

電磁流体コードによる大規模惑星磁気圏シミュレーション

深沢圭一郎^{1,2}、梅田隆行³、荻野瀧樹³

1. 九州大学大学院 理学研究院 地球惑星科学科
2. 日本学術振興会 特別研究員PD
3. 名古屋大学太陽地球環境研究所

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

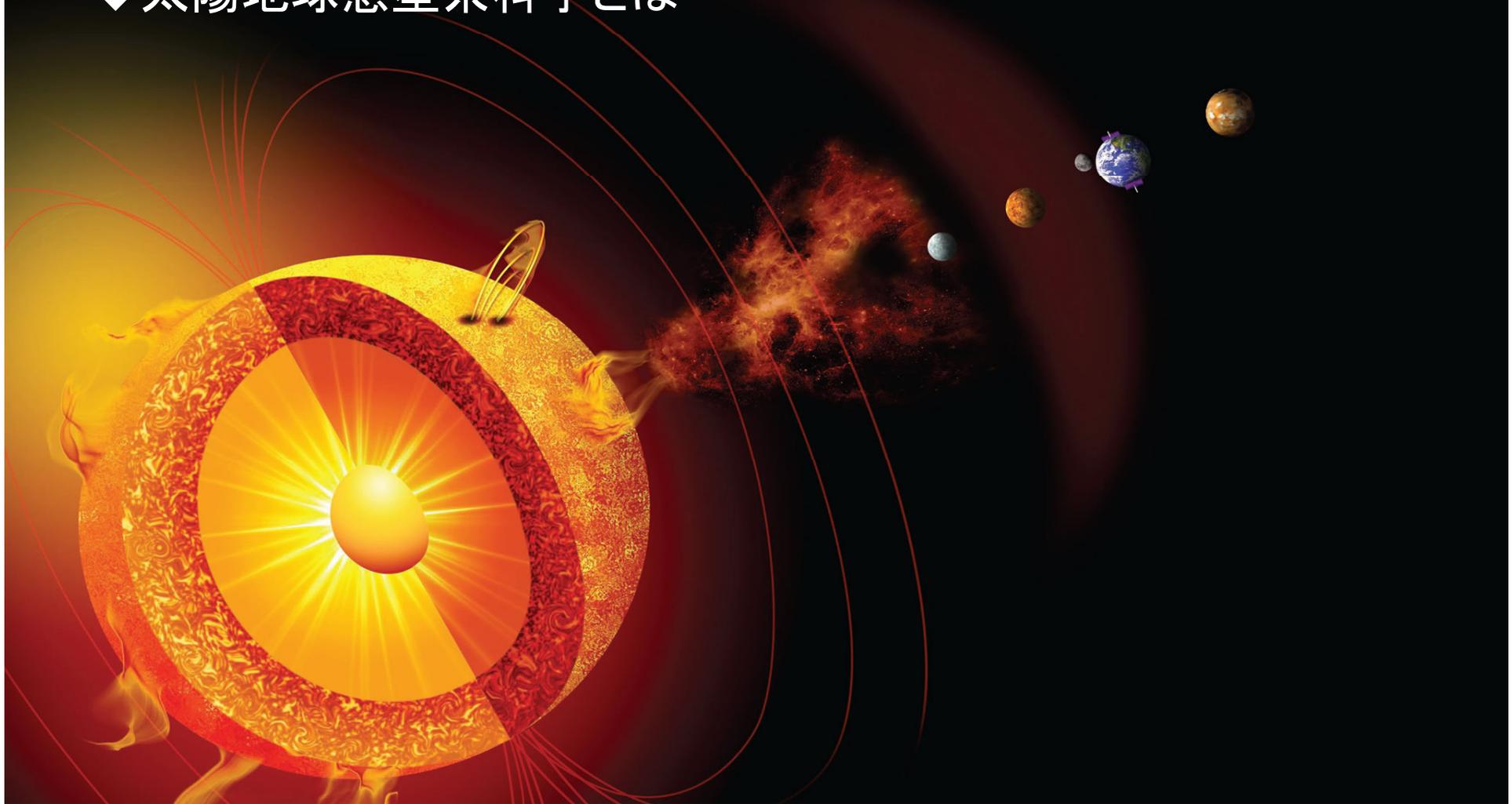
2. HA8000におけるMHDシミュレーションの性能評価

