サイエンティフィック・システム研究会　1999 年 8 月 20 日
科学技術計算分科会 第 1 回会合資料

# 京都大学における VPP800 の運用と性能評価

平野 彰雄 *　植木徹 *
hirano@kudpc.kyoto-u.ac.jp　ueki@kudpc.kyoto-u.ac.jp
赤坂浩一 *　浅岡香枝 *　金澤正憲 *
akasaka@kudpc.kyoto-u.ac.jp　asaoka@kudpc.kyoto-u.ac.jp　bwv147@mbox.kudpc.kyoto-u.ac.jp
* 京都大学大型計算機センター

## 1　はじめに

京都大学大型計算機センターは、全国の大学、高等専門学校などの研究者が学術研究に伴う科学技術計算や情報処理を行うための全国共同利用施設であり、その登録利用者数は 3,500 名を超える。年々増加する大規模科学技術計算需要に応えるために、本年 2 月末にスーパーコンピュータ (以下、スパコンと略す) のリプレースを行い、 Fujitsu VP2600/10E および Fujitsu VPP500/15 の 2 台のスパコン (1995 年 1 月導入) を Fujitsu VPP800/63 に置き換えて、サービスを行っている。

本稿では、まず、導入した VPP800 のハード構成および特徴について述べ、次に、本センターでのサービスの概要および今後の課題について明らかにする。また、 VPP800 の基本性能に関する評価を実測データに基づいて報告する。

## 2　VPP800 システムの構成と特徴

今回導入した VPP800 システムは図 1に示すように、 VPP800/63 本体とストレージサーバ Gen5 から構成される。

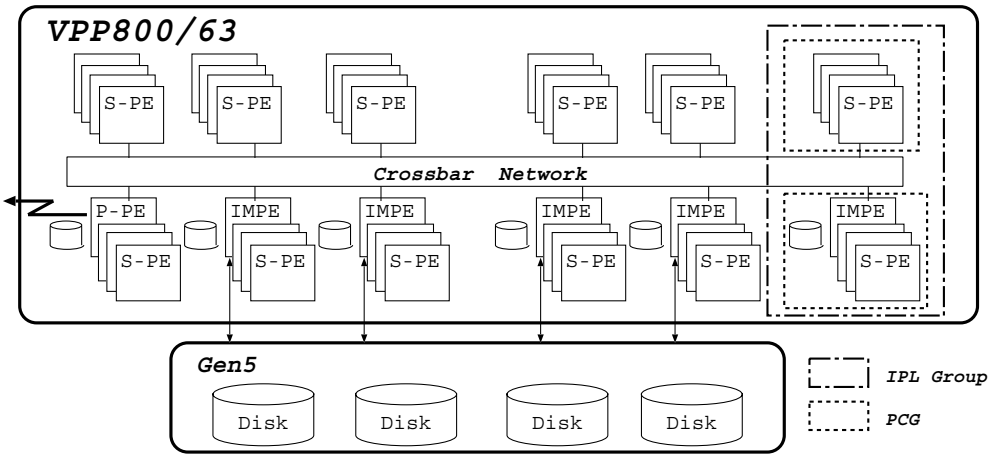


図 1. VPP800 システムの構成図

### 2.1 VPP800/63 システム

VPP800/63 システムは、これまでの VPP500/15 と同じタイプの分散メモリ型のベクトル並列計算機である。システムは、63 台のプロセッサ (PE: Processing Element) とプロセッサ間での通信のためのクロスバーネットワークで構成される。

#### 2.1.1 PE の構成と緒元

各 PE は 64 ビットアーキテクチャのベクトル計算機であり、ベクトルユニット、スカラユニット、メモリおよびデータ転送ユニットで構成される。表 1に、1PE 当りの主な緒元を示す。

