私家版「計算」とは何か？ —何故私はN.L.Trefethenによる「数値解析の定義」にムカついたのか？—

幸谷智紀　tkouya@na-net.ornl.gov

平成 15 年 1 月 8 日

1　発端

敬愛する小谷野敦先生の文 [1] に,「私はむっとしたのだ」ということを発端に「むっとした」理由について掘り下げて論じたものがあった。

私も感情的な人間なのでよく「むっ」とする。三十路過ぎてからは極力，ムカついても，感情を表に出した方が良いかどうかの「計算」を即座に行い,「怒鳴り散らした方がよい」と結論されればそのようにするし,「隠した方が得策」と判断すれば努めて平静を装うようにしている。だから，昔の私を知る人が今の私の行動を観察すれば「あいつは大人になった」と勘違いするかもしれない。が,「むっ」とすることは逆に増えている。単にそう見えないだけだ。特に，怒鳴り散らした方が何かと得である教師という職に就いてからは，努めて怒りを露わにするようにしている。学生さんの前では，だが。

しかし，体力に任せて声帯を振動させるのはちと辛くなっているので，小声で辛辣な小言をぼそっと言う技も習得すべく奮闘中である。そうなると今度は反論されることも考えておく必要が出てくる。キーキーと感情的にわめき散らせば，大抵，相手に「こいつは理性的な反論をしても無駄だ」と思われ見捨てられてしまうだろう。あるいは相手の逆ギレを誘引してしまうかもしれないが，そうなれば即座に「ごめんなさい」と謝ってしまうという最後の必殺技を繰り出して収集すれいいだけの話だ。が，小声での嫌味では,「こいつは勝てる」「反論できる」と判断され，それなりに相対することを想定しておく必要がある。で，小言を言う前は一応理論武装をする，という準備も怠らないようにしている。これを一言で表現するなら,「オヤジ臭くなった」のである。

L.N.Trefethen(トレフェセンでいいのかな？) による “The Definition of Numerical Analysis” の翻訳 [10] が日本応用数理学会誌に掲載されたのは今 (2002 年) から 10 年前になる。実は私はその内容を事前に知っていた。この翻訳が載る前に，ある先生から「これを読め」と原文を渡されていたからである。まだウブな修士の頃である。語学力が今よりも非常に低かったから，大体の文意を取るだけでもえらく苦労した覚えがある。原文を読み終えた時には，何とも思わなかったし，著者の言いたいことはほぼ肯定できたのだが，日本語訳が出，それを改めて読んだ時，私は「むっとした」のである。

何故むっとしたのか？　当時はその理由がちゃんと説明できなかった。で，10 年経った本日，書きたくもない書類を前にしてうーうー唸っていたら，いつもの現実逃避癖がにょっこり顔を出し始めた。その時突然私はムカ付いたことを 10 年ぶりに思い出し，今ならその理由を文章に起こすことが出来ることを悟ったのである。

で，私も小谷野先生に習って,「むっとした」理由について縷々述べることにしたい。学術論文じゃないので，あんまし身構えて読まないよーにお願いする。但し，批判やご意見は拝聴したいので（返信するかどーかは確約できないけど），何かあれば本文タイトルに記したメールアドレス宛に送って欲しい。

2　トレフェセンによる「数値計算の定義」

原文は著者のページから今でも読むことが出来る [8]。英語が得意な人はそちらを読んで貰うとして，ここでは岡田・三井による日本語訳 [10] を要約してその内容をお伝えしたい。

トレフェセンは自身が専門とする「数値解析」という学問の定義について，一般に流布されている次の定義は不満であると述べている。

> **定義 1**「数値解析とは，丸め誤差の研究である」

主要な（欧米の）テキストはまず「誤差」「丸め」「コンピュータ演算」から始まっているし，辞書を調べても「近似」「精度」「誤差」という用語が出てくる。これは若い学生諸君の「数値解析」への情熱を殺ぐものであると述べ，次の定義を使うべきであると主張する。

> **定義 2**「数値解析とは，連続数学 (continuous mathematics) の問題に対するアルゴリズムの研究である」

数値解析で対象とする問題の大部分は，連続的な問題であり，これは有限アルゴリズムでは解けないものである。定義 1 はウィルキンソン (J.H.Wilkinson) が Gauss の消去法の丸め誤差解析を徹底して行い，その結果があまりに喧伝されすぎて定着したものであり，これは「痛ましい」結果である，と述べる。

文意は以上の通りで，以後はその主張を補強する説明が続く。その下りでは「丸め誤差は重要」と，より戻し的な所も見受けられるが，最終的にはやっぱり定義 2 は正しい，ということを具体例を提示して終わっている。

