

課題番号 JHPCN12-NA04

並列プログラミング言語のシミュレーション宇宙物理学における実践

村主 崇行 (京都大学)

概要

原始惑星系円盤における磁気回転不安定性のなかでの放電現象のシミュレーション、重力波観測のためのテンプレート関数の最適化などの研究を行った。このための基礎として、機械学習や、関数型言語からの並列計算コード生成にまつわる広範囲のサーベイを行った。汎用な最適化のためのライブラリ、非同期分散計算のためのフレームワーク、配列変数上の並列演算のプリミティブセットなどの開発に貢献した。

1. 研究の目的と意義

本研究の目的は、並列計算の必要性の高まりに応じ、近年開発されてきた、またはされている様々な並列計算用プログラミング言語を、実際の応用へと結びつけることである。とくに本研究の代表者である私は、偏微分方程式の陽解法を、数学的記法をもちいて簡潔に記述し、コード生成と最適化を行うドメイン特化型言語である Paraiso を開発している。そこで、Paraiso によるアルゴリズム定義から複数の並列計算用プログラミング言語のソースコードを生成し、私が専門とする宇宙物理学のシミュレーションに使うことを目標とする。

ところで、本研究や Paraiso に関する発表をするうちに、次のようなアドバイスをたびたびいただいた。

- もっと共同研究者を増やしたほうがいい。
- 現行の Paraiso において、コード生成元のアルゴリズムに少しでも変更があると、時間のかかる自動チューニングをやり直すことになるから、機械学習などを取り入れてチューニング成果の再利用が図れないか。
- 最適化という新しい分野に打って出るには情報論の基礎が必要だし、物理を続けるにしても、もっと基礎から勉強した方が良い。

このような条件をみたしつつ、短期間でも成果をだすための方法を求め続けてきた。本年度前半は

おもに上記並列プログラムの生成や最適化のためのツールの開発を行った。これらのツールのいくつかは公開しており、プログラミング言語の研究者や、最近関数型プログラミングを勉強しはじめたという産業界の方々にも興味をもっていただいている。

2. 当拠点公募型共同研究として実施した意義

(1) 共同研究を実施した大学名と研究体制

当研究は、おもに東京工業大学の超大規模 GPU 並列計算機 TSUBAME2.0 および京都大学白眉センターの小規模 GPU クラスタ計算機を利用して行った。

(2) 共同研究分野

当研究の共同研究分野は超大規模数値計算系応用分野であった。本研究は、宇宙物理学およびコード生成分野として私が、流体計算分野担当として東京工業大学の青木尊之が、並列計算言語分野として東京工業大学の丸山直也が、宇宙物理学分野として名古屋大学の犬塚修一郎が共同して進めている(所属は申請時点)。

このほか、2012 年 5 月の Shonan Meeting や 9 月の ICFP (International Conference on

Functional Programming)では、プログラミング言語研究者からさまざまな指導をいただくことができた。同じく 8-9 月にかけて開催された機械学習夏の学校では、数多くの機械学習研究の現状のレクチャーを受けることができた。中でも産総研の津田宏治先生のグラフ最適化の研究や、数理研の岩田覚先生の確率的劣モジュラ最適化の研究を、プログラムのデータフローグラフの性能予測や最適化にむけて利用を図っている。

(3) 当公募型共同研究ならではの事項など

東京工業大学のスパコン、TSUBAME 2.0 は、国内で利用可能な GPGPU スパコンとしては比類ない規模の申し分ない環境である。また、ログインノードのみならず計算ノードでも Haskell コンパイラが動作するなど、マルチコア・LLVM・GPU コードの生成、fusion、ベンチマークなどの研究にはうってつけの環境であった。また、要望に応じて特定 Web サイトへのアクセス許可をいただけることから、github, bitbucket 等のレポジトリホスティングサービスと連携し、コミュニティーベースで開発されているコードを利用・開発に参加できる点も大変ありがたかった。