表 1. PE の主な緒元

<table>
<tr><td colspan="2">ベクトルユニット (VU)</td><td colspan="2">スカラユニット (SU)</td></tr>
<tr><td>理論ピーク性能</td><td>8GFLOPS</td><td>理論ピーク性能</td><td>1GOPS(※)</td></tr>
<tr><td>レジスタ容量</td><td>ベクトル　128KB<br>マスク　2KB</td><td>キャッシュメモリ</td><td>一次　128KB<br>二次　2MB</td></tr>
<tr><td rowspan="6">演算パイプ</td><td rowspan="6">(乗算 / 加算 /<br>乗算 & 加算 /<br>論理演算) 1<br>(除算 /SQRT) 1<br>(マスク) 1<br>(ロード / ストア) 1</td><td>レジスタ個数</td><td>汎用　32<br>浮動少数点　64</td></tr>
<tr><td>論理アドレス空間</td><td>256GB</td></tr>
<tr><td colspan="2">メモリ (MSU)</td></tr>
<tr><td>容量</td><td>8GB</td></tr>
<tr><td>転送速度</td><td>64GB/ 秒</td></tr>
<tr><td>PE 間スループット</td><td>1.6GB/ 秒× 2</td><td>インターリーブ</td><td>512 Way</td></tr>
</table>

※ GOPS(Giga Operation Per Second)、4 オペレーションが同時実行可能。

#### 2.1.2 PE の種類と機能

63 台の PE は、その構成と機能により IO-PE と S-PE(Secondary PE) に分類される。

IO-PE とは入出力機構を備えた PE であり、S-PE は入出力機構を持たない演算専用の PE である。

VPP800/63 は 55 台の S-PE と 8 台の IO-PE で構成されており、IO-PE は 36.4GB(9.1GB × 4) のディスクとネットワークインターフェースを備えている。

- **IPL グループ**

IPL グループとはシステムが分割して起動できる単位である。VPP800/63 では 1 台の IO-PE と 7 台の S-PE の計 8 台の PE で一つの IPL グループを構成している。IPL グループ内の IO-PE は IMPE と呼ばれ、物理的にディスクを持たない S-PE に対してディスクを提供する機能を持っている。また、8 台ある IMPE の内の 1 台が P-PE(Primary PE) と呼ばれ、システム全体を統括する。

VPP800/63 では、IPL グループ単位に平行して IPL を行うために、システムの起動時間は、以前のスパコン VPP500/15 が 30 分以上かかっていたのに比べて VP800 では 15 分以下と半分以下になっている。

- **PCG(Power Control Group)**

PCG とは PE への電源供給単位であり、PCG 単位での縮退、拡張運転が可能である。VPP800/63 では、4 つの PE で PCG を構成しているので、IO-PE を含まない 8 つ PCG 全てを停止し、32PE 構成のシステムまで縮退して運転でき、電力消費を押さえることができる。

### 2.2　ストレージサーバ Gen5

導入したストレージサーバ Gen5 は、 XLE モデルの 3 バンク構成である。バンクは 2 つのユニットグループから構成され、それぞれのユニットグループには 12 個のディスクが格納されている。 1 つのディスクの容量は、 18.2GB である。一方、 VPP800 システムとの接続用の HIPPI アダプタ (800Mbps) は、 4 つ実装しており 4 台の IO-PE にそれぞれを接続できる。したがって、 3 バンクで構成される Gen5 のディスクを各 IO-PE から独立して同時にアクセスするために、 4 つのファシリティに分割している。

