

惑星磁気圏超並列高効率 MHDシミュレーションの開発

深沢圭一郎¹、荻野瀧樹²、梅田隆行²、
石井 克哉³、永井 亨³

1. 九州大学情報基盤研究開発センター
2. 名古屋大学太陽地球環境研究所
3. 名古屋大学情報基盤センター

1. Introduction

- 太陽地球惑星系科学の紹介
- 土星磁気圏におけるプラズマ対流構造
- MHD方程式とVlasov方程式

2. FX1とHX600におけるMHDシミュレーションの性能評価

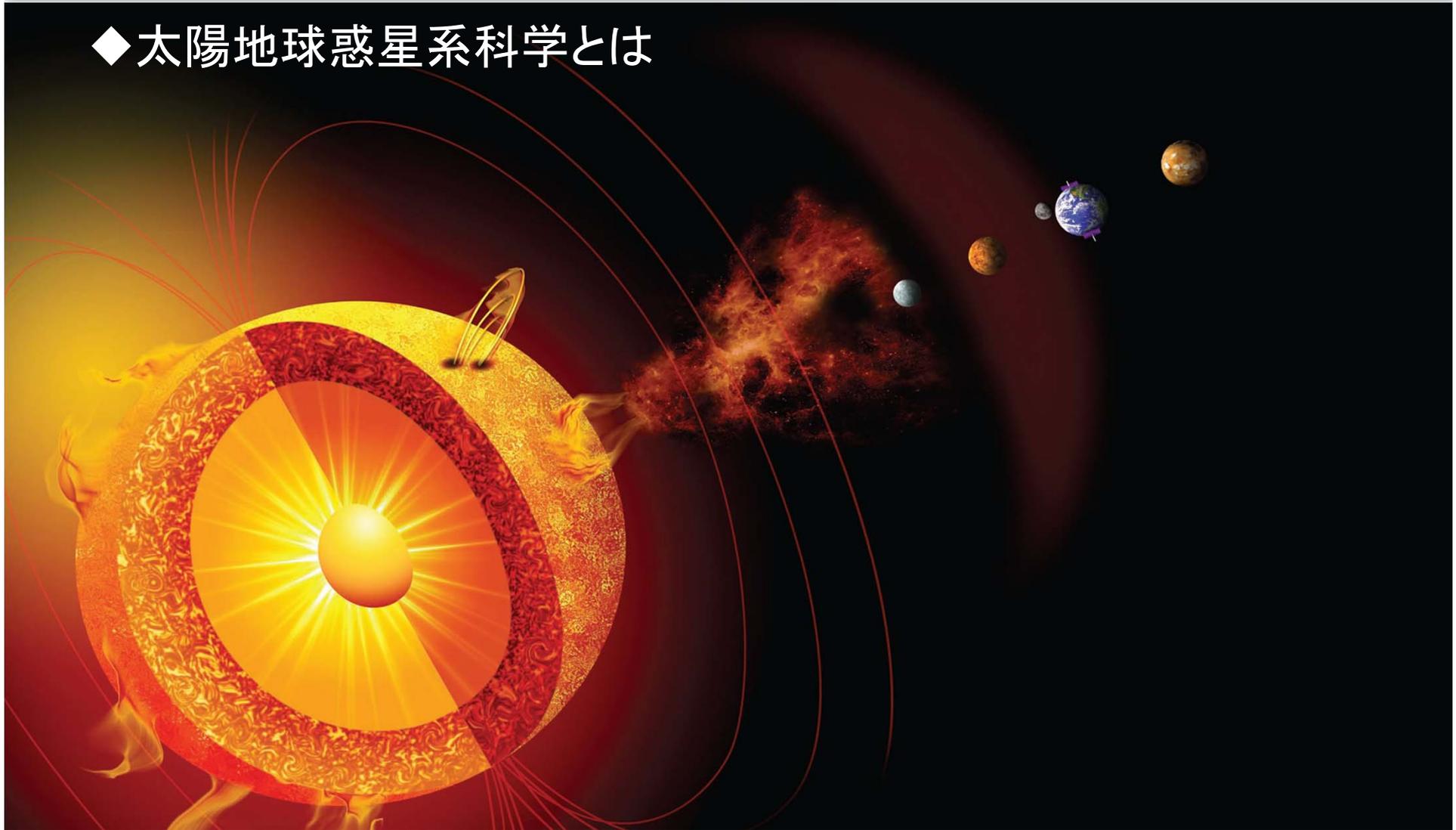
3. 性能プロファイル結果

4. 他計算機システムとの比較

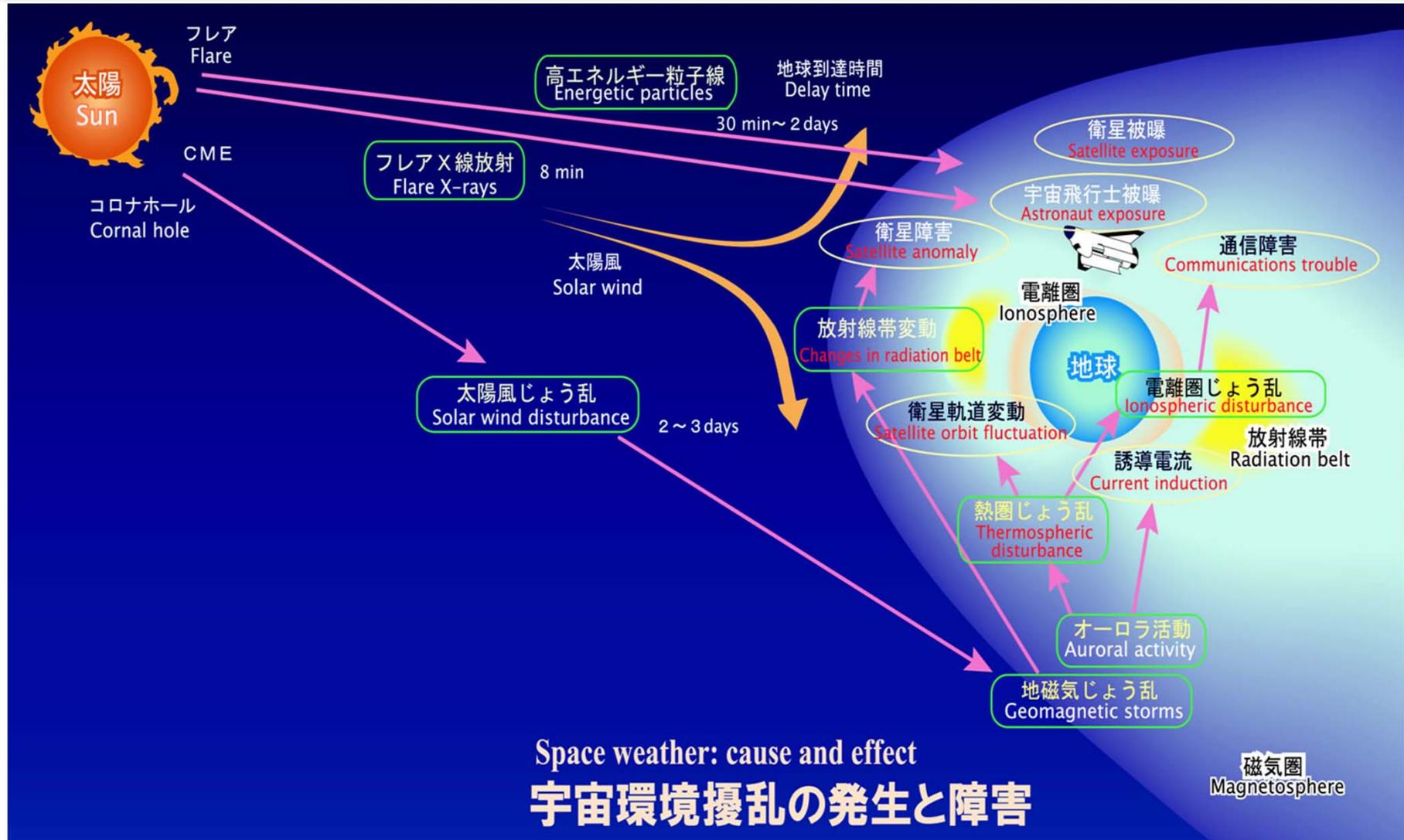
5. Summary



◆太陽地球惑星系科学とは



Introduction -2



◆地球、木星、土星における電磁気的特徴

	Jupiter	Saturn	Earth
Magnetic field [nT]	420,000	21,000	31,000
Magnetic polarity	N pole is north	N pole is north	N pole is south
Rotation period [hr]	10	10.65	24
Main plasma source	Io, ionosphere	Enceladus, ionosphere	ionosphere
Equatorial Radius [km]	71,492	60,268	6378
From Sun [A.U.]	5.2	9.55	1

