並列スーパーコンピュータの演算性能

~ Fortran プログラムでの実測データ ~

渡部 善隆
(九州大学大型計算機センター)

新スーパーコンピュータ VPP700/56 の運用にあたり，センターでは何本かのテストプログラムを走らせ，データを収集しました．[1] で述べられている通り，運用側としての関心は，「入れ換えられる計算機である FUJITSU VP2600/10 に蓄積された利用者プログラムがスムーズに移行できるか？」でした．そのため，浮動小数点形式の差異 (VP2600/10 は IBM 形式， VPP700/56 は IEEE 形式です．) や翻訳時オプションの動作確認などの「精度」とあわせて，翻訳時間や実行時間，スループットなどの「性能」も重要なチェック事項となりました．

本稿では，それらの実測データの中から，スーパーコンピュータの唯一の存在理由である浮動小数点演算性能について得られた結果を中心に報告します．また，1 章では VP2600/10 で数値計算を行なっていた者の立場から見た VPP700/56 の使い勝手を簡単に報告します．

1 VPP700/56 の使い勝手

1.1 VPP700/56 の外観

VPP700/56 は，合計 56 台のプロセッサで構成される並列計算機です．各プロセッサ (PE) は独立したベクトル計算機として機能しますので，「ベクトル計算機が 56 台並んだものが VPP700/56 なんだな」とイメージできます．図 1.1 は外観の写真です．

図 1.1 : VPP700/56 の外観

1.2 VPP700/56 の性能諸元

利用者は従来の VP2600/10 の Fortran プログラムおよび C プログラムを VPP700/56 の 1PE でそのまま実行することができます．ただし互換性はソースレベルです．また，並列処理のための指示行 (拡張最適化制御行) またはデータ通信の手続きをソースプログラムに記述することで最大 32PE を用いた並列処理を行なうことができます．

VPP700/56 の 1PE あたりの性能は表 1.1 の通りです．

表 1.1 : VPP700/56 の主な諸元 (1PE あたり)

項目	性能
理論ピーク性能	2.2GFLOPS
ベクトルパイプライン数	7 本
浮動小数点レジスタ	64bit×32 個
汎用レジスタ	32bit×32 個
ベクトルレジスタ	128KB
キャッシュ	32KB×2 個
ネットワーク通信速度	570MB/ 秒 ×2
主記憶容量	2GB
記憶素子	SDRAM
主記憶スループット	18.2GB/ 秒

1.3 VP2600/10 との違い

VP2600/10 と VPP700/56 で大きく異なる点を表 1.2 に示します．

表 1.2 : VP2600/10 と VPP700/56 の差異

項目	VPP700/56	VP2600/10
計算機	ベクトル並列計算機 (1PE での実行も可能)	ベクトル計算機 (1CPU)
OS	UXP/V	MSP, UXP/M
浮動小数点形式	IEEE 形式	M(IBM) 形式
Fortran	Fortran 90 (+ 富士通並列化仕様)	FORTRAN 77 (+ 富士通仕様)
最大記憶域	1.7GB/PE* (最大 1.7×32GB)	400MB*
理論ピーク性能	2.2GFLOPS/PE (最大 2.2×32GFLOPS)	4.9GFLOPS

∗) 利用者が確保できる最大リージョンサイズです．

1.4　利用方法の変更

従来，VP2600/10 の利用はバッチ処理でした．OS は IBM 型 OS である MSP と Unix である UXP/M の 2 つが動作していました．しかし，実際に利用者が見た場合，フロントエンドを受け持つ汎用計算機 M-1800/20U の OS が何であるかが重要であり，バッチジョブを処理する計算機の OS の種類は重要ではありません．

VPP700/56 の OS は Unix System V Release 4 をベースとした分散並列オペレーティングシステム UXP/V に一本化されました．その代わり，従来の VP2600/10 と同様の書式のジョブ制御文を汎用計算機の MSP から投入し，結果を再び MSP で検索する機能 (M-VPP 連携機能) がサポートされました．UXP/M からは従来と同じ NQS でのバッチの投入方法がサポートされました．

従って，利用者は基本的にパラメータのわずかな変更に注意すれば，従来の処理形態を踏襲することができます．また，1, 2 月に VP2600/10 と VPP700/56 を併行して運用できたこともあり，移行時期の大きな混乱は見られませんでした．

1.5　会話型処理

1997 年 4 月より VPP700/56 の会話型処理のテスト運用を開始しました ([2])．

VPP700/56 の OS である UXP/V は，コマンドの若干の非互換を除けば汎用計算機の UXP/M とほぼ同じ仕様の Unix OS です．並列ジョブの実行はできないものの，翻訳レベルのデバッグやオブジェクトファイル，アーカイブライブラリの作成が会話的にできるようになり，利用者の使い勝手はかなり良くなったと思われます．

ただし，Unix の実質的な標準ツールである emacs, tcsh などをこちらでインストールしなければならないのはかなりの負担でした (アーキテクチャの違いから，なかなか make 一発とは行きませんでした．)．一度富士通主催の研究会でフリーソフトへの取り組みを聞いたところ，《パブリックドメインソフトウェアであっても、製品として提供するためには，著作者の了解および将来に渡っての製品責任および保守責任を果たす必要があることから，基本的には慎重に取り組んでまいります．》というつれない内容の電子メールが返ってきました．

また，何人かの MSP 利用者 (愛好者) と話をした感触では，MSP から Unix 系 OS に乗り換える際の最大の問題は，使い慣れたエディタ (MSP の場合 PFD) を捨て，`Control` や `Esc` などの怪しげなキーを連発するエディタを勉強しなければならないということのようです．

1.6　浮動小数点形式の違い

VPP700/56 の導入によって最も心配していたことは浮動小数点形式の違いによる混乱でした．しかし，対策として，実行時オプションの指定により形式の変換が可能なこと，浮動小数点形式を変換するサービスサブルーチンがサポートされたこと，また M-VPP 連携機能ではデータの変換が自動的に行なわれること，さらに，講習会でこの部分を重点的に話したことなどから，スムーズな移行ができたと思います．

