

# 目次

第1章	序論	4
1.1	本研究の背景	4
1.1.1	移流の数値計算	4
1.1.2	流体とコンピューターグラフィックス	4
1.2	本研究の目的	5
1.3	本論文の構成	5
第2章	関連する研究	7
2.1	Euler 的な解法	7
2.2	Lagrange 的な解法	9
2.3	ALE 法, Semi-Lagrangian 法	10
2.4	テクスチャによる流れ場の可視化	11
第3章	写像による移流方程式の解法	12
3.1	移流方程式	12
3.1.1	Semi-Lagrangian 法による解法	13
3.1.2	Semi-Lagrangian 法の誤差の原因	13
3.2	手法の概要	14
3.2.1	従来手法との違い	14
3.2.2	フローチャート	15
3.3	移流の写像	16
3.3.1	写像の定義	16
3.3.2	写像の数値計算	16
3.3.3	順方向写像	16
3.3.4	逆方向写像	16
3.4	写像関数の定式化	18
3.4.1	速度場が時間定常の場合	18
3.4.2	速度場が時間非定常の場合	19
3.5	移流項の計算	20
3.5.1	写像による移流の計算	20
3.5.2	逆写像関数の更新	21
3.6	非移流項の計算	21
3.6.1	逆写像による移流前の値の更新	21
3.6.2	写像関数の更新	22
3.7	再初期化	22
3.7.1	写像関数のゆがみ	23
3.8	Semi-Lagrangian 法との関連	23

第 4 章 計算例と検証	25
4.1 速度が時間定常の場合	25
4.1.1 一次元の速度定常移流	25
4.1.2 二次元の剛体回転	29
4.2 速度が時間非定常の場合	33
4.2.1 流入を含む二次元の移流	33
4.2.2 非圧縮流体計算	38
第 5 章 結論	42
謝辞	43
参考文献	43

# 目次

2.1 Euler 的な解法 の概念 (実線は真の分布, 点線が補間による分布, 矢印が微分値)	7
2.2 Lagrange 的な解法 の概念 (実線は速度場, 点が計算点, 矢印が計算点上での速度)	9
2.3 テクスチャマッピングと写像の関係	11
3.1 Semi-Lagrangian 法 の概念	13
3.2 Semi-Lagrangian 法 における誤差の発生	14
3.3 Semi-Lagrangian 法 と提案手法の違い	14
3.4 提案手法のフローチャート	15
3.5 移流方程式における写像	16
3.6 写像関数と写像方向の概念	17
3.7 写像としての画像の回転 (左: 元画像, 中央: 逆方向写像, 右: 順方向写像)	17
4.1 1次元速度定常移流の計算体系	25
4.2 1次元速度定常移流の計算結果 (再初期化なし)	26
4.3 1次元速度定常移流の計算結果 (2ステップごとの再初期化)	27
4.4 1次元速度定常移流の計算結果 (再初期化あり)	27
4.5 再初期化による誤差の変化	28
4.6 Courant 数と再初期化間隔の値が整数になる場合の結果	28
4.7 移流させる剛体	29
4.8 2次元剛体回転の計算体系	30
4.9 Semi-Lagrangian 法 による結果, 解析解 (緑), $128^2$ (ピンク), $256^2$ (黄色)	31
4.10 提案手法による結果, 解析解 (緑), 提案手法 (10ステップ毎) (ピンク)	31
4.11 再初期化による誤差の変化	32
4.12 流入を含む二次元移流の計算体系	34
4.13 Semi-Lagrangian 法 による結果 (解像度, 上 $128^2$ , 中 $256^2$ , 下 $1024^2$ )	35
4.14 提案手法による結果 (ゆがみの閾値, 上 0.2, 中 0.6, 下 1.8)	36
4.15 提案手法で写像関数を更新しない場合の結果 (ゆがみの閾値, 1.8)	37
4.16 再初期化による誤差の変化	37
4.17 流体のキーフレーム制御の概念	38
4.18 計算に用いたキーフレームと初期分布	39
4.19 流体のキーフレーム制御の結果, 上: Semi-Lagrangian 法, 下: 提案手法	39
4.20 煙のシミュレーション, $64^3$ , 上: Semi-Lagrangian 法, 下: 提案手法	40
4.21 煙のシミュレーション, $128^3$ , 上: Semi-Lagrangian 法, 下: 提案手法	41
4.22 障害物を含む煙のシミュレーション, $64^3$ , 上: Semi-Lagrangian 法, 下: 提案手法	41

# 第1章 序論

## 1.1 本研究の背景

### 1.1.1 移流の数値計算

移流とは、速度場に乗って物理量が流れる現象である。移流を記述した方程式は移流方程式と呼ばれ、一般には解析的に解くことができない偏微分方程式となっている。移流は流体の運動や波の伝播をはじめとして、多くの物理現象に関わっているため、様々な物理現象の数値計算で、移流方程式を精度良く解く事が必要である。

移流方程式の数値解方法には、大きく分けて Lagrange 的な解法と Euler 的な解法とがある。Lagrange 的な解法とは、流れに乗った計算点を用いて、各時刻でその点の移動を求める解法で、Euler 的な解法とは、空間に固定した計算点を用いて、各時刻でその点を過ぎる物理量について求める解法である。

Lagrange 的な解法では、移流方程式は単に計算点を移動するだけで解くことができる [29]。離散化によって生じる誤差は、計算点の位置についてのみであるため、計算点上の物理量については正確である。しかし、計算点が移動することで場所によって計算点の粗密が生じ、計算の解像度が不均一になる。また、速度場によって計算点が不規則に移動することで、計算点の相互の位置が不規則になるため、計算点以外の位置における物理量の評価が困難であるという問題もある。

一方で、Euler 的な解法では、移流方程式を空間と時間で離散化して解く必要がある。離散化には本来の微分方程式に対する離散化誤差が生じるが、特に、空間に関する微分の離散化誤差が、本来の移流には関係ない非物理的な拡散や振動（数値拡散・数値振動）を引き起こすことが知られている [28]。しかしながら、Lagrange 的な解法とは違って空間に固定の計算点を用いているため、速度場に依存せず常に均一な解像度で計算できる。また、計算点の相互の位置も固定のため、補間などによって計算点以外の位置でも簡単に物理量の評価ができるという利点がある。

このように、Euler 的な解法と Lagrange 的な解法とは、それぞれ利点・欠点を持つものであり、移流方程式の数値計算にはまだ問題が残されていると言える。

### 1.1.2 流体とコンピューターグラフィックス

近年、数値計算による流体の計算（数値流体力学）の発展に伴い、コンピューターグラフィックスの分野で、数値計算を元にした水 [5] や煙 [7] などの、流体のアニメーションが盛んに研究されている。特に、視覚的になめらか（均一な解像度）な計算結果が得られる Euler 的な解法が注目を集めており、既に映画などでは従来達成することのできなかつた視覚効果を実現している [1][25]。

しかし既に述べたように、Euler 的な解法では移流方程式の離散化に起因する数値誤差が問題となるり、多くの文献 [6][5][7][20] でその問題が指摘されている。例えば、水のコンピューターアニメーションでは、界面を定義する物理量が数値拡散を受けると、水の形状が非物理的に変形・縮小してしまい、視覚的に非常に違和感のある結果となってしまう。そのため、数値拡散・数値振動に対処するための様々な手法が、コンピューターグラフィックスの分野においても研究されている。

数値流体力学とコンピューターグラフィックスという二つの分野を関連付けるもう一つの研究の分野が、数値流体力学によって得られた結果の可視化である。コンピューターグラフィックスにおける流体のアニメーションでは、数値計算によって写実的な流体の挙動を得ることが目的であるのに対して、計算の可視化では、数値計算の結果についての特徴を、視覚的に抽出することが目的である。可視化には抽出したい情報に応じて様々な手法が存在するが、本論文に関連がある可視化手法として、テクスチャマッピングによる流れ場の可視化 [15][11][24] がある。

テクスチャマッピングとは、コンピューターグラフィックスにおいて、ある模様データ(テクスチャ)を、割り当てられたテクスチャ座標を用いて、何らかの面に貼り付ける(マッピング)処理である。テクスチャマッピングによる流れ場の可視化では、流れ場に貼り付けたテクスチャを移流させることでテクスチャを変形させて、全体的な流れ場の挙動を可視化している。

テクスチャマッピングによる流れ場の可視化では、テクスチャによる模様の変化から、流れ場を見るため、可視化中のその模様は細かいディテールを保つ必要がある。そのため、多くの研究で細かいディテールを保ったままテクスチャを移流するための手法を導入している [15][11]。

## 1.2 本研究の目的

本研究の目的は、Euler 的な解法を用いて移流方程式を解いたときに発生する数値誤差を抑制する数値解法を提案することである。その基本となる解法として、Lagrange 的な考え方によって Euler 的な固定点における移流方程式を解く、Semi-Lagrangian 法 [22] を用いる。

Semi-Lagrangian 法は、移流方程式の計算を空間で離散化せずに、物理量が流れてくる上流の点の位置を計算し、その点で物理量の値を求めるという手順(タイムステップ)を繰り返すことで、移流方程式を解いている。Semi-Lagrangian 法は、上流の点の位置を速度場に乘って求めるという Lagrange 的な考え方をを用いているが、上流の点での物理量を求めるときに固定された計算点の間で補間をする必要があり、Euler 的な解法と同じく数値的な誤差が発生する。

従来、Semi-Lagrangian 法で移流方程式を解く際の誤差を抑制する場合には、上流の点で物理量の値を求めるときに使用する補間の精度を上げることによって、毎タイムステップで発生する誤差の大きさを抑制していた。これに対して、本研究では、移流方程式を解く際の数値誤差の原因が

1. 1 タイムステップで補間誤差が発生する
2. 発生した補間誤差が繰り返しによって蓄積する

という二段階に分けられることに注目し、繰り返しによって発生する補間誤差の蓄積を遅くすることで、移流方程式を解く際の数値誤差を抑制する方法を提案する。つまり、従来の Semi-Lagrangian 法では毎タイムステップで誤差が蓄積していたが、提案手法を用いることで、複数タイムステップ毎に誤差が蓄積するようになる。

そのための手段として、移流方程式の写像関数という概念を導入し、写像関数を用いて移流方程式を解く手法を提案する。写像関数の定式化は、テクスチャマッピングによる流れ場の可視化で提案されていたテクスチャ座標の定式化 [15] と形式的に同じだが、本研究ではそれが移流方程式の写像関数となっている事を示し、それによって移流方程式の解法を構築する。

## 1.3 本論文の構成

本論文は全 6 章から構成される。まず第 1 章では、序論として本研究の背景・目的について述べた。次に第 2 章では、本研究に関連する移流方程式の解法や流れ場の可視化の研究について述べる。第 3 章では、本研究で提案する写像関数による移流方程式の解法についてその詳細を述べ

る．第 4 章では提案した解法の計算例による検証および考察を行う．最後に第 5 章では，本研究の結論を述べる．

## 第2章 関連する研究

### 2.1 Euler 的な解法

移流方程式の Euler 的な解法とは、空間に固定した計算点の上で方程式を解く手法のことである。この時、移流方程式は以下のように定義される。

$$\frac{\partial H(\boldsymbol{x}, t)}{\partial t} + \boldsymbol{u}(\boldsymbol{x}, t) \cdot \nabla H(\boldsymbol{x}, t) = 0 \quad (2.1)$$

ここで、 $\boldsymbol{x}$  は位置、 $t$  は時間、 $H(\boldsymbol{x}, t)$  は移流する物理量、 $\boldsymbol{u}(\boldsymbol{x}, t)$  は速度場を表している。式 (2.1) を離散化する場合、時間と空間について離散化する必要があるが、移流方程式の場合に問題となるのは、空間一階微分を含む  $\nabla H(\boldsymbol{x}, t)$  である [28]。従って以下では、 $\nabla H(\boldsymbol{x}, t)$  の離散化について述べる。

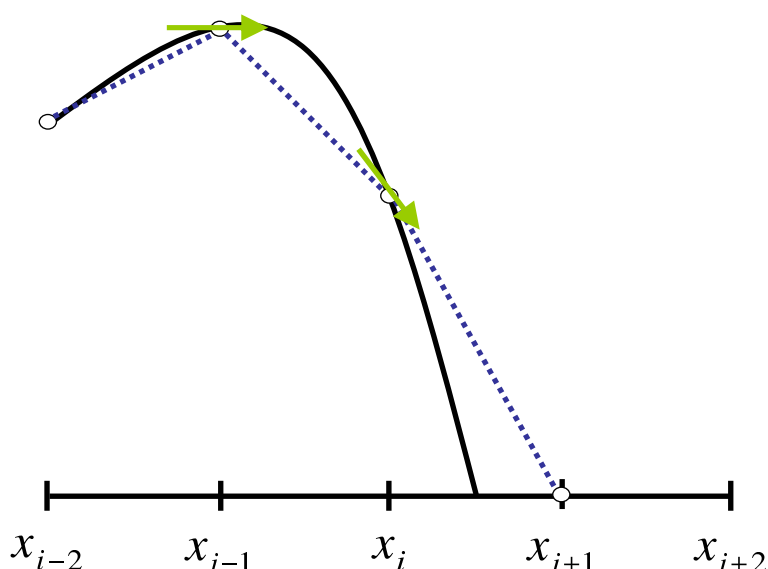


図 2.1: Euler 的な解法 の概念 (実線は真の分布、点線が補間による分布、矢印が微分値)

移流方程式の Euler 的な解法の基本的なアプローチは、離散的に与えられた  $H(x_i, t)$  について、計算点間の分布を何らかの関数によって近似し、その微分値から  $\nabla H(x, t)$  を求めるというものである (図 2.1)。

Euler 的な解法で最も基本的なのは、計算点間の分布を 1 次関数で近似する 1 次精度風上差分と呼ばれる手法である。風上差分とは以下のように流れの「上流」の点との間で差分を構築する方法である。

$$\frac{\partial H(\boldsymbol{x}_i, t)}{\partial x} = \begin{cases} \frac{H(\boldsymbol{x}_i, t) - H(\boldsymbol{x}_{i-1}, t)}{\Delta x} & u(\boldsymbol{x}_i, t) \geq 0 \\ \frac{H(\boldsymbol{x}_{i+1}, t) - H(\boldsymbol{x}_i, t)}{\Delta x} & u(\boldsymbol{x}_i, t) < 0 \end{cases} \quad (2.2)$$

ここで、 $\Delta x$  は空間刻み幅である風上差分は 2 次、3 次と計算点間の分布を近似する関数の次数を上げることでさらに高次精度の近似を構築できる。

Euler 的な解法では、数値的な安定性を保つために以下のような条件を満たす必要がある。

$$\Delta t < \alpha \frac{\Delta x}{|u_{max}|} \quad (2.3)$$

ここで、 $|u_{max}|$  は  $cu(x, t)$  の最大の絶対値であり、 $\alpha$  は、 $0 < \alpha \leq 1$  となるような定数である。これは CFL 条件と呼ばれ、この条件を満たさなければ計算が発散してしまう。計算手法によっては  $\alpha$  が小さくなることもあり、より条件が厳しくなる。局所的に速度場の値が大きくなるような場合であっても、 $|u_{max}|$  によって CFL 条件が定められるため、計算全体で CFL 条件を満たすような小さな  $\Delta t$  を用いる必要がある。また、一般に計算の解像度を上げるほど  $\Delta x$  が小さくなるため、解像度の高い計算では、小さな  $\Delta t$  を用いなければならない。

一般に高次精度の近似では、分布を近似する関数が本来の分布の最大値・最小値を超えてしまう、オーバーシュート・アンダーシュートが発生する。Gudnov の定理 [8] によれば、「オーバーシュート・アンダーシュートが発生せず、分布の単調性を維持する線形の差分法は、せいぜい 1 次精度でしかない」とされている。従ってどのような高次精度の近似でも線形であれば（前のタイムステップの値の線形和で、次のステップの値を計算するのであれば）、必ずオーバーシュート・アンダーシュートが発生するということである。

このようなオーバーシュート・アンダーシュートが発生しないための条件として、Harten [9] は Total-Variation-Diminishing (TVD) 条件という概念を導入し、TVD 条件を満たせばオーバーシュート・アンダーシュートが蓄積しない事を示している。また、TVD 条件を満たす解法として、Harten らは [10] ENO (Essentially Non-Oscillatory) を提案した。ENO では、オーバーシュート・アンダーシュートが発生しないような近似を複数の高次精度近似から選択している。後に Liu ら [14] は、ENO における高次精度近似の選択を、高次精度近似の重み付きの和に変えることによって、より近似精度を上げる WENO (Weighted Essentially Non-Oscillatory) を提案した。

Euler 的な移流方程式の高精度な解法の一つとして注目を集めているのが、Yabe ら [26] によって提案された CIP (Constrained-Interpolation-Profile) である。CIP 法では、物理量の微分（今の場合  $\nabla H(x, t)$ ）を陽に保持することによって、高次精度の近似を行っている。物理量そのものに加えて、物理量の微分も移流方程式に従って更新されるため、単に複数の計算点から高次精度の差分を構築するよりも、計算点の間の物理量の分布を高精度に近似するとされている。

Euler 的な解法の問題点は、固定点の間の分布をなんらかの関数で近似することによって生じる数値誤差である。既に述べたように、多くの手法が数値誤差を減らす方法を提案しているが、まだ問題が解決したとは言えないのが現状である [28]。特に、近年盛んに行われている流体の数値計算をベースとした流体のアニメーションの研究では、その問題が指摘されている [6][5]。

