流体解析ベンチマークコードによる ハイパフォーマンスコンピュータの性能評価

渡部 善隆*

概 要

これまでのハイパフォーマンスコンピュータは,「ベクトル並列型計算機 vs. PC クラスタ」という構図であり,並列計算機の性能を引き出すためには,MPI を代表とする分散メモリ環境を意識したプログラミングが必要でした.しかし最近は,32CPU 以上を持つ共有メモリ型の計算機が次々に開発され,OpenMP やスレッド並列,自動並列化による大規模数値計算も盛んに行なわれつつあります.

異なるアーキテクチャを持つ計算機群を評価するためには,CPU の単体演算性能の他にも,ネットワーク転送性能,メモリのロード / ストア性能,それらを実現する OS,コンパイラ性能など様々な角度からの検証が必要です.

本稿では,その「切口」のひとつとして,圧縮性および非圧縮性流体解析ベンチマークコードを用いて測定した共有メモリ型並列計算機,PC クラスタ,ベクトル並列計算機の性能評価結果を紹介し,得られた知見を幾つか報告します.

1 数値流体力学のベンチマークコード

性能評価に用いたプログラムを表 1 に示します.

表 1: ベンチマークコード

名称	HimenoBMT	ShimaBMT
作成者	理化学研究所・姫野 龍太郎 氏	川崎重工業・嶋 英志 氏
言語	Fortran	Fortran
分野	非圧縮性流体解析	圧縮性流体解析
URL	http://w3cic.riken.go.jp/HPC/HimenoBMT/	http://www.geocities.jp/esist88/benchmark/

HimenoBMT, ShimaBMT[1] とも,1 秒間に実行された浮動小数点演算数の値 FLOPS (FLoating Operations per Second) の値をメガ (Mega) 単位で測定します.以下,本稿では "MFLOPS" と記述します.

FLOPS 値は,計算機の演算性能を計る主要な指標のひとつです.HimenoBMT, ShimaBMT とも,実行に必要な浮動小数点演算数が算定可能です.その演算数を,実際の処理に要した

*九州大学情報基盤センター研究部 E-mail: watanabe@cc.kyushu-u.ac.jp

[1]嶋氏の Web サイトでは "ShimaBMT" という呼び方は行なっておりません.HimenoBMT と対比する便宜上,私が勝手に名付けたものです.

経過時間で割ることで FLOPS 値が計算できます．したがって，FLOPS 値が高いほど高速に浮動小数点演算の処理が行なわれたことを意味します．また，処理を並列に行なうことで，さらなる値の向上が期待されます．

なお，ShimaBMT では，MFLOPS 値を 3 回にわたって測定します．本稿のデータは測定値の中間を採用しました．

1.1 HimenoBMT

HimenoBMT は，熱伝導場や非圧縮性流体など物理・工学の広い範囲で現れる Poisson 方程式を 3 次元の一般座標系による差分法により離散化し，Jacobi 反復法により近似解を求めるプログラムです．HimenoBMT に関する詳細は文献 [1] を参照願います．

言語は Fortran と C が公開されており，それぞれ，逐次版，OpenMP による並列版，MPI による並列版に分かれています．今回は Fortran コードを用いて評価を行ないました．

1.1.1 計算サイズ

HimenoBMT の計算サイズは，3 次元立方体領域の分割数 $(i \times j \times k)$ によって図 2 の通り 5 種類が用意されています．記憶容量は NEC SX7 のハードウェア情報によるもので，特に小さいサイズは，この数値よりも小さい記憶容量で実行できると思われます．

表 2: HimenoBMT の計算サイズ

サイズ	$i \times j \times k$	記憶容量
XS	64 × 32 × 32	48MB
S	128 × 64 × 64	64MB
M	256 × 128 × 128	272MB
L	512 × 256 × 256	1856MB
XL	1024 × 512 × 512	14448MB

各計算サイズとも主要ループは四則演算のみであり，サイズを大きくしたときには，計算機のメモリアクセス能力が顕著に出ることが知られています．

1.1.2 MPI プログラム

MPI プログラムでは，分割数 i, j, k をそれぞれ $\hat{i}, \hat{j}, \hat{k}$ 分割し，$\hat{i} \times \hat{j} \times \hat{k}$ 並列実行します．また，配列も $\hat{i} \times \hat{j} \times \hat{k}$ 分割されます．このため，翻訳時に $\hat{i}, \hat{j}, \hat{k}$ の値をパラメータとして指定する必要があります．性能測定では，表 3 に示す分割指定を行ないました．

また，逐次版，OpenMP 版が動的配列宣言を行なっているのに対し，MPI 版は静的配列宣言を行なっているため，計算環境によっては MPI の 1 並列の方が逐次版より高速になることが確認されています[2]．

[2]当初は原因が分からずに苦労しました．コンパイラのレベルアップによってかなりの性能向上が得られることもありました．

表 3: MPI プログラムの分割指定

並列度	$\hat{i}$	$\hat{j}$	$\hat{k}$
1	1	1	1
2	2	1	1
4	2	2	1
8	2	2	2
16	4	2	2
32	4	4	2
48	4	4	3
60	5	4	3
64	4	4	4

1.1.3 倍精度版

HimenoBMT のオリジナルプログラムは，主要な変数が単精度です．科学技術計算の多くが倍精度で行なわれていることを考慮し，倍精度版を別途作成し性能を測定しました．

