

## 第 3 章

### 並列計算機システムの性能評価

並列計算機とは、複数台のプロセッサを相互結合網を介して並列に結合して、全体として高い処理能力を発揮させることを目的とした計算機システムである。[23]

このような観点からこれまでに様々な並列計算機が開発されてきた。本章では、本研究で用いた並列計算機システムの概要や、並列処理における評価方法などについて述べる。そしてその後、これらの環境を用いて行った性能評価実験について述べる。

#### 3.1 実行環境

本研究で用いた各並列計算機システムの仕様を、表 3.1 に示す。以下、システムごとの特徴を述べる。

表 3.1: 各計算機システムの仕様

システム	CP-PACS	RS/G000 SP	パソコン・サーバ
ピーク演算スピード / IPU	300MFLOPS	266.4MFLOPS	200MFLOPS
メモリ / IPU	64MBytes	256MBytes	64MBytes
キャッシュ・メモリ / IPU	512KBytes	128KBytes	256KBytes
結合網	ハイバ・クロスバ	多段スイッチ	スイッチング・ハブ
ピーク転送スピード	300MBytes/sec	150MBytes/sec	12.5MBytes/sec
PU 数	16	16	8
その他	疑似ベクトル化機構		

##### 3.1.1 CP-PACS

CP-PACS は、QCD 計算などの物理計算を高速に行うために、筑波大学と日立製作所により開発された並列計算機である。結合網は、多数のクロスバスイッチを x, y, z 各方向に配列した 3次元ハイバクロスバ (Hyper Crossbar) である。演算処理を並列に行う PU (Processing Unit) と、入出力を分散処理する IOU (Input/Output Unit) は各方向のクロスバを繋ぐ Exchanger に接続され、最大三段のクロスバを経由することにより任意のパターンの PU 間データ転送が可能である。各 PU 間を繋ぐ結合網の転送ピーク性能は 300MBytes/sec である。CP-PACS の構成図を図 3.1 に示す。

CP-PACS の各 PU が搭載するプロセッサには、PVP-SW (Pseudo Vector Processor based on Slide Window) 機構が搭載されている。この PVP-SW 機構により、主記憶のアクセス遅延が隠蔽

され、スーパースカラ・プロセッサでありながら効率のよいベクトル処理が可能となり、300MFLOPSのピーク速度を得ている。

OSは、CP-PACSのために開発されたマイクロカーネルとサーバ構成のHI-UX/MPPである。マイクロカーネルは、カーネギ・メロン大学で開発されたMach3.0をベースとしている。また、MPIは、Argonne National Laboratory/Mississippi State Universityにより開発されたMPICHをHI-UX/MPP用に移植したものである。[24]

尚、本研究では、16台のPUを用いた測定まで行った。

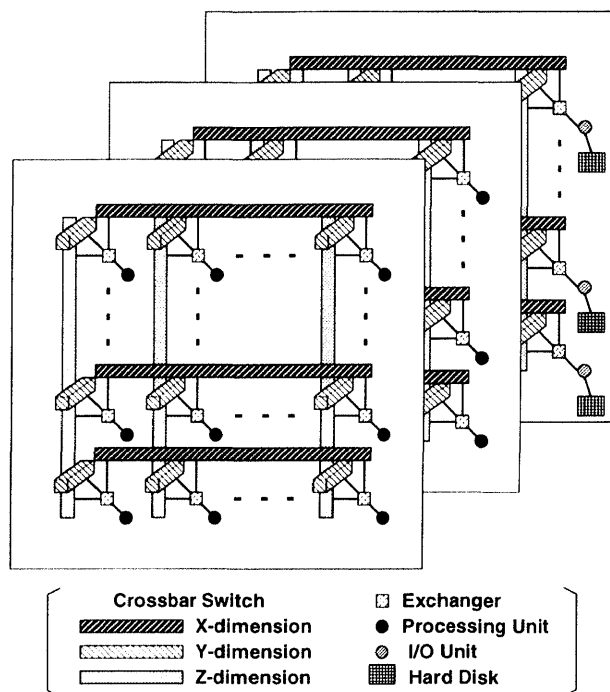


図 3.1: 並列計算機 CP-PACS の構成図

### 3.1.2 RS/6000 SP

並列計算機 RS/6000 SP は、IBM により開発された並列計算機である。各PUは、RS/6000のプロセッサ、メモリなどで構成されており、全体は複数のPUが格納されたフレームを接続することによって構成されている。システム全体の管理は、RS/6000 SPと接続されたコントロール・ワークステーションによって一元的に行われている。各PU間の通信は、RS/6000 SPスイッチと呼ばれるネットワークを介して行われる。これは多段相互結合網のネットワークであり、このピーク性能は150MBytes/secである。また、プロセッサのピーク速度は266.4MFLOPSである。OSは、IBMのAIXの並列処理サポート版である、AIX/6000 V4.1.3であり、MPIはIBMによりAIX用に移植されたものである。[25]

本研究では、16台のPUを用いた測定まで行った。

### 3.1.3 パソコン・クラスタ

パソコン・クラスタの機器構成は、AT互換機8台を100Mbits/secイーサネットに繋いだものである。各パソコンのMPUは、Intel PentiumPro200MHz、メモリは64MBytesである。MPU

のピーク速度は200MFLOPSである。(図3.2)

