

3次元 Span Space を用いた半透明複数等値面の高速生成

平野未来* 伊藤貴之** 白山晋*

*東京大学大学院工学系研究科環境海洋専攻

**お茶の水女子大学 理学部情報科学科

E-mail: micoo@race.u-tokyo.ac.jp

等値面とは、ボリュームデータから一定のスカラー値を有する点の集合で、一般的には曲面となる。異なるスカラー値から複数枚の等値面を生成し、これを半透明にして重ね合わせ表示することにより、ボリュームレンダリングにある程度近い効果を得ることが可能であり、ボリュームデータのスカラー場の概要を知るための可視化手法として用いることができる。

本研究は、ボリュームデータ中の3次元スカラー場の全体像を高速に可視化する一手法として、半透明複数等値面の高速生成手法を提案する。等値面生成の高速化手法の代表的なものに、ボリュームデータ中のセルを、そのスカラー値の最小値と最大値を座標軸とする空間(Span Space)に配置するISSUEという手法がある。本手法はこの拡張手法であると考えられる。本手法ではSpan Spaceに、「視点からの距離」という変数を加えた3次元Span Spaceを適用する。この空間上にて、複数枚の等値面のいずれかに交差するセルを、視点から遠い順に抽出することで、半透明複数等値面の高速生成を実現する。

Speeding Up the Generation of Semi-Translucent Multiple Isosurfaces Using 3D Span Space

Miku Hirano* Takayuki Itoh** Susumu Shirayama*

*Dept of Environment and Ocean Engineering, Faculty of Engineering, University of Tokyo

**Dept. of Information Sciences, Faculty of Science, Ochanomizu University

Isosurface is a set of points where scalar values are constant, and it generally forms surfaces. Simultaneous display of multiple semi-translucent isosurfaces on a screen is useful for visualizing large-scale 3D data. It realizes comprehensive representation of the 3D scalar fields, as well as volume rendering techniques.

In this paper, we focus on visualization of 3D scalar field composed of volume data by means of semi-translucent multiple isosurface plots. A new method for displaying semi-translucent multiple isosurfaces will be proposed to fulfill realtime rendering of scalar fields. ISSUE is a known algorithm to quickly generate isosurfaces. ISSUE uses 2D span spaces where minimum values of elements are assigned to horizontal axis, and maximum values of elements are assigned to vertical axis. In our algorithm, elements of volumes are placed onto a 3D span space. The third axis is the distance from the view point. Our algorithms can quickly search for elements intersecting with isosurfaces using the span space in the descending order of distances from the view point, and realize representation of semi-translucent multiple isosurfaces.

1. はじめに

等値面とは、ボリュームデータから一定のスカラー値を有する点の集合を抽出して形成される曲面であり、ボリュームデータの可視化のための代表的手法の一つである。複数のスカラー値を指定して複数の等値面を生成し、これらを半透明表示して重ね合わせる（図 1(左)参照）ことで、ボリュームレンダリングのようにボリュームデータの概略的な数値分布を表現することが可能になる。図 1(右)の画像と比較しても、半透明複数等値面の重ね合わせ表示により、ある程度ボリュームレンダリングに近い効果を得ることが可能であることがわかる。