3. HA8000における土星磁気圏シミュレーションの結果

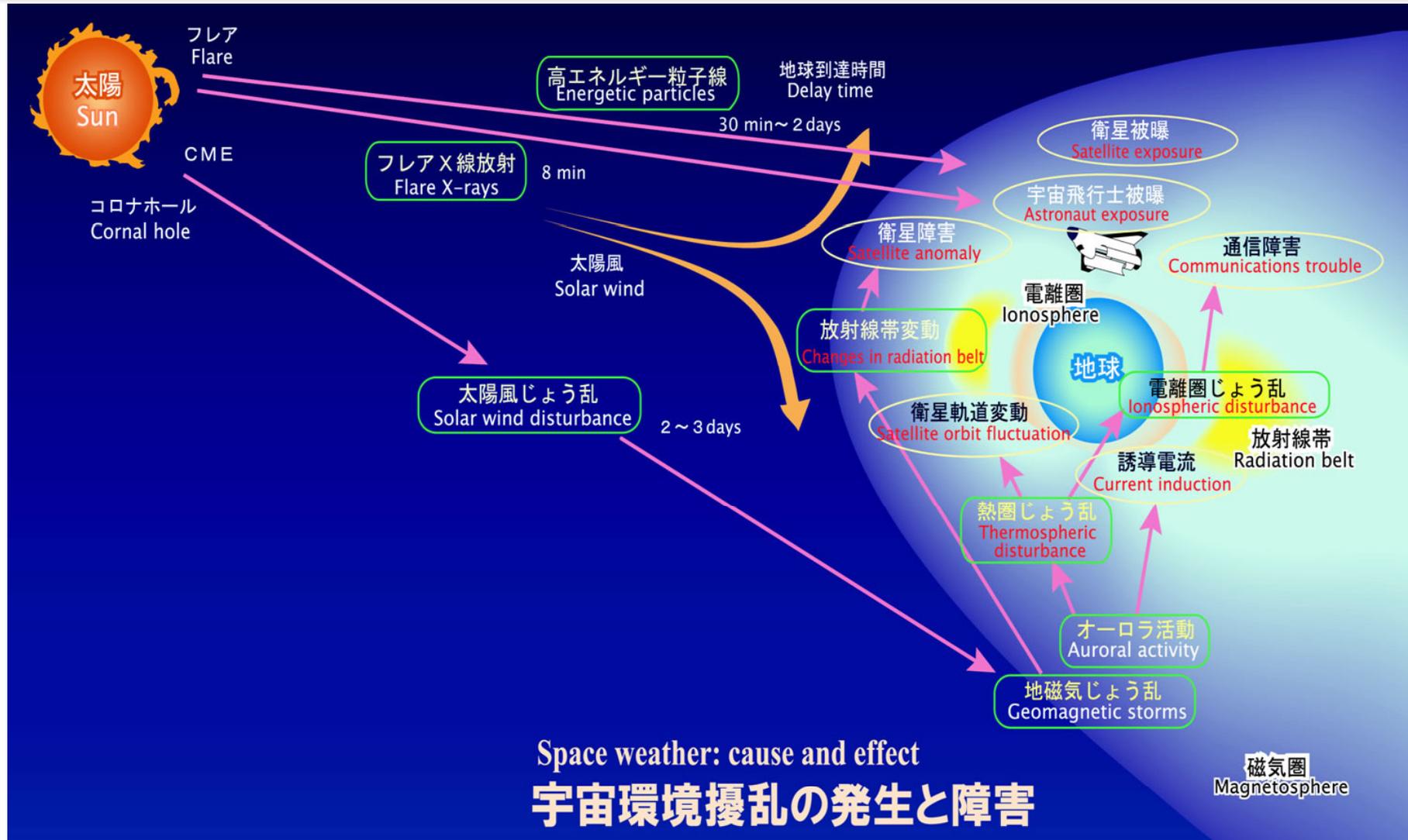
4. まとめ



◆太陽地球惑星系科学とは

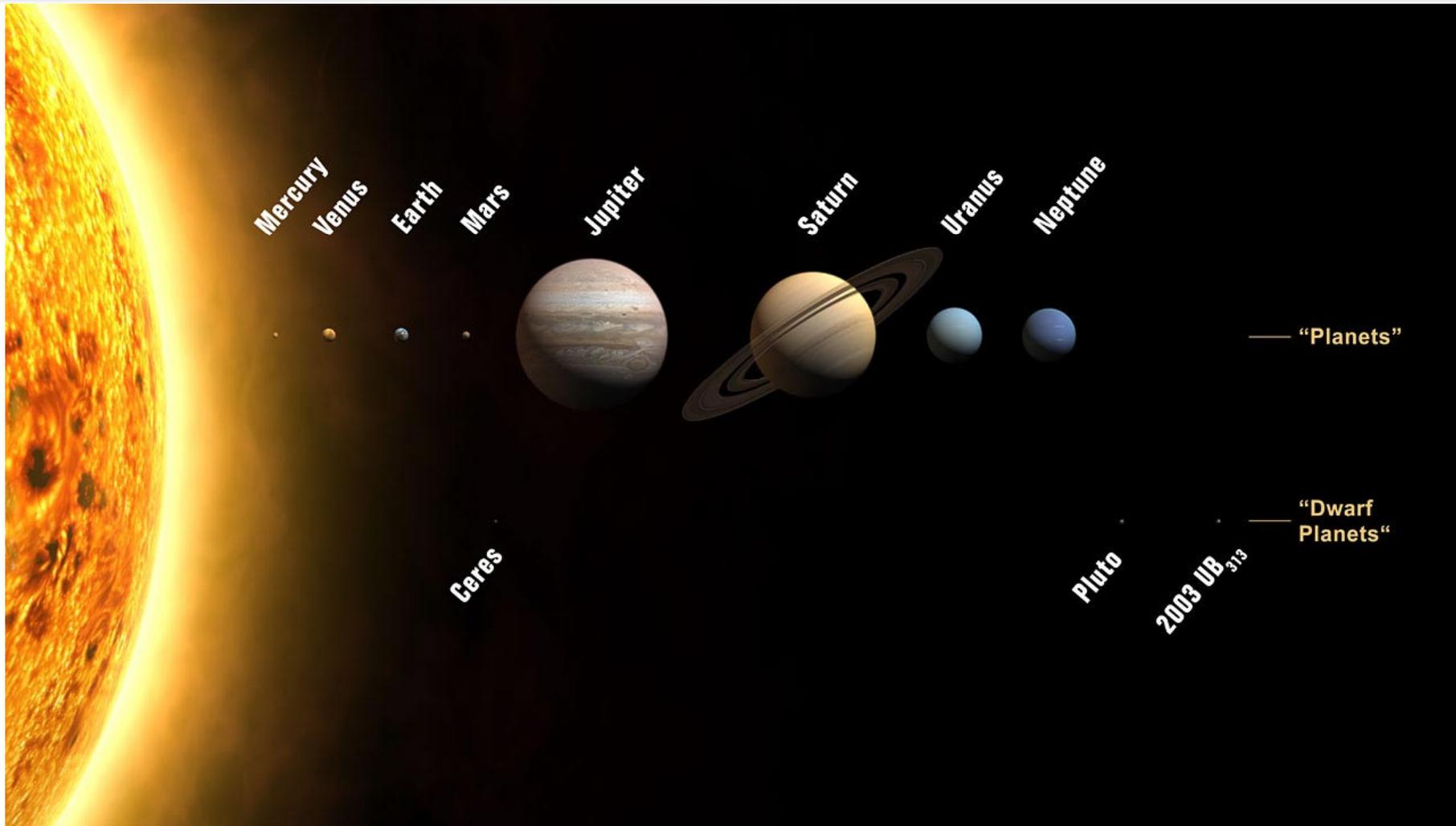


Introduction -2



Introduction -3

5



◆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

