

第20章 数値計算ソフトウェアの構造

偉大な科学上の計画を物語るには、明らかに異なる2つの方法がある。一つは正確な大きさと手続きという狭い意味において、単に事実を記録するのである。その場合は主として「ある」という動詞の各種の時の変化を用いる。他の一つは物語の血と肉に深くはいついて、事実は単に構成材料として使用し、これを「為す」という動詞で結びつけるのである。この場合、全体は部分よりも大きく、事実の代わりに真理が重んぜられる。

D.O.Woodbury/関・湯澤・成相訳
「パロマーの巨人望遠鏡(上)」(岩波文庫)

本章は、今まで見てきた数値計算アルゴリズムとそれを成立させる基盤的技法を、ソフトウェアの立場から階層化して提示し、改めて「ソフトウェアとしての」数値計算を見つめて貫く目的で執筆された。本書の最初に述べたように、現在の数値計算は、ユーザが自らアルゴリズムを実装しなければ実行できないものではない。むしろ、他者が作成したライブラリや統合ソフトウェアが溢れており、それらを積極的に活用することで効率的に自らの目的を達することが出来る、という状態にある。そういう現状を鑑みて、それに安住するか、反旗を翻してオリジナルの数値計算環境をスクラッチから作り上げるのかは、ユーザが判断すべき事であるが、どちらの道を選ぶにしろ、本章で述べるソフトウェアとして数値計算の「階層構造」を理解することは絶対に損にはならない、と断言する。

20.1 数値計算の階層構造

前章までに述べてきた数値計算に関するアルゴリズムをまとめて階層構造にしてみたのが図20.1である。かなり漠然とした図であるが、大まかなところは納得して頂けるだろう。矢印があるのは、下の階層に位置づけられた機能を上位の階層の機能が利用している、ということの意味する。例えば、浮動小数点形式は殆ど全ての数値計算アルゴリズムで利用されるため、どの階層へも矢印が引かれている、ということになる。

この階層構造を前提にすると、現状の数値計算環境の説明が非常に行きやすくなるのである。

前にも述べたように、現在の数値計算はPCやWS上で実行されることが多く、そこでは主としてIEEE754単精度・倍精度が利用される。この場合、(1)の浮動小数点数の形式はIEEE754という規格で定められたものを用いることになる。それを用いて四則演算や初等関数の計算を実行する(2)の大部分は、CPUの浮動小数点演算ユニットで直接処理される。従って、ユーザが自ら整えなければならない部分は、(2)の機能を利用するためのインターフェースと、(3)~(5)までの機能及びそれらを利用するためのインターフェースである。ちなみに(3)の部分はBLAS(Basic Linear

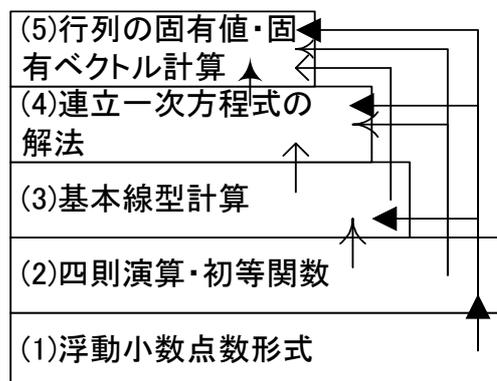


図 20.1: 数値計算の階層構造

Algebra Subprogram) というライブラリが、(4), (5) の部分は LAPACK というライブラリが de facto standard な IEEE754 浮動小数点数用のライブラリとなっている。

勿論、IEEE754 規格以上の長さの仮数部・指数部を持つ多倍長浮動小数点数を利用するには、(1) を独自に定め、(2) の機能を自らソフトウェアとして作り上げなければならないことになる。

この(2)~(5)までの機能を利用するためのインターフェイスに当たるのが、ソフトウェアライブラリ、もしくは統合型の数学ソフトウェアである。

これまでの章で紹介してきた数値計算アルゴリズムは、この(1)~(5)の機能を利用して実行される。よって、数値計算アルゴリズムを全般を眺めた場合、線型計算以下の機能が重要であることが理解されるであろう。

演習問題

次のアルゴリズムの階層構造を図示せよ。

1. CG 法 (アルゴリズム 18)
2. Newton 法 (アルゴリズム 27)
3. 内部反復に Newton 法 (アルゴリズム 39) を用いた陰的 Runge-Kutta 法 (アルゴリズム 38)

参考図書

ソフトウェアライブラリとして記述されたサブルーチン集として定評の高い書籍の日本語訳を紹介しておく。アルゴリズムの解説が丁寧で、参考文献が各節毎に紹介されており、数値計算の教科書としても優れている。

Numerica Recipes in C [日本語版]

W.H.Press et al./丹慶勝市他 訳

技術評論社

1993 年