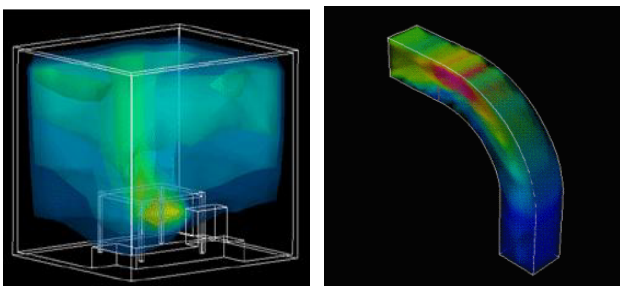


図 1. (左)半透明複数等値面の画像例。(右)ボリュームレンダリングの画像例。

近年の GPU 技術の発達により、ボリュームレンダリングのほうが等値面生成よりも高速に可視化を実現できる、という事例が生じ始めている。しかしそれでも、現在においても、等値面生成に関する研究事例は依然として多い。等値面生成の利点はいくつか考えられるが、本質的な利点の一つとして、ボリュームデータのセル数を N としたときに、等値面生成は平均 $O(N)$ 未満の計算時間で実現可能であり、また等値面生成結果であるポリゴンデータの容量も $O(N)$ より小さい、という点がある。よってボリュームデータが大規模になればなるほど等値面生成のほうが、計算量の観点で有利になること、またその結果の記憶媒体への保存やネットワーク転送において有利になること、が考えられる。

等値面を高速生成する手法は、旧来から非常に多く報告されている。その代表的なものを分類すると、以下ようになる。

[分類 1] 等値面と交差するセルを高速抽出する技術。言い換えれば、等値面と交差しないセルとの処理を省略する技術。代表的なものとして、各セルのセル頂点におけるスカラー値の最大値と最小値を座標軸とする 2 次元空間 (Span Space) 上でセルを探索する技術 [1,2,3,4] がある。またそれとは別に、ボリューム空間を多重解像度分解した木構造を用いる手法 [5] や、ボリューム中のスカラー場の極大点・極小点を連結する位相構造を活用した手法 [6,7] があげられる。

[分類 2] あらかじめ視点を固定した上で、等値面の各部位について、その視点からの可視性、奥行き方向の前後関係、などを利用した高速化手法 [8,9]。

[分類 3] 等値面をポリゴン化する処理の高速化 [10,11]、あるいはポリゴンではなく点群を用いることによる描画処理の高速化 [4,9,12]。

[分類 4] その他の手法として、メモリ空間に全体を配置できないような大規模ボリュームにおける Out-of-core な等値面生成手法 [13] や、時系列ボリュームにおける Span Space を用いた等値面生成手法 [14] など。

本報告では、半透明複数等値面の高速生成の一手法を提案する。半透明複数等値面の生成における処理時間の観点からの主な課題は、以下の 2 点である。

- ・ 複数のスカラー値をもつ等値面の同時生成。
- ・ 視点からの距離の遠い部位から順に描画すること。

この 2 つの課題に対する解決手段として提案手法では、ボリュームを構成するセルを、スカラー値の最小値および最大値、視点からの距離、の 3 値を座標軸とする空間 (3 次元 Span Space、以下 3DSS) に配置する。そして 3DSS 上で等値面と交差するエレメントを検索することにより、複数の等値面を高速に生成し、適切に半透明表示する。

本報告の提案手法の位置づけは以下の通りである。Span Space を用いて等値面と交差するセルを高速抽出するという意味で、本手法は [分類 1] に属する。それと同時に、視点を固定した上で等値面を高速生成するという意味で、本手法は [分類 2] にも属する。また後述するように、[分類 3] [分類 4] との融合によって提案手法を発展させることも可能である。

2. 関連研究

2.1 ISSUE とは

前述のとおり、等値面の高速生成手法に Span Space を活用した手法が多く報告されている [1,2,3,4]。これらの手法ではまず、ボリュームデータの各々のセルについて、各セル頂点におけるスカラー値の最小値 (S_{min}) と最大値 (S_{max}) を特定する。続いて、 S_{min} を横軸、 S_{max} を縦軸に取る 2 次元直交座標系 (Span Space、以下 SS) に、すべてのセルをプロットする。このときユーザーが指定したスカラー値 s の等値面と交差するセルは、図 2 の左上の塗りつぶされた領域上にある。

Span Space を用いる等値面高速生成手法の代表例に ISSUE [1] がある。ISSUE では Span Space を格子状に分割し、指定されたスカラー値を境界にして左上部分に属する格子内部を探索することで、等値面と交差するセルを高速抽出する。

ここで、ISSUE を用いて複数の半透明等値面を表示するには、以下の 3 ステップを行う必要がある。

1. 複数の等値面を個々に生成し、一時蓄える。
2. 半透明表示を実現するために、等値面を構成する点群を、視点からの距離の遠い順に並び替える。
3. 視点からの距離の遠い順に点群を描画する。

