



Title	FORTTRAN 77 (V) のススメ
Author(s)	大中, 幸三郎
Citation	大阪大学大型計算機センターニュース. 1982, 47, p. 59-68
Version Type	VoR
URL	<a href="https://hdl.handle.net/11094/65547">https://hdl.handle.net/11094/65547</a>
rights	
Note	

*The University of Osaka Institutional Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

# FORTRAN 77(V)のススメ

大阪大学大型計算機センター研究開発部

大 中 幸三郎

## 1. はじめに

FORTRANのJIS規格が改訂され<sup>1)</sup>、大阪大学大型計算機センターでもFORTRAN 77という名前で利用可能となっている。新規格のFORTRAN(以下FORTRAN 77と記す)は、旧規格のFORTRAN(以下FORTRAN 66と記す)にくらべて、機能の大巾な追加と変更がなされている<sup>2~4)</sup>。このような大改訂がなされて、使いやすい言語、あるいは時代の流れに合った言語となることは、好ましいことである。しかしながら、FORTRAN 77を用いたジョブの実行に要したCPU時間が、FORTRAN 66の場合と、ほぼ同程度以下でなければ、FORTRAN 66から77への移行はかなりむづかしい。

本稿では、FORTRANの仕様そのものではなく、ACOSのFORTRAN 77と66のコンパイラを用いて、ジョブを実行した場合の、CPU時間の比較を行う。テストに使用したプログラムは21本であり、用途、アルゴリズム、作成者の個性などの広がりを考えて、大阪大学内の利用者から7本、他大学の大型計算機センターから14本を入手した。21本程度のテストでは、とても充分とは言えないが、一応の目安になるものと思う。

## 2. テストプログラムの特徴とCPU時間

21本のテストプログラムの用途、アルゴリズムなどについては、明確なものと不明確なものとの混在している。しかしながら、ある程度の特徴はソースプログラムから判定が可能であり、その結果を表1に示す。これらのテストプログラムは、RとVの両モードで実行させるために、記憶容量は、Rモードで実行できる範囲内であり、記憶容量の大きなプログラムは含まれてはいない。

次にテストプログラムのコンパイル、リンクおよびランに要したCPU時間の合計を表2に示す。コンパイラは9月末日現在でサービス中のものと同一のコンパイラを用いている。また、コンパイラオプションは、条件をそろえることと、既定値を考慮して、下記のようにえらんだ。

FORTRAN 66	{ Rモード	NOPTZ または OPTZ, HEX, NROUND.
	{ Vモード	NOPTZ または OPTZ, HEX, NROUND.
FORTRAN 77	{ Rモード	OPT=1, NROUND. (OPT=2 は現在使用できない。)
	{ Vモード	OPT=1 または OPT=2, NROUND, INLINE=0, NIAP.

表2には1/1000秒まで示しているが、CPU時間は、システム構成、負荷の大小などに依存す

る。たとえ同一のシステム、同一のジョブであっても、±2～3%の差異は生ずるから、表2の値にもその程度の誤差を含むものと考えられる。

表1 テストプログラムの特徴

プログラム	特 徴 (用途、アルゴリズムなど)
A	組込み関数の精度測定
B	組込み関数の速度測定
C	数値積分
D	数値積分(二重積分)、ガウス法
E	逆行列、ガウス・ジョルダン法
F	固有値、固有ベクトル、ヤコビ法
G	固有値、固有ベクトル、QR法
H	連立一次方程式、ガウスの消去法
I	連立一次方程式、LU分解法
J	連立一次方程式、LU分解法、コレスキー法
K	熱伝導、偏微分方程式
L	電磁気学のシミュレーション
M	構造力学のシミュレーション
N	有限要素法
O	構造力学、有限要素法
P	素数、整数型演算
Q	グラフ理論、整数型演算
R	固体中の磁気モーメント、倍精度複素数型演算が多い
S	$\chi^2$ 分布
T	位相空間体積
U	ワーク用のディスクファイルを2個使用

表2 CPU時間(コンパイル、リンク、ランの合計)

(単位 秒)

