

# GPUを用いた 反応拡散系パターンダイナミクスの演算と表示の高速化

手山 奈緒子<sup>†</sup> 安藤 祥子<sup>‡</sup> 村木 茂<sup>§</sup> 藤代 一成<sup>‡</sup>

お茶の水女子大学 理学部 情報科学科<sup>†</sup>

お茶の水女子大学 大学院 人間文化研究科<sup>‡</sup>

産業技術総合研究所 ポリウムグラフィックス連携研究体<sup>§</sup>

## 1 背景と目的

従来の3次元グラフィックス処理は、CPU (Central Processing Unit) で物体の形状の定義や、ジオメトリ処理などの計算処理を行い、3次元グラフィックスボードで最終的なレンダリング処理を行っていた。近年3次元グラフィックスは、エンターテインメントだけでなく科学技術などの広い分野で3次元可視化技術としても急速に普及している。それに伴い、ただフォトリアリスティックな画像を作成するだけでなく、データ特有の可視化法を使い、より高速に表示することが求められるようになった。このような要求から開発されたのがプログラマブルGPU (Graphics Processing Unit) [1] である。

GPUは、そのプログラミング可能性により、グラフィックス処理だけでなく、計算処理にも利用可能である。格子構造をもった2次元データや3次元ボリュームデータなどは、画像と同じデータ構造である。これらのデータは、GPUのプログラマブルなピクセル描画機能を利用して計算処理を行うことができる。これにより、CPUでの計算処理量を削減することができ、全過程の処理高速化において、ひじょうに効果的である。GPUは、今やPCやゲーム機だけでなく携帯電話にまで搭載され始めており、CG分野での期待は大きく、研究も進んでいるが、応用面での可能性はまだ多く残されている。

本研究では、CPUとGPU上で単純な数値モデルを2次元表示し、その処理時間を比較する。またCPUの性能を最適化するためにSSE2[2]の適用を検討する。数値モデルには反応拡散系パターンダイナミクスを表すグレイ-スコットのモデル[3]を採用した。

## 2 グレイ-スコットモデル

自然界には、雪の結晶、動物の表皮の柄や水面を揺らしたときの干渉縞などのように、自己組織構造や時空間的秩序をもって変化していく構造がある。これらはパターンダイナミクスとよばれている。

グレイ-スコットモデルは、化学反応と拡散におけるパターンダイナミクスの一例である。1つのスポットが自己分裂を続け、領域が埋め尽くされた後は消滅と分裂を繰り返すパターンをもつ。このモデルは以下のような反応拡散方程式で表される：

$$\frac{\partial u}{\partial t} = D_u \Delta u - uv^2 + F(1 - u)$$

$$\frac{\partial v}{\partial t} = D_v \Delta v + uv^2 - (F + k)v$$

ここで、 $u$ 、 $v$ は反応物質の濃度、 $D$ 、 $F$ 、 $k$ はそれぞれ拡散係数、反応物の流入流出時間、反応速度を示す。

## 3 実装

CPUとGPU上で、グレイ-スコットモデルの計算と表示のプログラムを実装し、そのダイナミクスの様子を観察した。Dの値を変化させると異なるパターンが得られた。またDの値を固定してFやkの値を変化させると、パターンの変化や反応の遅れが見られた。図1にパターン例を示す(a)(b)(c)は、 $D_u=0.065$ 、 $D_v=0.0325$ としたときの時間変化による拡散の様子を示し、(d)、(e)(f)は、拡散係数を $D_u=0.095$ 、 $D_v=0.0125$ と変化させたときのパターンを示している。そしてそれぞれの処理時間を計測し、比較した。

実装環境には、単体のPC (CPU: Pentium4, 2.8GHz, OS: RedHat Linux) を使い、C言語でプログラミングを行った。計算結果はpgm形式でファイルに出力して表示した。また、CPU上での演算の高速化を図るために、反復演算の一部にアセンブラ言語を使い、SSE2を適用した。一方、GPUには代表的なプログラマブルGPUであるGeForceFX5900 (256MB) を使い、C++とCg[4]を用いてプログラミングを行い、OpenGLを用いて同時に描画を行った。

反応領域は128 × 128とし、処理時間の比較には10000回の演算とその表示を対象とした。CPUによ

Accelerating Computation and Visualization of Reaction-Diffusion System Pattern Dynamics with GPU

<sup>†</sup>Naoko Teyama, Department of Information Sciences, Ochanomizu University

<sup>‡</sup>Shoko Ando, Issei Fujishiro, Graduate School of Humanities and Sciences, Ochanomizu University

<sup>§</sup>Shigeru Muraki, Collaborative Research Team of Volume Graphics, National Institute of Advanced Industrial Science and Technology