また、一般に高次精度の解法ではその空間微分の近似に、複数の点を用いる必要がある（CIP の場合は物理量の微分値）ため、境界条件も複数の点で満たすように設定する必要がある。従って、高次精度になるほど境界条件の設定が複雑になるという問題がある。

本研究で提案する手法は、計算点間の分布を近似する関数の精度（次数）を上げずに、数値誤差を抑えることが可能であるため、高次精度に関連する上記の問題を避けながら、移流方程式を解くことができるという特徴がある。



## 2.2 Lagrange 的な解法

移流方程式の Lagrange 的な解法とは、流れに乗った計算点の上で方程式を解く手法のことである。この時、移流方程式は以下のように定義される。

$$\frac{DH(\mathbf{x}, t)}{Dt} = 0 \quad (2.4)$$

ここで、 $\frac{D}{Dt}$  は実質微分と呼ばれ、以下のように定義される微分演算子である。

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u}(\mathbf{x}, t) \cdot \nabla \quad (2.5)$$

従って、この定義に基づいて実質微分を展開すれば、

$$\begin{aligned} \left( \frac{\partial}{\partial t} + \mathbf{u}(\mathbf{x}, t) \cdot \nabla \right) H(\mathbf{x}, t) &= 0 \\ \frac{\partial H(\mathbf{x}, t)}{\partial t} + \mathbf{u}(\mathbf{x}, t) \cdot \nabla H(\mathbf{x}, t) &= 0 \end{aligned} \quad (2.6)$$

となつて、式 (2.4) は Euler 的な解法における式 (2.1) と同値であることがわかる。

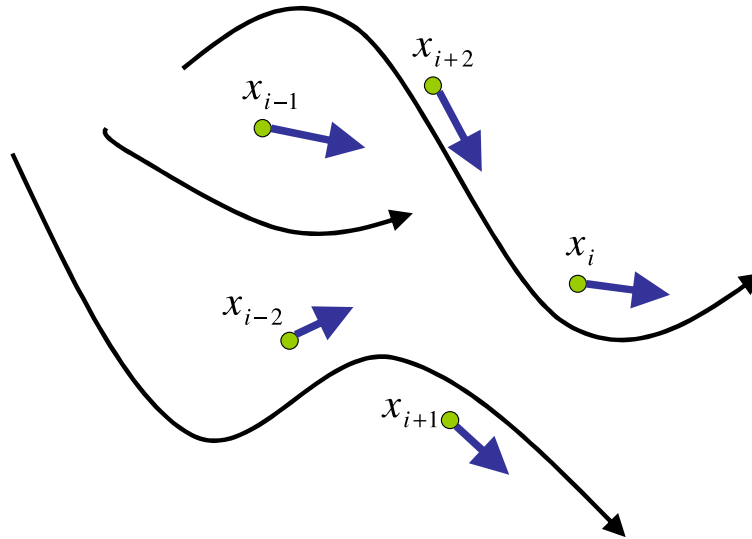


図 2.2: Lagrange 的な解法概念 (実線は速度場, 点が計算点, 矢印が計算点上での速度)

Lagrange 的な解法によって移流方程式を解くには、単に計算点を速度場に沿って移流させれば良いため (図 2.2), Euler 的な解法で問題となった数値拡散・数値振動の問題は起こらない。また、Euler 的な解法における CFL 条件 (式 (2.3)) の制限は、Lagrange 的な解法には無い。この Lagrange 的な解法を用いている計算手法の代表が、粒子法と呼ばれる方法である。

初期の粒子法は、天文学などで圧縮性流体の解析に用いられてきた、Smoothed Particle Hydrodynamics (SPH)[16] である。SPH では空間の物理量を粒子を中心としたカーネル関数の重ね合わせによって定義し、空間微分などは解析的に空間微分をしたカーネル関数の重ね合わせで計算している。

Koshizuka ら [13] は粒子による非圧縮性流体解析手法として MPS 法を提案した。MPS 法も SPH と同様に粒子を中心としたカーネル関数を定義するが、流体の計算で必要になる物理量の微分値を、粒子間相互作用として定義する点異なる。MPS は、他の Lagrange 的な解法と同様に

計算点の間に接続情報を持たないため、計算点の配置が大きく変化しても計算が可能である。そのため、特に流体が大変形する場合において様々な解析が行われている。

Lagrange 的な解法の問題は、計算点そのものが移流することにより、計算点の偏りが生じて、計算結果について十分な解像度を維持するのが難しいことである。そのため、ある点での物理量の計算に近傍粒子の情報を用いると、近傍粒子が時々刻々と変化するために、統計的な誤差から計算結果にノイズが生じてしまうことが指摘されている [28]。また、その際に近傍とみなす粒子の数や、重みの計算に用いるカーネル関数の形状は計算の安定性に大きく関連するにも関わらず、大きく任意性が残されていると言える。

## 2.3 ALE 法, Semi-Lagrangian 法

既に述べたように Euler 的な解法と Lagrange 的な解法はそれぞれ利点・欠点があるため、両方を組み合わせた手法が提案されている。そのひとつが ALE 法である [2]。ALE 法では Euler 的な手法と同じく、近傍情報が固定された計算点（例えばグリッド）を用いるが、移流の計算には Lagrange 的な手法と同じく、計算点を移動することによって達成する。そのため、Euler 的な手法で問題となる数値拡散・数値振動は生じず、Lagrange 的な手法と違って、空間的に連続な値を求めるのは容易となるという特徴がある。

一般に、ALE では Euler 的な計算点が過度にゆがむと、計算点によって構成されるメッシュが反転・重複し計算が不可能となってしまうため、適宜メッシュの再構築を行う必要がある。ところが、このメッシュの再構築では再び移流における数値誤差が生じてしまうだけでなく、特にトポロジーが変化する場合などにおいて処理が困難になるという問題がある。

本研究の基本としている、Euler 的な解法と Lagrange 的な解法を組み合わせた手法のひとつが、Semi-Lagrangian 法と呼ばれる手法である [3][22]。Semi-Lagrangian 法では、計算点には Euler 的な固定点を用いているが、移流の計算には計算点の上流の点を Lagrange 的な追跡によって求め、その点での値を元の固定された計算点の値に代入することで行う。通常、上流点での値は計算点の間に存在するので、何らかの補間関数を用いて上流点での値を求める。

Semi-Lagrangian 法の特徴は、Euler 的な解法と同様に固定点での物理量が求められるにも関わらず、Euler 的な解法における CFL 条件 (式 (2.3)) の制限が無いということである。これは、移流の計算に Lagrange 的な解法（計算点を上流に向かって動かす）を用いているからであり、特に補間関数が単調性を維持する場合、無条件安定（計算条件に関わらず安定）となる [22]。

しかしながら、Semi-Lagrangian 法は Euler 的な解法と同様に、移流方程式を解く際に数値誤差が発生する。これは、上流点での補間誤差が毎タイムステップで蓄積するためであり、Fedkiw ら [7] によって提案された Monotonic Cubic Interpolation や、Yabe らの CIP [26] などの高次精度の補間を用いて誤差を減らすことができる。しかし、高次精度の補間は複数の計算点を用いるため、Euler 的な解法における高次精度の計算と同様に、境界条件の設定が複雑になるという問題がある。

本研究で提案する手法では、移流方程式における数値誤差を抑制する方法として、従来の「毎ステップでの補間関数の精度（次数）を上げる」という手法以外に、「時間発展による誤差の蓄積を遅くする」という手法を提案している。そのため、提案手法は Semi-Lagrangian 法で使われている高次精度の補間関数と組み合わせることが可能であり、排他的な手法ではない。また、Semi-Lagrangian 法の持つ、CFL 条件に制限されないという特徴は、提案手法においても残っている。

## 2.4 テクスチャによる流れ場の可視化

本研究では、移流方程式を解くために移流の写像という概念を導入している。この概念に関連する研究が、テクスチャマッピングによる流れ場の可視化である。テクスチャマッピングとは、コンピュータグラフィックスにおいて、二次元で定義した物体の模様（テクスチャ）を、三次元のモデルデータの表面に貼り付ける手法である。

写像という概念を導入すれば、このテクスチャマッピングは、テクスチャ座標を用いたテクスチャの写像ということになる（図 2.3）。つまり、テクスチャ座標によってモデル空間上の位置がテクスチャ空間上の位置へと写像され、その点でのテクスチャの色を求めて、元のモデルデータ上の色としているということである。

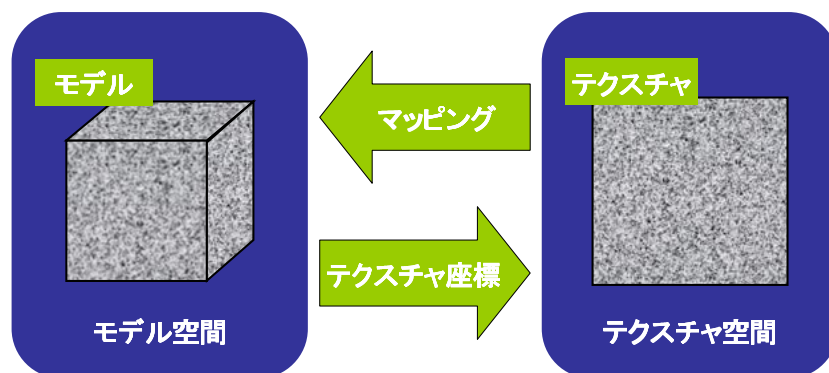


図 2.3: テクスチャマッピングと写像の関係

Max ら [15] は、このテクスチャマッピングを用いて、流れ場を可視化する方法を提案している。Max らは、速度場が定常の場合と非定常の場合にわけて、可視化のためのテクスチャ座標の移流の定式化を行っている。本研究の写像関数の定式化はこの Max らによる定式化に基づくが、Max らはそれが移流の写像関数となっている事を利用しておらず、本研究とは違って可視化にのみ特化しているため、提案手法とは全く異なる。

Max らの研究では、テクスチャ座標が流れによって過度にゆがむのを防ぐため、数タイムステップごとに現在の値をテクスチャに書き込み、テクスチャ座標を元の値にリセットする、再初期化という処理が提案されている。本研究でもこの扱いは同様であるが、Fabrice[17] と同様にテクスチャのゆがみの指標を導入し、写像関数のゆがみが一定値を超えないように再初期化する処理を行っている。

Jobard ら [11] は、Max らの定式化 [15] に基づいて、ノイズ状のテクスチャによる流れ場の可視化を行っている。Jobard らの研究は Max らと違い、毎タイムステップで再初期化を行っているため、移流に関して通常の Semi-Lagrangian 法と同様の数値誤差が生じている。これについて、Jobard らはノイズを毎タイムステップで再導入することにより対応しているため、この方法はノイズ状の物理量の移流にのみ制限されている。後の研究 [12] ではノイズの移流に加えて、色素の移流を可能としているが、毎タイムステップで非物理的な先鋭化フィルターをかけることによって、数値拡散を抑えているため滑らかな分布を持つ色素の移流には適用できない。

Stam[21] は、Max らの定式化 [15] を用いて高解像度のテクスチャを移流させれば、数値誤差を抑えた複雑な流れを表現できる事を示している。しかしながら、Stam[21] の応用ではテクスチャを移流させる場合には物理量の流入・流出がない場合に制限されている。本研究で提案する手法では、後に示すように任意の非移流項が扱えるため、通常の移流方程式の解法と同じように様々な問題に適用することが可能である。

## 第3章 写像による移流方程式の解法

### 3.1 移流方程式

本研究で対象とする移流方程式とは，以下のような偏微分方程式である．

$$\frac{\partial H(\boldsymbol{x}, t)}{\partial t} + \boldsymbol{u}(\boldsymbol{x}, t) \cdot \nabla H(\boldsymbol{x}, t) = N(\boldsymbol{x}, t) \quad (3.1)$$

ここで， $\boldsymbol{x}$  は位置， $t$  は時間， $H(\boldsymbol{x}, t)$  は移流する物理量， $\boldsymbol{u}(\boldsymbol{x}, t)$  は速度場， $N(\boldsymbol{x}, t)$  は移流以外の効果の項（例えば物理量の流入など）を表している．本研究では， $\frac{\partial H(\boldsymbol{x}, t)}{\partial t}$  について整理して

$$\frac{\partial H(\boldsymbol{x}, t)}{\partial t} = -\boldsymbol{u}(\boldsymbol{x}, t) \cdot \nabla H(\boldsymbol{x}, t) + N(\boldsymbol{x}, t) \quad (3.2)$$

以下のような二つの式に分けて

$$\frac{\partial H(\boldsymbol{x}, t)}{\partial t} = -\boldsymbol{u}(\boldsymbol{x}, t) \cdot \nabla H(\boldsymbol{x}, t) \quad (3.3)$$

$$\frac{\partial H(\boldsymbol{x}, t)}{\partial t} = N(\boldsymbol{x}, t) \quad (3.4)$$

式 (3.1) を解く．これは時間分離解法と呼ばれる方法で，Yabe らの CIP[26] でも用いられている．本研究では，特に式 (3.3) の右辺  $-\boldsymbol{u}(\boldsymbol{x}, t) \cdot \nabla H(\boldsymbol{x}, t)$  を移流項，式 (3.4) の右辺を  $N(\boldsymbol{x}, t)$  非移流項と呼ぶことにする．

なお，流体力学における移流方程式は以下のような形式となり本研究で扱う式 (3.1) とは違う．

$$\frac{\partial H(\boldsymbol{x}, t)}{\partial t} + \nabla \cdot (\boldsymbol{u}(\boldsymbol{x}, t) H(\boldsymbol{x}, t)) = 0 \quad (3.5)$$

しかしここで，式 (3.5) の左辺第二項について展開し

$$\frac{\partial H(\vec{x}, t)}{\partial t} + \vec{u}(\vec{x}, t) \cdot \nabla H(\vec{x}, t) + (\nabla \cdot \vec{u}(\vec{x}, t)) H(\vec{x}, t) = 0 \quad (3.6)$$

左辺第三項を移項すれば

$$\frac{\partial H(\vec{x}, t)}{\partial t} + \vec{u}(\vec{x}, t) \cdot \nabla H(\vec{x}, t) = -(\nabla \cdot \vec{u}(\vec{x}, t)) H(\vec{x}, t) \quad (3.7)$$

となり，右辺の  $-(\nabla \cdot \vec{u}(\vec{x}, t)) H(\vec{x}, t)$  を式 (3.1) における非移流項  $N(\vec{x}, t)$  として扱うことで，式 (3.1) と同様に解くことができることに注意する．

### 3.1.1 Semi-Lagrangian 法による解法

Semi-Lagrangian 法とは，移流方程式を解くための手法の一つであり，移流の計算に Lagrangian 的な考えを用いていながら，Euler 的な固定計算点上で計算を行うため，Semi-Lagrangian 法という名称になっている．

Semi-Lagrangian 法では，以下のような関数を用いる．

$$\begin{aligned} \frac{dP^{x,t}(s)}{ds} &= u(P^{x,t}(s), t) \\ P^{x,t}(0) &= x \end{aligned} \quad (3.8)$$

つまり， $P^{x,t}(s)$  は， $s = 0$  で位置  $x$  に存在する粒子の  $u(x, t)$  による流線をあらわしている．

この関数を用いると，ある時間  $t$  での物理量  $H(x, t)$  が与えられたとき，式 (3.3) に基づいて変化した微小な時間  $\Delta t$  後の物理量  $H(x, t + \Delta t)$  は，以下のように書くことができる．

$$H(x, t + \Delta t) = H(P^{x,t}(-\Delta t), t) \quad (3.9)$$

つまり， $H(x, t + \Delta t)$  を求めるには， $x$  における流線について上流に辿った点  $P^{x,t}(-\Delta t)$  で，現在の物理量  $H(x, t)$  の値を求めればよいということである (図)．

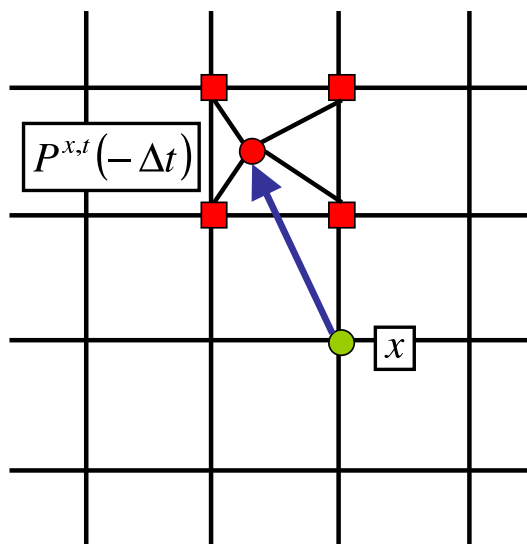


図 3.1: Semi-Lagrangian 法 の 概 念

一般に上流の点  $P^{x,t}(-\Delta t)$  は，計算点の上に無いことが多いため，その点で物理量を求めるには何らかの補間関数が必要である．

Semi-Lagrangian 法はオイラー的な手法で制限となっている，CFL 条件に制限がないことが特徴である．さらに，補間関数に単調増加・減少関数を用いれば，無条件安定となる事が示されている．

### 3.1.2 Semi-Lagrangian 法の誤差の原因

Semi-Lagrangian 法において数値誤差が発生するのは，上流点における補間関数の誤差が時間発展によって蓄積するためである．ここでは，簡単のために 1 次元での移流で説明をする．