言語 モード 最適化 プログラム	FORTRAN 66				FORTRAN 77		
	R		V		R	V	
	NOPTZ	OPTZ	NOPTZ	OPTZ	OPT=1 (NOPTZ)	OPT=1 (NOPTZ)	OPT=2 (OPTZ)
A	8.452	8.622	10.283	10.100	8.529	9.012	8.938
B	44.204	43.277	54.946	53.680	42.505	44.789	43.766
C	52.542	52.058	113.497	113.715	65.279	67.659	58.187
D	41.166	38.466	101.870	100.976	44.088	43.952	40.550
E	2.177	1.691	3.068	2.720	3.068	2.832	2.355
F	3.935	2.042	5.371	3.955	4.494	3.737	2.492
G	74.573	22.205	100.111	57.500	65.138	72.465	20.127
H	129.106	37.287	164.095	105.913	113.176	106.295	33.715
I	24.045	6.421	29.048	16.545	21.548	23.709	5.718
J	76.810	22.000	93.914	52.445	72.312	61.800	20.441
K	6.008	4.921	9.345	8.139	6.560	7.155	5.197
L	2.978	2.911	4.335	4.305	3.259	3.944	3.057
M	84.645	73.337	129.002	126.977	90.806	135.772	84.069
N	189.643	109.591	239.013	144.143	214.265	225.949	95.930
O	8.038	7.320	10.836	10.191	9.054	10.498	9.847
P	14.308	13.006	14.556	12.484	13.880	15.107	12.847
Q	45.263	44.955	61.076	60.575	43.787	56.311	42.233
R	37.598	36.363	75.777	74.867	37.687	36.858	35.307
S	48.972	47.874	108.295	108.361	52.427	59.303	56.607
T	33.771	29.387	69.541	67.379	38.772	41.995	37.913
U	2.610	2.254	12.480	11.865	6.112	9.433	2.421
合計	930.844	606.488	1410.459	1146.835	956.746	1038.575	621.717

### 3. FORTRAN66 と 77 の比較

各々のテストプログラムによって、多少のばらつきはあるが、表2から次の三つの傾向を読み取ることができる。

- (1) FORTRAN66 のVモードは、CPU時間が35～90%程度長くかかる。
- (2) NOPTZ (OPT=1) の場合には、FORTRAN66 と 77 のRモードはほぼ同一であるが、

FORTRAN 77のVモードでは、10%程度長くかかる。

- (3) OPTZ(OPT=2)の場合には、FORTRAN 66のRモードとFORTRAN 77のVモードはほぼ同一である。

まず(1)について考えてみよう。FORTRAN 66のVモードでCPU時間が余分に要する原因には、ページフォルト、セグメント制御およびドメイン制御のオーバーヘッドが考えられる。ところが、2.に述べたように、テストプログラムの必要とする記憶容量は小さく、また、実行レポート上からも、ページフォルトの影響は無視できることがわかる。したがって、主たる原因はFORTRAN 66のVモード固有の、実行時におけるセグメントとドメインの制御のためと考えられる。これらの影響を、FORTRANのソースプログラムの修正によって救済することは、実質上、不可能である。したがって、利用者の立場からみると、FORTRAN 66のVモードを使用することは、センターに負担金を寄付しているようなものであり、早急にFORTRAN 77のVモードに移行すべきである。

次に(2)のFORTRAN 77のVモードについて考えてみよう。この場合にも(1)の場合と同様に、セグメントとドメインの制御の影響により、10%の差を生じているものと推定している。FORTRAN 66のVモードにくらべて、差異が少ない原因は、それらの制御方式が大巾に変更されているためである。最適化を行わない場合には、セグメントとドメインの制御のための命令の最適化も行われない。したがって、10%という値の大小は別として、Rモードの場合よりもCPU時間が延びることは、やむを得ないだろう。

最後に(3)について考えてみよう。最適化が行われた場合には、前述のセグメントとドメインの制御のための命令も最適化の対象となる。したがって、これらの命令の影響はゼロにはならないが、表2のデータでは、ほとんど無視できる範囲となっている。

(2)および(3)より、FORTRAN 66のRモードからFORTRAN 77のVモードへ、積極的に移行するほどの結果は得られない。しかしながら、2.にも述べたように、表2の測定に用いたコンパイラオプションは、条件をそろえることを考慮している。FORTRAN 77のVモードには、表2の場合よりも、CPU時間を短縮するためのコンパイラオプションがあり、次節以降に、これらの効果について述べる。

#### 4. インライン展開の効果

FORTRAN 77のVモードでは、下記の演算を、外部手続きの呼び出しとするか、あるいは内部手続きとしてインライン展開するかを選択できる。<sup>5,6)</sup>

組込み関数            SQRT, DSQRT, SIN, DSIN, COS, DCOS, EXP, DEXP,  
                              ALOG, DLOG, ALOG10, DLOG10.

べき乗の演算            I\*\*I, R\*\*I, D\*\*I, R\*\*R, D\*\*D.