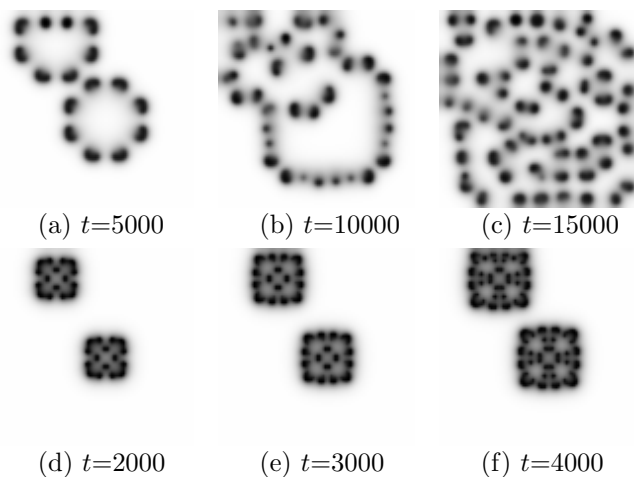


図 1: グレイ-スコットモデルのパターン例

る全処理, GPU も利用した全処理の時間をそれぞれ表 1 に示す. GPU を併用することにより, CPU だけの処理時間を約半分に短縮できたことがわかる.

表 1: 10000 回の演算と表示における処理時間の比較

	平均処理時間 (CPU 秒)
CPU	約 13.6
GPU	約 6.3

## 4 まとめと今後の課題

### 4.1 まとめ

グレイ-スコットモデルは, パラメタ値を変化させることで複数の異なるパターンを生成する. しかし, 4 つのパラメタ値を変化させ, それぞれのパターンを検証するためには, 処理の反復とそれに伴う膨大な時間が必要である. 莫大な回数の数値計算を必要とするシミュレーションの効率を上げるために, 処理の高速化は必須である. そこで本稿では, GPU による計算と表示の高速化の効果を検証した.

CPU と GPU 上での演算と表示の処理時間を比較し, その結果から GPU の有効性を確認することができた. CPU での処理の高速化には今回適用したようなアセンブラ言語を用いた解決策がある. しかし, GPU での処理は高速化だけでなく, Cg を用いることでより短いプログラムで実行できるという利点がある. よって GPU の使用はより効果的な解決策であると考えられる.

### 4.2 今後の課題

本グループでは, MRI の一種である拡散強調画像から得られる脳の拡散テンソル場を効果的に可視化法する拡散トラクトグラフィ法 (Diffusion-Based Tractography: DBT) [5] を提案している. DBT 法は拡散シミュレーションを使って, ランダムドット画像をぼかし, 神経軸索にそった水分子の拡散を可視化するので, 密な神

経走行の可視化や, 神経線維の交差を反映した扁平な拡散の追跡が可能であると考えられる. 図 2 に DBT 法の可視化結果を示す.

この DBT 法では, 拡散処理の反復において拡散方程式による拡散シミュレーションを使っており, 今回の計算とも類似性がある. グレイ-スコットモデルにおける GPU 利用の有効性の検証は, DBT 法の高速化への布石となる.

最終的には, この DBT 法を GPU 上で実装し, 拡散テンソル場の対話的な可視化環境の構築を目指す. また, 局所演算による拡散シミュレーションは並列処理に適している. よって本グループで開発中の, 高並列可視化システムである VG (Volume Graphics) クラスタ [6] を用いることで, さらに高速化を実現する予定である.

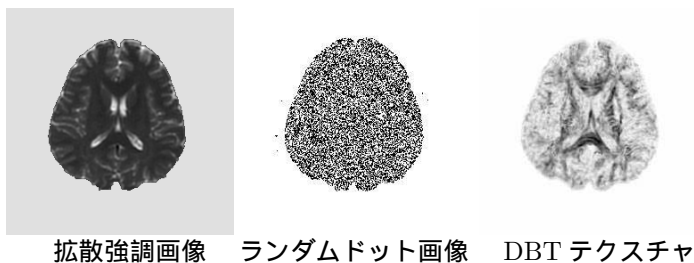


図 2: 拡散トラクトグラフィ法による可視化結果

## 謝辞

本研究は科学技術振興事業団計算科学技術活用型特定研究開発推進事業「広域ビジュアルコンピューティング技術」の成果の一部である.

## 参考文献

- [1] <http://pc.watch.impress.co.jp/docs/article/990902/idf02.htm>
- [2] インテル (R) Pentium (R) 4 プロセッサおよびインテル (R) Xeon (TM) プロセッサ最適化リファレンス・マニュアル (日本語)
- [3] 西浦 廉政: 自己複製と自己崩壊のパターンダイナミクス, 岩波書店, 2003 年
- [4] Randima Fernando, Mark J. Kilgard 著, 中本 浩訳: The Cg Tutorial 日本語版, 東京印書館, 2003 年 8 月
- [5] 安藤 祥子, 村木 茂, 藤代 一成, 「密な神経走行を強調する拡散テンソル場の可視化法」, 電子情報通信学会技術研究報告, MI2003-43, 2003 年 9 月
- [6] S. Muraki, M. Ogata, K. Kajihara, K-L. Ma, K. Koshizuka, X. Liu, Y. Nagano, K. Shimokawa, "Next-generation visual supercomputing using PC clusters with volume graphics hardware devices," In *Proc. IEEE SC2001*, November 2001.