今， $t = 0$  において図 3.2 のような物理量の分布 (左) が与えられていたとする．次のタイムステップでの物理量を求めるために，上流の点を求めてその点で 1 次関数で補間された物理量の値

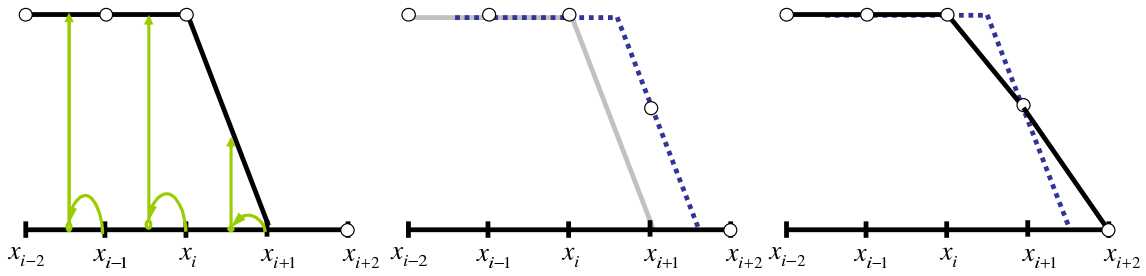


図 3.2: Semi-Lagrangian 法における誤差の発生

を求める（中央）．求められた離散的な物理量において，線形補間を行うと真の値に対して誤差が発生していることがわかる（右）．なぜなら，計算点の間を 1 次関数で補間しているため，ある計算点の間における移流後の分布が 1 次関数で近似できない場合に誤差が発生するからである．この場合，移流後の分布は区分 1 次のため 1 次関数では近似できない．

## 3.2 手法の概要

### 3.2.1 従来手法との違い

従来の Semi-Lagrangian 法は，直前のタイムステップでの値を用いて次のタイムステップでの値を求めている．次のタイムステップでの値は補間誤差を含むため，さらに後のタイムステップを計算したときに，誤差が蓄積していくという問題があった．提案手法では，移流前の値  $H_{pre}$  を別途保持しておき，各タイムステップでの値は写像によって求めている．従って，各タイムステップで誤差が発生したとしても，次のタイムステップには関係がないため，誤差の蓄積を抑えることができる．Semi-Lagrangian 法と提案手法の違いを示す概念図を図 3.3 に示す．提案手法には再初期化（図中の”Reinit”）という処理が入るため，その段階で誤差が蓄積するが，再初期化は数ステップごとにしか行われないため，Semi-Lagrangian 法よりも誤差の蓄積が遅くなっている．

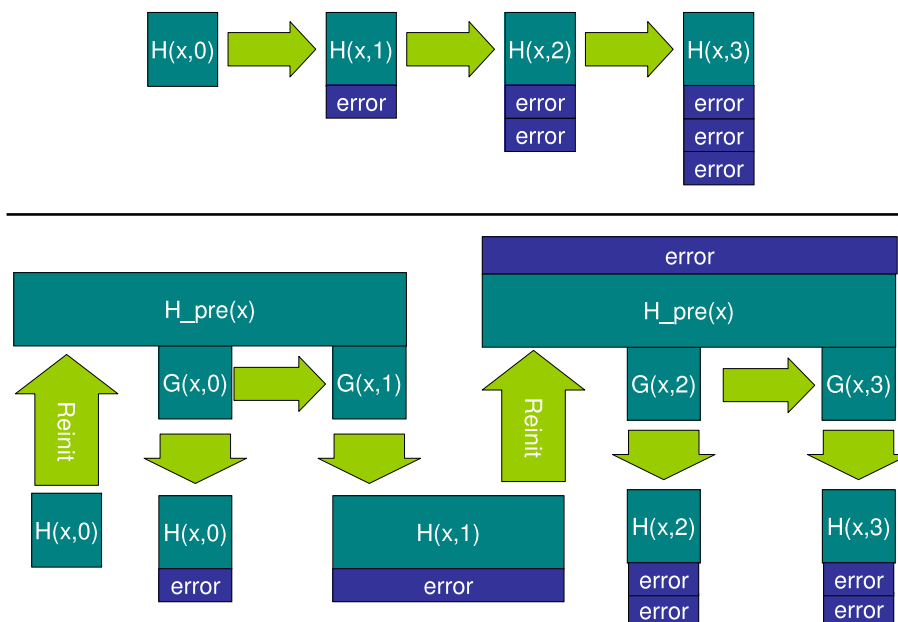


図 3.3: Semi-Lagrangian 法と提案手法の違い

### 3.2.2 フローチャート

提案手法の1タイムステップ分のフローチャートを以下に示す。

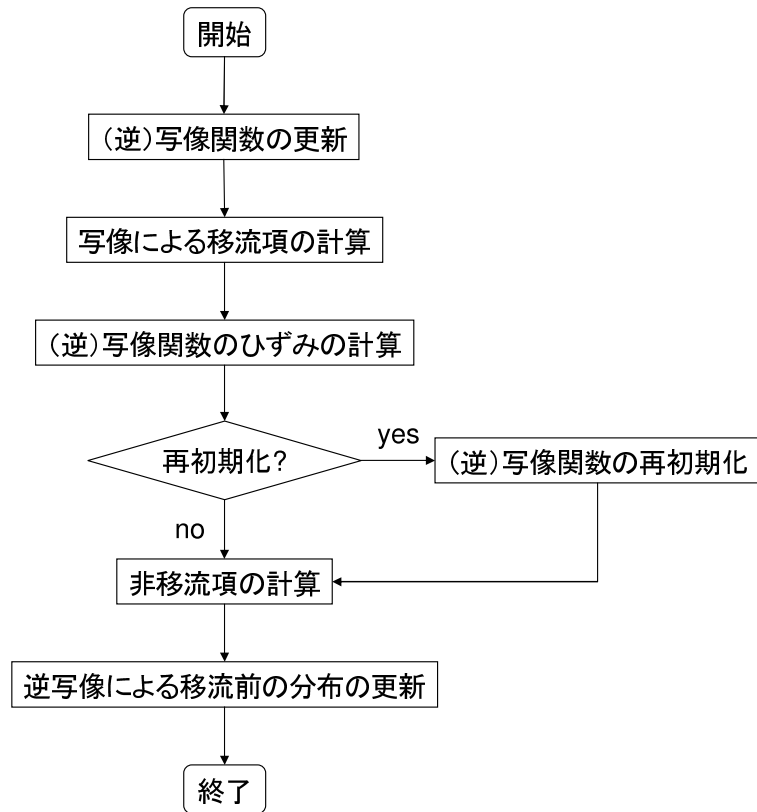


図 3.4: 提案手法のフローチャート

実際の計算は、これを必要な回数繰り返すことによって行う。以降ではこのフローチャートにおける個別の計算について述べる。

### 3.3 移流の写像

#### 3.3.1 写像の定義

移流方程式の写像関数とは，移流前の空間から移流後の空間への写像するような関数のことである．つまり，その写像関数を用いれば，移流前の分布が与えられたときに，移流後の分布が写像によって計算できるということである．従って，写像後の空間から写像前の空間へは逆写像となり，その逆写像関数を用いれば，移流後の分布から，移流前の分布を逆写像によって計算できるということになる．この概念図を図 3.5 に示す．

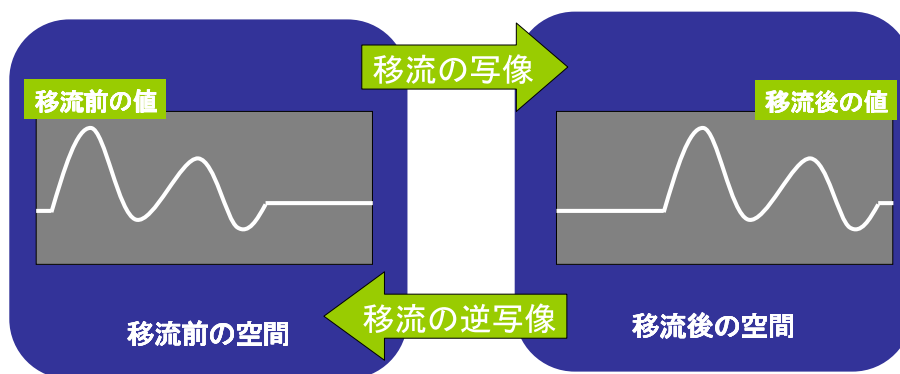


図 3.5: 移流方程式における写像

#### 3.3.2 写像の数値計算

#### 3.3.3 順方向写像

順方向写像とは，計算における写像の方向と，その写像に用いる写像関数の方向が同一となる写像の計算の事を言う．具体的には，写像関数を用いた写像，逆写像関数を用いた逆写像が順方向写像である．概念図を図 3.6 に示す．

順方向写像では，通常，場所が指定されることが多い写像先において，必ずしも値が得られるとは限らないという欠点がある．例えば，画像の回転に順方向写像を用いると，回転後の画像に穴が空いてしまう（図 3.7）．この画像の回転は以下のような式で表される．

$$\begin{bmatrix} x^* \\ y^* \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (3.10)$$

ここで， $x^*, y^*$  は回転後の位置， $x, y$  は回転前の位置， $\theta$  は回転角を示す．画像の回転で穴が開くのは， $x^*, y^*$  が必ずしも画像全体を埋めるとは限らないからである．

そのため，通常は次に示す逆方向写像を用いるのが好ましい．

#### 3.3.4 逆方向写像

逆方向写像とは，計算における写像の方向と，その写像に用いる写像関数の方向が逆となる写像の計算の事を言う．具体的には，写像関数を用いた逆写像，逆写像関数を用いた写像が逆方向写像である．概念図を図 3.6 に示す．



逆方向写像では，写像先の場所を指定することが可能であるため，常に写像先の場所に対応する写像元の場所を求められるのが特徴である．例えば，画像の回転に逆方向写像を用いれば，順方向写像のような穴を生じることなく回転後の画像が計算できる（図 3.7）．

これは，式 (3.10) について

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(-\theta) & \sin(-\theta) \\ -\sin(-\theta) & \cos(-\theta) \end{bmatrix} \begin{bmatrix} x^* \\ y^* \end{bmatrix} \quad (3.11)$$

と書き換えることで計算できる．つまり，回転後の位置  $x^*, y^*$  に対応する回転前の位置  $x, y$  を求めて，その値を画像に書き込むことで画像全体を必ず埋め尽くすようになる．

ここで，式 (3.10) と式 (3.11) の行列について，

$$\begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(-\theta) & \sin(-\theta) \\ -\sin(-\theta) & \cos(-\theta) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.12)$$

となり，式 (3.10) と式 (3.11) は写像・逆写像の関係にあることがわかる．

提案手法では，このような写像の方向に留意して，全ての写像に対して逆方向写像を用いているため，写像後の値は全ての場所で定義されており，順方向写像による穴は存在しない．

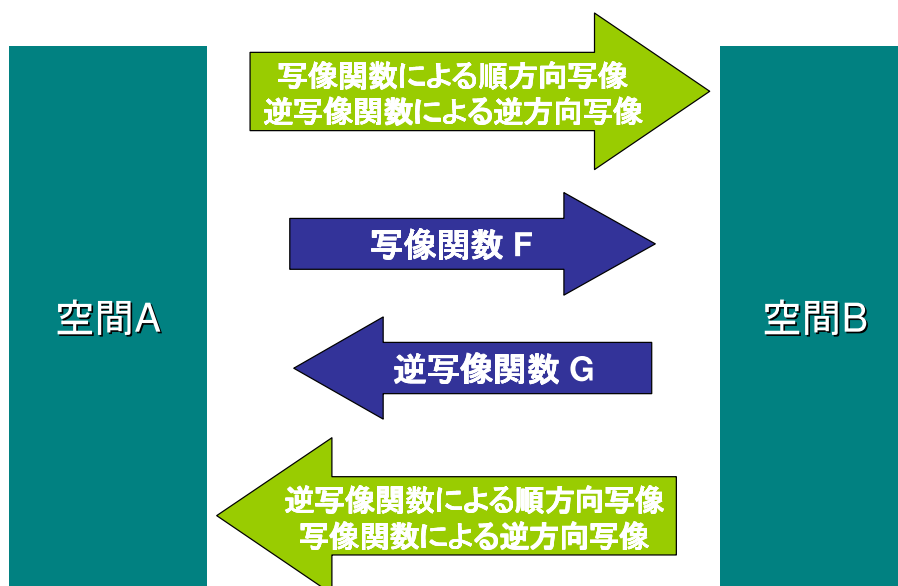


図 3.6: 写像関数と写像方向の概念



図 3.7: 写像としての画像の回転（左：元画像，中央：逆方向写像，右：順方向写像）

### 3.4 写像関数の定式化

#### 3.4.1 速度場が時間定常の場合

まずはじめに、速度場が定常の場合について移流方程式の写像関数の定式化を示す。移流の写像関数とは、移流前の空間から移流後の空間への写像を行う関数である。従って、その写像関数を用いれば、移流前の位置から移流後の位置が得られるはずである。

移流前の位置は、移流後の位置に対して上流にあると捉えることができるため、そのような関数は以下のように定義できる。

$$\frac{\partial \mathbf{F}(\mathbf{x}, t)}{\partial t} = \mathbf{u}(\mathbf{F}(\mathbf{x}, t)) \quad (3.13)$$

つまり、速度場が時間定常の場合の移流の写像関数は、時刻  $t$  で  $\mathbf{F}(\mathbf{x}, t)$  に存在する粒子の流線になっている。

この関数において、 $\mathbf{x}$  は粒子のラベルを示し、関数の値は粒子の位置を示す。従って、この関数について  $\mathbf{F}(\mathbf{x}, 0) = \mathbf{x}$  と初期化を行えば、時刻  $t$  にある粒子が移流後にどの位置に移動したかが得られる事になる。よって、移流前の位置  $\mathbf{x}$  から移流後の位置  $\mathbf{F}(\mathbf{x}, t)$  が計算されていることになる。

次に、写像の逆写像関数  $\mathbf{G}(\mathbf{x}, t)$  について求める。まず、写像・逆写像の関係から  $\mathbf{F}(\mathbf{x}, t)$  と  $\mathbf{G}(\mathbf{x}, t)$  について

$$\mathbf{F}(\mathbf{G}(\mathbf{x}, t), t) = \mathbf{G}(\mathbf{F}(\mathbf{x}, t), t) = \mathbf{x} \quad (3.14)$$

が成り立つ。今、写像関数  $\mathbf{F}(\mathbf{x}, t)$  は粒子  $\mathbf{x}$  についての流線になっている事を考えると、以下のような結合則が成り立つ。

$$\mathbf{F}(\mathbf{F}(\mathbf{x}, t), s) = \mathbf{F}(\mathbf{x}, t + s) \quad (3.15)$$

ここで、 $s = -t$  とおけば

$$\mathbf{F}(\mathbf{F}(\mathbf{x}, t), -t) = \mathbf{F}(\mathbf{x}, 0) = \mathbf{x} \quad (3.16)$$

従って、 $\mathbf{F}(\mathbf{x}, -t)$  は  $\mathbf{F}(\mathbf{x}, t)$  に対して、写像関数・逆写像関数の関係(式(3.14))を満たしていることになる。つまり  $\mathbf{G}(\mathbf{x}, t) = \mathbf{F}(\mathbf{x}, -t)$  となる。ここで  $\mathbf{F}(\mathbf{x}, t)$  の定義から、

$$\frac{\partial \mathbf{F}(\mathbf{x}, -t)}{\partial t} = -\mathbf{u}(\mathbf{F}(\mathbf{x}, -t)) \quad (3.17)$$

となるため、速度場が定常の場合その写像関数・逆写像関数は

$$\begin{aligned} \frac{\partial \mathbf{F}(\mathbf{x}, t)}{\partial t} &= \mathbf{u}(\mathbf{F}(\mathbf{x}, t)) \\ \frac{\partial \mathbf{G}(\mathbf{x}, t)}{\partial t} &= -\mathbf{u}(\mathbf{G}(\mathbf{x}, t)) \end{aligned} \quad (3.18)$$

と定式化できる。

### 3.4.2 速度場が時間非定常の場合

次に，速度場が時間非定常の場合について定式化を示す．まず，写像関数については速度場が時間定常の時と同様に

$$\frac{\partial \mathbf{F}(\mathbf{x}, t)}{\partial t} = \mathbf{u}(\mathbf{F}(\mathbf{x}, t), t) \quad (3.19)$$

となる．速度場が時間定常の場合には，写像関数  $\mathbf{F}(\mathbf{x}, t)$  の値は流線になっていたのに対して，速度場が時間非定常の場合には写像関数  $\mathbf{F}(\mathbf{x}, t)$  の値は流跡線となっている．従って，写像関数の定式化については速度が時間非定常  $\mathbf{u}(\mathbf{x}, t)$  となるだけである．

しかし，速度場が時間定常の場合の流線とは違い，移流後の位置  $\mathbf{F}(\mathbf{x}, t)$  を通る流跡線は必ずしも移流前の位置を通るとは限らない．従って流跡線については，結合則式 (3.16) が成り立たない．そのため，別のアプローチとして，写像関数・逆写像関数の関係 (式 (3.14)) から逆写像関数を定式化をする．