木星は巨大な磁場と豊富なプラズマを持って高速自転する。
土星は豊富なプラズマを持って高速自転する。



◆地球、土星磁気圏の形

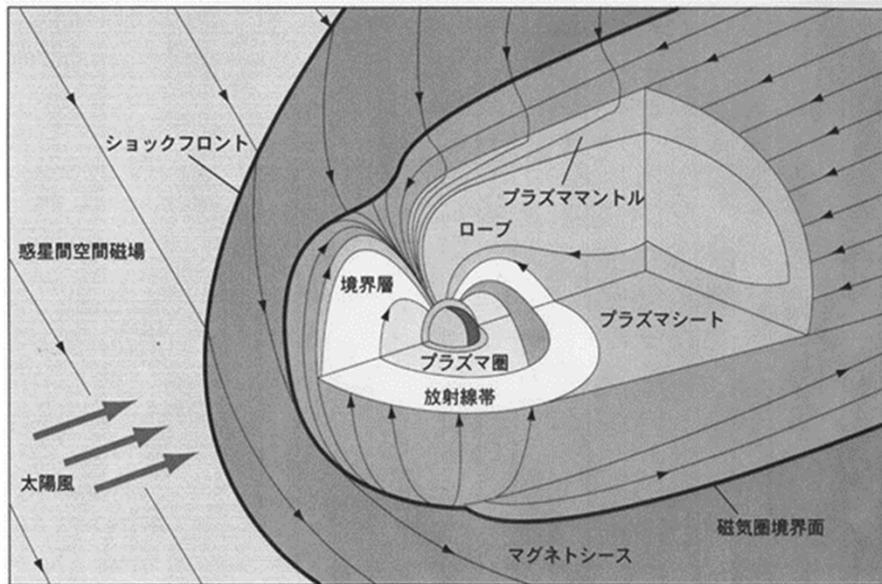


Fig.1. A schematic of Terrestrial magnetosphere

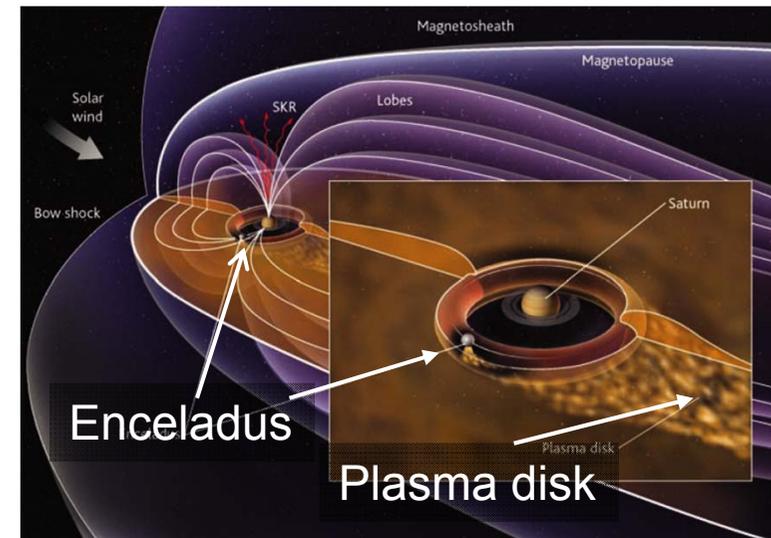
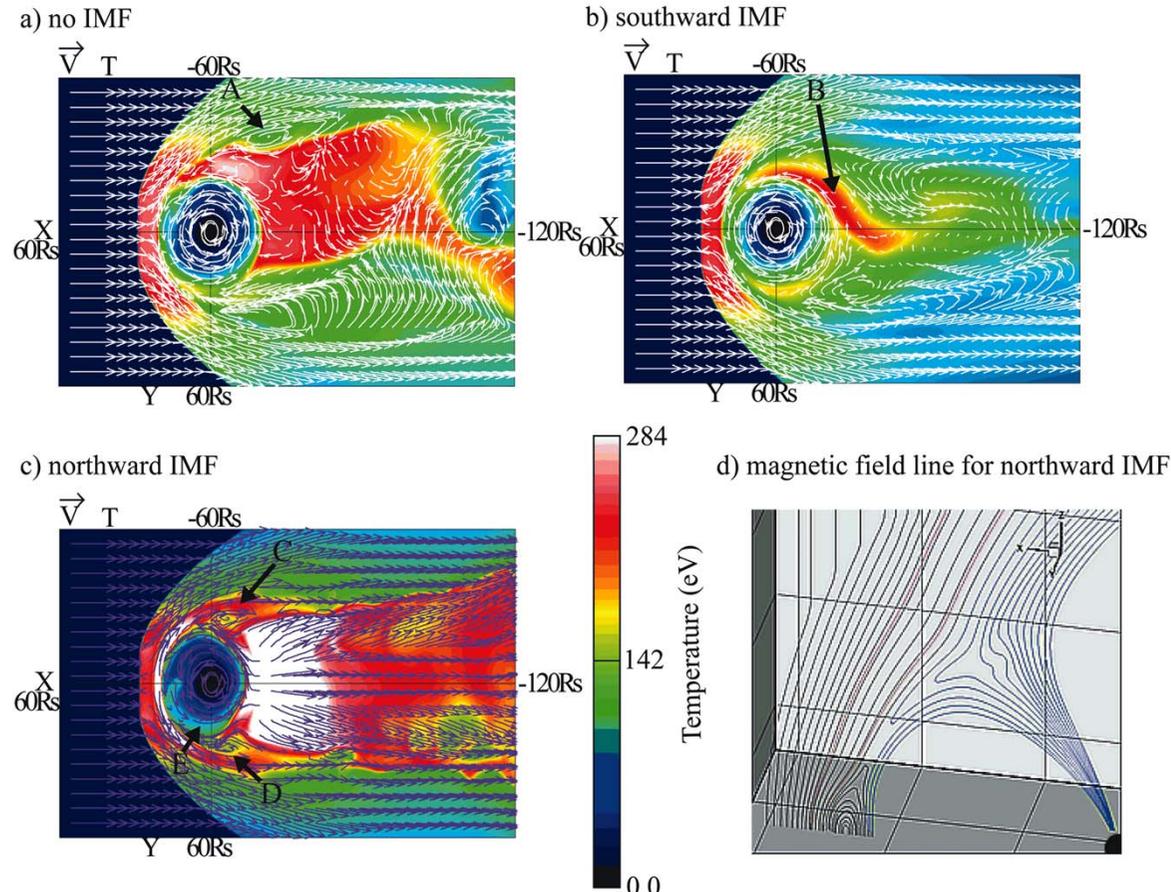


Fig.2. Schematic of Saturn's magnetosphere [Kivelson, 2006]

土星は高速自転しているため、磁気圏内のプラズマ対流が複雑。



