

論文 / 著書情報  
Article / Book Information

題目(和文)	局所補間微分オペレータ法による数値流体計算手法の研究
Title(English)	
著者(和文)	桜井浩一
Author(English)	
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第4929号, 授与年月日:2001年9月30日, 学位の種別:課程博士, 審査員:
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第4929号, Conferred date:2001/9/30, Degree Type:Course doctor, Examiner:
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

# 局所補間微分オペレータ法による 数値流体計算手法の研究

指導教官： 青木 尊之 教授  
堀岡 一彦 教授

総合理工学研究科創造エネルギー専攻

99D53050 桜井 浩一

2001 年 7 月 26 日

# 謝辞

本論文を執筆にあたり多くの方々に御指導、御助力を頂きましたことを大変感謝しております。

青木先生には、全く数値計算についての知識が無い修士1年のころより丁寧な御指導を頂き、論文投稿、学会発表、外部機関との共同研究など、研究者としてのスタンスを学ぶ良い機会をふんだんに与えて頂き、様々な経験を得ることができました。時にはディスカッションが白熱し、貴重な御時間を費して頂いたことが、いまさらながら先生の優しさを実感させます。

学部生の頃に研究室所属して以来これまで堀岡先生には大変な御迷惑をお掛けして来たことと思います。私の身勝手から数値計算に関わる研究を自由に進めさせて頂き、専門分野に関するディスカッションでも厳しい御指導と優しいご支援により大変楽しい研究を進めることができました。あわせまして中島助手には日頃よりの私の粗暴な態度にも関わらず、研究に関するディスカッションでは物理学的、数学的見地より叱咤、激励を頂きました。さまざまな失礼を働いたにも関わらず寛大に接して頂き、改めて私の精神的未熟さを反省しました。

本論文の一節となりました Semi-Lagrange CIP 法の構築に関しましては矢部先生、肖先生の御指導を頂き、世界の研究者の集うドイツ ブレーメンにおける国際会議 ISCFD 1999 にて矢部先生の御尽力により未熟な私が口頭発表できましたことはこれまでに無い良い経験になりました。また CFD Journal への投稿に際しても多くのご支援を頂きありがとうございます。

乱流 Burgers 方程式の節では店橋先生の御指導を頂き、また、検討のためにわざわざスペクトル法にて数値計算結果をお送り頂きましたこと、大変ありがとうございました。急な申し出にも関わらずスピーディーなレスポンスを頂きましたおかげで時間内に論文にまとめることができました。

残念ながら私の不徳により十分な結果が得られなかったため本論文に掲載できませんでしたが物質工学工業技術研究所の吉田先生には、材料破壊に関する御指導を頂き、大阪大学レーザー研究所や、物質研における実験見学など貴重な経験をさせて頂きありがとうございました。

また、細貝博士、長谷川博士、佐竹博士、平岡博士をはじめとする諸先輩方、同期、後輩の方々には学生生活を通じて、いろいろなアドバイスや、ご支援を賜わり、また御迷惑をお掛けしました。

ひとえに並々ならぬ皆様の御支援により本論文といった形で私の研究生活を綴ることができました。どうもありがとうございました。

最後に、今まで暖かく見守って下さいました両親への感謝の言葉を持ちまして謝辞を終らせて頂きます。

# 目次

1 序論	1
2 IDO 法の計算精度の評価	4
2.1 IDO 法で用いられる補間関数の高次微係数精度	4
2.1.1 補間関数の空間精度	6
2.1.2 3 次風上補間の高次微係数精度	10
2.1.3 5 次風上補間の高次微係数精度	12
2.1.4 5 次中心補間の高次微係数精度	15
2.2 IDO 法の計算手続きと CIP 法の計算手続きとの比較	17
2.3 単純な偏微分方程式を用いた基礎精度検証	35
2.4 まとめ	66
3 圧縮性流体問題への適用	67
3.1 1 次元衝撃波管問題 (1 次元 Riemann 問題)	67
3.1.1 人工粘性に対する考察	67
3.1.2 空間解像度、補間関数に対する計算精度の考察	71
3.1.3 1 次元衝撃波管問題のまとめ	76
3.2 1 次元乱流 Burgers 方程式	77
3.2.1 スペクトル法との比較	77
3.2.2 グリッド解像度、粘性係数 (レイノルズ数) に対する考察	84
3.2.3 まとめ	87
3.3 前線発達問題への適用	88
3.3.1 計算条件ならびに解析解	88
3.3.2 計算結果	88
3.3.3 まとめ	92
3.4 レーリー・テラー不安定性への適用	93
3.4.1 レーリー・テラー不安定性の概要	93
3.4.2 2 次元レーリー・テラー不安定性	93
3.4.3 3 次元レーリー・テラー不安定性	107
3.5 超音速混合流問題への適用	116
3.5.1 超音速混合の概要	116
3.5.2 混合過程における理論	117
3.5.3 基礎方程式	123
3.5.4 多次元圧縮性流体問題に対する人工粘性の検討	124
3.5.5 2 流体モデル	129
3.5.6 計算パラメータ	131
3.5.7 計算結果	133

3.5.8	まとめ	143
3.6	まとめ	143
<b>4</b>	<b>非圧縮性流体解法の高速化</b>	<b>144</b>
4.1	ポアソン方程式解法の高速化	144
4.1.1	ポアソン方程式の空間精度	144
4.1.2	マルチグリッド法の適用	152
4.1.3	レッドブラックアルゴリズムの適用	158
4.1.4	並列計算による計算速度向上	162
4.2	移流計算の高速化 (セミ・ラグランジュ CIP 法)	164
4.2.1	セミ・ラグランジュ手法	164
4.2.2	セミ・ラグランジュ CIP 法の構築	166
4.2.3	計算結果	168
4.2.4	まとめ	201
4.3	まとめ	202
<b>5</b>	<b>陰的 IDO 法による CFL 数に対する安定化</b>	<b>204</b>
5.1	陰解法スキーム	204
5.2	反復法	205
5.3	計算結果	206
5.4	まとめ	218
<b>6</b>	<b>接触不連続面を含む問題への移動カットセル手法の導入</b>	<b>219</b>
6.1	移動カットセル手法	221
6.2	1次元移動カットセル手法	222
6.2.1	カットセルを用いた接触不連続面の定義	222
6.2.2	接触不連続面上の補間関数の形成	223
6.2.3	接触不連続面の移動	225
6.2.4	局所 CFL 数による数値不安定性	227
6.2.5	1次元移動カットセル法のアルゴリズム	227
6.2.6	1次元計算例題	227
6.3	2次元移動カットセル手法	232
6.3.1	2次元移動カットセル導入に関する方針	232
6.3.2	LevelSet 関数構築のためのアルゴリズム	233
6.3.3	LevelSet 関数再構築 (Reinitialization) のためのアルゴリズム	235
6.3.4	2次元接触不連続面の位置更新	235
6.3.5	セミ・ラグランジュ的手法による接触不連続面上の物理量計算	236
6.3.6	2次元移動カットセル手法のアルゴリズム	240
6.3.7	2次元計算例題	241
6.4	まとめ	253
<b>7</b>	<b>結論</b>	<b>254</b>
<b>A</b>	<b>エルミート補間を用いた計算手法</b>	<b>257</b>
A.1	エルミート (Hermite) 補間	257
A.2	Cubic-Interpolated Propagation (CIP) 法	257
A.2.1	1次元 CIP 法	258

A.2.2	2次元 CIP 法	260
A.2.3	3次元 CIP 法	266
A.3	局所補間微分オペレータ法	274
A.3.1	補間関数	274
A.3.2	時間前進	276
A.3.3	多次元 IDO 法	277
<b>B</b>	<b>計算手法</b>	<b>280</b>
B.1	4次精度 Runge-Kutta 法	280
B.2	シンプソン則	280
<b>C</b>	<b>人工粘性</b>	<b>282</b>
C.1	1次元の人工粘性	282
C.2	多次元の人工粘性	284
<b>D</b>	<b>マルチグリッド法</b>	<b>287</b>
D.1	CS 法 (Correction Storage Scheme)	287
<b>E</b>	<b>PC クラスターの製作ならびに MPI を用いた並列プログラミング</b>	<b>290</b>
E.1	PC クラスターシステムの構築	290
E.1.1	ハードウェア構成	290
E.1.2	OS 並びに MPI ライブラリの導入	292
E.1.3	各種設定	292
E.2	MPI を用いた並列コードの開発	294
E.2.1	MPI (Message Passing Interface) ライブラリ	294
E.2.2	MPI で利用できるデータ型	294
E.2.3	グループ、コミュニケータの概念	295
E.2.4	グループ管理	298
E.2.5	コミュニケータ管理	300
E.2.6	プロセストポロジー	301
E.2.7	カルテシアン構造	301
E.2.8	グラフ構造	303
E.2.9	ブロッキング通信、ノンブロッキング通信	306
E.2.10	1対1通信	308
E.2.11	集団通信	315
E.2.12	実装時における注意	321
E.3	実装例	324
E.3.1	計算空間へのカルテシアン構造の導入	324
E.3.2	並列計算のための通信の実装	325
E.3.3	ポアソン方程式の並列解法	327
E.3.4	流体計算の並列解法	328

# 目 次

2.1	補間関数記述精度	8
2.2	補間関数記述精度	9
2.3	3次風上補間から得られた微係数誤差 (Sin プロファイル)	10
2.4	3次風上補間から得られた微係数誤差 (Cos プロファイル)	11
2.5	5次風上補間から得られた微係数誤差 (Sin プロファイル)	14
2.6	5次風上補間から得られた微係数誤差 (Cos プロファイル)	14
2.7	5次中心補間から得られた微係数誤差 (Sin プロファイル)	15
2.8	5次中心補間から得られた微係数誤差 (Cos プロファイル)	16
2.9	不連続を含む波の例	18
2.10	1次元 CIP 法における空間補間誤差 (波数 $k = 2\pi$ )	20
2.11	1次元 CIP 法における空間補間誤差 (波数 $k = \pi$ )	20
2.12	1次元 CIP 法における空間補間誤差 (波数 $k = 0.5\pi$ )	21
2.13	1次元 CIP 法における空間補間誤差	21
2.14	1次元 CIP 法における空間補間誤差 (波数 $k = 2\pi$ )	22
2.15	1次元 CIP 法における空間補間誤差 (波数 $k = \pi$ )	22
2.16	1次元 CIP 法における空間補間誤差 (波数 $k = 0.5\pi$ )	23
2.17	1次元 CIP 法における空間補間誤差	23
2.18	cos, sin でプロファイルを与えた時の CIP 法による空間補間誤差	24
2.19	2次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = 2\pi$ )	26
2.20	2次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = \pi$ )	26
2.21	2次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = 0.5\pi$ )	27
2.22	2次元 CIP 法 Type-B による空間補間誤差 ( $k_x = k_y = 2\pi$ )	27
2.23	2次元 CIP 法 Type-B による空間補間誤差 ( $k_x = k_y = \pi$ )	28
2.24	2次元 CIP 法 Type-B による空間補間誤差 ( $k_x = k_y = 0.5\pi$ )	28
2.25	2次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = 2\pi$ )	29
2.26	2次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = \pi$ )	29
2.27	2次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = 0.5\pi$ )	30
2.28	2次元 CIP 法による空間補間誤差	30
2.29	3次元 CIP 法 Type-A, Type-C による平均空間補間誤差	31
2.30	3次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = k_z = 2\pi$ )	32
2.31	3次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = k_z = \pi$ )	33
2.32	3次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = k_z = 0.5\pi$ )	33
2.33	3次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = k_z = 2\pi$ )	34
2.34	3次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = k_z = \pi$ )	34
2.35	3次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = k_z = 0.5\pi$ )	35
2.36	CIP 法による1次元移流方程式の数値拡散誤差	38
2.37	IDO 法による1次元移流方程式の数値拡散誤差	40

2.38	風上 5 次補間を用いた IDO 法による 1 次元移流方程式の数値拡散誤差	41
2.39	CIP 法による非一様速度場における 1 次元保存方程式の検証	44
2.40	IDO 法による 1 次元保存方程式の数値計算結果	45
2.41	CIP 法 Type-A の 2 次元移流方程式に対する数値拡散誤差	48
2.42	CIP 法 Type-B の 2 次元移流方程式に対する数値拡散誤差	48
2.43	CIP 法 Type-C の 2 次元移流方程式に対する数値拡散誤差	48
2.44	IDO 法による 2 次元移流方程式の数値拡散誤差	49
2.45	CIP 法による Zalesak の例題検証	50
2.46	3 次多項式による風上補間を用いた IDO 法による Zalesak 例題の数値計算結果	51
2.47	5 次多項式による風上補間を用いた IDO 法による Zalesak 例題の数値計算結果	51
2.48	CIP 法による 2 次元保存方程式の検証 (初期プロファイル)	53
2.49	CIP 法 Type-A による 2 次元保存方程式の検証	53
2.50	CIP 法 Type-B による 2 次元保存方程式の検証	54
2.51	CIP 法 Type-C による 2 次元保存方程式の検証	54
2.52	CIP 法による 2 次元保存方程式の集中速度場における保存誤差とプロファイル半径	55
2.53	IDO 法による 2 次元保存方程式 (集中速度場) 検証の初期プロファイル	55
2.54	風上 3 次補間を用いた IDO 法による 2 次元保存方程式 (集中速度場) 検証の計算結果	56
2.55	風上 5 次補間を用いた IDO 法による 2 次元保存方程式 (集中速度場) 検証の計算結果	56
2.56	IDO 法による 2 次元保存方程式 (集中速度場) の保存誤差	57
2.57	回転速度場における 2 次元保存方程式のための初期プロファイル	58
2.58	CIP 法 Type-A による回転速度場における 2 次元保存方程式の計算結果	59
2.59	CIP 法 Type-B による回転速度場における 2 次元保存方程式の計算結果	59
2.60	CIP 法 Type-C による回転速度場における 2 次元保存方程式の計算結果	60
2.61	回転速度場における 2 次元保存方程式の計算結果から見積もられた保存誤差	60
2.62	IDO 法による 2 次元保存方程式 (回転速度場) 検証の初期条件	61
2.63	風上 3 次補間を用いた IDO 法による 2 次元保存方程式 (回転速度場) 検証の計算結果	61
2.64	風上 5 次補間を用いた IDO 法による 2 次元保存方程式 (回転速度場) 検証の計算結果	62
2.65	IDO 法による 2 次元保存方程式 (回転速度場) の保存誤差	62
2.66	CIP 法 Type-A の 3 次元移流方程式に対する数値拡散誤差	63
2.67	CIP 法 Type-C の 3 次元移流方程式に対する数値拡散誤差	64
2.68	IDO 法による 3 次元移流方程式の数値拡散誤差	65
3.1	1 次元衝撃波管問題の人工粘性に対する応答 (差分判定)	68
3.2	1 次元衝撃波管問題の人工粘性に対する応答 (微係数判定)	69
3.3	1 次元衝撃波管問題の人工粘性に対する数値誤差	70
3.4	1 次元衝撃波管問題の空間解像度に対する精度	72
3.5	1 次元衝撃波管問題の数値誤差	73
3.6	高解像度グリッド使用時の数値不安定性	74
3.7	IDO 法による数値粘性を付加した 1 次元衝撃波管問題の数値解	75
3.8	IDO 法による数値粘性を付加した 1 次元衝撃波管問題の数値誤差	76
3.9	スペクトル法による Burgers 方程式の計算結果	78
3.10	IDO 法による Burgers 方程式の計算結果	79
3.11	スペクトル法による平均移流速度を考慮した Burgers 方程式の計算結果	80
3.12	IDO 法による平均移流速度を考慮した Burgers 方程式の計算結果	81
3.13	KK-Scheme、UTOPIA Method による乱流 Burgers 方程式の検討	82
3.14	KK-Scheme、UTOPIA Method による乱流 Burgers 方程式の検討	83



3.15	乱流 Burgers 方程式のグリッド数に対する傾向	84
3.16	乱流 Burgers 方程式の粘性係数に対する傾向	85
3.17	KK-Scheme, UTOPIA Method による計算結果	86
3.18	KK スキームによる乱流 Burgers 方程式の計算精度	87
3.19	IDO 法による Frontogenesis 問題の数値計算結果	90
3.20	前線発達問題の保存誤差ならびに数値拡散誤差	91
3.21	レーリー・テラー不安定性の初期プロファイルモデル	93
3.22	不連続密度プロファイルにおける静的釣合	95
3.23	2次元レーリー・テラー不安定性のための初期密度プロファイル	97
3.24	三角関数を用いた擾乱によるレーリー・テラー不安定性の発達	98
3.25	指数関数を用いた擾乱によるレーリー・テラー不安定性の発達	99
3.26	不連続プロファイルを用いた場合の密度擾乱の成長率	101
3.27	誤差関数を用いたプロファイルを用いた場合の密度擾乱の成長率	101
3.28	不連続初期密度プロファイルによるレーリー・テラー不安定性の数値計算	103
3.29	数値粘性を付加した不連続初期密度プロファイルによるレーリー・テラー不安定性の数値計算	104
3.30	誤差関数による初期密度プロファイルを用いたレーリー・テラー不安定性の数値計算	106
3.31	3次元レーリー・テラー不安定性の成長率	109
3.32	3次元レーリー・テラー不安定性の成長過程	111
3.33	3次元レーリー・テラー不安定性の流体構造	112
3.34	複雑な擾乱に対する3次元レーリー・テラー不安定性の成長	113
3.35	3次元レーリー・テラー不安定性の構造	114
3.36	ラムジェットエンジンの概念図	116
3.37	GasDynamic LASER (GDL) の概念図	117
3.38	Kelvin-Helmholtz 不安定性の例	118
3.39	2次元衝撃波伝搬問題におけるカラーバー	124
3.40	2次元衝撃波伝搬問題	125
3.41	2次元衝撃波伝搬問題	126
3.42	$c_0 = 0, c_L = 0.2, 0.4, 0.6, 0.8$ を用いた2次元衝撃波伝搬問題	127
3.43	$c_0 = 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, c_L = 0.0$ を用いた2次元衝撃波伝搬問題	128
3.44	密度関数を用いた2流体混合モデルの例	129
3.45	比熱比を識別関数として用いる2流体混合モデル	130
3.46	超音速混合実験装置 SSMIX テストセクション部の概念図	131
3.47	グリッド解像度に対する超音速混合流の流体構造	134
3.48	超音速混合流問題の数値計算結果	135
3.49	ノズル接合部における速度不連続	136
3.50	実験結果から得られる超音速混合流問題の流体構造	137
3.51	ノズル出口における速度擾乱を考慮した超音速混合流	138
3.52	ノズル出口における速度擾乱を考慮した超音速混合流	139
3.53	ノズル出口における速度擾乱を考慮した超音速混合流	140
3.54	ノズル出口における速度擾乱を考慮した超音速混合流	141
4.1	IDO 法による2次元ポアソン方程式の計算誤差	145
4.2	ノイマン境界条件下での補間関数(3次エルミート補間)	147
4.3	ノイマン境界条件下での補間関数(5次エルミート補間)	147
4.4	IDO 法によるノイマン境界条件下に於ける2次元ポアソン方程式の数値誤差	148

4.5	IDO 法による 3 次元ポアソン方程式の数値計算誤差	150
4.6	IDO 法によるポアソン方程式解法の計算時間	151
4.7	マルチグリッド法の計算時間に対する効果	157
4.8	レッドブラックアルゴリズムの必要性	158
4.9	レッドブラック法	158
4.10	レッドブラック法を適用した IDO 法による 2 次元ポアソン方程式の数値計算誤差	160
4.11	3 次元 レッドブラック格子配列	161
4.12	レッドブラック法を適用した IDO 法による 3 次元ポアソン方程式の数値計算誤差	161
4.13	IDO 法による 3 次元ポアソン方程式の計算時間	162
4.14	セミ・ラグランジュ CIP 法の移流計算手法	165
4.15	セミ・ラグランジュ手法の時間後退法による数値不安定性	166
4.16	セミ・ラグランジュ手法の概念図	167
4.17	セミ・ラグランジュ CIP 法による 1 次元移流方程式の数値誤差	169
4.18	一様速度場における移流方程式の解法	172
4.19	一様速度場における移流方程式の数値誤差	173
4.20	非一様速度場における保存方程式の解法	176
4.21	非一様速度場での 1 次元保存方程式の保存誤差	177
4.22	移流原点計算精度の検証	177
4.23	Zalesak の初期条件	179
4.24	非移流項に CIP 法の計算手法を用いたセミ・ラグランジュ法による Zalesak 例題の検証	180
4.25	非移流項に Simpson A を用いたセミ・ラグランジュ法による Zalesak 例題の検証	181
4.26	非移流項に Simpson B を用いたセミ・ラグランジュ法による Zalesak 例題の検証	182
4.27	2 次元移流方程式に対する数値誤差	183
4.28	SL-CIP 法による 3 次元移流方程式の初期プロファイル	184
4.29	CIP 法による CFL = 0.5 での 3 次元移流方程式の計算結果	185
4.30	SL-CIP 法による CFL = 5 での 3 次元移流方程式の計算結果	185
4.31	SL-CIP 法による CFL = 10 での 3 次元移流方程式の計算結果	186
4.32	SL-CIP 法による CFL = 30 での 3 次元移流方程式の計算結果	186
4.33	Frontogenesis 問題の解析解	188
4.34	CFL = 0.5 でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果	189
4.35	CFL = 1.0 でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果	190
4.36	CFL = 2.0 でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果	191
4.37	CFL = 4.0 でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果	192
4.38	SL-CIP 法による Frontogenesis 問題の解析解からの数値拡散誤差	193
4.39	長い移流経路に沿った粘性項計算	194
4.40	重みづけによる非移流相積分の高速化	195
4.41	オイラー的積分法の概念図	196
4.42	区分求積法の概念図	196
4.43	台形積分の概念図	196
4.44	Re = 1000、CFL 数 2 による Driven Cavity Flow 問題の数値解	197
4.45	Re = 3200、CFL 数 2 による Driven Cavity Flow 問題の数値解	197
4.46	Re = 3200、CFL 数 3 による Driven Cavity Flow 問題の数値解	198
4.47	Re = 3200、CFL 数 5 による Driven Cavity Flow 問題の数値解	198
4.48	Re = 3200、CFL 数 7 による Driven Cavity Flow 問題の数値解	199
4.49	Re = 5000、CFL 数 2 による Driven Cavity Flow 問題の数値解	199
4.50	Re = 5000、CFL 数 3 による Driven Cavity Flow 問題の数値解	200

4.51	Re = 5000、CFL 数 8 による Driven Cavity Flow 問題の数値解	200
5.1	陰的 IDO 法による 1 次元移流方程式の解法	208
5.2	陰的 IDO 法における CFL 数に対する数値拡散誤差	208
5.3	陰的 IDO 法による 1 次元移流方程式の数値拡散誤差	209
5.4	陰的 IDO 法による Burgers 方程式の計算結果	211
5.5	陰的 IDO 法による KdV 方程式の数値解	213
5.6	陰解法による 1 次元衝撃波管問題の数値解	214
5.7	陰解法による 1 次元衝撃波管問題の数値解	215
5.8	陰的 IDO 法による 2 次元移流方程式の数値拡散誤差	216
5.9	IDO 法の 2 次元移流方程式の安定領域	217
5.10	陰解法による Zalesak 例題の数値解	217
6.1	境界条件として利用されるカットセル	219
6.2	1 次風上差分法による 1 次元移流方程式の解法	220
6.3	不連続点にカットセルを適用した 1 次元移流方程式の解法	221
6.4	1 次元移動カットセルの構築例	223
6.5	カットセルを含むオイラーグリッド上での補間関数	224
6.6	1 次元移動カットセル上での補間関数	225
6.7	カットセル通過に伴うオイラーグリッド上の数値不安定性	226
6.8	移動カットセル手法を適用した 1 次元移流方程式の数値計算結果	228
6.9	移動カットセル手法による Harten 例題の検証	229
6.10	移動カットセル手法を適用した 1 次元保存方程式の計算結果	231
6.11	2 次元移動カットセルの定義に関する制限	232
6.12	移動カットセル手法における Level Set 関数の初期化	233
6.13	2 次元移動カットセル手法における Level Set 関数の初期化	234
6.14	2 次元移動カットセル手法におけるカットセル位置の時間更新	236
6.15	セミ・ラグランジュ的手法による多次元移動カットセル上での物理量計算	237
6.16	複雑な境界面上の不連続値の補間	238
6.17	多次元移動カットセル手法における境界面上での物理量の補間手法	239
6.18	カットセルを適用した IDO 法による 2 次元移流方程式のための初期プロファイル	241
6.19	カットセルを適用した IDO 法による 2 次元移流方程式の計算結果	242
6.20	2 次元移流方程式のための特異点を持つ初期プロファイル	243
6.21	特異点を持つ初期プロファイルに対する 2 次元移流方程式の計算結果	243
6.22	移動カットセル手法による 2 次元移流方程式の計算結果	244
6.23	移動カットセル法による正方形プロファイルの移流計算結果	244
6.24	移動カットセル手法による 2 次元移流方程式の保存誤差	245
6.25	移動カットセル手法を用いた Zalesak 例題の検証	246
6.26	移動カットセル手法による Zalesak 例題の計算結果	247
6.27	特異点近傍における Level Set 関数の不連続	247
6.28	カットセルを適用した IDO 法による 2 次元保存方程式のための初期プロファイル	248
6.29	カットセルを適用した IDO 法による 2 次元保存方程式の計算結果 (原点への収束)	249
6.30	カットセルを適用した IDO 法による 2 次元保存方程式の保存誤差 (原点への収束)	249
6.31	カットセルを適用した IDO 法による 2 次元保存方程式の計算結果 (回転速度場)	251
6.32	IDO 法による 2 次元保存方程式の数値計算結果	251
6.33	移動カットセル手法を適用した計算結果のプロファイル切断面	252
6.34	移動カットセル手法による 2 次元保存方程式の保存誤差	252

A.1	3次元エルミート補間を用いた場合のグリッド間プロファイル	258
A.2	1次元風上差分法による移流方程式の計算	259
A.3	CIP法による移流方程式の計算	259
A.4	CIP Type-Aによる2次元空間の補間	262
A.5	CIP Type-Bによる2次元空間の補間	264
A.6	CIP Type-Cによる2次元空間の補間	265
A.7	3次元CIP法 Type-A型補間	267
A.8	3次元CIP法 Type-Cの補間手順	268
A.9	3次元CIP法 Type-C型補間	272
A.10	風上5次元エルミート補間	276
A.11	中心5次元エルミート補間	277
C.1	Rankine-Hugoniotの関係	282
C.2	2次元人工粘性のための加速度の検出	284
C.3	2次元人工粘性のための代表グリッド長さ	285
D.1	マルチグリッド法で用いられるグリッド構造	287
D.2	マルチグリッドのサイクル図	288
E.1	PCクラスターシステム概念図	291
E.2	MPLCOMM_WORLD	296
E.3	MPIにおけるgroupの概念	296
E.4	コミュニケーター (Communicator) の概念	297
E.5	コミュニケーター間通信	297
E.6	カルテシアン構造を持つプロセス群	301
E.7	グラフ構造を持つプロセス群	304
E.8	仮想トポロジーとしてのグラフ構造の例	305
E.9	ノンブロッキング通信	307
E.10	ブロッキング通信	307
E.11	集団通信によるブロードキャスト	316
E.12	集団通信を用いたスキッター、ギャザー	317
E.13	集団通信を用いた全対全スキッター/ギャザー	318
E.14	3次元計算領域の分割例	321
E.15	不均一な計算領域分割	322
E.16	計算負荷の不均一による低速化	323
E.17	計算負荷の均一化	323
E.18	カルテシアン構造の周期境界条件	325
E.19	SendRecv()を用いた通信	326
E.20	MPIによるカルテシアントポロジーを用いた通信アルゴリズム	327
E.21	ポアソン方程式の並列解法における計算の流れ	328

# 表 目 次

2.1	非一様速度場における 1 次元保存方程式の検証	45
2.2	IDO 法による 1 次元保存方程式の数値拡散誤差ならびに保存誤差	46
2.3	Zalesak 例題の保存誤差と数値拡散誤差	51
2.4	IDO 2 次元移流方程式	52
3.1	Burgers 方程式の位相初期値	77
3.3	20 °C、1 気圧下の気体の粘度ならびにサザランドの定数	132
3.2	実験での標準状態における主流、副流の各パラメータ	132
3.4	超音速混合問題に対する計算パラメータ	133
3.5	超音速混合流問題における各流入条件	133
4.1	マルチグリッド法を用いた中心差分法によるポアソン方程式の計算時間	153
4.2	IDO 法を用いたポアソン方程式のマルチグリッド法による解法の最適化	154
4.3	緩和係数による最適化	154
4.4	3 グリッドサイクルマルチグリッド法の最適化	155
4.5	マルチグリッド法の最適化	156
4.6	マルチグリッド法を用いた IDO 法によるポアソン方程式の計算時間	157
E.1	hosts ファイルの書式	293
E.2	MPI で利用できる変数の型と対応する C 言語での型	295
E.3	大域的なリダクション操作の種類	319

# 第1章

## 序論

人工知能 (Artificial Intelligence; AI)、GPS (Global Positioning System)、ゲーム機などに始まり、身近な物では電気ポットや体温計、扇風機にまでコンピューターが利用されるようになった。このような特化されたコンピューター利用の繁栄に加え、汎用型コンピューターであるパーソナルコンピューター (Personal Computer; PC) や大規模計算機 (Super Computer, Ultra Computer) の性能向上も目覚ましい。汎用型コンピューターの利用目的は様々であり、PC に関しては家電製品としての一面性も見せるようになって来た。また、汎用コンピューター用 OS (Operation System) として Linux に代表される Free UNIX の開発速度も日増しに高速化され、堅牢性を持つ OS による最新のハードウェア制御が可能になりつつある。こういった恩恵のもと、汎用機の利用目的の一つである数値シミュレーションも様々な方面で活躍するようになって来た。

天体運動、気象、燃焼、相変化、拡散、素粒子反応など、さまざまな物理現象が自然界では日常的に見られる。しかし、これらの現象は最新の現代科学を持ってしても解析的に解を求める、つまり予測をしたり、結果を十分に理論的に説明することは困難である。ただし、局所的もしくはマクロ的に現象をとらえた場合、偏微分方程式 (Partial Differential Equation; PDE) で記述されるモデルを用いて現象を理解できることが多い。物理現象のみならず、近年ではこの手法に習って経済、金融問題、人口論、食物連鎖などまでもが偏微分方程式で記述されるようになった。つまり、偏微分方程式を高精度で解くことができれば世の中の多くの現象を数値的に解析できるようになる。しかし、単純な偏微分方程式には解析的な解が存在するものの、全ての偏微分方程式が解析的に解けるわけではない。天体問題も2体問題であれば解析的に解くことが可能であるが、現実には無数の天体が宇宙空間には存在しており、これらの運動を解析的に扱う手法は無い。また、流体力学で良くみかける Navier-Stokes 方程式も一般解が知られていない偏微分方程式として有名である。これらの偏微分方程式が提示された数世紀前にはそれぞれの方程式に対しなんらかの近似ないしは制限を行い特別な場合における現象の記述を行うことは可能であったが、それらにより複雑な現象を予測したり解析したりすることは出来なかった。近年のコンピューター技術の目覚ましい進歩により、そういった問題に対して我々は偏微分方程式を用いた数値シミュレーションを行うことにより近似解を求めることができるようになりつつある。単純な例題については多くの研究がなされている一方で、複雑な構造を示す現象についてはまだ解明が進んでおらず、数値流体計算技術の向上が望まれている。その中でも計算手法の高精度化、安定化、高速化は恒久的な命題である。

数値計算の歴史も古く、有限差分法 (Finite Difference Method; FDM) や、有限要素法 (Finite Element Method; FEM)、粒子法 (Particle in Cell; PIC) など、多種多様な手法がこれまでに開発されて来た。そういった中、近年になって流体問題を少ない格子点数で高精度に解く手法として Cubic-Interpolated Propagation (CIP) 法が提案された [1, 2]。また、CIP 法と同様にエルミート補間を用いて様々な偏微分方程式の数値計算を行うことを目的として局所補間微分オペレータ法 (Interpolated Differential Operator Scheme; IDO Scheme) が提案され [3]、これらのスキームの有効性も多くの論文で検証されている [3-9]。CIP 法や IDO 法と、有限差分法などの計算スキームとの主な差異は、グリッド点毎に物理量とその低次

空間微係数を従属変数として定義し、それらの値を用いて局所的にエルミート補間を行うことでグリッド間プロファイルを高精度に表記することにある。有限差分法などは偏微分方程式に含まれる微係数を微分の定義式に従い、各物理量を差分することで求める。グリッド間隔  $\Delta x$  が限りなく 0 に近づく場合この仮定は有効であるが、数値計算で十分なグリッド数を用いない限り得られた微係数は不適当な値を示す。その結果として、解析解が既知である移流方程式などに対して適用した例から分かるように、偏微分方程式の離散化の次数とグリッド間隔、つまり空間解像度により得られる解の精度が大きく変化する他、計算ステップが増大するに従い数値拡散によりプロファイルが減衰するなどの問題が生じる。CIP 法は移流項を含む双曲型偏微分方程式を精度良く解く手法として提案され、移流項の解法に解析的手法を採用している。CIP 法では移流計算を時間後退法 (Backward-in-time Method) により求められた移流原点における物理量を近傍のグリッド点に定義された物理量とその 1 階空間微分値を用いて形成されたエルミート補間により計算する。IDO 法では CIP 法と同様に局所的にエルミート補間を形成するが、その利用方法が異なる。偏微分方程式に含まれる微係数を有限差分法では微分の定義のようにして求める一方、IDO 法では補間関数によって解プロファイルを十分に表せているとみなし、補間関数を直接微分することにより求めて適用している。また、補間関数を対象となる偏微分方程式の性質によって使い分け、少ないメッシュ数で解像度の高いプロファイルを離散空間で表現することで精度の高い数値計算を達成している。更に、複雑境界問題に対しても補間関数を境界に適合させて形成することで [10] 有限要素法に比べ簡単に高精度な数値計算を行うことが可能である。

これまでに IDO 法を用いることで少ない格子点数で高精度計算が可能である事が示されてきたが、その基礎的な研究はまだ十分にされていない。IDO 法はオイラー的手続きによる計算手法であり、その計算アルゴリズム中で、形成された補間関数を対象計算格子点上で微分を行い、偏微分方程式中に含まれる高次空間微係数を計算する。その手続きにより得られる微係数の精度が数値計算結果に影響を及ぼすと考えられることから、数値流体計算を行う上で対象とする問題に必要なグリッド解像度を知ることが目的として、補間関数から得られる微係数の誤差を検証し、計算格子の解像度に対する高次微係数の精度について議論する。また、IDO 法はオイラー手法に基づいており、一方で CIP 法はセミ・ラグランジュ手法に基づいた計算手法であるにも関わらず定常一様速度場における 1 次元移流方程式の解法が CIP 法のものとは完全に一致する [3]。このことから CIP 法で用いられる補間関数の空間補間精度を検証することにより IDO 法による移流計算の CFL 数に対する計算精度の検証が可能であると考え、CIP 法で用いられる補間関数全てに対して補間精度誤差を検証し、IDO 法と CIP 法の関係に付いて考察を行う。また、各補間関数を利用した CIP 法の特徴についても併せて考察する。移流方程式、保存方程式を扱い、CIP 法による数値計算結果と比較を行いながら流体問題を扱うに先駆けてその基礎的な性質について考察する。

次に、非線形偏微分方程式の連立により記述される圧縮性流体問題を扱う。解析解が既知である 1 次元衝撃波管問題を扱うことで空間解像度、IDO 法における人工粘性の扱いについて考察を行う。とりわけ差分法では多くの計算格子点を必要とする複雑な流体構造を示す前線発達問題、慣性核融合のターゲット圧縮時に問題となるレーリー・テラー不安定性を扱うことで IDO 法の有効性について検討を行う。また、超音速混合流問題を IDO 法により扱い、数値計算結果から現象に対する考察を行う。

圧縮性流体問題は支配方程式を連立することにより陽的に時間更新可能であるが、非圧縮性流体問題を扱う場合、圧力波が無限小の時間で伝搬することから計算時間ステップ毎に圧力場を決定する必要があるため、楕円型偏微分方程式の一つであるポアソン方程式を解かなければならない。ポアソン方程式を緩和法により解く場合、移流計算に対してその負荷は非常に大きくなり、計算格子点数が増大するに従ってそのコストも大きくなる。1 次元ポアソン方程式に対し IDO 法を用いて求められた数値解が、同じ空間解像度で 2 次精度中心差分法を用いて得られたものよりも高精度で求まることが既に示されている [3] ことから、IDO 法を用いることにより、少ない格子点数で高精度解を示す多次元ポアソン方程式解法の開発を行う。次に、ポアソン方程式の高速解法に付いて検討を行う。緩和法が収束するまでに要する計算時間は計算メッシュ数  $n$  に対し、SOR 法 (Successive Over-Relaxation Method) を用いた場合で  $n^2$  に比例することが知られており、ICCG (Incomplete-Cholesky-preconditioned Conjugate Gradient) 法でも  $n^{1.5}$  で増加する。これに対しマルチグリッド法 (Multi-Grid Method) を用いることで理想的な場合には計算時

間はグリッド数  $n$  に比例することが知られている。IDO 法によるポアソン方程式の収束計算を、緩和計算を高速化することで知られているマルチグリッド法と組み合わせることにより計算の高速化を図り、高速かつ高精度のポアソン方程式の解法を開発する。3次元ポアソン方程式を扱う場合、現在の計算機能力では計算格子点数が2次元のものに比べ多くとれないことからマルチグリッド手法の効果が期待できない問題がある。3次元ポアソン方程式の高速化手法の一つとして考えられる並列化ならびにベクトル化を視野においてレッドブラックアルゴリズムを適用し、その収束性について議論する。また、並列化を行いそのスケラビリティについて考察する。非圧縮性流体問題を高速に解く他の手法として大きな計算時間刻幅  $\Delta t$  を用いることを考える。大きな  $\Delta t$  を用いることにより目的時間までの計算ステップ数を減少させ、ポアソン方程式を解く回数を減らすことにより計算時間の短縮が期待される。非圧縮性流体問題における流体構造の時間発展は圧縮性流体問題に比べ緩やかであることから、セミ・ラグランジュ法の導入を行う。ただしその基本概念から IDO 法に対して直接適用することが困難であるため、CIP 法に対してセミ・ラグランジュ手法を適用する。CIP 法はしばしばセミ・ラグランジュスキームとして分類されてきたが、一般的なセミ・ラグランジュスキームの特徴の一つである  $CFL > 1$  での安定計算がこれまで達成されていなかった。そこで、CIP 法を  $CFL > 1$  の条件下で安定に数値計算が行えるようなセミ・ラグランジュ CIP 法を開発する。非圧縮性流体問題の一つである2次元正方流れに適用を行い、その有効性を検討する。

IDO 法を用いて数値計算を行うと様々な偏微分方程式を一様に高精度で扱えるメリットがあるが、陽解法で用いると Courant-Friedrich-Levy (CFL) 数  $u\Delta t/\Delta x > 0.3$  の条件下で安定に解けない場合がある。計算対象問題の次元数が増大するに従い利用可能な CFL 数の最大値が小さくなる事も知られている。細かな流体構造をとらえるために局所的にメッシュ間隔を細かく切ったり、つむじ風のように局所的に移流速度が極度に大きな現象について数値計算を行うと、CFL 数の制約から、時間ステップ  $\Delta t$  が計算グリッド中の最も細かいグリッド幅によって制限されたり、計算空間中の最も速い移流速度に拘束され、 $\Delta t$  を大きくとることが出来ない。そのため、計算ステップ数が非常に多くなり、予想していない数値誤差が混入する他、計算時間自体も膨大になる傾向がある。この問題を改善する手法として IDO 法の陰解法化が挙げられる。有限差分法を陰解法化すると数値計算が CFL 数に制限されず安定性が増すことが von Neumann の安定解析から示されているが、IDO 法のように空間微係数を用いた数値計算手法に対しては従来の安定解析を行うことが困難であるため、陰解法化が一概に有効であるとは言いきれない。しかし、移流項に同様の手法を用いている CIP 法では陰解法化により安定化が可能であることが報告されていることから、IDO 法でも陰解法化による安定化が可能と考え、陰的 IDO 法の開発、検証を行う。

物体の境界面や衝撃波面、相境界面など様々な不連続面を日常的に見かける。しかし、IDO 法に限らず、オイラーグリッドを用いた数値計算を行う場合グリッド間プロファイルを有限次数の補間関数により表記する限りではグリッド上に定義された不連続面の伝搬を行う際に少なからず数値拡散が生じる。こうした不連続面の伝搬問題を高精度に記述する手法としてラグランジュ法が考えられ、多次元問題を扱う場合、グリッドの大変形に伴う計算格子の破壊が予測される。本論文では移動する接触不連続面 (Contact Discontinuity) を取り扱うことを目的として、1格子点上で多価の値が定義可能なカットセル手法を用いた新しい接触不連続面取扱手法を構築する。また、多次元問題への適用を目標に定め、不連続境界面の移動手段として Level Set 手法を導入した2次元移動カットセル手法を開発する。



## 第 2 章

# IDO 法の計算精度の評価

数値流体計算を行うにあたり、計算に用いるスキームの基礎的な精度を検証することは重要な問題である。数値計算スキームに多大な数値拡散誤差や保存誤差が含まれる場合、たとえ計算結果が得られたとしてもその結果に信頼性が無くなる場合や、扱っている偏微分方程式 (Partial Differential Equation; PDE) の性質に沿わない結果しか得られなくなる場合もある。

本章では局所補間微分オペレータ法 (Interpolated Differential Operator; IDO) 法に対して補間関数から得られる高次微係数の導出精度について考察する。また、IDO 法により一様定常速度場における 1 次元移流計算を行う場合その計算手続きがエルミート補間を用いた計算スキームである Cubic-Interpolated Propagation (CIP) 法と一致することが示されていることから [3] IDO 法と CIP 法の関係を明らかにし、CIP 法が用いる補間関数から得られる空間補間誤差について検証を行う。

また、流体問題で良く用いられる単純な偏微分方程式に適用し数値計算を行うことでそれぞれの計算スキームによる数値流体シミュレーションに対する基礎的な性質について議論する。本章では、流体方程式中での移流項のみを持つ移流方程式、保存則の一つである質量保存方程式に対して CIP 法による数値計算結果と比較を行いながら検証を行う。

### 2.1 IDO 法で用いられる補間関数の高次微係数精度

局所補間微分オペレータ法 (Interpolated Differential Operator Scheme; IDO Scheme) は時間前進型支配方程式を時間に関する微分オペレータとみなし数値計算上の時間前進のために用いるテーラー展開の式に含まれる時間微係数を全て空間微係数に置換し離散式を導き、数値計算を行う手法である。計算空間中でエルミート (Hermite) 補間を形成し、対象グリッド点上で補間関数を直接微分することにより置換された空間微係数を離散形に変換する。多次元問題に対しては CIP 法 Type-A、Type-B のように多次元補間関数を形成するのではなく、各次元方向に対してそれぞれ 1 次元補間関数を複数回形成し、微分することで各次元方向の高次空間微係数を見積もる。すなわち、IDO 法では時間微係数の空間微係数への置換ならびに局所的に形成されたエルミート補間関数の直接微分による空間微係数の計算が行われる。IDO 法は、支配方程式を記述する偏微分方程式中に含まれる項を移流項、非移流項に分類し、移流項から派生した高次空間微係数は風上補間と呼ばれるエルミート補間関数を微分することで計算し、非移流項から派生した高次微係数は中心補間と呼ばれるエルミート補間を用いて計算する。つまり補間関数から得られる高次空間微係数の計算誤差が数値計算の精度に直接反映されると言える。そこで、本節では IDO 法で用いられる 1 次元の補間関数である 3 次風上補間関数、5 次風上補間関数、5 次中心補間関数に対して高次微係数の計算精度の検証を行う。しかし、従来差分法で用いられる安定解析は IDO 法の性質上困難である。その理由について以下に述べる。

有限差分法 (Finite Difference Method; FDM) では数値誤差、数値安定性を理解するために打ちきり誤

差の評価を用いる。1次元移流方程式に対する数値計算での時間前進を1次精度で行う場合は、

$$f^{n+1} = f^n + \frac{\partial f}{\partial t} \Delta t \quad (2.1)$$

$$= f^n + \left( -u \frac{\partial f}{\partial x} \right) \Delta t \quad (2.2)$$

と記述されることから、打ちきり誤差は

$$f^{n+1} = f^n + \left( -u \frac{\partial f}{\partial x} \right) \Delta t + \frac{\partial^2 f}{\partial t^2} \Delta t^2 \quad (2.3)$$

と評価される。1次元移流方程式ではこの打ちきり誤差を示す項が

$$\frac{\partial^2 f}{\partial t^2} \Delta t^2 = u^2 \Delta t^2 \frac{\partial^2 f}{\partial x^2} \quad (2.4)$$

と評価されるため、実際に数値計算で解いている基礎方程式が

$$\frac{\partial f}{\partial t} = -u \frac{\partial f}{\partial x} - u^2 \Delta t \frac{\partial^2 f}{\partial x^2} \quad (2.5)$$

と等価になることが明らかである。ただし、打ちきり誤差に関しては最低次の項のみを評価した。打ちきり誤差により付加された項

$$-u^2 \Delta t \frac{\partial^2 f}{\partial x^2} \quad (2.6)$$

は、物理量に対する空間2階微係数とその係数とみなせることから、 $\Delta t > 0$ となる陽解法では負の拡散係数が陰的に含まれ、プロファイルの数値拡散を抑制し、 $\Delta t < 0$ となる陰解法ではプロファイルの数値拡散を促進すると説明できる。同様の手法の局所補間微分オペレータ (Interpolated Differential Operator; IDO) 法への適用を試みる。IDO法で1次元移流方程式に対し風上3次補間を用いた数値計算を行う場合には物理量の時間前進のために3次項まで展開したテーラー展開

$$f^{n+1} = f^n + \frac{\partial f}{\partial t} \Delta t + \frac{1}{2!} \frac{\partial^2 f}{\partial t^2} \Delta t^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial t^3} \Delta t^3 \quad (2.7)$$

を用いる。支配方程式を当てはめ、打ちきり誤差を評価すると、

$$f^{n+1} = f^n + \left( -u \frac{\partial f}{\partial x} \right) \Delta t + \frac{1}{2!} \left( u^2 \frac{\partial^2 f}{\partial x^2} \right) \Delta t^2 - \frac{1}{3!} \left( u^3 \frac{\partial^3 f}{\partial x^3} \right) \Delta t^3 + \frac{1}{4!} \frac{\partial^4 f}{\partial t^4} \Delta t^4 \quad (2.8)$$

と書き下すことができ、打ちきり項は

$$\frac{\partial^4 f}{\partial t^4} \Delta t^4 = u^4 \Delta t^4 \frac{\partial^4 f}{\partial x^4} \quad (2.9)$$

となる。このことから、IDO法で扱っている支配方程式は

$$\frac{\partial f}{\partial t} = -u \frac{\partial f}{\partial x} - u^4 \Delta t^3 \frac{\partial^4 f}{\partial x^4} \quad (2.10)$$

とあらわせる。同様にして風上5次補間を用いる場合はテーラー展開を5次まで利用することから、

$$\frac{\partial f}{\partial t} = -u \frac{\partial f}{\partial x} - u^6 \Delta t^5 \frac{\partial^6 f}{\partial x^6} \quad (2.11)$$

となることは自明である。両者ともに付加される項が偶数次であるから陽解法を行う場合には数値拡散を抑制、陰解法では数値拡散を促進する効果を持つ。ただし、IDO法は物理量に加え、その1次元空間微係数に関して同様にテーラー展開に基づいた時間前進を行っていることからその評価を行う必要がある。空間微係数の時間前進式に関しては物理量のそれと比較して微分の階数だけテーラー展開の次数が低下する。つまり、1次元問題を扱う場合には空間微係数の時間前進は次のようになる。

$$\left. \frac{\partial f}{\partial x} \right|^{n+1} = \left. \frac{\partial f}{\partial x} \right|^n + \frac{\partial}{\partial t} \frac{\partial f}{\partial x} \Delta t + \frac{1}{2} \frac{\partial^2}{\partial t^2} \frac{\partial f}{\partial x} \Delta t^2 \quad (2.12)$$

基礎方程式を代入し、打ちきり誤差を考慮すると、

$$\left. \frac{\partial f}{\partial x} \right|^{n+1} = \left. \frac{\partial f}{\partial x} \right|^n + \left( -u \frac{\partial^2 f}{\partial x^2} \right) \Delta t + \frac{1}{2} \left( u^2 \frac{\partial^3 f}{\partial x^3} \right) \Delta t^2 + \frac{1}{3!} \frac{\partial^3}{\partial t^3} \frac{\partial f}{\partial x} \Delta t^3 \quad (2.13)$$

と記述される。打ちきり項を支配方程式により展開すると

$$\frac{\partial^3}{\partial t^3} \frac{\partial f}{\partial x} \Delta t^3 = -u^3 \frac{\partial^4 f}{\partial x^4} \Delta t^3 \quad (2.14)$$

とあらわされることから、1次空間微係数に関する支配方程式

$$\frac{\partial}{\partial t} \frac{\partial f}{\partial x} = -u \frac{\partial}{\partial x} \frac{\partial f}{\partial x} \quad (2.15)$$

は数値計算中では

$$\frac{\partial}{\partial t} \frac{\partial f}{\partial x} = -u \frac{\partial}{\partial x} \frac{\partial f}{\partial x} + u^3 \Delta t^2 \frac{\partial^4 f}{\partial x^4} \quad (2.16)$$

と扱われていることが分かる。補間関数を5次精度とした場合は

$$\frac{\partial}{\partial t} \frac{\partial f}{\partial x} = -u \frac{\partial}{\partial x} \frac{\partial f}{\partial x} + u^5 \Delta t^4 \frac{\partial^6 f}{\partial x^6} \quad (2.17)$$

となることは明らかであり、両者ともに偶数次の空間微係数が付加されることから拡散に関する数値誤差を伴う。ただし、物理量の時間前進式とは異なり、移流速度に依存してその拡散係数の正負が逆転する。

しかしその一方で IDO 法は物理量ならびにその空間微係数を用いてエルミート補間を形成し、打ちきり項を示す高次空間微係数を表記するため、上記のような単純な考察を行うことができない。例えば式 (2.10) に含まれる打ちきり項が微係数で構成されていると考えた場合

$$\frac{\partial^4 f}{\partial x^4} = \frac{\partial^3}{\partial x^3} \frac{\partial f}{\partial x} \quad (2.18)$$

とあらわされ、微係数の奇数次微分により記述されると捕らえることができる。すなわち微係数の分散が打ちきり誤差として含まれることになり、数値振動の原因となる可能性がある。式 (2.16) についても同様の考察が可能であり、一方向的に誤差解析を行うことは困難である。実際には高次微係数は物理量ならびに微係数の両者により構成されるためさらに解析は困難となる。

空間微係数算出誤差に関しても同様の考察が可能であり、解析的なアプローチによる誤差解析は差分を用いた計算手法に比べ困難であることが理解できる。このような背景からエルミート補間を用いた計算手法はその誤差解析を数値的に行うことで検討して来ている [3, 11]。

### 2.1.1 補間関数の空間精度

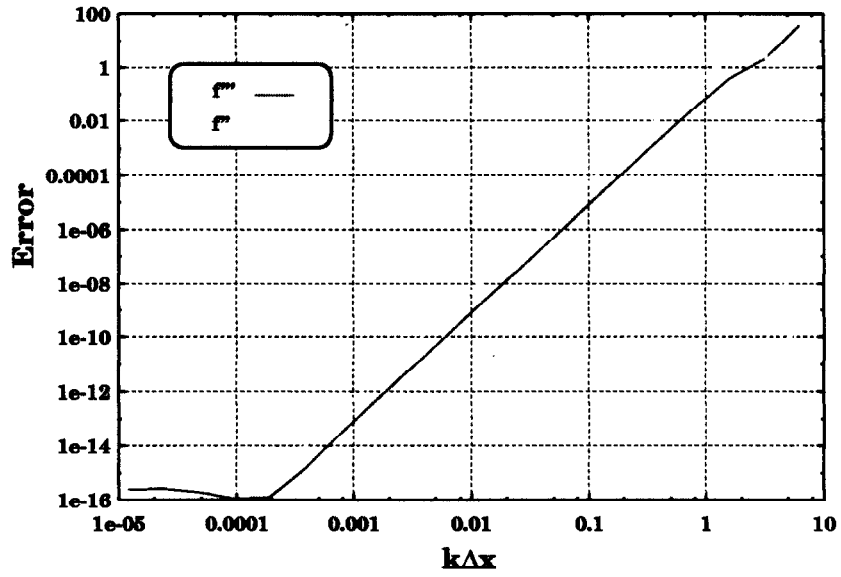
IDO 法は、多項式により記述されるエルミート補間がグリッド間プロファイルを高精度に記述できていることを前提として構築されている。このことから、グリッド間プロファイルの記述精度を検証することは重要な課題であり、特にグリッド近傍のプロファイル記述精度は空間微係数を求めるためにも必要となる。ここでは、各補間関数を用いて三角関数の一つである  $\cos$  プロファイルの記述精度を検証する。

計算空間を  $0 \leq x \leq 1, \Delta x = 1$  とし、 $\cos$  関数を用いたプロファイルを与えた。グリッド中プロファイルの波数を変化させ、空間解像度に対する補間関数記述精度を適合条件から決定される補間関数の各係数を

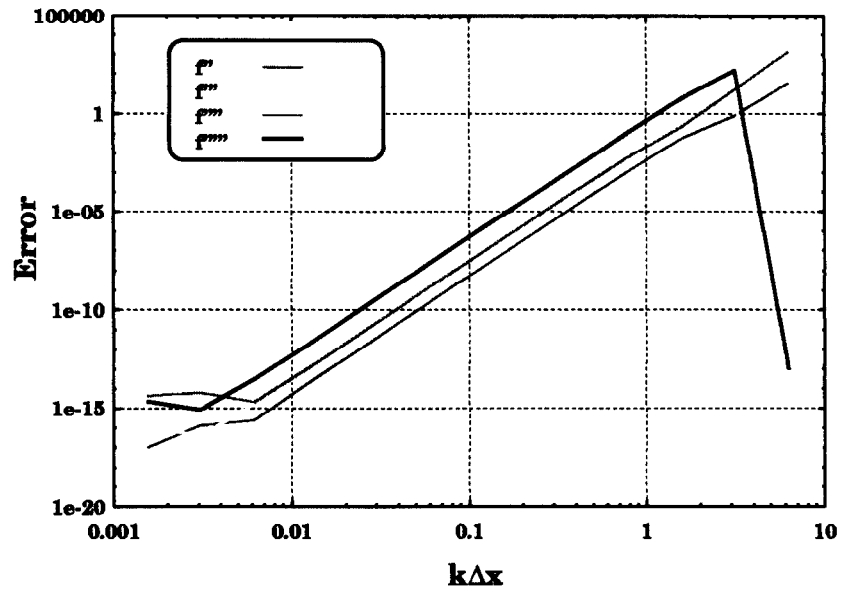
$$Error = \left| \frac{f_{analytic} - f_{calc}}{1} \right| \quad (2.19)$$

により評価する。得られた各係数は IDO 法の計算手続きからも分かるように高次微係数との関連性も強い。(a) 3次風上補間、(b) 5次風上補間、(c) 5次中心補間関数から得られた補間関数の係数導出誤差を図 2.1 に示す。

(a)



(b)



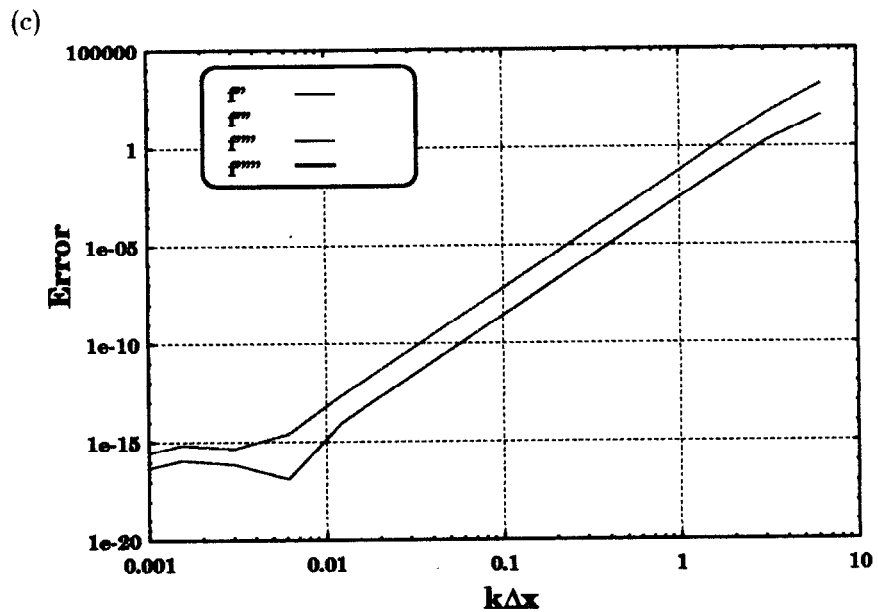
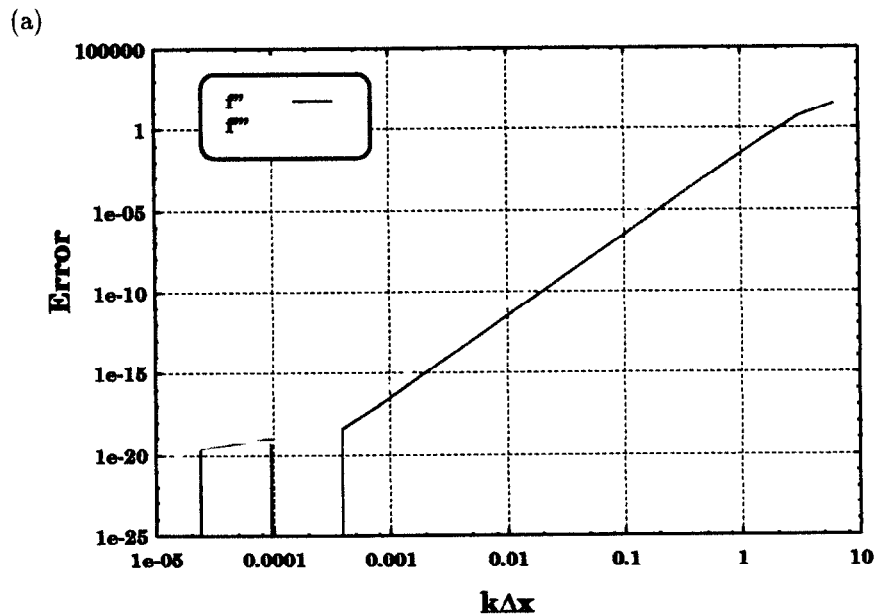


図 2.1: 補間関数記述精度:

空間解像度を増加させることで、(a) 3 次風上補間を用いた場合には  $\Delta x^4$ , (b) 5 次風上補間、(c) 5 次中心補間を用いた場合には  $\Delta x^6$  の空間精度が得られた。

空間精度を向上する事で 3 次多項式を用いた場合には  $\Delta x^4$ , 5 次多項式を用いた場合には  $\Delta x^6$  の空間精度が得られた。次に、グリッド間プロファイルとして  $\sin$  関数を与えてプロファイル記述精度を求めた結果を図 2.2 に示す。グリッド間プロファイルを  $\sin$  関数で与えた場合には、3 次補間に対して  $\Delta x^5$  精度、5 次補間に対して  $\Delta x^7$  精度が得られた。



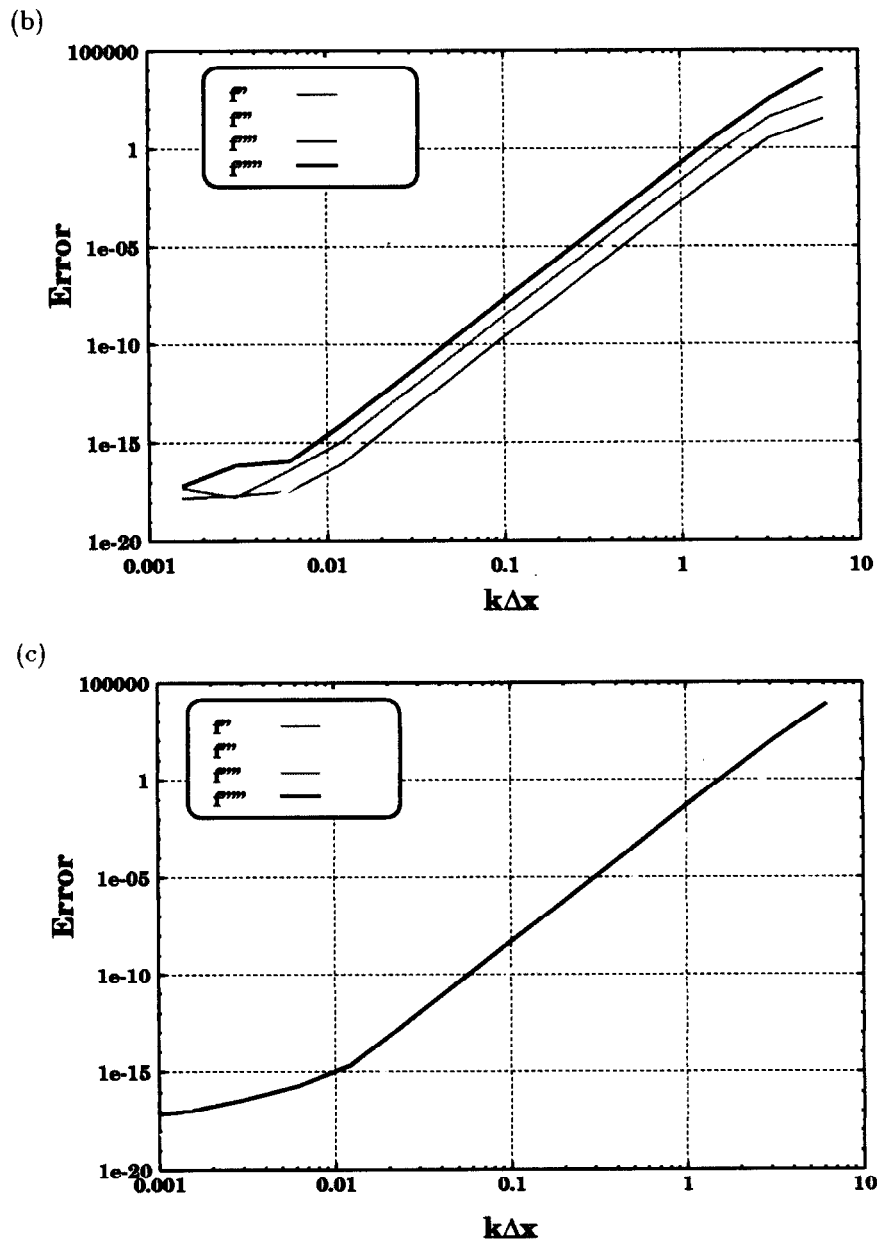


図 2.2: 補間関数記述精度:  
 空間解像度を増加させることで、(a) 3 次風上補間を用いた場合には  $\Delta x^5$ , (b) 5 次風上補間、(c) 5 次中心補間を用いた場合には  $\Delta x^7$  の空間精度が得られた。

それぞれの空間精度は  $\cos$ ,  $\sin$  関数をそれぞれテーラー展開した

$$\cos(x) = 1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \frac{1}{6!}x^6 + \dots \quad (2.20)$$

$$\sin(x) = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \frac{1}{7!}x^7 + \dots \quad (2.21)$$

を用いることで説明ができる。3 次多項式により  $\cos$ ,  $\sin$  関数を表現した場合、その表現限界によりそれぞれの表記誤差がそれぞれ  $O(\Delta x^4)$ ,  $O(\Delta x^5)$  になる。同様にして、5 次多項式によるそれぞれの表記誤差は  $O(\Delta x^6)$ ,  $O(\Delta x^7)$  となり、得られた空間誤差と一致する。

また、補間関数記述誤差は 3 次補間を用いた場合で  $k\Delta x < 0.0001$  程度で丸め誤差の影響が現れ、5 次

補間を用いた場合には  $k\Delta x < 0.01$  程度で丸め誤差の影響が現れた。これは、補間関数の空間精度が高いため、差分的手法に比べ速やかに空間解像度に対するプロファイル表記精度限界が現れることを示唆している。その一方で、少ない格子点数を用いた場合でも十分に高精度表記が可能であることを明らかにしており、エルミート補間を用いることで少ない格子点数で高精度計算が可能な理由の一つとしてとらえることができる。

IDO 法ではこれらのエルミート補間の記述精度を元に格子点上で補間関数を直接微分し、高次微係数を求める。以下では得られる高次微係数の精度について考察を行う。

### 2.1.2 3 次風上補間の高次微係数精度

IDO 法で偏微分方程式中に含まれる移流相から派生した高次空間微係数は、主に 3 次風上補間から求める。CIP 法では移流原点位置における物理量計算のためにグリッド間プロファイルの補間を目的に 3 次精度補間関数を形成する。その一方で IDO 法では移流相から派生する高次空間微係数すべてを 3 次風上補間を微分することで求めるため、微係数計算精度が重要になる。

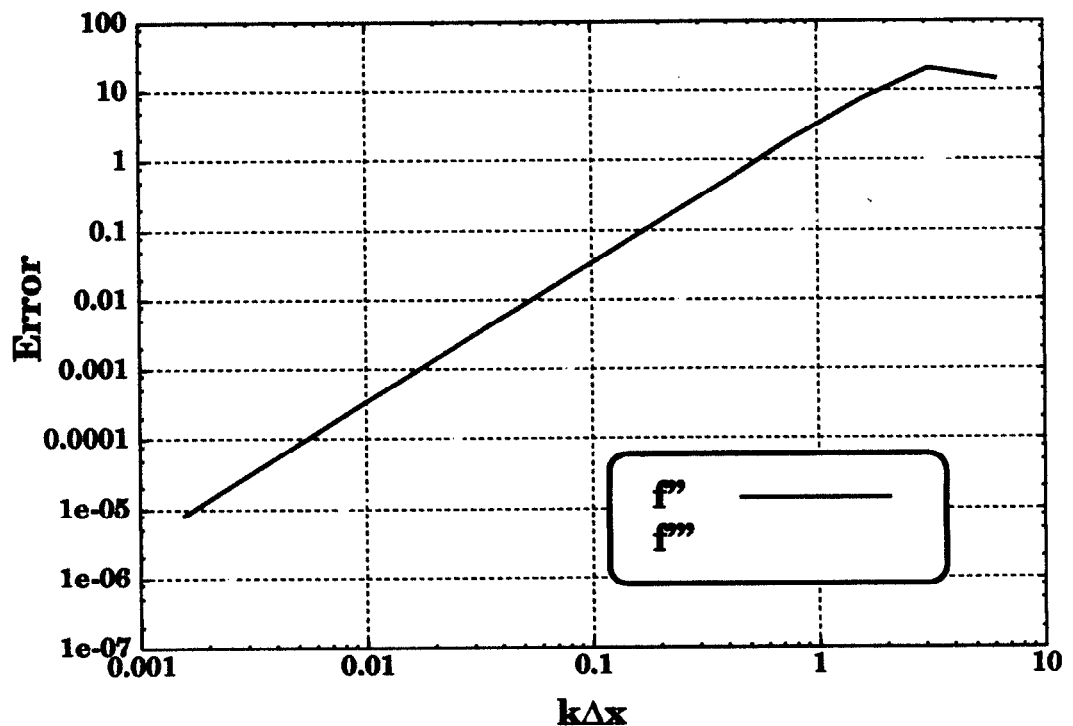


図 2.3: 3 次風上補間から得られた微係数誤差 (Sin プロファイル):

2 階空間微係数、3 階空間微係数ともに  $\Delta x^2$  程度の空間精度しか得られなかった。高波数成分 ( $k\Delta x \geq 3$ ) に関しては計算誤差が 1 を超え、誤差が 100 % 以上であることがわかる。

空間中に与えられたプロファイルが離散フーリエ級数により三角関数の線形結合により記述できる事から各波数成分に対し IDO 法で用いる補間関数により計算される高次微係数の精度を検証する事は有効である。グリッド間プロファイルを三角関数で与え、波数を変化させながら補間関数を微分して得られる高次微係数に関する検証を行う。計算空間を  $0.0001 \leq x \leq 1.0001$ ,  $\Delta x = 1$  とし、2 グリッド間にプロファイル

を sin 関数、cos 関数で与え、誤差を

$$\text{Error} = \left| \frac{f_{\text{analytic}} - f_{\text{calc}}}{f_{\text{analytic}}} \right| \quad (2.22)$$

で見積もる。ここで、 $f_{\text{analytic}}$  は解析解であり、 $f_{\text{calc}}$  は、補間関数から得られた空間微係数を表す。また、 $x=0$  では式 (2.22) から計算される  $f_{\text{analytic}}$  が 0 となる高次微係数が存在するため式 (2.22) による相対誤差の評価が困難になることから、計算空間の初期値を  $x$  を 0.0001 とした。

sin 関数で与えたグリッド間プロファイルに対する高次微係数誤差を図 2.3 に示す。2 階微係数、3 階微係数の両者ともに高波数を含むプロファイルに対しては誤差が 100% を超えており、空間精度も  $\Delta x^2$  程度でしか求まらないことがわかる。また、全波数空間に対して 3 階微係数が 2 階微係数に比べ精度が高い結果を得た。ただし、初期位置を 0.5 にした場合は 2 階微係数、3 階微係数は同程度の値を示したことから、今回用いた条件下では 2 階微係数が 0 に近くなるため丸め誤差などの影響から精度が低下したものと考える。

次にグリッド間プロファイルを cos 関数で与えた微係数誤差を図 2.4 に示す。

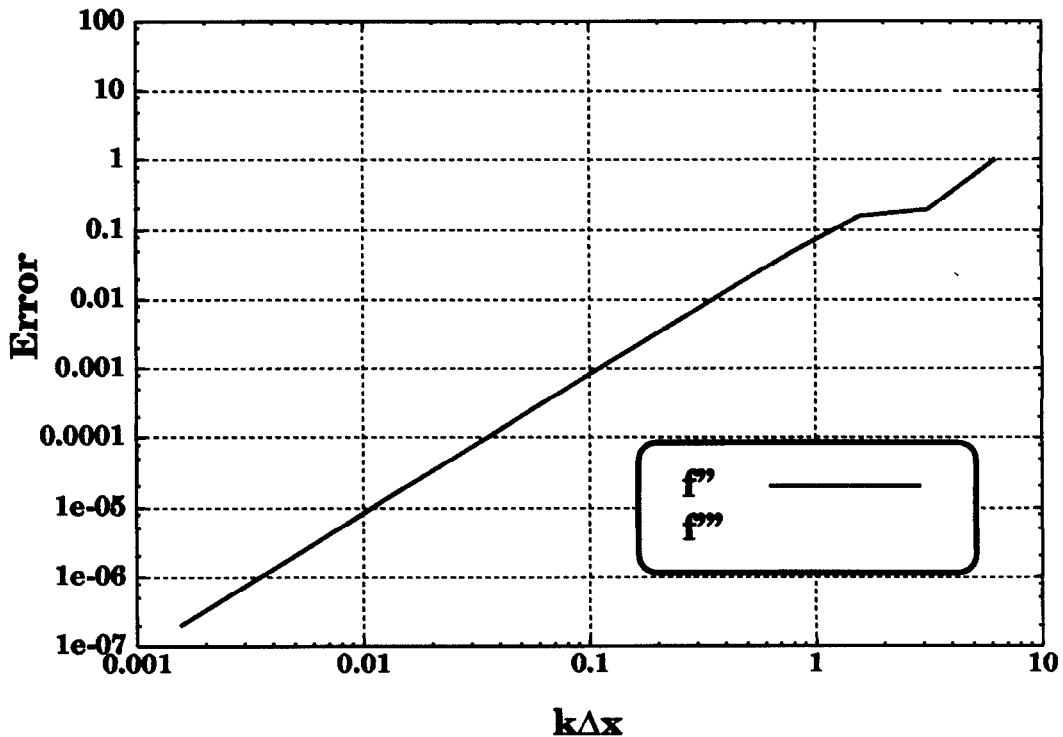


図 2.4: 3 次風上補間から得られた微係数誤差 (Cos プロファイル):

2 階空間微係数に関しては  $\Delta x^2$  程度の空間精度が得られたものの、3 階空間微係数に対しては  $\Delta x^0$  の空間精度しか得られなかった。3 階空間微係数に関しては計算誤差が 1 を超え、誤差が 100% 以上であることがわかる。

初期プロファイルを cos 関数で与えた場合は、3 階空間微係数計算誤差があらゆるプロファイル波数に対して 100% を超える結果となった。2 階微係数に関しては空間解像度に対して  $\Delta x^2$  の精度を示した。

sin 関数により表現されるプロファイル、cos 関数により表現されるプロファイルに対して 3 次風上補間が示す空間精度について考察を行う。sin 関数でプロファイルを与えた場合、その 2 階空間微係数、3 階



空間微係数はそれぞれ  $\sin$  関数、 $\cos$  関数で得られる。それぞれをテーラー展開により記述すると、

$$\sin(x) = x - \frac{1}{6}x^3 + \frac{1}{120}x^5 \dots \quad (2.23)$$

$$\cos(x) = 1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 \dots \quad (2.24)$$

で表される。式 (A.1) で表される補間関数を直接微分することにより、2 階空間微係数、3 階空間微係数はそれぞれ

$$\left. \frac{\partial^2 f}{\partial x^2} \right| = 6ax + 2b, \quad (2.25)$$

$$\left. \frac{\partial^3 f}{\partial x^3} \right| = 6a, \quad (2.26)$$

で表される。本節では式 (2.22) により、微係数の誤差を見積もるため、分母が 0 になる点を除くために初期値を  $x = 0.0001$  としていることから、初期プロファイルは主として初期値を与えるために用いた三角関数により表記されているものの、 $\sin$  関数、 $\cos$  関数の重ね合わせとして表記される。以上を踏まえて、改めて式 (2.23) - 式 (2.24) と式 (2.25) - 式 (2.26) を比較する。

$\sin$  関数を用いてグリッド間プロファイルを表記した場合、2 階空間微係数に関しては、式 (2.23) で示される 1 次、3 次の項を含む式を式 (2.25) で表される 1 次の式で表現することから、 $O(\Delta x^3)$  ないしは  $O(\Delta x^2)$  の精度を示すと考えられる。しかし、微係数計算誤差から  $O(\Delta x^2)$  の精度が得られていることから、プロファイル中に若干含まれる  $\cos$  関数により表記される波が影響し、精度を  $O(\Delta x^2)$  に低下させているものと考えられる。また、 $x = 0$  では、 $\sin(0) = 0$  となることから、丸め誤差等の影響も考慮される。3 階空間微係数は  $O(\Delta x^2)$  ないしは  $O(\Delta x)$  の精度が予想されるが、得られた結果が示した空間精度は  $O(\Delta x^2)$  であった。

同様に、初期プロファイルが  $\cos$  関数を用いて与えられた場合も同様の説明をつけられる。 $\cos$  関数で初期プロファイルを与えた場合、3 階空間微係数の計算精度が著しく低下していることを誤差解析から示した。式 (2.24) に示すように  $\cos$  関数で与えたプロファイル中では、3 階微係数は理論的には 0 になることから式 (2.26) で与えられた補間式で十分表現可能であり、 $O(\Delta x^2)$  を示す可能性がある。しかし、与えたプロファイル中には若干  $\sin$  関数で表記される波が存在するため、 $O(\Delta x)$  の精度まで低減することが考えられる。さらに、理論解が 0 に近くなることから、丸め誤差の影響もあり、3 次空間微係数の計算精度が  $O(\Delta x^0)$  まで低下したと考える。2 階微係数に関しては、 $\cos$  関数を式 (2.25) で示すことから、 $O(\Delta x^2)$  の精度を示すことが容易に理解できる。

### 2.1.3 5 次風上補間の高次微係数精度

今後、乱流問題等を扱うことを念頭においた場合、計算精度の高精度化が要求されることから、グリッド数の増加ならびに補間関数の精度向上が必要となる。グリッド数を増加させることは単純なアプローチである一方、ハードウェアに搭載された物理的な記憶容量は有限であるため、その制限から解像度向上により必要精度を達成することが出来ない場合がある。そこで、有限差分法では風上 7 次補間関数、中心 5 次補間関数などに代表される高次精度補間関数を形成することにより、補間精度を向上させる手法を主に用いている。IDO 法では 3 次風上補間関数よりも高精度な空間微係数計算アルゴリズムとして、5 次風上補間が提案されている。そこで、5 次風上補間に関して補間関数から得られる高次空間微係数の基礎的な精度検証を行う。

計算空間  $0.0001 \leq x \leq 1.0001$  において、 $\Delta x = 1$  とし、グリッド間プロファイルを  $\cos$  関数、 $\sin$  関数により与えた場合の高次微係数計算精度を計算空間中に与えるプロファイルの波数を  $2\pi$  から変化させることにより空間解像度に対して見積もる。ただし、計算誤差を式 (2.22) により見積もるため、分母が 0 にならないように、計算空間を  $x = 0$  からわずかにずらしてあるため、グリッド間プロファイルは、正確に  $\sin$ 、 $\cos$  関数で表記できるものではない。

図 2.5 に、グリッド間プロファイルを  $\sin$  関数で与え、2, 3, 4, 5 階空間微係数に対する誤差を見積もったものを示す。2 階空間微係数、3 階空間微係数に関しては  $O(\Delta x^4)$ 、4 階空間微係数、5 階空間微係数に関しては  $O(\Delta x^2)$  の精度が確認された。ただし空間解像度を上げていくと、図 2.5 からわかるように、高次空間微係数誤差が増大する結果を得た。この原因としてプロファイル波長が長くなるに従って高次空間微係数が 0 に近付くため、微係数計算時の丸め誤差による影響が生じたことが挙げられる。

次に、グリッド間プロファイルを  $\cos$  関数で与えた場合の高次空間微係数計算誤差について図 2.6 に示す。2 階空間微係数に関しては  $O(\Delta x^4)$ 、3 階、4 階の空間微係数に対しては  $O(\Delta x^2)$ 、5 階空間微係数に対しては  $O(\Delta x^0)$  の計算精度を得た。以下に空間解像度に関する考察を行う。

5 次風上補間を用いた場合、グリッド間プロファイルの補関数

$$F(x) = ax^5 + bx^4 + cx^3 + dx^2 + f_{x,i}x + f \quad (2.27)$$

を直接微分することで得られる各高次空間微係数に対する表記は、

$$\frac{\partial^5 f}{\partial x^5} = 120a, \quad (2.28)$$

$$\frac{\partial^4 f}{\partial x^4} = 120ax + 24b, \quad (2.29)$$

$$\frac{\partial^3 f}{\partial x^3} = 60ax^2 + 24bx + 6c, \quad (2.30)$$

$$\frac{\partial^2 f}{\partial x^2} = 20ax^3 + 12bx^2 + 6cx + 2d. \quad (2.31)$$

となる。グリッド間プロファイルを  $\sin$  関数で与えた場合、2, 3, 4, 5 階空間微係数はそれぞれ  $\sin$ 、 $\cos$ 、 $\sin$ 、 $\cos$  関数で表される。しかし、3 次風上補間での考察同様に、誤差解析において分母が 0 にならないように空間の初期値をずらしてあることから、 $\sin$  関数でグリッド間プロファイルを与えた場合のものには切片が含まれるため  $\cos$  関数の要素が含まれていると考えられる。式 (2.31) をテーラー展開された  $\sin$  関数と比較すると補関数は 3 次関数であることから、 $O(\Delta x^5)$  が望まれるが、 $\cos$  関数成分が含まれることを考慮すると  $O(\Delta x^4)$  次精度を示すことが理解できる。3 階微係数を示す式 (2.30) をテーラー展開された  $\cos$  関数と比較すると補関数は 2 次多項式であることから  $O(\Delta x^4)$  の精度が期待され、誤差計算結果からも  $O(\Delta x^4)$  精度を確認している。4 次微係数に関しては式 (2.29) が 1 次式で表されており、これにより  $\sin$  関数を表す場合  $O(\Delta x^3)$  精度が期待される。しかし、計算結果からは  $O(\Delta x^2)$  精度が得られており、 $\cos$  関数成分の影響が大きいと言える。5 次微係数を表す式 (2.28) は 0 次関数で表されていることから、テーラー展開された  $\cos$  関数と比較して  $O(\Delta x^2)$  精度になること予想され、誤差解析からも  $O(\Delta x^2)$  精度が得られた。

グリッド間プロファイルを  $\cos$  関数で与えた場合、2, 3, 4, 5 階空間微係数はそれぞれ  $\cos$ 、 $\sin$ 、 $\cos$ 、 $\sin$  関数で得られる。先と同様に、誤差解析において分母が 0 にならないように空間の初期値をずらしてあることから、 $\cos$  関数でグリッド間プロファイルを与えているが、 $\sin$  関数の要素が含まれていると考えることができる。2 階空間微係数を表す式 (2.31) が 3 次多項式で示されることからテーラー展開された  $\cos$  関数を表す式 (2.24) を表現する際には  $O(\Delta x^4)$  の精度を持つことが予想され、数値検証からも  $O(\Delta x^4)$  が示されている。3 階空間微係数に関しては、2 次多項式で表される式 (2.30) によりテーラー展開された  $\sin$  関数式 (2.23) を表す場合、 $O(\Delta x^3)$  の空間精度が期待されるが、与えられたプロファイル中に若干  $\cos$  関数により表記される成分が含まれていることから、検証結果からは  $O(\Delta x^2)$  精度が確認された。4 次微係数、5 次微係数を表す式 (2.29)、式 (2.28) についても同様に、それぞれが 1 次多項式、0 次多項式で表され、それらによってテーラー展開された  $\cos$  関数、 $\sin$  関数を表記する場合、 $O(\Delta x^2)$ 、 $O(\Delta x^1)$  が期待されるが、4 次微係数に関しては予想どおり  $O(\Delta x^2)$  精度が得られた。

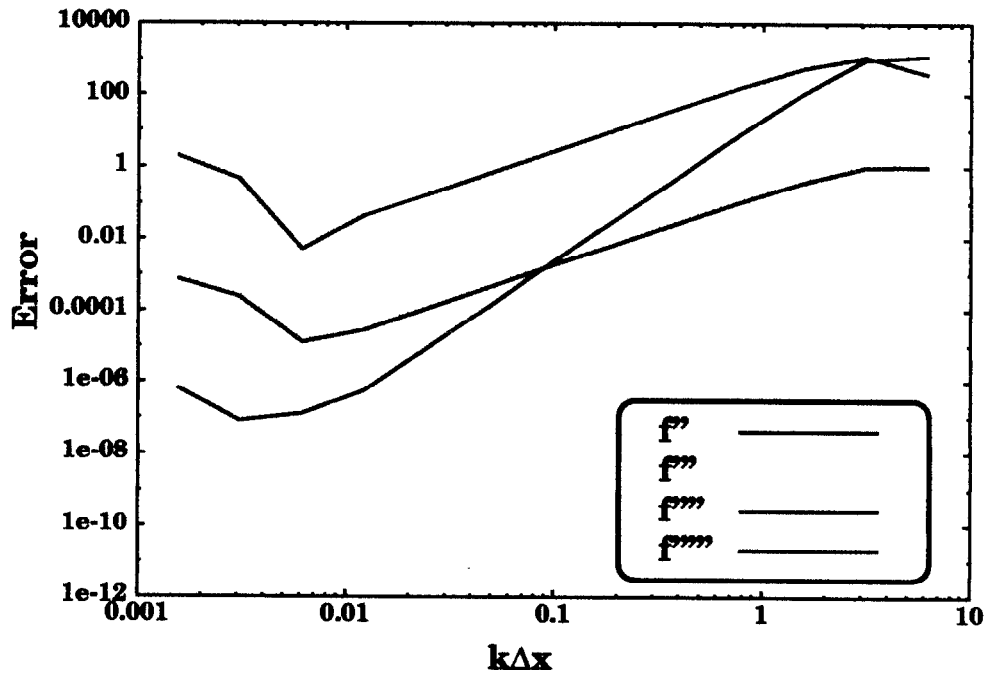


図 2.5: 5 次風上補間から得られた微係数誤差 (Sin プロファイル):  
 2 階空間微係数、3 階空間微係数ともに  $\Delta x^4$  程度の空間精度が得られたが、4 階空間微係数、5 階空間微係数に関しては  $O(\Delta x^2)$  しか得られなかった。また、空間解像度を上げるに従い、補間関数から得られた高次空間微係数と理論解の大小関係が逆転することから、誤差が増大する傾向が見られた。

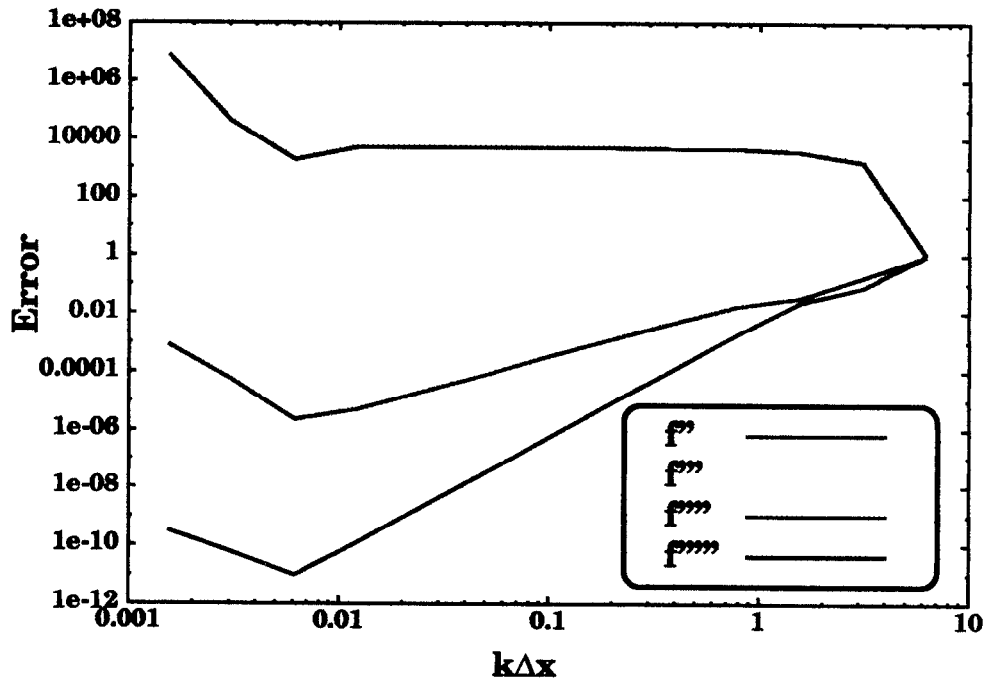


図 2.6: 5 次風上補間から得られた微係数誤差 (Cos プロファイル):  
 2 階空間微係数に関しては  $O(\Delta x^4)$  の解像度、3 階空間微係数、4 階微係数は  $\Delta x^2$  程度の空間精度、5 階空間微係数に関しては  $O(\Delta x^0)$  の解像度しか得られなかった。また、空間解像度を上げるに従い、補間関数から得られた高次空間微係数と理論解の大小関係が逆転することから、誤差が増大する傾向が見られた。

5 次微係数に関しては、グリッド間プロファイル中に含まれる  $\cos$  関数成分の影響により  $O(\Delta x^0)$  精度が現れているものと考えられる。

以上から、グリッド間プロファイルをテーラー展開した際に  $\cos$  関数で表現される成分が含まれる場合、 $\sin$  関数で表現される成分に比べて空間微係数計算精度に大きく影響することが理解できる。また、2 階、3 階空間微係数に関して 3 次風上補間との比較を行うと、 $\cos$  関数、 $\sin$  関数で表される波数成分に対して  $O(\Delta x^2)$  の精度を改善できることから、より高精度な高次空間微係数計算が可能であることを示した。

#### 2.1.4 5 次中心補間の高次微係数精度

IDO 法では非移流相から派生する高次空間微係数を計算する場合、5 次多項式

$$F(x) = ax^5 + bx^4 + cx^3 + dx^2 + \frac{\partial f}{\partial x}x + f_i \quad (2.32)$$

を用いた中心補間と呼ばれる補間関数を直接微分することで見積もる。5 次中心補間は計算対象グリッド点を中心に両側のグリッド点上の物理量とその空間微係数を境界条件として補間関数を形成する。流体問題を扱う場合は非移流相の影響により流体間ならびに流体内での相互作用が生じることから、非移流相計算精度は全体の計算精度に対して重要な影響を及ぼすことが考えられる。そこで、中心補間に対してもグリッド内のプロファイル波長に対する誤差解析を行う。

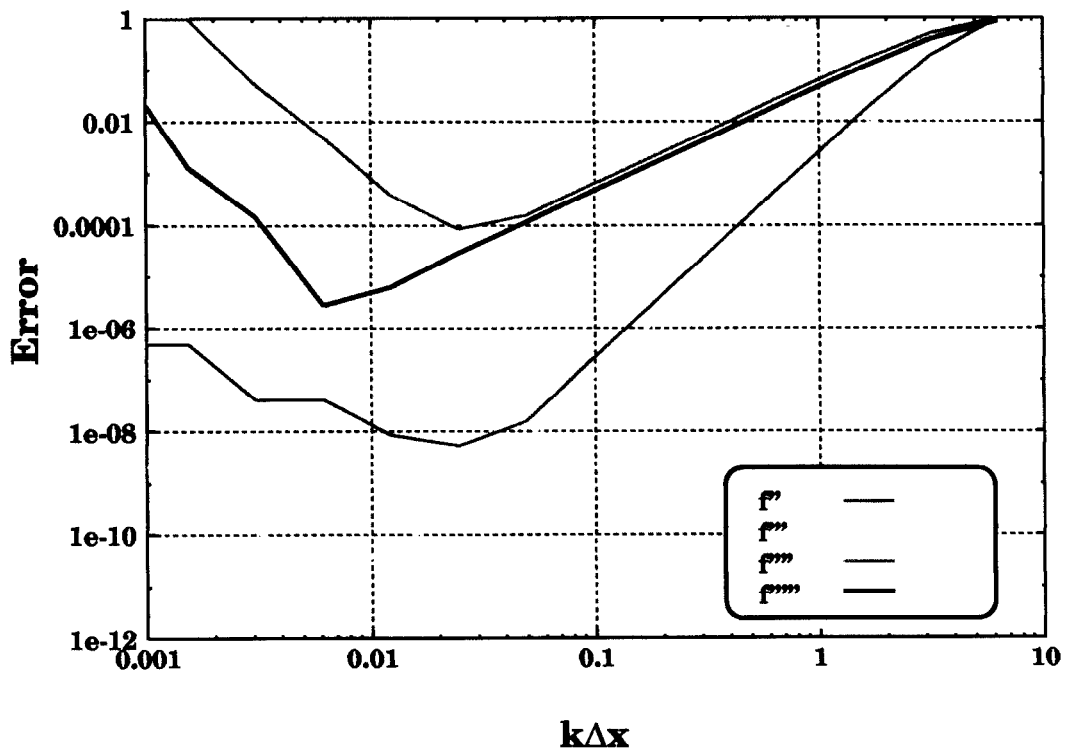


図 2.7: 5 次中心補間から得られた微係数誤差 (Sin プロファイル):

2 階空間微係数、3 階空間微係数に関しては  $O(\Delta x^4)$  精度、4 階空間微係数、5 階空間微係数に関しては  $O(\Delta x^2)$  の精度で見積もられることが示された。

計算対象空間  $0.0001 \leq x \leq 1.0001$ ,  $\Delta x = 1$  においてグリッド間プロファイルを  $\sin$  関数形、 $\cos$  関数形で与え、グリッド内のプロファイルの波数  $k$  を  $2\pi$  から変化させることで、 $k\Delta x$  に対する高次空間微係数計算誤差を式 (2.22) により見積もる。グリッド間プロファイルを  $\sin$  関数で与え、5 次中心補間により高次空間微係数を見積もった結果を図 2.7 に示す。

5 次風上補間の場合と同様に、2 階空間微係数、3 階空間微係数に関しては  $O(\Delta x^4)$  精度、4 階空間微係数、5 階空間微係数に関しては  $O(\Delta x^2)$  の精度で見積もることが出来ることが示された。ただし、波長が長くなるにつれて誤差が増大する傾向も同様に数値実験結果から明らかにした。

グリッド間プロファイルを  $\cos$  関数で与えた結果を図 2.8 に示す。2 階空間微係数、3 階空間微係数に関しては  $O(\Delta x^4)$  精度、4 階空間微係数、5 階空間微係数に関しては  $O(\Delta x^2)$  の精度で見積もられ、3 次、5 次風上補間とは異なり、初期プロファイルを  $\sin$  関数で与えた結果と同じ空間精度を得ることが出来ることを示した。風上補間は偏微分方程式中に含まれる移流相に対して適用することからその補間領域を計算グリッド上における移流速度により風上側に重点をおいて形成する。そのため、計算対象点が補間領域 (ステンシル形成領域) の片側に寄っているのに対し、中心補間は非移流相から派生する高次微係数に適用することから、特性線の数、その方向を限定することが困難であるため、計算対象点を中心として形成され、これによって風上補間の場合とは異なる境界条件を与えられる。風上補間に比べ、中心補間では補間関数中での計算対象点の位置に偏りが無いために高精度化ができ、 $\cos$  関数でグリッド間プロファイルを与えた場合でも  $\sin$  関数でグリッド間プロファイルを与えた場合と同様な結果が得られたと考えられる。しかし、 $\cos$  関数でグリッド間プロファイルを与えた場合でもグリッド間プロファイルの波長が長くなるにつれ高次空間微係数計算誤差が増大する傾向が見られた。

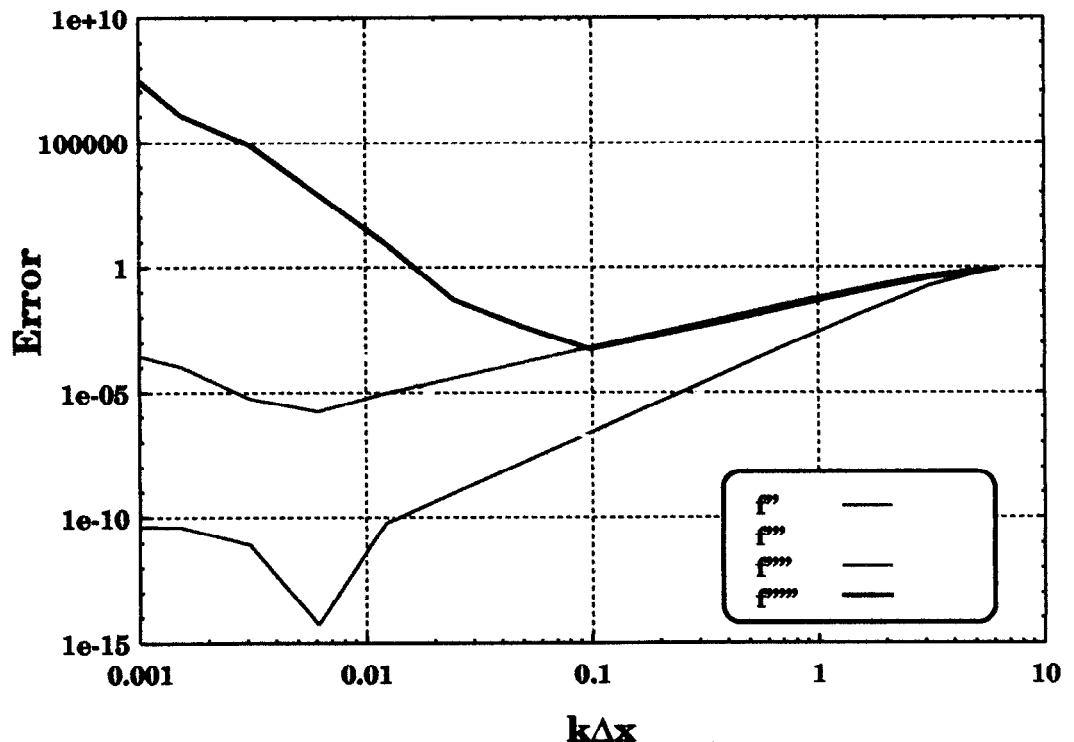


図 2.8: 5 次中心補間から得られた微係数誤差 (Cos プロファイル):

2 階空間微係数、3 階空間微係数に関しては  $O(\Delta x^4)$  精度、4 階空間微係数、5 階空間微係数に関しては  $O(\Delta x^2)$  の精度で見積もられることが示された。

波長が長くなるに従い数値誤差が増大する原因に関しては、風上補間での検証結果と同様に、高次空間微係数が 0 に近付くため丸め誤差の影響が現れたものとして説明することが出来る。また、同様に少ない格子点数で高精度計算可能である事をあわせて示した。

## 2.2 IDO 法の計算手続きと CIP 法の計算手続きとの比較

IDO 法により定常一様速度場 ( $U = \text{const.}$ ) において 1 次元移流方程式を扱う場合、その計算アルゴリズムが CIP (Cubic-Interpolated Propagation) 法と完全に一致する。IDO 法はオイラー的手法に基づき構成された計算手法である一方 CIP 法はセミ・ラグランジュ手法に基づき提案されている。その両者の計算アルゴリズムが一致することから、IDO 法はセミ・ラグランジュ的計算手法の視点から捕らえることも可能であるといえる。ただし、アルゴリズムが一致するのは先の命題のみであり、他の場合においては大きくその計算手順が異なる。本節ではエルミート補間を用いたもう一つの手法である CIP 法の補間関数に関して検証を行い、後節で議論を行う単純な偏微分方程式を用いた IDO 法と CIP 法との比較のための参考とする。

CIP 法は移流方程式

$$\frac{\partial f}{\partial t} = -(\mathbf{u} \cdot \nabla) f \quad (2.33)$$

に代表される双曲型偏微分方程式に対し、解析解を用いたセミ・ラグランジュ的手法により移流計算を行う手法である。CIP 法で移流計算を行う際、グリッド間のプロファイルをエルミート補間により空間補間し、移流原点位置におけるプロファイルの値を算出する。このことから、CIP 法により数値流体問題を扱う上でその主な特徴の一つである補間関数の空間精度に関して検証を行う。

### 1 次元補間手法の空間精度

1 次元移流方程式を CIP 法により数値計算する場合、3 次多項式

$$F(x) = ax^3 + bx^2 + cx + d, \quad (2.34)$$

により記述されるエルミート補間を用いてグリッド間プロファイルを空間補間し、移流原点位置の値を算出し ( $= F(-u\Delta t)$ )、セミ・ラグランジュ手法によりその値を次時刻のグリッド点上の値として移流させる。

$$\begin{aligned} f_i^{n+1} &= F_i(-u_i\Delta t) \\ &= a(-u_i\Delta t)^3 + b(-u_i\Delta t)^2 + c(-u_i\Delta t) + d. \end{aligned} \quad (2.35)$$

ここで、 $a, b, c, d$  は、補間関数を形成する計算セルの両端のグリッド点上で定義される物理量  $f$  ならびにその空間微係数 ( $\partial f/\partial x$ ) により決定される。このことから、CFL (Courant-Friedrich-Levy) 数 ( $= u\Delta t/\Delta x$ ) により、その補間される位置はグリッド中の全ての場所をとる可能性がある。論文 [3] でも示されているように IDO 法による定常一様速度場における 1 次元移流方程式の解法は式 (2.35) に完全に一致することから、CIP 法の 1 次元補間関数による補間誤差を算出することは、IDO 法による 1 次元移流計算を行う際の CFL 数に対する精度を検証することと同義となる。

計算空間中に不連続面を含むプロファイルが与えられた場合、フーリエ級数展開からも分かるようにさまざまな波数成分を含むことになる。例として、図 2.9 に示すのこぎり波を考えた場合、そのフーリエ級数は

$$f(t) = \frac{2}{\pi} \sum_{i=1}^{\infty} (-1)^{i+1} \frac{\sin(i\pi t)}{i} \quad (2.36)$$

であらわされる。

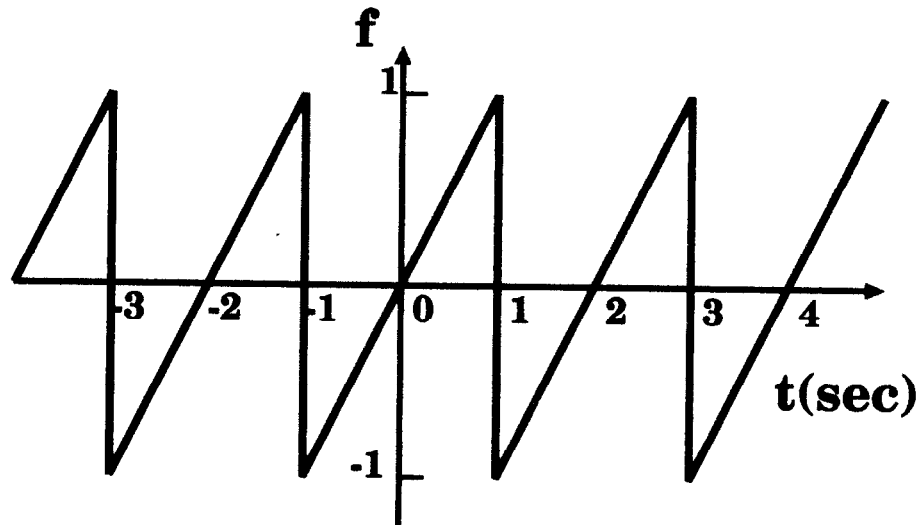


図 2.9: 不連続を含む波の例:

不連続面を持つこのぎり波をフーリエ級数により表現する場合、波数  $k = \infty$  までの三角関数の重ね合わせを行わなければならない。

このような不連続面の伝搬問題を考えた場合、各波数成分に対して数値誤差を検証することは格子点解像度に対するプロファイル表記精度を検証することと同義であることから、複雑な構造を示す計算問題を扱う上で重要となる。そこで、3次補間関数を計算領域  $0 \leq x \leq 1, \Delta x = 1/1$  において形成し、その間に含まれるプロファイルの波数  $k_x$  を  $2\pi$  から  $0.003125\pi$  まで変化させることでプロファイルの補間精度の空間分布を検証する。グリッド間プロファイルが  $\sin$  関数であらわされる場合と  $\cos$  関数で表される場合では同波長がグリッド中に含まれる場合でも、境界条件が異なることから、補間精度が異なることは容易に予測できる。そこで、以下ではグリッド間プロファイルを  $\sin$  関数で与えた場合と  $\cos$  関数で与えた場合の両者について精度検証を行う。

$k_x = 2\pi, \pi, 0.5\pi$  とし、グリッド間プロファイルを  $f = \sin(k_x x) + 2$  として与え、グリッド上の値を用いて3次多項式による補間から得られたプロファイルと解析解、ならびに

$$\text{Remapping Error} = \frac{f_{\text{analytic}} - f_{\text{interpolated}}}{f_{\text{analytic}}} \quad (2.37)$$

から見積もられた空間補間誤差を図 2.10 - 図 2.12 に示す。ここで、 $f_{\text{analytic}}$  は理論解を示し、 $f_{\text{interpolated}}$  は、補間関数から得られた値を示す。

$k_x = 2\pi$  では、3次多項式により  $\sin$  関数 1 波長を示すことから、計算グリッド中心付近 ( $x = 0.5$ ) で解析解に良く一致した結果を示すほか、計算グリッド点近傍  $x = 0, 1$  では補間関数の形成精度が高くなることから補間誤差が低減されている。前者はプロファイルの特性による特異的な精度であり、後者は補間関数の特徴から得られる精度と考えることができる。しかしその他の場所では式 (2.37) から得られる相対空間補間誤差が最大で 42% となる結果となった。

$k_x = \pi, k_x = 0.5\pi$  とグリッドに対する波長を長くするに従い空間補間による誤差を低減できている様子がわかる。 $k_x \Delta x = 0.03125\pi$  まで波数を変化させ、空間補間による誤差を空間に対して見積もった結果を図 2.13 に示す。グリッド間プロファイルの波長が長くなるに従い空間解像度 ( $\Delta x$ ) に対して  $\Delta x^{4.6}$  程度で空間補間精度が向上している様子がわかる。また、この図から読みとれる補間誤差は CFL 数による 1 次元移流問題の数値拡散の割合と考えることが出来る。例えば  $k\Delta x = \pi$  程度の波数の波がプロファイルに含まれる場合、CFL 数が 0.1 であれば 1 計算ステップで約 1% 程度の数値拡散が生じることを示している。これは、 $k\Delta x = \pi$  程度のプロファイルを CFL 数 0.1 で移流計算すると 100 ステップ程度で

$0.99^{100} = 0.37$  つまり、63% 近く減衰してしまうことを示している。その一方で波長に対する数値拡散誤差が  $\Delta x^{4.6}$  で改善されることから、ただだか4倍程度にプロファイルの波長が長くなると  $0.99999^{100} = 0.999$  となり、0.1(%) 程度のプロファイル減衰に低減できることになる。この性質から、CIP 法で不連続面を含むプロファイルの数値計算を行った場合、高波数成分は速やかに減衰するものの、数グリッド程度で不連続面を表記する空間精度をもつため、プロファイルが大きく数値拡散することはない。

また、グリッド中の誤差分布から、 $k_x = 2\pi$  等の特殊な場合を除くと、グリッド点近傍では境界条件を良く反映する補間関数を形成することから補間誤差が低減し、グリッド中央付近ではその影響が減少することから補間精度が低下することを示した。しかしその一方でグリッド中心付近の補間誤差はほぼ一様であることから、移流計算を行った場合のプロファイル減衰率が同波数に対しては CFL 数が  $0.1 \leq CFL \leq 0.9$  程度であればほとんど変わらないことを明らかにした。

次に初期プロファイルを  $f(x) = \cos(k_x x) + 2$  と与えた場合の空間補間の精度検証を行う。 $k_x = 2\pi, \pi, 0.5\pi$  としてグリッド上の物理量と1次空間微係数を境界条件としてグリッド間プロファイルを補間した結果と、式(2.37)から見積もった空間補間誤差をそれぞれ図 2.14 - 図 2.16 に示す。グリッド間に1波長が含まれる  $k_x = 2\pi$  の場合ではグリッド上に与えられた物理量とその1次空間微係数を用いた3次エルミート補間ではグリッド間プロファイルを直線で表現してしまうため、グリッド近傍以外の点では大きく異なる値を示すことがわかる。 $k\Delta x = 0.03125\pi$  まで波数を変化させ、空間補間誤差を見積もった結果を図 2.17 に示す。プロファイルを sin 関数で与えた場合は  $\Delta x^{4.6}$  の精度を示したが、cos 波形で与えた場合は  $\Delta x^{3.8}$  程度の精度しか確認できなかった(図 2.18)。そこで、その理由について以下に考察する。

sin 関数、cos 関数をそれぞれテーラー展開すると以下のように書き下すことが出来る。

$$\sin(x) = x - \frac{1}{6}x^3 + \frac{1}{120}x^5 \dots \quad (2.38)$$

$$\cos(x) = 1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 \dots \quad (2.39)$$

位相が  $\pi/2$  異なるプロファイルであるにも関わらず、両プロファイルを制限された空間内で表現する場合、境界条件が異なることから、テーラー展開を用いた多項式近似により sin 関数、cos 関数を表現すると式(2.38)、式(2.39)のように奇数次のみの多項式、偶数次のみの多項式で表現されることがわかる。3次多項式を用いた補間を行うことから理想的には sin 関数によるプロファイルに対しては  $O(\Delta x^5)$  精度、cos 関数を用いたプロファイルに対しては  $O(\Delta x^4)$  精度を示すと考えられる。しかし、今回の検証では計算対象空間を固定していることから、1波長以下のプロファイルを補間しているため、sin 関数、cos 関数を用いたプロファイルの定義を行っているにも関わらず、右端での境界条件により純粋な sin 関数、cos 関数によるプロファイルの定義が困難になっている。つまり、sin 関数、cos 関数の重ね合わせによりプロファイルが表現されていると言える。

このことから、プロファイルを  $\sin(x)$  で与えている場合は必ず左側境界に sin プロファイルを持つため補間精度は主に  $O(\Delta x^5)$  となるが、右側の境界条件により cos 成分が少なからず含まれるため若干精度が低下する。プロファイルを  $\cos(x)$  で与えている場合はプロファイルが主として cos で構成されると考えられるため  $O(\Delta x^4)$  の精度を示すと考えられるが右側の境界条件により若干精度が減少しているものと考えられる。 $k\Delta x = 0.5\pi$  では sin、cos 両プロファイルにおいて位相が異なるものの境界条件が一致することから図 2.13、図 2.17 からも理解されるように得られる空間補間誤差は完全に一致する結果が得られた。

以上から sin 関数、cos 関数を用いて定義されたプロファイルから得られた空間補間誤差が  $\Delta x^{4.6}$ 、 $\Delta x^{3.8}$  程度になることが説明できる。



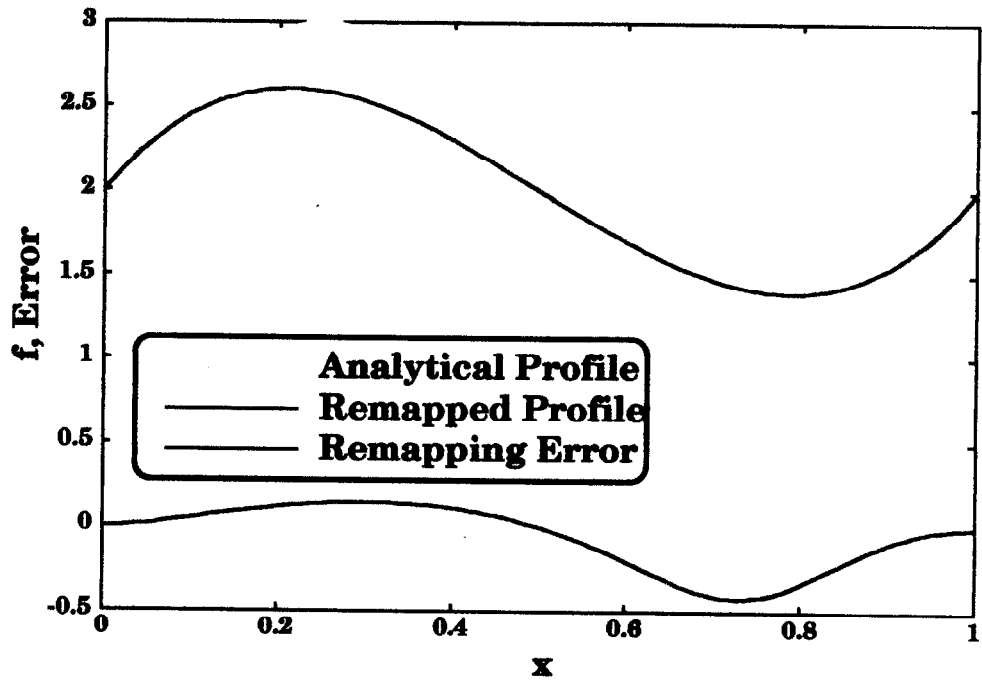


図 2.10: 1 次元 CIP 法における空間補間誤差 (波数  $k = 2\pi$ ):

$x = 0, 0.5, 1$  近傍では比較的精度の高い空間補間が行えているが、 $0.1 \leq x \leq 0.4$ 、とりわけ  $0.6 \leq x \leq 0.9$  では空間補間による誤差が最大 42% 程度まで発生することがわかる。

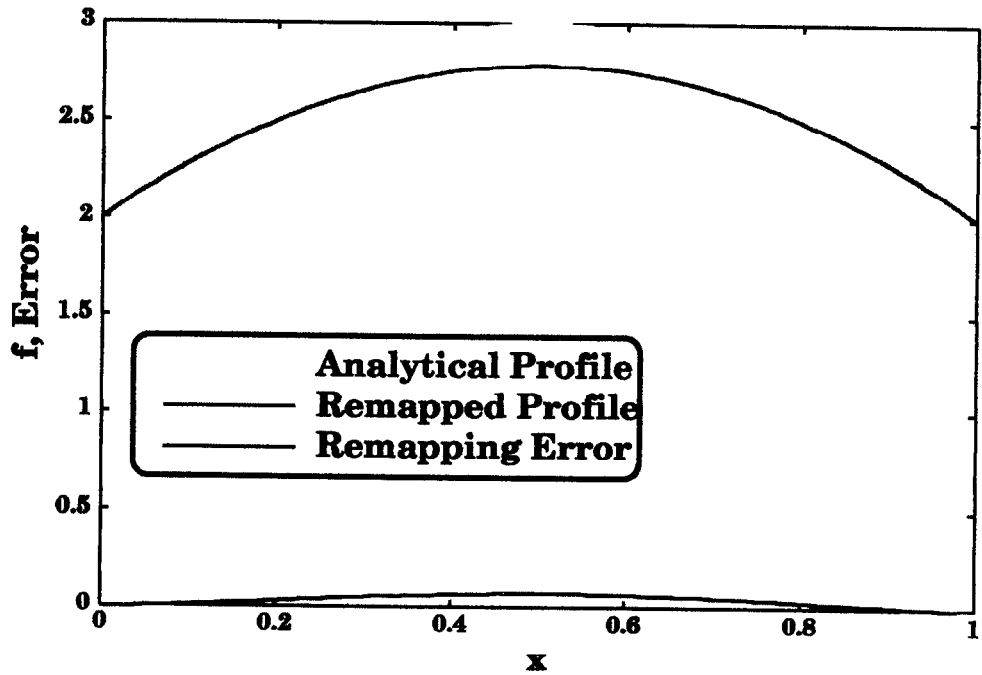


図 2.11: 1 次元 CIP 法における空間補間誤差 (波数  $k = \pi$ ):