ただし、Iは整数型、Rは実数型、Dは倍精度実数型とする。

複素数の乗除算 C\*C, C/C, CD\*CD, CD/CD.

ただし、Cは複素数型、CDは倍精度複素数型とする。

インライン展開の指定は、コンパイラオプションによって行う。バッチ処理の場合のオプションを次に示す。<sup>5,6)</sup> TSS処理の場合のオプションは少し異なるが、ここでは述べない。<sup>5,7)</sup>

INLINE = 0 すべてを外部手続きとする。

INLINE = 1 最深のDOループ中で引用されるときに、インライン展開とする。

INLINE = 2 プログラム中で引用されるすべてに対して、インライン展開とする。

インライン展開を行うことにより、実行時のCPU時間を短縮できることが予想される。表3にその例を示す。表3のCPU時間には、コンパイルとリンクは含まれていない。インライン展開によって、コンパイルとリンクに要するCPU時間も多少変化するが、無視できる範囲であり、インライン展開の効果が直接影響を与える実行時の値を比較した。

表3 インライン展開の効果

最適化 インライン プログラム	OPT=2 (OPTZ)		
	INLINE = 0	INLINE = 1	INLINE = 2
G	19.717 1	21.811 1.106	19.045 0.965
M	83.660 1	73.055 0.873	71.358 0.852
R	34.413 1	33.986 0.987	27.091 0.787
S	56.246 1	42.263 0.751	39.550 0.703
A ~ U の合計 (上記4本を含む)	606.297 1	583.762 0.969	測定せず

上段 コンパイルとリンクを除いたCPU時間(単位 秒)

下段 OPT=2, INLINE=0 を1とした比

インライン展開によるCPU時間への効果の主な要因を次に示す。

- 外部手続きを呼び出すための手順が省略できるため、CPU時間が短縮される。
- 引数などのチェックを省略しているため、<sup>6)</sup> CPU時間が短縮される。
- プログラムの手続き部分に必要な記憶容量が増加する。したがって、キャッシュメモリーのヒット率が低下し、CPU時間が延びることがある。

表3に示したG、M、RおよびSのプログラムは、インライン展開の影響が大きい例である。た

だし、プログラムGのように、 $INLINE = 0$  より  $INLINE = 1$  の方がCPU時間が長くかかるものは少数である。21本のテストプログラムの中で、CPU時間の増加が5%をこえるものは、プログラムGの他にはIとUにすぎない。

表3には、A～Uのすべてに対して  $INLINE = 1$  を指定した場合も示している。プログラムMとSによるデータと、A～Uすべての合計を比較すれば、プログラムの内容を問わずにインライン展開を行っても、効果のないことがわかる。インライン展開の効果をソースプログラム上から判定する指針を示すことはむづかしい。しかしながら、インライン展開の対象となる演算を数多く使用している場合には、インライン展開を行ってみるとよい。この場合に、 $INLINE = 1$  で効果が無かったからといって、あきらめてはいけない。プログラムRのように、 $INLINE = 2$  で初めて効果のあらわれる場合があることに注意を要する。

## 5. 最適化の効果

FORTRAN 77のVモードでは最適化のレベルは、次の四つをコンパイラオプションによって選択できる。<sup>4, 5)</sup>

- OPT=0 最適化をまったく行わない。
- OPT=1 一文中での最適化を行う。FORTRAN 66のNOPTZに対応する。
- OPT=2 複数の文にわたる最適化を行う。FORTRAN 66のOPTZに対応する。
- OPT=3 できるかぎりの全体的最適化を行う。ただし、一部でJIS規格<sup>1)</sup>に違反する。

実行時のCPU時間ももっとも短かくてすむのは、当然ながらOPT=3であろう。したがって、本節ではOPT=3の場合について述べる。この場合の最適化の項目は、OPT=2以下とくらべて、次のものが追加されている。<sup>5)</sup>

共通式およびDOループ内の不変式の評価時点の変更。

実引数が定数の場合の退避を削除。

割当て形GO TO文のエラーチェックの削除。

ゼロ回DOループの判定を削除。

レジスタ上にある実数値の共用。

DOが満足された時に、DO変数の終値は保証されない。

OPT=3を指定して実行した場合のCPU時間の例を表4に示す。OPT=3のときには、インラインの指定がなければ、 $INLINE = 2$ とみなされるから、<sup>4, 5)</sup>OPT=3かつ $INLINE = 2$ の値をも示した。なお、表4のCPU時間には、表3の場合と同様に、コンパイルとリンクは含まれてはいない。