倍精度版の変更点は，変数の宣言部分とそれに伴う定数などの単精度から倍精度への書き換えのみです．

1.2 ShimaBMT

ShimaBMT は，圧縮性流体解析に特徴的な近似リーマン流速の部分を用いてベンチマークコードとすることで，圧縮性数値流体力学コードの特徴を持たせることを狙ったものです．

ソースコードは Fortran の倍精度逐次版で，主要ループに平方根など四則演算以外の計算が入ります．メモリアクセスに比べ演算数が多いのが特徴です．

1.2.1 変更点

オリジナルプログラムは時間計測関数が CPU 時間を計測する上に機種依存のため，自動並列化によって CPU 時間の他に通信時間も発生することを考慮し，経過時間測定関数に差し換えました．また，FORTRAN 77 仕様のため，動的配列宣言を含め Fortran 95 規格に沿って書き換えを行ないました．主要ループについては，文法上の変更は行なっていません．

1.2.2 計算サイズ

ShimaBMT の計算サイズは，2 次元配列 $(i \times j)$ の大きさで決定します．次元数は任意に変更可能です．今回は，図 4 に示すサイズで測定を行ないました．記憶容量は，2 次元配列すべての確保に必要なバイト数を積算したもので，実行の際には他の変数の領域や OS の作業用領域などが更に必要です．

表 4: ShimaBMT の計算サイズ

サイズ	$i \times j$	記憶容量
100	100 × 100	0.85MB
200	200 × 200	3.4MB
2000	2000 × 2000	336MB
5000	5000 × 5000	2098MB
9000	9000 × 9000	6799MB
13000	13000 × 13000	14185MB

2 測定データ

この節では，5 つの計算機システムを用いて測定したデータを紹介します．値はすべて MFLOPS です．表中の”—” は，記憶容量不足で実行できなかったことを意味します．

2.1 FUJITSU VPP5000

九州大学情報基盤センターおよび筑波大学学術情報処理センターの分散メモリ型ベクトル並列計算機 FUJITSU VPP5000 を使用しました．仕様は以下の通りです．

プロセッサ	9.6GFLOPS/PE (Processing Element)
主記憶容量	8GB/PE または 16GB/PE
1 次キャッシュ	128KB
2 次キャッシュ	2MB
ネットワーク	クロスバー; 3.2GB/秒/PE×2
OS	UXP/V V20L10 X02111
コンパイラ	Fujitsu UXP/V Fortran V20L20 Driver L02091
最適化オプション	`-Kfast`

2.1.1 HimenoBMT

表 5 に HimenoBMT の測定結果を示します．逐次および MPI の 32 並列までが九州大学，MPI の 48 並列以上が筑波大学での測定結果です．

表 5: VPP5000: HimenoBMT

単精度

サイズ	XS	S	M	L	XL
逐次	1524	2628	3537	4363	4332
MPI 1	3048	3858	4061	4497	4573
MPI 2	2161	3842	5754	7454	8415
MPI 4	3845	8820	12048	14352	16745
MPI 8	4682	14810	22476	28459	32913
MPI 16	2928	12266	29058	36771	49473
MPI 32	2249	13834	46818	70246	95122
MPI 48	2147	14538	56958	97192	141434
MPI 60	2045	14531	61551	121719	154506
MPI 64	2060	14972	67478	130909	184653

倍精度

サイズ	XS	S	M	L	XL
逐次	1481	2336	3378	3911	—
MPI 1	2467	3183	3576	4214	—
MPI 2	2502	4248	5489	6172	7591
MPI 4	3718	7732	10592	11891	15316
MPI 8	4696	13381	19308	24673	30559
MPI 16	2970	12119	26677	34022	41861
MPI 32	2262	13889	44589	65157	84201
MPI 48	2097	14471	54124	89521	128493
MPI 60	2043	14500	55641	105344	145791
MPI 64	2087	15074	64780	119332	167256

単精度，倍精度ともに計算サイズが大きくなるにしたがって性能が高くなることがわかります．また，倍精度の値が単精度に比較して若干低くなっています．

2.1.2 ShimaBMT

表 6 に ShimaBMT の測定結果を示します．VPP5000 は分散並列型計算機のため，自動並列化はサポートされていません．

表 6: VPP5000: ShimaBMT

N	100	200	500	2000	5000	9000	13000
逐次	2853	3854	4221	4453	4486	4479	4435

サイズが大きくなるに従い，ベクトル長が長くなり，ピーク性能の 50%近くの性能が得られました．

2.2 FUJITSU PRIMEPOWER850

九州大学情報基盤センターのスカラ SMP 計算機 FUJITSU PRIMEPOWER 850 の 8CPU までを使用しました．仕様は以下の通りです．

プロセッサ	SPARC64 V; 1.35GHz; 16CPU
主記憶容量	24GB
1 次キャッシュ	256KB
2 次キャッシュ	8MB
OS	Solaris 8 Generic_114665-02
コンパイラ	Fujitsu Fortran Compiler Driver Version 5.3 P-id: 912528-01
最適化オプション	`-Kfast`

