

SS研科学技術計算分科会

科学技術計算分科会 2003年度 第2回会合

3次元MHDコードによるPRIMEPOWER HPC2500の性能評価
：名古屋大学における事例報告

名古屋大学太陽地球環境研究所

荻野竜樹

名古屋大学情報連携基盤センター

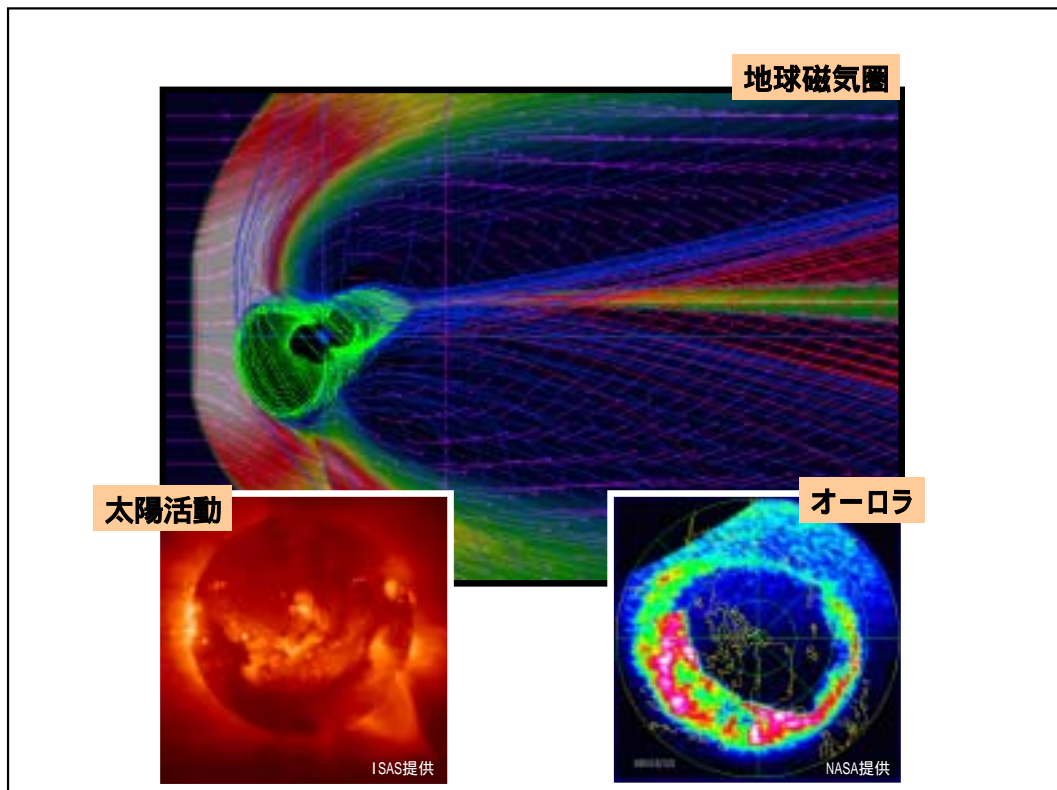
津田知子

アウトライン

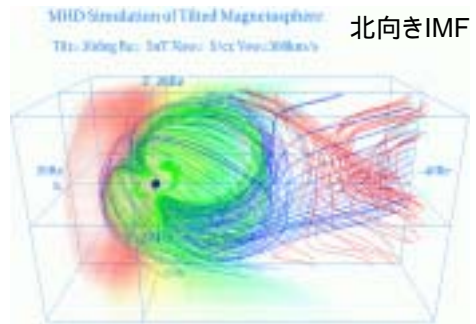
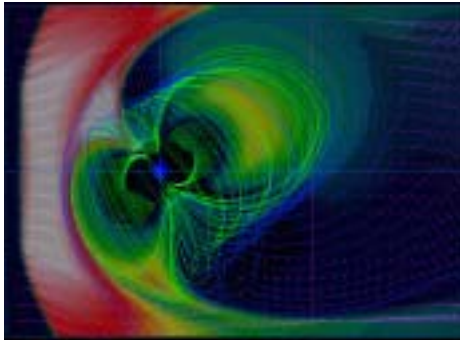
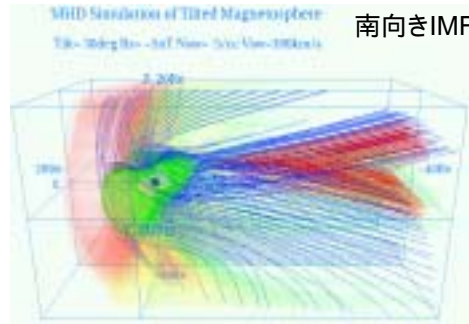
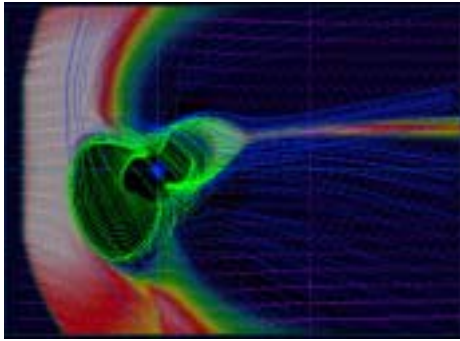
- 1．背景
- 2．地球磁気圏の3次元グローバルMHDモデル
- 3．VPPIにおける並列化
- 4．PRIMEPOWER HPC2500における並列化
- 5．HPC2500とVPPIにおける3次元MHDコード計算速度
- 6．計算速度の比較の考察と問題点
- 7．地球磁気圏の3次元グローバルMHDシミュレーションの将来
- 8．まとめ

1. 背景

- ・ 太陽活動とジオスペース（地球外圏）環境変化の因果関係を解明する研究
- ・ 外部境界をできるだけ遠くに設定してかつ空間分解能をできるだけ上げる
- ・ 計算方法の改良と最大級のスーパーコンピュータの効率的な利用
- ・ 並列計算共通プログラム言語 / 手法
 - High Performance Fortran (HPF)
 - Message Passing Interface (MPI)
- ・ ベクトル並列機からスカラー並列機への移行
 - 富士通株式会社：VPP5000からPRIMEPOWER HPC2500へ