図 2に Gen5 の構成を示し、表 2に各ファシリティのディスク個数と RAID5 構成時の容量を示す。

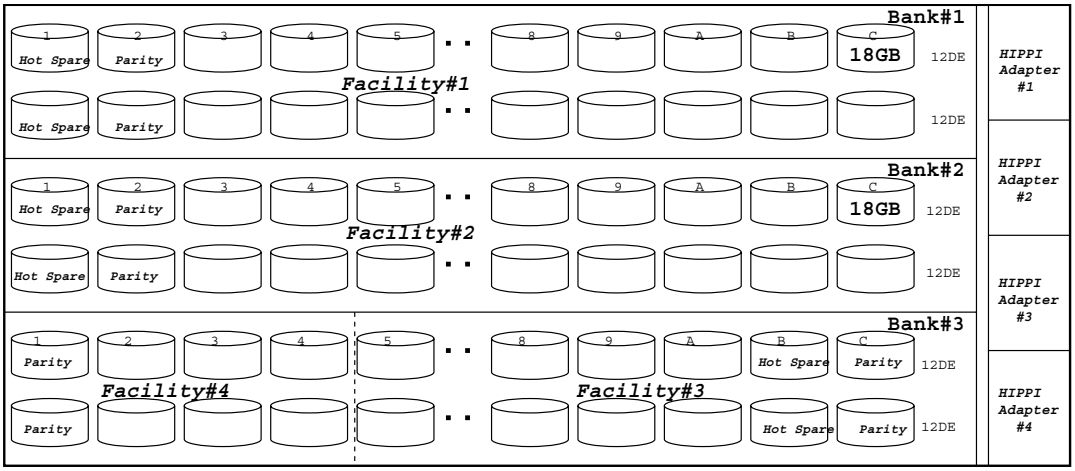


図 2. Gen5 の構成図

表 2. Gen5 XLE の構成

| バンク | ファシリティ | ディスク個数 | 容量 (RAID5) |
|---|---|---|---|
| #1 | #1 | 20+(パリティ 2、ホットスペア 2) | 360GB |
| #2 | #2 | 20+(パリティ 2、ホットスペア 2) | 360GB |
| #3 | #3 | 12+(パリティ 2、ホットスペア 2) | 216GB |
| | #4 | 6+(パリティ 2) | 108GB |

## 3　導入したソフトウエア

今回、 VPP800 システムで導入した、ベクトル処理および並列処理のためのコンパイラ、メッセージパッシングライブラリ (MPL)、数値計算ライブラリ、ツールの関係を表 3に示す。これらのソフトウエアは、基本的に 64 ビットおよび 32 ビットのアドレスモードをサポートしている。

また、次のようなアプリケーションも導入し、サービスしている。

- **Gaussian**
- **MSC/NASTRAN**
- **LS-DYNA3D**
- **MASPHYC**
- **MOPAC**
- **FEM5**
- **FSPICE**
- **AVS**

# 4　VPP800 の運用とシステム設計

## 4.1　VPP800 のサービス

### 4.1.1　会話型サービス

P-PE に直接ログインする形で会話型のサービスを行っている。

なお、 P-PE 以外の PE には /etc/nologin を設定し、一般ユーザがログインすることを禁止している。

また、会話型の許可量は、 CPU を 1 時間、メモリサイズを標準 1GB、最大 2GB としている。

### 4.1.2　NQS バッチ処理

NQS のキュー名と許可量を表 4に示す。キュー cl はコンパイル専用であり、 z は 40PE を超えるジョブのためのキューである。

### 4.1.3　MVPP 連携

VPP500 導入に開発した MVPP 連携機能によるサービスは、 NQS キューの内ベクトルジョブ d,e と並列ジョブ f,g,h についてサービスを行っている。また、コンパイル、リンクのステップは、専用のコンパイルキューを定義している。

## 4.2　NQS の定義とサービス

NQS のサービスに関するいくつかの設定パラメータがあるが、現在は、次のようにしている。

### 4.2.1　ジョブリストと実行多重度

まず、 NQS のキューと割付け PE を管理するジョブリストの設定は、次のようしている。 **1)** コンパイル専用キュー (cl) は、 IO-PE に割付ける。 **2)** 並列ジョブキュー (f,g,h) は、 S-PE だけに割付ける。 **3)** ベクトルジョブキュー (d,e) は、全ての PE に割付ける。

また、空き PE を極力少なくし、且つ、 PE 割付け待ちの発生を押さえるために、 NQS ジョブの実行多重度の制御は、キューの Run Limit と Complex Queue の Run Limit を次のように定義している。

### 4.2.2　NQS-JM(Job Manager) の利用と定義

NQS-JM は、ユーザ当りの受付ジョブ数と実行多重度およびジョブの資源量の設定値を管理する機能である。

現在は、実行多重度制御により、特定のユーザがシステムを占有しないようにしている。

また、ジョブの資源量の設定値管理を用い、ジョブ MRFS(Memory Resident File System) のサイズを標準 0GB、最大値 7GB に設定 (NQS の MRFS のサイズは、 7GB) し、利用者が任意のジョブ MRFS のサイズを指定できるようした。