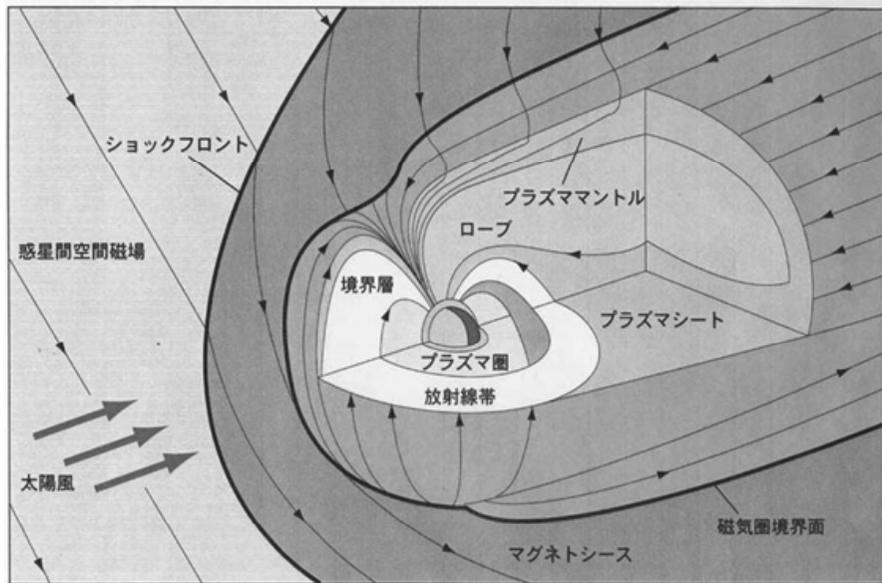


Fig.1. A schematic of Terrestrial magnetosphere

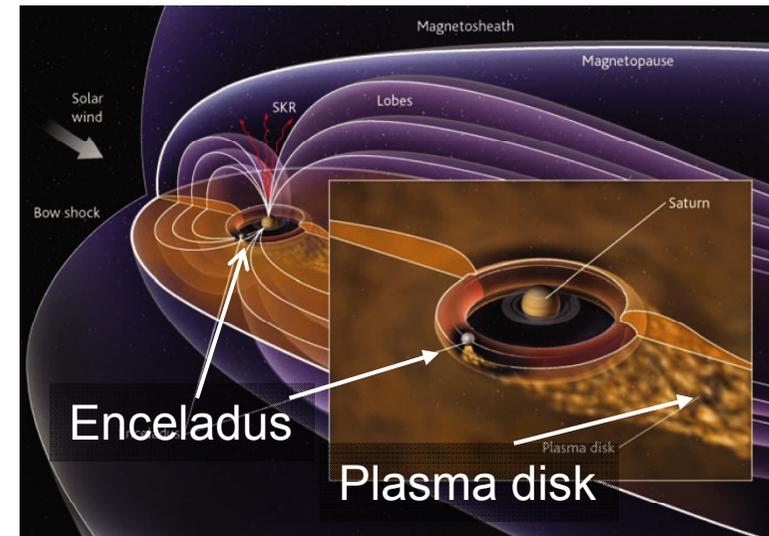
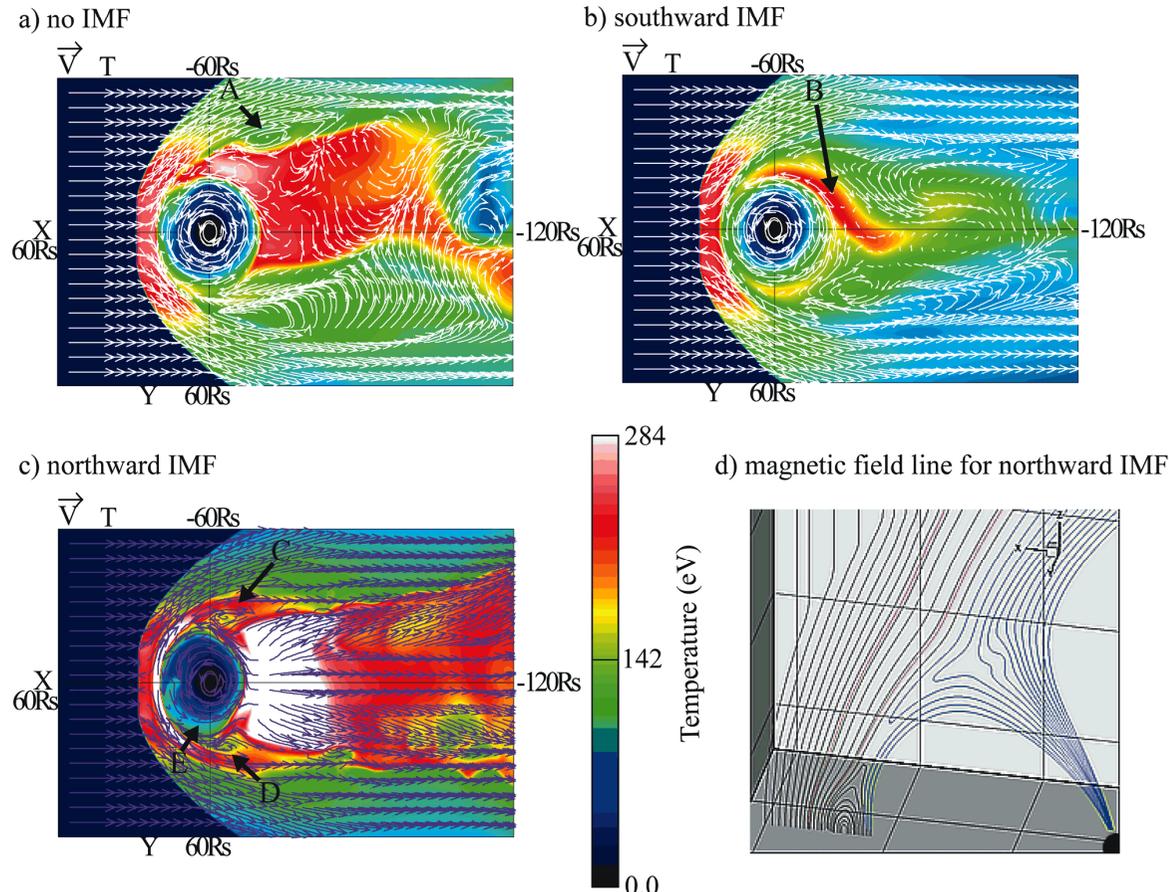


Fig.2. Schematic of Saturn's magnetosphere [Kivelson, 2006]