3. 研究成果の詳細と当初計画の達成状況

(1) 研究成果の詳細について

原始惑星系円盤の雷現象の研究

原始惑星系円盤は惑星形成の舞台となる天体であるが、その駆動源となっているのが磁気回転不安定性 (MRI) である。ここで、本来 MRI の存在に適さない条件下でも、ひとたび発生した MRI が雷放電によりプラズマ電離率を高く保ち、MRI が自己維持されるという現象がある。本年度は TSUBAME2.0 の並列計算性能、およびその上に構築した自家製タスク管理システム(後述)を利用し MRI の自己維持現象のサーベイ的研究を行い今まで知られていなかった新たな発生条件を発見した(文献[1])。またこの研究と、私が過去に行った

氷ダストを原因とする雷の発生条件の研究や、差動回転する円盤における雷の形状のシミュレーション研究などの成果をまとめ、学位申請論文を執筆し受理された(文献[2]、図 4)。

重力波天体の観測のための研究

京都大学基礎物理学研究所の柴田大先生のグループと共同で、彼らのシミュレーションにより得られた重力波シグナルを説明し、強烈なノイズにうもれた重力波信号中から極限状態物質の状態を読み取るのに必須のフィッティングモデルの構築を TSUBAME2.0 を利用して行った(図 5)。これには後述の CMA-ES アルゴリズムおよび遺伝的アルゴリズムによる部分最適化戦略の獲得を利用した。

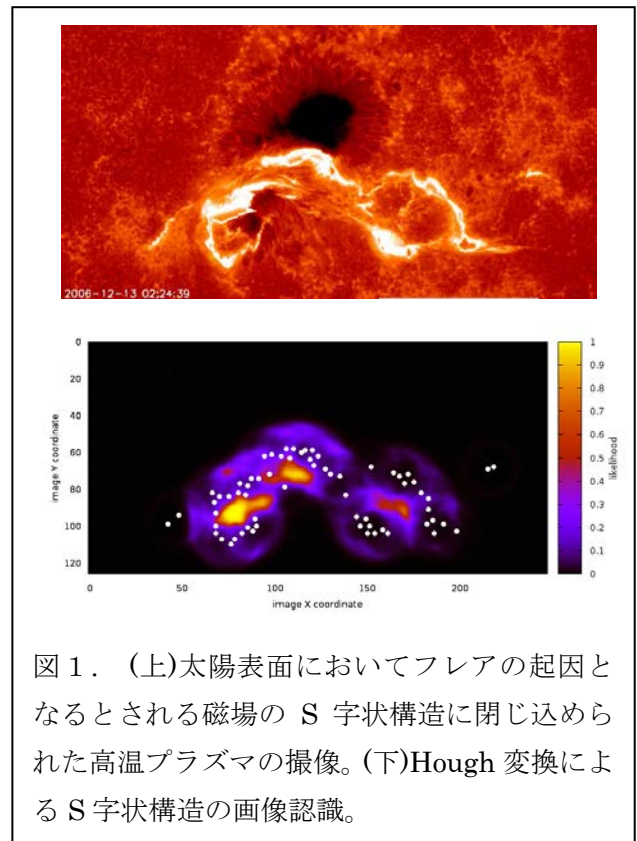


図 1. (上)太陽表面においてフレアの起因となるとされる磁場の S 字状構造に閉じ込められた高温プラズマの撮像。(下)Hough 変換による S 字状構造の画像認識。

宇宙天気予報のためのフレア予測の研究

京都大学宇宙物理学教室の柴田一成先生のグループ、および株式会社ブロードバンドタワーと共同で、ビッグデータ解析を利用した太陽地球環境の研究をはじめた。一年あたり 3.4 ペタバイトにもなる太陽画像データを分散計算環境を利用して分析し、モデルを構築し、数日将来の太陽活動を

表 1. CMA-ES の Haskell インターフェイス

```

minimize :: ([Double] -> Double) -> [Double] -> Config [Double]
minimizeT :: Traversable t => (t Double -> Double) -> t Double -> Config (t Double)
minimizeG :: Data a => (a -> Double) -> a -> Config a
run :: forall tgt. Config tgt -> IO tgt
    
```