NFSによりホームディレクトリの共有をしており、データの入出力などが行いやすくなっている。OSはFreeBSD2.2.1、MPIはLAM6.0を用いた。

パソコン・クラスタの特徴としては、非常に安価にシステムを構成出来る反面、通信速度が非常に遅いということが挙げられる。

尚、他の1台のパソコンからLAMの立ちあげ、プログラムの実行などを行い、8台のパソコンは計算だけを行うようにした。

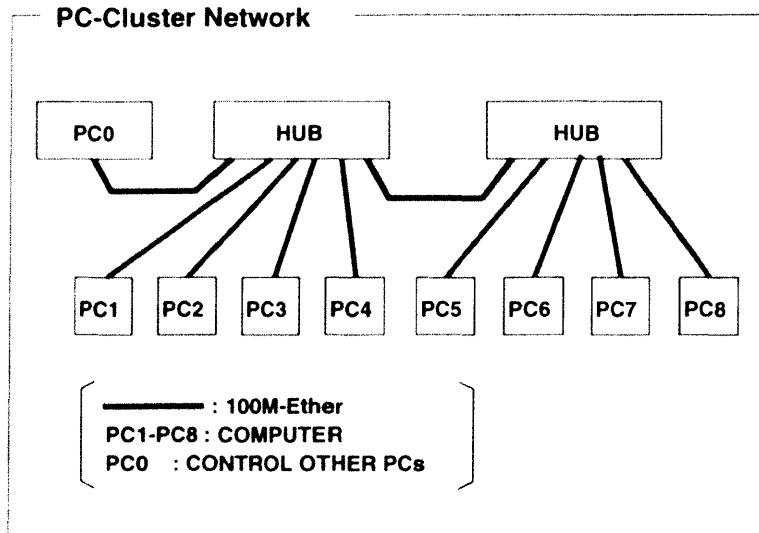


図 3.2: パソコン・クラスタの機器構成図

### 3.2 並列処理の評価基準

1つの仕事の分割数を  $P$  とし  $P$  台の演算装置 (Processing Unit: 以下 PU) により分担して並列に実行したと仮定し、その時の処理時間を  $T_p$  とする。逐次処理は  $P = 1$  のときに相当し、その処理時間を  $T_1$  とすると、理想的には並列処理時間は

$$T_p = \frac{T_1}{P} \quad (3.1)$$

のようになり、従って速度は  $P$  倍に増すと予測される。

しかし一般的には、そうならない場合が多い。例えば、1つの仕事を均等に  $P$  分割出来るとは限らず、場合によっては分割した仕事(タスク)のサイズに不均衡が生ずる。また、分割したためにタスク間で何らかのデータの受渡しが必要になる場合がある。また、ある PU がデータを必要とした時に、他の PU がビジー状態にありデータがすぐには取得出来ないことがある。この時、この PU は何もせずデータを受けとるまで待ち状態になることを意味し、このような場合、効率の著しい低下を招くことがある。

このようなデータ転送やそれに伴う同期などは逐次的処理には含まれていない並列処理特有の無駄な処理であり、これをオーバーヘッドという。

このオーバーヘッドのため実際には,

$$T_P \geq \frac{T_1}{P} \quad (3.2)$$

となる. そこで, 並列処理効率を

$$\alpha \equiv \frac{T_1}{PT_P} \quad (3.3)$$

とする ( $0 \leq \alpha \leq 1$ ). また,

$$S \equiv \frac{T_1}{T_P} \quad (3.4)$$

として速度向上比を定義すると,

$$S = \alpha P \quad (3.5)$$

と表すことが可能である.

このオーバーヘッドを削減し並列計算を効率良く行わせる, 即ち並列処理効率を向上させるためには, 仕事を如何に並列処理可能な部分に分割するか, 与えられたデータを如何にしてPUにマッピングするかなどが問題となる.

### 3.3 性能評価実験

前記の並列計算機システムを用いて, 基本統計量を求める処理, 重回帰分析, 主成分分析を, 変量数とサンプル数を変えながら実行し, 時間を測定した結果とその評価を示す. 文中, PU数を  $P$ , データの変量数を  $M$ , サンプル数を  $N$  と表すことにする.

#### 3.3.1 測定に用いた解析

測定に用いた解析には, オーバヘッドの面からそれぞれ以下に示すような特徴がある.

##### (1) 基本統計量

各PUで計算した値の比較・加算などのために, PU間で  $\log_2 P$  回のデータ転送を行っている. これが, 並列処理を行ったために発生する, PU数による計算のオーバーヘッドとなる.

##### (2) 重回帰分析

また, 逆行列の計算をする時にピボット行の全PUへのブロードキャスト処理が発生する. よって, データが  $M$  変量の場合,  $M$  回のブロードキャストが発生する. 以上が重回帰分析の並列処理によるオーバーヘッドとなる.

##### (3) 主成分分析

解析の前に各PUが受け持つデータの割当を決めてしまい, 計算途中には通信は行わない. よって, 処理の途中にオーバーヘッドは発生しない. 一方で, 固有値を求める処理を含んでいるため, データによっては収束に時間がかかり処理時間が長くなる場合がある.

(固有値計算に用いるデータは, 既に基本統計量を求めた時に全PUに転送済みである)

つまりそれぞれ, (1)PU数により通信回数が決まる処理, (2)変量数によりブロードキャストを行う回数が決まる処理, (3)計算途中には通信を行わない処理, ということが出来る. これら3つの

性質の違う処理を測定することにより、他の処理の適用性の予測にも役立つと考え、これらの解析を用いた性能評価を行うこととした。

(尚、重回帰分析、主成分分析とも、その処理時間はその処理に特有な時間を表しており、各処理の前に行う基本統計量を求める処理に要する時間は含めていない。)

以下に、プログラムのアルゴリズムの面から算出した、各処理にかかる時間 ( $T_{bstat}$ : 基本統計量、 $T_{mreg}$ : 重回帰分析、 $T_{pca}$ : 主成分分析) をあらわした式 (3.6)、(3.7)、(3.8) を示す。式中、 $a_1 \sim a_{10}$  は、各並列計算機システムにより違う係数を表す。また、1回のブロードキャストにかかる時間  $T_{br}$  は、CP-PACS と RS/6000 SP では  $P$  によらずほぼ一定であるが、パソコン・クラスタでは実際は  $P$  回の1対1通信処理を行うので、ほぼ  $P \times T_{tr}$  となる。パソコン・クラスタで重回帰分析を行った場合の処理時間を、式 (3.9) に示す。

$$T_{bstat} = a_1 \cdot M^2 + a_2 \cdot M + a_3 \cdot \frac{MN}{P} + a_4 \cdot \frac{M^2 N}{P} + (a_5 \cdot M^2 + a_6 \cdot M) \log_2 P + a_7 \cdot MT_{tr} \log_2 P \quad (3.6)$$

( $T_{tr}$ : 1回の転送にかかる時間)

$$T_{mreg} = a_1 \cdot M^3 + a_2 \cdot M^2 + a_3 \cdot M + a_4 \cdot \frac{M^4}{P} + a_5 \cdot \frac{M^2}{P} + a_6 \cdot \frac{M^2 N}{P} + a_7 \cdot M^2 T_{br} + a_8 \cdot MT_{br} + a_9 \cdot MT_{tr} \log_2 P + a_{10} \cdot M \log_2 P \quad (3.7)$$