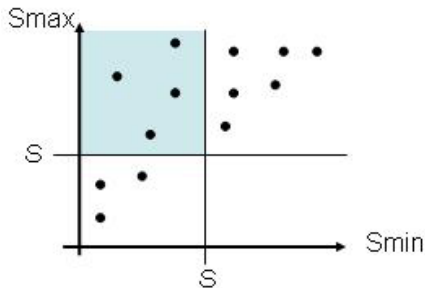


図 2. Span Space の原理。

2.2 ISSUE の実装方法

ISSUE では、Span Space の各座標軸を L 個に分割したものの (図 3 参照) を適用し、2 次元配列を用いてこれを実装する方法を提案している。以下、ISSUE における前処理とセル探索の処理を解説する。

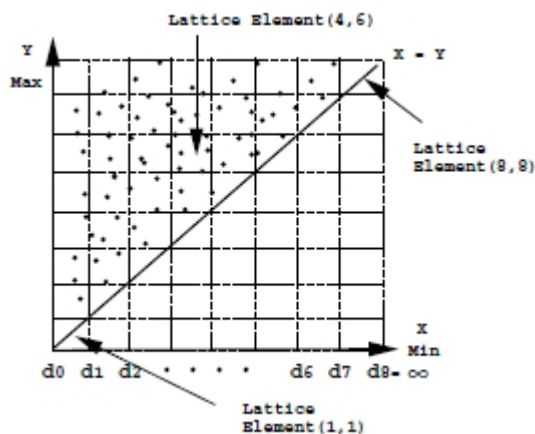


図 3. ISSUE の実装方法。

【前処理】

2 次元配列の一要素を (i,j) とする。また、あるセルのセル頂点におけるスカラー値の最小値および最大値を $Smin, Smax$ とする。このとき、要素 (i,j) に格納されるセルの条件は以下の式で表現される。

- $Amin(i) < Smin < Amin(i+1)$
- $Amax(j) < Smax < Amax(j+1)$

上記 2 式を満たす配列にセルを挿入する。

但し $Amin(i)$ とは、 $Smin$ に対する座標軸上の i 番目の要素に割り当てられたスカラー値で、 $Amin(i) < Amin(i+1)$ を満たすものとする。 $Amax(j)$ は $Smax$ に対して同様である。

【セル検索】

これから生成する等値面のスカラー値を s とすると、以下の条件を満たすセルは等値面と交差する。

- $s < Amin(i)$
- $s > Amax(j)$

上記 2 式を満たす要素 (i,j) には、等値面と交差するセルが格納されている可能性がある。ここで、 $Amin(i)$ とは s を越える最小値、 $Amax(j)$ は s を越えない最大値、と考えるとよい。

なお ISSUE では、Span Space を格子分割してできる小領域ごとに、多次元空間索引 (例えば k-d tree) を適用することで、さらに効率化を図れる、という点を主張している。これは言い換えれば、文献[2,3]の手法を局所的に ISSUE に併用することが望ましい、という主張であるとも言える。

3. 提案内容

3.1 提案手法のアルゴリズム

提案手法では、半透明複数等値面の生成の高速化を狙い、ISSUE で必要だった 3 ステップを、以下の 2 ステップにする。

1. 複数の等値面を一気に生成するために、等値面と交差するセルを、視点から遠い順に処理する。
2. この順で処理された部位から等値面を描画する。

この 2 ステップを実現するためには、1 章で提示した課題を解決する必要がある。以下に、この課題を解決するためのアルゴリズムを説明する。

【複数等値面を同時高速生成する方法】

提案手法では、複数等値面を同時に生成できるように、ISSUE を拡張する。例えば、フィールド値が s_1, s_2, s_3 の 3 枚の等値面を作るとする。このとき s_1 の等値面と交差するセルは、Span Space 上では図 4 (上) の領域に存在する。同様に s_2 の等値面と交差するセルは の領域、 s_3 の等値面と交差するセルは の領域に存在する。つまり、 s_1, s_2, s_3 の 3 枚の等値面を同時生成するためには、階段状の塗りつぶされている領域上のセルを一気に処理すればよい。