1.7　VP2600/10 の資産の移行

表 1.2 の通り，VPP700/56 の 1PE での CPU 時間は理論性能比で計算すれば 2 倍以上に伸びることになります．この対策として，演算負担金の事実上の値下げを実施しました ([1], 6 章)．また，主記憶容量の増大にともない，VP2600/10 でサポートしてきた SSU(システム記憶装置) は廃止しました．

利用者の立場から見れば，従来の VP2600/10 のジョブを VPP700/56 の 1PE で実行した場合，演算負担金の変更と複数 PE によるジョブ処理により，特に繁忙期のターンアラウンドが向上すること，また，記憶領域を従来の 4 倍以上確保できるようになることから，極端な計算環境の劣化は感じていないだろう，と期待しています．

1.8　プログラム言語

VPP700/56 で利用できるプログラム言語は Fortran, C, C++ です．ともにベクトル計算が可能です．Fortran は Fortran 90 が正式にサポートされました．機能的には FORTRAN 77 の完全上位互換であり，富士通版の拡張仕様もそのまま継続サポートとなったため，ソースレベルでの移行は全く問題ありませんでした．

また，ベクトル版の C は，導入後のレベルアップでベクトル演算機能がかなり充実しました．C 版のサブルーチンライブラリがサポートされるという計画もあるそうで，今後積極的に宣伝する必要が出てくると思います．

1.9　並列化の手段

VPP700/56 は分散メモリ型並列計算機です．システムの構成は [1], 2 章を御覧下さい．自動並列化の手段は，1997 年 9 月現在サポートされていません．利用者は，自身で並列化プログラミングを行なう必要があります．

VPP700/56 での並列化の手段は次の通りです．

- 富士通独自仕様である Fortran 90/VPP を用いる
- Fortran または C から PVM を利用する
- Fortran または C から MPI を利用する
- Fortran または C から PARMACS を利用する

PVM, MPI, PARMACS はともにメッセージパッシングライブラリ (MPL) です．

表 1.2 にあるように，並列ジョブでは最大 54.4GB の記憶領域が確保できます．これは理論性能と違って現実に確保可能です．

一方の理論処理性能は，最大で 70.4GFLOPS となります．もっとも，何かの役に立つ数値計算プログラムでこの性能を出すことは無理です．

一応，無理のない宣伝として，並列化により理論ピーク性能の 70% 以上を実現するプログラムが実際に報告されてることから，アプリケーションレベルでの実行性能は VP2600/10 の 10 倍だということにしています．

導入時には並列ジョブの利用は全くないのではと危惧しましたが，現在のところ予想外に利用されている ([1], 4.3 節) と思います．

2 翻訳性能

2.1 テスト環境

以降の章で報告するテストデータは，すべて Fortran の倍精度 (64 ビット) 実数データ型演算を行なうプログラムによる結果です．Fortran コンパイラ名は，1PE で動作するものが Fortran 90/VP V10L10，複数 PE で動作するものが Fortran 90/VPP V10L10 です．

2.2 翻訳時間の比較

[10] によれば，Fortran 90/VP は従来のベクトル化や最適化にプラスしてスカラープロセッサ向けの最適化を行っています．そのため翻訳時間は一般に VP2600/10 と比較して長くなります．

実際に VP2600/10 および汎用計算機 M-1800/20U と比較してどれくらい遅くなるかを，今回のスーパーコンピュータの調達で用いた性能試験プログラムを用いて測定しました．

プログラムの概要は表 2.1 の通りです．

表 2.1：性能試験プログラム群

番号	プログラムの概要	行数
job1	粘弾性流動解析	2571
job2	回転非同心円筒内の流体モード解析	2086
job3	シアー表面波の線形安定性問題	1218
job4	Volterra 型積分方程式の離散近似	486
job5	Navier-Stokes 方程式の有限要素解	4289
job6	Stokes 方程式の a priori 誤差評価	2090

行数はコメント行も含みます．比較の計算機 VP2600/10, M-1800/20U のコンパイラは双方 FORTRAN77 EX/VP V12L10 です．各測定日時は VP2600/10 が 1996 年 3 月，M-1800/20U は 1997 年 9 月 2 日，VPP700/56 の “Dec. 1996” は，1996 年 12 月 5 日に富士通沼津工場で，“Sep. 1997” は 1997 年 9 月 1 日です．オプションはメーカー推奨の標準値を採用しています．

表 2.2 に翻訳時間を，図 2.1 にその値をグラフ化したものをそれぞれ示します．翻訳時間は実行プログラムを作成するまでの CPU 時間を UXP の `timex` コマンドで計測しました．従って，編集・結合の時間も含みます．単位は秒です．有効数字を 2 桁としそれ以下は切捨てています．

表 2.2：翻訳性能の比較

job 名	VPP700/56 (Sep.1997)	VPP700/56 (Dec.1996)	VP2600	M-1800
job1	46sec.	58sec.	24sec.	15sec.
job2	43sec.	58sec.	23sec.	15sec.
job3	26sec.	36sec.	14sec.	9.2sec.
job4	5.4sec.	7.9sec.	3.6sec.	2.3sec.
job5	51sec.	74sec.	43sec.	28sec.
job6	21sec.	32sec.	13sec.	7.5sec.