## 4.3　ファイルシステムの設計とサービス

今回のスパコンリプレースの重要な課題の一つに、ユーザに高速、大容量ファイルを提供することがあった。

したがって、導入した Gen5 にスパコン専用のホームディレクトリおよび大容量ファイルの割当てが必須であったが、システム設計段階で、当面、使えるファイルシステムとしては UFS と VFL であり、 UFS では 2GB を超えるファイルシステムが作れない、また、 VFL では i-node が 3,2768 個しか作れないという大きな制約があった。

しかし、大容量、高速なアクセスが運用上の必須要件であるので VFL ファイルシステムを選択し、ホームディレクトリを 3 つに分けて配置、極力ファイル個数を減らすように設計を行った。表 6に Gen5 上のファイルシステムの配置と容量を示す。

ジョブ凍結、ジョブスワップのためのファイルスペースは、並列アクセスが有効なために 4 つのファシリティに分散して配置し、合計で 164GB を確保した。

また、ユーザホームおよび大容量ファイル領域として合計 800GB を確保した。これらの領域は、ユーザ当りの容量、個数を quota で管理している。現在のユーザ当りの許可量およびブロックサイズを表 7に示す。

さらに、ユーザホームで使用する i-node 数を極力減らすために、次のような処置を行っている。

**1)** 個々のユーザのホームディレクトリは、ユーザが VPP800 へ最初にログイン時に作成する。

**2)** 標準的な環境設定はシステム内で行うことで、ユーザがホームディレクトリに設定ファイル (.cshrc など) 作らなくても使えるようにした。

### 4.4 移植および開発したソフトウエア

#### 4.4.1 開発ソフトウエア

VPP800 の運用のために作成したコマンドには、次のようなものがある。

**1) qsub**

独自の課金処理のために独自のリクエスト名管理と、ジョブプロフィールの出力オプション (-oi) を標準で設定している。

**2) df**

Gen5 上のファイルシステムは、 HIPPI 接続された IO-PE にしか情報が無いために、 P-PE の会話型サービスでユーザがホームに関する情報が得られない。したがって、 P-PE で df コマンドが入力されると必要な IO-PE に rsh コマンド経由で df を実行し結果を整形し出力している。

**3) quota**

これも df コマンドと同じようなもので、 Gen5 上のファイルシステムの quota 管理情報を表示するために、必要な IO-PE に rsh 経由で quota を投げ、結果を出力している。

#### 4.4.2 移植ソフトウエア

VPP800 に移植したソフトウエアには、次のようなものがある。

- **emacs エディタ**
- **tcsh、 less**
- **GNU ツール gzip,patch,make,tar**
- **kterm, ウインドマネージャ fvwm**
- **Secure Shell**
- **parl5**
- **pgplot グラフックライブラリ**

# 5　今後の課題

## 5.1　ファイルシステムの移行

VFL ファイルシステムの i-node 数の制約はやはりきつく、せっかく導入した大容量ディスクを充分に活用できていない。したがって、現在、 VFL に代るファイルシステムとして FPFS(Flexible and high Performance File System) が準備されているので、早急に、移行し、ファイル個数というユーザへの制限を解除したいと考えている。

## 5.2　縮退運転による消費電力の削減

VPP800/63 と VPP500/15 の消費電力量は、カタログを見ると VPP800/63 が 240.18KVA、 VPP500/15 が 214.88KVA である。 PE 当りの消費電力量としては、 VPP500 に比べると 3 分の 1 以下であるが、PE 台数がその分増えているので結果として同じような値となっている。

前に述べたように VPP800 では PCG 単位での縮退、拡張実現されている。したがって、ジョブが少ない場合は、ジョブリストによる PE へのジョブの割付け制御と連動させ PCG の縮退を行い、消費電力面での効率的なシステム運転の自動化を検討している。