予測するためのプログラムを Haskell で書いている (図 1)。

などを最適化するプログラムを、ただ 1, 2 行で記述できる。

確率的最急降下法の高抽象度インターフェイス

CMA-ES (Covariance Matrix Adaptation Evolution Strategy) は確率的最急降下法の一つで、INRIA の Nikolaus Hansen らが開発している汎用的な最適化アルゴリズムである。CMA-ES は、フィット対象の関数が非凸であったり、noisy であったりする場合にも強く、またパラメータ空間の線形変形に対して結果が不変である等のロバストな性質をもつことから、「最適化の初手」としてよく用いられているアルゴリズムである。

私はこのアルゴリズムの Haskell インターフェイスを作った [12]。表 1. にあるように

- 実数のリストから実数への関数
- 実数に関して Traversable な構造から実数への関数
- 実数を含む Generic な値から実数への関数

P2P ネットワーク上の型付ソケットと状態共有

- 適合格子計算のような、計算領域が均一でない系のシミュレーションをロードバランスさせること。
- ベンチマークベースの最適化のためにさまざまな実行速度を持つプログラムを生成・実行し、計測情報を回収すること。

以上のような目的には、非同期並列計算が必要である。さらに遠隔ノードへとロードを移行したり、遠隔ノードで計測プログラムを起動したりするためには、継続(Closure)や、型情報のついた値を送受信できる分散計算フレームワークがあると便利である。

近年さかんに開発されている Haskell のライブラリの一つに Cloud Haskell がある。これは、分散環境において実績のあるプログラミング言語 Erlang 相当の機能を Haskell に取り込もうというものである。

私は Cloud Haskell に対し、その上で実装され、Peer-to-Peer 方式で状態の共有を実現する Distributed-process-state を貢献した [5]。表 2 は、Cloud Haskell で可能な操作を可視化するためのダイアグラムであり、ここで、☁️ 記号で表現されているのが、状態を共有するための P2P ネットワークである。図 2 は、Cloud Haskell を利用して設計した分散計測システムの例である。

TSUBAME2.0 上において、IP over Infiniband 上や、異なるユーザ間でも Peer-discovery が機能し状態共有ネットワークを組めることを確かめている。

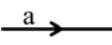

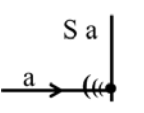
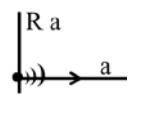
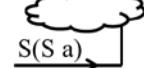
	a 型の値
	a 型のチャンネルの送信端 S a と受信端 R a の対生成
	送信端 S a からの a の送信
	受信端 R a からの a の受信
	Serializable な値のクラウドへの公開

表 2. distributed-process diagram の表現とその意味

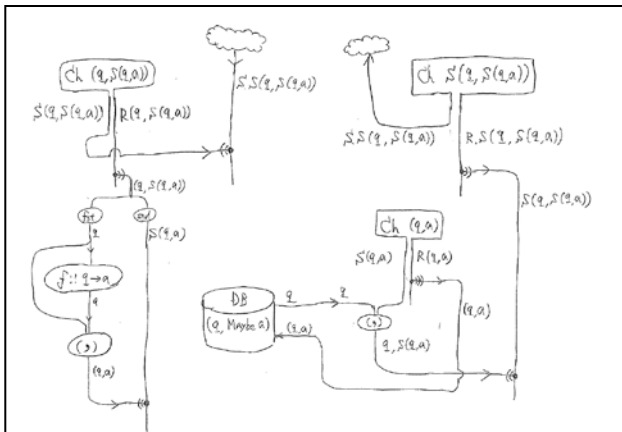


図 2. q 型の値を a 型の値に変換する関数 f を分散計算するサーバ・クライアントシステム的设计例。distributed-process では、送信ポートを含む高階型を送信できることから、サーバ(右側)はクラウドに $S(S(q, S(q, a)))$ 型の値を公開し、クライアント(左側)がそれを拾うだけで、以降のすべての通信を確立させることができる。

variable-arity zipWith

ICFP2012 では、Haskell の GPGPU バインディングである accelerate の開発者や、Nested Data Parallelism の研究者を中心に、データ並列演算のプリミティブ・セットの選定や、それらの fusion の実装について議論が交わされた。その ICFP2012 のホット・トピックの 1 つは、Glasgow Haskell Compiler の最新版 (7.6.1) で、可変引数 (variable-arity) zipWith をいかに実装するかであった。zipWith とは、複数個の配列変数に対し、同じ添字位置の要素どうしに演算を施す基本的な操作である。可変引数 zipWith は、 f を配列変数の型コンストラクタとして、以下のような型をとりうる多相な関数である。

```
repeat :: a -> f a
map :: (a -> b) -> f a -> f b
zipWith :: (a -> b -> c) -> f a -> f b -> f c
zipWith3 :: (a -> b -> c -> d) -> f a -> f b -> f c -> f d
zipWith4 :: (a -> b -> c -> d -> e) -> . . .
```