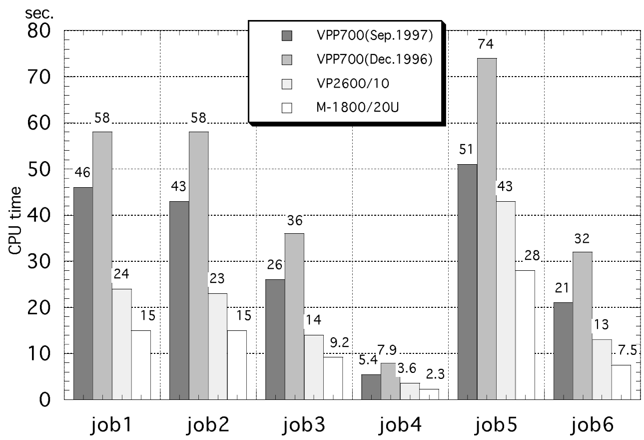

図 2.1：翻訳性能の比較 (表 2.2 のグラフ)

運用開始直前の測定結果では平均で VP2600/10 の約 2.3 倍の翻訳時間でしたが，最近の測定では約 1.6 倍にまで短縮しています．これは約 1.4 倍の高速化です．

実際に数値計算を行なう場合には，翻訳の度に繰り返し数千行のソースを翻訳するようなことはなく，手続き単位にオブジェクトモジュールを作成し，管理する利用者が多いはずです．

しかしながら，MSP からジョブを投入する場合には，中間段階のオブジェクトファイルを保存できないこと，また，VPP Fortran の PE 台数無依存の (どうも文法違反らしいです．) テクニックを用いない限り手続き単位で PE 台数を指定することが現状では避けられないことから，現在の VP2600/10 の約 1.6 倍，M-1800/20U の約 2.6 倍という翻訳時間は十分とはいえません．

今後のコンパイラのチューニングを大いに期待しています．

3 単一 PE の演算性能

3.1 利用者プログラムでの測定

計算機を更新する際に，従来の資産がそのまま継続して利用できるかどうかは重要なポイントとなります．今回のスーパーコンピュータの調達では，単一 PE の処理性能試験として，表 2.1 にあげたセンターの利用者プログラムの代表的と思われるものを 6 本用意しました．1CPU であった VP2600/10 を意識して作成されたこれ

らのプログラムが，提案されるシステムの単一 PE においてそのまま実行可能であり，かつ，計算機環境の継続性の観点から，極端な実行時間の劣化が生じないことを確認することが性能試験の目的でした．表 3.1 は，各ジョブが必要とする記憶領域と， VP2600/10 で実行した場合の VU 率 (全 CPU 時間の中でベクトルユニットが働いた割合) です．

表 3.1： 記憶領域と VU 率

番号	記憶領域	VU 率
job1	176MB	92%
job2	20MB	66%
job3	64MB	87%
job4	246MB	92%
job5	240MB	96%
job6	344MB	94%

メモリーサイズは VP2600/10 の利用者が確保できる大きさである 400MB に依存したものです．

性能評価試験の条件としては，基本的にコメント文字の修正・削除など移植に必要な最低限の修正を行なう以外は，ソースプログラムをそのまま翻訳・実行することとしました．一部のプログラムについては，同等な機能を持つ数値計算ライブラリとの差し替えや翻訳指示行の挿入によって大幅な性能向上が予想されるものもありました．しかし，利用者がこれまでのプログラムを「何も考えずに」流した場合の処理性能がより重要と考え，これらのチューニング行なわないことにしました．

表 3.2 に実行性能を，図 3.1 にグラフ化したものをそれぞれ示します．測定環境は 2 章と同じです．

表 3.2： 実行性能の比較

job 名	VPP700/56 (Sep.1997)	VPP700/56 (Dec.1996)	VP2600/10
job1	571sec.	565sec.	403sec.
job2	58sec.	52sec.	39sec.
job3	150sec.	146sec.	83sec.
job4	94sec.	99sec.	56sec.
job5	642sec.	641sec.	402sec.
job6	310sec.	309sec.	211sec.

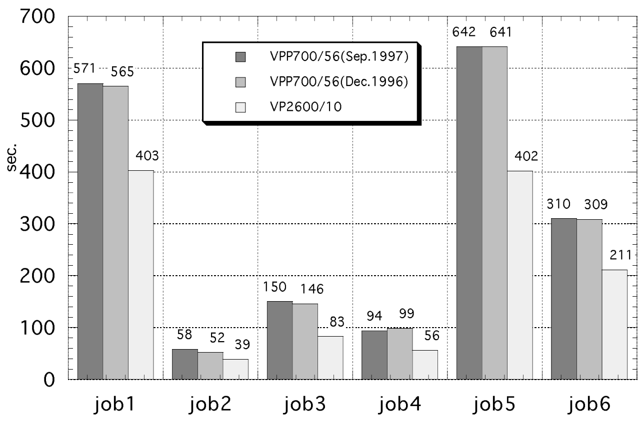

図 3.1： 実行性能の比較 (表 3.2 のグラフ)

表の時間はシステム CPU 時間とユーザー CPU 時間を合計した時間です．

理論ピーク性能は 0.44 倍ですが，このプログラムに限れば平均で 0.65 倍という結果が得られました．つまり，ピーク性能比ほどに実際のプログラムは遅くならないことが期待できます．

最新の測定結果の方が運用前の測定結果と比較してわずかながら悪くなっているジョブが見られます．原因として，テスト環境の違い (前回の測定は他のジョブがまったくない状態であるのに対し，今回は 1 利用者ジョブとして投入しています．また，ベクトルメモリーの割り当ても異なります)，コンパイラの最適化による実行性能への影響などが考えられますが，誤差の範囲だと見なしています．

また，参考までに，東北大学大型計算機センターの SX-3/44R，および，名古屋大学大型計算機センターの VPP500/42 で実行した結果を表 3.3, 図 3.2 に示します．測定は 1996 年 3 月に行ないました． Fortran コンパイラは東北大学が FORTRAN77/SX，名古屋大学が FORTRAN77 EX/VP です．翻訳時オプションはセンターの標準値を指定しました． CPU 時間は，それぞれのジョブログの出力によっています．

job5, job6 は NUMPAC のサブルーチンを利用しており， SX-3/44R でのテストでは job5, job6 を東北大学提供の NUMPAC サブルーチンに差し替えて実行しています．その他のプログラムの修正は行なっていません．VPP700/56 のデータは 1997 年 9 月のものです．

