

# 行列演算用言語 LAMAX-S (1)

八巻 直一, 内田 智史, 本郷 茂

## 1. はじめに

ORの発展は、コンピュータの進歩とともにその適用範囲を広げてきました。線形計画法におけるシンプレックス法の出現に端を発する数値計画法の進展と、それを効率よく計算するコンピュータに支えられて、今日のORがあるのです。特に、最近のスーパーコンピュータの存在は、ORのさらなる発展に拍車をかけることになるでしょう。

ここに、3回にわたって紹介します LAMAX-S[1] は、これからのORの進歩と普及および教育のために、すこしでも役立てたいとの目的で開発された言語です。LAMAX-S は Language for MAtrIX calculation on Super computer のことで、スーパーコンピュータに適用できる行列演算用言語を意味します。LAMAX-S については、本学会での発表[2]や本誌1992年12月号[3]の紹介記事で、ご存じの方も多とおもいます。そこで、今回からの連載と先の紹介記事とを併せてお読みいただければ幸いです。また、1992年12月号の内容との重複もありますが、あえて重ねて説明する部分もあることをお断りしておきます。

ORに現われる各種のアルゴリズムは、行列表現で記述されることが普通でしょう。しかし、FORTRANやC言語などでは、行列を明示的に取り扱うことはできないので、アルゴリズムのプログラミングには、結局アルゴリズムを要素演算に書き直す必要があります。この操作のために、もとのアルゴリズムの表現の読解性が失われることはもちろん、誤りの混入の危険性やプログラミングテクニックへの依存性が高まり、最終的な解の生産性と信頼性に悪影響を及ぼしてしまいます。一方、スーパーコンピュータにおいては、それぞれの機種に応じたチューニングが必要ですが、それは相当に熟練を要する

作業となります。いずれにしても、行列演算のプログラム記述は手作業にたよるのが現状ですので、多くの研究者や実務家の貴重な時間を無駄に費やしていることになります。

LAMAX-S は、このような問題点を解決するために開発が計画されました。LAMAX-S の設計のコンセプトは、以下のとおりです。

- A. 数学的表現が素直にできる言語を実現する。
- B. 個々のスーパーコンピュータのハードウェア知識を活かして、ユーザーが意図する性能が引きだせる能力を実現する。
- C. 有用な数学ライブラリ (LINPACK[4]など) を、有効に活かす能力を実現する。
- D. 過去の社会的資産を継承するとともに、技術計算のプログラミング文化を継承する。

LAMAX-S によって、研究者や実務家の生産性が格段に向上すれば、LAMAX-S は社会的に大いに貢献したといえましょう。なお、上記のコンセプトの帰結として、教育用のツールとしての活用が効果的であることが利用経験を得て発見されたことを付け加えておきます。

## 2. LAMAX-S の開発経緯

まず、LAMAX-S の歴史的経緯について、あらましを述べておきましょう。プログラミング言語の仕様は、一朝一夕に練れるものではありません。LAMAX-S もそれなりの時間を経て現在の形に落ち着いております。

LAMAX が本格的に公開されたのは、東京大学大型計算機センターが最初でした。東京大学大型計算機センターへの移植は1983年のことです。その後、1984年には北海道大学大型計算機センターに、さらに1987年に専修大学大型計算機センターに移植公開されました。このときの版はLAMAX-Eといわれるものでした。

LAMAX-E はそれなりの役割を現在も果たしていますが、その後のスーパーコンピュータをはじめとするコンピュータ技術の進歩への対応を考慮すると、まだまだ改善の余地が数多く残されていることも事実でした。ユ

やまき なおかず 銻システム計画研究所

〒150 渋谷区桜丘町2-9 カサヤビル

うちだ さとし 神奈川大学, ほんごう しげる 専修大学

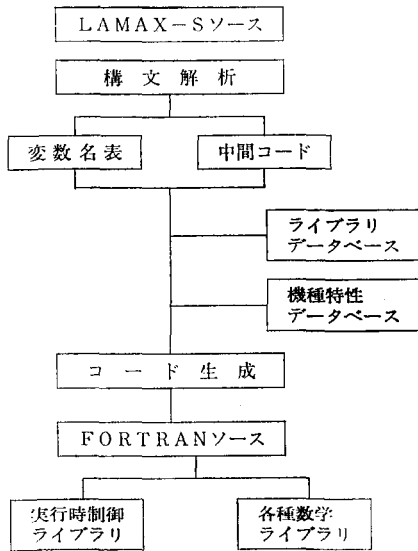


図 3.1 LAMA-S 処理系概念

ユーザーからも、もっと高い機能と使いやすさをとの要望が寄せられていました。そこで、新しい環境に適応した、より高い次元の言語を開発しようということになりました。それが LAMA-S プロジェクトです。