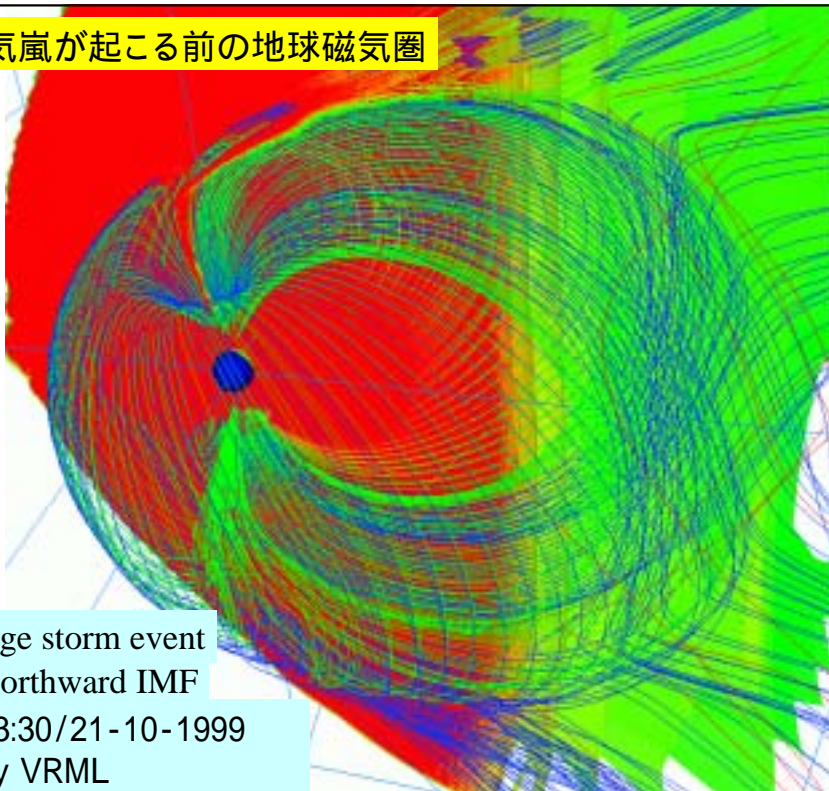


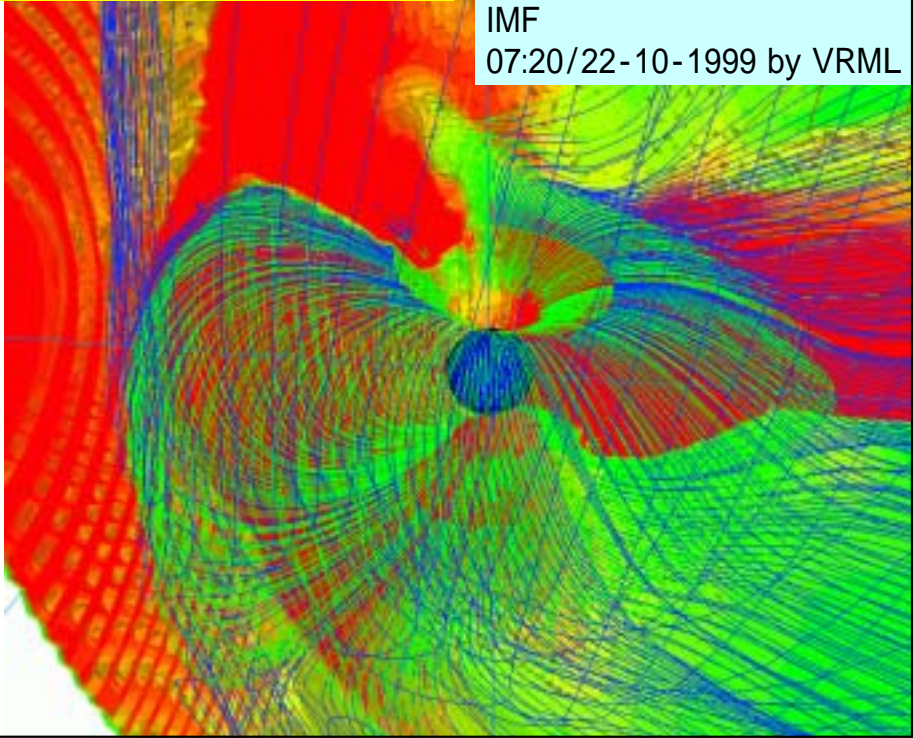
MHD Simulation of the Earth's Magnetosphere with Dipole Tilt for IMF Bz Component



磁気嵐が起こる前の地球磁気圏

Large storm event
Northward IMF
23:30/21-10-1999
by VRML





2 . 地球磁気圏の3次元グローバルMHDモデル

- ・ 太陽風磁気圏相互作用の3次元グローバルMHDモデル
- ・ MHD方程式とマクスウェル方程式をmodified leap-frog法を用いて、初期値境界値問題として時間発展を解く
- ・ 並列計算では、z方向に1次元の領域分割
- ・ z方向のdo loopは最大限に大きくとり、その出現回数を最少化
- ・ z方向のデータ転送は必要最小限の2カ所にコンパクトにまとめて、一括大量転送を実施

基礎方程式

- ・ MHDモデルの基礎となる規格されたMHD方程式とMaxwell方程式

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\mathbf{v}\rho) + D\nabla^2 \rho \quad (1)$$

$$\frac{\partial \mathbf{v}}{\partial t} = -(\mathbf{v} \cdot \nabla)\mathbf{v} - \frac{1}{\rho}\nabla p + \frac{1}{\rho}\mathbf{J} \times \mathbf{B} + \mathbf{g} + \frac{1}{\rho}\Phi \quad (2)$$

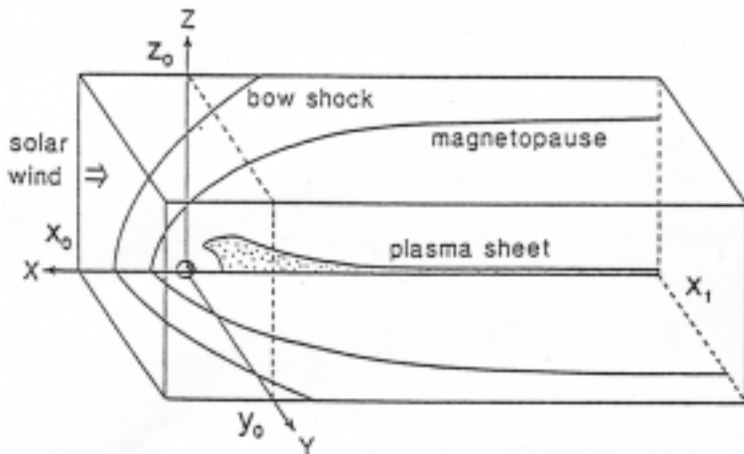
$$\frac{\partial p}{\partial t} = -(\mathbf{v} \cdot \nabla)p - \gamma p \nabla \cdot \mathbf{v} + D_p \nabla^2 p \quad (3)$$