まず，写像関数・逆写像関数の関係から以下のような式が成り立つ．

$$\mathbf{G}(\mathbf{F}(\mathbf{x}, t), t) = \mathbf{x} \quad (3.20)$$

ここで，両辺について  $t$  で微分すれば，右辺は  $t$  によらず一定 ( $\mathbf{x}$ ) なので

$$\frac{d\mathbf{F}(\mathbf{G}(\mathbf{x}, t), t)}{dt} = 0 \quad (3.21)$$

左辺は合成関数の微分法により

$$\frac{\partial \mathbf{G}(\mathbf{x}, t)}{\partial t} \frac{\partial \mathbf{F}(\mathbf{G}(\mathbf{x}, t), t)}{\partial \mathbf{G}(\mathbf{x}, t)} + \frac{\partial \mathbf{F}(\mathbf{G}(\mathbf{x}, t), t)}{\partial t} = 0 \quad (3.22)$$

式 (3.22) について，式を簡潔に書くために，左辺第一項を

$$\frac{\partial \mathbf{F}(\mathbf{G}(\mathbf{x}, t), t)}{\partial \mathbf{G}(\mathbf{x}, t)} = \frac{\partial \mathbf{F}(\mathbf{y}, t)}{\partial \mathbf{y}} \quad (3.23)$$

とおく．式 (3.22) の左辺第二項については，式 (3.19) の定義から  $\mathbf{F}(\mathbf{x}, t)$  が速度  $\mathbf{u}(\mathbf{F}(\mathbf{x}, t), t)$  で流れる粒子の位置であることがわかるので，粒子の初期位置として  $\mathbf{G}(\mathbf{x}, t)$  を代入すれば

$$\frac{\partial \mathbf{F}(\mathbf{G}(\mathbf{x}, t), t)}{\partial t} = \mathbf{u}(\mathbf{F}(\mathbf{G}(\mathbf{x}, t)), t) = \mathbf{u}(\mathbf{x}, t) \quad (3.24)$$

これらを式 (3.22) に代入すれば

$$\frac{\partial \mathbf{G}(\mathbf{x}, t)}{\partial t} \frac{\partial \mathbf{F}(\mathbf{y}, t)}{\partial \mathbf{y}} + \mathbf{u}(\mathbf{x}, t) = 0 \quad (3.25)$$

次に  $\frac{\partial \mathbf{G}(\mathbf{x}, t)}{\partial t}$  について解くと

$$\frac{\partial \mathbf{G}(\mathbf{x}, t)}{\partial t} = -\frac{1}{\frac{\partial \mathbf{F}(\mathbf{y}, t)}{\partial \mathbf{y}}} \mathbf{u}(\mathbf{x}, t) \quad (3.26)$$

また，式 (3.20) から

$$\frac{1}{\frac{\partial \mathbf{F}(\mathbf{y}, t)}{\partial \mathbf{y}}} = \frac{\partial \mathbf{G}(\mathbf{x}, t)}{\partial \mathbf{x}} = \nabla \mathbf{G}(\mathbf{x}, t) \quad (3.27)$$

ここで，左辺では  $y = G(x, t)$  の位置での微分になっているのに対して，右辺では  $x$  の位置での微分になっている事に注意する．これを式 (3.26) に代入すると

$$\frac{\partial G(x, t)}{\partial t} = -\nabla G(x, t) \cdot \mathbf{u}(x, t) \quad (3.28)$$

従って

$$\frac{\partial G(x, t)}{\partial t} + \mathbf{u}(x, t) \cdot \nabla G(x, t) = 0 \quad (3.29)$$

となることがわかる．つまり，逆写像関数は写像関数に対する Euler 形式の移流方程式 (式 (2.1)) になっている事がわかる．これらをまとめると

$$\begin{aligned} \frac{\partial F(x, t)}{\partial t} &= u(F(x, t), t) \\ \frac{\partial G(x, t)}{\partial t} + \mathbf{u}(x, t) \cdot \nabla G(x, t) &= 0 \end{aligned} \quad (3.30)$$

となる．この式は一見すると同一の移流方程式を解いているように見えるが，写像関数の  $x$  は粒子のラベル (移流前の位置) をあらわしているのに対して，逆写像関数の  $x$  は格子のラベル (移流後の位置) を表しているという違いがある．

## 3.5 移流項の計算

### 3.5.1 写像による移流の計算

提案手法では，移流後の物理量  $H(x, t)$  は移流前の物理量の写像によって行われる．つまり

$$H(F(x, t), t) = H_{pre}(x) \quad (3.31)$$

となる．ここで  $F(x, t)$  は移流の写像関数， $H_{pre}(x)$  は移流前の物理量である．そのため式 (3.31) は，移流前の空間にある物理量を，移流後の空間へと写像して，移流後の物理量を求めていることになる．なお，移流前の分布については初期値として  $H_{pre}(x) = H(x, t)$  を与える．

既に述べたように，写像の方向と写像関数の方向が一致する順方向写像は数値計算上好ましくない．この場合，写像の方向は「移流前の空間」から「移流後の空間」であり，写像関数は「移流前の空間」から「移流後の空間」へと写像する関数であるから，式 (3.31) は順方向写像となっている．

そのため，以下のように逆写像関数を使った逆方向写像により写像を行う．

$$H(x, t) = H_{pre}(G(x, t)) \quad (3.32)$$

これによって，写像の方向は「移流前の空間」から「移流後の空間」であるが，逆写像関数は「移流後の空間」から「移流前の空間」へと写像する関数であるから，式 (3.32) は，数値的に安定な逆方向写像となっている．

Semi-Lagrangian 法と同様に， $G(x, t)$  の位置は一般に計算点の上にないため， $H_{pre}(G(x, t))$  を求めるためには補間が必要である．提案手法の移流項についての空間精度は，この補間の精度で決定する．例えば，1次補間を用いると空間1次精度となる．これは，Semi-Lagrangian 法と同じであり，従来の Semi-Lagrangian 法で使われてきた様々な補間関数 [7][26] が利用できる．

### 3.5.2 逆写像関数の更新

既に述べたように，逆写像関数についての定式化は速度が時間定常の場合と非定常の場合で異なる．

まず，速度が時間定常の場合は

$$\frac{\partial G(x, t)}{\partial t} = -u(G(x, t)) \quad (3.33)$$

となる．これは，Lagrange 形式の移流方程式と同じであり， $G(x, t)$  という位置を持つ粒子について速度場に乗った追跡を行えばよい．具体的には，提案手法では修正 Euler 法によって以下のように更新している．

$$\begin{aligned} \hat{G}(x, t) &= G(x, t) - \Delta t u(G(x, t)) \\ G(x, t + \Delta t) &= G(x, t) - 0.5\Delta t \left( u(G(x, t)) + u(\hat{G}(x, t)) \right) \end{aligned} \quad (3.34)$$

次に，速度場が時間非定常の場合は

$$\frac{\partial G(x, t)}{\partial t} + u(x, t) \frac{\partial G(x, t)}{\partial x} = 0 \quad (3.35)$$

となる．これは Euler 形式の非保存形の移流方程式 (式 (3.1)) と同じであり，逆写像関数  $G(x, t)$  の値を移流する物理量として扱えばよい．具体的には，提案手法では 1 次精度の Semi-Lagrangian 法によって以下のように更新している．

$$G(x, t + \Delta t) = G(x - \Delta t u(x, t), t) \quad (3.36)$$

$x - \Delta t u(x, t)$  での  $G(x, t)$  の値は計算点間で 1 次補間を行って求める．

なお，速度場が時間非定常の場合の定式化を速度場が定常の計算に使用することは可能であるが，適用する問題によって使い分けることが好ましい．何故なら速度場が時間非定常の場合の式は，Euler 的な解法で解く式 (3.36) ために誤差が蓄積するからである．速度場が時間定常の場合の式は，Lagrangian 的な解法によって解く式 (3.34) ことができるため，逆写像関数の更新による誤差が問題にならないためである．

## 3.6 非移流項の計算

### 3.6.1 逆写像による移流前の値の更新

提案手法では，移流後の値は常に移流前の値から構築されるため，あるステップで移流後の値に非移流項を加えたとしても，次のステップの移流後の値には何ら影響が及ばないことになる (図 3.3 参照)．そのため，移流後の値について非移流項を加えた場合には，初期値に対してもその影響を加える必要がある．

移流前の値  $H_{pre}(x)$  に非移流項  $N(x, t)$  を加えるには，移流後の空間から移流前の空間への写像 (移流を写像として考えたときは，逆写像) を求めればよい．何故なら，非移流項  $N(x, t)$  は移流後の空間にあり，移流前の値  $H_{pre}(x)$  は移流前の空間にあるからである．なお，非移流項は移流以外の全ての効果であり，境界条件の適用による値の変化，圧力項，拡散項，局所的な値の変更などが含まれる．

写像による移流の計算と同様に，逆写像関数を用いて逆写像を行う事は順方向写像となるため，数値計算上好ましくない．従ってここでも，写像関数を用いて逆写像を行う逆方向写像を用いる．具体的には移流前の値  $H_{pre}(x)$  を時間に依存する関数として書き換えて，

$$\frac{\partial H_{pre}(x, t)}{\partial t} = N(F(x, t), t) \quad (3.37)$$

仮に Euler 陽解法法によって時間軸を離散化すると

$$H_{pre}(\boldsymbol{x}, t + \Delta t) = H_{pre}(\boldsymbol{x}, t) + \Delta t N(\boldsymbol{F}(\boldsymbol{x}, t), t) \quad (3.38)$$

となる．

ここで非移流項  $N(\boldsymbol{x}, t)$  は移流後の空間で解いているため，写像関数を意識することなく，移流後の値  $H(\boldsymbol{x}, t)$  などから求めることができる．つまり，非移流項の計算については従来の Semi-Lagrangian 法から変更する必要がない．

### 3.6.2 写像関数の更新

写像関数の更新は，既に述べたように，以下のような式を解くことで行われる．

$$\frac{\partial \boldsymbol{F}(\boldsymbol{x}, t)}{\partial t} = \boldsymbol{u}(\boldsymbol{F}(\boldsymbol{x}, t), t) \quad (3.39)$$

ただし，初期値として  $\boldsymbol{F}(\boldsymbol{x}, 0) = \boldsymbol{x}$  とする．これは，速度場が時間定常の場合の逆写像関数の更新と同じ (式 (3.34)) Lagrange 形式の移流方程式であり， $\boldsymbol{F}(\boldsymbol{x}, t)$  という位置を持つ粒子について速度場に乘った追跡を行えばよい．具体的には，提案手法では修正 Euler 法によって以下のように更新している．

$$\begin{aligned} \hat{\boldsymbol{F}}(\boldsymbol{x}, t) &= \boldsymbol{F}(\boldsymbol{x}, t) + \Delta t \boldsymbol{u}(\boldsymbol{F}(\boldsymbol{x}, t)) \\ \boldsymbol{F}(\boldsymbol{x}, t + \Delta t) &= \boldsymbol{F}(\boldsymbol{x}, t) + 0.5 \Delta t \left( \boldsymbol{u}(\boldsymbol{F}(\boldsymbol{x}, t)) + \boldsymbol{u}(\hat{\boldsymbol{F}}(\boldsymbol{x}, t)) \right) \end{aligned} \quad (3.40)$$

式 (3.34) との違いは速度場の符号だけである．

## 3.7 再初期化

提案手法では，写像関数  $\boldsymbol{F}(\boldsymbol{x}, t)$  と逆写像関数  $\boldsymbol{G}(\boldsymbol{x}, t)$  とを時間発展によって更新するため，数値的な誤差が蓄積してしまう．特に，速度場が非定常の場合の逆写像関数は (式 (3.36)) Euler 的な解法によって解くしかないため，その誤差が問題となる．また，仮に誤差が蓄積しないとしても，移流によって変化する写像関数の「ゆがみ」は，写像を計算する際の解像度のばらつきを引き起こすため，抑制することが好ましい．

従って，提案手法では写像関数の再初期化という処理を導入する．具体的には

$$\begin{aligned} \boldsymbol{F}(\boldsymbol{x}, t) &\Leftarrow \boldsymbol{x} \\ \boldsymbol{G}(\boldsymbol{x}, t) &\Leftarrow \boldsymbol{x} \\ H_{pre}(\boldsymbol{x}, t) &\Leftarrow H(\boldsymbol{x}, t) \end{aligned} \quad (3.41)$$

という処理を再初期化として行う．ここで  $\Leftarrow$  は右辺から左辺への代入を示す．

しかし，この再初期化では移流前の値  $H_{pre}(\boldsymbol{x}, t)$  を移流による誤差を含む  $H(\boldsymbol{x}, t)$  によって置き換えるため，ここで誤差の蓄積が起こる (図 3.3 参照)．そのため，移流による誤差の蓄積を抑えるには不必要な再初期化を行わないことが重要である．

最も単純な再初期化の方針は，経験的に設定したタイムステップ数ごとに再初期化を行うことである．しかしながら，速度場が変化したり時間刻み幅を変えたりすると，写像関数のゆがみの蓄積速度が変わるため，その場合には再初期化の間隔も変化することが望ましい．つまり，写像関数のゆがみを何らかの数値で計算し，その値が一定値を超えないように再初期化を行うことができればよい．そのため，写像関数のゆがみを計算する手法について以下に述べる．

### 3.7.1 写像関数のゆがみ

写像関数のゆがみ  $\|\epsilon\|$  は，以下のように定義される値である．

$$\begin{aligned}\|\epsilon\| &= \sum_{i,j} \epsilon_{i,j}^2 \\ \epsilon_{i,j} &= \frac{1}{2} \left( \frac{\partial \sigma_i}{\partial x_j} + \frac{\partial \sigma_j}{\partial x_i} \right) \\ \sigma_i &= \sigma_i(\mathbf{x}, t) = \mathbf{G}_i(\mathbf{x}, t) - \mathbf{x}\end{aligned}\tag{3.42}$$

ここで， $i, j$  は要素のラベル ( $x, y, z$  など) を示す．つまり，力学などで用いる通常の意味でのひずみテンソル  $\epsilon_{i,j}$  について，それぞれの要素の二乗の和を求めたものとなっている．

なお，同じような定式化は Neyret[17] によって提案されており，Neyret はテクスチャ座標について上記の値を求めている．テクスチャ座標はテクスチャマッピングにおける逆写像関数であるため，同様の値を写像関数のゆがみとして用いることは妥当である．

式 (3.42) によって定義されるゆがみは各計算点で求めるため，全ての計算点についてのゆがみの平均値を，再初期化の指標として用いる．つまり，計算点  $k = 1, 2, \dots, N$  について

$$S = \frac{1}{N} \sum_{k=1,2,\dots,N} \|\epsilon_k\|\tag{3.43}$$

という値を計算し， $S$  がある一定値を超えたら再初期化を行う．

## 3.8 Semi-Lagrangian 法との関連

提案手法は再初期化を毎ステップごとに行うことで，Semi-Lagrangian 法と等価な計算手法となる．このことを，速度場が時間非定常の場合の定式化を用いて示す．

まず，写像関数・逆写像関数，および移流前の分布について

$$\begin{aligned}\mathbf{F}(\mathbf{x}, t) &= \mathbf{x} \\ \mathbf{G}(\mathbf{x}, t) &= \mathbf{x} \\ H_{pre}(\mathbf{x}, t) &= H(\mathbf{x}, t)\end{aligned}\tag{3.44}$$

という初期値が与えられているとする．ここで，式 (3.36) に基づき逆写像関数を更新する．

$$\mathbf{G}(\mathbf{x}, t + \Delta t) = \mathbf{G}(\mathbf{x} - \Delta t \mathbf{u}(\mathbf{x}, t), t)\tag{3.45}$$

右辺について，式 (3.44) から

$$\mathbf{G}(\mathbf{x}, t + \Delta t) = \mathbf{x} - \Delta t \mathbf{u}(\mathbf{x}, t)\tag{3.46}$$

従って，この逆写像関数を用いて，写像による移流後の値を求めると

$$H(\mathbf{x}, t + \Delta t) = H_{pre}(\mathbf{G}(\mathbf{x}, t + \Delta t), 0)\tag{3.47}$$

式 (3.46) を代入して

$$H(\mathbf{x}, t + \Delta t) = H(\mathbf{x} - \Delta t \mathbf{u}(\mathbf{x}, t), t)\tag{3.48}$$

つまり，Semi-Lagrangian 法のように，上流の点を求めて，その点での値を現在の位置に代入する処理と等価になる．

この後に写像関数についての更新が入るが，毎ステップごとに再初期化を行うと，移流を求めた後すぐに再初期化が入るため，非移流項の計算に入る前に再び

$$\begin{aligned} \mathbf{F}(\mathbf{x}, t + \Delta t) &= \mathbf{x} \\ \mathbf{G}(\mathbf{x}, t + \Delta t) &= \mathbf{x} \\ H_{pre}(\mathbf{x}, t + \Delta t) &= H(\mathbf{x}, t + \Delta t) \end{aligned} \tag{3.49}$$

となり，写像関数の更新はリセットされる．よって非移流項による移流前の分布の更新を行うと

$$H(\mathbf{x}, t + \Delta t) = H(\mathbf{x}, t + \Delta t) + \Delta t N(\mathbf{F}(\mathbf{x}, t), t) \tag{3.50}$$

式 (3.49) を代入して

$$H(\mathbf{x}, t + \Delta t) = H(\mathbf{x}, t + \Delta t) + \Delta t N(\mathbf{x}, t) \tag{3.51}$$

となる．つまり，Semi-Lagrangian 法と同様に，写像関数を使用しない非移流項の扱いになる．



## 第4章 計算例と検証

提案手法と Semi-Lagrangian 法を比較するため，いくつかの検証計算を行った．その計算結果について考察する．なお全ての計算で，提案手法，Semi-Lagrangian 法ともに空間 1 次精度（補間関数に 1 次関数を使用）かつ時間 2 次精度（時間発展に修正 Euler 法を使用）である．