( $T_{br}$ : 1回のブロードキャストにかかる時間)  
( $T_{tr}$ : 1回の転送にかかる時間)

$$T_{pca} = a_1 \cdot M^2 + a_2 \cdot M + a_3 \cdot \frac{M^2 N}{P} + Iter \cdot (a_4 \cdot M^3 + a_5 \cdot M^4) \quad (3.8)$$

( $Iter$ : 固有値計算の収束回数)

[パソコン・クラスタで重回帰分析を行った場合]

$$T_{PCmreg} = a_1 \cdot M^3 + a_2 \cdot M^2 + a_3 \cdot M + a_4 \cdot \frac{M^4}{P} + a_5 \cdot \frac{M^2}{P} + a_6 \cdot \frac{M^2 N}{P} + a_7 \cdot M^2 P T_{tr} + a_8 \cdot M P T_{tr} + a_9 \cdot M T_{tr} \log_2 P + a_{10} \cdot M \log_2 P \quad (3.9)$$

### 3.3.2 各環境での特性

#### CP-PACS の測定結果について

基本統計量の処理では、表 3.2 に示すように、データが大量の場合には並列処理効率は 100% もしくは 100% 以上を示した。この測定結果のうち、PU 数と並列処理効率との関係をグラフで現したものが図 3.3 である。

特に、 $M=50,100$  で  $N=100000$  の時、 $P=8$  の時の効率が一番良いが、これはキャッシュ・メモリの効果が大きいためと思われる。  $P=16$  の時の効率が落ちているが、これはキャッシュ・メモリの利用と言う点では  $P=8$  の時と同じもしくはより有利な条件であるが、PUが増えた分転送回数も増えるので、結果的に効率が落ちていると思われる。

表 3.2において、同じPU数で比較すると、データ量 ( $M \times N$ ) が多いほど効率は良くなっている。これは、データ量が増加しても通信回数は  $P$  によって一定だからである。また、データ量が少ない時には効率がよくないが、これは、PVP-SW 機構はある程度大きなデータ (配列) を処理しないと性能を発揮しないからと思われる。

重回帰分析を行った場合の並列処理効率のグラフを図 3.4に示す。重回帰分析では、 $M$  回のブロードキャスト処理が必要になる。また、同じデータを処理する場合、PU数が多いほど各PUに割り当てられるデータ量は少なくなり、計算時間は減少する。 $P=8$  までは、計算時間はほぼ  $1/P$  となるが通信時間はほぼ変わらないので効率は悪化していった。 $P=16$  の時を見ると、 $M=2,10$  の時効率が改善している。これは、データがキャッシュ・メモリに入り切ってしまったために計算時間が大幅に短縮され、結果的に全体の処理時間が大幅に短縮されたためと思われる。

主成分分析では、計算途中には通信を行わないので、 $M=2,10$  の時にはほぼ 100% の並列処理効率を得られた。 $M=50,100$  と増加すると効率は悪化していくが、これは並列化出来ていない (全PUが行っている) 固有値計算部分の割合が、収束が遅くなることなどにより、並列化で高速に処理出来る部分の割合を越えてしまうからと思われる。(図 3.5)

表 3.2: 基本統計量の並列処理の速度向上比、並列処理効率 (CP-PACS)

M		2				10			
N		1000	10000	50000	100000	1000	10000	50000	100000
PU2/PU1	speed-up	0.40	0.70	2.16	1.61	1.61	2.27	3.41	2.52
	効率 (%)	20.00	34.96	107.89	80.30	80.36	113.34	170.43	125.92
PU4/PU1	speed-up	0.25	0.47	2.05	1.61	0.63	1.55	5.94	7.23
	効率 (%)	6.25	11.65	51.24	40.15	15.63	38.64	148.48	180.82
PU8/PU1	speed-up	0.13	0.20	1.32	1.41	0.28	0.87	4.13	8.37
	効率 (%)	1.56	2.50	16.53	17.67	3.52	10.90	51.58	104.56
PU16/PU1	speed-up	0.48	0.83	4.47	1.39	0.75	2.15	5.49	8.97
	効率 (%)	2.97	5.17	27.92	8.72	4.70	13.43	34.29	56.05

M		50				100			
N		1000	10000	50000	100000	1000	10000	50000	100000
PU2/PU1	speed-up	1.29	1.79	3.86	2.30	1.52	1.99	4.23	2.61
	効率 (%)	64.59	89.49	193.12	115.10	76.19	99.50	211.71	130.59
PU4/PU1	speed-up	0.94	2.42	14.42	9.89	1.48	3.38	17.39	10.58
	効率 (%)	23.49	60.58	360.58	247.30	36.92	84.47	434.76	264.49
PU8/PU1	speed-up	0.53	2.44	22.08	34.65	1.07	5.11	32.05	41.33
	効率 (%)	6.68	30.52	276.03	433.16	13.33	63.90	400.64	516.58
PU16/PU1	speed-up	0.59	2.51	29.66	53.15	0.96	5.57	53.30	74.17
	効率 (%)	3.68	15.69	185.38	332.22	5.99	34.78	333.13	463.55

### RS/6000 SP の測定結果について

基本統計量の処理では、データ量が多い場合には並列処理効率は、ほぼ 100% を示した。特に、 $M=100$ ,  $N=100000$  の時には、ほぼ 99% の効率を得ている。データ量が少ない時には、 $P=4$  をピークに効率は悪化しているが、これは計算量が少ないために通信の影響が大きく出てく

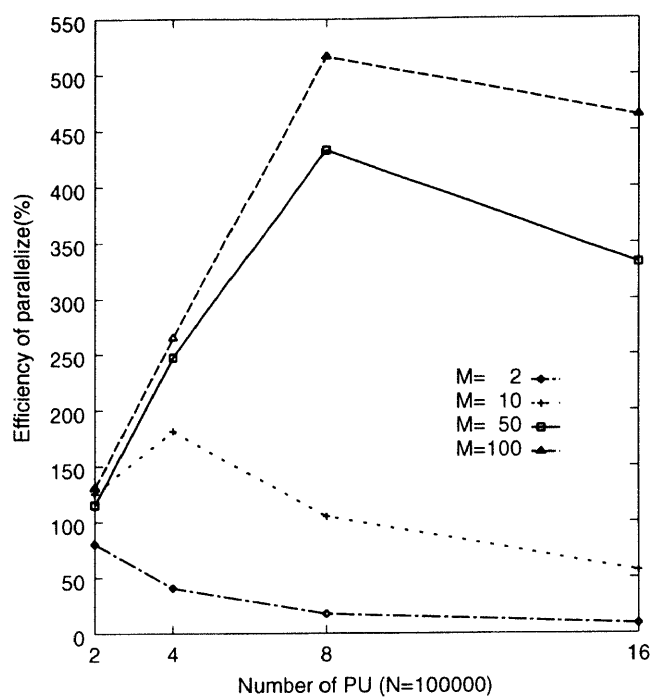


図 3.3: 基本統計量の並列処理効率 (N=100000) (CP-PACS)