2.2.1 HimenoBMT

表 7 に HimenoBMT の測定結果を示します．PRIMEPOWER 850 の理論ピーク性能は，1

表 7: PRIMEPOWER 850: HimenoBMT

単精度

サイズ	XS	S	M	L	XL
逐次	453	454	463	467	507
自動並列化 1	448	452	457	461	464
自動並列化 2	904	899	915	927	949
自動並列化 4	1678	1735	1749	1820	1875
自動並列化 8	3347	2646	2830	2868	3607
OpenMP 1	452	454	463	467	470
OpenMP 2	905	901	917	930	975
OpenMP 4	1679	1763	1776	1824	1923
OpenMP 8	3329	2642	2972	2813	3711
MPI 1	543	544	583	590	553
MPI 2	1037	1157	1164	1175	1181
MPI 4	2093	2056	2054	2057	2193
MPI 8	4093	2958	3027	2875	3726

倍精度

サイズ	XS	S	M	L	XL
逐次	355	355	328	337	—
自動並列化 1	350	340	336	340	—
自動並列化 2	695	677	653	650	—
自動並列化 4	1228	1116	1132	1174	—
自動並列化 8	2803	1481	1483	1595	—
OpenMP 1	350	348	334	342	—
OpenMP 2	687	678	635	653	—
OpenMP 4	1212	1138	1090	1189	—
OpenMP 8	2700	1255	1456	1660	—
MPI 1	312	311	320	333	—
MPI 2	553	540	540	579	—
MPI 4	1020	945	1000	1023	—
MPI 8	2537	1421	1421	1629	—

クロックで 4 命令の実行が可能なことから，1CPU で 4×1.35GFLOPS $\approx$ 5400MFLOPS です[3]．HimenoBMT に関しては，理論性能値の 1/10 程度にとどまりました．

[3] 1GFLOPS=1000MFLOPS とするか，1GFLOPS=1024MFLOPS とするかで若干数値が異なりますのでだいたいの値で表示します．以下同じです．

【参考】最適化オプション

測定したPRIMEPOWERについては，メモリの構成上インタリーブ数が少ないため，後継のHPC2500に比べて性能が出ていないのではという指摘を富士通さんよりいただきました．また，一般的な最適化オプション `-Kfast` の他に，HimenoBMTの性能を最大に引き出すメーカー推奨オプション`-Kfast_GP2=2 -Klargepage=2 -Kprefetch -Kstriping=11` を指定した測定結果を表8に示します．

表 8: PRIMEPOWER 850: HimenoBMT(最適化)

	XS	S	M	L
逐次・単精度	693	643	662	649
逐次・倍精度	327	330	321	324

最適化オプションの指定によって，単精度版は1.4倍前後の性能向上を得ています．倍精度版は「微減」という結果になりました．

2.2.2 ShimaBMT

表9にShimaBMTの測定結果を示します．

表 9: PRIMEPOWER 850: ShimaBMT

N	100	200	500	2000	5000	9000	13000
逐次	507	474	446	448	448	449	449
自動並列化 1	490	456	436	439	440	430	441
自動並列化 2	707	574	739	838	856	852	868
自動並列化 4	866	592	1192	1591	1656	1676	1700
自動並列化 8	783	556	1331	2662	2787	2983	3051

ShimaBMTに関しても，HimenoBMTと同様の性能値（ピーク性能の 1/10 程度）となりました．PRIMEPOWER850の特徴として，サイズの大きさによる性能差が少ないことがあげられます．

2.3 Hewlett Packard hpserver zx2000

PCクラスタ代表として，情報基盤センターに導入されたhpserver zx2000を用いて性能を測定しました．仕様は以下の通りです．

プロセッサ	Itanium2 900MHz; 12CPU
主記憶容量	512MB/CPU
1 次キャッシュ	32KB
2 次キャッシュ	256KB
3 次キャッシュ	1.5MB
ネットワーク	Myricom Myrinet2000 M3F-PCI64B (SRAM 2MB) GM1.6.4; MPICH-GM 1.2.5..10
OS	Red Hat Linux Advanced Workstation 2.1
コンパイラ	Intel Compiler version 7.1
最適化オプション	`-O3`

2.3.1 HimenoBMT

表 10 に HimenoBMT の測定結果を示します．

表 10: zx2000: HimenoBMT

単精度

サイズ	XS	S	M	L	XL
逐次	1021	1037	931	—	—
MPI 1	1028	1054	1367	—	—
MPI 2	1887	1739	1831	14	—
MPI 4	3204	3301	3417	65	—
MPI 8	5602	6372	6974	8782	—

倍精度

サイズ	XS	S	M	L	XL
逐次	684	715	1.9	—	—
MPI 1	692	728	41	—	—
MPI 2	823	1244	1284	—	—
MPI 4	1636	2331	2531	13	—
MPI 8	3516	4329	5213	66	—

単精度の L サイズ，倍精度の M, L サイズの一部で極端な性能低下が見られる理由は，一部の記憶領域をディスクで代替したためと考えられます．MPI プログラムでは各 CPU が記憶領域を分割して保持するため，例えば L サイズの単精度プログラムでは 8 並列において実行可能です．

2.3.2 ShimaBMT

表 11 に ShimaBMT の測定結果を示します．zx2000 は分散並列型計算機のため，自動並列化はサポートされていません．

表 11: zx2000: ShimaBMT

N	100	200	500	2000	5000	9000	13000
逐次	765	652	658	663	—	—	—

2.4 IBM eServer pSeries 615

情報基盤センターの保有する POWER4+の共有メモリ (2CPU) 型計算機 eServer pSeries 615 を用いて性能を測定しました．仕様は以下の通りです．

