

惑星磁気圏超並列高効率 MHD シミュレーションの開発

九州大学情報基盤研究開発センター 深沢 圭一郎

1. Introduction

宇宙空間には太陽から吹いてくる磁場を伴ったプラズマの風と惑星の磁場との相互作用によって形成される惑星磁気圏というグローバルな領域が存在し、そこでは様々な物理現象が起きている。近年地球磁気圏において我々が行い始めた高解像度グローバル地球磁気圏 MHD (電磁流体) シミュレーションにより、今まで解像できていなかったパッチ状のプラズマ流、磁気圏の境界に発生する渦構造などが世界で初めて示され、惑星磁気圏シミュレーションにおいて高精細化が強く必要とされる動機になっている。しかしながら磁気圏という巨大な領域を高精細に解くためには、多くの計算機資源が必要であり、また計算効率を高めるためにシミュレーションコードのチューニングが不可欠であるため、高実行効率、超並列対応シミュレーションコードの開発を目的としている。昨年度では時間の制限もあり、名古屋大学 FX1、HX600 に対してコードの最適化のために性能測定を行った。

2. Simulation Model and Benchmark Model

電磁流体コードの基礎方程式は無衝突 Boltzmann 方程式である Vlasov 方程式のモーメントをとることで求まる MHD (Magnetohydrodynamics) 方程式を利用する。実際の方程式系を式 (1) に示す。上から、質量保存式、運動方程式、エネルギー方程式、磁場の誘導方程式である。詳細な導出は参考文献[1]を参考にされたい。

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

MHD 方程式を解く数値計算法としては、Ogino *et al.* [2]によって開発された Modified Leap Frog 法を使用する。これは最初の 1 回を two step Lax-Wendroff 法で解き、続く $(l - 1)$ 回を Leap-frog 法で解き、その一連の手続きを繰り返す。 l の値は数値的に安定な範囲で大きい方が望ましいので、2 次精度の中心空間差分を採用するとき、数値精度の線形計算と予備的シミュレーションから $l = 8$ に選んでいる。Modified leap-frog 法は、two step Lax-Wendroff 法の数値的安定化効果を一部取り入れて、Leap-frog 法の数値的減衰と分散の小さい効果をより多く取り入れた、数値的減衰と分散にバランスの良くとれた一種の組

み合わせ計算方法となっている。また、パラメータを変化させることによって、性質の良く分かった 2 つの計算方法に一致させることができるので、結果に与える数値誤差の影響も理解し易い利点を持っている。更にこの手法を用いた計算で、今まで様々な計算機で性能評価を行ってきたこともあり、同様の手法をもちいることで、過去の結果と比較できる利点もある。

並列化には MPI を使用する。並列化手法としては 3 次元空間を分割する領域分割法を用いる。領域分割では 2 次元、3 次元の評価を行う。スカラ機で性能を出すにはキャッシュの有効活用が重要である。MHD シミュレーションにおいては、物理変数がプラズマ密度、速度 3 成分、圧力、磁場 3 成分の計 8 変数となる。そのため、配列を $u(i, j, k, m)$ と定義し (Type A)、 $m = 8$ としている。数値計算時に、同じ場所の物理変数を何度も使うことになるので、一般に $u(m, i, j, k)$ と定義した方がキャッシュヒット率は上がると考えられる (Type B) [3]。そのため、本性能評価においてもこの配列定義を使った評価も行う。

計算サイズとして 1 core あたり $u(x, y, z, 8) = 16387064 (127^3 \times 8)$ grid を設定し、weak scaling の測定を行う。また計算上この 7 倍の配列数を実際には使用する。

3. Results

3.1. FX1 の結果

FX1 における性能評価結果を図 1 に載せる。図 1 より、3 次元領域分割 Type B が最も性能が出ていることがわかる。1024 コアを利用した場合、2TFlops を超える実効性能、21%の実行効率を達成した。一方で 3 次元領域分割 Type A は性能が良くなく配列交換の効果が明らかに出ている。

