻

3606U036-8 木原 瞬

目次

1	序論	3
1.1	背景	4
1.2	本論文の目的	4
1.3	本論文の構成	4
2	数学的予備知識	5
2.1	はじめに	6
2.2	関数解析の予備知識	6
2.3	不動点定理	10
2.3.1	不動点定理について	10
2.3.2	Schauder の不動点定理について	10
2.4	浮動小数点数	11
2.5	浮動小数点数への丸めと四則演算	13
2.6	関数空間の定義	14
3	本論文で扱う方程式	16
3.1	はじめに	17
3.2	楕円型偏微分方程式	17
3.2.1	楕円型方程式	17
3.2.2	Poisson 方程式	17
3.2.3	扱う方程式	17
4	中尾先生の方法	18
4.1	はじめに	19
4.2	基本的な理論	19
4.3	概要	19
4.3.1	近似解と誤差	19
4.3.2	不動点定理の導出	20
4.3.3	解の存在の証明	20
4.3.4	計算機での表現	22

5	有限要素法	23
5.1	はじめに	24
5.2	有限要素法の考え方	24
5.3	2次元有限要素法	25
5.3.1	近似関数の構成	25
5.3.2	有限要素分割	25
5.3.3	面積座標について	27
5.3.4	要素マトリックスの計算	27
5.3.5	自然境界の一般化	29
5.3.6	近似方程式の組み立て	29
6	有限要素近似結果	32
6.1	はじめに	33
6.2	有限要素近似グラフ	33
6.3	有限要素近似結果	35
7	精度保証法	36
7.1	はじめに	37
7.2	有限次元の精度保証	37
7.3	無限次元の精度保証	38
8	精度保証結果	39
8.1	はじめに	40
8.2	有限次元	40
8.3	無限次元	40
8.4	有限次元+無限次元	40
8.5	実行時間	41
9	プログラム	42
9.1	はじめに	43
9.2	本論文で作成したプログラム	43
10	統括	46
10.1	はじめに	47
10.2	統括	47
10.3	結論	47
10.4	今後の展望	47
	謝辞	48
	参考文献	50

第 1 章

序論

1.1 背景

現在, 多様な現象をモデル化し, その現象を支配する基礎方程式を導き, それを解析していくことによって現象を解析していく研究が多面でなされている. そのため, 連続数学の問題を解くということはとても重要な位置を占めている. ところが, 連数数学の問題を解くことは難しい場合が多い. そこで, 計算機を利用して連続数学を解こうとする研究が始まり, それにともなって数値計算の理論, 研究に対して多くの人が携わりその結果, 順調に発展してきたといえる. また今日, コンピュータ技術も大幅な発展をし, いまや少し昔の大型計算機がパソコンとして個人で使える時代となった.

しかし, そのような時代になっても, 数値計算には次のような基本問題が残っている. 計算機を用いる数値計算では, それぞれの計算を正確に行わなければならない. しかし, 実際には, 四則演算の結果は丸められ, また, 無限演算は全て近似して行われる. つまり, 計算結果は常にある程度の誤差を伴っているということである. そこで, 計算結果から正しい結論を得るためには, この計算結果に含まれている誤差を評価して真の値の範囲を確定することが必要になる.

従来, 数値計算を行っている人々の間ではこの点についてあまり取り上げられることがなかった. それは, 数値計算によって上記のように数学的に正しいことを保証するためには, 計算コストや手間ひまがかかるといった理由のためであろう.

しかし, 近年では数値計算の誤差解析の精度を厳密に保証する手法が飛躍的に発達し, 多くの場合, 近似解の精度保証が近似解を得る手間の数倍程度で可能であることがわかった. また, それに加えて計算機の高速化も伴っていくため, 近い将来, 数値計算において, 任意に精度保証を付加することが日常的になると考えられる.

1.2 本論文の目的

本論文の目的は, 楕円型偏微分方程式の解に対する精度保証を行うことである. 偏微分方程式の解の一意存在を証明する代表的な手法として様々な手法が知られている. その中で中尾先生の方法を用い, 今まで研究されていない方程式に適用する.

1.3 本論文の構成

本論文の構成は以下の通りである. 第2章では, 本論文で扱う数学的予備知識について述べる. 第3章では, 本論文で扱う方程式について述べ, 第4章では, 中尾先生の方法について述べる. 第5章では, 有限要素法について述べる. 第6章では, 有限要素法近似の数値実験結果を示す. 第7章では, 精度保証法について述べる. 第8章では, 精度保証の数値実験結果を示す. 第9章では, 本論文で作成したプログラムを紹介する. そして, 第10章では, 本論文の総括を行う.

第 2 章

数学的予備知識

2.1 はじめに

この章では, 中尾先生の方法 [3] で偏微分方程式の精度保証を行う為に用いられる数学的予備知識について説明する.

2.2 関数解析の予備知識

中尾先生の手法で扱う関数空間を理解するために, 以下の定義について記述する.

部分空間

ある構造を持った集合 X について, それを空間と呼ぶとき部分空間とは, その構造を保つような X の部分集合あるいは, 構造を保つように X に埋め込まれた別の集合 A のことをいう

距離空間

集合 X 上に 2 変数実数値の写像 d , すなわち X の二つの直積 $X \times X$ 上で定義され次数に値をとる関数

$$d : X \times X \rightarrow \mathbb{R} : (x, y) \rightarrow d(x, y)$$

が定義されていて, x, y, z を X の任意の点としたときに, d が距離の公理とよばれる次のような性質

1. 非負性 : $d(x, y) \geq 0$
2. 同一性 $d(x, y) = 0 \Leftrightarrow x = y$
3. 対称性 : $d(x, y) = d(y, x)$
4. 三角不等式 : $d(x, y) + d(y, z) \geq d(x, z)$

を満たすならば, d は X 上の距離関数であるといい, 対 (X, d) を距離空間と呼ぶ.

ノルム空間

K を実数体 \mathbb{R} または複素数体 \mathbb{C} とし, K 上のベクトル空間 V を考える. このとき任意の $a \in K$ と任意の $u, v \in V$ に対して,

1. 正値性: $\|v\| \geq 0$
2. $\|v\| = 0 \Leftrightarrow v = 0$
3. 斉次性: $\|av\| = |a|\|v\|$
4. 三角不等式: $\|u + v\| \leq \|u\| + \|v\|$

を満たすような関数 $\|\cdot\| : V \rightarrow \mathbb{R}; x \rightarrow \|x\|$ を V のノルムと呼ぶ.
また, ノルムの定義された空間をノルム空間という.

不動点定理について理解するために, 以下の定義について記述する.

連続

連続性は各点の周りで考えられる概念である. 一変数関数 $f(x)$ がある点 x_0 で連続であるとは, x が x_0 に限りなく近づけば, $f(x)$ が $f(x_0)$ に限りなく近づくことをいう.

$$\lim_{x \rightarrow x_0} f(x) = f(x_0)$$

凸集合

C を実, または複素ベクトル空間とする. C が凸集合であるとは, C に含まれる任意の x, y と区間 $[0, 1]$ に含まれる任意の t について, 点 $(1-t)x + ty$ が C に含まれることを言う. 言い換えれば, x と y を結ぶ線分が C に含まれることである.

写像

集合 A の各元に対してそれぞれ集合 B の元をただひとつずつ指定するような規則 f が与えられているとき, f を A から B への写像であるといい,

$$f : A \rightarrow B$$

と表す. また, A の元 a が f によって B の元 b に移されるとき, b を f による a の像と呼び, $b = f(a)$ と表す.

全射・単射・全単射

$f : A \rightarrow B$ について, $f(A) = B$ が成り立つとき, f を全射という. また, 任意の A の元 a_1, a_2 に対して, $a_1 \neq a_2$ ならば $f(a_1) \neq f(a_2)$ が成り立つとき, f を単射という. さらに, 全射で単射な写像のことを全単射という.

逆写像

f を A から B への全単射とする. $f(a) = b$ によって, b を a に対応させると, f は全射の為, 全ての b がある a に対応していて, f が単射であることからそのような a は一つしかないことが分かる. こうして作られる写像を f の逆写像という.

連続写像

また, 距離空間 X から距離空間 Y への写像 f が X の各点で連続であるとき, f を連続写像という.

同相写像

位相空間 A から位相空間 B への連続写像 f が全単射でその逆写像も連続であるとき, f を同相写像という.

埋め込み

ある写像 f が単射, 連続とする. その像 $f(x)$ に限定すると, f は全単射になるため, 逆像が連続ならば $f : X \rightarrow f(x)$ は同相写像になる. このとき, f を X から $f(x)$ への埋め込みという.