プロセッサ	POWER4+ 1.45GHz; 2CPU
主記憶容量	2GB/CPU
1 次キャッシュ	96KB
2 次キャッシュ	1.5MB
3 次キャッシュ	8MB
OS	AIX 5L Version 5.2
コンパイラ	XL Fortran for AIX version 8.1
最適化オプション	`-O4`

2.4.1 HimenoBMT

表 12 に HimenoBMT の測定結果を示します．

表 12: pSeries 615: HimenoBMT

単精度

サイズ	XS	S	M	L	XL
逐次	373	229	231	—	—
自動並列化 1	375	229	233	—	—
自動並列化 2	377	231	234	—	—
OpenMP 1	542	298	300	—	—
OpenMP 2	1044	548	542	—	—

倍精度

	XS	S	M	L	XL
逐次	239	153	165	—	—
自動並列化 1	237	153	164	—	—
自動並列化 2	240	155	164	—	—
OpenMP 1	284	181	206	—	—
OpenMP 2	543	316	343	—	—

pSeries 615 は 4 命令/クロックですので ,理論ピーク性能は 4×1.45 GFLOPS $\approx$ 5800MFLOPS です . この値に比べると ,(このプログラムに関しては) 十分な性能が出ているとはいえません .

2.4.2 ShimaBMT

表 13 に ShimaBMT の測定結果を示します .

表 13: pSeries 615: ShimaBMT

N	100	200	500	2000	5000	9000	13000
逐次	1007	927	642	636	—	—	—
自動並列化 1	740	600	225	81	—	—	—
自動並列化 2	1418	1134	308	108	—	—	—

サイズが大きくなるにしたがって , 数値が低くなっています .

2.5 Hewlett Packard hpserver rx2600

Hewlett Packard 社の好意により ,Itanium2 の (現時点での)最新クロックの計算機 hpserver rx2600 を利用する機会がありました . 仕様は以下の通りです .

プロセッサ	Itanium2 1.5GHz; 1CPU
主記憶容量	8GB
1 次キャッシュ	32KB
2 次キャッシュ	256KB
3 次キャッシュ	6MB
OS	HP-UX 11.23
コンパイラ	Fortran: HP Fortran90 コンパイラ バージョン 2.7
最適化オプション	`+Ofast +O3 +DD64`

2.5.1 HimenoBMT

表 14 に HimenoBMT の測定結果を示します .

rx2600 は 4 命令/クロックですので , 理論ピーク性能は 4×1.5 GFLOPS $\approx$ 6000MFLOPS です . XS サイズにおいては , ピーク性能の約半分の性能を達成しています . この数値は VPP5000 を上回るものです . しかしながら , サイズが大きくなると数値が低くなります .

表 14: rx2600: HimenoBMT

	XS	S	M	L	XL
単精度	2900	1812	1915	1671	—
倍精度	1408	1035	1024	823	—

2.5.2 ShimaBMT

表 15 に ShimaBMT の測定結果を示します．

表 15: rx2600: ShimaBMT

N	100	200	500	2000	5000	9000	13000
逐次	2214	2260	1552	1551	1509	—	—

こちらも理論ピーク性能の 37% ~ 25%の値を達成しています．

3 考察

この節では，前節の測定データより得られた知見をいくつか報告します．

3.1 単精度と倍精度

図 1 に HimenoBMT の S サイズにおける単精度版と倍精度版プログラムの性能比を示します．ベクトル計算機 (性能比 88%) に比べ，他の計算機では性能の劣化 (57% ~ 78%) が目立

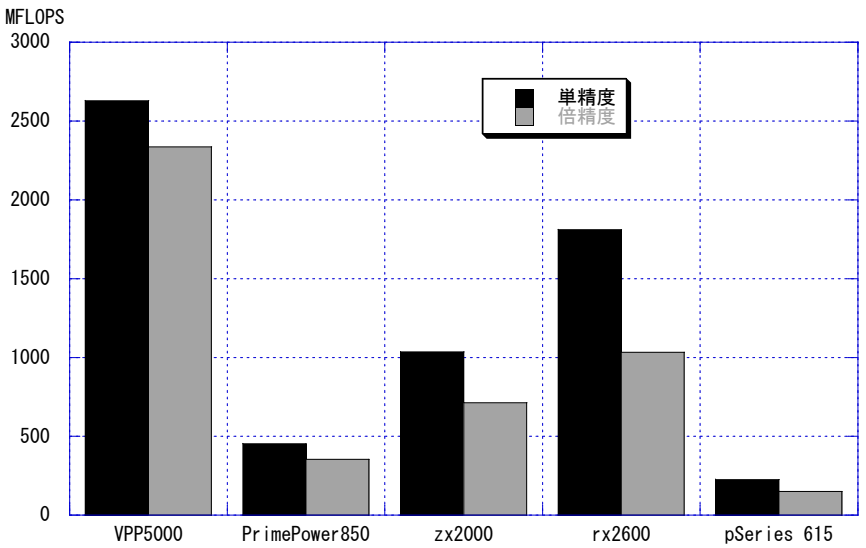

図 1: 単精度と倍精度の比較 (HimenoBMT; S サイズ)