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{v} \times \mathbf{B}) + \eta \nabla^2 \mathbf{B} \quad (4)$$

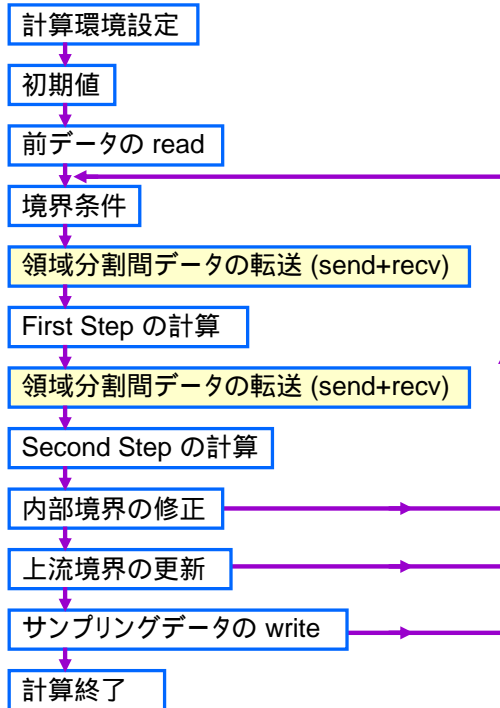
$$\mathbf{J} = \nabla \times (\mathbf{B} - \mathbf{B}_d) \quad (5)$$

座標系と境界条件

3次元MHDシミュレーションに用いる太陽地球磁気圏座標系



3 . MHDコードにおけるHPF/JA



3 . VPPにおける並列化

- VP200やCRAY-1などでフルベクトル化されていた3次元MHDコード
- VPP Fortranを用いて、VPP500用にフルベクトル化フル並列化された3次元MHDコードを作成 (1997)
- HPF/JAを用いて、VPP5000用にフルベクトル化フル並列化された3次元MHDコードを作成 (2000)
- VPP FortranとHPF/JAからMPIへ書き換えて、フルベクトル化フル並列化された3次元MHDコードを作成 (2002)
- 3次元MHDコードの基本構造は、MPIとHPF/JAとVPP Fortranで同じ

4 . PRIMEPOWER HPC2500における並列化

- ・ VPP Fortranで使用したMPIの3次元MHDコードと基本的に同じ
- ・ 配列の順序は、 $f(nx2,ny2,nz2,nb)$ のように、x方向、y方向、z方向、8個のMHD方程式の成分の順序で、z方向に1次元の領域分割
- ・ 高速化のために $f(nx2,ny2,nz2,nb)$ から $F(nb,nx2,ny2,nz2)$ への配列の順序の並び替えを宣言
- ・ 領域分割の両端のデータをcpu間で通信には、wild cardと`mpi_sendrecv`の利用及び通信データのコンパクト化を実行
- ・ コンパイルオプションでは、キャッシュのレベルの最適化、`largepage`の利用、64bit対応用のオプションV9のオフを採用

5 . HPC2500とVPPにおける3次元MHDコード計算速度

- ・ VPP5000/64では、VPP FortranとHPF/JAとMPIのいずれでも同等の計算速度がでていて、スケーラビリティもかなり良い
- ・ PRIMEPOWER HPC2500では、32cpuまではかなりよいスケーラビリティが得られた
- ・ HPC2500 (ngrid)の8cpuで8.36 Gflopsの計算速度を得たように、VPP5000の1cpuに対してHPC2500の7cpuで同等の速度を得ることができた。
- ・ 33cpu以上で計算速度が急に遅くなったが、HPC2500 (ngrid)において33cpu以上では高速ネットワークが働かなくなるからである
- ・ 領域分割間の通信停止の場合の計算速度を測定すると48cpuまで確実によいスケーラビリティが得られている

<< 名古屋大学情報連携基盤センターのPRIMEPOWER HPC2500 (grid) の性能 >>

【CPU】

32CPU構成 × 2筐体 = 計64CPU

(このうち、Batchには24CPU × 2を使用、あとは、TSSとGrid用)

- ・ SPARC64 V(1.3GHz)
- ・ 内部キャッシュ(1次キャッシュ:命令128KB+データ128KB、2次キャッシュ:2MB)
- ・ 理論計算速度: 5.2 GFlops/CPU, 332.8 GFlops/TOTAL

【メモリ】

64GB × 2筐体 = 計128GB

【高速インターコネクト】

ノード間通信性能: 4GB/s × 2 = 8GB/s (理論性能値)

(A) 加減乗除の四則演算の実測 (Mflops)

1. F(I)=FLOAT(I)
2. P(I)=F(I)
3. U(I)=F(I)+P(I)
4. U(I)=F(I)-P(I)
5. U(I)=F(I)*P(I)
6. U(I)=F(I)/P(I)

Computer	1	2	3	4	5	6
PRIMEPOWER HPC2500 (Mflops)	190	465	366	353	366	94
VPP5000 (scalar)	42	291	210	212	217	61
VPP5000 (vector)	2327	8533	3200	3200	3200	2844
HPC P4Linux	384	889	628	630	610	51

(B) 四則演算の速度の比較

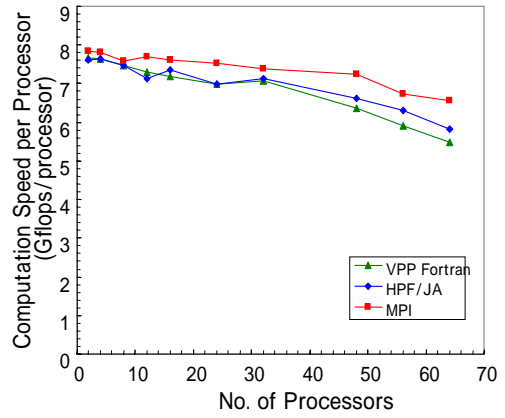
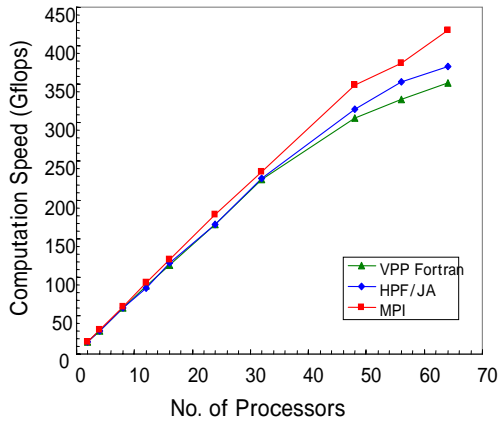
computer	compiler option	processing capability (MFLOPS)
Fujitsu VPP-500 (1PE)	frtpx, -sc	19.8
Fujitsu VPP-500 (1PE)	frtpx	730.3
Fujitsu VPP-5000 (1PE)	frt, -sc	189.8 (1999.12.27)
Fujitsu VPP-5000 (1PE)	frt	3,073.7 (1999.12.27)
SunFireV800 (sv080)	f90 -0	138.1 (2002.08.28)
GP7000Fモデル900(gpcs)	frt -03	95.8 (2003.04.08)
HPC2500 (ngrd)	frt -03	273.3 (2003.04.08)
DEC Alpha DS-20(833MHz)	f90 -fast	246.9 (2002.11.08)
SONY VAIO Pentium4	lf95	547.6 (2002.09.03)
HPC P4Linux	ifc	478.7 (2003.02.06)