計算グリッド近傍 ( $x = 0, 1$ ) では比較的精度の高い空間補間が行えているが、グリッド中心付近で補間精度がい低下している様子が分かる。

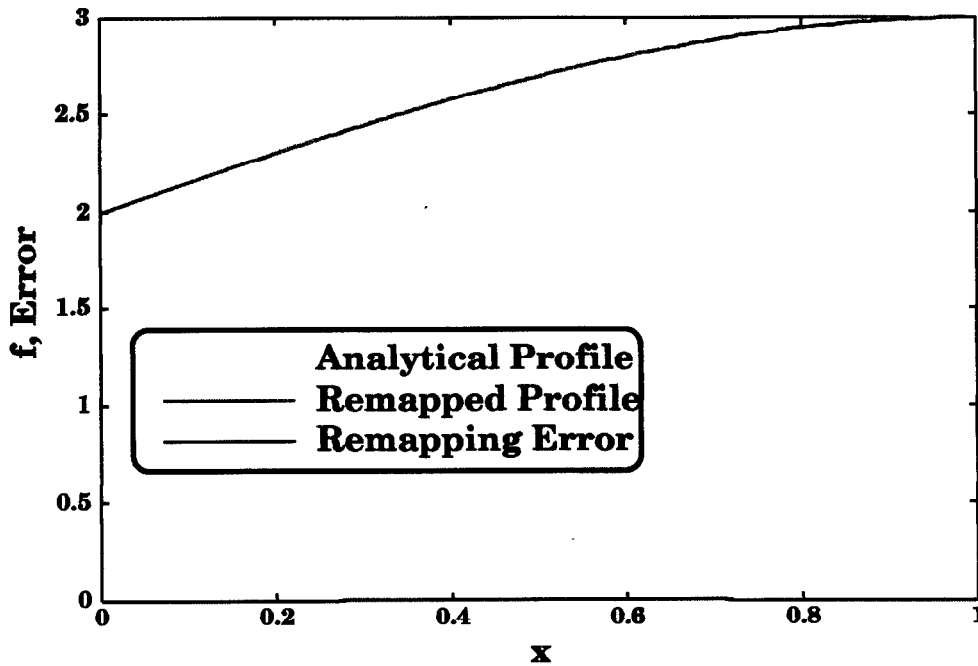


図 2.12: 1 次元 CIP 法における空間補間誤差 (波数  $k = 0.5\pi$ ):  
 $x = 0, 0.5, 1$  近傍では比較的精度の高い空間補間が行えているが、計算グリッド近傍 ( $x = 0, 1$ ) では比較的精度の高い空間補間が行えているが、グリッド中心付近で補間精度が低下している様子が分かる。

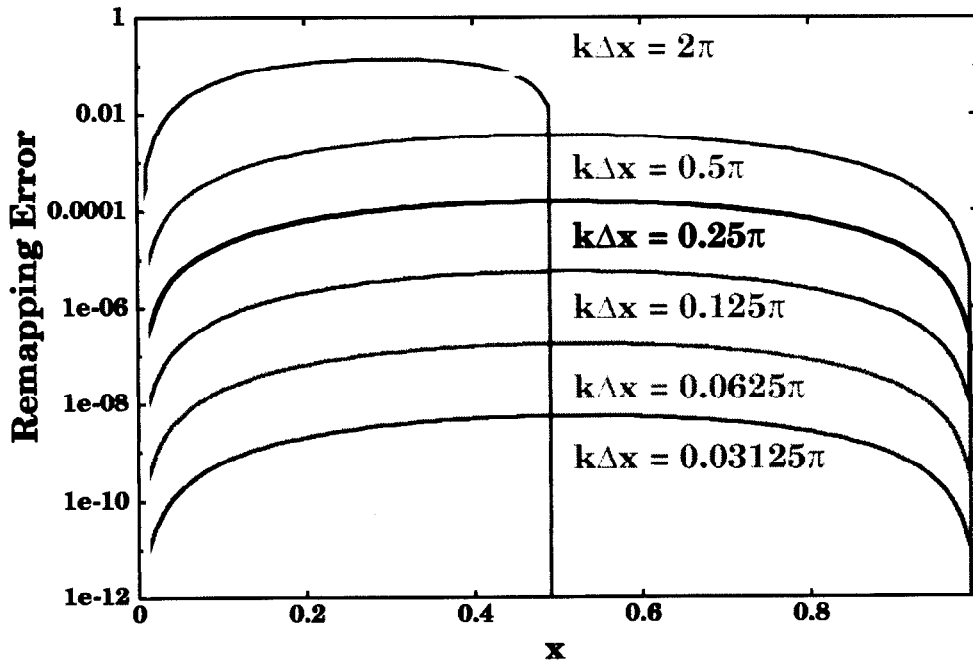


図 2.13: 1 次元 CIP 法における空間補間誤差:  
 グリッド間プロファイルの波長が長くなるに従い空間補間の精度が  $\Delta x^{4.6}$  程度で向上する。

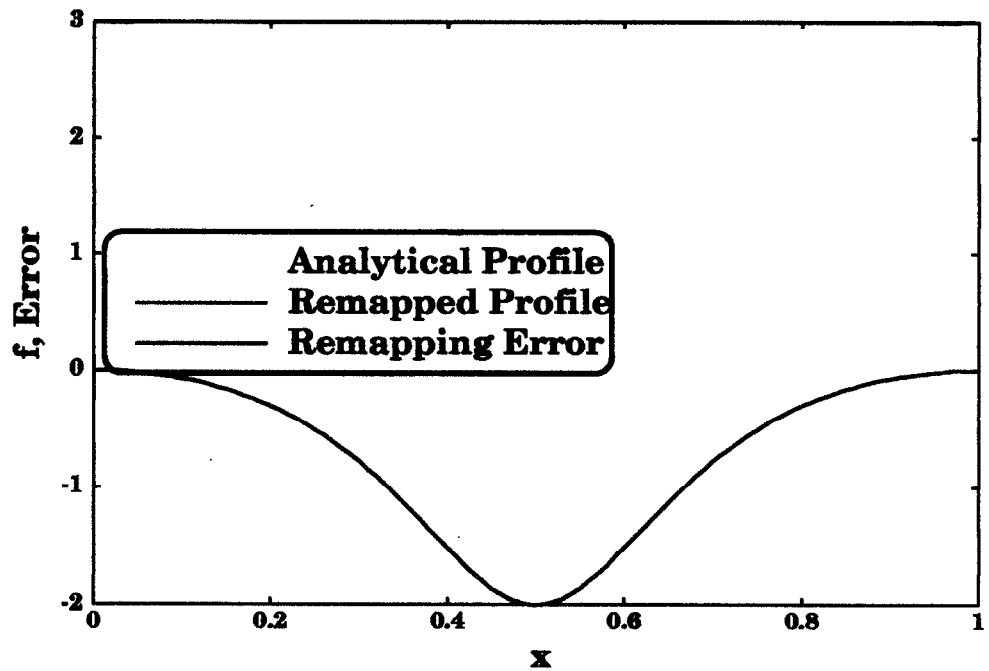


図 2.14: 1次元 CIP 法における空間補間誤差 (波数  $k = 2\pi$ ):

cos 関数でプロファイルを与える場合、 $k_x = 2\pi$  のプロファイルをグリッド点上の物理量とその微係数のみで表現することが出来ず、グリッド近傍以外では大きく異なる値を示す。

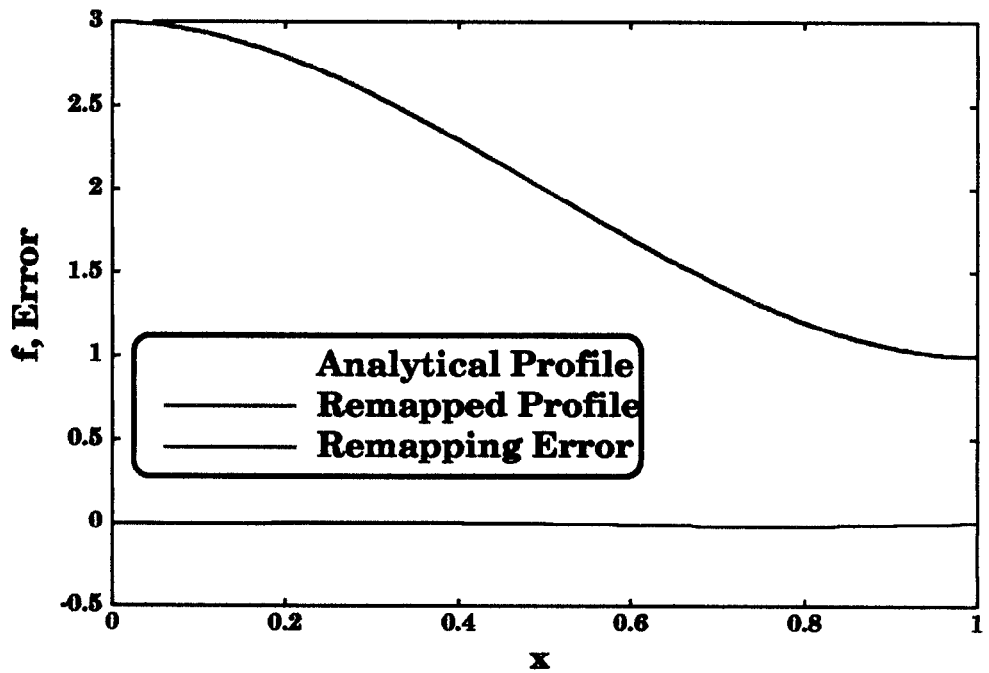


図 2.15: 1次元 CIP 法における空間補間誤差 (波数  $k = \pi$ ):

グリッド中全体で高精度補間が行えているが、セル中心付近では計算グリッド近傍に比べ若干精度の低下が見られる。

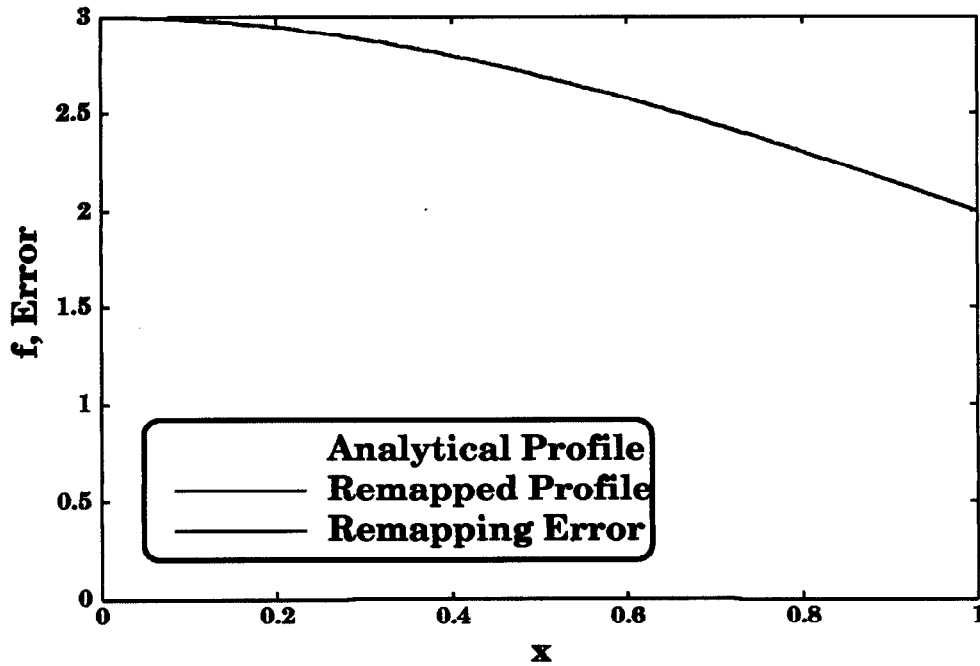


図 2.16: 1 次元 CIP 法における空間補間誤差 (波数  $k = 0.5\pi$ ):  
 グリッド中全体で高精度補間が行えているが、セル中心付近では計算グリッド近傍に比べ若干精度の低下が見られる。

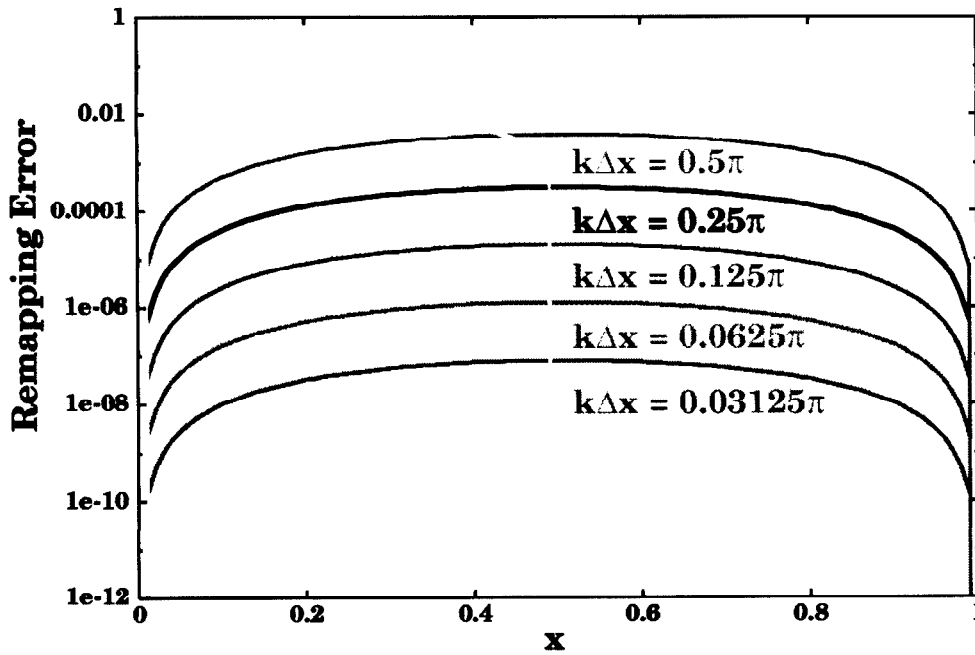


図 2.17: 1 次元 CIP 法における空間補間誤差:  
 グリッド間プロファイルの波長が長くなるに従い空間補間の精度が  $\Delta x^4$  程度で向上する。

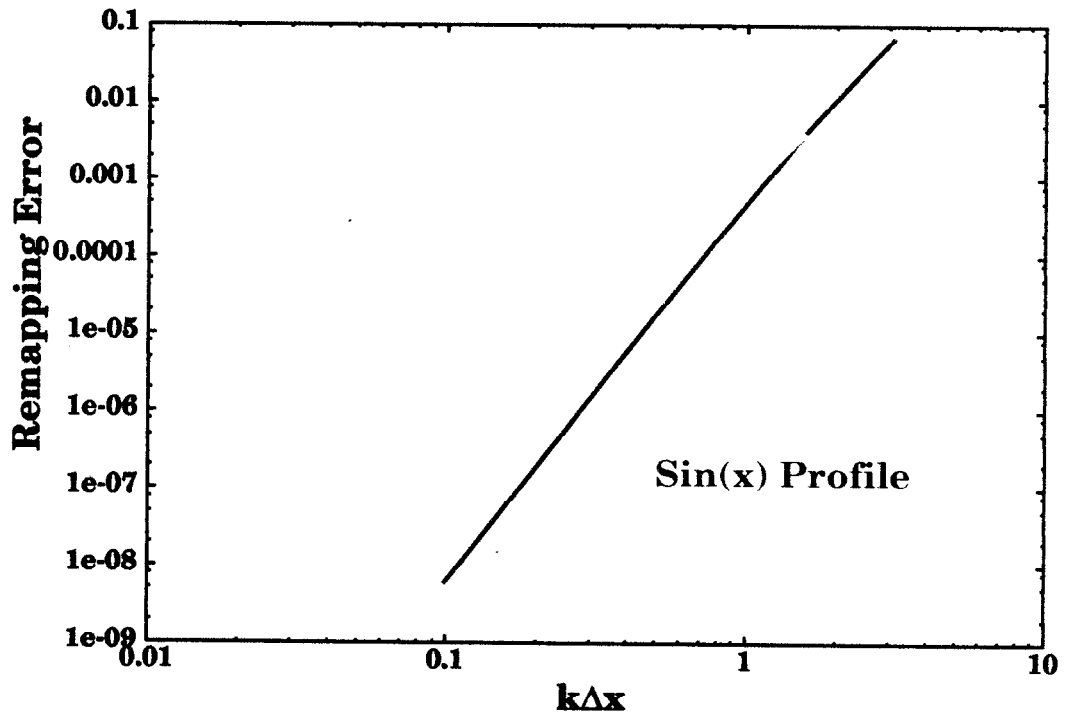


図 2.18:  $\cos, \sin$  でプロフィールを与えた時の CIP 法による空間補間誤差:

$\cos(x)$  でプロフィールをグリッド間を与えるとき空間補間による誤差は  $\Delta x^{3.8}$  程度の精度を示すが、 $\sin(x)$  でプロフィールを与えるとき空間補間誤差は  $\Delta x^{8.8}$  程度の精度となる。これは  $\sin, \cos$  をそれぞれテーラー展開すると奇数次により構成される多項式、偶数次により構成される多項式で表現できることに起因する。

#### 多次元補間手法の空間精度

IDO 法により 2 次元問題を扱う場合、A.3.3 節に示すように 1 次元 IDO 法で用いる補間関数を各計算軸方向に適用を行い、計算対象格子点上でオイラー的時間前進を行うために必要となる空間微係数の算出を行う。その一方で 2 次元 CIP 法における計算手順も 1 次元 CIP 法と同様に、計算対象点に対応する移流原点位置における物理量をエルミート補間により見積り、セミ・ラグランジュ的に移流計算を行う手法をとる。今後 IDO 法による多次元問題の精度検証を CIP 法によるものと比較検討を行いながら示すことを目的として、CIP 法の多次元補間手法から得られる空間補間精度について検証を行う。

2 次元 CIP 法で用いられる空間補間手法には Type-A, Type-B, Type-C の 3 方式が提案されている。Type-A, Type-B では、2 元関数  $F(x, y)$  を形成する手法をとり、Type-C では、1 次元 CIP 法で用いた関数を複数回使用することにより対象点の物理量を算出する手法をとる。また、それぞれの手法で補間関数を形成する際に利用する境界条件も異なる。その結果としてそれぞれの手法で扱える CFL 数や計算結果から得られる数値拡散誤差などに差が生じることが知られている。

その性質を明らかにすることを目的として、2 次元 CIP 法 Type-A, Type-B, Type-C で用いる補間関数についても空間補間誤差を検証することは重要な課題である。1 次元補間による空間補間誤差から 2 次元問題に対してもそれぞれの補間手法で用いている補間関数の次数が各軸方向に対して最大 3 次であることを考慮すると計算領域に与えるプロフィールを  $\cos$  関数を用いて与えた場合と  $\sin$  関数を用いて与えた場合の精度の傾向が容易に推測できる。このことから、ここでは補間のためのグリッド間プロフィールを  $\cos$  関数を用いた  $f(x, y) = \cos(k_x x) \cos(k_y y) + 2$  として与えるのみとする。ただし直交等間隔オイラー座

標系を用い、プロファイル中に各軸方向に含まれる波数を、 $k_x = k_y = 2\pi, \pi, \dots$  と同じ値を与えることとした。Type-A、B、C それぞれから得られた空間補間誤差を図 2.19 - 図 2.27 に示す。

全ての空間補間手法において、プロファイルを  $\cos$  関数で与えていることから 1 次元問題の場合と同様に  $k_x = k_y = 2\pi$  のプロファイルを表現することは不可能であった。Type-A では補関関数を形成するための境界条件として、 $(x, y) = (0, 0), (0, 1), (1, 0)$  における物理量と空間微係数 ( $\partial f/\partial x, \partial f/\partial y$ ) の連続条件ならびに  $(x, y) = (1, 1)$  における物理量の連続条件を用いていることから 図 2.20、図 2.21 に示すように計算対象グリッド点  $(x, y) = (0, 0)$  の近傍では比較的高精度な補間が行えるものの、計算対象点から遠ざかるにつれて補間誤差が増大する傾向を示した。ただし、 $(x, y) = (1, 1)$  近傍では物理量の連続条件を利用して、図 2.20 からわかるように急速に補間精度を回復している。Type-A の手順により 2 次元数値流体問題を扱う場合、経験的に CFL 数を 0.5 以下で扱う必要があることが知られている。今回得られた補間誤差を元に考察すると、CFL 数が 0.5 以下程度で参照される領域では補間誤差が比較的少なく、CFL 数が 0.1 以下で参照される領域に限って言えば更に高精度補間が行われていることから、Type-A を用いた高精度移流計算を行う場合は CFL 数を 0.1 以下程度とすることが望ましいと言える。

Type-B では  $(x, y) = (0, 0), (1, 0), (0, 1), (1, 1)$  における物理量に加え、空間微係数  $\partial f/\partial x, \partial f/\partial y$  の連続条件を境界条件として補関関数を形成することから Type-A のように空間補間誤差が空間内で局所的に増大することは無い結果を得た。今回検証した波数では Type-A に比べ空間補間誤差が少ない結果となった。Type-B による補関関数形成手法を採り入れた 2 次元 CIP 法では CFL 数が 1 を越えない範囲で経験的に安定な数値計算が行えることが知られており、Type-A における説明と同様にその理由は補関関数から得られる補間誤差の分布を用いて説明できる。ただし、境界条件の影響を良く反映するグリッド点近傍から遠くなるにつれて補間精度が低下するため、Type-B においても同様に高精度数値計算を行うためには CFL 数を小さくすることが望ましい。

Type-C による補関手順では、 $(x, y) = (0, 0), (1, 0), (0, 1), (1, 1)$  における物理量ならびに空間微係数  $\partial f/\partial x, \partial f/\partial y$  の連続条件に加え、 $\partial^2 f/\partial x \partial y$  を境界条件として補関関数を形成すると考えることができる。このことから、Type-C による手続きに基づきグリッド中を補関する場合、Type-A、Type-B による空間補間に比べ境界条件として与える情報量が増加することから高精度な空間補間が可能であることを計算結果から明らかにした。また、Type-B での補関手法と同様に、計算セルに含まれる全グリッド点において等価的な境界条件を与えていることから、セル内部における誤差の局所的な分布は低減できている。Type-C による補関手法を用いた CIP 法での移流計算では、Type-B と同様に CFL 数が 1 以下であれば数値安定的に計算できることが知られており、このことはセル内の均一な補関精度に言及される。

2 次元 CIP 法における補関関数として提案されている Type-A、B、C が持つそれぞれの平均補関誤差を見積もるため、

$$\text{Remapping Error} = \sum_i^N \sum_j^M \left| \frac{f_{\text{Interpolated}} - f_{\text{Analytic}}}{f_{\text{Analytic}}} \right| / (N \times M) \quad (2.40)$$

を用いて検討する。ここで、 $f_{\text{Interpolated}}$  は補関された値を示し、 $f_{\text{Analytic}}$  は解析解を示す。また、各軸方向の補関領域分割数  $N, M$  をそれぞれ 101 とした。得られた相対誤差を図 2.28 に示す。ほぼ全ての波数空間において Type-C で用いられている補関手法から見積もられた空間補関誤差は Type-A、Type-B で用いられている補関手法から見積もられたものよりも小さくなる結果が得られた。これは補関関数形成のために 2 階空間微係数  $\partial^2 f/\partial x \partial y$  を用いることで、境界条件が増加することから、補関精度が向上していることに起因すると考えられる。図 2.28 から Type-A、Type-B、Type-C で用いられている補関手法はそれぞれ  $\cos$  関数により表記されるプロファイルに対し、 $\Delta x^4$  の精度を示すことを示した。

また、グリッド中に含まれるプロファイルの波長が長い領域では Type-A、Type-B による補関から得られた空間補関誤差は相対的に 1% 未満の差が見られるもののほぼ同じ結果となった。

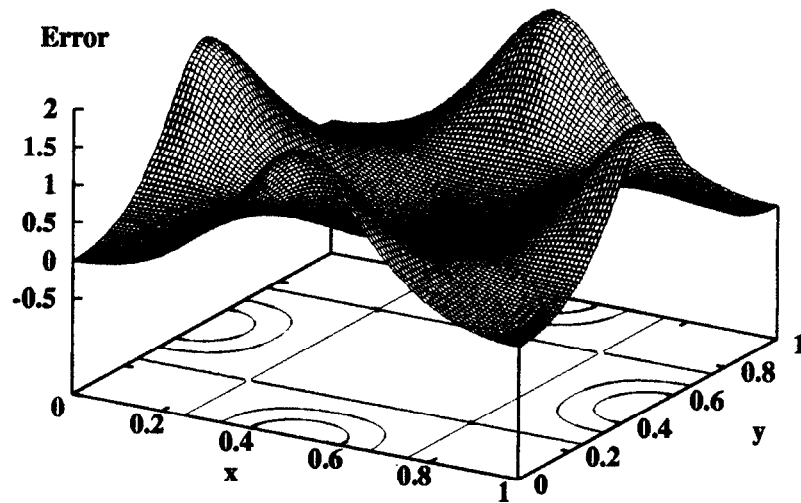


図 2.19: 2次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = 2\pi$ ):

CIP 法の 1 次元補間手法と同様に、波数を  $2\pi$  として与えた  $\cos$  プロファイルに対しては、補間関数の境界条件として与えるグリッド上の情報のみでは適切な補間関数を形成することが不可能である。

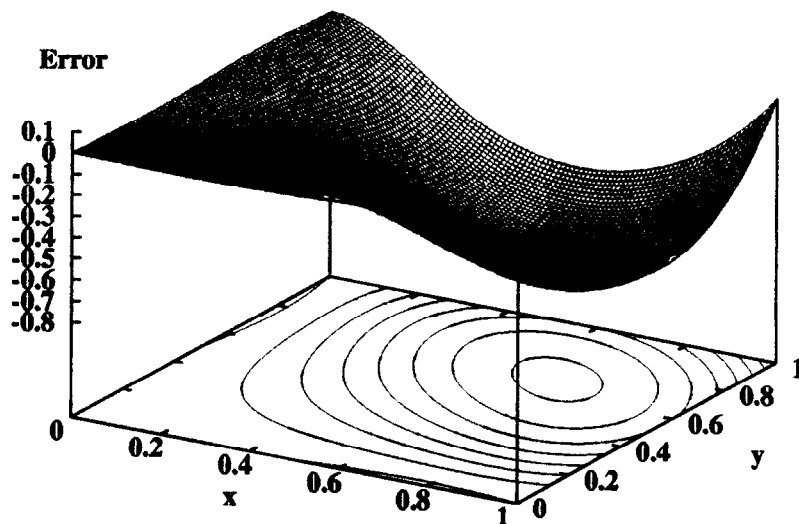


図 2.20: 2次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = \pi$ ):

補間関数を形成するために利用する境界条件が計算対象点近傍に局在しているため、計算対象点から遠ざかるにつれ補間誤差が増大する。

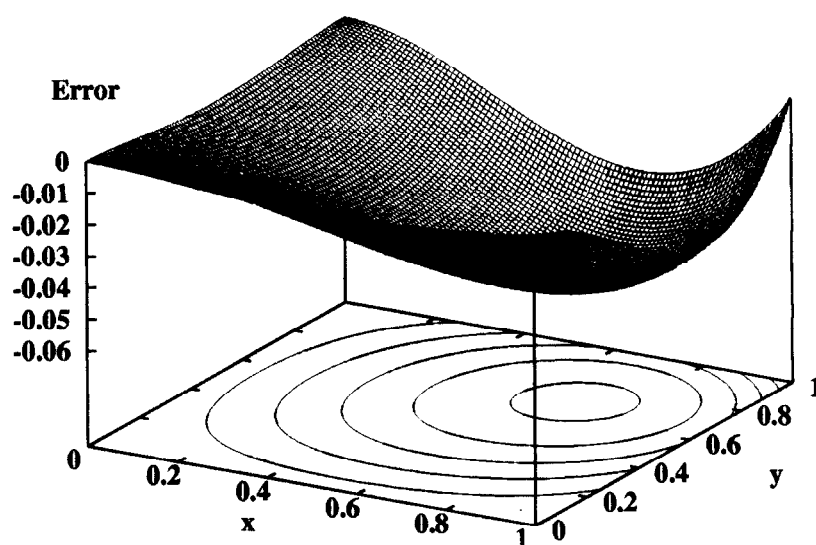


図 2.21: 2次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = 0.5\pi$ ):  
補間関数を形成するために利用する境界条件が計算対象点近傍に局在しているため、計算対象点から遠ざかるにつれ補間誤差が増大する。

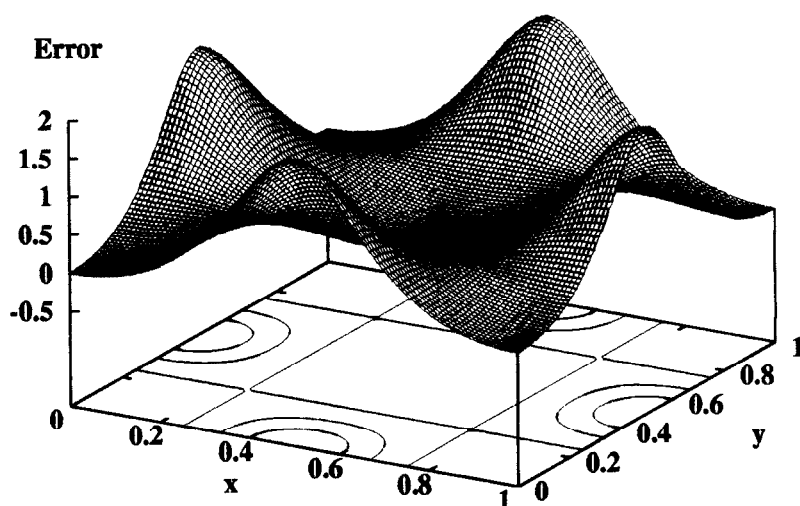


図 2.22: 2次元 CIP 法 Type-B による空間補間誤差 ( $k_x = k_y = 2\pi$ ):  
Type-A における補間手法と同様に、波数  $2\pi$  を持つ  $\cos$  プロファイルに対しては補間関数の境界条件として用いられるグリッド上の値からでは適切な補間関数を形成することができない。



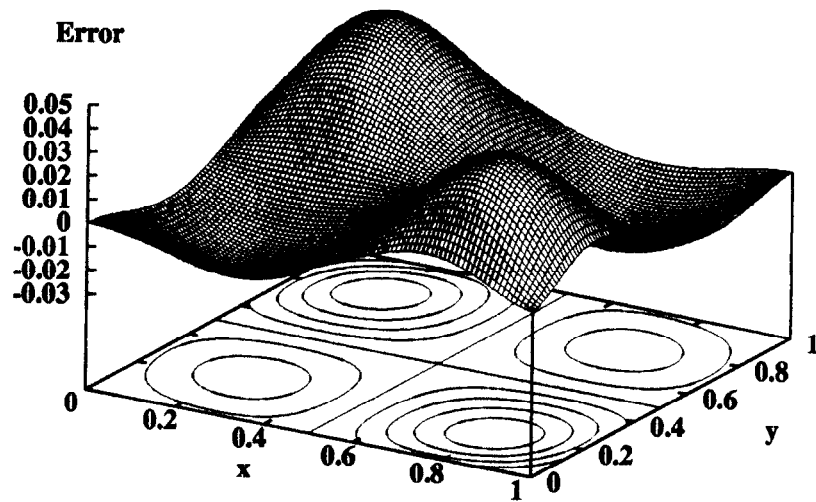


図 2.23: 2次元 CIP 法 Type-B による空間補間誤差 ( $k_x = k_y = \pi$ ):

Type-B における補間手法ではセルに含まれる全格子点上の境界条件を均等に用いることから補間誤差の局所分布は Type-A と比較して低下していることが分かる。

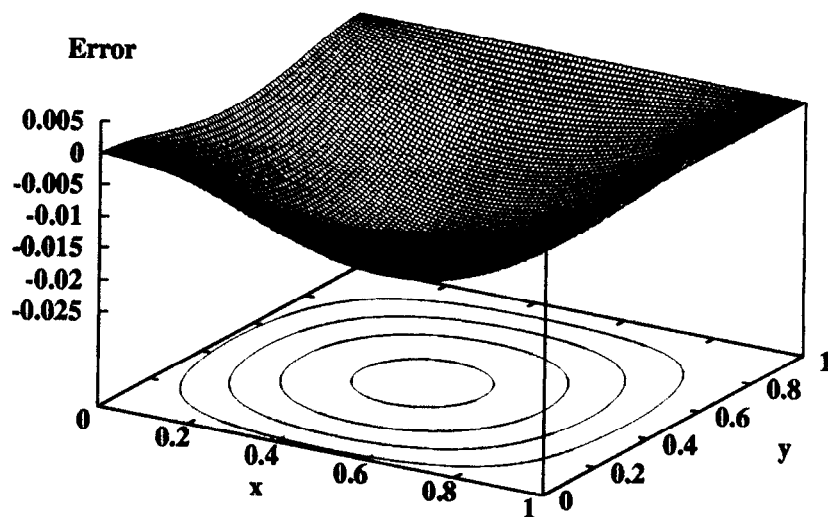


図 2.24: 2次元 CIP 法 Type-B による空間補間誤差 ( $k_x = k_y = 0.5\pi$ ):

Type-B で用いられる空間補間手法では、計算セル内に含まれる全グリッド点上の情報を均等に用いることから、補間誤差はグリッドの中心付近でほぼ最大値をとる。

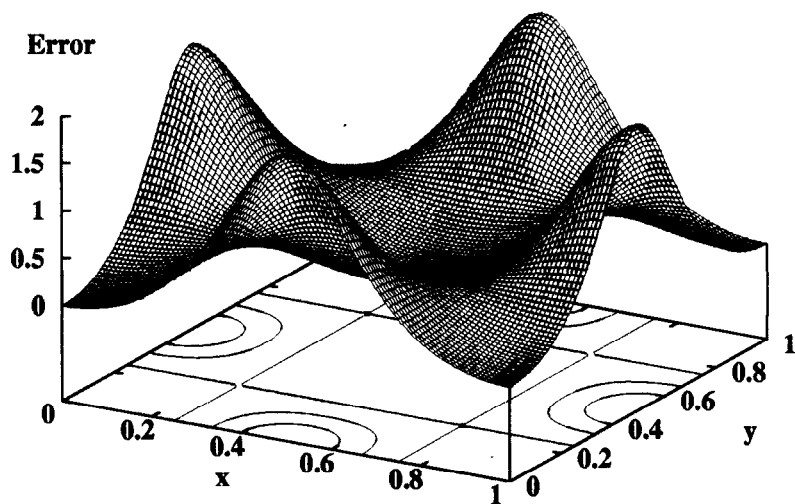


図 2.25: 2次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = 2\pi$ ):

Type-C を用いた場合でも、Type-A、Type-B における補間手法と同様に 3 次多項式に基づいた補間手法であるため、波数  $2\pi$  を持つ  $\cos$  関数により表現されるプロファイルを適切に補間することができない。

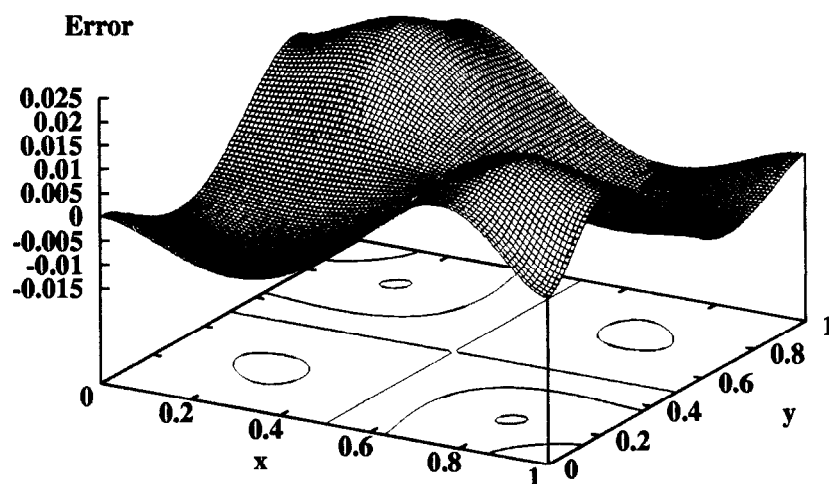


図 2.26: 2次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = \pi$ ):

Type-C ではセルに属する 4 グリッド点上の物理量とその一次空間微係数ならびに  $\partial^2 f / \partial x \partial y$  を用いて補間関数を形成することから Type-A と比べてわかるようにセル内で空間補間誤差の局所分布は無く、また Type-B と比べても空間補間誤差を低減できていることがわかる。

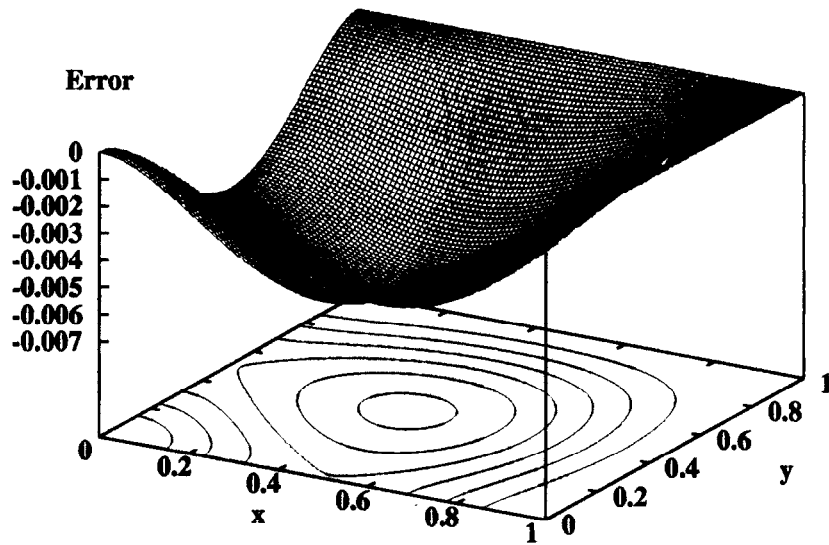


図 2.27: 2次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = 0.5\pi$ ):

Type-C ではセルに属する 4 グリッド点上の物理量とその一次空間微係数ならびに  $\partial^2 f / \partial x \partial y$  を用いて補間関数を形成することから Type-A と比べてわかるようにセル内で空間補間誤差の局所分布は無く、また Type-B と比べても空間補間誤差を低減できていることがわかる。

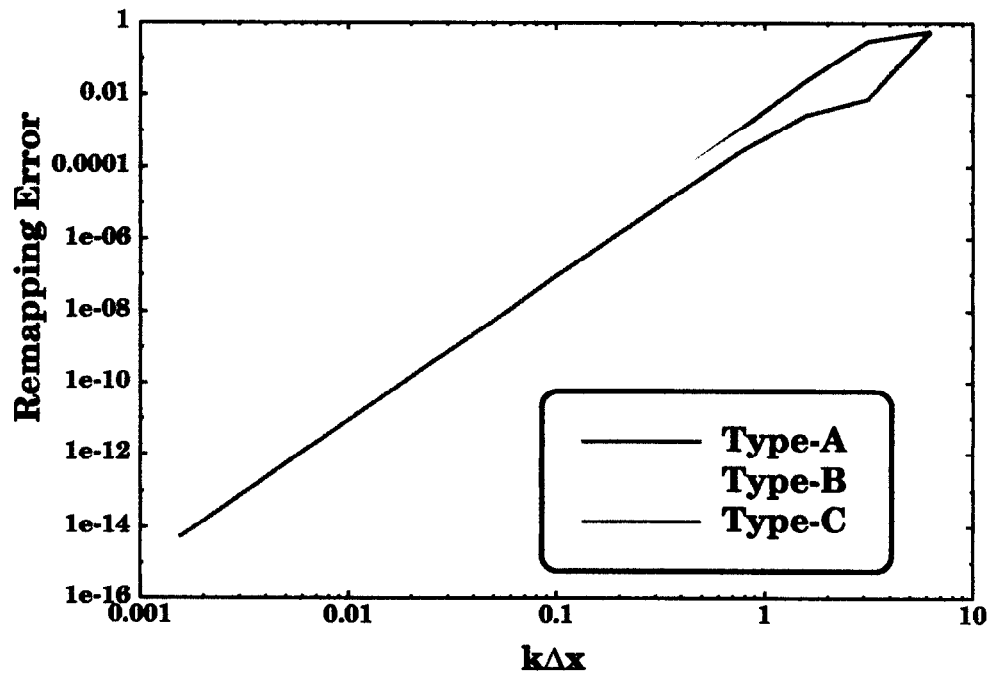


図 2.28: 2次元 CIP 法による空間補間誤差:

Type-A, Type-B に比べ Type-C による空間補間誤差は若干であるもののほぼ全波長空間において少ない結果が得られた。Type-A, B, C 全ての空間補間手法において空間補間誤差は  $\Delta x^4$  の精度を示した。

3次元に拡張された CIP 法は付録 A で示すように 2次元 CIP 法で用いられる補関数形成手法 Type-A, Type-C の計算手順を基本としているものが提案されている。それぞれの補関数形成手法は異なるものの、CIP 法の基本的概念である Remapping + Semi-Lagrangian 手法という計算手順を元に双曲型偏微分方程式を扱うことは、1, 2次元の CIP 法と同様であることから、3次元 CIP 法で用いている補関数に関しても空間補関誤差を検査することは有用であると考えられる。

1次元補関による空間補関精度検証より、初期プロファイルを  $\cos$  関数、 $\sin$  関数で与えた場合の空間補関誤差精度の傾向を明らかにしていることから 3次元補関手法に対する精度検証は 2次元補関関数の精度検証と同様に、初期プロファイルを  $f(x, y, z) = \cos(k_x x) \cos(k_y y) \cos(k_z z) + 2$  とし、グリッド間に含まれるプロファイルの波数  $k_x, k_y, k_z$  を変化させることにより空間補関誤差の検証を行う。

図 2.29 に、平均空間補関誤差を

$$\text{Remapping Error} = \sum_i^N \sum_j^M \sum_k^T \left| \frac{f_{\text{interpolated}} - f_{\text{analytic}}}{f_{\text{analytic}}} \right| / (N \times M \times T) \quad (2.41)$$

により見積もった結果を示す。ここで、 $f_{\text{interpolated}}$  は補関された値を示し、 $f_{\text{analytic}}$  は解析解を示す。また、 $N, M, T$  は補関領域の分割数を示し、それぞれ 101 とした。図 2.29 から、Type-A, Type-C の平均空間補関誤差は 1次元、2次元の場合と同じように、 $\Delta x^4$  で得られることを示した。その一方で、Type-C は、Type-A に比べおよそオーダー程度精度が高いことも図 2.29 からわかる。これは、Type-C は全グリッド点において物理量ならびに空間微係数  $\partial f / \partial x, \partial f / \partial y, \partial f / \partial z, \partial^2 f / \partial x \partial y, \partial^2 f / \partial x \partial z, \partial^2 f / \partial y \partial z, \partial^3 f / \partial x \partial y \partial z$  を定義し、それらの全ての連続条件をセルに含まれる 8グリッド上で満たすように補関関数を形成する一方、Type-A では物理量とその空間一次微係数のみを用い、更に空間微係数の連続条件を適用するグリッド点数が少ないことにより、補関関数の空間解像度が異なることが原因として考えられる。

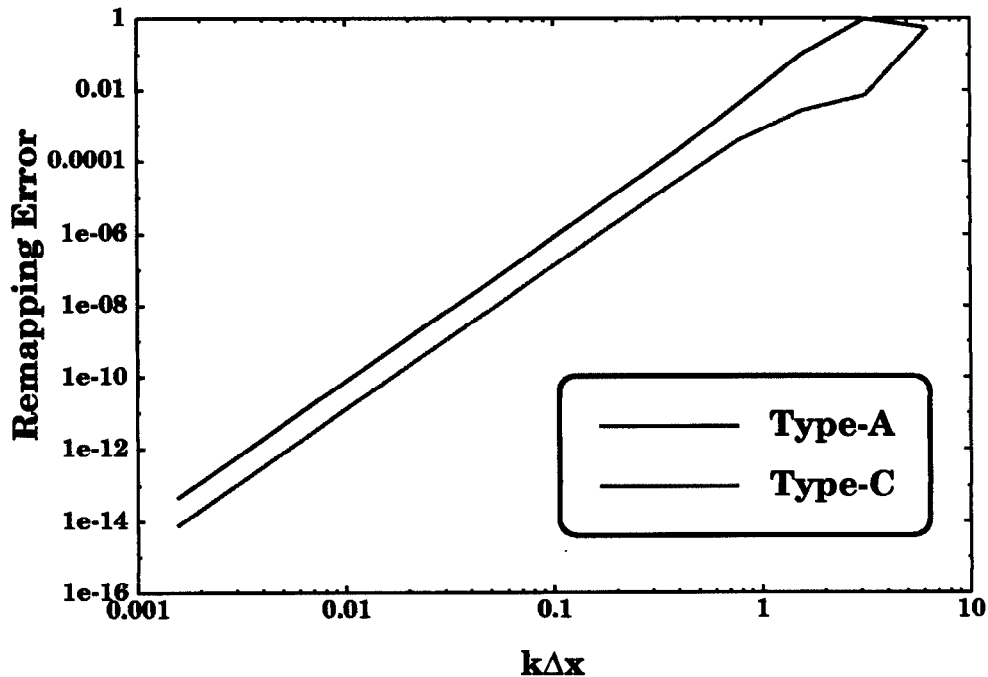


図 2.29: 3次元 CIP 法 Type-A, Type-C による平均空間補関誤差:

セル全体に対する空間補関誤差を平均すると Type-A に比べ Type-C はおよそオーダー程度の精度が改善される。

次に空間補間誤差を式 (2.37) を用いて評価した結果を示す。波数  $k_x = k_y = k_z = 2\pi, \pi, 0.5\pi$  とし、Type-A, Type-C による空間補間誤差の空間分布を見積もったものを図 2.29、図 2.30 - 図 2.32 ならびに図 2.33 - 図 2.35 に示す。全ての図の中で計算対象点は右手前下のグリッドとした。

図 2.30 では式 (2.41) から得られる空間補間誤差の等値面の値は青、緑、黄(赤)色の順に 0.5, 1.0, 1.5 を示しており、同様に図 2.31 では補間誤差 0.2, 1.2, 2.0、図 2.32 では 0.02, 0.11, 0.2、図 2.33 では 0.5, 1.0, 1.5、図 2.34 では 0.002, 0.01, 0.02、図 2.35 では 0.001, 0.005, 0.0075 を表している。図 2.30、図 2.33 から Type-A, Type-C 共に波数  $k_x = k_y = k_z = 2\pi$  を用いた  $\cos$  関数によるプロファイルを表現できないことがわかる。図 2.31 - 図 2.32 から Type-A による空間補間誤差は計算対象点から遠くなるにつれて大きくなる傾向がある。これは Type-A による補間が計算対象点近傍の 4 点における物理量とその 1 階空間微係数の連続条件とその他の 4 点における物理量の連続条件を元に形成されていることを考えると計算対象点近傍では境界条件を適用したグリッドから近いことから高精度補間が行われていると考えられるのでその理由が説明できる。その一方 Type-C から得られた空間補間誤差は空間にほぼ均等に分布している様子がわかり、その絶対値も Type-A から得られたものより小さい値を示した。これは、Type-C 補間では補間対象点を含む計算セルに属する全てのグリッド上の物理量ならびに空間微係数  $\partial f/\partial x, \partial f/\partial y, \partial f/\partial z, \partial^2 f/\partial x\partial y, \partial^2 f/\partial x\partial z, \partial^2 f/\partial y\partial z, \partial^3 f/\partial x\partial y\partial z$  の連続条件を用いた補間関数形成手法を用いていることから、補間対象セル内で補間誤差の局所的な分布が Type-A から得られる補間誤差に比べ低減でき、また境界条件としての情報量も Type-A に比べて多いことから補間精度向上に貢献できることを考慮すると説明付けることが出来る。

また、図 2.31 に着目すると、空間補間誤差が 1.2 つまり 120% の誤差を含む等値面がおおよそ CFL 数で 0.5 近傍に見られ、20% の誤差を示す等値面ですら CFL 数でおおよそ 0.1 近傍に見られる。このことから、3次元 CIP 法 Type-A が大きな CFL 数 ( $\geq 0.1$ ) に対して不安定性を生じる理由を説明できる。

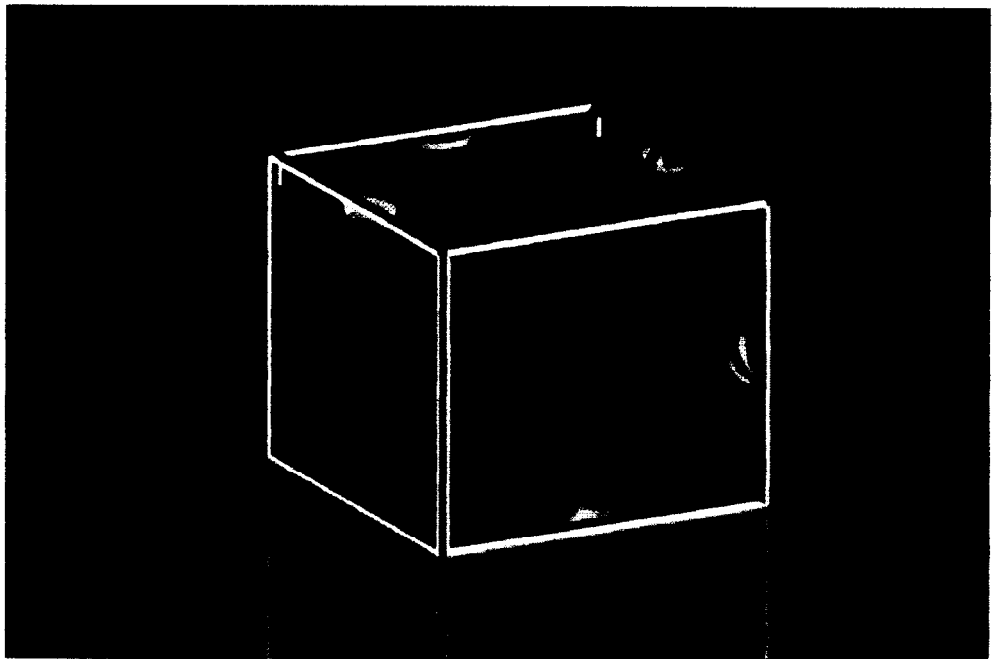


図 2.30: 3次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = k_z = 2\pi$ ):

波数が  $2\pi$  で、 $\cos$  プロファイルを与えた場合、補関数により計算空間内のプロファイルを記述することが出来ない。それぞれの等値面は理論解からの相対誤差 0.5(青), 1.0(緑), 1.5(赤) を示している。

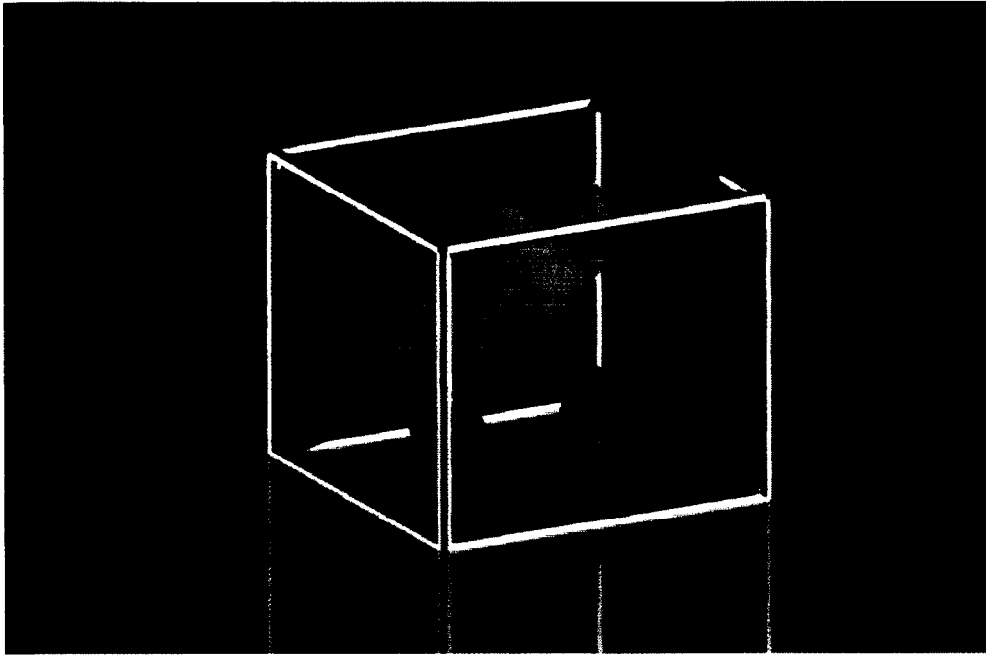


図 2.31: 3次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = k_z = \pi$ ):

Type-A ではセルに属する 8 グリッド点上のうち、計算対象点近傍の 4 点における物理量とその 1 階空間微係数の連続条件とその他の 4 点における物理量の連続条件を用いて補間関数を形成するため、計算対象点から遠い場所では計算誤差が増大する。それぞれの等値面は理論値からの相対誤差 0.2(青), 1.2(緑), 2.0(赤) をあらわしている。

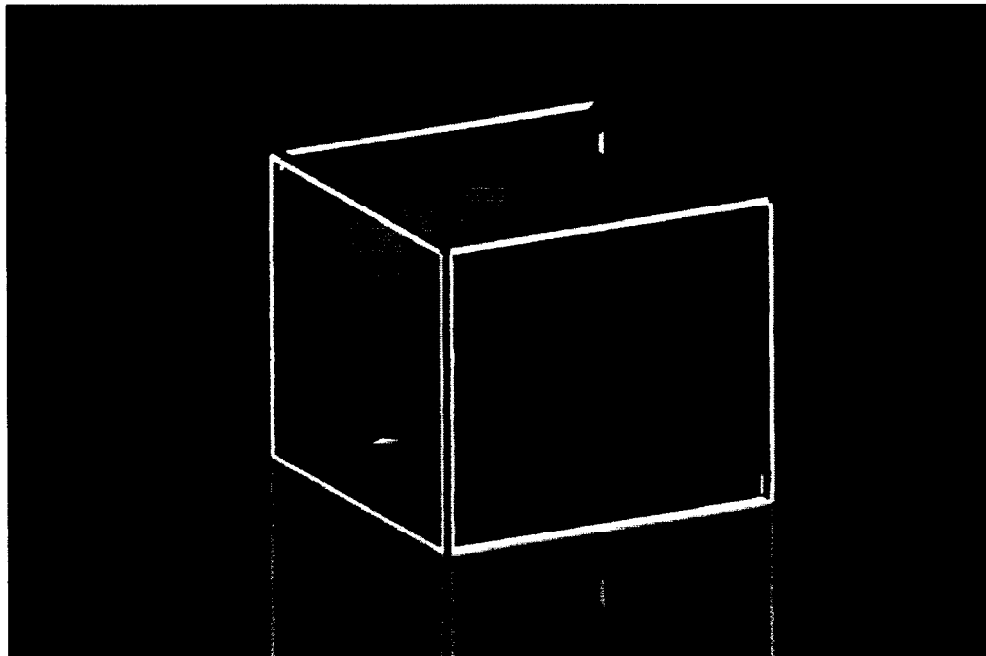


図 2.32: 3次元 CIP 法 Type-A による空間補間誤差 ( $k_x = k_y = k_z = 0.5\pi$ ):

Type-A ではセルに属する 8 グリッド点上のうち、計算対象点近傍の 4 点における物理量とその 1 階空間微係数の連続条件とその他の 4 点における物理量の連続条件を用いて補間関数を形成するため、計算対象点から遠い場所では計算誤差が増大する。それぞれの等値面は理論値からの相対誤差 0.02(青), 0.11(緑), 0.2(赤) をあらわしている。

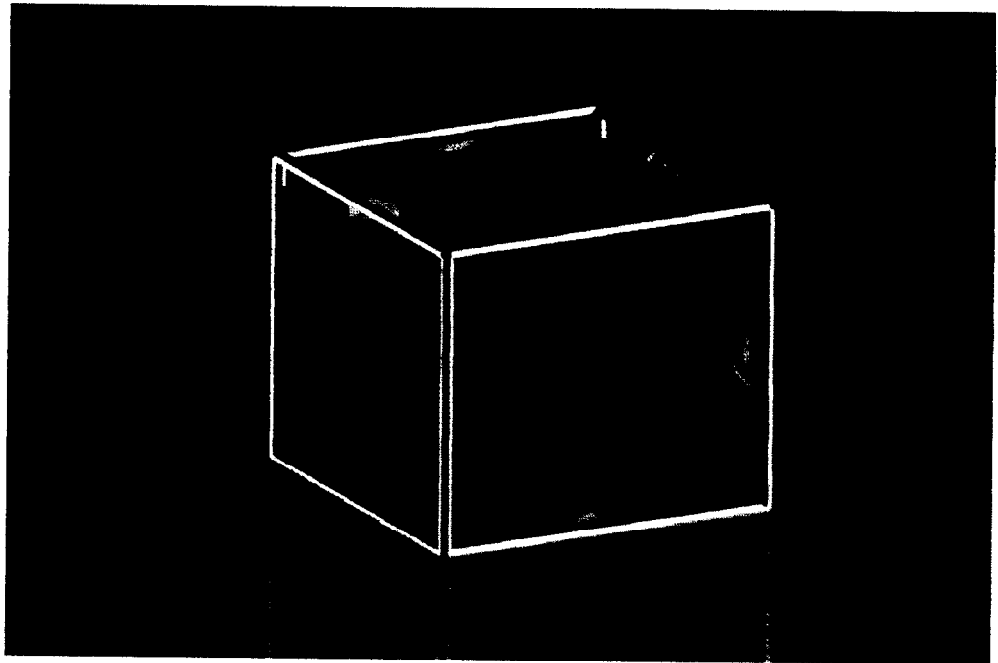


図 2.33: 3次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = k_z = 2\pi$ ):  
波数が  $2\pi$  で、 $\cos$  関数を用いたプロフィールを与えた場合、補間関数により計算空間内のプロフィールを記述することが出来ない。それぞれの等値面は理論解からの相対誤差 0.5(青), 1.0(緑), 1.5(赤) を示している。

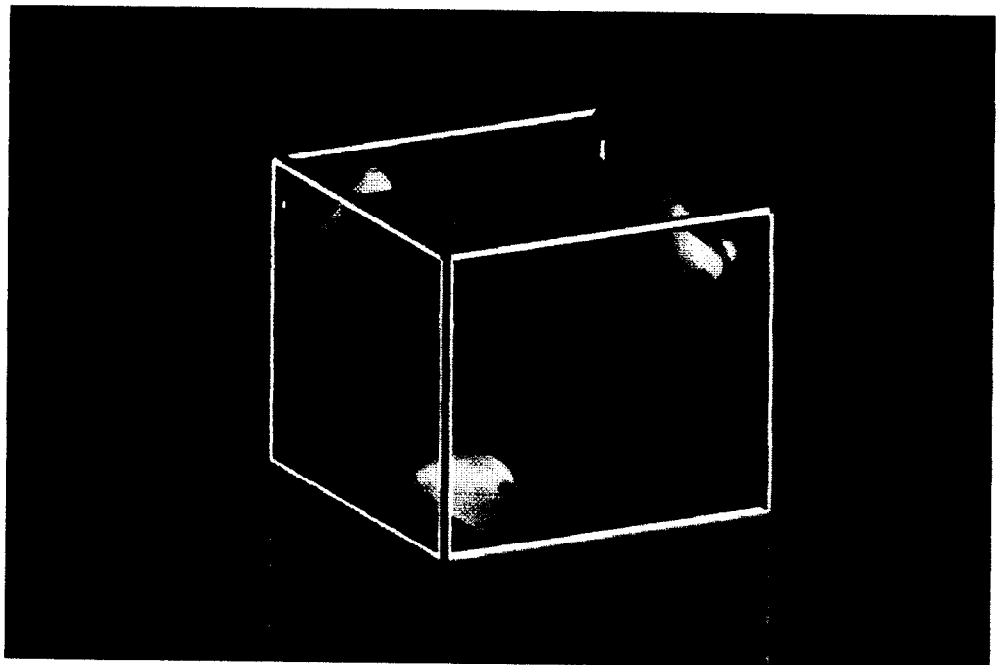


図 2.34: 3次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = k_z = \pi$ ):  
Type-A ではセルに属する 8 グリッド点上のうち、計算対象点近傍の 4 点における物理量とその 1 階空間微係数の連続条件とその他の 4 点における物理量の連続条件を用いて補間関数を形成するため、計算対象点から遠い場所では計算誤差が増大する。それぞれの等値面は理論解からの相対誤差 0.002(青), 0.01(緑), 0.02(黄) を示している。

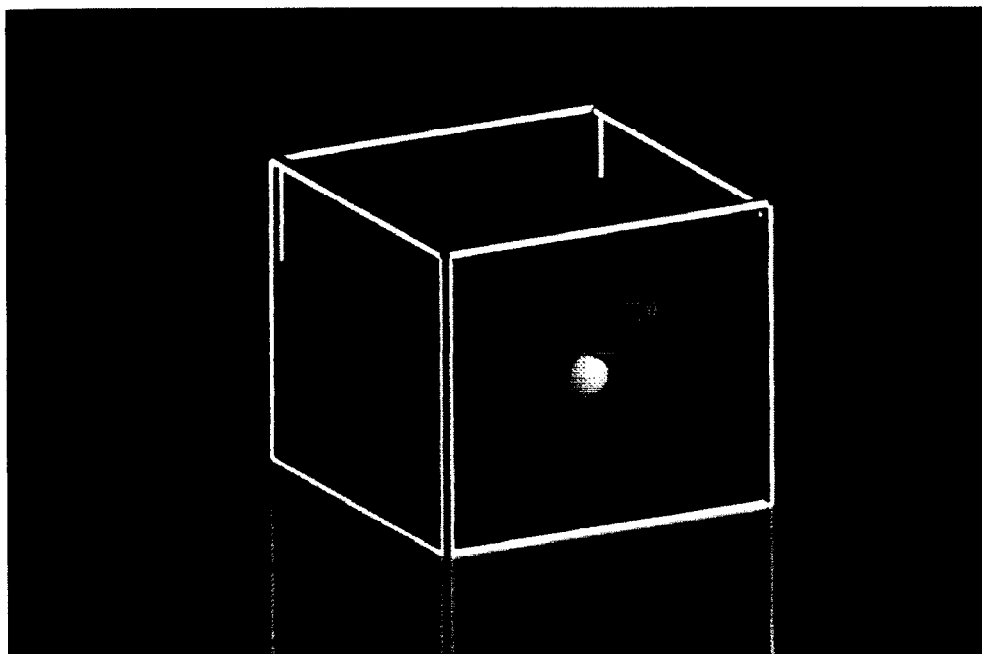


図 2.35: 3次元 CIP 法 Type-C による空間補間誤差 ( $k_x = k_y = k_z = 0.5\pi$ ):

Type-A ではセルに属する 8 グリッド点上のうち、計算対象点近傍の 4 点における物理量とその 1 階空間微係数の連続条件とその他の 4 点における物理量の連続条件を用いて補間関数を形成するため、計算対象点から遠い場所では計算誤差が増大する。それぞれの等値面は理論解からの相対誤差 0.001(青), 0.005(緑), 0.0075(黄)を示している。

以上から、CIP 法の補間関数に関してその特徴を示し、これまで経験的にしか知られていなかった CFL 数に対する数値安定性の原因を空間補間誤差の見地から明らかにした。以下では基礎的な流体方程式を扱い、本節で行った誤差解析を元に IDO 法と CIP 法の計算結果を比較する。

## 2.3 単純な偏微分方程式を用いた基礎精度検証

IDO 法は様々な偏微分方程式 (Partial Differential Equation; PDE) を高精度で一様に扱うために提案された数値計算スキームであり、CIP 法と同様にグリッド点毎に物理量とその低次空間微係数を定義し、それらの値を用いてグリッド間プロファイルをエルミート補間することで有限差分法 (Finite Difference Method; FDM) と比較して少ないメッシュ数で高解像度でプロファイルを表現することを可能にしている。IDO 法は形成した補間関数を対象となるグリッド点上で直接微分し、時間に関してテーラー展開した支配方程式中に含まれる高次空間微係数を求めるオイラー的手法である。

IDO 法を用いた補間関数の形成方法は対象となる偏微分方程式の性質により 2 通りに分類される。偏微分方程式に含まれる移流相を構成する空間微係数を求める際に利用する補間関数は移流相のみで構成される移流方程式の解析解が 1 本の風上方向に向かう特性線によって表現できることから、計算対象グリッド点の移流速度を用いて風上方向を検出し、風上方向に対して風上補間と呼ばれる補間関数を形成する。エルミート補間を用いて風上補間を行う場合、計算対象点を含み 2 グリッド点上の情報をを用いる 3 次風上補間と、4 グリッド点上の情報をを用いる 5 次風上補間が提示されており、5 次風上補間を用いることで 3 次風上補間に比べ高精度計算が可能であることを先の精度検証から示した。また、定常一様速度場に対し 3 次風上補間を用いて 1 次元移流方程式を IDO 法により離散化した場合、CIP 法の離散式に完全に一致することが知られている。



Navier-Stokes 方程式などの流体運動を記述する支配方程式は移流相に加え、それ以外の項を含む時間前進型偏微分方程式により記述される。これらの非移流相に関してはその解を理論的に解析すると特性線の数が 2 を越えるため情報が移流する風向きを限定できない。そこで、非移流相を構成する高次空間微係数を計算するための補間関数として計算対象グリッド点を中心に 3 グリッドにわたって 5 次多項式を用いた中心補間とよばれるエルミート補間を形成する。

時間発展型偏微分方程式を IDO 法により扱う場合、その時間前進は時間に関するテーラー展開

$$f^{n+1} = f^n + f_t^n \Delta t + \frac{1}{2!} f_{tt}^n \Delta t^2 + \frac{1}{3!} f_{ttt}^n \Delta t^3 + \dots \quad (2.42)$$

に基づいて行われる。ここで上付の  $n$  は、計算時間ステップをあらわし、添字の  $t$  は時間  $t$  に関する偏微分をあらわしている  $f_t = \partial f / \partial t$ 。式 (2.42) に含まれる高次時間微係数は対象となる偏微分方程式を時間に関する微分オペレータとみなし計算される。この作業により全ての時間微係数は空間微係数に置換され、それらの空間微係数は風上補間、中心補間によりグリッド点上で計算されることから支配方程式は離散化される。

その一方で、Cubic-Interpolated Propagation (CIP) 法は物理量とその低次空間微係数をグリッド点毎に定義し、それらの値を用いてグリッド間プロファイルを 3 次多項式を用いた 3 次エルミート補間により表現する。また、双曲型偏微分方程式を解析解を用いたセミ・ラグランジュ的手法により補間された値を移流計算することで有限差分法 (Finite Difference Method; FDM) に比べ少ないメッシュ数で高精度な計算結果を示して来た。CIP 法は非移流相に対しては数値安定性の問題から 2 次精度中心差分法を適用した数値計算を行う。2 次元 CIP 法は、対象となるグリッド点を含む近傍の 3 点における物理量、空間微係数 ( $\partial f / \partial x, \partial f / \partial y$ ) の連続条件ならびに計算対象点の対角位置に置けるグリッド上の物理量の連続条件を用いて 2 次元補間関数を形成する Type-A、対象となるグリッド点を含むセルを構成するグリッド 4 点における物理量、空間微係数 ( $\partial f / \partial x, \partial f / \partial y$ ) の連続条件を用いて 2 次元補間関数を形成する Type-B、計算対象グリッドを含むセルを構成する 4 グリッド点上に物理量、空間微係数 ( $\partial f / \partial x, \partial f / \partial y, \partial^2 f / \partial x \partial y$ ) を定義し、それらを用いて 1 次元 CIP 法を多数回利用することで 2 次元補間を行う Type-C の 3 手法が提案されている。それぞれ数値安定性、計算精度の点から異なる性質を示すが、有限差分法と比較するといずれも少ない格子点数で構成される計算空間中で高精度で移流計算を行えるスキームである。2 次元 CIP 法においても非移流相計算には 2 次精度中心差分法を用いている。3 次元 CIP 法では、2 次元 CIP 法の Type-A, Type-C の手順にならった手法で補間関数を形成する手法が提案されている。

IDO 法の多次元化を考えた場合 CIP 法の様に多次元補間関数を形成するのではなく、各軸方向に 1 次元 IDO で用いられるエルミート補間式を適用することで  $\partial^3 f / \partial x \partial y^2$  などの高次空間微係数を見積もり、偏微分方程式を離散化する手法をとる。ただし、CIP 法、IDO 法ともに対象となる物理量に加えその空間微係数もグリッド点毎に定義するため、多次元化を行う場合、計算機の記憶容量を大量に消費する欠点を持っている。ただし、有限差分法に比べ空間解像度が高いことから半分以下の計算グリッド数で有限差分法と同等以上の精度で計算できることも 1 次元ポアソン方程式の解法から示されている。

以上の背景を元に IDO 法を用いて以下では 1 次元移流方程式、1 次元保存方程式、2 次元移流方程式、2 次元保存方程式、楕円型方程式の一つであるポアソン方程式を解き、精度検証を行う。移流計算を含む問題に関してはもう一つのエルミート補間を用いた計算手法である CIP 法との比較検証を併せて行う。

## 1 次元移流方程式

一様定常速度場における 1 次元移流方程式

$$f_t = -u f_x \quad (2.43)$$

の解法は CIP 法による 1 次元移流方程式の解法に完全に一致することが論文中 [3] で示されている。ここで、添字  $t, x$  はそれぞれ時間、空間に関する偏微分を示す;  $f_t = \partial f / \partial t, f_x = \partial f / \partial x$ 。CIP 法は双曲型偏微分方程式の一つである移流方程式を高精度で解くことを目的として開発されたスキームであり、移流計

算に対する精度の高さはこれまでに様々な論文で示されて来た。ここでは、移流速度が空間で一様である速度場に初期プロファイルを正弦関数で与え、初期プロファイルの波数に対し保存性と数値拡散を検証する。対照計算として1次精度風上差分法、3次精度風上差分法について同様の検証を行った。また、CIP法と完全にアルゴリズムが一致する風上3次補間式を用いた数値計算結果とIDO法特有の風上5次補間を用いた数値計算結果の比較を併せて行い、高次補間式を用いる際の性質について検証する。

保存誤差、数値拡散誤差の計算方法はそれぞれ以下の通りである。

$$\begin{aligned} & \sum_i^N (f_{Analytic,i} - f_i) / \sum_i (f_{Analytic,i}) \quad (\text{保存誤差}) \\ & \sum_i^N (|f_{Analytic,i} - f_i| / f_{Analytic,i}) / N \quad (\text{数値拡散誤差}) \end{aligned} \tag{2.44}$$

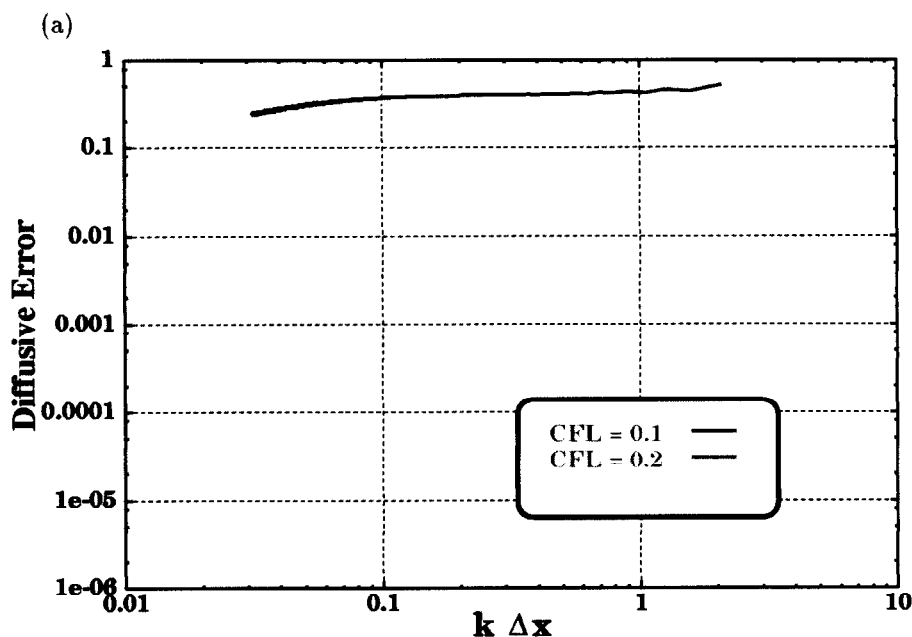
ここで、 $f_{Analytic}$  は、理論解を表す。また、1次精度風上差分法、3次精度風上差分法で用いた離散式は以下の通りである。

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = -u_i^n \frac{f_i^n - f_{i-1}^n}{\Delta x} \quad \text{1次精度風上差分法} \tag{2.45}$$

$$\begin{aligned} \frac{f_i^{n+1} - f_i^n}{\Delta t} &= -u_i^n \bar{f}_x + \frac{1}{2} (u_i^n)^2 \bar{f}_{xx} \Delta t \\ \bar{f}_x &= (2f_{i+1}^n + 3f_i^n - 6f_{i-1}^n + f_{i-2}^n) / 6\Delta x \\ \bar{f}_{xx} &= (-f_{i+2}^n + 16f_{i+1}^n - 30f_i^n + 16f_{i-1}^n - f_{i-2}^n) / 12\Delta x^2 \end{aligned} \tag{2.46}$$

3次精度風上差分法

計算空間  $0 \leq x \leq 1$  に対し、初期プロファイルを  $f(x) = 2 + \sin(2\pi x)$ 、移流速度を  $u(x) = 1$  と与え、周期境界条件 (Periodic Boundary Condition) を用いて計算を行う。 $t = 12$  まで計算した結果から得られた数値拡散誤差を  $\Delta x$  を変化させて1グリッドに含まれる初期プロファイルの波数  $k\Delta x$  に対して見積もった結果を図 2.36 に示す。



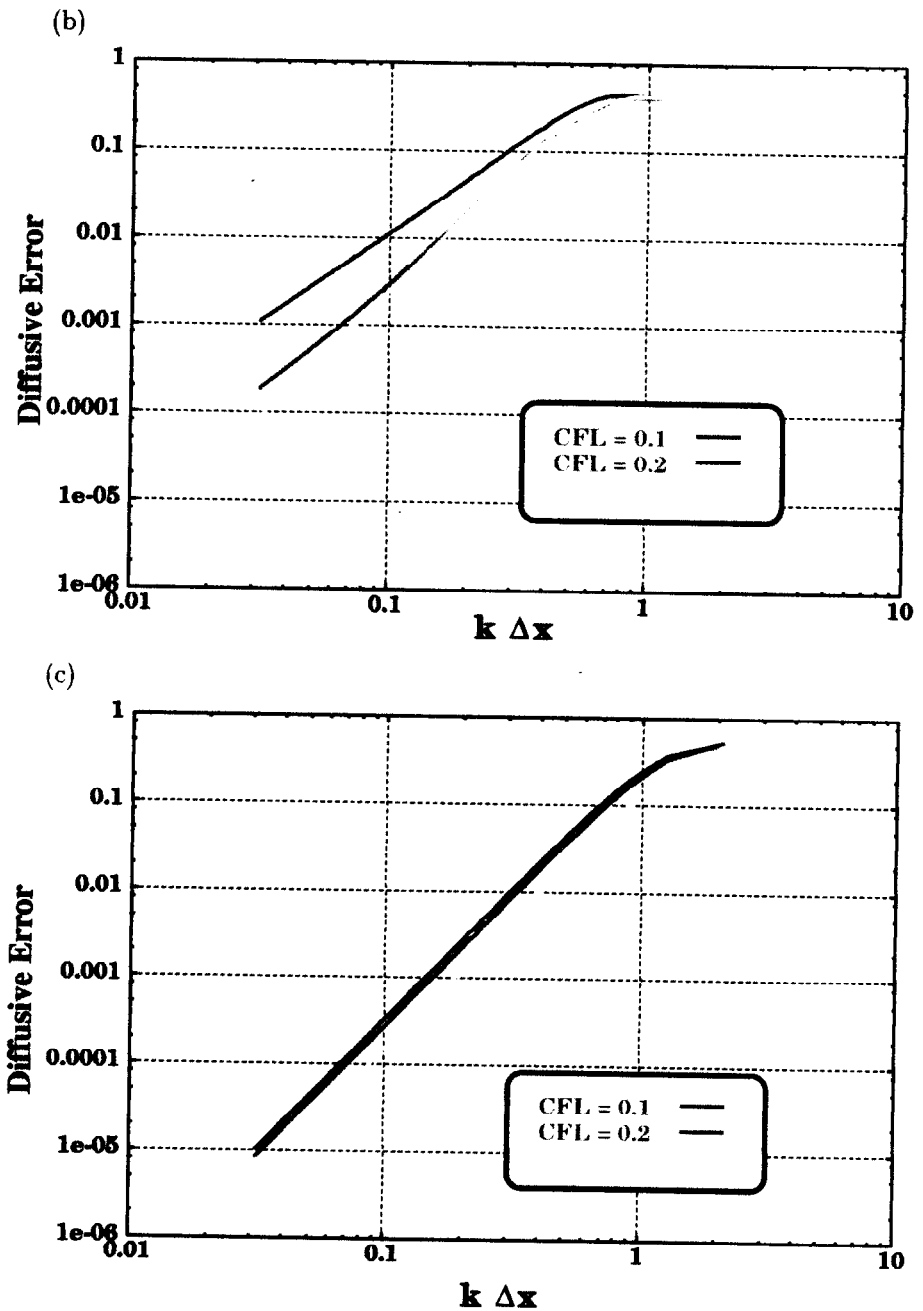


図 2.36: CIP 法による 1 次元移流方程式の数値拡散誤差:

初期プロファイルを正弦関数で与え、1 次元移流方程式を (a) 1 次精度風上差分法、(b) 3 次精度風上差分法、(c) CIP 法でそれぞれ解き、解析解からの数値拡散誤差を、初期プロファイルの波数に対して見積もった。1 次風上差分法では初期プロファイルの波数によらず、一般的に計算時間にもなつてプロファイルが大きく減衰する結果を得た。3 次精度風上差分法による数値計算結果は 1 次精度風上差分法と比較して低数値拡散誤差で解を得ることが出来たが、CFL 数により数値拡散誤差に多少のばらつきが見られる。CIP 法を用いることにより、数値拡散誤差は他の 2 手法に比べ大きく改善され、空間解像度に対して  $\Delta x^3$  の精度を示す結果を得た。また、CFL 数の変化に対する数値拡散誤差の変動も 3 次風上差分法に比べ低減できることを示した。

保存誤差は1次風上差分法、3次風上差分法、CIP法の全てのスキームでCFL数に依存せず $10^{-14}$ 以下で得られた。数値拡散誤差に関しては初期プロファイル中に含まれる波数にも依存するが、1次精度風上差分法とCIP法とでは誤差に $10^4$ 程度の差が生じ、3次精度風上差分法とでも $10^1 \sim 10^2$ 程度の差が生じた。また、3次精度風上差分法はCFL数によって数値拡散誤差に変動が見られたが、CIP法では一貫して同程度の数値拡散誤差で数値計算が行えることを示した、これはCIP法で用いている補間関数から得られる補間誤差が図2.18に示したようにグリッド中でほぼ一様に分布していることから説明付けることが出来る。1次精度風上差分法、CIP法共にCFL=1.0で数値計算を行った場合、風上側のグリッド点に於ける情報を計算対象としているグリッド点に完全に移流させるアルゴリズムを用いていることから安定な数値計算が可能であり、数値計算結果から得られる数値拡散誤差、保存誤差は共に丸め誤差程度になる。一方、3次精度風上差分法ではCFL=1.0で計算すると初期プロファイルの波数 $k\Delta x$ が小さくなるにつれ数値不安定が生じた。原因として有限差分法では物理量のみで高次の補間を行うためにステンシルが数グリッドにわたって形成されるため、高波数成分を含むプロファイルを補間する場合、計算対象となるグリッド点から遠くなるに従い補間された値に誤差が多く含まれることが考えられる。つまり、CFLを大きくとることにより移流原点が計算対象グリッド点から離れるため、数値不安定性が生じたものと思われる。

離散グリッド点により表現される計算空間中では、与えられたプロファイルは離散フーリエ級数展開すると量子化可能な範囲で三角関数による波数成分に分解可能であり、今回示した結果から波数 $\pi$ 以下の三角関数成分に対してCIP法は1次精度風上差分法、3次精度風上差分法に比べ数値拡散、CFL数に対する数値安定性の点で移流方程式に対して優れていることを示した。ただし、プロファイルが何ら波数成分を含まない場合( $f = \text{const.}$ )は全てのスキームで全く同じ数値計算結果が得られる。

次に、風上3次補間式を用いたCIP法による数値計算結果と風上5次補間式を用いた数値計算結果の比較を行う。風上5次補間式を用いた場合、時間に関するテーラー展開を5次まで利用することが可能であるが、風上3次補間との比較を行うことを目的として3次までの展開とした。移流速度 $u$ が空間で一様であることを利用すると、扱う偏微分方程式の時間前進に必要な関係式は以下のようになる。

$$f^{n+1} = f^n + f_t \Delta t + \frac{1}{2!} f_{tt} \Delta t^2 + \frac{1}{3!} f_{ttt} \Delta t^3 \quad (2.47)$$

$$f_x^{n+1} = f_x^n + f_{tx} \Delta t + \frac{1}{2!} f_{ttx} \Delta t^2 \quad (2.48)$$

$$f_t = -u f_x \quad (2.49)$$

$$f_{tx} = -u f_{xx} \quad (2.50)$$

$$f_{ttx} = -u f_{xxx} \quad (2.51)$$

$$f_{tt} = -u f_{tx} \quad (2.52)$$

$$f_{ttx} = -u f_{txx} \quad (2.53)$$

$$f_{ttt} = -u f_{ttx} \quad (2.54)$$

すべての高次空間微係数は移流項から派生していることから風上補間を用いて計算する。先と同様の検証例題を扱った結果、保存誤差は風上3次補間、風上5次補間を用いた計算結果から、共に丸め誤差程度( $< 10^{-15}$ )で求まることを確認した。数値拡散誤差に関しては $k\Delta x = 0.01$ での3次風上補間、5次風上補間から得られた値を比較した場合、10倍程度3次風上補間から得られた誤差が大きくなる結果を示した。この結果から、高次補間関数を用いることで数値拡散誤差に関して改善できることを明らかにした。

CFL数を変化させ、数値拡散誤差を見積もった結果を図2.37に示す。

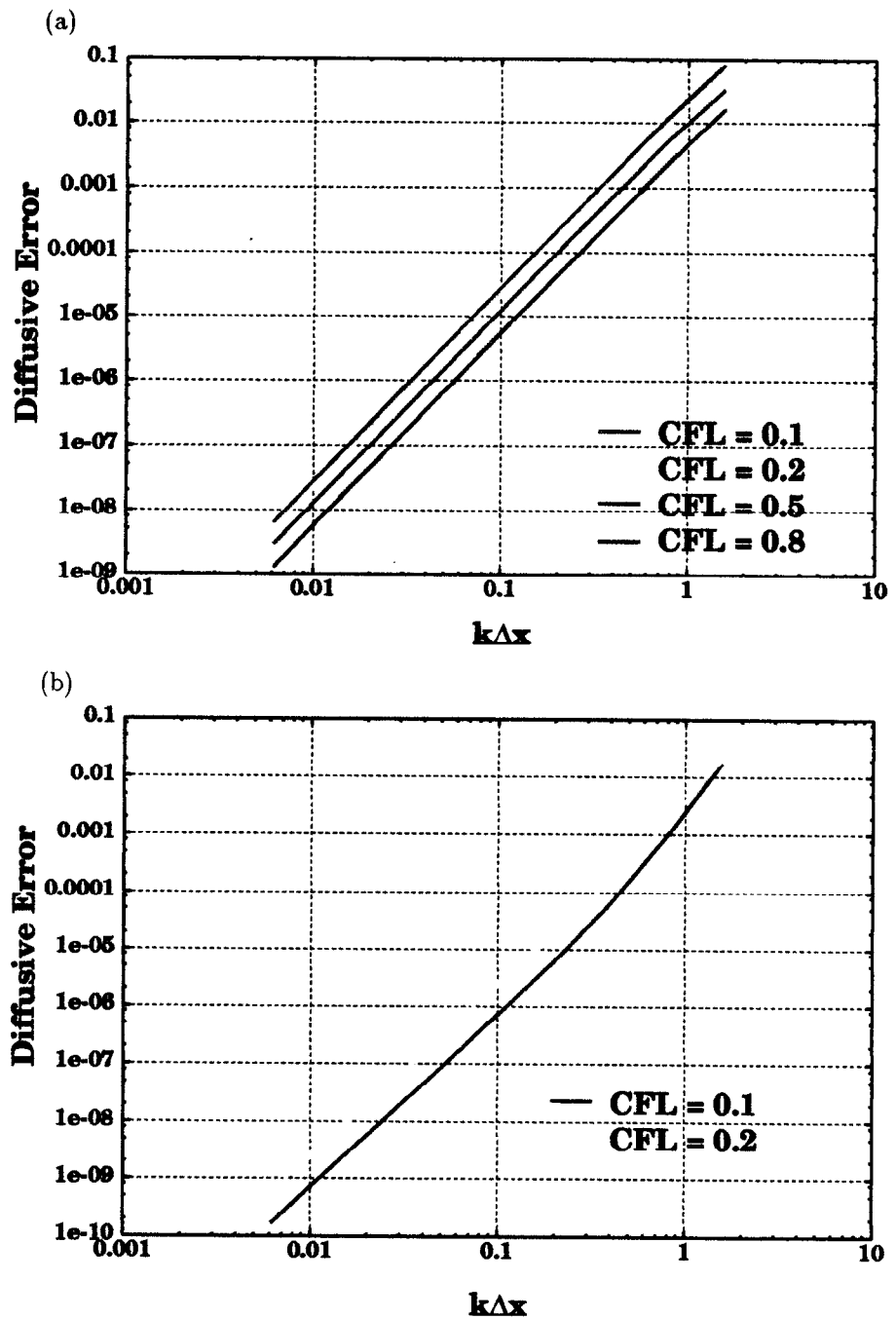


図 2.37: IDO 法による 1 次元移流方程式の数値拡散誤差

(a) 風上 3 次補間、(b) 風上 5 次補間を用いた結果から得られた数値拡散誤差は共に  $\Delta x^3$  精度を示したが、風上 5 次補間では CFL 数が増大するに従って誤差が拡大する傾向が見られた。

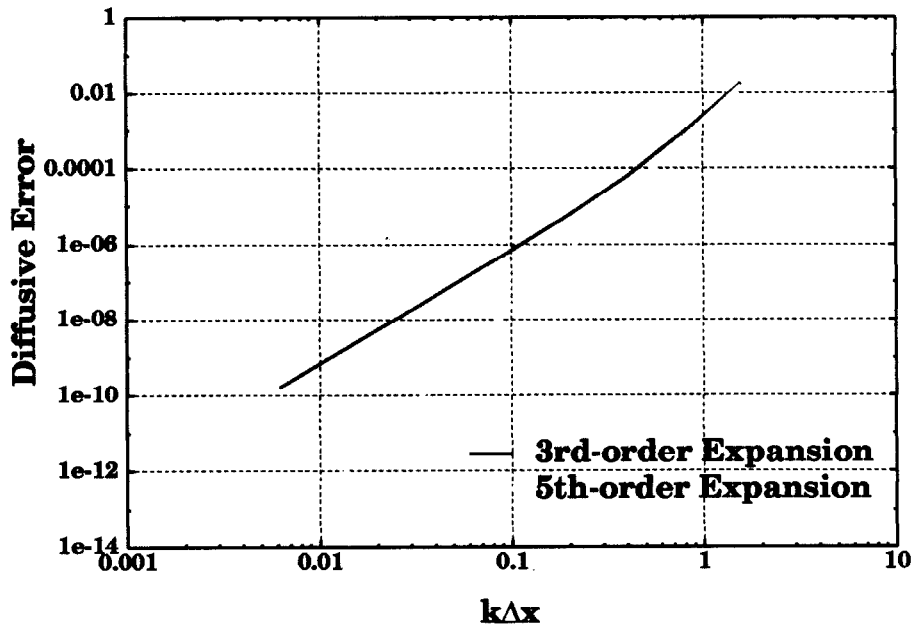


図 2.38: 風上 5 次補間を用いた IDO 法による 1 次元移流方程式の数値拡散誤差  
 テーラー展開の次数を 5 次まで用いることによって風上 5 次補間を用いた IDO 法の 1 次元移流方程式に対する数値拡散誤差は  $\Delta x^5$  の精度で得られる。

CFL 数を増加させるに従って数値拡散誤差が低減される性質を数値実験的に示した。その原因として、CFL 数を大きくすることで 1 計算時間刻み幅  $\Delta t$  でプロファイルが進行する距離が大きくなることから、目的となる計算時間までに必要となる計算ステップ数が減少し、計算ステップ毎に生じる計算誤差を低減できることが挙げられる。風上 5 次補間を用いた場合では  $CFL > 0.3$  で数値不安定性が見られた。この原因として、先の例題としてあげた有限差分法による風上 3 次補間を用いた 1 次元移流方程式の場合と同様に補間領域が拡張することにより高波数成分がグリッド中に含まれる場合、計算対象グリッドから離れるに従い計算誤差が増大することが挙げられる。風上 5 次補間を用いた計算結果から、 $\Delta x^3$  の精度を得ることが出来たが、これは時間前進のためのテーラー展開の次数を 3 次に制限していることが原因と考えられる。その理由について以下に述べる。

IDO 法は、オイラー的手法をとる数値計算スキームであると説明したが、3 次補間関数を用いた離散式が CIP 法の離散式に完全に一致することから、離散式を導く際の手順を異なる視点から捕らえることでセミ・ラグランジュ手法によりその離散式を導くことが出来る。今回の場合では、風上 5 次補間式を CIP 法に利用したと考え、1 次元移流方程式に対する離散式を導くことを考える。5 次風上補間式

$$F(x) = ax^5 + bx^4 + cx^3 + dx^2 + f_{x,i}x + f_i \quad (2.55)$$

を CIP 法で適用する場合に、その未知係数  $a, b, c, d$  を決定するために用いる適合条件を IDO 法での適合条件と同じであると仮定した場合、グリッド点  $i$  における移流速度が負であることを仮定するとその補間関数は

$$F(-\Delta x) = f_{i-1}, F(\Delta x) = f_{i+1}, F(2\Delta x) = f_{i+2}, F_x(\Delta x) = f_{x,i+1}$$

となる。これらから、式 (A.195) - 式 (A.198) で示される関係式が CIP 法に対しても同様に導かれる。CIP 法では、移流方程式の特性線を利用した解析解に従い移流原点位置  $(-u\Delta t)$  における物理量を上記 5 次多項式により補間し、次時刻の値として用いる手法をとることから、 $f^{n+1}$  は、

$$f_i^{n+1} = a(-u\Delta t)^5 + b(-u\Delta t)^4 + c(-u\Delta t)^3 + d(-u\Delta t)^2 + f_{x,i}(-u\Delta t) + f_i \quad (2.56)$$

として求められる。

その一方で、一様定常速度場を仮定し IDO 法で時間 5 次まで展開したテーラー展開の式

$$f_i^{n+1} = f_i^n + \frac{\partial f}{\partial t} \Delta t + \frac{1}{2!} \frac{\partial^2 f}{\partial t^2} \Delta t^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial t^3} \Delta t^3 + \frac{1}{4!} \frac{\partial^4 f}{\partial t^4} \Delta t^4 + \frac{1}{5!} \frac{\partial^5 f}{\partial t^5} \Delta t^5 \quad (2.57)$$

により 1 次元移流方程式の時間前進を行うことを考える。式 (2.49) - 式 (2.54) を空間微係数との関係式に変換した式

$$f_t = -u f_x \quad (2.58)$$

$$f_{tt} = (-u)^2 f_{xx} \quad (2.59)$$

$$f_{ttt} = (-u)^3 f_{xxx} \quad (2.60)$$

$$f_{tttt} = (-u)^4 f_{xxxx} \quad (2.61)$$

$$f_{ttttt} = (-u)^5 f_{xxxxx} \quad (2.62)$$

ならびに、空間微係数と補間関数の未知係数との関係式

$$f_{xxxxx} = 5!a, f_{xxxx} = 4!b, f_{xxx} = 3!c, f_{xx} = 2!d \quad (2.63)$$

を利用すると、式 (2.57) は

$$f^{n+1} = f_i^n + (-u) f_{x,i} \Delta t + (-u)^2 d \Delta t^2 + (-u)^3 c \Delta t^3 + (-u)^4 b \Delta t^4 + (-u)^5 a \Delta t^5 \quad (2.64)$$

と書換えられ、式 (2.56) に完全に一致する。いま、時間に関するテーラー展開を 3 次に抑えて数値計算を行っていることから、時間前進式は

$$f^{n+1} = f_i^n + (-u) f_{x,i} \Delta t + (-u)^2 d \Delta t^2 + (-u)^3 c \Delta t^3 \quad (2.65)$$

で表現される。

つまり時間精度を低下することは CIP 法のようにセミ・ラグランジュ手法的視点から捕らえる場合、補間関数の次数を低減していることと等価になる。これによりグリッド間プロファイルの空間解像度に対する精度が低下する理由が説明付けられる。また、5 次補間式から求めた未知定数  $a, b, c, d$  の内、 $a, b$  を利用していないことから情報が不足することにより数値不安定性が生じたものとする。テーラー展開の展開次数を 5 次として同様に定常一様速度場に対する移流方程式を扱ったところ、CFL 数 1 まで安定かつ高精度に数値計算できることを確認した。

風上 5 次補間関数の本来の精度を検証するため、テーラー展開の次数を 5 次として先と同じ条件下で  $CFL = 0.1$  を用い数値拡散誤差を見積もったものを図 2.38 に示す。テーラー展開を 5 次まで用いることで数値拡散誤差は  $\Delta x^5$  の空間精度で得られることを数値実験的に示した。

### 1 次元保存方程式

IDO 法は非保存系で用いられるため、保存方程式を解くことで保存性について議論を行うことは重要である。ここでは 1 次元保存方程式

$$f_t + u f_x + u_x f = 0 \quad (2.66)$$

を用いて 1 次精度風上差分法、3 次精度風上差分法、CIP 法との比較を用いて検証を行う。また、移流項から派生する高次空間微係数の計算手段として 3 次風上補間、5 次風上補間を適用し、それぞれから得られる計算結果についても検証する。

IDO 法で時間発展型偏微分方程式を解くには式 (A.208)、式 (A.209) に対象となる偏微分方程式を時間に関する微分オペレータとして作用し、時間前進の式を全て空間微係数で表記しなければならない。1

次元保存方程式を移流相に 3 次風上補間、非移流相に 5 次中心補間で扱う場合、時間に関するテーラー展開は 3 次に制限されることから、数値計算に用いる関係式は次のようになる。

$$f^{n+1} = f^n + f_t \Delta t + \frac{1}{2!} f_{tt} \Delta t^2 + \frac{1}{3!} f_{ttt} \Delta t^3 \quad (2.67)$$

$$f_x^{n+1} = f_x^n + f_{tx} \Delta t + \frac{1}{2!} f_{ttx} \Delta t^2 \quad (2.68)$$

$$f_t = -u f_x - u_x f \quad (2.69)$$

$$f_{tx} = -u f_{xx} - u_x f_x - u_{xx} f - \widetilde{u_{xx} f} \quad (2.70)$$

$$f_{ttx} = -u f_{xxx} - 2u_x f_{xx} - u_{xx} f_x - u_x \widetilde{f_{xx}} - 2\widetilde{u_{xx} f_x} - \widetilde{u_{xxx} f} \quad (2.71)$$

$$f_{txxx} = -u f_{xxxx} - 3u_x f_{xxx} - 3u_{xx} f_{xx} - u_{xxx} f_x - u_x \widetilde{f_{xxx}} - \widetilde{u_{xx} f_{xx}} - 2\widetilde{u_{xxx} f_x} - \widetilde{u_{xxxx} f} \quad (2.72)$$

$$f_{tt} = -u_t f_x - u_{tx} f - u f_{tx} - u_x f_t \quad (2.73)$$

$$f_{ttx} = -u_t f_{xx} - 2u_{tx} f_x - u_{ttx} f - u f_{ttx} - 2u_x f_{tx} - \widetilde{u_{xx} f_t} \quad (2.74)$$

$$f_{tttx} = -u_t f_{xxx} - 3u_{tx} f_{xx} - 3u_{ttx} f_x - u_{tttx} f - u f_{tttx} - 3u_x f_{ttx} - 3\widetilde{u_{xx} f_{tx}} - \widetilde{u_{xxx} f_t} \quad (2.75)$$

$$f_{ttt} = -u_{tt} f_x - u_{ttx} f - 2u_{tx} f_t - 2u_x f_{tt} - u_x f_{itt} - u f_{itt} \quad (2.76)$$

ここで、'~' のついた項は非移流項から派生した高次微係数であるので、中心補間を用いて求められる。風上 5 次補間を用いる場合は物理量の 5 階空間微係数まで求めることが出来るので、時間 5 次のテーラー展開の式を利用することが出来る。このことから、風上 5 次補間を用いた精度検証例題では物理量の 5 階時間微分値 ( $f_{tttt}$ ) まで用いた数値計算を行う。

また、下記の手法により求めた解析解を数値解との比較を行うために付記する。

1 次元保存方程式をラグランジュ座標で考え、以下の関係式を得る。

$$\frac{\partial f}{\partial t} = - \left( \frac{\partial u}{\partial x} \right) f \quad (2.77)$$

ここで、物理量  $f$ 、移流速度  $u$  はラグランジュ座標  $x = x_0 + \int u dt$  上に定義される。 $x_0$  は時刻 0 での座標である。上記偏微分方程式を変形し、以下の形を得る。

$$\frac{\partial f}{f} = - \frac{\partial u}{\partial x} \partial t \quad (2.78)$$

移流速度  $u$  が、時間の関数でない場合、

$$\frac{\partial f}{f} = - \partial u \frac{\partial t}{\partial x} \quad (2.79)$$

$$= - \frac{\partial u}{u} \quad (2.80)$$

と表記でき、両辺を定積分することで下記の関係を得る。

$$\int_{f^n}^{f^{n+1}} \frac{\partial f}{f} = \int_{u^n}^{u^{n+1}} \frac{\partial u}{u} \quad (2.81)$$

$$f^{n+1} = f^n \frac{u^n}{u^{n+1}} \quad (2.82)$$

式 (2.82) はラグランジュ座標上での流束 (flux) の保存を示している。関係式から時間  $n+1$  におけるラグランジュ座標  $x_0 + \int_0^{n+1} u dt$  を求めることで  $u^{n+1}$  を決定できることから、 $f^{n+1}$  を解析的に求めることが出来る。時刻  $n+1$  におけるラグランジュ座標は以下のように求められる。速度の定義式

$$\frac{\partial x}{\partial t} = u(x) \quad (2.83)$$



より、

$$\int_{x_0}^{x_0+\Delta x} \frac{1}{u(x)} dx = \int_0^{\Delta t} dt \quad (2.84)$$

の関係が得られる。積分を簡単にするため、以後の数値計算に用いる速度場を次のように定める。

$$u(x) = \frac{1}{1 + 0.5 \sin(kx)} \quad (2.85)$$

ここで、 $k = 2\pi$  とした。式 (2.84) は定積分することで、

$$\left[ x - \frac{0.5}{k} \cos(kx) \right]_{x_0}^{x_0+\Delta x} = \Delta t \quad (2.86)$$

と表される。数学的に式 (2.86) を解くことは不可能だが、数値積分手法を用いることで  $\Delta x$  を求めることが可能である。本論文では式 (2.86) を Newton 法を用いて解き、式 (2.82) から解析解を求めた。

計算条件は次の通りである。空間  $0 \leq x \leq 1$  において速度場を  $u(x) = 1/(1 + 0.5 \sin(kx))$ ;  $k = 2\pi$  と与え、初期プロファイルを  $f = 1(0.25 \leq x \leq 0.45)$ ,  $f = 0$ (その他の点) とした。  $\Delta x = 0.01$ 、周期境界条件を採用し、 $CFL = \Delta t/\Delta x = 0.2$  で  $t = 0.8$  まで計算を行った。CFL 数は移流速度場が一様でないことから代表速度を 1 として計算した。

1 次精度風上差分法、3 次精度風上差分法、CIP 法による計算結果を図 2.39 に示す。

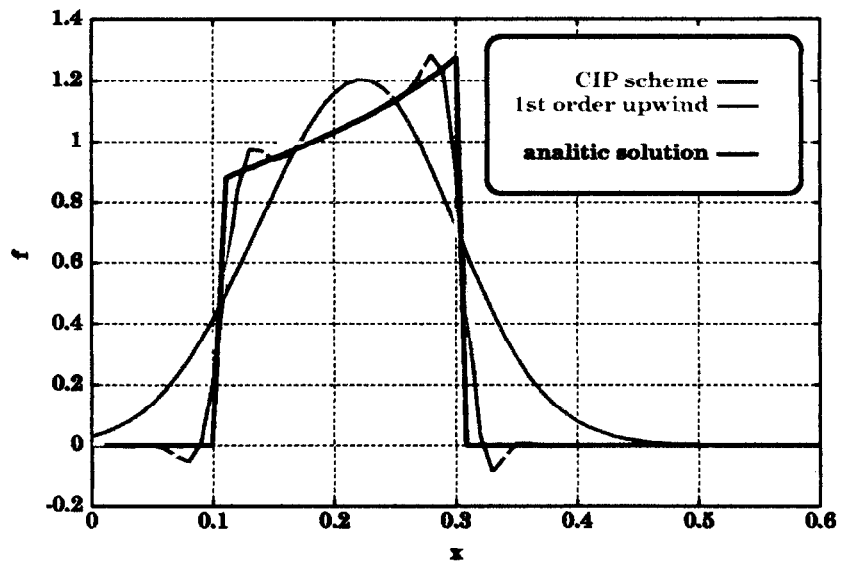


図 2.39: CIP 法による 非一様速度場における 1 次元保存方程式の検証:

非一様速度場で 1 次元保存方程式を 1 次精度風上差分法、3 次精度風上差分法、CIP 法を用いて解いた数値計算結果。

表 2.1: 1次元保存方程式の数値解の保存誤差

スキーム	数値拡散誤差	保存誤差
CIP 法	$1.693251 \times 10^{-2}$	$1.538685 \times 10^{-4}$
1次精度風上差分法	$3.125472 \times 10^{-1}$	$1.925281 \times 10^{-1}$
3次精度風上差分法	$3.143044 \times 10^{-2}$	$3.078503 \times 10^{-3}$

1次精度風上差分法での数値計算結果は位相は解析解によく一致しているものの、プロフィールに関しては解析解と比較した場合、不連続面の記述精度、プロフィールの表現精度が低い。3次精度風上差分法による数値計算結果は解析解と良く一致した形状を示しているが、CIP法による数値計算結果と比較すると数値拡散がやはり大きいことが分かる。 $t=1$ における数値拡散誤差、保存誤差を表 2.1 に示す。保存性に関しては、移流速度が一樣な速度場ではスキーム間にほとんど差が見られなかったが、保存方程式を非一樣速度場で扱うと、CIP法は、1次精度風上差分法、3次精度風上差分法に比べ  $10^1 \sim 10^2$  程度良い結果を得た。数値拡散誤差に関しては  $2 \sim 10^1$  程度良い結果を示した。

次に、IDO法による数値計算結果を図 2.40 に示す。

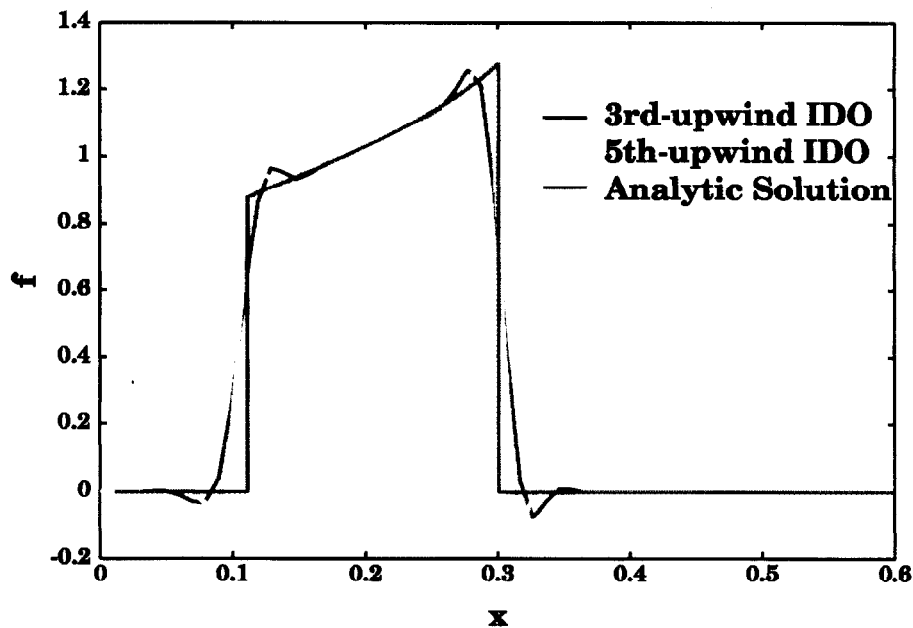


図 2.40: IDO法による1次元保存方程式の数値計算結果

IDO法の風上3次補間、風上5次補間を用いて1次元保存方程式を数値計算した計算結果

IDO法で1次元保存方程式を扱った場合、風上補間に5次多項式を用いて得られた数値計算結果は、3次多項式を用いて得られた数値計算結果のプロフィールに比べ、不連続面の記述精度が向上する結果を得た。 $t=1$ における数値計算結果と理論解との数値拡散誤差、保存誤差は表 2.2 のようになる。

表 2.2: IDO 法による 1 次元保存方程式の数値拡散誤差ならびに保存誤差:

風上 5 次補間を用いた場合と風上 3 次補間を用いた場合では、数値拡散誤差はほぼ同程度で求められるが、保存誤差に関しては風上 5 次補間を用いることでオーダー程度改善できることを示した。

	数値拡散誤差	保存誤差
風上 3 次補間	$1.596787 \times 10^{-2}$	$1.003441 \times 10^{-6}$
風上 5 次補間	$1.122212 \times 10^{-2}$	$6.052395 \times 10^{-8}$

保存誤差は補間式によらず CIP 法により計算した結果より高い保存性を示した。これは、非移流相計算を CIP 法に比べ高精度で計算していることが寄与していると考えられる。3 次多項式を用いた数値計算結果と 5 次多項式を用いた数値計算結果から得られた保存誤差にはオーダー以上の差がみられたが、対象となる初期プロファイル形状、速度場によりこれらの値は変動すると思われる。

以上の結果から、IDO 法を用いることで風上差分法、また CIP 法に比べ高精度な数値計算が可能であることを示した。

## 2 次元移流方程式

1 次元問題に対するの基礎的な検証から、IDO 法は有限差分法 (Finite Difference Method; FDM) や、CIP 法に比べ高精度数値計算が可能であることを示した。より一般的な問題に対応させるため多次元問題への適用を考え、2 次元移流方程式

$$f_t + uf_x + vf_y = 0 \quad (2.87)$$

を解き検証を行う。多次元問題を扱う場合、IDO 法の計算アルゴリズムは CIP 法による手続きと一致することが無いため、相互関係を検証することを目的として、CIP 法で 2 次元問題を扱う手法として提案されている Type-A、Type-B、Type-C の 3 通りと、IDO 法との比較検討を式 (2.88) から得られる保存誤差、数値拡散誤差を用いて行う。

$$\frac{\sum_{i=1}^N \sum_{j=1}^M f_{analytic,i,j} - \sum_{i=1}^N \sum_{j=1}^M f_{i,j}}{\sum_{i=1}^N \sum_{j=1}^M (f_{analytic,i,j})} \quad (\text{保存誤差})$$

$$\sum_{i=1}^N \sum_{j=1}^M \left| \frac{f_{analytic,i,j} - f_{i,j}}{f_{analytic,i,j}} \right| / \text{MeshNumber} \quad (\text{数値拡散誤差}) \quad (2.88)$$

定常速度場に於ける 2 次元移流方程式を IDO 法で扱う場合、以下の関係式を用いる。

$$f^{n+1} = f^n + f_t \Delta t + \frac{1}{2!} f_{tt} \Delta t^2 + \frac{1}{3!} f_{ttt} \Delta t^3 \quad (2.89)$$

$$f_x^{n+1} = f_x^n + f_{tx} \Delta t + \frac{1}{2!} f_{ttx} \Delta t^2 \quad (2.90)$$

$$f_y^{n+1} = f_y^n + f_{ty} \Delta t + \frac{1}{2!} f_{tty} \Delta t^2 \quad (2.91)$$

$$f_{xy}^{n+1} = f_{xy}^n + f_{txy} \Delta t + \frac{1}{2!} f_{ttxy} \Delta t^2 \quad (2.92)$$

$$f_t = -uf_x - vf_y \quad (2.93)$$

$$f_{tx} = -uf_{xx} - u_x f_x - v f_{xy} - v_x f_y \quad (2.94)$$

$$f_{ty} = -uf_{xy} - u_y f_x - v f_{yy} - v_y f_y \quad (2.95)$$

$$f_{txy} = -uf_{xxy} - u_x f_{xy} - u_y f_{xx} - u_{xy} f_x - v f_{xyy} - v_x f_{yy} - v_y f_{xy} - v_{xy} f_y \quad (2.96)$$

$$f_{txx} = -uf_{xxx} - 2u_x f_{xx} - u_{xx} f_x - v f_{xxy} - 2v_x f_{xy} - v_{xx} f_y \quad (2.97)$$

$$f_{tyy} = -uf_{xyy} - u_y f_{xy} - u_{yy} f_x - v f_{yyy} - v_y f_{yy} - v_{yy} f_y \quad (2.98)$$

$$f_{txxy} = -uf_{xxyy} - u_y f_{xxx} - 2u_x f_{xxy} - 2u_{xy} f_{xx} - u_{xx} f_{xy} - u_{xxy} f_x \\ - v f_{xxyy} - v_y f_{xxy} - 2v_x f_{xyy} - 2v_{xy} f_{xy} - v_{xx} f_{yy} - v_{xxy} f_y \quad (2.99)$$

$$f_{txyy} = -uf_{xyyy} - u_x f_{xyy} - u_y f_{xxy} - u_{xy} f_{xy} - u_{yy} f_{xx} - u_{xyy} f_x \\ - v f_{xyyy} - v_x f_{yyy} - v_y f_{xyy} - v_{xy} f_{yy} - v_{yy} f_{xy} - v_{xyy} f_y \quad (2.100)$$

$$f_{tt} = -u f_{tx} - v f_{ty} \quad (2.101)$$

$$f_{tx} = -u f_{txx} - u_x f_{tx} - v f_{txy} - v_x f_{ty} \quad (2.102)$$

$$f_{ty} = -u f_{txy} - u_y f_{tx} - v f_{tyy} - v_y f_{ty} \quad (2.103)$$

$$f_{txy} = -u f_{txxy} - u_x f_{txy} - u_y f_{txx} - u_{xy} f_{tx} - v f_{txyy} - v_x f_{tyy} - v_y f_{txy} - v_{xy} f_{ty} \quad (2.104)$$

$$f_{ttt} = -u f_{ttx} - v f_{tty} \quad (2.105)$$

1次元保存方程式と同様に時間に関するテーラー展開に含まれる時間微分項を全て空間微分項で置き換えなければならない。 $f_{xx}, f_{xxx}, \dots, f_{yy}, f_{yyy}, \dots$ 等は式 (A.192) ないし、式 (A.203) で求めることができるが、 $f_{xxy}, f_{xyy}, \dots$ は1次元IDO法をそれぞれ $f_y, f_x$ に対して $y$ 方向、 $x$ 方向に適用することで求める。

計算空間 $-0.5 \leq x \leq 0.5, -0.5 \leq y \leq 0.5$ において、初期プロファイルを $f = \cos(2\pi x) \cos(2\pi y)$ と与え、グリッド数を変化させることでグリッド当たりのプロファイルの波長に対する誤差を見積もる。CIP法を用い、周期境界条件下において、移流速度を全格子点上で $u = -1, v = -1$ と与え、 $\Delta t$ を、 $0.1\Delta x, 0.2\Delta x, 0.5\Delta x$ と変化させ、 $t = 1$ まで計算を行う。式 (2.88) を用いて求めた数値拡散誤差を図 2.41 ~ 図 2.43 に示す。保存誤差は全ての場合において丸め誤差程度で求められることを確認した。