この文の一体どこにムカ付いたのか，それは最後に述べることにして，ちょっと寄り道をする。この定義 2 に対する異見を述べるため,「計算」という言葉を掘り下げて（それ程でもないか）考えてみたい。

3　「計算」とは何か？

私の数値計算のテキストの下書き [2] は「計算とは何か？」という章から始まっている。まあ大体今時の「数値計算」を名乗る専門書は大抵同じことしか書いていないし，自分もそれ程学識があるわけではないので，最初ぐらいは目新しいことを書こうと，ない知恵を絞った結果そうなったに過ぎない。

そこではトレフェセンのように，国語辞典から「計算」の意味を引用している。ここでもいつも使っている国語大辞典（小学館）から「計算」の項を引用してみる。

1. 計量を計ること。はかりかぞえること。勘定すること。
2. 見積もること。予想すること。
3. あたえられた数，量，式を一定の規則に従って処理すること。また，未知の数，量，式を公式などを用い，演算の規則に従って求めること。

このうち，１の文例としては三国志の一節が引用されており，現在この意味で「計算」という語を使うことはほぼないと思われる。日常会話では２，３の意味で「計算」という語を使っていることになる。そして，ここで対象とするのは３の意味が中心となる。シミュレーションを行うための科学技術「計算」は，３を大量に行った結果，２の意味を持っているとも言えるが，それはここでは問題にしない。

私は計算を次のように定義したい。

> **「計算」の定義**「人間の頭脳で咀嚼できる範囲内の規則に従って，直線的に結果を導き出す一連の操作」

「計算」の例としては，まず整数・有理数・実数・複素数の四則演算が挙げられる。正確には「算術 (arithmetic)」の部類に属するものだろうが，計算の一種と見なして差し支えない。あるいは関数の微分・積分といった代数的演算も計算と言える。

逆に「計算」とは言えない例としては，ある種の定理や補題，系の証明がある。例えば初等幾何の問題で，ある程度試行錯誤を要し，センスがないと解けないようなものがそれに当たる。古い時代の指導要綱ではこの初等幾何がかなりのウェイトを占めていて，その復活を主張する者も多い。これは所謂「計算」より高度な思考力を要する問題であるから，能力開発にはもってこいというのがその論旨だろう。教育的な効果の是非は兎も角，この種の「証明」は「計算」とは別種のものだという認識がある。「直線的に結果を導き出」せ

ないから計算ではない，ということになる。逆に初等幾何の分野でも，公式さえ知っていれば「直線的に結果を導き出せる」，例えば面積を求める問題等は，面積を「計算」によって得ることが出来る。

但し，時代が変わると，今までは「証明」しないと解けなかった問題が「計算」によって解決してしまうことも起こり得る。例えば，全てのパターンを調べ尽くすのは人力では難しい問題でも，Mathematica を使えば短時間で終わってしまうこともある。また，全体としては「直線的でない」筋道をあれこれたどることになったとしても，その過程で，人力では不可能な大量の「計算」を，コンピュータの手助けを借りて解くといった解決方法は今や日常茶飯事と言って良い。一昔前ならそれなりのセンスが不可欠なものも，コンピュータの馬力に任せた力業の「計算」によって可能になることは珍しくない。これも「コンピュータを使って」「Mathematica の能力を使って」はいるものの,「人間の頭脳で咀嚼できる範囲内の規則に従って」いることは間違いない。そこでどのような計算が行われているか，全てを把握することは出来なくとも,「コンピュータを使えば」「Mathematica を使えば」解決できるという「規則」を知っているからこそ，その内部で行われている大量の小さな計算の過程を知ることなしでも，部分的とはいえ，ある程度「直線的に結果を導き出す」ことが出来ているのだから。

ということで,「一連の操作」の対象が数であれ量であれ式であれ，もっと漠然としたものであれ，人間が「コンピュータを使う」「電卓を使う」「手計算で十分」などの判断をし，必要な道具を揃えてそれを活用し，結果として直線的に結果を得ることが出来れば，その操作は全て「計算」と見なすべきだろう。

4 「数値計算」とは？

「計算」の定義を前述のように定めたのは,「数値計算」をその真部分集合として扱いたかったからである。但しここでは「数値」を扱う「計算」という意味で「数値計算」という語を扱わない。あくまで「数値計算 (Numerical Computation)」と一連の語としての意味を問題にする。