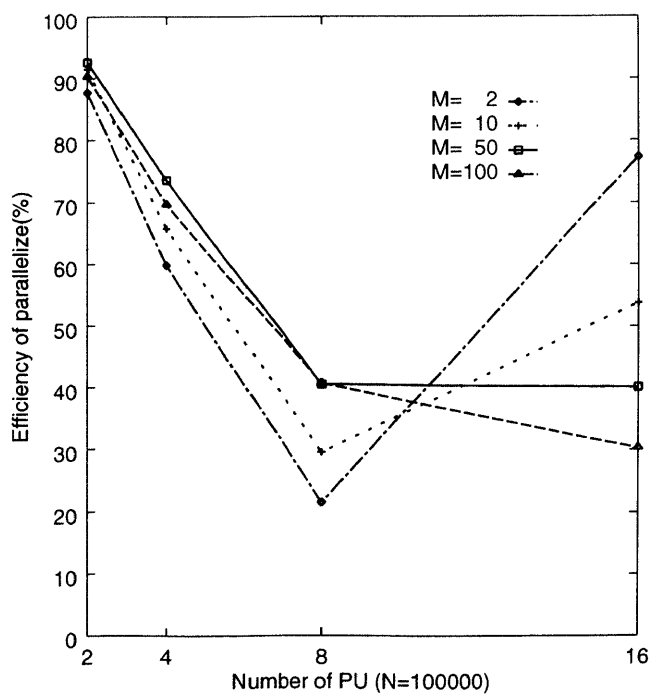


図 3.4: 重回帰分析の並列処理効率 (N=100000) (CP-PACS)

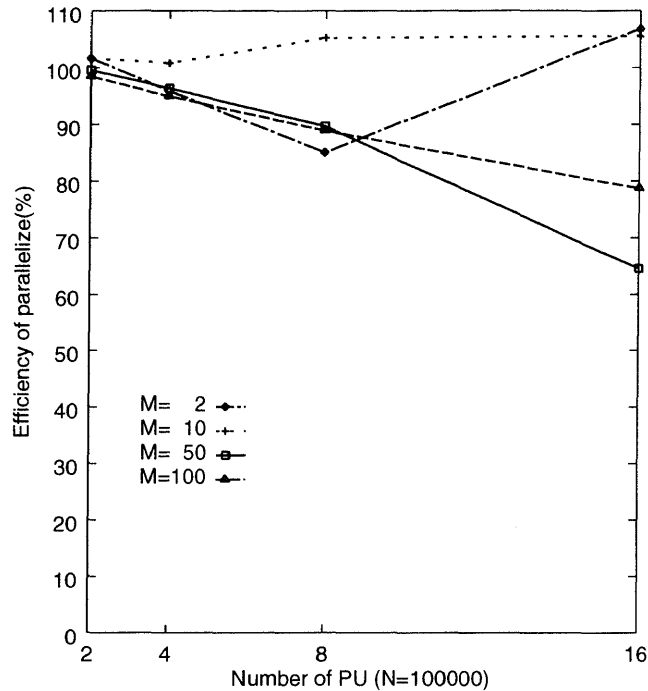


図 3.5: 主成分分析の並列処理効率 (N=100000) (CP-PACS)

るからである。M=100, N=100000 の時でも、PU 数が増えれば効率自体は悪化すると思われる。(図 3.6)

重回帰分析は、CP-PACS の項で述べたのと同様の理由で、P が増加するほど効率は悪化したと考えられる。CP-PACS に比べてキャッシュ・メモリ容量が小さいため、P = 16 まで増えても特に計算時間が劇的に改善されるということは無く、素直な傾向を示した。(図 3.7)

主成分分析は、ほぼ CP-PACS と同じような傾向が見られた。M=100, N=100000 の時、PU 数が増えるほど効率が悪化しているが、CP-PACS の場合と同様に固有値計算部分の影響だと思われる。(図 3.8)

#### パソコン・クラスタの測定結果について

基本統計量では、データ量が少ない時は PU が増えると効率は急速に悪化する。これは、計算量(時間)が少なく通信が遅いためである。キャッシュ・メモリの容量から考えて、一番効果があるのは P=8 の時だと思われるが、通信速度が非常に遅いので、その兼ね合いで P=2 もしくは P=4 の時の効率の方が良いという結果が得られた。(図 3.9)

重回帰分析では、行列計算で M 回のブロードキャスト処理が発生するが、実際は式(3.9)に示したように、M × P 回の 1 対 1 通信処理を行っているので非常に効率は悪い。大量のデータを P=2 で処理した場合でも、20 ~ 30% 前後であった。(図 3.10)

主成分分析では、M=10, N=10000 では、ほぼ 100% の効率を得ているが、M=50,100 と増加すると効率は 71.6%, 54.8% と悪化していく。これも、他のマシンでの傾向と同じように、M に依存する固有値計算部分の割合が、収束の遅さなどにより大きくなっていくためであると思われる。(図 3.11) しかし、処理の途中に通信処理が無いために、他の解析に比べると効率悪化の仕方は小さかった。