完備・コーシー列

無限点列 $\{x_n\}_{n=1}^{\infty}$ について

$$\lim_{n,m \rightarrow \infty} \|x_n - x_m\| = 0$$

が成立するとき, 点列 $\{x_n\}_{n=1}^{\infty}$ はコーシー列であるという.

また, 距離空間 M が完備であるとは, 距離空間 M におけるいかなるコーシー列も M 内の点に収束することである.

被覆

集合 S に対し, I を添字集合とする S の部分集合族 $\{U_i\}_{i \in I}$ が $S = \bigsqcup_{i \in I} U_i$ を満たすとき,

集合族 $\{U_i\}_{i \in I}$ を集合 S の被覆と呼ぶ.

また, 添字集合 I が有限集合である場合を有限被覆, U_i が全て開集合である場合を開被覆, さらに, $V_j = U_k$ となるような任意の j とある k が J の元となるとき, $\{V_j | j \in J\}$ を $\{U_i | i \in I\}$ の部分被覆という. 開集合からなる部分被覆を部分開被覆, 有限被覆となる部分被覆を有限部分被覆という. 同様に開被覆からなる有限部分被覆は有限部分開被覆という.

コンパクト

位相空間 X の部分集合 A について, A のどんな開被覆にも有限部分被覆が存在するとき, A はコンパクトであるという.

作用素

作用素とは, 関数空間上の変換, すなわち関数を別の関数にうつす写像のことである.

コンパクト作用素

X, Y をノルム空間とし, $Y_0 \subset Y$ とする. 線形作用素 $T : X \rightarrow Y_0$ について, X の任意の有界点列 $\{u_n\} (n = 1, 2, \dots)$ に対し, それらの部分列 $\{u_{n_i}\}$ をうまく選んだとき, 点列 $\{Tu_{n_i}\} (n, i = 1, 2, \dots)$ が必ず Y のある点に収束するとき, T はコンパクト作用素という.

ニュートン法

数値計算の分野において, ニュートン法とは方程式を数値計算によって解くための反復解法の1つである. 方程式に対する条件は領域における微分可能性と2次微分に関する符号だけであり, 方程式系の線形性等は特に要求しない. 収束の早さも2次収束なので古くから数値計算で使用されている.

直交補空間

線形空間 V とその部分空間 W に対して, $w \in W$ を任意に定めたとき, W の直交補空間を次のように定義する

$$W^\perp = \{v | (v, w) = 0\}$$

2.3 不動点定理

2.3.1 不動点定理について

X をノルム空間とする. 連続写像 $T : X \rightarrow X$ に対し $Tx = x$ を満たす $x \in X$ を T の不動点という.

また, 適当な条件の下に不動点の存在を主張する定理を不動点定理という.

不動点定理は方程式 $F(x) = 0$ (ただし, $F : X \rightarrow X$ は連続とし, $x \in X$ とする.) の解の存在と関係している. なぜなら $T = I - F$ と置けば, x が $F(x) = 0$ の解であることと x が T の不動点であることは同値であるからである.

2.3.2 Schauder の不動点定理について

X をノルム空間とする. S を X のコンパクト凸集合とする. このとき連続写像 $T : S \rightarrow S$ は不動点をもつ.

(proof)

k を自然数とし $\epsilon_k = \frac{1}{k}$ と置く. Theorem 1 より, S はコンパクトであるから, 全有界であり, S 内の点からなる有限 ϵ_k ネット x_1, \dots, x_n が存在する.

Theorem 1 S が距離空間の相対コンパクト集合ならば S は全有界である. したがってコンパクト集合は全有界である.

このとき,

$$S \subseteq \bigsqcup_{i=1}^n B(x_i, \epsilon_k), \quad B(x_i, \epsilon_k) = \{x \in X \mid \|x - x_i\| < \epsilon_k\}$$

ここで

$$S_k = \left\{ \sum_{i=1}^n \lambda_i x_i \mid \lambda_i \geq 0 \ \forall i, \sum_{i=1}^n \lambda_i = 1 \right\}$$
$$\phi_i(x) = \begin{cases} \epsilon_k - \|x - x_i\| & (x \in B(x_i, \epsilon_k)) \\ 0 & (x \notin B(x_i, \epsilon_k)) \end{cases}$$

と置けば,

$$\sum_{i=1}^n \phi_i(x) > 0 \quad (x \in S)$$

となる. さらに,

$$\mu = \phi_i(x) / \sum_{j=1}^n \phi_j(x) \quad (i = 1, 2, \dots, n)$$

$$P_k(x) = \sum_{i=1}^n \mu_i x_i \quad (x \in S)$$

と置くことにより, 写像 $P_k : S \rightarrow S_k$ が定義される. ϕ_1, \dots, ϕ_n は x の連続関数であるから P_k は S 上連続である. したがって

$$P_k T : S \rightarrow S_k$$

も連続である. $S_k \subset S$ であるから, $P_k T$ の S_k 上への制限を Φ とすれば, $\Phi : S_k \rightarrow S_k$ は連続かつ S_k は有限次元空間内の有界閉凸集合である. ゆえに, Theorem 2 によって Φ は S_k 内に不動点 y_k を持つ.

Theorem 2 \mathbb{R}^n の有界閉凸集合を D とするとき, 連続写像 $T : D \rightarrow D$ は不動点をもつ.

このとき $\{y_k\}$ は S 内の点列で S はコンパクトと仮定しているから, S の元に収束する部分列 $\{y_{k_j}\}_{j=1}^{\infty}$ を含む. その収束先を y とすれば y は T の不動点である. 以下にこのことを示す. $x \in S$ ならば

$$\begin{aligned} P_k x - x &= \sum_{i=1}^n \mu_i x_i - x = \sum_{i=1}^n \mu_i (x_i - x) \\ \|P_k x - x\| &\leq \sum_{i=1}^n \mu_i \|x_i - x\| \leq \epsilon_k \sum_{i=1}^n \mu_i = \epsilon_k \end{aligned}$$

ゆえに

$$\begin{aligned} \|T y - y\| &\leq \|T y - T y_{k_j}\| + \|T y_{k_j} - y_{k_j}\| + \|y_{k_j} - y\| \\ &= \|T y - T y_{k_j}\| + \|T y_{k_j} - \Phi y_{k_j}\| + \|y_{k_j} - y\| \\ &= \|T y - T y_{k_j}\| + \|T y_{k_j} - P_{k_j} T y_{k_j}\| + \|y_{k_j} - y\| \\ &< \|T y - T y_{k_j}\| + \epsilon_{k_j} + \|y_{k_j} - y\| \rightarrow 0 \quad (j \rightarrow \infty) \end{aligned}$$

よって $\lim_{j \rightarrow \infty} T y_{k_j} = y$ となつて $y \in S$ は T の不動点である.

2.4 浮動小数点数

IEEE754 に基づく浮動小数点数システムは, 浮動小数点数の集合とその上の演算によって定義される. IEEE754 によって規定される浮動小数点数としては, 規格化二進浮動小数点数, 零, 非規格化二進浮動小数点数, NaN の4つのタイプの数が用意されている.

規格化二進浮動小数点数

$$a = \pm \left(\frac{1}{2} + \frac{d_2}{2^2} + \frac{d_3}{2^3} + \dots + \frac{d_N}{2^N} \right) \times 2^e, \quad d_i = 0 \text{ or } 1$$

と書ける数をいう. e_{\min} を負の整数, e_{\max} を正の整数として, e は $e_{\min} \leq e \leq e_{\max}$ となる整数である.

$$m = \pm \left(\frac{1}{2} + \frac{d_2}{2^2} + \frac{d_3}{2^3} + \dots + \frac{d_N}{2^N} \right)$$

を符号付き仮数 (signed mantissa) といい, e を指数 (exponent) という. 指数 e も二進数で表される. 通常, 単精度 (4 byte = 32 bit), 倍精度 (8 byte = 64 bit), 拡張倍精度 (16 byte = 128 bit) の浮動小数点システムがあるが, それらは次のような浮動小数点システムである.

$$\begin{aligned} N = 24, & \quad -126 \leq e \leq 127 \\ N = 53, & \quad -1022 \leq e \leq 1023 \\ N = 64, & \quad -16382 \leq e \leq 16383 \end{aligned}$$

ただし, 拡張倍精度浮動小数点数は標準化されておらず, コンピュータによって異なる. したがって本論文では, 特に断ることのない限り, 浮動小数点数といえば, 倍精度浮動小数点のことである.