表 3.3： 実行時間の比較

job 名	VPP700/56	VPP500/42	SX-3/44R
job1	571sec.	576sec.	636sec.
job2	58sec.	56sec.	31sec.
job3	150sec.	140sec.	57sec.
job4	94sec.	106sec.	118sec.
job5	642sec.	687sec.	190sec.*
job6	310sec.	358sec.	168sec.*

* SX-3/44R 版 NUMPAC を使用

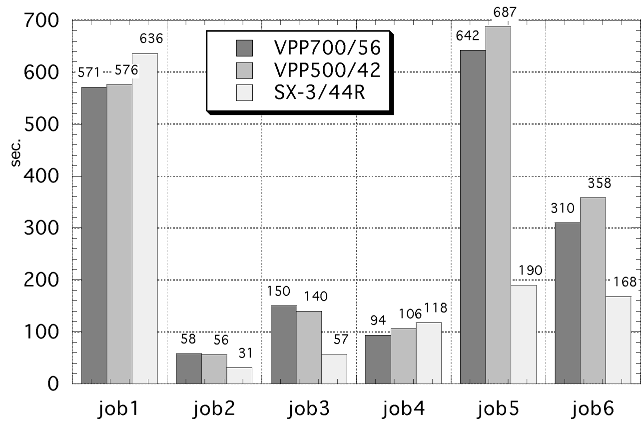

図 3.2： 実行時間の比較 (表 3.3 のグラフ)

他の計算機で実行した大きな理由は，性能評価試験プログラムを違う環境で実行し，正しい結果が得られているかを確認することにありました．もちろん，計算機のきちんとした性能評価をするためには，各プログラムのコスト分布やベクトル化状況などをきちんと分析する必要があります．が，計算機によって得意なプログラムと不得意なプログラムがあるということは少なくとも言えそうです．

3.2　スカラー演算性能

次に，VPP700/56 のスカラー演算を測定したデータを紹介します．汎用計算機 M-1800/20U と比較可能とするため，job4 と job6 を M-1800/20U でも実行可能となる (約 100MB の記憶領域) ように修正し，それぞれを job4-a, job6-a とします．表 3.4 はベクトル化オプションを有効とした場合の実行に要する CPU 時間です．M-1800/20U の理論最大性能は 1.2GFLOPS です．

表 3.4：ベクトル演算性能

ジョブ名	VPP700/56	M-1800/20U
job4-a	12sec.	25sec.
job6-a	1.7sec.	2.2sec.

job4-a, job6-a に対し，ベクトル化を抑止する翻訳時オプションを指定した実行結果を表 3.5, グラフを図 3.3 に示します．S-4/1000E は，センターでサポートしているライブラリサーバーでの実行結果です．

表 3.5：スカラー演算性能

ジョブ名	VPP700/56	M-1800/20U	S-4/1000E
job4-a	1291sec.	522sec.	2244sec.
job6-a	69sec.	48sec.	127sec.

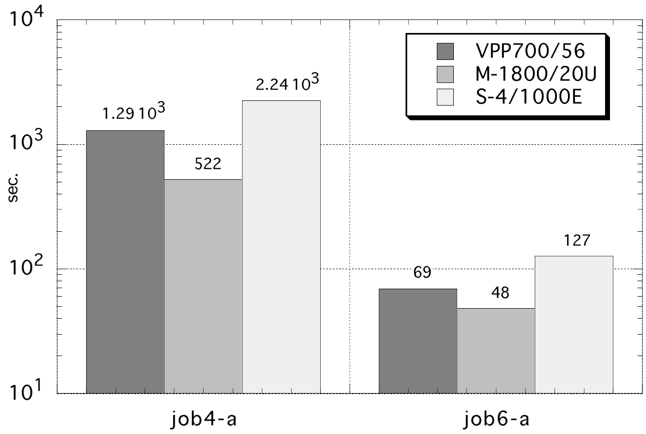

図 3.3：実行性能の比較 (表 3.5 のグラフ)

job4-a については，汎用機の倍以上の時間がかかってしまいました．少なくともこの二つのジョブの結果を見る限り，VPP700/56 のスカラー演算性能はあまり優れていないように思えます．

3.3　基本的な線形計算

この節では，Fortran 90 (cf.[3], [5]) の新しい機能であるベクトルおよび行列の乗算関数が “使いものになるか” を調べることも兼ねて行なったごく基本的な線形計算のテストについて報告します．

○行列積 (MATMUL)

Fortran 90 には，組み込み関数 MATMUL が標準でサポートされています ([3])．アルゴリズムは処理系依存です．ブラックボックスとなるため気味が悪いことは否定できませんが，従来の FORTRAN 77 のサブルーチンの様に整合寸法や次元数をいちいち入力する手間が省けることは魅力です．

テストでは次元 n の正方行列 A, B を乱数発生の関数 RANDOM_NUMBER で作成しました．また，比較のため富士通提供の SSL II/VP([13]) の DVMGGM と，2 回アンローリングを施した自作の行列積サブルーチン MYLIB を用意しました．CPU 時間の測定は組み込み関数 CLOCKV を手続きの前後に埋め込んで行ないました．

図 3.4 は演算数を $2n^3$ 回とした FLOPS 値をプロットしたものです．“MATMUL(July)” はレベルアップ以前に測定したデータ，その他は 1997 年 9 月 2 日の結果です．