ちます．原因としては，キャッシュのヒット率の差や，記憶容量が倍になることによるロード / ストア数の違いなどが考えられます．

3.2 単体性能

図 2 に HimenoBMT の単精度版の各サイズにおける性能比を，図 3 に ShimaBMT の各サイズにおける性能比をそれぞれ示します．ベクトル計算機では，サイズが大きくなるほどベ

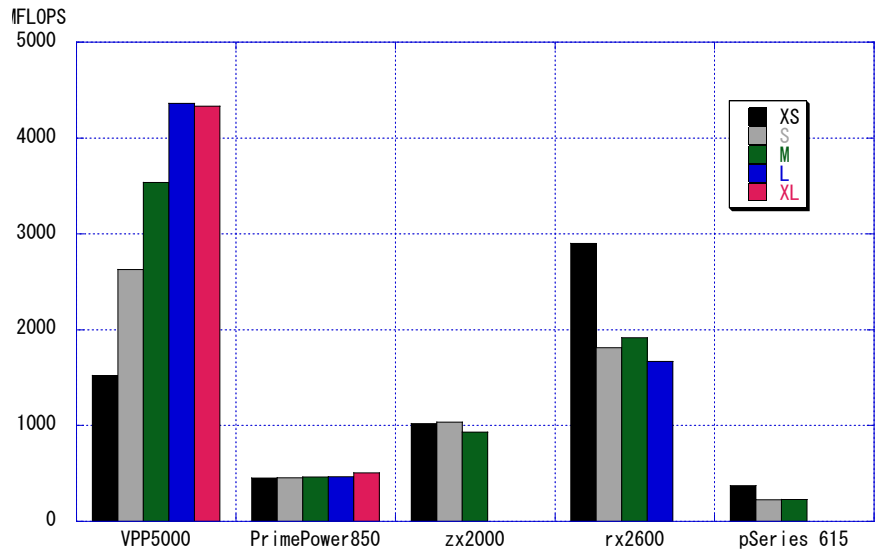

図 2: 単体性能 (HimenoBMT; 単精度)

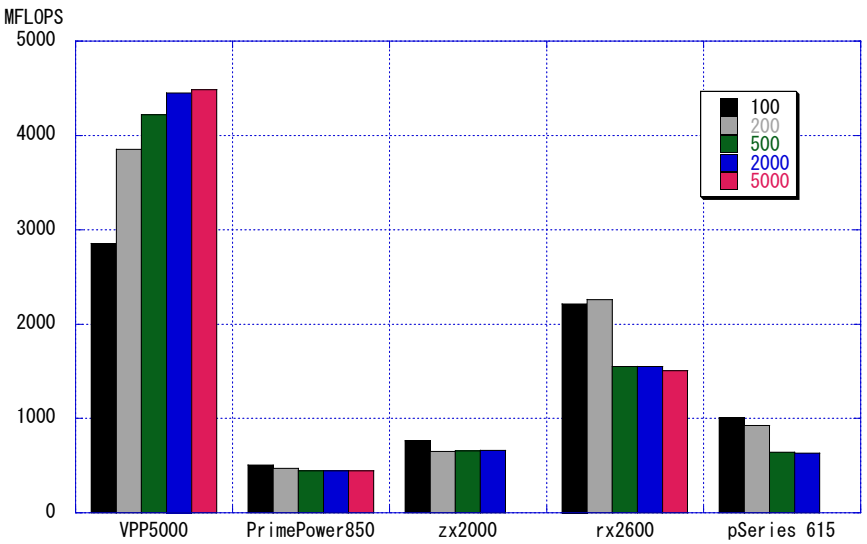

図 3: 単体性能 (ShimaBMT)

クトル長の長い計算が可能となり，性能が向上することが確認できました．一方，XS サイズの計算では Itanium2 プロセッサがベクトル計算機以上の性能を達成しており，小さな計算サイズにおけるベクトル計算機の利点は少なくなりつつあります．

また，スカラ計算機では，キャッシュを活用することにより性能が大きく変化します．今回の測定は，標準的な最適化オプションを用いました．しかし，2.2.1 節でも触れたように，

プログラムにあわせた最適化オプションを指定することにより，更に高い数値を達成できる可能性もあると思われます．

3.3 MPI による並列化性能 (HimenoBMT)

VPP5000，PRIMEPOWER850，zx2000 における単精度版 MPI プログラムの並列化性能を，図 4，図 5，図 6 にそれぞれ示します．ベクトル計算機では，サイズが大きくなるにつ

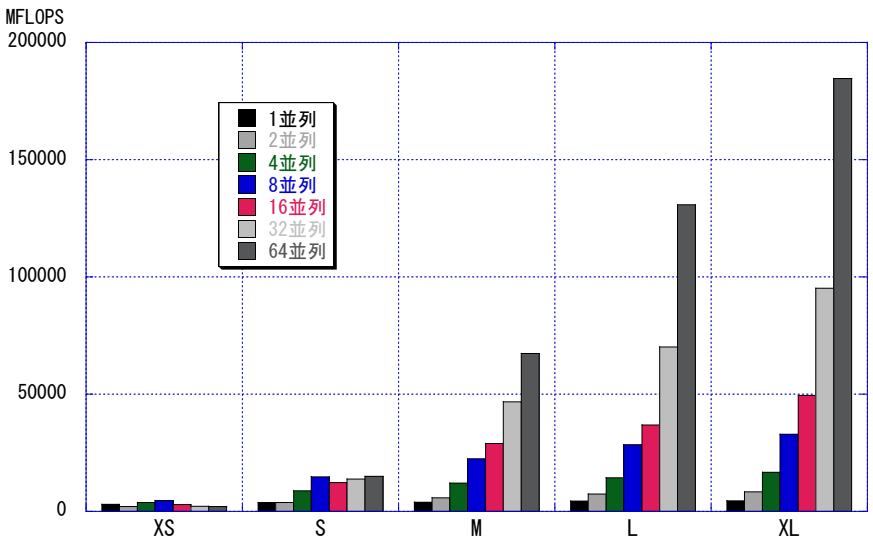

図 4: MPI の並列化性能 (VPP5000; HimenoBMT; 単精度)