規格化二進浮動小数点システムにおいて表される数の絶対値の最大値は

$$x_{\max} = \left(\frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^N} \right) \times 2^{e_{\max}}$$

であり, その最小値は

$$x_{\min} = \frac{1}{2} \times 2^{e_{\min}}$$

である. 倍精度ではそれぞれ

$$\begin{aligned} \left(\frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^{53}} \right) \times 2^{1023} &= 1.79769 \cdots \times 10^{308} \\ \frac{1}{2} \times 2^{-1022} &= 2.22507 \cdots \times 10^{-308} \end{aligned}$$

である. $|x| > x_{\max}$ のときにオーバ フロー (overflow) が生じたという.

倍精度浮動小数点数においては, 仮数部が 53 ビットであるから

$$2^{-53} = 1.110223 \cdots \times 10^{-16}$$

より倍精度浮動小数点数は十進法で約 16 桁の精度がある.

零

零は規格化され

$$+ \left(\frac{0}{2} + \frac{0}{2^2} + \frac{0}{2^3} + \cdots + \frac{0}{2^N} \right) \times 2^{e_{\min}}$$

と表される.

非規格化二進浮動小数点数

IEEE754 では, 浮動小数点数は指数部が e_{\min} となったとき, 仮数部の最初の桁が 1 より小さい数を表すために, デフォルトで最初の桁を 1 とすることをやめ, ここが 0 となる

数を置くことを許す規格となっている．これを，非規格化数 (denormalized number) という．非規格化数の範囲に数が入ることを漸化アンダーフロー (gradual underflow) という．このような非規格化浮動小数点数の正で最も小さな数は

$$2^{-1074} = 4.940656 \cdots \times 10^{-324}$$

これ以下の数になると，アンダーフロー (underflow) が生じたという．

NaN

この他に，IEEE 754 では次のような特別な数が用意されている．

- NaN (Not a Number) は $\sqrt{-5}$, $\frac{\infty}{\infty}$, $+\infty$, $+(-\infty)$ などの不当な演算の結果として得られる．
- $\pm\infty$ はオーバーフローの結果や零で割った結果として得られる．
- ± 0 はアンダーフローから $\pm\infty$ での割り算の結果として得られる．

本論文では，以上の四つのタイプの数のうち NaN を除いたもの，すなわち規格化浮動小数点数，零，非規格化浮動小数点数の集合を \mathbb{F} で表す． \mathbb{F} の要素の数は有限個で，区間 $[-x_{\max}, x_{\max}]$ の上に原点对称に分布する．実数 x が与えられた時，それを挟む二つの浮動小数点数の距離を $\text{ulp}(x)$ で表す．ulp は unit in the last place のことである．

2.5 浮動小数点数への丸めと四則演算

IEEE 754 では，浮動小数点数の集合 \mathbb{F} 上での演算は丸めを用いて定義されている．

また，次の四つの丸めのモードを指定できるように，コンピュータが設計されていることを要請している．これは，ほとんどすべてのコンピュータで実現されている． c を実数とする．

- 上向きの丸め (round upward) $\triangle : \mathbb{R} \rightarrow \mathbb{F}$
 c 以上の浮動小数点数の中で最も小さい数に丸める．
- 下向きの丸め (round downward) $\nabla : \mathbb{R} \rightarrow \mathbb{F}$
 c 以下の浮動小数点数の中で最も大きい数に丸める．
- 最近点への丸め (round to nearest) $\square : \mathbb{R} \rightarrow \mathbb{F}$
 c に最も近い浮動小数点数に丸める．もしも，このような点が 2 点ある場合は，仮数部の最終ビットが偶数である浮動小数点数に丸める．これを偶数への丸めという．
- 切り捨て (round toward 0)
絶対値が c 以下の浮動小数点数の中で最も c に近いものに丸める．

丸めの演算を写像として $\circ : \mathbb{R} \rightarrow \mathbb{F}$ と書く。すなわち， \circ は Δ, ∇, \square のいずれかと考える。IEEE 754 では，丸めの演算はつぎの条件をみたすことが要請されている。

$$\begin{aligned} \circ x &= x && (\text{任意の } x \in \mathbb{F} \text{ について}) \\ x \leq y &\implies \circ x \leq \circ y && (\text{任意の } x, y \in \mathbb{R} \text{ について}) \end{aligned}$$

また， $x \in \mathbb{F}$ のとき，符号を変えることにより， $-x$ や $|x|$ が得られるので，これらは正確に計算される。

IEEE 754 では，次の性質が成立することも要請されている。

$$\begin{aligned} \square(-x) &= -\square x && (\text{任意の } x \in \mathbb{R} \text{ について}) \\ \Delta(-x) &= -\Delta x && (\text{任意の } x \in \mathbb{R} \text{ について}) \\ \nabla(-x) &= -\nabla x && (\text{任意の } x \in \mathbb{R} \text{ について}) \end{aligned}$$

IEEE 754 では，浮動小数点数演算 (\mathbb{F} 上での四則演算) は丸めとの関係により次のように定義されている。 $\cdot \in \{+, -, \times, /\}$ ， $\circ \in \{\Delta, \nabla, \square\}$ のとき，

$$x \circledast y = \circ(x \cdot y) \quad (\text{任意の } x, y \in \mathbb{R} \text{ について})$$

この式は，左辺の浮動小数点数の四則演算の結果 $x \circledast y$ は，右辺の数学的に正しい実数としての四則演算の結果 $x \cdot y$ を指定された丸めを行なって得られた数 $\circ(x \cdot y)$ に一致するように計算する規格を表している。

また，平方根も

$$(\sqrt{x})_{fp} = \circ(\sqrt{x}) \quad (\text{任意の } x \in \mathbb{F} \text{ について})$$

と，浮動小数点演算によって計算された $(\sqrt{x})_{fp}$ は，正確な実数演算で計算された平方根 \sqrt{x} で指定された丸め方向へ丸めた数となる規格である。注意すべきことは，指数関数や三角関数などはこのような規格をみたすようには規格されていないことである。これは，これら初等関数の値を精度保証付きで求めるためには別途工夫が必要であることを意味する。

また，十進数を二進数に変換すること，また，その逆の変換をすることは，しばしば数値計算で必要となる。IEEE 754 では，そのような基数の変換に関する規定もあるが，精度保証の観点からでは，それは不十分である。

十進数整数は二進整数へ誤差無しで変換できること，および十進小数を二進小数へ変換すると誤差が生じることに注意する。

2.6 関数空間の定義

m を非負整数とし， $\Omega \subset \mathbb{R}^2$ を有界領域とする。本論文が扱う関数空間 $L^2(\Omega)$, $H^m(\Omega)$, $H_0^1(\Omega)$ は標準的なものをつかう。すなわち

$$\begin{aligned} L^2(\Omega) &:= \{v \mid \int_{\Omega} |v|^2 dx dy < \infty\}, \\ H^m(\Omega) &:= \{v \in L^2(\Omega) \mid \partial^\alpha v \in L^2(\Omega), |\alpha| \leq m\}, \\ H_0^1(\Omega) &:= \{v \in H^1(\Omega) \mid v = 0 \text{ on } \partial\Omega\}. \end{aligned}$$

$H_0^1(\Omega)$ の内積を

$$(u, v)_{H_0^1} := (\nabla u, \nabla v) \quad u, v \in H_0^1(\Omega)$$

と定義する. ここで,

$$(\nabla u, \nabla v) = \int_{\Omega} (\partial_x u \partial_x v + \partial_y u \partial_y v) dx dy.$$

とおいた. このとき, $H_0^1(\Omega)$ がノルム空間になることはよく知られている.

また S_h としてパラメータ h に依存する $H_0^1(\Omega)$ の有限次元部分空間とし, S_h^\perp を S_h の直交補空間, すなわち

$$S_h^\perp = \{v \in H_0^1(\Omega) \mid (v, u)_{H_0^1} = 0 \quad \forall u \in S_h\}$$

とする. さらに射影 P_h を $H_0^1(\Omega)$ から S_h への射影とする.

第 3 章

本論文で扱う方程式

3.1 はじめに

この章では, 本論文で扱う楕円型偏微分方程式を紹介する. 中尾先生の方法により, これまで扱われていなかった偏微分方程式について考える.

3.2 楕円型偏微分方程式

3.2.1 楕円型方程式

二階線形偏微分方程式の分類において, ラプラシアンを含み, 時間についての偏微分を含まない方程式の総称である. 各係数の関係式により, 他に放物線型や双曲線型がある.

3.2.2 Poisson 方程式

Poisson 方程式とは, ラプラシアン Δ とある関数 f を用いて書かれる偏微分方程式の総称であり, $\Delta u = f(u)$ の形をとる. また, 偏微分方程式の分類において楕円型方程式の代表例といえる.