(尚、パソコンクラスタでは、メモリ容量の関係で 1 台当たり約 10000 サンプルまでの処理しか行



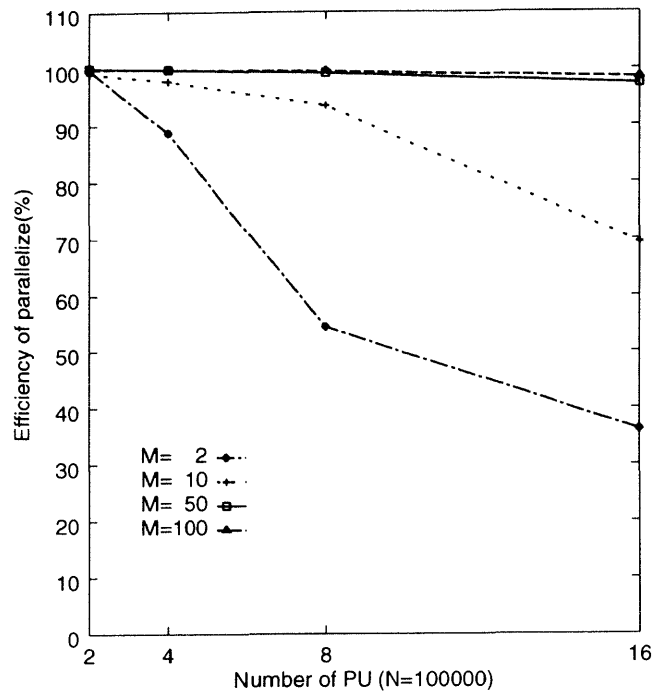


図 3.6: 基本統計量の並列処理効率 (N=100000) (RS/6000 SP)

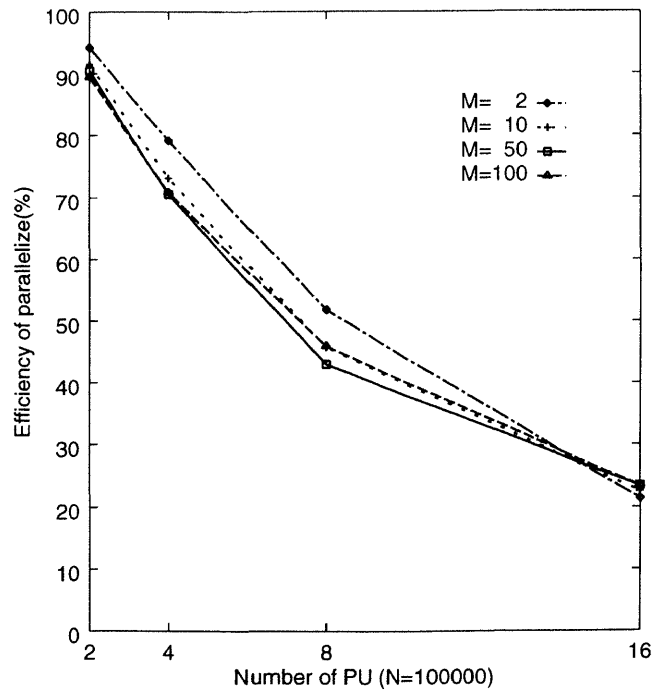


図 3.7: 重回帰分析の並列処理効率 (N=100000) (RS/6000 SP)

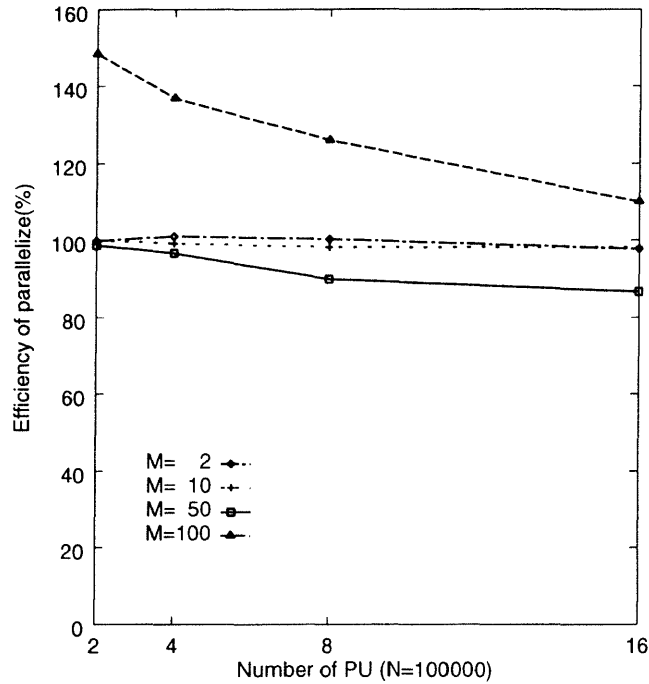


図 3.8: 主成分分析の並列処理効率 (N=100000) (RS/6000 SP)

うことが出来なかった。従って、効率算出にはその測定結果を用いている.)

### 3.3.3 CP-PACS,RS/6000 SP, パソコン・クラスタの相対性能評価

本節では、各処理における各測定環境ごとの実時間の比較とその考察を行う。

(尚、表中の「\*\*\*」の記号は、パソコン・クラスタにおいて、PU 数やメモリ容量の関係上測定出来なかったことを示す。)

#### 基本統計量

表 3.3 より、 $M=2,10$  と小さい時は、 $P=4,8$  で RS/6000 SP の方が CP-PACS よりも速い。また、 $M=50,100$  と大きくなっていくと CP-PACS の方が大幅に速くなっていく。これは、データ量が少ない時は CP-PACS の PVP-SW 機構は効果が発揮されず、大きなデータを扱う場合に効果が得られるからである。また、PU 数が増えると、ハイパクロスバ構造により、指定した PU に衝突無く直接データを転送出来る CP-PACS の方が有利になる。

各環境での速度比は表 3.3 より、 $P=8, M=100, N=50000$  の時、パソコン・クラスタ : CP-PACS : RS/6000 SP = 1 : 14.041 : 1.154。また、表 3.6 より、 $P=16, M=100, N=100000$  の時、CP-PACS:RS/6000 SP = 1 : 0.087。キャッシュ・メモリや CP-PACS の PVP-SW 機構などの影響があり、データ量や PU 数によって完全に同じ割合にはなっていない。

実際に処理にかかった実時間を、表 3.7 に示す。表 3.7 を見ると、測定を出来た範囲内では、パソコン・クラスタは RS/6000 SP とほぼ同じ時間で処理出来ているが、CP-PACS と比べると数倍から十数倍の差があった。