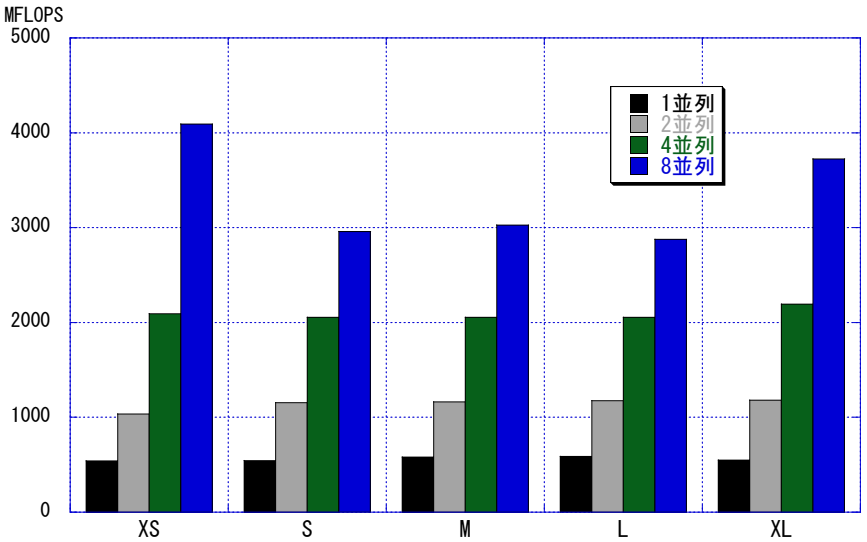

図 5: MPI の並列化性能 (PRIMEPOWER; HimenoBMT; 単精度)

れ，良好な並列化性能を示します．逆に，小さいサイズではベクトル長が短くなることにより，並列化によって逆に性能が低下します．

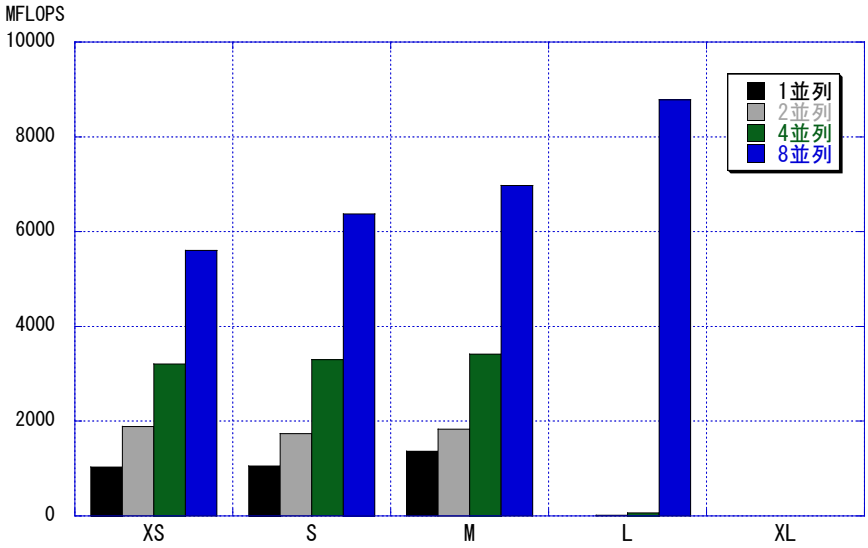

図 6: MPI の並列化性能 (zx2000; HimenoBMT; 単精度)

PRIMEPOWER850(SMP 計算機) , zx2000(PC クラスタ) では , 8 並列までについては , サイズによらずほぼ同じ並列化性能を示しました .

3.4 OpenMP による並列化性能 (HimenoBMT)

PRIMEPOWER850 , pSeries 615 における単精度版 OpenMP プログラムの並列化性能を , 図 7 , 図 8 にそれぞれ示します . 理論ピーク性能との差はともかく , どちらとも良好な並列