OR学会賞の対象となったバージョンは、プロトタイプ P C 98 版といわれるものです。P C 98 版の仕様については、本稿の最後を参照してください。

### 3. 処理系の概要

LAMAX-S の文法および、行列の取扱のあらまはしは、すでに 1992 年 12 月号に記載しましたので、ここでは処理系の概要を述べることにします。図 3.1 に処理概念を示します。

#### 3.1 中間コードと実行制御

構文解析部では、FORTRAN77 の構文も同時に解析します。LAMAX-S では中間コードが大きな働きをします。上記の各種数学ライブラリの知識を導入するとき、各種コンピュータの最適化（並列化とベクトル化）の知識を導入するときも、基本的には中間コードに対して加工を行ないます。上記の他に、数学的意味をもとに、メモリ量や計算量を最小にする操作も、中間コードに対して行なうように考慮しています。

実行時制御は、主としてメモリ管理を受けもちます。LAMAX-S では、行列は初めから次元を決定するもの（静的行列）と、代入時に次元が動的に決定されるもの（動的行列）[3]があります。動的行列と計算途中に現

われるワークエリアは、大きな COMMON 領域に必要なとき配置され、不用になると削除されます。このようなメモリの制御機構が、メモリ管理です。

#### 3.2 知識のデータベース

LAMAX-S の鍵は、スーパーコンピュータを中心とする数学ライブラリの知識（機能の種類と特徴、引き数情報、例外処理、上手に使う選択基準など）をいかに収集し、いかに活用するかにあります。もう 1 つの鍵は、FORTRAN コードをいかにすれば、スーパーコンピュータをはじめとする各種コンピュータの計算機構を、最も効率的に活かせるかです。それには、数多くのテストを重ねなければなりません。現在は、テストプログラムを作成して、データを収集しているところです。さらに集められた知識をどのように活かすかですが、現在稼働中の P C 98 版は、数学ライブラリとして LINPACK、は計算機構としてはインテル社のベクトルプロセッサ i 80860 の知識が用いられているだけです。そのうえ、i 80860 はまだあまり普及してはいませんので、実際に i 80860 を用いる際には、若干の工夫を要する状態です。現在作成中の版では、NEC ACOS SX シリーズ、HITAC S シリーズ、FACOM VP シリーズ、CONVEX あるいは HP、SUN などのワークステーションの知識を盛り込みたいと考えております。

### 4. プログラム例

ここでは、LAMAX-S のアルゴリズム記述能力を、代表的な例によって示すことにします。1992 年 12 月号の文法説明を併せて確かめていただければ、その便利さが実感されるでしょう。

以下に、行列宣言のいくつかを示します。[5]

1. real : matrix [100, 100 : lower\_tr(1)] A
2. real : matrix [100, 100 : upper\_tr(1)] B
3. real : matrix [100, 100 : sparse(200), symmetric] C
4. real : matrix [10, 10 : band(2, 2)] D

上記の各行列は表 4.1 のように定義されます。

行列の演算式も以下に示すように、自然な表現をほぼそのまま用いることができます。

1.  $A = B' * C$
2.  $A = B^{\wedge} * C$
3.  $A = B[* , i] * c[j , *] * X$
4.  $k = x * Q * x + b' * x$

1. は B の転置と C の積を A に代入、2. は B の逆行列との C 積を A に代入、3. は B の i 列ベクトルと C の j 行ベ

表 4.1 行列の定義

行列	型	構造と数学的性質
A	実数	下三角 (対角要素は0)
B	実数	上三角 (対角要素は0)
C	実数	スパース (要素数<200) 対称
D	実数	バンド (5重)

クトル, および X の積を A に代入, 4. の右辺は 2 次形式を表わします。

次に紹介するプログラムは, 連立 1 次方程式に対する共役傾斜法です。学生がプログラミングを行なうことを想定してテキストの記述を忠実になぞってみましょう。テキストとして, 森正武他の『数値計算』[6]を選びました。LAMAX-S と FORTRAN のプログラムを示しますので, その違いがよくわかると思います。

## 5. おわりに

本連載は, 本誌1992年12月号を事実上の第1回とするかたちになりましたので, 本編だけでは説明不足を余儀なくされました。ぜひ上記記事を合わせてお読みいただくようお願いいたします。次回では, 事例の紹介を主にし, 最終回ではスーパーコンピュータでの自動チューニングと知識の獲得および蓄積について述べる予定です。