◆土星磁気圏対流構造

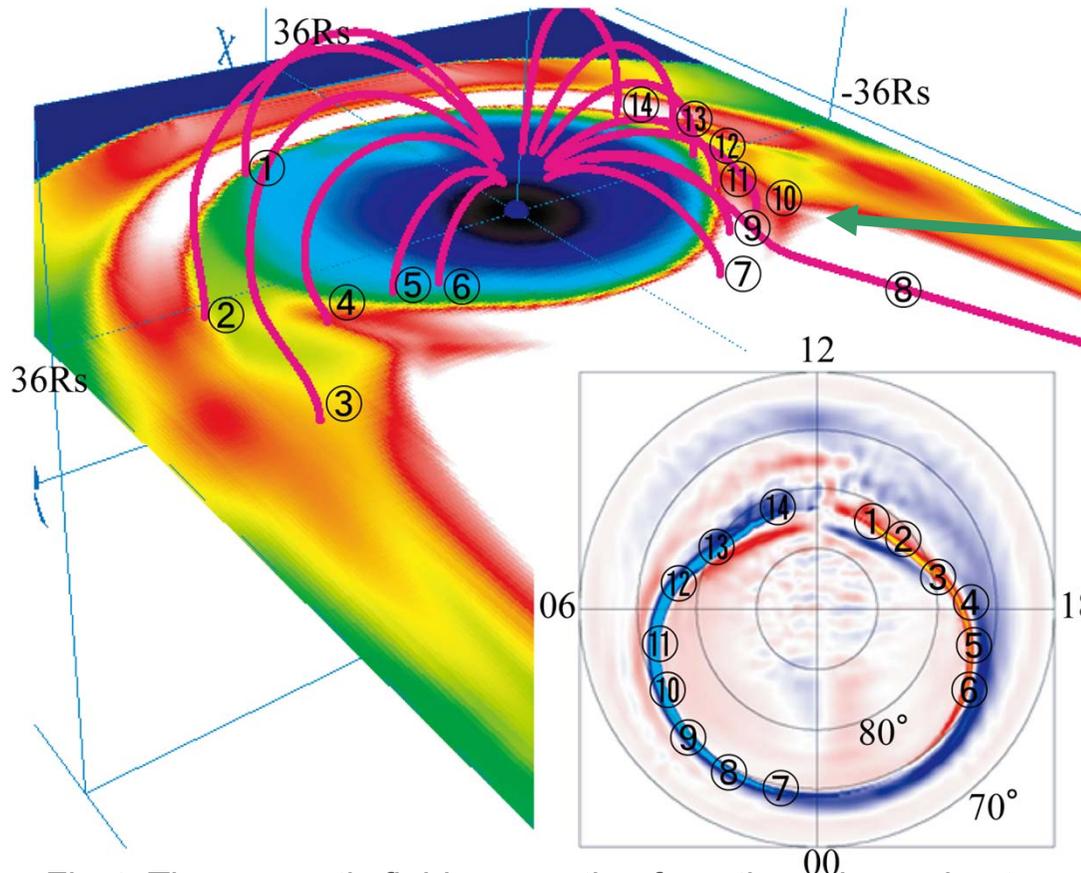


どの条件においても土星磁気圏では渦構造が発達している。

no IMF、southward IMFではスケールの大きな渦、northward IMFでは小さいが、強い渦が見える。

Fig.3. The temperature and flow vectors in the equatorial plane for the simulations with no IMF (a), southward (b) and northward IMF (c) [Fukazawa et al., 2007a]