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

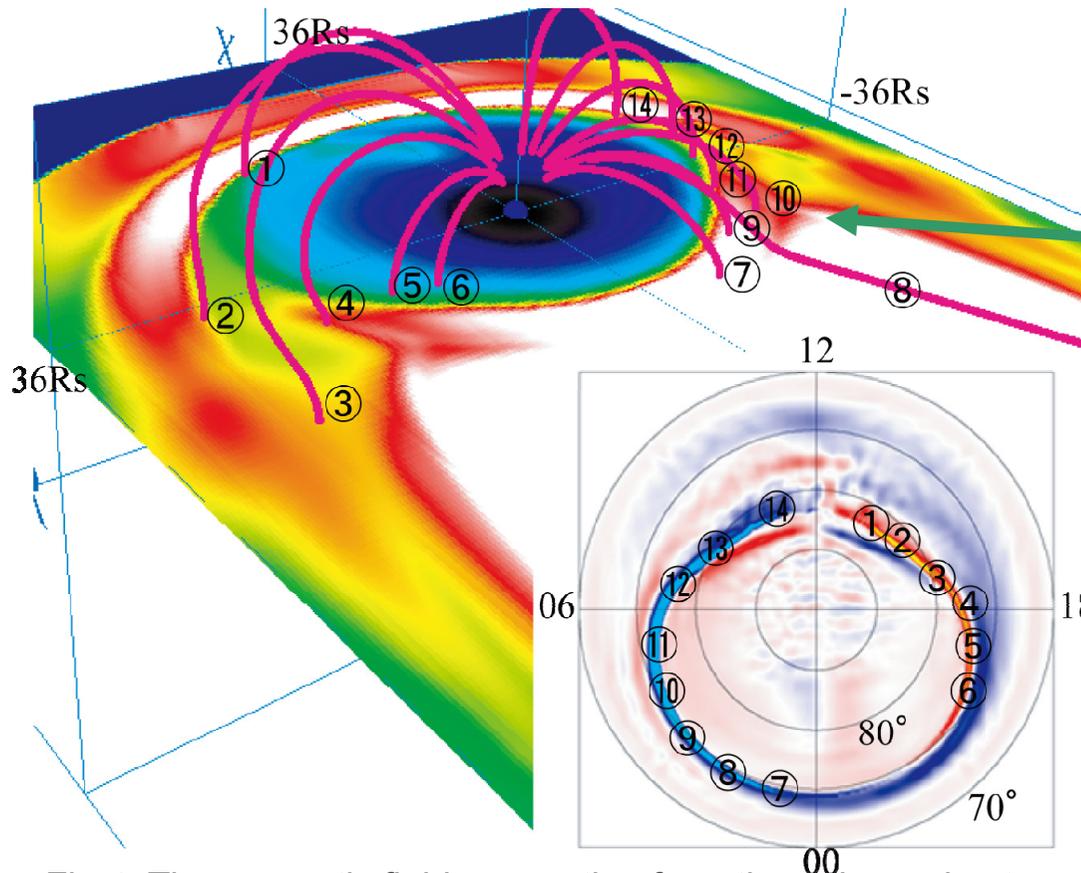


どの条件においても土星磁気圏では渦構造が発達している。

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]