(C) MHDコードを用いての計算速度の比較

computer	compiler	(a)3D-MHD	(b)3D-MHD	(c)2D-MHD
		50x50x26 sec (MFLOPS)	62x32x32 sec (MFLOPS)	402x102 sec (MFLOPS)
Fujitsu VPP-5000(1PE)	frt, -sc	0.717 (169)	0.694 (168)	0.1238 (225)
Fujitsu VPP-5000(1PE)	frt	0.0233 (5201)	0.0227 (5110)	0.0044 (6316)
SunFireV800 (sv080)	f90 -0	0.305 (397)	0.336 (348)	0.0781 (355)
GP7000Fモデル900(gpcs)	frt -03	0.2515 (481)	0.2905 (403)	0.0562 (493)
HPC2500 (ngrd)	frt -03	0.0938 (1291)	0.0938 (1246)	0.0234 (1184)
DEC Alpha (833MHz)	f90 -fast	0.1016 (1186)	0.0859 (1355)	0.0234 (1155)
SONY VAIO Pentium4	lf95	0.203 (596)	0.312 (375)	0.0430 (645)
HPC P4Linux	ifc	0.0906 (1335)	0.0943 (1241)	0.0240 (1172)

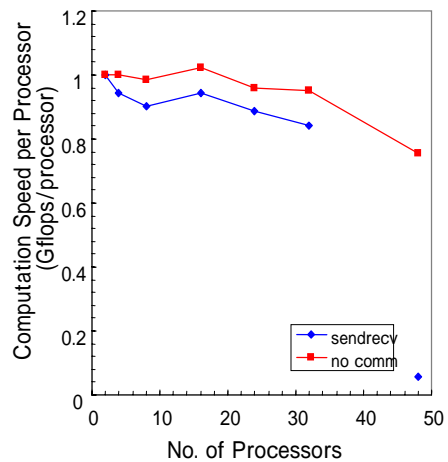
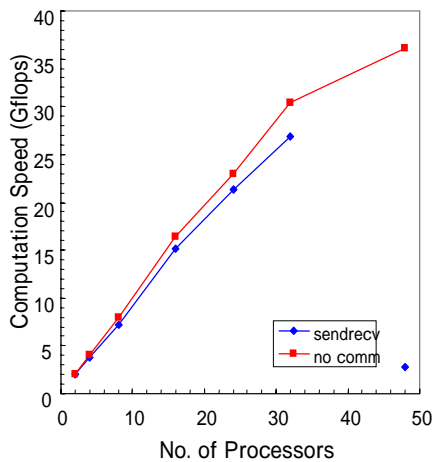
Computer Processing Capability in the MPI, HPF/JA and VPP Fortran by Using VPP

(nx,ny,nz)=(800,200,670)



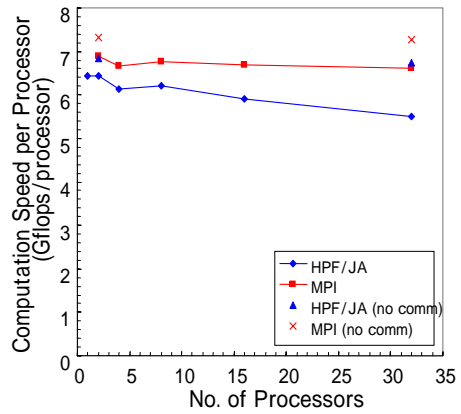
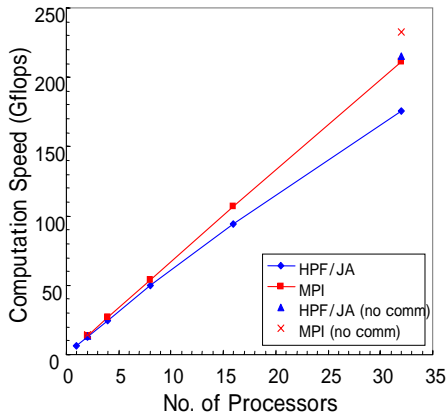
Computer Processing Capability in the MPI by Using HPC2500

(nx,ny,nz)=(250,158,158)



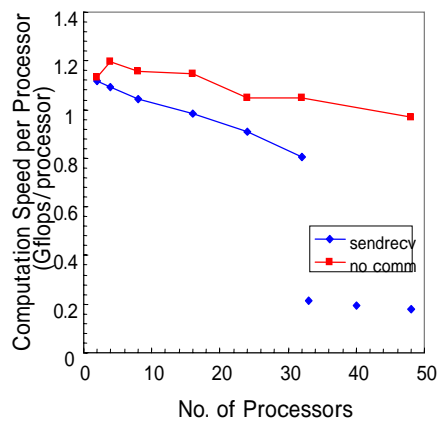
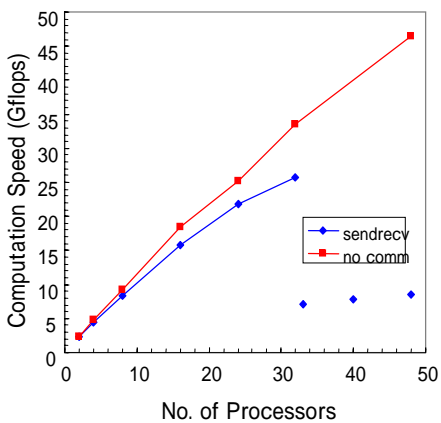
Computer Processing Capability in the MPI and HPF/JA by Using VPP