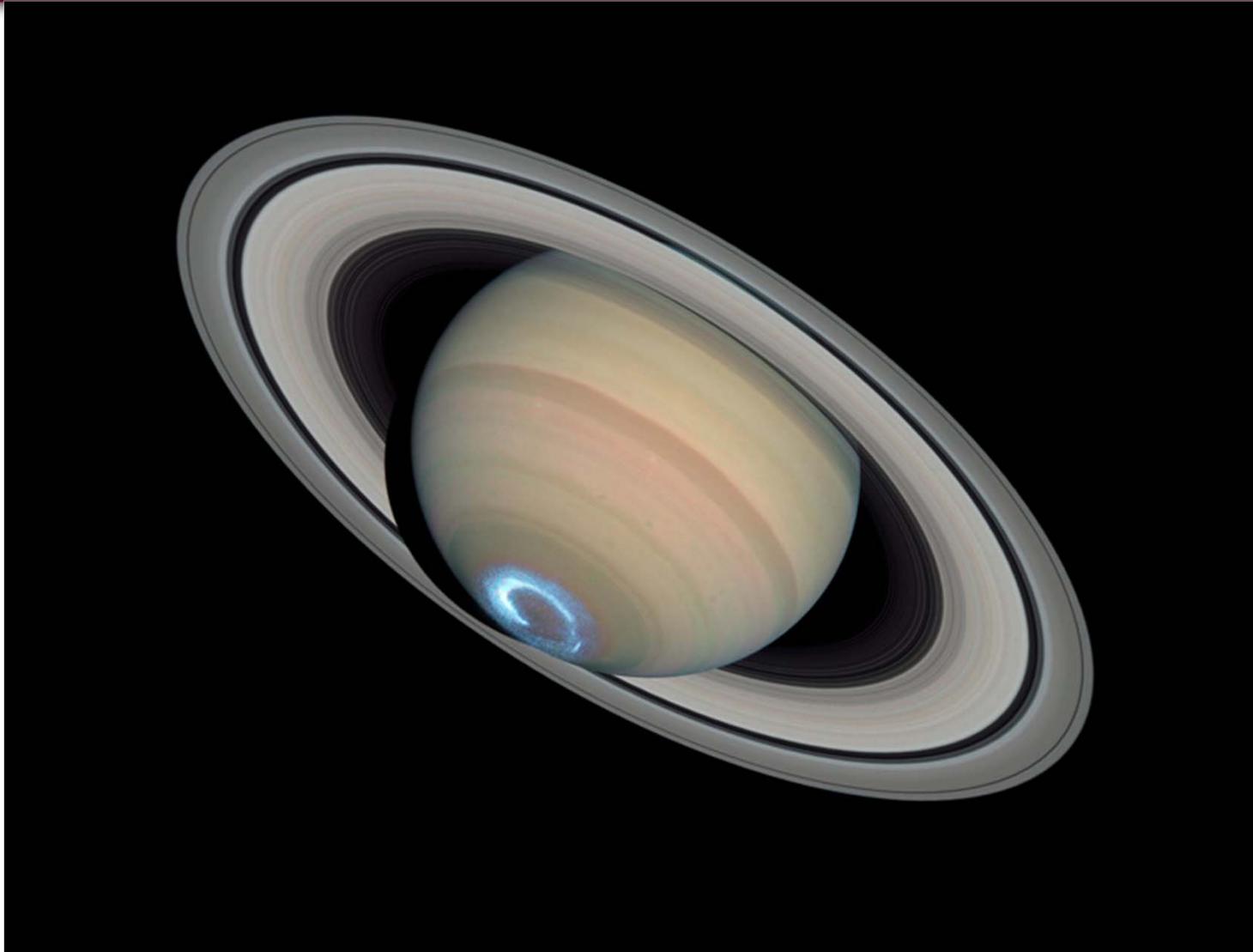
◆対流構造と沿磁力線電流の関係



dawn側の渦が生成されている場所で上向き沿磁力線電流が強い。

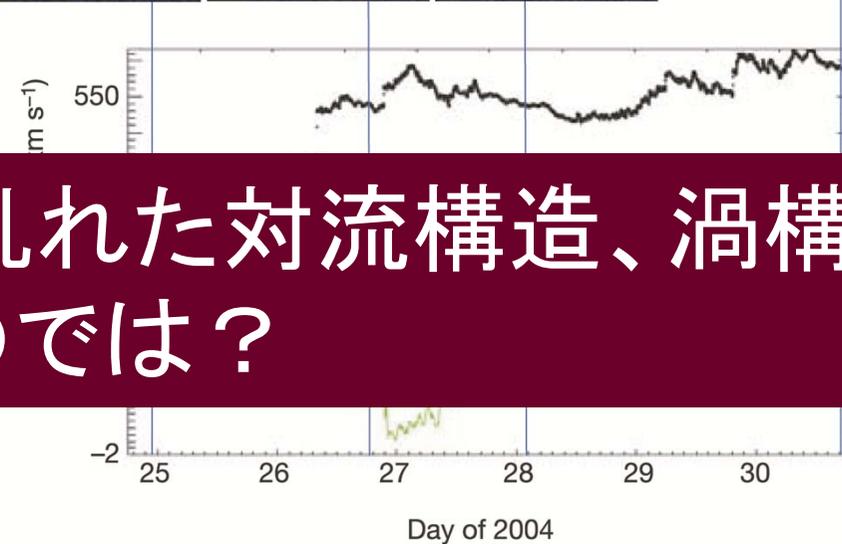
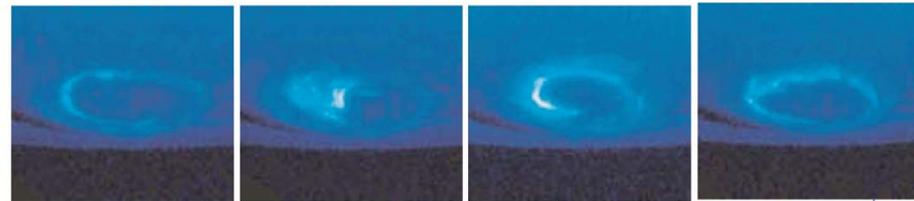
*南極側のプロットなので、左右が反転する。

Fig.4. The magnetic field connection from the polar region to the magnetosphere [Fukazawa et al., 2007a]



◆HSTとCassiniの同時観測(2004年)

- Cassiniが太陽風を観測して($500R_S$ 程度上流)、HSTが土星のオーロラを撮像。



- 土星磁気圏の乱れた対流構造、渦構造に関係するのでは？

Fig.4. Comparison between HST images and solar wind conditions propagated to Saturn for the period 25–30 January 2004 [Crary et al., 2005]