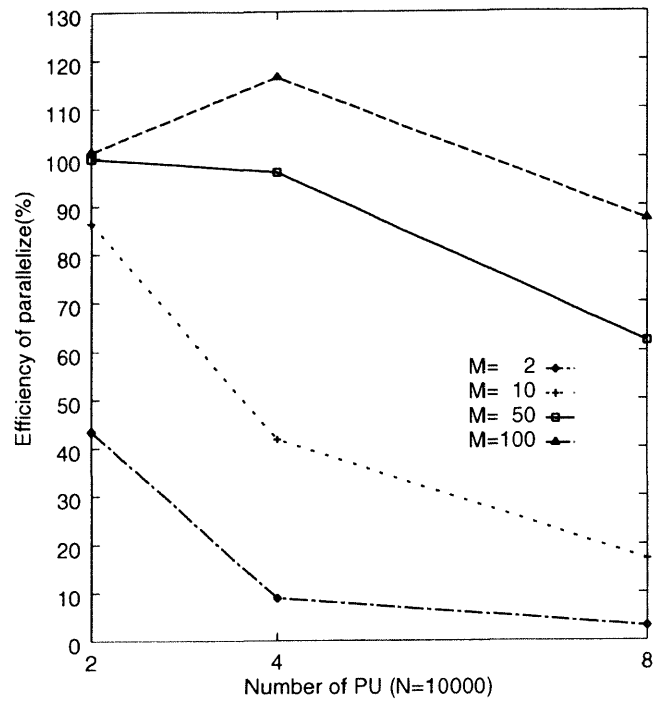


図 3.9: 基本統計量の並列処理効率 (N=10000) (パソコン・クラスタ)

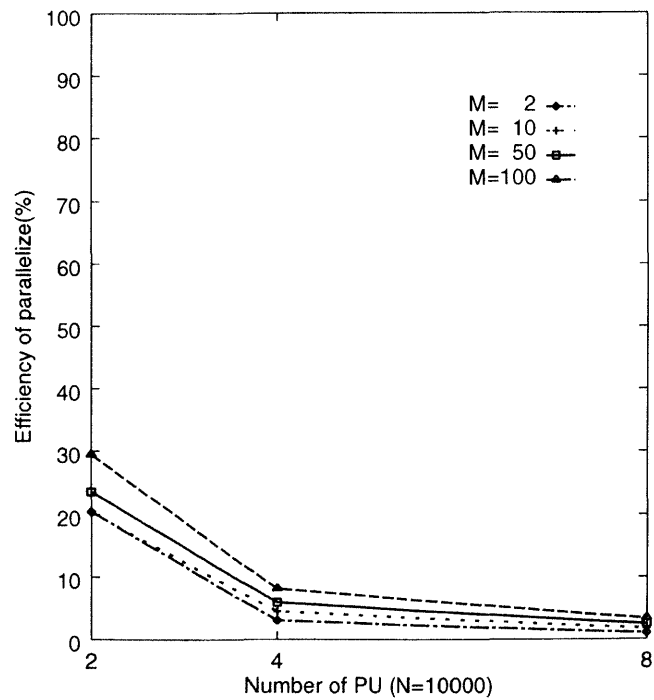


図 3.10: 重回帰分析の並列処理効率 (N=10000) (パソコン・クラスタ)

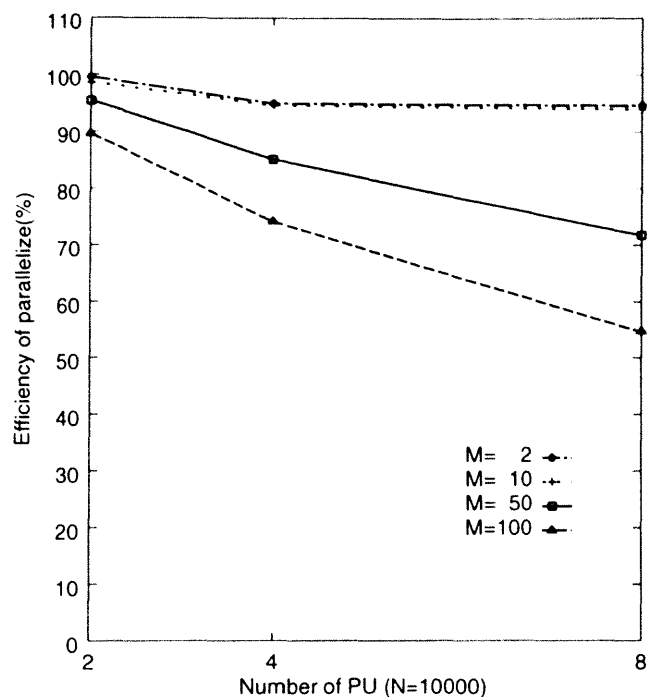


図 3.11: 主成分分析の並列処理効率率 (N=10000) (パソコン・クラスタ)

### 重回帰分析

表 3.4 より,  $M, N$  がいくつでも,  $P=8$  までは RS/6000 SP の方が速い. ただし, データ量が多くなるほどその差は縮まり,  $M=100$  ではほぼ同じになる. この内容を詳しく見ると, 通信を頻繁に行う部分で差がついてしまっている. 本来, ハイパクロスバ構造を取る CP-PACS の方が高速なはずであるが, MPI のコンパイラの最適化が不十分なのか, 少ないデータを頻繁に送るという処理では高速性が発揮されていない. データ量が増加すると差が縮まっているが, これは計算部分の実行時間の割合が大きくなっていくので, PVP-SW 機構で高速な処理が出来る分だけ CP-PACS の処理効率率が向上しているからだと思われる.

表 3.6 より,  $P=16$  の時 RS/6000 SP の速度が急に遅くなっているが, これは RS/6000 SP の構造上の特徴によると思われる. RS/6000 SP は 16PU を一つのフレームに格納し, それを繋ぎ合わせるという構造を取っている. 各フレーム内には, ゲートウェイや単体演算用の PU が設定されており, 全部を並列計算に使用出来る訳ではない. そこで, 16PU を使用した計算をする場合は, 二つ以上のフレームをまたいで処理を行うことになり, その分だけオーバーヘッドが大きくなっているのではないかと考えられる.

パソコン・クラスタでの効率が極端に悪くなっているが, これは行列計算で行われるブロードキャスト処理のためだと思われる. パソコン・クラスタで使用した LAM/MPI システムでは, 一般的な TCP/IP 通信を用いて通信を行っている. そこで, 前述したように, ブロードキャスト処理とはいっても実際はパソコン・クラスタの台数分だけ一対一通信を繰り返しているだけである. 従って, パソコン・クラスタの台数が増えるほどブロードキャスト処理にかかる時間は長くなっていき, 全体の効率は悪化していくのである.