通常、それぞれの関数は別物とみなされ、別個の名前が与えられている。しかし私は、型クラスや型族などの機構を使い、以上全てを一気に引き

受けうる単一の関数を作れることを示した (参考リンク [14]) さらに、Haskell 上の GPGPU プログラミング・フレームワークである accelerate にも拡張すべく、Oleg らと検討を進めている。zipWith をはじめとする並列計算プリミティブでプログラムを記述すれば、REPA と accelerate など、複数のフレームワークを抽象化したプログラムを書くことができる---コードを生成し分ける必要すらなく、同一のプログラムから複数のフレームワークを利用できるのである。こうして、並列計算言語同士の比較という当初の目標に近づくことができる。

(2) 当初計画の達成状況について

当初計画では、GPU 大規模並列計算の上に、磁気流体方程式などの高度な偏微分方程式の陽解法ソルバを乗せ、宇宙物理に資する大規模計算、とくに MRI を分解する解像度での原始惑星系円盤の大局シミュレーションを実現する予定であったが、を行う予定であった。MRI と相互作用しうるさまざまな物理素過程の研究は進んだものの、今年度中に、件の大目標は果たせなかった。この原因とは、さまざまな相互作用とカップルした磁気流体方程式の離散化はまだ完全に既知とはいいがたく、結果として本研究代表者の能力とプログラム自動生成によっては数値不安定性を排除したコードを作れなかったことにある。

しかしながら、限られた時間のなかで成果を出すべく積極的に課題を探しにゆき、結果として 2 本の宇宙物理分野の学術論文および学位申請論文を出すなど、一定の成果を上げることはできた。