実現に向けて行ったテストでは、 PCG 単位の縮退 (4 つの S-PE の shutdown 時間を含む)、拡張 (4 つの S-PE が IPL され、ジョブが流せる状態になるまで) に要した時間は、縮退で 1 分、拡張で 1 分 40 秒ほどであった。

## 5.3　システム資源利用効率とターンアラウンドの改善

現在、 VPP800 は NQS ジョブで使えるメモリサイズを 7GB であり、各 PE には一つのジョブしか割付けない simplex モードで運用している。

しかし、ジョブの使用メモリ量が 2GB を超えるジョブはベクトル、並列ジョブともに、まだ少ない。

したがって、 NQS-JM の資源割当て管理機能を用いてメモリ許可量を標準 2GB、最大 7GB に変更して、並列ジョブも含めて share モードで各 PE に 2 つのジョブを割付けることを考えている。

また、現在の NQS ジョブの CPU 許可量は、 1 時間と 6 時間の二種類だけにしている。しかし、このような設計では、ジョブが混むと CPU 時間の短いジョブもプロダクションランのジョブと同じように FIFO(First In First Out) で制御されてしまうので、試行錯誤的なプログラム開発およびデバッグが不利になる。

このような問題に対応し、戦略的なスケジューリングを実現する機能として NQS-JS(Job Schedule) がある。

したがって、センターでの NQS ジョブのキュー定義の見直しと合わせて NQS-JS を適用して、ジョブが要求するシステム資源量に応じたスケジューリング、ターンアラウンドタイムの保証をぜひ実現したいと考えている。

# 6　性能の評価

## 6.1　ベクトルの基本性能

VPP800 の PE のベクトル性能の評価を目的に、幾つかの基本ベクトルループでの CPU 時間を計測し、これまでのスパコン VPP500 との比較をおこなったので、これについて報告する。

なお、プログラムは Fortran で記述し、 CPU 時間の計測にはサービスサブルーチン CLOCKV を用い、図の縦軸は消費した CPU 時間を表し、横軸の N はベクトル長を示している。

### 6.2 メモリアクセスの性能

メモリアクセス性能を評価するために、 `x(i)=0.0` および `x1(i)=x2(i)` のループの計測した。結果を図3と図 4に結果を示す。図 3では、ベクトルストアでは VPP500 と比べると N=500 で約 2 倍、 N=2048 で約 4 倍の性能向上が見られる。また、図 4も最大ベクトル長の 2048 単位で、若干の性能の劣化が見られるが、 VPP500 との性能差はベクトルストアーと同じような傾向であった。

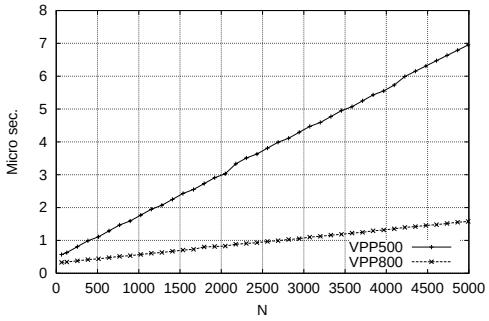


図 3. x(i)=0.0

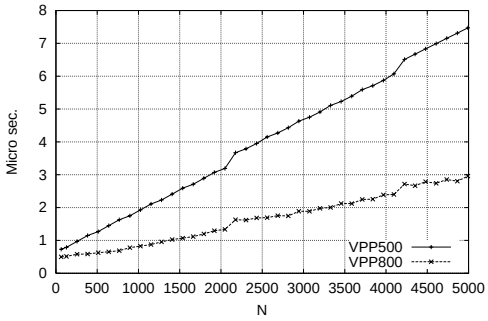


図 4. x1(i)=x2(i)

### 6.3 ベクトル演算の性能

次に、ベクトル演算の基本性能に関する計測を行った。乗算と除算について計測した結果を図 5と図 6に示す。乗算は N=2048 において VPP500 と比べて約 3.2 倍、除算で約 7.5 倍の性能差が見られる。

