

# コンピュータの可能性を促進する基盤ソフトウェアの構造とは -数理演算を通して考えるノイマン型と人との異同-

慶應義塾大学 木下博之

## 1. 目的

コンピュータの限界は、ノイマン型構造に由来するというよりも、コンパイラ、インタプリタ等のソフトウェア基盤の保守的姿勢にあると指摘したい。すなわち、コンピュータは人間の数学的能力を再現する機器であり(Nohzawa 2003)、その数学的認識を通して概念の共有をするのが情報化社会のあり方である(脇 2003)という論点に立っている。それにおいて重要となる、確率、統計演算、及び、テキストマイニングを視野に入れつつ、数理演算のあり方について検討する。

## 2. 各コンパイラ、インタプリタの優位性

### I. コンパイラ型

#### (i) Fortran

Fortran は、最初期に起草された高級言語である。しかしながら、配列処理、複素数処理を、直接に記述する方法を有している点で優位にある。

例えば、 $f(x,y,z) = 2g^2(x,y,z) - g(x,y,z) + 5$  について、  
“f(:, :, :) = 2.0\*g(:, :, :)\*\*2 - g(:, :, :) + 5.0”

とすれば、配列の演算を行うことが出来、数学的感覚に沿っているといえる。

メインフレームにおいて開発されたこともあり、並列処理を前提とするシステムと適合度が高い。

#### (ii) C++, C

汎用性が高い言語である。ただし、ベクトル、行列、複素数などについては、変数型として定義されておらず直接的に記述することはできない。

### II. インタプリタ型

#### (iii) R

ベクトルを変数として扱うことが出来る。このことにより、統計解析の手順を直接的、普遍的に記述することが出来る。

#### (iv) Mathematica

数式処理インタプリタである。数学的な式変形、多桁計算等を、数学的な記述で指定できる。

#### (v) MATLAB, Octave

数学的概念に沿った記述が可能であり、ループ等の基本的言語機能を有する。

#### (vi) 各種統計解析システム

SPSS Statistics, STATA, Mplus, EViews 等、各分野を想定したものが多くある。

### 3. 特徴の比較

コンパイラを用いる場合、単純な命令を多く連ねる必要があるがより自由に独立したソフトウェアとして仕上げる事が出来る。ただ、ごく単純なものを除き数学的概念を直接に記述できない場合が多く、煩雑さを感じさせることがある。

インタプリタ型のシステムは、高度な処理を直接に指定する命令を有し、操作者との対話を通じ成果を生み出すことに有利である。ただし、一定の手順を繰り返す場合には、実行速度、操作の手間が増す場合がある。また、統計処理という同じ範疇にあっても、経済分野、生物分野など、特定の分野ごとに特別化して提供されていることが多い。それぞれの分野に最適化されているともいえるが、一般性、汎用性に欠けているとみなすこともできる。

### 4. 提案

数学的概念に基本的に対応するコンパイラが求められる。R, Mathematica, Octave など、インタプリタ型としては実例があるので、これらを参考に試みるべきである。その試みの中で、数理言語、統計解析言語、テキストマイニング言語の収斂を試みるべきである。

方程式自体を扱う変数を定義し、さらに柔軟な式変形を実現できればなお良い。また、その影響についての検討は不可欠である。

GUI, 表形式編集機能, テスト機能, グラフ機能を兼ね備えた統合開発環境となればなお望ましい。

統計と数理の融合にもつながる。

### 参考文献

Ronen Feldman (2010). “テキストマイニングハンドブック” 東京電機大学出版局.

北川源四郎(1993). “時系列解析プログラミング—FORTRAN77” 岩波書店.

松田七美男(2011). “Octave の精義” カットシステム.

Toru Nohzawa (2003). “コンピュータの発明” テクノレビュー.

Scott Rosenberg (2009). “プログラマーのジレンマ” 日経 BP 社.

櫻井鉄也(2003). “MATLAB/Scilab で理解する数値計算” 東京大学出版会.

Stan Wagon(1991). “Mathematica に見える現代数学” ブレーン出版.

脇英世(2003). “インターネットを創った人たち” 青土社.