### 連立一次方程式 $Ax = b$ に対する共役傾斜法

$x_0$  を任意の初期ベクトルとして,

$$r_0 = b - Ax_0, \quad p_0 = r_0$$

とおく。

$k = 0, 1, \dots$  について次の反復を行う。

$$(1) \quad \alpha_k = \frac{(r_k, r_k)}{(p_k, Ap_k)}$$

$$(2) \quad x_{k+1} = x_k + \alpha_k p_k$$

$$(3) \quad r_{k+1} = r_k - \alpha_k Ap_k$$

$$(4) \quad \beta_k = -\frac{(r_{k+1}, Ap_k)}{(p_k, Ap_k)}$$

$$(5) \quad p_{k+1} = r_{k+1} + \beta_k p_k$$

### c 共役傾斜法

```
real:matrix[*,*,symmetric] A
real:vector[*] x,b,r,p,v
```

c

```
read(*,*) n
call minput(A)
call minput(b)
x := 0::vector[n]
r = b - A*x
p = r
```

c

```
do 100 k = 0,2*n
  if( absmax(p) .lt. 1.e-8 )go to 200
  v = A*p
  w = p'*v
  alpha = (p'*r)/w
  x = x + alpha*p
  r = r - alpha*v
  beta = -(r'*v)/w
  p = r + beta*p
100 continue
write(*,*) ' 収束せず'
200 write(*,*) ' 解'
call mprint(x)
end
```

図 4.1 共役傾斜法の LAMAX-S によるプログラム

### c 共役傾斜法

```
real A(100,100)
real b(100),x(100),r(100),p(100),v(100)
```

c

```
read(*,*) n
read(*,*) ( (A(i,j),j=1,n), i=1,n )
read(*,*) ( b(i), i=1,n )
```

c

```
x := 0::vector[n]
do 1 i = 1,n
  x(i) = 0
```

1 continue

c

```
r = b - A*x
do 3 i = 1,n
  r(i) = b(i)
  do 2 j = 1, n
    r(i) = r(i) - A(i,j)*x(j)
```

2 continue

3 continue

```

c   p = r
    do 4 i = 1, n
      p(i) = r(i)
4   continue
c
    do 100 k = 0, 2*n
      absmax = 0
      do 5 i = 1, n
        if( abs(p(i)) .gt. absmax ) absmax = abs(p(i))
5   continue
      if( absmax .lt. 1.e-8 ) go to 200
c
      v = A*p
      do 7 i = 1, n
        v(i) = 0
        do 6 j = 1, n
          v(i) = v(i) + A(i,j)*p(j)
6   continue
7   continue
c
      w = p'*v

```

図 4.2 共役傾斜法の FORTRAN によるプログラム

```

      w = 0
      do 8 i = 1, n
        w = w + p(i)*v(i)
8   continue
c
      alpha = (p'*r)/w
      alpha = 0
      do 9 i = 1, n
        alpha = alpha + p(i)*r(i)
9   continue
      alpha = alpha/w
c
      x = x + alpha*p
      do 10 i = 1, n
        x(i) = x(i) + alpha*p(i)
10  continue
c
      r = r - alpha*v
      do 11 i = 1, n
        r(i) = r(i) - alpha*v(i)
11  continue
c
      beta = -(r'*v)/w
      beta = 0
      do 12 i = 1, n
        beta = beta - r(i)*v(i)
12  continue
      beta = beta/w

```

LAMAX-S プロトタイプ版として PC98版が 1991年に完成し、OR学会に発表されました。PC98版の仕様について、補足しておきます。なお、PC98版は無償で配布しておりますので、ご興味のある方はシステム計画研究所 (03-5489-0211) までご連絡ください。

PC98版の仕様

ハードウェア	NEC PC98シリーズ
数値演算プロセッサ	ある方が望ましい
OS	MS-DOS(マイクロソフト社)
コンパイラ	MS-FORTRAN (同上)
ライブラリ	LINPACK
行列の制限	スパース行列は対応しない。
言語仕様	本稿と若干の違いがあるが、本稿の文法に対応するツールが用意されている。

#### 参 考 文 献

- [1] ASNOP研究会, 非線形最適化プログラミング, 日刊工業新聞社, 1991
- [2] 八巻直一 他, 行列演算用言語 LAMAX-S について, 1991年ORの学会春期予稿集, 1991
- [3] 八巻直一 他, 行列演算用言語 LAMAX-S, オペレーションズ・リサーチ, Vol.37, No.12, 1992
- [4] Dongarra, J. J., Bunch, J. R., Moler, C. B, and Stewart, G. W., LINPACK Users' Guide, SIAM, 1979
- [5] 八巻直一, 内田智史, 本郷 茂, 行列演算用言語 LAMAX-S, オペレーションズ・リサーチ, Vol.37, No.12, 1992
- [6] 森正武他, 数値計算, 岩波書店, 1982

```

c   p = r + beta*p
    do 13 i = 1, n
      p(i) = r(i) + beta*p(i)
13  continue
100 continue
    write(*,*) ' 収束せず'
200 write(*,*) ' 解'
    write(*,*) ( x(i), i=1,n )
end

```