現代非線形科学シリーズ　6

精度保証付き数値計算

工学博士　大石進一　著

コロナ社

刊行のことば

理工学においては，実在する現象に対し，それをある程度理想化した物理モデルをつくる。理工学が今日のように発展したのは，この物理モデルから，微分方程式で記述される数学モデルを導き，これを解くことによって，未知の現象を予測したり，新しい工学的な製品を設計することが可能であったからである。

このような，物理モデルを経て，数学モデルをつくり，これを解くことによって，現象を説明する方法を確立したのはニュートン (Newton) である。ニュートンは力学現象の物理モデルをつくり，それからニュートンの運動方程式と呼ばれる微分方程式の導き方を示した。そして，微積分学を創始して，これを解く方法を与えた。これを契機として，電磁気学，相対性理論，量子力学などが作られ，半導体などの発明に結びつき，コンピュータが実現されるようになった。このような方法は，生体や脳，経済現象などの社会科学にまで適用されるようになっている。このように，現象の物理モデルをつくり，それから，数学モデルとして微分方程式を導き，これを解いて，現象の予測や，設計を行うという方法により，現代の高度に発展した科学技術が築かれてきた。なぜ，このような数学モデルがこれほどまでに有効なのか，それは謎である。逆にいえば，科学が成立できたのは，困難の連続の中で，ほとんど唯一の成功といってよい，微分方程式によるモデル化という方法論を得たからということができよう。

こうしてどのような理工学の分野にたずさわっても，現象のモデル化によって得られた微分方程式を解析して，現象に対する知見を導きだすことが要求されるようになる。従来，小さな入力を加えれば，それに対する応答は入力に比例して大きくなるような現象を利用して，工学的なシステムが作られることが多かった。すなわち，線形性の利用である。しかし，科学技術が高度に発展するにつれて，そのような線形性の仮定が成立しないような領域での現象を取り

扱うことが普通となりつつある。すなわち，現代は非線形現象と相対することが，分野を越えて共通する時代となっているのである。

従来，非線形現象のモデルである非線形微分方程式を解くことは容易なことではなった。しかし，計算機環境の飛躍的な発展により，現代では，コンピュータを駆使して数値計算によって近似解を求め，それによって微分方程式から現象に関する情報を引き出すことが可能となっている。こうして，カオス，ソリトン，フラクタルなど予想もしなかったような非線形現象および新しい概念が発見されている。また，コンピュータ技術の発展により，離散系（ディジタル系）の研究，応用がめざましく進展している。さらに，コンピュータはニューラルコンピュータなど，人間の脳をめざして新しい方向へ発展しつつあるが，これらも非線形科学にその基礎を置く部分が多い。

本シリーズは，このように現代理工学の学習，研究においていまや必修となった非線形科学・工学について，数学的，物理的基礎から工学的研究の第一線までを体系的に習得するための専門教科書シリーズとして編まれたものである。すなわち，本シリーズの内容は非線形解析入門，非線形物理などの基礎から始めて，カオス，ニューラルネット，ソリトン，フラクタルなどの非線形科学の新しい基礎，精度保証付き数値計算，高速自動微分などの数値計算法から生体，経済現象に至るまで，非線形科学全般にわたっている。それぞれの巻の著者は各分野の気鋭の研究者にお願いすることができた。本シリーズにより，理工学における新しい共通基礎分野としての非線形科学が基礎から第一線まで総合的に学習できるようになると考えている。

編者は，内容を吟味して，時には書き直しまでお願いしている。御協力頂いた各著者に深謝する次第である。また，新しい時代のシリーズとして共通のスタイルファイル（LaTeX）による執筆も行った。スタイルファイルを作成して下さった中央大学牧野光則助教授に感謝する。また，本シリーズの企画において大変お世話になったコロナ社の各位に心から御礼申し上げる。

1997 年春

編集委員長　大石　進一

まえがき

数値計算法とは，連続数学の問題を計算機上で四則演算に還元して解く手法のことをいう。計算機が発明されたのは1940年代であるから，まだ半世紀しか経過していない。計算機の発明に深く関わったフォンノイマンとチューリングは，計算機の発明後，ただちに連立一次方程式の数値計算における誤差解析論を発表している。これにより，かなり一般的な連立一次方程式の解の，かなりよい近似が数値計算によって求められることが示された。これらは現代数値解析の研究の始まりである。その後半世紀経過し，多くの優れた研究者により数値解析理論は高度に発展した。また，計算機技術も驚くような発展をし，いまや少し前の大型計算機がパソコンとして一人に一台占有して使える時代となった。この間，多くの科学的発見に数値計算は貢献した。また，さまざまな工学的技術が数値計算をもとに構築された。