4. 今後の展望

私は気づいてしまったのだが、そもそも対象となる系を大解像度で表現し、過去から未来へと時間積分する、というアプローチには

```

spec :: Spec
spec = describe "Bonner Ebert Sphere by " ++ citet bonner1956 ++ " & " ++ citet ebert1955 $ do
  it "temperature is 10K" $ do
    soundSpeed `is0fOrder0f` (100 *| meter |/) second)
  prop "radius is positive" $
    \xyz -> xyzToR xyz >=| (0::Double) *| astronomicalUnit

  prop "has correct gravitational potential 2" $
    \xyz ->
      let
        den' :: Vec3 (Double | Meter) -> Value Density '[ (Meter, NThree), (Kilo Gram, POne)] Double
        den' = gravitationalPotentialToDensity (bonnerEbertPotential soundSpeed . xyzToR)
        den = bonnerEbertDensity soundSpeed . xyzToR

      in
        xyzToR xyz >| (0 *| meter) ==>
          abs(den' xyz |-| den xyz) |<=| 1e-10 *| den xyz

  prop "is hydrostatic" $
    \xyz ->
      let
        pre :: Floating f => Vec3 (f | Meter) -> Value Pressure (U Pascal) f
        pre = (to pascal) .(|*| square soundSpeed) . bonnerEbertDensity soundSpeed . xyzToR
        den :: Vec3 (Double | Meter) -> Value Density '[ (Meter, NThree), (Kilo Gram, POne)] Double
        den = bonnerEbertDensity soundSpeed . xyzToR

        acc :: (f ~ Double) => Vec3 (f | Meter) ->
              Vec3 (Value Acceleration '[ (Second, NTwo), (Meter, POne)] f)
        acc = (fmap $ to (meter `per` square second)) . pressureToAcc pre den

        acc' :: (f ~ Double) => Vec3 (f | Meter) ->
              Vec3 (Value Acceleration '[ (Second, NTwo), (Meter, POne)] f)
        acc' = (fmap $ to (meter `per` square second)) . grad (bonnerEbertPotential soundSpeed . xyzToR)

      in
        xyzToR xyz >| (0 *| meter) ==>
          foldl1 (|+|) (compose $ \i -> abs ((acc xyz | i) |+| (acc' xyz | i)))
          |<=|
          foldl1 (|+|) (compose $ \i -> 1e-10*| abs (acc xyz | i))

hydrostatic ::
forall x
dimLen dimPre dimDen dimAcc dimGpr dimNegLen dimNegDen dimAcc'
uniLen uniPre uniDen uniAcc uniGpr uniNegLen uniNegDen uniAcc'
( Fractional x
, dimLen ~ LengthDimension
, dimPre ~ Pressure
, dimDen ~ Density
, dimAcc ~ Acceleration
, dimNegDen ~ [ '(Length, PThree), '(Mass, NOne) ]
, dimGpr ~ [ '(Length, NTwo), '(Mass, POne), '(Time, NTwo) ]
, Convertible' dimLen uniLen
, Convertible' dimPre uniPre
, Convertible' dimDen uniDen
, Convertible' dimAcc uniAcc
, Convertible' dimAcc' uniAcc'
, MapNeg dimLen dimNegLen
, MapMerge dimPre dimNegLen dimGpr
, MapNeg dimDen dimNegDen
, MapMerge dimGpr dimNegDen dimAcc'
, MapEq dimAcc' dimAcc
, MapNeg uniLen uniNegLen
, MapMerge uniPre uniNegLen uniGpr
, MapNeg uniDen uniNegDen
, MapMerge uniGpr uniNegDen uniAcc'
) =>
(forall s. AD.Mode s =>
Vec3 (Value dimLen uniLen (AD s x)) -> Value dimPre uniPre (AD s x))
-> (Vec3 (Value dimLen uniLen x) -> Value dimDen uniDen x)
-> (Vec3 (Value dimLen uniLen x) -> Vec3 (Value dimAcc uniAcc x))
-> (Vec3 (Value dimLen uniLen x) -> Vec3 (Value dimAcc uniAcc x))

hydrostatic pressure density externalAcc r
= compose $ \i -> to (undefined :: Value dimAcc uniAcc x) $
{externalAcc r | i} |+| (gradP r | i) |/) (density r)

where
gradP :: Vec3 (Value dimLen uniLen x) -> Vec3 (Value dimGpr uniGpr x)
gradP = grad pressure
    
```

図 3. Physics-in-Haskell のソースコードからの抜粋 (左) 出典文献情報、ランダムサンプルテストによる検証が組み込まれた、天体モデルの Haskell 上での表現。(右) 型レベル次元付き演算による証明つきの、物理法則の Haskell 上での記述。

- 新しい方程式に対し数値不安定性を取り除くのが困難で、しばしば異様に小さな時間刻みや、NaN の出現に悩まされる。
- 安定なアルゴリズムが作れたとしても、時間刻みの制限により長時間積分するほど現実の時間も消費するため、系の長い時間発展を追うのが苦手

という共通の短所があるのではないか。無論、多くの人が努力によりこれらの短所を乗り越えられて成果を挙げているのは承知の上だけれども。

第一の点に関しては、ランダムテストにより間違いの確率を 0 に近づけること、型レベル次元付き演算やプログラムに対する定理証明などにより、あるカテゴリの論理バグを根絶することなどを通じて、数値不安定性の原因となりうる誤ちを 1 つ 1 つ潰していこう (参考文献[11]、図 3)。

第二の点に関していえば、原始惑星系円盤の大局的 3 次元シミュレーションが、現状の世界的に

最先端のものでも 1000 周期程度にとどまっていること、それに対し原始惑星系円盤の寿命は数百万年と目されている。ということは、原始惑星系円盤の全寿命を解き明かすには今の 1000 倍規模の時間刻みが必要ということになり、これはいかにこの先計算機の並列化が進んでも当分の間は実現しそうにない。

そもそも、原始惑星系円盤の歴史を解き明かしたいという本分野の究極の動機に対して、そのような大解像度計算の陽解法によるアプローチがはたして向こう 10 年といった期間に十分な貢献ができるのかは疑問となってきた。