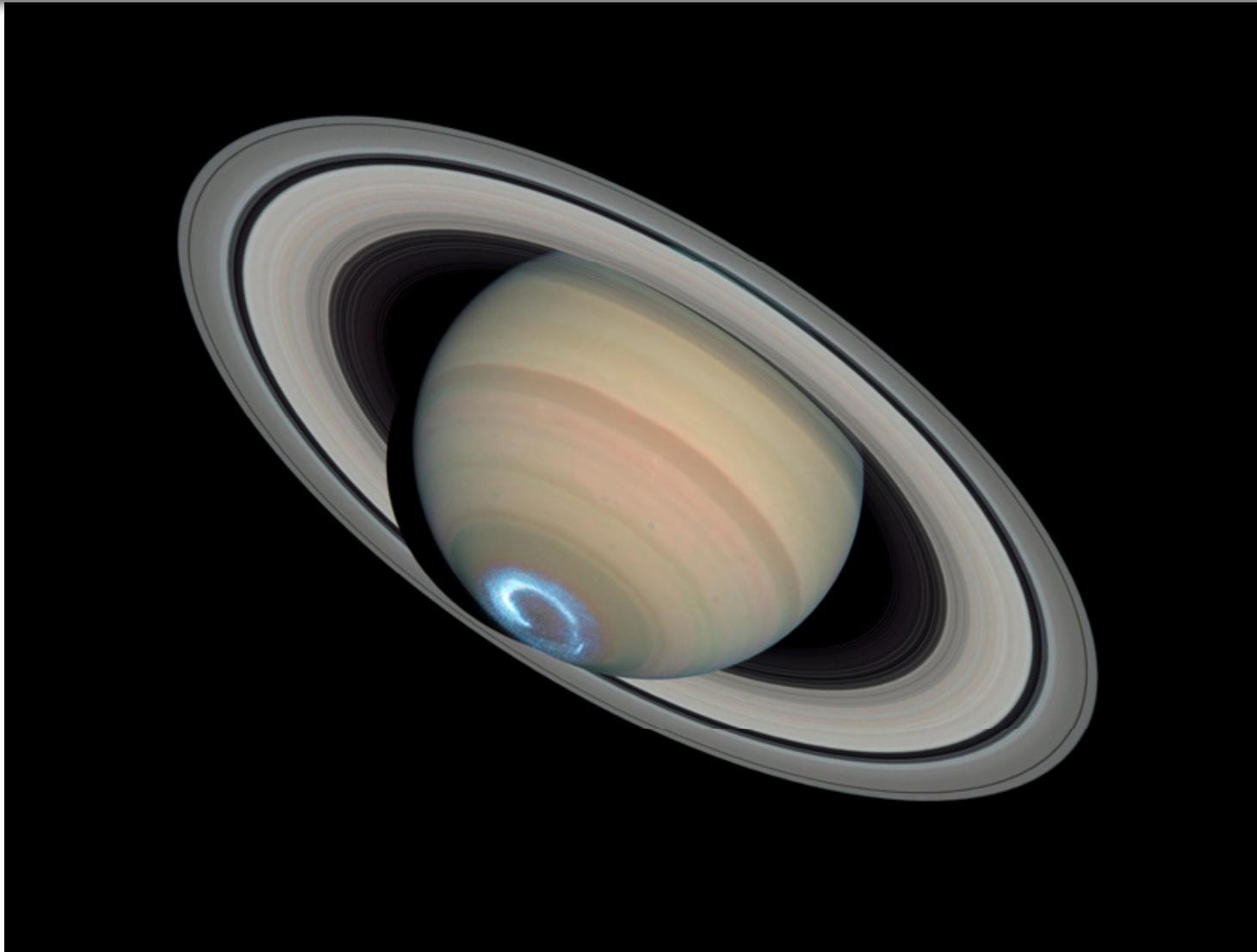
◆対流構造と沿磁力線電流の関係



dawn側の渦が起
こっている場所で上
向き沿磁力線電流
が強い。

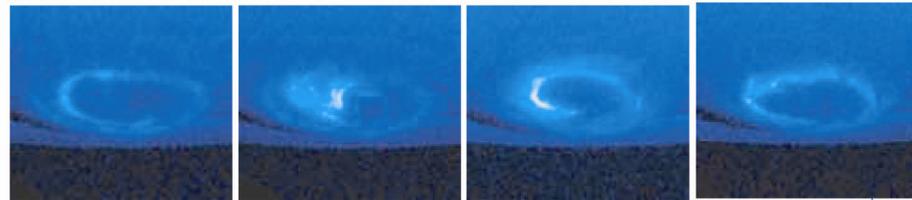
*南極側のプロットなので、左右
が反転する。

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？

因は不明。

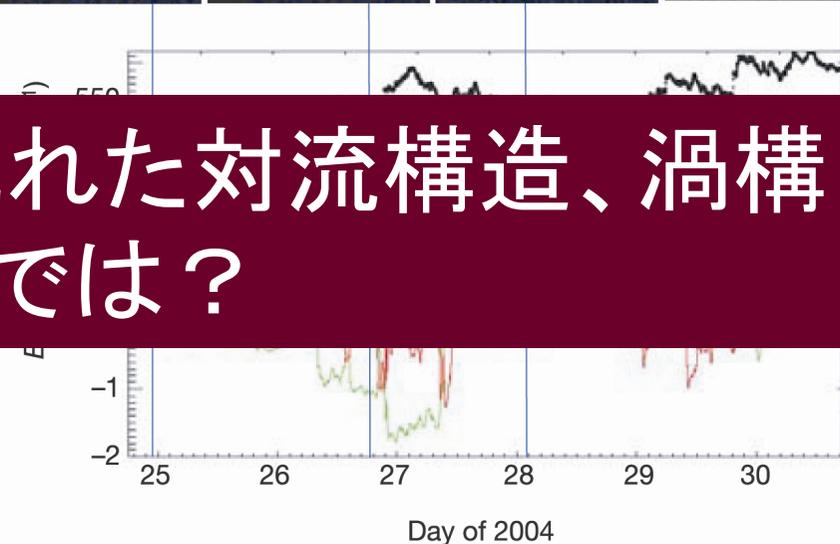


Fig.4. Comparison between HST images and solar wind conditions propagated to Saturn for the period 25–30 January 2004 [Crary *et al.*, 2005]

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方程式を用いて、HA8000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。
- 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



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 $f(\mathbf{x}, \mathbf{v}, t)$ 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

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



MHDシミュレーション による性能評価



◆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られなければならず、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、HA8000の性能が出やすいと考えられる。

□基本サイズ

- 1coreあたり $f(x, y, z, 8) = 691200$ gridを設定。
- 計算上この7倍の配列数を使用する。

□MPI並列

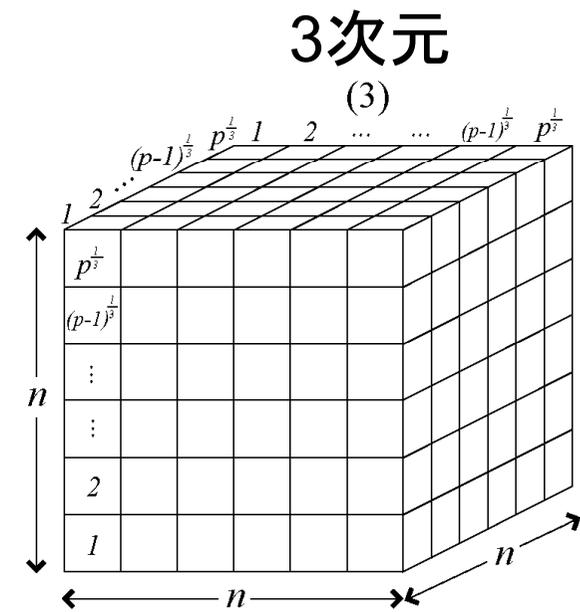
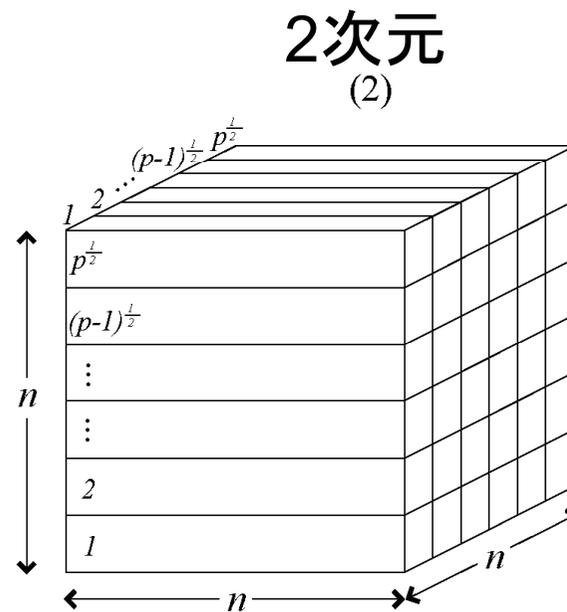
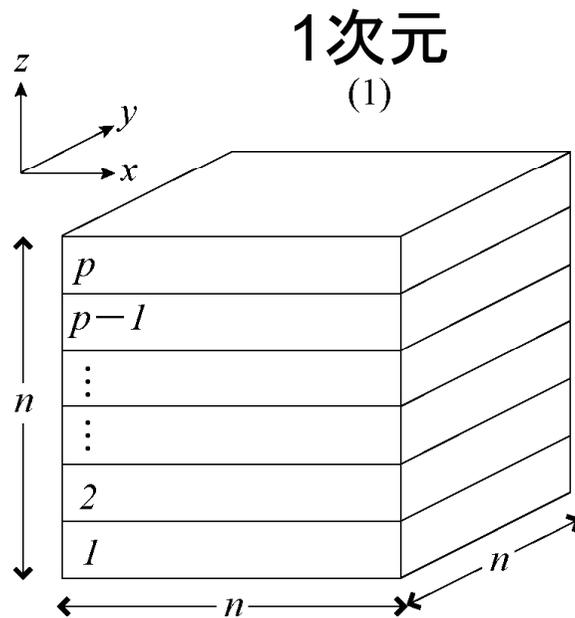
- のりしろは前後一つ。OpenMPは使用せず。

□コンパイラオプション

- 日立製: `-Oss -noparallel -autoinline=2 -looptiling -pvfunc=0 -nolimit -noscope -divopt`
- Intel製: `-O3 -msse3 -xSSE3 -ipo`



◆3種類の領域分割法



計算時間 (T_S)

$$T_{S1} = k_2 n^2 (p - 1)$$

$$T_{S2} = k_2 n^2 (p - 1)$$

$$T_{S3} = k_2 n^2 (p - 1)$$

通信時間 (T_C)