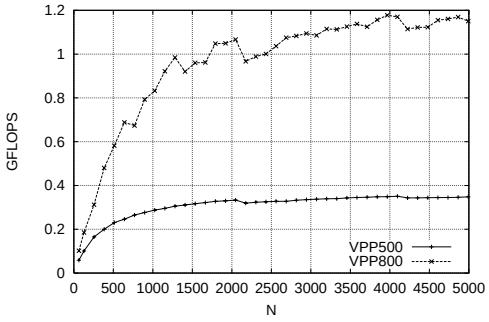


図 5. x1(i)=x2(i) × x3(i)

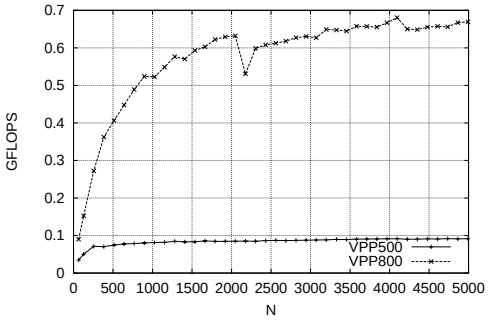


図 6. x1(i)=x2(i)/x3(i)

また、図 7は、 VPP800 で新たにサーポートされた sqrt についての比較である。その性能差は N=2048 において約 34 倍という結果であった。

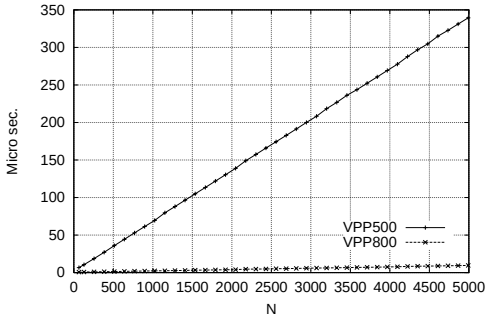


図 7. x1(i)=sqrt(x2(i))

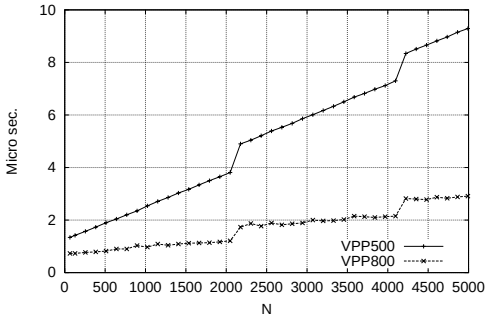


図 8. xmax=max(xmax,x(i))

### 6.4 マクロ演算の性能

マクロ演算の性能比較のために、最大値検索、総和演算、内積の 3 つについて、計測した。

結果を図 8、図 9、図 10にぞれぞれ示す。

VPP500 との性能差は、 N=2048 において最大値検索で約 3.2 倍、総和計算で約 2.7 倍、内積で 3.6 倍であった。

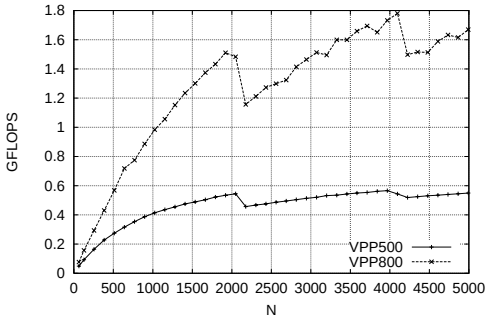


図 9. s=s+x(i)

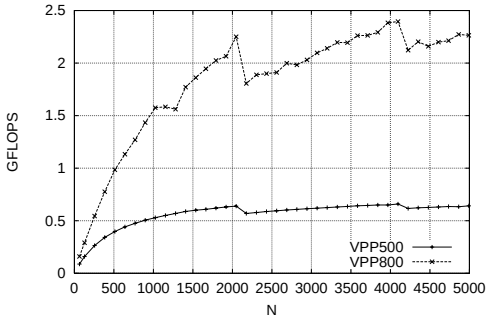


図 10. s=s+x1(i)×x2(i)