【視点から遠い順に点群生成する方法】

提案手法では、視点から遠い部位から順に等値面を描画するために、ISSUE の座標系の 3 次元目として、視点からの距離という軸を追加 (図 4 (下) 参照) する。これによって SS を 3DSS に拡張する。なお ISSUE が SS 上で 2 次元配列を構成してセルを格納していたのと同様に、提案手法では 3 次元配列を構成して 3DSS 上のセルを構築する。このデータ構造により、図 4 (上) の階段状の領域を、視点から遠い順に処理することが容易になる。

なお 3DSS においても ISSUE と同様に、Span Space の小領域ごとに、多次元空間索引を適用することで、さらに効率化が図れると考えられるが、現時点ではまだ実装していない。

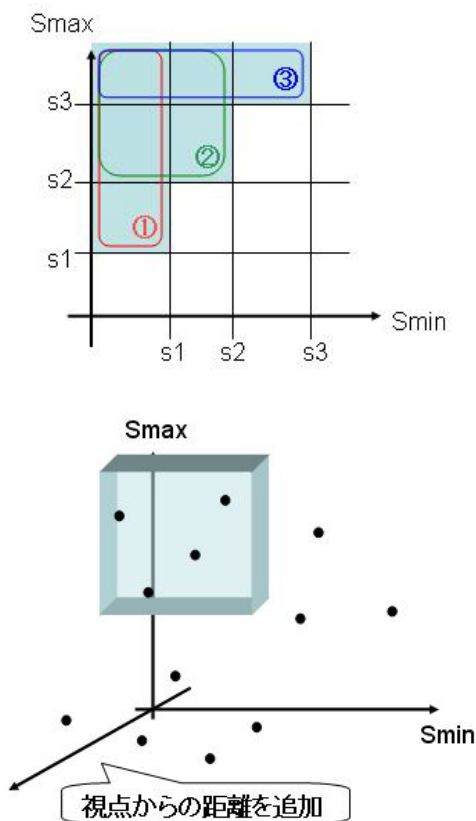


図4. (上)複数等値面の同時高速生成。(下)視点から遠い順にセルを処理。

3.2 提案手法の定量評価

提案手法のセル探索部における計算量を定量評価する。ボリュームデータのセル総数を N 、ISSUE および 3DSS における座標軸の分割数を L 、1 枚の等値面と交差するセル数の平均値を K 、等値面の枚数を M とする。

非高速化手法を使って複数の等値面を生成し、視点から遠い順にソートして半透明表示した場合の処理時間は、

$$O(MN + MK \log K)$$

と評価される。非高速化手法の代わりに ISSUE を使ったときのセル探索の処理時間は、文献[1]での分析より、

$$O(M \log(N/L) + \sqrt{NM}/L + MK + MK \log K)$$

と評価される。一方、複数の等値面のうち最低 1 枚以上の等値面と交差するエレメント数を K_2 とすると、本手法のセル探索の処理時間は、

$$O(\log(N/L) + \sqrt{NM}/L + K_2)$$

と評価される。

一般的に K_2 は $O(N)$ よりも小さいことから、提案手法は大規模なデータになるほど、既存手法に対する優位性が大きいということが定量評価できる。

続いてメモリ使用量を定量評価する。ISSUE でも 3DSS でも、Span Space を表現する配列を確保するために、非高速化手法に対してメモリ使用量が増加する。ISSUE の非高速化手法に対するメモリ使用量の増加は

$$O(L^2 + N)$$

であり、3DSS の場合には、

$$O(L^3 + N)$$

であると考えられる。

文献[1]の実行例には、 $L=200$ 程度のときに処理時間が最適であったと記載されている。仮に 3DSS においても $L=200$ 程度のときに処理時間が最適であるとする、非高速化手法のみならず ISSUE と比較しても、メモリ使用量の増加は無視できない大きなものになる。逆に言えば、 L が小さくても十分な高速化を実現することも、今後の課題の一つであると考えられる。