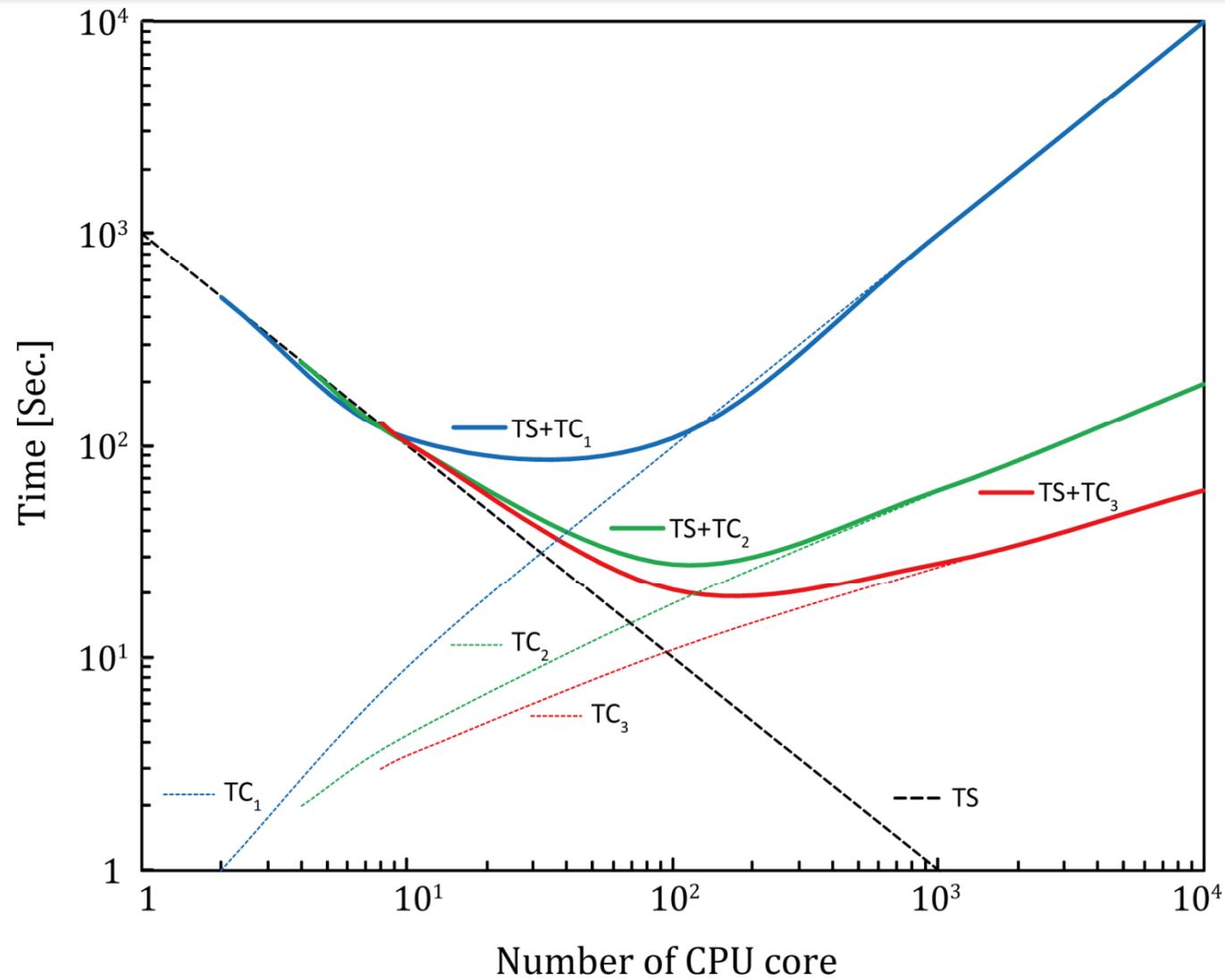
$$T_{C1} = k_2 n^2 (p - 1)$$

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

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



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。

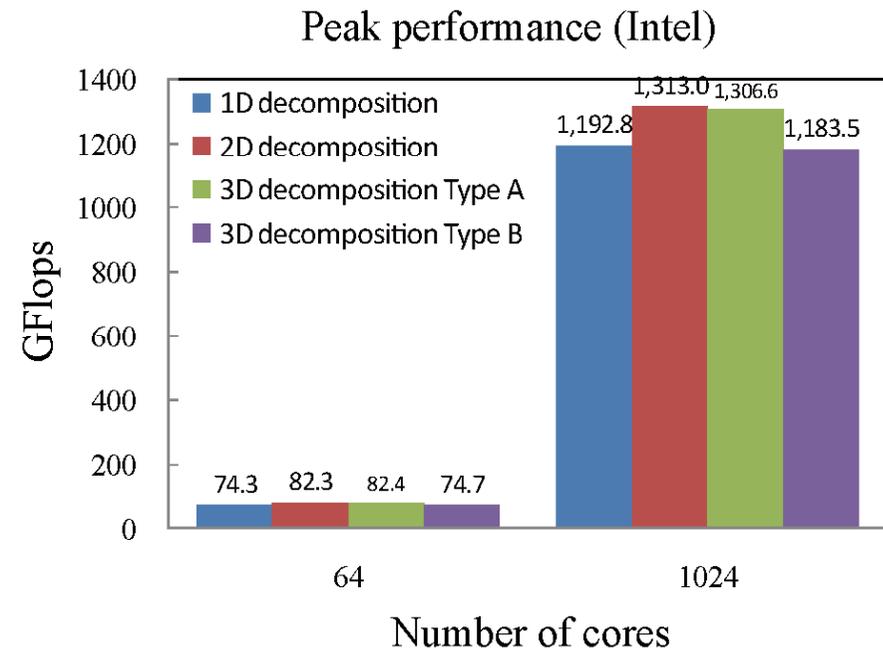
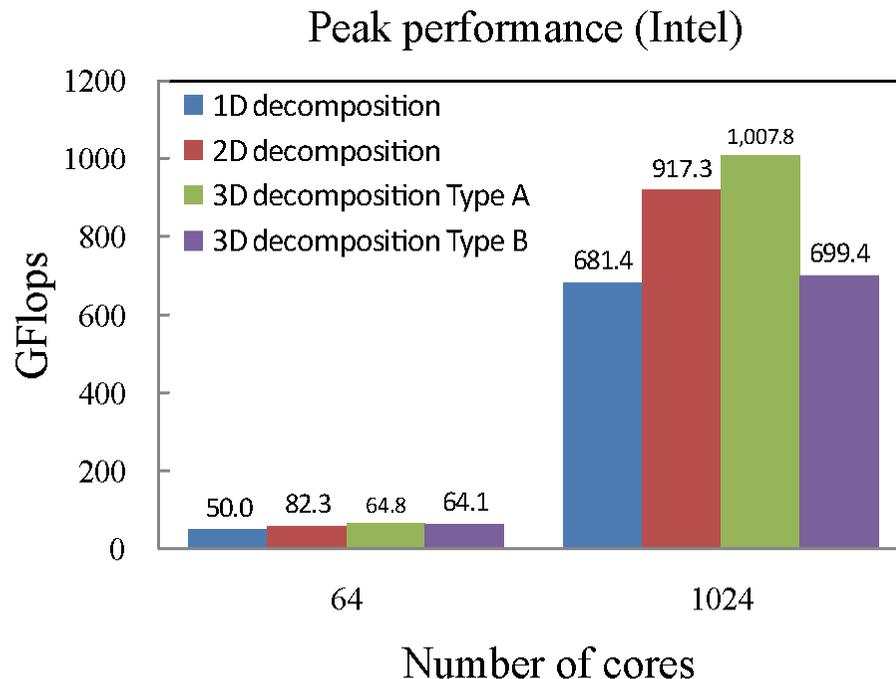
□性能評価