しかし，そのような現代になっても，数値計算のつぎのような基本問題が残されたままである。占部 実の言葉を借りよう。

「電子計算機を用いる数値計算にあっては，その計算は正確には行われない。四則演算の結果はそのつど四捨五入によって丸められるし，極限を含む無限演算はすべて有限演算で近似して行われる。したがって計算結果は，つねにある程度の誤差をともなっている。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。このことは，数値計算に従事している人たちの間ではあまりとり上げられなかったが，これは一つには結果を得るのに急でそこまで手がまわらなかったということ，また一つにはこのような仕事は労が多くて報われるものが少ないと考えられていたこと，この二つに原因があるように思われる。(中略) 今後は計算結果の誤差評価を行って真の値の範囲を確定し，数値計算によっ

て得られた結論を論理的にも正しいものにする必要がある。」(京都大学数理解析研究所講究録 (1972) より抜粋)

連続数学の問題の真の解の存在を証明したり，真の解との誤差範囲が保証された精度のよい近似解を得る数値計算を精度保証付き数値計算という。著者は1990年以来この分野の研究に取り組んできたが，最近になって，近似解の精度保証が近似解を得る手間の数倍の手間 (2倍程度) で可能であることに気づいた。基本がシンプルなので数値計算のどの分野にも適用できる。こうして，精度保証付き数値計算の研究を始めて以来，10年来の夢であった“精度保証付き数値計算が簡単かつ高速にできる”ことが実現した。結果的にみれば，線形計算においては，丸めの制御を適切な場所で行い，従来の数値計算の資産を利用して精度保証付き数値計算を実行するという，シンプルな方法論になった。ただし，非線形系の計算については，区間演算の利用が必要となり，効率的な解析ができるようになるためには，CPUに実装する初等関数の精度に関する新しい規約づくりや，高速な区間演算ライブラリの開発など，従来の資産だけでは不十分で，新しい展開が必要となる場面も生じるものと思われる。

いずれにしても，数値計算の精度保証が簡単かつ高速にできるようになったので，近い将来，数値計算に精度保証を付加するのが日常的となるであろう。本書は，そのような時代に向けて，数値計算の各分野について高速で効率的な精度保証付き数値計算のための理論と，この10年間に蓄積した実際的なノウハウを展開したものである。記述の仕方としては，数値計算して近似解を求める段階の概要をまとめ，その精度を保証する過程を詳述するというスタイルをとった。これにより，数値計算の初学者であっても，本書を理解できるように工夫した。本書が“数値計算といえば精度保証付き数値計算のことである”時代が到来するのに少しでも役立てば，著者の喜びはこれに過ぎるものはない。

1999年11月

大石　進一

丸めの制御などに関わる最新情報が http://www.oishi.info.waseda.ac.jp/~oishi/FAQ/FAQ.html にあるので参照されたい。

目　　　次

1.　精度保証の原理

2.　連立一次方程式

3.　固 有 値 問 題

4.　線形計画法–単体法による解法

5.　関数の補間と積分

6.　非 線 形 方 程 式

7.　関数方程式

8.　多倍長計算

索　　引

1 精度保証の原理

本章では，精度保証付き数値計算とは何かについてから説明し，その基礎となる浮動小数点システム，区間解析について学ぶ。

1.1 精度保証付き数値計算とは

コンピュータは数値計算のために開発された。その後，数値計算以外にもその適用範囲が広がり，現代のコンピュータ技術に至っている。しかし，依然として，数値計算がコンピュータ利用の目的の上位にあることには変わりがない。そして，高速に数値計算するための技術がつねに探求され，その成果がコンピュータに実装されている。このような高速な数値計算の応用例をいくつか挙げると

1. 大気の動きのシミュレーション(数値計算)によって，天気予報の精度が向上した
2. コンピュータグラフィックス(CG)用の数値計算が高速に行えるので，リアルタイムに動くゲームが楽しめる