残念ながら、A～Uの21本の約2/3をOPT=3で実行したが、表4に示したように、OPT

=3 が有効な例を見出すことはできなかった。しかし、上記の項目は、常識的には、CPU時間を短くする方向に働くはずであり、特定のプログラムでは有効であろう。

表4 最適化の効果

最適化 インライン プログラム	OPT = 2	OPT = 3	
	INLINE = 0	INLINE = 0	INLINE = 2
G	19.717 1	19.347 0.981	19.227 0.975
H	33.402 1	32.978 0.987	33.181 0.993
R	34.413 1	34.301 0.996	26.390 0.766

上段 コンパイルとリンクを除いたCPU時間(単位 秒)  
下段 OPT=2, INLINE=0 を1とした比

## 6. 統合アレイプロセッサ(IAP)の効果

FORTRAN77のVモードでは、コンパイラオプション(NIAPまたはIAP)あるいは、ベクトル関数の呼び出しによってIAPを使用することができる。<sup>4,8,9)</sup> 本節では、ソースプログラムの修正を行わなくてもよいことを考えて、コンパイラオプションによるIAPの使用について述べる。

IAPオプションにより、ベクトル命令が生成され、IAPを使用するオブジェクトプログラムが作られるが、そのためにはいくつかの条件を満足しなければならない。<sup>5,8)</sup> 適用条件の個々については省略するが、代表的なものを四つだけ示しておく。

配列を使用している。

最深のDOループにのみ有効。

最深のDOループ内にIF文などの分岐命令がない。

最深のDOループ内に、特定の組込み関数以外の外部手続きの引用がない。

IAPオプションを指定して実行した場合のCPU時間の例を表5に示す。なお、表5のCPU時間には、表3および表4の場合と同様に、コンパイルとリンクは含まれていない。

表5に示したプログラムは、行列関係の計算であり、IAPが有効に働くことが期待されるプログラムである。たしかに、プログラムGとJでは著しい効果があらわれている。ところがプログラムHではまったく効果がない。その主たる原因は、IAPの適用条件を満足する度合であり、この場合にはIF文の有無である。たとえば、プログラムH中のピボットの選択の部分には必ずIF文があらわれてくる。表5の値はソースプログラムをまったく修正しない場合であるが、プログラム

表5 IAP の効果

プログラム	最適化	OPT=2		OPT=3
	インライン	INLINE=0		INLINE=2
	IAP	NIAP	IAP	IAP
G		19.717 1	14.592 0.740	13.719 0.695
H		33.402 1	32.048 0.959	36.637 1.096
J		19.672 1	9.625 0.489	9.531 0.484

上段 コンパイルとリンクを除いたCPU時間(単位 秒)

下段 OPT=2, INLINE=0, NIAPを1とした比

Hでは、ベクトル関数の引用を含むソースプログラムの一部修正によって、CPU時間を40~50%短縮することができる。

上記の例からもわかるように、IAPの効果はソースプログラムに大きく依存し、そのわずかな修正によってCPU時間が大巾に変動する。修正の指針としては、ACOS以外の計算機との互換性を考慮する場合には、IAPの適用条件を満足するように変更するしかない。互換性を考慮しなくてもよい場合には、ベクトル関数を積極的に使用することである。FORTRAN 77のVモードで使用できるベクトル関数は、ベクトル関数を使用しなければ、必ずIF文が必要となり、現在のIAPオプションでは適用条件に違反するものである。IAPの適用条件は、今後、徐々にではあるがゆるめられるものと思う。しかしながら、ACOSの場合にかぎらず、一般にアレイプロセッサを有効に使用するためには、従来とは異なった観点からの、アルゴリズムとプログラムの見直しが必要となろう。

## 7. コンパイラオプションの選択

FORTRAN 77のコンパイラオプションの既定値の主なものは、

V, OPT=1, NROUND, INLINE=0, NIAP,

である。これらを既定値としてえらんだ要因は、次の四つである。

- (1) RよりもVの方が機能が多様であり、コンパイラオプションの選択によって、CPU時間を短縮できる可能性が高いこと。
- (2) デバッグ機能のSUBCHKとFDSは、OPT=0またはOPT=1でのみ有効であること。

(3) ROUND よりも NROUND の方がCPU時間が短かくてすむこと。<sup>10)</sup>ならびに、NROUNDのときにのみ IAP が使用できること。<sup>5)</sup>