(nx,ny,nz)=(500,318,318)



Computer Processing Capability in the MPI by Using HPC2500

(nx,ny,nz)=(500,318,318)



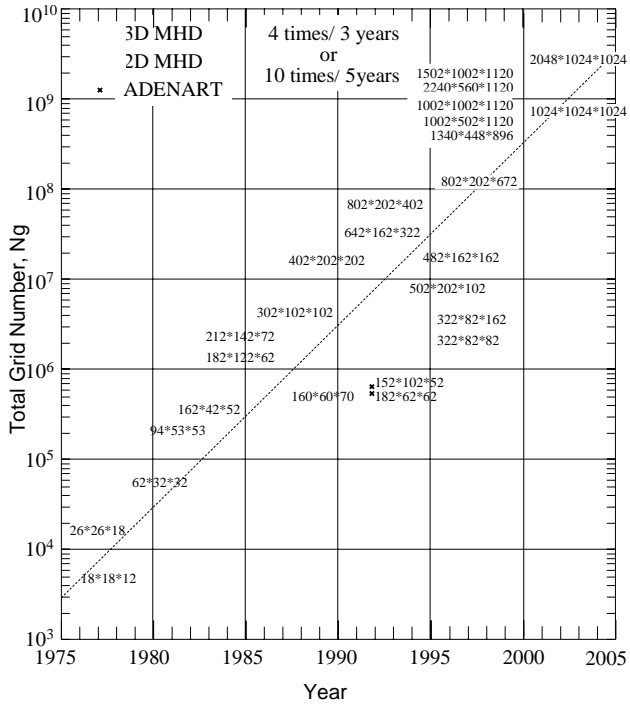
6 . 計算速度の比較の考察と問題点

- ・ PRIMEPOWER HPC2500で、33cpu以上では並列計算の顕著な劣化がみられたが、33cpu以上ではデータ転送に高速光インターコネクト（4GB/sec）を用いずに、GbEを使用しているため(-ldt 0)。
- ・ 24cpu使用時のコンパイルオプションの違いによるハードバリアとソフトバリア利用の計算速度は、それぞれ21.82 Gflopsと20.27 Gflopsであり、その差は約7.1%。
- ・ ベクトル並列機VPP5000からスカラー並列機PRIMEPOWER HPC2500への移行で、高効率の並列計算が保証されるのか？
- ・ ベクトル並列機では最内のdoloopでベクトル化して、最外のdo loopで並列化するのが基本。計算効率を上げるためには最内のdo loopは長い方が望ましい。
- ・ スカラー並列機はキャッシュのヒットが高速計算に不可欠。1次キャッシュと2次キャッシュのヒット、largepageの機能で高速計算を保証。従って、計算に関連する変数の番地はあまり離れていないことが望ましい。