### 4.1 速度が時間定常の場合

#### 4.1.1 一次元の速度定常移流

最も単純な計算として，以下のような速度一定の 1 次元移流について解いた結果を示す．

$$\frac{\partial H(x,t)}{\partial t} + U \frac{\partial H(x,t)}{\partial x} = 0 \quad (4.1)$$

計算条件として，図 4.1 のように計算領域全体の長さを 1.0 として等間隔に 500 分割し， $U = 0.1, \Delta t = 0.004, \Delta x = 0.002$  とした．

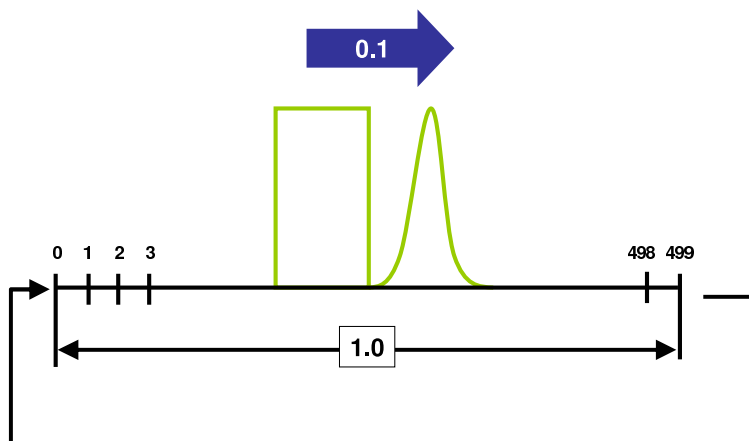


図 4.1: 1 次元速度定常移流の計算体系

従って Courant 数は  $C = U\Delta t/\Delta x = 0.2$  である．なお，初期値として

$$H(x,0) = \begin{cases} 1.0 & 0.4 \leq x < 0.5 \\ \exp(-750(x-0.6)^2) & 0.5 \leq x < 0.7 \\ 0.0 & \text{other} \end{cases} \quad (4.2)$$

を与えた．境界は周期境界とし，2500 ステップの計算を行った．1 ステップでは  $\Delta tU = 0.0004$  だけ分布が平行移動するので，2500 ステップ後の解析解は初期値と一致するはずである．

この計算体系では写像関数のゆがみが生じないため，提案手法における写像関数の再初期化は必要ない．従って，再初期化を行わないで得られた結果を図 4.2 に示す．1 次元の速度定常移流は，移流方程式の解法の精度を検証する問題として良く使われているが [28]，この問題に提案手

法を用いるとほぼ解析解に一致した結果が得られることがわかる。これは提案手法では、再初期化を行わない限り誤差が蓄積しないからである。つまり、計算体系として写像関数のゆがみが生じない体系であれば、提案手法はかなり高い精度で移流を解くことができると言える。

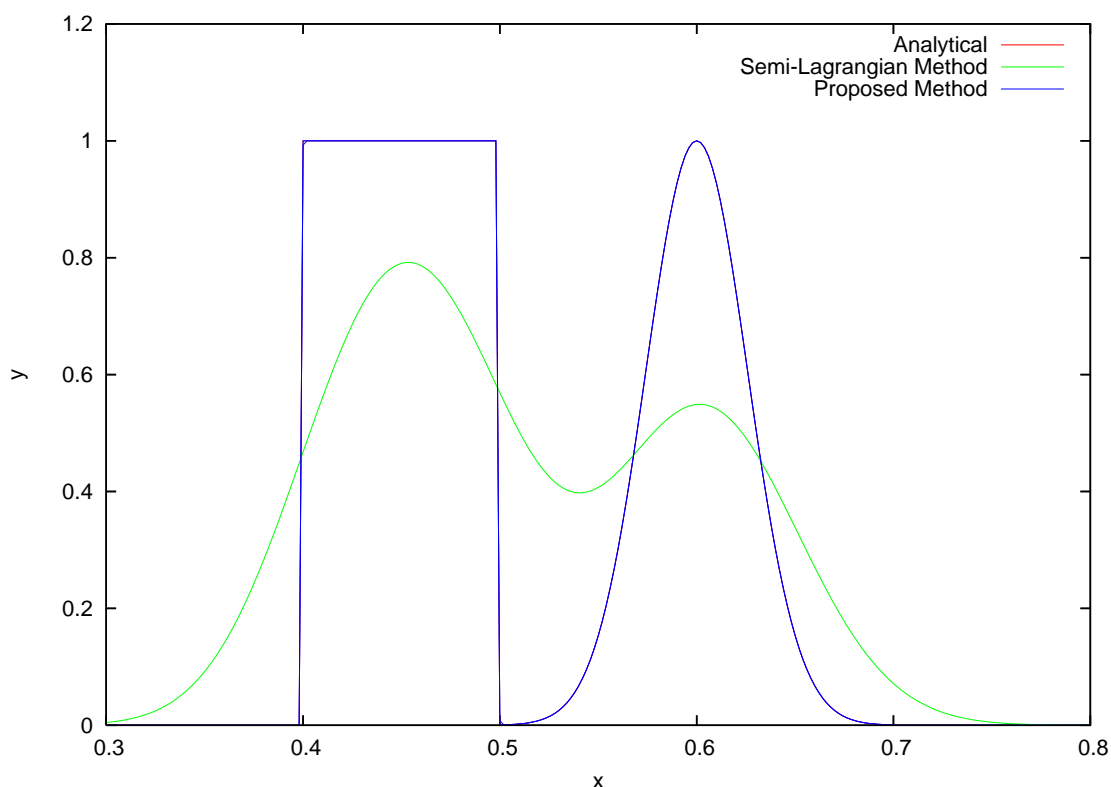


図 4.2: 1次元速度定常移流の計算結果 (再初期化なし)

提案手法は、Semi-Lagrangian 法では毎ステップで蓄積していた補間誤差を、複数ステップで蓄積するようにすることで、最終的な誤差を減らしている。従って、最悪で2ステップごとに再初期化を行ったとしても従来手法よりも誤差が小さくなることを図 4.3 に示す。この場合、解析解との差の L2 ノルムは、Semi-Lagrangian 法が 3.9554 で、2ステップで再初期化を行った提案手法が 3.5672 であり、空間時間精度が同じであっても、再初期化の間隔を2以上にすることで誤差を減らすことができている。これは、再初期化間隔を人為的に設定するのは、それほど難しくないと示している。

再初期化の間隔を大きくすると誤差が減ることを確認するため、再初期化の間隔を強制的に設定して(12ステップ, 32ステップ, 92ステップごと)得られた結果を図 4.4 に示す。再初期化間隔と誤差(解析解との差の L2 ノルム)についての変化は図 4.5 に示す。図 4.4 を見ると、再初期化の間隔が大きくなるに従って解析解の形状へと近づいていることがわかる。数値的な誤差については図 4.5 のように、再初期化の間隔を大きくするに従って小さくなっている。

図 4.5 で誤差が急激に0に近づいている点では、Courant 数と再初期化間隔の積が整数になっている。つまり、写像を計算するときちょうど計算点に一致した位置で移流前の値を求められるため、補間が必要なくなるということである。このことを示すため、図 4.6 には Courant 数(この計算では0.2)と再初期化間隔の積が整数になる場合の結果を示す。実際に、Courant 数と再初期化間隔の積が整数(1, 2, 3)になる場合では、補間誤差がほとんど生じずに、解析解にほぼ一致する結果が得られている。従って、提案手法では、再初期化間隔の設定と問題によっては、ほぼ解析解と一致する結果が得られることがわかった。

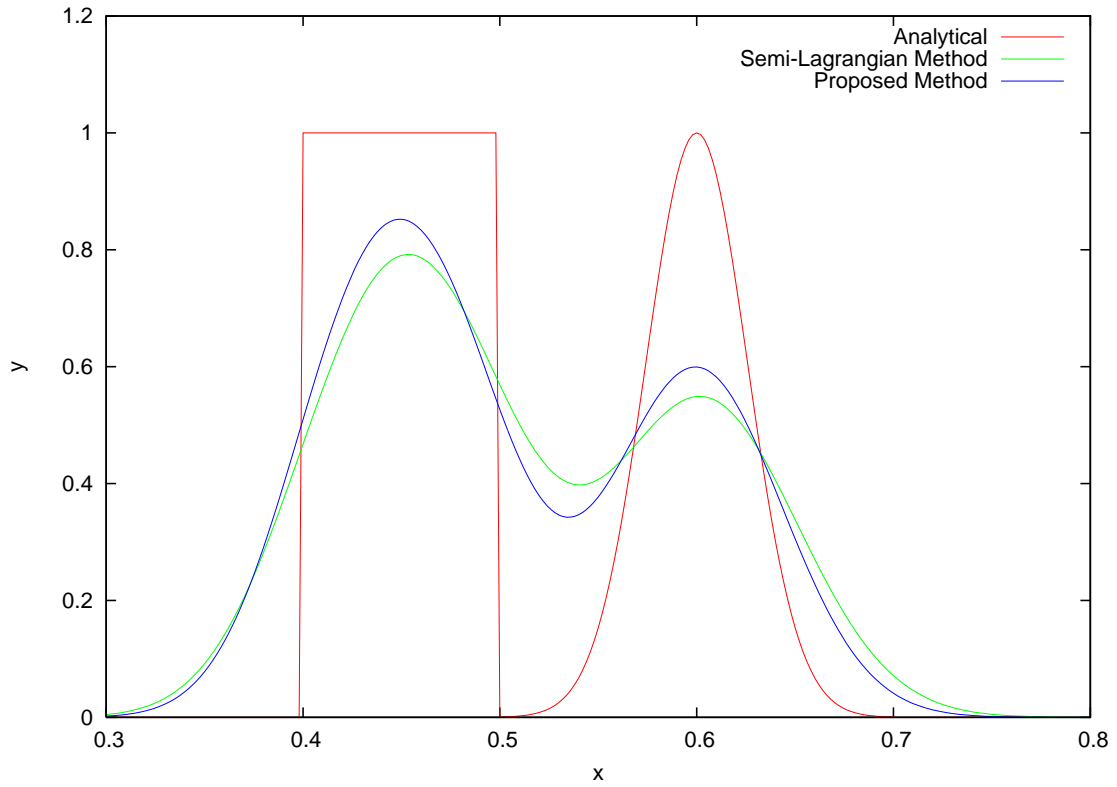


図 4.3: 1次元速度定常移流の計算結果 (2ステップごとの再初期化)

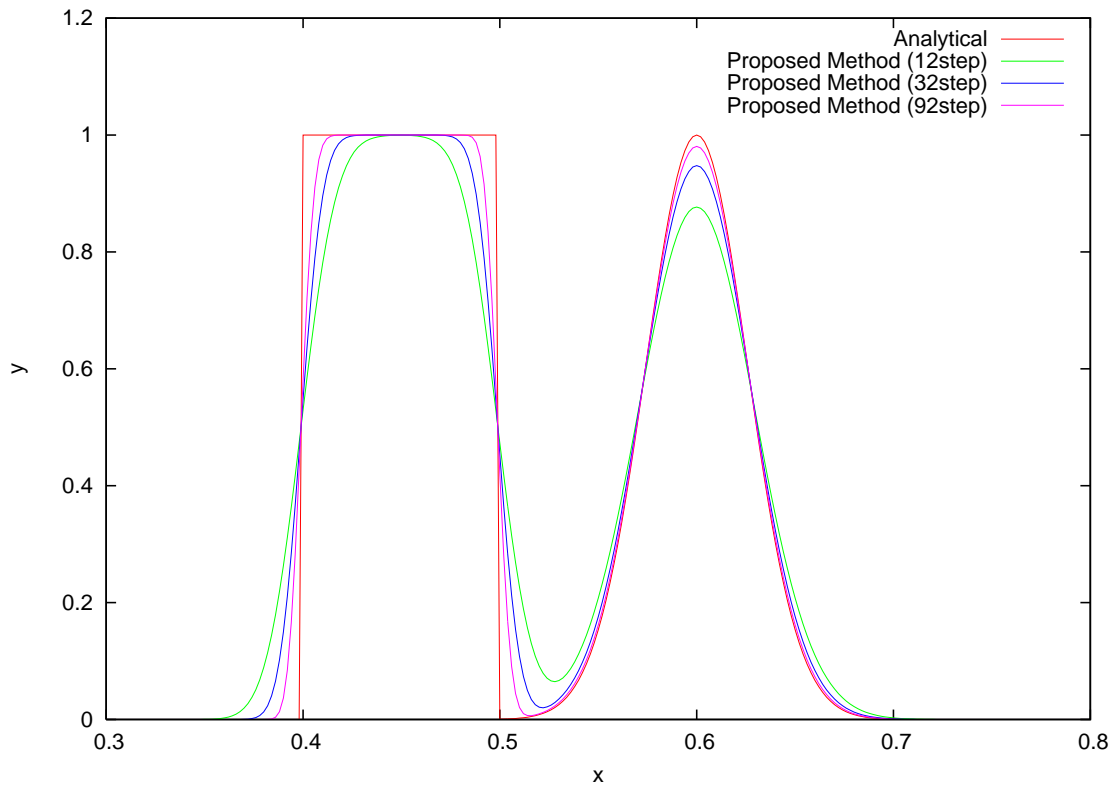


図 4.4: 1次元速度定常移流の計算結果 (再初期化あり)

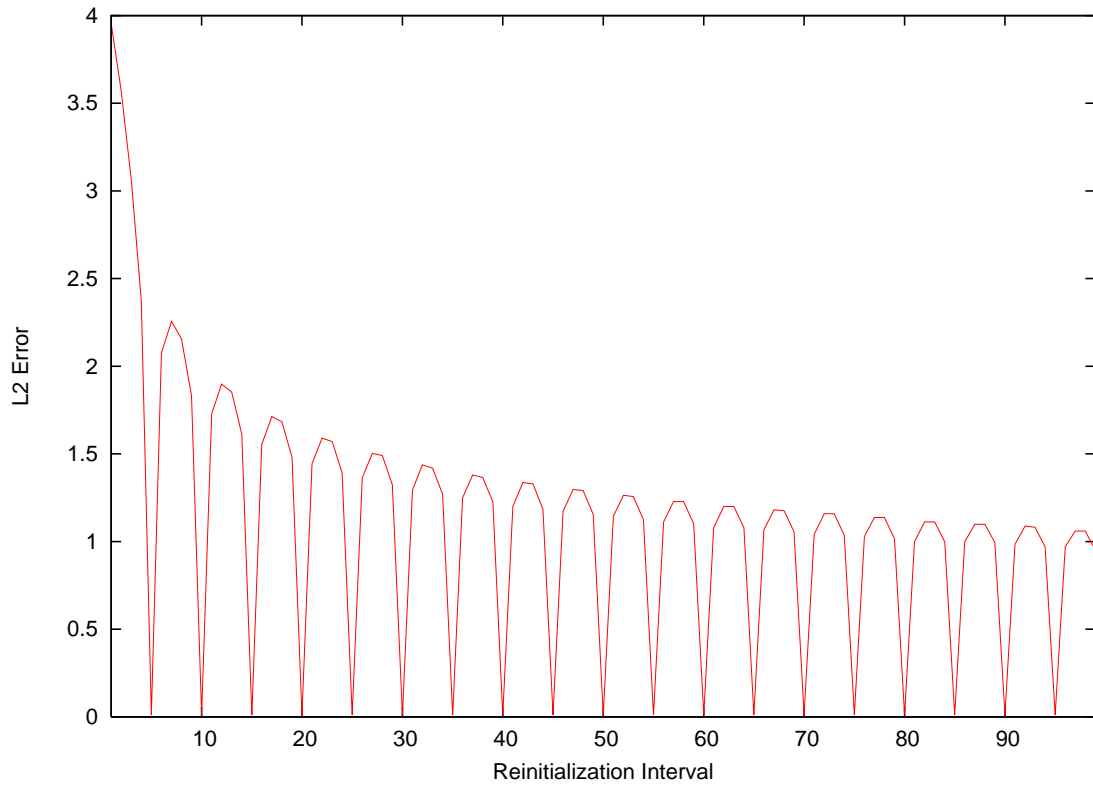


図 4.5: 再初期化による誤差の変化

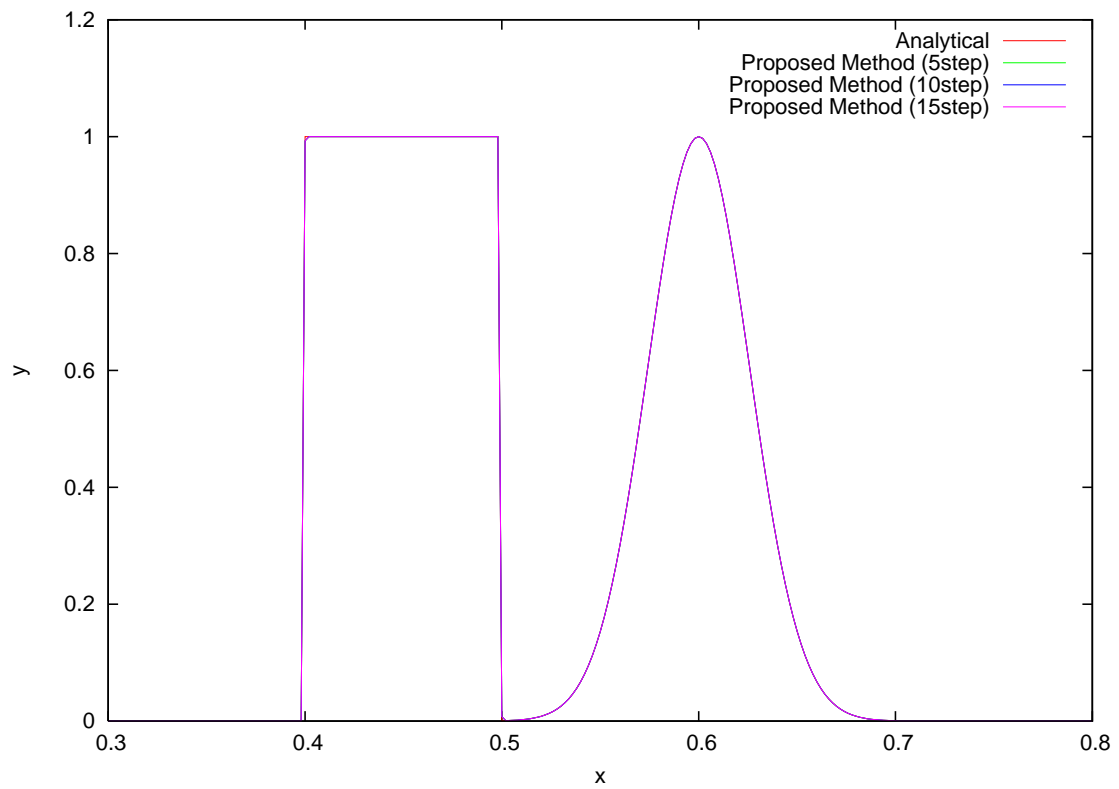


図 4.6: Courant 数と再初期化間隔の値が整数になる場合の結果

#### 4.1.2 二次元の剛体回転

二次元の速度時間定常移流の計算例として, Zalesak[27] によって提案された, 時間定常な速度場での剛体の回転を行った. この場合の支配方程式は以下ようになる.