- 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: $f(m, i, j, k)$



MHDシミュレーションの性能評価 (HA8000) -1 20

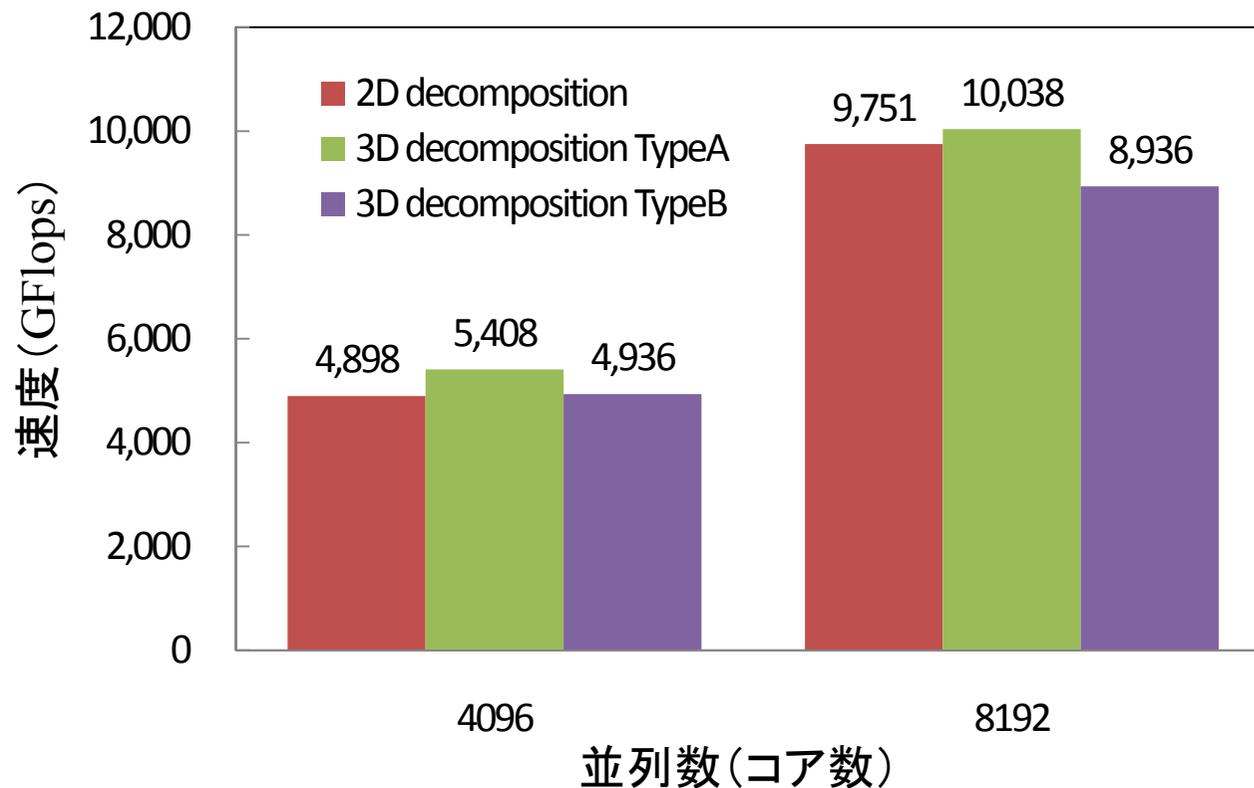
◆ 1次元領域分割 vs 2次元領域分割 vs 3次元領域分割



HA8000には2次元領域分割が適している(日立、Intelともに)
1024coreで最大実行効率Hitachiで2次元領域分割時に13%、
Intelで2次元+3次元Type Aで14%



◆ 512ノード(8192コア)利用結果(Intelコンパイラ)



※配列サイズに問題有り。動かなかったため2048コア以上では1024コアの配列サイズを利用している。

2048コア以上利用時に、使用可能メモリの制限？

3次元領域分割TypeAが最速。

8192並列で最大実行効率は13%、4096並列では14%。



◆ 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	2058	6553	31	2次元	ベクトル
SX-8R	8/8	80	282	28	1次元	ベクトル
HA8000	1024/256	1313	9420	14	2次元	Opteron
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

	SX-9	SX-8R	HA8000	HX600	FX1	SR1600
@core	1.0000	0.3100	0.0400	0.0658	0.0632	0.1244
@CPU	1.0000	0.3100	0.1595	0.2631	0.2528	0.2487



◆まとめ

- 1024コアまでは領域分割は3次元ではなく、2次元がもっとも良い性能。
- 最高でIntel製では2次元+3次元TypeAで14%がでた。
- 2000コアを越えると3次元TypeAが最速だが、まだ調べる余地有り。
- x86系であるのでIntel製コンパイラのSSEサポートが強力
- TypeBが効果無いのは、HA8000ではキャッシュにうまく入らない(サイズが効果のあったものと違う)ため？
- スカラ機でのMHDコードは実効性能がHA8000では15%近く、SR16000、FX1、HX600では20%を越える効率が出る。



◆シミュレーションセッティング

□解像度を従来の3倍に

- Grid幅 $0.3R_S \rightarrow 0.1R_S$
- 対流構造、渦構造の解像

□Gridサイズ

- $(nx, ny, nz, nmhd) = (1800, 1216, 608, 8)$ *約42GB
- 1coreあたり $(nx, ny, nz, 8) = (1800, 40, 20, 8)$ を設定。
- 計算上この7倍の配列数を使用する。