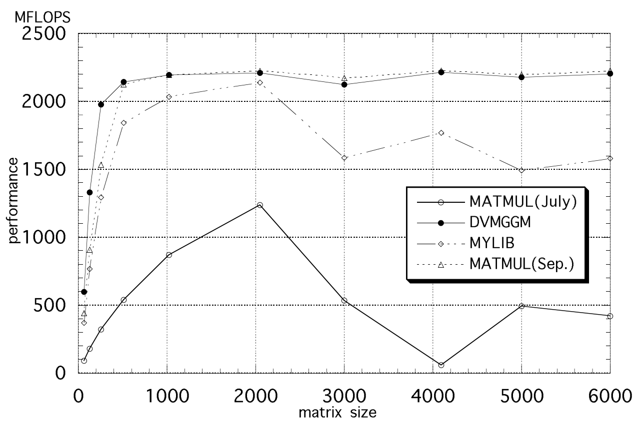

図 3.4：行列積の性能

Fortran 90/VP のレベルアップにより，MATMUL の性能が DVMGGM と同等にまで向上したことがわかります．

○ベクトルの内積

次の結果は，同じく Fortran 90 で組み込み関数となったベクトルの内積を求める関数 DOT_DRODUCT と，自作の関数 (何もテクニックは使っていません) MYLIB とを比較したものです．横軸は対数とし，ベクトルの長さ n を意味します．縦軸は演算を $n(n-1)$ 回とした FLOPS 値をプロットしています．図 3.5 がベクトル演算，図 3.6 がスカラー演算の結果です．測定は 1997 年 9 月 2 日に行ないました．“○”, “●” の部分が測定した n に対応しています．

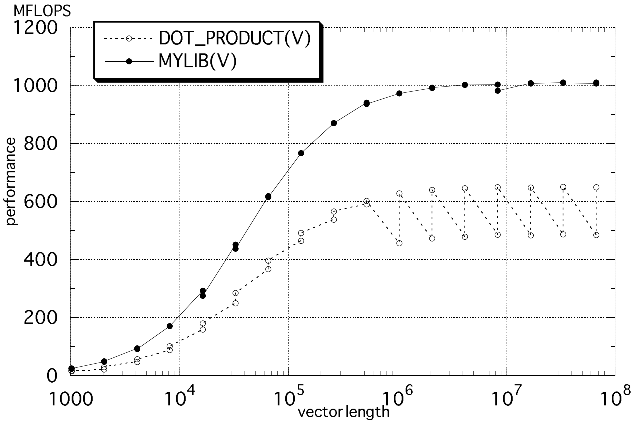

図 3.5 : ベクトルの内積の性能 (ベクトル)

グラフが “ふらふら” しているのは，バンクコンフリクト (cf. [4]) の影響と思われます (意図的に n を 2 のベキ乗になるような値を設定しています)．

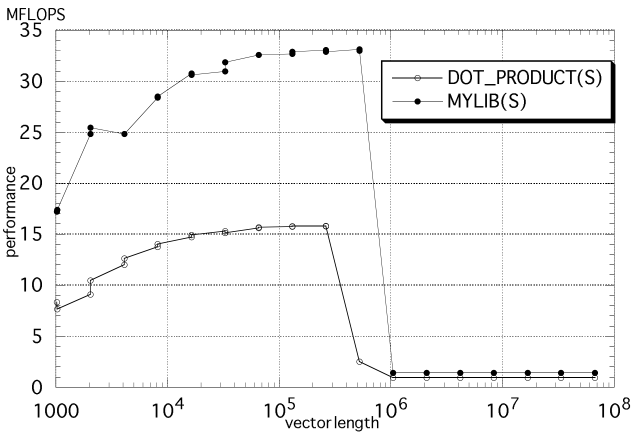

図 3.6 : ベクトルの内積の性能 (スカラー)

スカラー演算で極端に性能が悪くなった原因は今のところわかりません．

両方のテストともに，`DOT_DRODUCT` は自作の関数に大きく差をつけられています．今後のチューニングに期待します．

3.4　メモリーアクセス

次に，n 次元の正方行列を以下の方法でアクセスした場合に要する CPU 時間の合計を測定してみました．

`COLUMN`	列ごとに行方向を連続してアクセス
`ROW`	行ごとに列方向を連続してアクセス
`R-COLUMN`	列ごとに行方向をランダムに異なりなくアクセス
`R-ROW`	行ごとに列方向をランダムに異なりなくアクセス

図 3.7 がベクトル，図 3.8 がスカラーで翻訳した実行した値をプロットしたものです．図では “`COLUMN`” を 1 とし，他のアクセスがその何倍になるかを示しています．

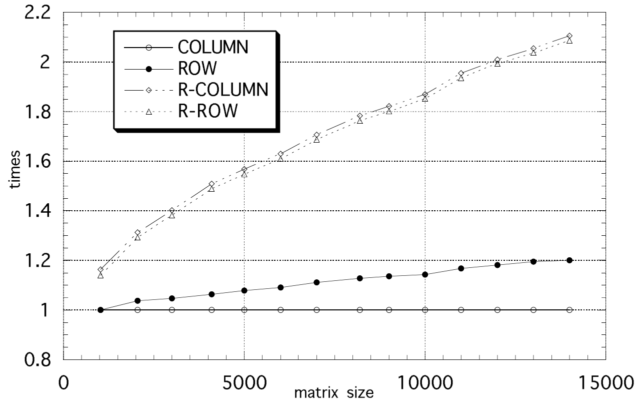

図 3.7 : メモリーアクセス (ベクトル)

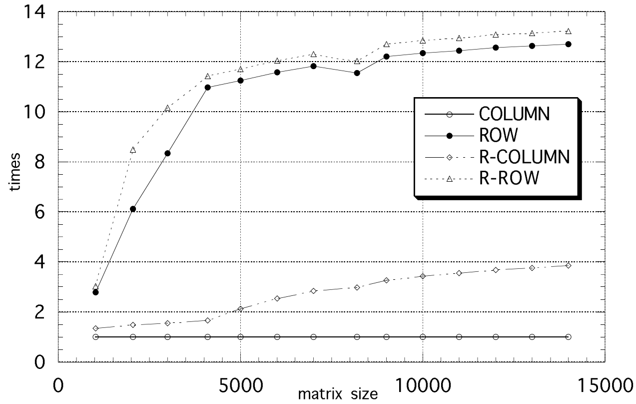

図 3.8 : メモリーアクセス (スカラー)