図 2.41 ~ 図 2.43 から、数値拡散誤差はCIP法 Type-A, B, C に依らず $\Delta x^3$ の精度を示した。また、Type-Aによる数値計算結果から求めた数値拡散誤差は、CFL数( $\Delta t$ )に対する依存性があまり生じなかったが、Type-B、Type-Cを用いた計算ではCFL数を大きくすることで数値拡散誤差が低減できることを明らかにした。通常CFL数を大きくすることで目標計算時間までに必要となる計算時間ステップ数が減少することから生じる数値拡散誤差は低減するが、Type-Aを用いた空間補間では、CFL数を増大させることで移流原点位置が計算対象点から離れるため、補間誤差が増大することが計算時間ステップ数減少による精度向上効果を打ち消しているものと考えられる。Type-B、Type-Cによる空間補間では、前節で示したようにType-Aに比べ補間誤差の空間分布がほぼ一様に得られることからその様な問題が生じなかったものと説明することが可能である。Type-A, B, Cの3手法から得られた数値拡散誤差の内、もっとも小さい値を示したのはType-Cであり、今回検証を行った最小の $k\Delta x$ での数値拡散誤差を比較すると約1/2程度になることを示した。その理由についても前節で示した補間精度で、Type-Cが最も優れていたことを考慮すれば容易に理解できる。

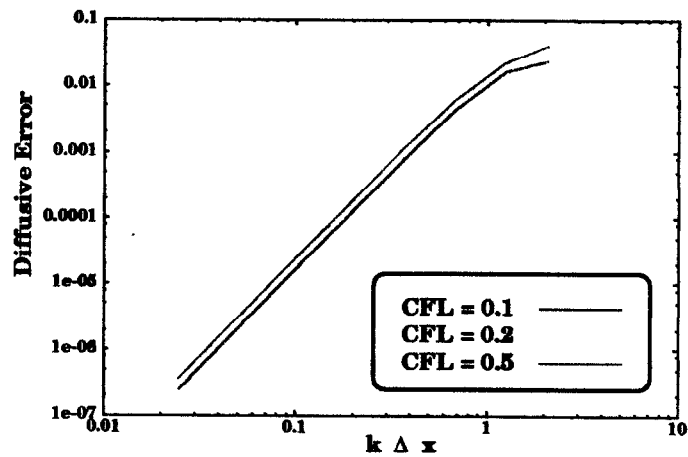


図 2.41: CIP 法 Type-A の 2 次元移流方程式に対する数値拡散誤差  
計算結果から見積もられた数値拡散誤差は  $\Delta x^3$  の精度を示した。

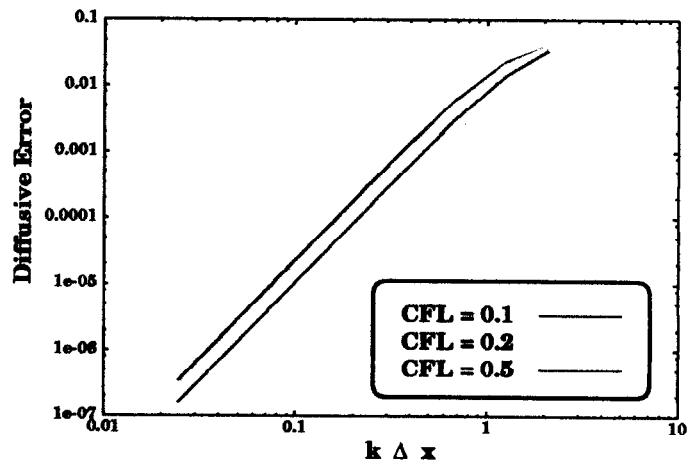


図 2.42: CIP 法 Type-B の 2 次元移流方程式に対する数値拡散誤差:  
計算結果から見積もられた数値拡散誤差は  $\Delta x^3$  の精度を示した。

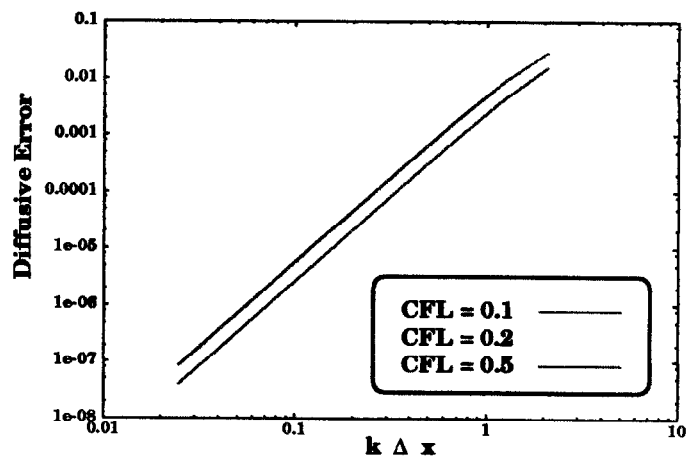


図 2.43: CIP 法 Type-C の 2 次元移流方程式に対する数値拡散誤差:  
計算結果から見積もられた数値拡散誤差は  $\Delta x^3$  の精度を示した。

IDO 法により同様の問題を扱った結果を図 2.44 に示す。移流相から派生した高次空間微係数計算のために風上 3 次補間、風上 5 次補間を用いた場合について検証した結果、両者ともに保存誤差は丸め誤差程度 ( $\approx 10^{-16}$ ) となることを確認した。また、数値拡散誤差は図 2.44 に示すように、風上 3 次補間、風上 5 次補間から得られた結果は共に  $\Delta x^3$  の精度を示すことを示した。ただし、時間に関するテーラー展開の次数を 3 次制限している。1 次元移流方程式の結果から風上 5 次補間を用いた場合、時間前進のためのテーラー展開の次数を 5 次上げることで計算結果から得られる数値拡散誤差は空間に対して  $\Delta x^5$  になることが予想される。

また、風上 5 次補間を用いた方が精度が高くなる傾向が得られた。ただし、風上 5 次補間を用いた場合では CFL (Courant - Friedrich - Levy) 数を高くすることで、数値不安定性が風上 3 次補間を用いた場合に比べ顕著に現れる結果となった。その理由として、1 次元移流方程式で考察したように高波数プロファイルに対する補間精度が計算対象グリッド点から遠くなるに従って低下することが原因と考えられる。この対処法としても時間に関するテーラー展開の次数を 5 次までとることが有効であることは 1 次元移流方程式の数値計算結果から明らかである。

CFL 数が 0.2 を超えると、風上 3 次補間、風上 5 次補間に関わらず両者ともに不安定性のため計算が不可能となった。IDO 法の CFL 数に対する安定性については第 5 章にて検証を行う。

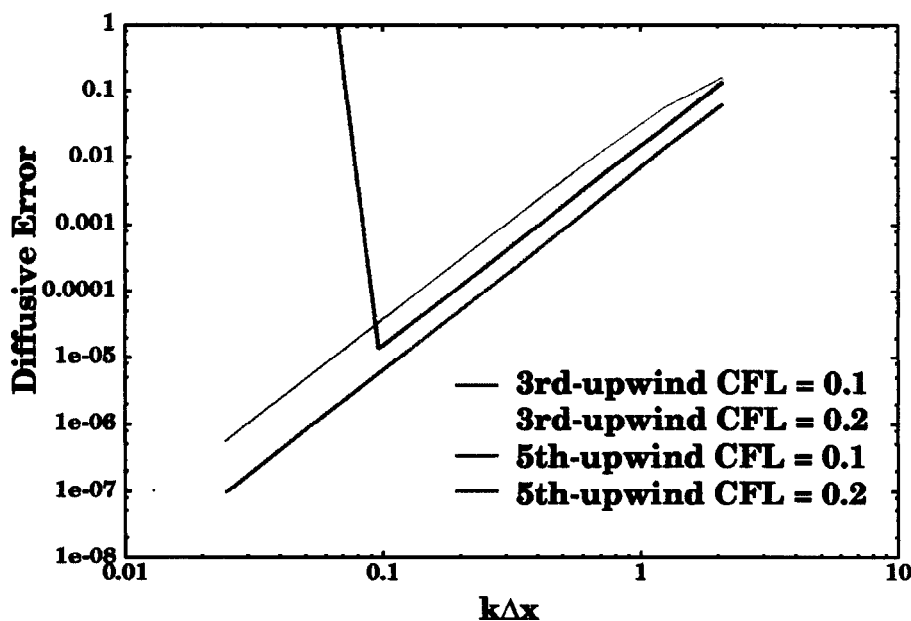
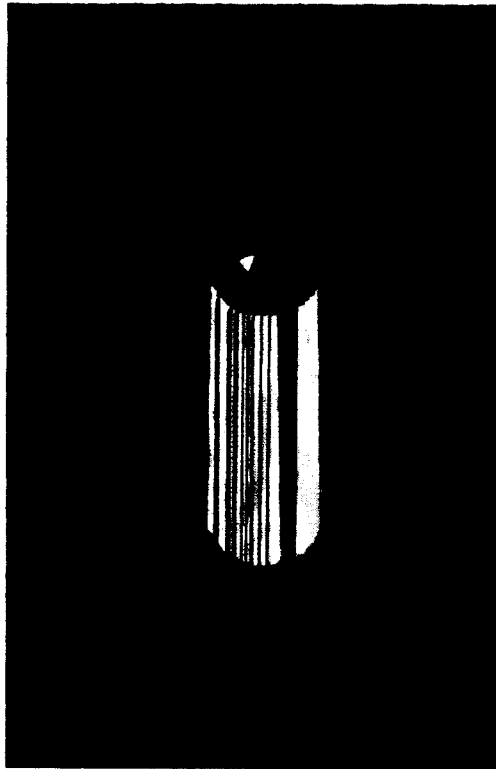


図 2.44: IDO 法による 2 次元移流方程式の数値拡散誤差:

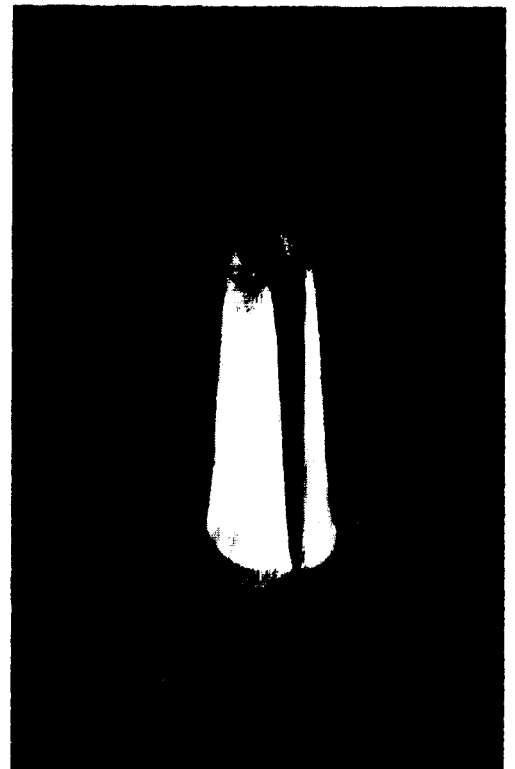
移流項計算のために 3 次風上補間、5 次風上補間を用いた結果から得られた空間精度は共に  $\Delta x^3$  の精度を示した。

以上をふまえ、さまざまな波数成分を含むプロファイルを用いて 2 次元移流方程式の数値拡散誤差を検証するため、複雑な初期プロファイルを持つ例題として有名な Zalesak の例題 [12] を検証する。計算空間  $-0.5 \leq x \leq 0.5, -0.5 \leq y \leq 0.5; \Delta x = 1/100, \Delta y = 1/100$  に対し、回転速度場  $u(x, y) = y, v(x, y) = -x$ 、Zalesak の鍵型プロファイルを初期プロファイルとして与え、 $t = 2\pi$  まで計算を行った。 $\Delta t = 0.005$  とした。CIP 法による計算結果を図 2.45 に示す。

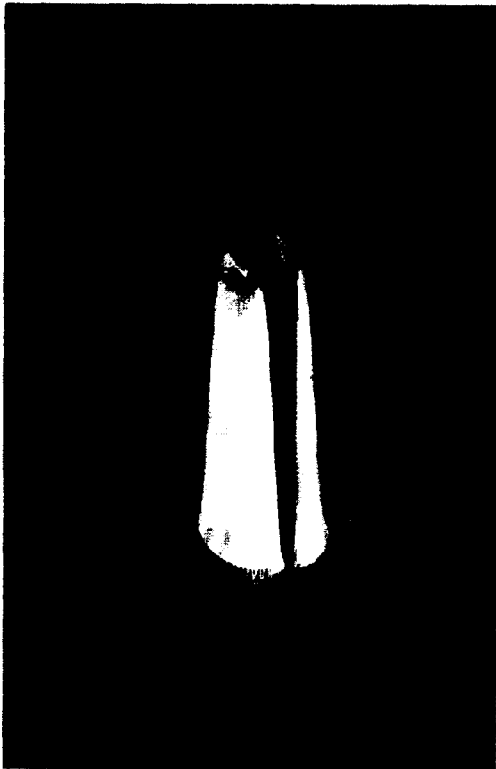
(a)



(b)



(c)



(d)

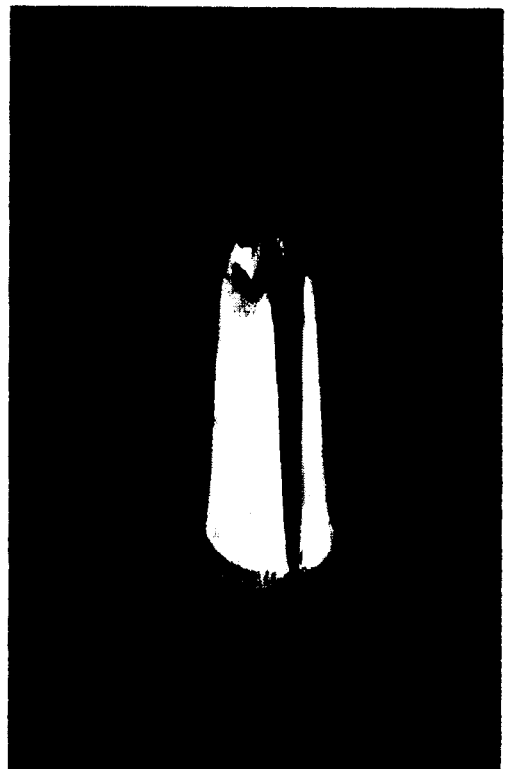


図 2.45: CIP 法による Zalesak の例題検証:

(a) に初期プロファイルを示す。(b)、(c)、(d) はそれぞれ Type-A、Type-B、Type-C による計算結果を示す。

表 2.3: CIP 法による Zalesak 例題の保存誤差と数値拡散誤差

Type	保存誤差	数値拡散誤差
A	$2.437138 \times 10^{-2}$	$1.729665 \times 10^{-2}$
B	$2.437177 \times 10^{-2}$	$1.738157 \times 10^{-2}$
C	$2.396955 \times 10^{-2}$	$1.536648 \times 10^{-2}$

表 2.3 に数値計算結果から求めた保存誤差と数値拡散誤差を示す。保存誤差、数値拡散誤差共に使用する Type によらず数 % 程度であった。その中でも保存誤差、数値拡散誤差に関して Type-C が優れていることが分かる。ただし、どのアルゴリズムを用いても Zalesak 例題を高精度で扱っており、保存誤差、数値拡散誤差も 2 倍と違わない値を示していることから、全ての手法が十分に高精度解を示すと言える。

次に、IDO 法による数値計算結果を図 2.46、図 2.47 に示す。また、それぞれの計算結果から得られた保存誤差、数値拡散誤差を表 2.4 に示す。保存誤差、数値拡散誤差において CIP 法の全ての計算アルゴリズムに対し IDO 法は高精度計算可能であることを明らかにした。数値拡散誤差、保存誤差に関して補間関数の次数による差がほとんど見られなかったが、計算結果から得られたプロファイルから風上 5 次補間を用いた方がオーバーシュートが見られるものの不連続面を風上 3 次補間を用いて得られた数値計算結果と比較して、少ないメッシュ数で表現できていることが分かる。しかし、1 次元問題以上に離散式の導入が複雑になることと、数値計算負荷が大きくなることから、高精度計算が要求されない限り時間精度を 3 次として計算しても表 2.4 から、精度の良い計算が可能と考える。

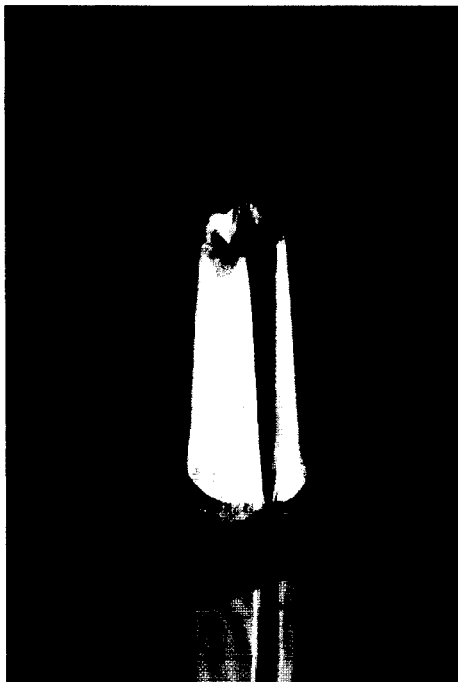


図 2.46: 3 次多項式による風上補間を用いた IDO 法による Zalesak 例題の数値計算結果

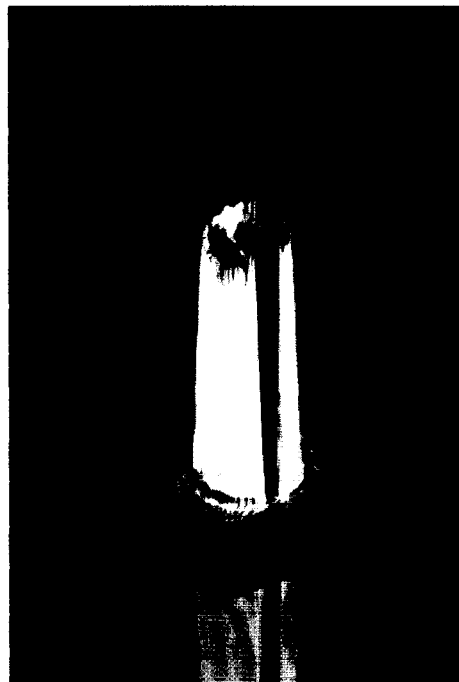


図 2.47: 5 次多項式による風上補間を用いた IDO 法による Zalesak 例題の数値計算結果



表 2.4: IDO 法による 2 次元移流方程式の数値誤差:

補間関数	保存誤差	数値拡散誤差
風上 5 次補間	$1.060467 \times 10^{-7}$	$8.264590 \times 10^{-3}$
風上 3 次補間	$1.696008 \times 10^{-7}$	$1.118160 \times 10^{-2}$

## 2 次元保存方程式 (集中速度場)

一般的な流体现象等を扱うことを考慮し、多次元問題における IDO 法の保存性を 2 次元保存方程式

$$\frac{\partial f}{\partial t} = -u \frac{\partial f}{\partial x} - v \frac{\partial f}{\partial y} - \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) f \quad (2.106)$$

を解き、保存誤差に関する検証を行う。保存誤差は以下の方法を用いて見積もった。

$$\text{Conservation Error} = \left| \frac{\sum_i \sum_j f_{\text{analytic}} - \sum_i \sum_j f_{\text{calc}}}{\sum_i \sum_j f_{\text{analytic}}} \right| \quad (2.107)$$

ここで、 $f_{\text{analytic}}$  は解析解を表し、 $f_{\text{calc}}$  は計算結果のプロファイルをあらわしている。1 次元問題の検証と同様に、IDO 法の風上補間として 3 次多項式を用いた結果と 5 次多項式を用いた解法を比較しながら検証を行う。

計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, \Delta x = 2/100, \Delta y = 2/100$  において、初期プロファイルを  $f = 1(\sqrt{x^2 + y^2} < 0.3^2), f = 0(\text{other points})$  として与え、速度場を  $u(x, y) = -x, v(x, y) = -y$  として与えた。初期プロファイルならびに、 $\Delta t = 0.004$  を用いて  $t = 0.96$  まで CIP 法 Type-A, Type-B, Type-C でそれぞれ計算した結果得られたプロファイルをそれぞれ 図 2.48-2.51 に示す。プロファイル形状は Type-A, B, C に関わらず、ほぼ同形状で求められることがわかる。

周期境界条件を適用し、保存誤差を Type-A, B, C について求めた結果、ならびに解析解から見積もったプロファイル半径を図 2.52 に示す。CIP 法 Type-A, B から見積もられた保存誤差は相対的に 0.1% 程度の差があるものの、ほぼ同じ値を示すことが図 2.52 から理解できる。一方 Type-C から見積もられた保存誤差から、計算開始直後に Type-A, B に比べ保存誤差が急激に増大するものの、その後は保存誤差の悪化は緩やかになることがわかる。また、Type-A, B, C 全ての手法において、理論解から得られるプロファイル半径がおよそ 0.01、つまり  $\Delta x/2$  程度間でプロファイルを圧縮した結果得られる保存誤差が 0.23 程度であることから、上記のような速度場でプロファイルを 1 メッシュ以下に圧縮した場合には 20% 程度の保存誤差が生じることを示した。プロファイルが計算空間中心に集中することで少ない格子点数で不連続面を記述しなければならなくなり、高波数成分が発生する。基礎精度検証の結果からその記述精度が低いことが明らかであり、計算開始直後とプロファイル半径が数グリッド程度になってから誤差の増大が顕著であることから保存誤差増大の原因として不連続面記述精度が挙げられる。

同様の計算条件で IDO 法により計算を行った結果を図 2.54, 図 2.55 に示す。速度場に従ってプロファイルが計算空間の原点に対して集中する計算が精度良く行えていることがわかる。図 2.54 に示した風上 3 次補間を用いた数値計算結果に比べ、図 2.55 に示した風上 5 次補間を用いた数値計算結果の方が少ないグリッド数で不連続面を記述できている様子がわかる。しかしその一方で図 2.55 に示したプロファイルはアンダーシュートやオーバーシュートが図 2.54 に示したプロファイルに比べ顕著に現れている。

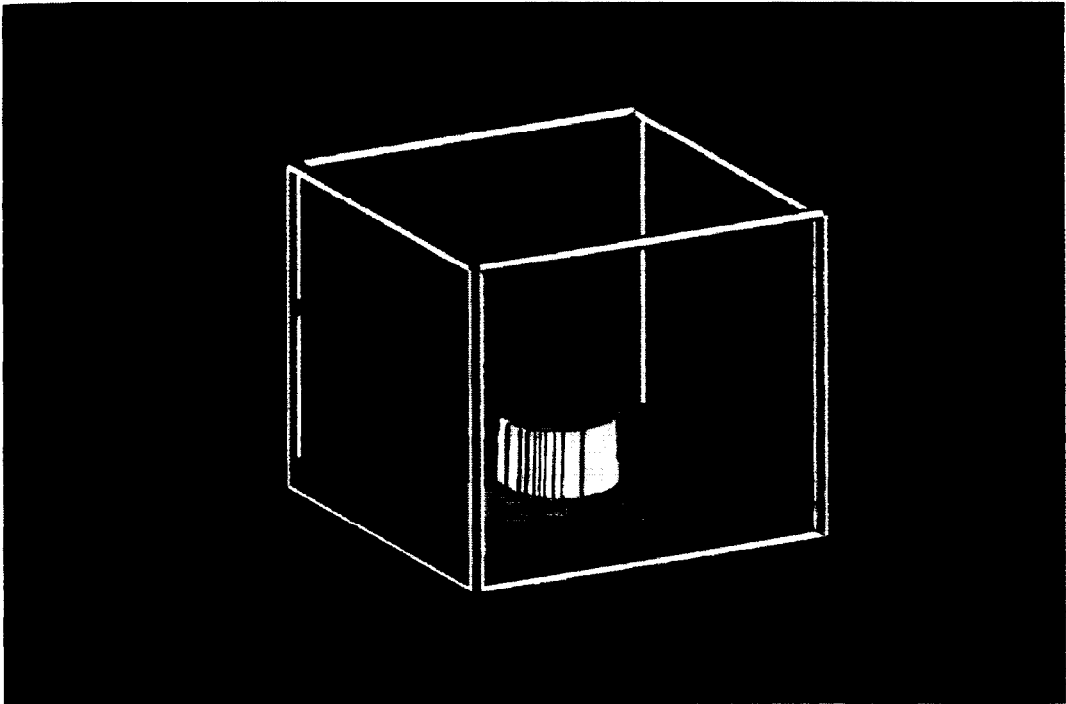


図 2.48: CIP 法による 2 次元保存方程式の検証 (初期プロファイル):

計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, \Delta x = 2/100, \Delta y = 2/100$  において、初期プロファイルを  $f = 1, (\sqrt{x^2 + y^2} < 0.3^2), f = 0(\text{other points})$  として与えた。

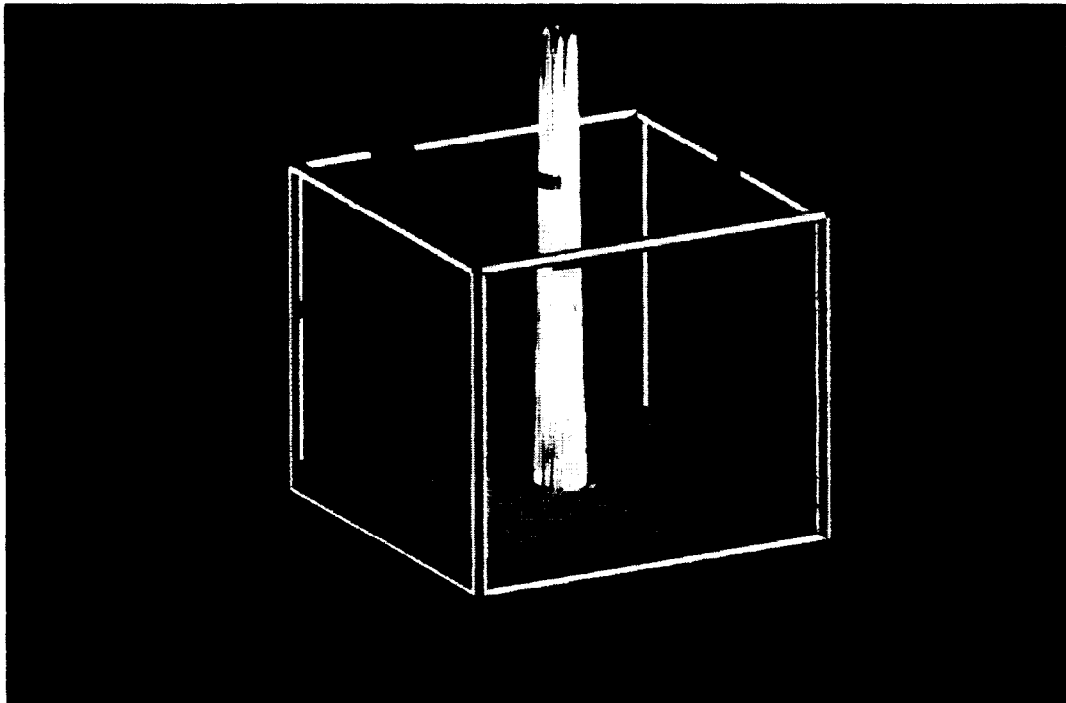


図 2.49: CIP 法 Type-A による 2 次元保存方程式の検証:

図 2.48 で示される初期プロファイルを速度空間  $u(x, y) = -x, v(x, y) = -y$  において、 $\Delta t = 0.004$  を用い  $t = 0.96$  まで CIP 法 Type-A により計算した結果。

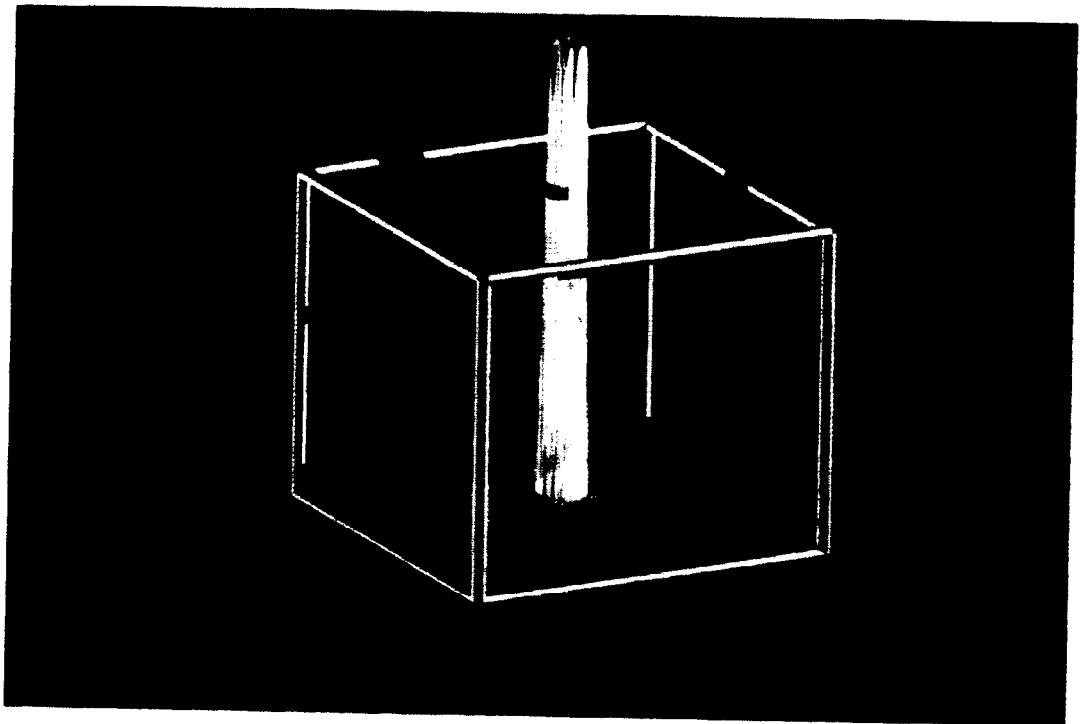


図 2.50: CIP 法 Type-B による 2 次元保存方程式の検証:  
図 2.48 で示される初期プロファイルを速度空間  $u(x, y) = -x, v(x, y) = -y$  において、 $\Delta t = 0.004$  を用い  $t = 0.96$  まで CIP 法 Type-B により計算した結果。

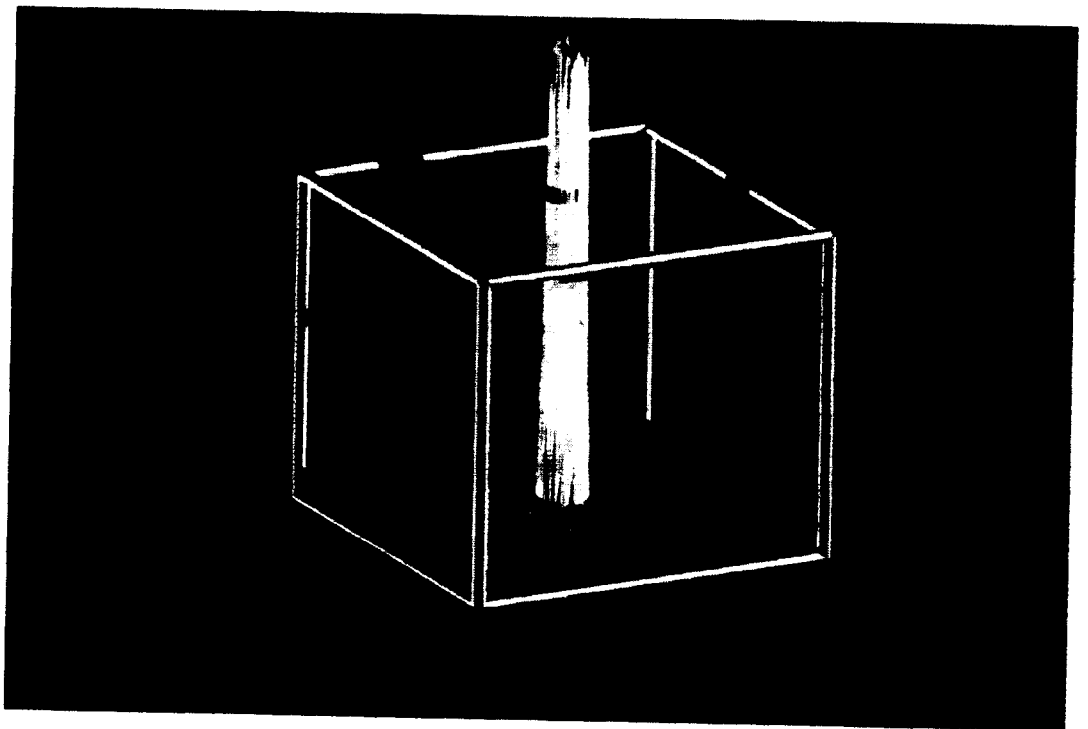


図 2.51: CIP 法 Type-C による 2 次元保存方程式の検証:  
図 2.48 で示される初期プロファイルを速度空間  $u(x, y) = -x, v(x, y) = -y$  において、 $\Delta t = 0.004$  を用い  $t = 0.96$  まで CIP 法 Type-C により計算した結果。

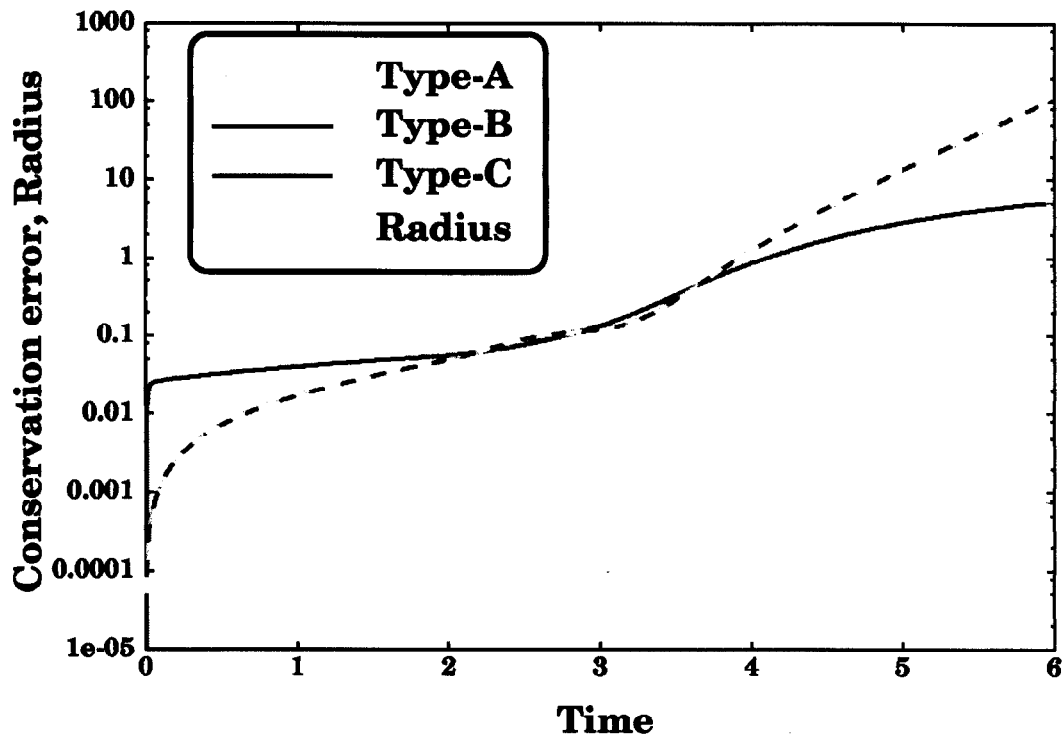


図 2.52: CIP 法による 2 次元保存方程式の集中速度場における保存誤差とプロフィール半径:  
 図 2.48 で示される初期プロフィールを速度空間  $u(x, y) = -x, v(x, y) = -y$  において、 $\Delta t = 0.004$  を用い CIP 法 Type-A, B, C によりそれぞれ計算し、保存誤差を式 (2.107) により見積もった結果。Type-A, Type-B では、保存誤差は 0.1% 程度しか差が見られなかった。

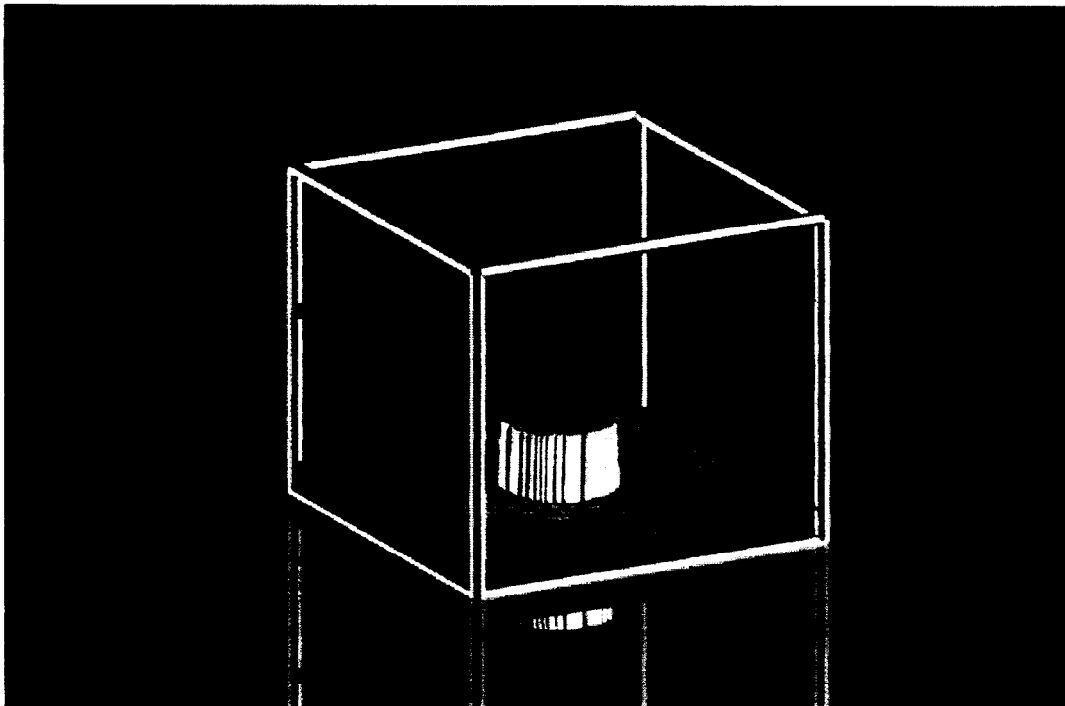


図 2.53: IDO 法による 2 次元保存方程式 (集中速度場) 検証の初期プロフィール:  
 計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, \Delta x = 2/100, \Delta y = 2/100$  において、初期プロフィールを  $f = 1, (\sqrt{x^2 + y^2} < 0.3^2), f = 0(\text{other points})$  として与えた。

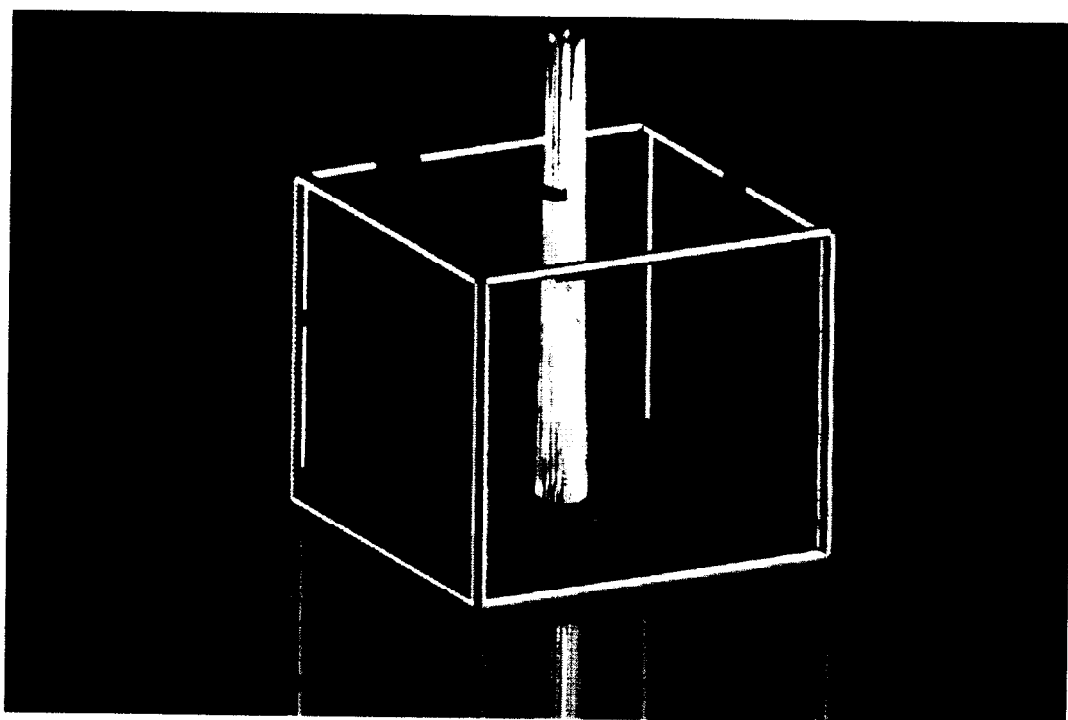


図 2.54: 風上 3 次補間を用いた IDO 法による 2 次元保存方程式 (集中速度場) 検証の計算結果: 計算空間の原点に向かう速度場によりプロファイルが 1 点に向かって集中している様子がわかる。IDO 法を用いることで不連続面が少ないグリッド数で表現可能である。

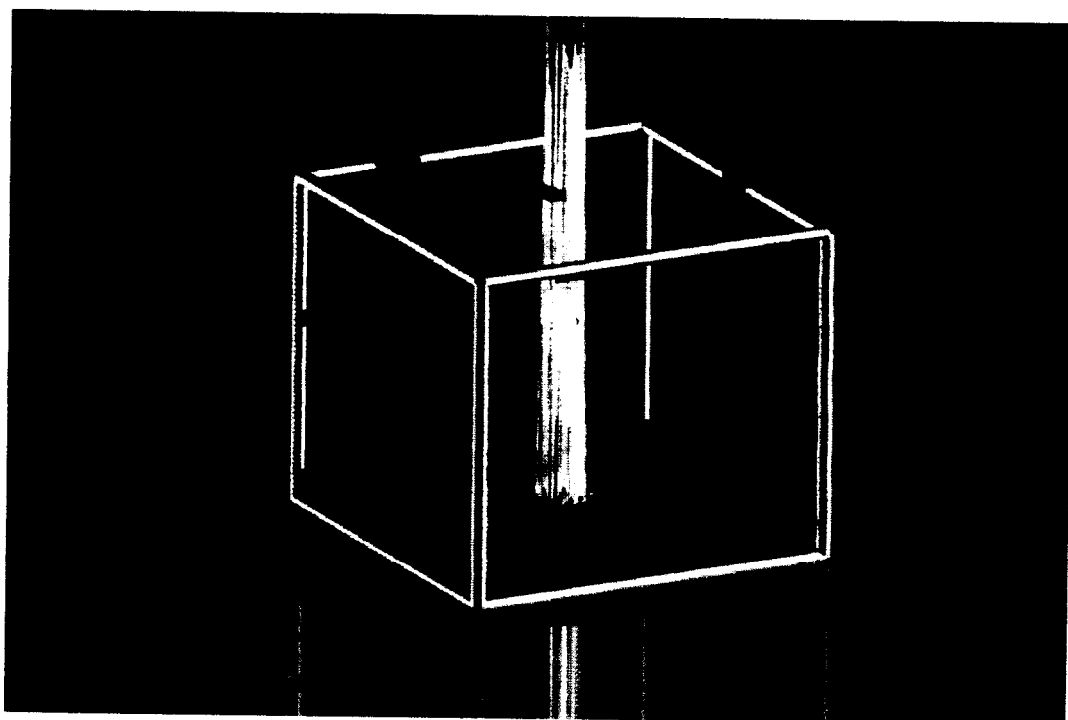


図 2.55: 風上 5 次補間を用いた IDO 法による 2 次元保存方程式 (集中速度場) 検証の計算結果: 計算空間の原点に向かう速度場によりプロファイルが 1 点に向かって集中している様子がわかる。IDO 法を用いることで不連続面が少ないグリッド数で表現可能である。

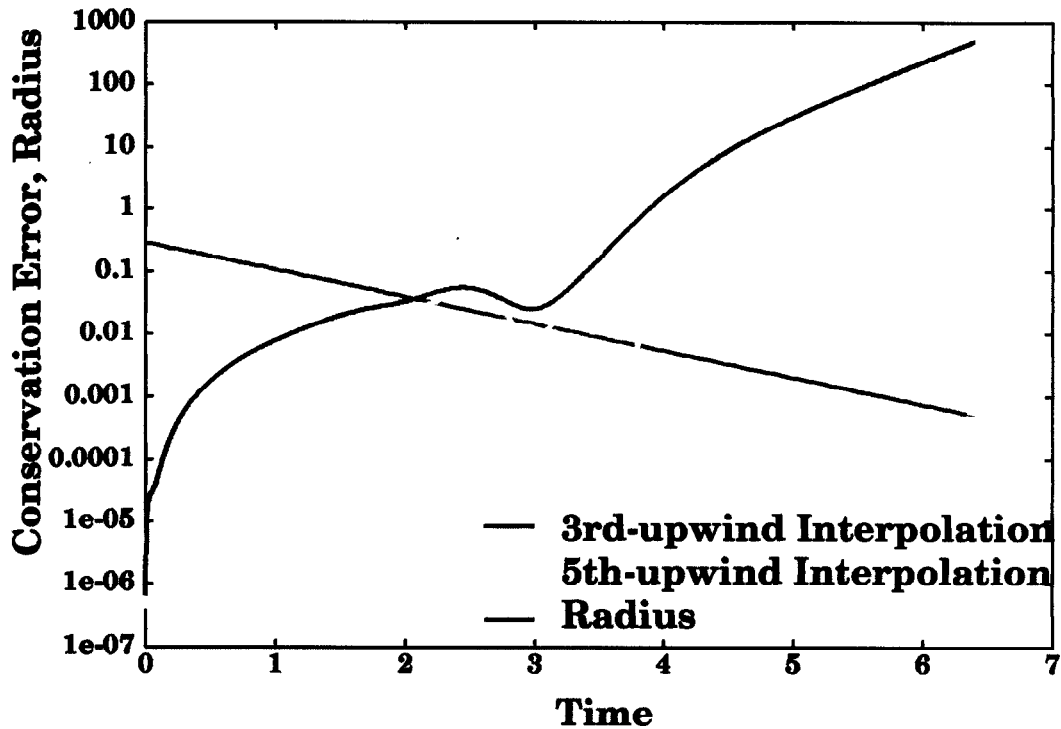


図 2.56: IDO 法による 2 次元保存方程式 (集中速度場) の保存誤差:

プロファイル半径が小さくなるに従って保存誤差が増大する。風上 3 次補間を用いた場合は  $r = 0.5\Delta x$ 、風上 5 次補間を用いた場合は  $r = 0.33\Delta x$  程度で保存誤差が 10% を超える。

式 (2.107) を用いて求めた保存誤差を図 2.56 に示す。初期プロファイル半径 0.3 に対して  $1/10$  の  $r = 0.03 = 1.5\Delta x$  まで ( $t = 2.3$ ) の計算であれば風上 3 次補間、風上 5 次補間から得られる両者の保存誤差は 5% 程度であるが、プロファイル半径が  $r = 0.005 = 0.25\Delta x$  程度になると保存誤差が 100% を超える結果となった。保存誤差を 10% まで許容できると考えると、風上 3 次補間を用いた場合は  $r = 0.01 = 0.5\Delta x$ 、風上 5 次補間を用いた場合は  $r = 0.0067 = 0.33\Delta x$  までの計算が許容範囲に収まる。この結果から、IDO 法による数値流体計算を行う場合、グリッド解像度以下のプロファイルを扱うことは可能であるが、保存性などの計算精度の面からは有限差分法などの計算手法と同様に困難であることを示した。また、図 2.52 に示す CIP 法による数値計算結果と比較した場合、 $t < 3$  では、CIP 法に比べ IDO 法による数値計算結果は若干であるがすぐれていることが理解できる。しかし計算時刻に伴う誤差の増大傾向は近似しており、プロファイルの数値拡散傾向、グリッド解像度に対するプロファイル記述精度が酷似していることを明らかにした。

## 2 次元保存方程式 (回転速度場)

2 次元保存方程式の検証例題として回転速度場中のプロファイルの変形問題について扱う。計算空間中でプロファイルが圧縮、膨張するように回転速度空間中で角速度を変化させることを考える。

計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, \Delta x = 2/100, \Delta y = 2/100$  中で、速度プロファイルを

$$u(x, y) = \frac{y}{1 - A \cos(\theta)}, \quad v(x, y) = \frac{-x}{1 - A \cos(\theta)} \quad (2.108)$$

と与えた。ここで、 $A = 0.2, \cos(\theta) = x/\sqrt{x^2 + y^2}$  とした。

初期プロファイルを図 2.57 に示すように  $f = 1, \text{if}(\sqrt{(x-x_c)^2 + (y-y_c)^2} < 15\Delta x), x_c = 0, y_c = 25\Delta y, f = 0(\text{other points})$  として円形で与えた。プロファイルが計算空間を 1600 ステップで原点を中心に 1 回転するように計算時間刻幅を  $\Delta t = 2\pi/1600$  として与え、時間  $t = 2\pi$  まで CIP 法 Type-A, Type-B, Type-C により計算した結果をそれぞれ図 2.58 - 図 2.60 に示す。

得られた計算結果から、Type-A, Type-B はほぼ同じような形状のプロファイルを示したが、Type-C では Type-A, Type-B に比べ不連続面を少ないグリッド数で表現した結果が得られた。これは Type-C が少ない数値拡散誤差で計算できていることを示している。

また、Type-A, Type-B, Type-C から得られた計算結果から式 (2.107) を用いて見積もった保存誤差を図 2.61 に示した。先の原点集中速度場における 2 次元保存方程式の場合と同様に Type-A, Type-B から得られた保存誤差は 1% 未満の相対誤差を持つものの、ほぼ同値で求められることがわかった。Type-C ではそれらとは異なる結果となったが、オーダーではそれほど変わらない結果となった。

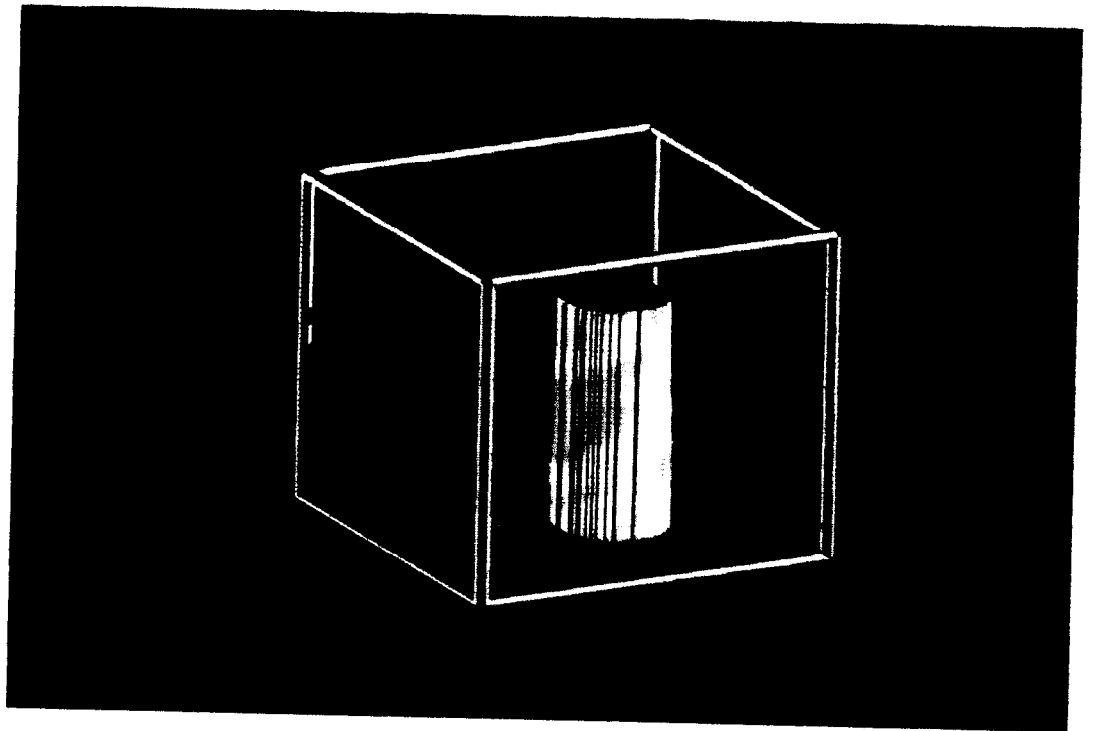


図 2.57: 回転速度場における 2 次元保存方程式のための初期プロファイル:

2 次元保存方程式を計算空間中でプロファイルが圧縮、伸長される速度場で検証を行うため、図のような円形プロファイルを用いる。

次に、IDO 法により移流相計算に風上 3 次補間、風上 5 次補間を用いて時間  $t = 2\pi$  まで計算した結果をそれぞれ図 2.63、図 2.64 に示す。風上 3 次補間、風上 5 次補間を用いた結果は共に不連続面を少ないメッシュ数であらわせており、数値振動なども見られず、高精度な数値解を得ることが出来た。風上 5 次補間を用いた結果は風上 3 次補間を用いた結果に比べ不連続面を少ないグリッド数で表現できている一方、オーバーシュートやアンダーシュートは風上 3 次補間を用いたものに比べ大きくなっていることがわかる。

次に保存誤差の検証を行うため、計算時間刻幅をプロファイルが 3200 ステップで原点を中心に 1 回転するように  $\Delta t = 2\pi/3200$  と与えた。周期境界条件下で移流相計算に風上 3 次補間、風上 5 次補間を用いた計算手法からそれぞれに関して式 (2.107) から得られた保存誤差を図 2.65 に示す。

図 2.65 から、風上補間の次数に関わらず計算時間中ほぼ  $10^{-5}$  程度に安定している様子がわかる。また、CIP 法の結果である図 2.61 と比較すると、保存誤差に関しては  $10^{-2}$  程度 IDO 法の方が優れた結果を得た。

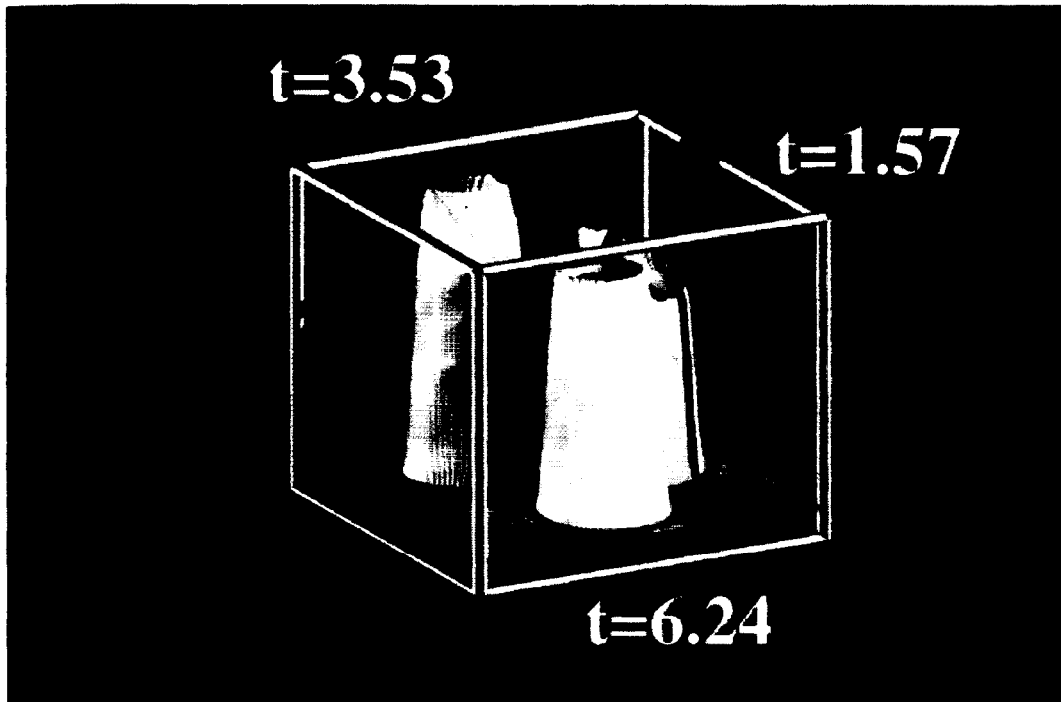


図 2.58: CIP 法 Type-A による回転速度場における 2 次元保存方程式の計算結果:  
 時間が経過するとともにプロファイルが伸長、圧縮される。プロファイルの不連続面はたかだか数  
 グリッドで表現されており、グリッド間補間による数値拡散は非常に少ない。その一方でオーバ  
 シュートやアンダーシュートが少なからず見られる。

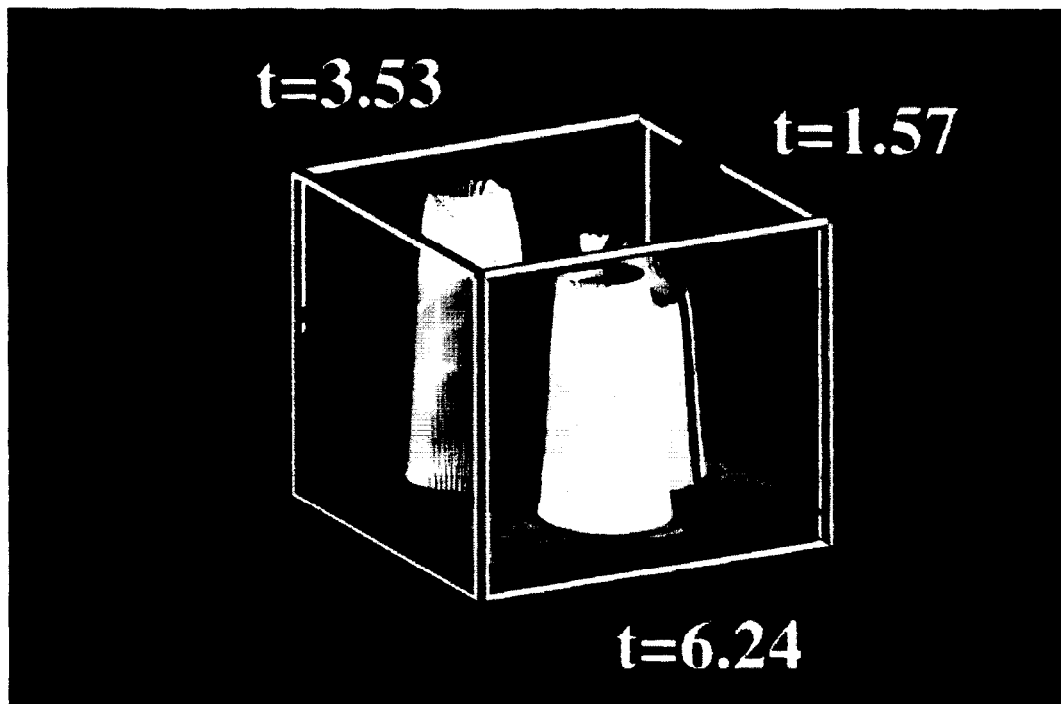


図 2.59: CIP 法 Type-B による回転速度場における 2 次元保存方程式の計算結果:  
 時間が経過するとともにプロファイルが伸長、圧縮される。プロファイルの不連続面はたかだか数  
 グリッドで表現されており、グリッド間補間による数値拡散は非常に少ない。その一方でオーバ  
 シュートやアンダーシュートが少なからず見られる。



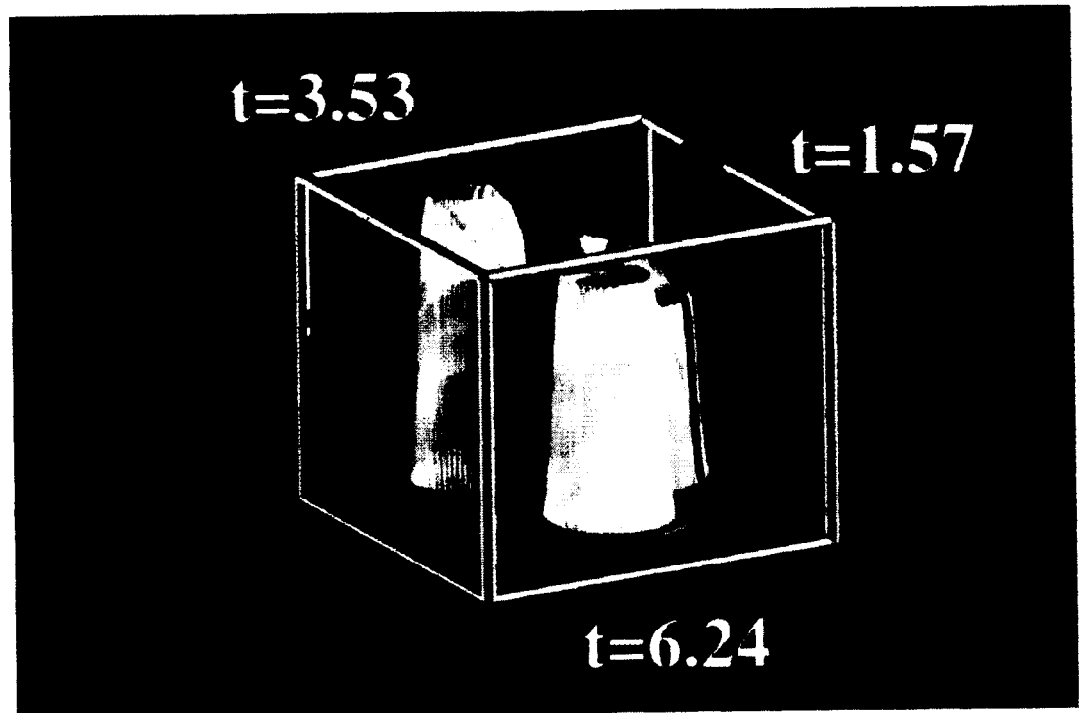


図 2.60: CIP 法 Type-C による回転速度場における 2 次元保存方程式の計算結果:  
 時間が経過するとともにプロファイルが伸長、圧縮される。プロファイルの不連続面はたかだか数  
 グリッドで表現されており、グリッド間補間による数値拡散は非常に少ない。その一方でオーバ  
 シュートやアンダーシュートが少なからず見られる。

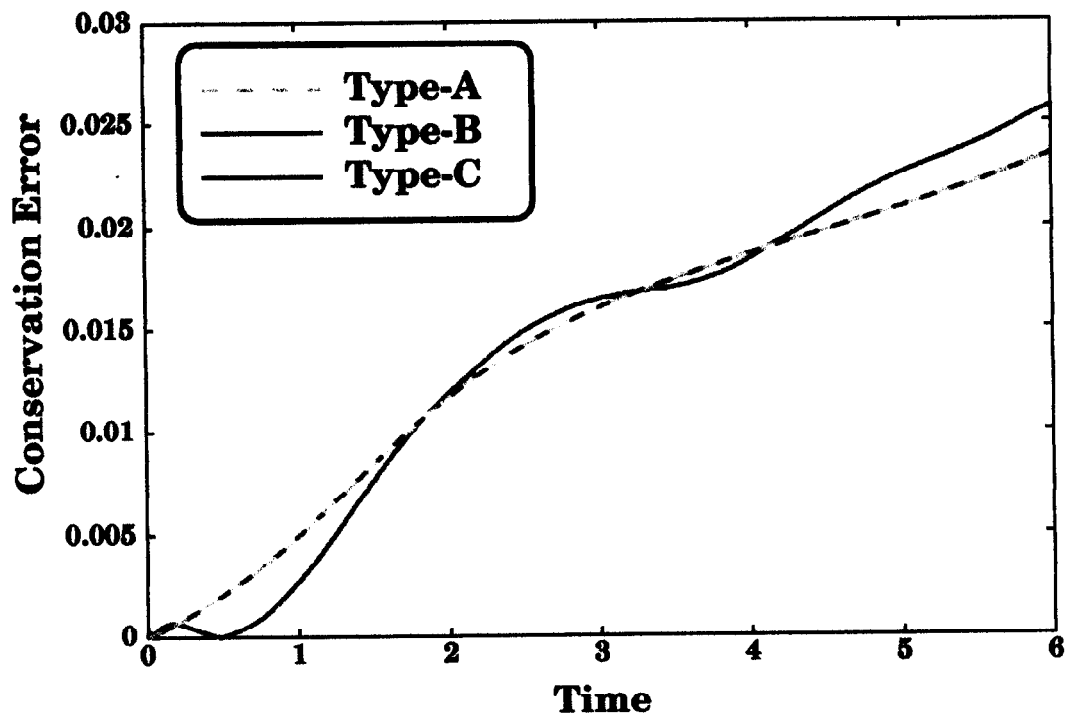


図 2.61: 回転速度場における 2 次元保存方程式の計算結果から見積もられた保存誤差:  
 計算が進行するに従い、Type-A, Type-B は相対的に 1% 未満で差があるもののほぼ同じ程度の保  
 存誤差を示した。Type-C では Type-A, Type-B とはわずかに異なるものの、オーダーで見るとほ  
 ぼ変わらない程度の保存誤差で計算が進行している様子がわかる。

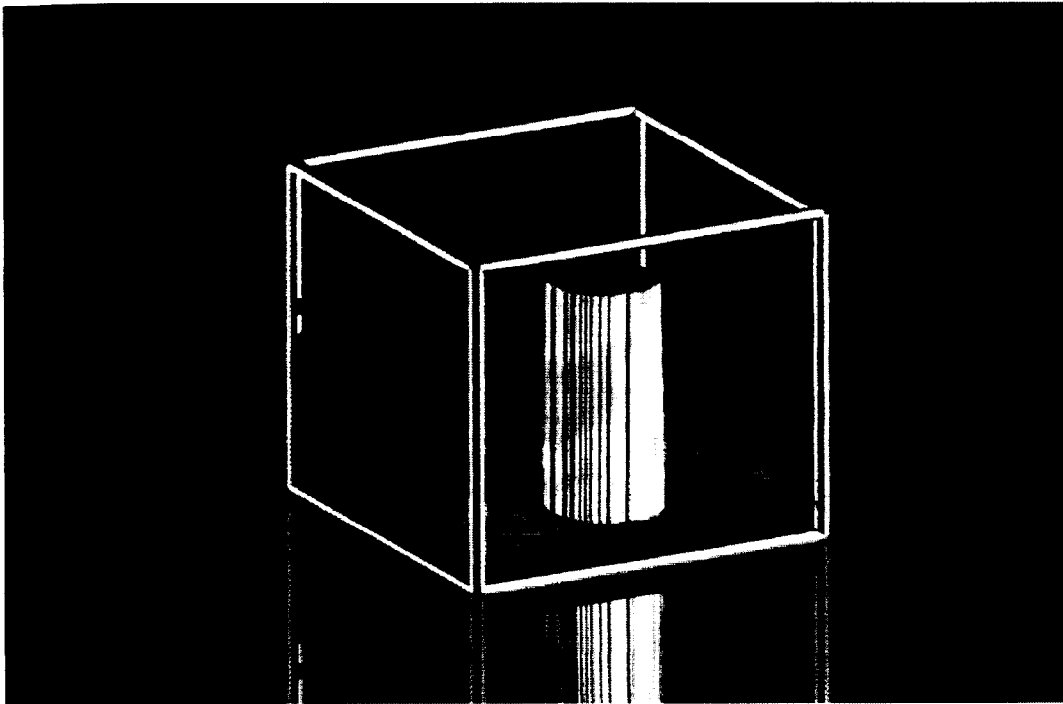


図 2.62: IDO 法による 2 次元保存方程式 (回転速度場) 検証の初期条件:  
2 次元保存方程式を計算空間中でプロファイルが圧縮、伸長される速度場で検証を行うことを目的として、図のような円形プロファイルを用いる。

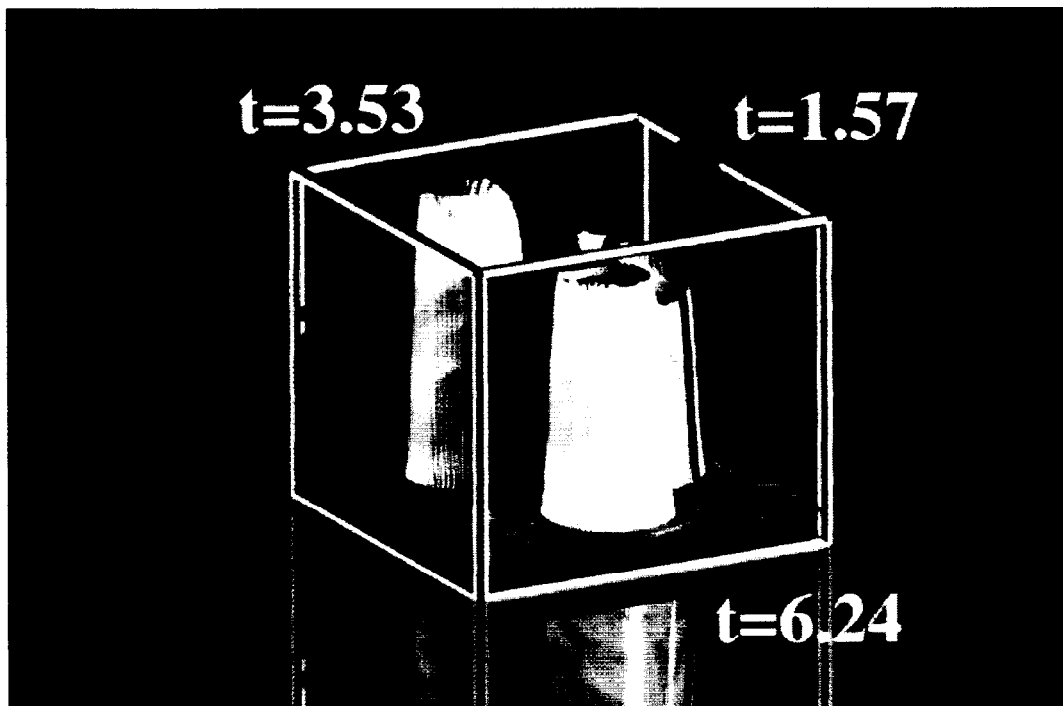


図 2.63: 風上 3 次補間を用いた IDO 法による 2 次元保存方程式 (回転速度場) 検証の計算結果:  
時間が経過するとともにプロファイルが伸長、圧縮される。プロファイルの不連続面はたかだか数グリッドで表現されており、グリッド間補間による数値拡散は非常に少ない。その一方でオーバーシュートやアンダーシュートが少なからず見られる。

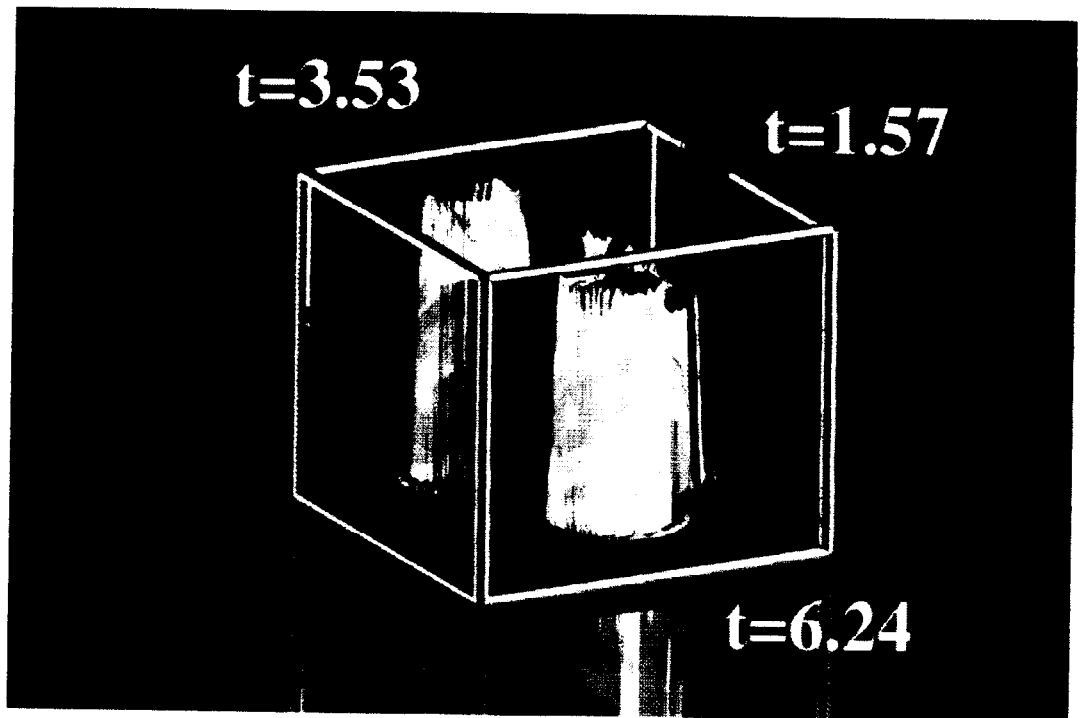


図 2.64: 風上 5 次補間を用いた IDO 法による 2 次元保存方程式 (回転速度場) 検証の計算結果: 時間が経過するとともにプロファイルが伸長、圧縮される。プロファイルの不連続面はたかだか数グリッドで表現されており、グリッド間補間による数値拡散は非常に少ない。その一方でオーバーシュートやアンダーシュートが少なからず見られる。

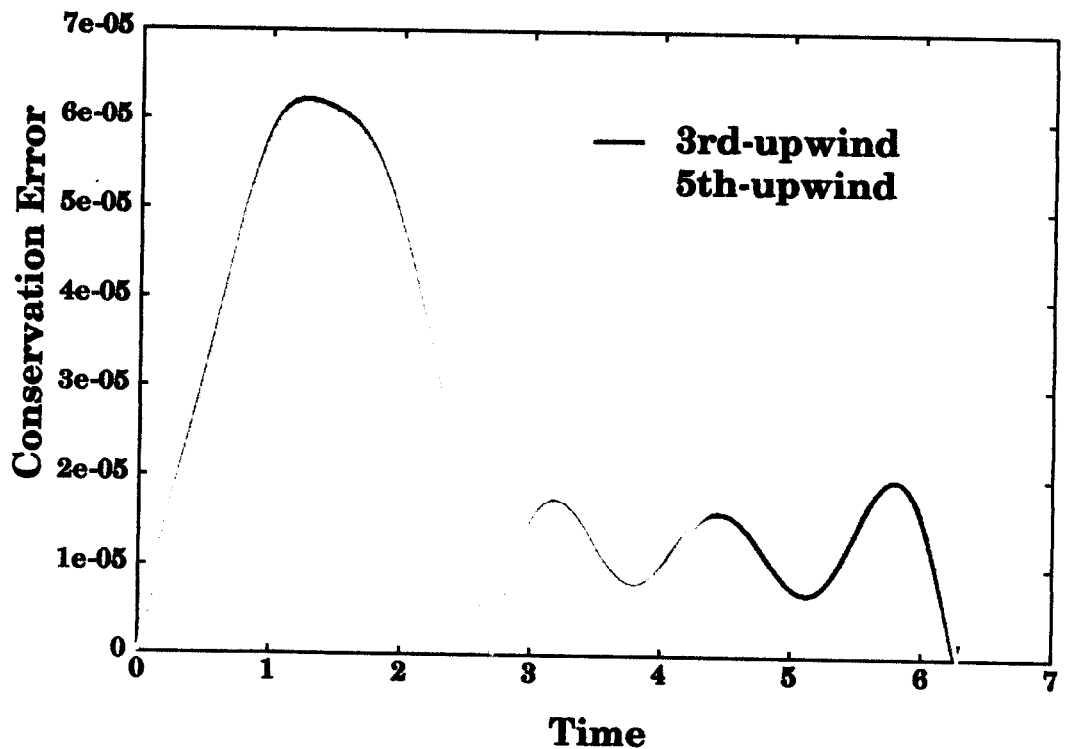


図 2.65: IDO 法による 2 次元保存方程式 (回転速度場) の保存誤差: プロファイルが計算空間の原点を中心に 1 回転するまでの計算時間では  $t \leq 2\pi$  保存誤差は  $10^{-5}$  程度で求められる結果となった。

### 3 次元移流方程式

自然界の現象は全て 3 次元的な構造を持っており、特殊な条件下でのみ 1 次元的、2 次元的にとらえることが可能である。すなわち 2 次元以下での数値計算では何らかの拘束条件が働いていることになる。そこで、より一般的な数値計算を行うためには 3 次元的な構造をとらえるために 3 次元計算を行うことが望ましい。しかし計算機の記憶容量や計算速度などの能力限界により、3 次元計算を行うための手法が確立されても実際に計算を行うことは非常に困難であった。近年、コンピューターの計算能力が劇的に進化し、数十年前のスーパーコンピューターと同等の機能がパーソナルコンピューター (Personal Computer; PC) で備えられていることから 3 次元計算が比較的現実的なものになって来ているとは言え、高精度計算を行うためにはまだ並列計算機やスーパーコンピューターに頼らざるをえない状況にあることは間違いない。

これまでに示して来たように IDO 法を用いることで従来の有限差分法 (Finite Differential Method; FDM) に比べ少ないメッシュ数 (コンピューター記憶容量) で高精度な解を得ることが可能なことから現在の PC 能力程度で 3 次元問題の高精度計算を行える可能性がある。以下で 3 次元移流計算を行い、3 次元問題に対する IDO 法の可能性を検証する。

計算空間  $-0.5 \leq x \leq 0.5, -0.5 \leq y \leq 0.5, -0.5 \leq z \leq 0.5$  において移流速度  $u(x, y, z) = 1, v(x, y, z) = 1, w(x, y, z) = 1$  を用いて  $f(x, y, z) = \cos(k_x x) \cos(k_y y) \cos(k_z z) + 10, k_x = 2\pi, k_y = 2\pi, k_z = 2\pi$ 、で表される初期プロファイルを周期境界条件下でプロファイルが計算空間を完全に一周するまで計算を行い、数値拡散誤差ならびに保存誤差を

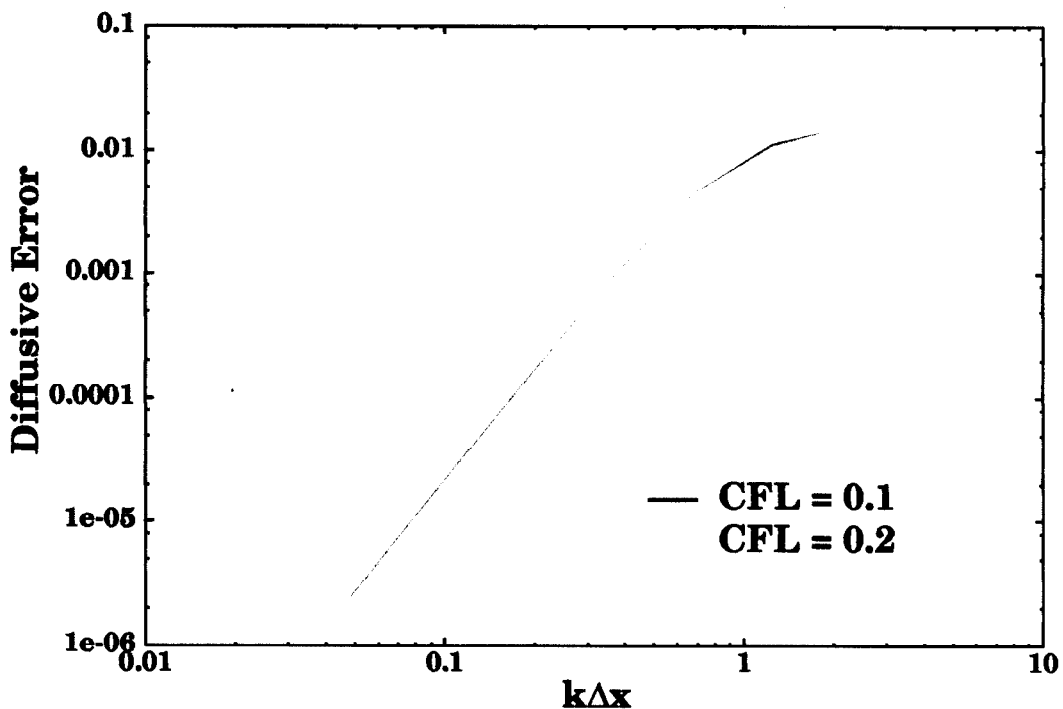


図 2.66: CIP 法 Type-A の 3 次元移流方程式に対する数値拡散誤差:

3 次元 CIP 法 Type-A の波数に対する数値拡散誤差の見積もりから、その精度は  $\Delta x^3$  であることが示された。CFL 数による変動はほとんど見られなかった。

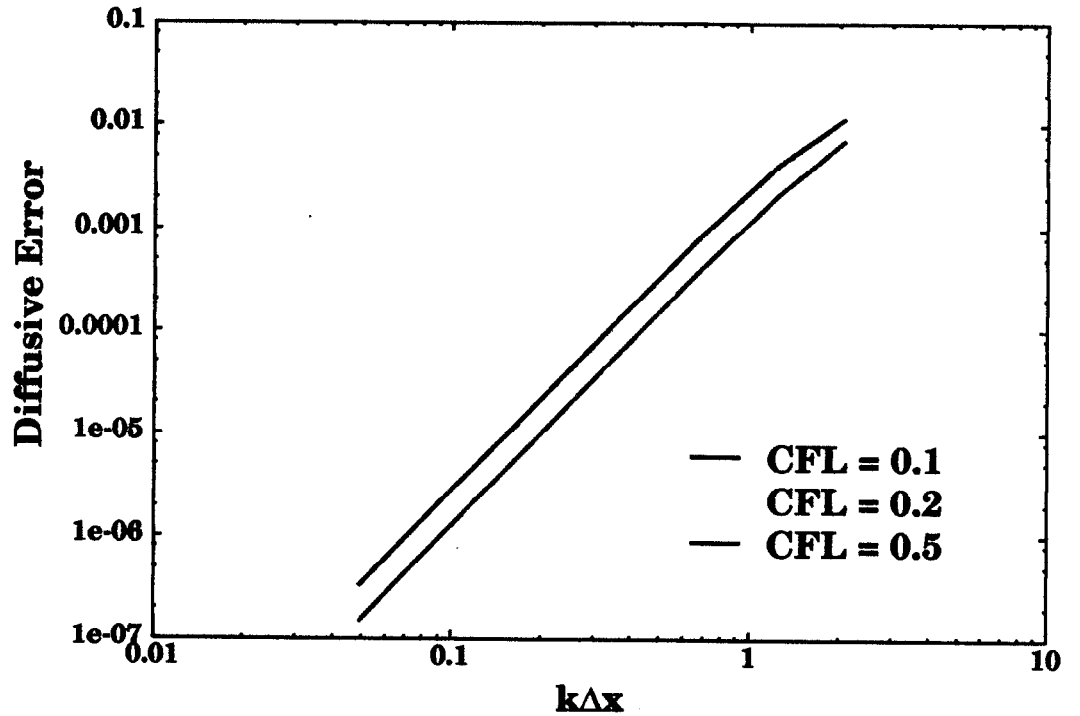


図 2.67: CIP 法 Type-C の 3 次元移流方程式に対する数値拡散誤差:

3 次元 CIP 法 Type-C の波数に対する数値拡散誤差の見積もりから、その精度は  $\Delta x^3$  であることが示された。CFL 数が増大することで、わずかではあるが数値拡散誤差を低減できることがわかる。

$$\text{Conservation Error} = \frac{\sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^S f_{\text{analytic},i,j,k} - \sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^S f_{i,j,k}}{\sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^S (f_{\text{analytic},i,j})} \quad (2.109)$$

$$\text{Diffusive Error} = \sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^S \left| \frac{f_{\text{analytic},i,j,k} - f_{i,j,k}}{f_{\text{analytic},i,j,k}} \right| / \text{MeshNumber} \quad (2.110)$$

を用いて計算する。ここで、 $f_{\text{analytic}}$  は解析解を示している。

3 次元 CIP 法として提案されている Type-A, Type-C に対してそれぞれ CFL 数を変化させ、式 (2.109)、式 (2.110) を用いて見積もった数値拡散誤差をそれぞれ図 2.66 - 図 2.67 に示す。

3 次元 CIP 法においても 1 次元、2 次元の場合と同様に定常等速速度場に対しては保存誤差が丸め誤差程度 ( $< 10^{-14}$ ) で求まることを確認した。Type-A, Type-C の両計算手法から得られた数値拡散誤差から、共に空間解像度に対して  $\Delta x^3$  の精度を示すことを数値実験的に明らかにした。Type-A では、 $CFL = \Delta t / \Delta x > 0.2$  では安定な数値計算が困難であった一方、Type-C を用いることで  $CFL \leq 1$  で安定に計算できることを確認した。

移流方程式に対して、Type-A が CFL 数により数値不安定性を示し、Type-C は安定に扱える理由は、先に示した空間補間誤差を考慮することで理解できる。3 次元 CIP 法 Type-A による空間補間では計算格子空間に含まれるプロファイルが持つ波数によっては、計算対象格子点近傍以外の領域で補間された値の持つ誤差は 100% を超える結果となった。本節で扱ったプロファイルは不連続面を持つため、離散フーリエ変換すると無限の波数を持つ三角関数により表現されると言える。CFL 数が増大するに従い移流原点

位置は計算対象点から遠ざかることを考えると補間精度の低下が容易に予想され、そのために数値不安定性が生じたと考えられる。その一方で、3次元 CIP 法 Type-C による補間手法では、Type-A による補間結果に見られるような誤差の計算格子点内に置ける局所分布が存在しないため、移流原点が格子点内に含まれている限りでは ( $CFL < 1$ ) 高精度で移流計算が可能であると言える。

IDO 法を用いた計算結果を以下に示す。保存誤差は風上 3 次補間、風上 5 次補間を用いた数値計算結果ともに丸め誤差程度 ( $\leq 10^{-14}$ ) で求められることを確認した。図 2.68 に示すように風上 3 次補間、風上 5 次補間を用いた計算結果から数値拡散誤差はともに  $\Delta x^3$  の精度を示すことを明らかにした。また、風上 5 次補間を用いることで風上 3 次補間を用いた計算結果に対して低数値拡散誤差での移流計算が可能であることを示した。ここで風上 5 次補間から得られた空間精度が  $\Delta x^3$  であった理由として時間前進に用いるテーラー展開の次数を 3 次に制限したことが挙げられる。1 次元移流方程式で示したように、風上 5 次補間を用い、テーラー展開の次数を 5 次までとることにより、空間精度を  $\Delta x^5$  まで上げることが可能であると考えられる。また、風上 3 次補間、風上 5 次補間を用いた数値計算では、 $CFL$  数が 0.2 を越えると両者ともに数値不安定性が生じた。この結果から、数値計算対象となる空間の次元数が増大するに従ってより  $CFL$  数に対する条件が厳しくなり、大きな  $\Delta t$  を用いた計算が CIP 法に比べ困難になることを明らかにした。

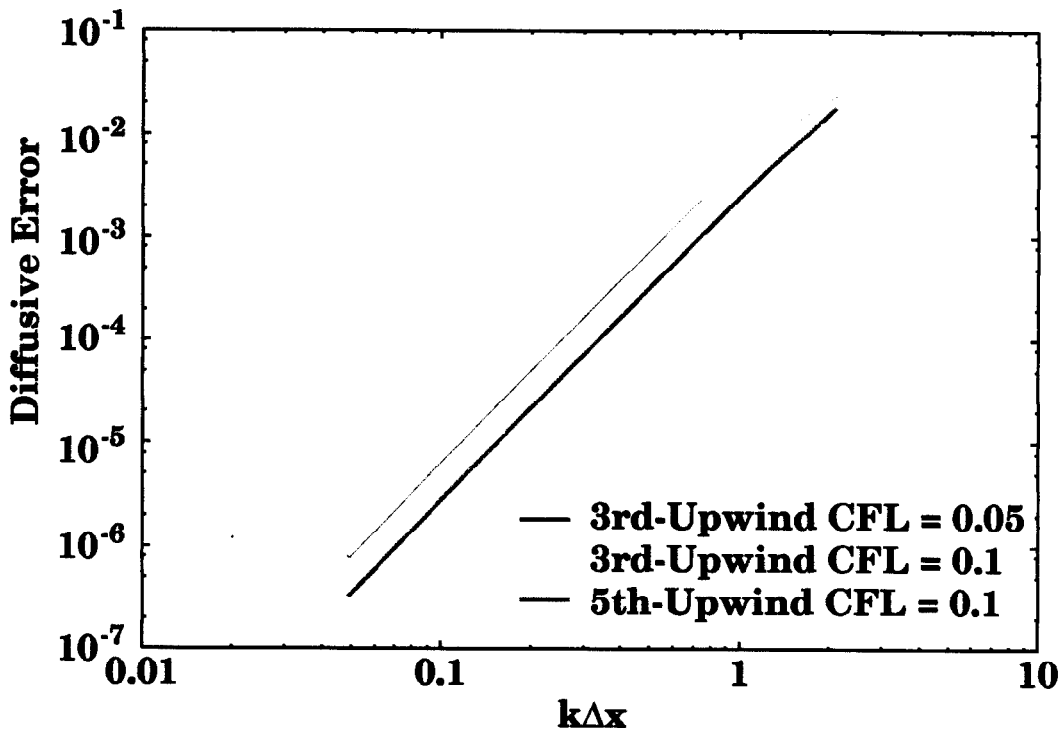


図 2.68: IDO 法による 3 次元移流方程式の数値拡散誤差

風上 3 次補間、風上 5 次補間を用いて 3 次元移流方程式を扱った数値計算結果から得られた数値拡散誤差は、 $\Delta x^3$  の精度を示した。

IDO 法では、計算対象グリッドを中心として、隣接するグリッド上の情報のみを用いた計算を行うため、CIP 法に比べて  $CFL$  数に対する数値安定範囲が低下することが考えられる。特に CIP 法 Type-C との比較を行った場合、Type-C では、計算対象点における風上方向の計算セルに含まれる全 8 グリッド点上の情報から補間関数を形成するのに対し、IDO 法では、風上 3 次補間を考えた場合、風上方向に対して隣接する 3 点の情報のみにより全ての高次空間微係数を計算することから、1 次元的な補間に関しては  $CFL$  数が大きくとれるものの、多次元的な補間を行った場合計算対象点から遠くなる領域 ( $CFL$  数が増大する方向) においては補間精度が低下する結果となる。このことから、CIP 法に比べ IDO 法は  $CFL$  数に対す

る安定面で不利であることが説明できる。

## 2.4 まとめ

### IDO 法で用いられる補間関数の高次微係数精度

IDO 法はグリッド間プロファイルが風上補間ならびに中心補間により満足に記述できることを前提に、高次空間微係数を補間関数を直接微分することにより見積もることを手続きの 1 つとして用いていることから、グリッド間プロファイルを三角関数により与え、その波長を変化させることでグリッド間プロファイルの波数 ( $k\Delta x$ ) に対する高次微係数誤差を見積もった。

3 次風上補間関数、5 次風上補間関数、5 次中心補間関数から得られるそれぞれの最高次微係数までの計算誤差を見積もったところ、得られる微係数の中でも低次のもの程高精度で計算できることを示した。また、得られた空間解像度に対する計算誤差は  $\sin$  関数、 $\cos$  関数をそれぞれテーラー展開した式を用いることで充分議論できることを示した。ただし、グリッド間プロファイルの波長が長くなるに従って微係数計算誤差が増大する傾向が数値実験的に観測された。理論的な高次微係数の値が  $10^{-17}$  程度を下回ると丸め誤差の影響により補間関数から得られる高次微係数計算誤差が増大する結果を得た。現象のスケールがわかる場合は  $k\Delta x > 0.01$  程度でグリッドを作成することが望ましいと言える。

### IDO 法の計算手続きと CIP 法の計算手続きとの比較

定常一様速度場における 1 次元移流方程式を IDO 法により扱う場合、その計算アルゴリズムが CIP 法のものとは完全に一致することが知られている。このことから、CIP 法の概念に基づき 1 次元補間関数の空間補間精度を検証する事で CFL 数に対する数値誤差の影響を検証した。精度検証結果から、1 次元移流計算に対して CFL 数が 0.1 ~ 0.9 での数値計算を行う場合は数値誤差はほぼ同程度で得られ、0.1 未満であれば大きく補間精度を改善できることを示した。また、CIP 法の多次元補間手法に付いて空間補間精度を検証することで、これまで明らかにされていなかった多次元 CIP 法の CFL 数に対する数値不安定性の原因を明らかにするとともに、多次元 IDO で大きな CFL 数をとれない理由として補間関数形成領域が計算対象格子点近傍に偏っていることを示した。

### 単純な偏微分方程式を用いた基礎精度検証

IDO 法に関して 1 次元質量保存式、2 次元移流方程式を CIP 法との比較を行い精度検証を行った。1 次元保存方程式の数値解から得られる保存誤差より、CIP 法の非移流相計算に比べ高精度で非移流相を扱っている IDO 法では CIP 法に比べ、オーダー以上少ない数値計算が可能であることを示した。また、風上 5 次補間式を用いることで 2 次元移流問題から CIP 法よりも数値拡散誤差の少ない数値解が得られることを示した。保存誤差に関しても IDO 法は CIP 法に比べ高精度な計算を行える結果を得た。

## 第3章

# 圧縮性流体問題への適用

基礎的精度検証の結果、有限差分法 (Finite Difference Method; FDM) に比べ高精度な数値計算が局所補間微分オペレータ法 (Interpolated Differential Operator; IDO) 法を用いることで可能であることを示した。本章では IDO 法をベンチマーク問題として良く知られる例題に適用し、流体問題への適用を検証する。その後、現在研究段階にある超音速混合流問題へ適用を行い現象に対する考察を行う。

### 3.1 1次元衝撃波管問題 (1次元 Riemann 問題)

1次元衝撃波管問題は1次元衝撃波管内の流体の挙動を記述するベンチマーク問題として良く引用されている。IDO 法は非保存系スキームであるため衝撃波を扱うためには衝撃波位置において Rankine-Hugoniot の関係式から得られる人工粘性を適切に加える必要がある。1次元衝撃波管問題を扱うことで圧縮性流体問題 (Compressible Fluid Dynamics) への IDO 法の適用性に加え、人工粘性の IDO 法における効果を検討することも目的とする。

1次元圧縮性流体の支配方程式は密度  $\rho$ 、速度  $u$ 、内部エネルギー  $E$ 、圧力  $P$  に関する以下の質量保存則、運動量保存則、エネルギー保存則からなる。

$$\rho_t = -u\rho_x - u_x\rho \quad (3.1)$$

$$u_t = -uu_x - \frac{P_x}{\rho} \quad (3.2)$$

$$E_t = -uE_x - (\gamma - 1)u_x E \quad (3.3)$$

ここで、添字の  $t, x$  はそれぞれの変数による偏微分を表す ( $f_t = \partial f / \partial t$ )。  $\gamma$  は比熱比を表し、数値計算では 1.4 とした。

IDO 法は、非保存系で記述される基礎方程式により数値流体計算を行うため、衝撃波問題などの圧縮性流体問題を扱うためには基礎方程式に人工粘性を考慮する必要がある。人工粘性を用いた記述を行うと、上式は

$$\rho_t = -u\rho_x - u_x\rho \quad (3.4)$$

$$u_t = -uu_x - \frac{P_x + q_x}{\rho} \quad (3.5)$$

$$E_t = -uE_x - \frac{u_x(P + q)}{\rho} \quad (3.6)$$

と表せる。

#### 3.1.1 人工粘性に対する考察

有限差分法などで人工粘性を導入する場合、von Neumann 型、Landshoff 型人工粘性 (式 (C.19)、式 (C.20)) が良く用いられている [13]。しかし、有限差分法ではこれらの人工粘性を前後の計算グリッド点



上の物理量を用いた差分形で用いていることから、微係数を導入した IDO 法への適用が可能であることは明確に示されていない。そこで、ここでは Landshoff 型人工粘性を IDO 法による圧縮性流体問題に適用し、その効果について議論する。IDO 法では物理量ならびに空間微係数に対してテーラー展開に基づいた時間発展を行うため、基礎方程式を空間に関して微分した関係式を求める必要がある。そのため人工粘性

$$q = c_L \rho \Delta x a \left| \frac{\partial \hat{x}}{\partial x} \right| \quad (3.7)$$

に関する高次微係数を各物理量の高次空間微係数を用いて表現することになることから、有限差分法で用いられる手法とは大きくその性質が変わることが予想される。また、人工粘性は圧縮領域 ( $\partial u / \partial x < 0$ ) にのみ適用される。

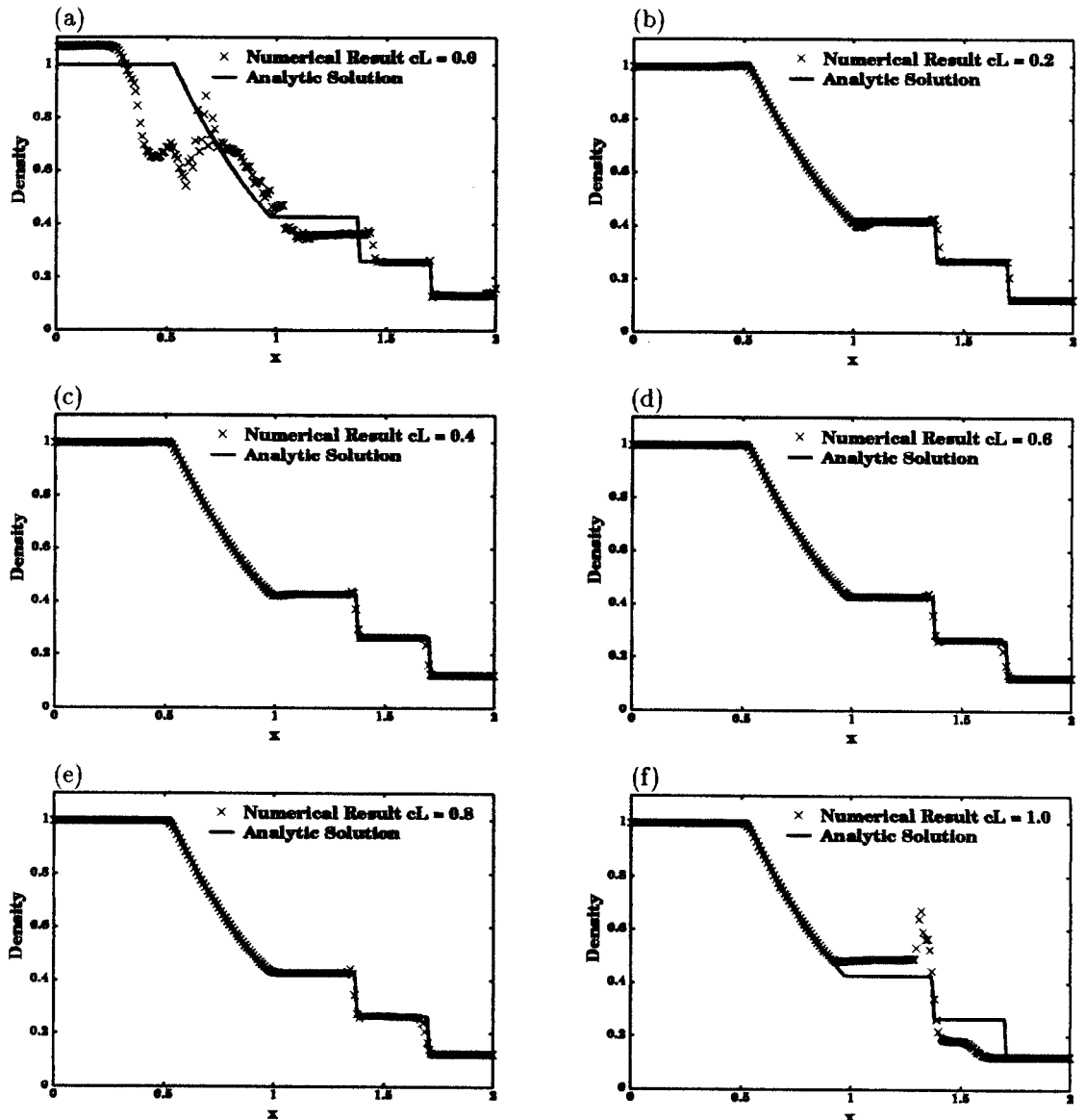


図 3.1: 1 次元衝撃波管問題の人工粘性に対する応答 (差分判定)

差分により圧縮領域を特定し、人工粘性を加える。(a) - (f) では人工粘性の係数  $c_L$  を 0, 0.2, 0.4, 0.6, 0.8, 1.0 としている。

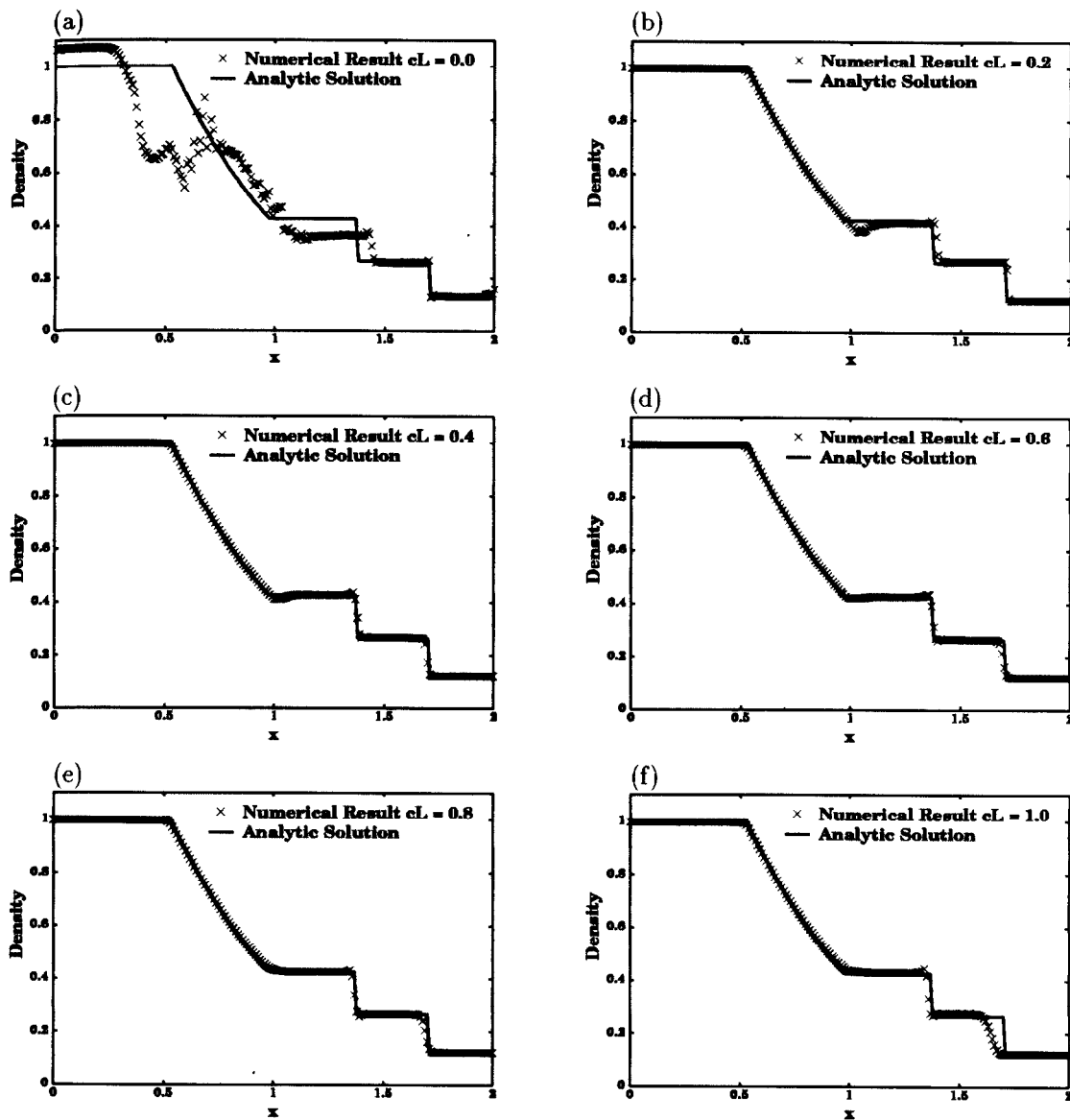


図 3.2: 1次元衝撃波管問題の人工粘性に対する応答 (微係数判定)

グリッド点に定義されている速度の空間微係数により圧縮領域を特定し、人工粘性を加える。(a) - (f) では人工粘性の係数  $c_L$  を 0, 0.2, 0.4, 0.6, 0.8, 1.0 としている。

IDO 法では速度の 1 階空間微係数  $u_x$  を従属変数としてグリッド点毎に定義していることから圧縮領域の判定に直接、速度の 1 階空間微係数を使用する手法と、有限差分法で用いられているように前後グリッド上の移流速度の差分から圧縮領域を特定する手法が考えられる。そこで、両者に対して人工粘性の係数  $c_L$  を変化させながら、計算結果から得られたプロファイルを検討する。有限差分法では、 $c_L \simeq 1$  で用いられる。

計算空間  $0 \leq x \leq 2, \Delta x = 2/200$  において、初期密度、初期圧力をそれぞれ  $\rho = 1, P = 1 (x \leq 1), \rho = 0.125, P = 0.1 (x > 1)$  と与え、 $CFL = \Delta t / \Delta x = 0.1$  を用いて  $t = 0.4$  まで計算した結果を図 3.1 (a) - (f)、図 3.2 (a) - (f) に示す。ここで、 $CFL$  数は移流速度が空間で変動することから、代表速度 1 を用いて定義した。また、高次空間微係数を計算するための補間関数は風上 5 次補間、中心 5 次補間を用いる。

差分形での圧縮領域判定 (図 3.1 (a) - (f)) は計算対象グリッド ( $i$ ) 上の移流速度ならびにその前後グリッド ( $i+1, i-1$ ) における移流速度との差分による判定 ( $u_i - u_{i-1}, u_{i+1} - u_i$ ) ならびに前後グリッドの移流速

度の差分 ( $u_{i+1} - u_{i-1}$ ) を用いた判定を併用して行った。その一方、微係数による圧縮領域判定 (図 3.2 (a) - (f)) は、対象計算グリッド ( $i$ ) 上の微係数 ( $u_x|_i$ ) に加え、前後の計算グリッド上の移流速度の空間微係数 ( $u_x|_{i-1}, u_x|_{i+1}$ ) 正負判定を用いた。いずれの場合においても人工粘性を含めない場合 ( $c_L = 0.0$ ) では計算が不安定になっている様子を数値実験的に示した。また、微係数により判定を行った場合では  $c_L = 1.0$  で、衝撃波面が解析解に比べて遅れる傾向が見られ、差分による判定でも  $c_L = 1.0$  では数値不安定が見られる結果を得た。

解析解と比較すると、 $c_L \leq 0.4$  では膨張波面と接触不連続面 (Contact discontinuity) の接合部 ( $x = 1$ ) においてアンダーシュートが生じる結果を得た。 $c_L$  を大きくするに従い、衝撃波面を記述するために要するグリッド点数が増大する傾向が得られた。これは、式 (3.5) に式 (3.7) を変形した式

$$q = -c_L \rho \Delta x a \frac{\partial u}{\partial x} \quad (3.8)$$

を代入することで、

$$u_t = -u u_x - \frac{P_x}{\rho} + c_L \Delta x a \frac{\partial^2 u}{\partial x^2} \quad (3.9)$$

と書き表せることから、人工粘性が数値的な拡散項として作用していることを考えると容易に説明することが出来る。ここで、圧縮領域では  $\partial u / \partial x < 0$  であることを利用した。

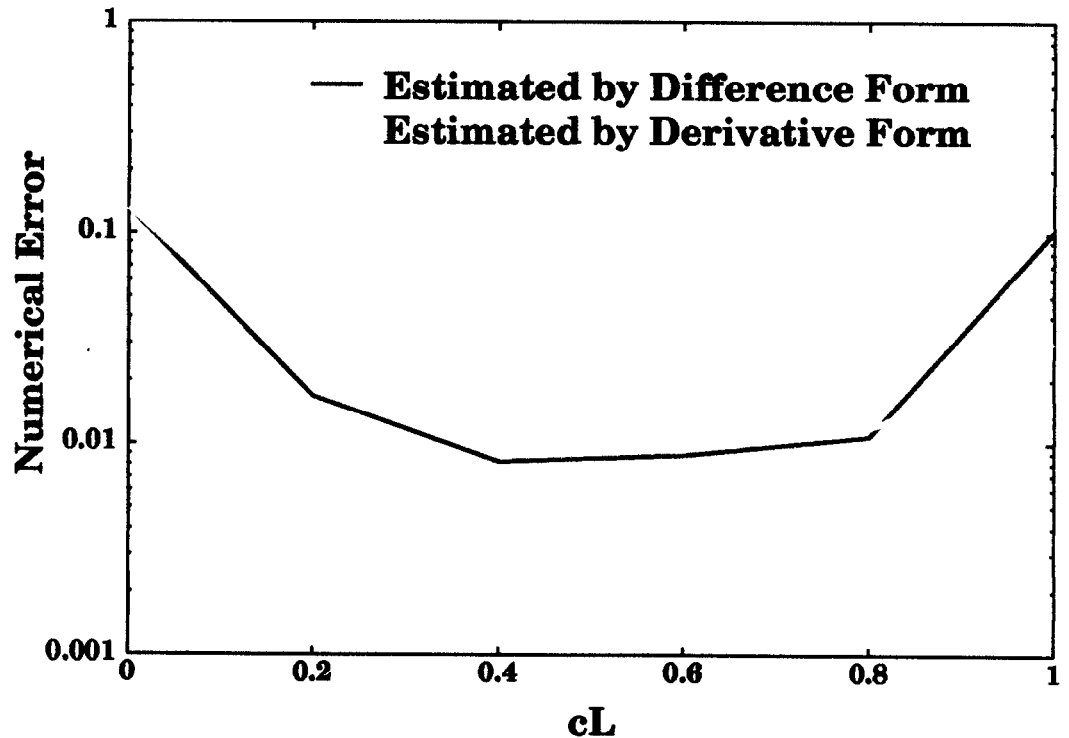


図 3.3: 1 次元衝撃波管問題の人工粘性に対する数値誤差

圧縮領域の判定を差分により行ったもの (赤線) とグリッド上に定義された速度の空間微係数を用いたもの (橙線) とでは数値誤差にほとんど差が見られない。

得られた計算結果の解析解からの誤差を

$$\text{Numerical Error} = \sum_i \left| \frac{f_{\text{analytic},i} - f_{\text{calc},i}}{f_{\text{analytic},i}} \right| / \text{Mesh Number} \quad (3.10)$$

により求めたものを図 3.3 に示す。ここで、 $f_{\text{analytic}}$  は解析解を示し、 $f_{\text{calc}}$  は計算結果を示す。圧縮領域の判定を差分形で行った場合とグリッド上に定義されている速度の空間微係数を用いた場合を比較しても数値誤差にほとんど差が生じないことを計算結果から明らかにした。以降は安定性のメリットから、微係数形式による圧縮領域の判定を行い、 $0.4 \leq c_L \leq 0.8$  を用いる。

### 3.1.2 空間解像度、補間関数に対する計算精度の考察

1次元衝撃波管問題は解析解が存在することから、空間精度、補間精度に対する計算精度の検討を行うことは有効である。グリッド数、風上補間関数、CFL 数を変化させ、流体問題に対する IDO 法の基礎的な精度を以下で検討する。

計算空間  $0 \leq x \leq 2$  において、初期密度、圧力をそれぞれ  $\rho = 1, P = 1(x \leq 1)\rho = 0.125, P = 0.1(x > 1)$  と与える。CFL 数 ( $\Delta t/\Delta x$ ) を 0.1 として空間解像度  $\Delta x$  を変化した計算結果を図 3.4 に示す。また、CFL 数、空間解像度をそれぞれ 0.025, 0.05, 0.1, 1/10, 1/20, 1/40, 1/80, 1/160, 1/320, 1/640, 1/1280 と変化させ、 $t = 0.4$  まで計算した結果に対して式 (3.10) により見積もった数値誤差を図 3.5 に示す。人工粘性は圧縮領域に対して式 (3.7) で示される Landshoff 型の人工粘性を  $c_L = 0.8$  で導入した。図 3.5 から、グリッド数を増大し計算空間に対する解像度を増大させた場合 IDO 法では数値誤差が増大することが明らかになった。その理由を明らかにするため、風上 5 次補間を用い、 $\Delta x = 2/1280$  での計算結果と解析解を図 3.6 に示す。