Introduction -10

◆目的

□惑星磁気圏コードの性能評価

- 惑星磁気圏を解くMHDコードを用いて、名古屋大学のスーパーコンピュータシステムでどれだけ速度が出るかを調べる。
→どのようなコードの組み方が最も効率的かを調べる。
- 最終的には上記のコードを利用して、高解像度土星磁気圏シミュレーションを行い、プラズマ対流の渦構造を詳しく調べることを目標とする。



電磁流体コード -1

◆宇宙プラズマを取り扱う方程式(1)

□Vlasov方程式

- 無衝突Boltzmann方程式とMaxwell方程式から成るプラズマの振る舞いを最も正確に表現できる方程式系。

速度分布関数 $f(x, v, t)$ を考えると、

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \frac{\partial f}{\partial \mathbf{x}} + \frac{q}{m} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f}{\partial \mathbf{v}} = 0$$

これとMaxwell方程式を連立して解く。

- しかし、位置 (x, y, z) 3次元、速度 (V_x, V_y, V_z) 3次元と時間から成る非線形方程式系で、解くことが困難。



電磁流体コード -2

◆宇宙プラズマを取り扱う方程式(2)

□MHD (Magnetohydrodynamics)方程式

- Vlasov方程式のn次モーメント取ることで、求められる。

0次(速度空間で積分)、1次(\mathbf{v} かけて積分)、2次(\mathbf{v}^2 かけて積分)より、

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\mathbf{v} \rho)$$

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

$$\frac{\partial p}{\partial t} = -(\mathbf{v} \cdot \nabla) p - \gamma p \nabla \cdot \mathbf{v}$$

を得る。これらと磁場の誘導方程式

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{v} \times \mathbf{B}) \quad \text{をまとめてMHD方程式という。}$$



◆ベンチマークセッティング

□解法はLeap-frog+Lax Wendroff法の混合手法

- 磁場を扱う方程式では $\text{div} \cdot \mathbf{B} = 0$ を保証する手法を用いる必要があり、適用可能手法に限界がある。
- 今までのベンチマークとの比較、また、simpleな手法を用いることで、計算機の性能が出やすいと考えられる。

□基本サイズ

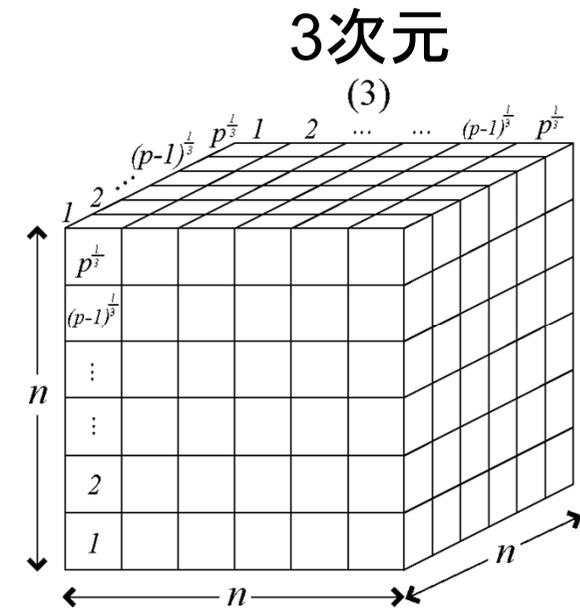
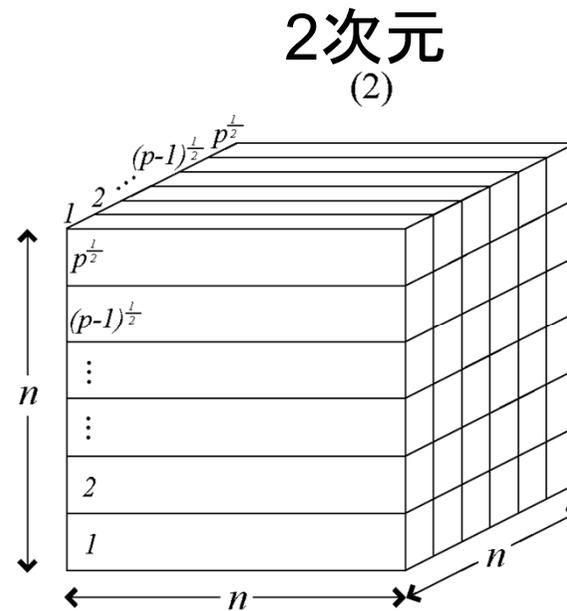
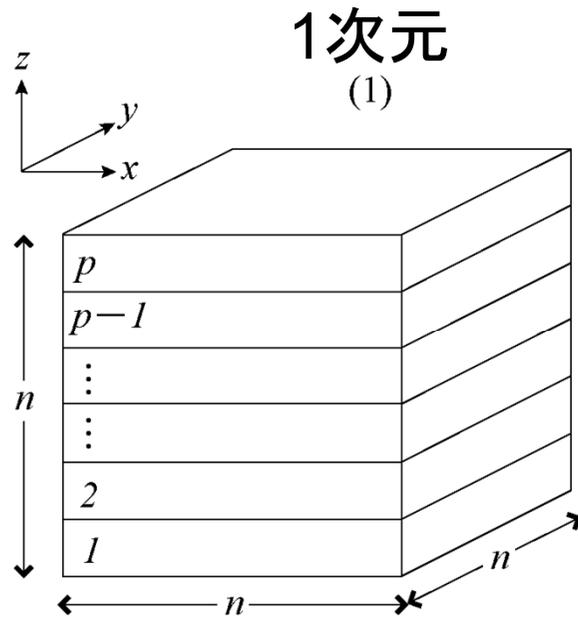
- 1coreあたり $f(x, y, z, 8) = 16387064 (127^3 \times 8)$ grid を設定。
→weak scaling
- 計算上この7倍の配列数を使用する。

□MPI並列

- のりしろは前後一つ。OpenMPは使用せず。
→グリッドを1/2メッシュ幅ずらした2個のシミュレーションボックスを利用し、1ループで2段計算するため、実際は2次精度。



◆3種類の領域分割法



計算時間 (T_S)

$$T_{S1} = k_1 n^2 (p - 1)$$

$$T_{S2} = k_1 n^2 (p - 1)$$

$$T_{S3} = k_1 n^2 (p - 1)$$

通信時間 (T_C)