このデータは，講習会で “Fortran の配列のアクセスは行方向に連続してアクセスした方が速いよ” と説明する目的で測定したものです．

3.5　単一 PE の演算性能のまとめ

これまでのテストから，VPP700/56 のスカラー演算性能は “大したことない” ことがわかりました．結果として，利用者が単一 PE でプログラムを実行する場合，ベクトル化率が高くない限り十分なベクトル演算性能を享受できないことになります．また，ベクトル化率を高くする努力は，VP2600/10 や M-1800/20U に比較して一層必要になったということを感じました．

自動ベクトル化技術はある程度完成されてしまったと言われています．しかし，逆に考えれば，それから先の性能アップの努力は利用者に任されているということです．九州大学大型計算機センターの利用者の VU 率は他センターと比べて決して高いとはいえません．大規模な記憶領域を確保し，行列を「ぶんまわす」数値シミュレーションの分野において，スカラー並列計算機が自動並列化手法による明るい未来を保証してくれていない以上，これからもベクトル化プログラミングの重要性を広報していく必要があると考えています．

4　複数 PE の演算性能

VP2600/10 が 1CPU のベクトル計算機であったこともあり，VPP700/56 の導入当時，センターには並列計算用の利用者プログラムの蓄積がありませんでした．そこで，複数 PE の演算性能のデータとして，富士通提供の並列数値計算ライブラリ SSL II/VPP([15]) の性能測定結果で代替させていただきます．

少しだけいいわけをすれば，数値計算プログラムの多くが連立 1 次方程式，固有値問題, FFT などの汎用的な線形計算に帰着される (もしくは帰着できるようなアルゴリズムを構成できる) こと，また，その場合，全体の計算コストがこれらの部分に大きく依存することから，これらの線形計算の性能を調べることは無駄ではないだろうと思います．

利用者の中には，すべてプログラムを自前で作成することに意気が揚がり，メーカーの提供するサブルーチンは「中身が見えない」理由で使いたがらない人もいます．しかし，Fortran 90/VPP が属する “VPP Fortran” の言語仕様は富士通の完全な独自仕様であり，拡張最適化制御行だらけのプログラムを (一応 “!” でコメントアウトされるので動作しないことはないのですが) 他の計算機に移植する気はあまり起こらないと思います．従って，SSL II/VPP を利用するにあたっての “敷居” は低いのではないかと考えます．

4.1　SSL II/VPP

SSL II/VPP(Scientific Subroutine Library II for Vector Parallel Processors) は，オーストラリア国立大学の数値計算グループと富士通が共同開発した並列数値計算ライブラリです．以下の機能をサポートしています．

- 行列積
- 実・複素行列の連立 1 次方程式
- 正定値対称行列の連立 1 次方程式
- 実バンド・スパース行列の連立 1 次方程式
- 固有値問題 (3 重対角, 実対称, 一般化固有値問題)
- 逆行列
- 実・複素 Fourier 変換 (2, 3, 5 の混合基底)
- 線形最小二乗解
- 一様乱数生成

また，次回のバージョンアップでは，幾つかの機能追加を予定しています ([16])．

SSL II/VPP は，Fortran 90/VPP から CALL 文により利用します．従って，Fortran 90/VPP に関する基本的な知識が必要です．精度はすべて倍精度です．単一 PE 用の SSL II/VP とは機能範囲や呼び出し方法が大きく異なります．

VPP700/56 が導入された当時，富士通の発行している Fortran 90/VPP のマニュアルは，無味乾燥な文法だけが列挙された [11] のみでした．そのため，テスト用のプログラムの作成には，VPP Fortran の解説である京都大学大型計算機センターの広報記事 ([6], [7])，および名古屋大学大型計算機センターのニュース記事 ([8], [9]) が大変参考になりました．

富士通からは，プログラミングを解説した [12] がようやく発行される予定です．

4.2　56PE でのピーク性能試験

1996 年 12 月 5 日，富士通沼津工場で VPP700/56 の現地テストを行ないました．その際，ピーク性能試験として，行列積演算の Fortran プログラムを 56PE で並列実行しました．プログラムは，次数 n の倍精度実数データ型の正方行列 $A,\ B$ の積を計算し，結果を次数 n の正方行列 C に格納するものです．

行列積演算には SSL II/VPP の `DP_VMGGM` を用います (ソースリストを見たことがあります．A4 判 29 頁におよぶ解読不能なものでした)．行列のデータは各要素を $A_{ij},\ B_{ij}$ とするとき，

$$A_{ij} = B_{ij} = \left(\frac{2}{n+1}\right)^{\frac{1}{2}} \sin\left(\frac{ij\pi}{n+1}\right)$$

で定義します．なお π は $4\tan^{-1}1$ とします．

このように定義した理由は，行列積演算の精度の確認として理論的に $AB = E$(E は単位行列) となることを用いて，行列 C に対する要素を C_{ij}，E の要素を E_{ij} とするとき，最大行和ノルムの意味での誤差 ($||E||_\infty = 1$ より絶対誤差かつ相対誤差となります．)

$$||C-E||_\infty = \max_{1\le i\le n}\sum_{j=1}^{n}|C_{ij}-E_{ij}|$$

が知りたかったからです．ウォールクロックにより行列積演算の開始から結果格納までの経過時間を測定しました．次元 n の大きさと $A,\ B,\ C$ に必要な記憶領域および経過時間，誤差は表 4.1 の通りです．経過時間は小数点第 1 位以下を，誤差は小数点第 4 位以下を切捨てています．

表 4.1：56PE でのピーク性能試験結果

n	記憶領域	経過時間	誤差
18000	7.2GB	105sec.	0.138e-09
40000	35.7GB	1135sec.	0.393e-09
55000	67GB	2865sec.	0.713e-09
60000	80GB	3683sec.	0.681e-09
65000	94GB	5197sec.	0.819e-09