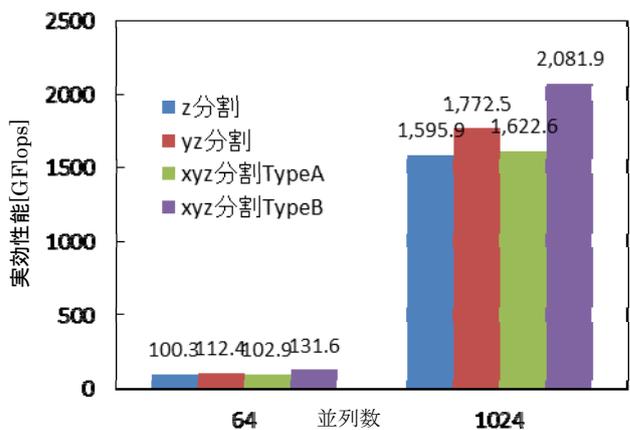


図 1 : FX1 における各領域分割の性能

3.2. HX600 の結果

図 2 に測定結果を示す。HX600 では、3次元領域分割 Type A の性能が良く TypeB が明らかに性能が悪い。これは FX1 とは逆の結果となった。1024 コアで約 2.1TFlops と FX1 に比べわずかに性能が高かった (HX600 と FX1 のコア当たりの理論性能は同等である)。実行効率は 21% を達成した。

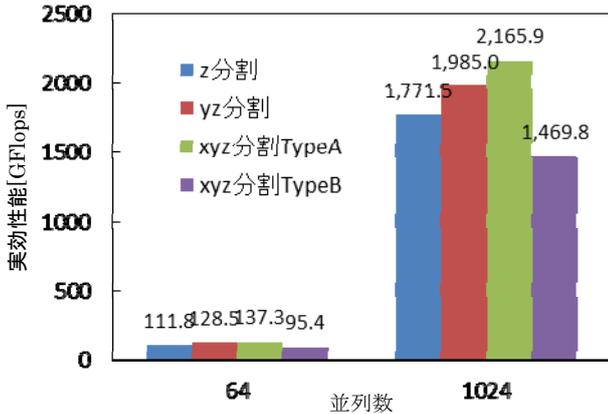


図 2: HX600 における各領域分割の性能

4. 性能プロファイル結果

性能測定の結果 FX1 と HX600 はともにスカラ計算機であるが、CPU のアーキテクチャが異なるため最適化コードに違いが出た。FX1 ではキャッシュヒット率向上が見込めるコード、HX600 ではベクトル計算機向けの最適化が良い性能を示した。それらの違いを見るために、性能プロファイルを取り、コードの振る舞いを調べた。表 1 に FX1 の性能プロファイルを載せる。表中の赤い文字は性能が良い値を示している。各値の詳細な説明は省略するが、TypeB の方が全体的には性能が良い一方で、細部まで見ると TypeA が良いところもある。小を捨てて大をとることもチューニングには必要だが、まさにその結果になっている。ただし、TypeB については性能が出ていない点が多すぎるため、チューニングの余地を感じている。

表 1: FX1 の性能プロファイル結果

	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

表 2 に HX600 のプロファイル結果を載せる。表 2 を見ると、全体的には TypeB の方が良い結果に見える。しかしながら浮動小数点演算では演算器が働き、メモリアクセスも良好な TypeA の方が実効性能では高くなっている。

表 2: HX600 の性能プロファイル結果

	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

5. Summary

本プロジェクトでは 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 Type B が最も性能が良く、1024 コアを利用した場合実行効率 21%、実効性能 2TFlops を達成した。HX6003 では 3次元領域分割 Type A が最高で 1024 コア利用時に 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わかった。

References

- [1] R. O. Dendy, 『Plasma Dynamics』, Oxford University Press, 1990.
- [2] T. Ogino, R. J. Walker, M. Ashour-Abdalla, A global magnetohydrodynamic simulation of the magnetopause when the interplanetary magnetic field is northward, IEEE Trans. Plasma Sci.20, 817.828, 1992.
- [3] Fukazawa, K., et al., Performance measurement of magneto-hydro-dynamic code for space plasma on the various scalar type supercomputer systems, IEEE Transactions on Plasma Science, Vol. 38, No. 9, pp2254, 2010.