グリッドの解像度を増大させた場合、膨張波面において数値振動が生じていることがわかる。膨張波領域では移流速度が座標軸の負方向に対して正である、つまり流体は向かって右側に流れているにもかかわらず、膨張波面は向かって左側に進行する。すなわち、両方向に音波が進行することから膨張領域では移流速度を重視した風上補間関数から得られるプロファイル記述精度が他の領域と比較して低下することが考えられる。

そこで、IDO 法で膨張波面を扱う際に風上補間を用いることで生じる数値振動を低減するために数値粘性の導入を考える。1次元衝撃波管問題のように衝撃波が生じるような問題ではレイノルズ数が非常に大きいと考え、式 (3.5) に示す非粘性 Navier-Stokes 方程式を利用して数値計算を行うが、数値粘性を付加するためにあえて

$$u_t = -uu_x - \frac{P_x}{\rho} + \kappa u_{xx} \quad (3.11)$$

で表される粘性を考慮した Navier-Stokes 方程式を利用した数値計算を行う。ここで、 $\kappa$  は数値粘性係数として作用する。

数値粘性の妥当性を検討するために、計算空間  $0 \leq x \leq 2, \Delta x = 2/1280$  において、初期密度、圧力をそれぞれ  $\rho = 1, P = 1(x \leq 1)\rho = 0.125, P = 0.1(x > 1)$  と与え、CFL 数 ( $\Delta t/\Delta x = 0.05$ ) を用いて数値粘性  $\kappa = 10^{-5}, 10^{-4}, 10^{-3}$  を適用した計算結果を図 3.7 (a)、(b)、(c) にそれぞれ示す。数値粘性  $\kappa = 10^{-3}$  を付加した計算結果は解析解に比べ数値拡散の大きな結果となり、衝撃波面位置も解析解と異なる結果となった。 $\kappa = 10^{-4}$  では比較的良好な結果を示したが、 $\kappa = 10^{-5}$  を用いた場合では数値粘性としての効果を確認することが出来なかった。

それぞれの  $\kappa$  に対する解析解からの数値拡散誤差を式 (3.10) を用いて計算した結果を図 3.8 に示した。 $\kappa = 10^{-4}$  以下で数値粘性を適用した場合は数値粘性を適用しない結果と変わらない数値拡散誤差で解を求めることが可能であることを示した。これにより、数値振動や数値不安定性などが生じる問題に対しては流体现象を大きく変えない程度の数値粘性を付加することで対処することが可能であることを示した。

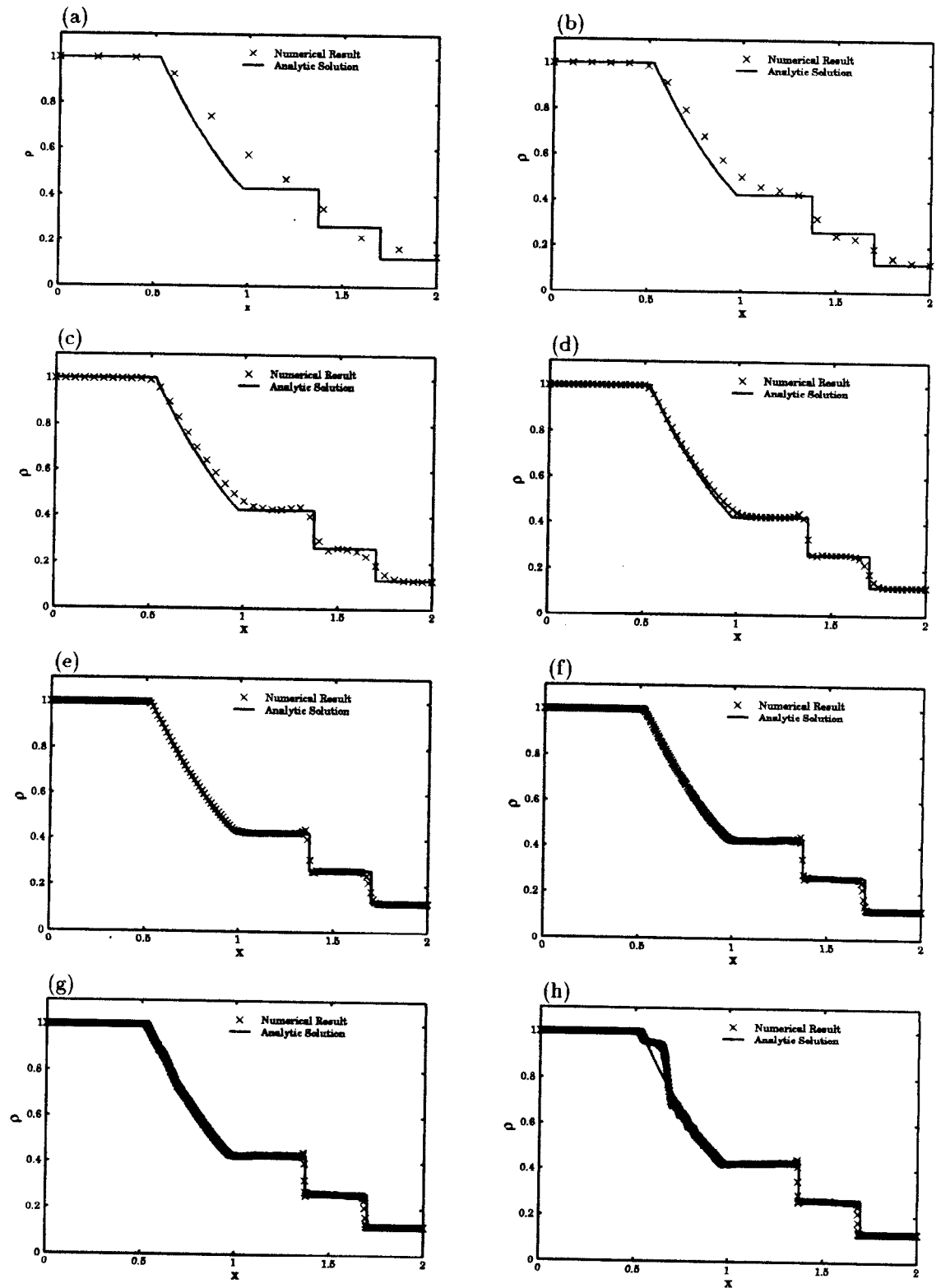
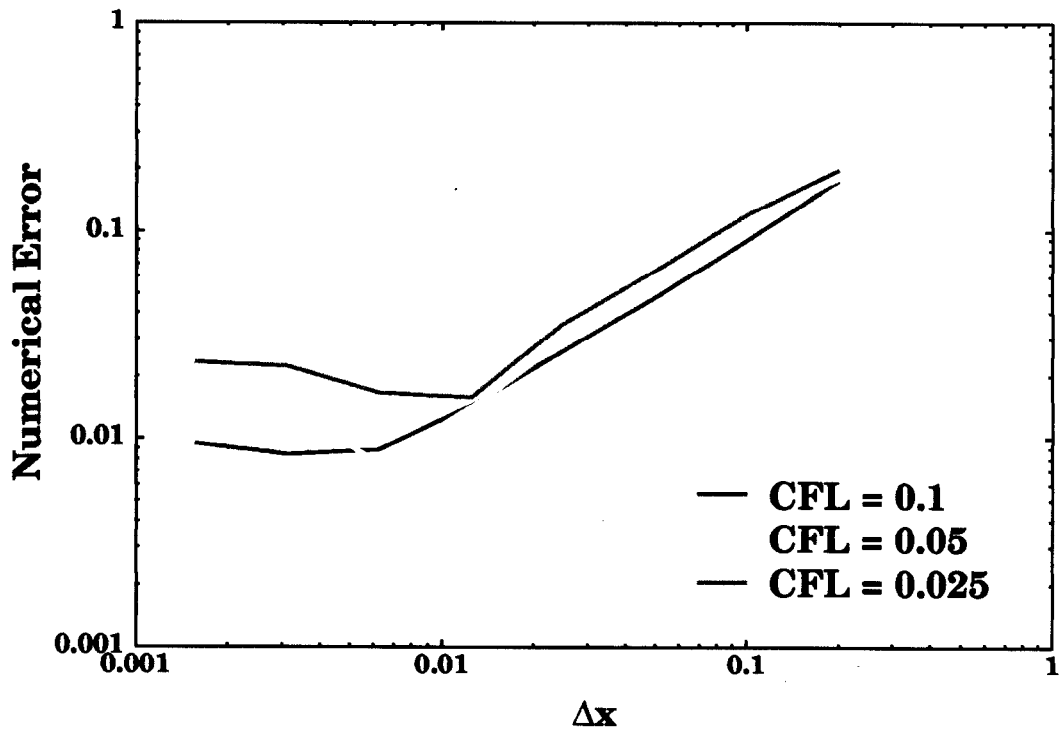


図 3.4: 1次元衝撃波管問題の空間解像度に対する精度

$\Delta x =$  (a)  $1/10$ , (b)  $1/20$ , (c)  $1/40$ , (d)  $1/80$ , (e)  $1/160$ , (f)  $1/320$ , (g)  $1/640$ , (h)  $1/1280$  により計算した結果。多くの計算格子点数を用いることで計算精度は向上するが、精度向上に伴い数値粘性が低下するため、膨張波面で数値振動が生じる。

(a)



(b)

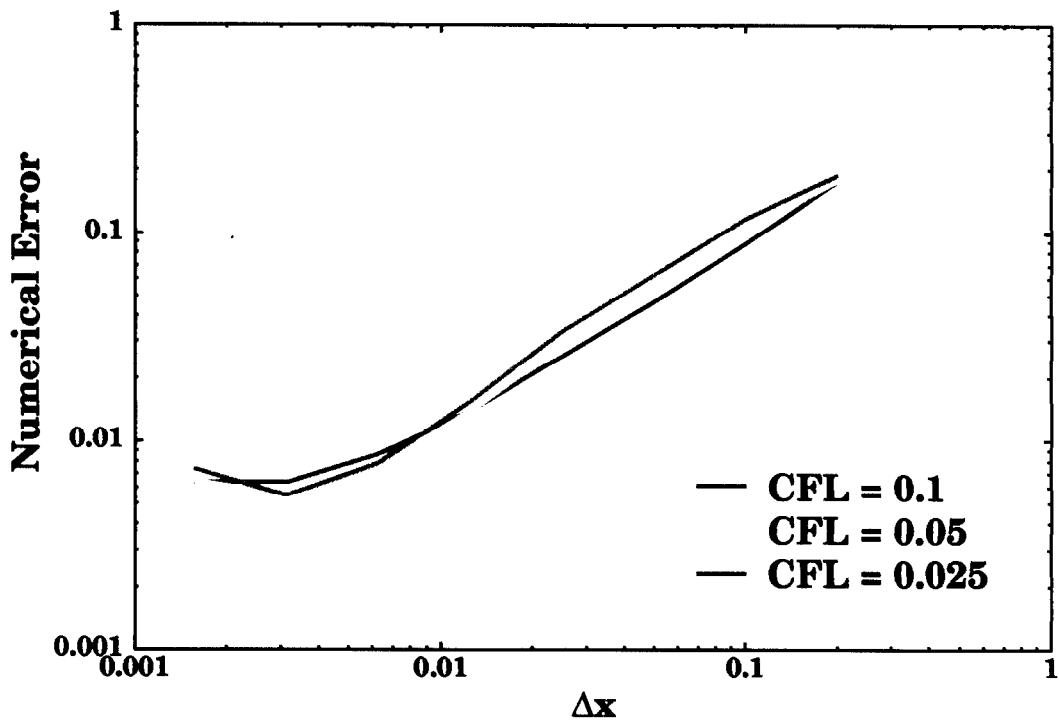


図 3.5: 1次元衝撃波管問題の数値誤差

解析解が存在する1次元衝撃波管問題に対しての数値誤差をIDO法による(a)風上3次補間を用いた数値計算結果、(b)風上5次補間を用いた数値計算結果に対して見積もった。

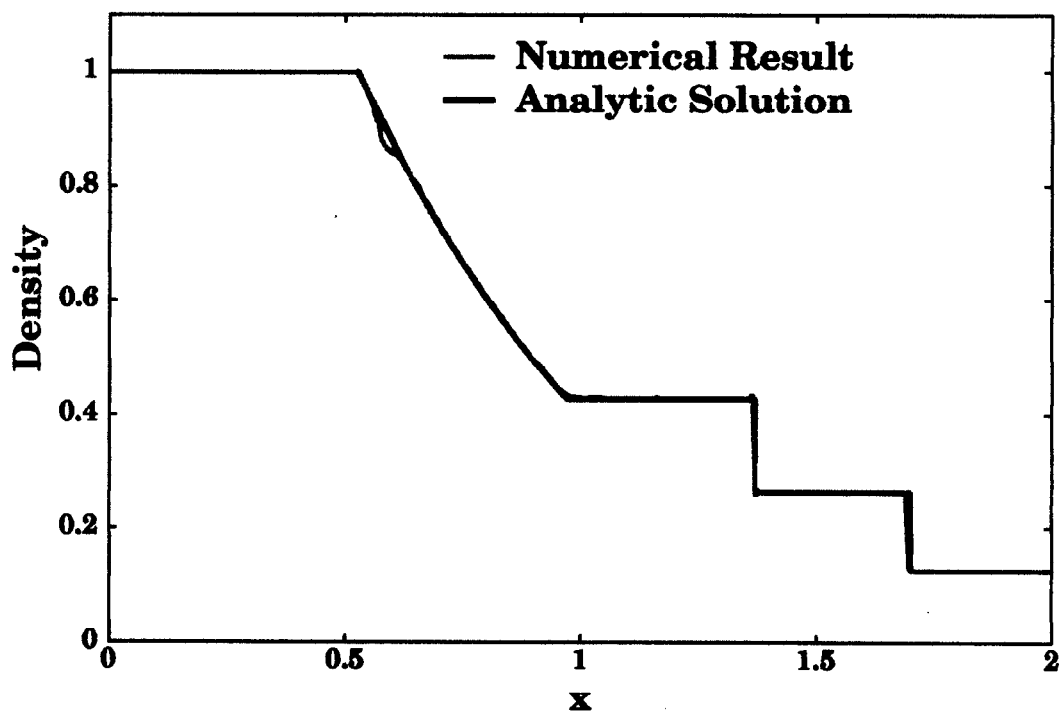
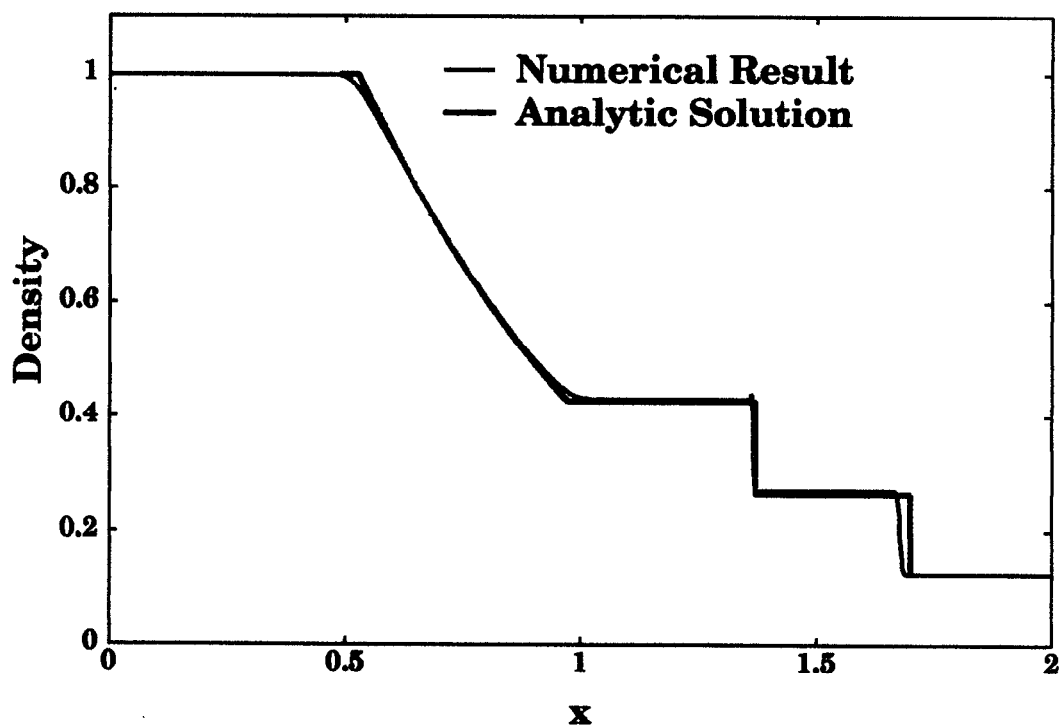


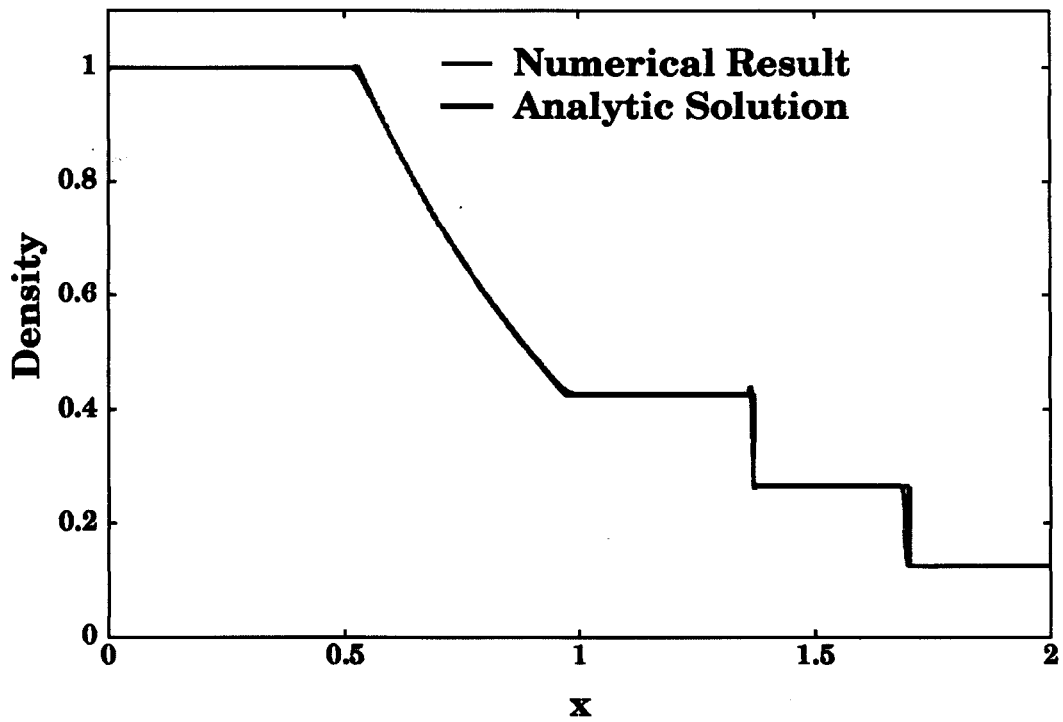
図 3.6: 高解像度グリッド使用時の数値不安定性

音波が 2 方向に伝播する膨張波面の移動を高解像度グリッドを用いて扱う場合、風上補間関数から得られる空間微係数精度が向上することで数値粘性として作用する数値誤差が低減するため数値振動が生じる。

(a)



(b)



(c)

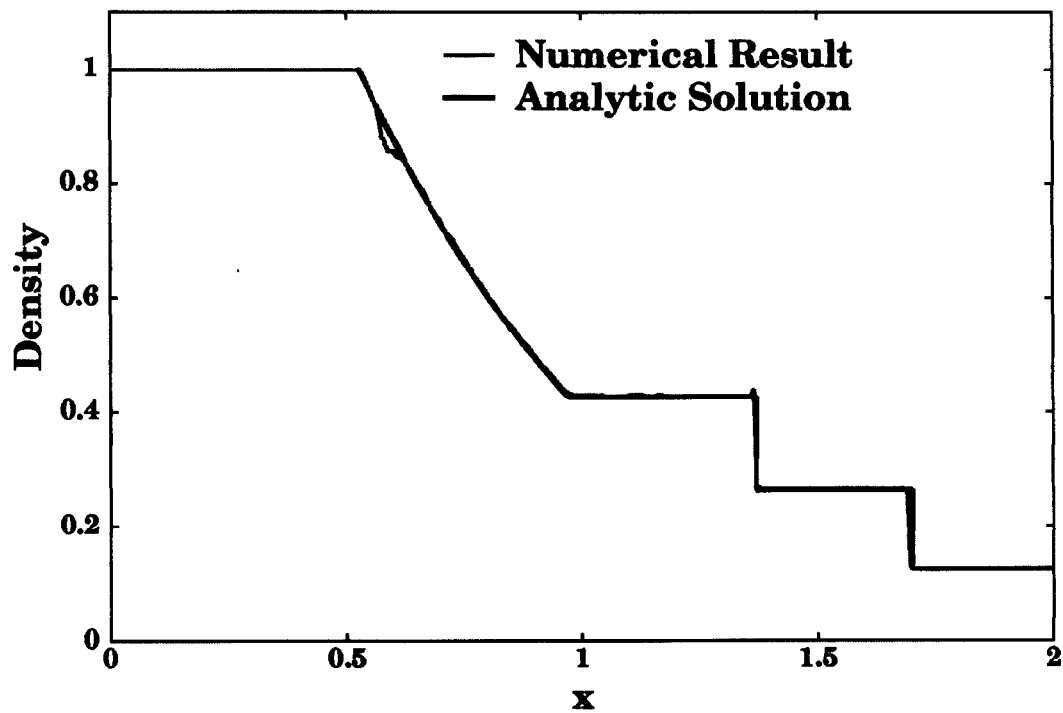


図 3.7: IDO 法による数値粘性を付加した 1 次元衝撃波管問題の数値解

数値粘性として (a)  $\kappa = 10^{-3}$ , (b)  $\kappa = 10^{-4}$ , (c)  $\kappa = 10^{-5}$  を考慮して  $\Delta x = 2/1280$  を用いて計算した結果。 $\kappa = 10^{-4}$  程度を用いて数値粘性を付加することで膨張波面の数値計算を安定化できる。 $\kappa = 10^{-3}$  程度では数値拡散が大きくなっていることがわかる。逆に  $\kappa = 10^{-5}$  程度では膨張波面の不安定性を除くことが出来ない。



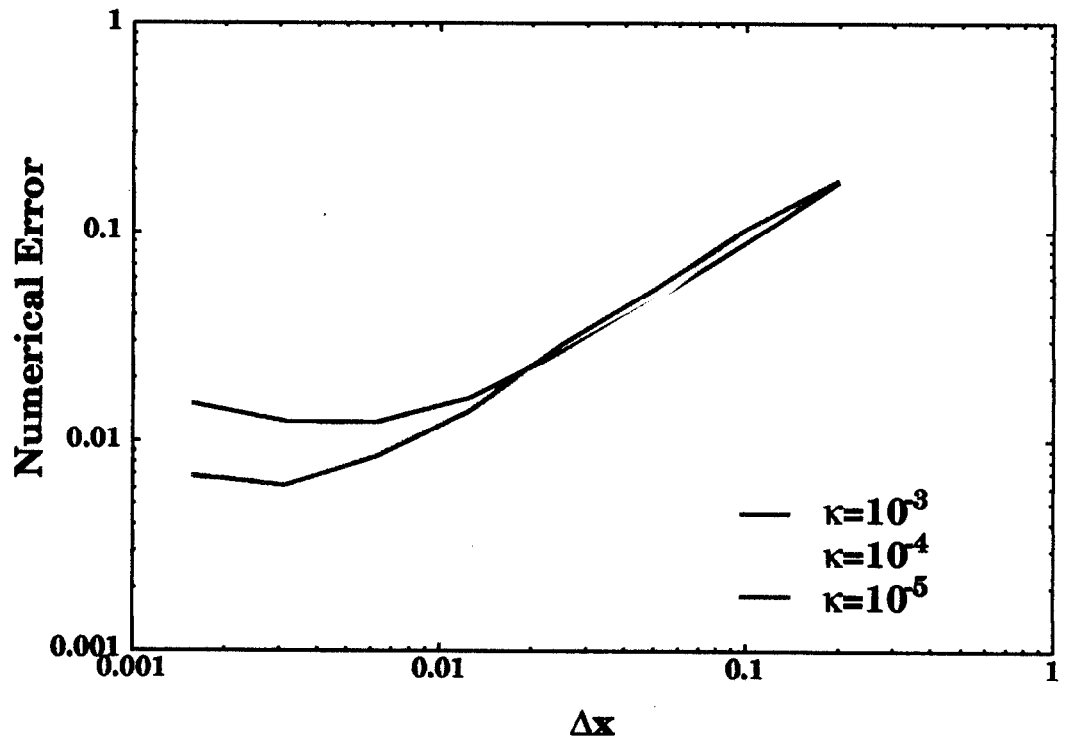


図 3.8: IDO 法による数値粘性を付加した 1 次元衝撃波管問題の数値誤差

$\kappa = 10^{-4}$  を用いた場合は数値粘性を付加しない数値計算結果と変わらない程度の数値拡散誤差で計算が行える。

### 3.1.3 1 次元衝撃波管問題のまとめ

1 次元衝撃波管問題を扱うことによって IDO 法の流体問題への適用性を検討した。圧縮性流体問題を、非保存形で記述される支配方程式を用いた IDO 法により扱う場合、人工粘性の導入が必要になる。有限差分法で利用されている von-Neumann 型人工粘性を IDO 法に対して適用し、その有効性について検討した。はじめに人工粘性を適用する圧縮領域の判定に関して有限差分法で用いられる速度の差分形式を用いた手法と IDO 法で利用可能な速度の空間微係数を直接利用する手法について比較した結果、両者ともに同程度の計算誤差で数値計算結果を得ることに成功した。ただし、速度の空間微係数により判定を行うことでグリッド点上での評価を行うことが可能であることから、より局所的に現象を扱うことが出来る。

また、計算グリッド数を増加させることでより高精度な数値計算結果を得ることが出来るが、その反面膨張領域において数値振動を生じることを示した。膨張領域では移流方向に対しプロファイルの進行方向が異なり、音波が 2 方向に進行するため流体速度を重視した風上補間を用いる移流計算では現象をとらえきれないことが原因として挙げられる。計算グリッドを高解像度にすることで膨張波面に生じる数値誤差は Navier-Stokes 方程式に粘性項として適度に数値粘性を付加することで抑制することが可能であり、数値計算を安定化できることを明らかにした。

## 3.2 1次元乱流 Burgers 方程式

Burgers 方程式

$$\frac{\partial \mathbf{u}}{\partial t} = -\mathbf{u} \cdot \nabla \mathbf{u} + \nu \Delta \mathbf{u} \quad (3.12)$$

は波の伝搬、音波、プラズマ物理などの理論において出現する様々な非線形現象を記述する。また、Navier-Stokes 方程式とエネルギー散逸における非線形性において良く似た性質を示す。その一方で初期条件に対する敏感性は無く (Chaos) 可積分形であることから Navier-Stokes 方程式による乱流解析のモデルとして Burgers 方程式は注目を集めて来た [14, 15]。さまざまな研究も進められており、右辺に対してホワイトノイズを加えながらの界面成長についての議論や、ランダムな初期プロファイルによる検討、計算時間進行に伴うエネルギー散逸がとりわけ盛んな分野である。

ここでは、乱流 Burgers 方程式を扱うことで局所補間微分オペレータ (Interpolated Differential Operator; IDO) 法の特徴的数値粘性について考察し、乱流場に対する直接数値計算 (Direct Numerical Simulation; DNS) への適用性を検討する。

### 3.2.1 スペクトル法との比較

Burgers 方程式

$$\frac{\partial u}{\partial t} = -u \frac{\partial u}{\partial x} + \nu \frac{\partial^2 u}{\partial x^2} \quad (3.13)$$

を支配方程式として扱う。初期プロファイルが一様でなければ計算時間が進行するに従い衝撃波面が形成され、それに伴い物理量  $u$  の各波数成分のエネルギーを波数 ( $k$ )-エネルギー ( $E$ ) 平面において記述すると高波数領域では

$$E_0(k) \propto |k|^{-2} \quad (3.14)$$

の傾向が得られることが解析的に知られている [14]。

初期プロファイルを計算領域  $[0 : 2\pi]$  に対して周期関数により

$$u(x) = \sum_k \cos(kx + 2\pi \text{Phase}[k]) \quad (3.15)$$

として与える。ここで、 $k = 1 \sim 20$  とし、それぞれの Phase は表 3.1 に示すとおりである。

表 3.1: Burgers 方程式の位相初期値

Phase[1] = 0.296540739946067,	Phase[2] = 0.317679069936275,
Phase[3] = 0.336732201278210,	Phase[4] = 0.01553758233785629,
Phase[5] = 0.01259063929319382,	Phase[6] = 0.309237979352474,
Phase[7] = 0.710570029914379,	Phase[8] = 0.241913042962551,
Phase[9] = 0.300905816257000,	Phase[10] = 0.809423826634884,
Phase[11] = 0.784906439483166,	Phase[12] = 0.740388952195644,
Phase[13] = 0.306427948176861,	Phase[14] = 0.157073281705379,
Phase[15] = 0.833270482718945,	Phase[16] = 0.289010964334011,
Phase[17] = 0.492871634662151,	Phase[18] = 0.786237858235836,
Phase[19] = 0.421671681106091,	Phase[20] = 0.554008327424526

計算格子点数 2048、計算時間刻幅  $\Delta t = 1 \times 10^{-5}$ 、粘性係数  $\nu = 1/50$  としたスペクトル法、IDO 法それぞれについての計算結果を図 3.9、図 3.10 に示す。ただし、IDO 法での計算は空間微係数計算をすべて中心補間により行う。計算が進行するに従い急峻なプロファイル(衝撃波)が形成され、伝搬して行く様子が伺える。この衝撃波の生成に伴いプロファイル中に高波数成分が発生するため、プロファイルの高精度記述のためには差分法では非常に多くの計算格子点数を必要とする。その一方でスペクトル法と計算結果を比較して見ても、IDO 法は遜色無い計算結果を示している。

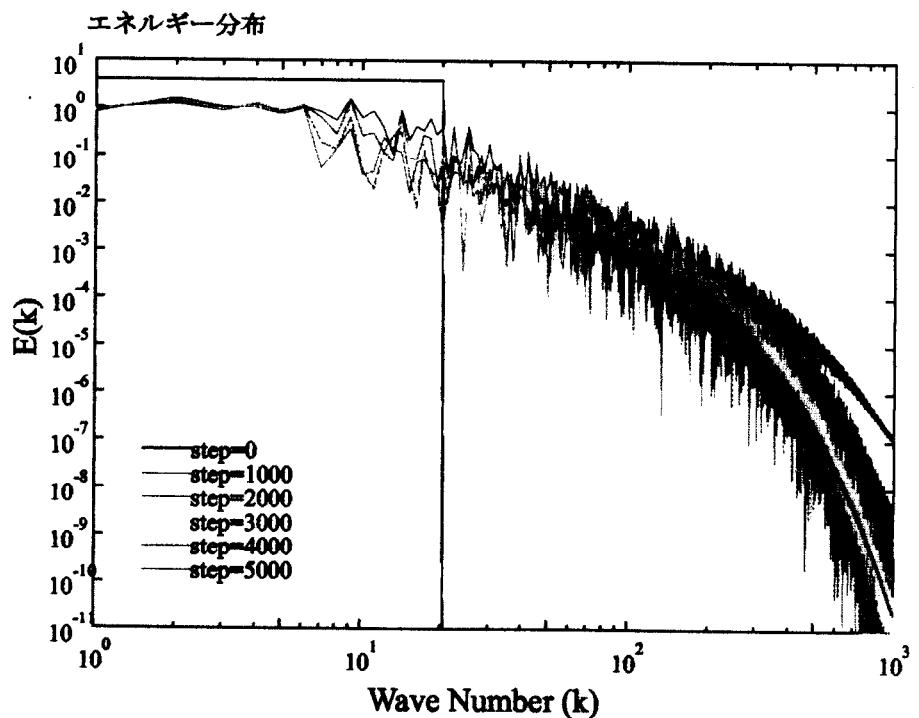
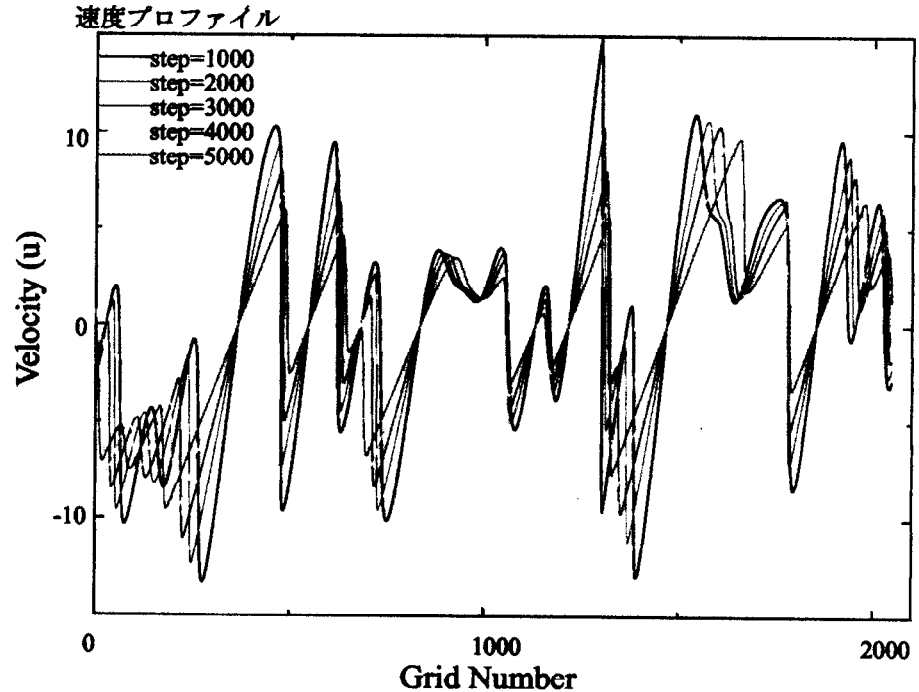


図 3.9: スペクトル法による Burgers 方程式の計算結果

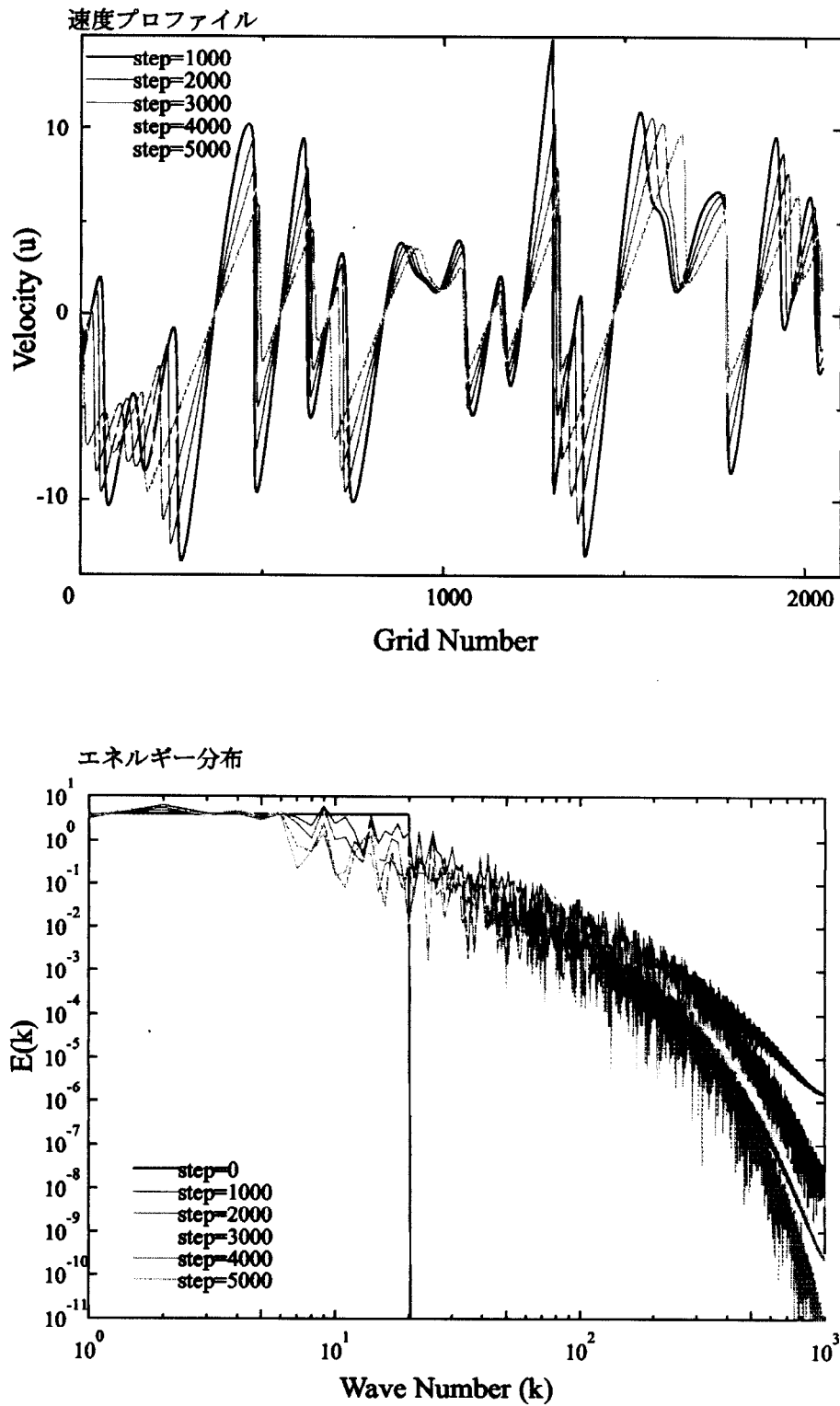


図 3.10: IDO 法による Burgers 方程式の計算結果

次に得られた結果の正当性を検証するため、物理量  $u$  を

$$u(x) = \sum_k \cos(kx + 2\pi \text{Phase}[k]) + 5 \quad (3.16)$$

として与え、同様の問題を扱う。波数分解できる成分に関しては先の例題と同様であることから得られるエネルギースペクトルは先のものと一致することが期待される。計算結果を図 3.11、図 3.12 に示す。

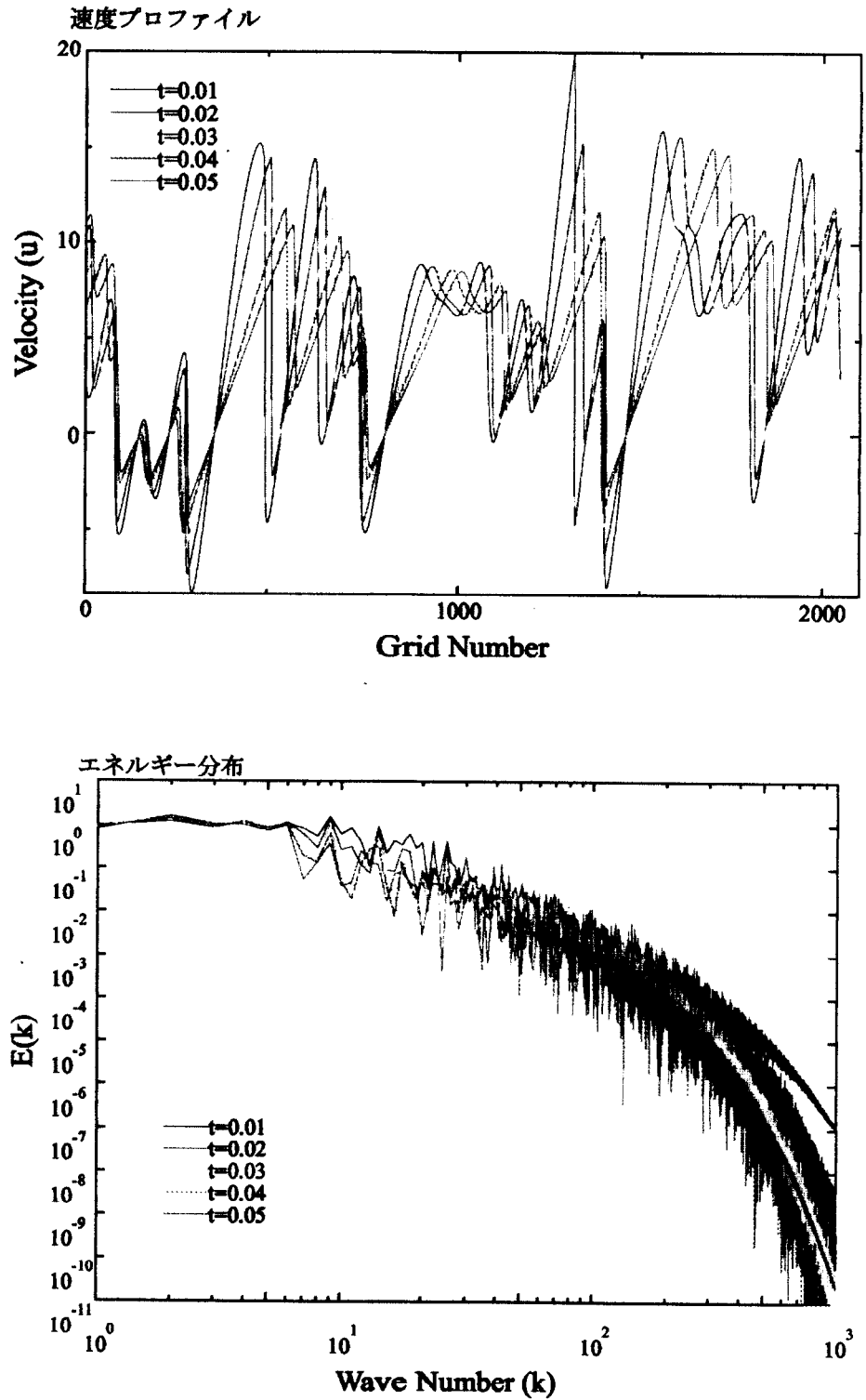


図 3.11: スペクトル法による平均移流速度を考慮した Burgers 方程式の計算結果

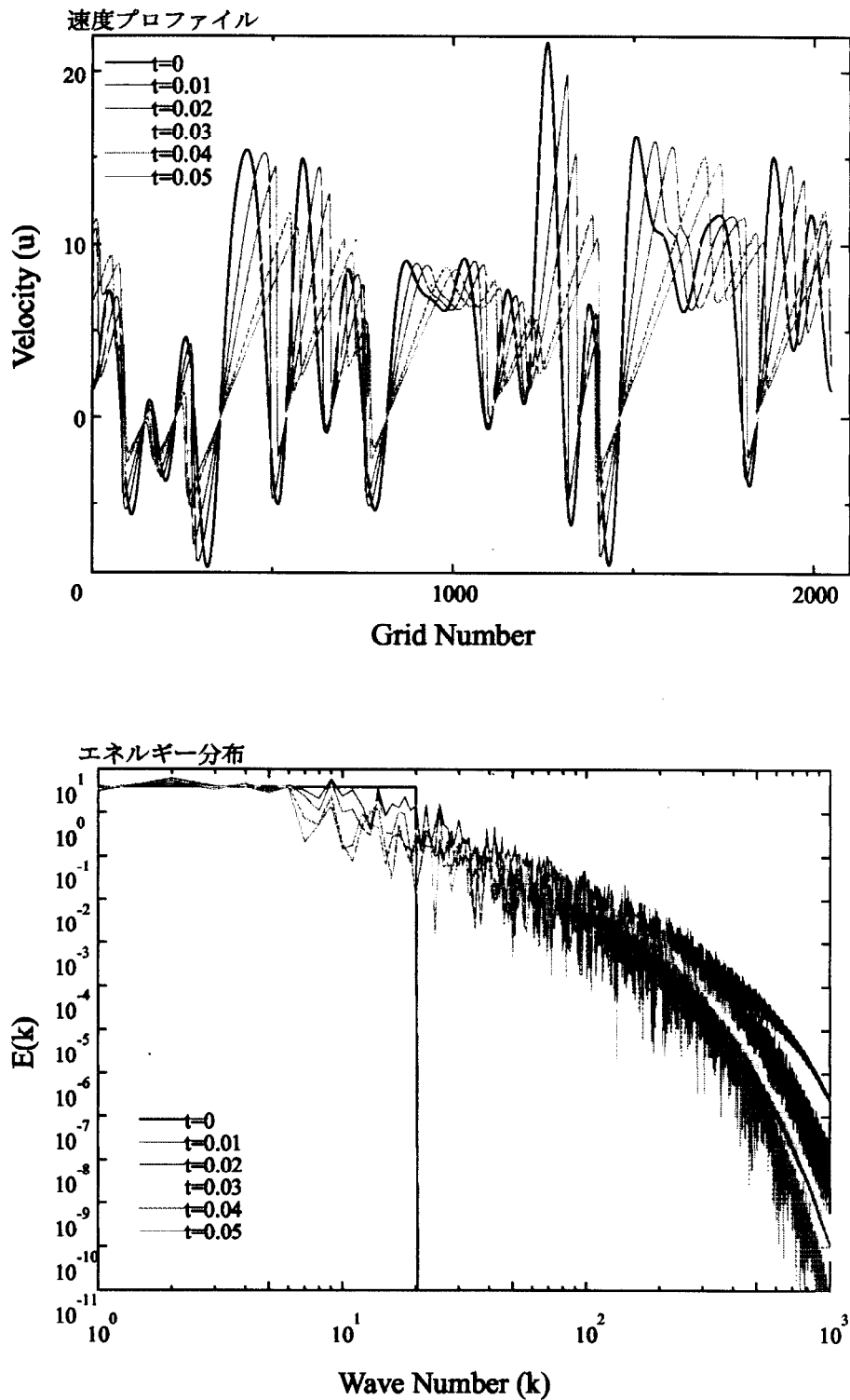


図 3.12: IDO 法による平均移流速度を考慮した Burgers 方程式の計算結果

移流速度を全体に 5 加算した場合においても IDO 法の数値計算結果はスペクトル法の結果と良く一致しており、波数-エネルギー平面を先の結果と比較しても良く一致した結果が得られていることが分かる。同様の例題を差分法でよく用いられる KK-Scheme、UTOPIA Method と比較を行う。数値計算結果を図

3.13 - 図 3.14 に示す。

オフセット速度を含まない図 3.13 に示す結果と、オフセット速度 5 を考慮した図 3.14 に示す両結果を比較した場合には良く一致しているが、スペクトル法、IDO 法の結果と比較した場合、低波数領域 ( $k < 50$ ) におけるエネルギースペクトルは良く一致するものの、高波数領域ではエネルギーの減衰が生じている。

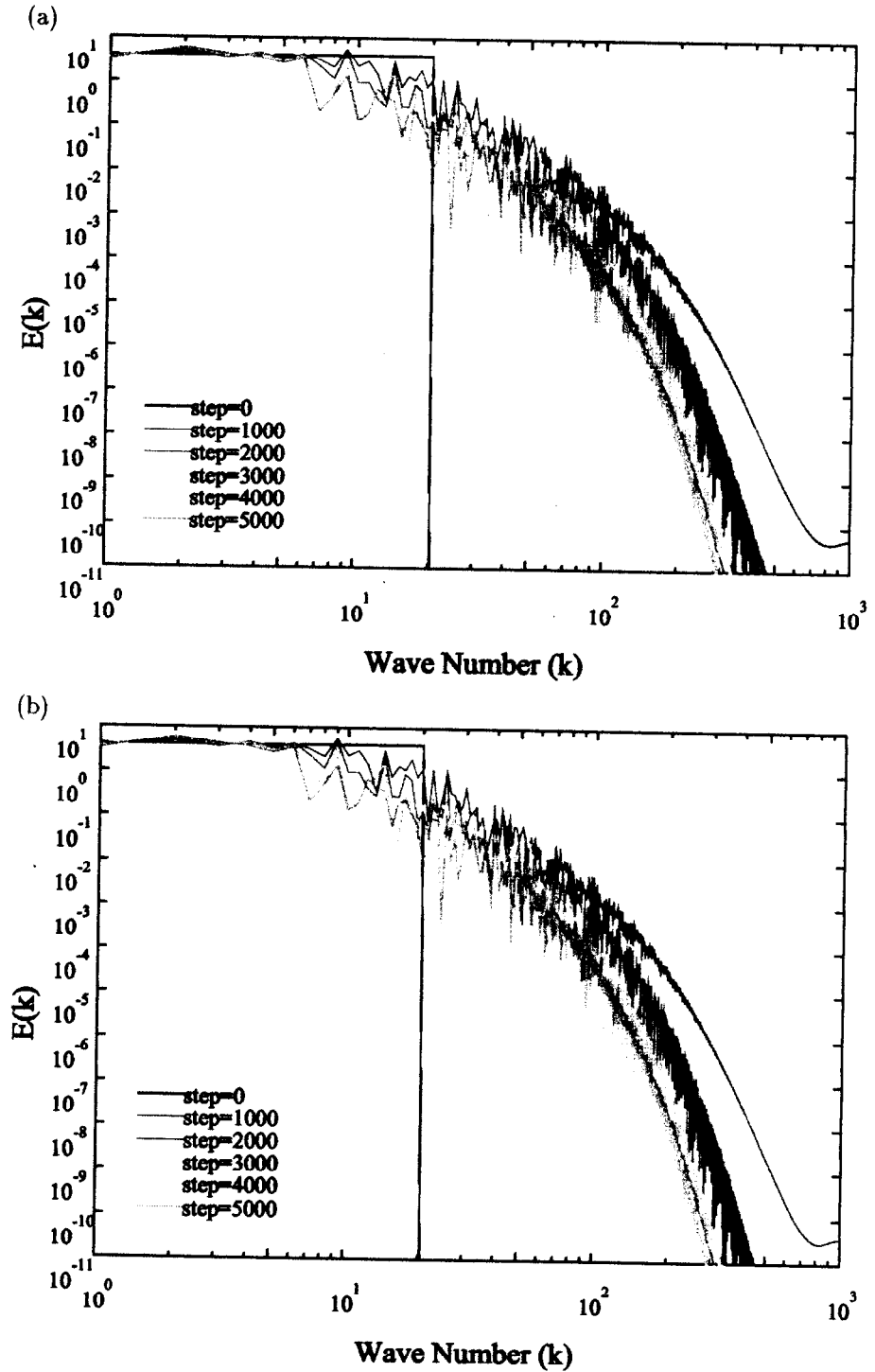


図 3.13: (a)KK-Scheme、(b)UTOPIA Method による乱流 Burgers 方程式の検討:

低波数領域 ( $k < 50$ ) ではスペクトル法、IDO 法に良く一致したエネルギースペクトルを示すが、高波数領域ではエネルギーの減衰が観測される。

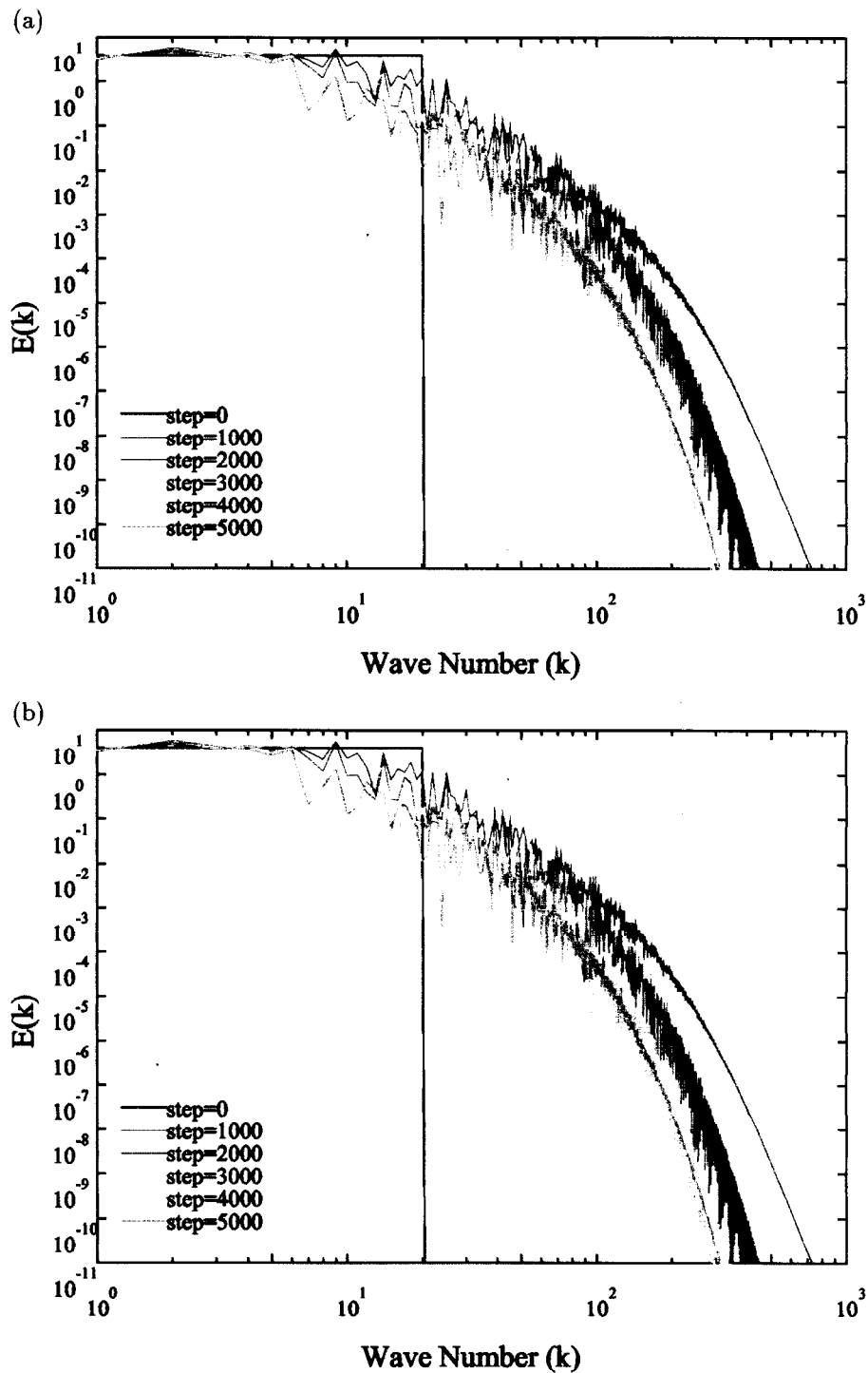


図 3.14: (a)KK-Scheme、(b)UTOPIA Method による乱流 Burgers 方程式の検討:  
オフセット速度 5 を加えた場合でも低波数領域 ( $k < 50$ ) ではスペクトル法、IDO 法に良く一致したエネルギー スペクトルを示すが、高波数領域ではエネルギーの減衰が観測される。

この結果から、KK-Scheme、UTOPIA Method には数値粘性が陰的に含まれていることが考察される。その一方で IDO 法はスペクトル法と比較しても十分に数値粘性が抑制されていることを明らかにした。



### 3.2.2 グリッド解像度、粘性係数 (レイノルズ数) に対する考察

IDO 法を用いた乱流 Burgers 方程式の計算結果がスペクトル法によるものと良く一致した結果を示したことから、その妥当性を明らかにした。次に、グリッド解像度に対する傾向について検討を行う。また、より高レイノルズ数での計算を目的に粘性係数を変化させ、計算結果について考察を行う。

グリッド解像度に対する傾向を検討することを目的として、式 (3.15)、表 3.1 に示す先の初期プロファイルを利用し、粘性係数  $\nu = 1/50$  により計算格子点数を 512, 1024, 2048, 16384 を利用して、 $t = 5 \times 10^{-5}$  まで計算した結果を図 3.15 に示す。

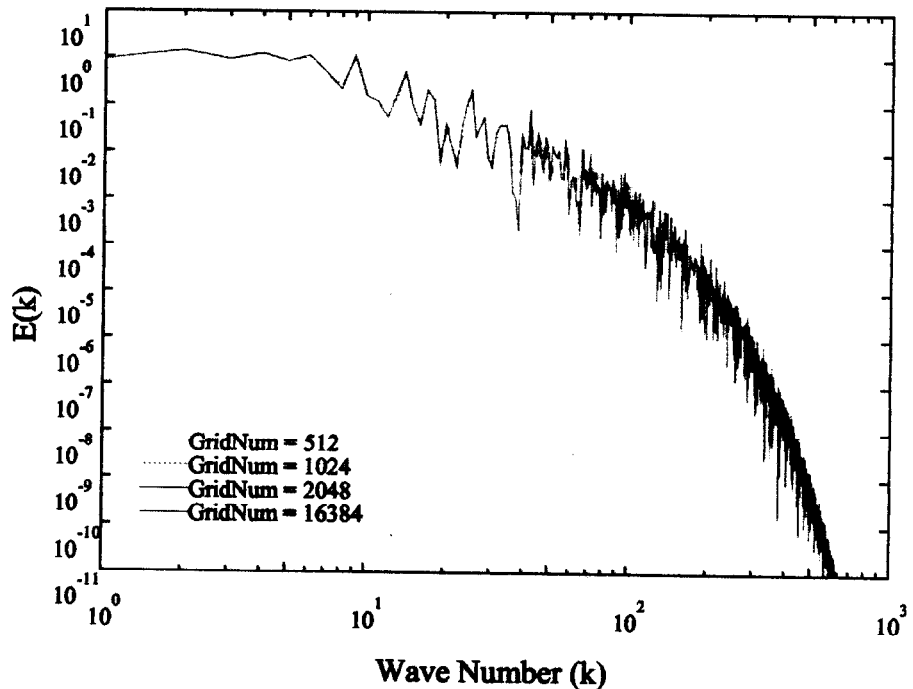


図 3.15: 乱流 Burgers 方程式のグリッド数に対する傾向:

格子点数 512 では他の計算格子点数との結果と若干異なるエネルギースペクトルを示すが、格子点数を 1024 以上にする場合にはほぼ同形のエネルギースペクトルが得られた。

この計算結果から、計算格子点数にかかわらず良く一致したエネルギースペクトルが得られることを明らかにした。また、先のスペクトル法との比較を目的とした数値計算結果がグリッド解像度に依存する結果ではないことがこの結果から示された。この計算結果を背景に、粘性係数に対するエネルギースペクトルの傾向をグリッド数 2048 にて考察する。

粘性係数  $\nu = 1/50, 1/100, 1/150, 1/250$  について数値計算を行い、 $t = 5 \times 10^{-2}$  での波数に対するエネルギースペクトルを図 3.16 に示す。解析的背景より、粘性項を含まない Burgers 方程式では衝撃波が誘起されるまで時間が経過すると、その特性から波数に対するエネルギースペクトルを観測した場合、高波数領域は  $E(k) \propto k^{-2}$  の傾向を示すことが知られている。このことから、高レイノルズ数での数値シミュレーションを目的とした場合、高波数領域におけるエネルギースペクトル分布形状がその基礎精度の指標となる。

図 3.16 から、粘性係数を低下させるに従い高波数領域におけるエネルギースペクトル分布は  $E(k) \propto k^{-2}$  に従っている様子が確認され、高レイノルズ数乱流場に対する適用可能性を明らかにした。また、本計算では初期プロファイル中の速度最大値が 8.32、計算空間が  $[0 : 2\pi]$  であることから、粘性係数  $\nu = 1/250$  に相当するレイノルズ数は初期値に対しては  $Re = 250 \times 8.32 \times 2\pi \simeq 13000$  となり、 $10^4$  程度のレイノル

ズ数での数値計算が安定に行えることを明らかにした。

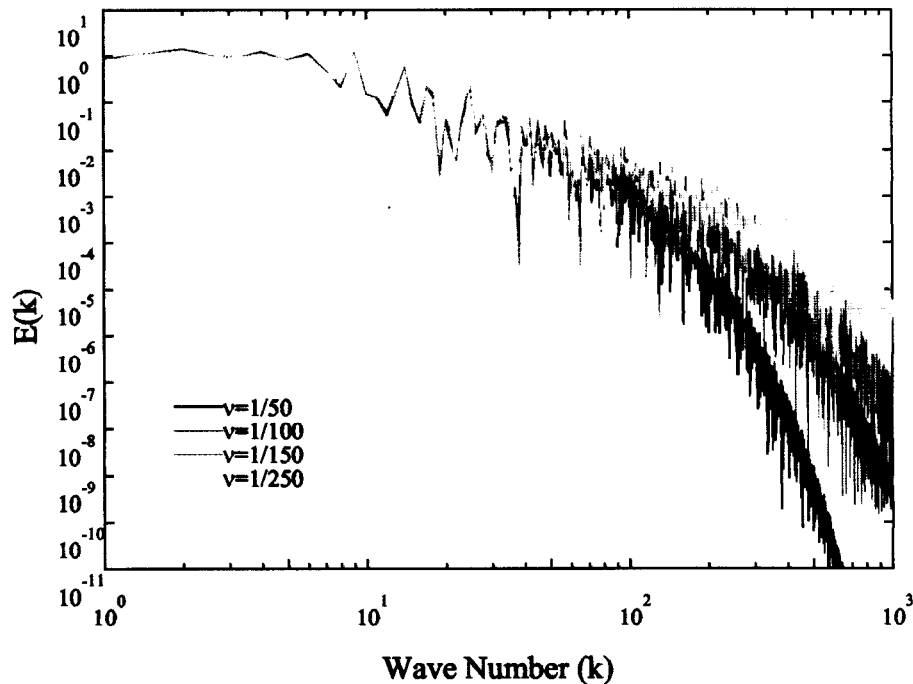


図 3.16: 乱流 Burgers 方程式の粘性係数に対する傾向:

粘性係数を小さくするに従い、解析的背景から得られる高波数領域におけるエネルギースペクトルの  $k^{-2}$  での減衰が現れる傾向を得た。

これらの IDO 法による数値計算結果を KK-Scheme、UTOPIA 法 [16] による計算結果と比較検討を行う。両手法における時間前進は 4 次精度の Runge-Kutta 法を用いる。図 3.17 に示す計算結果は IDO 法による計算結果に比べ高波数領域におけるエネルギースペクトルが減衰している様子が伺える。論文 [16] 中では計算格子点数により KK-Scheme が UTOPIA 法に比べ少ない格子点数で高精度計算が可能であることを述べているが、ここでは両者にほとんど差が見られない計算結果が得られた。また、KK-Scheme により乱流場を扱う長所として、論文中で以下の点を挙げている。

これまでの数値流体計算手法では乱流場を扱う場合、グリッドサイズ以下の細かな渦構造を扱うことができないことからそのような細かい渦構造の現れる乱流場を扱う場合にはなんらかのモデルを導入する必要があった。しかし、その乱流モデルは Burgers 方程式や Navier-Stokes 方程式に見られる拡散項と同形状をしており、高精度計算手法を用いたとしてもあえてその上で数値粘性を加えている形をとっている。また、乱流場を扱う場合でも細やかな渦構造には実質的に興味無く、大きな渦構造を捕らえることが乱流場を扱う上で重要である。つまり、あえて高精度計算手法を扱う必要は無く、数値粘性により陰的に乱流モデルが考慮されるこれらの計算手法を用いることは乱流解析に対して有効である。

そこで、低波数領域におけるエネルギースペクトルについて検討を行う。粘性係数が  $1/100$  以下では波数 50 程度までは計算格子点数を大幅に少なくした場合でも (図 3.18) IDO 法による計算結果と良く一致した結果が得られている。しかし、粘性係数が大きくなると更にエネルギースペクトルの減衰が激しくなり、せいぜい 20 程度までが良く表現できていると言える。工業的利用に使用目的を限定した場合には充分有効な計算結果を示すと考えるが、DNS などの粘性を除いた計算を行いたい場合には自由度が無いため、IDO 法による計算がその点では有利であると考えられる。

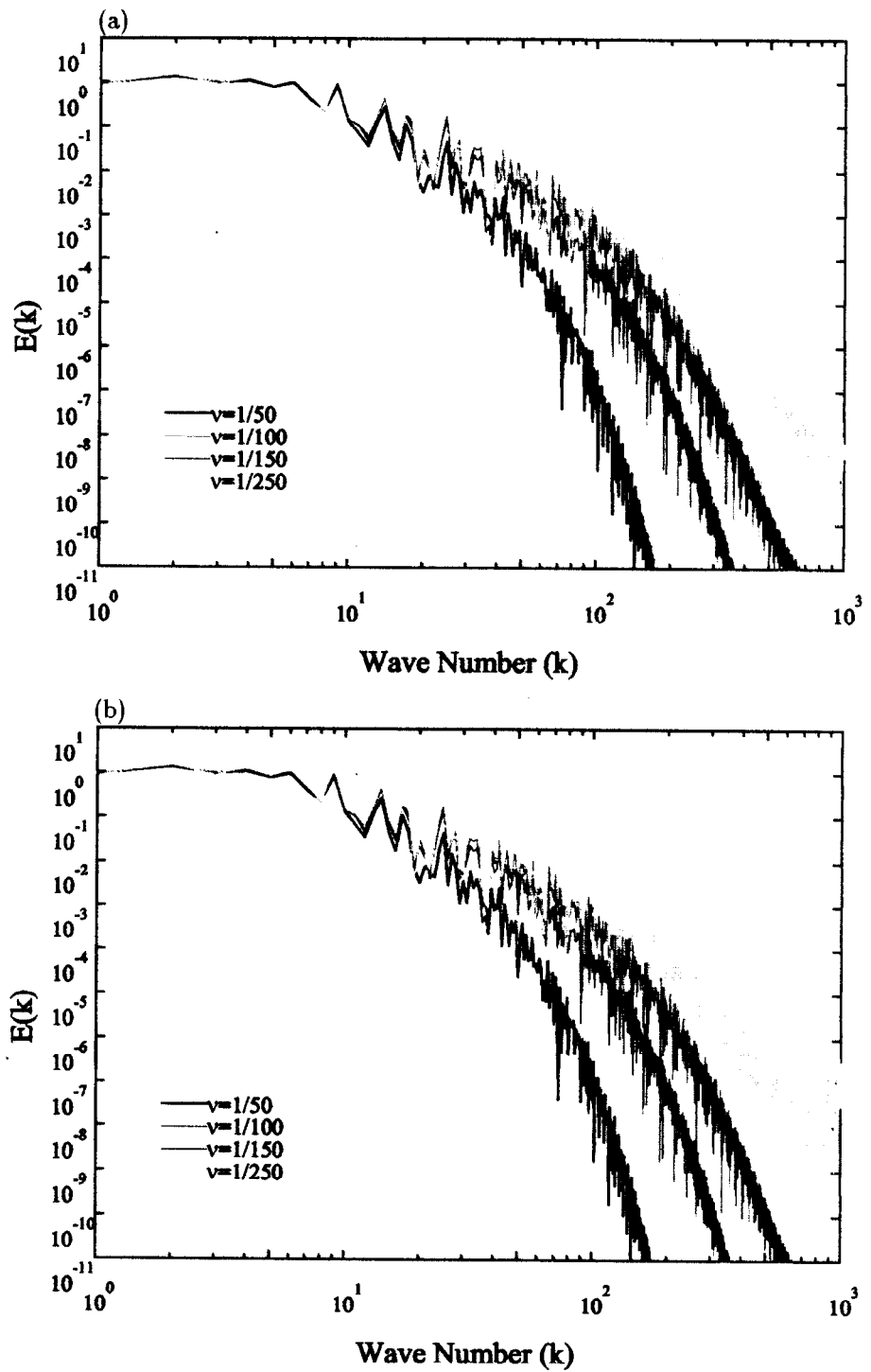


図 3.17: KK-Scheme, UTOPIA Method による計算結果:

(a)KK-Scheme、(b)UTOPIA Method による乱流 Burgers 方程式の計算結果:

それぞれの計算結果は低波数領域 ( $k < 20$ ) では IDO 法と良く一致した結果を示すが高波数領域ではエネルギースペクトルの減衰が見られる。

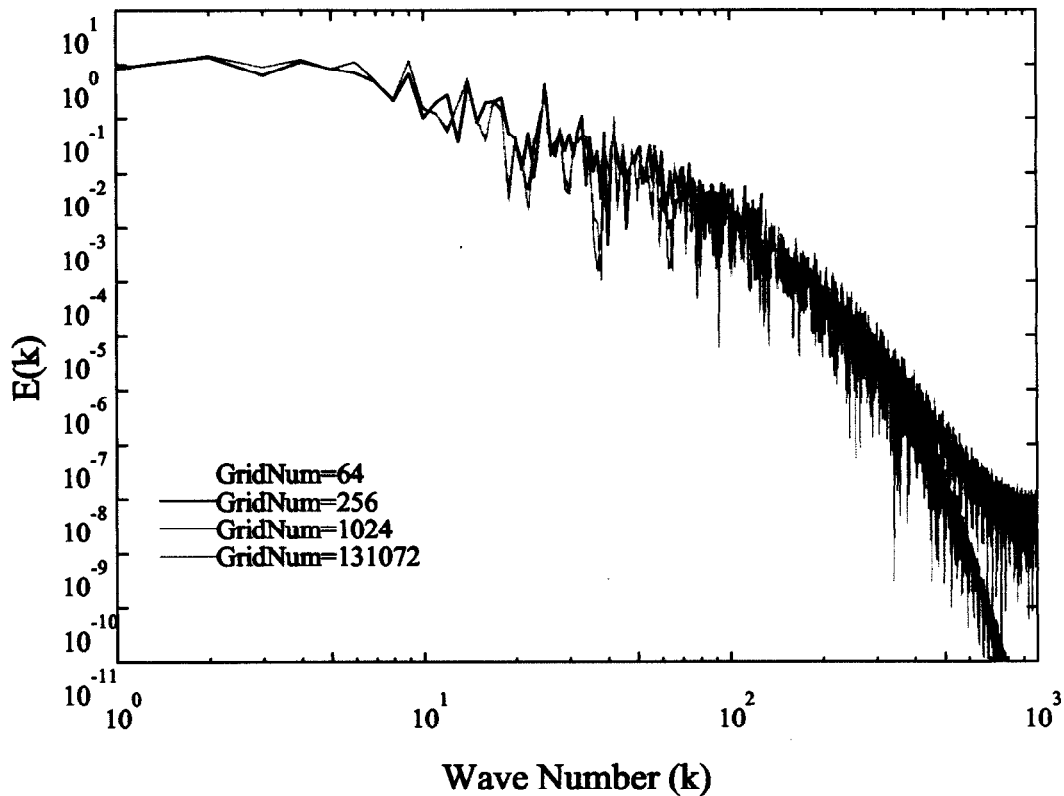


図 3.18: KK スキームによる乱流 Burgers 方程式の計算精度:  
 グリッド点数を大幅に少なくした場合でも低波数領域では高解像度グリッドを用いた計算結果と良く一致している。

### 3.2.3 まとめ

乱流 Burgers 方程式を扱うことにより IDO 法の数値粘性について検討を行った結果、IDO 法の中心補間を用いた計算手法を利用することによりスペクトル法と同程度の数値粘性で乱流問題を扱える可能性を示した。また、差分法の代表として KK-Scheme により同様の計算を行い比較検討を行った結果、IDO 法は KK-Scheme に含まれるような高波数領域におけるエネルギーを減衰させる数値粘性は十分に抑制されており、直接数値計算への適用可能性があることを明らかにした。

### 3.3 前線発達問題への適用

理想化された前線発達 (Frontogenesis) の運動は一般的な Navier-Stokes 方程式やエネルギー保存式を扱う流体問題を扱うことなく解くことが出来る。しかし、静的な速度場であるにも関わらず得られる流体構造が複雑であることや、その解析解が知られていることから、検証例題として良く用いられている。

#### 3.3.1 計算条件ならびに解析解

温度場  $\Psi$  を計算対象としており、保存方程式

$$\Psi_t = -u\Psi_x - v\Psi_y + (u_x + v_y)\Psi \quad (3.17)$$

により、プロファイルの時間発展が行われる。Doswell [17] によって提案され、Rančić [18] により改良された初期温度分布は次のようにハイパボリックタンジェント (Hyperbolic Tangent; tanh) を用いてあらわされる。

$$\Psi(x, y, 0) = -\tanh\left(\frac{y - y_c}{2}\right) \quad (3.18)$$

ここで、 $y_c$  は計算領域における  $y$  座標値の中心値をあらわす。回転速度場を仮定し、計算領域中心  $(x_c, y_c)$  からの距離  $r$  を用いることで初期速度は次のように与えられる。

$$v_T = \text{sech}^2(r) \tanh(r) / v_{T_0} \quad (3.19)$$

ここで、 $v_{T_0}$  は速度の最大値が 1 となるように与えられる。式 (3.19) を用いて 2 次元保存方程式を式 (3.18) で表される初期プロファイルに対して解くと、解析解

$$\Psi(x, y, t) = -\tanh\left[\frac{(y - y_c) \cos(\omega t)}{2} - \frac{(x - x_c) \sin(\omega t)}{2}\right] \quad (3.20)$$

が得られる [18]。解析解からの相対誤差を見積もるため、初期プロファイルが 0 となる点を除くことを目的として以下では

$$\Psi(x, y, 0) = -\tanh\left(\frac{y - y_c}{2}\right) + 5 \quad (3.21)$$

として与える。よって、解析解は

$$\Psi(x, y, t) = -\tanh\left[\frac{(y - y_c) \cos(\omega t)}{2} - \frac{(x - x_c) \sin(\omega t)}{2}\right] + 5 \quad (3.22)$$

となる。

#### 3.3.2 計算結果

$x, y$  方向に対して無次元長さ 8 の計算領域を用い前節で説明した初期プロファイルにより計算を行う。計算グリッド数は  $100 \times 100$  とし、固定境界条件を用いる。

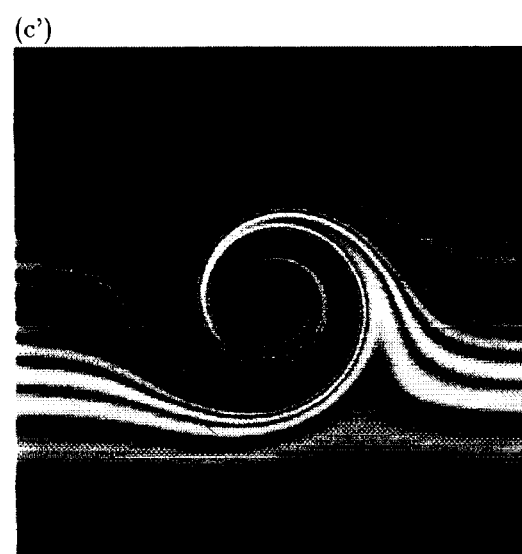
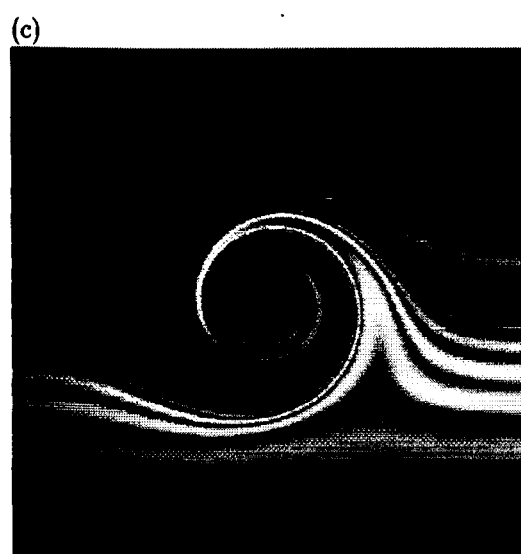
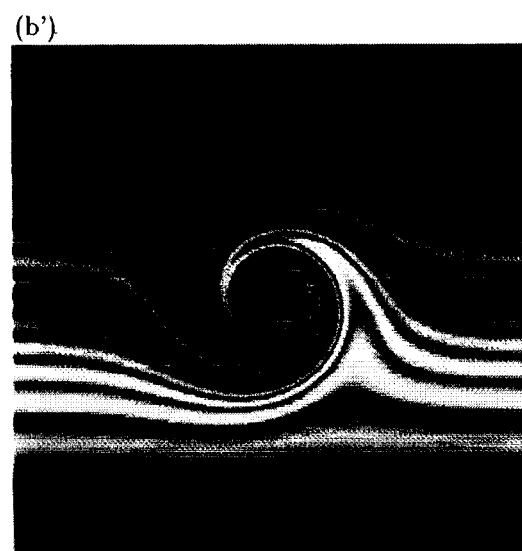
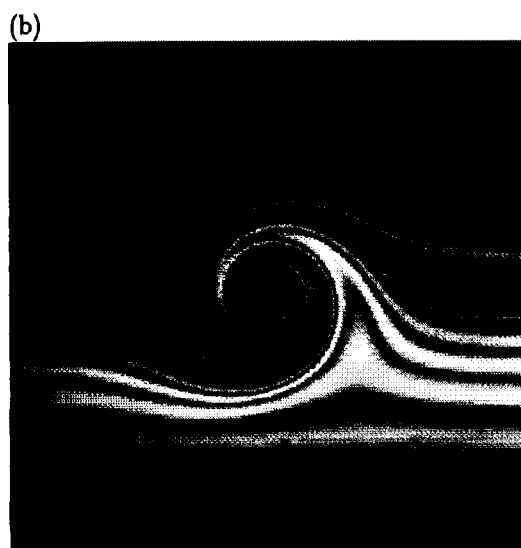
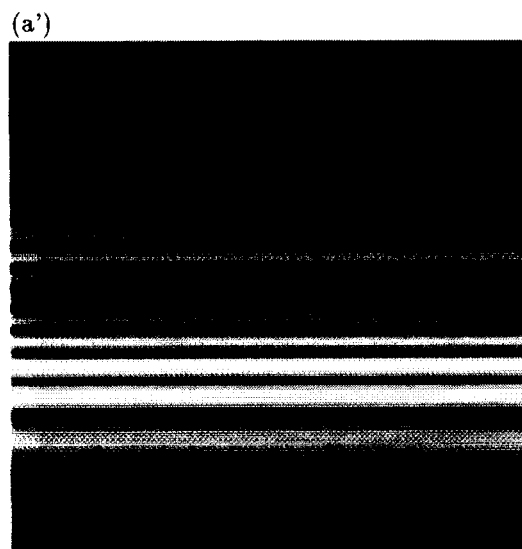
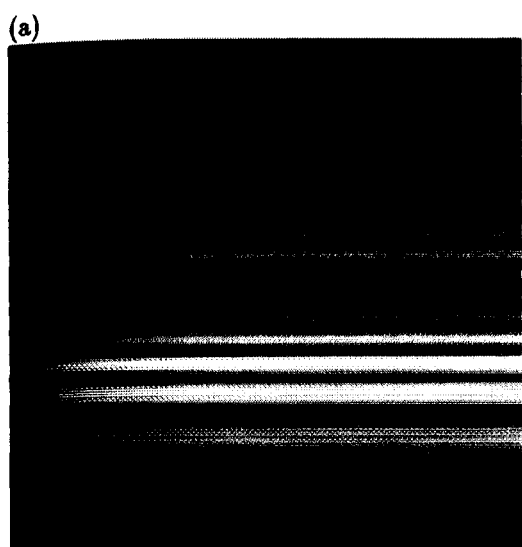
CFL 数  $(\Delta t / \Delta x)$  0.1 を用いて見積もられた時刻  $t = 0, 3.2, 6.4, 9.6, 12.8, 16$  における数値解と解析解を図 3.19 (a) - (f), (a') - (f') に示す。いずれの時刻においても解析解によく一致した結果を得ることに成功した。

解析解が得られていることを利用して、数値拡散誤差、保存誤差の時間履歴について検証することを考える。計算結果の保存誤差、数値拡散誤差をそれぞれ

$$\text{ConservationError} = \frac{\sum_{i=1}^N \sum_{j=1}^M f_{\text{Analytic}}(i, j) - \sum_{i=1}^N \sum_{j=1}^M f(i, j)}{\sum_{i=1}^N \sum_{j=1}^M (f_{\text{Analytic}}(i, j))} \quad (3.23)$$

$$\text{DiffusiveError} = \sum_{i=1}^N \sum_{j=1}^M \frac{|f_{\text{Analytic}}(i, j) - f(i, j)|}{f_{\text{Analytic}}(i, j)} / \text{MeshNumber} \quad (3.24)$$

を用いて計算する。ここで、 $f_{\text{Analytic}}$  は解析解を示す。



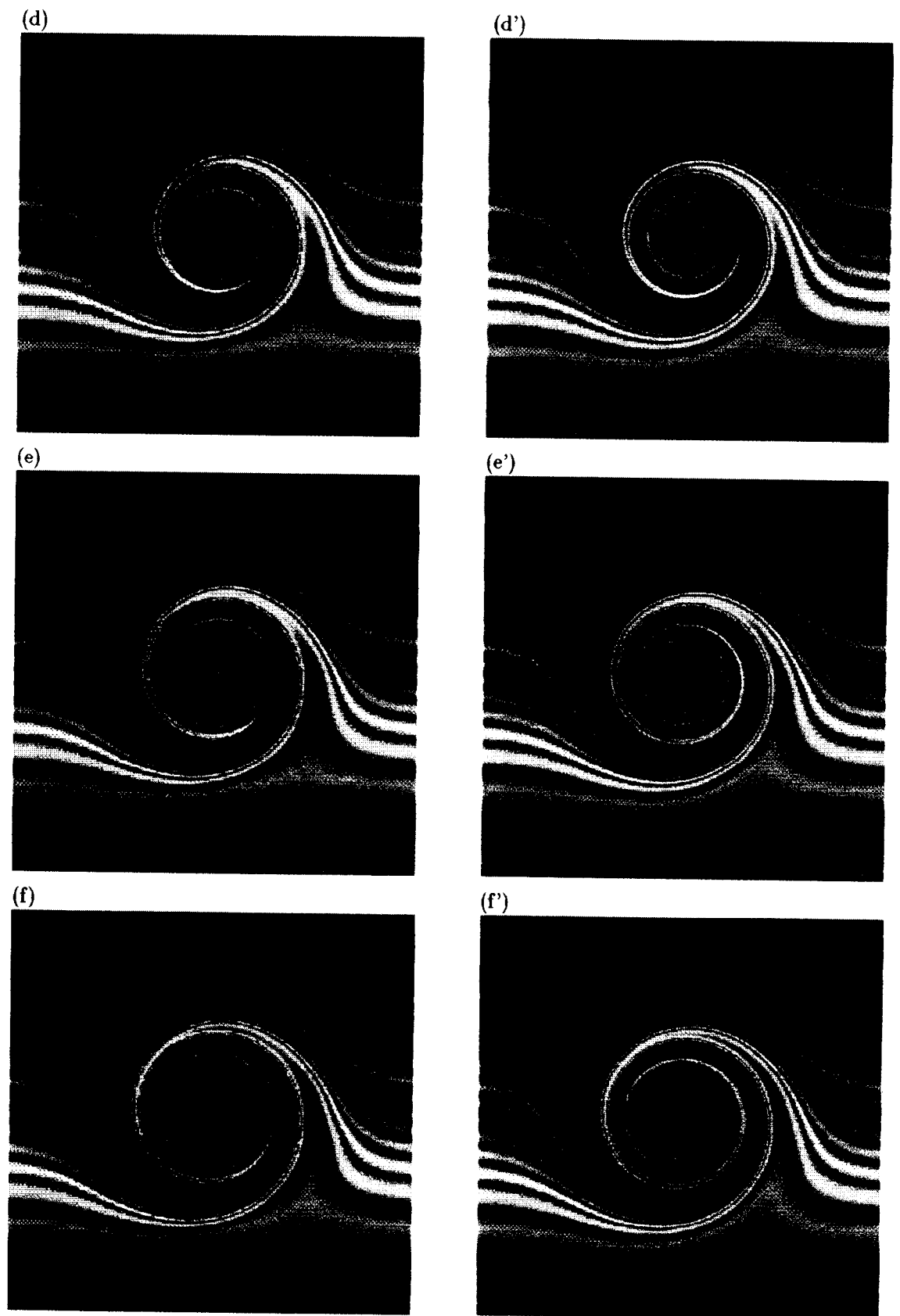


図 3.19: IDO 法による Frontogenesis 問題の数値計算結果  
 $t = 0(a), 3.2(b), 6.4(c), 9.6(d), 12.8(e), 16(f)$  における温度分布と  $t = 0(a'), 3.2(b'), 6.4(c'), 9.6(d'), 12.8(e'), 16(f')$  における解析解

計算格子点数を変化させ、各誤差を見積もった結果を図 3.20 に示す。

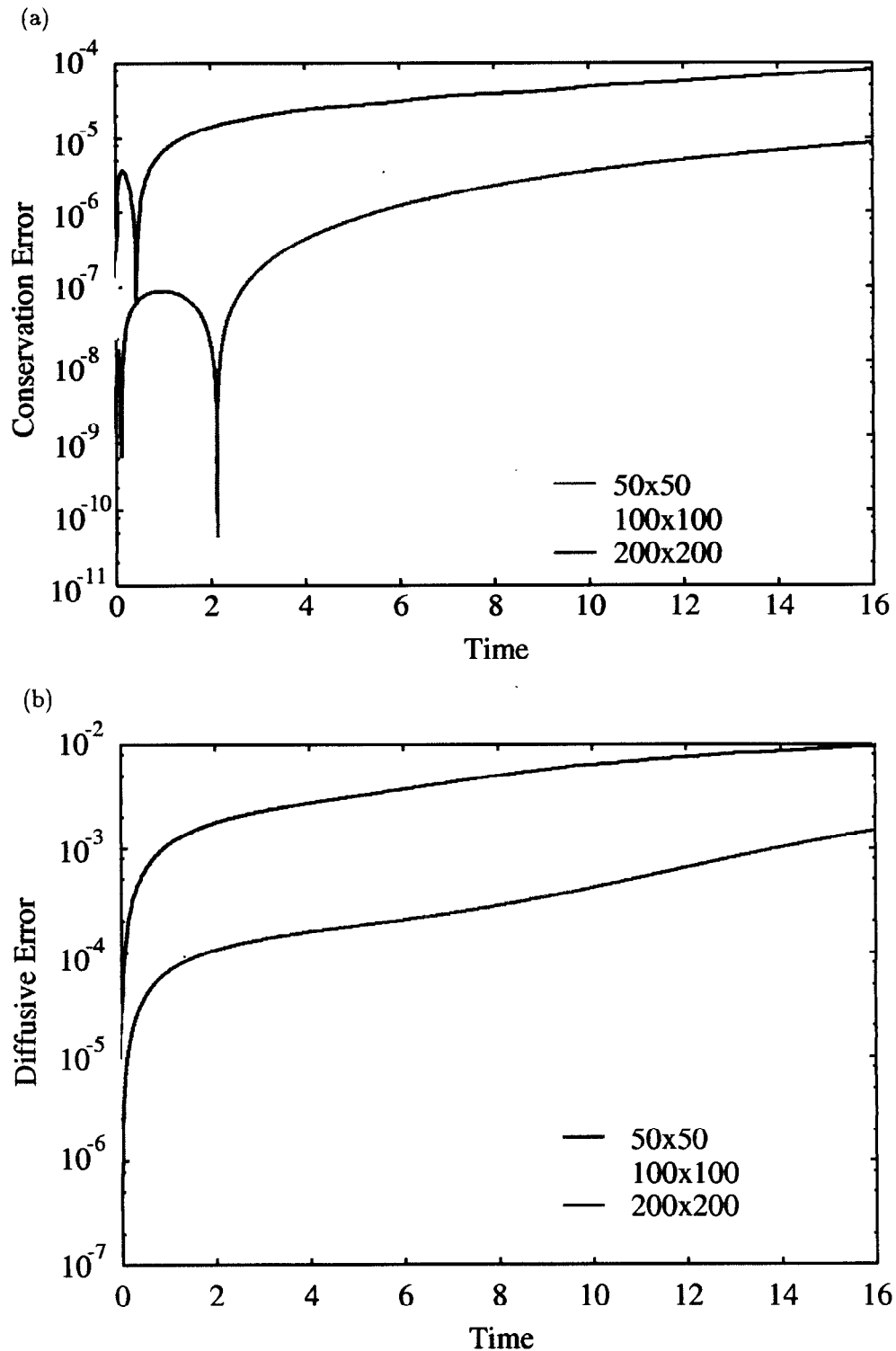


図 3.20: 前線発達問題の (a) 保存誤差ならびに (b) 数値拡散誤差  
IDO 法は単純な移流計算に対して  $\Delta x^3$  の空間精度を示すが、さまざまな波数成分が継続的に発生する本問題に対しては空間精度の低下が見られた。

ハイパボリック・タンジェントにより与えられた初期プロファイル中にはさまざまな波数成分が含まれて



おり、計算初期段階ではこれらの減衰が生じるとともに計算が進行するにしたがって微細構造が発達する事からその際に高波数成分が継続的に生じる。単純な移流計算に対しては IDO 法は  $\Delta x^3$  の空間精度を示すが、こうした継続的な高波数成分の生成が計算精度を低下させているものと考えられる。保存誤差に関しては計算格子点数に依存して減少するものの、グリッド数が増加すると単調な減少傾向は得られなかった。周期境界条件を適用しなかったことがその原因として挙げられる。また、数値拡散誤差は  $t = 10$  近傍を超えると次第に増加する傾向があるが、これは渦中心の流体構造が複雑化するため IDO 法を用いた場合でも表現能力が不足し、数値拡散が生じるためと考える。

### 3.3.3 まとめ

解析解の存在する理想化された前線発達 (Frontogenesis) 問題を扱うことで保存誤差ならびに解析解からの数値拡散誤差を見積もった結果、保存誤差は計算格子点数依存があまり見られず、数値拡散誤差は  $\Delta x^2$  の精度を示すことを明らかにした。この結果から、IDO 法による数値計算は単純な一様速度場における移流方程式等に対しては  $\Delta x^3$  の精度を示すものの、継続的に高波数成分が生じる問題に対しては空間精度が低下する可能性を明らかにした。

ただし、IDO 法を用いることで  $50 \times 50$  の計算格子点数であっても複雑な流体構造を十分に高精度で扱うことができ、今回扱った範囲内では保存誤差に関しては  $10^{-4}$  以下、数値拡散誤差は最大で 1% 程度と良好な結果を得ることに成功した。

### 3.4 レーリー・テラー不安定性への適用

圧縮性流体問題へのIDO法の適用例として複雑な流体構造を示すレーリー・テラー不安定性を扱う。この問題は非圧縮性流体問題として扱うことも可能であるが、圧縮性流体の解法を用いた数値計算により解を得ることが出来る。ただし、現象が流体不安定性を見せることから初期条件ならびに数値誤差により大幅に計算結果が異なることが予想される。IDO法は先に示した1次元衝撃波管問題の結果からも示されたように高解像度計算グリッドを利用した場合、計算精度が大きく改善されることから数値拡散により消滅していたプロファイル成分が支配方程式の非線形性により増大し、不必要に数値誤差が発生する問題がある。このような数値誤差の影響を考察すると共にレーリー・テラー不安定性の持つ細かい流体構造を表現するために、初期プロファイル、擾乱について検討を行う。

#### 3.4.1 レーリー・テラー不安定性の概要

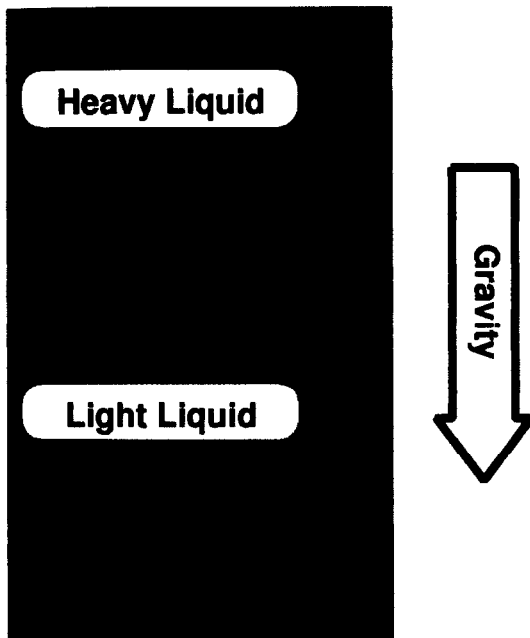


図 3.21: レーリー・テラー不安定性の初期プロファイルモデル

図 3.21 に示すように重力場において重い流体が軽い流体の上部に位置する場合、全く擾乱が存在しなければ流体は静的なつりあいを保つ。しかし、流体中に擾乱が発生した場合、不安定性により流体は混合を開始する。レーリー・テラー不安定性 (Rayleigh-Taylor Instability, Rayleigh-Taylor unstable) と呼ばれるこの現象は混合プロセスの過程でマッシュルーム型の複雑な流体構造を見せることからこの問題は物理学分野で長年にわたり研究が続けられている [19, 20, 21, 22]。近年研究が推められている慣性核融合 (Internal Confinement Fusion; ICF) の分野では、燃料ペレット爆縮過程においてレーリー・テラー不安定性が生じ、均一な圧縮が困難になることからレーリー・テラー不安定性に対する現象の理解が重要な課題となっている。また、Super Nova 1987A における物質の混合過程の一つとしてレーリー・テラー不安定性がとりあげられており、数値流体計算による現象の解明も進んでいる。

本節ではIDO法のレーリー・テラー不安定性問題への適用を行い、その成長速度の議論を行うことで、IDO法のレーリー・テラー不安定性への適用性について検討する。

#### 3.4.2 2次元レーリー・テラー不安定性

##### 基礎方程式

IDO法による2次元圧縮性流体の陽解法を用いた。人工粘性を考慮しないため、密度 $\rho$ 、速度 $u, v$ 、内部エネルギー $e$ 、圧力 $P$ を関係付ける基礎方程式群は以下ようになる。

$$\rho_t = -u\rho_x - v\rho_y - \rho(u_x + v_y) \quad (3.25)$$

$$u_t = -uu_x - vu_y - \frac{P_x}{\rho} \quad (3.26)$$

$$v_t = -uv_x - vv_y - \frac{P_y}{\rho} \quad (3.27)$$

$$e_t = -ue_x - ve_y - \frac{P(u_x + v_y)}{\rho} \quad (3.28)$$

$$P = (\gamma - 1)\rho e \quad (3.29)$$

ここで、 $\gamma$  は比熱比をあらわし、1.4 と与えた。ただし、重力を  $x$  方向に対して付加するため、 $u$  に関する偏微分方程式は次のようになる。

$$u_t = -uu_x - vu_y - \frac{P_x}{\rho} + g \quad (3.30)$$

計算中では  $g = 1$  とした。

### 初期密度プロファイル

計算結果の検討に先立ち、初期プロファイルの妥当性について示す。レーリー・テラー不安定性では初期条件として、重力方向に比重が小さくなるように流体を配置する。

$$\frac{\partial P}{\partial x} = \rho g \quad (3.31)$$

$$P = \int \rho g dx \quad (3.32)$$

を満足するように初期圧力を決定することから、初期に擾乱が存在しなければ流体重力とのバランスを保つことで静止状態を持続する。レーリー・テラー不安定性を数値計算で検討する場合、密度プロファイルとして流体の境界面上下で一様に与える。しかし、IDO 法により同様のプロファイルを扱う場合、初期状態での静止を保持するためには単純に式 (3.32) により与えることが出来ない。そこで、以下の手法により圧力場を決定する手法を提案する。

一様な密度プロファイル中 ( $\rho(x) = \rho_0$ ) での運動方程式について考える。速度擾乱が完全に存在しない場

$$u(x, y) = 0, \quad v(x, y) = 0 \quad (3.33)$$

に対して、流体が静止を保持するためには、式 (3.30) で示される重力を付加したナビエ・ストークス方程式において、 $\partial u / \partial t = 0$  を保持する必要がある。 $u = 0, u_x = 0$  を用いるとその関係式は式 (3.31) で示されることがわかる。つまり、空間に対する圧力は式 (3.32) で示され、重力方向に対して空間上部での初期圧力を  $P_0$  であらわすと、圧力分布は単純に

$$P = \rho_0 g x \quad (3.34)$$

であらわされる。ここで、重力方向を  $x$  軸の正方向とした。IDO 法では、従属変数として 1 階空間微係数を各グリッド上に定義することから、 $u_x$  の時間発展についても考慮しなければならない。 $u_x$  が 0 でなければ、エネルギー保存式が時間発展し、結果として圧力擾乱を引き起こすため、 $u_{tx}$  もまた 0 でなければならない。式 (3.30) を、空間微分した式

$$u_{tx} = -uu_{xx} - u_x u_x - vu_{xy} - v_x u_y - \frac{P_{xx}}{\rho} + \frac{P_x \rho_x}{\rho^2} \quad (3.35)$$

が、静的つりあいを保つためには

$$P_{xx} = \frac{P_x \rho_x}{\rho} \quad (3.36)$$

を満足しなければならない。今、 $\rho_x = 0$  であることから、 $P_{xx}$  の見積もりを行う。簡単のため、座標系  $x = i\Delta x$  を用いる。ここで、 $i$  はグリッド番号をあらわす。IDO 法の 5 次中心補間式から、

$$P_{xx} = \frac{P_{i+1} - 2P_i + P_{i-1}}{\Delta x^2} - \frac{f_{x,i+1} - f_{x,i-1}}{4\Delta x} \quad (3.37)$$

$$= \frac{\rho_0 g(i+1)\Delta x - 2\rho_0 g i\Delta x + \rho_0 g(i-1)\Delta x}{\Delta x^2} - \frac{\rho_0 g - \rho_0 g}{4\Delta x} \quad (3.38)$$

$$= 0 \quad (3.39)$$

が求められ、一様な密度プロファイルに対しては静的なつりあいを保持することがわかる。つぎに、図 3.22 に示すような不連続面位置におけるつりあいを考える。

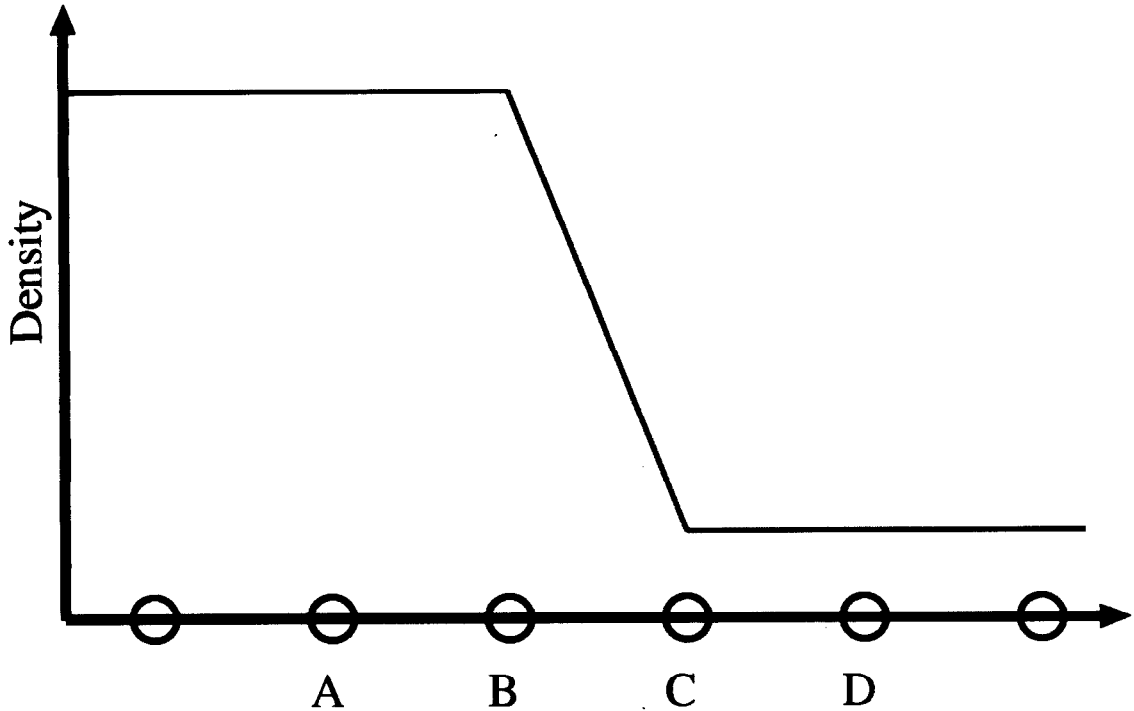


図 3.22: 不連続密度プロファイルにおける静的釣合

点 A までの流体の静的つりあいはこれまでの一様密度プロファイルに対する安定性の検討により保証されている。ここからは高密度側の密度を  $\rho_H$ 、低密度側の密度を  $\rho_L$  であらわす。点 B, C における速度の静的つりあい条件から、

$$P_x|_B = \rho_H g, \quad (3.40)$$

$$P_x|_C = \rho_L g, \quad (3.41)$$

が得られ、また、 $u_x$  の静的つりあい条件から、

$$P_{xx}|_B = \frac{\rho_x P_x}{\rho} = \rho_x|_B g, \quad (3.42)$$

$$P_{xx}|_C = \frac{\rho_x P_x}{\rho} = \rho_x|_C g, \quad (3.43)$$

が得られる。中心 5 次補間から得られる空間 2 階微係数の離散式を代入することで、以下の関係式が得られる。

$$2 \left( \frac{P_C - 2P_B + P_A}{\Delta x^2} - \frac{\rho_L g - \rho_H g}{4\Delta x} \right) = \rho_x|_B g \quad (3.44)$$

$$2 \left( \frac{P_D - 2P_C + P_B}{\Delta x^2} - \frac{\rho_L g - \rho_H g}{4\Delta x} \right) = \rho_x|_C g \quad (3.45)$$

点 A, B 間ならびに点 C, D 間はそれぞれ一定密度  $\rho_H, \rho_L$  で満たされていることから、 $P_B, P_D$  は、

$$P_B = P_A + \rho_H g \Delta x, \quad (3.46)$$

$$P_D = P_C + \rho_L g \Delta x \quad (3.47)$$

と求めることが出来る。式 (3.46)、式 (3.47) をそれぞれ式 (3.44)、式 (3.45) に代入することで、

$$2 \left( \frac{P_C - P_A}{\Delta x^2} - \frac{7\rho_H g + \rho_L g}{4\Delta x} \right) = \rho_x|_B g \quad (3.48)$$

$$2 \left( \frac{-P_C + P_A}{\Delta x^2} + \frac{5\rho_H g + 3\rho_L g}{4\Delta x} \right) = \rho_x|_C g \quad (3.49)$$

が得られ、密度の微係数に関する関係式

$$\rho_x|_B + \rho_x|_C = \frac{\rho_L - \rho_H}{\Delta x} \quad (3.50)$$

が得られる。密度の微係数に関する拘束条件は他に存在しないため、式 (3.50) を満たす限り  $\rho_x|_B, \rho_x|_C$  を任意に与えることが出来るが、ここでは以下のように与える。

$$\rho_x|_B = \rho_x|_C = \frac{\rho_L - \rho_H}{2\Delta x} \quad (3.51)$$

また式 (3.48)、式 (3.49) から、

$$P_C = P_A + \frac{\rho_x|_B g \Delta x^2}{2} + \frac{7\rho_H + \rho_L}{4} g \Delta x, \quad (3.52)$$

$$P_C = P_A - \frac{\rho_x|_C g \Delta x^2}{2} + \frac{5\rho_H + 3\rho_L}{4} g \Delta x \quad (3.53)$$

が得られ、 $P_C$  は

$$P_C = P_A + \frac{3\rho_H + \rho_L}{2} g \Delta x \quad (3.54)$$

と求められる。

上記手法は、1 階の時間微係数のみに対する静的つりあい条件により圧力等を決めているが、IDO 法は時間 3 次微係数までを用いたテーラー展開に基づき支配方程式の時間発展を行うため、より複雑な関係式を扱うべきである。しかし、上記手法により初期プロファイルを与え、静的つりあいについて検討したところ、時間 3 次微係数までを扱った場合でも十分に流体を静止させることが出来た。

不連続初期プロファイルを用いて得られた数値計算結果は、後に示すように数値粘性を付加しなければ数値発散する結果となった。空間微係数の急激な不連続が一つの原因として考えられるが、数値粘性を加えることでレーリー・テーラー不安定性から成長する流体のマッシュルーム構造の詳細が数値拡散してしまう。そこで、密度不連続を誤差関数を用いて表現することで、空間微係数の不連続点を除く手法を提案する。

図 3.23 に、誤差関数

$$\rho = a \operatorname{erf}(b(c-x)) + d \quad (3.55)$$

$$p = 0.1 + \int \rho dx \quad (3.56)$$

$$\begin{aligned} &= -a \frac{\exp(-b^2(c-x)^2)}{b\pi} - d(c-x) - a(c-x) \operatorname{erf}(b(c-x)) \\ &\quad + \frac{a \exp(-b^2 c^2)}{b\pi} + d \times c + a \times c \operatorname{erf}(b \times c) + 0.1 \end{aligned} \quad (3.57)$$

$$a = 0.35, \quad c = 0.3, \quad d = 0.65 \quad (3.58)$$

を用いた密度プロファイルを示す。 $a, c, d$  はそれぞれ  $x = 0.3$  において境界面を持ち、流体の密度がそれぞれ 0.3, 1.0 となるように定めた。流体間の密度勾配を  $b$  を変化させることでより急勾配の密度プロファイルを与えることが出来る。 $b \leq 41$  では安定にプロファイルは静止状態を保つことが可能であるものの、 $b > 41$  では、数値不安定性が生じた。これは、流体間の密度勾配を急峻になるとプロファイル中に高波数成分を持つため、プロファイル記述精度が低下したことが原因と考えられる。

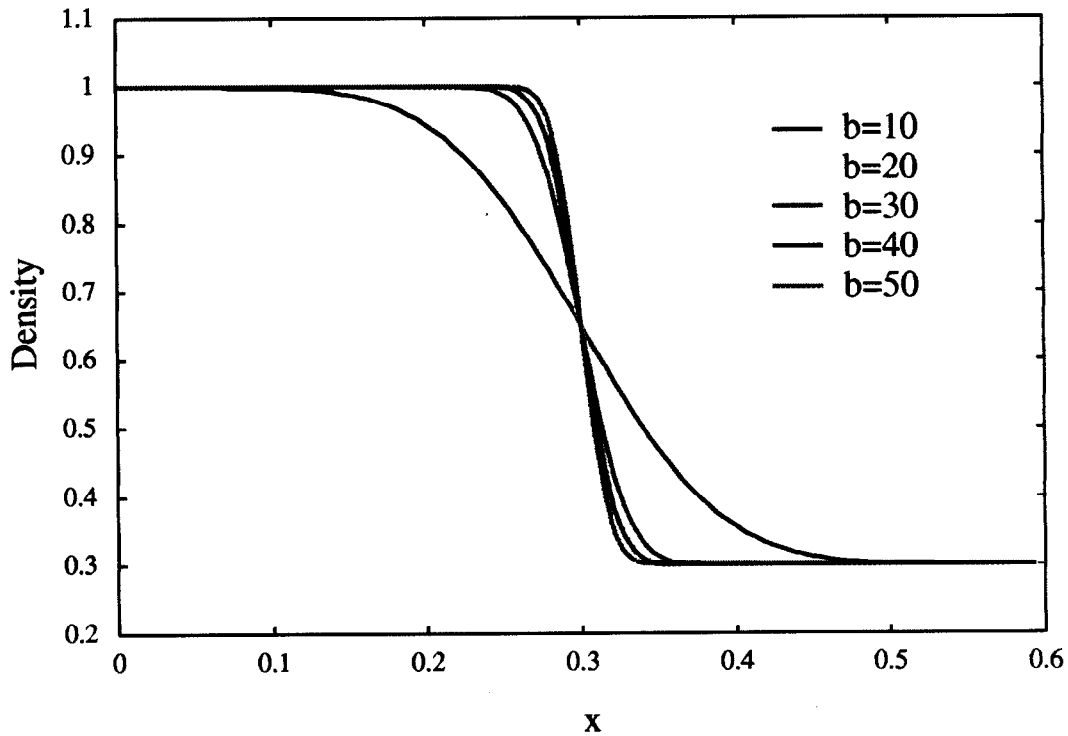


図 3.23: 2次元レーリー・テラー不安定性のための初期密度プロファイル

### 初期擾乱に対する考察

レーリー・テラー不安定性は、初期プロファイルに対して何らかの擾乱を与えることにより流体運動を開始し、不安定性の成長が観測される。その基礎的な振舞いを検討するため、1つのマッシュルーム構造が成長する様子を捕らえることを目的とし、シングルモードを誘発する擾乱をここでは議論する。

シングルモードが成長するためには、重力方向と垂直方向に対して擾乱を1波長与える必要がある。このことからもっとも単純な擾乱として三角関数を用いた以下の擾乱を考える。

$$u = C_a \cos(k_x x) \cos(k_y y) \quad (3.59)$$

$$u_x = -C_a k_x \sin(k_x x) \cos(k_y y) \quad (3.60)$$

$$u_y = -C_a k_y \cos(k_x x) \sin(k_y y) \quad (3.61)$$

$$u_{xy} = C_a k_x k_y \sin(k_x x) \sin(k_y y) \quad (3.62)$$

$$v = C_b k_x \sin(k_x x) \sin(k_y y) \quad (3.63)$$

$$v_x = C_b k_x^2 \cos(k_x x) \sin(k_y y) \quad (3.64)$$

$$v_y = C_b k_x \sin(k_x x) \cos(k_y y) \quad (3.65)$$

$$v_{xy} = C_b k_x^2 k_y \cos(k_x x) \cos(k_y y) \quad (3.66)$$

ここで、 $x$  は境界面上で0となるように定めた重力方向の座標であり、 $y$  は、それに垂直方向に定めた座標である。また、 $\nabla u = 0$ となるように  $C_a = k_x C_b$  として擾乱を与える。境界面上に集中的に重力方向成分に対する擾乱を与えることを目的として、 $k_x = \pi/l_x$ とした。ここで、 $l_x$  は、 $x$ 方向の計算領域長さをあらわす。この擾乱を用いて計算した結果の一例を図3.24に示す。



図 3.24: 三角関数を用いた擾乱によるレーリー・テラー不安定性の発達  
 計算空間全体に渡って擾乱が与えられるため、マッシュルーム構造が発達しにくい。(a)-(e) はそれぞれ  $t = 0, 0.2, 0.6, 1.0, 1.4$  における密度プロファイル。

速度擾乱が計算空間全体に渡って与えられているため、マッシュルーム構造が成長しにくい結果となった。そこで、境界面付近により集中して速度擾乱を与えるために、指数関数を用いた速度擾乱を考える。

$$\begin{cases} u = -a \cos(k_y y) \exp(bx) & (if x < 0) \\ v = -ab/k_y \sin(k_y y) \exp(bx) & \\ \\ u = -a \cos(k_y y) \exp(-bx) & (if x \geq 0) \\ v = ab/k_y \sin(k_y y) \exp(-bx) & \end{cases} \quad (3.67)$$

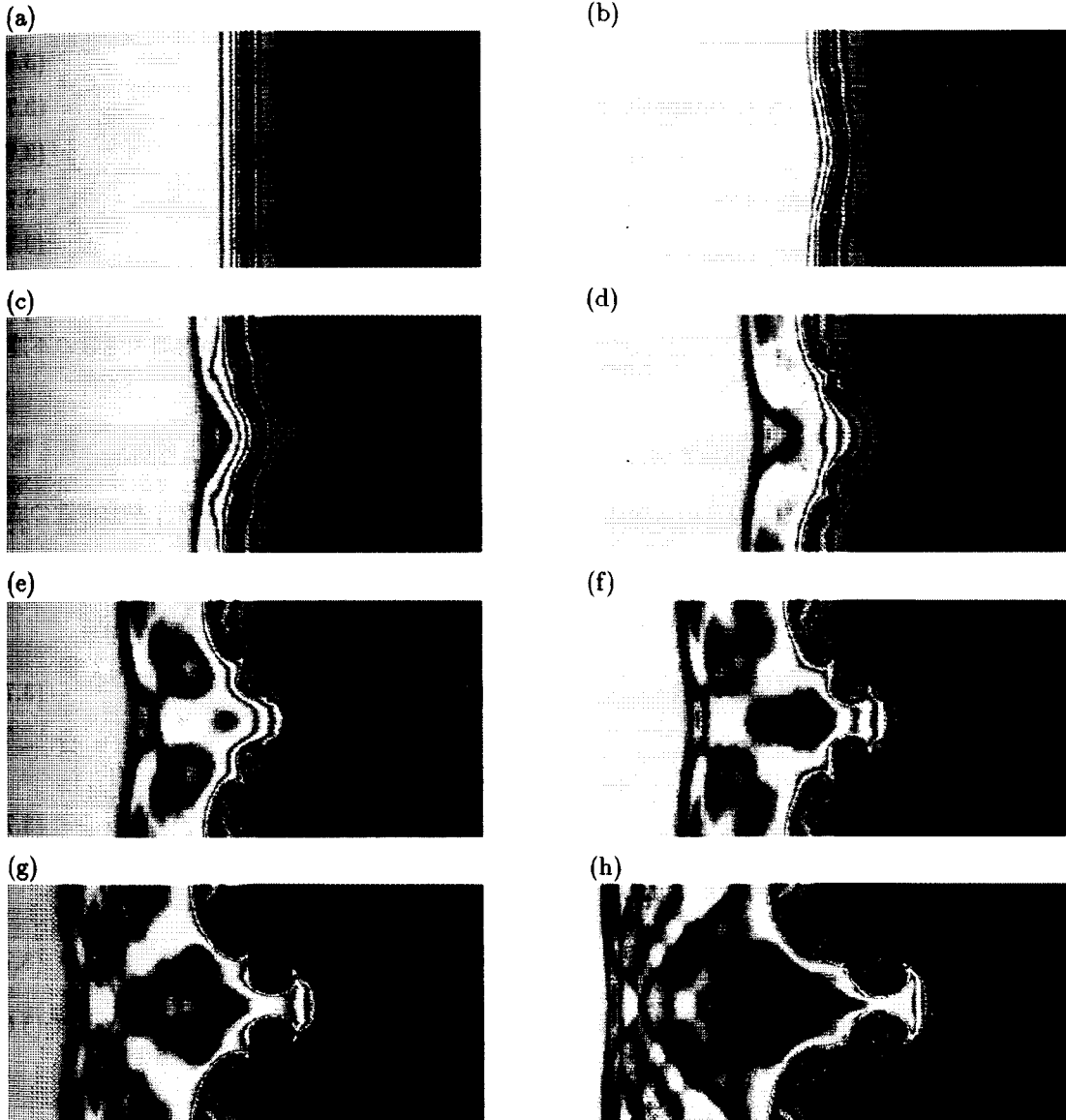


図 3.25: 指数関数を用いた擾乱によるレーリー・テラー不安定性の発達指数関数により速度擾乱を与えた場合はマッシュルーム構造の発達が観測されたが、シングルモードでの擾乱を与えているにも関わらず、2 次的な渦の成長が観測された。(a) - (h) はそれぞれ  $t = 0, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.7$  における密度プロファイルを示す。