いっぽうで、今年度までの経験から、純粋関数型言語は、粒度は荒いけれども依存関係は複雑なタスクの並列化や、関数空間など複雑な構造をもった対象の最適化問題を記述するには向いていることがわかってきた。

このことから、最近私は、偏微分方程式の求解問題を、最小作用形式に書き直し、関数解析に基づいた、時空間最適化問題として解く方法に注目

している。これには、従来の非メッシュ解法の開発の指導原理として用いられてきた G ノルム最適化を時空間に拡張するアプローチ、分子生物学にて同様に時間の壁を乗り越えるために用いられている Generalized Ensemble 法の応用、また分散計算が必要なほどの大規模データの最適化問題を部分最適化問題の組に帰着する Incremental Gradient Descent 法などのアプローチおよびその組み合わせが考えられる。

まず解が求まり、次にそれを最適化するのだ。ノーベル物理学賞受賞者アンソニー・レゲット教授が指摘する物理次の 100 年の課題、「時間の矢」の常識に、シミュレーションの観点から挑戦するのも愉快ではないか。

5. 研究成果リスト

(1) 学術論文 (投稿中のものは「投稿中」と明記)

- [1] “*Interdependence of Electric Discharge and Magnetorotational Instability in Protoplanetary Disks*”, T. Muranushi, O. Satoshi and I. Shu-ichiro (2012), *The Astrophysical Journal*
- [2] “*Lightning in Protoplanetary Disks*,” T. Muranushi (2013), Dissertation for the Degree of Doctor of Philosophy, Kyoto University
- [3] “*Remnant massive neutron stars of binary neutron star mergers: Evolution process and gravitational waves*”, Kenta Hotokezaka, Kenta Kiuchi, Koutarou Kyutoku, Takayuki Muranushi, Yu-ichiro Sekiguchi, Masaru Shibata, and Keisuke Taniguchi *Physical Review D* (投稿中)

(2) 国際会議プロシーディングス

(3) 国際会議発表

- [4] T. Muranushi, “Global, 3D, 2π simulations of MRI in protoplanetary disks --- current state and the future.” *Extrasolar*

Planets. Kyoto. April 18, 2012

(4) 国内会議発表

- [5] 村主崇行「偏微分方程式の陽解法のための自動チューニングフレームワーク」GTC Japan, Tokyo July 26, 2012 (招待講演)
- [6] 村主崇行「偏微分方程式の陽解法のための自動チューニングフレームワーク」GTC Japan, Tokyo July 26, 2012 (招待講演)
- [7] K. Hotokezaka, K. Kiuchi, K. Kyutoku, T. Muranushi, Y. Sekiguchi, M. Shibata and K. Taniguchi. “Constructing Gravitational Wave Signal Templates: Neutron Star Merger Remnants.” *Innovative Areas Meeting*. Atami, Shizuoka. January 9, 2013
- [8] 村主崇行、犬塚修一郎、奥住聡、高橋実道、富康達也「原始惑星系円盤の一生と雷現象」大阪大学宇宙進化セミナー、2013 年 2 月 18 日 (招待講演)

(5) その他 (特許, プレス発表, 著書等)

- [9] Miran Lipovaca 著、田中英行・村主崇行 (共訳) 2012『すごい Haskell たのしく学ぼう!』オーム社、401 ページ
- [10] <http://hackage.haskell.org/package/Paaiso> : 並列プログラムの自動生成・自動チューニングフレームワーク
- [11] <https://github.com/nushio3/physics-inhaskell> : 物理的知識の証明、文献情報を伴う Haskell による表現
- [12] <http://hackage.haskell.org/package/cmaes> : 汎用最適化アルゴリズム CMA-ES の Haskell ラッパ
- [13] <http://hackage.haskell.org/package/dynamic-object> : Haskell における動的オブジェクト指向プログラミング
- [14] <https://github.com/nushio3/zipWithN> : N 個のコンテナ型コンストラクタに対する可変長引数関数適用の一般的な表現