4. 実行結果

図 5 は、流れに対して垂直に置かれた円板のまわりの圧力分布を、提案手法を用いて可視化した例である。この例では 6~9 枚の等値面を同時生成している。

続いて表 1 に、セル探索の処理時間測定結果を示す。非高速化手法とは ISSUE ではなく、すべてのセルに対して等値面との交差判定をする実装のことである。単位はミリ秒、セル数は Data1 で $24 \times 24 \times 24$ 、Data2 で $50 \times 50 \times 52$ 、Data3 で $100 \times 100 \times 102$ である。非高速化手法と比べ本手法は、Data1 では約 15.7 倍、Data2 では 56 倍、Data3 では 134 倍の高速化に成功した。

表 2 は Data1 を基準とした際の処理時間およびセル数の相対値を示す。この結果から提案手法が、データが大規模になればなるほど有利であることがわかる。

今回の実行環境は、Panasonic CH-W4 (CPU 速度 1.20GHz, メモリ 0.99GB)および Windows XP である。

表 1 処理時間測定結果

	Data1	Data2	Data3
提案手法	0.96	2.92	11.61
非高速化手法	15.2	163.64	1551.58

表 2 Data1 を基準にした際の処理時間の比較

	Data2	Data3
本手法の処理時間	3.042 倍	12.094 倍
非高速化手法の処理時間	10.77 倍	102.08 倍
セル数	9.404 倍	73.785 倍

5. まとめ

本報告では、3次元 Span Space を利用した半透明複

数等値面の高速生成手法を提案した。提案手法ではボリュームデータを構成するセルを、スカラー値の最小値および最大値、視点からの距離、の3値を座標軸とする3次元空間に配置し、この空間上で等値面と交差するセルを高速検索する。

なお、提案手法はまだ研究の初期段階にあり、多くの課題と技術的可能性を残している。以下、そのいくつかについて議論したい。

まず4章での実行結果では、処理時間やメモリ使用量の観察が十分ではない。特にSpan Spaceの分割数Lと処理時間やメモリ使用量の相関性、Span Spaceの各小領域に多次元空間索引を採用した場合の処理時間の改善度、ISSUEとの処理時間の比較、などを検証すべきであると考えている。

図1にも示したように、半透明複数等値面と同様な視覚的効果は、多峰性の伝達関数を適用したボリュームレンダリングにおいても表現可能である。両者の処理時間を比較すると、量的には、ボリュームデータが大規模化すれば半透明複数等値面が有利であると考えられる。実際に両者の処理時間を測定し、どのような状況下においてどちらが有利か、という観点での実行結果を出すことも重要であると考えている。

1章で論じたように、等値面生成手法は視点を固定するという制約下においても活発に議論されている。また最近では、ボリュームデータにおける最適視点を自動算出する手法も議論されており[15]、これらと提案手法の併用にも意義があると考えられる。しかし、視点を固定せずに利用したいという要求があるのも当然である。提案手法の技術的なチャレンジとして、現時点での処理速度を損ねずに視点操作を可能にする、という点が考えられる。一案として、時系列に沿って徐々にスカラー値が変化するボリュームデータに対するSpan Spaceの探索手法[14]を、視点操作に沿って徐々に視点からの距離値が変化するボリュームデータに対する手法、というように置き換えての適用が考えられる。

本手法ではボリュームデータを構成するセルに、スカラー値の最小値、最大値、視点からの距離、という3変数を加えて、この3変数を3DSSの座標軸においているが、セルの変数をさらに増やすことで等値面表現の機能性を拡張することも考えられる。例えば等値面の位相構造上の特徴(例えば包含関係[16])から得られる重要度を、あらかじめ変数として各セルに反映させることが可能であると考えられる。