個人的な感想としては，演算数の増大によって着々と増え続ける誤差の方が気になります．もちろん，誤差ノルムの計算自体に混入する丸め誤差も考慮に入れる必要がありますが，大規模な数値計算を行なう場合の丸め誤差の評価は今後きちんと考える必要があるのではないでしょうか．

図 4.1 は演算数を $2n^3$ とした GFLOPS 値をプロットしたものです．“`Theoretical_Peak`” は，56 台での理論ピーク性能値です．

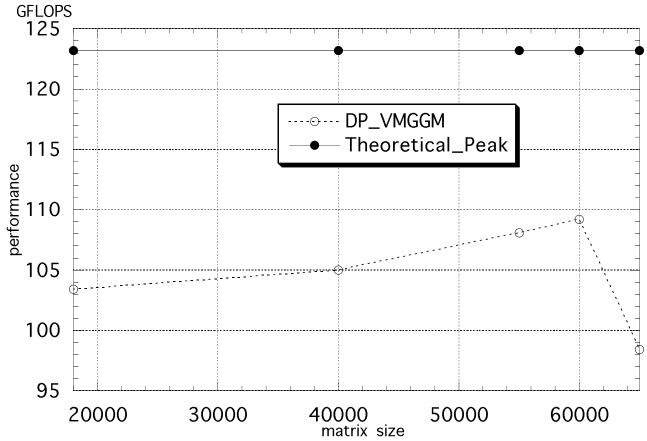

図 4.1 : 56PE でのピーク性能試験結果

$N = 65000$ で処理性能が低下した原因を開発元に問い合わせたところ，“たまたま” この値に対して性能低下を起こす要因があり，DP_VMGGM のソースの修正により，$N = 60000$ 以上の性能が出るとの報告を受けました．しかし，56PE の性能試験を行なうためにはすべての PE の割り当てを変更する必要があるため，運用を開始した現在ではなかなかテストできません．

4.3　台数効果

現在，利用者が使用できる最大 PE 台数は 32 です．以下，4 つの SSL II/VPP サブルーチンに対し，PE 台数を変化させ，台数効果を調べてみました．測定は 1997 年 4 月 ~ 6 月に行ないました．

FLOPS 値は，Analyzer([17]) の浮動小数点演算数の採取機能 PEPA(PE Performance Analyzer) の値を信用して用いています．

○行列積

図 4.2 は 56PE でのピーク性能試験で使用した同じプログラム (サブルーチンDP_VMGGM) の $n = 8500$ に対し PE 台数を変化させたものです．“Theoretical_Peak” は，その台数での理論ピーク値です．

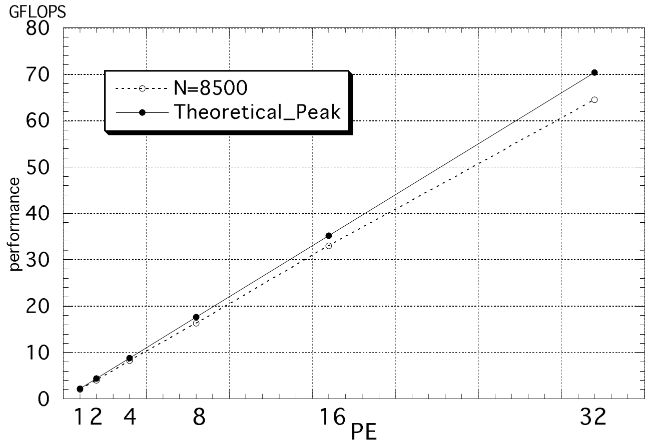

図 4.2 : DP_VMGGM の台数効果

$N = 8500$ にした理由は，ぎりぎり 1PE で実行可能だからです．各 PE 台数分のメモリーの上限に迫るように N を変化させたデータも採取しましたが，それほど性能は変わりませんでした．

図 4.3 は，実スパース行列と実ベクトルの積を求める DP_VMVSD の性能を示すものです．実スパース行列は，領域 $\Omega = [0,1]^3$ を各方向 $N+1$ 等分割したときの楕円型偏微分演算子を中心差分を用いて離散化した時に得られるものです．[13] による対角形式格納法により格納されています．

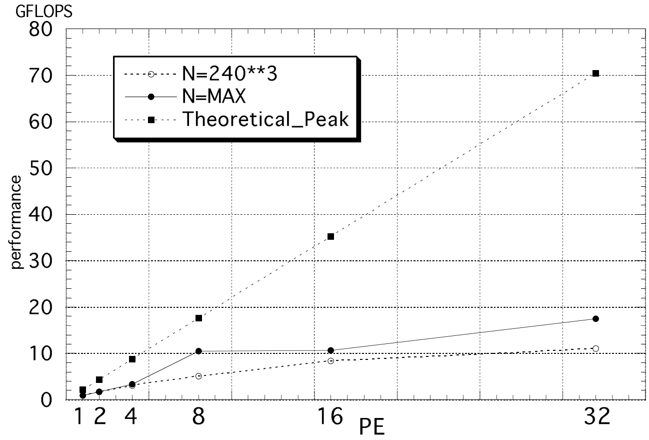

図 4.3 : DP_VMVSD の台数効果

“N=MAX” は，各次元の分割数 N を実行可能な記憶領域の上限近くに設定して採取した値をプロットしたものです．

○連立 1 次方程式

図 4.4 は密実行列に対する連立 1 次方程式 DP_VLAX の性能を示すものです．行列は DP_DVMGGM と同じものを用いました．

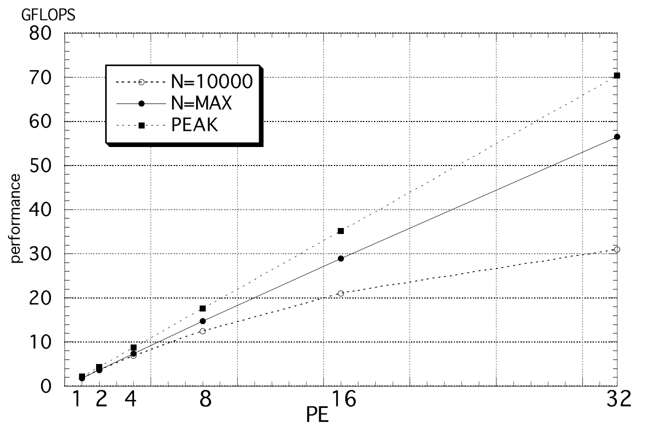

図 4.4 : DP_VLAX の台数効果

"N=MAX" の意味は先ほどと同じです．

図 4.5 は非対称または不定値のスパース行列に対する連立 1 次方程式 (MGCR 法) DP_VCRD の性能を示すものです．行列は DP_VMVSD と同じものを用いました．