広い意味では次の定義が無難だろう。

> **「数値計算」の定義 1**「コンピュータ上で扱うことの出来る数値 (整数，固定・浮動小数点数) を対象とした「計算」。」

但し，今の所謂科学技術計算に携わる人々が認識する狭義の「数値計算」は

> **「数値計算」の定義 2**「コンピュータ上において，主として浮動小数点数を対象とした「計算」。」

となるだろう。

「トレフェセンの定義はどこ行った？」と思われるかもしれないが，ある連続問題に対してそれを解くアルゴリズムが定まれば，それを実行するのは「計算」に他ならない。従って，彼の定義 2 は『数値解析とは,「数値計算」の研究である』と言い換えられる。面白くも何ともない話になっちゃうけどな。

個人的には，現状としては「数値計算の定義 2」が better だと思われるが，将来を考えると「数値計算の定義 1」にしておくのが無難かなぁと考えている。定義 1 については数式処理と呼ばれる機能の一部分も含まれてしまいそうだが，実はそれを狙っているのである。

コンピュータにおける「実数」の表現は，現状，整数とそれを組み合わせた有理数，更に広い範囲の，かつ細かい単位を扱うための浮動小数点数，という区分がなされている。数値計算を必要とする科学技術計算の分野は，大量の計算を行う必要があるため，有理数止まりではオーバーフローしてしまう可能性が高くなる。これは固定小数点数でも事情は同じである。従って，固定された bit 長で出来うる限り広範囲の実数を表現しようとすれば指数部を設ける他ない。また，有限桁で良い理由として，実行しようとする問題そのものに誤差が含まれている場合，計算途中の精度をある程度以上向上させても，得られる結果の精度には上限が出てきてしまうという事情もある。所謂多倍長計算があまり流行らないのも，計算時間が延びるという以外に，そんなに精度を必要とする計算は相対的に限られるので，IEEE754 倍精度以上を要求するニーズはそんなにないというのが主な原因だろう。兎も角，主に利用するのが浮動小数点数，というのはこのような理由があってのことで，逆にそれがなければ計算途中に誤差が発生しない整数・有理数の「数値計算」でことが足りる。

数式処理と数値計算の境目は？ということになると，本稿の範囲を超えるので，一つだけ事例を挙げてその

分離の難しさを示しておく。

多項式の演算が絡んでくるアルゴリズムは数値計算なのか数式処理なのか？　例えば，因数分解を行うのに，浮動小数点演算による Newton 法で根を求めたりするとこれは数値計算で，係数が整数・有理数で根もその体に含まれていることがあらかじめ分かっている場合は，整数・有理数演算による GCD 計算が可能なわけだが，これは数式処理？　しかし根が有限桁の整数や有理数で収まらない時に浮動小数点数を利用したりすると数値計算になるのか？

あるいは ODE における Taylor 展開法は果たして数式処理なのか数値計算なのか？最初から全て浮動小数点数で展開式の係数を表現すれば数値計算で，整数・有理数で事足りるなら数式処理か？　それもやはり桁が足らずに浮動小数点数を途中から利用したりすると数値計算になるのか？どうやらボーダーラインはこの辺にありそうだが，近似代数 [5] ってのは果たして，どっちになるんでしょう？そもそも分けることが無意味なのか？

「数式処理」にグラフィック・マルチメディアの機能と，IEEE754 ベースの浮動小数点演算を最初に大々的に取り入れたのが Mathematica だと言われている。だとすれば，Mathematica は数式処理の出来る数値計算ソフトウェア?それとも数値計算の出来る数式処理ソフトウェア?　こうなってくると，近年の対話的インターフェースを備えた統合的な数学ソフトウェア，Matlab, MuPad，Maple 等々は果たして何なのか？訳が分からなくなってくる。ユーザの望む機能を，ふくれあがる PC の処理能力に応じて取り入れた結果，お互い同じような機能を取り揃えるに至ってしまったのである。そして，昔ならもうちょっと明確に分離できていた「数値計算」と「数式処理」の機能が，一つのソフトウェアに組み込まれるに至って，混在，というか融合してしまったと言える。Mathematica なら，FindRoot 関数は数値計算的だが，Solve 関数となるとさてどうなることやら。更にそれらの関数を組み合わせた一連の「アルゴリズム」は一体どう仕分けできるというのだろう？　こうなるともう殆ど諸星大二郎の「生物都市」[3] の世界である。それがユートピアなのかどうかは分からないが，数値をベースとした「計算」と記号処理を重ねた「計算」が一緒くたになって，更に数学公式集というデータベースも取り入れた，機能の多寡はあれ，統合的なソフトウェアになっていることだけは間違いない。こうなるともう単に「数学ソフトウェア」とか両者を含む「計算」ソフトウェアと呼ぶべきだという気がするが如何だろうか？