などである。

現代のコンピュータにおいて，高速な数値計算ができるのは，技術的には，実数を浮動小数点数で近似して計算を行うからである。実数には円周率 π などの無理数もあるが，それを

$$0.314\,15 \times 10^1 \tag{1.1}$$

のような形の有限桁の小数で近似したのが**浮動小数点数**である。浮動小数点数による数値計算は高速ではあるが，近似である。近似であるというのは，日常的な意味では数値計算の結果はかなりいい線をいっているが，数学的な意味では厳密に正しいことが保証されていないということである。また，病的な例においては，数値計算の結果がまったく見当はずれになってしまうこともある。

それでは，数値計算結果がどのくらい正しいのか検算をしてはどうか，ということになる。これがじつは難しいのである。10 年前までは，数値計算によって近似解を求めること (これは，普通，数値計算と呼ばれる) に要する計算時間を 1 とすると，検算の時間は 1 万くらいであると考えられていた。また，検算の方法すらわからない問題も多数存在していた。

ところが，数値計算結果の検算の方法の研究がこの 10 年で急速に進み，検算の時間が 10 くらいになった。特に，行列にからむ問題など，線形計算においては検算の時間が 1 となる場合もある画期的な方法が発見された (著者とドイツの研究者 Rump 教授との共同研究など)。こうして，数値計算による近似計算の検算が，高速にできるようになった。これを，**精度保証付き数値計算**という。

本書は，精度保証付き数値計算の入門から第一線までを，やさしく紹介することを目的として書かれている。

1.2　浮動小数点数

本節では，浮動少数点数とは何かを明らかにする。浮動小数点数を利用して，近似計算し (数値計算)，浮動小数点数を用いてその結果を検算 (精度保証) するからである。

浮動小数点数システムについては，IEEE 標準 754 (IEEE standard 754, IEEE 754 と略記する) がパソコンやワークステーションなどをはじめとして，多くのコンピュータで標準的に用いられている。この規格は W. Kahan を中心とするグループの提案に基づき，1977 年から制定作業が開始され，1985 年に承認された。例外として有名なのはクレイのスーパーコンピュータである。これは，高速化を狙うために，一部の仕様を意識的にみたさないようにしている。本書では，以下，IEEE 推奨方式 754 に基づく 2 進数浮動小数点数システムを考えることにする。これは IEEE 754 が理論的にも優れたシステムで，その骨格となる仮定をみたす他の浮動小数点数システムでも以下の議論は同様に成立するからである。

すなわち，本書では，浮動小数点数とはIEEE推奨方式754に基づく2進数浮動小数点数システムのことである。

1.2.1 浮動小数点数

IEEE 754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。まず，IEEE 754によって規定される浮動小数点数の集合について述べよう。

IEEE 754によって規定される浮動小数点数としては四つのタイプの数が用意されている。それは規格化2進浮動小数点数，零，非規格化2進浮動小数点数，NaN (Not a Number, 非数) である。

〔*1*〕 **2進規格化浮動小数点数** 2進規格化浮動小数点数とは

$$a = \pm\left(\frac{1}{2} + \frac{d_2}{2^2} + \frac{d_3}{2^3} + \cdots + \frac{d_N}{2^N}\right) \times 2^e, \quad d_i = 0 \text{ か } 1 \tag{1.2}$$

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$e_{\min} \leqq e \leqq e_{\max}$となる整数である。

$$m = \pm\frac{1}{2} + \frac{d_2}{2^2} + \frac{d_3}{2^3} + \cdots + \frac{d_N}{2^N} \tag{1.3}$$

を**符号付き仮数** (signed mantissa) といい，eを**指数** (exponent) という。指数eも2進数で表される。通常，単精度，倍精度 (8 byte = 64 bit)，拡張倍精度 (10 byte = 80 bit) の浮動小数点数システムがあるが，それらはそれぞれつぎのような浮動小数点システムである。

$$N = 24, \quad -126 \leqq e \leqq 127$$

$$N = 53, \quad -1\,022 \leqq e \leqq 1\,023$$

$$N = 64, \quad -16\,382 \leqq e \leqq 16\,383$$

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，本書では，特に断ることのない限り，浮動小数点数といえば，倍精度浮動小数点数のことである。

規格化2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

$$x_{\max} = \left(\frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^N}\right) 2^{e_{\max}} \tag{1.4}$$