$$\frac{\partial H(x, y, t)}{\partial t} + u(x, y) \frac{\partial H(x, y, t)}{\partial x} + v(x, y) \frac{\partial H(x, y, t)}{\partial y} = 0 \quad (4.3)$$

剛体の形状は図 4.7 に示す形状であり, 計算体系については図 4.8 に示す. つまり, 計算体系の大きさは  $([0.0, 1.0], [0.0, 1.0])$  で,  $t = 0$  で剛体は  $(0.5, 0.25)$  に置かれている. 速度場としては

$$\begin{aligned} u(x, y) &= \frac{\pi}{314} (0.5 - y) \\ v(x, y) &= \frac{\pi}{314} (x - 0.5) \end{aligned} \quad (4.4)$$

を与える. なお,  $\Delta t = 0.01$  であり,  $\Delta x = \Delta y = 1/N$  ( $N$  は  $x$  方向,  $y$  方向の分割数) である. 従って, 628 タイムステップで剛体が  $(0.5, 0.5)$  を中心にちょうど一回転して初期値と同じ位置に来ることになる.

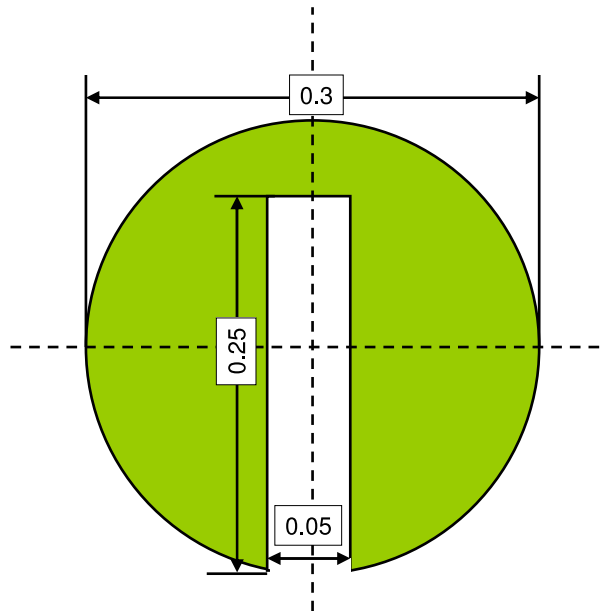


図 4.7: 移流させる剛体

移流する物理量としては, Level Set 法 [18] で用いられている符号付距離関数を使った. 符号付距離関数とは, 絶対値が最も近い面からの距離で, 剛体の内側なら負の値外側なら正の値になるような関数である. つまり, 剛体の表面は符号付距離関数が 0 となる場所で定義される. 実際に図 4.7 について正確な符号付距離関数を求めるのは困難なため, 以下のような近似値を用いた.

$$\begin{aligned} r(x, y) &= \sqrt{(x - 0.15)^2 + (y - 0.15)^2} - 0.15 \\ H(x, y, 0) &= \begin{cases} -r(x, y) & r(x, y) < 0 \& |x - 0.15| < 0.025 \& (y - 0.15) > -0.10 \\ r(x, y) & other \end{cases} \end{aligned} \quad (4.5)$$

誤差の評価方法として, 剛体表面の位置が正しく追跡されていることが重要なため, Enright ら [4] の式を用いて誤差を評価した.

$$Error_{rigid} = \int |S(H_{rotated}(\mathbf{x})) - S(H_{initial}(\mathbf{x}))| dx dy \quad (4.6)$$

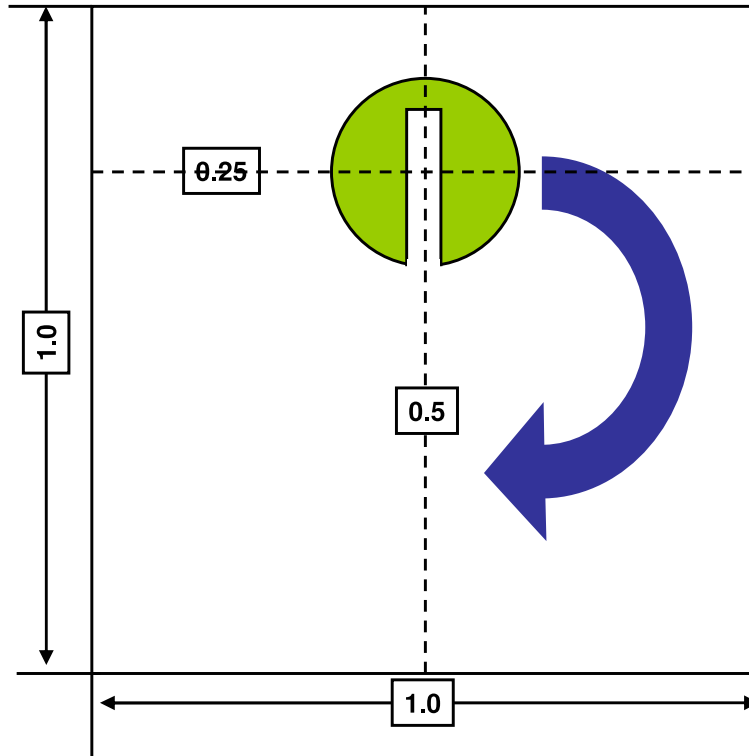


図 4.8: 2次元剛体回転の計算体系

ここで  $S(r)$  は

$$S(r) = \begin{cases} 1 & r \leq 0 \\ 0 & r > 0 \end{cases} \quad (4.7)$$

と定義される関数である．つまり，剛体内部で1，剛体の外で0となる．実際には領域を  $1000^2$  に分割し，補間によって得られた値で評価した．

まず図 4.9 に Semi-Lagrangian 法による結果を示す．計算結果は空間を  $128^2$ ,  $256^2$  で分割して得られた結果である．既に述べたように，剛体の表面は  $H(x, y, t) = 0$  で定義されるので， $H(x, y, t) = 0$  についての等高線を描いている．Semi-Lagrangian 法では，数値誤差の蓄積によって剛体の表面の位置が大きく変化してしまっていることがわかる．特に，中央の切れ込み部分については  $256^2$  で計算した結果であっても表現できていないことがわかる．

次に図 4.10 に提案手法による結果を示す．解像度は  $128^2$  で，再初期化は10ステップごとに行った．図 4.9 の Semi-Lagrangian 法による結果と比べると，倍の解像度の  $256^2$  の結果と比較しても，提案手法はより高精度に剛体表面の位置を捉えている．

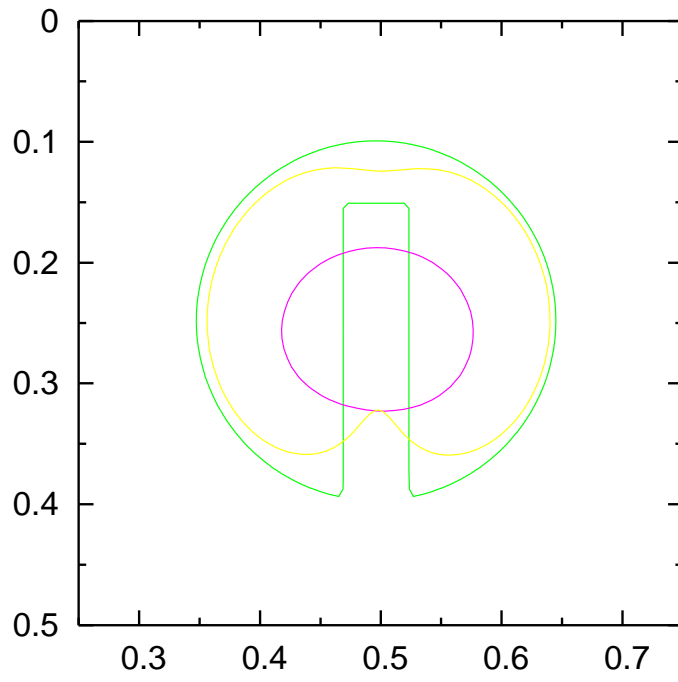


図 4.9: Semi-Lagrangian 法による結果, 解析解 (緑),  $128^2$ (ピンク),  $256^2$ (黄色)

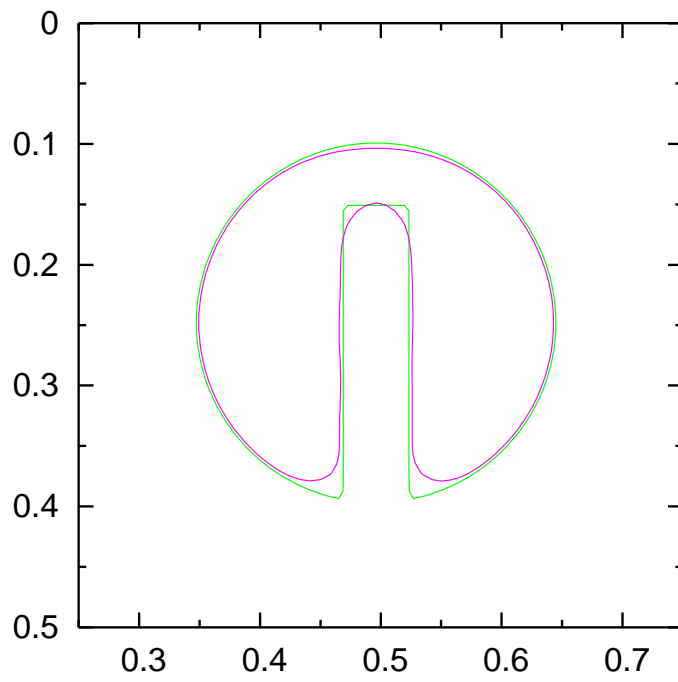


図 4.10: 提案手法による結果, 解析解 (緑), 提案手法 (10 ステップ毎) (ピンク)

図 4.11 には再初期化の間隔を増やした場合の，式 (4.6) によって定義される誤差の変化を示す．なお，比較として Semi-Lagrangian 法による誤差も同時に示す．再初期化の間隔がおよそ 10 までに急峻に誤差が減少しており，再初期化間隔 16 では解像度  $512^2$  で Semi-Lagrangian 法を用いた結果と，ほぼ等しい誤差になっていることがわかる．

表 4.1 には，同様の問題を解いている Enright らの結果 [4] との比較を示す．誤差の評価については Enright らの手法と統一した．Enright らの Particle Level Set Method は剛体の境界に粒子を集中して配置することで，計算の精度を上げている．ところが，提案手法は境界について特別な扱いをしていないにも関わらず，Particle Level Set Method に匹敵する結果が得られている．また，提案手法は空間 1 次精度で計算しているにも関わらず，空間 5 次精度の WENO よりも低い解像度で，精度の良い結果が得られていることがわかる．

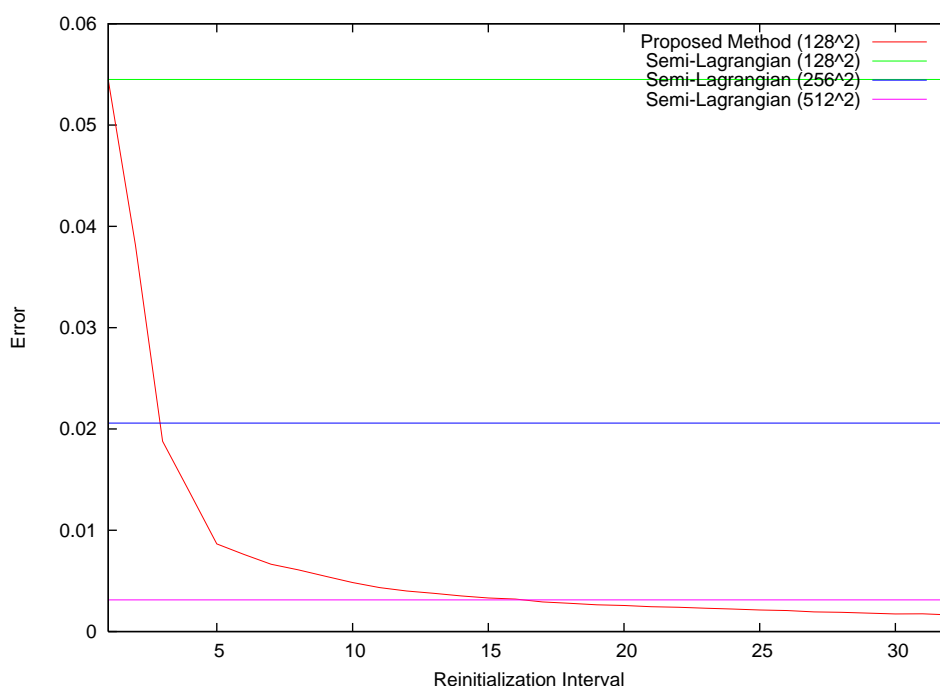


図 4.11: 再初期化による誤差の変化

	式 (4.6) による誤差
Semi-Lagrangian 法 ( $128^2$ )	0.6976
Semi-Lagrangian 法 ( $256^2$ )	0.2634
Semi-Lagrangian 法 ( $512^2$ )	0.0401
提案手法 (4 ステップ毎) ( $128^2$ )	0.1754
提案手法 (16 ステップ毎) ( $128^2$ )	0.0411
提案手法 (32 ステップ毎) ( $128^2$ )	0.0209
5 次精度の WENO ( $200^2$ ) [4]	0.0800
Particle Level Set Method ( $200^2$ ) [4]	0.0200

表 4.1: 従来手法と提案手法の計算誤差の比較 (括弧内は計算の解像度)



## 4.2 速度が時間非定常の場合

### 4.2.1 流入を含む二次元の移流

速度が非定常の場合の計算例として，流入を含む密度場の移流を計算する．この場合の支配方程式は以下ようになる．

$$\frac{\partial H(x, y, t)}{\partial t} + u(x, y) \frac{\partial H(x, y, t)}{\partial x} + v(x, y) \frac{\partial H(x, y, t)}{\partial y} = N(x, y, t) \quad (4.8)$$

計算体系の大きさは  $([0.0, 1.0], [0.0, 1.0])$  で，等間隔格子に区切って計算している．密度場の初期値は  $H(x, y, 0) = 0$  である．時間刻み幅  $\Delta t$  は，全ての計算で  $\Delta t = 0.002$  で，628 ステップまで計算した．速度場については

$$\begin{aligned} u1(x, y) &= \frac{\pi}{314} (0.5 - y) \\ v1(x, y) &= \frac{\pi}{314} (x - 0.5) \end{aligned} \quad (4.9)$$

および

$$\begin{aligned} \Psi(x, y) &= \frac{1}{4\pi} \sin(4\pi(x + 0.5)) \cos(4\pi(y + 0.5)) \\ u2(x, y) &= \frac{\partial \Psi(x, y)}{\partial y} \\ v2(x, y) &= -\frac{\partial \Psi(x, y)}{\partial x} \end{aligned} \quad (4.10)$$

を用いて，以下のように与えた．

$$\begin{aligned} c(t) &= \frac{t}{628\Delta t} \\ u(x, y, t) &= (1 - c(t)) u1(x, y) + c(t) u2(x, y) \\ v(x, y, t) &= (1 - c(t)) v1(x, y) + c(t) v2(x, y) \end{aligned} \quad (4.11)$$