そんなご時世では「数値」を浮動小数点数に限定しすぎるのもどうかと思うのである。浮動小数点数が非常に強力な，実数を表現する離散化手法であることは事実だが，整数や有理数で済む範囲の計算なら無理して浮動小数点数を使う必要もあるまい。また，その範囲に収まらないようなら，必要に応じて浮動小数点数に切り替えれば良いだけの話だ。ということで，現実を踏まえると,「数値計算」は定義１ぐらいの意味とし，狭義には定義２の意味で使っている人が多い，という説明が無難な線だろう。

5　ムカ付いた理由

私は数値計算における丸め誤差に関連する事項に興味を持つ人間であるが，トレフェセンの言う定義１が数値解析の定義としてはふさわしくないという意見には賛成する。むしろ，私は「今更そんなこと言われんでもわかっとるわい」という理由でムカ付いたのである。

日本の場合，私の恩師筋にあたる現在６０歳以上の先生方は，基本的なサブルーチン群さえロクにないという計算機黎明時代を過ごしており，まず初等関数のルーチン群から作り上げ，その上に現在まで伝えられている頑健な数値計算アルゴリズムを研究しながら整えていく必要があった。また，使用できる浮動小数点数の桁数も少なかったから，丸め誤差に対して今以上に過敏にならざるを得ないという事情があって，そういう先生方が，かなり誤差に関しては「うるさ方」であることは否定しない。しかしながら，その弟子筋に当たる人で，今に至るまで主として丸め誤差に拘泥しているのは私や兄弟子ぐらいなもんで，他の方々はとっくの昔に「連続数学に対するアルゴリズムの研究」に邁進している。トレフェセンの文の翻訳が掲載された 1993 年時点で「数値計算は丸め誤差の研究である」などと考えている人がどれほどいたのか？かなり疑問だ。現に戸川 [7] は誤差に関する BIT 増刊号においてすら,「アルゴリズムの研究が格好いい」と述べているのだ。これが発行されたのが昭和 50 年だから，1975 年の当時においてすら，既にそういう認識があったのである。

今更何で声高に，丸め誤差の研究が主流ではない，な

どと言う必要があったのか。そしてそれが何故，日本語に翻訳されて応用数理に掲載されたのか？　そこに若い私は言外の「ニュアンス」を感じ取ったのである。

それはつまり,「丸め誤差の研究」を主流と考える頭の古い人間が未だに多いから，啓蒙の意味でこの文章を掲載する必要がある，というニュアンスだ。単なる妄想？　ならいいが，このトレフェセンのエッセイはSIAM News に掲載されたもので，それをわざわざ日本語訳にして数値解析屋の集う応用数理学会誌に再掲載するなんぞは，当時の論文誌編集委員の中に「啓蒙してやろう」という精神があったと勘ぐられても仕方あるまい。まーだそんな段階なの？日本の数値解析は？と言われたような気がして，それでムカ付いちゃったのである。「日本はそんなに遅れてはいない！」と 10 年前の私は言いたかった，が，それをちゃんと言葉にまとめることが出来なかった，という訳である。

しかし，今から冷静に考えてみると，このトレフェセンの文はかなりアジテーション的なものと思われる。大体, 私も含めて「数値計算の研究」に従事する「数値解析」屋は，この手の教条的な議論は酒の肴という程度にしか思っていないのが普通だ。「数値計算」「数値解析」の「定義」なんぞという論文のネタになりそうもない話題に労力を注ぎ込むよりは，新規な発見を盛り込んだ研究に邁進するのが得策と考える自分勝手な人間が所謂理系の研究者の際だった特徴なのである。トレフェセン先生は最近も「100 ドル 100 桁問題」[9] という出題を SIAM News に掲載したりして，かなり扇情的な文章を書くのがお好きな人のようであるが,「100 ドル」という極めて現実的な金銭感覚を持っているあたり，世間ズレしたオトナの感性が感じられ，どーもウブな研究者を煽ってその反応を楽しんでやろう，という意図も見えてきたりする。10 年前に私がムカ付いた文章も，他の (英語圏の) 研究者が「数値解析」をどのように認識しているか，一度試してやろう，という目的で執筆されたものとしか思えない。従って，その文を読んで「ムカ付いた」10 年前の私は，まんまとトレフェセンというお釈迦様の手の上で踊らされたということになろう[1]。おかげで本エッセイが完成したのだから，ま，この辺で勘弁しといてやらぁ(ここは吉本新喜劇風にズッコケてほしい)。