3.2.3 扱う方程式

$\Omega = (0, 1) \times (0, 1)$ に対して, 次の非線形楕円型偏微分方程式を考える:

$$\begin{cases} -\Delta u + u = f(u) & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega. \end{cases} \quad (3.1)$$

ここで,

$$\Delta = \partial_x^2 + \partial_y^2$$

$$\Omega = (0, 1) \times (0, 1)$$

$$f = (8\pi^2 + 1) \sin 2\pi x \sin 2\pi y$$

である.

第 4 章

中尾先生の方法

4.1 はじめに

この章では, 偏微分方程式の解に対する精度保証の手法の一つである, 中尾先生の方法について説明する.

4.2 基本的な理論

対象とする方程式を不動点形式に変換し, 真の解を含む可能性のある関数の集合を適切に設定した上で, Schauder の不動点定理を適用する. そして, 解の存在範囲を数学的に得ることを目指す.

4.3 概要

4.3.1 近似解と誤差

次の方程式について考える.

$$\begin{cases} -\Delta u = f(u) & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega. \end{cases} \quad (4.1)$$

ここで, 有限要素法を用いて,

$$(\nabla u_h, \nabla \phi_h) = (f(u_h), \phi_h) \quad \forall \phi_h \in S_h$$

を満たす 4.1 の有限要素近似解 $u_h \in S_h$ を見つける.

このとき u_h は

$$\begin{cases} -\Delta \hat{u} = f(u_h) & \text{in } \Omega, \\ \hat{u} = 0 & \text{on } \partial\Omega. \end{cases} \quad (4.2)$$

の解 \hat{u} の S_h への H_0^1 射影となっている. すなわち,

$$u_h = P_h \hat{u},$$

$$(\nabla \hat{u} - \nabla u_h, \nabla \phi_h) = 0 \quad \forall \phi_h \in S_h.$$

よって 4.1, 4.2 より

$$\begin{cases} -\Delta(u - \hat{u}) = f(u) - f(u_h) & \text{in } \Omega, \\ u - \hat{u} = 0 & \text{on } \partial\Omega. \end{cases} \quad (4.3)$$

となる. ここで

$$w = u - \hat{u}$$

とおくと

$$\begin{cases} -\Delta w = f(\hat{u} + w) - f(u_h) & \text{in } \Omega, \\ w = 0 & \text{on } \partial\Omega. \end{cases} \quad (4.4)$$

と書ける. つまり 4.1 を解く代わりに 4.4 を解くことを考える.

4.3.2 不動点定理の導出

ここで一つ補助定理を紹介する.

Lemma 1

任意の $\psi \in L^2(\Omega)$ に対して,

$$\begin{cases} -\Delta\phi = \psi & \text{in } \Omega \\ \phi = 0 & \text{on } \partial\Omega \end{cases}$$

を満たすような $\phi \in H^2(\Omega) \cap H_0^1(\Omega)$ が一意に存在することが知られている.

このときの $\psi \in L^2(\Omega)$ に対して, $\phi \equiv A\psi$ と定義する.

$$A : L^2(\Omega) \rightarrow H_0^1(\Omega) \cap H^2(\Omega)$$

であるが, 埋め込み $H^2(\Omega) \hookrightarrow H_0^1(\Omega)$ は compact なので,

$$A : L^2(\Omega) \rightarrow H_0^1(\Omega) \quad \text{compact}$$

となる. いま, 非線形作用素 F を

$$F(w) \equiv A(f(\hat{u} + w) - f(u_h))$$

として定義すれば F は $H_0^1(\Omega)$ から $H_0^1(\Omega)$ への compact 作用素となる.

これにより

$$w = F(w) \quad (4.5)$$

と書ける.

4.3.3 解の存在の証明

4.5 を次のように有限次元の部分と無限次元の部分に分けて考える.

$$\begin{aligned} P_h w &= P_h F(w) \\ (I - P_h)w &= (I - P_h)F(w) \end{aligned} \quad (4.6)$$

4.6 の上の式に Newton Method を用い, それを $P_h N(w)$ とおく, すなわち

$$P_h N(w) \equiv P_h w - [P_h - P_h F'(u_h)]^{-1} (P_h w - P_h F(w))$$

と定義する.

ここで, $P_h N(w)$ に現れる作用素の仮定をしておく.

Assumption 1

$F'(u_h)$ を F の u_h での Frechet 微分とすると, 写像 $(P_h - P_h F'(u_h)) : H_0^1(\Omega) \rightarrow S_h$ を S_h に制限したものは逆作用素 $[P_h - P_h F'(u_h)]^{-1} : S_h \rightarrow S_h$ をもつ.

これは $(\nabla(P_h - P_h F'(u_h))\phi_i, \nabla\phi_j)$ が可逆であることが必要十分条件であることを意味している.

いま作用素 T を

$$T(w) \equiv P_h N(w) + (I - P_h)F(w)$$

と定義すると, 次の Proposition が成り立つ.

Proposition 1

$$w = T(w) \Leftrightarrow w = F(w)$$

(proof)

$w = T(w)$ より,

$$\begin{cases} w_h = P_h N(w) \\ w_h^\perp = (I - P_h)F(w) \end{cases}$$

となる $w_h \in S_h, w_h^\perp \in S_h^\perp$ が一意に存在し, $w = w_h + w_h^\perp$ と書ける. すると, 上式より,

$$w_h = P_h w - [P_h - P_h F'(u_h)]^{-1} (P_h w - P_h F(w))$$

よって,

$$P_h w = P_h F(w)$$

したがって,

$$\begin{aligned} w &= w_h + w_h^\perp \\ &= P_h w + w_h^\perp \\ &= P_h F(w) + (I - P_h)F(w) \\ &= F(w) \end{aligned}$$

となる.

Schauder の不動点定理より

$$T(W) \equiv \{T(w) | w \in W\} \subset W$$

となる $H_0^1(\Omega)$ の有界凸閉部分集合 W が見つければ 4.5 を満たす $T(W)$ の元 w が一意に存在することが言える.

4.3.4 計算機での表現

そこで, $T(W) \subset W$ なる $W \subset H_0^1(\Omega)$ をどのようにして計算機内で表現するか考える. 作用素 $T(w)$ の構成に着目して無限次元である W を有限次元の部分 S_h と無限次元の部分 S_h^\perp に分ける. すなわち

$$\begin{aligned} W &= W_h \oplus W_h^\perp \\ W_h &\subset S_h \quad W_h^\perp \subset S_h^\perp \end{aligned}$$

となる.

そして, このときの検証条件は

$$\begin{aligned} T(W) &= P_h N(W) \oplus (I - P_h)F(W) \\ &\subset W_h \oplus W_h^\perp \\ &= W \end{aligned}$$

となればよいので,

$$\begin{aligned} P_h N(W) &\subset W_h \\ (I - P_h)F(W) &\subset W_h^\perp \end{aligned}$$

であることが示せればよい.