### 6.5 並列性能

本来、並列処理についても基本性能を評価すべきであるが、データがそろっていないので、本センターの VPP800 で行った連立一次方程式の並列版のベンチマーク結果を表 8に示す。

LINPACK Parallel とは、 VPP800/63 の全ての PE を使った LINPACK の結果である。実効性能比 95.7 %は、非常にすばらしい値である。

残りベンチマークは、センターサービスで一般的に使える最大 PE 数 40 での実測結果である。

各ベンチマークの名称は、次のようになっている。

1) **VPP-ScaLAPCK**

現在、センター行っている ScaLAPACK の VPP Fortran への移植バージョンである。

2) **MPI-ScaLAPCK**

これは、 VPP800 で導入した製品版の MPI ベースの ScaLAPACK である。

3) **SSLII/VPP**

これは、 VPP Fortran 用の SSLII/VPP の連立一次方程式ルーチン`dp_vlax` を指す。

これらのベンチマークは、 VPP-ScaLAPCK で解ける問題サイズ 18 万元で他のルーチンとの性能比較のための実測したものである。また、ルーチンの構造から SSLII/VPP の`dp_vlax` では問題サイズが 13 万元し指定できなかった。したがって、この問題サイズ 13 万元について VPP-ScaLAPACK、 MPI-ScaLAPACK も計測し直したので、この値も表に載せてある。

なお、 MPI-ScaLAPACK では、プロセッサを 2 次元に割付けるが、今回のベンチマークでは、 VPP-ScaLAPCK に合わせ 1 × 40 として実測した値であり、他の形状でプロセッサを割付ければ、より良い性能が得られるかもしれないが、今回、これを検証していない。

## 7 まとめ

VPP800/63 は理論最大ベクトル性能で 504GFlops、総メモリ量も 504GB というスパコンである。これは以前のスパコン VPP500/15 と比べると性能で 21 倍、総メモリ量では 131.25 倍という値である。今

回、このようなスパコンリプレースができたことは、非常に喜ばしいことであるが、システムを管理する側の立場のものとして戸惑いもある。

VPP800 の利用状況は、ホームディレクトリを Gen5 に移してから 2ヶ月であるが登録ユーザ数は 120 名ぐらいであり、一挙に PE 数が 4 倍になり演算速度も速いので、処理能力は充分といった状況である。

利用者からのプログラム相談も VPP500 の時代に比べるとベクトル C、 MPI など多彩になり、色々な使われ方をしていることがうかがえる。

また、本文中の今後の課題でも述べたが、 VPP800 システムは、バックエンドプロセッサであった VPP500 と比べるとセンターサービスにとって有用な様々な機能が提供されているが、本センターでは、当面、 VPP500 サービスの継承と利用者資産の移行を最優先の課題として運用を検討したために、これらの新規機能を活かしたシステム設計ができていない。

これらの新機能を取入れたシステム設計とサービスについては、今年度中を目標に段階的に進めることを予定している。

## 謝辞

本センター研究開発部、安岡孝一助教授ならびに本学情報学研究科、岡部寿男助教授にはシステム設計段階で、色々と適切なアドバイスを頂いた。また、小澤義明技術専門職員には、システム設計およびプログラム開発で援助を頂いた。ここに記して感謝致します。

## 参考文献

[1] 富士通マニュアル: 「FUJITSU VPP800 シリーズハードウエア使用手引書」

[2] 富士通マニュアル: 「UXP/V ネットワークキューイングシステム -JM 説明書 V20 用」

[3] 富士通マニュアル: 「UXP/V ネットワークキューイングシステム -JS 説明書 V20 用」

[4] 浅岡香枝, 平野彰雄: 「VPP800 のベクトル性能について」, 京都大学大型計算機センター広報,Vol.32No.3(1999)

[5] 浅岡香枝, 平野彰雄, 岡部寿男, 金澤正憲: 「線形計算ライブラリ ScaLAPACK の VPP への移植」, 京都大学大型計算機センター研究開発部研究発表報告集第 14 号 (1999)