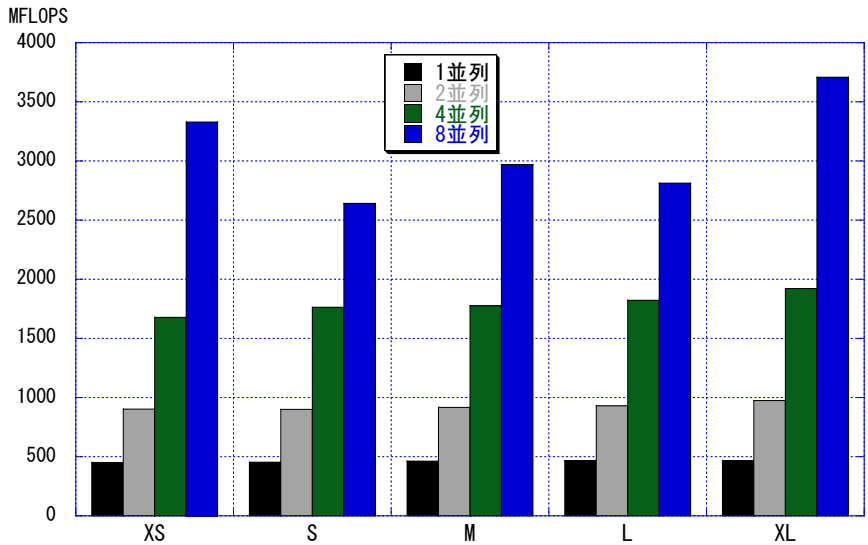

図 7: OpenMP の並列化性能 (PRIMEPOWER; HimenoBMT; 単精度)

化効果が得られています .

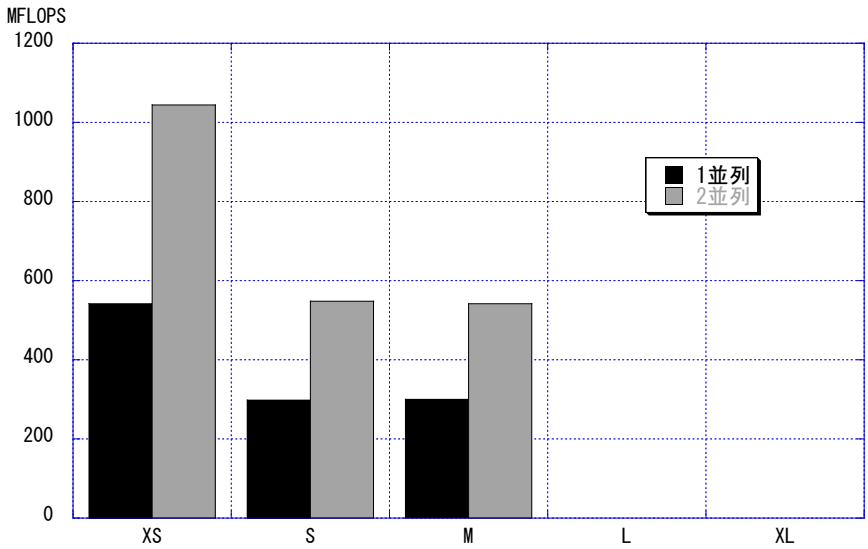

図 8: OpenMP の並列化性能 (pSeries 615; HimenoBMT; 単精度)

3.5 自動並列化性能

図 9 に，PRIMEPOWER850 における HimenoBMT の単精度版プログラムの M サイズの自動並列化性能を示します．自動並列化機能により，MPI や OpenMP で陽に並列化するの

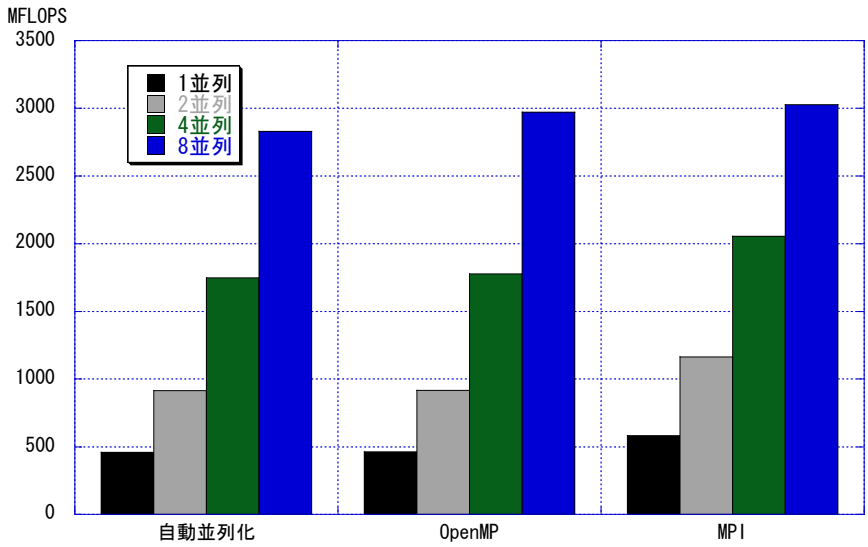

図 9: 自動並列化 (PRIMEPOWER; HimenoBMT; 単精度; M サイズ)