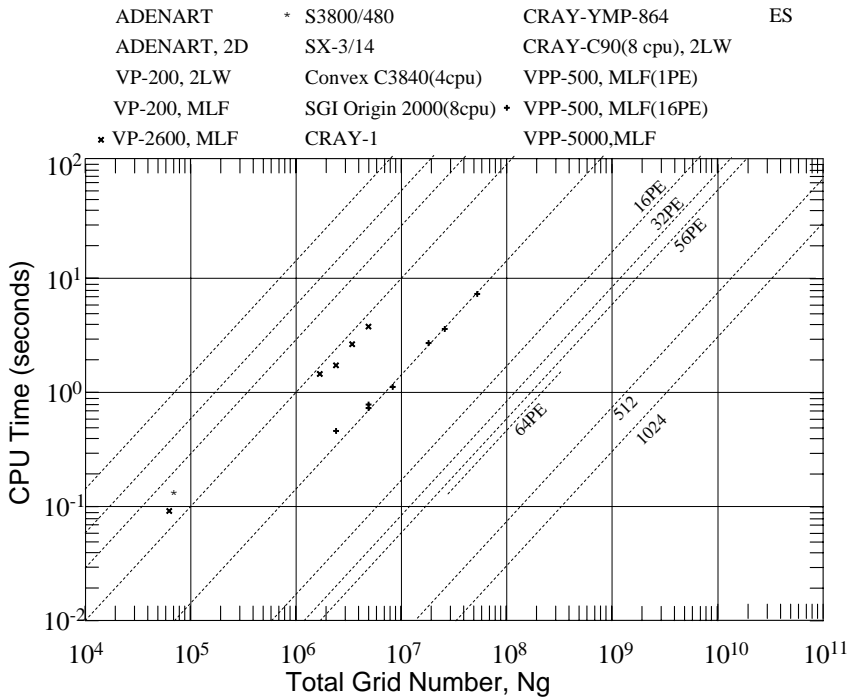
7 . 地球磁気圏の3次元グローバルMHDシミュレーションの将来

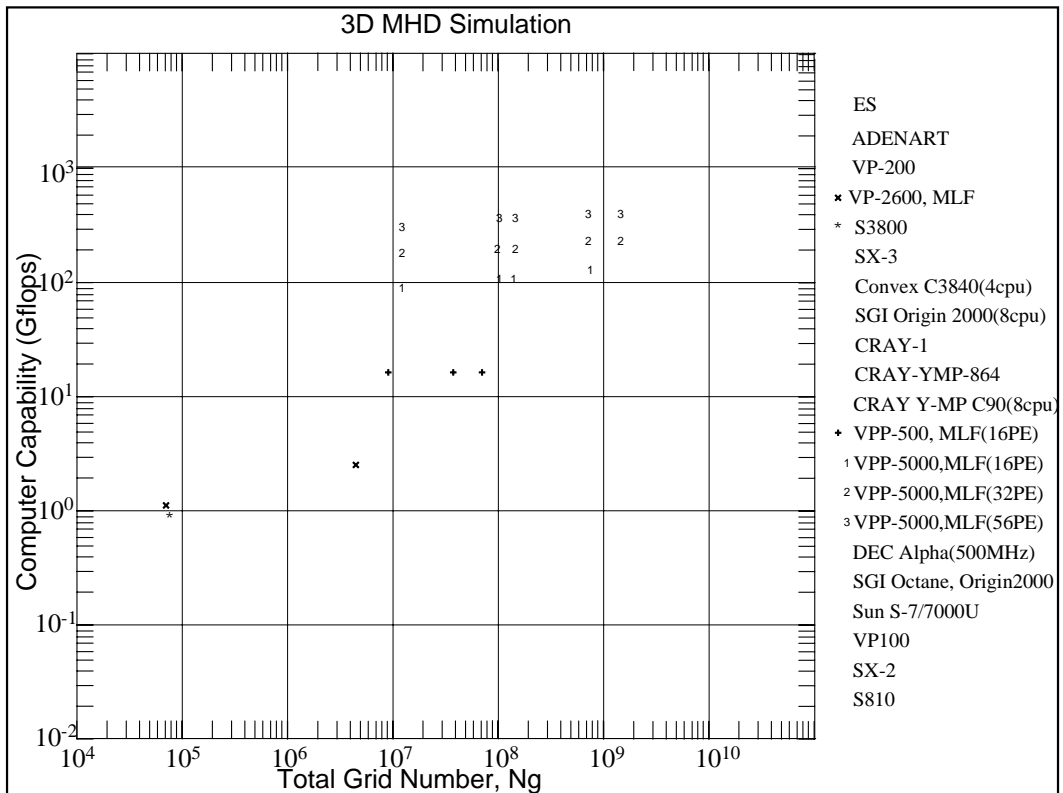
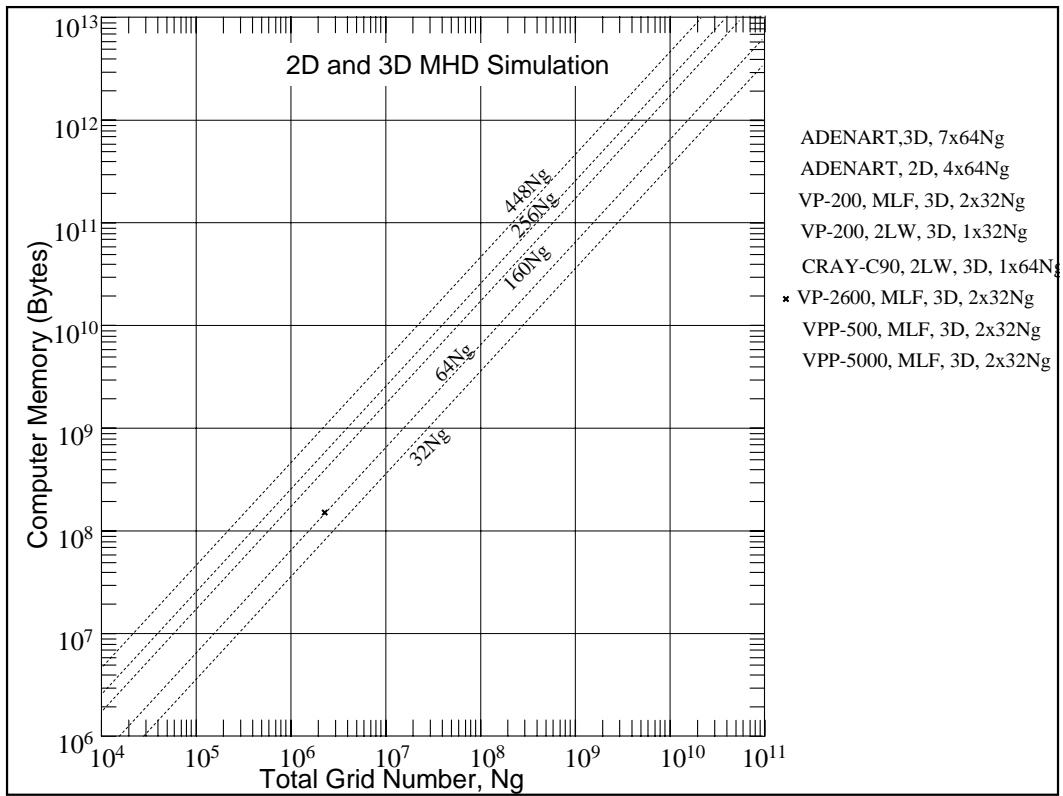
- ・ 利用できる総格子点数からみたMHDシミュレーションの年毎の発展拡大の傾向をみると、この25年間で平均して3年で4倍あるいは5年で10倍の割合で増大している。
- ・ 最近の高性能のワークステーションやパーソナルコンピュータの計算速度の伸びは著しいが、最大級のスーパーコンピュータは常にそれらの約千倍の高速計算処理能力を有している。
- ・ 多数の3次元格子点を用いた大容量シミュレーションでは、メモリ不足やRISC計算機でのcache不足などによって、更に決定的な差が生じる。
- ・ フルベクトル化/フル並列化されたプログラムを用いて、VPP5000等のスーパーコンピュータで100時間かかる3次元MHDシミュレーションは、スーパーコンピュータ以外の通常のワークステーションやパーソナルコンピュータで実行することは全く不可能である。
- ・ 地球磁気圏の3次元グローバルMHDシミュレーションには、最大級のスーパーコンピュータの利用は常に必須の条件と言える。

2D and 3D MHD Simulation



2D and 3D MHD Simulation



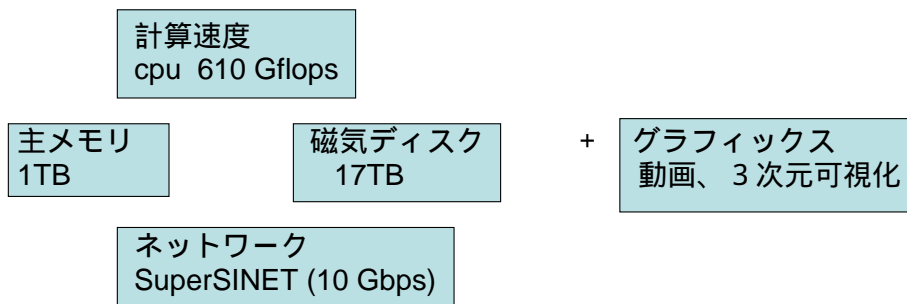


8 . まとめ

- ・ベクトル並列機VPP5000から、クラスター型スカラー並列機PRIMEPOWER HPC2500への移行
- ・地球磁気圏の3次元MHDシミュレーションコードを用いたPRIMEPOWER HPC2500の性能評価
 - 32cpuまでの利用ではスケーラビリティはかなりよい
 - グリッド数 (nx,ny,nz) =(500,318,318)を用いて、2cpuで21%(=0.116/5.2)の実行効率
 - 8cpu で8.4 Gflops、32cpuで25.8 Gflopsの計算速度を実現
- ・今回はスレッド並列は用いずに、MPIによるプロセス並列ジョブの測定のみ実行
- ・ノード内自動並列、ノード間利用者による並列化で高効率計算を実現するのは、非常に困難な問題である。
- ・PRIMEPOWER HPC2500は、設計思想としてスレッド並列（自動並列、あるいはOpenMPを使用したノード内並列）とプロセス並列（MPI等を使用したノード間並列）との併用による並列ジョブの実行を想定していて、最大性能を引き出すにあたってはその構成が望ましい。
- ・PRIMEPOWER HPC2500で更にcpu数を増やしてTflopsの実行速度が実現すること、ベクトル並列機に負けない計算速度を実現することを強く期待する。