式 (3.67) で記述される速度擾乱を用いることで、境界面近傍に集中的に擾乱を与えることが出来る。この擾乱を用いた数値計算結果の一例を図 3.25 に示す。マッシュルーム構造を持つ成長が観測されたが、シングルモードでの速度擾乱を与えたにも関わらず 2 次的な渦の発達を確認できた。その原因として、境界面を境に速度ならびにその微係数の不連続が存在することが考えられる。

そこで、新たに次に示す速度擾乱を提案する。

$$u = C_a \cos(k_y y) \exp(-k_x x^2) \quad (3.68)$$

$$u_x = -2C_a k_x x \cos(k_y y) \exp(-k_x x^2) \quad (3.69)$$

$$u_y = -C_a k_y \sin(k_y y) \exp(-k_x x^2) \quad (3.70)$$

$$u_{xy} = -2C_a k_x k_y \sin(k_y y) \exp(-k_x x^2) \quad (3.71)$$

$$v = C_b x \sin(k_y y) \exp(-k_x x^2) \quad (3.72)$$

$$v_x = C_b \sin(k_y y) \exp(-k_x x^2) - 2C_b k_x x^2 \cos(k_y y) \exp(-k_x x^2) \quad (3.73)$$

$$v_y = -C_b k_y x \cos(k_y y) \exp(-k_x x^2) \quad (3.74)$$

$$v_{xy} = -C_b \cos(k_y y) \exp(-k_x x^2) + 2C_b k_x x^2 \cos(k_y y) \exp(-k_x x^2) \quad (3.75)$$

ここで、 $C_a, C_b$  は、 $u_x + v_y = 0$  を満たすように  $2C_a k_x = C_b k_y$  を満足するように与える。式 (3.68) - 式 (3.75) を用いることで、初期擾乱を境界面付近に局所的に、かつ速度不連続、速度の空間微係数の不連続のない擾乱を与えることが可能になる。

## 2 次元レーリー・テラー不安定性の成長速度

式 (3.68) - 式 (3.75) における速度擾乱の大きさ  $C_a, C_b$  を変化させ、レーリー・テラー不安定性の成長速度について議論する。

低密度側の流体密度を  $\rho_1$ 、高密度側の流体密度を  $\rho_2$  とした場合、表面張力を持たない非粘性流体において線形密度擾乱

$$\exp\{i(k_y y + k_z z) + nt\} \quad (3.76)$$

の成長率  $n$  は、

$$n^2 = gk \left( \frac{\rho_2 - \rho_1}{\rho_2 + \rho_1} \right) \quad (3.77)$$

であらわされる [19, 20]。ここで、 $k_y, k_z$  はそれぞれ重力に対して垂直方向の速度擾乱の波長であり、

$$k^2 = k_y^2 + k_z^2 \quad (3.78)$$

で与えられる。また、 $g$  は重力加速度を示している。単純なモード (Single Mode) を選択するため、式 (3.68) - 式 (3.75) における  $k_y$  を計算空間中で 1 波長の速度擾乱が付加されるように、

$$k_y = \frac{2\pi}{l_y} \quad (3.79)$$

とした。ここで、 $l_y$  は、重力方向に対して垂直方向の計算領域の長さである。

はじめに、計算空間  $0 \leq x \leq 1.2, 0 \leq y \leq 0.3, \Delta x = 1.2/120, \Delta y = 0.3/30, \Delta t = 0.1\Delta x$  を用いて初期密度プロファイルを不連続に与えた場合の擾乱の成長速度について議論する。高密度側の密度を 1、低密度側を 0.3 とし、式 (3.34) - 式 (3.54) を用いてその他の初期プロファイルを与え、速度擾乱を式 (3.68) - 式 (3.75) を用いて  $k_x = 1/(100 \times 1.2), k_y = 2\pi/0.3, C_a = 0.3, 0.6$  とした。

計算空間中心における密度擾乱の成長を検討した結果をそれぞれ図 3.26 に示す。

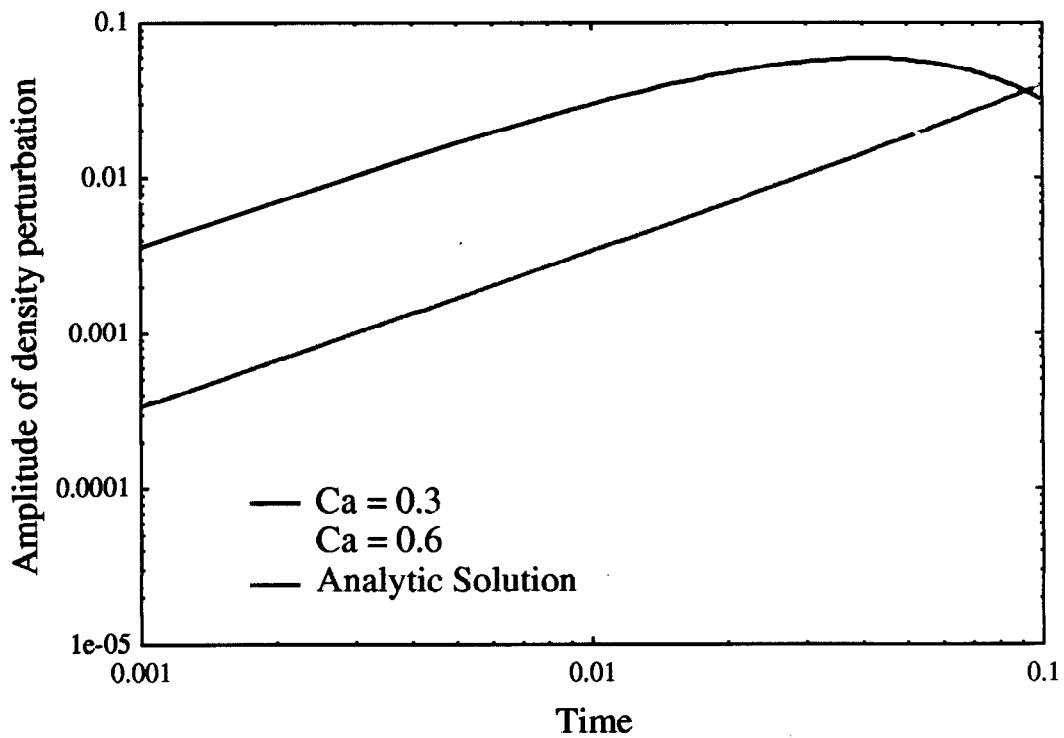


図 3.26: 不連続プロファイルを用いた場合の密度擾乱の成長率  
解析解から得られる密度擾乱の成長率(傾き)と、計算結果から得られた成長率は線形成長が観測される範囲内で良く一致した結果を得た。

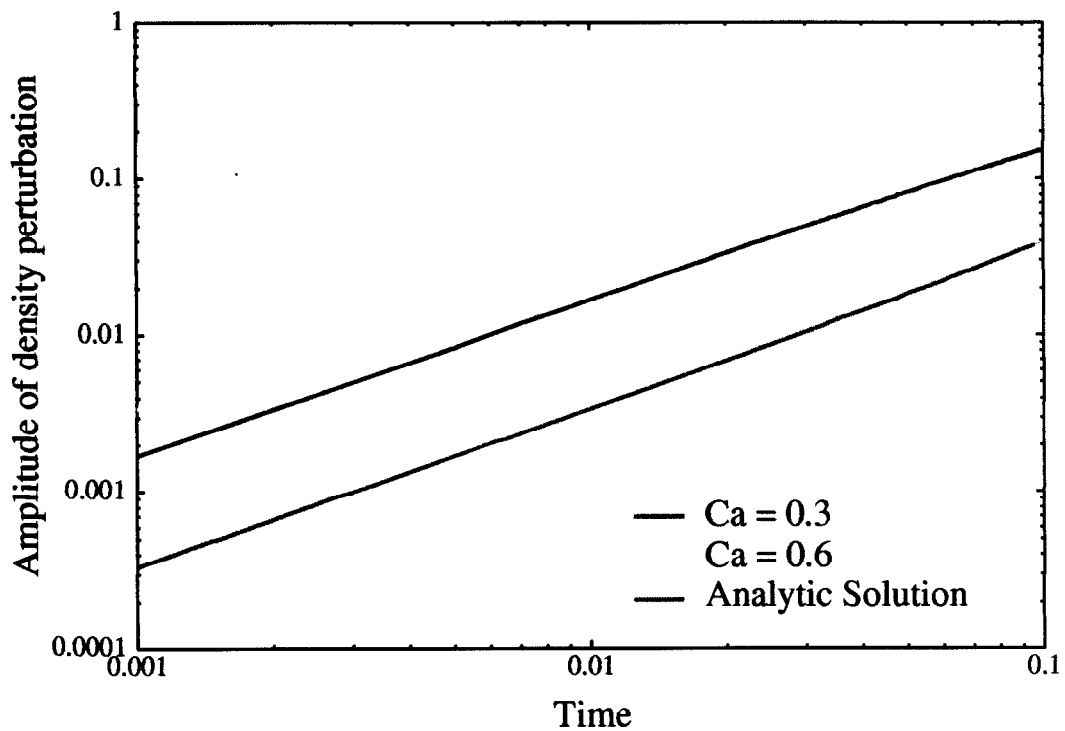


図 3.27: 誤差関数を用いたプロファイルを用いた場合の密度擾乱の成長率  
解析解から得られる密度擾乱の成長率(傾き)と、計算結果から得られた成長率は線形成長が観測される範囲内で良く一致した結果を得た。

解析解から得られる成長率(グラフの傾き)と、数値計算結果から得られた擾乱の成長率は良く一致していることがわかる。

つぎに、同じ計算空間、初期擾乱を用いて誤差関数を用いた密度プロファイル

$$\rho = a \operatorname{erf}(b(c-x)) + d \quad (3.80)$$

$$p = 0.1 + \int \rho dx \quad (3.81)$$

$$a = 0.35, \quad b = 40, \quad c = 0.3, \quad d = 0.65 \quad (3.82)$$

に対し、同様に密度擾乱の成長率を検討した。

図 3.27 に示した検討結果から、誤差関数により初期密度プロファイルを与えた場合でも擾乱の成長率は解析解に良く一致する結果を得た。

以上から、IDO 法を用いてレーリー・テラー不安定性の数値計算を行った場合、線形擾乱の成長率に関しては初期密度プロファイルを不連続に与えた場合でも誤差関数を用いてなだらかに与えた場合でも解析解に良く一致することを示した。

### レーリー・テラー不安定性の流体構造

レーリー・テラー不安定性で見られる流体のマッシュルーム構造について議論することを目的として、計算空間  $0 \leq x \leq 1.2, 0 \leq y \leq 0.3, \Delta x = 0.6/240, \Delta y = 0.3/60$  に、高密度側密度 1、低密度側密度 0.3 とした不連続初期密度プロファイルを与え、式 (3.68) - 式 (3.75) に示した初期速度擾乱を用いて流体計算を行った結果を示す。初期速度擾乱のパラメータは、 $k_x = 100/1.2, k_y = 2\pi/0.3, C_a = 0.6$  と与えた。また、計算時間刻幅を  $\Delta t = 0.1\Delta x$  とした。

図 3.28 (a) - (f) はそれぞれ時刻  $t = 0, 0.1, 0.2, 0.4, 0.6, 0.8, 1.0$  における密度プロファイルを示した。右方向に  $x$  軸の正方向をとっており、重力加速度  $g = 1$  を導入している。左側に配置された高密度流体が次第に低密度流体側に浸透して行き、マッシュルーム構造を形成している様子がわかる。しかし、その過程で数値不安定性が生じた。

1 次元衝撃波管問題でも行ったように、IDO 法での数値計算に対してわずかに数値粘性を加えることで数値不安定性を低減できることから、同様の手法による安定化を考える。 $\kappa = 10^{-4}$  を用いた拡散項を Navier-Stokes 方程式に対して加え、先と完全に同じ計算条件下で行った計算結果を図 3.29 に示す。(a)-(f) はそれぞれ  $t = 0.25, 0.5, 0.75, 1.0, 1.25, 1.5$  における密度プロファイルを表している。数値粘性を加えることにより計算が安定化し、マッシュルーム構造の成長が観測された。しかし、非粘性流体問題としてレーリー・テラー不安定性は議論されていることから、可能な限り数値粘性を含めない計算を行うことが望ましい。そこで、先に提案した誤差関数を用いた初期密度プロファイルによる数値計算を検討する。

式 (3.80) - 式 (3.82) に示される誤差関数を用いた初期密度プロファイルを計算空間  $0 \leq x \leq 1.2, 0 \leq y \leq 0.3, \Delta x = 0.6/240, \Delta y = 0.3/60$  に与え、式 (3.68) - 式 (3.75) に示した初期速度擾乱を用いて計算時間刻幅  $\Delta t = 0.1\Delta x$  により流体計算を行った結果を示す。初期速度擾乱のパラメータは、 $k_x = 100/1.2, k_y = 2\pi/0.3, C_a = 0.6$  と与えた。

図 3.30 に示した計算結果から、誤差関数を用いた初期密度プロファイルを用いることで、数値粘性を加えずに安定にマッシュルーム構造の成長を計算できた。しかし、計算時間が進行するにつれて圧縮領域に数値不安定が見られる。これは、支配方程式を非保存系で扱っているため、人工粘性を適用しなければ圧縮領域を正しく解くことが出来ないことに起因している。しかし実際に衝撃波が生じているわけではなく、人工粘性を加えた計算結果から期待するものが得られなかった。

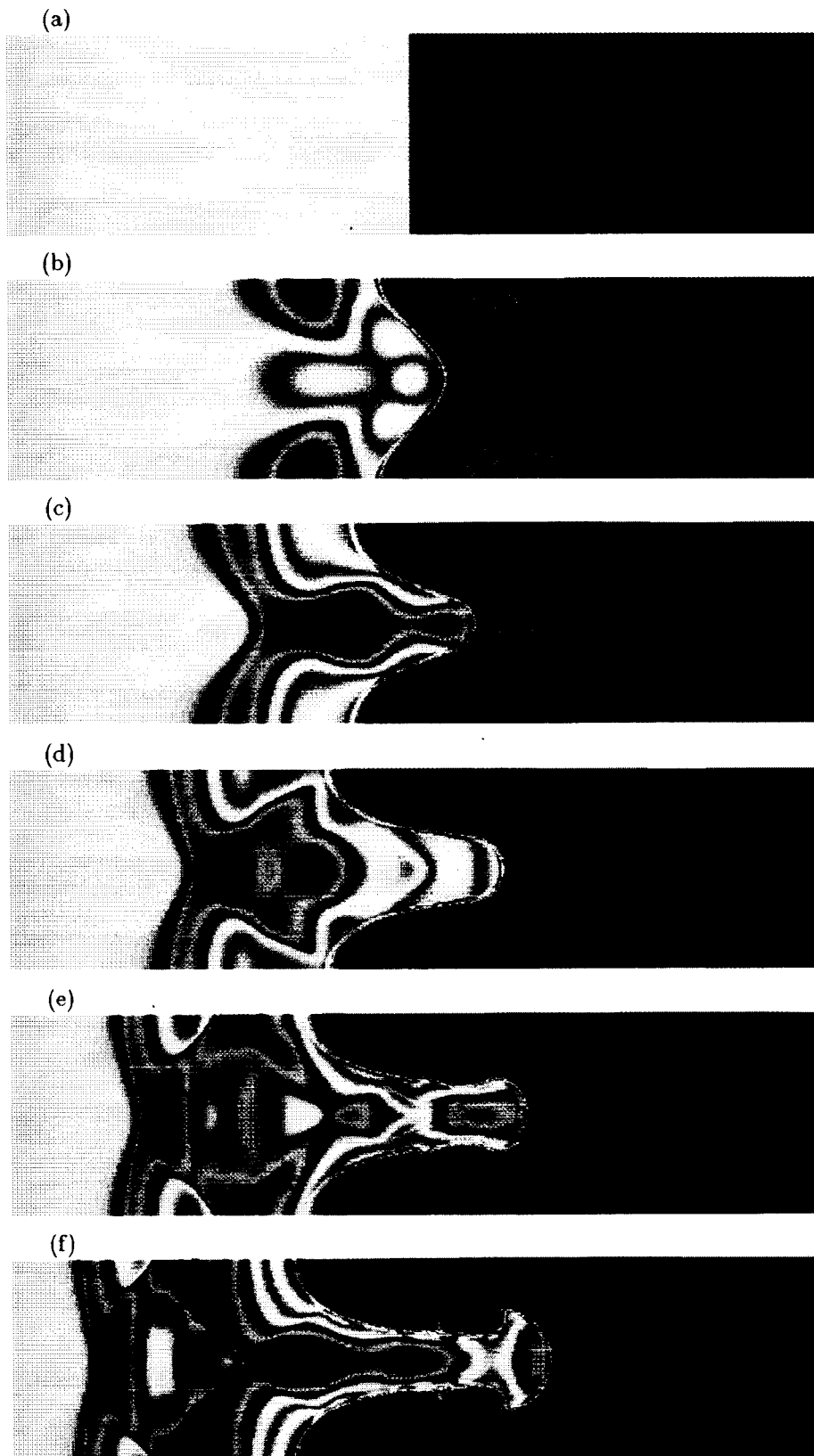


図 3.28: 不連続初期密度プロファイルによるレーリー・テラー不安定性の数値計算  
(a) - (h) はそれぞれ  $t = 0, 0.2, 0.4, 0.6, 0.8, 1.0$  における密度プロファイルを示す。成長の過程で数値不安定性が生じる。

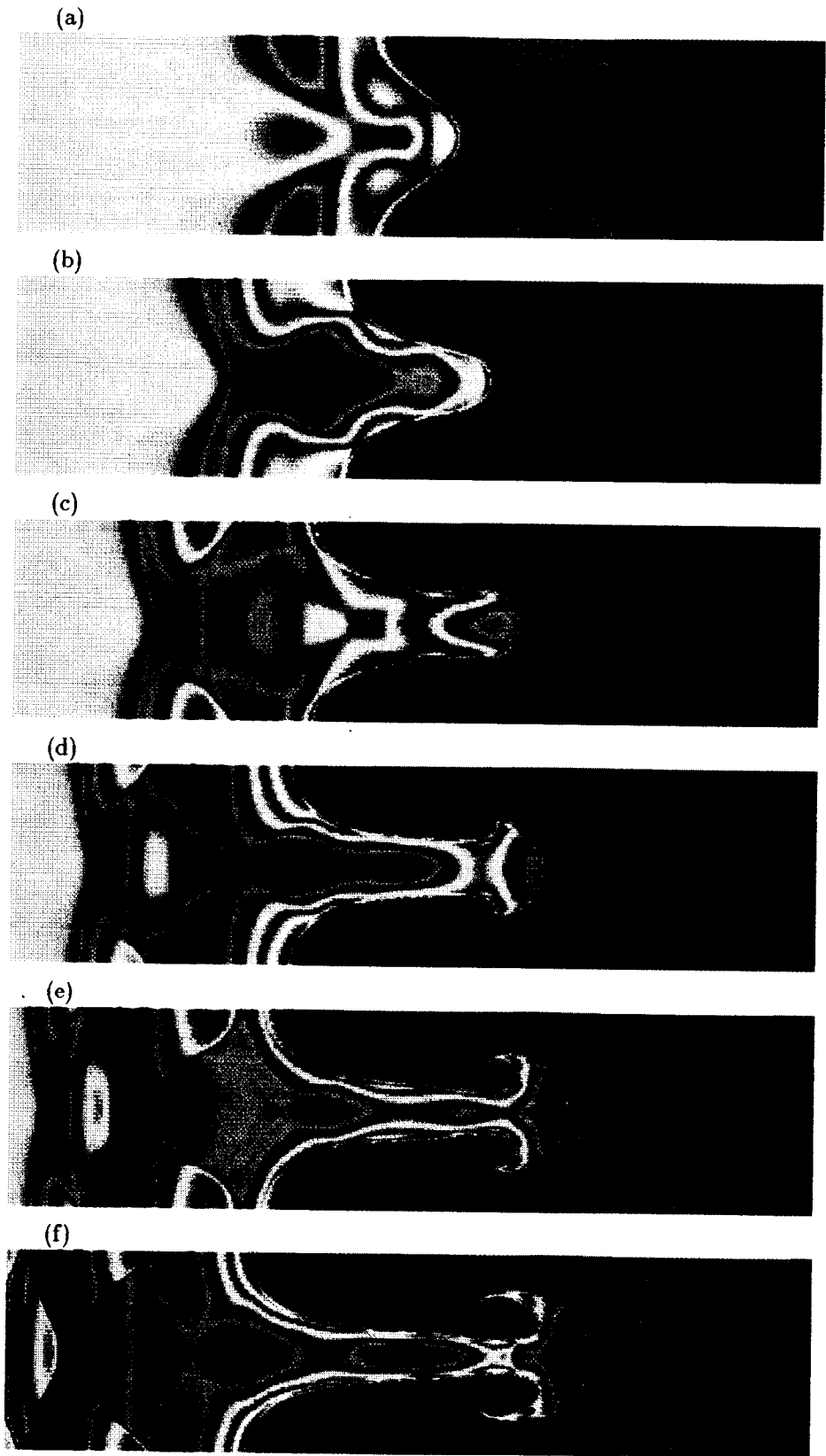
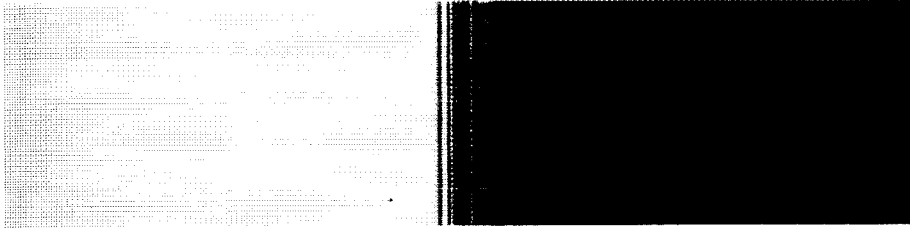


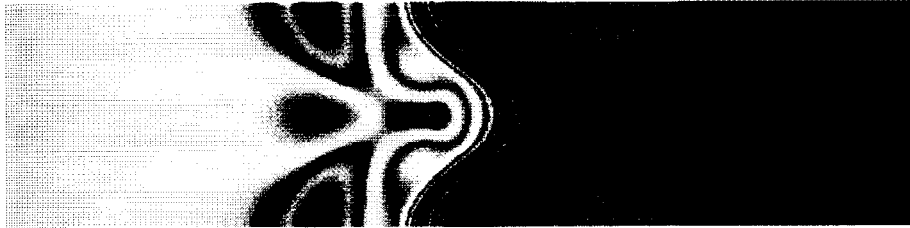
図 3.29: 数値粘性を付加した不連続初期密度プロファイルによるレーリー・テラー不安定性の数値計算

(a) - (h) はそれぞれ  $t = 0.25, 0.5, 1.5, 1.0, 1.25, 1.5$  における密度プロファイルを示す。数値粘性を付加することで数値不安定性を低減できる。

(a)



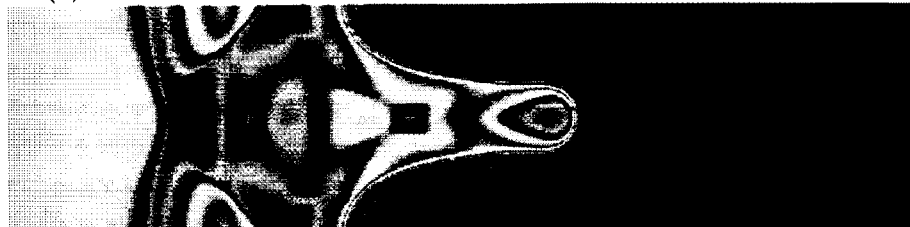
(b)



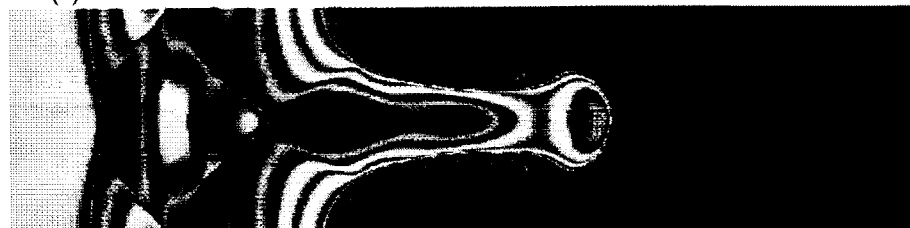
(c)



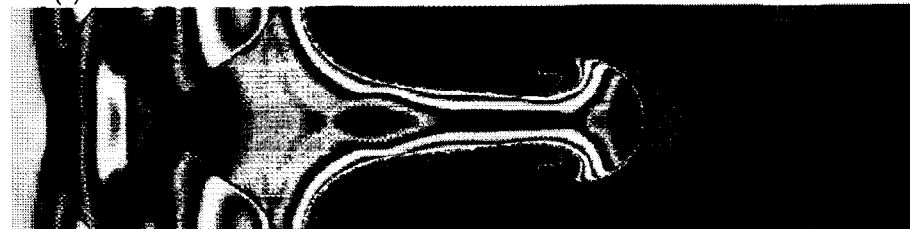
(d)



(e)



(f)



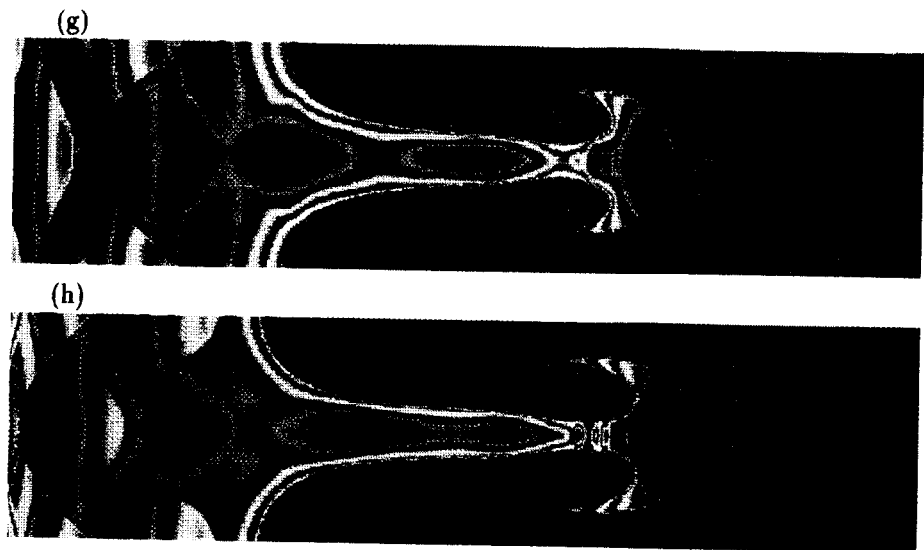


図 3.30: 誤差関数による初期密度プロファイルを用いたレーリー・テラー不安定性の数値計算  
(a) - (h) はそれぞれ  $t = 0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4$  における密度プロファイルを示す。数値粘性を付加しなくても比較的安定に数値計算が可能になる。

## 2 次元レーリー・テラー不安定性のまとめ

IDO 法で 2 次元レーリー・テラー不安定性を扱うための初期密度プロファイルとして不連続境界を与えるための手法を提案し、十分に流体が静的条件を満足することに成功した。その一方で不連続境界を持つ初期密度プロファイルでの流体計算は数値粘性を含めた計算を必要とすることから、数値粘性を必要としない誤差関数による初期密度プロファイルを併せて提案し、流体安定性を満足することを確認した。

レーリー・テラー不安定性を誘発する初期擾乱として速度擾乱を用いることを考え、 $\nabla u = 0$  を満足する 3 タイプの初期擾乱を提案し、その中でもレーリー・テラー不安定性のシングルモードが安定にかつマッシュルーム構造がすみやかに成長する擾乱を指数関数を用いることで満足することに成功した。

初期密度プロファイルとして不連続関数、誤差関数を用いた両者に対して、線形擾乱の成長を解析解と比較した結果、現象が線形成長する間は解析解に一致する成長率を見せることを示した。

不連続関数を用いた初期密度プロファイルからのレーリー・テラー不安定性の成長は、ナビエ・ストークス方程式に対して粘性項を付加することにより安定に計算できることを示し、誤差関数を初期密度プロファイルとして与えた場合は粘性項を用いずに安定にマッシュルーム構造が成長することが出来ることを示した。

### 3.4.3 3次元レーリー・テラー不安定性

レーリー・テラー不安定性が生じる現象は一般的に3次元構造を持っており、シングルモードだけではなく様々な擾乱が重畳されより複雑な振舞いを見せる。これまでに様々な計算スキームを用いた3次元レーリー・テラー問題の数値計算がなされて来たが、流体的構造が非常に複雑であるため従来の有限差分法等の計算スキームでは多くの計算格子点数を必要とするため [21]、これまでの計算機能力では現象を数値計算することすら困難であった。エルミート補間を用いた計算手法の先駆である CIP 法を用いることで少ない計算格子点数で3次元レーリー・テラー不安定性を扱うことが可能であることが示されている [22]。IDO 法も有限差分法等の計算スキームに比べ少ない格子点数で十分に高精度な計算が可能であることから、単純なシングルモードの計算に関しては少ない格子点数で十分に記述できると考えられる。また、様々な波数成分からなる擾乱をもつレーリー・テラー不安定性の成長の数値流体計算を行い、レーリー・テラー不安定性を引き起こす大規模問題への適用可能性を考察する。

#### 基礎方程式

2次元レーリー・テラー不安定性で用いた圧縮性流体解法を陽解法で用い、数値計算を行う。本問題に対しても人工粘性を考慮しないことから、支配方程式は以下の質量保存則、運動量保存則、エネルギー保存則、状態方程式によって記述される。

$$\rho_t = -u\rho_x - v\rho_y - w\rho_z - (u_x + v_y + w_z)\rho \quad (3.83)$$

$$u_t = -uu_x - vu_y - wu_z - \frac{1}{\rho} \frac{\partial P}{\partial x} \quad (3.84)$$

$$v_t = -uv_x - vv_y - wv_z - \frac{1}{\rho} \frac{\partial P}{\partial y} \quad (3.85)$$

$$w_t = -uw_x - vw_y - ww_z - \frac{1}{\rho} \frac{\partial P}{\partial z} \quad (3.86)$$

$$e_t = -ue_x - ve_y - we_z - \frac{P(u_x + v_y + w_z)}{\rho} \quad (3.87)$$

$$P = (\gamma - 1)\rho e \quad (3.88)$$

ここで、 $\rho, u, v, w, e, P, \gamma$  はそれぞれ密度、 $x, y, z$  方向移流速度、内部エネルギー、圧力、比熱比をあらわす。以下では  $\gamma = 1.4$  とした。ただし、 $x$  方向に重力加速度を付加するため、速度  $u$  に関する運動量保存式は

$$u_t = -uu_x - vu_y - wu_z - \frac{1}{\rho} \frac{\partial P}{\partial x} + g \quad (3.89)$$

と書き換えられる。

#### 初期プロファイル

3次元レーリー・テラー不安定性では2次元レーリー・テラー不安定性に比べその流体構造が複雑化することが予想されるため、より安定な数値計算を行うことを目的として先に示した誤差関数により初期密度プロファイルを与えることとする。

$$\rho = a \operatorname{erf}(b(c-x)) + d \quad (3.90)$$

$$p = 0.1 + \int \rho dx \quad (3.91)$$

$$\begin{aligned} &= -a \frac{\exp(-b^2(c-x)^2)}{b\pi} - d(c-x) - a(c-x) \operatorname{erf}(b(c-x)) \\ &\quad + \frac{a \exp(-b^2c^2)}{b\pi} + d \times c + a \times c \operatorname{erf}(b \times c) + 0.1 \end{aligned} \quad (3.92)$$

$$a = 0.1, \quad b = 40, \quad c = 0.6, \quad d = 0.65 \quad (3.93)$$



## 初期擾乱

2次元レーリー・テラー不安定性で考察した指数関数を用いた初期速度擾乱を3次元問題へ適用するために、拡張を行う。重力方向 ( $x$  軸方向) に対して、指数関数を用いて擾乱を減少させることを考え、3次元レーリー・テラー不安定性では以下に示す擾乱を用いる。

$$u = C_a \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.94)$$

$$u_x = -2C_a k_x x \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.95)$$

$$u_y = -C_a k_y \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.96)$$

$$u_z = -C_a k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.97)$$

$$u_{xy} = 2C_a k_x k_y x \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.98)$$

$$u_{yz} = C_a k_y k_z \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.99)$$

$$u_{xz} = 2C_a k_x k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.100)$$

$$u_{xyz} = -2C_a k_x k_y k_z \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.101)$$

$$v = C_b x \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.102)$$

$$v_x = C_b \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) - 2C_b k_x x^2 \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.103)$$

$$v_y = C_b k_y x \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.104)$$

$$v_z = -C_b k_x x \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.105)$$

$$v_{xy} = C_b k_y \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) - 2C_b k_x k_y x^2 \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.106)$$

$$v_{yz} = -C_b k_y k_z x \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.107)$$

$$v_{xz} = -C_b k_x \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) + 2C_b k_x k_z x^2 \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.108)$$

$$v_{xyz} = -C_b k_y k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) + 2C_b k_x k_y k_z x^2 \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.109)$$

$$w = C_c x \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.110)$$

$$w_x = C_c \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) - 2C_c k_x x^2 \cos(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.111)$$

$$w_y = -C_c k_y x \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.112)$$

$$w_z = C_c k_z x \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.113)$$

$$w_{xy} = -C_c k_y \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) + 2C_c k_x k_y x^2 \sin(k_y y) \sin(k_z z) \exp(-k_x x^2) \quad (3.114)$$

$$w_{yz} = -C_c k_y k_z x \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.115)$$

$$w_{xz} = C_c k_z \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) - 2C_c k_x k_z x^2 \cos(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.116)$$

$$w_{xyz} = -C_c k_y k_z \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) + 2C_c k_x k_y k_z x^2 \sin(k_y y) \cos(k_z z) \exp(-k_x x^2) \quad (3.117)$$

非圧縮を仮定した擾乱を含めるためには  $u_x + v_y + w_z = 0$  とならなければならないため、

$$C_a k_x + C_b k_y + C_c k_z = 0 \quad (3.118)$$

を満足する必要がある。

## 3次元レーリー・テラー不安定性の成長速度

レーリー・テラー不安定性の線形擾乱の成長率は式 (3.77) で与えられる。3次元問題でシングルモードを扱うため、式 (3.78) における  $k_y, k_z$  を  $2\pi$  と設定することから、2次元レーリー・テラー不安定性の成長速度に対して $\sqrt{2}$ 倍遅くなることが考えられる。

計算空間  $0 \leq x \leq 1.2, 0 \leq y \leq 0.3, 0 \leq z \leq 0.3, \Delta x = 1.2/40, \Delta y = 0.3/10, \Delta z = 0.3/10$  に対して誤差関数であらわされるプロファイル式 (3.90) - 式 (3.93) を与え、初期擾乱として式 (3.94) - 式 (3.117) で

示される速度擾乱を与える。ここで、 $k_x = 100\pi/1.2$ ,  $k_y = 2\pi/0.3$ ,  $k_z = 2\pi/0.3$ ,  $Ca = 2C_b = 2C_c$  とした。計算空間中央における密度擾乱の時間発展を図 3.31 に示す。

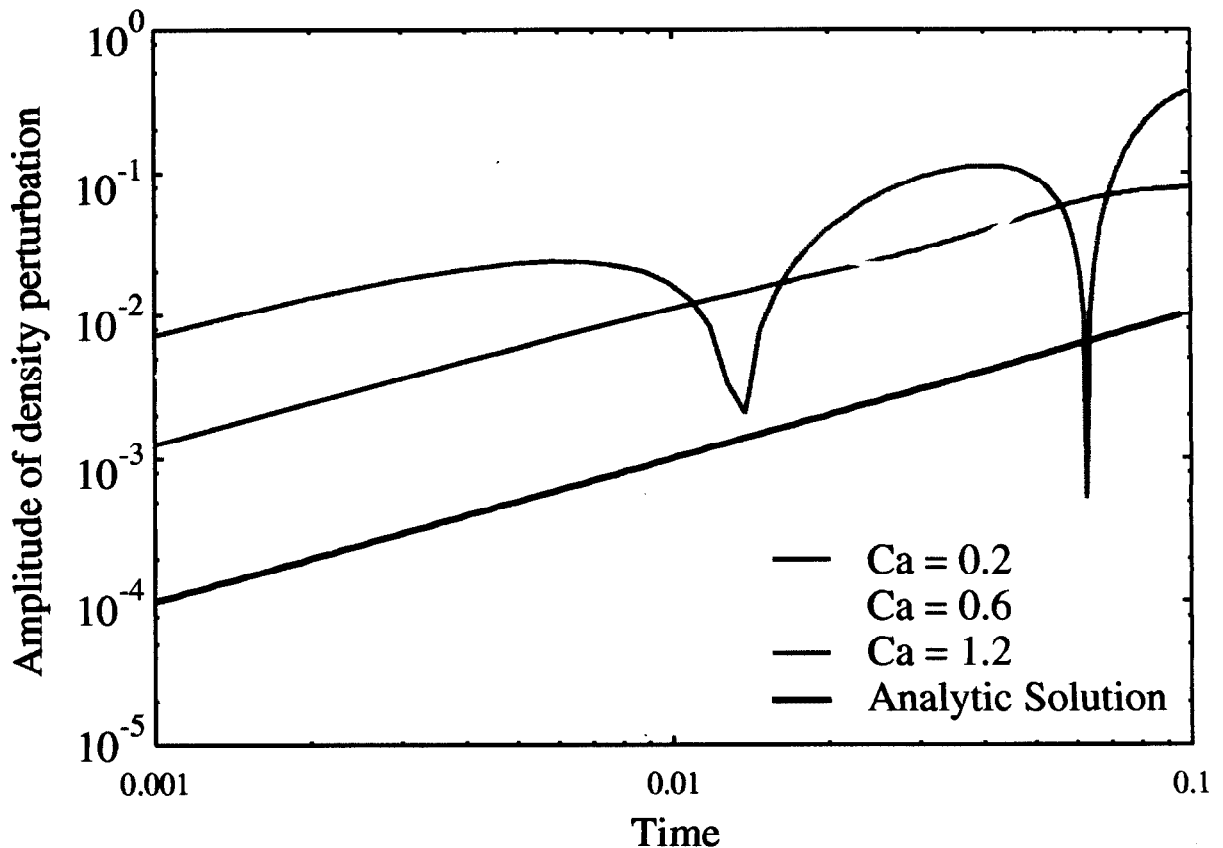


図 3.31: 3次元レーリー・テラー不安定性における密度擾乱の成長率

初期速度擾乱の大きさに関わらず、3次元レーリー・テラー不安定性における密度擾乱の成長率(傾き)は、解析解から得られるものによく一致した結果を得た。

初期速度擾乱の大きさに関わらず、擾乱が線形成長を続ける間ではその成長率(傾き)は解析解によく一致した結果を得ることに成功した。このことにより、IDO法による圧縮性流体支配方程式系に対する陽解法が3次元レーリー・テラー不安定性に対して有効であることを示した。

### 3次元レーリー・テラー不安定性の流体構造

3次元レーリー・テラー不安定性は複雑な流体構造を持つことで知られており、これまでも様々な計算スキームを用いてその解析が行われて来た。有限差分法のようなスキームではその複雑な構造を表現するために多くの格子点数を必要とするため、大規模計算機による数値流体計算を必要としていたが、CIP法を用いることで少ない格子点数( $60 \times 30 \times 30$ )でその構造を捕らえることに成功している。[22]

CIP法に比べ非移流相計算精度が高いIDO法を用いることでさらに少ない格子点数での計算が行えると考え、計算領域  $0 \leq x \leq 0.6$ ,  $0 \leq y \leq 0.3$ ,  $0 \leq z \leq 0.3$ ,  $\Delta x = 0.6/20$ ,  $\Delta y = 0.3/10$ ,  $\Delta z = 0.3/10$  での数値計算を行う。ただし、計算格子点上の値のみを用いた可視化を行うと計算結果が数値拡散を伴って観測される。CIP法と同様に、IDO法はグリッド間プロファイルを表現するために空間微係数を用いていることから、他の有限差分法などのスキームに比べて高解像度計算が可能になっていることが考えられる。この特徴を考え、以下ではグリッド間プロファイルを3次元CIP法 Type-C で用いられる補間関数により

$\Delta x/5$  の解像度でプロファイルを補間し、描画する。

式 (3.94) - 式 (3.117) で示される速度擾乱を用いて数値計算を行った所、マッシュルーム構造の時間発展が非常に遅いため、計算領域内でマッシュルーム構造が発達するまでの数値計算を行うことは困難であった。そこで、あえて不連続性が入るように 2 次元レーリー・テラー問題に対する初期擾乱で提案した  $\exp(-k_x|x|)$  により減衰させる以下の速度擾乱を加え、非線形成長を比較的早い段階で成長させることを考える。

$$u = -a \cos(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.119)$$

$$u_x = a k_x \cos(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.120)$$

$$u_y = a k_y \sin(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.121)$$

$$u_z = a k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.122)$$

$$u_{xy} = -a k_x k_y \sin(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.123)$$

$$u_{xz} = -a k_x k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.124)$$

$$u_{yz} = -a k_y k_z \sin(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.125)$$

$$u_{xyz} = a k_x k_y k_z \sin(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.126)$$

$$v = a k_x/k_y \sin(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.127)$$

$$v_x = -a k_x k_x/k_y \sin(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.128)$$

$$v_y = a k_x \cos(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.129)$$

$$v_z = -a k_x k_z/k_y \sin(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.130)$$

$$v_{xy} = -a k_x k_x \cos(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.131)$$

$$v_{xz} = a k_x k_x k_z/k_y \sin(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.132)$$

$$v_{yz} = -a k_x k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.133)$$

$$v_{xyz} = a k_x k_x k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.134)$$

$$w = -2.0 a k_x/k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.135)$$

$$w_x = 2.0 a k_x k_x/k_z \cos(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.136)$$

$$w_y = 2.0 a k_x k_y/k_z \sin(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.137)$$

$$w_z = -2.0 a k_x \cos(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.138)$$

$$w_{xy} = -2.0 a k_x k_x k_y/k_z \sin(k_y y) \sin(k_z z) \exp(-k_x x) \quad (3.139)$$

$$w_{xz} = 2.0 a k_x k_x \cos(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.140)$$

$$w_{yz} = 2.0 a k_x k_y \sin(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.141)$$

$$w_{xyz} = -2.0 a k_x k_x k_y \sin(k_y y) \cos(k_z z) \exp(-k_x x) \quad (3.142)$$

ここで、 $x$  軸は計算領域中心を 0 としており、上記に示したのは、 $x > 0$  の場合のみのものである。各係数  $a, k_x, k_y, k_z$  はそれぞれ  $a = 6 \times 10^{-2}/k_y, k_x = 4\pi, k_y = 2\pi, k_z = 2\pi$  と与えた。

密度、圧力の初期プロファイルとして、式 (3.90) - 式 (3.92) に示される誤差関数プロファイルに対して  $a = 0.10, b = 50, c = 0.3, d = 0.65$  を用いて与え、 $t = 0, 1.3, 2.6, 3.9, 5.2, 6.5, 7.8, 9.2 (\times 10^{-2})$  における密度プロファイルの  $\rho = 0.65$  の等値面を描画したものを図 3.32 に示す。初期条件下では計算空間中心においてその等値面が描かれており、時間の経過とともに初期擾乱に対応したプロファイルの線形成長が観測される (図 3.32 (c))。次第に成長したその先端部分において球状を形成し始め (図 3.32 (d),(e))、最終的にマッシュルーム構造をとる (図 3.32 (g),(h)) ことを確認した。

更にその流体構造の詳細を議論するため、図 3.32 (g) を拡大したものを図 3.33 に示す。周期境界条件を用いて計算を行っていることから、右方向に見えるプロファイルの切断面は中央に現れているプロファ

イルと完全に相似形で求められている。その構造から、マッシュルーム構造の上方先端部では半球に近い構造をとっていることが分かる。また、マッシュルーム構造の傘部分の下方ではくぼみが観測されている。胴体部において凹凸が確認できるが、同初期擾乱により2次元計算した際に得られた2次系の渦が数値誤差により拡散した結果であると考えられる。

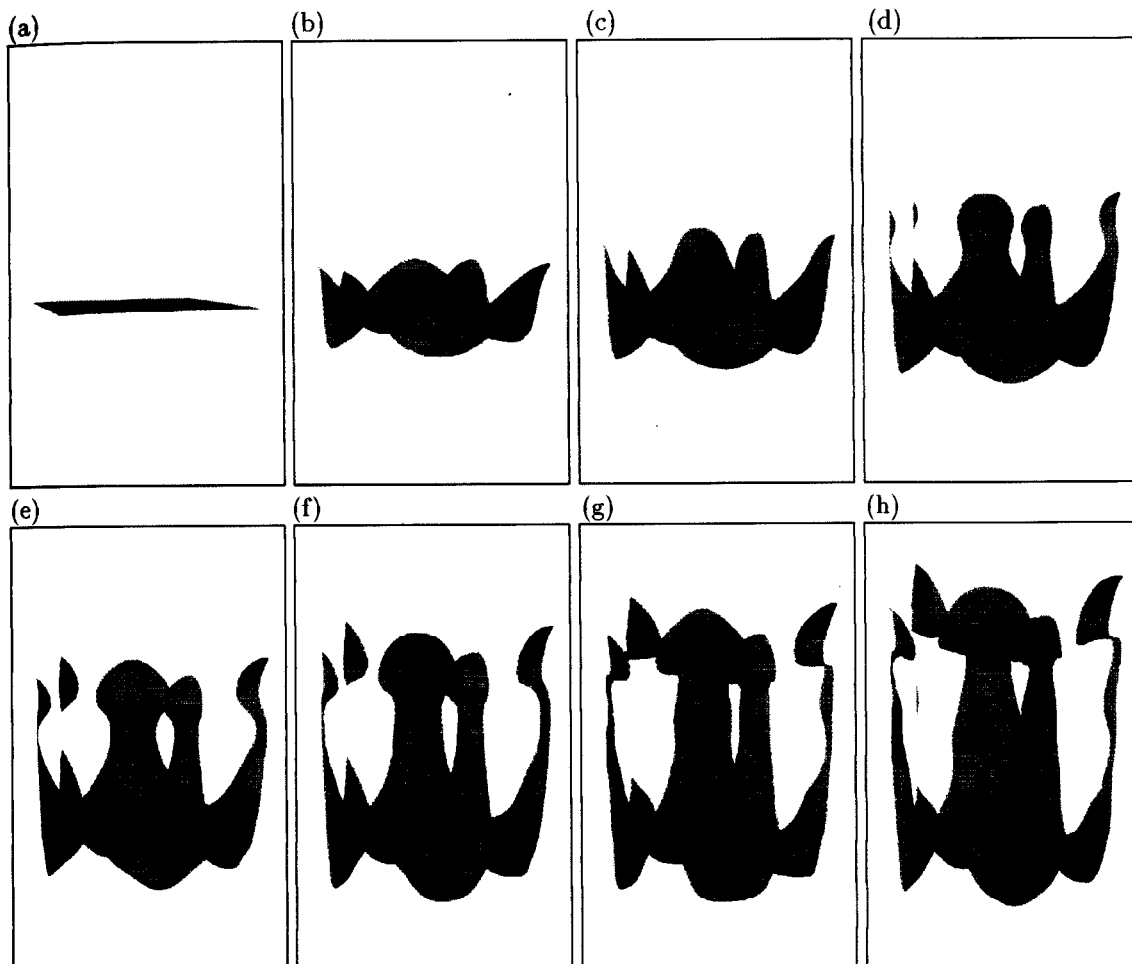


図 3.32: 3次元レーリー・テラー不安定性の成長過程

(a)-(h) はそれぞれ  $t = 0, 1.3, 2.6, 3.9, 5.2, 6.5, 7.8, 9.2(\times 10^{-2})$  における密度  $\rho = 0.65$  の等値面を示す。

非常に少ない格子点数 ( $20 \times 10 \times 10$ ) を用いているにも関わらず、レーリー・テラー不安定性から生じる複雑な流体構造を高精度に捕らえており、その胴体部の直径はおよそ  $3 \sim 4\Delta y$  程度で表現されている。今回作成したアプリケーションが要するコンピューターメモリ量はおよそ 5 MByte 程度であり、またその計算負荷もパーソナルコンピューター (Personal Computer; PC) 程度で充分なものであった。このことから、IDO 法を用いることで劇的に必要格子点数を減少させることができ、またそれとともなって計算負荷を大幅に減らすことが出来ることから、数値流体計算が PC でも充分な精度で行えるといえる。

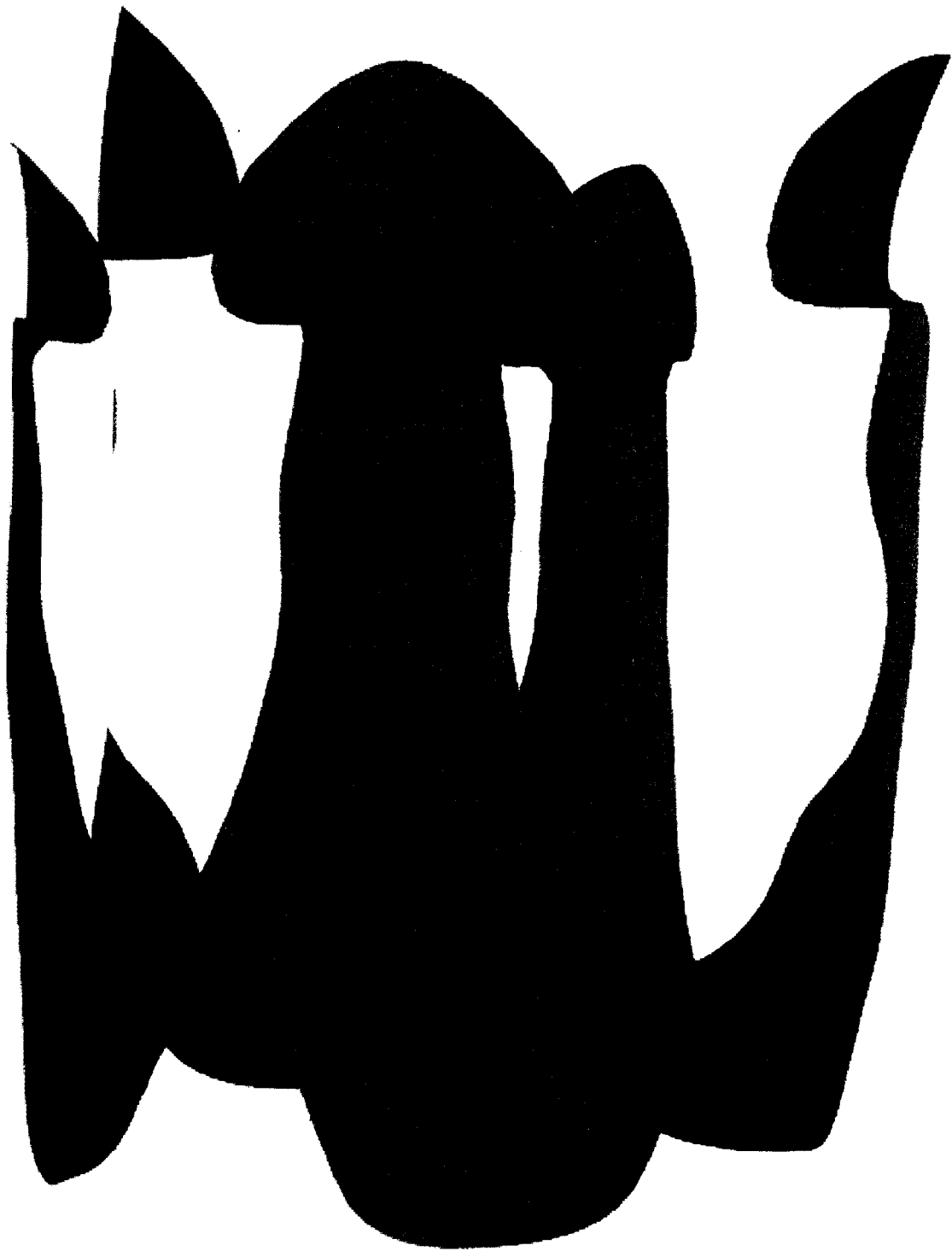


図 3.33: 3次元レーリー・テラー不安定性の流体構造  
非常に少ない格子点数 ( $20 \times 10 \times 10$ ) を用いているにも関わらず、複雑な 3次元レーリー・テラー不安定性の流体構造を的確に捕らえている。

## 様々なモードを含むレーリー・テラー不安定性

IDO 法を用いることで細かな構造を少ないグリッド数で表現できるとは言え、3次元流体問題を扱う場合にはわずかに格子点数を増加させることにも莫大な計算機メモリ容量、計算時間を要求される。本節では3次元レーリー・テラー不安定性の一般的な流体構造をデモンストレーションすることを目的として様々な周波数成分を持つ擾乱を流体に加え、複雑な構造の発達を数値計算する。計算空間  $0 \leq x \leq 0.9, 0 \leq y \leq 0.3, 0 \leq z \leq 0.3$  に対して格子点数を  $90 \times 30 \times 30$  とし、式 (3.90)、式 (3.92) で示される誤差関数を用いた初期プロファイルを与える。初期プロファイルを決定する各係数  $a, b, c, d$  はそれぞれ

$$a = 0.2, \quad b = 40, \quad c = 0.45, \quad d = 0.65 \quad (3.143)$$

とした。初期擾乱として式 (3.94) - 式 (3.117) で示される擾乱を

$$C_a = 1.2, C_b = 8, C_c = 8, k_x = \frac{40\pi}{0.9}, k_y = \frac{2\pi}{0.3}, k_z = \frac{2\pi}{0.3} \quad (3.144)$$

$$C_a = 0.6, C_b = 1, C_c = 1, k_x = \frac{20\pi}{0.9}, k_y = \frac{4\pi}{0.3}, k_z = \frac{4\pi}{0.3} \quad (3.145)$$

$$C_a = 0.72, C_b = 0.8, C_c = 0.8, k_x = \frac{20\pi}{0.9}, k_y = \frac{2\pi}{0.3}, k_z = \frac{4\pi}{0.3} \quad (3.146)$$

の3タイプに関して線形に重ね併せて導入する。ただし、複雑な構造が得られることを期待して、それぞれの擾乱の位相を変えて与えた。

計算結果から得られた流体境界面 ( $\rho = 0.65$ ) の成長過程を図 3.34 に示す。

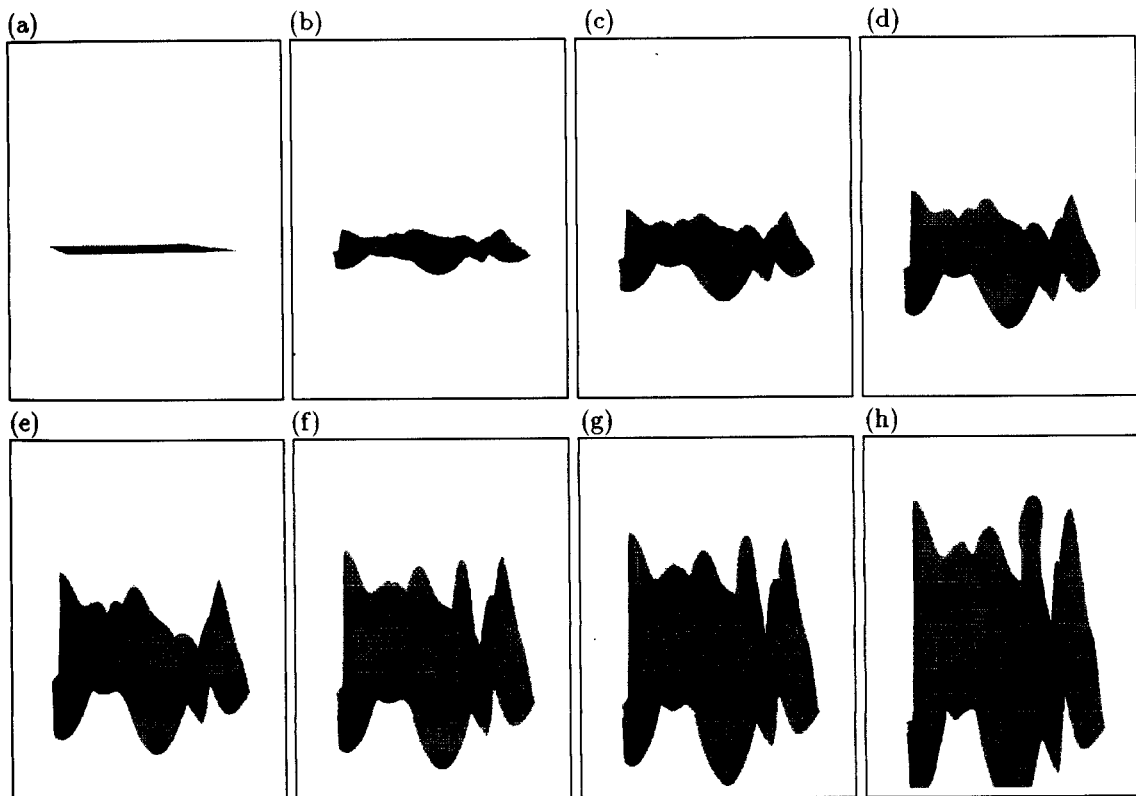


図 3.34: 複雑な擾乱に対する 3次元レーリー・テラー不安定性の成長:

シングルモードを誘発する初期擾乱を重ね合わせて導入すると得られる流体構造はシングルモードに比べて複雑な形状が得られる。

シングルモードを誘発する速度擾乱を重ね合わせて導入することで、得られる流体構造はシングルモードから得られるもの比べて複雑化し、また非線形成長においてはその成長過程を議論することが困難になる。しかし、一般的に観測されるマッシュルーム構造に成長する過程の構造が図 3.34 (f) に見られている。また他方から同境界面を観測した結果を図 3.35 に示す。



図 3.35: 3次元レーリー・テラー不安定性の構造:

複雑な構造を持つ3次元レーリー・テラー不安定性問題の数値計算結果から、重力方向に垂直方向に対して伸びるマッシュルーム構造を得ることに成功した。

境界面が複雑に成長し、重力方向(紙面下方向)に対して垂直方向にマッシュルーム構造が成長している様子をとらえることに成功した。また、それと隣接する構造もマッシュルーム構造をとりつつあることが確認できる。このことから、少ない格子点数を用いた場合においてもIDO法を用いることで複雑な3次元レーリー・テラー不安定性に対してアプローチ可能であることを示した。今後、慣性核融合(Inertial Confinement Fusion; ICF)や、Zピンチ(Z-Pinch)などのレーリー・テラー不安定性が重要になるア

アプリケーションを扱う際に IDO 法による数値計算が効果を発揮すると結論づける。

### 3 次元レーリー・テラー不安定性のまとめ

これまで有限差分法を用いた場合には大規模計算が必要とされた 3 次元レーリー・テラー不安定性の成長過程の数値計算をエルミート補間を用いるもう一つの計算スキームである CIP 法に比べより少ないグリッド数で捕らえることに成功し、IDO 法の 3 次元流体問題への適用性を明らかにすると共に、エルミート補間関数を用いた数値計算ではグリッド内流体構造を高精度で捕らえていることからその可視化も重要であることを示した。

より一般的なレーリー・テラー不安定性を扱うことを考慮し、3 タイプのシングルモードを誘発する速度擾乱を与えたレーリー・テラー不安定性の数値計算を行ったところ、単純な擾乱の重ね合わせを用いた場合でもシングルモードでは捕らえられない複雑な流体構造が成長する様子を捕らえることに成功した。この結果から、ICF や Z-Pinch などのレーリー・テラー不安定性が重要になる大規模アプリケーションに対しても効果的な数値流体計算を行う精度を IDO 法が持つことを明らかにした。



### 3.5 超音速混合流問題への適用

これまでにエルミート補間を用いた数値流体計算スキームである CIP 法ならびに IDO 法の性質を考慮した基本的な精度検証、基礎的な偏微分方程式に対する精度検証、ベンチマーク問題として利用される流体問題への適用を行い、少ない格子点数で高精度解を示すことを明らかにした。

本節では、より具体的なアプリケーション例として超音速流を用いた混合過程を扱い、現象に対して流体構造の視点から実験結果との比較、考察を行うことを目的とする。

#### 3.5.1 超音速混合の概要

輸送機関の発達により高速に飛行する超音速旅客機や、スペースプレーン (宇宙往還機) が近年開発されている。言うまでもなく超音速航行は従来のプロペラを用いたレシプロエンジンでは達成することが出来ず、ラムジェットエンジンや、ターボジェットエンジンなどを利用する必要がある。エンジン内で吸入流体が亜音速となるラムジェットエンジン (図 3.36) に比べ燃焼器内で超音速流れを維持したまま燃料を燃焼させるスクラムジェットエンジンは強い衝撃波の発生に伴うエントロピー増加を抑制できることから高飛行マッハ数領域での高効率推進に有利であることが知られている。これらのエンジンは化学燃料の燃焼エネルギーを用いた熱推進機関であり、効果的に作動燃料を燃焼させることで高効率化を図ることが出来る。

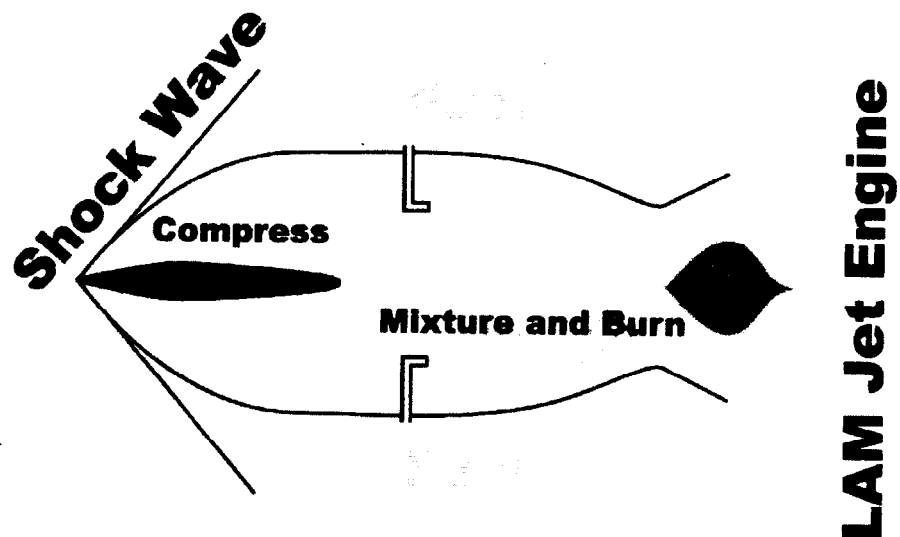


図 3.36: ラムジェットエンジンの概念図:

ラムジェットエンジンでは流体採り入れ口において衝撃波を発生させ、そのラム圧を利用すると共にさらに燃料の燃焼エネルギーにより推力を得る。

そのためには燃料をエンジン内に吸入された気体と効果的に混合することが重要な課題となるが、スクラムジェットエンジンに着目すると、超音速流中の 2 流体混合はこれまでのいくつかの論文で指摘されているように非圧縮流体の混合と比較して効率が悪いとされているが、詳細な混合過程は未解明である。

また、超音速混合過程が重要なアプリケーションの例として図 3.37 に示す GasDynamic LASER (GDL) が挙げられる。異なる作動流体を超音速により混合することにより、反転分布をもつレーザー媒質を形成し、それにより入射レーザーを増幅することで、高強度レーザーを得ることが出来る。この場合においても作動流体の分子レベルでの急速な混合が重要であり、また、衝撃波発生に伴うエントロピー増大 (温度上昇) がレーザー媒質の利得 (Gain) を低下させるためレーザーの高強度化においても混合過程の解明が重要な問題となる [23]。

## Population Inversion Driven by Super Sonic Mixing

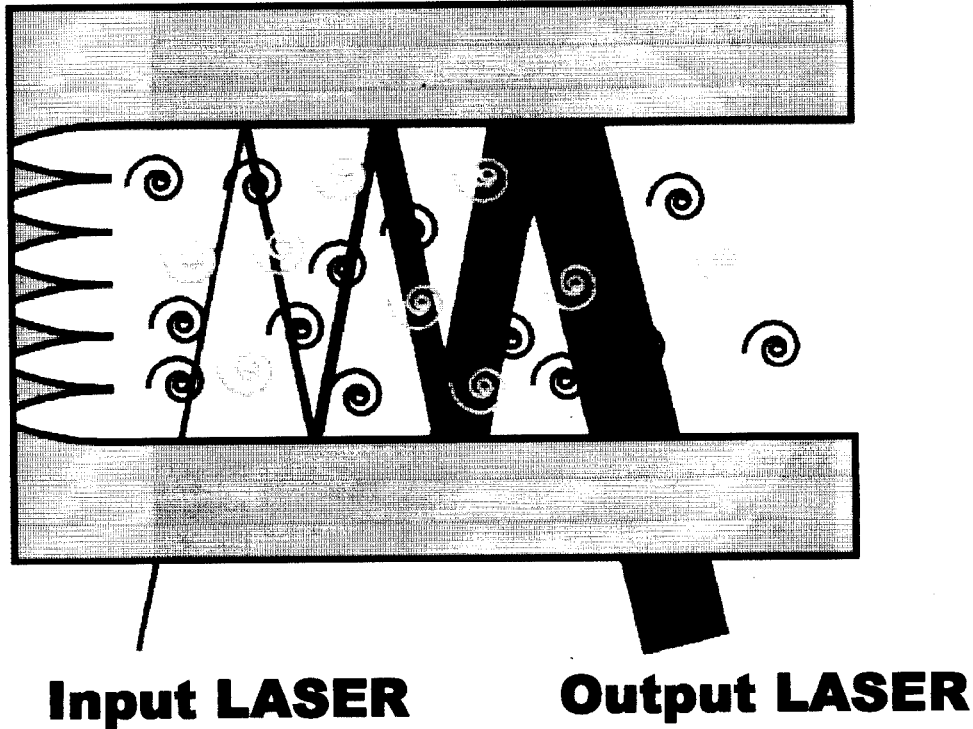


図 3.37: GasDynamic LASER (GDL) の概念図:

超音速に加速された作動流体を急速に混合することにより反転分布を形成し、LASER の増幅に用いる。

一方、圧縮性の影響が大きい高マッハ数領域での混合過程は、渦と衝撃波の相互作用とこれに伴って起こるエネルギー散逸過程が流れを支配していると予測され、流体物理学的視点からも興味深い。

このように、超音速混合流は工学的、理学的に興味深い現象 [24, 25] であり、本章では IDO 法を用いた数値計算結果と実験結果との比較を行うことでその流体構造に関する議論を行う。

### 3.5.2 混合過程における理論

非圧縮性の条件が成立する流体に対しては並行に流れる 2 流体間に擾乱が存在する場合の流体不安定性による渦構造の成長が詳細に議論されている。圧縮性流体においては流体不安定性による渦の成長が非圧縮性流体に比べて抑制される傾向がある。このことから、超音速流体を用いた 2 流体混合は比較的困難とされ、さまざまな課題が残されている [26]。本節ではそれぞれの混合過程に関する理論を紹介する。

#### Kelvin-Hermholtz 不安定性

我々は風によって水面上に波紋が生じる現象 [27] や、空に浮かぶ雲の周期的な列 (図 3.38 (a)) を日常的に見かける。これらは Kelvin-Hermholtz 型不安定性 [28] が成長するメカニズムと同じ原理で生成される。速度の異なる 2 流体が相接する界面では渦構造が発生し、その流速差がある基準値を越えると不安定性を示すようになる。非圧縮、非粘性流体におけるこのような流速差により成長する渦列の不安定成長は

Kelvin-Helmholtz 不安定性によって説明されている。その代表的なものに図 3.38 (b) に示す木星の大赤斑が挙げられる。近年では乱流混合層の発達過程を Kelvin-Helmholtz 不安定性により議論している論文も見られる [29]。

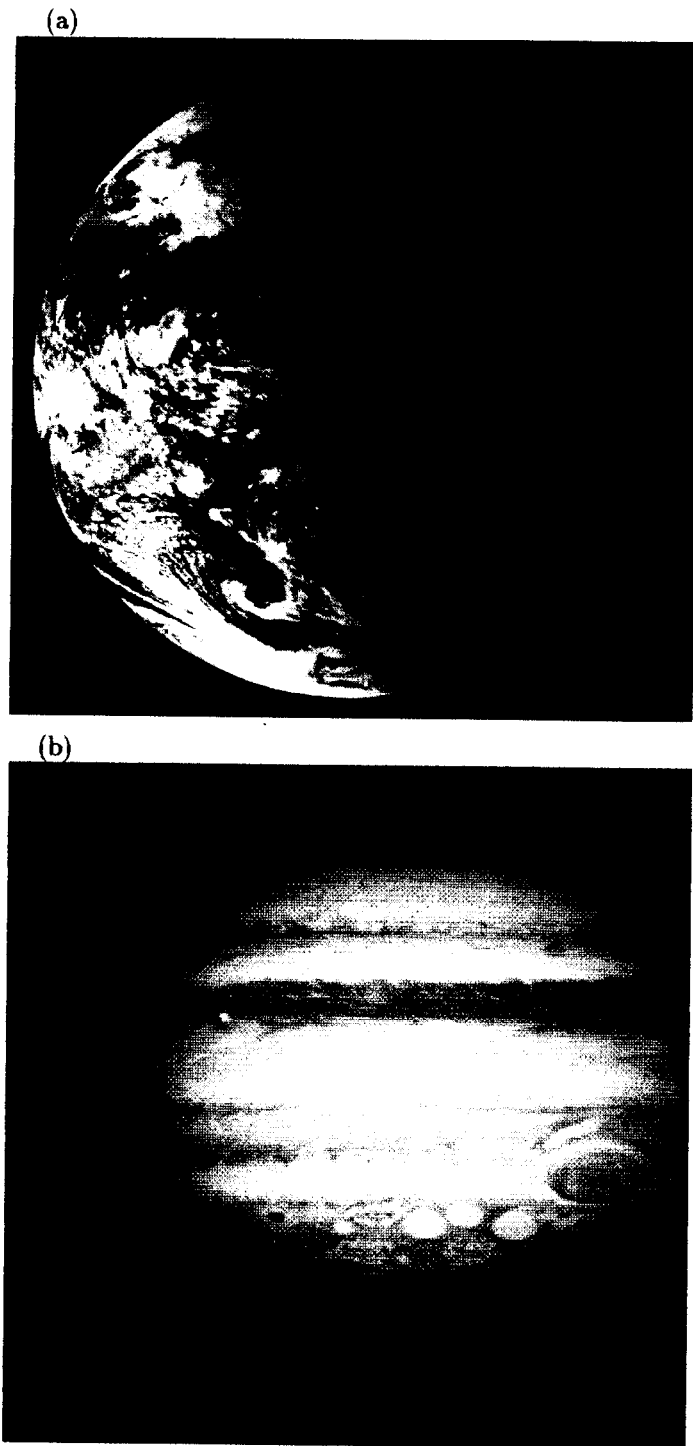


図 3.38: Kelvin-Helmholtz 不安定性の例:  
日常的に観測される (a) うろこ雲の生成メカニズムは Kelvin-Helmholtz 不安定性のものと類似しており、(b) 木星の大赤斑はまさに Kelvin-Helmholtz 不安定性により駆動されている。

数学的に Kelvin-Helmholtz 不安定性について考える。一様流  $\mathbf{u} = (U(z), 0, 0)$  中の擾乱  $\delta\rho, \delta p, u, v, w$  の成長について非粘性 Navier-Stokes 方程式を記述すると、

$$(U + u)_t = -(U + u)(U + u)_x - v(U + u)_y - w(U + u)_z - \frac{(\delta p)_x}{\rho}, \quad (3.147)$$

$$v_t = -(U + u)v_x - vv_y - ww_z - \frac{(\delta p)_y}{\rho}, \quad (3.148)$$

$$w_t = -(U + u)w_x - vw_y - ww_z - \frac{(\delta p)_z}{\rho} - g\frac{\delta\rho}{\rho}, \quad (3.149)$$

とあらわされる。ここで、添字  $x, y, z, t$  はそれぞれ空間  $x, y, z$  方向もしくは時間  $t$  による偏微分を示している ( $f_x = \partial f / \partial x$ )。また、密度、流速は  $z$  軸方向 (重力方向) に分布している ( $\rho = \rho(z), \mathbf{u} = \mathbf{u}(z)$ ) ものとし、 $z$  方向に対して重力加速度  $g$  を考慮した。微小擾乱に関する 2 次の項を省略し、 $\partial U(z) / \partial t = 0$  とすると、上記支配方程式は、

$$\frac{\partial u}{\partial t} + U \frac{\partial u}{\partial x} + w \frac{\partial U}{\partial z} = -\frac{\partial}{\partial x} \left( \frac{\delta p}{\rho} \right), \quad (3.150)$$

$$\frac{\partial v}{\partial t} + U \frac{\partial v}{\partial x} = -\frac{\partial}{\partial y} \left( \frac{\delta p}{\rho} \right), \quad (3.151)$$

$$\frac{\partial w}{\partial t} + U \frac{\partial w}{\partial x} = -\frac{\partial}{\partial z} \frac{\delta p}{\rho} - g \frac{\delta p}{\rho}, \quad (3.152)$$

と表される。さらに流体境界面の運動方程式、質量保存式ならびに非圧縮性を仮定 ( $\nabla \cdot \mathbf{u} = 0$ ) から得られる関係式、

$$\frac{\partial}{\partial t} \delta z_s + U_s \frac{\partial}{\partial x} \delta z_s = w(z_s), \quad (3.153)$$

$$\frac{\partial}{\partial t} \delta\rho + U \frac{\partial}{\partial x} \delta\rho = -w \frac{\partial \rho}{\partial z}, \quad (3.154)$$

$$u_x + v_y + w_z = 0, \quad (3.155)$$

を用いる。ここで、添字  $s$  は境界面における値であることをあらわしている。ここで、 $x, y, t$  に依存する

$$\exp i(k_x x + k_y y + nt) \quad (3.156)$$

で表される擾乱を解の形式の一つとして仮定した場合、式 (3.150) - 式 (3.155) はそれぞれ下記のように書き換えられる。

$$i\rho(n + k_x U)u + \rho(DU)w = -ik_x \delta p, \quad (3.157)$$

$$i\rho(n + k_x U)v = -ik_y \delta p, \quad (3.158)$$

$$i\rho(n + k_x U)w = -D\delta p - g\delta\rho, \quad (3.159)$$

$$i(n + k_x U)\delta\rho = -wD\rho, \quad (3.160)$$

$$i(n + k_x U_s)\delta z_s = w_s, \quad (3.161)$$

$$i(k_x u + k_y v) = -Dw, \quad (3.162)$$

ここで、 $D$  は微分演算子で、 $\partial / \partial z$  をあらわす。式 (3.157)、式 (3.158) にそれぞれ  $-ik_x, -ik_y$  を掛け、式 (3.162) を利用することにより、

$$i\rho(n + k_x U)Dw - i\rho k_x (DU)w = -k^2 \delta p, \quad (3.163)$$

が得られる。ここで、 $k^2 = k_x^2 + k_y^2$  とした。また、式 (3.159) に対して式 (3.160) を代入することより、

$$i\rho(n + k_x U)w = -D\delta p - ig(D\rho) \frac{w}{n + k_x U} \quad (3.164)$$

の関係式が得られる。式 (3.163)、式 (3.164) を連立し、 $\delta p$  を消去すると、

$$D\{\rho(n+k_xU)Dw - \rho k_x(DU)w\} - k^2\rho(n+k_xU)w = gk^2(D\rho)\frac{w}{n+k_xU} \quad (3.165)$$

の関係式が得られる。上式を展開し、まとめると以下のように記述される。

$$(n+k_xU)(D^2-k^2)w - k_x(D^2U)w - gk^2\frac{D\rho}{\rho}\frac{w}{n+k_xU} + \frac{D\rho}{\rho}[(n+k_xU)Dw - k_x(DU)w] = 0 \quad (3.166)$$

擾乱の速度成分  $w$  が境界面上で連続であるためには、

$$\frac{w}{n+k_xU} \quad (3.167)$$

が境界上で連続でなければならない [30]。

ここで改めて 2 流体モデルを考える。 $z$  軸下方の流体密度、速度を  $\rho_1 = \text{const.}$ ,  $\mathbf{u} = (U_1, 0, 0) = \text{const.}$ 、上方の流体を  $\rho_2 = \text{const.} < \rho_1$ ,  $\mathbf{u} = (U_2, 0, 0) = \text{const.}$  とし、式 (3.166) を境界面における拘束条件とすると、それぞれの流体中では式 (3.166) は、

$$(n+k_xU)(D^2-k^2)w = 0, \quad (3.168)$$

と書き下すことができる。本条件下では  $w/(n+k_xU)$  が境界面上で連続であることから  $(n+k_xU)$  を省略することができる、

$$(D^2-k^2)w = 0, \quad (3.169)$$

が満足される。式 (3.169) の一般解は境界面における座標を  $z = 0$  として  $\exp(+kz)$ ,  $\exp(-kz)$  の線形結合により記述される。境界面から離れるに従い  $w$  が指数関数的に増加することは物理的に矛盾しているため、それぞれの流体に対して  $w$  を、

$$w_1 = A(n+k_xU_1)e^{+kz} \quad (z < 0) \quad (3.170)$$

$$w_2 = A(n+k_xU_1)e^{-kz} \quad (z > 0) \quad (3.171)$$

と与えることができる。式 (3.165) を境界面近傍 ( $z_s - \varepsilon \sim z_s + \varepsilon$ ) においてとらえ、 $\varepsilon = 0$  の極限を考えた式

$$\Delta_s \{\rho(n+k_xU)Dw - \rho k_x(DU)w\} = gk^2 \{\Delta_s(\rho)\} \left( \frac{w}{n+k_xU} \right)_s \quad (3.172)$$

に対して、式 (3.170)、式 (3.171) で与えられる解を適用することで特性方程式

$$\rho_2(n+k_xU_2)^2 + \rho_1(n+k_xU_1)^2 = gk(\rho_1 - \rho_2) \quad (3.173)$$

を得る。ここで、 $\Delta_s$  は、境界面上での物理量の変化量を示している ( $\Delta_s(f) = f_{z=z_s, +0} - f_{z=z_s, -0}$ )。新しく変数

$$\alpha_1 = \frac{\rho_1}{\rho_1 + \rho_2}, \quad \alpha_2 = \frac{\rho_2}{\rho_1 + \rho_2}, \quad (3.174)$$

を導入することで、式 (3.173) は、

$$\alpha_2(n+k_xU_2)^2 + \alpha_1(n+k_xU_1)^2 = gk(\alpha_1 - \alpha_2) \quad (3.175)$$

とあらわされ、成長率  $n$  に関して展開すると、

$$n^2 + 2k_x(\alpha_1U_1 + \alpha_2U_2)n + k_x^2(\alpha_1U_1^2 + \alpha_2U_2^2) - gk(\alpha_1 - \alpha_2) = 0 \quad (3.176)$$

が得られ、その成長率は

$$n = -k_x(\alpha_1 U_1 + \alpha_2 U_2) \pm \sqrt{gk(\alpha_1 - \alpha_2) - k_x^2 \alpha_2 \alpha_1 (U_1 - U_2)^2} \quad (3.177)$$

で与えられる。擾乱を式 (3.156) で与えていることから、成長率が虚数になる場合に擾乱が成長し不安定性が生じることが分かる。このことをふまえて式 (3.177) から、

$$k_x^2 \alpha_1 \alpha_2 (U_1 - U_2)^2 > gk(\alpha_1 - \alpha_2) \quad (3.178)$$

であれば Kelvin-Helmholtz 不安定性が生じ、その差が大きい程成長が促進される。つまり、密度比の差  $(\alpha_1 - \alpha_2)$  が小さく、速度差  $(U_1 - U_2)$ 、擾乱波数  $k_x$  が大きくなるに従って不安定性がより成長することがわかる。また、その成長率は圧力に依存しないことは明らかである。

### Brown-Roshko 渦

Brown-Roshko 渦は超音速流中ないしは非圧縮性流体中における平板乱流混合領域の渦構造の成長について議論したものである [31]。先の Kelvin-Helmholtz 不安定性と異なる点として、超音速流中ではマッハ数が大きくなるに従って混合の度合が非圧縮性流に比べて低下する傾向が観測されることにある。マッハ 5.7 による実験では非圧縮性流と混合層厚さを比較した場合、300% もの減少が観測されている。

理論的背景から現象について考察する。 $i$  種の成分から流体が成る場合その拡散方程式は保存則から、

$$\frac{\partial n_i}{\partial t} + \nabla n_i \mathbf{u}_i = 0, \quad (3.179)$$

が得られる。ここで、 $n_i$  は  $i$  種の体積当たりの分子数、 $\mathbf{u}_i$  は平均速度をあらわす。また、 $\Gamma$  を拡散係数とした。Fick の法則から、

$$\frac{n_i}{\rho} (\mathbf{u}_i - \mathbf{u}) = -\Gamma \nabla \frac{n_i}{\rho} \quad (3.180)$$

が得られる。ここで、 $\rho$  は密度を示し、 $\mathbf{u}$  は混合分子流中の質量中心における速度をあらわす。両方程式から拡散方程式

$$\frac{\partial n_i}{\partial t} + \nabla n_i \mathbf{u} = \nabla \left( \rho \Gamma \nabla \frac{n_i}{\rho} \right), \quad (3.181)$$

が得られる。一様温度を持つ低マッハ数領域では  $\Delta p/p, \Delta T/T$  が  $M^2$  のオーダーになることから熱力学的圧力、温度の変化量が流体中で小さい。また、それぞれの分圧から得られる全圧 (Total Pressure) は  $p = nkT (n = \sum n_i)$  と得られ、分子数  $n$  の変化量も流体中で小さくなる ( $\Delta n/n \sim M^2$ )。このことから、式 (3.181) を全流体成分に対して積算すると一般的な非定常拡散方程式

$$\nabla \mathbf{u} = \nabla \left\{ \rho \Gamma \nabla \left( \frac{1}{\rho} \right) \right\} \quad (3.182)$$

が得られる。乱流場における記述を行うことを目的として、 $\mathbf{u} = \mathbf{U} + \mathbf{u}'$  を導入し、粘性応力がレイノルズ応力に比べて充分小さいことを考慮すると分子運動論的拡散を省略することができることから、式 (3.182) は、

$$\nabla \mathbf{U} = 0 \quad (3.183)$$

と書き直すことができる。

次に、支配方程式について考える。 $x$  軸方向に対して支配的な流速  $\mathbf{U} = (U, 0)$  を持つ乱流速度場  $\mathbf{u} = \mathbf{U} + \mathbf{u}'$  では質量保存方程式、運動量保存方程式は  $x$  軸方向に対する微係数が  $y$  軸方向に対するものに比べて充分小さく、速度擾乱  $\mathbf{u}' = (u', v')$  の自乗平均 (r.m.s.) が同等程度 (Comparable) と仮定して次のようにあら

わされる。

$$\frac{\partial}{\partial x}(\rho U) + \frac{\partial}{\partial y}(\rho V + \overline{\rho'v'}) = 0, \quad (3.184)$$

$$\frac{\partial}{\partial x}(\rho U^2) + \frac{\partial}{\partial y}(\rho UV + U\overline{\rho'v'}) = -\frac{\partial}{\partial y}(\overline{\rho u'v'}), \quad (3.185)$$

$$\frac{\partial}{\partial y}(\overline{\rho v'^2}) = -\frac{\partial p}{\partial y}. \quad (3.186)$$

乱流速度場における運動方程式は以上の質量保存方程式、運動量方程式、拡散方程式からなり、これらから

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0 \quad (3.187)$$

が導かれる。 $\rho\tilde{V} = \rho V + \overline{\rho'v'}$  と定めると、式 (3.184)、式 (3.185)、式 (3.187) はそれぞれ

$$\frac{\partial}{\partial x}(\rho U) + \frac{\partial}{\partial y}(\rho\tilde{V}) = 0, \quad (3.188)$$

$$\frac{\partial}{\partial x}(\rho U^2) + \frac{\partial}{\partial y}(\rho U\tilde{V}) = \frac{\partial}{\partial y}\left(\nu_T \rho \frac{\partial U}{\partial y}\right), \quad (3.189)$$

$$\frac{\partial U}{\partial x} + \frac{\partial \tilde{V}}{\partial y} = \frac{\partial}{\partial y}\left(\rho \Gamma_T \frac{\partial 1}{\partial y \rho}\right), \quad (3.190)$$

とあらわすことができる。ここで、 $\Gamma_T, \nu_T$  は渦拡散係数をあらわし、

$$\overline{\rho'v'} = -\Gamma_T \frac{\partial \rho}{\partial y}, \quad (3.191)$$

$$\overline{u'v'} = -\nu_T \frac{\partial U}{\partial y}, \quad (3.192)$$

で与えられる。

超音速流問題を考慮する場合、式 (3.182) であらわされる拡散方程式はエネルギー保存式

$$\rho \frac{Dh}{Dt} - \frac{Dp}{Dt} = \mu \left(\frac{\partial u}{\partial y}\right)^2 + \frac{\partial}{\partial y}\left(k \frac{\partial T}{\partial y}\right), \quad (3.193)$$

に置き換えられる。理想気体 ( $h = \gamma/(\gamma-1)p/\rho$ ) を仮定すると、式 (3.193) は以下の式で書き下すことができる。

$$\frac{1}{\gamma} \frac{Dp}{Dt} - \frac{\gamma}{\gamma-1} \frac{p}{\rho} \frac{D\rho}{Dt} = \mu \left(\frac{\partial u}{\partial y}\right)^2 + \frac{\partial}{\partial y}\left(k \frac{\partial T}{\partial y}\right). \quad (3.194)$$

ここで、連続方程式

$$\frac{D\rho}{Dt} = -\rho \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right), \quad (3.195)$$

を用いることで、

$$\frac{1}{\gamma-1} \frac{\partial p}{\partial t} + \frac{\gamma}{\gamma-1} \left[ \frac{\partial}{\partial x}(pu) + \frac{\partial}{\partial y}(pv) \right] - u \frac{\partial p}{\partial x} - v \frac{\partial p}{\partial y} = \mu \left(\frac{\partial u}{\partial y}\right)^2 + \frac{\partial}{\partial y}\left(k \frac{\partial T}{\partial y}\right) \quad (3.196)$$

が導かれる。静的 (steady) な乱流場 ( $P + p', U + U' \dots$ ) を考慮した場合、上記エネルギー方程式は単純に

$$P \left[ \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} \right] + \frac{\partial}{\partial y} \overline{p'v'} + \frac{1}{\gamma} V \frac{\partial P}{\partial y} = 0 \quad (3.197)$$

と表される。ただし、 $y$  方向の物理量勾配が  $x$  方向のそれに比べ大きく、 $\overline{p'u'} \sim \overline{p'v'}$ ,  $\partial P/\partial x = 0$ ,  $(\gamma-1)/\gamma \overline{v'\partial p'/\partial y} < \partial/\partial y(\overline{p'v'})$  を仮定している。

ここで、混合境界層の発達について概算を行う。 $\alpha$  を混合層の広がり角とすると ( $= \Delta y / \Delta x$ )、式 (3.185) は式 (3.184) を利用して、

$$\rho U \frac{\partial U}{\partial x} + \rho V \frac{\partial U}{\partial y} + \frac{\partial U}{\partial y} \overline{\rho'v'} = -\frac{\partial \bar{p}}{\partial y} \overline{u'v'} - \frac{\partial u'}{\partial y} \overline{\rho v'} - \frac{\partial \bar{v}'}{\partial y} \overline{\rho u'} \quad (3.198)$$

と書き換えられる。微小擾乱の 2 次項を省略しまとめると

$$U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} + \frac{\partial}{\partial y} (\overline{u'v'}) = 0 \quad (3.199)$$

となり、先の仮定  $U = (U, 0)$  から  $U \gg V$  を考慮すると、

$$\overline{u'v'} \sim -\alpha U \Delta U \quad (3.200)$$

と概算される [31]。また、 $\overline{u'v'} \sim u'v'$ 、 $u' \sim \Delta U$  を仮定すると、式 (3.200) は

$$v' \sim -\alpha U \quad (3.201)$$

と見積もることができる。また、式 (3.186) から

$$\frac{\partial p}{\partial y} \sim -\frac{\rho \alpha^2 U^2}{\Delta y} \quad (3.202)$$

が得られ、 $\overline{p'v'} \sim p'v'$  を仮定すると

$$\overline{p'v'} \sim -\rho \alpha^3 U^3, \quad (3.203)$$

と見積もられる。また、エネルギー式 (式 (3.197)) から、

$$\alpha \Delta U + \Delta V - \gamma \alpha^3 U M^2 - V \alpha^2 M^2 = 0 \quad (3.204)$$

が得られる。ここで、 $M = U/a$  はマッハ数を表しており、 $a^2 = \gamma p / \rho$  である。マッハ数が充分小さい ( $M \ll 1$ ) 場合、第 3 項、第 4 項は省略でき、式 (3.187) が導かれる。また、充分マッハ数が大きい場合は  $V \sim -\alpha U$  より、第 4 項が第 3 項の形であらわされ、 $V = 0$  から、

$$\alpha \sim M^{-1} \left( \frac{\Delta U}{U} \right)^{\frac{1}{2}} \quad (3.205)$$

が導かれる。上式は超音速流ではマッハ数が増加するに従い混合層の発達が弱くなることを示唆している。

### 理論的背景からの考察

非圧縮性流体における渦の成長を扱った Kelvin-Helmholtz 不安定性解析から、密度の異なる 2 流体を並行に流し、混合を行う場合その速度差、初期擾乱の波数が大きく、密度比の差が小さくなるに従って混合が促進される傾向が得られることが理解できる。また、超音速流における準静的乱流を考慮した Brown-Roshko の理論から、マッハ数が増大するに従って混合層の広がり角が鈍化することが示唆されている。

しかし、超音速流体中では衝撃波の生成が避けられないと考えられ、そのような非線形現象を解析的に説明することは困難である。本章で扱う超音速混合流問題では、2 流体の速度を変化させることで得られる効果、また、実験結果、計算結果から得られる流体構造について議論する。

### 3.5.3 基礎方程式

超音速流体問題を扱う場合、レイノルズ数が極度に高いことから通常は非粘性流体として扱うことが出来る。しかし、その混合過程を議論するためには流体不安定に伴う流体混合に加え、せん断流に起因する



粘性効果による流体の相互作用が重要になるため、本章で扱う支配方程式は以下に示す粘性を考慮した Navier-Stokes 方程式ならびに質量保存式、エネルギー保存式、状態方程式とする。

$$\rho_t = -u\rho_x - v\rho_y - (u_x + v_y)\rho \quad (3.206)$$

$$u_t = -uu_x - vu_y - \frac{P_x}{\rho} + \kappa(u_{xx} + u_{yy}) \quad (3.207)$$

$$v_t = -uv_x - vv_y - \frac{P_y}{\rho} + \kappa(v_{xx} + v_{yy}) \quad (3.208)$$

$$e_t = -ue_x - ve_y - \frac{P(u_x + v_y)}{\rho} \quad (3.209)$$

$$P = (\gamma - 1)\rho e \quad (3.210)$$

ここで、 $\rho, u, v, e, P$  はそれぞれ密度、 $x$  軸方向流速成分、 $y$  軸方向流速成分、内部エネルギー、圧力をあらわしており、 $\kappa$  は粘性係数である。

また、伝熱に関しては超音速混合流問題に対しては温度による流体の拡散速度に比べ移流速度、圧力差による運動が支配的であると見え、ここでは考慮しない。

### 3.5.4 多次元圧縮性流体問題に対する人工粘性の検討

超音速混合流問題では衝撃波の発生が考えられることから、その扱いを適切に行うために非保存系で支配方程式を扱う IDO 法では人工粘性を加える必要がある。1 次元問題における人工粘性の検討に関しては 1 次元衝撃波管問題の計算結果から十分な精度で衝撃波を捉えることを明らかにした。本節では均一な密度場に対して点対称の圧力場を設定し、衝撃波が形成、伝搬する問題を扱うことで 2 次元問題に対する人工粘性の検討を行う。

初期密度として  $\rho(x, y) = 1$  を計算領域  $0 \leq x \leq 2, 0 \leq y \leq 2, \Delta x = 2/50, \Delta y = 2/50$  に対して与える。初期圧力として、計算領域中心から同心円状に衝撃波を伝搬させることを目的として

$$P(x, y) = A \exp\left(-\frac{r^2}{dr^2}\right) + 1 \quad (3.211)$$

$$A = 40, dr = 0.1, r = \sqrt{(x - x_c)^2 + (y - y_c)^2} \quad (3.212)$$

を用いた。ここで、 $x_c, y_c$  はそれぞれ計算空間の  $x, y$  座標の中心値をあらわす。初期速度は空間全体でゼロとした。

多次元問題では C.2 節に示すように式 (C.26) で表される人工粘性を付加することから、 $c_0, c_L$  に対して考察する。有限差分法では  $c_0 \simeq 2, c_L \simeq 1$  で用いられる。先の初期条件による衝撃波の伝搬過程を時間履歴で図 3.40 に示す。カラーバーは図 3.39 に示すものを用いており、左端で密度 0、右端で密度 3 とし、線形に密度を 256 分割して色を決定した。

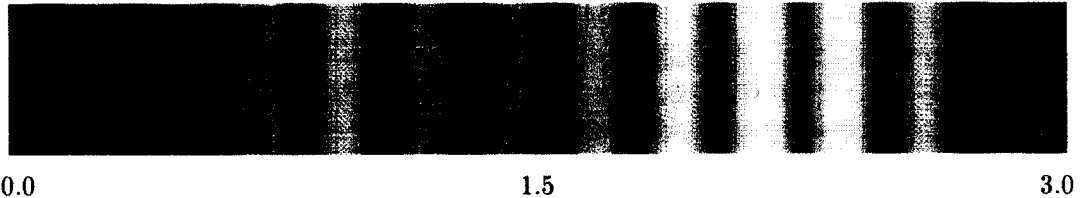


図 3.39: 2 次元衝撃波伝搬問題におけるカラーバー: 密度 0 ~ 3 を線形に色分けした。

また、図 3.41 に、エントロピーを  $S = PV^\gamma$  で評価した結果を示す。図 3.39 に示すカラーバーを最小値を 0、最大値を 50 とし、線形に 256 分割して色を決定している。

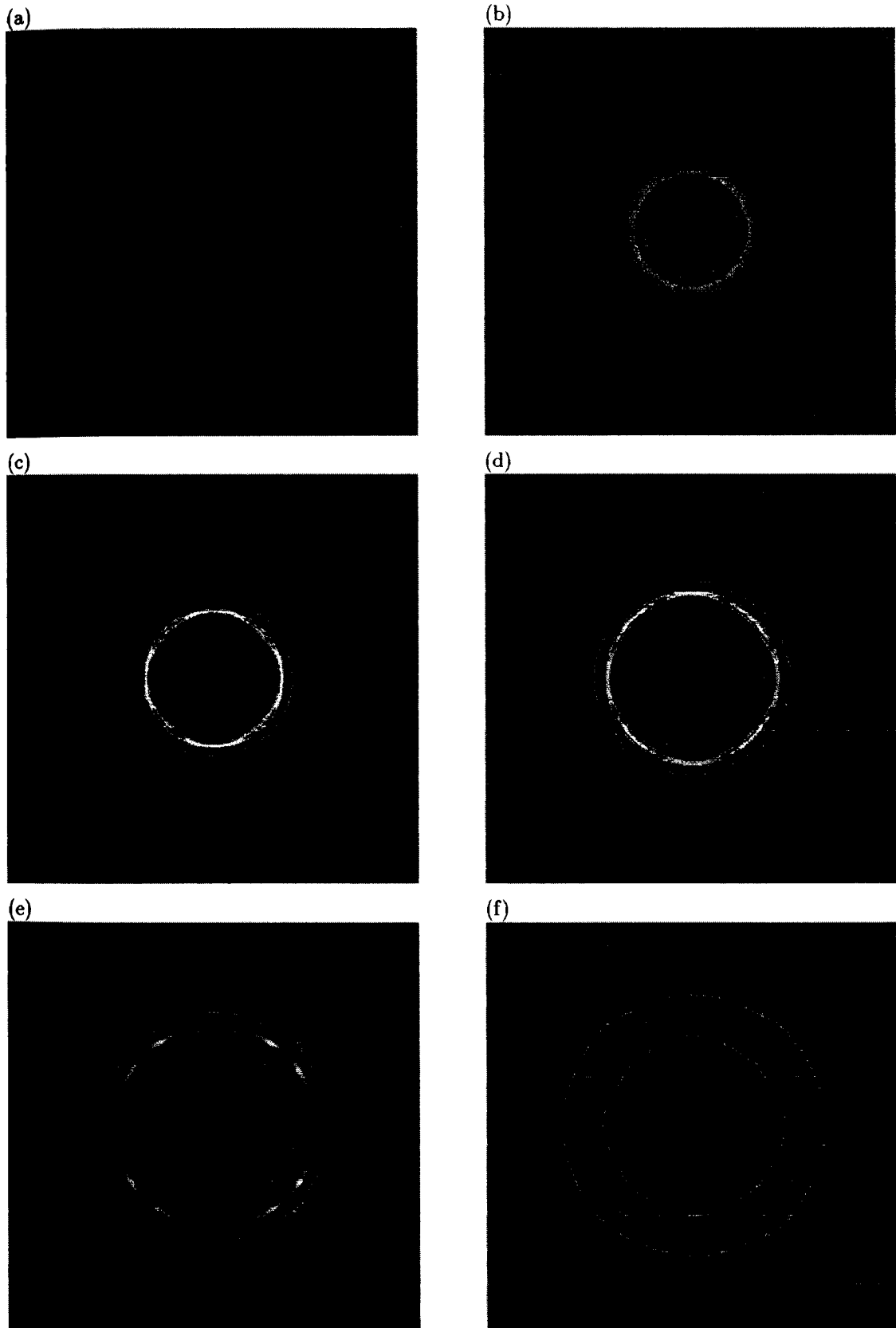


図 3.40: 2次元衝撃波伝搬問題:

(a) - (f) はそれぞれ  $t = 0, 3.13 \times 10^{-2}, 6.96 \times 10^{-2}, 1.73 \times 10^{-1}, 1.10 \times 10^{-1}, 1.53 \times 10^{-1}, 1.96 \times 10^{-1}$  における密度プロファイルを表している。

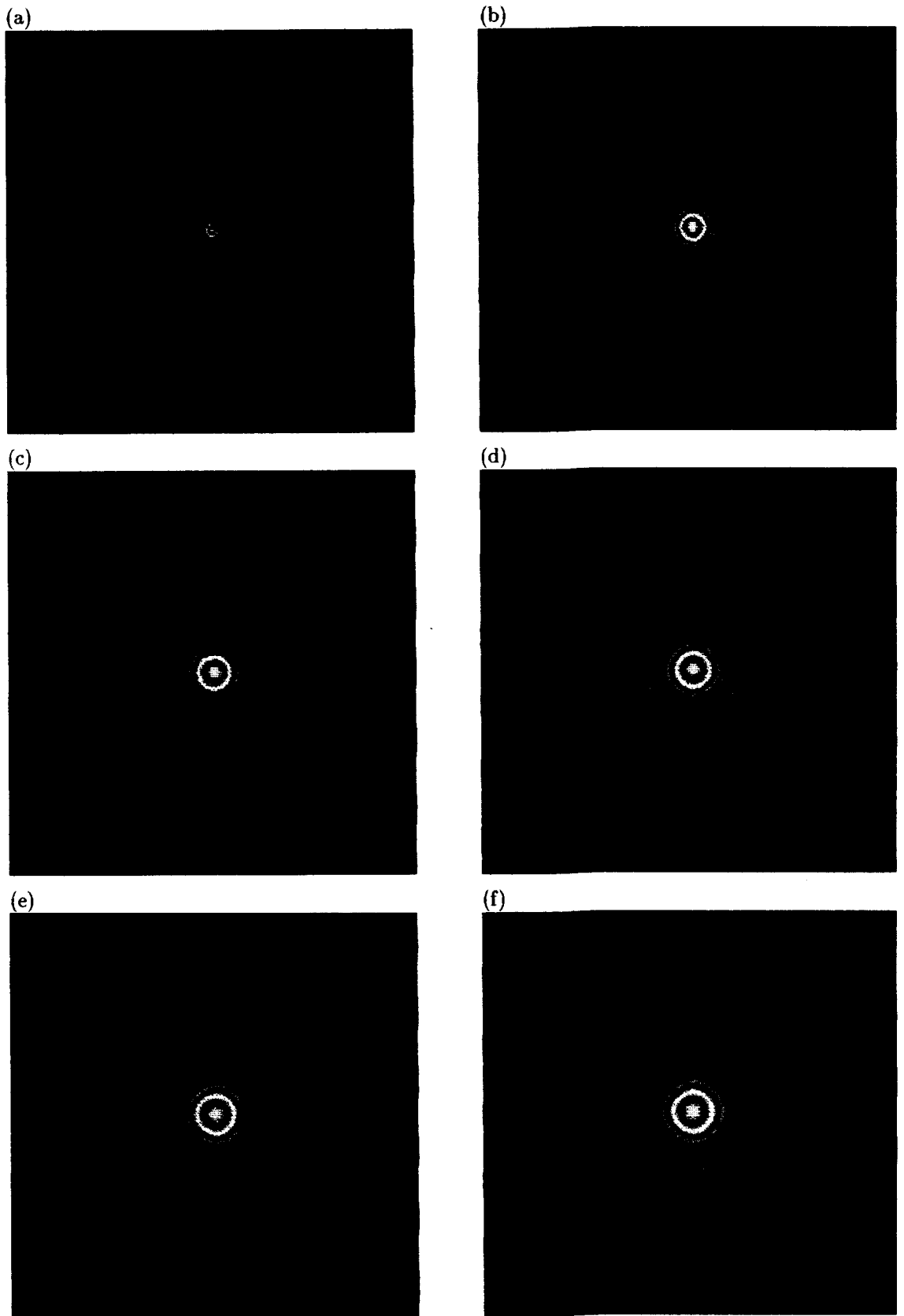


図 3.41: 2次元衝撃波伝搬問題:

(a) - (f) はそれぞれ  $t = 0, 3.13 \times 10^{-2}, 6.96 \times 10^{-2}, 1.73 \times 10^{-1}, 1.10 \times 10^{-1}, 1.53 \times 10^{-1}, 1.96 \times 10^{-1}$  における密度プロファイルを表している。

初期プロファイルでは密度が一様であるが、高圧力場を適用することで計算空間中心において急激に密度が下がり、同心円状に衝撃波が伝搬して行く様子が良くとらえられている。また、エントロピーを評価した図 3.41 から、衝撃波の移動に伴うエントロピー等値面の伝搬が観測された。弱い衝撃波が伝搬する場合には流体は等エントロピー的運動を行うことから、得られた計算結果の妥当性を示すことができた。

この例題に対して  $c_0 = 0, c_L = 0.2, 0.4, 0.6, 0.8$  とした結果を図 3.42 に示す。また、 $c_0 = 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, c_L = 0.0$  とした計算結果を図 3.43 に示す。 $c_0 = 0$  を用いた場合には  $c_L < 0.2, c_L > 0.8$  での数値計算は前者に対しては 1 次元の場合と同様に衝撃波を適切に記述することができなため数値不安定性が生じ、数値計算が困難であった。後者では図 3.42 (d) でも見られるように、衝撃波面に数値不安定性が見られるようになり波面が破壊する結果となった。 $0.2 \leq c_L \leq 0.8$  では衝撃波はそれぞれの計算結果から同心円状に伝搬している様子が伺えるが、 $c_L$  の値を大きくするにつれて衝撃波面、膨張波面が四角い形状をとる傾向が見られた。これは、直交等間隔グリッドを用いていることに原因があり、グリッド依存性が生じることを表していると考えられる。 $c_L = 0$  とし、 $c_0$  を変化させた場合では、 $c_L < 0.4, c_L > 1.4$  では同様に数値不安定性により数値計算を行うことが困難であった。人工粘性は拡散項として作用することから、 $c_L$  を大きくした際に生じる不安定性は第 5 章で行う陰解法化により安定な計算が可能と考える。

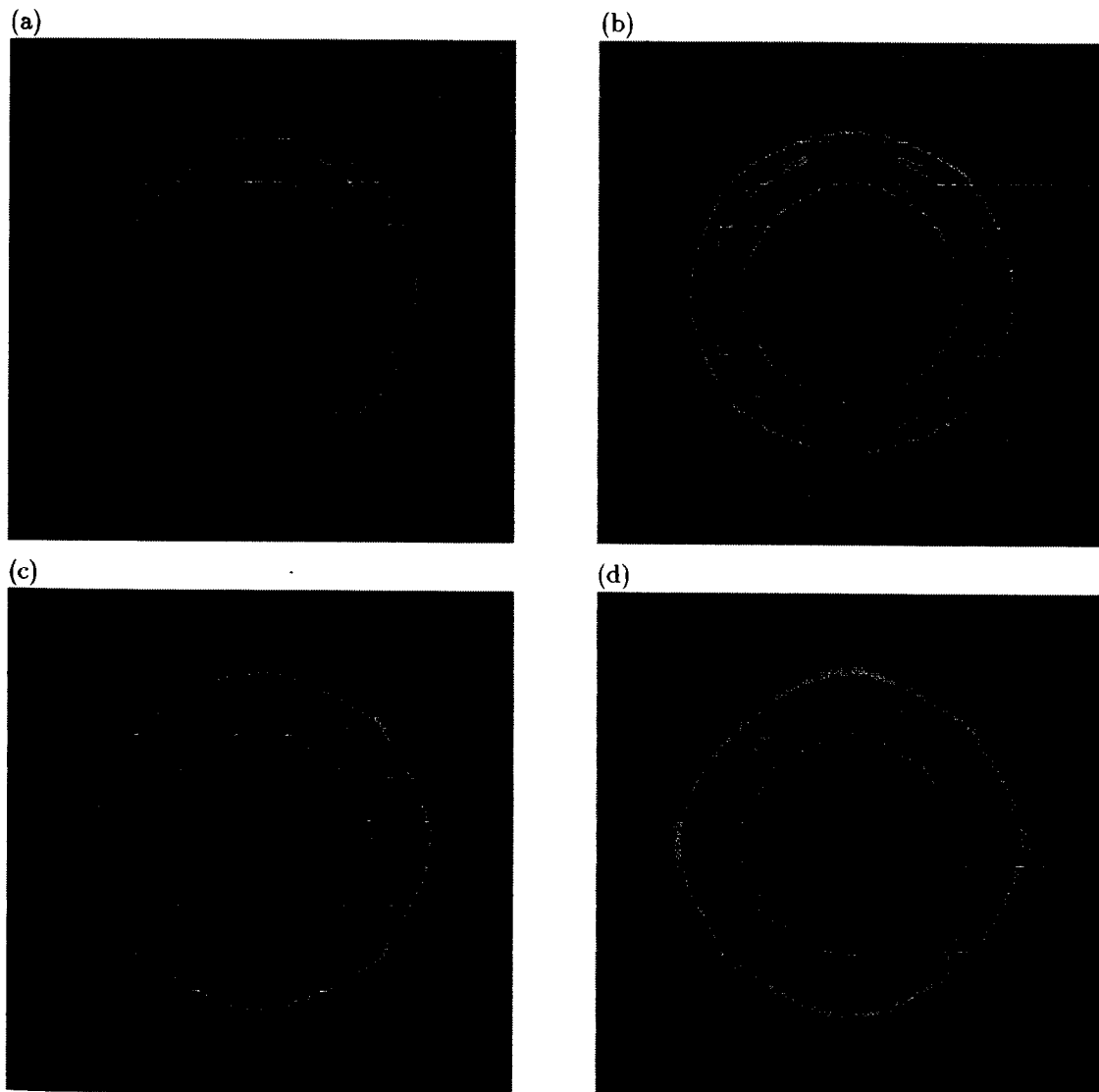


図 3.42:  $c_0 = 0, c_L = 0.2, 0.4, 0.6, 0.8$  を用いた 2 次元衝撃波伝搬問題

(a) - (f) はそれぞれ  $c_0 = 0, c_L = 0.2, 0.4, 0.6, 0.8$  を用いた、時刻  $t = 1.96 \times 10^{-1}$  における密度プロファイルを表している。

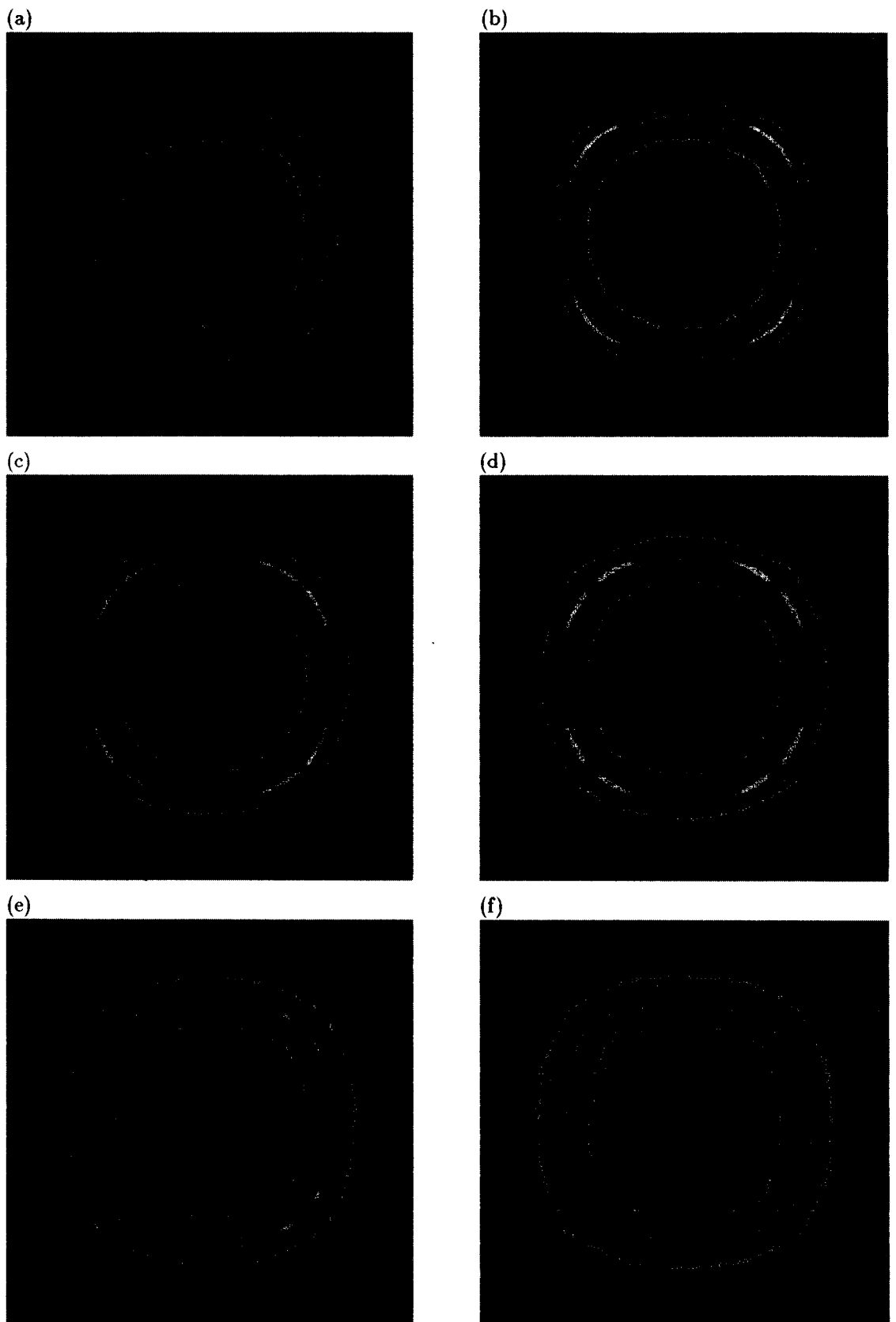


図 3.43:  $c_0 = 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, c_L = 0.0$  を用いた 2 次元衝撃波伝搬問題:  
 (a) - (f) はそれぞれ  $c_0 = 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, c_L = 0.0$  における密度プロファイルを表している。