[1] 当時の応用数理編集委員諸氏も踊らされたのかどうか。その辺は不明である。

補注 1

私は狭義の「数値計算」において発生する丸め誤差に関心を持つ。故に，最近流行の「精度保証」について言及しておきたい。

精度保証つき数値計算の研究が盛んな現在では，丸めモードを変えた計算結果の差を取ったり（保証にはならない場合が多い），桁数を変えた任意精度計算結果の差を取るなんて原始的な方法で丸め誤差を測定する手法を好きこのんで使っている私はかなり時代遅れだ。が，そのスタンスを変えるつもりはないし，トレフェセン先生が言うところの主流ではない数値解析をこれからも続けていく予定である。

精度保証グループの研究報告書を読んでみると，理論誤差も丸め誤差も含めてトータルな精度保証を行うのが目的のようである。ユーザの負担を極力少なくした精度保証可能な数値計算ライブラリが登場すれば，かなり普及すると思われるし，私も利用したいと思っている（勝手な奴)。

但し，現状，全てのアルゴリズムで精度保証が出来るわけではなさそうで，暫くは古典的な誤差解析手法も併用せざるを得ない。従って，現実的には，保証できるアルゴリズムについてはオプションで「精度保証付き」と「保証なし」が切り替えられると，よりユーザフレンドリーだろう。もし実装しようという人がいれば，その観点からお願いしたい。自分で作るつもりはないが，STL として提案してもらえると，もっとユーザは増えるんじゃないか？

全ての数値計算アルゴリズで精度保証が可能になるのかどうかは不明だが，そうなる日までは，ある程度古典的な誤差解析手法に親しんでおいても無駄ではないということは断言できる。まずは自分でトレフェセン流ではない，めっちゃold style な数値計算マニュアルを書いてみたい。少なくとも骨董品的な「味わい」ぐらいは出るだろうしねぇ。

私の大師匠は常々，数値計算アルゴリズムは “Simple is best”だと言っている。私は “Simple is almost better”だと考える。Bestってのは人により状況により変化しうるものだから，“Simple”が “Best”かどうかは正直分からないことが多い。が，大概のケースでは “Better”じゃないの？ぐらいは言って良いだろうという日和見的なニュアンスを加味した標語である。今のところ，現状肯定の old style 野郎である私は，Simple

な丸め誤差測定法が better だと思って使い続けることだろうし，絶対的な「精度保証」なしの数値計算が全て否定されることもないと考えている。

補注 2

S 先生より,「数値計算とは，四則演算だけで全てを計算する」ことを言うのではないか？というご意見を拝聴した。確かに田原総一郎の本 [6] でも，数値計算について同様の説明を富士通の技術者が行っている下りがある。私はそれも「狭義の」数値計算だと思っているが，どうだろうか？

また，欧米の「数値解析」の本に「誤差」から始まっていないものも結構存在するのでは?という指摘もあった。これは「数値計算」の理論を説明することを目的としているからではないだろうか。「理論」なら丸め誤差は無視しても良い，という立場を取ることも可能だろう。また,「数値計算は丸め誤差研究」ではないという例証でもある。

参考文献

[1] 小谷野敦, 「軟弱者の言い分」, 晶文社, 2001.

[2] 幸谷智紀, 「ソフトウェアとしての数値計算」, `http://www.pas-net.jp/nasoft/`

[3] 諸星大二郎, 「生物都市」, 週刊少年ジャンプ 31 号, 1974.

[4] 佐々木建昭, 「数式処理：研究道具として、研究対象として」, 応用数理 Vol.1, No.2, pp.135-152, 1991.

[5] 佐々木建昭 (研究代表者), 「近似代数の算法と応用の研究」, 科学研究費補助金報告書, 2000.

[6] 田原総一郎, 「日本コンピュータの黎明」, 文藝春秋, 1992

[7] 一松　信・戸川隼人編, BIT 増刊「数値計算における誤差」, 1975.

[8] N.L.Trefethen, “The Definition of Numerical Analysis”, `http://web.comlab.ox.ac.uk/oucl/work/nick.trefethen/defn.ps.gz`

[9] N.L.Trefethen, “The SIAM 100-dollar, 100-digit Callenge”, `http://web.comlab.ox.ac.uk/oucl/work/nick.trefethen/hundred.html`

[10] N.L.Trefethen(岡田・三井訳), “数値解析の定義”, 応用数理 Vol.3, No.2, pp.133-137, 1993.