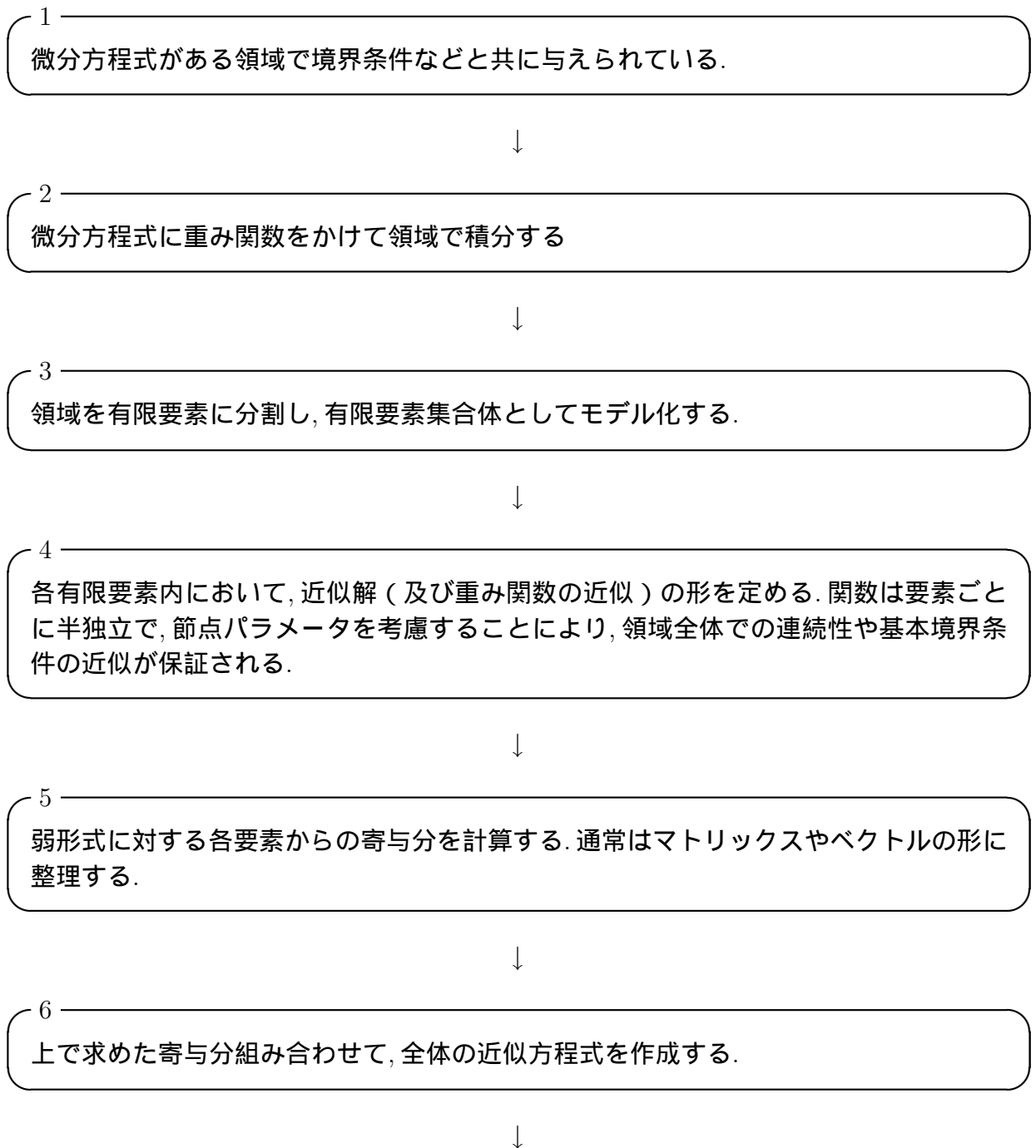
第 5 章

有限要素法

5.1 はじめに

この章では有限要素法による近似解の求め方について説明する。有限要素法は誤差解析が知られており精度保証ができるため、解析学において有限差分法と並びよく使われている。

5.2 有限要素法の考え方



得られた方程式を解けば未知の節点パラメータ値が決定され、したがって近似関数が求められる。

5.3 2次元有限要素法

5.3.1 近似関数の構成

$$\begin{aligned} \Omega & : 2 \text{次元領域} \\ \Gamma = \partial\Omega & : \text{境界 } (\Gamma_1 \cup \Gamma_2) \\ f & : \Omega \text{上の与えられた関数} \\ g_1 & : \Gamma_1 \text{上の与えられた関数} \end{aligned}$$

に対して次の方程式を考える.

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ u = g_1 & \text{on } \Gamma_1, \\ \frac{\partial u}{\partial \nu} = 0 & \text{on } \Gamma_2. \end{cases}$$

このとき,

$$\Delta = \partial_x^2 + \partial_y^2, \quad \partial \nu: \text{外向き法線方向微分.}$$

そして, 次を定義する.

$$\langle u, v \rangle = \int_{\Omega} \int_{\Omega} \nabla u \cdot \nabla v dx dy$$

$$(u, v) = \int_{\Omega} \int_{\Omega} \nabla u \nabla v dx dy$$

ここで,

$$\langle u, v \rangle = (u, v)$$

$$\forall v \in S = \{v \in H^1(\Omega) | v = 0 \text{ a.e. on } \Gamma_1\}$$

を満たす $u \in H^1(\Omega)$, $u = g_1$ a.e. on Γ_1 となるものを求めたい.

5.3.2 有限要素分割

様々な方法があるが, ここでは三角形要素による分割を考える. 重なったり隙間ができないように, さらに, 頂点が他の要素の辺上にこないようように分割しなければならない.

ここで, 次のような三角形有限要素 e を考える.

- 3頂点を節点とする. ここでは $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ をそれぞれ $1^*, 2^*, 3^*$ で表す.

- 節点上で近似関数値を考え、その値を各要素でお補間パラメータとして用いる。
- 全節点に対する通い番号と各要素ごとの局所的な番号とを考え、その対応を表にする。

$\alpha_1, \alpha_2, \alpha_3$ をパラメータとすると要素 e 内における近似関数 \hat{u} の形は

$$\hat{u} = \alpha_1 + \alpha_2 x + \alpha_3 y$$

となる。また、隣接要素間で \hat{u} は連続であるため、各節点の \hat{u} の値 u_i は要素間で共通である。そして、 e と e' で共通に u_i, u_j を用いれば、边上での \hat{u} の表現は一致する。

$$u_i = \alpha_1 + \alpha_2 x_i + \alpha_3 y_i$$

$$i = 1, 2, 3 \quad \text{局所番号}$$

ここで以下の式が成り立つ。

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \frac{1}{\begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}} \begin{bmatrix} x_2 y_3 - y_3 x_2 & x_3 y_1 - x_1 y_3 & x_1 y_2 - x_2 y_1 \\ y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

ここで、 $D = \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}$ は e の面積の 2 倍であり、

$$D = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)$$

ここで、

$$a_i^* = x_j y_k - x_k y_j, b_i^* = y_j - y_k, c_i^* = x_k - x_j$$

$$a_i = \frac{a_i^*}{D}, b_i = \frac{b_i^*}{D}, c_i = \frac{c_i^*}{D}$$

とし、また、 $(i, j, k) = (1, 2, 3)$ の偶置換とする。さらに、

$$\hat{u} = \sum_{i=1}^3 (a_i + b_i x + c_i y) u_i = \sum_{i=1}^3 L_i u_i$$

となり、 (L_1, L_2, L_3) を点 (x, y) の面積座標という。

5.3.3 面積座標について

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

に注意すれば,

$$L_i(x_i, y_i) = f_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

全要素に対して以上を行う. ここで, 基底関数 ψ_i (i は全体の通し番号) を定義する.

$$\hat{u} = \sum_{i=1}^N u_i \psi_i$$

$$\psi = L_1^* \quad \text{ine} \quad \text{if } i \leftrightarrow i^*$$

すなわち, ψ は i 上で 1, それ以外で 0 となる. さらに, ここで

$$S = |e| : e \text{ の面積}$$

$$L_i(x, y) = S_i/S \quad (i = 1, 2, 3)$$

とする. つまり,

$$L_1 + L_2 + L_3 = 1 \quad (\forall p \in e)$$

となる. さらに,

$$x = \sum_{i=1}^3 x_i L_i, \quad y = \sum_{i=1}^3 y_i L_i$$

なので, (x, y) から $X = L_1, Y = L_2$ と変数変換するときの Jacobian は,

$$\begin{vmatrix} \frac{\partial x}{\partial X} & \frac{\partial x}{\partial Y} \\ \frac{\partial y}{\partial X} & \frac{\partial y}{\partial Y} \end{vmatrix} = \begin{vmatrix} x_1 - x_3 & x_2 - x_3 \\ y_1 - y_3 & y_2 - y_3 \end{vmatrix} = D$$

5.3.4 要素マトリックスの計算

\hat{u} に対する近似方程式を得るために要素 e からの寄与分を算定する.

$$\langle u, v \rangle_e = \int \int_e \nabla u \cdot \nabla v \, dx dy \quad (f, v)_e = \int \int_e f v \, dx dy$$

↓

$$\langle u, v \rangle_e = \left\langle \sum_{i=1}^3 u_i L_i, \sum_{i=1}^3 v_i L_i \right\rangle_e$$

$$\begin{aligned}
&= \sum_{i=1}^3 v_j A_{ij}^{(e)} u_i \\
(f, \hat{v})_e &= (f, \sum_{i=1}^3 v_i L_i)_e \\
&= \sum_{i=1}^3 v_i f_i^{(e)}
\end{aligned}$$

ここで、要素パラメータベクトル、要素自由項ベクトル、要素係数マトリックスとして以下の
ように置く：

要素節点パラメータベクトル

$$\mathbf{u}_e = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}, \quad \mathbf{v}_e = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix},$$

要素自由項ベクトル

$$\mathbf{f}_e = \begin{bmatrix} f_1^e \\ f_2^e \\ f_3^e \end{bmatrix},$$

要素係数マトリックス

$$A_e = (A_{ij}^e).$$

このとき、 $\langle \hat{u}, \hat{v} \rangle_e, (f, \hat{v})_e$ は

$$\begin{aligned}
\langle \hat{u}, \hat{v} \rangle_e &= \mathbf{v}_e^T A_e \mathbf{u}_e, \\
(f, \hat{v})_e &= \mathbf{v}_e^T \mathbf{f}_e
\end{aligned}$$