であり，その最小値は

$$x_{\min} = \frac{1}{2} 2^{e_{\min}} \tag{1.5}$$

である。倍精度数ではそれぞれ

$$(2 - 2^{52}) \times 2^{1\,023} = 1.797\,693\,134\,862\,315\,7 \cdots \times 10^{308} \tag{1.6}$$

$$2^{-1\,023} = 2.225\,073\,585\,072\,01 \cdots \times 10^{-308} \tag{1.7}$$

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 bit である。

$$2^{-53} = 1.110\,223\,024\,6 \cdots \times 10^{-16} \tag{1.8}$$

より，倍精度浮動小数点数は 10 進数で約 16 桁の精度がある。

〔2〕　**零**　零は規格化されて

$$+\left(\frac{0}{2} + \frac{0}{2^2} + \frac{0}{2^3} + \cdots + \frac{0}{2^N}\right) 2^{e_{\min}} \tag{1.9}$$

と表される。

〔3〕　**非規格化 2 進浮動小数点数**　IEEE 754 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**漸近アンダフロー** (gradual underflow) という。

このような非規格化浮動小数点数の正で一番小さな数は

$$2^{-1\,074} = 4.940\,656\,458\,412\,465\,44 \cdots \times 10^{-324} \tag{1.10}$$

である。これ以下の数になると，**アンダフロー** (underflow) が生じたという。

〔4〕　**NaN**　このほかに，IEEE 754 では，つぎのような特別な数が用意されている。

(a) NaN (Not a Number) は $\sqrt{-5}$, ∞/∞，$+\infty + (-\infty)$ など不当な演算の結果として得られる。

(b) $\pm\infty$ はオーバフローの結果や零で割った結果として得られる。

(c) ± 0 はアンダフローか$\pm\infty$ での割り算の結果として得られる。

本書では，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$\boldsymbol{F}$ で表す。$\boldsymbol{F}$ の要素の数は有限個で，区間 $[-x_{\max}, x_{\max}]$ の上に原点対称に分布する。実数 x が与えられたとき，それを挟む二つの浮動小数点数間の距離を $\mathrm{ulp}(x)$ で表す。ulp は unit in the last place（ウルプ）のことである。

1.2.2 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$\boldsymbol{F}$上での演算は丸めを用いて定義されている。

また，つぎの四つの丸めのモードが指定できるように，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，現在のパソコン，ワークステーションをはじめとして，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 $(c \in \boldsymbol{R})$ とする。

(1) **上向きの丸め** (round upward)　　c 以上の浮動小数点数の中で最も小さい数に丸める。これを $\triangle : \boldsymbol{R} \to \boldsymbol{F}$と表す。

(2) **下向きの丸め** (round downward)　　c 以下の浮動小数点数の中で最も大きい数に丸める。これを $\triangledown : \boldsymbol{R} \to \boldsymbol{F}$と表す。

(3) **最近点への丸め** (round to nearest)　　c に最も近い浮動小数点数に丸める。これを $\Box : \boldsymbol{R} \to \boldsymbol{F}$と表す。もし，このような点が 2 点ある場合には，仮数部の最後のビットが偶数である浮動小数点数に丸める。これを**偶数丸め方式** (round to even) という。

(4) **切り捨て** (round taward 0)　　絶対値が c 以下の浮動小数点数の中で c に最も近いものに丸める。

索　　引

——著者略歴——

1976年　早稲田大学理工学部電子通信学科卒業

1981年　早稲田大学大学院博士後期課程修了（電子通信学専攻）

　　　　工学博士（早稲田大学）

1984年　早稲田大学助教授

1989年　早稲田大学教授

　　　　現在に至る

精度保証付き数値計算

Verified Numerical Computation

2000年 1 月 5 日　初版第1刷発行

2010年 6 月 15 日　初版第4刷発行

検印省略

著　者　大　石　進　一

発 行 者　株式会社　コ ロ ナ 社

代 表 者　牛 来 真 也

印 刷 所　壮光舎印刷株式会社

112-0011　東京都文京区千石 4-46-10

発行所　株式会社　**コ　ロ　ナ　社**

CORONA PUBLISHING CO., LTD.

Tokyo　Japan

振替 00140-8-14844・電話(03)3941-3131(代)

ホームページ　http://www.coronasha.co.jp

ISBN 978-4-339-02605-4　（中原）　（製本：牧製本印刷）

Printed in Japan