[6] 島崎眞昭: 「スーパーコンピュータとプログラミング」, 共立出版 (1989)

表 3. 導入ソフトウエアの一覧

<table>
<tr><th colspan="3">ベクトル</th><th colspan="3">並列</th></tr>
<tr><td>コンパイラ</td><td>ライブラリ</td><td>ツール</td><td>コンパイラ</td><td>ライブラリ</td><td>ツール</td></tr>
<tr><td rowspan="3">Fortran/VP</td><td>SSL II/VP</td><td rowspan="5">アナライザ<br>sampler<br>fdb<br>pa</td><td>Fortran/VPP</td><td>SSL II/VPP</td><td rowspan="2">アナライザ</td></tr>
<tr><td rowspan="2">BLAS/VP<br>LAPACK/VP</td><td>HPF2.0</td><td></td></tr>
<tr><td colspan="3">MPL</td></tr>
<tr><td>C/VP</td><td rowspan="2">C-SSL II/VP</td><td>MPI2.0</td><td>ScaLAPACK</td><td>TotalView</td></tr>
<tr><td>C++/VP</td><td>PVM3.3</td><td></td><td>MPTool</td></tr>
</table>

表 4. キュー名と許可量

| キュー名 | PE 数 | CPU 時間 | 経過時間 | メモリサイズ |
|---|---|---|---|---|
| cl | 1 | 30 分 | 60 分 | 2GB |
| d | 1 | 60 分 | 90 分 | 7GB |
| e | 1 | 360 分 | 540 分 | 7GB |
| f | 10 | 60 分 | 90 分 | 7GB × 10 |
| g | 10 | 360 分 | 540 分 | 7GB × 10 |
| h | 40 | 360 分 | 540 分 | 7GB × 40 |
| z | 41 以上 | — | — | — |

表 5. 実行多重度の制御

| Complex Queue | Run Limit | Queue | Run Limit |
|---|---|---|---|
| Vector | 20 | d | 12 |
| | | e | 8 |
| Parallel10 | 5 | f | 3 |
| | | g | 2 |
| Parallel40 | 1 | h | 1 |

表 6. ファイルシステムの割付けと用途

| ファシリティ | 用　　途 | 容　　量 |
|---|---|---|
| #1 | ユーザホーム 1 | 200GB |
| | センター ID ホーム | 100GB |
| | ジョブ凍結、ジョブスワップ | 60GB |
| #2 | 大容量ファイル領域 | 300GB |
| | ジョブ凍結、ジョブスワップ | 60GB |
| #3 | ユーザホーム 2 | 200GB |
| | ジョブ凍結、ジョブスワップ | 16GB |
| #4 | MVPP 連携ホーム | 80GB |
| | ジョブ凍結、ジョブスワップ | 28GB |

表 7. ユーザ当りの許可量とブロックサイズ

| 項目 | Soft Limit | | Hard Limit | | ブロックサイズ |
|---|---|---|---|---|---|
| | 容量 | 個数 | 容量 | 個数 | |
| ホーム | 20GB | 1500 個 | 40GB | 3000 個 | 64KB |
| 大容量ファイル領域 | 100GB | 20 個 | 200GB | 40 個 | 640KB |

表 8. 並列版連立一次方程式のベンチマーク

| ベンチマーク名 | PE 数 | 問題サイズ | 実測値 (GFlops) | 実効性能比 (%) |
|---|---|---|---|---|
| LINPACK Parallel | 63 | 234,360 × 234,360 | 482.5 | 95.7 |
| VPP-ScaLAPACK (1) | 40 | 180,000 × 180,000 | 286.4 | 89.5 |
| MPI-ScaLAPACK (1) | 40(1 × 40) | 180,000 × 180,000 | 271.4 | 84.8 |
| SSLII/VPP | 40 | 130,000 × 130,000 | 253.9 | 79.3 |
| VPP-ScaLAPACK (2) | 40 | 130,000 × 130,000 | 271.0 | 84.6 |
| MPI-ScaLAPACK (2) | 40(1 × 40) | 130,000 × 130,000 | 268.3 | 83.3 |