と書ける。さらに、

$$\begin{aligned}
\frac{\partial L_1}{\partial x} &= b_i, \\
\frac{\partial L_1}{\partial y} &= c_i
\end{aligned}$$

なので、

$$\begin{aligned}
A_{ij}^{(e)} &= \int \int_e (b_i b_j + c_i c_j) dx dy \\
&= S(b_i b_j + c_i c_j)
\end{aligned}$$

さらに、

$$f_i^{(e)} = \int \int_e f(x, y) L_i(x, y) dx dy$$

ここで e において、 $f(x, y) = \bar{f}^e$ が定数であれば、

$$f_1^{(e)} = \bar{f}^e \int \int_e L_1(x, y) dx dy = \frac{1}{3} \bar{f}^e S$$

このとき、 \bar{f}^e を $f(x, y)$ の e での平均値や中央値に取れば、一種の近似とみなせる。

5.3.5 自然境界の一般化

Γ_2 上での境界条件が $\frac{\partial u}{\partial \gamma} = g_2$ となる場合の扱いを述べる. 弱形式は,

$$\langle u, v \rangle = (f, v) + \int_{\Gamma_2} g_2 v d\gamma = [g, v]$$

となる. $[g, \hat{v}]_e$ への e からの寄与を考える.

$$[g, \hat{v}]_e = \int_{\Gamma_2^{(e)}} g_2 \hat{v} d\gamma$$

ここで $\Gamma_2^{(e)}$ は Γ_2 の一部, もしくはその近似で e の一部をなすものである. 辺 2^*3^* に沿って座標 S^* を考え, 2^* で $S = 0, 3^*$ で $S = 1$ とすると,

$$\hat{v} = (1 - S^*)v_2 + S^*v_3$$

と表せる. したがって,

$$[g, \hat{v}]_e = v_2 \int_{\Gamma_2^{(e)}} (1 - S^*) g_2 d\gamma + v_3 \int_{\Gamma_2^{(e)}} S^* g_2 d\gamma$$

L_{23} を辺 2^*3^* の長さとしたとき,

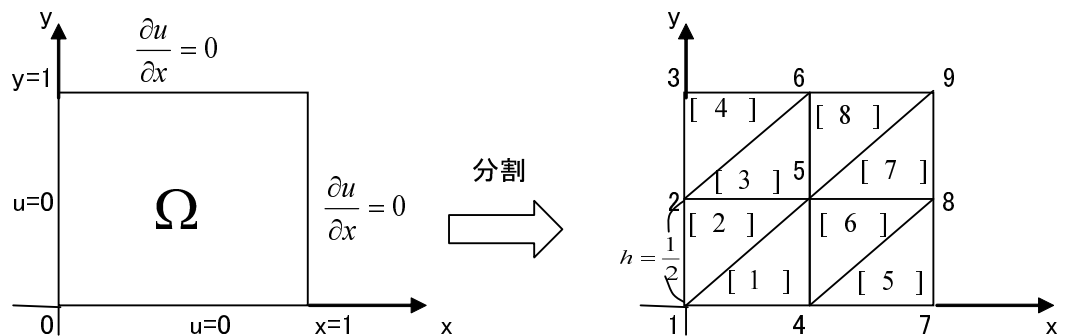
$$\sigma = L_{23} S^*$$

もし, g_2 が $\Gamma_2^{(e)}$ 上で定数 $\bar{g}_2^{(e)}$ であれば,

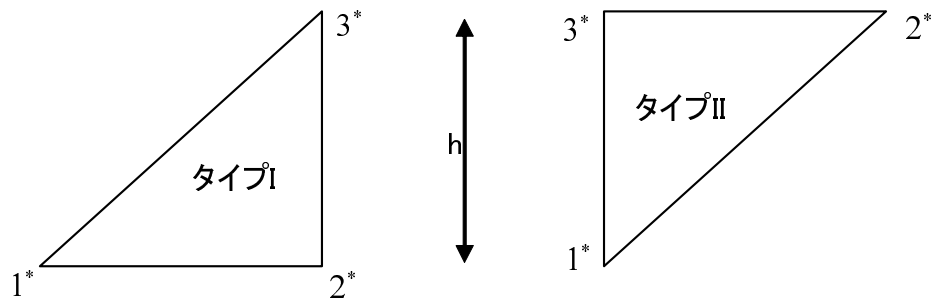
$$[g_2, \hat{v}]_e = (v_2 + v_3) \frac{\bar{g}_2^{(e)} L_{23}}{2} = v_e^T g_e$$

となる.

5.3.6 近似方程式の組み立て



領域を三角形要素に分割し, 節点番号対応表を作成する. このとき, 三角形要素は2種類存在する.



ここで、タイプIについて考える.

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \frac{1}{h^2} \begin{bmatrix} -h \\ h \\ 0 \end{bmatrix}, \quad \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \frac{1}{h^2} \begin{bmatrix} 0 \\ -h \\ h \end{bmatrix}$$

よって,

$$A_e^I = \frac{1}{2} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}, \quad f_e^I = \frac{\bar{f}h^2}{6} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

となる. また、同様にタイプIIについて考えると,

$$A_e^II = \frac{1}{2} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix}, \quad f_e^II = \frac{\bar{f}h^2}{6} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

節点番号対応表

要素	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
1*	1	1	2	2	4	4	5	5
2*	4	5	5	6	7	8	8	9
3*	5	2	6	3	8	5	9	6

要素 [1] について

$$A_e^I = \frac{1}{2} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{matrix} 1 \\ 4 \\ 5 \end{matrix}$$

なので、拡大要素マトリックスは,

- 1-1 要素に 1 を加える,
- 1-4 要素に -1 を加える,
- 1-5 要素に 0 を加える,

という操作を行う。また、拡大自由項ベクトルについても、第 1,4,5 要素に $\bar{f}h^2/6$ をそれぞれ加える。

以上を各要素について行うことで全体係数マトリックス、全体自由項ベクトルを得ることができる。

$$A^* \begin{bmatrix} u_1 \\ \vdots \\ u_9 \end{bmatrix} \leftrightarrow f^*$$

これから基本境界条件を考慮して変形を行う。

Γ_1 上にある節点 (1, 2, 3, 4, 7) は除くので、 A^* の第 1,2,3,4,7 行は除く。また、 $u_1 = u_2 = u_3 = u_4 = u_7 = 0$ とする。

$$Au^* = f$$

$$A = \frac{1}{2} \begin{bmatrix} 8 & -2 & -2 & 0 \\ -2 & 4 & 0 & -1 \\ -2 & 0 & 4 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}, \quad u^* = \begin{bmatrix} u_5 \\ u_6 \\ u_8 \\ u_9 \end{bmatrix}, \quad f = \frac{\bar{f}h^2}{6} \begin{bmatrix} 6 \\ 3 \\ 3 \\ 2 \end{bmatrix}$$

これより u^* を求めればよい。

第 6 章

有限要素近似結果

6.1 はじめに

この章では, 先の有限要素法で近似した方程式をグラフで可視化したものを提示する. また, 真の解との誤差がどれほどのものとなっているかを示す.

6.2 有限要素近似グラフ

扱う方程式

$$\begin{cases} -\Delta u + u = f(u) & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega. \end{cases}$$

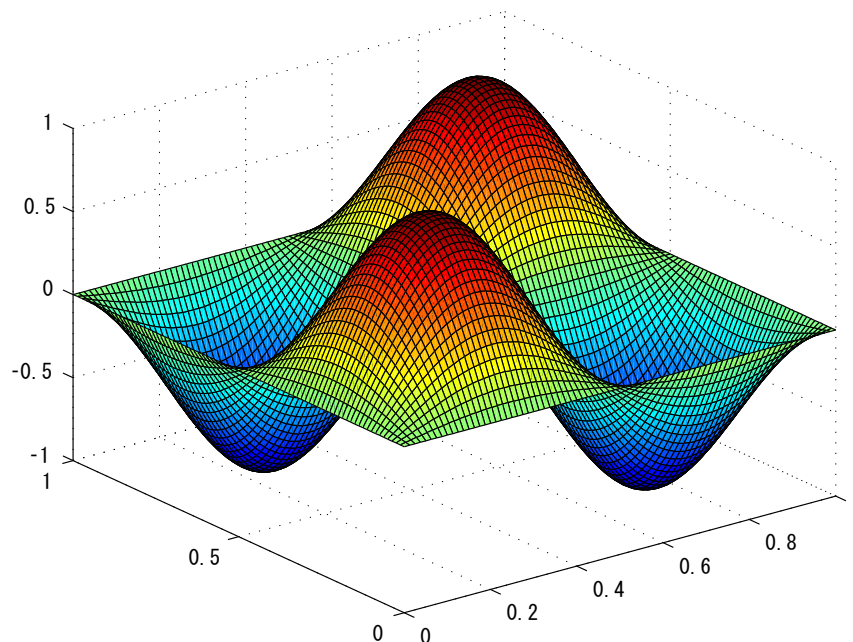
$$f = (8\pi^2 + 1) \sin 2\pi x \sin 2\pi y$$