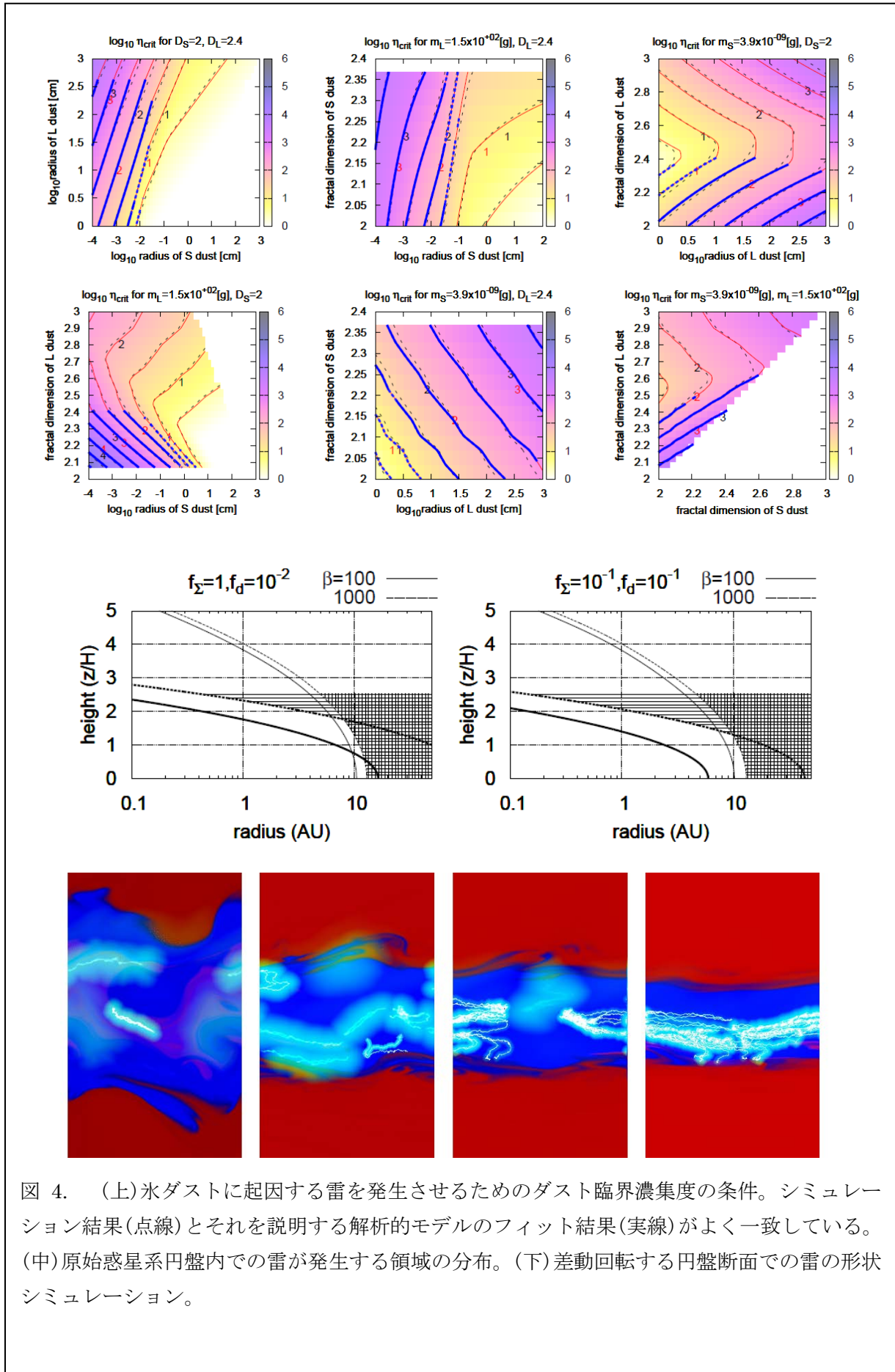


図 4. (上)氷ダストに起因する雷を発生させるためのダスト臨界濃集度の条件。シミュレーション結果(点線)とそれを説明する解析的モデルのフィット結果(実線)がよく一致している。(中)原始惑星系円盤内での雷が発生する領域の分布。(下)差動回転する円盤断面での雷の形状シミュレーション。