また、図 3.43 から分かるように  $c_0$  により人工粘性を考慮した場合、衝撃波面が直交等間隔グリッドの性質を反映し、ほぼ正方形に伝搬している様子が伺える。また、 $c_0$  を用いる von Neumann 型の人工粘性では  $c_L$  を用いる人工粘性に比べてその係数を大きくするに従い速やかに衝撃波面が数値的に拡散する様子を捉えることに成功した。

グリッド依存性を考慮した場合 von Neumann 型人工粘性に比べ Landshoff 型の人工粘性が有効な解を示すと考えられることから、今後の数値計算では  $c_L = 0.8$  を用いた Landshoff 型の人工粘性を扱うこととする。

### 3.5.5 2 流体モデル

超音速混合流問題では、密度、圧力、流速、比熱比 ( $\gamma$ ) の異なる 2 流体を混合させるため、異なる物質を計算空間中で適切に扱う手法について考慮する必要がある。図 3.44 に示すように、計算領域内で異なる物質の密度を明示的に与え、2 流体を独立に扱う方法が一つの解決手法であると考えられる。

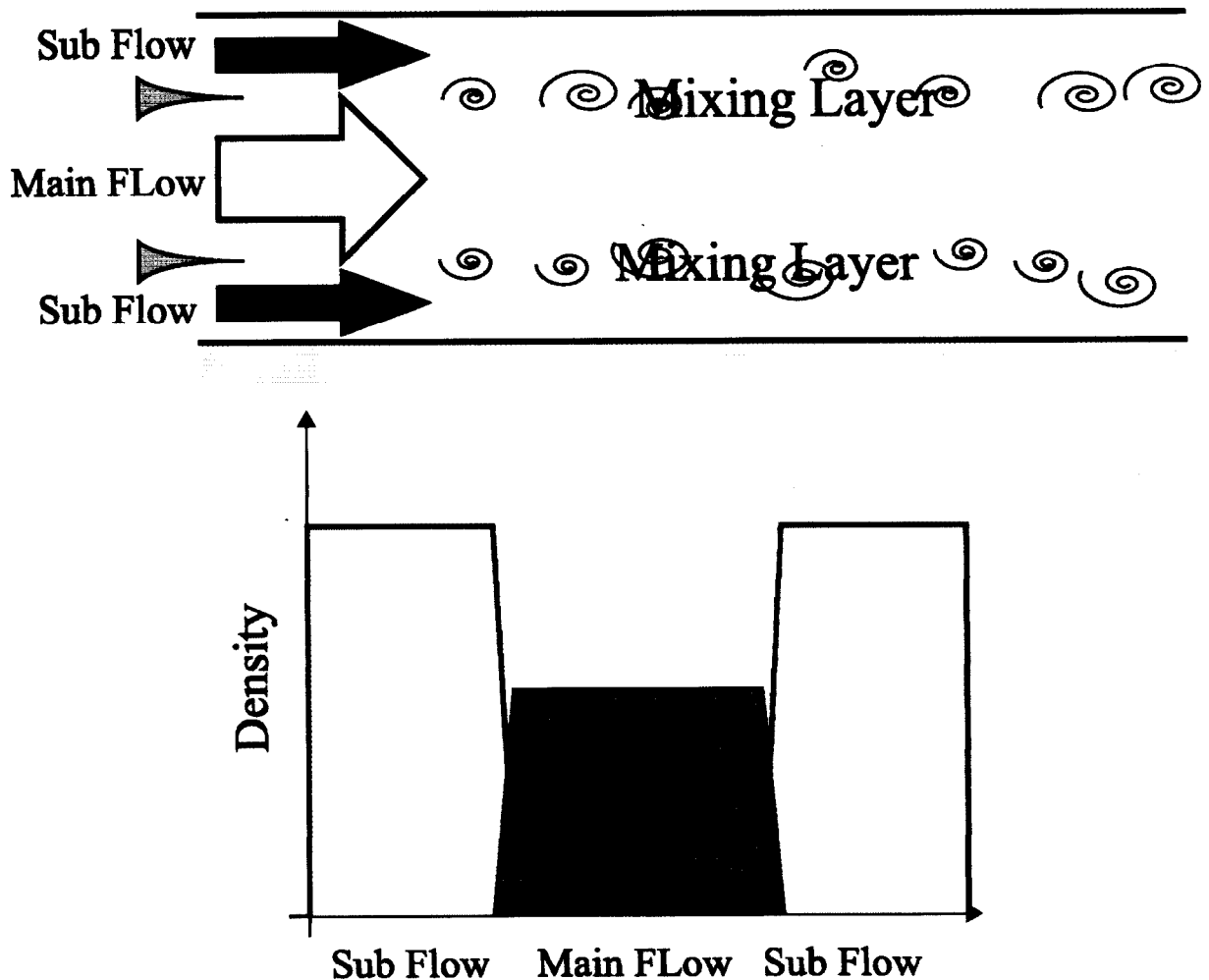


図 3.44: 密度関数を用いた 2 流体混合モデルの例

密度、速度、圧力をそれぞれの流体に対して定義することで 2 流体を記述する。

それぞれの流体が持つ物理量 (密度、分圧、流速、物性値) を個別に定義することで 2 流体を記述するこ

とが出来るが、IDO 法で本モデルを用いた場合、急激な勾配を記述する際にアンダーシュートが生じることから、空間的にわずかな領域ではあるが圧力ならびに密度が負となる計算グリッドが生じるため計算が不安定になることが考えられる。この問題の解決策として、物理量、基礎式を  $\tan$  関数を用いて変換する手法、 $\exp$  関数を用いた変換などが考えられるが、圧縮性流体を扱っていることや、これらの変換によって計算精度低下が考えられるため、より単純な手法の導入を検討する。

他の手法として識別関数を用いることが考えられる。その一例として Level Set 法を採り入れたモデル導入方法が考えられる。Level Set 関数 ( $\Psi$ ) が 0 となる面を流体の境界面と定義し、正值をとる領域では流体 A、負値をとる領域では流体 B として扱うことで空間に連続的に与えた物理量を用いて 2 流体を記述することが出来る。しかし、Level Set 関数を用いた手法は 2 流体を完全に分離してその境界面を追跡する手法であるため、その混合過程を検討することが出来ない。そこで、単純に比熱比を識別関数として用いる手法を用いる。

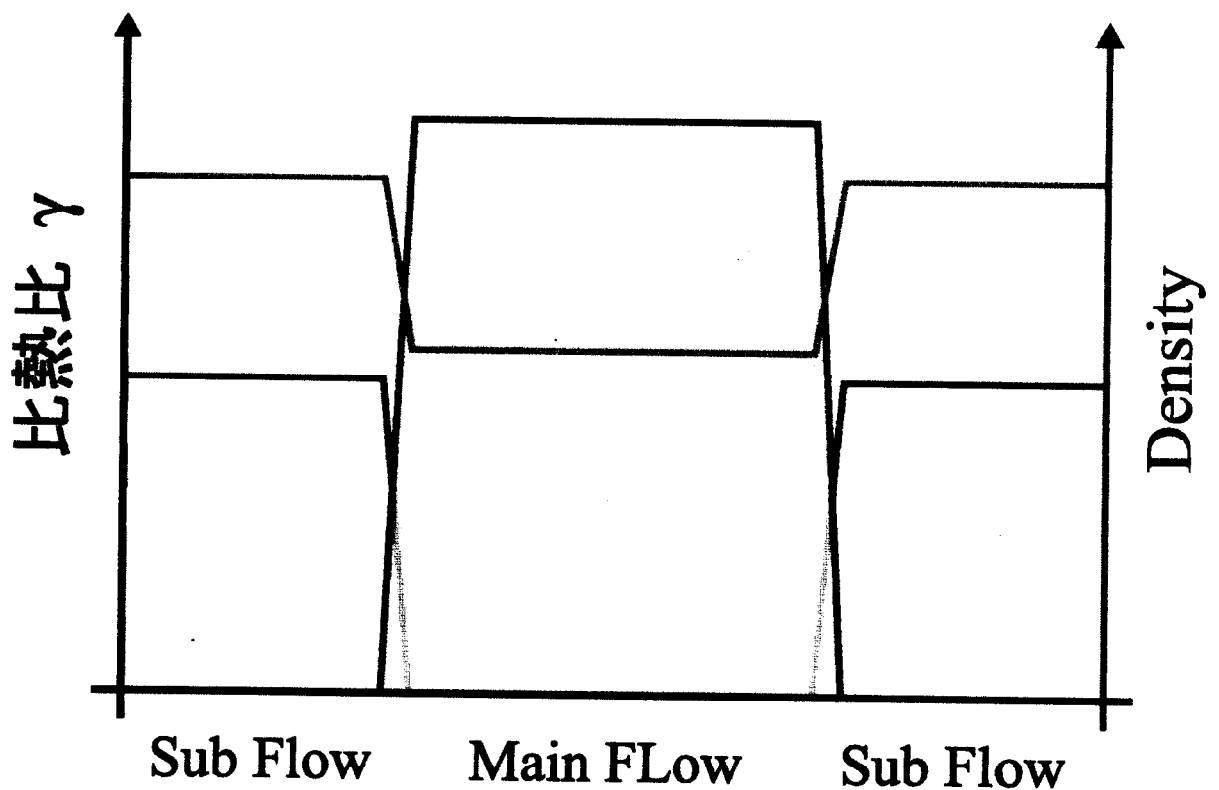


図 3.45: 比熱比を識別関数として用いる 2 流体混合モデル

比熱比がアンダーシュートにより負値をとった場合でも計算が破綻することはないことを利用し、識別関数として用いる。

各作動流体、ここでは主流、副流に対してそれぞれ比熱比を関数として定義する。図 3.45 に示すオレンジ色の部分が副流の比熱比を表し、青色の部分が主流の比熱比を表している。密度、圧力、流速に関しては赤線で示すように計算空間中で 1 関数により表記するため連続に与えられ、このことから密度が負になる状況は生じることが無い。この連続して与えられた物理量を示す関数に対して比熱比を示す識別関数を考慮することで対象となる流体の混合を捕らえることが可能になる。その一方で図 3.44 で示したものと同様に比熱比  $\gamma$  がアンダーシュートにより負値をとる問題が生じるが、数値計算がそれによって破綻することはないため問題に対して何らかの解を得ることが出来る。また、数値拡散による流体混合の発

生についても問題点として考えられるが、数値拡散は主として移流方向に発生し、今回用いるグリッドは移流方向に対して平行に直交等間隔格子を適用することから、流速と垂直方向の数値拡散は比較的少ないものとする。また、識別関数は移流速度により時間発展することから、流体境界面を比較的良好的に記述できるものと考えられ、識別関数の混合も可能な手法である。

具体的には識別関数として用いた比熱比を各グリッド上で平均し、

$$\gamma_{\text{Fluid}} = \frac{\gamma_{\text{FluidA}} + \gamma_{\text{FluidB}}}{2} \quad (3.213)$$

により示される平均された比熱比を用いて流体方程式を扱う。このことから、流入条件として分布を与えた連続な密度、圧力、流速を用い、2流体モデルを記述出来るものとする。

### 3.5.6 計算パラメータ

本章で扱う問題は実際の実験との比較を目的としている。その実験装置並びにパラメータについて述べる。

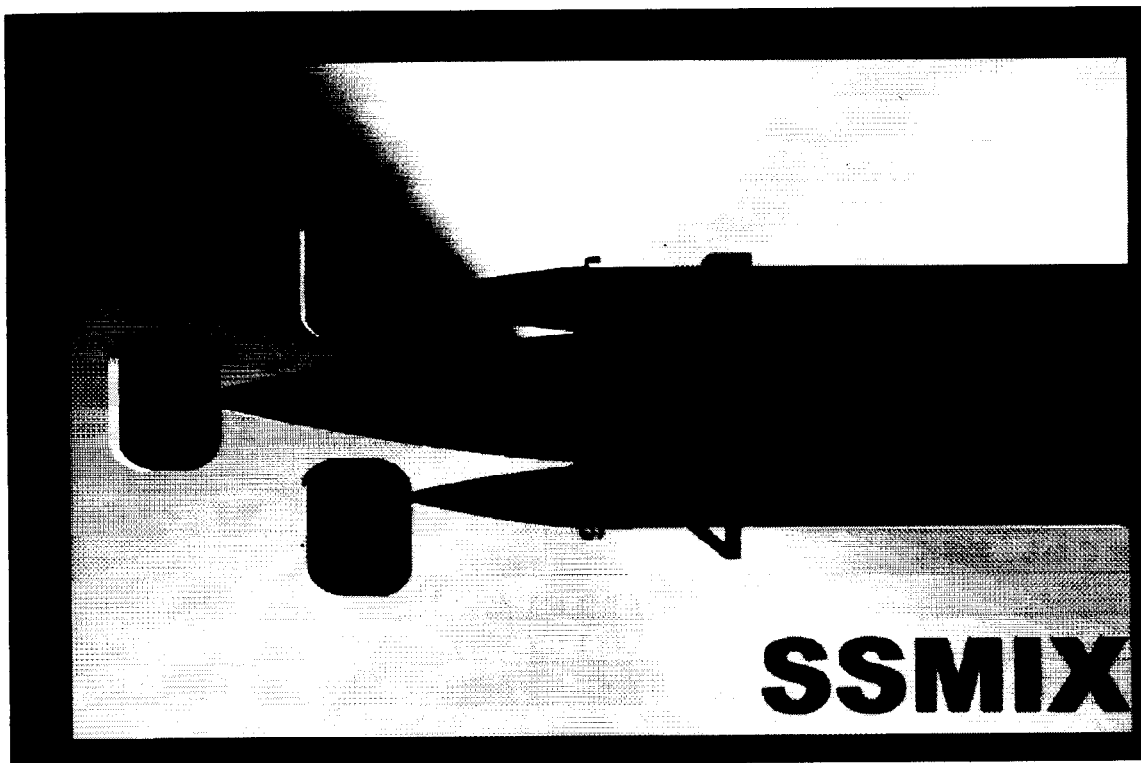


図 3.46: 超音速混合実験装置 SSMIX テストセクション部の概念図

中心に位置するメインフロー用ノズルとその上下に配置されたサブフロー用ノズルにより形成され、下流方向に 350mm の等高さ部を持つ。

計算対象となるテストセクション部について説明する。超音速混合実験装置 SSMIX (Super Sonic MIXture) は、中心に主流用ノズルを持ち、その上下に副流用のノズルを配置しており、超音速混合は主流と副流の境界層で生じる。主流ノズルの高さは 22[mm]、副流ノズル高さは 9 [mm] で、特性曲線法を用いて出口で平行流となるように設計されており、厚さ方向は 20[mm] となっている。計算では、流れ下流方向 400[mm] までの計算領域を確保する。



表 3.3: 20 °C、1 気圧下の気体の粘度並びにサザランドの定数

	粘度 ( $\times 10^{-6}$ Pa s)	サザランドの定数 C
窒素ガス	17.6	104
ヘリウムガス	19.6	72.9
アルゴンガス	22.3	142

表 3.2: 実験での標準状態における主流、副流の各パラメータ

	圧力 [kPa]	温度 [K]	音速 [m/s]	速度 [m/s]	質量流量 [g/s]	比熱比	マッハ数
主流	3.877	71.43	172.27	689.06	55.44	7/5	4
副流	3.877	76.15	192.91	689.06	31.28	5/3	3.572

それぞれのノズル出口における流体の速度勾配、圧力勾配は、流体を構成するガスの質量分率ならびに淀み点圧力を変化させることで独立に変化させることが可能である。主流並びに副流のノズル出口圧力、音速、流速は標準状態 (流体間圧力差、速度差無し) では表 3.2 に示す値をとるように設計されている。

数値計算では実験パラメータを SI 単位系で直接扱うこととする。気体定数  $R$  [J/kg·K] は主流を構成する窒素、副流を構成するアルゴン、ヘリウムについてそれぞれ 296.75、200.19、2078.0 を用いた。また、副流の質量分率は  $c_{Ar} : c_{He} = 0.95 : 0.05$  である。

各作動流体の動粘性係数に関しては表 3.3 に示す 1 気圧 20°C における粘性と、サザランド (Sutherland) の定数  $C$  から、関係式

$$\eta_2 = \eta_1 \left( \frac{T_1 + C}{T_2 + C} \right) \left( \frac{T_2}{T_1} \right)^{\frac{3}{2}} \quad (3.214)$$

を用いて対象温度における粘度を計算し、計算領域における密度で割ることで求める。混合領域では更に平均値をとる。

数値計算では、主流、副流のノズル出口温度がそれぞれ 71.43[K]、76.15[K] であることから、それぞれの粘度を

$$\eta_{N_2} = 17.6 \times 10^{-6} \left( \frac{293.15 + 104}{71.43 + 104} \right) \left( \frac{71.43}{293.15} \right)^{\frac{3}{2}} = 4.79 \times 10^{-6} \quad (3.215)$$

$$\eta_{Ar} = 22.3 \times 10^{-6} \left( \frac{293.15 + 142}{76.15 + 142} \right) \left( \frac{76.15}{293.15} \right)^{\frac{3}{2}} = 5.89 \times 10^{-6} \quad (3.216)$$

$$\eta_{He} = 19.6 \times 10^{-6} \left( \frac{293.15 + 72.9}{76.15 + 72.9} \right) \left( \frac{76.15}{293.15} \right)^{\frac{3}{2}} = 6.37 \times 10^{-6} \quad (3.217)$$

として用いる。

次に、初期密度について考察する。初期密度はノズル出口面積並びに質量流量、流速より簡単に見積もる。

実験では、副流を構成する作動流体の構成比を変化させることで流体間速度差を与え、その影響を考察可能であるように設計している。ただし、ノズルの設計上マッハ数を変化させることは不可能である。数

値流体計算でも流体間速度差による影響を考察することを目的として、実験結果が得られている副流にアルゴンガスのみを用いた計算パラメータ (Pattern2) について検討を行う。その際の流入条件は、マッハ数が同じであることから、

$$\frac{U_{\text{標準状態}}}{a_{\text{標準状態}}} = \frac{U}{a} \quad \Rightarrow \quad \rho = \rho_{\text{標準状態}} \frac{P}{P_{\text{標準状態}}} \frac{U_{\text{標準状態}}^2}{U^2} \quad (3.218)$$

の関係が得られる。初期密度を求めると以下ようになる。

表 3.4: 超音速混合問題に対する計算パラメータ:

実験結果との比較を目的として、以下に示す 2 パターンについて数値実験を行う。

	$\rho_{N_2}$ [kg/m <sup>3</sup> ]	$\rho_{Ar}$ [kg/m <sup>3</sup> ]	$\rho_{He}$ [kg/m <sup>3</sup> ]	$\rho_{\text{sub}}$
Pattern 1	0.183	0.240	0.0126	0.252
Pattern 2	0.183	0.254	0.0	0.254

以上をまとめると、流入条件は表 3.5 に示すものとなる。

表 3.5: 超音速混合流問題における主流ならびに副流における各初期条件。

Pattern 1	圧力 [Pa]	密度 [kg/m <sup>3</sup> ]	速度 [m/s]	粘度 [ $\times 10^{-6}$ Pa s]	比熱比
主流	3877	0.183	689.06	4.79	1.4
副流	3877	0.252	689.06	5.91	1.67
Pattern 2					
主流	3877	0.183	689.06	4.79	1.4
副流	3877	0.254	569.36	5.89	1.67

### 3.5.7 計算結果

超音速混合流問題が持つ流体構造の大きさの最小値は、乱流混合を考えるとほぼ分子レベルとなる。しかし、連続体近似を行う Navier-Stokes 方程式からはそのような構造を得ることが出来ない。しかし、直接数値計算を目的としていることから、乱流モデルなどの付加的な項を導入することはここでは考えない。このことから、空間解像度と得られる流体構造との整合性を検討することを目的として、同じ現象をグリッド数を変化させつつ検討を行い、ほぼ同等の流体構造が得られる計算グリッド数により表 3.5 に示すパラメータにより検討を行う。

#### グリッド解像度に関する考察

グリッド解像度を変化させた数値計算を行い、同程度の流体構造が得られる計算格子点数を見積もる。例題として、標準状態 (Pattern 1) を用いた。

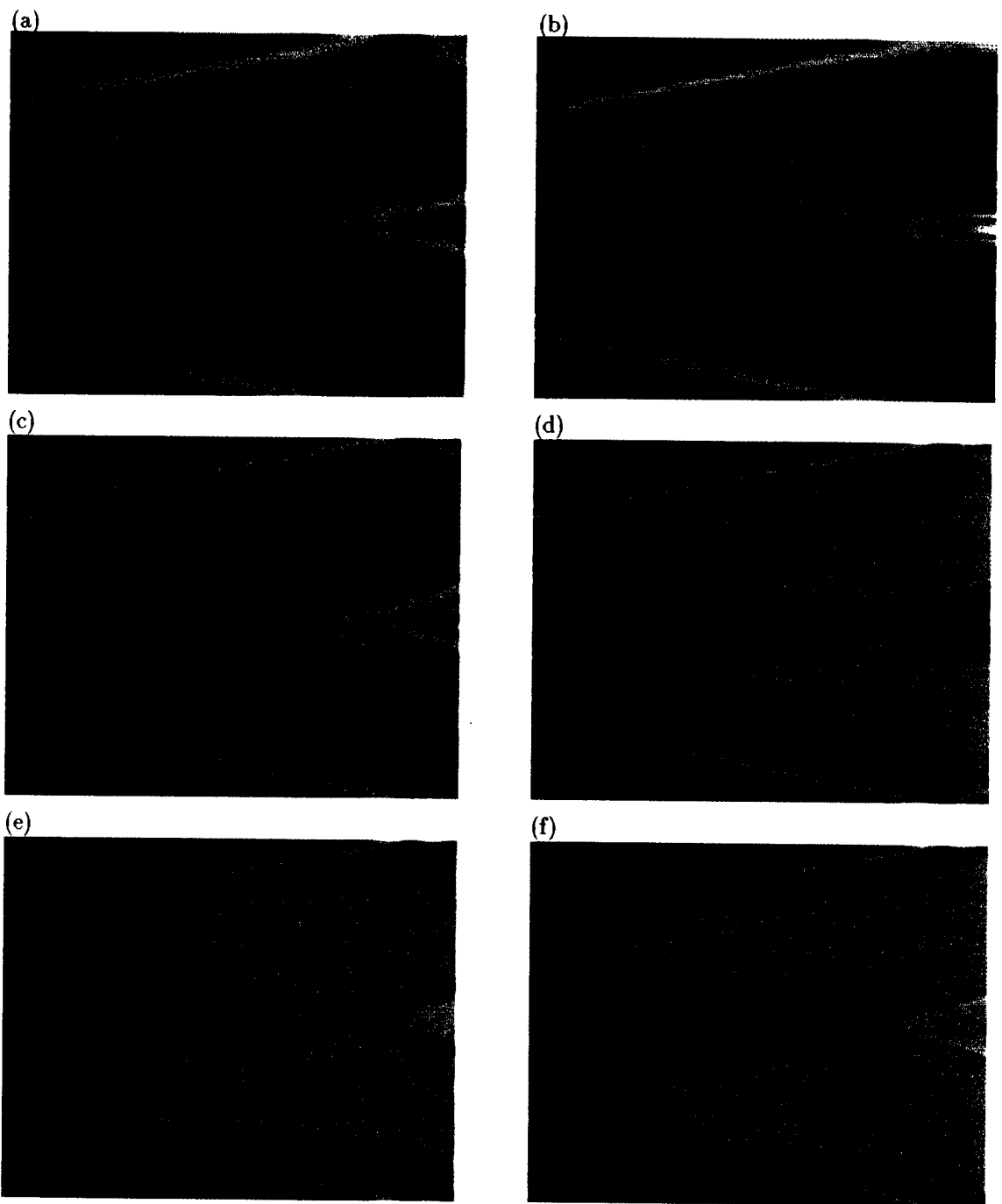


図 3.47: グリッド解像度に対する超音速混合流の流体構造:

(a)  $20 \times 16$ 、(b)  $25 \times 20$ 、(c)  $50 \times 40$ 、(d)  $100 \times 80$ 、(e)  $200 \times 160$ 、(f)  $300 \times 240$  のグリッド解像度により、標準状態での混合流問題を扱った結果。

図 3.47 に示すように計算格子点数が少ない場合には同じ流体問題を扱っているにも関わらず、他の数値計算結果から得られる構造と比較して、異なる結果としてとらえられる。その一方で、IDO 法は少ない格子点数で高精度計算が可能なることから、図 3.47 (b) に示す  $20 \times 16$  の計算格子点数でも流体構造としてはその 10 倍の格子点数を用いた図 3.47 (f) に示す  $200 \times 160$  の解像度から得られる流体構造とほぼ同等の結果を示した。この問題に関しては解析解が存在しないため、グリッド解像度に対する流体問題への適

用性に関する考察は難しいが、一般的にグリッド解像度が高くなるにつれて、流体計算は高精度化する。IDO 法に関しても同様の傾向が見られることを先の 1 次元衝撃波管問題などから示した。しかし、不必要に高解像度グリッドを用いることは計算コストの面から好ましくないことは明らかである。今回問題の対象としている微小衝撃波列 (Shock Let) の構造がサブ mm 程度になることが実験結果から示されているため、今後の計算は図 3.47 (d) による結果が図 3.47 (f) に示す流体構造とほぼ同等の結果を得ており、図 3.47 (d) のグリッド間隔が、 $20[\text{mm}]/80[\text{Grids}] = 0.25[\text{mm}/\text{Grid}]$  となり、更に IDO 法ではエルミート補間を行うことからグリッド間隔以下の空間解像度を得ることが可能であることから、 $100 \times 80$  の計算空間解像度での計算を進めることとする。

#### 実験パラメータを用いた数値計算および流体構造に関する考察

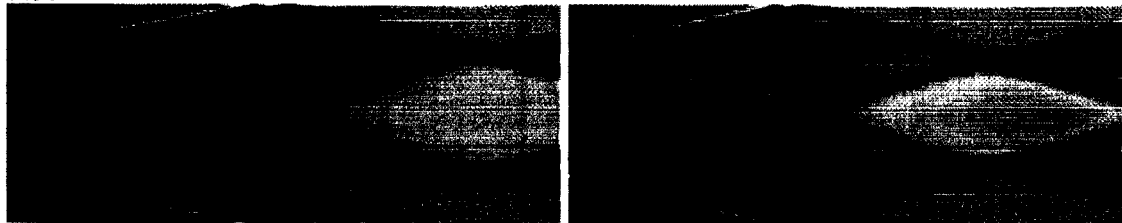
実験との対比を行うことを目的として、実験で設計されたパラメータに対して数値計算を行い、その傾向について議論する。超音速流問題では基本的には下流側の流体構造が上流側の流体構造に影響しないことが保証されていることから、上下壁面に反射境界条件を適用し、ノズル出口において実験パラメータを用いた流入条件、ディフューザー方向に流出境界条件を適用する。

初期プロファイルとしてノズル出口におけるパラメータを流れ方向に対して一様に与えた。これは、超音速流問題においてはいかなる初期条件を与えた場合でも十分な時間が経過した後は同じ数値計算結果を得ることができるため適当である。実験パラメータによる数値計算結果を図 3.48 に示す。それぞれの図は上から密度、圧力、比熱比の分布関数を表している。

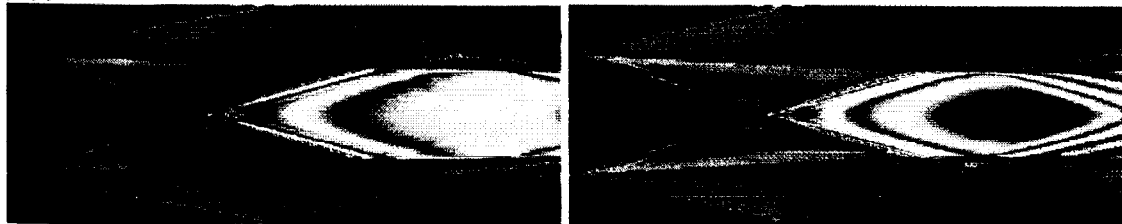
Pattern 1

Pattern 2

密度



圧力



識別関数

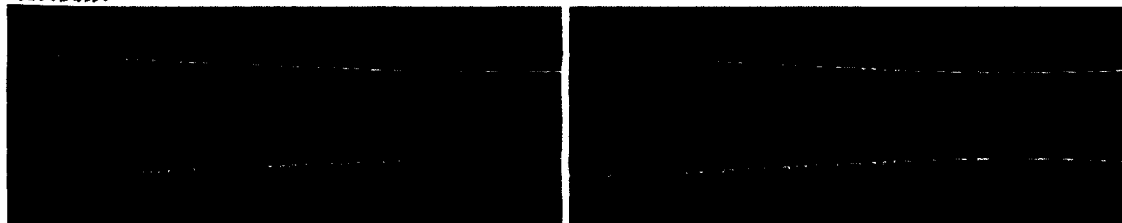


図 3.48: 超音速混合流問題の数値計算結果:

それぞれの計算結果は密度、圧力、比熱比の分布関数を示している。数値計算から得られた結果では混合層の成長を確認することは困難であった。

速度差の有無にかかわらず下流 100mm までには混合層の目立った発達は観測されず、流体構造に関してもほぼ同様な結果を得た。本数値計算では流体粘性を付加しているとは言え、標準状態における流入条件下で主流、副流の動粘性係数はそれぞれ

$$\mu_{\text{主流}} = \frac{4.79 \times 10^{-6}}{0.183} = 2.62 \times 10^{-5} \quad (3.219)$$

$$\mu_{\text{副流}} = \frac{5.91 \times 10^{-6}}{0.252} = 2.35 \times 10^{-5} \quad (3.220)$$

となる。代表長さをノズル出口高さ 22 mm、9mm としたレイノルズ数が

$$Re_{\text{主流}} = \frac{0.022 \times 689.06}{2.62 \times 10^{-5}} = 5.79 \times 10^5 \quad (3.221)$$

$$Re_{\text{副流}} = \frac{0.009 \times 689.06}{2.35 \times 10^{-5}} = 2.63 \times 10^5 \quad (3.222)$$

と求められ、流体間の粘性が流体の持つ慣性力と比較してほとんど無視できることから今回得られた計算結果に差が見られないことが理解できる。

しかし、図 3.50 で示す実験結果からは混合層付近からの微小衝撃波列の生成が確認され、混合領域の成長も見られる。実験におけるセットアップと今回用いた計算条件を比較した場合、その結果を大きく変える要因として考えられるのがノズル出口 (流入条件) における速度擾乱が考えられる。実験では主流ノズルに相接するように副流ノズルが配置されている。

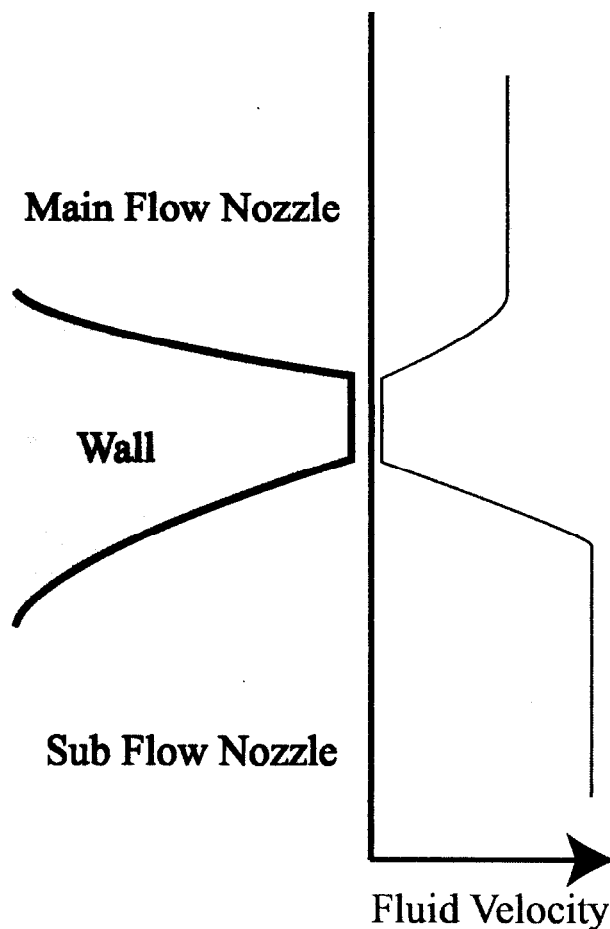


図 3.49: ノズル接合部における速度不連続:

実験では主流ノズル、副流ノズルを接合する場合、数学的な 0 厚みで接合することは困難である。そのため、ノズル出口において速度不連続が生じる。



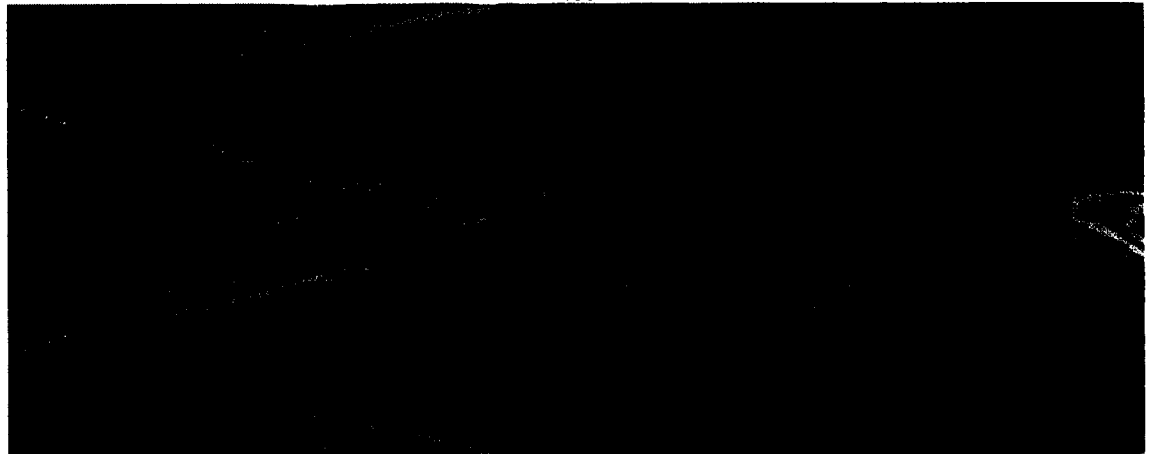
図 3.50: 実験結果から得られる超音速混合流問題の流体構造:  
実験から超音速 2 流体混合領域で一定間隔に現れる微小衝撃破列 (Shocklet) と呼ばれる構造を捕らえることができる。

Pattern 1

密度分布



圧力分布



分配関数

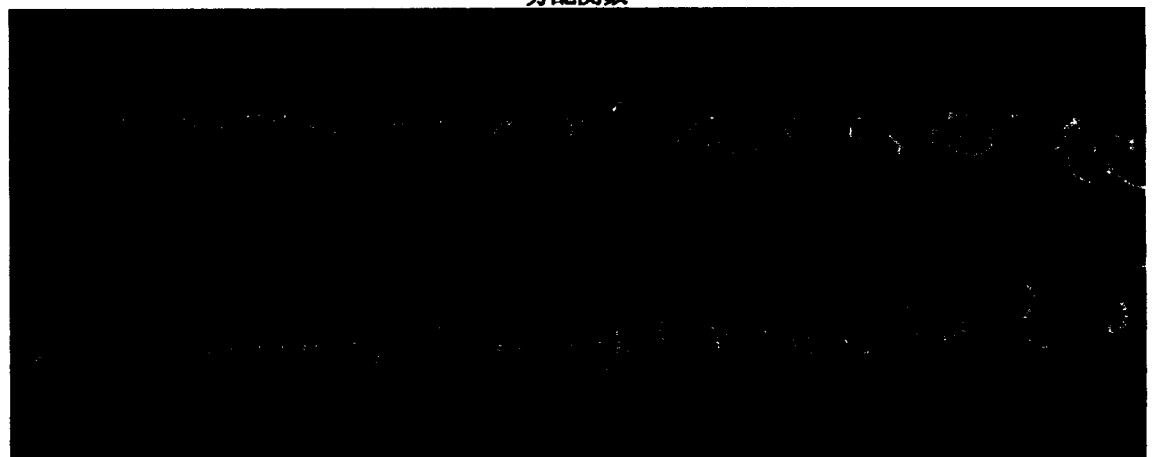


図 3.51: ノズル出口における速度擾乱を考慮した超音速混合流:  
ノズル接合部は有限厚さを持つため実験ではそこで速度擾乱が発生することを考慮し、流入条件として主流-副流相接部において速度不連続となる点を考慮した結果擾乱の周期的な発生を観測することに成功した。

## Pattern 2

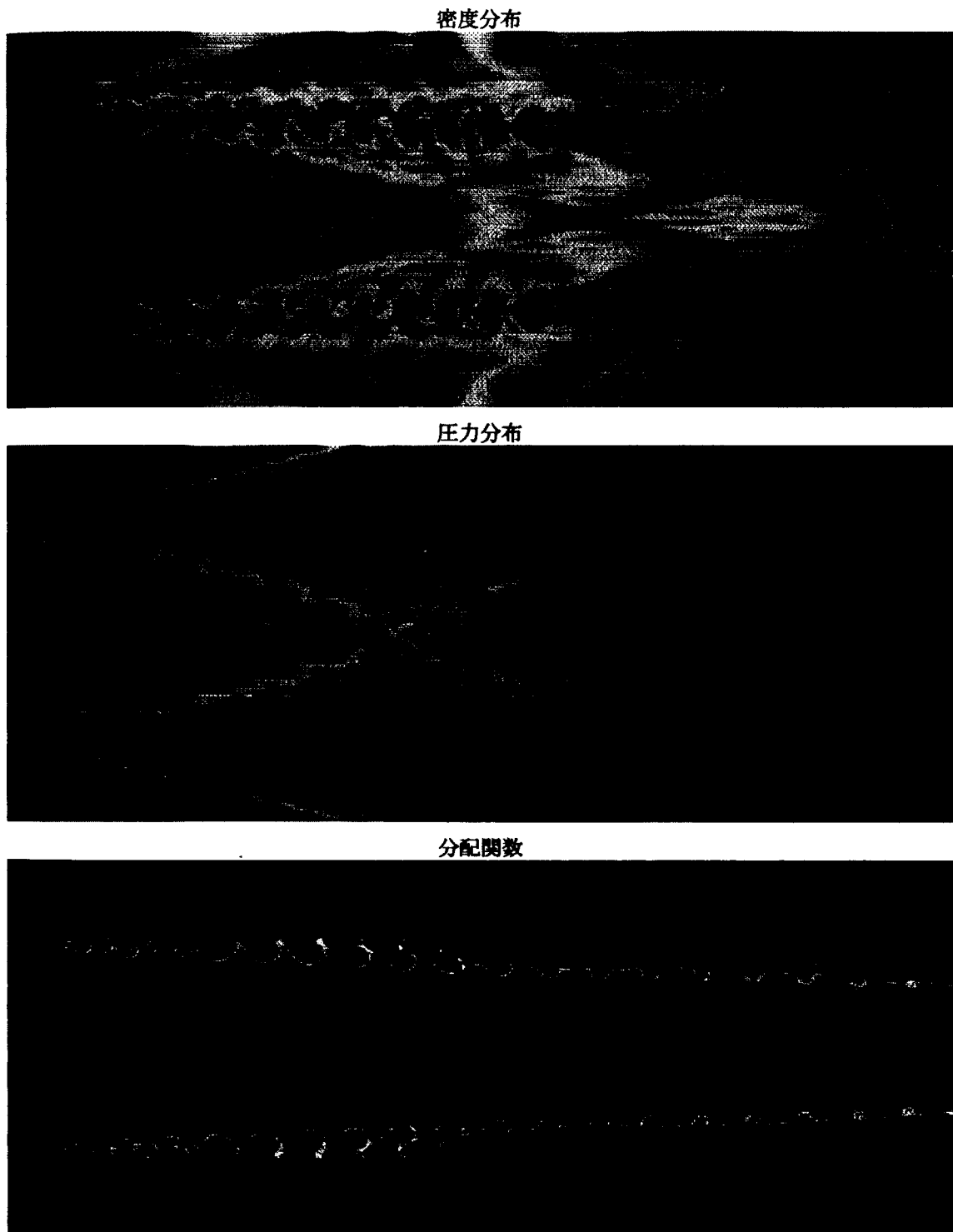


図 3.52: ノズル出口における速度擾乱を考慮した超音速混合流:

ノズル接合部は有限厚さを持つため実験ではそこで速度擾乱が発生することを考慮し、流入条件として主流-副流相接部において速度不連続となる点を考慮した結果擾乱の周期的な発生を観測することに成功した。



Pattern 1

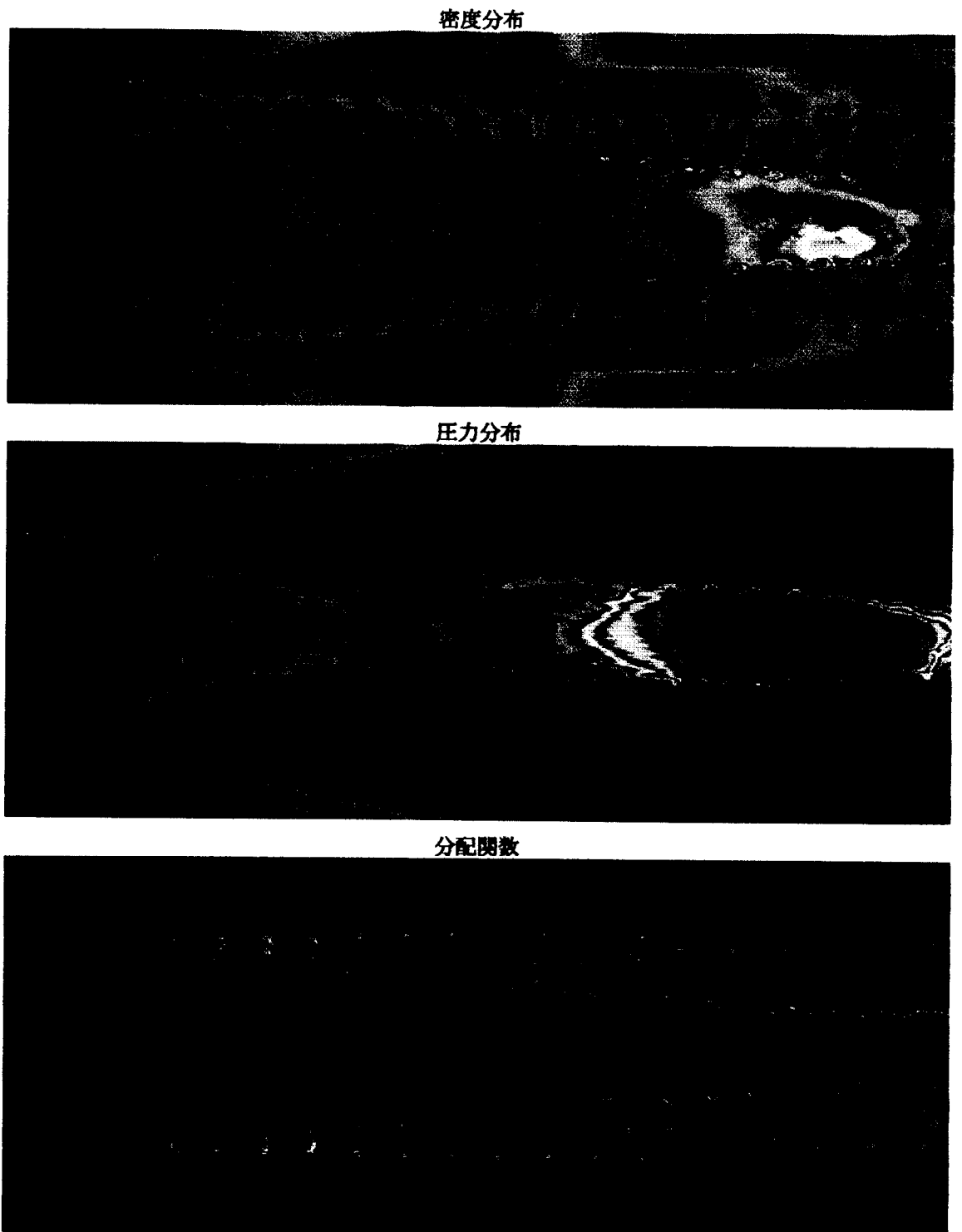


図 3.53: ノズル出口における速度擾乱を考慮した超音速混合流:  
ノズル接合部は有限厚さを持つため実験ではそこで速度擾乱が発生することを考慮し、流入条件として主流-副流相接部において速度 0 となる点を考慮した結果擾乱の周期的な発生を観測することに成功した。

## Pattern 2

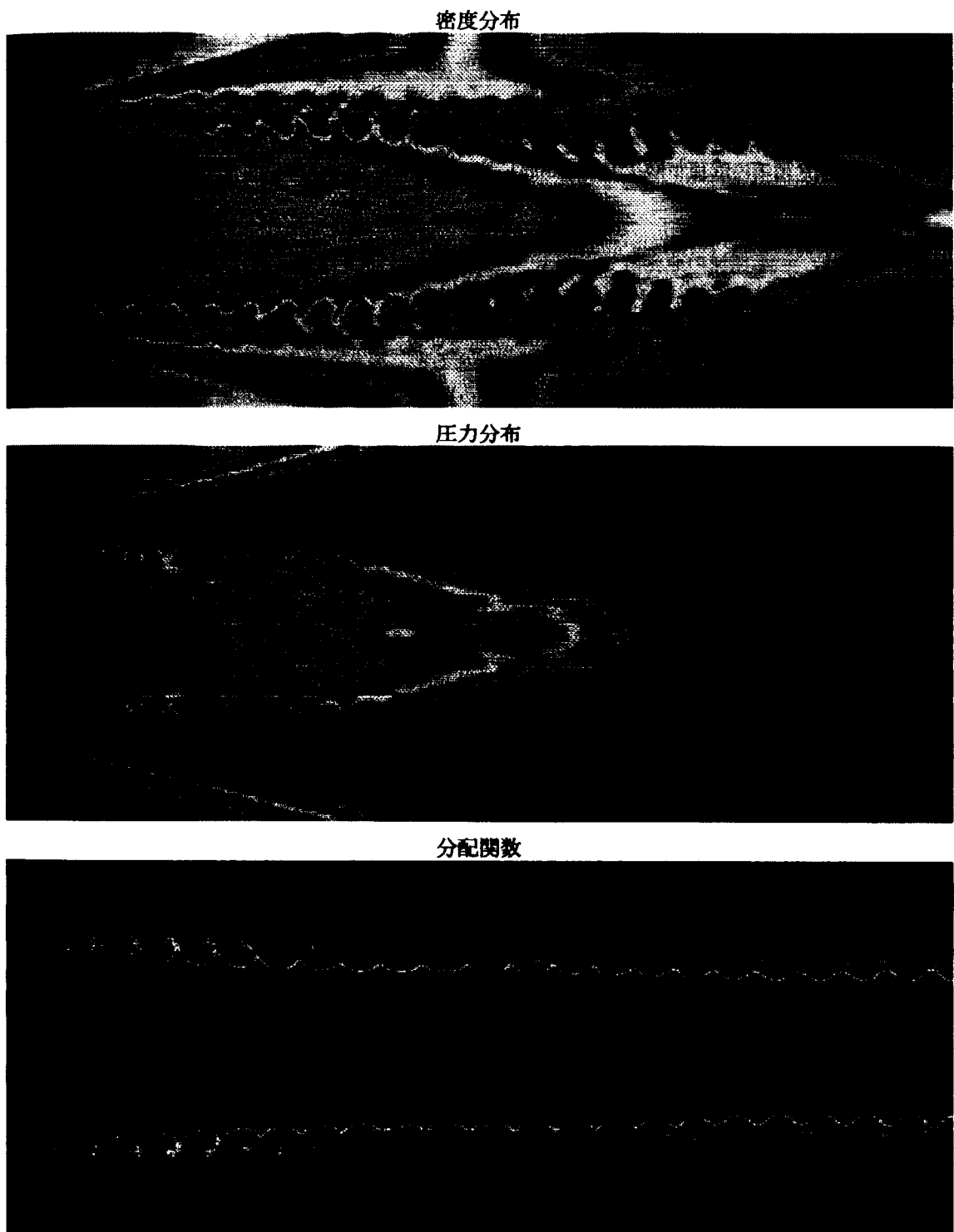


図 3.54: ノズル出口における速度擾乱を考慮した超音速混合流:  
ノズル接合部は有限厚さを持つため実験ではそこで速度擾乱が発生することを考慮し、流入条件として主流-副流相接部において速度不連続となる点を考慮した結果擾乱の周期的な発生を観測することに成功した。

図 3.49 に示すように現実的に主流ノズル、副流ノズル接合部の厚みを数学的 0 にすることは困難であるため、得られるノズル出口における流速分布は全ての実験条件下で不連続となる。そこで、ノズル出口主流-副流相接部において速度を低下させることを考慮する。更に厳密にはそのノズル接合部厚さならびに境界層厚さを吟味する必要があるが、擾乱の発生原因として利用することを考えることから流速を低下させる 1 グリッドを導入するにとどめた。得られた計算結果を図 3.51 ~ 図 3.54 に示す。図 3.51、図 3.52 では、ノズル接合部の流体速度を接合部厚さを考慮しない場合の 40% としており、図 3.53、図 3.54 では、ノズル接合部の流体速度を接合部厚さを考慮しない場合の 10% とした。ノズル出口において速度擾乱が存在することから混合領域の成長が擾乱の無い場合に比べて顕著に観測され、その構造が周期的に現れることを数値実験的に示した。

Pattern 1 では、主流-副流の速度差  $\Delta u$  が 0 であるため、ノズル出口における擾乱の成長は式 (3.177) から示される Kelvin-Helmholtz の理論からは成長することは考えられず、また本計算では重力を考慮していないことから主流-副流間で密度差はあるもののレーリー・テラー不安定性のような混合プロセスをとることもない。しかしその一方で現象を局所的に捉えると、風上で与えられた速度擾乱により流体構造が乱されその影響で流体境界面付近では速度差が生じるため、Kelvin-Helmholtz 不安定性的混合プロセスの駆動力となり、下流域まで擾乱の成長が持続すると捉えることができる。ただし、図 3.51 に示す結果からは、周期的な構造が上流 20 mm 付近で終了しており、それ以降の流体構造は不規則に乱れている。その一方で図 3.53 に示す結果では下流 100 mm まで周期的な構造を得ることに成功した。両結果ともに流体間速度差の無い標準状態により流入条件を扱っているが、初期速度擾乱の大小により流体境界面近傍の局所的な速度差が異なるため、不安定性の成長過程に差が生じる結果となった。

図 3.52、図 3.54 の結果からは両者ともに周期的な流体構造を持つ混合領域を観測した。これは、不安定性を促進する 2 流体間流速差  $\Delta U$  が下流域に至るまで保持され続けていることを示唆している。非圧縮性流体では 2 流体間流速差を大きくつけることは比較的困難であるが、超音速流においては僅かな擾乱により引き起こされる速度差が Kelvin-Helmholtz 不安定性的成長を引き起こす駆動力として十分に機能したものと考える。

混合実験から得られた結果から、今回用いたパラメータではノズル出口より 100 [mm] 程度下流域において流体構造に多くの渦が含まれるようになり混合が促進されている傾向が得られている。Kelvin-Helmholtz の不安定性解析から得られる擾乱の成長過程では流体間に速度差 (駆動力) が存在する限り 2 流体は決して混合することは無い。しかし一旦その駆動力が消滅すると流体構造を保持することが困難になり流体混合が生じる。下流域まで対象とした数値計算結果から下流方向 90 [mm] 程度において流入時に無限小幅で接合していた流体間になだらかな速度勾配が生成され、渦構造の成長を促進する駆動力が減少する傾向を得た。密度プロファイル、比熱比を用いた識別関数からもノズル出口から 90 [mm] 程度からは渦構造の破壊が観測されその事実の裏付けとして捉えることができる。

また、分配関数を比較した場合 2 流体が混合する領域の広がり角は標準状態 (Pattern 1) の結果に比べ速度差を与えた計算結果 (Pattern 2) の方が小さくなっていることを図 3.53、図 3.54 から示した。この結果は、式 (3.205) の裏付けとなり、標準状態に近い流入条件を与えることが超音速流の混合に効果的であることを示唆している。

残念ながら数値計算結果からは図 3.50 で見られた様な鮮明な衝撃波列を得ることができなかったが、その駆動力となり得る Kelvin-Helmholtz 不安定性の周期的な構造を捕らえることに成功した。微小衝撃波列を観測できなかった原因として、2 流体境界面に発生する渦構造の強さが考えられる。渦の持つ流速が流体に対して相対速度差がマッハ 1 を超える場合微小衝撃波列が観測されると考える。このことから実験ではノズル出口において今回検討したパラメータ以上の密度、圧力ないしは流速に不連続が生じているものと考えられる。

### 3.5.8 まとめ

具体的なアプリケーションを IDO 法による数値流体計算で扱うことを目的として、超音速流を用いた混合問題に適用した。

数値計算結果から、ノズルエッジから発生する衝撃波や周期的な渦列などの実験で観測された混合流の構造をとらえることができた。主流-副流間速度差に関わらずノズル出口において理想的な流速分布が得られる場合、流体混合層はほとんど成長しないことを明らかにした。実験ではノズル接合部の後流が Kelvin-Helmholtz 不安定性により流体境界面に成長している可能性を示した。その結果周期的な渦列が流体境界面に発生することにより、実験で観測された等間隔に発生する微小衝撃波列が誘引されている可能性を示した。

また、Brown-Roshko 渦の理論から示されているように流体間速度差が大きい場合は混合領域の拡大角が小さくなる結果を得た。流体混合を行うためには、流体間速度差を大きくすることにより Kelvin-Helmholtz 不安定性の成長を促進させることがより効果的であるが、ノズル接合部から擾乱が発生することにより流体間速度差が無い場合でも Kelvin-Helmholtz 不安定性が成長することから、実験に用いたような条件下では 2 流体間速度差は超音速混合を促進させるための重要な因子ではないことを明らかにした。

## 3.6 まとめ

1 次元衝撃波管問題をはじめ、複雑な流体構造が発展する例題を IDO 法で扱い、流体問題への適用性を検討した結果、単純な 1 次元移流方程式などを扱う場合とは異なり、計算グリッド解像度に対する計算精度の変化率、高解像度計算グリッドを用いた場合の高次微係数計算精度の影響などが重要になることを示した。しかしその一方で少ない格子点数で IDO 法は流体計算に対して高精度解を与えることが可能であり、この特徴は 3 次元流体計算を行う場合には、計算機記憶容量の低減、計算負荷の低減をはかることが可能にし、これまでに大規模計算機を要した 3 次元レーリー・テラー不安定性の成長問題に対して PC (Personal Computer) による数値流体計算を可能にした。線形成長時の成長率に関して解析解によく一致した結果が得られ、マッシュルーム構造の成長を少ない格子点数で可能にした。この結果は有限差分法ではアプローチ困難な複雑な流体構造を示すアプリケーションに対してエルミート補間を用いた計算スキームが十分有効な結果を示す可能性があることを示唆している。流体問題に対するアプリケーションとして超音速混合流問題を扱い、実験結果から得られるカラーシュリーレン像と比較することで流体構造に関する議論を行った。数値計算では実験で設計されているパラメータを理想的に与えることが可能であるが、得られた結果からは標準状態における混合層の成長を捉えることは困難であった。実験ではノズル出口において流体間に有限厚さの境界が存在することから数値実験における流入条件として流体間に速度擾乱を付加した数値計算を行った所、擾乱から成長する渦構造を確認できた。この結果から実験における流体混合の主な要因としてノズル先端部の有限厚さに起因する流体擾乱が上流部に生成される衝撃波を誘起し、流体構造を乱すとともに下流域に向かい擾乱が成長することで混合領域における渦構造を発達させることが考えられる。そこで、ノズル出口における流体擾乱を付加した数値計算を考慮した結果、大きな渦構造の発達を標準状態に対しても得ることができた。微小衝撃波列を数値計算から得ることができなかったが、その生成要因として流体境界領域に発生する Kelvin-Helmholtz 不安定性に起因する周期的渦列が一つとして挙げられることを示した。

Brown-Roshko 渦の成長の議論に従い流体間速度差が大きくなるに伴って混合領域の増大角が減少し、混合が抑制される結果を得た。しかし、Kelvin-Helmholtz 不安定性の議論から流体間速度差が大きくなるに従い不安定性の成長、つまり混合が促進される考察がされるが、数値実験からは、若干の流体擾乱を持つ流入条件下では大域的な流体間速度差が無い場合に対しても Kelvin-Helmholtz 不安定性の成長が確認された。このことから、実験では流体間速度差を少なくすることでより混合が促進される可能性があることを示した。

## 第 4 章

# 非圧縮性流体解法の高速度化

数値流体計算により非圧縮性流体問題を扱う場合、MAC 法、SMAC 法、C-CUP 法に代表される、圧力ベースの解法を用いる。これらの計算手続きには各計算時間毎に圧力場決定のためのポアソン方程式を解く必要が生じる。ポアソン方程式の解法は移流計算に比べて計算時間の面から計算コストが大きく、計算格子点数が増加するにしたがって計算時間増大の要因となる。IDO 法を用いることで移流計算に対し少ない格子点数で精度の高い計算結果を得ることができることから、ポアソン方程式に対する空間精度を検証する。しかし、差分法によるポアソン方程式解法に比べ従属変数の増大、計算アルゴリズムの複雑化が原因となり計算時間が増大する傾向が予想されることから、本章では計算時間の短縮を図るため、緩和計算を高速化する手法として差分法で用いられて来たマルチグリッド法、並列計算ならびにベクトル計算を行う。計算領域における情報依存性を排除するレッドブラック法の適用性について検証する。また、非圧縮性流体問題を扱う場合各自間ステップ毎にポアソン方程式を解く必要性があることに着目し、計算時間刻幅  $\Delta t$  を大きくとることによって目標とする計算時間までに必要となる計算ステップ数を減少させ、ポアソン方程式を解く回数を減らすことで非圧縮性流体問題の高速解法構築を目指す。

### 4.1 ポアソン方程式解法の高速度化

#### 4.1.1 ポアソン方程式の空間精度

##### 2 次元ポアソン方程式の解法 (ディリクレ境界条件)

IDO 法は楕円型偏微分方程式の一つであるポアソン方程式を高精度で解くことが可能なスキームである。1 次元ポアソン方程式に対する精度検証から、IDO 法により求められた数値解は  $\Delta x^4$  の空間精度を持つことが既に示されている [3]。また、同グリッド数で 2 次精度中心差分法により求められた数値解を持つ誤差と IDO 法を用いて得られた数値解の誤差を比較してもより高い精度で求められることが示されている。ここでは 2 次元ポアソン方程式

$$f_{xx} + f_{yy} = \rho \quad (4.1)$$

を IDO 法を用いて解く手法を提示し、そのアルゴリズムを用いて求められた数値計算結果の精度の検証を行う。

IDO 法で 2 次元ポアソン方程式を解くためには 4 従属変数 ( $f, f_x, f_y, f_{xy}$ ) を利用していることから、方程式を閉じるために以下の 4 式を従属的に解かなければならない。

$$f_{xx} + f_{yy} = \rho \quad (4.2)$$

$$f_{xxx} + f_{xyy} = \rho_x \quad (4.3)$$

$$f_{xxy} + f_{yyx} = \rho_y \quad (4.4)$$

$$f_{xxx} + f_{xyy} = \rho_{xy} \quad (4.5)$$

多次元 IDO 法では、 $f_{xy}, f_{yy}$  等は  $f_y, f_x$  を  $x$  方向、 $y$  方向にそれぞれ 1 次元補間することで求めることが提案されている [32]。偏微分方程式の性質から、全ての空間微係数は 5 次中心補間を微分することで計算する。以上から、IDO 法で 2 次元 ポアソン方程式を扱う離散式は次のようになる。

$$\begin{aligned} & \frac{2}{\Delta x^2} (f_{i+1,j} - 2f_{i,j} + f_{i-1,j}) - \frac{1}{2\Delta x} (f_{x,i+1,j} - f_{x,i-1,j}) \\ & + \frac{2}{\Delta y^2} (f_{i,j+1} - 2f_{i,j} + f_{i,j-1}) - \frac{1}{2\Delta y} (f_{y,i,j+1} - f_{y,i,j-1}) = \rho_{i,j} \end{aligned} \quad (4.6)$$

$$\begin{aligned} & \frac{15}{2\Delta x^3} (f_{i+1,j} - f_{i-1,j}) - \frac{3}{2\Delta x^2} (f_{x,i+1,j} + 8f_{x,i,j} + f_{x,i-1,j}) \\ & + \frac{2}{\Delta y^2} (f_{x,i,j+1} - 2f_{x,i,j} + f_{x,i,j-1}) - \frac{1}{2\Delta y} (f_{xy,i,j+1} - f_{xy,i,j-1}) = \rho_{x,i,j} \end{aligned} \quad (4.7)$$

$$\begin{aligned} & \frac{2}{\Delta x^2} (f_{y,i+1,j} - 2f_{y,i,j} + f_{y,i-1,j}) - \frac{1}{2\Delta x} (f_{xy,i+1,j} - f_{xy,i-1,j}) \\ & + \frac{15}{2\Delta y^3} (f_{i,j+1} - f_{i,j-1}) - \frac{3}{2\Delta y^2} (f_{y,i,j+1} + 8f_{y,i,j} + f_{y,i,j-1}) = \rho_{y,i,j} \end{aligned} \quad (4.8)$$

$$\begin{aligned} & \frac{15}{2\Delta x^3} (f_{y,i+1,j} - f_{y,i-1,j}) - \frac{3}{2\Delta x^2} (f_{xy,i+1,j} + 8f_{xy,i,j} + f_{xy,i-1,j}) \\ & + \frac{15}{2\Delta y^3} (f_{x,i,j+1} - f_{x,i,j-1}) - \frac{3}{2\Delta y^2} (f_{xy,i,j+1} + 8f_{xy,i,j} + f_{xy,i,j-1}) = \rho_{xy,i,j} \end{aligned} \quad (4.9)$$

式 (4.6) ~ 式 (4.9) は全グリッドにおける  $f$  に関する全従属変数を含むベクトル  $F$ 、全グリッドにおける定数項  $\rho$  に関する全従属変数を含むベクトル  $R$  を用いて

$$AF = R \quad (4.10)$$

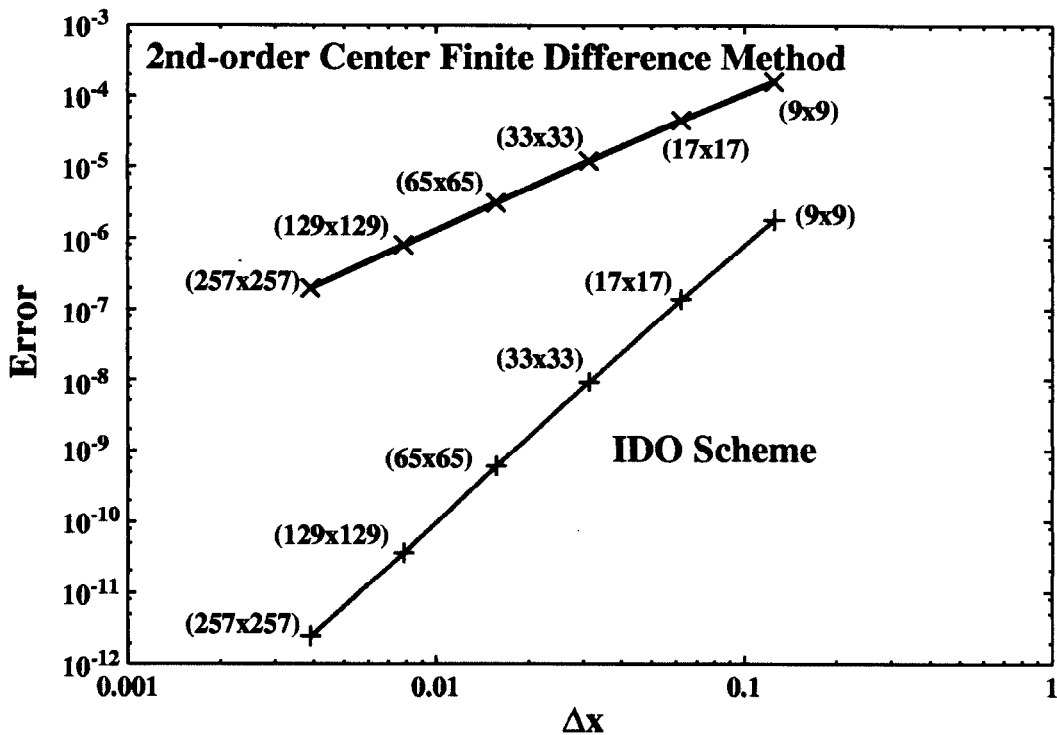


図 4.1: IDO 法による 2 次元ポアソン方程式の計算誤差  
 $\Delta x$  を変化させると 2 次精度中心差分法では  $\Delta x^2$  の精度で誤差が減衰し、IDO 法では  $\Delta x^4$  で誤差が減衰することを数値的に明らかにした。

と表すことが出来る。ここで、 $A$  は係数行列である。直接法を用いて  $A$  の逆行列を求め、解くことも可能であるが、係数行列  $A$  は計算グリッド点数を  $N \times M$  とすると、IDO 法では  $4N \times 4M$  行列となりその逆行列を解くこと自体困難になる。計算時間、計算機の記憶領域の制限から、一般的にポアソン方程式の解法に用いられている緩和法の一つである逐次の過剰緩和法 (Successive Over Relaxation; SOR) 法を用いて収束計算を行った。その計算式は次のようになる。

$$f_{i,j}^* = \left( \frac{2}{\Delta x^2} (f_{i+1,j}^k + f_{i-1,j}^k) - \frac{1}{2\Delta x} (f_{x,i+1,j}^k - f_{x,i-1,j}^k) + \frac{2}{\Delta y^2} (f_{i,j+1}^k + f_{i,j-1}^k) - \frac{1}{2\Delta y} (f_{y,i,j+1}^k - f_{y,i,j-1}^k) - \rho_{i,j} \right) / \left( \frac{4}{\Delta x^2} + \frac{4}{\Delta y^2} \right) \quad (4.11)$$

$$f_{x,i,j}^* = \left( \frac{15}{2\Delta x^3} (f_{i+1,j}^k - f_{i-1,j}^k) - \frac{3}{2\Delta x^2} (f_{x,i+1,j}^k + f_{x,i-1,j}^k) + \frac{2}{\Delta y^2} (f_{x,i,j+1}^k + f_{x,i,j-1}^k) - \frac{1}{2\Delta y} (f_{xy,i,j+1}^k - f_{xy,i,j-1}^k) - \rho_{x,i,j} \right) / \left( \frac{12}{\Delta x^2} + \frac{4}{\Delta y^2} \right) \quad (4.12)$$

$$f_{y,i,j}^* = \left( \frac{2}{\Delta x^2} (f_{y,i+1,j}^k + f_{y,i-1,j}^k) - \frac{1}{2\Delta x} (f_{xy,i+1,j}^k - f_{xy,i-1,j}^k) + \frac{15}{2\Delta y^3} (f_{i,j+1}^k - f_{i,j-1}^k) - \frac{3}{2\Delta y^2} (f_{y,i,j+1}^k + f_{y,i,j-1}^k) - \rho_{y,i,j} \right) / \left( \frac{4}{\Delta x^2} + \frac{12}{\Delta y^2} \right) \quad (4.13)$$

$$f_{xy,i,j}^* = \left( \frac{15}{2\Delta x^3} (f_{y,i+1,j}^k - f_{y,i-1,j}^k) - \frac{3}{2\Delta x^2} (f_{xy,i+1,j}^k + f_{xy,i-1,j}^k) + \frac{15}{2\Delta y^3} (f_{x,i,j+1}^k - f_{x,i,j-1}^k) - \frac{3}{2\Delta y^2} (f_{xy,i,j+1}^k + f_{xy,i,j-1}^k) - \rho_{xy,i,j} \right) / \left( \frac{12}{\Delta x^2} + \frac{12}{\Delta y^2} \right) \quad (4.14)$$

$$f_{i,j}^{k+1} = \omega f_{i,j}^* + (1 - \omega) f_{i,j}^k \quad (4.15)$$

$$f_{x,i,j}^{k+1} = \omega f_{x,i,j}^* + (1 - \omega) f_{x,i,j}^k \quad (4.16)$$

$$f_{y,i,j}^{k+1} = \omega f_{y,i,j}^* + (1 - \omega) f_{y,i,j}^k \quad (4.17)$$

$$f_{xy,i,j}^{k+1} = \omega f_{xy,i,j}^* + (1 - \omega) f_{xy,i,j}^k \quad (4.18)$$

SOR 法の収束判定は  $f$  の収束誤差による判定、 $f, f_x, f_y, f_{xy}$  の収束誤差による判定等、必要な精度に応じて決定される。非圧縮流体の計算では圧力の勾配を高精度で求めることが重要であるため、後者で判定した方が良く考える。

数値計算結果の精度評価は

$$\sum_i^N \sum_j^M \left| \frac{f(i,j) - f_{\text{analytic}}(i,j)}{f_{\text{analytic}}(i,j)} \right| / (N \times M) \quad (4.19)$$

で表される数値拡散誤差で評価を行った。ここで、 $f_{\text{analytic}}$  は解析解を表す。計算空間  $0 \leq x \leq 1, 0 \leq y \leq 1$  においてポアソン方程式の定数項を  $\rho = \sin(k_x x) \cos(k_y y)$ ,  $k_x = 5, k_y = 5$  と与え境界値を解析値で与えるディリクレ (Dirichlet) 型の境界条件下で、IDO 法、2 次精度中心差分法でそれぞれ収束計算をさせ、解析解からの数値拡散誤差を上記評価法で求めた結果をグリッド間隔  $\Delta x (\Delta y)$  に対してプロットしたものを図 4.1 に示す。図 4.1 から、本計算で用いた定数項に対して  $10^{-7}$  程度の誤差で 2 次元ポアソン方程式を解きたい場合、2 次精度中心差分法では IDO 法の 10 倍程度のメッシュが必要になること分かる。また、1 次元ポアソン方程式を IDO 法により扱った場合、その計算誤差は空間解像度に対して  $\Delta x^4$  の精度を示すことが知られており、2 次元ポアソン方程式に対しても IDO 法による数値計算結果から得られた数値誤差は  $\Delta x^4$  の精度を示すことを明らかにした。グリッド幅  $\Delta x (\Delta y)$  を変化させることでグリッドに対するプロファイルの波長が変化することを考えると、検証を行った範囲内では 2 次精度中心差分法に比べ IDO 法は数値拡散誤差に関して優れていることが数値的に明らかである。プロファイルはフーリエ級数によりグ

リッド数に応じた量子化が可能な範囲で周波数成分に分解可能であることから、上記計算結果により IDO 法による数値計算結果は 2 次精度中心差分法の数値計算結果に比べ高精度解を示すことを明らかにした。

## 2 次元ポアソン方程式の解法 (ノイマン境界条件)

非圧縮性数値流体計算ではしばしば固定境界面上の圧力を決定するためにノイマン条件を適用する。ノイマン境界条件下では境界条件として境界に対する垂直方向の微係数のみを拘束条件として与えるため、境界上における計算手法が精度に大きく影響することが考えられる。そこで、境界上の補間関数に関して説明を行う。境界面以外の点では、ディリクレ境界条件を用いた場合同様に 5 次精度中心補間を用いる。

境界に対して垂直方向の微係数を計算するために 3 次のエルミート補間を利用する手法がまず考えられる。図 4.2 に示すように、計算対象グリッドを中心として、境界面に沿って 5 次中心エルミート補間を形成し、境界に対して垂直方向に 3 次のエルミート補間を形成することで要求される空間微係数はおおよそ計算できる。境界面に対して垂直方向に形成した補間関数の形式ならびにその離散式、空間微係数計算手法は完全に CIP 法の補間式に一致するのでここでは式の導入を省略する。

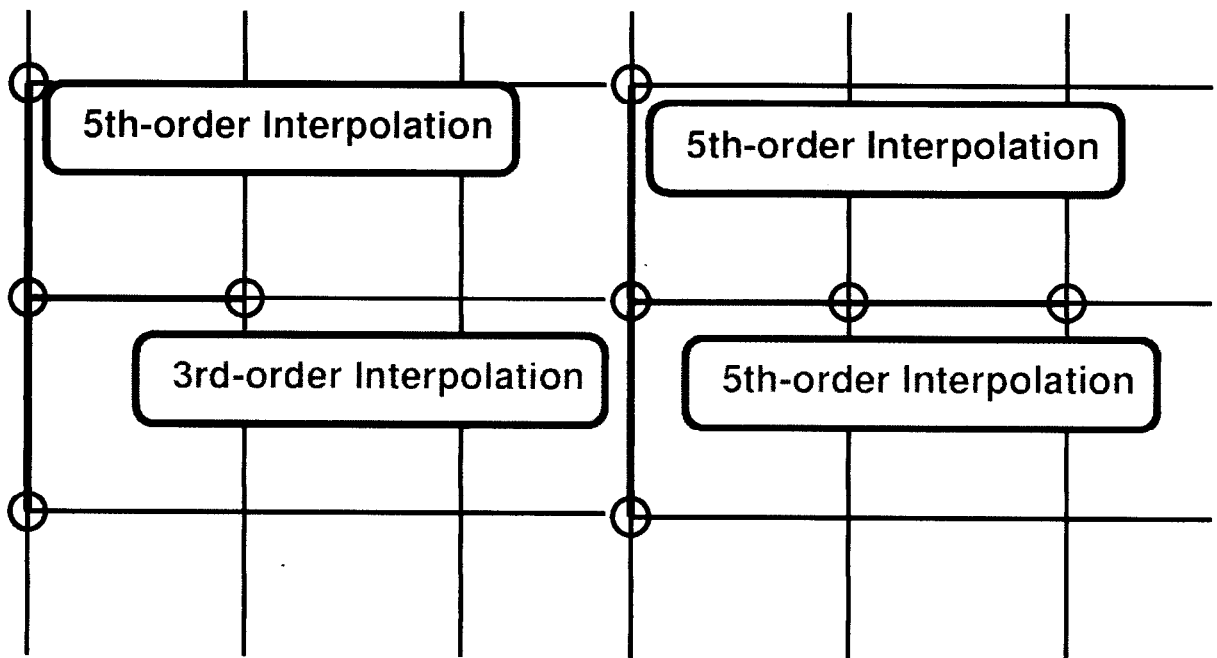


図 4.2: ノイマン境界条件下での補間関数 (3 次エルミート補間)  
境界に対して垂直方向に 3 次のエルミート補間を形成し、微係数を計算する

図 4.3: ノイマン境界条件下での補間関数 (5 次エルミート補間)  
境界に対して垂直方向に 5 次のエルミート補間を形成し、微係数を計算する

更なる高精度計算を達成するためには、境界面に対して垂直方向に 5 次のエルミート補間を形成する方法が考えられる。補間式は、

$$F(x) = ax^5 + bx^4 + cx^3 + dx^2 + f_{x,i}x + f_i \quad (4.20)$$

で表され、未知係数  $a, b, c, d$  はそれぞれ適合条件

$$F(\Delta x) = f_{i+1}, F_x(\Delta x) = f_{x,i+1} \quad (4.21)$$



$$F(2\Delta x) = f_{i+2}, F_x(2\Delta x) = f_{x,i+2} \quad (4.22)$$

or

$$F(-\Delta x) = f_{i-1}, F_x(-\Delta x) = f_{x,i-1} \quad (4.23)$$

$$F(-2\Delta x) = f_{i-2}, F_x(-2\Delta x) = f_{x,i-2} \quad (4.24)$$

より、

$$a = \frac{3f_i - 3f_{i+2}}{4\Delta x^5} + \frac{f_{x,i} + 4f_{x,i+1} + f_{x,i+2}}{4\Delta x^4} \quad (4.25)$$

$$b = \frac{-17f_i + 4f_{i+1} + 13f_{i+2}}{4\Delta x^4} + \frac{-6f_{x,i} - 20f_{x,i+1} - 4f_{x,i+2}}{4\Delta x^3} \quad (4.26)$$

$$c = \frac{33f_i - 16f_{i+1} - 17f_{i+2}}{4\Delta x^3} + \frac{13f_{x,i} + 32f_{x,i+1} + 5f_{x,i+2}}{4\Delta x^2} \quad (4.27)$$

$$d = \frac{-23f_i + 16f_{i+1} + 7f_{i+2}}{\Delta x^2} + \frac{-12f_{x,i} - 16f_{x,i+1} - 2f_{x,i+2}}{4\Delta x} \quad (4.28)$$

or

$$a = -\frac{3f_i - 3f_{i-2}}{4\Delta x^5} + \frac{f_{x,i} + 4f_{x,i-1} + f_{x,i-2}}{4\Delta x^4} \quad (4.29)$$

$$b = \frac{-17f_i + 4f_{i-1} + 13f_{i-2}}{4\Delta x^4} - \frac{6f_{x,i} - 20f_{x,i-1} - 4f_{x,i-2}}{4\Delta x^3} \quad (4.30)$$

$$c = -\frac{33f_i - 16f_{i-1} - 17f_{i-2}}{4\Delta x^3} + \frac{13f_{x,i} + 32f_{x,i-1} + 5f_{x,i-2}}{4\Delta x^2} \quad (4.31)$$

$$d = \frac{-23f_i - 16f_{i-1} + 7f_{i-2}}{\Delta x^2} - \frac{12f_{x,i} - 16f_{x,i-1} - 2f_{x,i-2}}{4\Delta x} \quad (4.32)$$

と求められる。

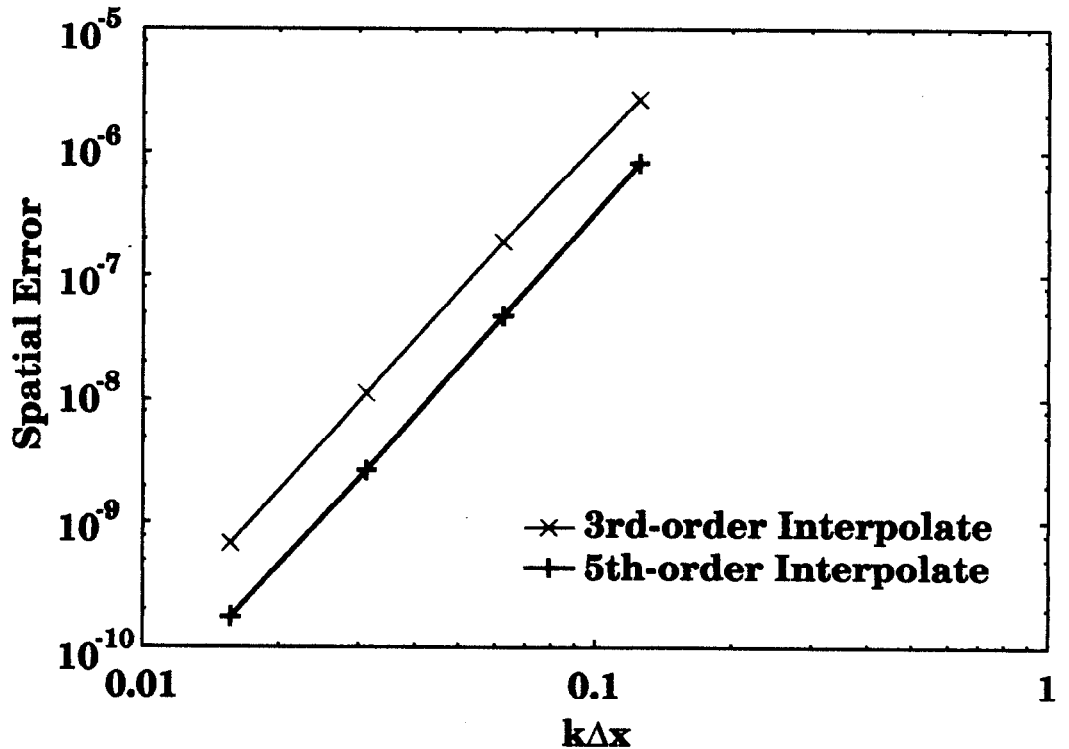


図 4.4: IDO 法によるノイマン境界条件下に於ける 2 次元ポアソン方程式の数値誤差  
2 次元ポアソン方程式を IDO 法によりノイマン境界条件下で扱った数値計算結果は空間解像度に対して  $\Delta x^4$  の精度を示す。

全境界においてノイマン条件を適用した 2 次元ポアソン方程式の数値誤差を検証するため、計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1$  中で、定数項  $\rho = \cos(k_x x) \cos(k_y y)$ ,  $k_x = \pi, k_y = \pi$  を与え、グリッドきざみ幅  $\Delta x(\Delta y)$  に対する数値計算誤差を見積もる。境界では物理量の垂直方向の微係数のみを解析解を用いて与えた。また、境界に対して垂直方向に形成する補間関数として 3 次エルミート補間を用いた。また、全境界でノイマン条件を適用しているため、SOR 法の収束が困難になることが予想されることから、 $x = -1, y = -1$  において物理量に対する解析解を与えた。

グリッド数を変化させることで空間解像度に対するポアソン方程式の解析解からの誤差を見積もったものを図 4.4 に示す。ディリクレ条件下でと同様に境界条件としてノイマン条件を適用した場合でも空間解像度に対して  $\Delta x^4$  の精度を示した。ただし解析解からの誤差の絶対値は本検証ではディリクレ条件で得られたものに比べ 10 倍程度大きい値が得られた。また、境界における補間関数の次数を 5 次とすることで空間解像度に対する数値誤差を低減することが可能になり、ディリクレ条件から得られた数値誤差とほぼ同程度でポアソン方程式を扱うことが可能になることを数値的に示した。

### 3 次元ポアソン方程式の解法

より一般的な流体問題への適用を考慮し、3 次元ポアソン方程式の精度検証を行う。3 次元問題を IDO 法で扱うためには物理量  $f$  に加えて、その空間微係数

$$f_x, f_y, f_z, f_{xy}, f_{xz}, f_{yz}, f_{xyz} \quad (4.33)$$

を用いる必要があり、その結果 3 次元ポアソン方程式

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} = \rho, \quad (4.34)$$

を解くためには式 (4.34) を空間微分した

$$\frac{\partial^3 f}{\partial x^3} + \frac{\partial^3 f}{\partial x \partial y^2} + \frac{\partial^3 f}{\partial x \partial z^2} = \frac{\partial \rho}{\partial x}, \quad (4.35)$$

$$\frac{\partial^3 f}{\partial x^2 \partial y} + \frac{\partial^3 f}{\partial y^3} + \frac{\partial^3 f}{\partial y \partial z^2} = \frac{\partial \rho}{\partial y}, \quad (4.36)$$

$$\frac{\partial^3 f}{\partial x^2 \partial z} + \frac{\partial^3 f}{\partial y^2 \partial z} + \frac{\partial^3 f}{\partial z^3} = \frac{\partial \rho}{\partial z}, \quad (4.37)$$

$$\frac{\partial^4 f}{\partial x^3 \partial y} + \frac{\partial^4 f}{\partial x \partial y^3} + \frac{\partial^4 f}{\partial x \partial y \partial z^2} = \frac{\partial^2 \rho}{\partial x \partial y}, \quad (4.38)$$

$$\frac{\partial^4 f}{\partial x^3 \partial z} + \frac{\partial^4 f}{\partial x \partial y^2 \partial z} + \frac{\partial^4 f}{\partial x \partial z^3} = \frac{\partial^2 \rho}{\partial x \partial z}, \quad (4.39)$$

$$\frac{\partial^4 f}{\partial x^2 \partial y \partial z} + \frac{\partial^4 f}{\partial y^3 \partial z} + \frac{\partial^4 f}{\partial y \partial z^3} = \frac{\partial^2 \rho}{\partial y \partial z}, \quad (4.40)$$

$$\frac{\partial^5 f}{\partial x^3 \partial y \partial z} + \frac{\partial^5 f}{\partial x \partial y^3 \partial z} + \frac{\partial^5 f}{\partial x \partial y \partial z^3} = \frac{\partial^3 \rho}{\partial x \partial y \partial z}. \quad (4.41)$$

を連立して解かなければならない。3 次元 IDO 法の補間式に従い、式 (4.34) を離散化することで

$$\begin{aligned} & \frac{2}{\Delta x^2} (f_{i+1,j,k} - 2f_{i,j,k} + f_{i-1,j,k}) - \frac{1}{2\Delta x} (f_{x,i+1,j,k} - f_{x,i-1,j,k}) \\ & + \frac{2}{\Delta y^2} (f_{i,j+1,k} - 2f_{i,j,k} + f_{i,j-1,k}) - \frac{1}{2\Delta y} (f_{y,i,j+1,k} - f_{y,i,j-1,k}) \\ & + \frac{2}{\Delta z^2} (f_{i,j,k+1} - 2f_{i,j,k} + f_{i,j,k-1}) - \frac{1}{2\Delta z} (f_{z,i,j,k+1} - f_{z,i,j,k-1}) = \rho_{i,j,k}. \end{aligned} \quad (4.42)$$

が得られる。式 (4.35) - 式 (4.41) に関しては 2 次元ポアソン方程式での離散化と同様の手順を用いることで離散化することが可能であることからここでは記述を省略する。

物理量ならびにその空間微係数で与える境界条件を解析解を用いて与えるディリクレ型境界条件に対する IDO 法による 3 次元ポアソン方程式の数値拡散誤差を計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, -1 \leq z \leq 1$  に対してポアソン方程式の定数項  $\rho$  を

$$\rho(x, y, z) = \cos(k_x x) \cos(k_y y) \cos(k_z z) \tag{4.43}$$

と与え、空間解像度に対して見積もる。ここで、 $k_x = \pi, k_y = \pi, k_z = \pi$  と与えた。数値計算誤差は、

$$\sum_i \sum_j \sum_k \left| \frac{f(i, j, k) - f_{\text{analytic}}(i, j, k)}{f_{\text{analytic}}(i, j, k)} \right| / \text{MeshNumber} \tag{4.44}$$

を用いて評価する。ここで、 $f_{\text{analytic}}$  は解析解を示す。以上により求められた数値誤差を図 4.5 に示す。

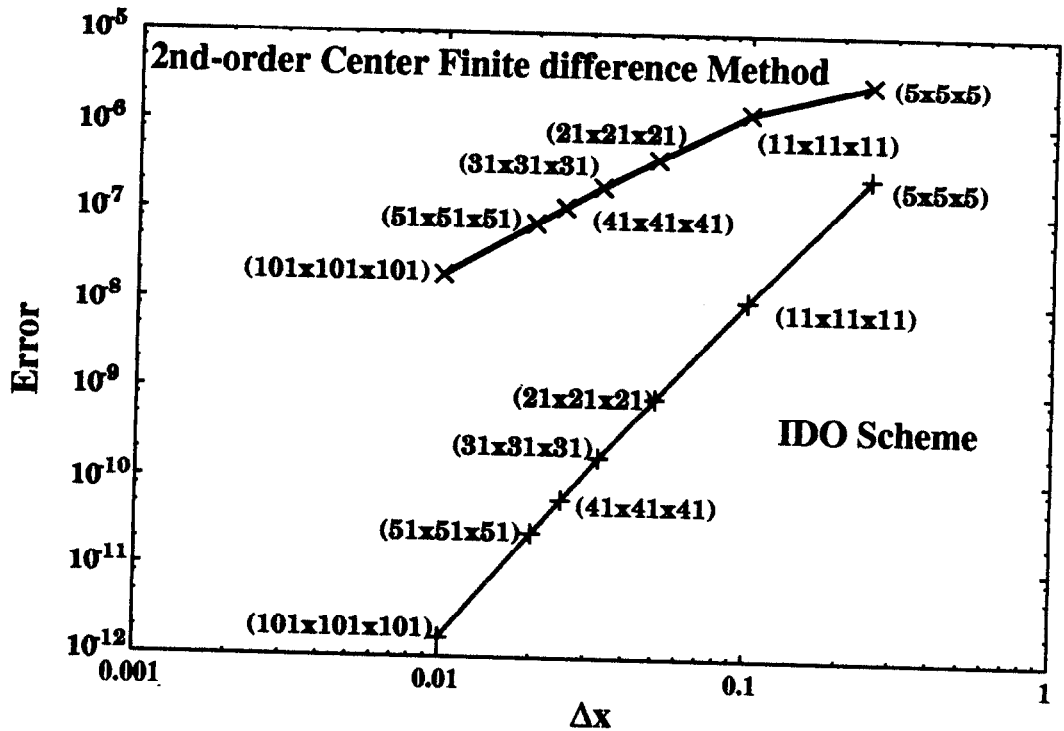


図 4.5: IDO 法による 3 次元ポアソン方程式の数値計算誤差  
3 次元ポアソン方程式に対しても IDO 法は  $\Delta x^4$  精度を示すことを数値実験的に明らかにした。

数値計算結果から、IDO 法は 3 次元ポアソン方程式に対しても  $\Delta x^4$  の精度を示すことを明らかにした。計算結果から得られた数値誤差を 2 次精度中心差分法と比較すると、今回検証した範囲内では常に IDO 法による数値計算結果から得られた誤差が優位を示した。3 次元問題ではグリッド解像度に対する精度向上がもたらす効果は大きい。仮に IDO 法により  $10 \times 10 \times 10$  計算格子点数を用いた 3 次元ポアソン方程式の計算結果が  $100 \times 100 \times 100$  計算グリッド点数を用いた 2 次精度中心差分法による計算結果と等しい精度を示すと仮定すると、両者が要求する計算機記憶容量はそれぞれが必要とする変数を考慮した場合、

$$\text{for 2nd-FDM} = (100 \times 100 \times 100)[\text{Grids}] \times 2[\text{Variables}] \times 8[\text{Byte}] = 16[\text{MByte}] \tag{4.45}$$

$$\text{for IDO} = (10 \times 10 \times 10)[\text{Grids}] \times 8[\text{Variables}] \times 8[\text{Byte}] = 64[\text{kByte}] \tag{4.46}$$

と概算され、およそ 3 乗程度の差が生じることになる。CPU コストについては単純な比較が困難であるが、同一 CPU により、2 次元ポアソン方程式、3 次元ポアソン方程式を扱った際の計算に要した計算時間を見積もったものを図 4.6 に示す。

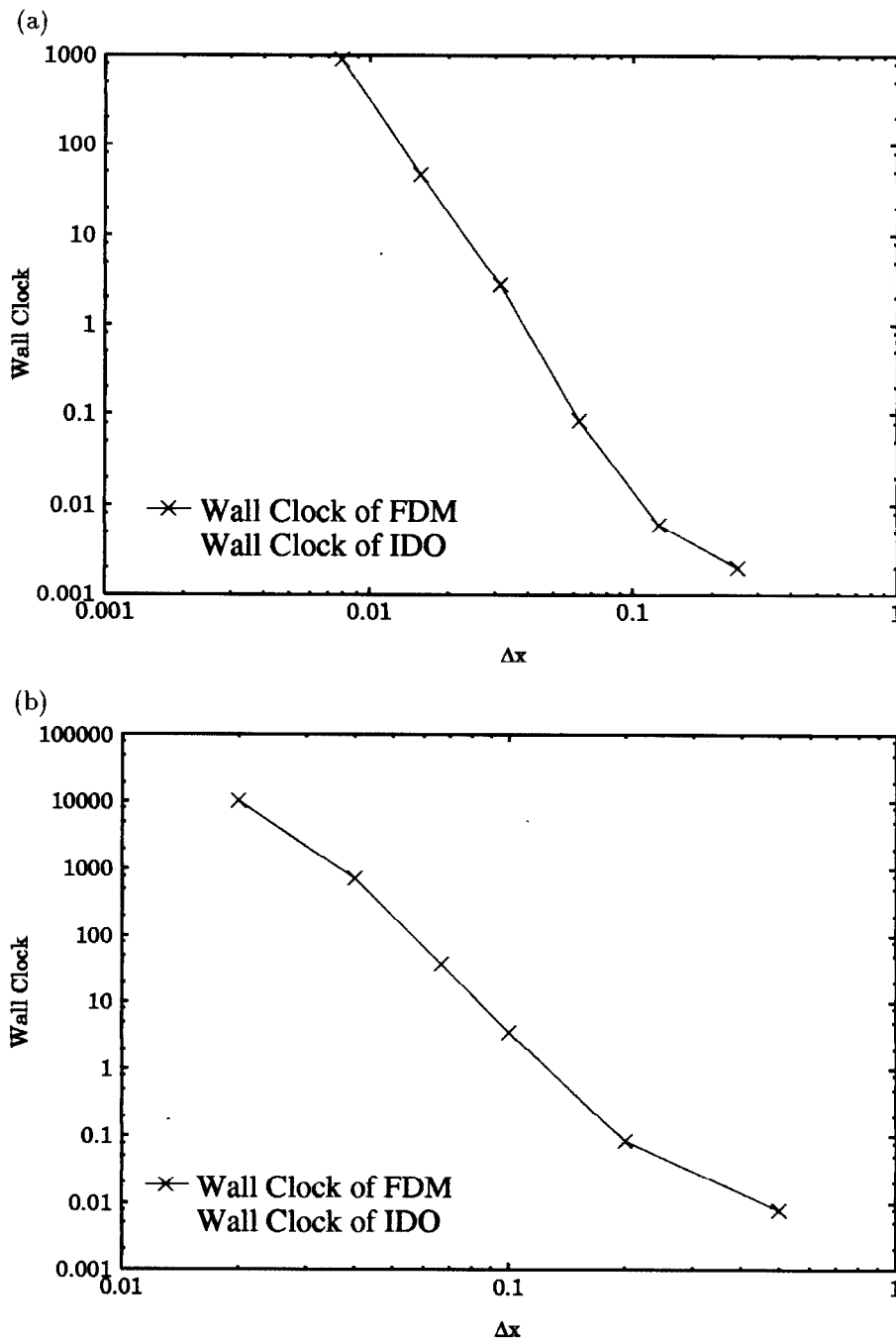


図 4.6: IDO 法によるポアソン方程式解法の計算時間:

(a) 2次元ポアソン方程式に対する計算時間、(b) 3次元ポアソン方程式に対する計算時間を表している。

計算に用いた CPU は、Athlon 800MHz で、ディスクスワップの生じない計算機記憶容量 384MByte を保持している。システムは FreeBSD を利用しており、コンパイラならびに最適化は gcc version 2.95.2、-O2 を用いた。演算時間ではなく、メモリ確保から解放を含む全ての時間を含んでいるため若干のばらつきが見られるが、2次元ポアソン方程式に関しては計算時間が 4 次のオーダーで増大しており、3次元ポアソン方程式に対しては 5 次のオーダーで増大することを明らかにした。また、IDO 法による計算時間は有限差分法で 2 倍程度の格子点数を用いた場合の計算時間とほぼ一致しており、先の有限要素法により IDO

法と同精度計算を行うために 10 倍程度の計算格子点数が必要であることを考慮すると、計算時間においても IDO 法はすぐれていることを明らかにした。このことから、IDO 法を用いることでこれまでスーパーコンピュータと呼ばれる大型計算機を用いなければならなかった計算問題をパーソナルコンピュータ (Personal Computer; PC) 程度の性能、記憶容量で充分計算可能になる可能性がある。実際に本問題は PC 上で検証された。

#### 4.1.2 マルチグリッド法の適用

非圧縮性流体問題や電磁気学の問題などを数値計算で扱うには、楕円型偏微分方程式の一つであるポアソン方程式を解く必要が生じる。楕円型偏微分方程式は係数行列を直接法により解くことも可能ではあるが、記憶領域の問題や計算時間の問題から行列が大きくなるほど計算が困難になるため、緩和法によって解いた方が効率が良く考えられる。これまでに様々な反復法が提案されており、代表的な手法として下記のようなものが挙げられる。

- Jacobi 法
- Gauss-Seidel 法
- 逐次的過剰緩和法 (SOR 法)
- 対称逐次的過剰緩和法 (SSOR 法)
- 共役勾配法 (CG 法)
- 最小残差法 (MINRES 法) と対称 LQ 法 (SYMLQ 法)
- 正規方程式に対する共役勾配法 (CGNE 法と CGNR 法)
- 一般化最小残差法 (GMRES 法)
- 双共役勾配法 (BiCG 法)
- 疑似最小残差法 (QMR 法)
- 2 乗共役勾配法 (CGS 法)
- 安定化双共役勾配法 (Bi-CGSTAB 法)
- チェビシェフ反復法

ポアソン方程式に対する精度検証から、IDO 法を用いることで有限差分法に比べ高精度で解を得られることを示したが、IDO 法を用いることで従属変数の増大、計算手順の複雑化が原因となり同グリッド数を用いた 2 次精度中心差分法による計算手順に比べて緩和計算における計算負荷が増大する。この問題を改善するため、緩和計算において真の解からの誤差を早く収束させるため、解像度の高いメッシュでの緩和中に格子点数の少ないメッシュに移り、真の解からの修正量を計算し、解像度の高いメッシュでの緩和に反映させることで緩和速度を劇的に改善することが可能な手法であるマルチグリッド法の導入を考える。

緩和法を用いた収束計算をさせると始めの数回の収束計算で急速に誤差が減少し緩和回数が増加するにしたがって誤差の減少速度が低下する現象が見られる。これは、緩和法が格子長さと同波長程度の誤差を効率良く緩和し、長波長の誤差に対しては効率が悪いという性質を表している。そこで、解像度の高いグリッドでは緩和しにくい長波長の誤差を粗いグリッドで緩和するのがマルチグリッド法 [33, 34] の狙いである。ただし、細かいグリッド程度の波長の誤差は粗いグリッド上で表現できないのでそのような高周波成分は粗いグリッドに移る前に長波長成分に対し、十分小さくしておかなければならない。以上をふまえ、マルチグリッド法概念を示す。

1. 緩和法により高波数成分を緩和する。
2. より粗い格子で誤差を打ち消すための修正量を計算し、グリッド長さに対応した波数成分の誤差を緩和する。
3. 粗い格子から元の解像度の高い格子に修正量を付加しながら戻していく。

マルチグリッド法の一つである CS 法 (Correction Storage Scheme) について Appendix D に詳細を付記した。マルチグリッド法をスカラー計算機において先に示したの緩和手法と組み合わせることにより緩和速度を高速化することが可能である。また、計算領域を保持した状態でグリッド解像度のみを変更するアルゴリズムの性質上並列化も容易に行える計算手法である。ベクトル計算機であればマルチカラー法などが高速化手法として良く利用されている。本節では IDO 法を用いたポアソン方程式の解法に SOR 法とマルチグリッド法を合わせた手法について示し、その有効性について検証する。

2 次精度中心差分法、IDO 法でそれぞれ CS 法を用い、2 次元ポアソン方程式 ( $f_{xx} + f_{yy} = \rho$ ) を解くことでマルチグリッド法の有効性を検証する。計算領域  $0 \leq x \leq 1, 0 \leq y \leq 1, \Delta x = 1/256, \Delta y = 1/256$  に対して定数項

$$\rho(x, y) = \sin(k_x x) \sin(k_y y); k_x = 5, k_y = 5 \quad (4.47)$$

を与え、解析解を用いた境界条件下で検証を行う。

CS 法における制限補間 ( $I_k^{k-1}$ ) は 2 次精度中心差分法、IDO 法共に細かいグリッド点の情報をそのまま与える方針を用いた。延長補間は中心差分法では線形補間 ( $f_i = (f_{i+1} + f_{i-1})/2$ ) から求め、IDO 法では 5 次中心補間関数を用いて求めた。緩和法として採用した SOR 法の緩和係数は  $\omega = 1.7$  とした。

2 次精度中心差分法を用いて 2 次元ポアソン方程式をマルチグリッド法における残差 (式 (D.3)) がグリッド当たり  $2 \times 10^{-12}$  となるまで収束計算を行った。マルチグリッド法で用いる格子の数  $k$  を段数とし、段数に対する収束までの時間を表 4.1 に示す。

表 4.1: マルチグリッド法を用いた中心差分法によるポアソン方程式の計算時間

1 段での計算は SOR 法を用いた通常の緩和計算を示す。マルチグリッド法を用いた 2 段、3 段、4 段の収束計算を行うと、計算時間がほぼログスケールで減少する結果が得られた。

段数	CPU 時間	実時間	CPU 使用率
1 段	313.975u	5:21.84	97.5%
2 段	70.712u	1:12.11	98.1%
3 段	18.376u	0:18.85	97.8%
4 段	8.304u	0:08.52	97.7%

段数が増加するに従いそれぞれのグリッドでの緩和回数に自由度が増し、計算時間短縮のための最適化を行うのは困難になる。そのため表 4.1 に示した計算時間が必ずしも最適であると保証出来ないが、大まかな目安、指標としては用いることが可能と考える。マルチグリッド法を SOR 法と組み合わせ用いた場合、段数を増加するに従って計算時間短縮に及ぼす影響が少なくなる傾向が通常見られるが、上記条件下で 2 次精度中心差分法を用いて緩和計算を行った限りではグリッド段数の増加に従って計算時間を大幅に短縮できていることが分かる。単純な SOR 法 (1 段) と 4 グリッドサイクルマルチグリッド法での計算時間には CPU 時間で 35 倍以上の加速を確認した。

次に IDO 法のマルチグリッド化について検証する。収束判定条件は中心差分法の場合と同じようにマルチグリッド法で用いる物理量の残差行列を構成する要素の最大値が  $4 \times 10^{-12}$  より小さくなることとし

た。IDO 法を用いた緩和計算に対して、マルチグリッド法による計算時間の最適化を行う場合、グリッド段数増加によって各グリッドサイクルに於ける緩和回数自由度増加のみならず、微係数の制限補間、延長補間の選択にも考慮の余地があるため 2 次精度中心差分法に比べ更に最適化が困難になる。例として 2 グリッドサイクル、3 グリッドサイクルの最適化について考察する。ただし、計算負荷のため、グリッド数を減らし、 $\Delta x = 1/64, \Delta y = 1/64$  とした。制限補間は粗い格子に細かい格子の値をそのまま与える手法をとり、延長補間は IDO 法の 5 次精度中心補間を用いて補間した。各微係数についても同様である。SOR 法の加速係数を 1.7 で固定し、各グリッドに於ける緩和回数をパラメータとして収束時間を計測した結果を示す。

表 4.2: IDO 法を用いたポアソン方程式のマルチグリッド法による解法の最適化:

IDO 法を用いてポアソン方程式をマルチグリッド法により解く場合、マルチグリッド法中で生成する修正方程式を各グリッドで緩和する条件により計算時間は大幅に変わる。各グリッドでの緩和回数と収束計算に要した時間から、最終段での収束条件により下記の例では収束時間に約 30% 程度の変動が見られた。

stage	stage	stage	CPU	wall	CPU
1 (1)	2 (1)	1 (2)	time	clock	(%)
10	50	10	6.558u	0:06.72	97.4%
10	100	10	5.325u	0:05.47	97.8%
10	150	10	5.874u	0:05.98	98.1%
10	200	10	6.966u	0:07.11	97.8%
10	100	5	4.808u	0:04.93	97.5%
5	100	10	4.804u	0:04.90	97.9%
10	100	1	6.073u	0:06.20	97.9%
10	100	6	4.663u	0:04.77	97.6%

表 4.3: 緩和係数による最適化:

2 グリッドサイクルマルチグリッド法の緩和係数に対する緩和計算時間の減少。

$\omega$	CPU	wall	CPU
	time	clock	(%)
0.5	72.018u	1:13.64	97.8%
0.8	35.900u	0:37.37	96.1%
1.0	24.086u	0:24.73	97.4%
1.2	16.238u	0:16.70	97.3%
1.5	8.419u	0:08.71	96.5%
1.7	5.312u	0:05.42	97.9%
1.8	8.026u	0:08.21	97.6%

表 4.2 から、各グリッドでの緩和演算回数により収束までの計算時間に 28% 程度の差が確認できる。マルチグリッド法を用いずに SOR 法のみでの計算時間が約 25 (sec) であることからいずれの場合でも高速化は果たしているが、緩和演算回数を微妙に変化させることで計算時間が簡単に 3 割程度変動することから、マルチグリッド法の最適化は非常に困難であると言える。上記結果から、粗いグリッドで十分長波長誤差を緩和しなければならないが、粗いグリッド上で修正方程式が完全に収束するまで緩和計算を行ったとしても、細かいグリッド上で緩和できなかった高周波数誤差が修正方程式中に含まれるため、全計算時間に対して悪効果を及ぼす結果となった。逆に粗いグリッドでの緩和演算が少なれば長波長誤差を細かいグリッドで緩和することになるため、同様に計算時間に対して悪効果を及ぼした。また、2 グリッドで V サイクルを用いたマルチグリッド法を適用しているため、1 段目の緩和回数 (1) と、1 段目の緩和回数 (2) を入れ替えても全く等価な結果が得られた。

次に、SOR 法の緩和係数に対する計算時間について考察する。ここで用いる各グリッドの於ける緩和計算回数は、表 4.2 で最速であった、1 段目緩和回数 (1)、2 段目緩和回数、1 段目緩和回数 (2) をそれぞれ 10、100、6 とした。結果を表 4.3 に示す。緩和係数を増加するに従い計算時間が 10 倍以上短縮できることが示された。ただし、通常の SOR 法では緩和係数が 2 まで安定に収束するのにに対し、マルチグリッド法を用いた今回の計算では緩和係数が 1.8 を越えると数値不安定性により収束に向かわなくなる傾向が見られた。また、各グリッドに於ける緩和回数によっては緩和係数を増大させても単純に緩和速度が高速化しない例もあった。

表 4.4: 3 グリッドサイクルマルチグリッド法の最適化:

格段に於ける緩和計算回数によりマルチグリッド法の最適化が行えるが、3 グリッドサイクルになると自由度が高くなるため最適化は困難になってくる。

stage	stage	stage	stage	CPU	wall	CPU
1 (1)	2 (1)	3 (1)	2 (2)	time	clock	(%)
4	10	10	10	5.485u	0:05.67	96.8%
4	10	20	10	4.756u	0:04.89	97.1%
4	10	50	10	4.796u	0:04.99	95.9%
4	10	100	10	5.497u	0:05.65	97.1%
4	10	200	10	6.767u	0:06.95	97.4%
4	5	20	10	4.454u	0:04.64	96.1%
4	10	20	5	5.708u	0:05.91	96.7%
4	2	20	10	3.493u	0:03.65	96.4%
4	3	20	10	6.832u	0:07.05	96.8%
4	4	20	10	7.783u	0:08.09	96.4%
4	2	20	15	3.472u	0:03.59	96.9%
4	2	20	20	3.454u	0:03.58	96.6%
4	2	20	22	3.531u	0:03.68	96.4%
3	2	20	22	8.748u	0:09.06	96.6%
6	2	20	22	4.272u	0:04.39	97.2%
6	2	20	22	4.153u	0:04.31	96.5%



更に最適化の難しい3グリッドサイクルマルチグリッド法の最適化について検証する。先と同様に緩和係数を1.7とし、グリッドは計算負荷の関係から、 $\Delta x = 1/64, \Delta y = 1/64$ として検証を行った。各段における緩和回数に対する計算時間を表4.4に示す。

2グリッドサイクルの場合と同様に、最も格子の粗いグリッドでの緩和回数が少なすぎると長波長誤差の緩和を細かいグリッドで行うことになるので計算負荷が高くなるため計算時間が改善されず、粗い格子での緩和計算が多すぎても十分に長波長誤差が緩和されていけば不必要に計算時間が増大する結果が得られた。また、2段目の緩和回数1を2段目の緩和回数2を入れ換えた場合、収束までの計算時間が変化する結果が得られた。各グリッドに於ける緩和回数1は最終段での計算の前での緩和演算であるため、格子長さ程度の誤差を十分に緩和できていれば、より粗いグリッドに於ける修正方程式から高波数誤差を除くことが出来る。緩和回数1での緩和演算は計算対象となっているグリッド幅にほぼ適合している高波数誤差を緩和対象としているので数回の緩和計算で十分に減少させることが出来る。一方緩和回数2での緩和演算は、1段細かいグリッドに対してより精度の高い修正量を提供しなければならないため、十分に誤差を緩和しておかなければならない。このことから最適化を行った場合、緩和回数2は緩和回数1に比べ増大する傾向がみられた。Vサイクルを用いたマルチグリッド法を適用しているため、1段目の緩和回数は2グリッドサイクルの場合と同様に緩和回数1、緩和回数2の入れ換えを行っても収束までの計算時間にほとんど差が見られなかった。また、1段目での緩和回数もグリッドに適合した誤差を減少させるだけで良いので数回程度で十分である結果が得られた。

次に、緩和係数に対する収束時間について考察を行う。

表4.5: 3グリッドサイクルマルチグリッド法の緩和係数に対する最適化:

単純に緩和係数を増大させると計算は高速化されるが、数値不安定性により緩和係数が1未満、1.8以上では計算が発散した。

緩和係数	CPU 時間	実時間	CPU 使用率
1.0	291.108u	5:01.53	96.6%
1.2	16.908u	0:17.56	96.4%
1.5	4.488u	0:04.69	95.7%
1.7	3.466u	0:03.59	96.6%

3グリッドサイクルではSOR法の加速係数を1.7として各グリッドでの緩和演算回数を最適化した後、緩和演算回数を固定したままで加速係数を1未満に変更した場合、数値不安定性により収束しない結果が得られた。また、加速係数が1.2以下では計算時間が2グリッドサイクルよりも遅くなり、加速係数を1.8以上にすると再び数値不安定性により収束しない結果が得られた。加速係数に対する計算時間の傾向は、2グリッドサイクル同様、加速係数を大きくするに従い計算時間は短縮される傾向が見られた。今回のように2グリッドサイクルよりも計算時間が大きくなった要因として、グリッド毎の緩和回数に対する最適化を加速係数1.7に対して行ったことが考えられる。最適化を行った状態で加速係数を小さくすることにより各グリッドでの誤差の収束量が減少するため修正方程式を形成する前の緩和演算回数1が不足することが原因の一つと考える。また、緩和演算回数2に関しても十分に収束可能な演算回数にならないため精度の高い修正量を提供できないこともまた原因となる。以上から、計算で用いる加速係数毎にそれぞれ緩和回数の最適化が必要であると結論づける。

IDO法を用いたポアソン方程式の緩和演算手法に対しマルチグリッド法を適用した結果について考察する。表4.6に挙げたように今回用いた定数項に対して $257 \times 257$ グリッドでマルチグリッド法を適用した場合3グリッドサイクルまではマルチグリッド法を用いたことによる計算の高速化が十分に示されたが、3グリッドサイクルと4グリッドサイクルではほとんど計算時間に差がみられなかった。計算のオーバー

ヘッド、最適化等様々な原因が考えられるが、SOR 法とマルチグリッド法を併用した場合、グリッド数がある程度増やすと計算の高速度化に限界があることが示されていることから、IDO 法に対して同手法を用いた場合、2 次精度中心差分法を用いた場合よりもグリッド段数による高速化効果の減衰が早いのかも知れない。

本節で行ったマルチグリッド手法のうち、最も高速化できた 4 グリッドサイクルマルチグリッド法と SOR 法を比較した場合、計算時間に 35 倍程度の短縮が達成され、この結果からだけでもマルチグリッド法が非常に有効な手段であることが分かる。2 次精度中心差分法と IDO 法の、グリッド数に対する CPU 時間を図 4.7 に示す。

表 4.6: マルチグリッド法を用いた IDO 法によるポアソン方程式の計算時間

段数	CPU 時間	実時間	CPU 使用率
1 段	10340.717u	2:54:18.16	98.9%
2 段	1851.478u	31:02.21	99.4%
3 段	285.583u	4:46.20	99.8%
4 段	284.155u	4:44.77	99.8%