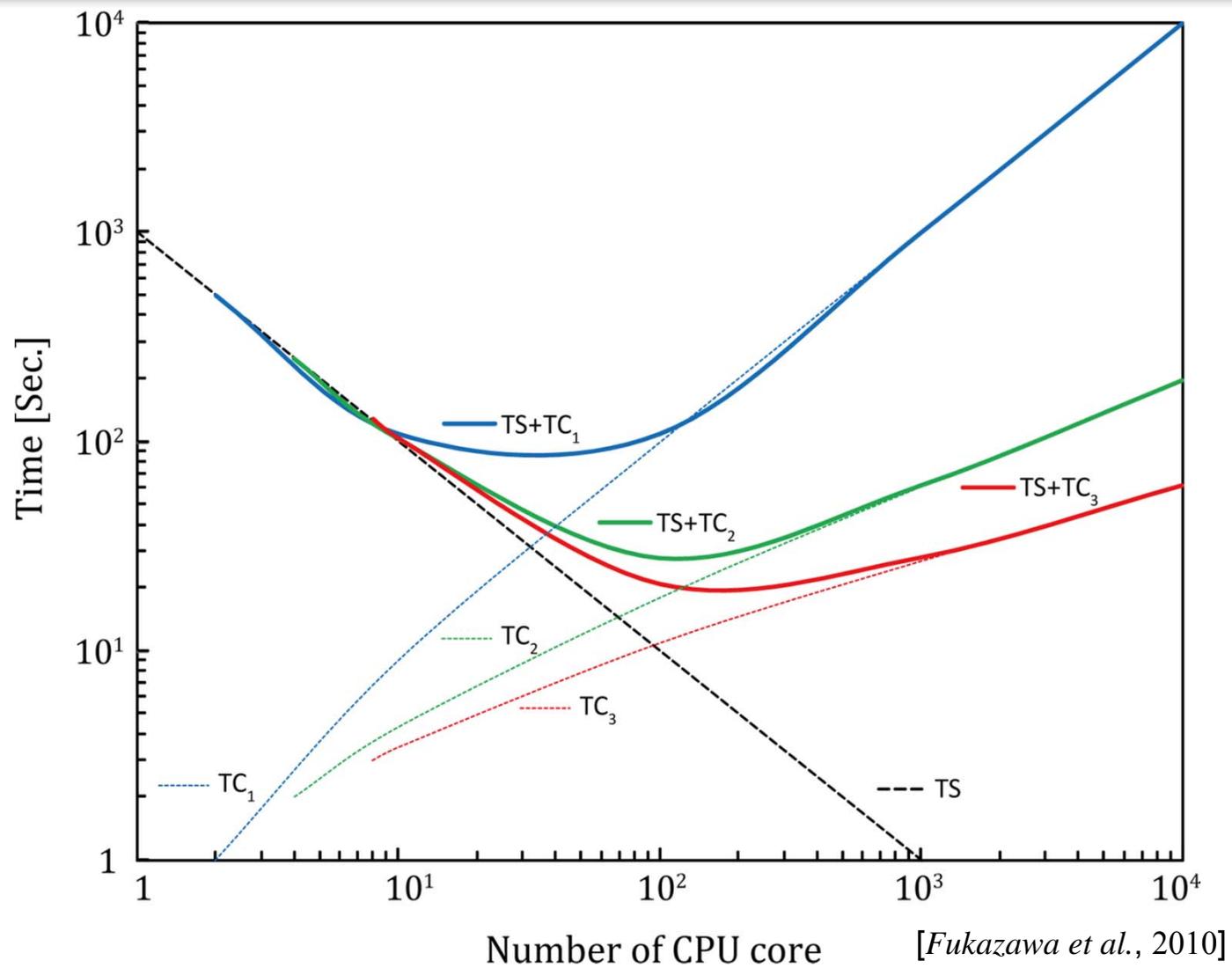
$$T_{C1} = k_2 n^2 (p - 1)$$

$$T_{C2} = 2k_2 n^2 (p^{\frac{1}{2}} - 1)$$

$$T_{C3} = 3k_2 n^2 (p^{\frac{1}{3}} - 1)$$

[Fukazawa et al., 2010]

Estimated Time



◆スカラーチップではキャッシュが重要

□キャッシュの基本的働き

- CPU/core内部に設けられた高速な記憶装置。ここに使用頻度の高いデータを蓄積しておくことにより、低速なメインメモリへのアクセスを減らすことができ、処理を高速化することができる。
- メインメモリにアクセスしたときに、その周辺データをキャッシュに格納する。なので、連続データアクセスが多いプログラムは効率的にキャッシュを利用可能。

□性能評価

- スカラ機ではキャッシュヒット向上を見込んで、配列の並び替えを行うと性能が上がる機種がある(非x86系)ので、その効果も評価する。
- Type A: $f(i, j, k, m)$
- Type B: $f(m, i, j, k)$



FX1におけるMHD シミュレーションによる 性能評価



◆ 富士通製スカラSMP計算機



SPARCアーキテクチャのため、
プレタコンと呼ばれる。
SMT機能が有り。
※今回は使わず

		FX1
CPU	型式	quad core SPARC64 VII
	周波数	2.5GHz (40GFlops)
	Cache	L1: 64KB/core L2: 6MB/CPU
Memory	Band幅	40GB/s /node
B/F値	40/40	1
Node	Core数	4
	メモリ	32GB
System	Node数	768 (3072core)
	理論性能	30.72TFlops
	Node間通信	InfiniBand DDR (2GB/s) with IMPACT