における真の解 u は

$$u = \sin 2\pi x \sin 2\pi y$$

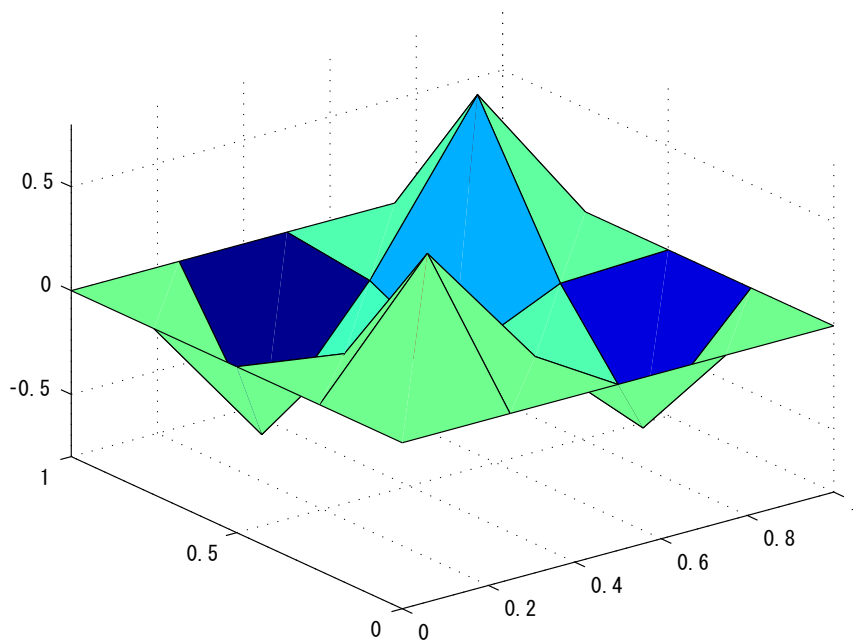
となる.

分割数 $n = 80$ における真の解のグラフは次のようになった.

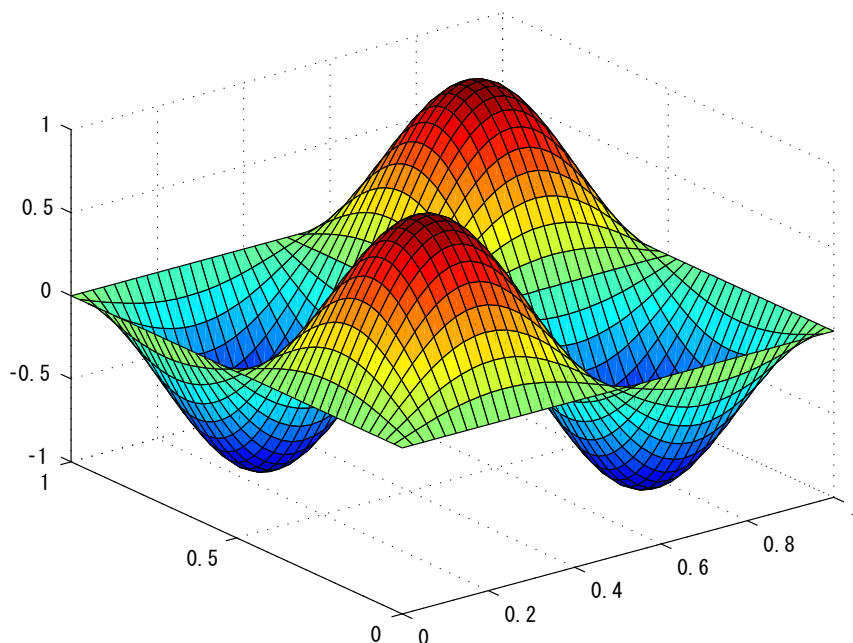


次に、有限要素法で近似したグラフを示す。

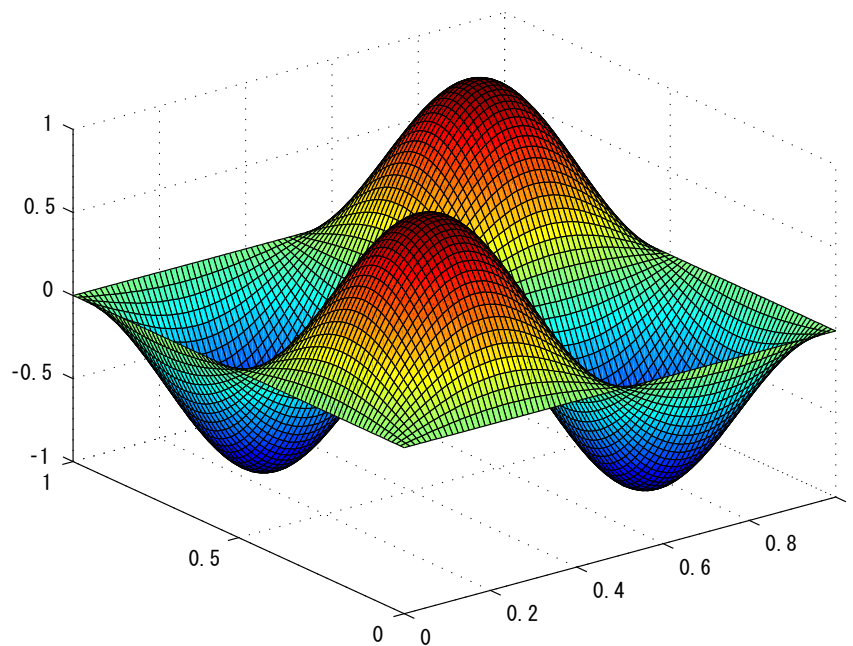
分割数 $n = 4$ の近似解のグラフは次のようになった



分割数 $n = 40$ の近似解のグラフは次のようになった



分割数 $n = 80$ の近似解のグラフは次のようになった



6.3 有限要素近似結果

ここで, 各分割数における誤差の最大値ノルムは次のようになった.

	近似解			真の解
分割数 n	4	40	80	-
$\max(u)$	0.7051	0.9947	0.9985	1.0000
最大誤差	0.4116	0.0063	0.0015	-

第 7 章

精度保証法

7.1 はじめに

中尾先生の手法では, 有限次元と無限次元に分けて精度保証をしている. この章では, 精度保証の方法について有限次元と無限次元のそれぞれについて説明する.

7.2 有限次元の精度保証

有限次元における精度保証法について述べる.

$$P_h T(w) \equiv P_h w - [P_h - P_h F'(u_h)]^{-1} (P_h w - P_h F(w))$$

ここで, $F'(u_h) = -A$ より,

$$P_h T(w) = [I + P_h A]^{-1} (P_h F(w) + P_h A P_h w)$$

上式を変形させると,

$$(I + P_h A) P_h T(w) = P_h F(w) + P_h A P_h w$$

$$\begin{aligned} & (\nabla(I + P_h A) P_h T(w), \nabla \phi_h) \\ &= (\nabla(P_h F(w) + P_h A P_h w), \nabla \phi_h) \end{aligned}$$

ここで, 右辺と左辺をそれぞれ部分積分すると,

$$(\nabla P_h T(w), \nabla \phi_h) + (P_h T(w), \phi_h) = (-V_0, \phi_h)$$

さらに, $P_h T(w) = \sum_{I=1}^N T_p^I \phi_h^I$ とおくと,

$$\sum_{I=1}^N T_p^I \{(\nabla \phi_h^I, \nabla \phi_h) + (\phi_h^I, \phi_h)\} = (-V_0, \phi_h)$$

ここで,

$$G = ((\nabla \phi_h^i, \nabla \phi_h^j) + (\phi_h^i, \phi_h^j))_{ij}$$

とおくと, 有限次元の誤差 $|T|$ は,

$$|T| \leq |G^{-1}| |(-V_0, \phi_h)|$$

でおさえることができる.

7.3 無限次元の精度保証

無限次元における精度保証法について述べる.

$$(I - P_h)T(w) = (I - P_h)F(w)$$

誤差評価

$$(\nabla v, \nabla w) = (f(u), w) \quad w \in S_h$$

に対して,

$$\|V - P_h v\|_{H_0^1} \leq C_0 h \|f(u)\|_L^2$$

と知られている.