文献[10,11]でも指摘されているように、Marching Cubes法に代表される等値面ポリゴン化手法には、処理時間上の問題がある。これを改善する一案として本研究でも、等値面をポイントレンダリングする[4,9,12]ことを視野に入れている。しかし図5は非常に単純な描画プログラムによって生成されており、等値面を表現

する点群に穴や重なりが生じている。この描画プログラムを改善した上で処理時間を再測定したい。またこの際に、点群の詳細制御を3DSS上で実現できれば、等値面と交差するセルの抽出処理、および描画処理、の両面において高速化が期待できる。なお、点群の詳細制御手法[17]、点群の半透明複数表示手法[18]、点群とポリゴンの併用手法[19,20]、などは等値面生成以外の分野でも活発に議論されており、これを等値面生成に適用することも課題の一つと考えている。

参考文献

- [1] Shen H.-W., Hansen C. D., Linvat Y., Johnson C. R., Isosurfacing in Span Space with Utmost Efficiency, IEEE Visualization '96, pp.287-294, 1996.
- [2] Livnat Y., Shen H., Johnson C. R., A Near Optimal Isosurface Extraction Algorithm Using the Span Space, IEEE Transactions on Visualization and Computer Graphics, Vol. 2, No. 1, pp. 73-84, 1996.
- [3] Cignoni P., Marino P., Montani C., Puppo E., and Scopigno R., Speeding Up Isosurface Extraction Using Interval Trees, IEEE Transactions on Visualization and Computer Graphics, Vol. 3, No. 2, pp. 158-170, 1997.
- [4] Rymon-Lipinski B., Hanssen N., Jansen T., Ritter L., Keeve E., Efficient Point-Based Isosurface Exploration Using the Span-Triangle, IE Visualization 2004, pp. 441-448, 2004.
- [5] Wilhelms J., Van Gelder A., Octrees for Fast Isosurface Generation, ACM Transactions on Graphics, Vol. 11, No. 3, pp. 201-227, 1992.
- [6] Kreveld M., Oostrum R., Bajaj C. L., Pascucci V., Schikore D. R., Contour Trees and Small Seed Sets for Isosurface Traversal, Proceedings of 13th ACM Symposium of Computational Geometry, pp. 212-219, 1997.
- [7] Itoh T., Yamaguchi Y., Koyamada K., Fast Isosurface Generation Using the Volume Thinning Algorithm, IEEE Transactions on Visualization and Computer Graphics, Vol. 7, No. 1, pp. 32-46, 2001.
- [8] Livnat Y., Hansen C. D., View dependent isosurface extraction, Visualization '98, pp.175-180, 1998.
- [9] Livnat Y., Tricoche X., Interactive Point-Based Isosurface Extraction, IEEE Visualization 2004, pp. 457-464, 2004.
- [10] Howie C. T., Blake E. H.: The Mesh Propagation Algorithm for Isosurface Construction, Computer Graphics Forum (Eurographics), Vol. 13, No. 3, pp. C-65-74, 1994.
- [11] 伊藤, 山口, 小山田, 等値面生成のための高速ポリゴン構築方法, 情報処理学会論文誌, Vol. 42, No. 5, pp. 1076-1083, 2001.
- [12] Christopher S. Co, Bernd Hamann, Kenneth I. Joy,

Reading in Iso-splatting: A Point-based Alternative to Isosurface Visualization, Pacific Graphics '03, p.325, 2003.

[13] Chiang Y.-J., Silva C. T., Schroeder W. J., Interactive Out-Of-Core Isosurface Extraction, IEEE Visualization '98, pp. 167-174, 1998.

[14] Shen H.-W., Isosurface Extraction in Time-varying Fields Using a Temporal Hierarchical Index Tree, IEEE Visualization '98, pp. 159-166, 1998.

[15] Takahashi S., Fujishiro I., Takeshima Y., Nishita T., A Feature-Driven Approach to Locating Optimal Viewpoints for Volume Visualization, IEEE Visualization 2005, pp. 495-502, 1998.