□利用計算機資源

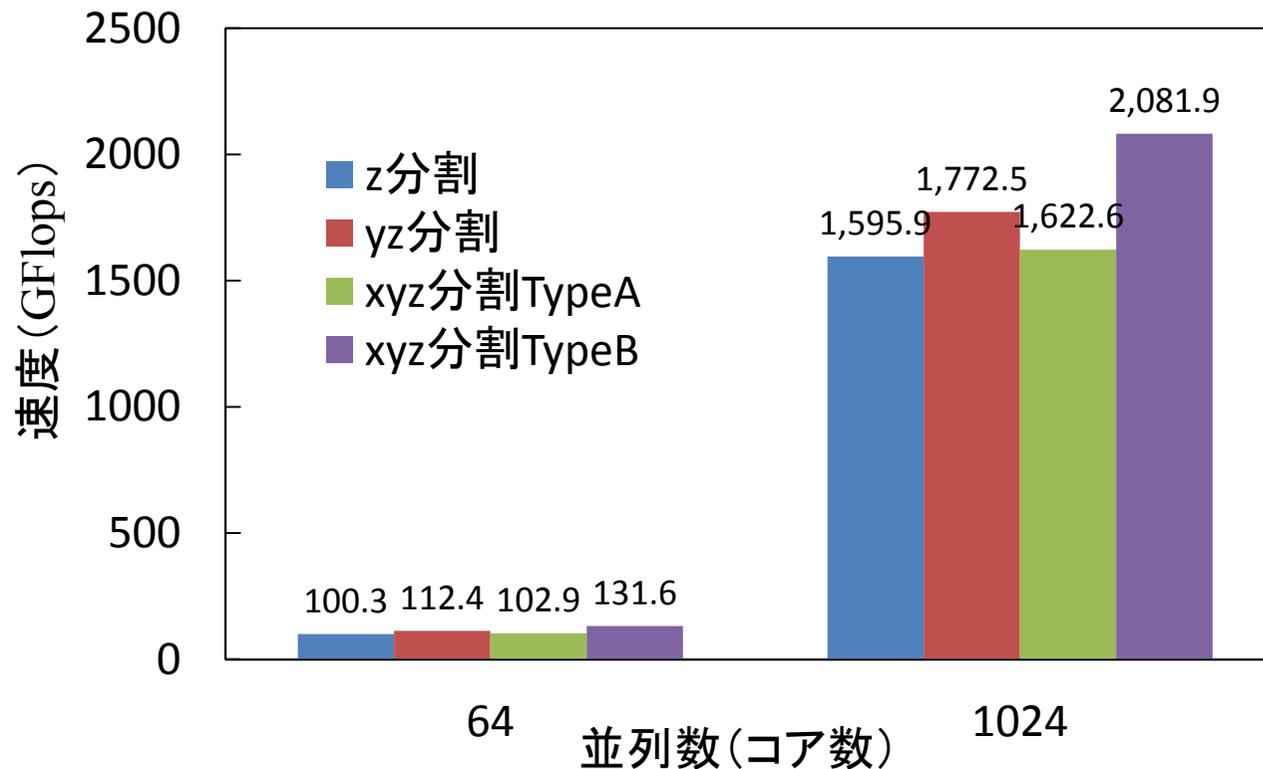
- 名古屋大学HPC計算科学連携研究プロジェクトSTE研計算機共同利用研究で利用した最大256CPU(1024core)で10TFlopsの理論性能。

□コンパイラオプション

- -Am -Kfast, prefetch_model=FX1, prefetch_indirect, prefetch_cache_level=3, prefetch_sequential, prefetch_strong, V9, impact, frecipro, loop, nomfunc, noparallel, swp, tiling



◆ 1次元領域分割 vs 2次元領域分割 vs 3次元領域分割



3次元領域分割TypeBが最速。他は大差なし。
1024並列で最大実行効率は21%



HX600におけるMHD シミュレーションによる 性能評価



◆富士通製T2K型計算機



初期型と比べ、CPU性能の向上、メモリの容量増加、バンド幅が向上

		HX600
CPU	型式	Quad Core Opteron 8380
	周波数	2.5GHz (40GFlops)
	Cache	L1: 64KB/1core L2: 512KB/1core L3: 6MB/CPU
Memory	Band幅	12.8GB/s /CPU
B/F値	12.8/40	0.32
Node	Core数	16
	メモリ	64GB
System	Node数	150
	理論性能	25.6TFlops
	Node間通信	Infiniband DDR (2GB/s) × 4

□利用計算機資源

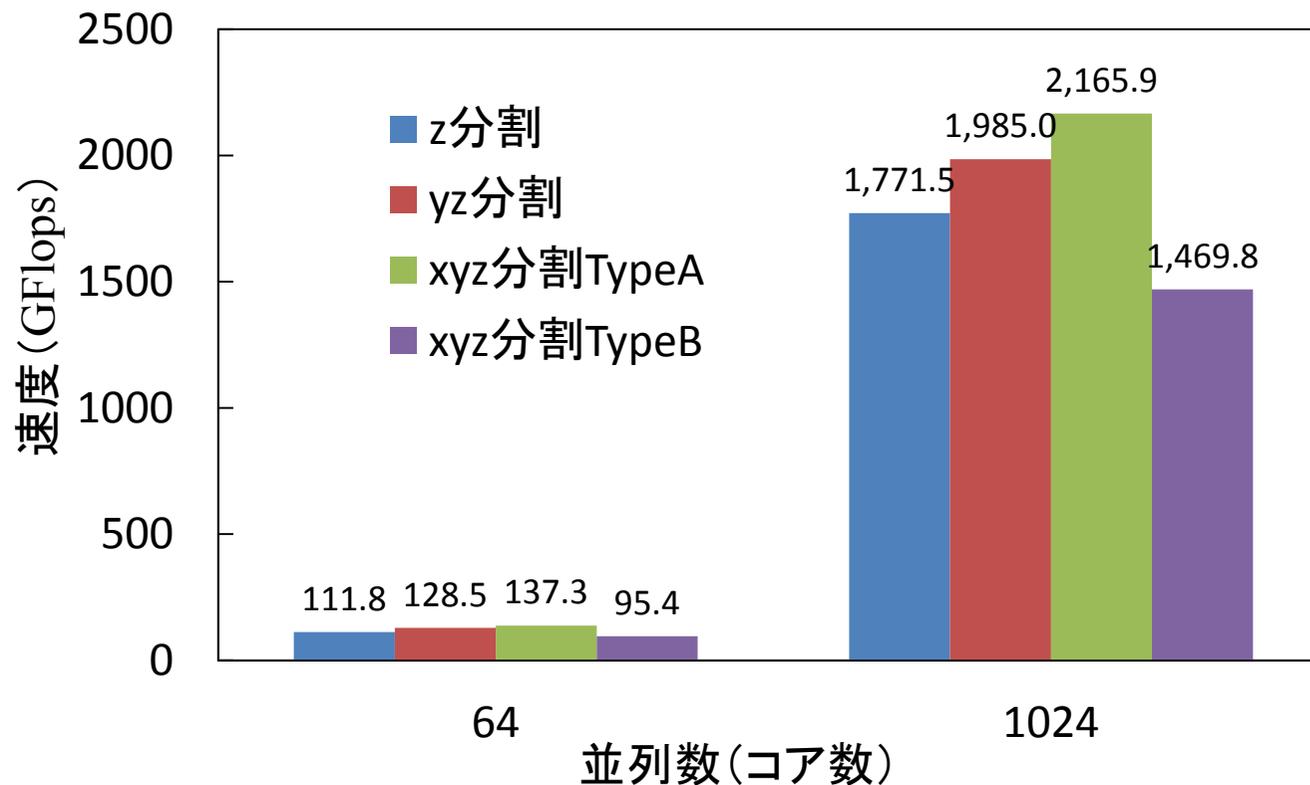
- 名古屋大学HPC計算科学連携研究プロジェクトで使
した最大256CPU(1024core)で10TFlopsの理論性能。