今, $(\nabla w, \nabla \phi) = -(w + V_0, \phi)$ である為,

$$\begin{aligned} \|\nabla(I - P_h)T(w)\|_{L^2} &\leq C_0 h \|w + V_0\|_{L^2} \\ &\leq C_0 h (\|w_h\|_{L^2} + \|w_\perp + V_0\|_{L^2}) \end{aligned}$$

ここで Aubin Nitsche の技巧を用いると,

$$\|\nabla(I - P_h)T(w)\|_{L^2} \leq C_0 h (W_h + \frac{\sqrt{2}}{4} \|\nabla V_0\|_{L^2} + \alpha)$$

となる. さらに, $\epsilon = \frac{\sqrt{2}}{4} \|\nabla V_0\|_{L^2}$ とおくと,

$$\|\nabla(I - P_h)T(w)\|_{L^2} \leq C_0 h (W_h + \epsilon + \alpha)$$

と表すことができる. よって, 無限次元の誤差 α は

$$\alpha \leq \frac{C_0 h}{1 - C_0 h} (W_h + \epsilon)$$

でおさえることができる.

第 8 章

精度保証結果

8.1 はじめに

この章では、先の精度保証法で得た誤差について、有限次元、無限次元それぞれについて示す。また、有限要素近似の実行時間と精度保証の実行時間を示す。

8.2 有限次元

有限次元における精度保証結果は次のようになった。

分割数 n	4	20	40	60
W_h	0.0509	0.0402	0.0250	0.0184

8.3 無限次元

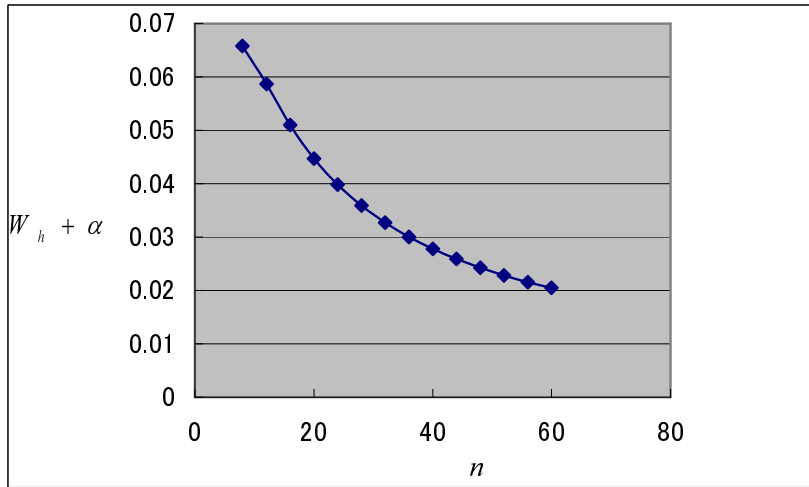
無限次元における精度保証結果は次のようになった。

分割数 n	4	20	40	60
α	0.0057	0.0045	0.0028	0.0021

8.4 有限次元+無限次元

ここで、有限次元と無限次元の精度保証結果を合わせると次のようになった。

分割数 n	4	20	40	60
$W_h + \alpha$	0.0566	0.0447	0.0278	0.0205



8.5 実行時間

各分割数における有限要素近似,そして精度保証にかかった時間は以下の通りである.ここでは5回の実行における平均の数値を示す.

分割数 n	4	20	40	60
有限要素近似 (s)	0.0154	0.2030	3.4404	24.9314
精度保証 (s)	0.0282	0.3252	6.8500	50.2282

第 9 章

プログラム

9.1 はじめに

この章では, 本論文で作成した Matlab プログラムの紹介をする.

9.2 本論文で作成したプログラム

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%有限要素法%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear
format long
n=20;
t=n-1;
N=t^2;
h=1/n;

a=zeros(N);
a=sparse(a);
for p=1:N
a(p,p)=4;

end

for q=1:N-1
a(q+1,q)=-1;
a(q,q+1)=-1;
end

for r=1:N-t
a(r,r+t)=-1;
a(r+t,r)=-1;
end

for s=1:n-2
a(t*(s-1)+n,t*(s-1)+t)=0;
a(t*(s-1)+t,t*(s-1)+n)=0;
end

b=zeros(N);
b=sparse(b);
for p=1:N
b(p,p)=1/2;

end

for q=1:N-1
b(q+1,q)=1/12;
b(q,q+1)=1/12;
end

for r=1:N-t
b(r,r+t)=1/12;
b(r+t,r)=1/12;
end
```

```

for r=1:N-t
b(r,r+t-1)=1/12;
b(r+t-1,r)=1/12;
end

for s=1:n-2
b(t*(s-1)+n,t*(s-1)+t)=0;
b(t*(s-1)+t,t*(s-1)+n)=0;
end

b(1,1)=7/12;
b(N,N)=7/12;

B=b*(h^2);

f=zeros(N,1);
for i=1:t;
    for j=1:t;
        f(t*(i-1)+j)=(8*(pi^2)+1)*sin(2*pi*i*h)*sin(2*pi*j*h);
    end
end

F=B*f;

M=a+B;
u=bicgstab(M,F);

U=zeros(t+2);
for k=1:N;
    z1=rem(k-1,t);
    z2=floor((k-1)/t);
    U(z2+2,z1+2)=u(k);
end
v=zeros(N,1);
for i=1:t;
    for j=1:t;
        v(t*(i-1)+j)=sin(2*pi*i*h)*sin(2*pi*j*h);
    end
end

V=zeros(t+2);
for k=1:N;
    z1=rem(k-1,t);
    z2=floor((k-1)/t);
    V(z2+2,z1+2)=v(k);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%有限要素法近似グラフの表示%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[X,Y]=meshgrid(0:h:1);
surf(X,Y,U)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%有限次元精度保証%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
G1=inv(M);

```

```
G2=abs(G1);
G3=max(max(G2));
g1=u.'*a*u;
g2=sum(sum(B));
g3=(h/pi)*sqrt(g1)*sqrt(g2);
Th=G3*g3;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%無限次元精度保証%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C=(1/pi)^2;
Ch=0;
A=(C/(1-C))*(Th+Ch);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%有限次元+無限次元%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
W=Th+A;
```

第 10 章

統括

10.1 はじめに

本章では, 本論文の統括と結論, 及び今後の展望について述べる.

10.2 統括

第2章では, 準備として, 本論文で扱う数学的予備知識について述べた.

第3章では, 本論文で扱う方程式について述べた.

第4章では, 中尾先生の方法について述べた.

第5章では, 有限要素法について述べた.

第6章では, 有限要素法近似の数値実験結果を示した.

第7章では, 精度保証法について述べた.

第8章では, 精度保証の数値実験結果を示した.

第9章では, 本論文で作成したプログラムを紹介した.

本論文の構成は以上のものであった.

10.3 結論

第8章で示した結果により, (4.1) で挙げた楕円型偏微分方程式に対して, 中尾先生の方法による精度保証が可能であり, 近似にかかる時間の2倍程度の短い実行時間で行えることが分かった.

10.4 今後の展望

今後の研究の展望として, 次のような問題が挙げられる.

- 非線型方程式への応用
- Plum の方法との比較
- 非定常な問題への応用

いずれの問題も興味深いものばかりであり, 今後の研究の発展が楽しみである.

謝辭

本研究を進めるに当たり、終始丁寧な御指導及び御激励を賜り、最後まで面倒を見て下さいました大石進一教授に深く感謝いたします。

また、大変お忙しい中でも私の質問に対して熱心に答え、御教示して下さいました、大石研究室客員講師久保隆徹氏にも深く感謝いたします。

また、同じ大石研究室修士2年として意見の交換や協力をして下さいました山中脩也氏に深く感謝いたします。

最後に本論文に御指導、御協力を頂いた皆様に心から感謝いたします。どうもありがとうございました。

参考文献

- [1] 大石進一: 精度保証付き数値計算, コロナ社 (2000).
- [2] 山本哲郎: 2点境界値問題の数理, コロナ社 (2006).
- [3] 菊池文雄: 有限要素法概説, サイエンス社 (1980).
- [4] 河村哲也: 線形代数と数値解析, 朝倉書店 (2005).
- [5] Mitsuhiro T.Nakao, Nobito Yamamoto:
Numerical Verification of Solutions for Nonlinear Elliptic Problems Using an L^∞ Residual Method, Journal of Mathematical Analysis and Applications 217, 246-262(1998), Article No.AY975712.