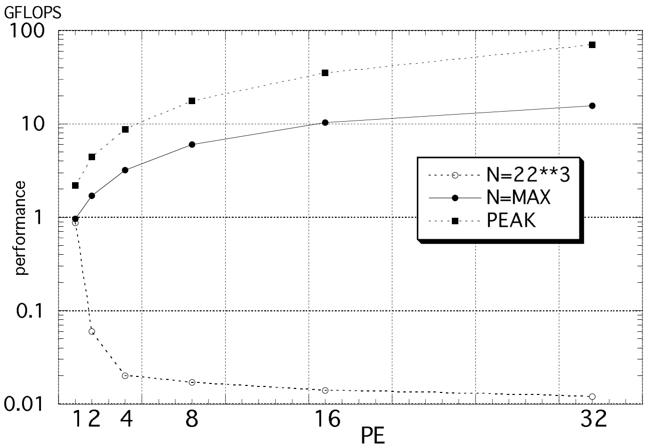

図 4.5 : DP_VCRD の台数効果

"N=MAX" の意味は先ほどと同じです．このテストでは，N(ここでは分割数です) の値を小さめに設定してみました．

$N = 22$(従って方程式の次元は 22^3) の場合に方程式を解く時間は数秒です．しかし，PE が増えるに従い，浮動小数点演算に要する時間よりも PE 間のデータ転送にかかるコストが増大するため，性能は悪くなっていきます．このような小規模な問題では PE 台数を増やしてもあまり意味がないことがわかります．

5 おわりに

本稿で報告した VPP700/56 のテストは，計算機の性能を総合的に評価したものとはとてもいえません．例えば，C, C++ での演算性能やメッセージパッシングライブラリでの並列化がどれくらいの性能を示すか，またファイル入出力や PE 間通信など，ここに提示できていない試験項目は山積しています．これらについては今後の課題とさせていただきます．

最後に，VPP700/56 を使ってみた感想は，《きちんとベクトル並列計算機の特性に注意してプログラムを組めばそれなりの結果は得られるかなぁ》という極めてあたりまえのものでした．ただし，最適化制御行を埋め込むプログラミングはやはり厄介なものですし，「メモリーが足りないから仕方なくやってるんだ」という虚しさを感じることも否定できません．自動並列化コンパイラがあるにこしたことはないというのも正直な感想です．

参考文献

[1] 橋倉 聡, 川寄 正子, 河津 秀利, 福田 賢一, 天野 浩文 : 並列スーパーコンピュータの効率的運用をめざして, 第 19 回全国共同利用大型計算機センター研究開発連合発表講演会 (1997).

[2] 渡部 善隆 : VPP700/56 での対話型処理, 九州大学大型計算機センター広報, Vol.30, No.2, pp.167–185 (1997).

[3] M. Metcalf, J. Reid (西村 恕彦, 和田 英穂, 西村 和夫, 高田 正之 訳) : 詳解 Fortran 90, bit 別冊, 共立出版 (1993).

[4] 島崎 眞昭 : スーパーコンピュータとプログラミング, 計算機科学／ソフトウェア技術講座 9, 共立出版 (1989).

[5] 浅岡 香枝, 平野 彰雄 : はじめての Fortran90, 京都大学大型計算機センター広報, Vol.30, No.4, pp.213–224 (1997).

[6] 安岡 孝一, 金澤 正憲 : VPP FORTRAN 入門, 京都大学大型計算機センター広報, Vol.27, No.6, pp.262–266 (1994).

[7] 平野 彰雄 : 並列プログラム入門 : 京都大学大型計算機センター広報, Vol.28, No.3, pp.116–136 (1995).

[8] 永井 亨 : VPP Fortran 入門, 名古屋大学大型計算機センターニュース, Vol.26, No.4, pp.315–339 (1995).

[9] 小河 正基 : ベクトル並列計算機 VPP500 体験談, 名古屋大学大型計算機センターニュース, Vol.27, No.2, pp.120–151 (1996).

[10] UXP/V Fortran 90/VP 使用手引書 V10 用, J2U5-0050, 富士通株式会社 (1995).

[11] UXP/V Fortran 90/VPP 使用手引書 V10 用, J2U5-0080, 富士通株式会社 (1995).

[12] UXP/V VPP プログラミングハンドブック V10, 富士通株式会社 (発行予定).

[13] FIJITSU SSL II 拡張機能使用手引書 (科学用サブルーチンライブラリ), 99SP–4070, 富士通株式会社 (1991)．

[14] FUJITSU SSL II 拡張機能使用手引書 II(科学用サブルーチンライブラリ), J2X0-1360, 富士通株式会社 (1995)．

[15] FUJITSU SSL II/VPP 使用手引書 (科学用サブルーチンライブラリ)V11 用, J2X0-1370, 富士通株式会社 (1996)．

[16] FUJITSU SSL II/VPP 使用手引書 (科学用サブルーチンライブラリ)V12 用, J2X0-1371, 富士通株式会社 (1997)．

[17] UXP/V アナライザ使用手引書 V11 用, J2U5–0130, 富士通株式会社 (1995).