つまり，計算開始時は  $u1(x, y), v1(x, y)$  だった速度場が，628 タイムステップ後に  $u2(x, y), v2(x, y)$  へ線形に変化する．従ってこの速度場は空間時間非定常である．非移流項  $N(x, y, t)$  については

$$\begin{aligned} s(t) &= \cos\left(16\pi \frac{t}{628\Delta t}\right) \\ r(x, y, t) &= (x - 0.5)^2 + (y - 0.25 + 0.1s(t))^2 \\ N(x, y, t) &= \begin{cases} 512(0.0005 - r(x, y, t)) & r(x, y, t) < 0.0005 \\ 0 & other \end{cases} \end{aligned} \quad (4.12)$$

とした．つまり， $(0.5, 0.25)$  を中心に  $0.1$  の範囲で振動する流入条件となっている．計算体系の模式図を図 4.12 に示す．

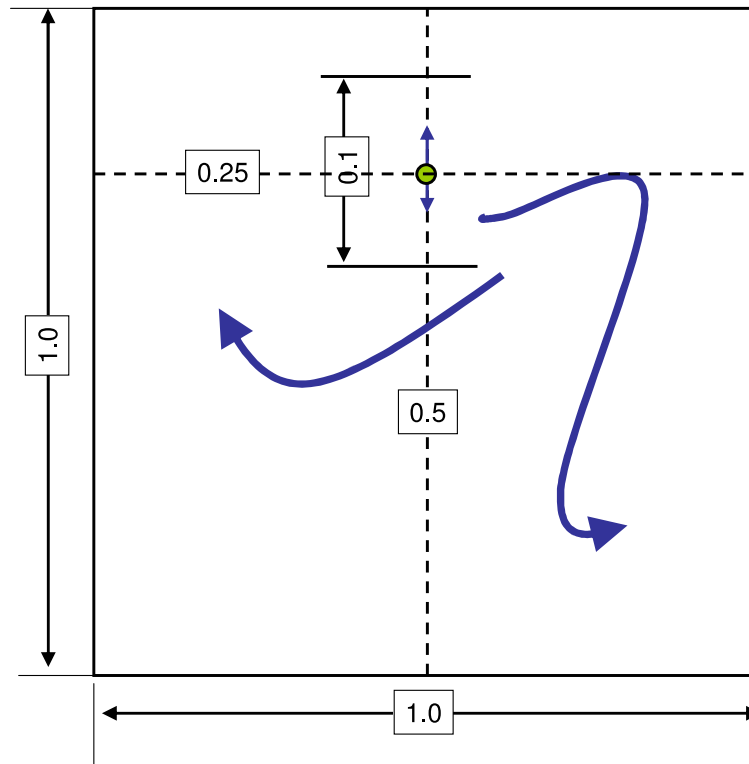


図 4.12: 流入を含む二次元移流の計算体系

これらの条件で、解像度を変えて Semi-Lagrangian 法で計算した結果を図 4.13 に示す。また、提案手法によって解像度を  $128^2$  に固定して写像関数を再初期化する写像関数のゆがみについての閾値を変えて計算した結果を図 4.13 に示す。図 4.13 と図 4.13 を比較すると、Semi-Lagrangian 法による低解像度 ( $128^2$ ,  $256^2$ ) の結果では、数値誤差によって消えてしまっている渦の構造が、提案手法では  $128^2$  でも解像できており、定性的に良い結果が得られている。また、この計算では時間空間で変化する非移流項  $N(x, y, t)$  が含まれるが、提案手法においても、写像関数によって正しく非移流項  $N(x, y, t)$  を扱えていることがわかる。仮に写像関数を更新しない場合の結果は図 4.15 のようになり、定性的に異なる結果になってしまっている。

提案手法の誤差について定量的に評価するため、速度時間定常の計算例と同じく、計算誤差と再初期化の関連について図 4.16 に示す。なお、この計算では厳密解を計算するのは不可能なため、図 4.13 に示した Semi-Lagrangian 法による  $1024^2$  の結果を厳密解として誤差を計算した。図 4.16 を見るとある値までは誤差が減少し、その後は誤差が逆に増加している。

この原因は、速度が時間非定常の場合には、逆写像関数の更新に通常の移流方程式と同様の数値誤差が現れるためである。この場合、再初期化をある程度の頻度で行わなければ、逆写像関数についての誤差が蓄積してしまう。従って、効果的に提案手法を用いるには、再初期化の間隔について適切な値を設定する必要がある。

しかしながら、誤差が増加している範囲においても、提案手法は同じ解像度 ( $128^2$ ) での Semi-Lagrangian 法よりも誤差が小さいため、再初期化の頻度は決定が難しいパラメータではないと言える。実際の問題では、まず安全なパラメータとして再初期化の頻度を十分に高くしておき、移流における誤差をさらに減少させたい場合に再初期化の頻度を下げて調整するといった方法が考えられる。

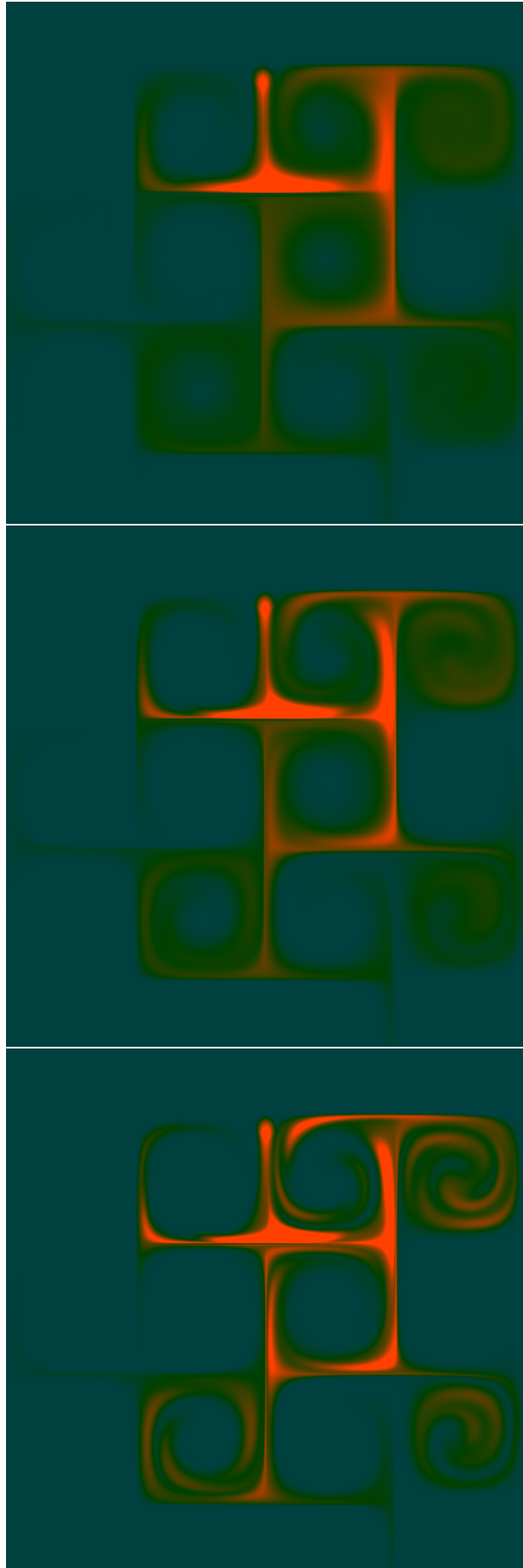


図 4.13: Semi-Lagrangian 法による結果 (解像度, 上  $128^2$ , 中  $256^2$ , 下  $1024^2$ )

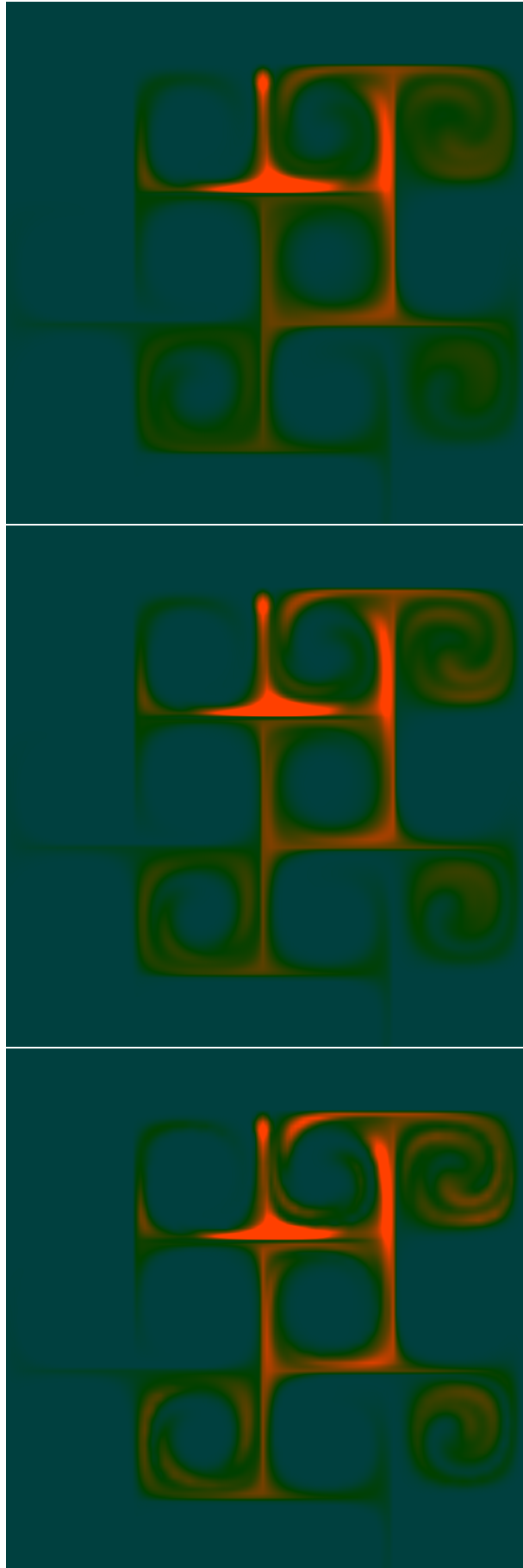


図 4.14: 提案手法法による結果 (ゆがみの閾値, 上 0.2, 中 0.6, 下 1.8)

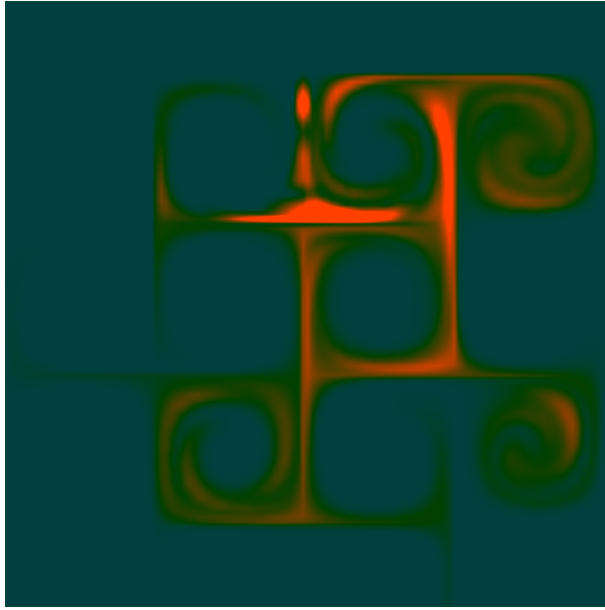


図 4.15: 提案手法で写像関数を更新しない場合の結果 (ゆがみの閾値, 1.8)

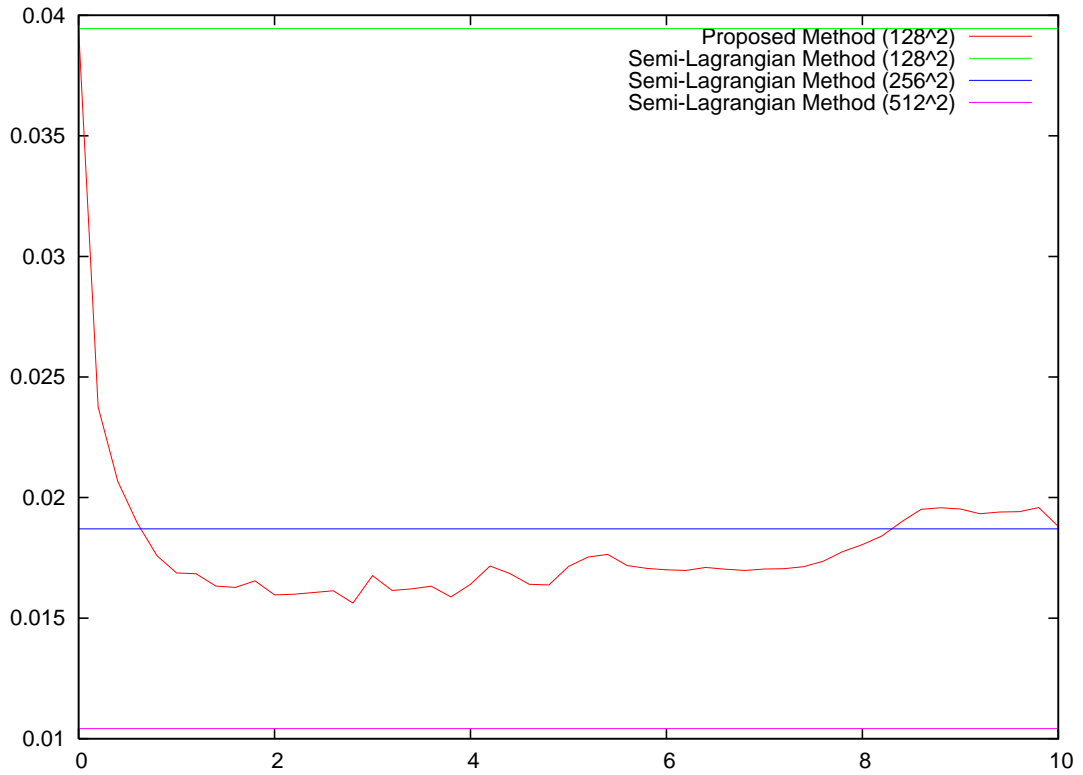


図 4.16: 再初期化による誤差の変化

## 4.2.2 非圧縮流体計算

速度が時間非定常で、かつ非線形（速度場が自身の速度で移動する場合）の計算として Euler 方程式に基づく非圧縮性流体の計算を行った．支配方程式は以下ようになる．

$$\nabla \cdot \mathbf{u} = 0 \quad (4.13)$$

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \nabla p + \mathbf{f} \quad (4.14)$$

ここで、 $\mathbf{f}$  は外力、 $p$  は圧力を示す．なお、表記を簡潔にするため  $\mathbf{u}(x, t) = \mathbf{u}$ 、 $p(x, t) = p$  とした．また、二次元では  $\nabla = (\partial/\partial x, \partial/\partial y)$ 、三次元では  $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)$  である．

本研究では、Fractional Step 法によって Euler 方程式を解いている．まず、式 (4.14) から  $p$  についての項を無視して時間軸を Euler 法によって離散化し、仮の速度場  $\mathbf{u}^*$  を求める．

$$\frac{\mathbf{u}^* - \mathbf{u}}{\Delta t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} + \mathbf{f} \quad (4.15)$$

次に、 $\mathbf{u}^*$  を用いて圧力  $p$  を以下の Poisson 方程式で計算する．

$$\nabla^2 p = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^* \quad (4.16)$$

そして、求められた圧力  $p$  を用いて、式 (4.13) を満たす  $\mathbf{u}$  を計算する．

$$\mathbf{u} = \mathbf{u}^* - \Delta t \nabla p \quad (4.17)$$

これを 1 ステップ分として計算を繰り返すことによって、流体の運動の時間発展を求める．煙の密度など、受動的に運動する物理量については、上記で求められた速度場を用いて移流させている．なお、以下の 2 つの応用ではどちらも上記の手法によって流体の運動を計算している．

### 流体のキーフレーム制御

Fattal ら [6] は、流体シミュレーションのコンピューターグラフィックスへの応用として、ユーザーが任意に与えた形状（キーフレーム）に流体の形状をコントロールする手法を提案している（図 4.17）．

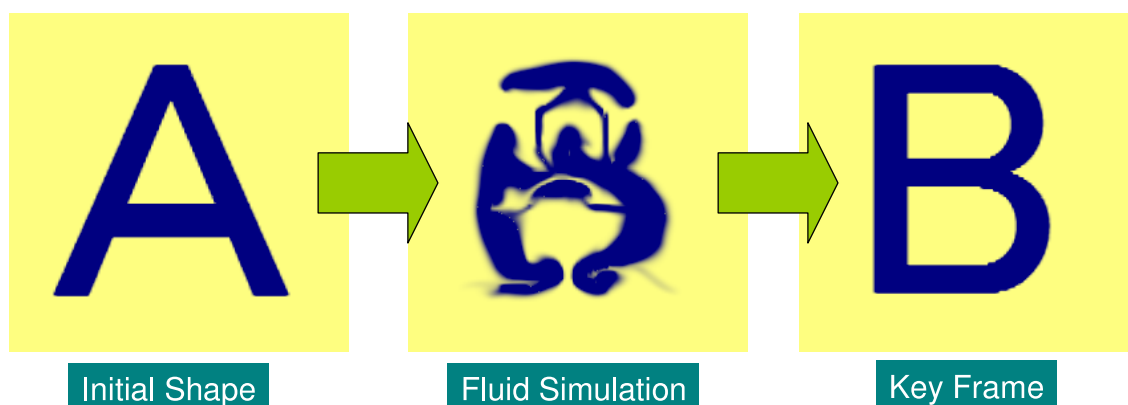


図 4.17: 流体のキーフレーム制御の概念

Fattal らは、従来の Semi-Lagrangian 法では移流方程式における数値誤差が大きく、複雑な形状のキーフレームを扱うことが難しいことを指摘している．そのため、強制的にキーフレームの