□MPI並列

- 1024coreによるFlat MPI、2次元領域分割

□コンパイラオプション

- Intel製: `-O3 -msse3 -xSSE3 -ipo`

□解析

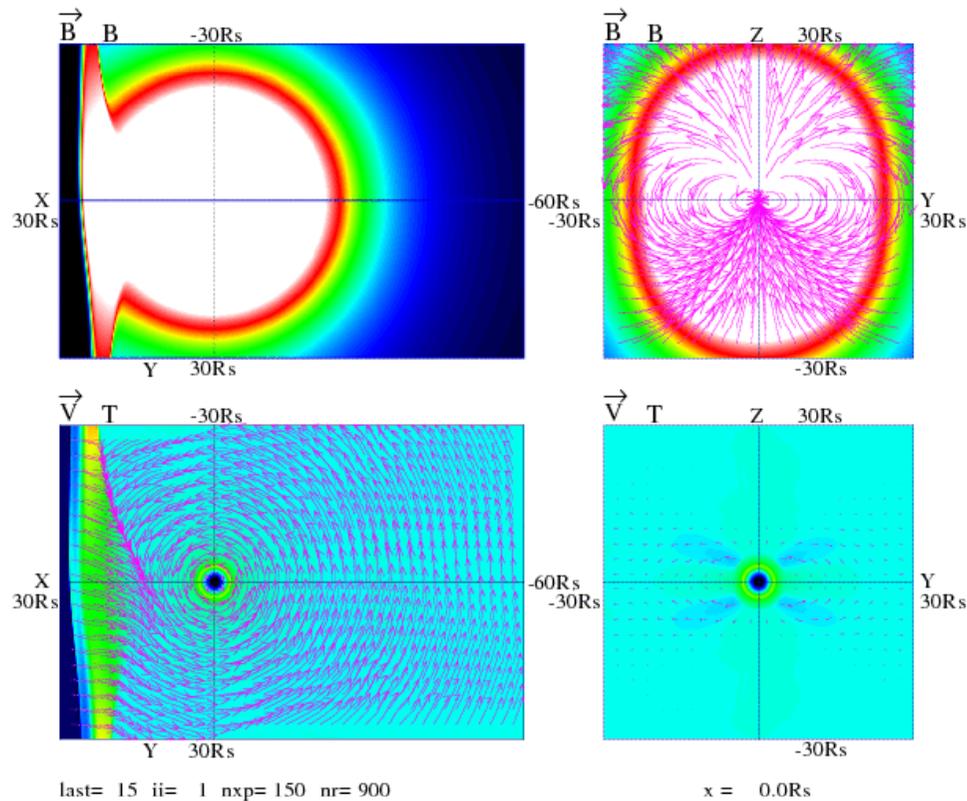
- データが大きいため、各次元1/2にして可視化



◆土星磁気圏対流構造

Saturnian Magnetosphere (Low plasma source)

$B_z = 0.4 \text{ nT}$ $D_{sw} = 0.0082 \text{ nPa}$ $t = 0.7 \text{ hours}$

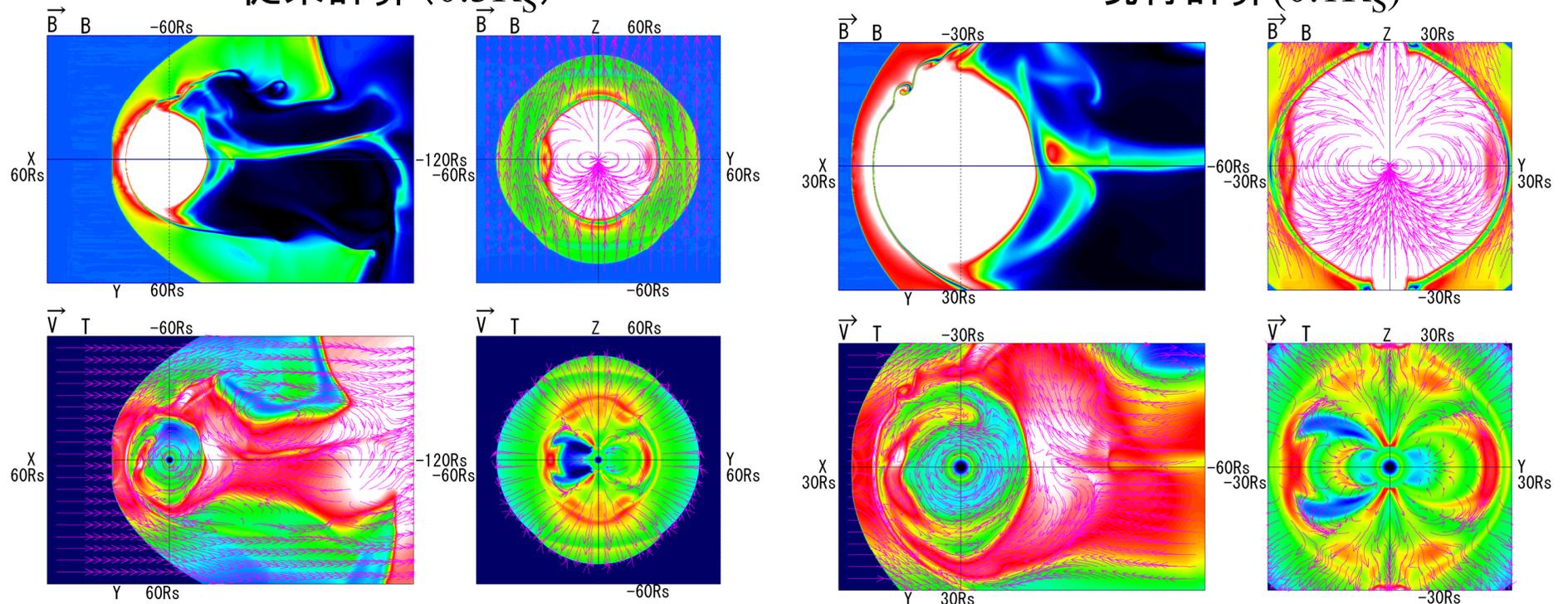


土星磁気圏途中計算結果

◆ 北向きIMF($t = 10h$)

従来計算 ($0.3R_S$)

現行計算 ($0.1R_S$)



$600 \times 400 \times 200$

$1800 \times 1200 \times 600$
 $x = 0.0R_S$

高解像度ではdawn側のmagnetopauseに渦構造が見える

◆ MHDコードの性能評価と磁気圏シミュレーション結果

1. MHDコードでは、1024コアまでは2次元領域分割が早いですが、それ以上のコア数では3次元TypeAが早い可能性有り。
2. 8192コアなどの利用時にはまだ性能測定が足りていないが、3次元TypeAが2048コア時に15%を越える実効性能を出しているため、より詳しく調べる必要がある
3. 高精細土星磁気圏シミュレーションはまだ準定常状態に達せず、継続計算がもう少し必要。ただし、従来とは違った構造は見えてきている。