各環境での速度比は, 表 3.4 より,  $P=8, M=100, N=50000$  の時, パソコン・クラスタ:CP-PACS:RS/6000 SP = 1 : 9.180 : 10.330 である. また, 表 3.6 より,  $P=16, M=100, N=100000$  の時, CP-PACS:RS/6000 SP = 1 : 0.738 である. 重回帰分析の処理も, システムの構造的な特性

からデータ量によって同じ割合にはなっていない。

実際に処理にかかった実時間を、表 3.8 に示す。表 3.8 を見ると、パソコン・クラスタは PU 数が増えるに従って処理時間が増大しており、他の 2 システムに比べると、大量のデータの場合に実用的な時間で処理することは難しいと言える。

### 主成分分析

表 3.5 より、全体的に RS/6000 SP の方が高速である。特に、データ量が少ない時の方がより高速である。また、データ量が多くても、約 1.2 ~ 1.3 倍前後の差がある。これは、CP-PACS のコンパイラではコンパイル時に PVP-SW 機構を使えるかどうかの判別をしているが、まだ完全ではなく、ループ処理で PVP-SW 機構を使えていない場合が見られることと、PVP-SW 機構を使える場合でも、少ないデータでは効果が発揮出来ないからである。ただし、今回実験を行ったよりも更に大量のデータを処理していく場合には、PVP-SW 機構を使用出来ている部分の処理の高速化が望めるので、差は縮まっていくと思われる。

また、表 3.6 より、重回帰分析の時に見られたような、RS/6000 SP の PU が 16 台の時の顕著な速度低下は見られない。これは、主成分分析の処理の途中には、通信処理が含まれていないからである。

各環境での速度比は、表 3.5 より、 $P=8$ 、 $M=100$ 、 $N=50000$  の時、パソコン・クラスタ:CP-PACS:RS/6000 SP = 1 : 3.204 : 3.984 である。また、表 3.6 より、 $P=16$ 、 $M=100$ 、 $N=100000$  の時、CP-PACS:RS/6000 SP = 1 : 1.309 である。計算途中に通信処理がないという特性のため、パソコン・クラスタでもそれほど大きな速度低下はなく、表 3.9 から実用に耐え得る時間で処理出来ていると言える。

表 3.3: PC-CLUSTER, CP-PACS, RS/6000 SP の速度比 (基本統計量)

PU	M	N	PC	CP	SP	PU	M	N	PC	CP	SP
1	2	1000	1	0.189	0.277	4	2	1000	1	1.785	5.215
		10000	1	1.159	0.723			10000	1	2.079	3.826
		50000	***	***	***			50000	1	2.358	2.651
	10	1000	1	1.000	0.963		10	1000	1	4.303	8.638
		10000	1	5.323	1.249			10000	1	4.932	2.750
		50000	***	***	***			50000	1	4.880	1.744
	50	1000	1	9.109	0.826		50	1000	1	11.260	7.018
		10000	1	13.525	1.107			10000	1	8.450	1.180
		50000	***	***	***			50000	1	12.752	1.162
	100	1000	1	11.747	0.761		100	1000	1	13.281	5.884
		10000	1	14.369	1.058			10000	1	10.417	0.952
		50000	***	***	***			50000	1	14.172	1.090
2	2	1000	1	0.730	1.433	8	2	1000	1	1.428	4.950
		10000	1	1.281	1.099			10000	1	1.335	5.195
		50000	1	2.377	1.511			50000	1	1.716	3.724
	10	1000	1	3.389	2.328		10	1000	1	2.982	9.331
		10000	1	6.979	1.356			10000	1	3.449	5.093
		50000	1	4.332	1.390			50000	1	2.450	2.346
	50	1000	1	8.731	1.670		50	1000	1	9.040	13.665
		10000	1	12.132	1.098			10000	1	6.663	1.813
		50000	1	6.504	1.115			50000	1	10.830	1.265
	100	1000	1	11.320	1.600		100	1000	1	12.540	12.914
		10000	1	14.145	1.046			10000	1	10.538	1.257
		50000	1	7.474	1.175			50000	1	14.041	1.154

表 3.4: PC-CLUSTER, CP-PACS, RS/6000 SP の速度比 (重回帰分析)

PU	M	N	PC	CP	SP	PU	M	N	PC	CP	SP
1	2	1000	1	0.024	0.798	4	2	1000	1	1.246	14.508
		10000	1	0.247	1.452			10000	1	1.251	14.663
		50000	***	***	***			50000	1	1.095	7.763
	10	1000	1	0.135	1.628		10	1000	1	3.640	18.034
		10000	1	0.729	2.923			10000	1	3.345	15.020
		50000	***	***	***			50000	1	2.945	9.611
	50	1000	1	0.967	1.668		50	1000	1	9.432	14.919
		10000	1	2.035	2.971			10000	1	8.631	13.316
		50000	***	***	***			50000	1	6.488	9.579
	100	1000	1	1.367	1.451		100	1000	1	10.595	11.289
		10000	1	2.232	2.236			10000	1	10.068	10.476
		50000	***	***	***			50000	1	8.640	8.471
2	2	1000	1	0.511	11.250	8	2	1000	1	0.911	16.256
		10000	1	0.606	4.911			10000	1	0.981	14.577
		50000	1	0.626	2.447			50000	1	1.032	11.145
	10	1000	1	2.102	12.046		10	1000	1	2.559	15.629
		10000	1	2.428	7.819			10000	1	2.525	14.744
		50000	1	1.839	4.711			50000	1	2.646	11.350
	50	1000	1	6.457	8.939		50	1000	1	7.214	14.093
		10000	1	5.186	7.240			10000	1	7.351	13.596
		50000	1	3.674	5.285			50000	1	6.530	11.165
	100	1000	1	5.322	5.545		100	1000	1	9.656	12.173
		10000	1	5.232	5.282			10000	1	9.745	11.718
		50000	1	5.015	4.731			50000	1	9.180	10.330

表 3.5: PC-CLUSTER, CP-PACS, RS/6000 SP の速度比 (主成分分析)