□コンパイラオプション

- -Am -Kfast, fsimple, fuse, prefetch_indirect,
prefetch_sequential, SSE4, loop, nomfunc, noparallel, tiling
※fastにSSE2、SSE3が含まれる



◆ 1次元領域分割 vs 2次元領域分割 vs 3次元領域分割



3次元領域分割TypeAが最速(21%)。2次元領域分割もまあまあ。
FX1などに有効なTypeBはHX600には有効ではない。

性能プロファイルの結果

◆ FX1の計算時におけるハードウェア情報

	TypeA	TypeB
L1データミス率 (%)	0.0041	0.0034
L1 命令ミス率 (%)	0.9197	0.7233
L2キャッシュミス率(%)	0.1432	0.1562
メモリアクセス命令率 (%)	47.3556	48.8111
データアクセスコスト率 (%)	35.8635	28.139
メモリスループット (MB/S)	72687.0156	86305.5078
浮動小数点演算命令率(%)	26.9919	33.9202
高速命令率 (%)	7.2932	8.3666
分岐命令率 (%)	0.6168	0.4226
分岐予測ミス率 (%)	5.0339	3.8461
Prefetch(%)	10.2553	4.2402
通信時間割合 (%)	6.5636	5.3289

プロファイラの利用法

コンパイル(リンク時にも)オプションに `-Ktl_trt`をつける。

実行時に `fpcoll` オプション `mpixec`で実行。

・オプション

`-lhwm,mpi`

`-lhwm,mpi -Hevent=MEM_access`

`-lhwm,mpi -Hevent=Cache`

`-lhwm,mpi -Hevent=Cache_overlap`

このオプションにより、収集できるデータが異なる。

今回はコード自体の情報(通信時間、MPI処理時間、サブルーチンのコスト)と処理中のハードウェア情報(キャッシュヒット率、メモリアクセス命令率、データアクセスコスト率、浮動小数点演算命令率、高速命令率、分岐命令率、分岐命令予測ミス率、prefetch命令率)を用いた。

←赤文字の値が良い数値



性能プロファイルの結果

◆ HX600の計算時におけるハードウェア情報

	TypeA	TypeB
L2キャッシュミス率 (%)	1.4674	0.4678
L3 キャッシュミス率 (%)	1.1848	0.4211
命令実行レイテンシ (%)	26.4721	53.6704
命令実行ストール率 (%)	73.5279	46.3296
メモリスループット (MB/S)	78868.5234	60146.1328
分岐予測ミスストール (%)	3.0427	0.7298
命令デコードストール (%)	19.4821	26.2697
整数データレジスタ 依存ストール率 (%)	1.7587	0.5458
浮動小数点データ 依存ストール率 (%)	3.6592	7.2851
ロード/ストアストール率 (%)	48.6351	12.3438
通信時間割合 (%)	7.7250	4.1374

プロファイラの利用法
コンパイル(リンク時にも)オプションに `-Ktl_trt`をつける。
実行時に `fpcoll` オプション `mpiexec`で実行。

・オプション
`-lhwm,mpi`
`-lhwm,mpi -Hevent=MEM_access`
`-lhwm,mpi -Hevent=Cache`
`-lhwm,mpi -Hevent=Exec_status`
`-lhwm,mpi -Hevent=Stall_BE_flush`
`-lhwm,mpi -Hevent=Stall_unit`

このオプションにより、収集できるデータが異なる。
今回はコード自体の情報(通信時間、MPI処理時間、サブルーチンのコスト)と処理中のハードウェア情報(キャッシュヒット率、命令実行レイテンシ、メモリスループット、浮動小数点、整数データレジスタ依存ストール率、分岐命令率、分岐命令予測ミス率)を用いた。

←赤文字の値が良い数値



MHDシミュレーションの性能評価(対他システム) 27

◆ 現状の各種スパコンとの比較

	Core数 /CPU数	実効性能 [GFlops]	理論性能 [GFlops]	実行効率 [%]	領域分割	CPU種類
SX-9	64/64	2188	6553	33	2次元	ベクトル
SX-8R	8/8	80	282	28	1次元	ベクトル
HA8000	8192/1024	10038	75366	13	3次元A	Opteron
HX600	1024/256	2166	10240	21	3次元A	Opteron
FX1	1024/256	2081	10240	21	3次元B	SPARC64VII
SR16000	1344/672	5375	25267	21	3次元B	POWER6
PQ580	2048/1024	2145	13107	16	3次元B	Itanium2
PG S3	1536/768	2536	18432	14	3次元A	Xeon

スカラCPUではx86系は3次元TypeA、それ以外はTypeBが最適という傾向が強い。



◆ SX-9と各種スパコンとの比較

	@core	@CPU	CPU種類
SX-9	1.0000	1.0000	ベクトル
SX-8R	0.2925	0.2925	ベクトル
HA8000	0.0358	0.1432	Opteron
HX600	0.0618	0.2474	Opteron
FX1	0.0594	0.2377	SPARC64VII
SR16000	0.1170	0.2340	POWER6
PQ580	0.0306	0.0613	Itanium2
PG S3	0.0483	0.0966	Xeon

実アプリではFX1、SR16000やHX600が4CPUでSX-9の1CPUと同等の性能になっている。



- ◆ MHDシミュレーションの性能評価をおこなった
 - FX1では3次元領域分割TypeBが最も性能が良く、1024coreを利用した場合実行効率21%、実効性能2TFlopsを達成した。
 - HX600では3次元領域分割TypeAが最高で1024core利用時に21%の効率、2.1TFlopsの性能だった。
 - 性能プロファイルを細かに見ていくと、性能向上余地があり、シミュレーションを実行する前に、より一層のチューニングが必要と考えられる。
 - 他の計算システムと比較すると、スカラではx86系は3次元領域分割TypeA、それ以外では3次元領域分割TypeBが最適であった。
 - スカラCPU単体がベクトルCPU単体の性能に近づきつつある。