位置に流体を流入させることで、複雑な形状を扱う手法を提案している。しかしながら、この流入は流体の物理的な挙動ではないため、視覚的に不自然な挙動となることも指摘しており [6]、本質的には移流方程式における数値誤差を抑えることによって改善するとしている。

このような背景を受けて、提案手法を Fattal ら [6] の研究に基づいた流体のキーフレーム制御に応用した。その結果を図 4.19 に示す。計算の解像度は Semi-Lagrangian 法、提案手法共に  $128^2$  である。提案手法では、5 ステップ毎に再初期化を行った。設定したキーフレームおよび初期の分布は図 4.18 に示す。

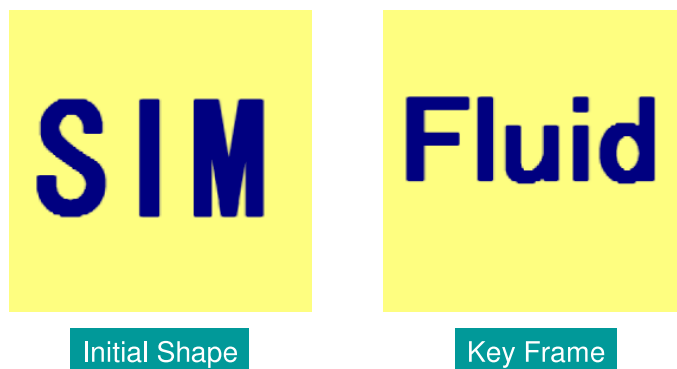


図 4.18: 計算に用いたキーフレームと初期分布



図 4.19: 流体のキーフレーム制御の結果, 上: Semi-Lagrangian 法, 下: 提案手法

図 4.19 を見ると、Semi-Lagrangian 法による結果では、キーフレームの”Fluid”の  $d$  について、中央の穴が数値誤差によって塞がれてしまっている。一方、提案手法では移流による数値誤差を減らすことによって、 $d$  についても形状が上手く表現できていることがわかる。

## 煙のシミュレーション

Fedkiw らは [7] Euler 方程式に基づいた、煙のアニメーションの手法を提案している。Fedkiw らは、従来の Semi-Lagrangian 法では、数値誤差によって細かい渦が拡散してしまい、煙の写実的なアニメーションが計算できない事を指摘し、Vorticity Confinement を導入した。

Vorticity Confinement [23] とは、数値誤差による細かいスケールの渦の拡散を防ぐ項を外力項として加える手法である。この項を加えることで、移流による数値誤差で散逸した渦を再び加えることが可能となる。しかしながら、この外力項は物理的な意味を持つ外力項ではないため、その大きさによっては流体が非物理的な運動をしてしまう。Selle らは [19] この外力項の大きさを渦粒子による別のシミュレーションから決めることを提案しているが、やはりパラメータによっては非物理的な挙動となることを指摘している。

提案手法は、移流方程式を解いたときの誤差そのものを小さくするため、Vorticity Confinement のような非物理的な項を加えることなく、細かい渦を維持した計算ができることが予想される。そのため、Fedkiw らは [7] の実装に基づいて、移流方程式の解法について Semi-Lagrangian 法を用いた場合と、提案手法に変えた場合について比較を行った。なお、全ての計算で提案手法は 4 ステップごとの再初期化を行っている。図 4.20 に  $64^3$  で計算した結果、図 4.21 に  $128^3$  で計算した結果を示す。初期値として  $(0.5, 0.8, 0.5)$  を中心とした一辺 0.25 の立方体の領域に、密度を 1 として与えた。

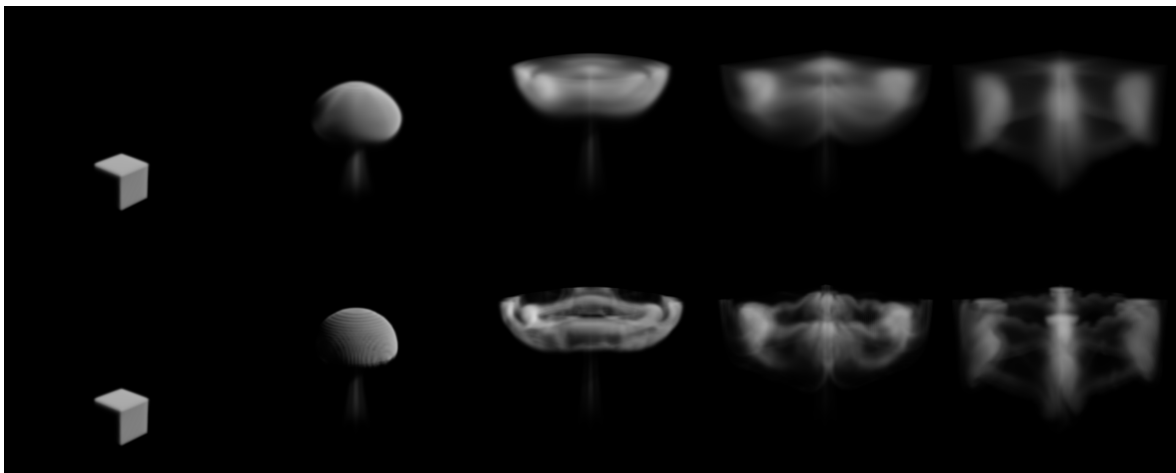


図 4.20: 煙のシミュレーション,  $64^3$ , 上: Semi-Lagrangian 法, 下: 提案手法

既に述べたように、この計算では Euler 方程式を用いているため、厳密解では煙の密度の拡散や渦の散逸は起こらない。そのため、計算結果に現れている煙の拡散や渦の散逸は、移流方程式による数値拡散が主な原因である。図 4.20 と図 4.21 を見ると、同じ解像度で、提案手法では細かい渦が表現できていることがわかる。計算時間については、 $64^3$  の場合について表 4.2 に示す。提案手法では、3次元の写像関数・逆写像関数について移流を計算しなければならないため、移流項については提案手法はおよそ 3 倍の計算時間がかかっている。しかし、この計算では圧力の Poisson 方程式を解くことに最も時間がかかっているため、移流項については相対的に計算時間が短く、全体としての計算時間はほとんど差がない。従って、提案手法を用いることで、ほぼ同じ計算時間でより詳細な煙の挙動が計算できる。

図 4.22 には  $64^3$  で計算した、障害物を含む煙のシミュレーションを示す。このような障害物について、Fedkiw ら [7] の実装では、障害物について法線方向の速度をゼロにする境界条件を設定している。提案手法でもこの境界条件を非移流項として扱うことで同様に計算を行う事ができている。また、図 4.20 や図 4.21 と同様に、提案手法の方が細かい流れが計算できている事がわかる。



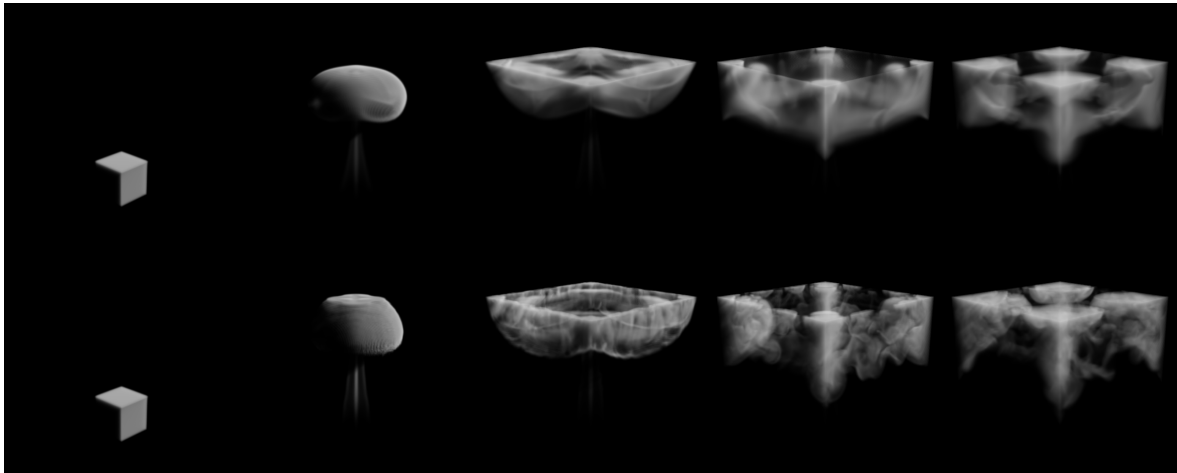


図 4.21: 煙のシミュレーション,  $128^3$ , 上: Semi-Lagrangian 法, 下: 提案手法

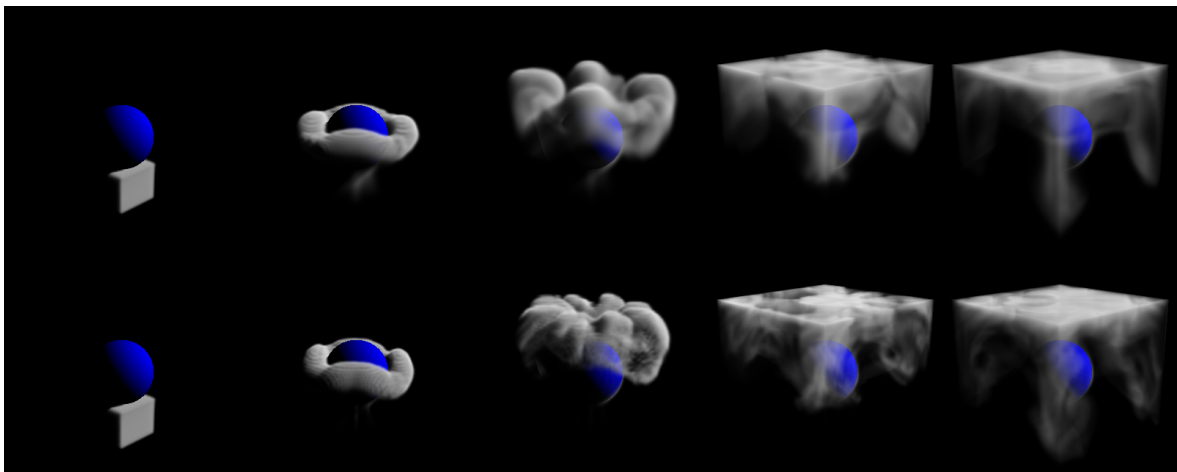


図 4.22: 障害物を含む煙のシミュレーション,  $64^3$ , 上: Semi-Lagrangian 法, 下: 提案手法

	Semi-Lagrangian 法	提案手法
移流項	320	960
圧力項	1510	1730
全体	2430	3145

表 4.2: Semi-Lagrangian 法と提案手法の計算時間の平均値 (単位は [msec])

## 第5章 結論

本研究では，移流方程式を高精度に解くための手法として，写像関数による移流方程式の解法を提案した．また，従来はテクスチャによる流れ場の可視化で使われていた定式化が，実は移流方程式の写像関数・逆写像関数を求める式となっている事を示した．

提案手法では，移流方程式における数値誤差が毎ステップでの数値誤差の蓄積の結果であることに注目して，誤差の蓄積を遅くすることで，最終的な数値誤差を減らしている．その結果，本研究で扱った計算例では，同じ解像度・次数であっても従来手法よりも精度の良い結果が得られることがわかった．

今後は，提案手法を様々な問題に適用することで，その適用範囲や有効性を確認したい．特に，再初期化を行わない間に写像関数に蓄積するゆがみが，最終的に計算にどのような影響を与えるかについてはさらなる検討が必要である．

# 謝辞

本研究を進めるにあたって、多くの方にお世話になりました。

指導教員の越塚先生には、研究の進め方について多くのアドバイスをいただきました。また、研究室では粒子法をメインに研究しているにも関わらず、粒子法とは関係のない、一学生の研究アイデアを認めていただき、大変感謝しています。さらには、個人的な事情により研究に割ける時間が無かった時期があったのを、大変柔軟に対応していただき、とても助かりました。

コース長の岡本先生には、本研究をはじめのきっかけとなったアイデアについて、そのままでは研究にまで発展しなかったであろうところを、研究を進めるよう後押しをしていただきました。この後押しがなければ、ここまで研究を進めることはできませんでした。この場を借りてお礼を述べます。

研究員の鈴木さんには、写像関数の定式化について貴重なご意見をいただきました。また、数値解析上の提案手法の性質についての様々な議論は、本研究を進める上で非常に役立ちました。また、同じ研究室の先輩方には、研究についてのアドバイスを数多くいただき、研究に関して一人で考えてると見逃していたであろう点について、気がつくことができました。

平成 18 年 蜂須賀恵也

## 参考文献

- [1] C. Batty and B. Houston. The visual simulation of wispy smoke. In *Proceedings of the SIGGRAPH 2005 Conference on Sketches & Applications*. ACM Press, 2005.
- [2] D. J. Benson. Computational methods in lagrangian and eulerian hydrocodes. *Comput. Methods Appl. Mech. Eng.*, 1992.
- [3] R. Courant, E. Issacson, and M. Rees. On the solution of nonlinear hyperbolic differential equations by finite differences. *Communication on Pure and Applied Mathematics* 5, 243–255., 1952.
- [4] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell. A hybrid particle level set method for improved interface capturing. *J. Comput. Phys.* 183, 83-116, 2002.
- [5] D. Enright and R. Marschner, S.and Fedkiw. Animation and rendering of complex water surfaces, 2002.
- [6] R. Fattal and D. Lischinski. Target-driven smoke animation. *ACM Trans. Graph.*, Vol. 23, No. 3, pp. 441–448, 2004.
- [7] R. Fedkiw, J. Stam, and H. W. Jensen. Visual simulation of smoke. *SIGGRAPH 2001: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001.
- [8] S. Godunov. Finite difference method for numerical computation of discontinuous solution of the equations of fluid dynamics. *Matematicheskii Sbornik*, 47:271, translated from Russian by I. Bohachevsky., 1959.
- [9] A. Harten. On a class of high order resolution total-variation-stable finite-difference schemes. *SIAM Journal of Numerical Analysis* 21, 1-23., 1974.
- [10] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes iii. *J. Comput. Phys.* 71, 231-303, 1987.
- [11] B. Jobard, G. Erlebacher, and M. Y. Hussaini. Hardware-accelerated texture advection for unsteady flow visualization. *11th IEEE Visualization 2000 (VIS'00)*, 2000.
- [12] B. Jobard, G. Erlebacher, and M. Y. Hussaini. Lagrangian-eulerian advection of noise and dye textures for unsteady flow visualization. *IEEE Transactions on Visualization and Computer Graphics*, 2002.
- [13] S. Koshizuka and Y. Oka. Moving particle semi-implicit method : A gridless approach based on particle interactions for incompressible flow simulation. *Proc. 3rd Workshop on Supersimulators for Nuclear Power Plants*, 1995.

- [14] X.-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.* 126, 202-212, 1996.
- [15] N. Max and B. Becker. Flow visualization using moving textures. *ICASW/LaRC Symposium on Visualizing Time-Varying Data*, 1995.
- [16] J. J. Monaghan. An introduction to sph. *J. Comput. Phys.*, 1988.
- [17] F. Neyret. Advected textures. *ACM-SIGGRAPH/EG Symposium on Computer Animation (SCA)*, 2003.
- [18] S. Osher and R. Fedkiw. The level set method and dynamic implicit surfaces. *Springer-Verlag*, 2002.
- [19] A. Selle, N. Rasmussen, and R. Fedkiw. A vortex particle method for smoke, water and explosions. *ACM Trans. Graph.*, Vol. 24, No. 3, pp. 910–914, 2005.
- [20] O. Y. Song, H. Shin, and H. S. Ko. Stable but nondissipative water. *ACM Trans. Graph.*, Vol. 24, No. 1, pp. 81–97, 2005.
- [21] J. Stam. Stable fluids. *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999.
- [22] A. Staniforth and J. Cote. Semi-lagrangian integration schemes for atmospheric models - a review. *Monthly Weather Review* 119, 2206-2223, 1991.
- [23] J. Steinhoff and D. Underhill. Modification of the euler equations for vorticity confinement : Application to the computation of interacting vortex rings. *Physics of Fluids*, 6(8):2738. 2744, 1994.
- [24] D. Weiskopf, T. Schafhitzel, and T. Ertl. Real-time advection and volumetric illumination for the visualization of 3d unsteady flow. *Proceedings of Eurovis 2005 (EG/IEEE TCVG Symposium on Visualization)*, 13-20, 2005.
- [25] M. Wiebe and B. Houston. The tar monster: Creating a character with fluid simulation. In *Proceedings of the SIGGRAPH 2004 Conference on Sketches & Applications*. ACM Press, 2004.
- [26] T. Yabe and T Aoki. A universal solver for hyperbolic equations by cubic-polynomial interpolation. *Computational Physics Commun.* 66, 219-242., 1991.
- [27] S. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Comput. Phys.* 31, 335.362, 1979.
- [28] 矢部孝, 尾形陽一, 内海隆行. C I P 法 原子から宇宙までを解くマルチスケール解法. 森北出版, 2003.
- [29] 越塚誠一. 粒子法 計算力学レクチャーシリーズ. 丸善, 2005.