PU	M	N	PC	CP	SP	PU	M	N	PC	CP	SP
1	2	1000	1	0.142	0.410	4	2	1000	1	0.054	0.566
		10000	1	0.230	0.454			10000	1	0.160	0.453
		50000	***	***	***			50000	1	0.225	0.467
	10	1000	1	0.662	1.526		10	1000	1	0.172	1.495
		10000	1	0.816	1.592			10000	1	0.595	1.613
		50000	***	***	***			50000	1	0.782	1.601
	50	1000	1	2.302	3.385		50	1000	1	1.483	2.166
		10000	1	3.232	4.328			10000	1	2.864	3.797
		50000	***	***	***			50000	1	3.374	4.281
	100	1000	1	2.794	3.097		100	1000	1	2.580	2.742
		10000	1	3.263	4.372			10000	1	3.007	3.393
		50000	***	***	***			50000	1	3.323	4.391
2	2	1000	1	0.103	0.470	8	2	1000	1	0.015	0.626
		10000	1	0.207	0.454			10000	1	0.085	0.460
		50000	1	0.451	0.883			50000	1	0.208	0.560
	10	1000	1	0.406	1.466		10	1000	1	0.066	1.432
		10000	1	0.715	1.589			10000	1	0.288	1.548
		50000	1	0.913	1.727			50000	1	0.735	1.708
	50	1000	1	1.912	3.000		50	1000	1	1.534	2.177
		10000	1	2.844	3.948			10000	1	2.356	3.410
		50000	1	7.054	8.891			50000	1	3.011	4.134
	100	1000	1	2.663	2.912		100	1000	1	2.531	2.116
		10000	1	3.154	4.065			10000	1	2.854	2.966
		50000	1	3.489	4.800			50000	1	3.204	3.984



表 3.6: N=100000 の時の, CP-PACS, RS/6000 SP の速度比

基本統計量				重回帰分析				主成分分析			
PU	M	CP	SP	PU	M	CP	SP	PU	M	CP	SP
1	2	1	0.959	1	2	1	3.524	1	2	1	1.988
	10	1	0.690		10	1	2.442		10	1	1.959
	50	1	0.425		50	1	1.535		50	1	1.280
	100	1	0.411		100	1	0.957		100	1	0.750
2	2	1	1.193	2	2	1	3.783	2	2	1	1.954
	10	1	0.543		10	1	2.440		10	1	1.930
	50	1	0.370		50	1	1.498		50	1	1.270
	100	1	0.315		100	1	0.948		100	1	1.413
4	2	1	2.117	4	2	1	4.664	4	2	1	2.095
	10	1	0.373		10	1	2.714		10	1	1.927
	50	1	0.172		50	1	1.473		50	1	1.282
	100	1	0.155		100	1	0.973		100	1	1.349
8	2	1	2.951	8	2	1	8.526	8	2	1	2.349
	10	1	0.616		10	1	3.767		10	1	1.829
	50	1	0.097		50	1	1.631		50	1	1.286
	100	1	0.079		100	1	1.077		100	1	1.325
16	2	1	3.988	16	2	1	0.976	16	2	1	1.820
	10	1	0.853		10	1	1.026		10	1	1.823
	50	1	0.125		50	1	0.898		50	1	1.717
	100	1	0.087		100	1	0.738		100	1	1.309

表 3.7: 基本統計量の処理時間 [second]

M	N	PU	PC	CP	SP
50	50000	1	***	21.347	64.293
		2	35.948	5.527	32.233
		4	18.872	1.480	16.242
		8	10.470	0.967	8.273
		16	***	0.720	4.222
100	50000	1	***	89.850	269.711
		2	158.607	21.220	135.009
		4	73.222	5.167	67.189
		8	39.362	2.803	34.100
		16	***	1.686	16.796
100	100000	1	***	220.270	536.353
		2	***	84.337	268.048
		4	***	20.820	134.294
		8	***	5.330	67.398
		16	***	2.970	34.064

表 3.8: 重回帰分析の処理時間 [second]

M	N	PU	PC	CP	SP
50	50000	1	***	12.637	8.353
		2	26.599	7.240	5.033
		4	35.726	5.507	3.730
		8	38.964	5.967	3.490
		16	***	3.215	3.787
100	50000	1	***	37.628	39.331
		2	111.725	22.278	23.615
		4	143.230	16.577	16.908
		8	155.093	16.894	15.015
		16	***	11.964	15.914
100	100000	1	***	64.190	67.066
		2	***	35.523	37.469
		4	***	23.053	23.698
		8	***	19.673	18.267
		16	***	13.201	17.893

表 3.9: 主成分分析の処理時間 [second]

M	N	PU	PC	CP	SP
50	50000	1	***	5.476	4.371
		2	19.926	2.825	2.241
		4	5.071	1.503	1.184
		8	2.630	0.874	0.636
		16	***	0.553	0.358
100	50000	1	***	23.172	16.566
		2	41.819	11.986	8.712
		4	21.270	5.167	4.844
		8	11.572	3.612	2.904
		16	***	2.229	1.838
100	100000	1	***	45.571	60.754
		2	***	23.172	16.405
		4	***	11.997	8.896
		8	***	6.405	4.835
		16	***	3.620	2.766

### 3.4 まとめ

基本統計量では、データのPU数によって通信回数は決まっている。データ量が少ない時、特に $M=2,10$ など変量数が小さい時にはパソコン・クラスタでも実用に耐え得る速度で処理出来る。しかし、 $M, N$ が大きくなってくるとその差は大きくなり、 $M=100, N=100000$ 前後ではPVP-SW機構により高速なベクトル演算が出来、ハイパクロスバ構造でデータ転送の高速なCP-PACSが一番有利である。

重回帰分析では、ブロードキャスト処理によりデータを頻繁にやりとりする。そのため、ネットワーク速度の遅いパソコン・クラスタでは非常に時間がかかる。また、本来一番高速な通信能力を持っているはずのCP-PACSは、ユーザによるチューニングが行われることをある程度前提として開発されているのか、本研究のようにMPIで記述したプログラムではスペック通りの速度を発揮出来なかった。そのため、本研究で測定した範囲内ではRS/6000 SPが一番適しているという結果が得られた。

主成分分析では、処理の途中には通信を行っていない。よって、計算の速度のみが影響すると思われるが、本研究ではコンパイラの関係でCP-PACSでPVP-SW機構を使い切れていない部分があり、RS/6000 SPの方が高速であるという結果が出た。また、通信がないおかげでパソコン・クラスタの処理の速度低下もそれほど大きくはなく、実用に耐え得るレベルの速度が得られた。

本研究では、プログラムの汎用的な使用を考え、通信ライブラリとしてMPIを使用し、敢えて各機種固有のライブラリを用いたチューニングなどは行わなかった。その影響で、CP-PACSでは十分に能力を発揮出来なかった部分もあるが、様々なプラットフォームでそのまま利用出来るプログラムということの利点は大きいと考える。