[16] 徳永, 竹島, 高橋, 藤代, 位相解析に基づくボリュームビジュアライゼーションの高度化, 画像電子学会誌, Vol. 32, No.4, pp. 418-427, 2003.

[17] Pfister H., Zwicker M., van Baar J., Gross M., Surfels: Surface elements as rendering primitives, SIGGRAPH 2000, pp. 335-342, 2000.

[18] Zwicker M., Pfister H., van Baar J., Gross M. H., Surface Splatting. SIGGRAPH 2001, pp. 371-378, 2001.

[19] Chen B., Nguyen M. X.. POP: A hybrid point and polygon rendering system for large data. IEEE Visualization 2001, pp. 45--52, 2001.

[20] Ji J., Li S., Liu X., We E., P-Quadtrees: A Point and Polygon Hybrid Multi-Resolution Rendering Approach, Computer Graphics International 2004, pp. 382-385, 2004.

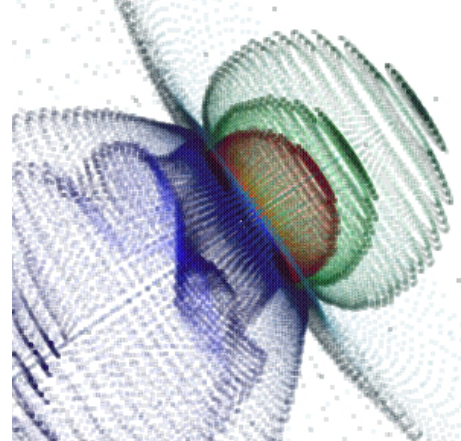
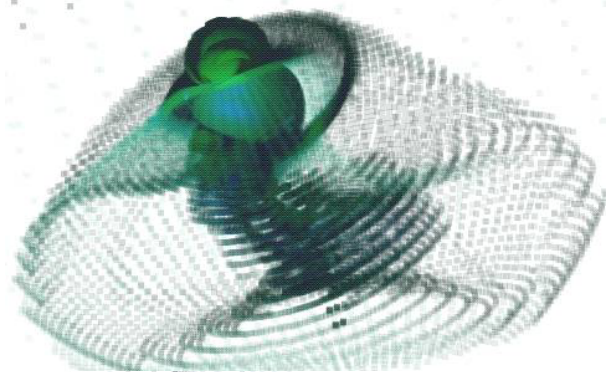
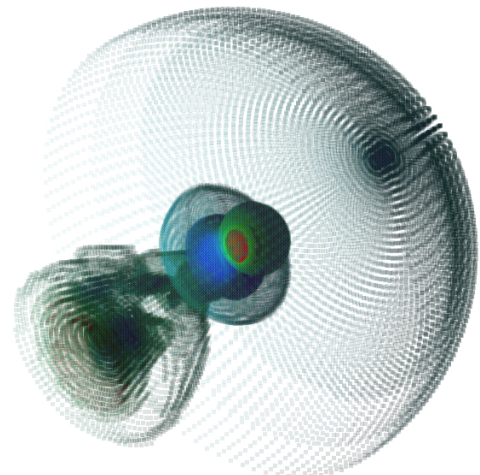
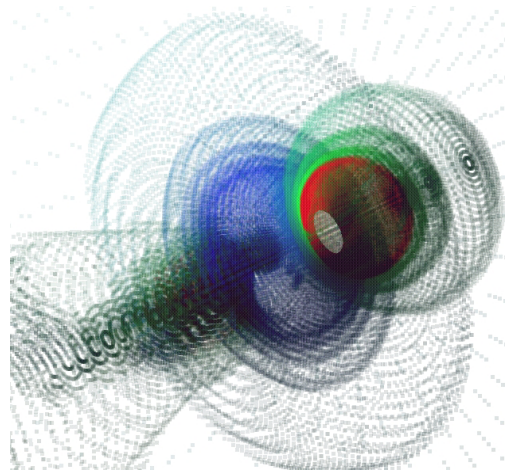


図5 提案手法による可視化の例。