と同程度の並列化効率が達成されています．

図 10 に，PRIMEPOWER850 における ShimaBMT のいくつかのサイズにおける自動並列化性能を示します．

サイズが小さい場合にはほとんど自動並列化が効いていません．しかし，サイズが大きくなるに従い，効果が現れることがわかります．

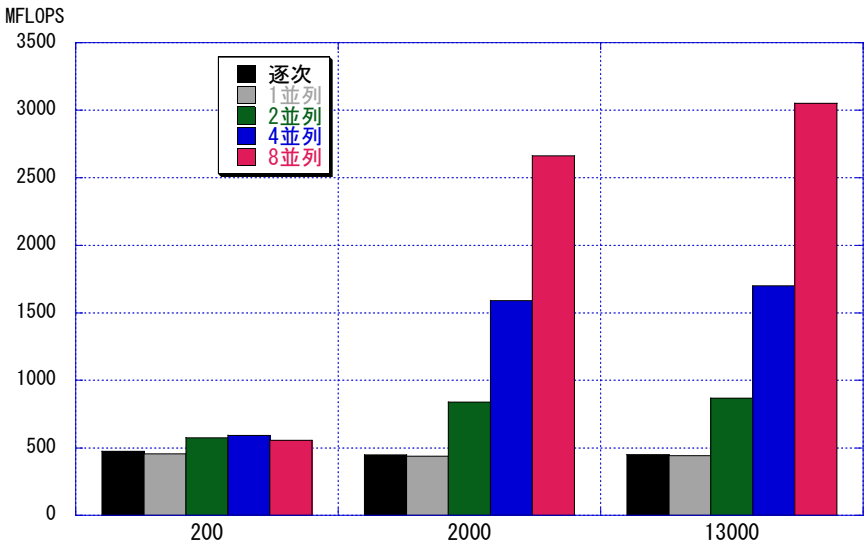

図 10: 自動並列化 (PRIMEPOWER; ShimaBMT)

図 11 に，pSeries 615 における ShimaBMT のいくつかのサイズにおける自動並列化性能を示します．

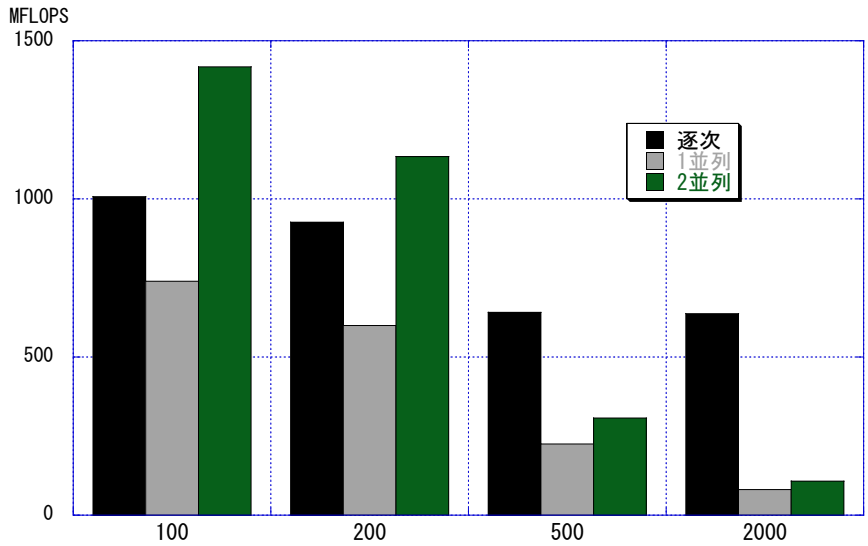

図 11: 自動並列化 (pSeries 615; ShimaBMT)

こちらは，PRIMEPOWER の自動並列化コンパイラのような結果とはならず，自動並列の 1 スレッドの性能が，逐次版に比べ悪くなり，そこから並列化効果が出るという結果になっています．これらの数値は，コンパイルオプションやレベルアップによって今後改善されるのではないかと期待しています．

4　おわりに

本稿では，HimenoBMT および ShimaBMT を用いて測定した共有メモリ型並列計算機，PC クラスタ，ベクトル並列計算機の性能評価結果を紹介しました．

HimenoBMT, ShimaBMT とも PRIMEPOWER850 においては自動並列化による性能向上が得られることから，これらのプログラムに関しては，並列プログラムを記述する必要がない共有メモリ型並列計算機が最適ではないかと思われます．

HimenoBMT, ShimaBMT とも，ベクトル計算機 VPP5000 においてはサイズが大きくなるほど性能が向上するものの，小さな計算サイズにおいては Itanium2 の計算機がベクトル計算機の性能を超える場合があることを確認しました．

今回の数値実験を通して，計算機性能においては，コンパイラと計算機構造とプログラミングの 3 つの技術が複雑に絡んでいることを再認識しました．

したがって，汎用的と思われる高速な数値計算アルゴリズムを提案する際には，計算サイズ，並列化を行なった場合には並列度との関係を複数の計算機環境で試み，それらの結果を開示することにより，より提案手法の信頼性が増すのではと思われます．

今後は，より多くの応用プログラムを用いた並列化効率の測定，疎行列を圧縮した形式としてよく用いられるリストベクトルに対するアクセスおよび演算性能，Gaussian03 などのより具体的なアプリケーションプログラムを用いた性能評価などを行なう予定です．

謝辞

ベンチマークプログラムを提供していただいた，理化学研究所 姫野 龍太郎 氏，川崎重工業 嶋 英志 氏，および HimenoBMT の VPP5000 における実行データの提供をいただきました筑波大学学術情報処理センター (電子・情報工学系 数値解析研究室) の伊藤 祥司様に心より感謝いたします．

参考文献

[1] 渡部 善隆, 南里 豪志, 藤野 清次: Himeno BMT によるハイパフォーマンスコンピュータの性能評価, 情報処理学会研究報告 2003-HPC-95 (2003) pp.137-142.
http://www.cc.kyushu-u.ac.jp/RD/watanabe/RESERCH/MANUSCRIPT/PAPER/SWoPP2003/index.html
よりダウンロード可能．