8 . まとめ - 2

スーパーコンピューティングの4つの機能



近隣のアジア諸国の研究者・学生に日本の最大級のスーパーコンピュータを利用してもらう機会を提供できないか。

韓国、台湾、中国、インドなど

表1 名古屋大学情報連携基盤センターの富士通VPP5000/64を用いて、VPP Fortran、HPF/JA及びMPIで書かれた太陽風と地球磁気圏相互作用の3次元グローバル電磁流体力学的(MHD)コードで調べた場合の計算処理速度の比較。

Table 1. Comparison of computer processing capability between VPP Fortran, HPF/JA and MPI in a 3-dimensional global MHD code of the solar wind-magnetosphere interaction by using Fujitsu VPP5000/64.

Number of PE	Number of grids	VPP Fortran		HPF/JA		MPI	
		cpu time	Gflops Gf/PE	cpu time	Gflops Gf/PE	cpu time	Gflops Gf/PE
1PE	200x100x478	119.607 (0.17)	0.17 (scalar)				
1PE	200x100x478	2.967 (6.88)	6.88	3.002 (6.80)	6.80		
2PE	200x100x478	1.458 (14.01)	7.00	1.535 (13.30)	6.65	1.444 (14.14)	7.07
4PE	200x100x478	0.721 (28.32)	7.08	0.761 (26.85)	6.71	0.714 (28.60)	7.15
8PE	200x100x478	0.365 (55.89)	6.99	0.386 (52.92)	6.62	0.361 (56.55)	7.07
16PE	200x100x478	0.205 (99.38)	6.21	0.219 (93.39)	5.84	0.191 (107.19)	6.70
24PE	200x100x478	0.141 (144.49)	6.02	0.143 (143.02)	5.96	0.1302(157.24)	6.55
32PE	200x100x478	0.107 (191.23)	5.98	0.110 (186.13)	5.82	0.1011(202.50)	6.33
48PE	200x100x478	0.069 (297.96)	6.21	0.074 (276.96)	5.77	0.0679(301.51)	6.28
56PE	200x100x478	0.064 (319.53)	5.71	0.068 (299.27)	5.34	0.0639(320.39)	5.72
64PE	200x100x478	0.0662(308.91)	4.83	0.0627(324.57)	5.07	0.0569(359.80)	5.62

Number of PE	Number of grids	VPP Fortran		HPF/JA		MPI	
		cpu time	Gflops Gf/PE	cpu time	Gflops Gf/PE	cpu time	Gflops Gf/PE
1PE	500x100x200	2.691 (7.94)	7.94	2.691 (7.94)	7.94		
2PE	500x100x200	1.381 (15.47)	7.73	1.390 (15.37)	7.68	1.357 (15.74)	7.87
4PE	500x100x200	0.715 (29.97)	7.47	0.712 (29.99)	7.50	0.688 (31.03)	7.76
8PE	500x100x200	0.398 (53.65)	6.71	0.393 (54.38)	6.80	0.372 (57.50)	7.19
16PE	500x100x200	0.210 (101.87)	6.37	0.202 (105.74)	6.61	0.193 (110.70)	6.92
24PE	500x100x200	0.160 (133.70)	5.57	0.150 (142.40)	5.93	0.135 (158.26)	6.59
32PE	500x100x200	0.131 (163.55)	5.11	0.120 (175.50)	5.48	0.1084(197.10)	6.15
48PE	500x100x200	0.100 (214.48)	4.46	0.091 (231.69)	4.82	0.0811(263.44)	5.49
56PE	500x100x200	0.089 (239.48)	4.28	0.086 (244.85)	4.37	0.0688(310.54)	5.55
64PE	500x100x200	0.0956(222.95)	3.48	0.0844(249.49)	3.90	0.0687(310.99)	4.86
2PE	800x200x478	10.659 (15.33)	7.66	10.742 (15.21)	7.60	10.428 (15.67)	7.83
4PE	800x200x478	5.351 (30.53)	7.63	5.354 (30.52)	7.63	5.223 (31.28)	7.82
8PE	800x200x478	2.738 (59.67)	7.46	2.730 (59.85)	7.48	2.696 (60.61)	7.58
12PE	800x200x478	1.865 (87.58)	7.30	1.911 (85.49)	7.12	1.771 (92.25)	7.68
16PE	800x200x478	1.419 (115.12)	7.19	1.389 (117.66)	7.35	1.342 (121.81)	7.61
24PE	800x200x478	0.975 (167.54)	6.98	0.976 (167.45)	6.98	0.905 (180.59)	7.52
32PE	800x200x478	0.722 (226.33)	7.07	0.717 (227.72)	7.12	0.690 (236.63)	7.39
48PE	800x200x478	0.534 (305.70)	6.36	0.515 (317.26)	6.61	0.469 (348.38)	7.25
56PE	800x200x478	0.494 (330.95)	5.91	0.464 (352.49)	6.29	0.433 (377.73)	7.74
64PE	800x200x478	0.465 (351.59)	5.49	0.438 (373.41)	5.83	0.389 (420.45)	6.57