(4) 最適化、インライン展開、IAPの使用の機能は、意識して用いてほしいこと。すなわち、既定値としては、基本的な組合せとしたいこと。

上記の要因には、異論があるかもしれない。とくに(4)については、保守的にすぎるとの意見もあるだろうが、了承していただきたい。

利用者の立場からのコンパイラオプションの選択について、少し考えてみよう。まず、デバッグの段階にある場合には、デバッグ機能および、4.に述べたインライン展開時の引数などのチェックの省略から考えて、すべて既定値を用いることが順当であろう。次に、デバッグ段階をすぎて、本格的に実行させる場合であるが、この場合には、一般的な指針を示すことは困難である。実行時のCPU時間が極端に短いジョブは、既定値でもよいが、ある程度以上のCPU時間が必要なジョブに対しては、既定値では能率が悪い。とくに、大きなジョブをデータをかえて、多数回実行させるときには、既定値は避けるべきである。このような場合には、3.に述べたことから、少なくともOPT=2を指定するべきであると言ってもよいが、それ以上のことを言うのはむづかしい。常識的に考えて、OPT=3、INLINE=2、IAPの場合が、最もCPU時間が短くなるはずである。しかしながら、4~6.に述べたように、これらの効果はソースプログラムに大きく依存する。各オプションの意味を考慮して、適切な選択を心掛けるように努力されたい。

## 8. お わ り に

1.にも述べたように、21本のテストプログラムに対する考察程度では、一般的あるいは断定的な結論を出すことはできない。しかしながら、FORTRAN 77のVモードはFORTRAN 66のRモードと比較して、CPU時間がほぼ同程度、あるいはコンパイラオプションのえらび方によっては、より短いCPU時間で実行できる目安は得られた。これらのテストプログラムは、2.にも述べたように、記憶容量の大きなものは含まれていない。Vモードでのみ実行可能なプログラムについては、2、3についてしか比較していないが、表2と同じ傾向が得られている。すなわち、FORTRAN 66のVモードを用いることは、まったく利点がないと言ってよいだろう。

JIS規格の改訂および、ACOSのFORTRAN 77の機能と性能から考えて、新しくプログラムを作成するときには、プログラムの大小にかかわらず、最初からFORTRAN 77のVモードを使ってほしい。また、既存のプログラムについても、FORTRAN 66のVモードからはもちろん、Rモードからも積極的に、FORTRAN 77のVモードに移行することを奨励する。既存のプログラムの移行に際しては、FORTRAN 77と66の文法仕様の相異を考慮する必要がある。この点については、本稿ではふれなかったが、21本のテストプログラムは本来FORTRAN 66用に作られたも

のにもかかわらず、一行の書き換えも必要とはしなかった。また、センター・ライブラリーの FORTRAN 77 の V モード化においても、数本のプログラムのごくわずかな修正でよかった。したがって、新、旧の JIS 規格の比較ではなく、ACOS においての FORTRAN 66 から 77 への書き換え作業は、通常、大きな作業量とはならないものと思う。

JIS 規格が改訂されたことにより、FORTRAN 77 は、今後の FORTRAN の中心となる言語である。当センターの運用においても、FORTRAN 66 から、FORTRAN 77 の V モードに重点をうつすように切り換えて行くことになる。本稿が、FORTRAN 77 の V モードへの移行の手助けとして、有用な資料となれば幸いである。

#### 〔 参 考 文 献 〕

- 1) JISハンドブック情報処理 1982, pp. 729-907, 日本規格協会(1982).
- 2) 大中, 後藤: FORTRAN 77 概説(1), 大阪大学大型計算機センターニュース, Vol.11, No.3, pp.59-78(1981).
- 3) 大中, 後藤: FORTRAN 77 概説(2), 大阪大学大型計算機センターニュース, Vol.11, No.4, pp.123-141 (1982).
- 4) 大中, 後藤: FORTRAN 77 概説(3), 大阪大学大型計算機センターニュース, Vol.12, No.1, pp.73-89(1982).
- 5) FGB 09-1 FORTRAN 77 (V) プログラミング説明書, 日本電気(1982).
- 6) FORTRAN 77 の V モードにおけるインライン展開について, 大阪大学大型計算機センター速報, No.94, p.10(1982).
- 7) FORTRAN 77(V)の RUN コマンドについて, 大阪大学大型計算機センター速報, No.96, pp.12-13(1982).
- 8) 泉谷, 北脇, 諸木: 統合アレイプロセッサについて, 大阪大学大型計算機センターニュース, Vol.12, No.1, pp.59-72(1982).
- 9) FGB 07-1 FORTRAN 77 文法説明書, 日本電気(1982).
- 10) 大中: FORTRAN ジョブから見た ACOS 900 の性能, 大阪大学大型計算機センターニュース, No.32, pp.49-56(1979).