Number of PE	Number of grids	VPP Fortran			HPF/JA			MPI		
		cpu time	Gflops	Gf/PE	cpu time	Gflops	Gf/PE	cpu time	Gflops	Gf/PE
4PE	800x200x670	7.618	(30.06)	7.52	8.001	(28.62)	7.16	7.433	(30.81)	7.70
8PE	800x200x670	3.794	(60.36)	7.54	3.962	(57.81)	7.23	3.683	(62.17)	7.77
12PE	800x200x670	2.806	(81.61)	6.80	3.005	(76.21)	6.35	2.696	(84.95)	7.08
16PE	800x200x670	1.924	(119.00)	7.44	2.012	(113.85)	7.12	1.854	(123.53)	7.72
24PE	800x200x670	1.308	(175.10)	7.30	1.360	(168.44)	7.02	1.254	(182.61)	7.60
32PE	800x200x670	0.979	(233.85)	7.31	1.032	(221.88)	6.93	0.955	(239.77)	7.49
48PE	800x200x670	0.682	(335.62)	6.99	0.721	(317.80)	6.62	0.662	(346.21)	7.21
56PE	800x200x670	0.595	(384.61)	6.87	0.628	(364.87)	6.52	0.572	(400.59)	7.15
64PE	800x200x670							0.519	(441.50)	6.90
16PE	1000x500x1118	9.668	(123.52)	7.72	9.619	(125.50)	7.84			
32PE	1000x500x1118	5.044	(236.73)	7.40	4.992	(241.83)	7.56			
48PE	1000x500x1118	3.550	(336.40)	7.01	3.479	(346.97)	7.23			
56PE	1000x500x1118	2.985	(400.04)	7.14	2.935	(411.36)	7.35			
32PE	1000x1000x1118	9.979	(239.33)	7.48	9.813	(243.37)	7.61			
48PE	1000x1000x1118	7.177	(332.79)	6.93	7.028	(339.85)	7.08			
56PE	1000x1000x1118	5.817	(410.55)	7.33	5.794	(412.23)	7.36			

: Mflops is an estimated value in comparison with the computation by
1 processor of CRAY Y-MP C90.

表2 名古屋大学情報連携基盤センターの富士通HPC2500 (ngrd) とVPP5000/64を用いて、MPIとHPF/JAで書かれた太陽風と地球磁気圏相互作用の3次元グローバル電磁流体力学的 (MHD) コードで調べた場合の計算処理速度の比較。

Kind of Date computer	Number of PEs	HPF/JA reflect			MPI mpi_send and		
		cpu time (sec)	Gflops	Gf/PE	cpu time (sec)	Gflops	Gf/PE
mpi_recv							
VPP5000	1PE	16.7886	(6.44)	6.44			
VPP5000	2PE	8.3927	(12.87)	6.44	7.8305	(13.80)	6.90
VPP5000	4PE	4.4073	(24.52)	6.13	4.0589	(26.61)	6.65
VPP5000	8PE	2.1753	(49.67)	6.21	1.9973	(54.10)	6.76
VPP5000	16PE	1.1446	(94.40)	5.90	1.0103	(106.95)	6.68
VPP5000	32PE	0.6139	(176.01)	5.50	0.5116	(211.21)	6.60
mpi_irecv							
VPP5000	2PE	7.8334	(13.79)	6.90	7.8227	(13.81)	6.90
VPP5000	16PE	1.0109	(106.88)	6.68			
VPP5000	32PE	0.5127	(210.74)	6.59	0.5089	(212.31)	6.63
VPP5000	2PE	7.8898	(13.69)	6.85	7.3934	(14.61)	7.31
VPP5000	32PE	0.5016	(215.40)	6.73	0.4639	(232.90)	7.28

HPC2500 (ngrd)	2PE (250,158,158)	sendrecv -KV9	6.7873 (1.999)	0.999
HPC2500 (ngrd)	4PE (250,158,158)	sendrecv -KV9	3.5883 (3.780)	0.945
HPC2500 (ngrd)	8PE (250,158,158)	sendrecv -KV9	1.8780 (7.224)	0.903
HPC2500 (ngrd)	16PE (250,158,158)	sendrecv -Kfast	0.8991 (15.091)	0.943
HPC2500 (ngrd)	24PE (250,158,158)	sendrecv -Kfast	0.6378 (21.273)	0.886
HPC2500 (ngrd)	32PE (250,158,158)	sendrecv -Kfast	0.5036 (26.940)	0.841
HPC2500 (ngrd)	48PE (250,158,158)	sendrecv -Kfast	4.8827 (2.779)	0.058
HPC2500 (ngrd)	2PE (250,158,158)	no comm -KV9	6.7865 (1.999)	0.999
HPC2500 (ngrd)	4PE (250,158,158)	no comm -KV9	3.3942 (3.997)	0.999
HPC2500 (ngrd)	8PE (250,158,158)	no comm -KV9	1.7190 (7.892)	0.986
HPC2500 (ngrd)	16PE (250,158,158)	no comm -Kfast	0.8289 (16.366)	1.023
HPC2500 (ngrd)	24PE (250,158,158)	no comm -Kfast	0.5898 (23.001)	0.958
HPC2500 (ngrd)	32PE (250,158,158)	no comm -Kfast	0.4464 (30.387)	0.950
HPC2500 (ngrd)	48PE (250,158,158)	no comm -Kfast	0.3755 (36.125)	0.753

HPC2500 (ngrd)	2PE (500,318,318)	sendrecv -Cpp	48.4191 (2.231)	1.116
HPC2500 (ngrd)	4PE (500,318,318)	sendrecv -Cpp	24.7140 (4.372)	1.093
HPC2500 (ngrd)	8PE (500,318,318)	sendrecv -Cpp	12.9242 (8.360)	1.045
HPC2500 (ngrd)	16PE (500,318,318)	sendrecv -Cpp	6.8664 (15.736)	0.983
HPC2500 (ngrd)	20PE (500,318,318)	sendrecv -Cpp	5.6745 (19.041)	0.952
HPC2500 (ngrd)	24PE (500,318,318)	sendrecv -Cpp	4.1945 (21.820)	0.909
HPC2500 (ngrd)	32PE (500,318,318)	sendrecv -Cpp	4.1945 (25.760)	0.805
HPC2500 (ngrd)	33PE (500,318,318)	sendrecv -Cpp	15.2646 (7.078)	0.214
HPC2500 (ngrd)	40PE (500,318,318)	sendrecv -Cpp	13.9701 (7.733)	0.193
HPC2500 (ngrd)	48PE (500,318,318)	sendrecv -Cpp	12.5962 (8.577)	0.179
HPC2500 (ngrd)	2PE (500,318,318)	no comm -Cpp	47.6551 (2.268)	1.134
HPC2500 (ngrd)	4PE (500,318,318)	no comm -Cpp	22.6370 (4.774)	1.194
HPC2500 (ngrd)	8PE (500,318,318)	no comm -Cpp	11.6999 (9.237)	1.155
HPC2500 (ngrd)	16PE (500,318,318)	no comm -Cpp	5.8821 (18.376)	1.148
HPC2500 (ngrd)	24PE (500,318,318)	no comm -Cpp	4.3032 (25.118)	1.047
HPC2500 (ngrd)	32PE (500,318,318)	no comm -Cpp	3.2285 (33.479)	1.046
HPC2500 (ngrd)	48PE (500,318,318)	no comm -Cpp	2.3294 (46.401)	0.967