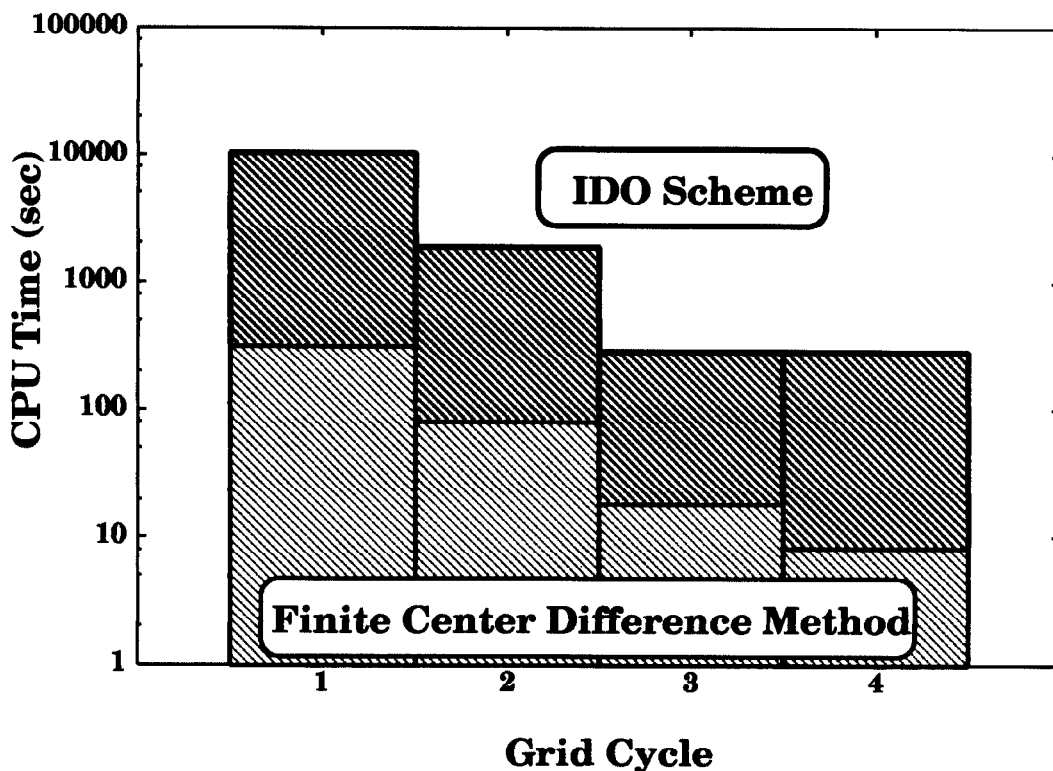


図 4.7: マルチグリッド法の段数が計算時間に及ぼす効果

グリッドサイクルを増加させると計算時間がログスケールで減少する傾向が見られた。ただし、緩和法に SOR 法を用いているため、ある程度グリッドサイクルを増加させると計算時間の改善に対する効果が薄れる結果が得られた。

グリッド数を増やすことで2次精度中心差分法、IDO法ともに計算時間がログスケールで減少していることが分かる。SOR法などの計算時間はグリッド点数の自乗に比例するが、マルチグリッド法を用いると単純にグリッド点数に比例することがわかっている。更に段数を増やすことで計算時間がログスケールで減少するのであれば、マルチグリッド法は緩和計算に非常に有効な手段と言える。ただし、グリッド数がある程度大きくすると計算時間に対する効果が減少することも報告されている。

更なる高速化を図るためにSOR法の加速係数を変化させたり制限補間、延長補間の手法を工夫してみた。通常、IDO法を用いてポアソン方程式をSOR法により緩和計算すると緩和係数を1.7~1.8程度で最も収束率が高くなるが、マルチグリッド法を適用した場合緩和係数の最適値は緩和手順に大きく依存し、1.7~1.8での収束が最適にならない場合が良く見られた。また、制限補間、延長補間に関してもIDO法では補間式を用いることで高精度な補間が可能であるが、単純に線形補間を用いて延長補間を行った方がIDO法の補間式を用いた手法に比べはるかに演算回数を減らせるため補間精度の低下にも関わらずより高速に計算できることが分かった。

以上から、IDO法とマルチグリッド法を組み合わせることで2次精度中心差分法でSOR法を用いた場合に要した時間以下で2次精度中心差分法に比べ $10^2$ ~倍計算精度の高い2次元ポアソン方程式の解法の開発に成功した。

#### 4.1.3 レッドブラックアルゴリズムの適用

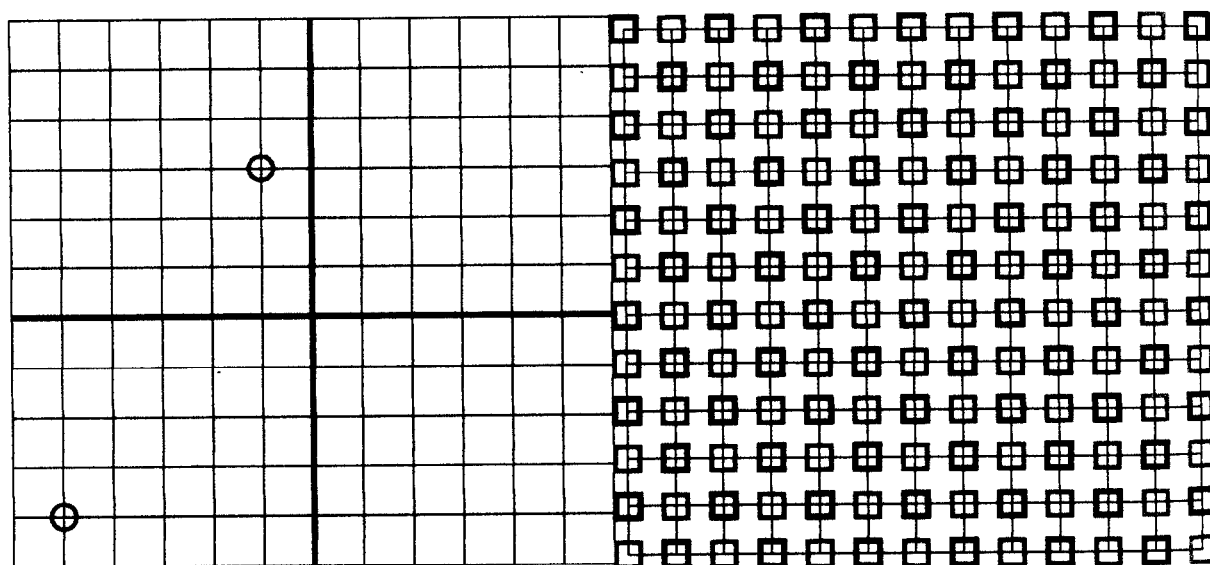


図 4.8: レッドブラックアルゴリズムの必要性  
自然な手続きによる緩和法では図中に示された一方の点の情報を更新するためには他方の情報の更新を待つ必要がある。そのため、太線のように領域分割を行い、並列計算することを考えた場合、各ノードにおける計算は他のノードの計算を待たなければならない。

図 4.9: レッドブラック法  
2次元格子を図のようにRedグリッドとBlackグリッドに分類し、それぞれの色毎に情報を更新することにより2次精度中心差分法を用いた2次元ポアソン方程式の解法では情報更新の空間依存性を排除することが出来る。

ポアソン方程式を含め、様々な数値計算に対する高速化のアプローチとして並列計算機、ベクトル計算機を用いた並列化やベクトル化が挙げられる。しかし、そのためには数値計算において、データ更新の空間依存性を完全に排除する必要がある。図 4.8 で示されたグリッド上でポアソン方程式を解くことを考え

る。SOR 法などの緩和法で収束計算することを考え、図 4.8 左下端から計算を開始する。緩和法を自然なアルゴリズムで利用した場合、下部に丸で示されたグリッド上の情報更新を元に上部の丸で示されたグリッドを含む下流側の情報更新を行うことから情報更新の依存関係が生じるため、それぞれのグリッドを同時に解くことが出来ない。

つまり、図 4.8 の太線で示されたように領域分割を行い並列計算を行った場合、それぞれの領域毎に計算が完了しなければ他の領域の計算が進行しないため並列計算を行ったとしても、複数の計算機を利用することによる記憶領域の拡張といったメリットはあるものの、計算速度に関するメリットが全く無く、収束計算は 1 CPU で計算する時間と同じ時間を要する。そこで、レッドブラック アルゴリズムの適用を考える。レッドブラック アルゴリズムは 2 次精度中心差分法のように緩和計算における情報依存が前後左右の計算グリッド点に限られた離散式に対し、反復計算法の更新依存性を排除し、計算空間のいずれの場所からの計算を可能にする。

レッドブラックアルゴリズムでは全計算格子点を、図 4.9 に示すように Red 格子と Black 格子に分類し、Red 格子の計算を先に行い、Black 格子の計算をその後に行う。Red 格子の計算のために Black 格子の時間更新が必要であるが、Red 格子同士の依存関係が無くなるため、Red 格子に関しては同時に計算が可能である。つまり、Red 格子の計算順序には拘束が無くなる。これにより、先のように領域分割をした場合でも Red 格子を計算し、その後に Black 格子を計算するというルールを守ることによって空間的計算順序から依存性を排除することが出来る。

有限差分法を用いた計算手順に対してこれまでに レッドブラックアルゴリズムの有効性が示されて来た。中心 5 次補間を用いた IDO 法によるポアソン方程式の解法でも計算対象グリッド点を中心としてその前後左右におけるグリッド点上の情報を用いた離散式により緩和計算が行えることから、レッドブラックアルゴリズムの適用は可能である。しかし、空間微係数を用いる IDO 法は新しいスキームであるため、これまでに レッドブラックアルゴリズムを適用した場合の収束性について議論されていないことから、その適用性を検証することは有用である。本節では、レッドブラックアルゴリズムを適用した IDO 法によるポアソン方程式の計算誤差を自然なアルゴリズムによる結果と比較することで検証を行う。

計算空間  $0 \leq x \leq 1, 0 \leq y \leq 1$  において 2 次元ポアソン方程式方程式

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \rho \quad (4.48)$$

の定数項 ( $\rho$ ) を

$$\rho(x, y) = \cos(k_x x) \cos(k_y y) \quad (4.49)$$

と与えた。ここで、 $k_x = 5, k_y = 5$  とした。IDO 法に対してディリクレ境界条件を与えるため、物理量 ( $f$ ) とその空間微係数 ( $f_x, f_y, f_{xy}$ ) に関して解析解を与えた。レッドブラックアルゴリズムを適用した数値計算手法による計算結果から

$$\sum_i \sum_j \left| \frac{f(i, j) - f_{\text{analytic}}(i, j)}{f_{\text{analytic}}(i, j)} \right| / \text{MeshNumber} \quad (4.50)$$

を用いて求めた数値誤差を図 4.10 に示す。

図 4.1 に示した自然なアルゴリズムを用いた SOR 法による数値計算から得られた数値計算誤差と比較すると、完全に同じ誤差で計算が可能であることがわかる。これにより、IDO 法による 2 次元ポアソン方程式の解法に対してレッドブラックアルゴリズムが適用可能であることを数値的に示したと言える。

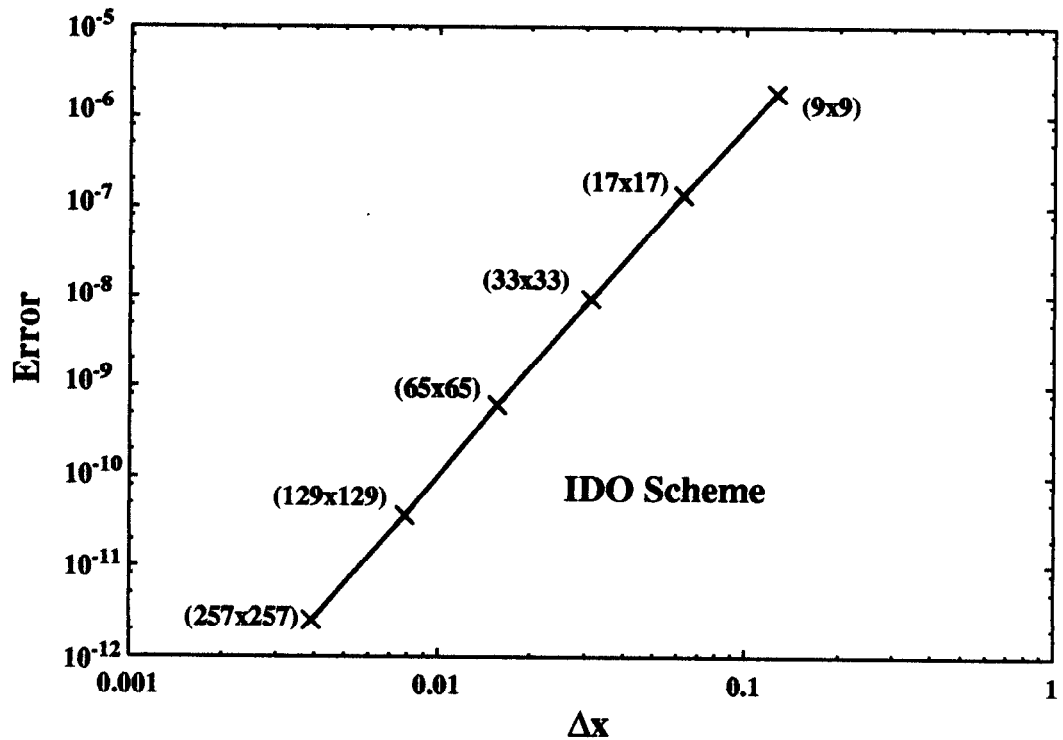


図 4.10: レッドブラック法を適用した IDO 法による 2 次元ポアソン方程式の数値計算誤差  
 レッドブラックアルゴリズムを適用した IDO 法による 2 次元ポアソン方程式の数値計算結果から得られた数値計算誤差は自然な計算手順による 2 次元 IDO 法による数値計算結果から得られた数値計算誤差と完全に一致する。

3 次元 IDO 法によるポアソン方程式解法の高速度化を考えた場合、現在の計算機では 1 次元問題や 2 次元問題と比較して 3 次元問題に対してグリッド数を大きく利用することが困難であることから、マルチグリッド法の適用を行うことに比べ、並列化やベクトル化がその効力を大きく発揮できると考えられる。そこで、3 次元ポアソン方程式解法への適用を考え、3 次元での レッドブラックアルゴリズムの適用方法を簡単に説明する。図 4.11 に示したように、2 次元で用いた レッドブラック配列を 3 次元的に交互に組み合わせた配列が 3 次元問題に於けるレッドブラックアルゴリズムの配列となる。この様な配列を用いることにより IDO 法による 3 次元ポアソン方程式の解法を用いる場合、Red 格子間ならびに Black 格子間での更新計算に依存関係が消滅する。

物理量ならびにその空間微係数で与える境界条件を解析解を用いて与えるディリクレ型境界条件に対する レッドブラックアルゴリズムを適用した IDO 法による 3 次元ポアソン方程式の数値拡散誤差を計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, -1 \leq z \leq 1$  に対してポアソン方程式の定数項  $\rho$  を

$$\rho(x, y, z) = \cos(k_x x) \cos(k_y y) \cos(k_z z) \tag{4.51}$$

と与え、空間解像度に対して見積もる。ここで、 $k_x = \pi, k_y = \pi, k_z = \pi$  と与えた。数値計算誤差は、

$$\sum_i \sum_j \sum_k \left| \frac{f(i, j, k) - f_{\text{analytic}}(i, j, k)}{f_{\text{analytic}}(i, j, k)} \right| / \text{MeshNumber} \tag{4.52}$$

を用いて評価する。ここで、 $f_{\text{analytic}}$  は解析解を示す。

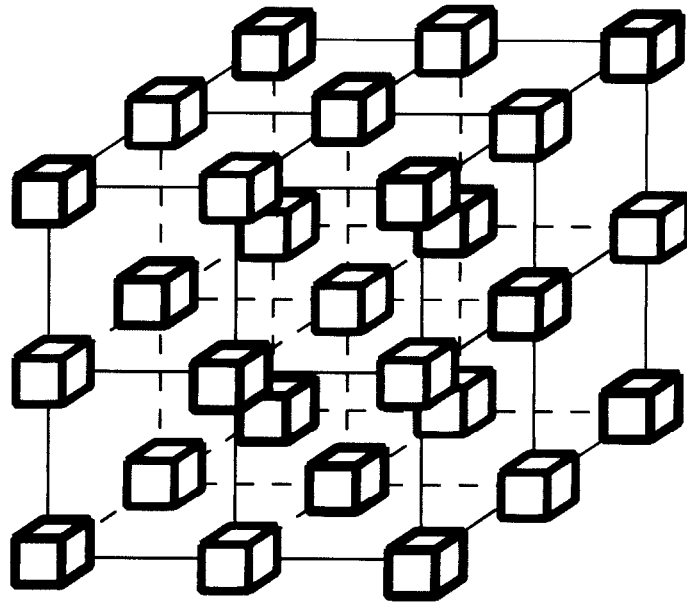


図 4.11: 3次元 レッドブラック格子配列  
 2次元問題で利用した レッドブラック格子を交互に重ね合わせることで3次元 レッドブラックアルゴリズムの格子を形成できる。

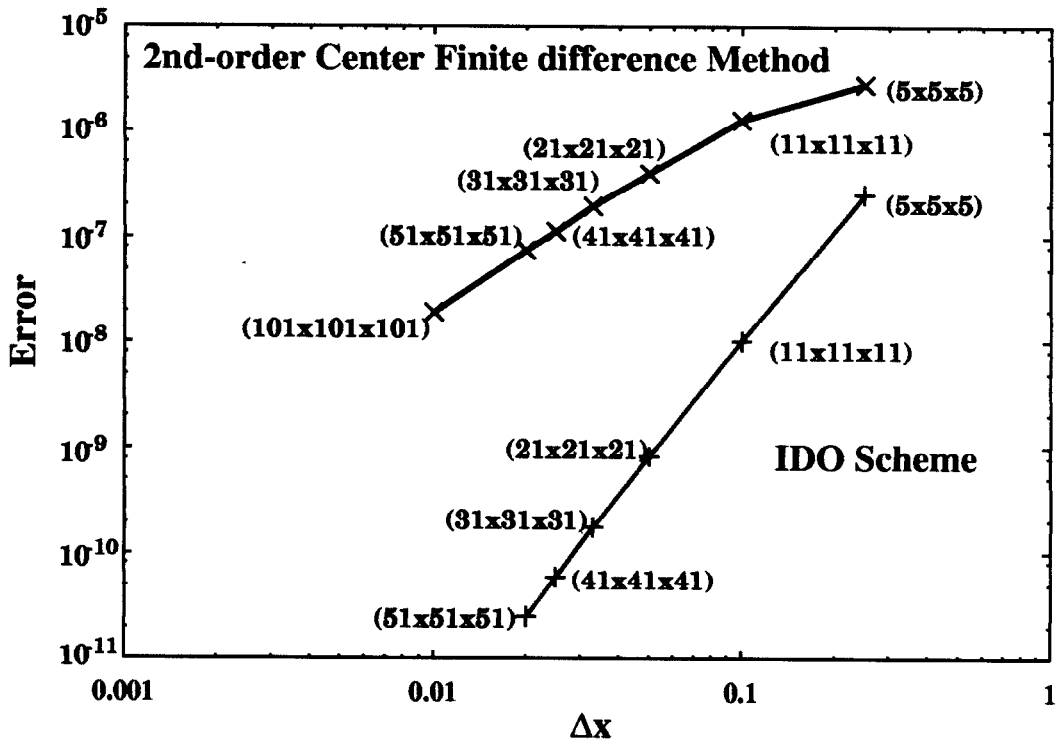


図 4.12: レッドブラック法を適用した IDO 法による 3次元ポアソン方程式の数値計算誤差  
 レッドブラックアルゴリズムを適用した IDO 法による 3次元ポアソン方程式の数値計算結果から得られた数値計算誤差は自然な計算手順による 3次元 IDO 法による数値計算結果から得られた数値計算誤差と完全に一致する。

図 4.12 に示した計算結果は 図 4.5 に示した自然なアルゴリズムによる計算結果から得た数値誤差と一致する結果を得ることが出来た。このことから、3 次元ポアソン方程式に対してもレッドブラックアルゴリズムが適用可能であることを数値実験的に明らかにした。

#### 4.1.4 並列計算による計算速度向上

図 4.1.2 に示したマルチグリッド法による計算の高速度化はアルゴリズム的な視点をもちいたソフトウェアの高速度化であるとする、並列計算機、ベクトル計算機による高速度化はハードウェア的な高速度化であると言える。IDO 法による 2 次元ポアソン方程式の解法では現在の計算機記憶容量でマルチグリッド法に対して充分効果的な 2 次元計算空間を確保できることからその適用を行い、計算時間改善について議論した。しかし、3 次元問題を考えた場合、マルチグリッド法は計算格子点数増加に伴う緩和時間の増大を改善する手法であるため、現在の計算機能力ではマルチグリッド法の効果を十分に発揮できるだけの計算格子点数を確保するのが困難であることから、並列化による計算時間の高速度化が有効であると考え、その検証を行う。

プログラムを並列化ならびにベクトル化可能なコードに書き換えるためにはレッドブラックアルゴリズムのように計算手順から空間依存性を除く手法を適用する必要があるが、前節で IDO 法による 3 次元ポアソン方程式の解法に対してレッドブラックアルゴリズムが適用可能であることを数値実験的に既に示した。本節ではレッドブラックアルゴリズムを適用した IDO 法による 3 次元ポアソン方程式の解法を用いた並列計算を行い、計算ノード (Processor Element; PE) 数に対する計算時間を検証する。

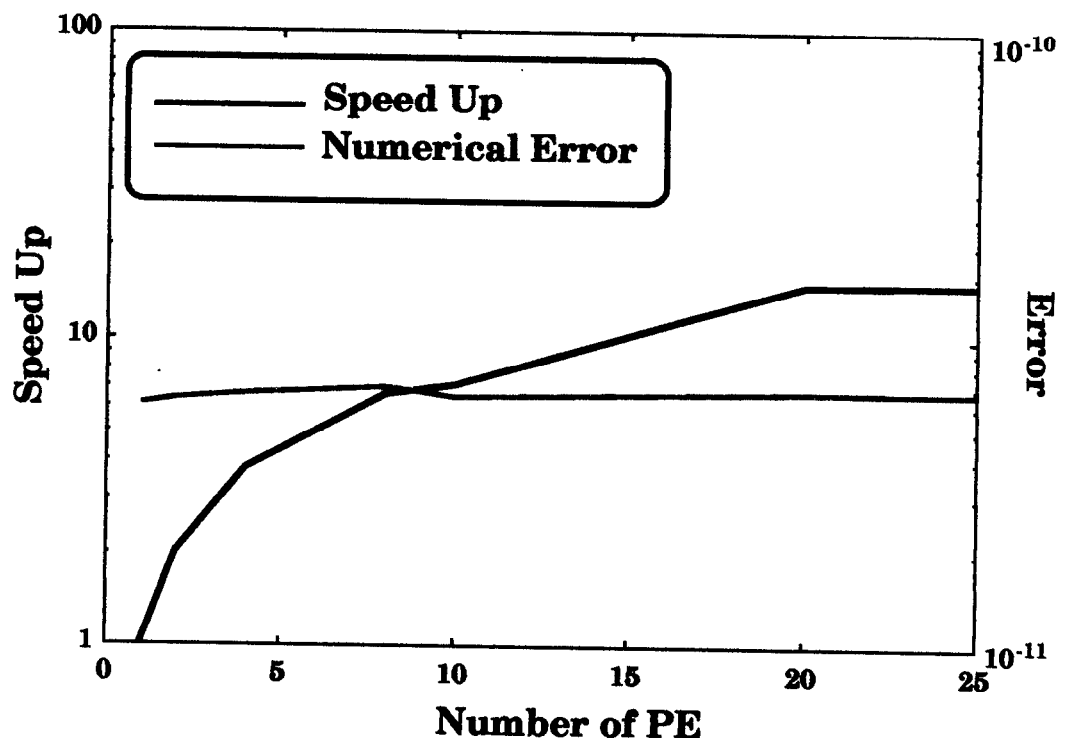


図 4.13: IDO 法による 3 次元ポアソン方程式の計算時間

計算ノード数を増加させた結果から得られる計算誤差が 1 ノードを用いた計算結果から得られる数値誤差にほぼ一致していることから数値計算が正しく行われていることがわかる。計算速度向上は 20 ノードで約 15 と得られた。

並列計算を可能にするためにノード間の通信アルゴリズムを実装するため、MPI (Message Passing Interface) ライブラリを用いた。並列計算を高速化するためには数値計算とノード間通信を同時に行う非同期通信を利用すべきであるが、ここでは通信の安全性を確保するために数値計算とノード間通信を交互に行う同期通信を用いたプログラムを独自に開発した。並列計算機は東京工業大学情報処理センターが所有する MIPS RISC R12000 300MHz を 128 個搭載した SGI の Origin 2000 system を利用した。

計算グリッド数  $51 \times 51 \times 51$  を使い、計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, -1 \leq z \leq 1$  においてポアソン方程式の定数項を  $\rho(x, y, z) = \cos(k_x x) \cos(k_y y) \cos(k_z z)$  と与え、ディリクレ境界条件化で数値誤差を

$$\sum_i \sum_j \sum_k \left| \frac{f(i, j, k) - f_{\text{analytic}}(i, j, k)}{f_{\text{analytic}}(i, j, k)} \right| / \text{MeshNumber} \quad (4.53)$$

を用いて評価した。ここで、 $k_x = k_y = k_z = \pi$  とした。また、ノード数 1 における計算時間を基準として計算時間向上率を算出した結果を図 4.13 に示す。

計算ノード数に関わらず、計算誤差は図 4.12 に示した値によく一致した結果を得ることが出来たことから、並列計算が適当な精度で行えていると言える。また計算時間向上率を検証した結果、少ない格子点数  $51 \times 51 \times 51$  で検証しているにも関わらず 20 ノードで 15 倍程度の加速を確認した。20 ノード以上を用いた計算では 1 ノードに含まれる計算格子点数が少なくなることから計算負荷に対してノード間通信に要する負荷が増大するため計算時間向上率が頭打ちになる結果となった。ノード間通信を非同期通信にすることにより計算時間向上率に若干の改善が見られると思われる。



## 4.2 移流計算の高速化 (セミ・ラグランジュ CIP 法)

セミ・ラグランジュ手法は気象問題 [35] などの比較的移流速度の時間変動が緩やかな問題に対して大きな  $\Delta t$  を利用することにより高速な数値計算を行う場合に適用される。

IDO 法はオイラー手法を用いた計算スキームであるため、セミ・ラグランジュ手法の適用を行うためにはその計算手続きの根本を見直す必要がある。一方、エルミート補間を用いたもう一つの計算手法である CIP 法は基礎的なアルゴリズムとしてセミ・ラグランジュ手法を採り入れており、速やかに高速解法を適用できると考える。しかし CIP 法はセミ・ラグランジュ法の特徴の一つである大きな CFL 数に対する安定移流計算を行うことができず、オイラースキーム同様に  $CFL = |u\Delta t/\Delta x| \leq 1$  の条件を満たさなければならない [36]。CIP 法では計算対象点を含む 1 グリッド内を補関数により Remapping し、セミ・ラグランジュ的手法により移流計算を行うため、 $CFL > 1$  での計算では補関領域を越えた場所からの情報の移流、つまり補関数の外挿を行うことになる。この外挿が数値不安定性の原因と考えられる。また、CIP 法では計算領域で移流速度が変化する場合、 $|\Delta u\Delta t/\Delta x| \leq 1$  を満たしていなければならなかった。ここで、 $\Delta u = u_{i+1} - u_i$  (or  $u_i - u_{i-1}$ ) を表している。CIP 法では移流原点を時間後退法 (backward - in - time) で求めているため、メッシュ間の速度差で  $\Delta t$  で進む距離がメッシュ間隔を越えると風上のプロファイルが風下のプロファイルを追い越してしまい、保存則に従わなくなることから数値不安定が生じる。

これらの問題を解消するため、CFL が 1 を越えても安定に計算できるように補関数を移流原点位置で適切に形成することと、移流原点を 4 次 Runge-Kutta 法で求めることで時間後退法を用いた場合に見られたような計算誤差による矛盾を減少させることを提案し、CFL  $> 1$  で安定動作するセミ・ラグランジュ CIP 法の構築を目指す。

### 4.2.1 セミ・ラグランジュ手法

一般的に非保存系の流体方程式は下記の偏微分方程式 (Partial Differential Equation; PDE) によって記述できる。

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \nabla f = RHS, \quad (4.54)$$

ここで、 $f$  は物理量を表し、RHS は移流速度  $\mathbf{u}$  や  $f$  とその空間微係数から構成される非移流相を示す。移流速度  $\mathbf{u}$  は、 $\mathbf{u} = \partial \mathbf{x} / \partial t$  と定義される。式 (4.54) を移流原点  $(\mathbf{x}_0, t_0)$  から数値積分することにより  $(\mathbf{x}_1, t_1)$  における物理量  $f$  を見積もることができる。このことから、式 (4.54) は空間、時間からなる連続空間  $\mathbb{R}^3 \times \mathbb{R}^1$  内の 2 点  $(\mathbf{x}_1, t_1)$ 、 $(\mathbf{x}_0, t_0)$  における物理量  $f$  の関係を表していると言える。ストークスの定理から、2 点  $(\mathbf{x}_1, t_1)$ 、 $(\mathbf{x}_0, t_0)$  をつなぐ軌跡  $C$  に沿った積分を下記のように表すことができる [37]。

$$f(\mathbf{x}_1, t_1) = f(\mathbf{x}_0, t_0) + \int_C (d\mathbf{x} - \mathbf{u}dt) \cdot \nabla f + \int_C RHS dt \quad (4.55)$$

式 (4.55) は、 $\mathbf{x}_0, \mathbf{x}_1$  の選択により以下の 3 通りの表記方法に分類できる。

- $\mathbf{x}_0 = \mathbf{x}_1$  とした場合  
 $d\mathbf{x} = 0$  であるから、式 (4.55) は、

$$f(\mathbf{x}_1, t_1) = f(\mathbf{x}_0, t_0) + \int_C -(\mathbf{u} \cdot \nabla f) dt + \int_C RHS dt \quad (4.56)$$

のようにオイラー表記となる。

- $\mathbf{x}_0$  を  $\mathbf{x}_1$  の移流原点とした場合

$$\mathbf{x}_1 = \mathbf{x}_0 + \int_0^1 \mathbf{u}(\mathbf{x}, t) dt \quad (4.57)$$

より、参照座標は移流速度  $u$  で動くと考えると、式 (4.55) は、

$$f(x_1, t_1) = f(x_0, t_0) + \int_C RHS dt \tag{4.58}$$

と、ラグランジュ表記される。

- 上記の両方に当てはまらない場合  
 オイラー法にもラグランジュ法にも属さない  $x_0, x_1$  の採り方は無限にある。これらの手法は混合オイラーラグランジュ法と呼ばれている。

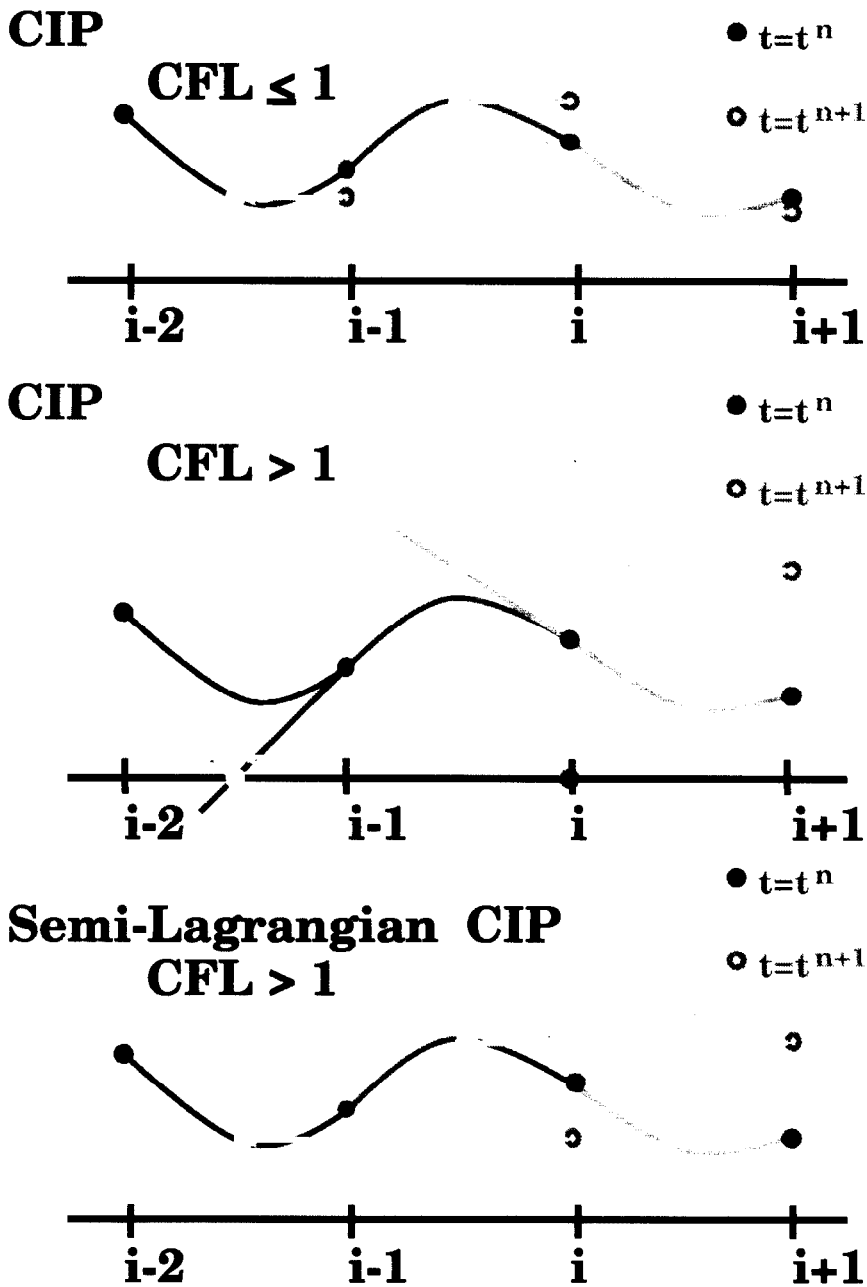


図 4.14: セミ・ラグランジュ CIP 法の移流計算手法  
 CIP 法では  $CFL > 1$  で計算を行う場合、補間関数による外挿のため数値不安定性が生じる。この問題を解決するため、移流原点位置において 3 次多項式による補間を行うことを提案する。

混合オイラーラグランジュ法の中で、オイラー座標上の点を到着点  $(x_1, t_1)$  とし、後退積分によって移流原点  $(x_0, t_0)$  を求めるものをセミ・ラグランジュ法と一般的に呼んでいる。

#### 4.2.2 セミ・ラグランジュ CIP 法の構築

CIP 法はセミ・ラグランジュスキームに分類されることがあるが、セミ・ラグランジュ法の代表的な性質の一つである、精度を保ったまま CFL 条件に制約されない計算手法がこれまで示されていない。図 4.14 に示すように CFL  $> 1$  での計算では、計算対象点を含む 1 グリッド内に補間関数を形成する CIP 法の手法を用いると、補間関数を形成領域外の値を補間関数から求めるため補間精度が悪くなることの原因と考えられる。セミ・ラグランジュ CIP 法では CFL  $> 1$  での計算を安定に行うために移流原点を含む計算領域 (cell) で補間関数を形成し、移流原点に於ける物理量、空間微分値等をセミ・ラグランジュ的に求める手法を提案する (図 4.14)。

また、通常の CIP 法では移流原点位置  $(x_0, t_0)$  を時間後退法により求めるため、 $u_i < 0$  の場合、 $\Delta u = u_i - u_{i+1}$  が正であれば、風上の速度が風下の速度より大きいことになり、 $u_i > 0$  の場合、 $\Delta u = u_{i-1} - u_i > 0$  であれば、同様に風上の速度が風下の速度より大きいことになるので、移流速度によりプロファイルが 1 タイムステップで進む距離の差  $\Delta u \Delta t$  が、メッシュ間隔を越えるため ( $\Delta u \Delta t / \Delta x > 1$ )、風上のプロファイルは風下のプロファイルを追い越し、保存則を満たさなくなる (図 4.15)。そこで、移流速度が時間の関数でない ( $\partial u / \partial t = 0$ ) ことを仮定し、移流原点を 4 次精度 Runge-Kutta 法で求めることで、積分精度による数値誤差の減少を図る。

$$(x_0, t_0) = (x_1, t_1) + \int_{t_1}^{t_0} \mathbf{u}(\mathbf{x}, t) dt \quad (4.59)$$

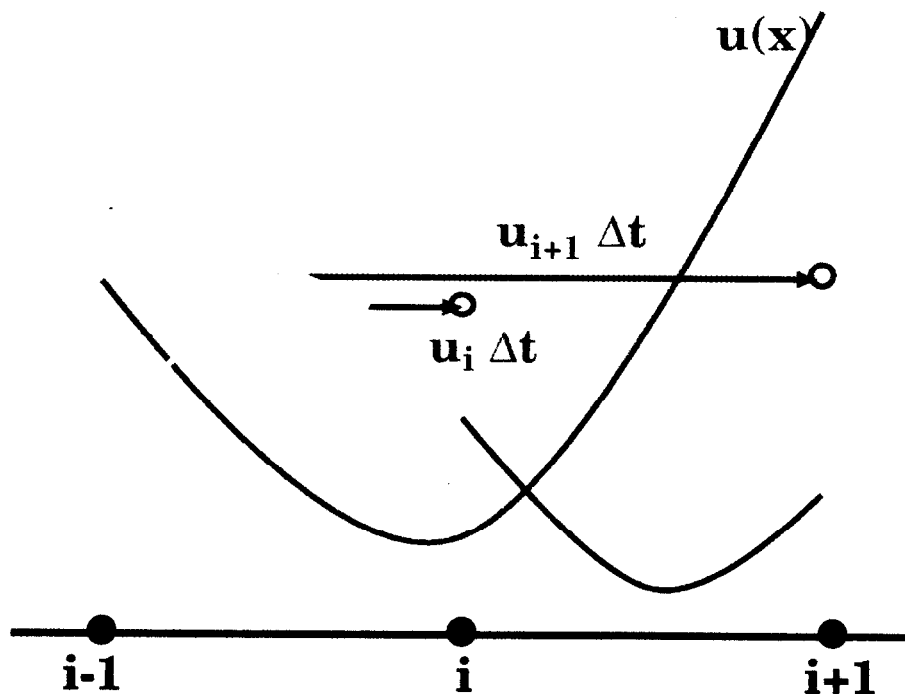


図 4.15: セミ・ラグランジュ手法の時間後退法による数値不安定性:

非一様速度場で時間後退法を用いるとグリッド間に大きな速度差がある場合、風下の情報が風上の情報を追い越してしまう場合がある。このことが原因となり、保存則に従わなくなる点が生じることから数値不安定が起こるため、移流原点位置の高精度な計算が必要となる。

ただし、Runge-Kutta 法にも分解能の限界があるため、より複雑な移流経路への対応を考え、式 (4.59) の計算に、仮想時間ステップ  $\lambda_s$  を導入する ( $t_1 - t_0 = \Delta t = \sum_s \lambda_s$ )。  $\lambda_s$  を用いることで式 (4.59) は、

$$(\mathbf{x}_0, t_0) = (\mathbf{x}_1, t_1) + \sum_{s=0}^{S-1} \int_{\lambda_s}^{\lambda_{s+1}} \mathbf{u}(\mathbf{x}, t) dt \quad (4.60)$$

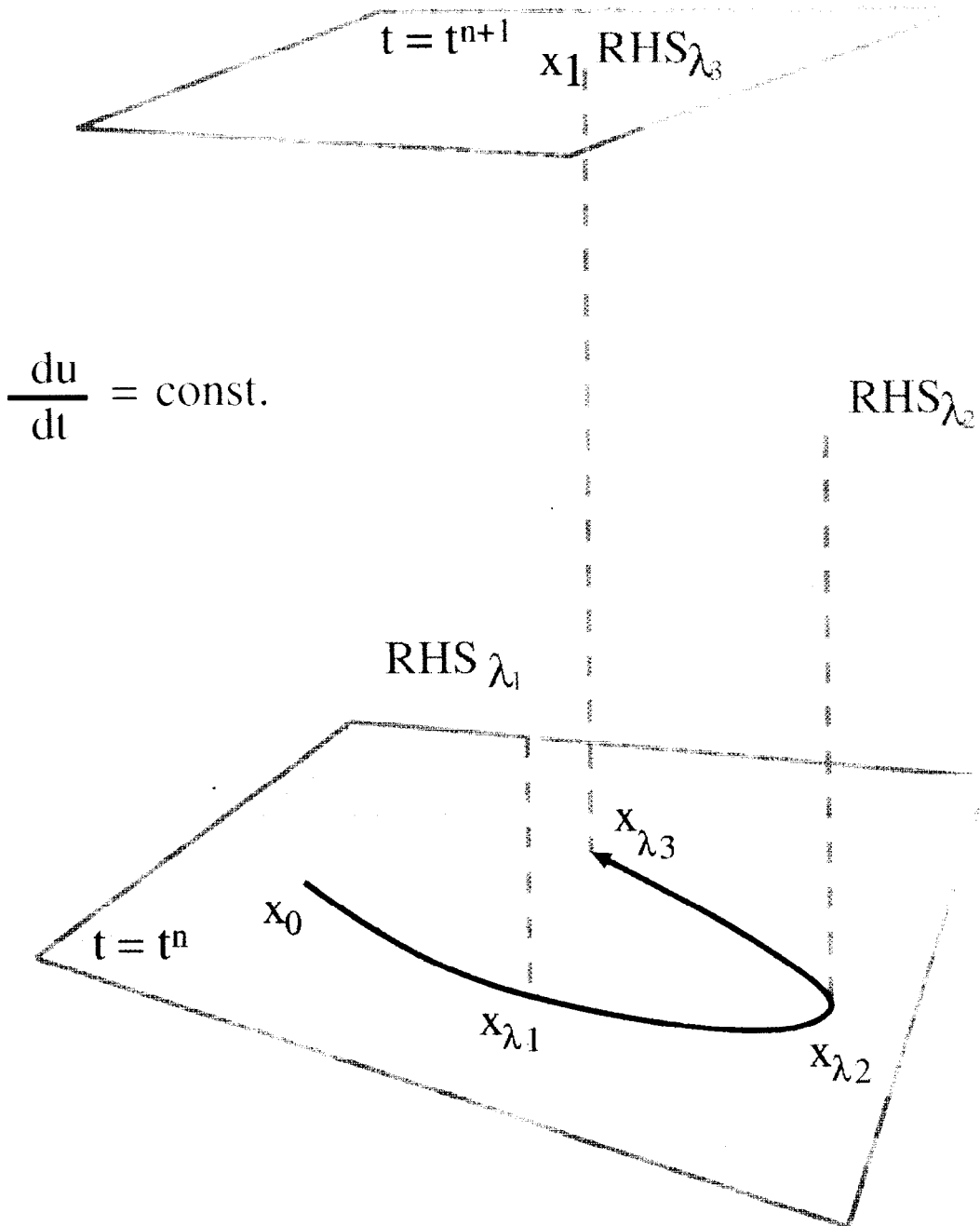


図 4.16: セミ・ラグランジュ法 の概念図

定常速度度場を仮定すると、移流経路が  $\Delta t$  内で求められるため、その移流原点位置における物理量の計算と、移流経路に沿った非移流相積分を行うことが可能である。

と表すことができる。4次精度 Runge-Kutta 法により  $\int_{\lambda_s}^{\lambda_{s+1}} \mathbf{u}(\mathbf{x}, t) dt$  を計算することで移流原点  $(\mathbf{x}_0, t_0)$  を精度良く求めることができる。また、 $\partial \mathbf{u} / \partial t = 0$  を仮定していることから、時間  $\lambda_s$  毎の位置、つまり移流経路  $C$  を求めることができるので、 $\mathbf{u}(\mathbf{x}, \lambda_s), f(\mathbf{x}, \lambda_s)$  と、その1階空間微分値も CIP 法の3次エルミート補間式から移流経路  $C$  に沿って求めることができる。速度の空間変化が急な領域でも、十分に空間解像度が高ければ仮想時間  $\lambda$  への分割数  $S$  を増加させることで複雑な移流経路  $C$  にも対応が可能な手法である。また、移流経路  $C$  を精度良く求めることは非移流相 (式 (4.62)) の積分精度の向上につながる。セミ・ラグランジュ CIP 法では式 (4.55) を下記のように fractional step で計算を行う。

$$f^* = f(\mathbf{x}_0, t_0) + \int_C -\mathbf{u} dt \cdot \nabla f \quad (4.61)$$

$$f(\mathbf{x}_1, t_1) = f^* + \int_C RHS dt. \quad (4.62)$$

移流相 (式 (4.61)) は解析的に解くことが可能であり、その解は  $f(\mathbf{x}_1, t_1) = f(\mathbf{x}_1 - \int_C \mathbf{u}(\mathbf{x}, t) dt, t_1 - \Delta t) = f(\mathbf{x}_0, t_0)$  となる。

式 (4.55) 中の非移流相成分  $RHS$  は通常、物理量  $f$ 、移流速度  $\mathbf{u}$  と、それらの空間微分項で形成されている。CIP 法ではこれらの値を式 (A.13) や、式 (A.43) ~ (A.45) を用いて時間積分する。

## 4.2.3 計算結果

### 1 次元移流方程式の検証

#### 1 次元移流方程式

$$f_t + u f_x = 0 \quad (4.63)$$

をセミ・ラグランジュ CIP 法で解き、移流問題に対する基本的な性質について検証する。ここで、添字  $t, x$  はそれぞれ時間、空間に関する偏微分を示す;  $f_t = \partial f / \partial t, f_x = \partial f / \partial x$ 。セミ・ラグランジュ CIP 法の構築にあたり、移流原点位置計算を非一様速度空間において高精度化するため時間後退法に替えて4次精度 Runge-Kutta 法を採用入れたことから、空間、時間一定の移流速度場と、時間変化無し、空間変化有りの2通りについて検証を行う。

空間変化が無い速度場 ( $u_x = 0$ ) における1次元移流方程式の数値解から得られる数値拡散誤差と、保存誤差について検証する。CIP 法では物理量に加えてその1次空間微係数を従属変数として独立に与えるため、それぞれの時間更新を行う必要があることから支配方程式は以下ようになる。

$$f_t + u f_x = 0 \quad (4.64)$$

$$(f_x)_t + u (f_x)_x = 0 \quad (4.65)$$

移流速度は全空間で1とし、初期プロファイル  $f$  を、空間  $0 \leq x \leq 1; \Delta x = 1/100$  に、正弦関数で与えた ( $\sin(k_x x)$ )。数値計算結果から得られる数値拡散誤差 ( $\sum_i (|f_{analytic,i} - f_i| / f_{analytic,i})$ )、保存誤差 ( $(\sum_i f_{analytic,i} - \sum_i f_i) / \sum_i (f_{analytic,i})$ ) をそれぞれ初期プロファイルの波数に対して見積もった結果を図 4.17 に示す。ここで、 $f_{analytic}$  は解析解を示し、 $f_i$  は計算結果をあらわす。

セミ・ラグランジュ CIP 法は CIP 法と同様に CFL 数が整数で表される場合、グリッド点上の値をグリッド点上に移流させる計算になるため、数値拡散誤差は丸め誤差程度で得られる。CFL 数が整数でない場合の数値拡散誤差は、CIP 法で得られる数値拡散誤差と完全に一致した結果を得た。CFL 数が1を越えた場合では計算手順の性質から、CFL 数の小数点以下の値での CIP 法による数値計算結果と同程度の数値拡散誤差を示す。例として、セミ・ラグランジュ CIP 法により CFL 数 100.5 で数値計算した場合では、各計算ステップ毎に生じる数値誤差は、CIP 法により CFL 数 0.5 で計算した場合と完全に同じものとなる。ただし、セミ・ラグランジュ CIP 法による数値計算では目的計算時刻までの計算ステップ数を減少することが可能であることから、計算結果から得られる数値誤差が CIP 法に比べ少ない結果となる。

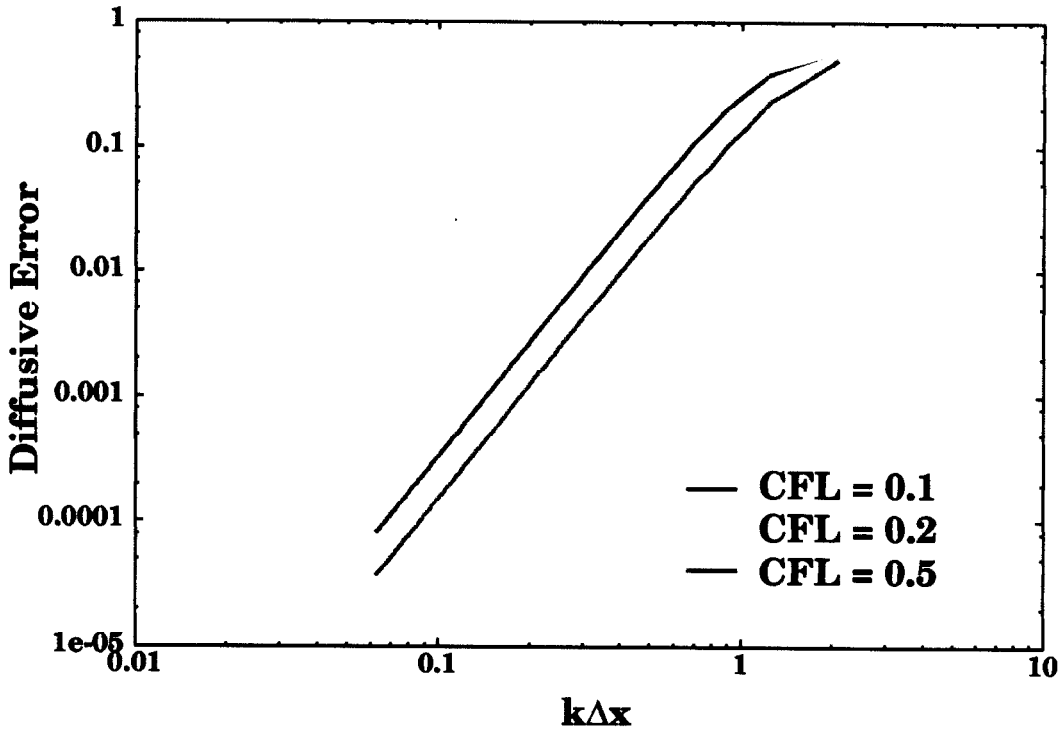


図 4.17: セミ・ラグランジュ CIP 法による 1 次元移流方程式の数値誤差:

1 次元移流方程式における初期プロファイルの波数  $k$  対解析解からの数値拡散誤差。移流速度は計算空間  $0 \leq x \leq 1$ ;  $\Delta x = 1/100$  で、常に 1。

CFL 数が整数値でない場合には、より大きな CFL 数で計算することにより  $\Delta t$  が大きくなるため対象計算時間までに要する時間ステップ数を減少することができるため、演算毎に発生する誤差が低減することから数値拡散誤差を減少させることが可能になる。また、セミ・ラグランジュ CIP 法においても CIP 法と同様に波数、CFL 数にかかわらず移流方程式から得られた数値計算結果の保存誤差は丸め誤差程度になり、定常一様速度場に於ける 1 次元移流方程式の数値計算では非保存系スキームを用いているにも関わらず保存が成立していることを確認した。

空間非一様速度場に於ける 1 次元移流方程式をセミ・ラグランジュ法で扱い、非移流相の計算手法について検証する。CIP 法を用いて空間速度変化のある場で移流方程式を計算する場合、グリッド毎に定義されている物理量の 1 階空間微分値の時間前進に於ける速度の空間微係数の取り扱いに注意しなければならない。扱う偏微分方程式は以下のようになる。

$$f_t + u f_x = 0 \quad (4.66)$$

$$(f_x)_t + u(f_x)_x = -u_x f_x \quad (4.67)$$

速度が一様な空間では、 $u_x = 0$  であったため、空間微係数に関する時間前進型偏微分方程式の右辺が 0 であった。しかし、移流速度の空間変動を考えた場合、非移流相の時間積分を考えなければならない。移流項の空間微分によって出現したこの項を、移流経路  $C$  に沿って時間積分する手段として、CIP 法で用いられている 2 次精度中心差分法を用いた計算手法と、微係数を用いないシンプソン則 (Simpson A)、微係数を用いるシンプソン則 (Simpson B) による積分の 3 通りの手法について検証を行う。シンプソン則に関するアルゴリズムは Appendix B.1 に付記した。1 次元移流方程式の非移流項積分をそれぞれの手法で扱う

場合、その計算手続きは次のようになる。

- CIP 法で用いられる非移流相積分

$$f_i^{n+1} = f_i^* \quad (4.68)$$

$$f_{x,i}^{n+1} = f_{x,i}^* + \frac{f_{i+1}^{n+1} - f_{i-1}^{n+1}}{2\Delta x} - \frac{f_i^{n+1} - f_i^*}{2\Delta x} - u_{x,i} f_{x,i}^* \Delta t \quad (4.69)$$

ただし、 $f^*$  は、セミ・ラグランジュ CIP 法で移流相の計算を行った後の、更新された値を示す。

- Simpson A による非移流相積分

$$f_i^{n+1} = f_i^* \quad (4.70)$$

$$\begin{aligned} f_{x,i}^{n+1} &= f_{x,i}^* + \int_t^{t+\Delta t} (-u_x f_x) dt \\ &= f_{x,i}^* + \sum_i \int_{\lambda_i}^{\lambda_{i+1}} (-u_x f_x) dt \\ &= f_{x,i}^* + \sum_i -\frac{\Delta\lambda}{6} \left[ u_x(\lambda_i) f_x(\lambda_i) + 4u_x \left( \frac{\lambda_i + \lambda_{i+1}}{2} \right) f_x \left( \frac{\lambda_i + \lambda_{i+1}}{2} \right) + u_x(\lambda_{i+1}) f_x(\lambda_{i+1}) \right]. \end{aligned} \quad (4.71)$$

ただし、 $f^*$  はセミ・ラグランジュ CIP 法で移流相計算を行った後の、更新された値を示す。

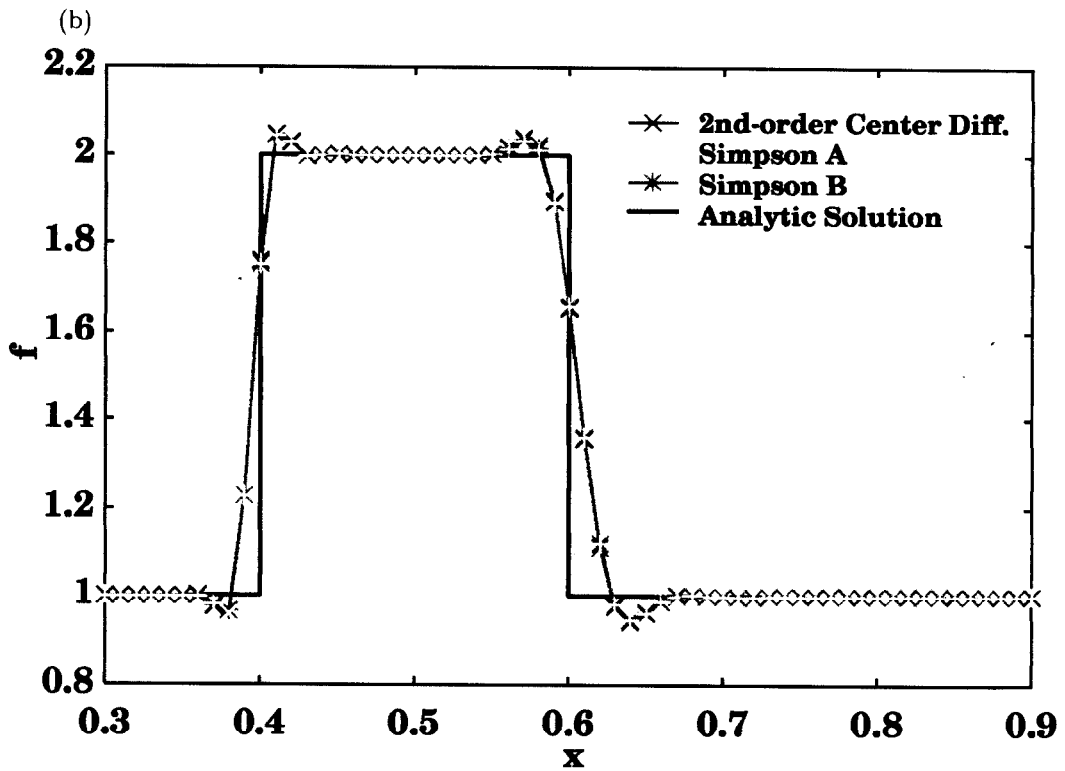
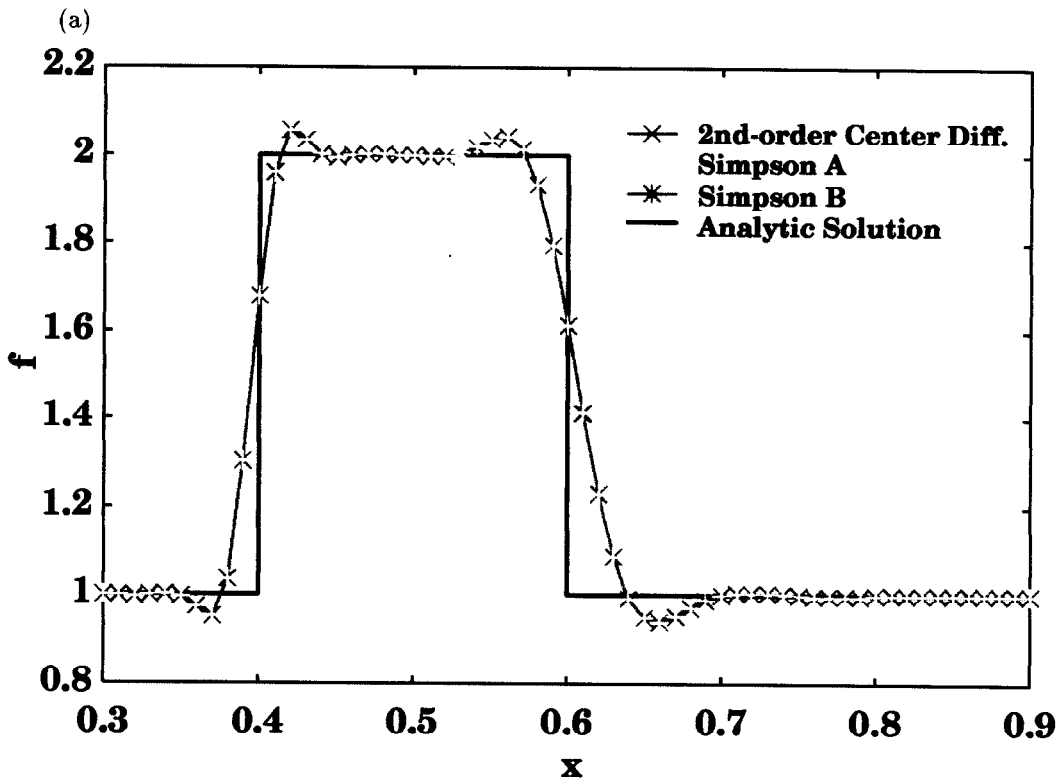
- Simpson B による非移流相積分

$$f_i^{n+1} = f_i^* \quad (4.72)$$

$$\begin{aligned} f_{x,i}^{n+1} &= f_{x,i}^* + \int_t^{t+\Delta t} (-u_x f_x) dt \\ &= f_{x,i}^* + \sum_i \int_{\lambda_i}^{\lambda_{i+1}} (-u_x f_x) dt \\ &= f_{x,i}^* + \sum_i \left[ -\frac{\lambda}{2} [u_x(\lambda_i) f_x(\lambda_i) + u_x(\lambda_{i+1}) f_x(\lambda_{i+1})] \right. \\ &\quad \left. + \frac{\lambda^2}{12} [(-u_x f_{x,i} - u_{x,i} f_i) - (-u_{x,i+1} f_{x,i+1} - u_{x,i+1} f_{i+1})] \right] \end{aligned} \quad (4.73)$$

ただし、 $f^*$  はセミ・ラグランジュ CIP 法で移流相計算を行った後の、更新された値を示す。Simpson B では、時間の微係数を必要とするため、対象としている時間前進型偏微分方程式を直接代入することで計算を行う。

空間  $0 \leq x \leq 1$ ,  $\Delta x = 1/100$  に対し、速度場を  $u(x) = 1/(1 + 0.5 \sin(2\pi x))$  と与え、初期プロファイルとして  $f = 2(0.4 \leq x \leq 0.6)$ ,  $f = 1$ (その他の場所) を与えた。CFL =  $\Delta t/\Delta x = 0.5, 2, 5, 20$  を用いて時間  $t = 1$  まで計算した結果をでの計算結果を図 4.18 (a), (b), (c), (d) に示す。CFL 数は空間で速度が非一様であるため代表速度 1 を用いて定義した。解析解は、速度場から初期プロファイルに完全に一致する。2 次精度中心差分法、Simpson A, Simpson B による非移流項積分計算を行った結果、全ての積分手法において  $\Delta t/\Delta x$  を増加した場合でも解析解に良く一致した解を得た。ただし、CFL 数を増加させるに従いオーバーシュートが目立つ結果となった。計算結果から保存誤差ならびに数値拡散誤差をそれぞれ





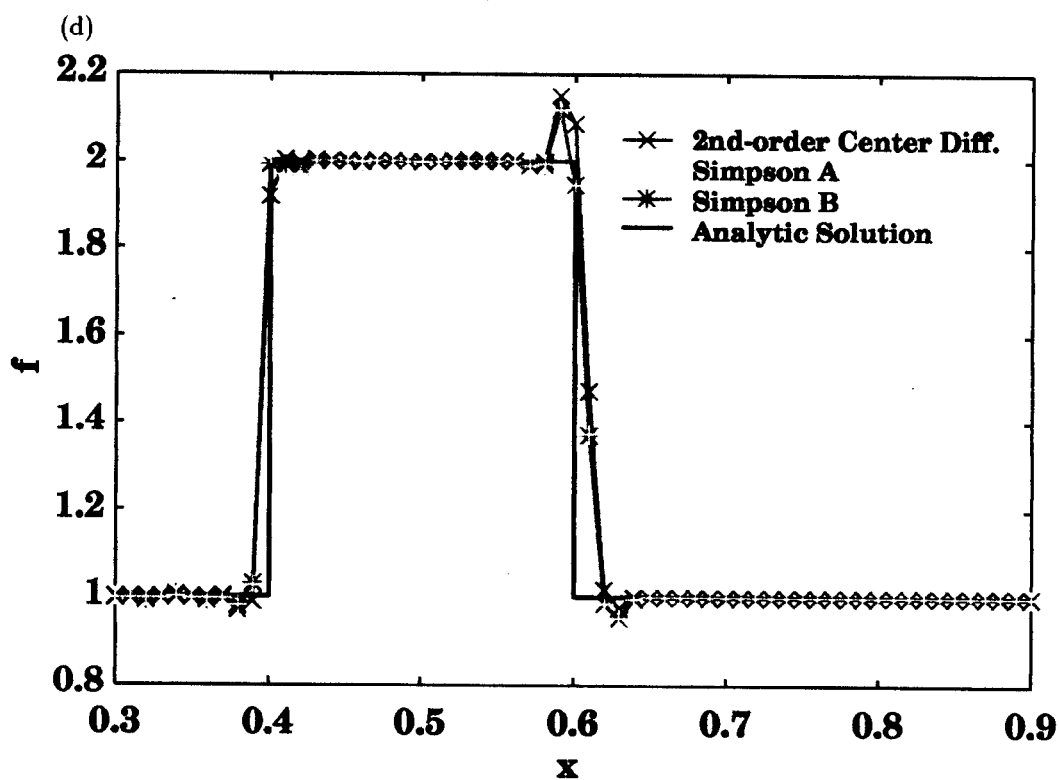
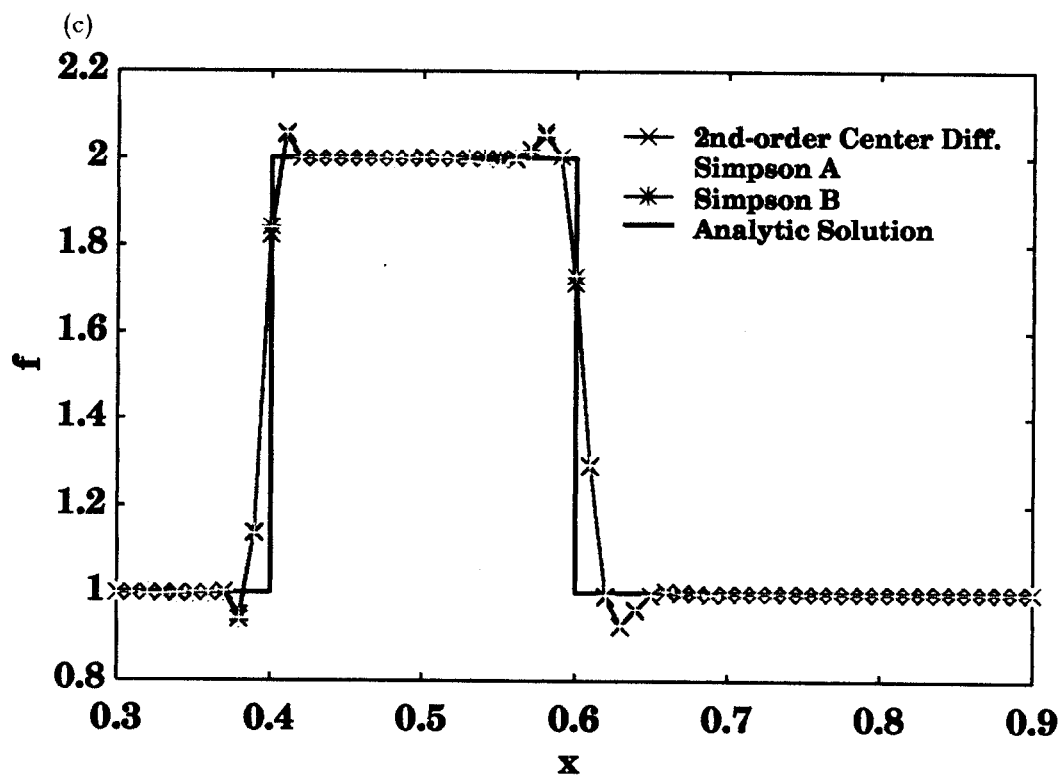


図 4.18: 非一様速度場における移流方程式の解法

空間  $0 \leq x \leq 1$ ,  $\Delta x = 1/100$  において速度場  $u(x) = 1/(1 + 0.5 \sin(2\pi x))$  を与え、1次元移流方程式を解いた。(a)  $\Delta t/\Delta x = 0.5$ , (b)  $\Delta t/\Delta x = 2$ , (c)  $\Delta t/\Delta x = 5$ , (d)  $\Delta t/\Delta x = 20$  での結果を示す。

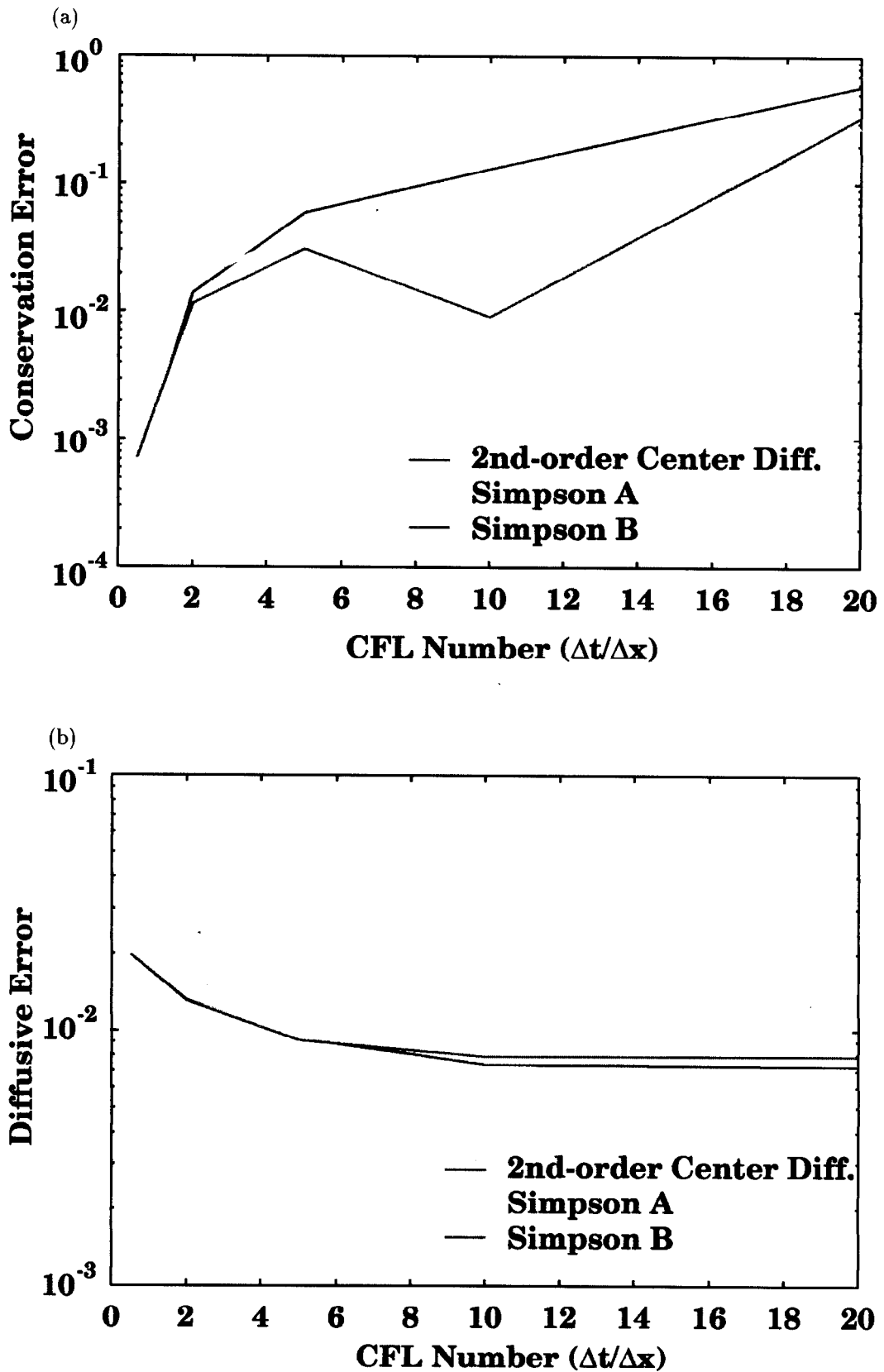


図 4.19: 非一様速度場における移流方程式の数値誤差  
 セミ・ラグランジュ CIP 法から得られる (a) 保存誤差ならびに (b) 数値拡散誤差。非移流相積分を高精度化することで最大  $10^1$  程度の保存性を改善できることを示した。ただし、数値拡散誤差に関しては、数値不安定性の問題から 2 次精度中心差分法を用いた手法が良い結果を示した。

$$\text{保存誤差} = \left| \frac{\sum_i f_i - \sum_i f_{Analytic,i}}{\sum_i f_{Analytic,i}} \right| \quad (4.74)$$

$$\text{数値拡散誤差} = \sum_i \left| \frac{f_i - f_{Analytic,i}}{f_{Analytic,i}} \right| \quad (4.75)$$

を用いて計算した結果を図 4.19 に示す。ここで、 $f_i$  は計算結果を示し、 $f_{Analytic,i}$  は解析解を示す。全ての非移流相積分手法において CFL 数 ( $= \Delta t / \Delta x$ ) が増大するに従い保存誤差が増大する傾向が見られ、数値拡散誤差に関しては減少する傾向が見られた。保存誤差に関しては 2 次精度中心差分法に比べシンプソン則により非移流相積分を行う手法が精度の高い値を示したが、数値拡散誤差に関しては今回対象とした全ての手法で同程度の値が得られた。

### 1 次元保存方程式の検証

非一様速度場に対するセミ・ラグランジュ CIP 法の有効性を 1 次元保存方程式

$$f_t + u f_x + u_x f = 0 \quad (4.76)$$

を解くことで保存性の観点から検証を行う。1 次元保存方程式に対する解析解は第 1 章 で用いた手法と同様に、式 (2.77) ~ 式 (2.86) を用いて計算した。

初期条件を  $u = 1/(1 + 0.5 \sin(2\pi x)) (0 \leq x \leq 1)$ ,  $f = 1 (0.4 \leq x \leq 0.6)$ ,  $f = 0$  (その他の点),  $\Delta x = 0.01$  として、 $CFL = \Delta t / \Delta x = 0.5, 1, 2, 3$  で計算した結果をそれぞれ 図 4.20 (a), (b), (c), (d) に示す。移流速度が空間で非一様であるため代表速度 1 を用いて CFL 数の定義を行った。 $CFL = \Delta t / \Delta x \leq 1$  では、非移流項の計算手法によらず、解析解に良く一致した結果を得た。 $CFL = \Delta t / \Delta x > 1$  では非移流項を Simpson A、Simpson B で計算した結果はプロファイルの右端で数値拡散により、鋭角に求まるべきプロファイルが鈍化した結果となった。また、Simpson B では、CFL 数を大きくとるに従い、数値不安定性によりプロファイル中に高波数を持つ誤差が生じている様子が分かる (図 4.21 (d))。

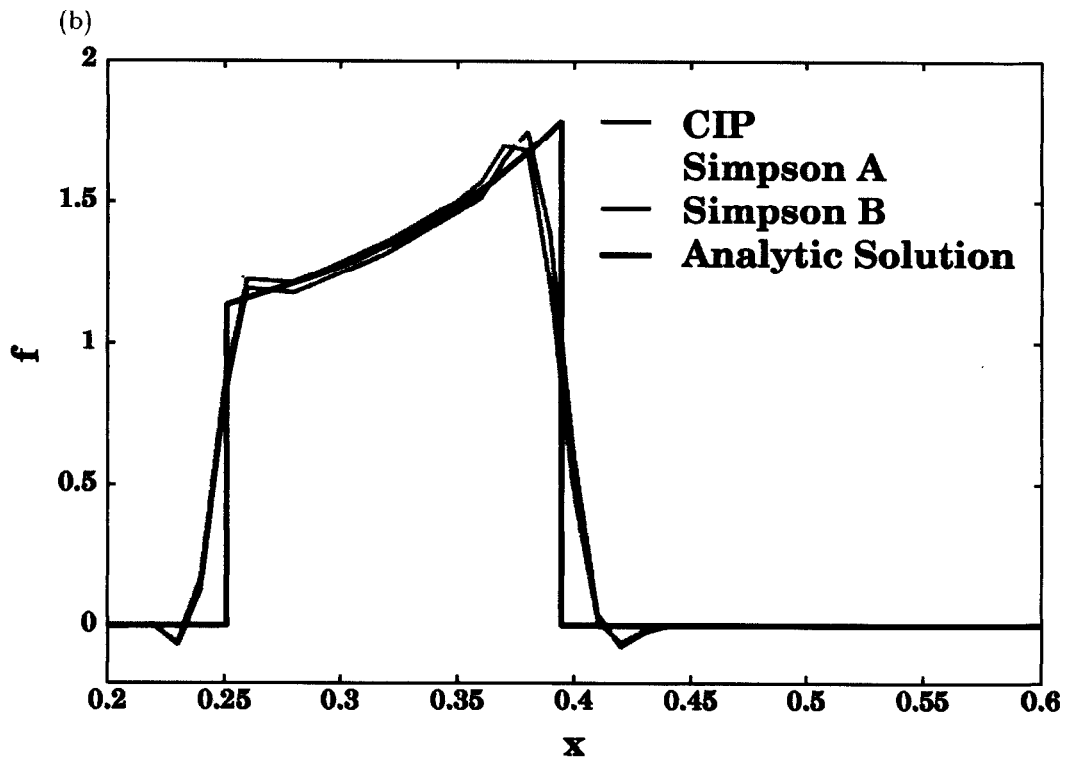
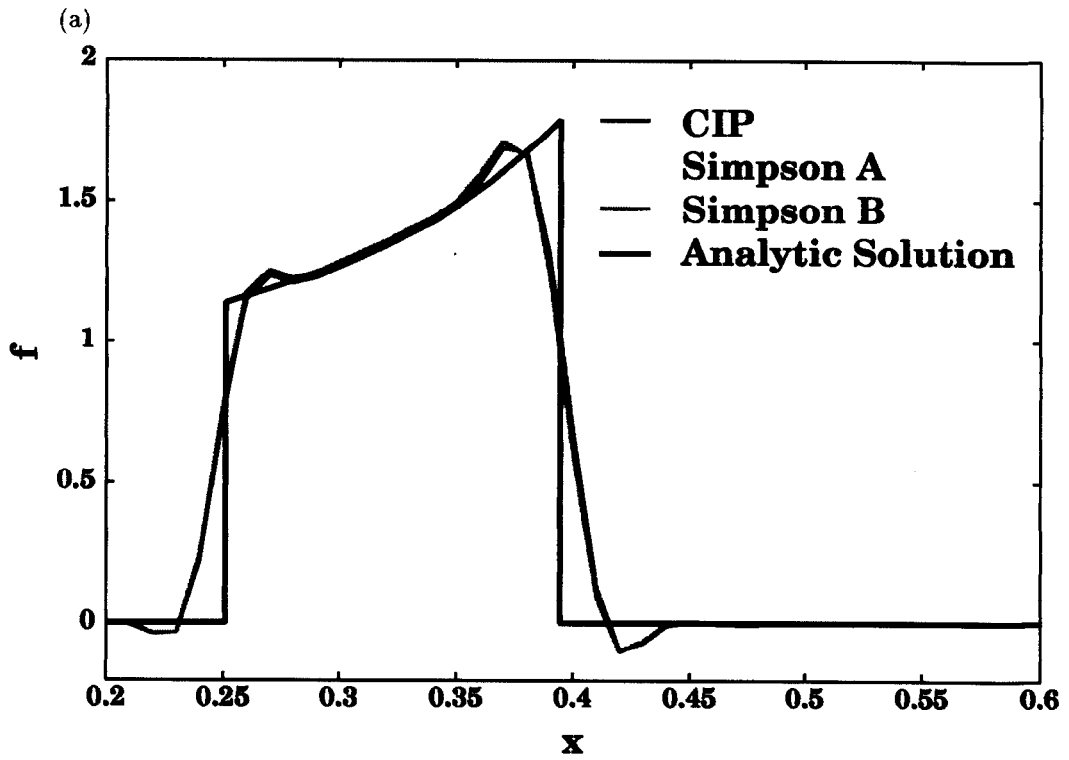
その一方で、CIP 法で用いられている 2 次中心差分法で非移流項時間積分した結果はプロファイルが減衰するものの、解析解の形状を良く保存した計算結果を示した。その理由として、シンプソン則を用いた両手法では移流経路に沿って非移流相積分を行っているため、数値計算誤差がより敏感に計算結果に影響していることが考えられる。その一方で、2 次精度中心差分法を用いた非移流相積分では、物理量計算時の非移流相積分をオイラー手法と同様に

$$\int R.H.S dt = R.H.S \Delta t \quad (4.77)$$

として求めていることから、計算精度が低下するものの安定な計算を提供することを可能にしていると考えられる。

計算結果から得られた保存誤差を図 4.21 に示す。保存誤差に関して微係数を用いるシンプソン則 (Simpson B) が良い結果を示しているが、 $\Delta t / \Delta x$  を大きくすると数値振動が生じた。微係数を用いないシンプソン則 (Simpson A) による非移流相積分に関しても CFL 数を増大させるに従い数値振動が見られる結果となった。一方 2 次精度中心差分法を用いた非移流相積分を行った場合、計算ステップの進行にともないプロファイルが減衰して消滅してしまうものの、 $\Delta t / \Delta x = 100$  でも安定な数値計算が可能であった。

以上から、セミ・ラグランジュ CIP 法を用いて本節で用いた条件下で 1 次元保存方程式を扱う場合、保存誤差が少なからず生じるものの、非移流項計算を 2 次中心差分法を用いる積分手法がシンプソン則を用いたものに比べ安定した数値解を与えることを示した。



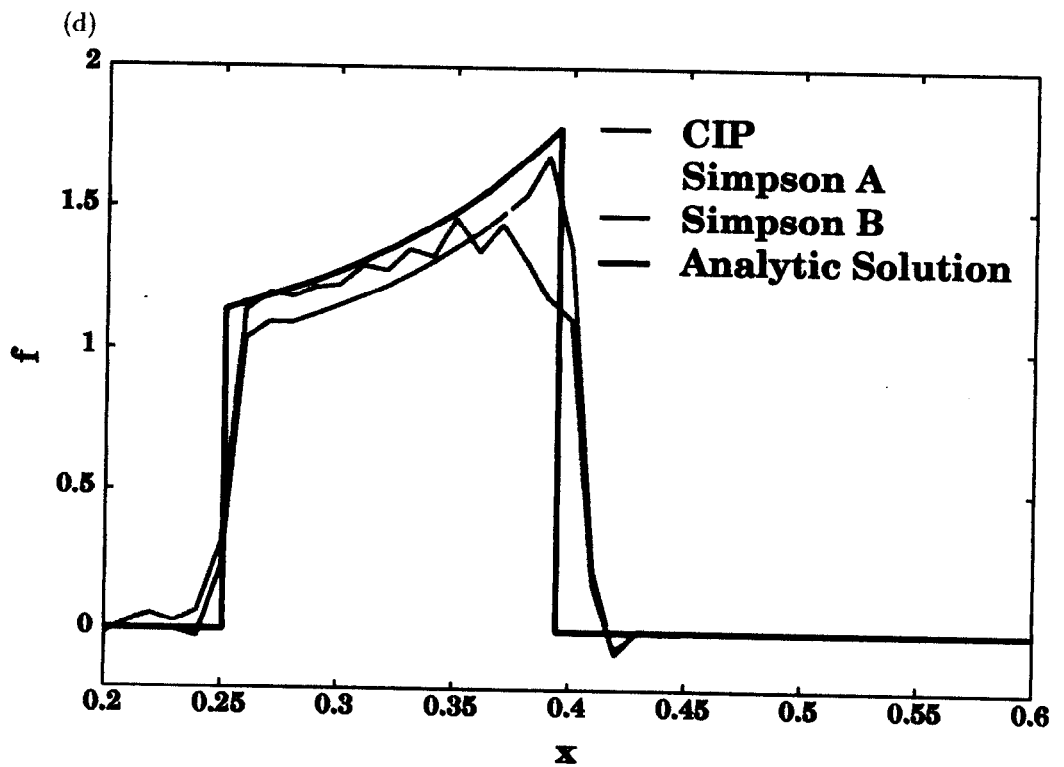
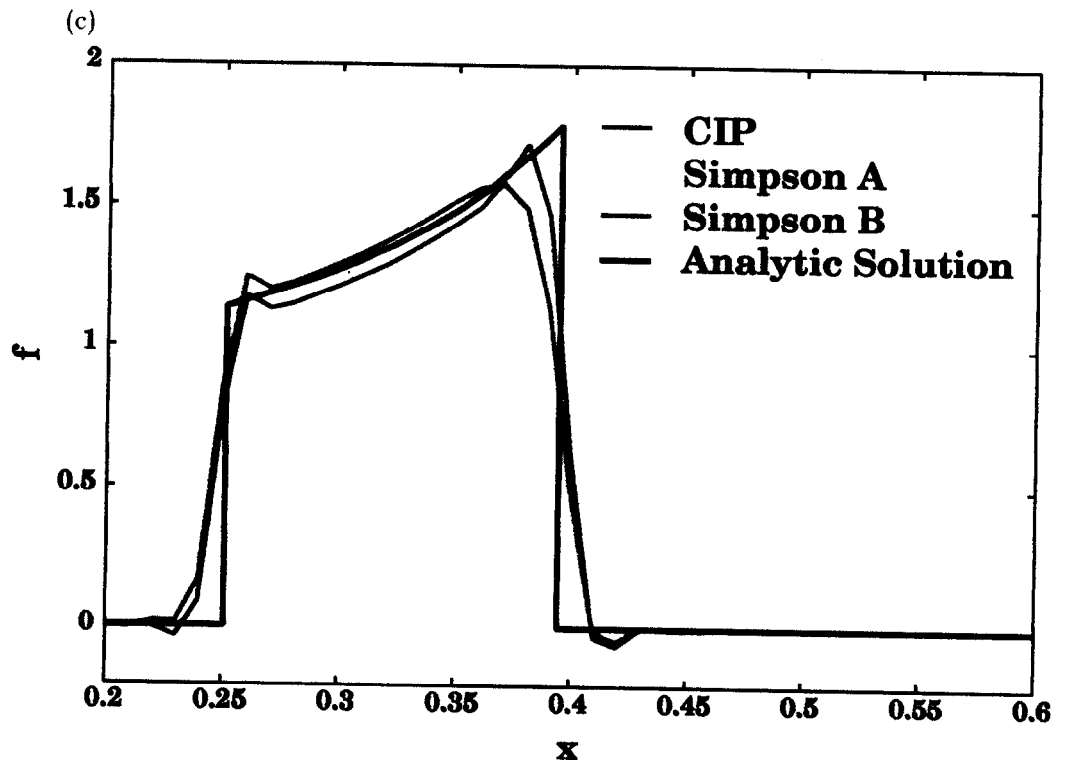


図 4.20: 非一様速度場における保存方程式の解法

空間  $0 \leq x \leq 1$ ,  $\Delta x = 1/100$  において速度場  $u(x) = 1/(1 + 0.5 \sin(2\pi x))$  を与え、1次元保存方程式を解いた。 $\Delta t/\Delta x = 0.5$  (a),  $\Delta t/\Delta x = 1.0$  (b),  $\Delta t/\Delta x = 2.0$  (c),  $\Delta t/\Delta x = 3.0$  (d) での結果を示す。

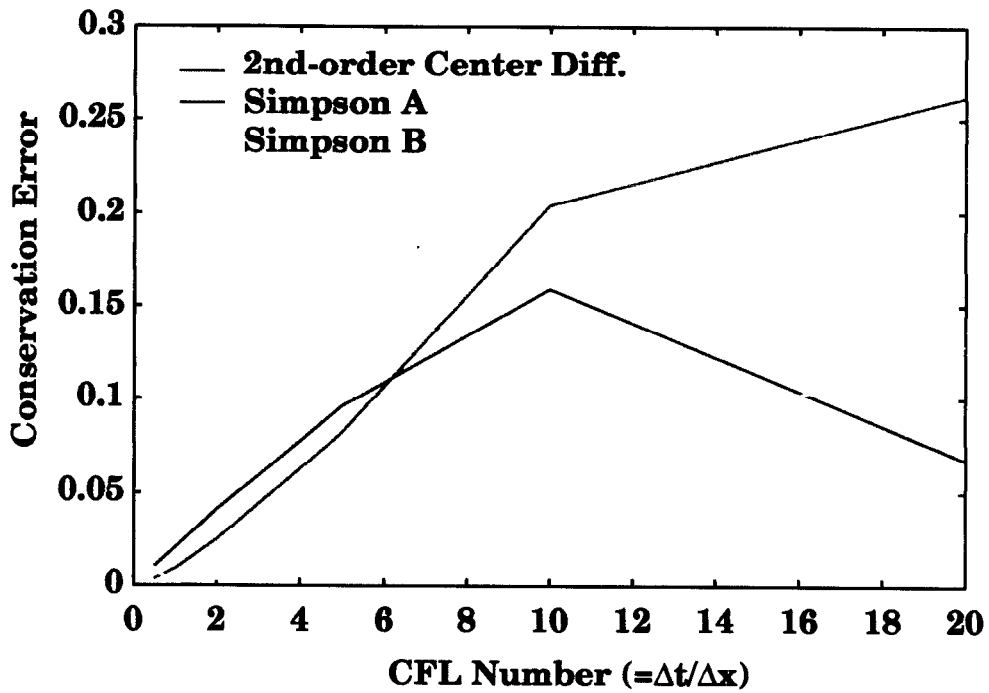


図 4.21: 非一様速度場での 1 次元保存方程式の保存誤差:  
通常の CIP スキーム、シンプソン則 (Simpson A)、微係数を用いるシンプソン則 (Simpson B) を用いて非移流相を扱った時の保存誤差

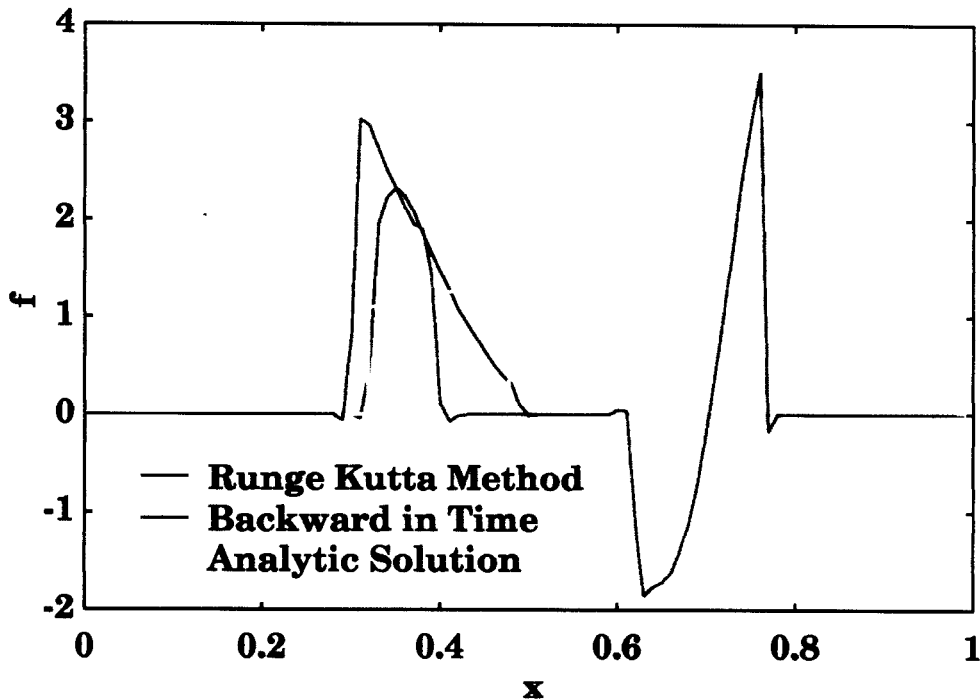


図 4.22: 移流原点計算精度の検証:  
移流原点計算を時間後退法により求めるセミ・ラグランジュ法では、空間に対する移流速度勾配の急激な場所で数値不安定性が生じる。

次に、移流原点の計算アルゴリズムとして採用した 4 次精度 Runge-Kutta 法について考察を行うため、対照実験として CIP 法で用いられている時間後退法を用いて移流原点計算を行い、1 次元保存方程式を扱う。非移流項の積分演算には高い安定性を保証する CIP 法で用いられている 2 次精度中心差分法を用いた。

計算空間  $0 \leq x \leq 1; \Delta x = 1/100$  に対して、移流速度  $u(x) = 1/(1 + 0.5 \sin(2\pi x))$  を与え、初期プロファイルを、 $f = 1(0.4 \leq x \leq 0.6)$ ,  $f = 0$ (その他の点) と与えた。 $\Delta u$  ( $u_i - u_{i+1}$  or  $u_{i-1} - u_i$ ) の最大値は  $5.3 \times 10^{-2}$  であるため、先に提案した条件  $\Delta u \Delta t / \Delta x < 1$  を満たす CFL ( $= \Delta t / \Delta x$ ) は、 $CFL < 18.9$  であるので、ここでは  $CFL = 30$  とし、時間  $t = 0.9$  まで計算を行う。ここで、CFL 数は移流速度が空間非一様場であるため、代表移流速度を 1 として定義した。

計算結果を図 4.22 に示す。非移流項計算の精度が低いため、プロファイル形状は、移流原点計算手法によらず解析解を表現しきれていない。移流原点計算に時間後退法を用いた計算結果では速度勾配が急激な領域 ( $0.6 \leq x \leq 0.9$ ) において、物理的には決して生じないプロファイルがみられるが、4 次精度 Runge-Kutta 法を用いた計算ではこのような顕差を生じることは無かった。 $CFL \leq 10$  では、時間後退法、4 次精度 Runge-Kutta 法、両手法で移流原点計算を行い 1 次元保存方程式を扱った場合、このような数値不安定性が生じなかったことから、速度勾配が急激な空間に於ける移流原点計算には高次精度スキームを用いる必要性があると結論付ける。

## 2 次元移流方程式の検証

セミ・ラグランジュ CIP 法を 2 次元に拡張し 2 次元移流方程式

$$f_t + uf_x + vf_y = 0 \quad (4.78)$$

に適用する。計算空間  $-0.5 \leq x \leq 0.5, -0.5 \leq y \leq 0.5, \Delta x = 1/100, \Delta y = 1/100$  で、剛体回転速度場  $u(x, y) = y, v(x, y) = -x$  を与え、Zalesak の例題を検証した。

非移流相の積分を CIP 法で用いられている手順、シンプソン則 (Simpson A)、微係数を用いたシンプソン則 (Simpson B) の 3 通りを用いて計算を行った。Zalesak 例題の初期プロファイルを図 4.23 に示す。回転速度場において 1 回転するまで計算を行うため、解析解も同型となる。非移流項の計算を CIP 法、Simpson A, Simpson B で行った計算結果をそれぞれ図 4.24、図 4.25、図 4.26 に示す。CIP 法では  $CFL = \Delta t / \Delta x \leq 1$  を満たさなければ安定に計算できないのに対し、セミ・ラグランジュ CIP 法では全ての非移流相計算手順において  $CFL = 200\pi$  で安定に計算することが可能であることを示した。CFL 数は空間非一様速度場であることから代表速度を 1 として定義した。

$CFL = 200\pi$  で計算した場合、1 ステップでプロファイルを計算空間の原点を中心に 1 回転させる計算になる。つまり、図 4.24 - 図 4.26 において、(a) は 4 ステップ、(b) は 3 ステップ、(c) は 2 ステップ、(d) は 1 ステップで計算が完了することになる。計算ステップ毎にプロファイルがそれぞれ  $\pi/2, \pi/3, \pi, 2\pi$  回転することから、直交座標系を用いていることを考慮すると、(a), (c), (d) のケースでは初期プロファイルの特性から理想的には、計算ステップ毎に計算グリッドに完全に写像できるため、(a), (c), (d) から見られるように解析解に良く一致した結果が得られた。(b) の場合でも多少のオーバーシュートがあるものの、プロファイルの境界を 1~2 メッシュで表現できていることが分かる。

Simpson A, Simpson B による非移流相計算を行うセミ・ラグランジュ CIP 法では、計算空間内に数値振動が生じる結果となった。とりわけ、計算が進行するに従って数値誤差が増大する傾向があることが (a), (b), (c) の結果から伺える。1 ステップで計算が完了する (d) では計算領域に誤差が見られなかったが、一般的な計算を行う場合 1 ステップで計算が終了するようなことは無いので、大きな CFL をとることによる誤差が無視できる場合のみシンプソン則を用いた非移流相積分が適用可能であると考えられる。

数値計算結果から得られた保存誤差と数値計算誤差を図 4.27 に示す。Zalesak の例題で用いられている初期プロファイルは CFL 数  $50\pi, 100\pi, 200\pi$  で計算ステップ毎に完全にグリッド上に写像できることから、それらの CFL 数では他と比較して飛躍的に保存誤差、数値拡散誤差を低減させることができたが、それらのばらつきを無視した場合においても非移流項計算にいずれの手法を用いても CFL 数を大きくとる

ことでセミ・ラグランジュ法の特徴の一つである、数値拡散誤差の低減を図ることが可能であった。また CIP 法に比べ保存誤差を改善することができた。

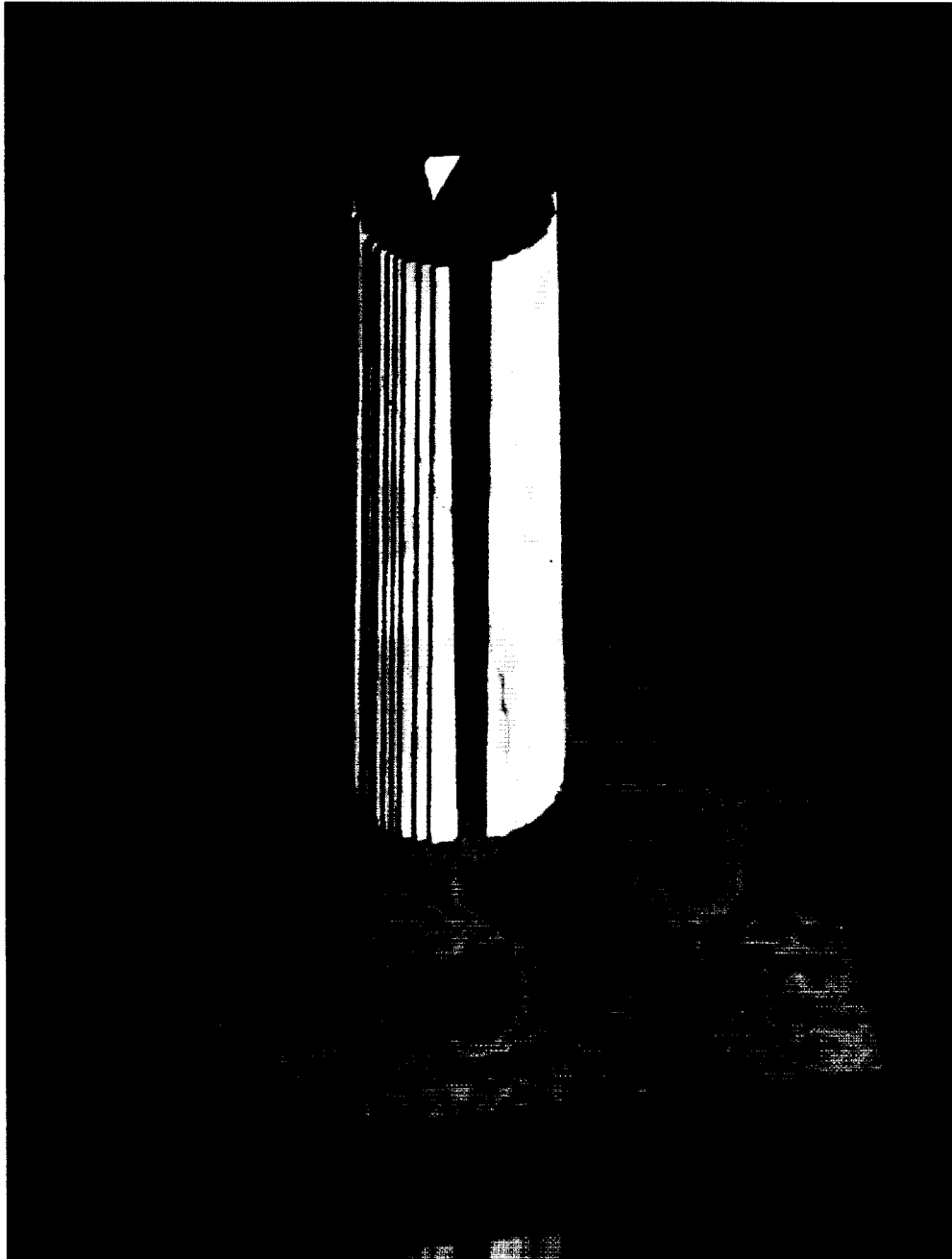


図 4.23: Zalesak 例題の初期プロファイル



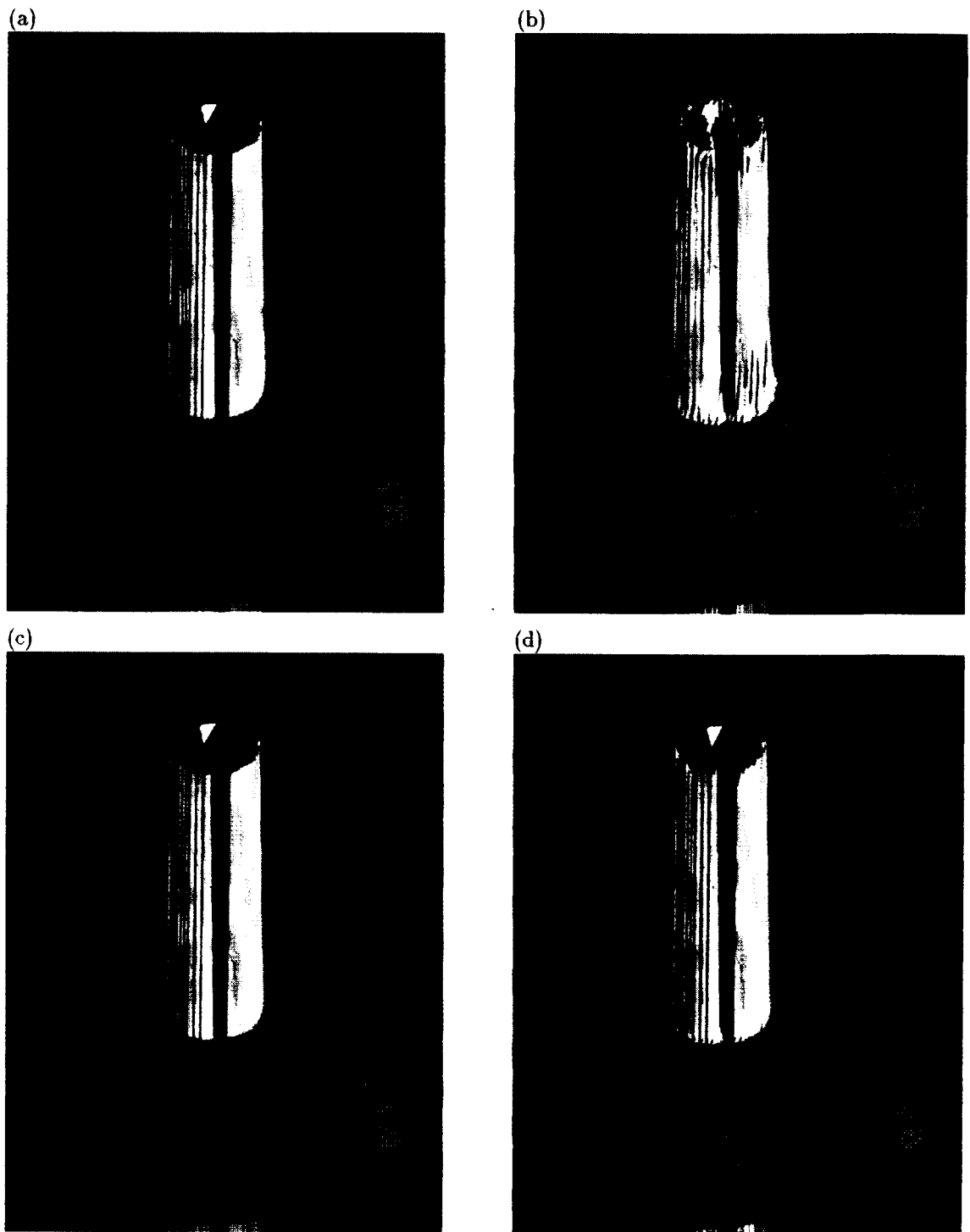


図 4.24: 非移流項に CIP 法の計算手法を用いたセミ・ラグランジュ法による Zalesak 例題の検証:  $CFL = \Delta t / \Delta x$  をそれぞれ (a)  $CFL = 50\pi$ , (b)  $CFL = 66,6\pi$ , (c)  $CFL = 100\pi$ , (d)  $CFL = 200\pi$  で計算した。

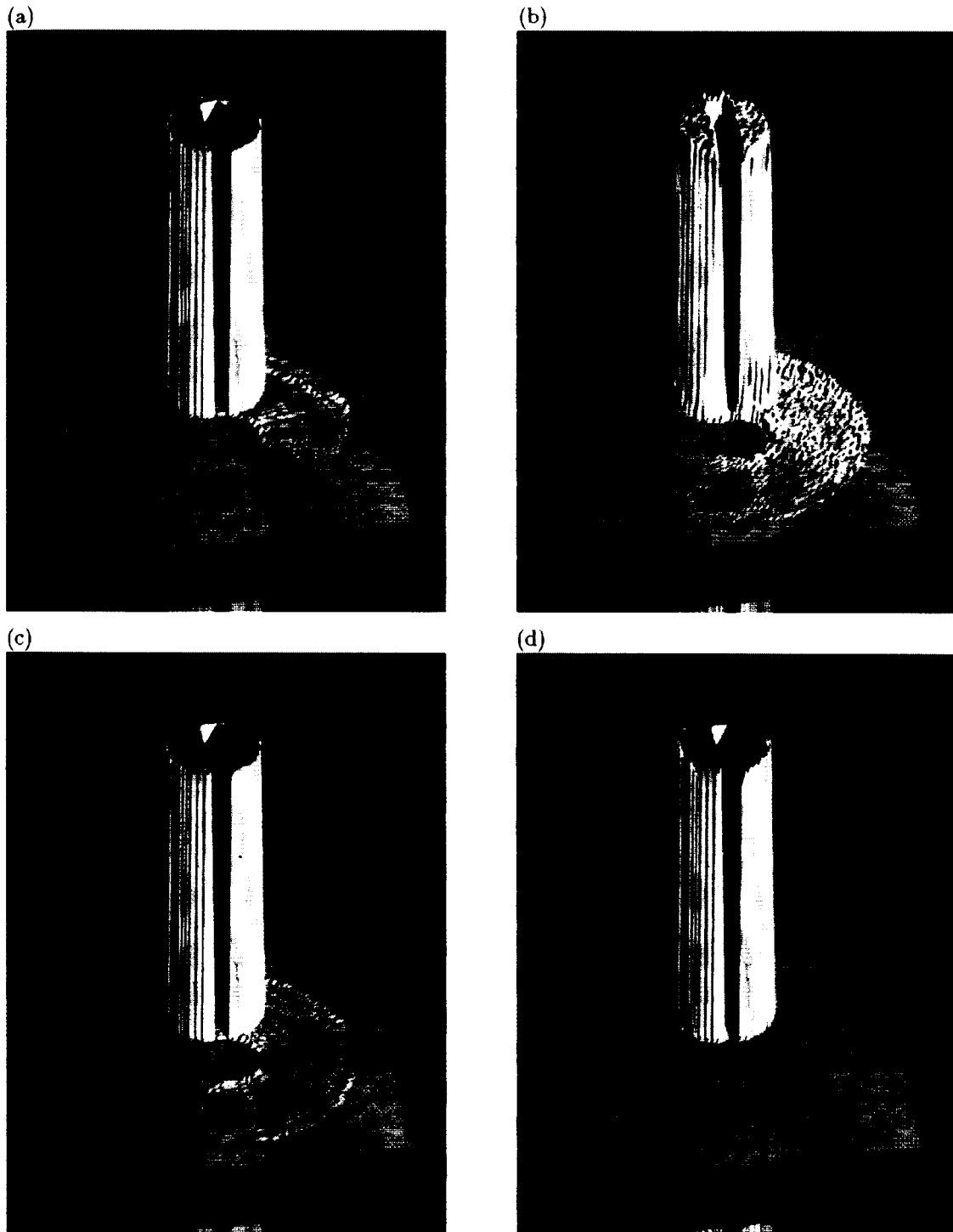


図 4.25: 非移流項に Simpson A を用いたセミ・ラグランジュ法による Zalesak 例題の検証:  $CFL = \Delta t / \Delta x$  をそれぞれ (a)  $CFL = 50\pi$ , (b)  $CFL = 66\pi$ , (c)  $CFL = 100\pi$ , (d)  $CFL = 200\pi$  で計算した。

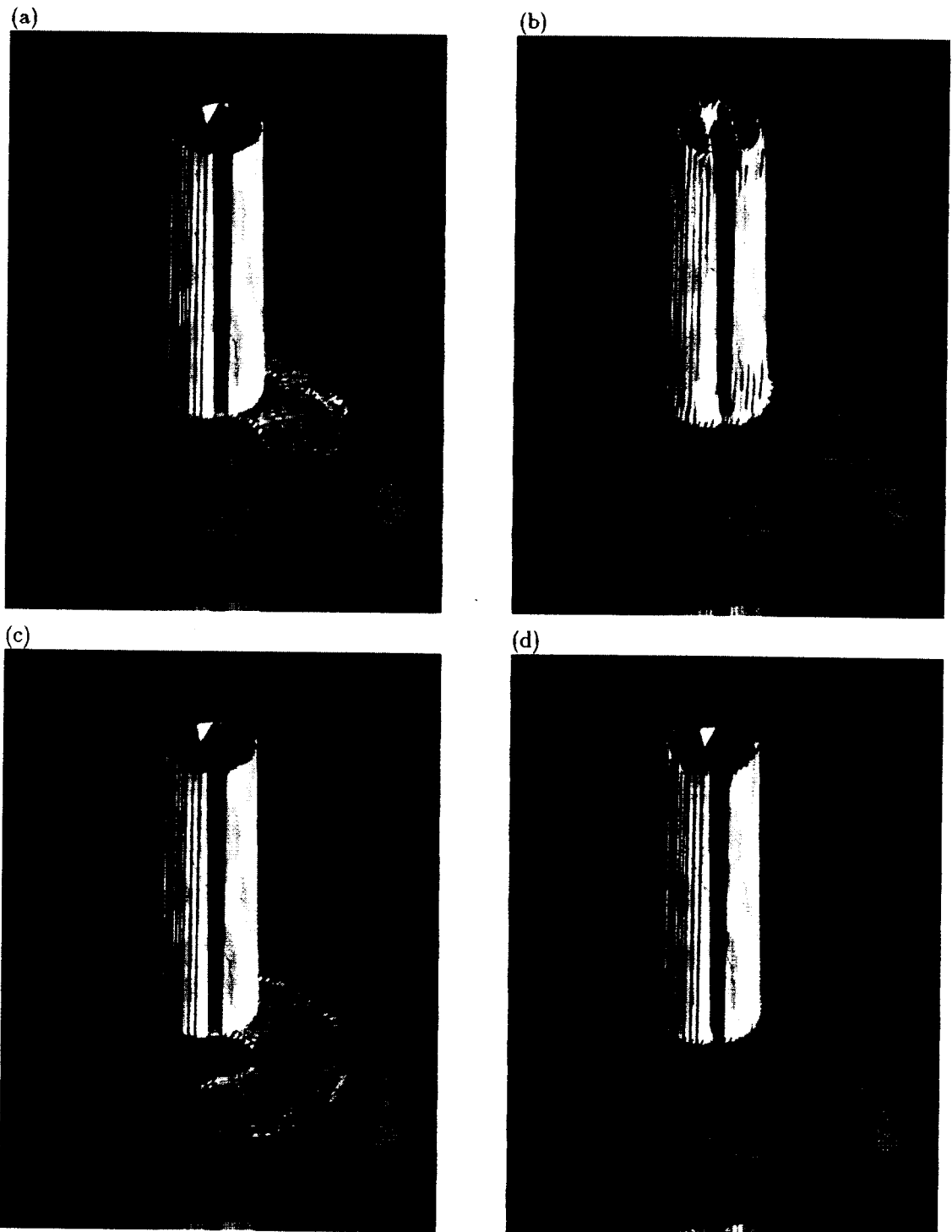


図 4.26: 非移流項に Simpson B を用いたセミ・ラグランジュ法による Zalesak 例題の検証:  $CFL = \Delta t / \Delta x$  をそれぞれ (a)  $CFL = 50\pi$ , (b)  $CFL = 66\pi$ , (c)  $CFL = 100\pi$ , (d)  $CFL = 200\pi$  で計算した。

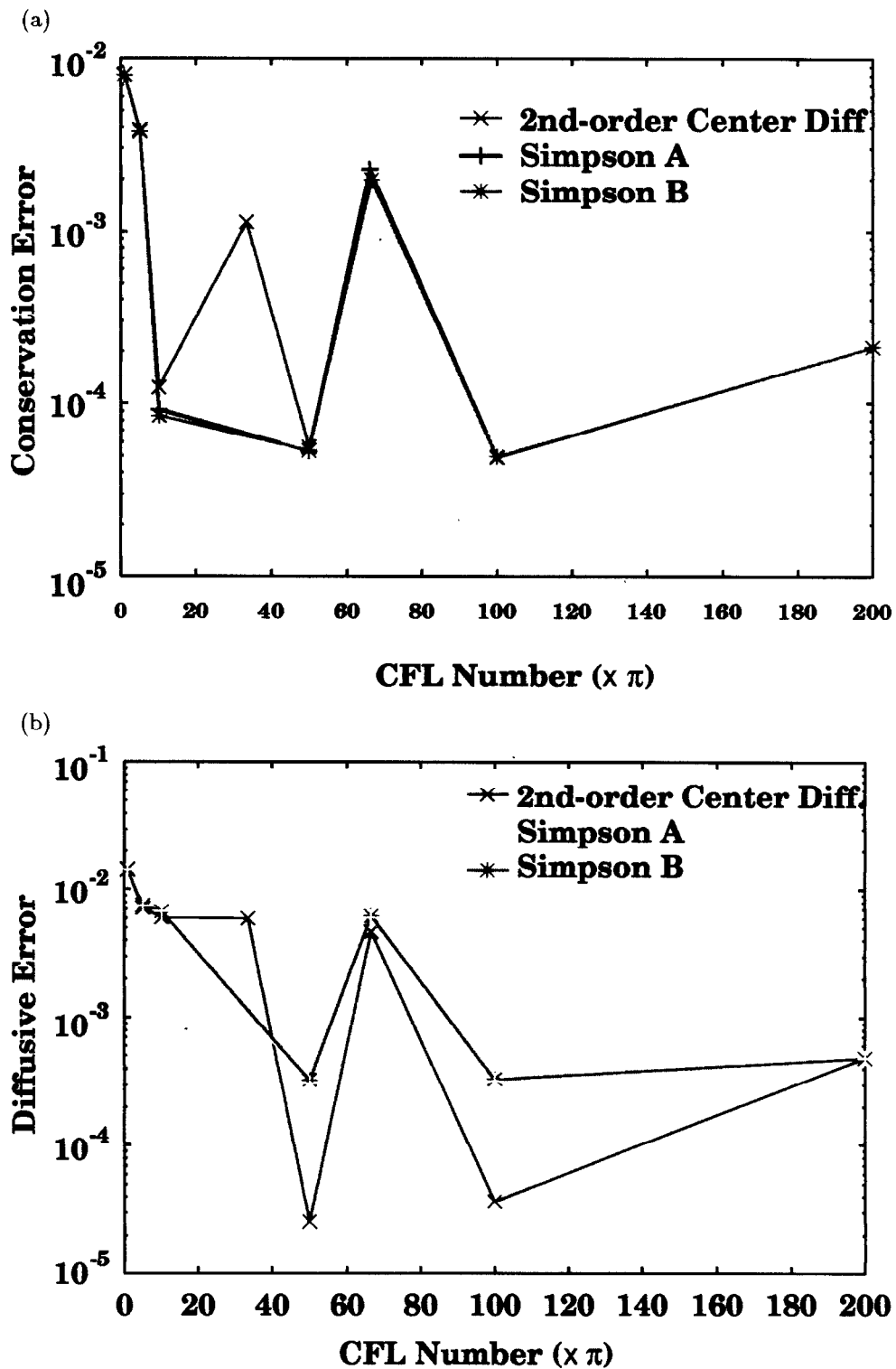


図 4.27: 2次元移流方程式に対する数値誤差:

セミ・ラグランジュ CIP 法を用い、Zalesak の例題を検証した結果から得られる (a) 保存誤差ならびに (b) 数値拡散誤差。直交座標系において初期プロファイルを各自間ステップ毎に完全に写像できる CFL 数  $50\pi$ ,  $100\pi$ ,  $200\pi$  では少ない誤差で計算できるため、値がばらつく結果となったが、CFL 数を大きくするに従い数値誤差が減少する傾向が得られた。ただし、CFL 数を大きくとりすぎると逆に誤差が増大する傾向も併せて見られる。

## 3 次元移流方程式の検証

セミ・ラグランジュ CIP 法の 3 次元問題への適用を考え、3 次元移流方程式

$$f_t = -uf_x - vf_y - wf_z \quad (4.79)$$

を扱う。 $u, v, w$  はそれぞれ移流速度の  $x, y, z$  方向成分を表す。 $x$ - $z$  平面で回転速度場を、 $y$  軸方向に並進速度を与え計算を行うため、移流速度を  $u(x, y, z) = -z, v(x, y, z) = 0.02, z(x, y, z) = x$  と与えた。計算空間の座標値を  $(-0.5 \leq x \leq 0.5, -0.5 \leq y \leq 0.5, -0.5 \leq z \leq 0.5)$  として  $50 \times 50 \times 50$  の等間隔メッシュを用いた。初期プロファイル形状を図 4.28 に示すように  $f = 1$  ( $-0.08 \leq x \leq 0.08, -0.4 \leq y \leq -0.24, -0.4 \leq z \leq -0.24$ ),  $f = 0$  (その他の点) とし、立方体で与えている。 $t=30$  まで計算した結果を図 4.29 - 図 4.32 に示す。計算は  $y$  軸周りに 4.75 周回転しながら  $y$  軸方向に並進することになる。

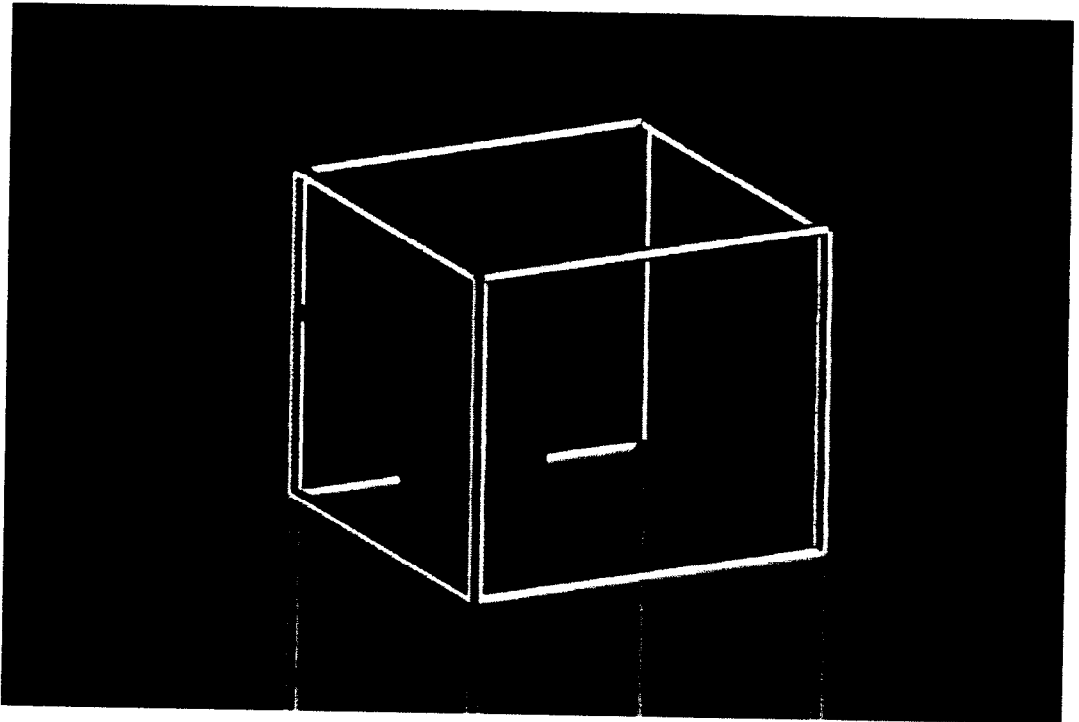


図 4.28: SL-CIP 法による 3 次元移流方程式の初期プロファイル

計算結果はそれぞれ  $f = 0.5$  の等値面を示している。CIP 法による計算結果は数値拡散によりプロファイルが丸みを帯びた形状で求められているが、セミ・ラグランジュ CIP 法で大きな CFL 数により計算することで数値拡散誤差を低減し、より元の形状を保持した計算結果が得られた。プロファイルの変形の原因となる数値拡散誤差は計算ステップ毎に生じる補間誤差により生じたものと考えられる。CIP 法での計算は、代表速度を 1 として、 $CFL = 0.5$  で計算を行っていることから、 $t = 30$  までの計算では 3000 ステップを要する。その一方でセミ・ラグランジュ CIP 法により、 $CFL = 5, 10, 30$  を用いた計算ではそれぞれ 300, 150, 50 計算ステップで  $t = 30$  までの計算を行うことが出来る。このことから、同じ計算内容であっても CFL 数を大きくとることにより低数値拡散誤差での計算が可能であることが理由付けられる。

ただし、CFL 数をより大きくとると、非移流相を 2 次精度中心差分法によって行っているにもかかわらず、数値不安定性の問題がみられた。これは、物理量ならびに空間微係数の移流相計算が高精度であったとしても移流速度が空間で変化することにより生じる微係数の時間発展式の非移流相積分精度が低下していることが問題であると考えられる。更に高 CFL 数での計算を行うためには非移流相積分の高精度化が重要になると考えられる。

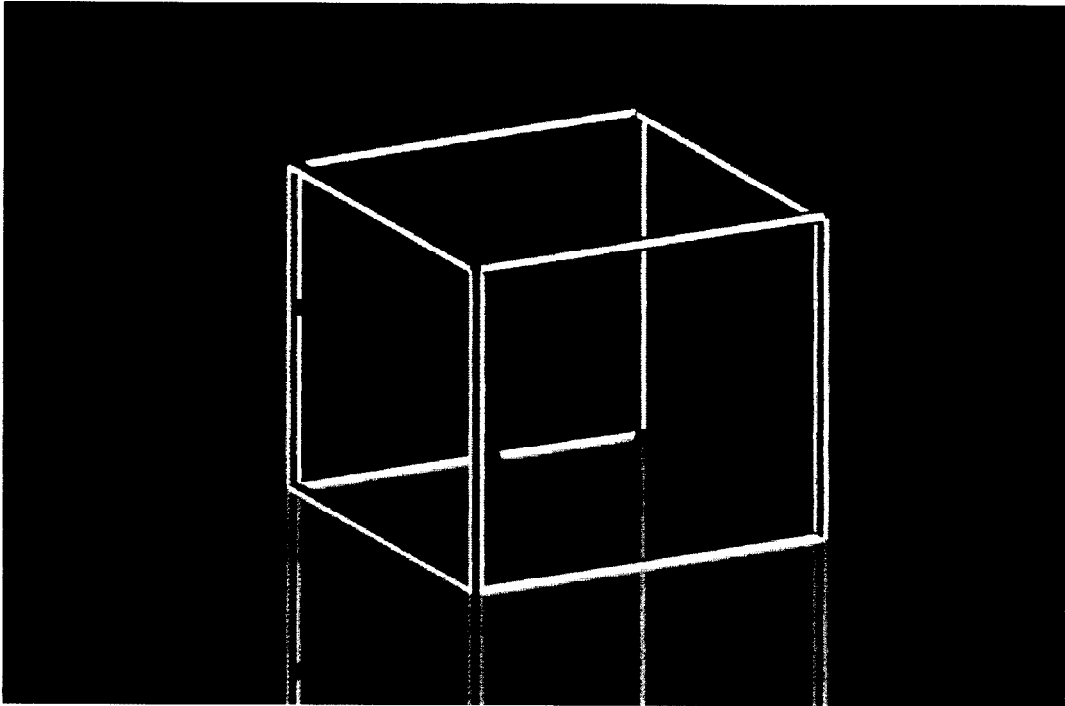


図 4.29: CIP 法による  $CFL = 0.5$  での 3 次元移流方程式の計算結果

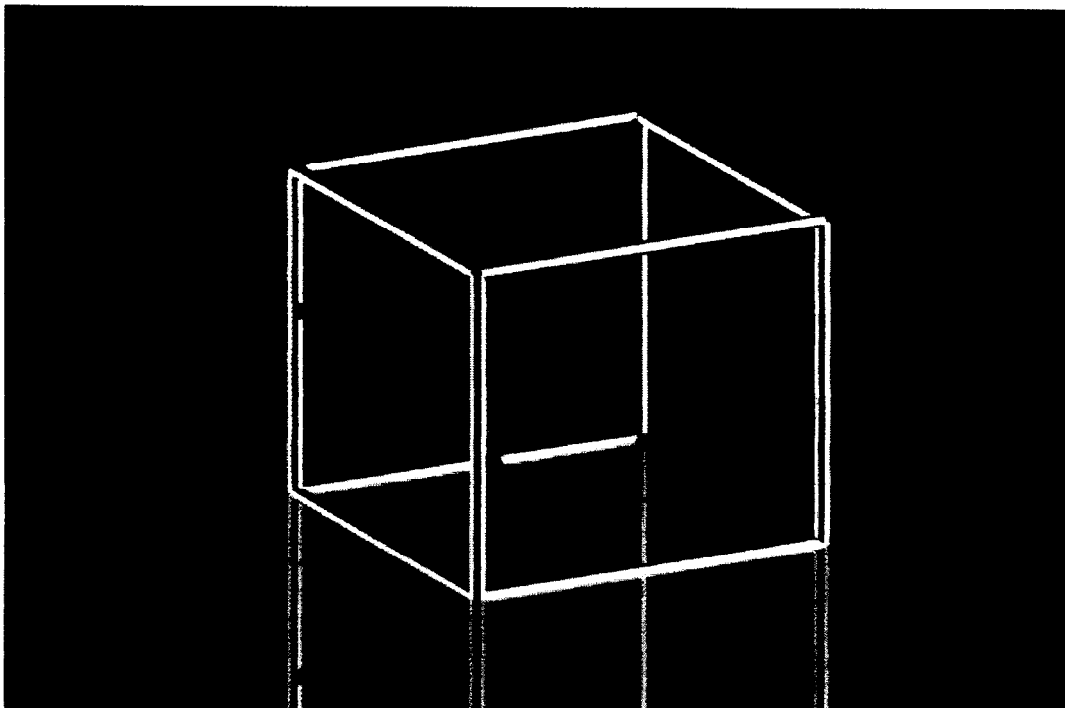


図 4.30: SL-CIP 法による  $CFL = 5$  での 3 次元移流方程式の計算結果

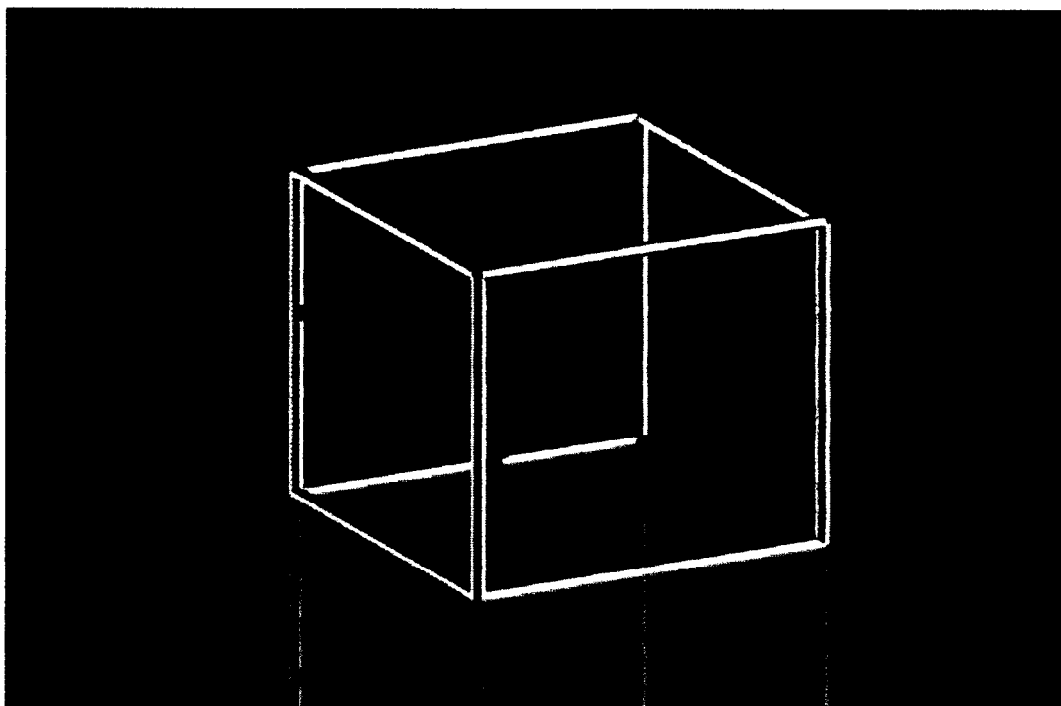


図 4.31: SL-CIP 法による  $CFL = 10$  での 3 次元移流方程式の計算結果

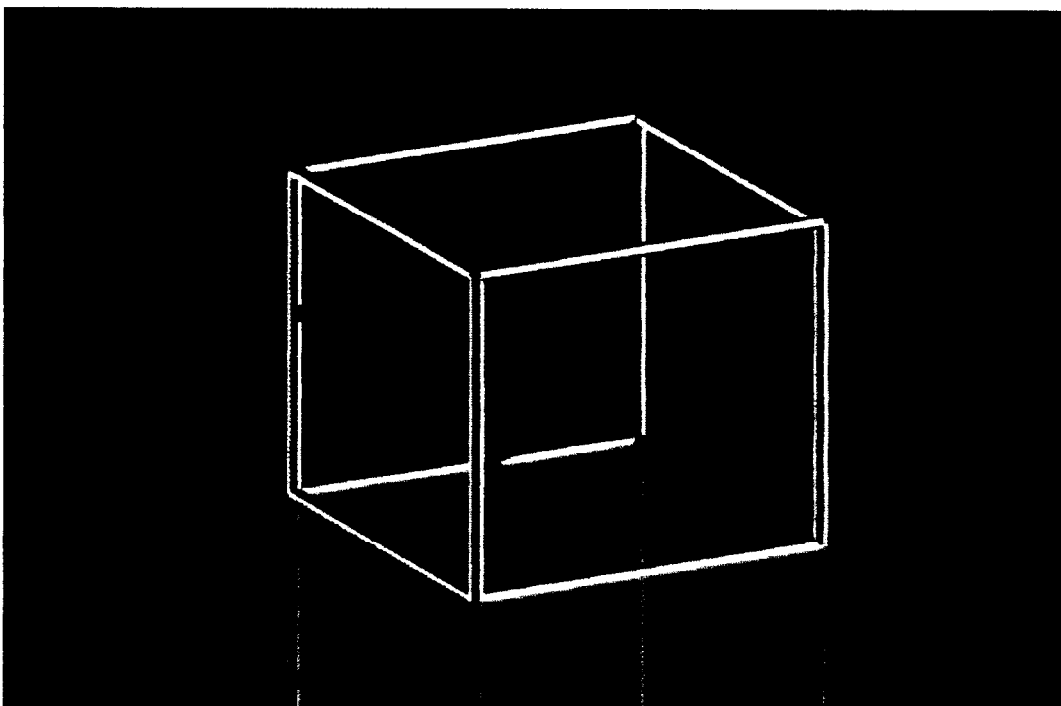


図 4.32: SL-CIP 法による  $CFL = 30$  での 3 次元移流方程式の計算結果

### 前線の発達 (Frontogenesis) 問題の検証

定常速度場中で複雑な流体構造が発達する事から第 3 章 3.3 節で扱った理想化された前線の発達 (Frontogenesis) 問題をセミ・ラグランジュ CIP 法の検証例題として扱う。

計算グリッド  $100 \times 100$  を用い、 $CFL = \Delta t / \Delta x$  を、0.5, 1.0, 2.0, 4.0 として計算した結果を図 4.34、図 4.35、図 4.36、図 4.37 にそれぞれ示す。また、解析解を図 4.33 に示した。 $CFL \leq 2$  では時間が経過し、渦が発達するにしたがって渦中心部のプロファイルにおいて数値振動が生じる結果となったがその他の点では解析解によく一致した計算結果を得ることが出来た。ただし、 $CFL > 4$  では、このプロファイル中心部における数値振動が原因となり、数値不安定性が生じる結果となった。その一方で  $CFL \leq 1$  とした場合では、数値不安定も見られず解析解と良く一致した結果を得た。

さらに、解析解が存在することを利用して解析解からの数値拡散誤差を

$$\text{Diffusive Error} = \sum_{i,j} \left| \frac{f_{\text{Analytic}}(i,j) - f_{\text{Calc}}(i,j)}{f_{\text{Analytic}}} \right| \quad (4.80)$$

により計算した結果を図 4.38 に示す。ここで、 $f_{\text{Analytic}}(i,j)$  はグリッド  $(i,j)$  上の解析解を示す。

$CFL$  数が増大させることで低数値拡散誤差で数値計算が行えることが数値計算結果から示した。ただし、 $CFL$  数が 2 以上ではプロファイル中心部で数値振動が生じることなどから、数値拡散誤差に大きな相違が見られない結果となった。セミ・ラグランジュ CIP 法は移流方程式に対しては十分な精度が得られるものの、保存方程式などの非移流相積分が重要になる問題に対しては  $CFL$  数が 4 程度までは効果が得られるがそれ以上の  $CFL$  数を用いた計算を行うことを考える場合、より高精度な非移流相積分手法の開発が必要になると考える。



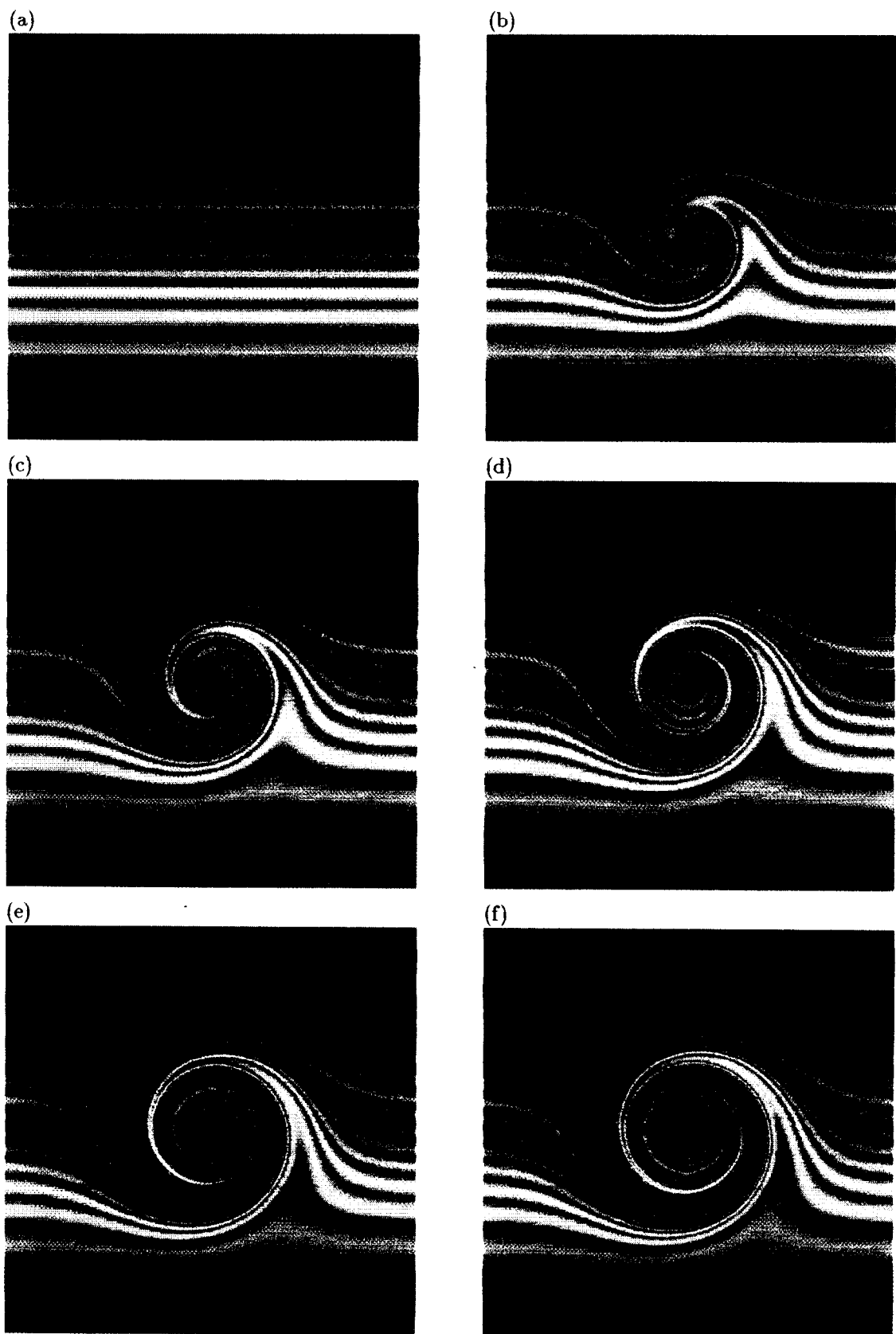


図 4.33: Frontogenesis 問題の解析解  $t = 0(a), 2(b), 4(c), 6(d), 8(e), 10(f)$  における温度分布

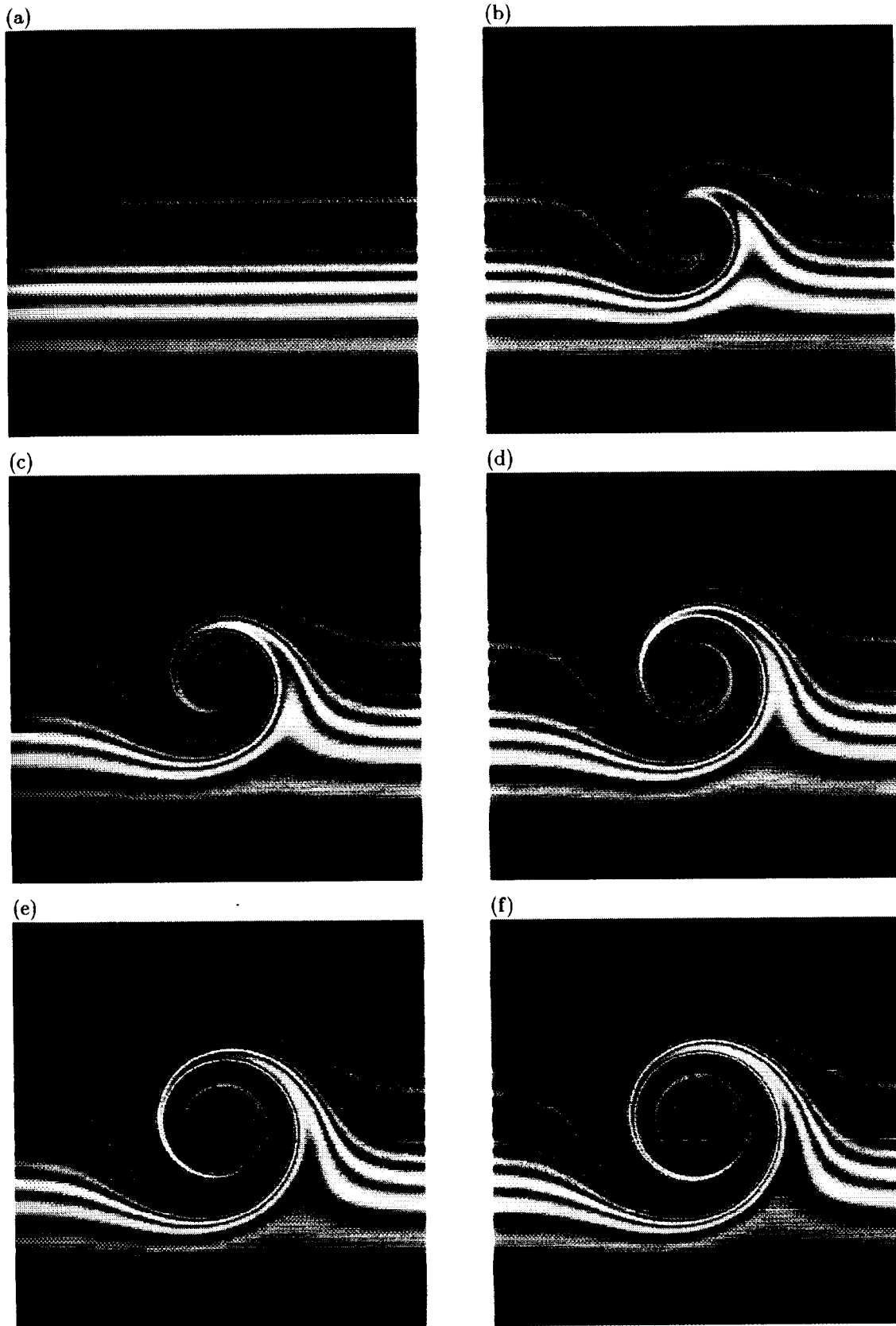


図 4.34: CFL = 0.5 でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果  $t = 0(a), 2(b), 4(c), 6(d), 8(e), 10(f)$  における温度分布

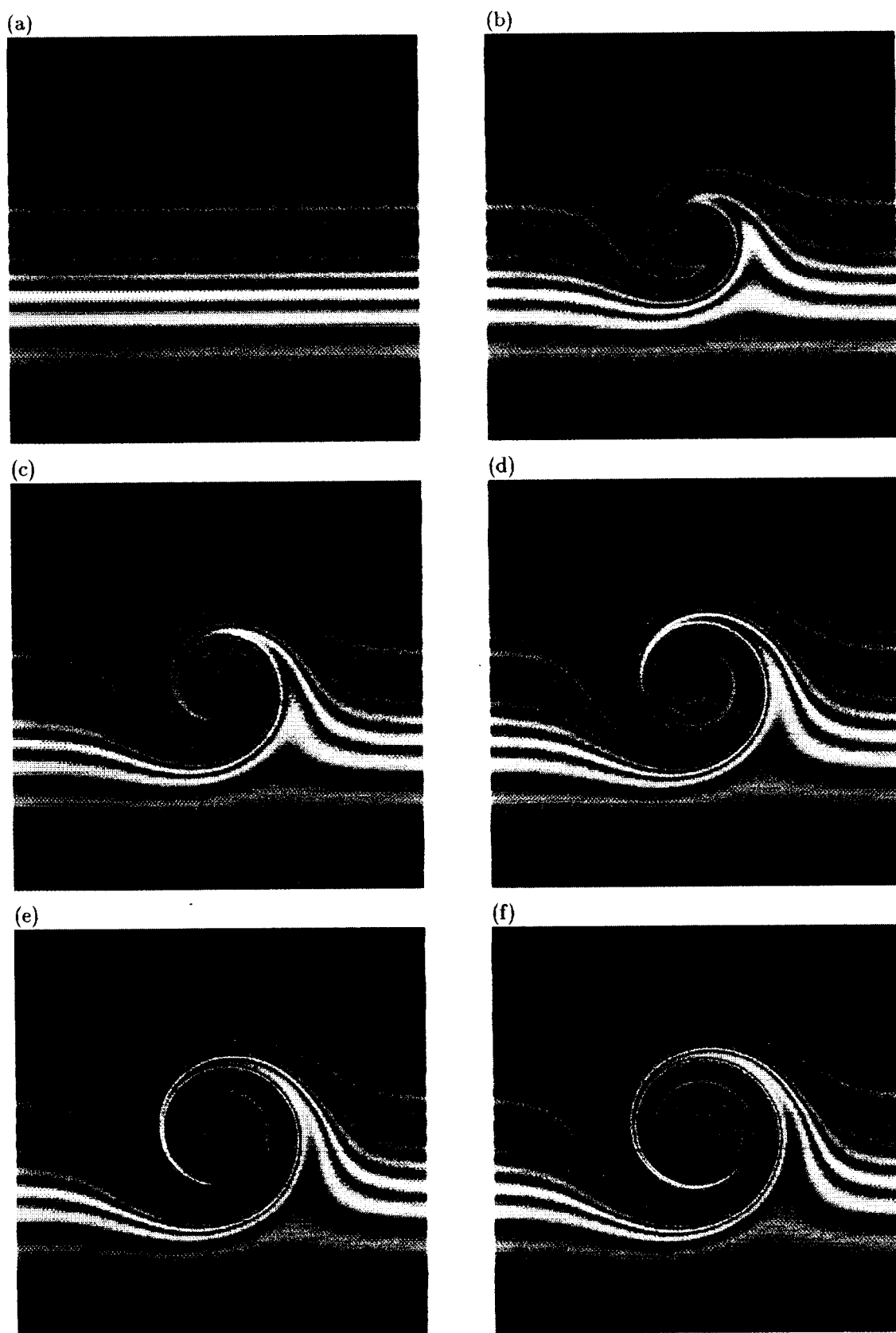


図 4.35: CFL = 1.0 でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果  $t = 0(a), 2(b), 4(c), 6(d), 8(e), 10(f)$  における温度分布

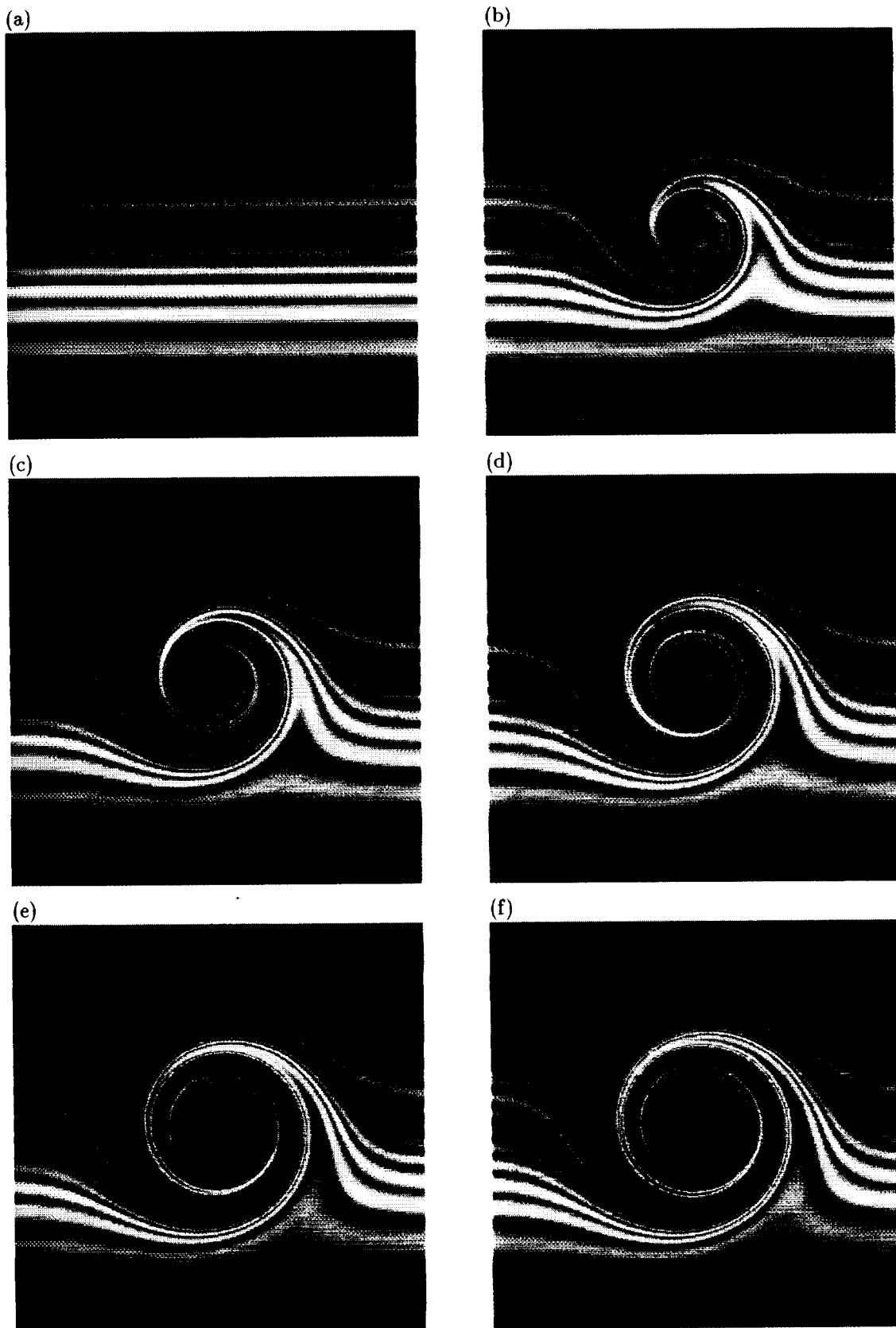


図 4.36: CFL = 2.0 でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果  $t = 0(a), 3.2(b), 6.4(c), 9.6(d), 12.8(e), 16(f)$  における温度分布

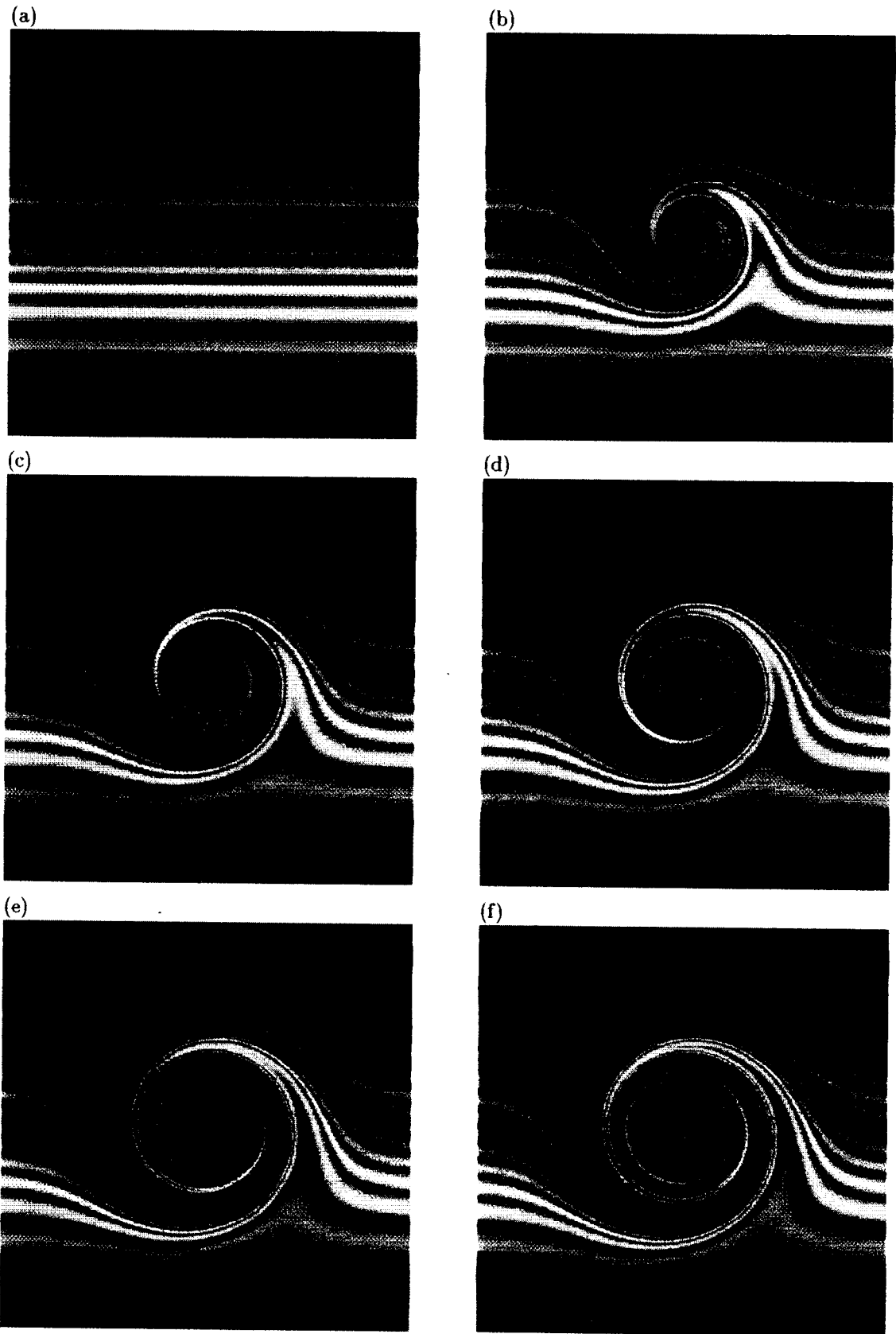


図 4.37:  $CFL = 4.0$  でのセミ・ラグランジュ CIP 法による Frontogenesis 問題の数値計算結果  $t = 0(a), 3.2(b), 6.4(c), 9.6(d), 12.8(e), 16(f)$  における温度分布

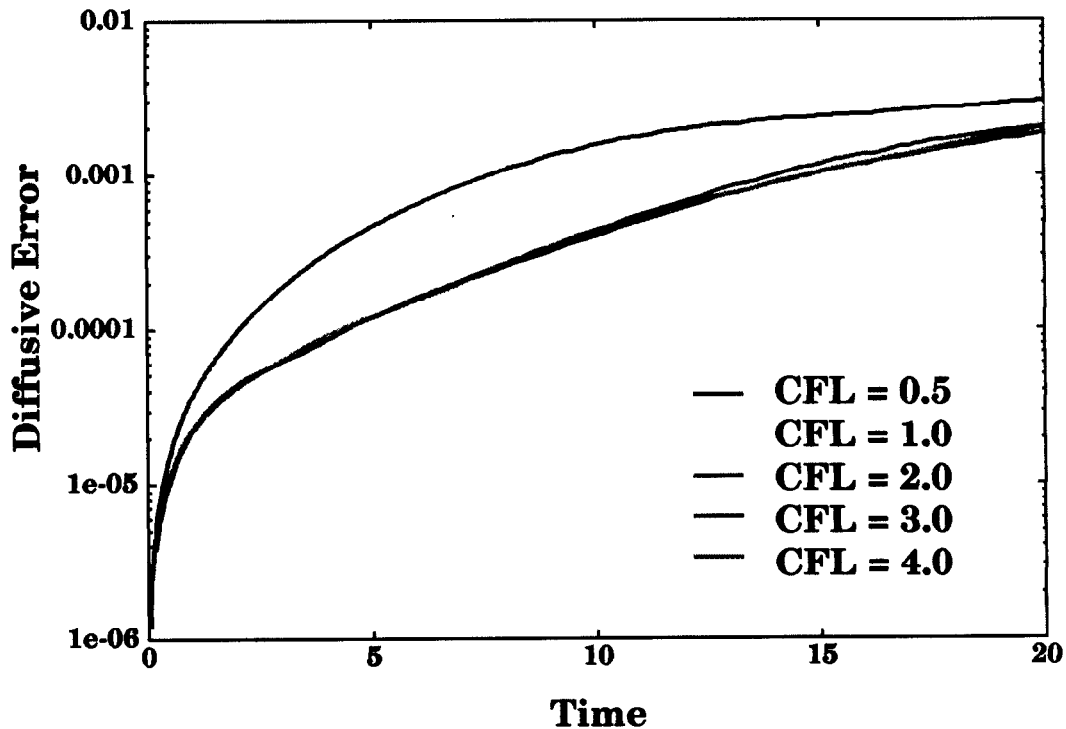


図 4.38: SL-CIP 法による Frontogenesis 問題の解析解からの数値拡散誤差

解析解を用いて CFL 数に対応した数値拡散誤差を見積もった。その結果、CFL 数を大きくとることにより数値拡散誤差を低減できている様子がわかる。ただし、CFL 数が 2 以上では数値拡散誤差に大きな相違が見られない結果となった。

## 2 次元正方キャビティー流れの検討

セミ・ラグランジュ手法は定常速度場を前提としたスキームであるが、現実的には速度場が時間的に変動しない問題は非常に少ない。そこで、速度場の時間変動が比較的緩やかな非圧縮性流体問題の一つである 2 次元 Driven Cavity Flow 問題を用いてセミ・ラグランジュ CIP 法の流体問題への適用性を検討する。非圧縮性流体問題を選択したもう一つの理由として、音速の問題が挙げられる。圧縮性流体問題を扱うためには必ず音速による制限が生じる。セミ・ラグランジュ手法により音速以上の速度で情報を移流させることが可能であるが、現実的に衝撃波を越えて伝搬する流体现象はありえない。つまり、圧縮性流体問題に対してセミ・ラグランジュ手法を適用した場合、音速によって  $\Delta t$  が制限されるため、その有効性を十分に発揮することが出来ない。また、圧縮性流体問題は時間的に速度変化が激しいため、セミ・ラグランジュ手法では扱いにくいといった問題点がある。

## 非移流相積分の高精度化

これまでの例題から  $CFL > 1$  においてセミ・ラグランジュ CIP 法を用いることで定常速度場では高精度移流計算が可能であることを示した。しかし流体問題を扱う場合、現象に対して移流相が寄与する影響と同様に非移流相が現象に対して及ぼす影響も無視することは出来ない。これまでのセミ・ラグランジュ CIP 法で用いた 2 次精度中心差分法による非移流相積分手法は式 (A.13) や、式 (A.43) ~ (A.45) に示すようにオイラー的な積分を行っていると言える。大きな CFL 数 ( $> 1$ ) を用いた場合、3 次元移流方程式でも見られたようにオイラー的手法で非移流相積分を行うことは、計算精度の低下を導く。

セミ・ラグランジュ CIP 法は仮想時間ステップ ( $\lambda_s$ ) 毎の移流原点位置を 4 次 Runge-Kutta 手法を用いて精度良く求めていることから、移流経路に沿った非移流相積分が可能であると考えられる。しかし、シンプソン則を用いた非移流相積分では数値不安定性の原因となることをこれまでの単純な例題から明らかにした。本節では CIP 法の補間関数を用いて移流経路に沿って物理量ならびにその空間微係数を補間し、高精度積分手法により非移流相積分の高精度化を考える。2 次元移流問題での空間補間にはグリッド内で高精度 Remapping を行える CIP 法の Type-C を用いた。

流体問題での非移流相積分を高精度化するにあたり、計算負荷増大の問題がまず挙げられる。その根拠を以下に示す。

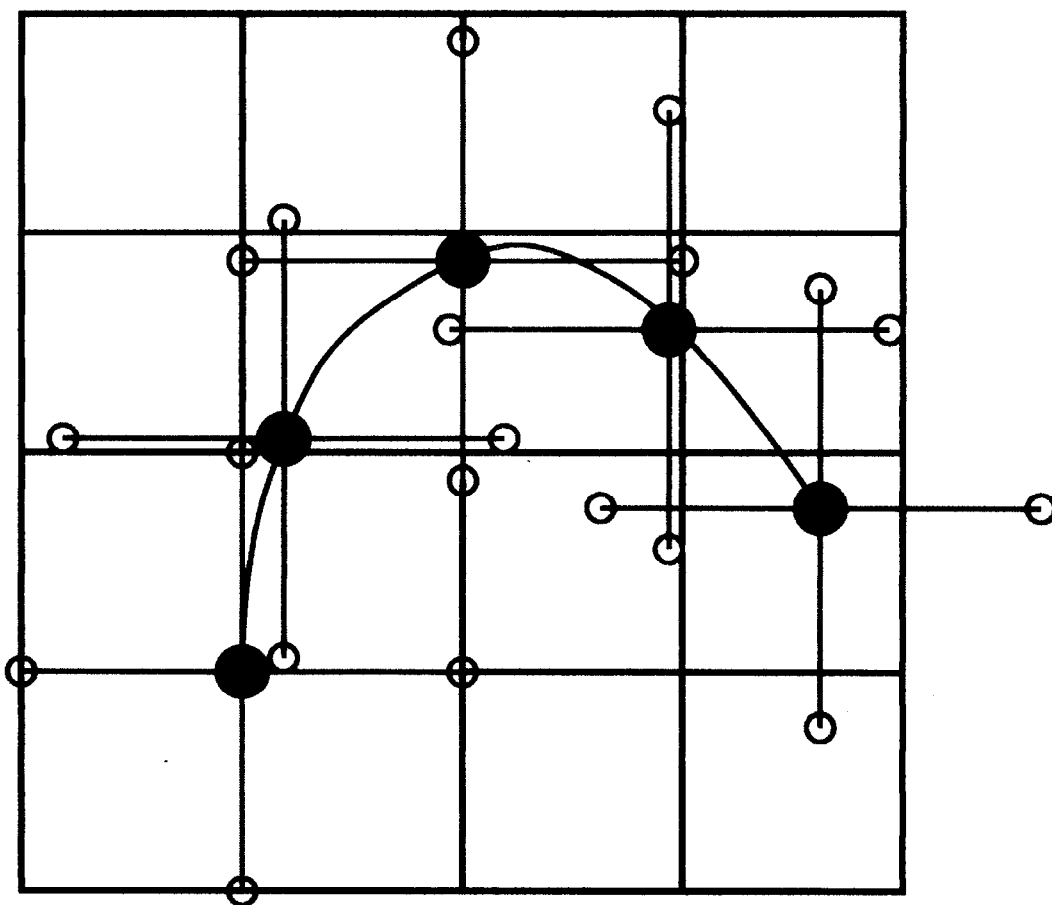


図 4.39: 長い移流経路に沿った粘性項計算

長い移流経路に沿って粘性項の積分計算を行うためには多数の物理量を移流経路に沿って補間する必要がある。

例として移流経路に沿った粘性項 ( $\Delta f = \partial^2 f / \partial x^2 + \partial^2 f / \partial y^2$ ) 計算を図 4.39 を元に考える。移流原点計算のために計算時間幅  $\Delta t$  を 4 分割している場合 ( $\Delta t = \sum_{s=1}^4 \lambda_s$ )、それぞれの仮想時間ステップにおける移流原点での粘性項計算を行うことを考える。2 次精度中心差分法を用いて粘性項計算を行う場合、各仮想時間ステップにおける移流原点毎にグリッド上に無い 4 点での移流速度を見積もらなければならない。セミ・ラグランジュ CIP 法では 2 次元問題における Remapping 計算に CIP Type-C を用いているため、少なくとも CIP 法 Type-C の 5 倍の計算負荷を要することになる。この他にも移流原点位置計算のために 4 次精度 Runge-Kutta 手法を用いているため時間後退法を用いている CIP 法の 3 倍程度計算

負荷が増大するなどの問題があるため、より高速な粘性項計算手法を用いることが望ましい。そこで、精度は幾分低下するが、グリッド点毎に2階微分値を計算し、重みづけにより移流原点での粘性項を見積もる手法を検討する。

図 4.40 に示すように各仮想時間ステップに対応する移流原点を含むセルの各頂点で粘性項計算を行う。この計算では必要な情報が全てグリッド上に存在するため、先のように Type-C による Remapping を多用しなくても簡単に計算が可能となる。各頂点での粘性項を用い、面積による重みづけで移流原点位置での粘性項を見積もることで先の手法に比べ計算負荷を大幅に低減できる。各位流原点毎に同様の手法で粘性項を見積もることで様々な積分手法により高精度積分が可能と考える。

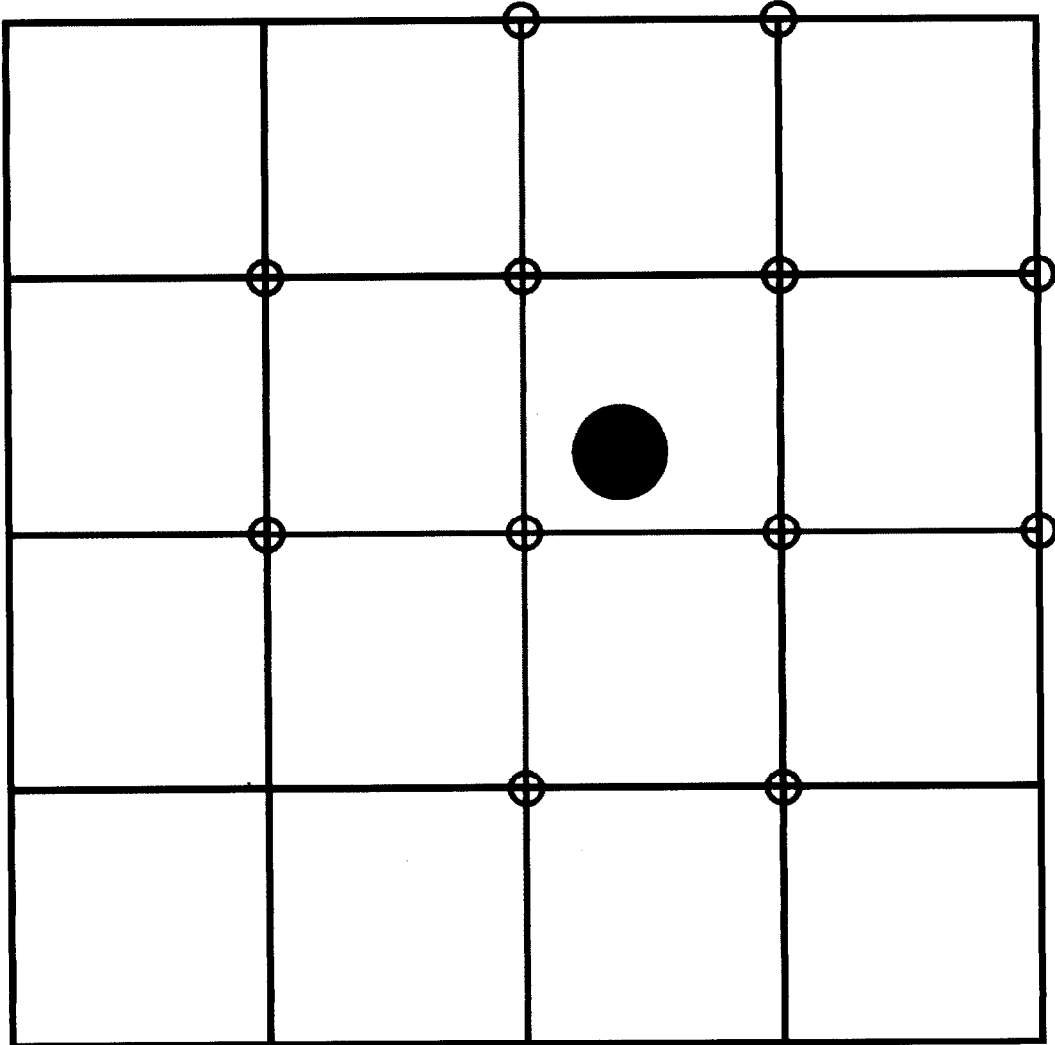


図 4.40: 重みづけによる非移流相積分の高速化

グリッド上で粘性項計算を行い、面積による重み付けにより計算対象点での粘性項を概算することで計算負荷を大きく減らすことが出来る。

これまでは積分手法として2次精度中心差分法を用いたオイラー的積分手法、一般的なシンプソン則 (Simpson A)、微係数を用いたシンプソン則 (Simpson B) の3手法について検討を行ったが、非圧縮性流体問題を扱った場合圧力に関する時間微分項を求めることが比較的困難であることと、補間関数から得られる微係数を用いて非移流項計算を行うことで数値不安定性が生じやすくなることから、今回は Simpson



B については考慮しない。ただし、それらの手法に加えて単純な積分手法である区分求積法と台形則を用いた。以下でそれぞれについて簡単に説明する。

オイラー的積分法

$$\int RHS.dt = RHS.\Delta t \quad (4.81)$$

区分求積法

$$\int RHS.dt = \sum_s RHS.\lambda \quad (4.82)$$

台形積分

$$\int RHS.dt = \sum_s \frac{(RHS.(a) + RHS.(b)) \lambda}{2} \quad (4.83)$$

シンプソン則

$$\int RHS.dt = \sum_s \frac{(RHS.(a) + 4 \times RHS.(\frac{a+b}{2}) + RHS.(b)) \lambda}{6} \quad (4.84)$$

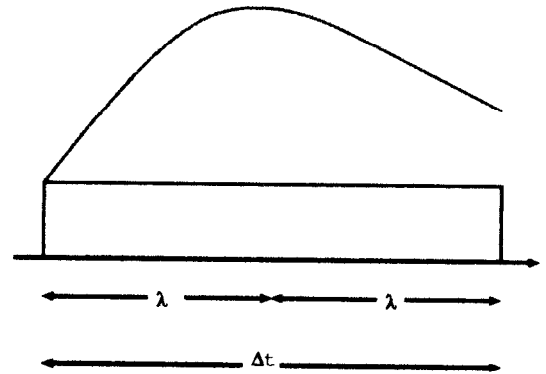


図 4.41: オイラー的積分法の概念図

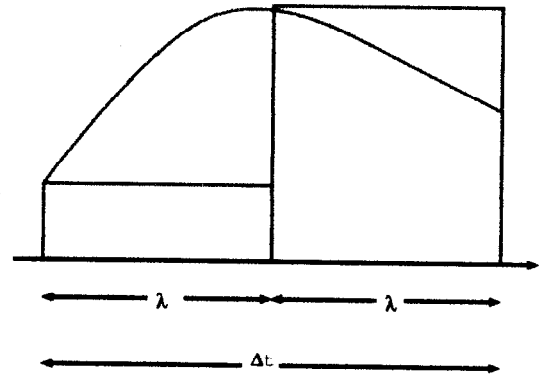


図 4.42: 区分求積法の概念図

それぞれの手法を比較した場合、後者になるに従い一般的に精度の高い積分が可能であると言える。計算負荷は、オイラー的積分を行った場合、時間ステップ毎に 1 点の情報しか用いないのに比べ、区分求積法で、各仮想時間ステップ λ 毎に 1 点の情報、台形積分で各仮想時間毎に 2 点の情報、シンプソン則では仮想時間毎に 3 点の情報が必要になる事が分かる。計算の手続きから、各仮想時間ステップ毎に始点と終点を必ず計算することから、台形積分までは計算負荷の差がほとんど生じないが、シンプソン則を用いる場合は他の手法に比べ負荷の高い計算になると予想される。

上記計算手法による 2 次元正方キャビティ流れの検討を行った結果、区分求積法を用いることで安定に、かつオイラー的積分手法に比べて高精度数値解を求めることが出来たことから、以下では非移流相計算を区分求積法により行っている。

以上をふまえ、2 次元 Driven-Cavity Flow 問題へのアプローチ方法を説明する。計算空間  $0 \leq x \leq 1, 0 \leq y \leq 1$  において、 $65 \times 65$  の等間隔メッシュを使用した。全壁面上でノンスリップ条件とし、上壁は速度

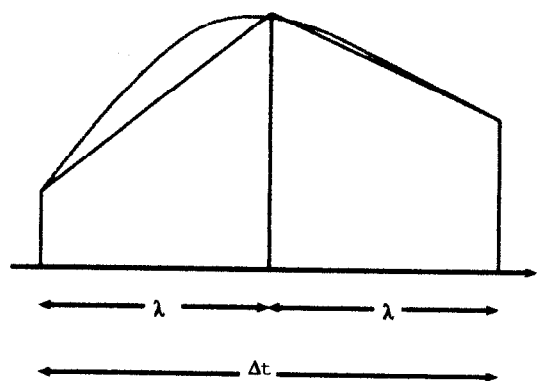


図 4.43: 台形積分の概念図

$u = 1$  で移動するものとする。非圧縮性流体問題を解くための手法として SMAC 手法を採り入れ、その計算手続きに含まれるポアソン方程式の解法にはマルチグリッド手法を適用し、高速化を図った。

計算結果の検討は、Ghia らによる結果 [38] と比較するとともに、CIP 法により  $CFL = 0.1$  で計算した結果とも併せて行う。それぞれの流線図はセミ・ラグランジュ手法による結果から得られたものである。扱うレイノルズ数は Ghia らの結果との検討を前提としていることから、1000, 3200, 5000 の 3 通りを行った。

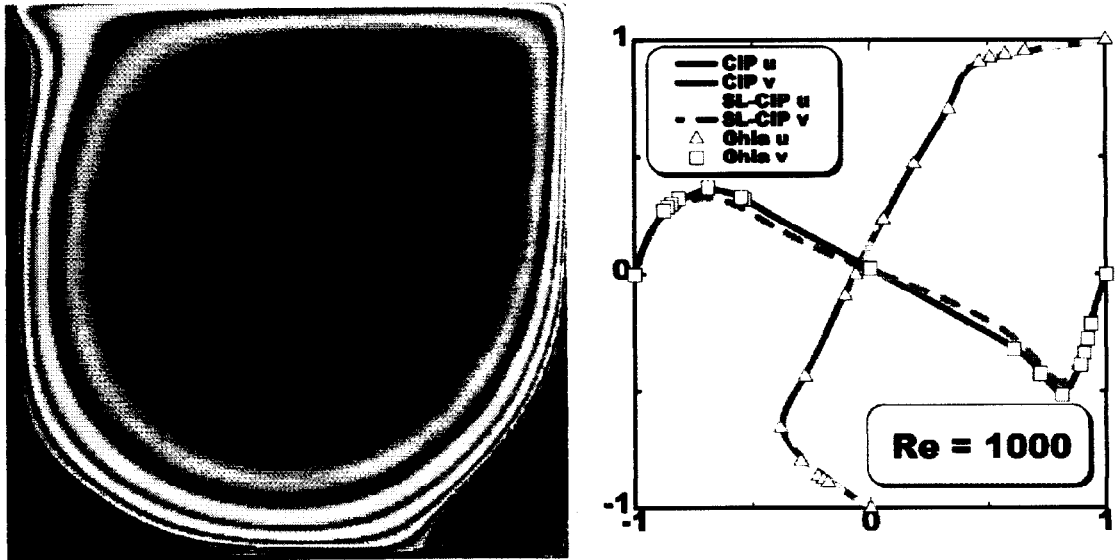


図 4.44:  $Re = 1000$ 、 $CFL$  数 2 による Driven Cavity Flow 問題の数値解

計算空間の中心線に沿った速度場は Ghia らによる計算結果と比較的によく一致した結果が得られた。

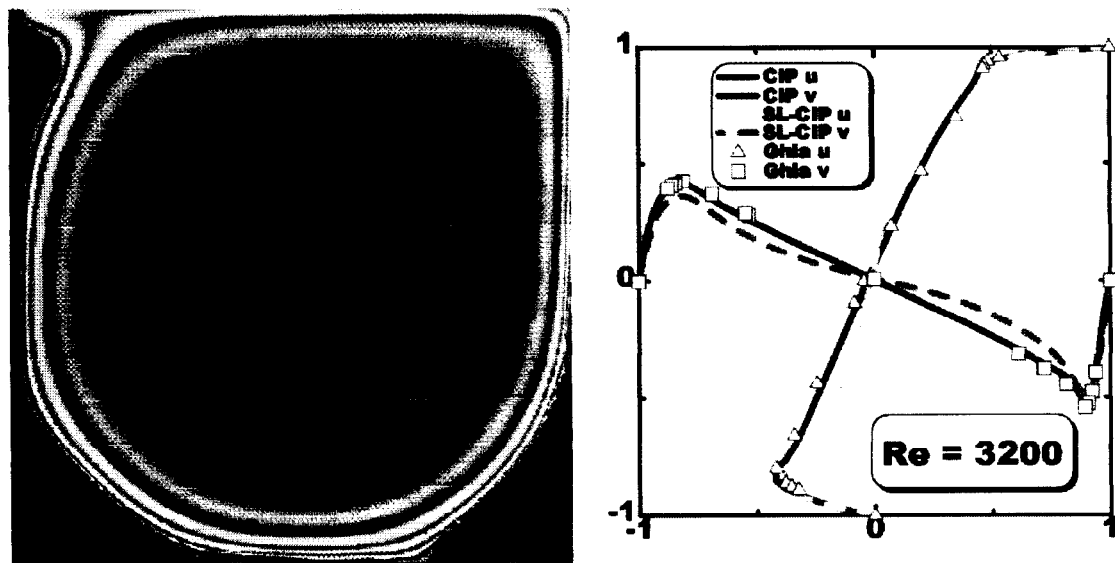


図 4.45:  $Re = 3200$ 、 $CFL$  数 2 による Driven Cavity Flow 問題の数値解

中心線に沿った移流速度は Ghia らによる計算結果と比較して若干小さい値となった。しかし、プロファイルの中心位置はよく一致した結果を得た。

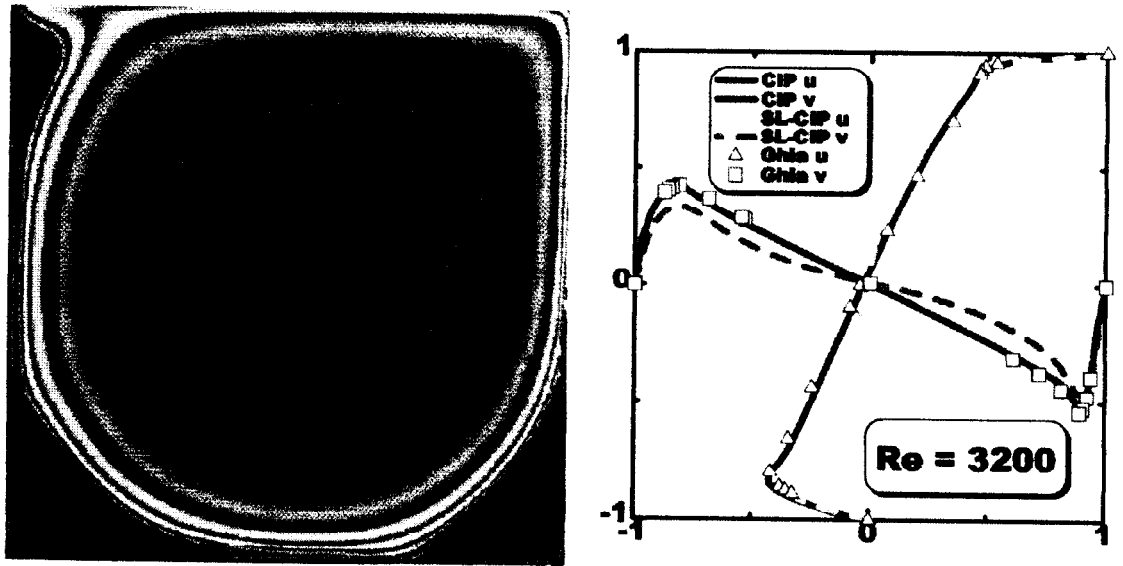


図 4.46:  $Re = 3200$ 、CFL 数 3 による Driven Cavity Flow 問題の数値解  
 中心線に沿った移流速度は Ghia ちによる計算結果と比較して若干小さい値となった。しかし、プロファイルの中心位置はよく一致した結果を得た。

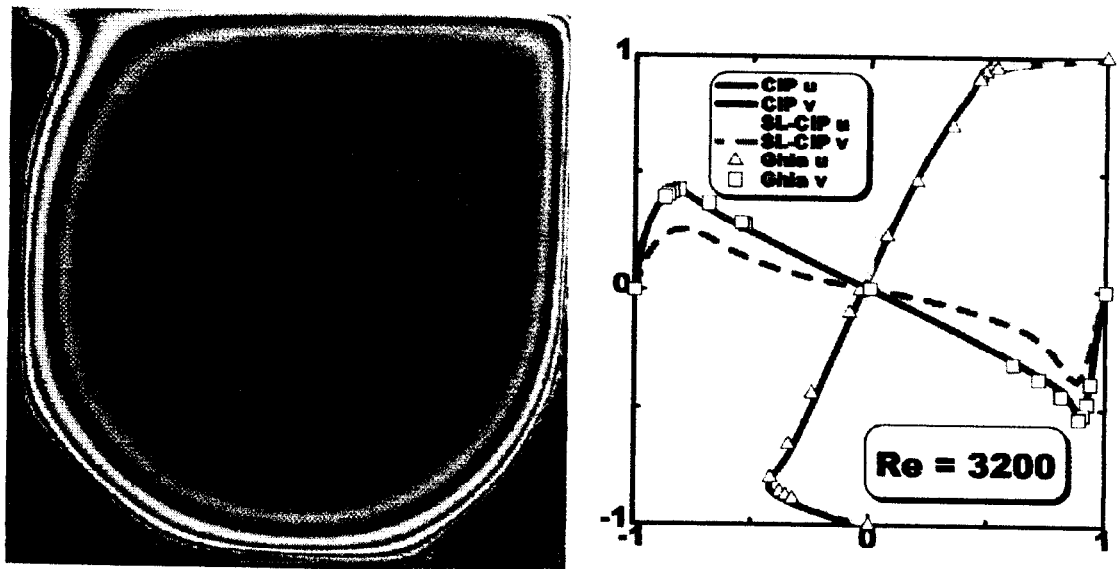


図 4.47:  $Re = 3200$ 、CFL 数 5 による Driven Cavity Flow 問題の数値解  
 CFL 数を増大させるに従って次第に中心線に沿った移流速度の見積もりに誤差が生じている。

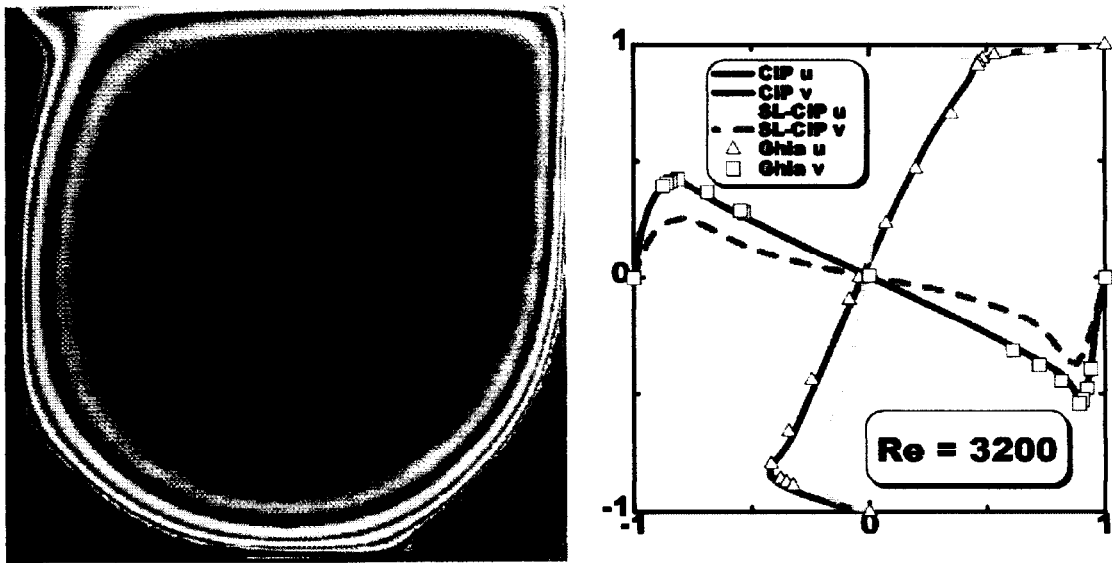


図 4.48:  $Re = 3200$ 、CFL 数 7 による Driven Cavity Flow 問題の数値解  
CFL 数を増大させるに従って次第に中心線に沿った移流速度の見積もりに誤差が生じている。

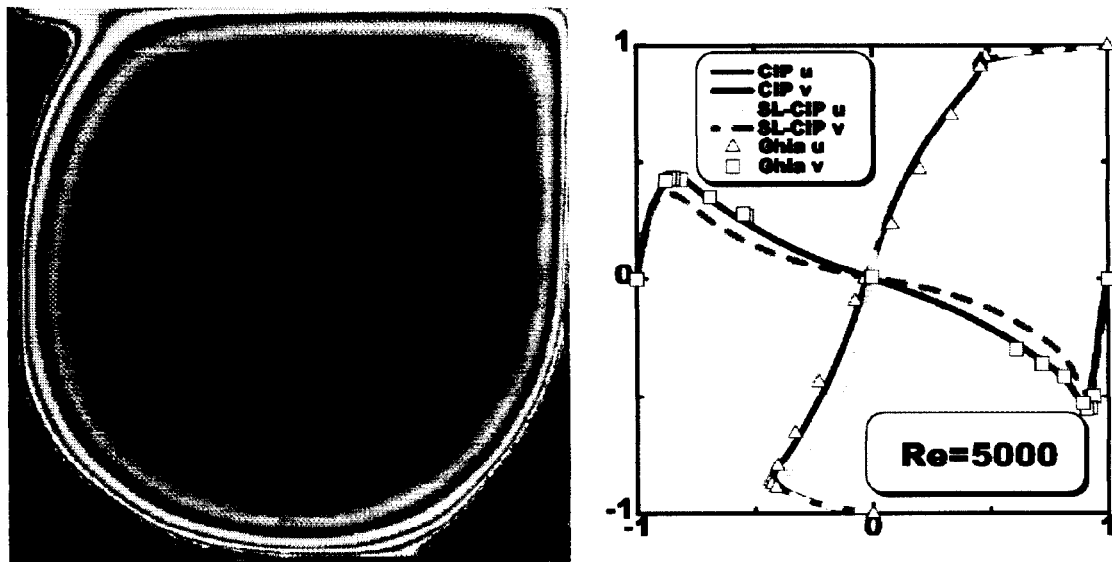


図 4.49:  $Re = 5000$ 、CFL 数 2 による Driven Cavity Flow 問題の数値解  
CFL 数 2 での計算では比較的良好な数値計算結果を示している。しかし、Ghia らによる数値計算結果からは若干ずれた結果となっている。

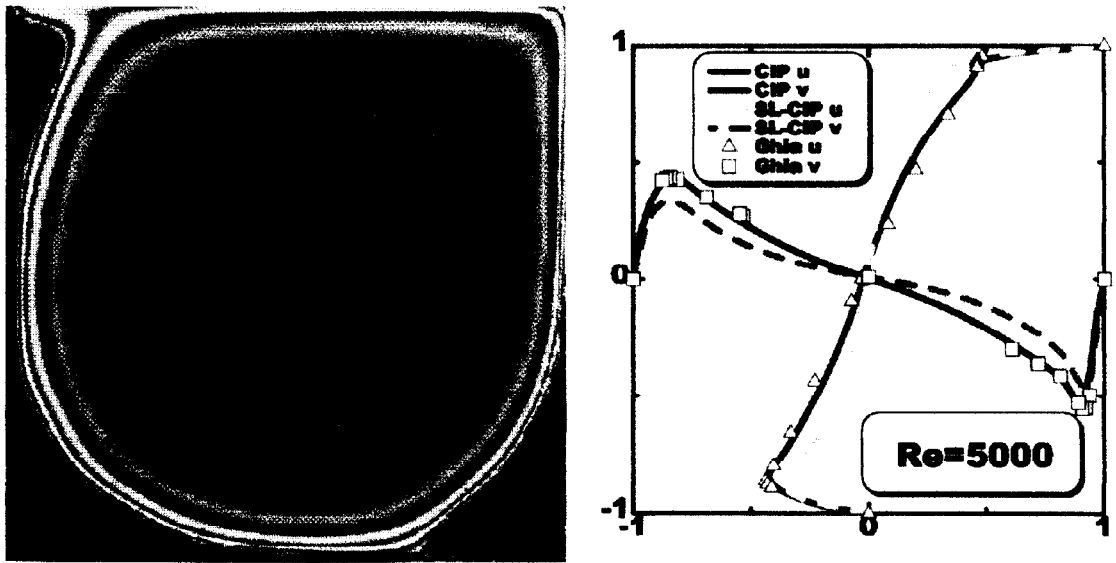


図 4.50:  $Re = 5000$ 、CFL 数 3 による Driven Cavity Flow 問題の数値解  
 CFL 数 3 では、右下の 2 次系渦が定常的に安定にならない現象が見られるようになった。また、中心線に沿った移流速度も Ghia らによって示された結果より若干小さな値を示している。

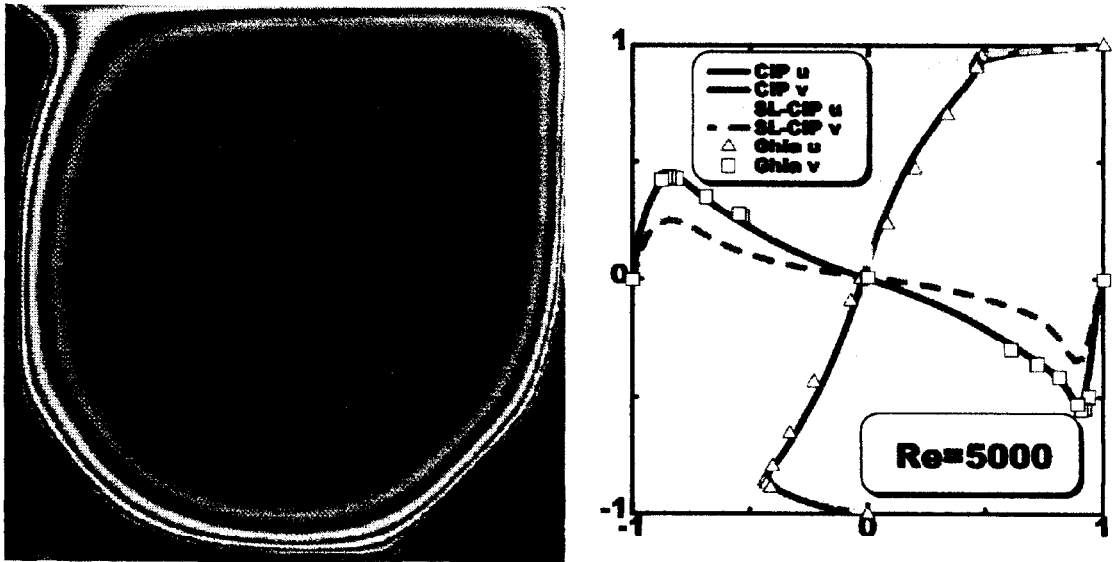


図 4.51:  $Re = 5000$ 、CFL 数 8 による Driven Cavity Flow 問題の数値解  
 レイノルズ数 5000 では CFL 数 8 まで安定に計算できたものの、中心線に沿った移流速度は Ghia らによる結果から大きくずれる結果となった。

レイノルズ数を大きくとるに従い、流速の時間変化が緩やかになるため、大きな CFL 数をとれる傾向が見られた。また、レイノルズ数 3200 での結果から、CFL 数がある値以上に大きく ( $\approx 5$ ) した場合、全線発達問題でも見られたように精度向上が期待されず、Driven Cavity Flow 問題では得られる定常解における速度プロファイルが大きく変化しない結果が得られることをしめした。これは、Runge-Kutta 法の分解性能の限界によるものではなく、非移流相積分精度の限界によるものであることが、計算サブステップ数の分割数の変化に対して精度向上が期待されなかったことから明らかである。また、レイノルズ数 3200, CFL=5, 7 の Stream Contour から、右下隅の 2 次的渦の計算精度が低下する結果が見られる。これは大きな CFL 数をとることで細かな現象を記述できなくなる問題点を明らかにしている。しかし、CIP 法に比べ、CFL 数を 3 としたセミ・ラグランジュ CIP 法による計算では、定常状態を達成するまでに要する計算時間を 16 倍高速化することに成功した。

#### 4.2.4 まとめ

より大きな CFL 数 ( $> 1$ ) で流体方程式を解くことを目的としてセミ・ラグランジュ CIP 法を開発した。保存誤差、数値拡散誤差を一樣速度場での 1 次元移流方程式で検証した結果から、同 CFL 数では CIP 法と同じ数値拡散誤差、保存誤差での計算が可能であることを示した。セミ・ラグランジュ CIP 法では大きな CFL 数を用いて安定に計算できることから、目的計算時間までの計算ステップ数を CIP 法に比べて減らすことが可能であり、そのような状況下では CIP 法に比べて少ない数値誤差で計算できることを併せて示した。非一樣速度場における 1 次元保存方程式では非移流項成分が発生するため、非移流項成分の数値積分手法として CIP 法で用いられている 2 次精度中心差分法、シンプソン則 (Simpson A)、微係数を用いたシンプソン則 (Simpson B) を用いて検証を行った。Simpson B を用いて非移流相の積分を行った場合、2 次精度中心差分法に比べ、少ない保存誤差で計算できることを示したが、大きな CFL 数 ( $\Delta t/\Delta x$ ) での計算では Simpson B は数値振動を示す結果が得られた。その一方で 2 次精度中心差分法により非移流項計算を行うことで、 $\Delta t/\Delta x$  が 100 を越えてもプロファイルが減衰するものの、安定な計算が可能であることを示した。

移流原点位置計算精度の検討を目的として時間後退差分法、4 次精度 Runge-Kutta 法を用いて数値計算を行ったところ、 $\Delta u \Delta t/\Delta x > 1$  の条件下で移流原点計算を時間後退差分法により行った場合、風上の情報が風下の情報を越えて伝搬されるため数値不安定性が生じることを示した。より計算精度の高い 4 次精度 Runge-Kutta 法により移流原点を求めることで数値安定性の改善を行うことが可能であった。しかし、非移流相積分の精度に問題があるため、得られたプロファイルは解析解と一致せず、2 次精度中心差分法以外の手法に対しては数値安定性の問題がより顕著に現れる結果となった。高次精度非移流項積分に関しては課題が残されているものの、空間非一樣速度場に於ける移流原点計算に於いて高次スキームを用いることがセミ・ラグランジュ法を用いる上で必要であることを示した。

一般的な問題に適用するためセミ・ラグランジュ CIP 法を 2 次元に拡張し、Zalesak の例題を非移流項の計算手法として 1 次元問題で用いた 3 手法について検証し、それぞれの方法で大きい CFL 数で安定な数値計算が可能であることを示した。前線の発達 (Frontogenesis) 問題を扱うことで、複雑な移流経路に対しても高精度数値計算が可能であることを示した。また、2 次元問題に対しても非移流項の計算手法により数値拡散誤差の低減、数値安定性の改善が可能であることを示した。

セミ・ラグランジュ法を用いることで計算ステップ数を減少させることが可能であることから計算ステップ毎に生じる数値拡散誤差の低減を図ることが出来たが、移流原点を 4 次精度 Runge-Kutta 法で求めているため従来の時間後退法を用いる手法に比べ計算負荷が増大する欠点もあるため、移流原点の計算アルゴリズムについて今後更なる改良が必要であると考えた。計算精度が低下するものの、移流原点位置計算時に用いるサブタイムステップ数を減少させることで、セミ・ラグランジュ CIP 手法は CIP 法よりもはるかに高速な計算が可能であるが、2 次元移流問題の一つである Zalesak 例題による検証を行ったところ、サブタイムステップ時間  $\lambda$  を大きくするにしたがって、移流精度が大幅に低減する結果となった。このことから、移流経路計算精度の重要性を確認した。

流体問題への適用を考え、非圧縮性流体問題の一つである 2 次元正方キャビティー流れを扱った。セミ・ラグランジュ手法は移流速度の時間変動が小さいことを仮定し、大きな  $\Delta t$  を用いて移流計算を行うが、2 次元正方キャビティー流れでは過渡期に速度場が大きく変動するため、大きな CFL 数を扱うことは困難であった。また、 $\Delta t$  を大きくとることにより小さな構造の渦を適切に捕らえることが出来なくなることから定常状態における移流速度分布に関しても若干の誤差が確認された。その一方で CFL 数を 3 としたセミ・ラグランジュ CIP 法による数値計算は CIP 法による数値計算時間に対し 16 倍の高速度に成功した。

2 次元キャビティー流れ問題では細やかな渦や速度の時間発展がセミ・ラグランジュ手法では扱えない程度の変化を見せるため、誤差の大きい結果となったが、本節で開発したセミ・ラグランジュ CIP 法は定常速度場における問題に対して高精度数値計算結果を示したことから、従来セミ・ラグランジュ手法が計算対象としている気象問題や、ジェット流などの問題に対しては十分な効果を発揮すると考える。

### 4.3 まとめ

非圧縮性流体問題を数値計算で扱う場合、圧力波が無限小の時間で伝搬することから計算ステップ毎にポアソン方程式を解く必要がある。そのため、圧縮性流体問題の解法に比べ計算時間が増大することから、非圧縮性流体問題の高速度解法を試みた。

#### ポアソン方程式解法の高速度化

IDO 法は移流問題に限らず様々な偏微分方程式の数値計算に用いることが可能なことから、圧縮性流体計算で重要な楕円型偏微分方程式の一つであるポアソン方程式の数値計算に適用し、精度検証を行った。対照検証として 2 次精度中心差分法による数値解と比較したところ、今回用いた例題に対しては同程度の精度でポアソン方程式の数値解が必要な場合、2 次精度中心差分法では少なくとも IDO 法で用いている 10 倍程度のメッシュ数を必要とすることを示した。IDO 法で 2 次元ポアソン方程式を扱う場合、ディリクレ境界条件、ノイマン境界条件を問わず空間精度に対して  $\Delta x^4$  の精度を示すことを明らかにした。3 次元ポアソン方程式に対する検証結果からも  $\Delta x^4$  精度を得た。高解像度計算手法を用いることにより、少ない格子点数で計算が可能であることを示した。

ポアソン方程式の高速度解法を達成するため、有限差分法で緩和計算を高速化することで知られているマルチグリッド法の IDO 法への適用検証を行ったところ、2 次元ポアソン方程式の解法では SOR 法と比較しても約 36 倍高速に同精度までの収束を可能にした。このことにより、これまでは有限差分法に対する緩和計算に有効性を検証されていたマルチグリッド法が、物理量に加え微係数も従属変数として扱う IDO 法を用いた緩和計算に対しても有効であることを明らかにした。マルチグリッド法を用いるにあたり、記憶容量の最適化を行ったとしても通常の 2 倍程度の記憶容量を必要とするが、単一 CPU で計算時間が数十倍から百倍程度短縮できることを考えると、本論文で開発した IDO 法による 2 次元ポアソン方程式の解法は高速かつ高精度解を得ることができるところからスカラー型計算機に於いて非常に有効な手段であると言える。

マルチグリッド法とは異なるポアソン方程式収束計算の高速度化を目的としたベクトル計算機もしくは並列計算機による並列計算を行うことを前提としてレッドブラックアルゴリズムを適用した IDO 法によるポアソン方程式の収束性を数値的に検証した。その結果、レッドブラックアルゴリズムを適用した場合でも自然な計算手順での SOR 法を用いた数値計算結果と同等の数値誤差でレッドブラックアルゴリズムを適用した IDO 法によるポアソン方程式の緩和計算が可能であることを示した。このことから、有限差分法の様に収束対象が 1 変数の場合については理論的にその収束性が保証されているレッドブラックアルゴリズムが、IDO 法のように 4 変数がそれぞれ依存関係にあるケースについても適用可能であることを数値的に明らかにした。また、3 次元 IDO ポアソン方程式に対してもレッドブラックアルゴリズムを適用し、その収束性を確認した。更に、MPI ライブラリを用いた並列計算を行い、 $51 \times 51 \times 51$  グリッド点数によるポアソン方程式の解法において少ない格子点数にもかかわらず高いスケラビリティを得ることに成功した。

### セミ・ラグランジュ CIP 法

非圧縮性流体問題を高速に解く手法として大きな時間刻幅  $\Delta t$  を利用するセミ・ラグランジュ手法の導入を考える。CIP 法はその計算手順からセミ・ラグランジュ手法として扱われることがあったが、代表的な特徴の一つである CFL 数により  $\Delta t$  が制限されないといった性質を備えていなかった。そこで、各計算格子点に対する移流原点位置を 4 次精度 Runge-Kutta 法により計算し、移流原点位置において適切に補間関数を形成する手法を採り入れたセミ・ラグランジュ CIP 法 (SL-CIP) を提案した。SL-CIP 法の検証結果から、1 次元、2 次元移流方程式に対しては CFL 数が 100 を越えても、3 次元計算では CFL 数が 30 程度でも安定に高精度計算を行うことが可能であることを示した。また、非圧縮性流体問題における速度場の時間発展が比較的緩やかであることに着目し、2 次元正方キャピティ流れ問題を検証した結果 CFL 数を 3 程度とした場合、CIP 法に比べて 16 倍程度の高速計算を行うことが可能であることを示した。



## 第 5 章

# 陰的 IDO 法による CFL 数に対する安定化

局所補間微分オペレータ (Interpolated Differential Operator; IDO) 法を用いることで双曲型偏微分方程式、楕円型偏微分方程式、放物型偏微分方程式を同じ手続きで離散化することが可能であり、少ない計算格子点数で高精度数値シミュレーションが可能である反面、多次元問題を扱う場合には有限差分法 (Finite Difference Method; FDM) や、CIP 法と比較して大きな CFL ( $> 0.3$ ) での計算が不安定になる場合がある [39]。この問題を克服することを目的として有限差分法で数値安定化のために用いられる陰解法 (Implicit Method) を IDO 法に適用するとともに、その精度検証を行うことで有用性について検証する。

陰的 IDO 法の時間発展は時間前進を陽解法と同様にテーラー展開に基づいて行う。ただし、陽解法ではテーラー展開の時間基準を  $t^n; t^{n+1} = t^n + \Delta t$  に置いていたが、陰解法では  $t^{n+1}; t^n = t^{n+1} - \Delta t$  を基準とした以下の展開式を用いる。

$$f^n = f^{n+1} - f_t^{n+1} \Delta t + \frac{1}{2} f_{tt}^{n+1} \Delta t^2 - \frac{1}{6} f_{ttt}^{n+1} \Delta t^3 \dots \quad (5.1)$$

$$f_x^n = f_x^{n+1} - f_{tx}^{n+1} \Delta t + \frac{1}{2} f_{ttx}^{n+1} \Delta t^2 - \frac{1}{6} f_{tttx}^{n+1} \Delta t^3 \dots \quad (5.2)$$

ここで、時間に関するテーラー展開の条件は、陽解法と同様に空間補間関数の多項式の次数により制限される。補間関数として用いる多項式の最高次数が 3 次である場合、時間に関するテーラー展開は最大で 3 次までとなる。陽解法 IDO 法では移流相計算に対する空間精度、時間精度の向上を考えた場合風上補間に 5 次補間関数を用いることで達成されたことから、陰解法においても同様に時間精度、空間精度が要求される場合は風上補間として 5 次補間式を用いて計算を行うことを考える。ただし、風上 5 次補間式を用いた場合陰解法では各時間ステップにおける収束までの計算コストが増大することが考えられるので、以下では、1 次元移流方程式などの単純な方程式に対しては風上 5 次補間による陰解法化も行うが、他の例題では風上 3 次補間を用いて陰解法化を行う。

### 5.1 陰解法スキーム

陰解法スキームを用いて数値計算を行うと一般的に  $CFL > 1$  であっても安定に計算できるようになる。その理由を 1 次精度風上差分法を用いて以下に説明する。

擾乱  $\phi_j^n = \delta \phi^n e^{ikj\Delta x}$  (正弦波) を仮定し、von Neumann の手法を用いて安定解析を行う。  $i$  は虚数単位、  $k$  を波数、  $j$  を格子点番号とした。1 次精度風上差分法で陰解法を用いると離散式は、以下のようになる。

$$\begin{aligned} \phi_j^{n+1} &= \phi_j^n - u \frac{\phi_j^{n+1} - \phi_{j-1}^{n+1}}{\Delta x} \Delta t \\ &= \phi_j^n - C(\phi_j^{n+1} - \phi_{j-1}^{n+1}) \end{aligned} \quad (5.3)$$

$\phi_j^{n+1}$  を解く時点で  $\phi_{j-1}^{n+1}$  は既知であるから、

$$\phi_j^{n+1} = \frac{\phi_j^n + C\phi_{j-1}^{n+1}}{1+C} \quad (5.4)$$

と表される。ただし、 $C = \text{CFL}$  数と略記した。擾乱の式から、

$$\frac{\delta\phi_j^{n+1}}{\delta\phi_j^n} = \frac{1}{1+C - Ce^{-ik\Delta x}} \quad (5.5)$$

の関係が得られる。

$$e^{-ik\Delta x} = \cos(k\Delta x) - i\sin(k\Delta x) \quad (5.6)$$

より、

$$\frac{\delta\phi_j^{n+1}}{\delta\phi_j^n} = \frac{1}{1+C - C\cos(k\Delta x) + iC\sin(k\Delta x)} \quad (5.7)$$

が得られ、複素数のノルムを考慮すると、

$$\left| \frac{\delta\phi_j^{n+1}}{\delta\phi_j^n} \right| = \frac{1}{\sqrt{(1+C - C\cos k\Delta x)^2 + C^2 \sin^2 k\Delta x}} \quad (5.8)$$

と書き下すことが出来る。つまり、CFL を大きくした場合 (CFL 数  $> 1$ ) でも、計算ステップ毎に解が減衰するものの、振幅が増幅されて数値不安定性を示すようなことは無い。

同様の数値安定性解析を IDO 法について行うことは補間関数形成のために微係数を従属変数として用いることから困難である。そのため、陰解法化を行うことで必ずしも安定化が図れるかどうかは不明である。CIP 法に関しては陰解法化を行ったことで安定化可能であることが報告されていることから [40, 41]、IDO 法についても定常一様速度場における移流方程式に関しては陰解法化により安定度を増すことが出来ると考える [39]。以下で線形移流方程式、非線形移流方程式、圧縮性流体問題などの様々な偏微分方程式 (Partial Differential Equation; PDE) に対して陰解法化を行いその有効性を数値的に検証する。

## 5.2 反復法

IDO 法を用いた陰解法スキームは以下の関係式で表すことが出来る。

$$\mathbf{A}\mathbf{F}^{n+1} = \mathbf{B}\mathbf{F}^n. \quad (5.9)$$

ここで、ベクトル  $\mathbf{F}$  は、対象となる物理量とその空間微係数からなり、行列  $\mathbf{A}$  は時間  $t = t^{n+1}$  における物理量とその空間微係数の係数、行列  $\mathbf{B}$  は時間  $t = t^n$  における物理量とその空間微係数の係数を表す。上記ベクトル方程式は  $\mathbf{A}$  の逆行列を求めることで時間前進を行うことが出来る  $\mathbf{F}^{n+1} = \mathbf{A}^{-1}\mathbf{B}\mathbf{F}^n$ 。行列  $\mathbf{A}$  の大きさは計算グリッド点数に比例するが、それに加えて IDO 法では空間微係数も扱うことから、多次元化を進めるに従って、行列  $\mathbf{A}$  は非常に大きな行列となる。特定の偏微分方程式に対して有限差分法による解法のために逆行列  $\mathbf{A}^{-1}$  を高速に解く有効な手段が多く提案されているが [42, 43]、新しいスキームである IDO 法のための逆行列算出に対する最適な手法はまだ提案されておらず、また本節ではあらゆる偏微分方程式の陰解法化を対象としているため行列形状が特定できないこともあり、Gauss-Seidel 法による反復解法で上記ベクトル方程式を扱うことにする。他の反復法として良く知られている Jacobi 法や SOR 法などを用いて収束速度や収束安定性の比較検討を行ったが、移流方程式や保存方程式などの単一偏微分方程式により表記される現象に対しては Gauss-Seidel 法がその中でも最も安定に収束した。流体計算のように数種類の偏微分方程式により記述される方程式系ではその特性により SOR 法を用いた方が収束が安定する場合も見られた。

また、1 次元移流方程式や保存方程式などは、非線形項を含まないことから直接行列を解くことが可能であるが、非線形項を含む流体方程式に対しては直接解法は適用することは困難である。非線形項の扱いを可能にすることも反復解法を用いる目的の一つである。

### 5.3 計算結果

#### 1 次元移流方程式の検証

陰的 IDO 法の基礎的な精度を検証するため、最も簡単な偏微分方程式として 1 次元移流方程式

$$f_t + uf_x = 0 \quad (5.10)$$

に対して陰的 IDO 法を適用し、数値拡散誤差を見積もる。ここで、添字  $t, x$  はそれぞれ時間、空間に関する偏微分を示す;  $f_t = \partial f / \partial t, f_x = \partial f / \partial x$ 。また、移流速度  $u$  は空間、時間において定常、一様とする ( $u_t = 0, u_x = 0$ )。この仮定から、物理量  $f$  の時間の 2 次微係数、3 次微係数等を元の PDE を用いて空間微分により表現すると以下ようになる;  $f_{tt} = u^2 f_{xx}, f_{ttx} = u^2 f_{xxx}, f_{ttt} = -u^3 f_{xxx} \dots$ 。これらの関係式を用い、陰的解法における時間前進の式 (式 (5.1), 式 (5.2)) に含まれる時間微係数を空間微係数で書き換えると以下ようになる。

$$f^{n+1} = f^n - u f_x^{n+1} \Delta t - \frac{u^2}{2} f_{xx}^{n+1} \Delta t^2 - \frac{u^3}{6} f_{xxx}^{n+1} \Delta t^3 \quad (5.11)$$

$$f_x^{n+1} = f_x^n - u f_{xx}^{n+1} \Delta t - \frac{u^2}{2} f_{xxx}^{n+1} \Delta t^2 \quad (5.12)$$

空間の補間関数に 3 次風上エルミート補間式 (式 (A.2)) を用いると、式 (5.11), 式 (5.12) は移流速度  $u_i$  が正である場合、以下のように離散化される。

$$\begin{aligned} f_i^{n+1} &= f_i^n - u \Delta t f_{x,i}^{n+1} \\ &\quad - \frac{(u \Delta t)^2}{2} \left( -6 \frac{f_i^{n+1} - f_{i-1}^{n+1}}{\Delta x^2} + 2 \frac{f_{x,i-1}^{n+1} + 2 f_{x,i}^{n+1}}{\Delta x} \right) \\ &\quad - \frac{(u \Delta t)^3}{6} \left( -12 \frac{f_i^{n+1} - f_{i-1}^{n+1}}{\Delta x^3} + 6 \frac{f_{x,i-1}^{n+1} + f_{x,i}^{n+1}}{\Delta x^2} \right) \end{aligned} \quad (5.13)$$

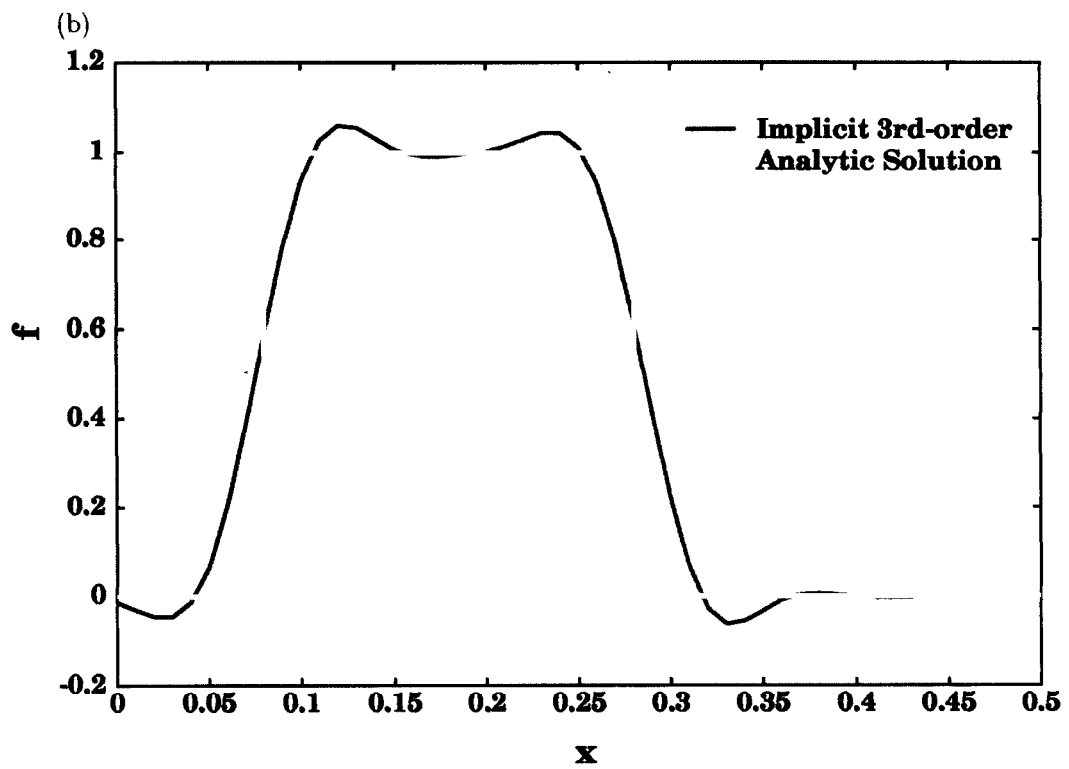
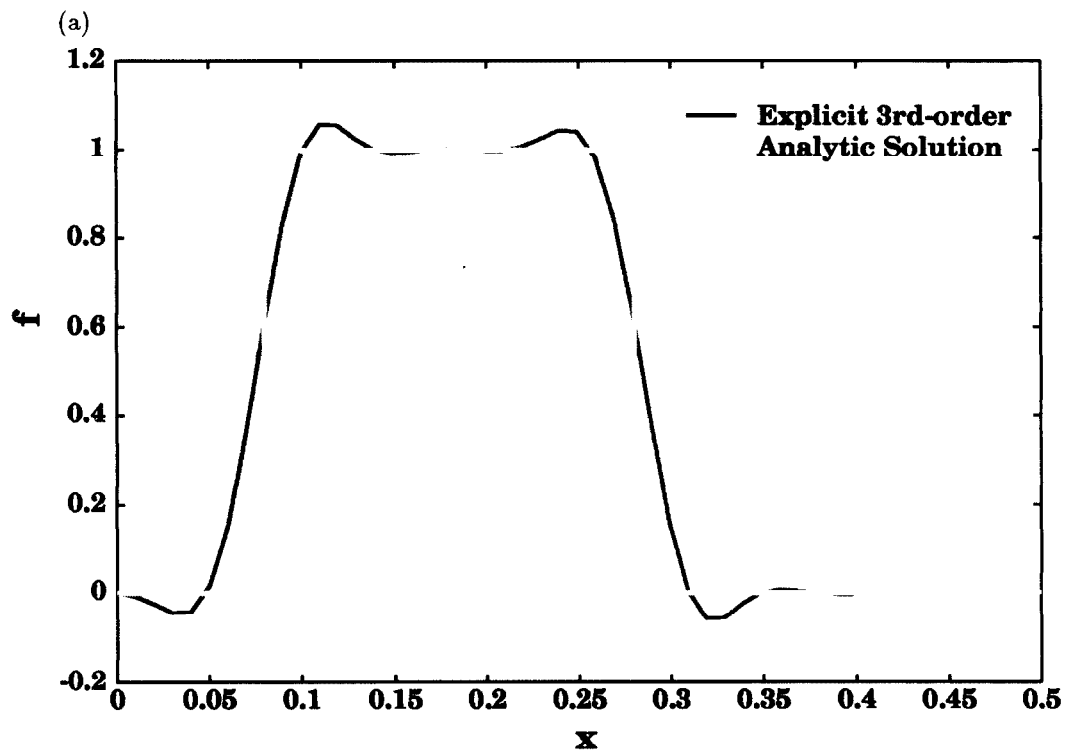
$$\begin{aligned} f_{x,i}^{n+1} &= f_{x,i}^n - u \Delta t \left( -6 \frac{f_i^{n+1} - f_{i-1}^{n+1}}{\Delta x^2} + 2 \frac{f_{x,i-1}^{n+1} + 2 f_{x,i}^{n+1}}{\Delta x} \right) \\ &\quad - \frac{(u \Delta t)^2}{2} \left( -12 \frac{f_i^{n+1} - f_{i-1}^{n+1}}{\Delta x^3} + 6 \frac{f_{x,i-1}^{n+1} + f_{x,i}^{n+1}}{\Delta x^2} \right) \end{aligned} \quad (5.14)$$

陽的解法の場合と同じように、この離散式は陰的 CIP 法の移流相の離散式と完全に一致する。従って CFL 数に依存せず安定な数値計算が経験的に保証されている。[40]

式 (5.13), 式 (5.14) は、Gauss-Seidel 法による緩和計算が行われる。Gauss-Seidel 法における緩和ステップを  $m$  で表すと反復計算中で利用される以下の式が得られる。

$$\begin{aligned} f_i^{m+1} &= f_i^n - u \Delta t f_{x,i}^{m+1} \\ &\quad - \frac{(u \Delta t)^2}{2} \left( -6 \frac{f_i^{m+1} - \overline{f_{i-1}^{m+1}}}{\Delta x^2} + 2 \frac{\overline{f_{x,i-1}^{m+1}} + 2 f_{x,i}^{m+1}}{\Delta x} \right) \\ &\quad - \frac{(u \Delta t)^3}{6} \left( -12 \frac{f_i^{m+1} - \overline{f_{i-1}^{m+1}}}{\Delta x^3} + 6 \frac{\overline{f_{x,i-1}^{m+1}} + f_{x,i}^{m+1}}{\Delta x^2} \right) \end{aligned} \quad (5.15)$$

$$\begin{aligned} f_{x,i}^{m+1} &= f_{x,i}^n - u \Delta t \left( -6 \frac{f_i^{m+1} - \overline{f_{i-1}^{m+1}}}{\Delta x^2} + 2 \frac{\overline{f_{x,i-1}^{m+1}} + 2 f_{x,i}^{m+1}}{\Delta x} \right) \\ &\quad - \frac{(u \Delta t)^2}{2} \left( -12 \frac{f_i^{m+1} - \overline{f_{i-1}^{m+1}}}{\Delta x^3} + 6 \frac{\overline{f_{x,i-1}^{m+1}} + f_{x,i}^{m+1}}{\Delta x^2} \right) \end{aligned} \quad (5.16)$$



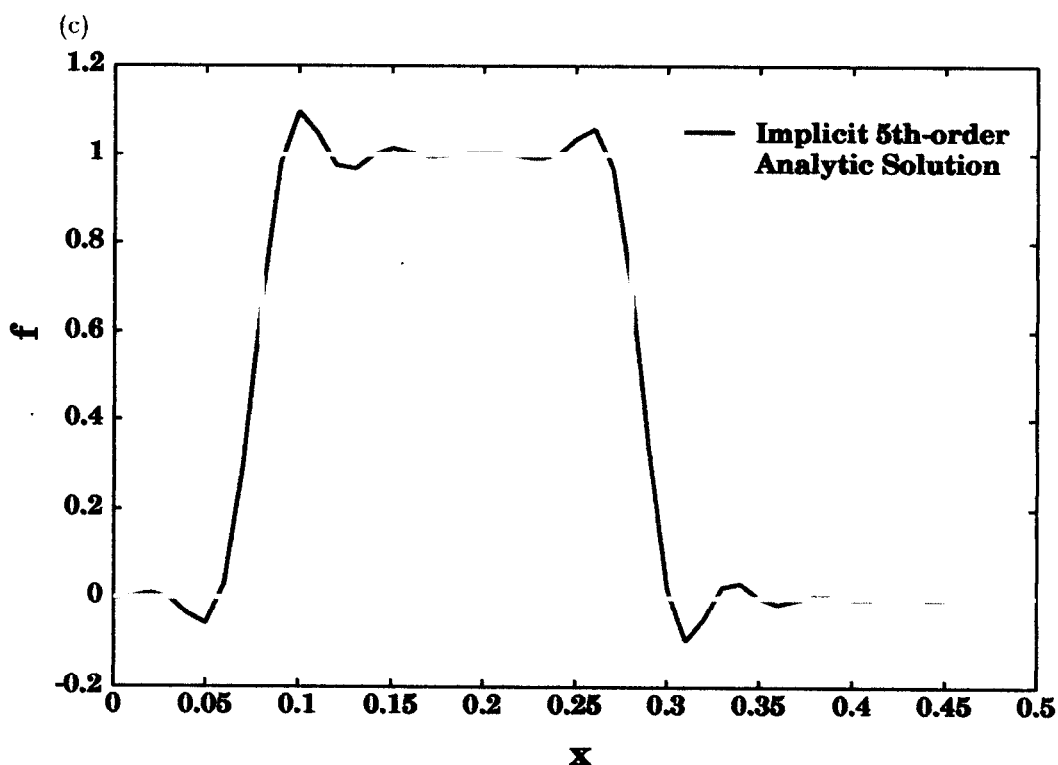


図 5.1: 陰的 IDO 法による 1 次元移流方程式の解法。

(a) 3 次精度 IDO 法陽解法、(b) 3 次精度陰的 IDO 法、(c) 5 次精度陰的 IDO 法による一次元移流方程式の計算結果

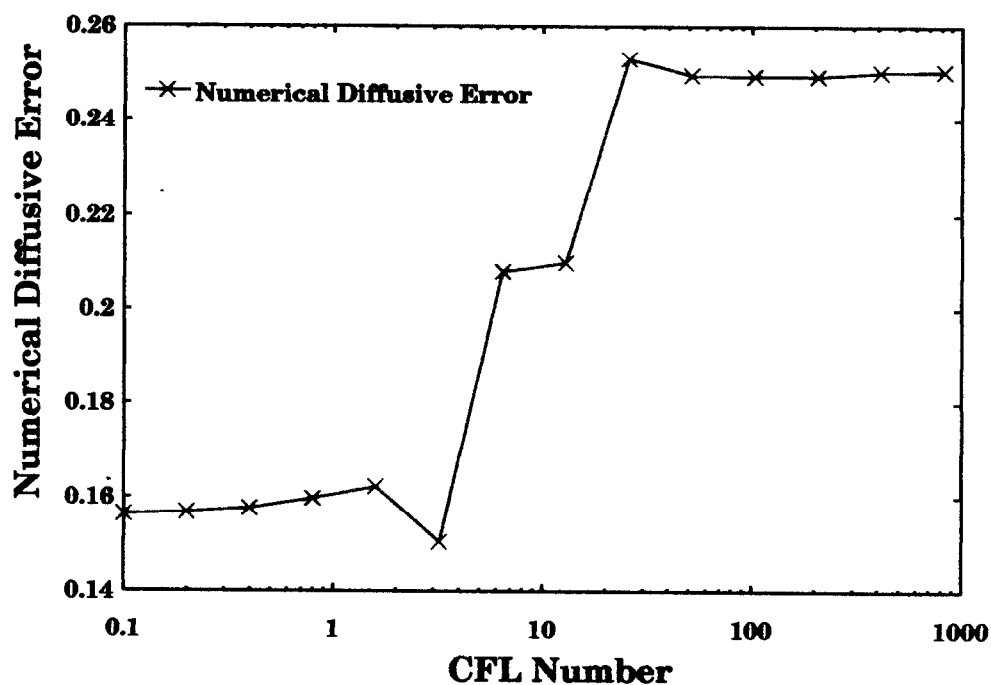


図 5.2: 陰的 IDO 法における CFL 数に対する数値拡散誤差:

CFL 数を大きくすることで対象計算時間までの計算ステップ数を減少することができることから、数値拡散誤差は CFL 数に対して大きく影響を受けることは無い。また、大きな CFL 数 ( $\approx 1000$ ) に対しても安定に計算できることを確認した。

初期条件を  $f_i^{(m=0)} = f_i^n$  として、 $f_i^{m+1}$  を反復計算により逐次更新する。ここで、記号 '—' を付記した値は緩和ステップ内に更新された値を得ることが出来るので、最新の情報を利用した。緩和計算を  $\sum_i |f_i^{m+1} - f_i^m| < \epsilon$  が成立するまで行うことで  $f_i^m$  を  $f_i^{m+1}$  に十分な精度で漸近させることが可能である。

計算空間  $0 \leq x \leq 1$ ;  $\Delta x = 1/100$  で、移流速度  $u(x) = 1$  を与え、初期条件  $f = 1 (0.05 \leq x \leq 0.15)$ ,  $f = 0$  (その他の点) として周期境界条件を用い、数値計算を  $t = 20$  まで行った。CFL =  $1 \times \Delta t / \Delta x = 0.2$  とした。計算結果から得られたプロファイルと解析解を図 5.1 に示す。風上 3 次補間を用いた陽解法 (図 5.1 (a)) と風上 3 次補間を用いた陰解法 (図 5.1 (b)) では、陰解法から得られたプロファイルの方が若干陽解法から得られるプロファイルよりも数値拡散が見られた。その一方で風上 5 次補間を用いた陰的 IDO 法ではオーバーシュートやアンダーシュートが、風上 3 次補間を用いたものに比べ大きくなったが数値拡散の少ない解を得られた。また、同じ計算条件を用い CFL 数に対する数値拡散誤差を見積もった結果を図 5.2 に示す。

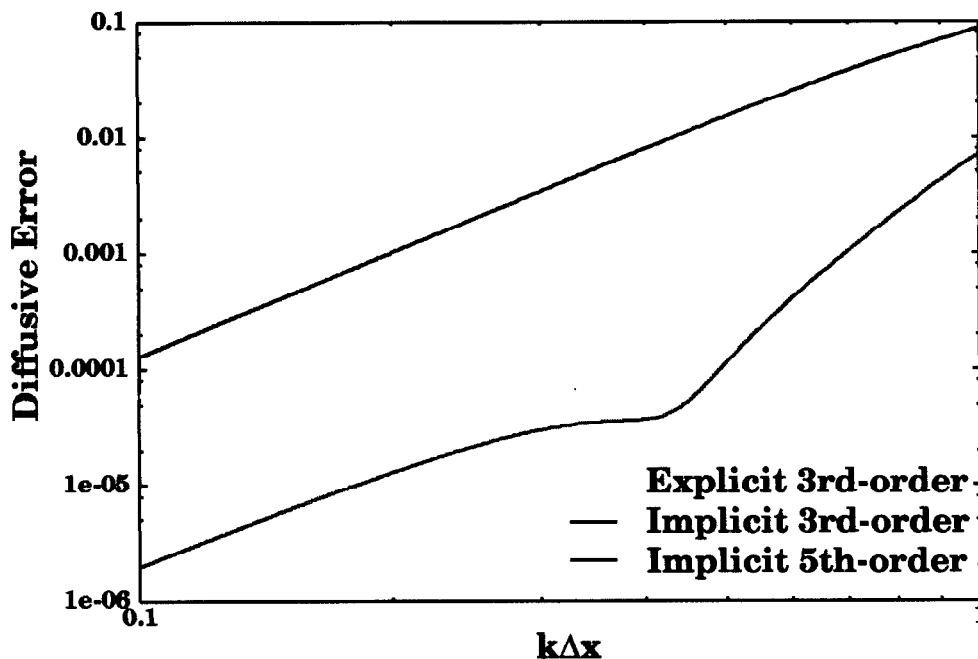


図 5.3: 陰的 IDO 法による 1 次元移流方程式の数値拡散誤差

CFL = 0.2 での波数に対する数値拡散誤差。陽解法 IDO 法に比べ風上補間に 3 次多項式を用いた陰的 IDO 法は若干数値拡散誤差が増加する傾向が見られる。その一方で、風上 5 次補間を用いた陰的 IDO 法は風上 3 次補間を用いた計算に比べ低数値拡散誤差で計算できることを示した。

CFL 数を大きくとることで対象計算時間までに要する計算ステップ数が減少することから、数値拡散誤差は 2 倍と変わらない程度で計算できている様子が分かる。また、大きな CFL 数  $\approx 1000$  でも安定に計算できることを確認した。

次に、空間  $0 \leq x \leq 1$  に一様定常速度場  $u(x) = 1$  を与え、初期プロファイルを正弦関数 ( $\sin kx$ ) とし、CFL = 0.2 で時間  $t = 1$  まで計算することで、初期プロファイルの波数に対する数値拡散誤差を

$$\sum_i^N (|f_{exact} - f_i| / f_{exact}) \quad (5.17)$$

を用いて見積もることを考える。その結果を図 5.3 に示す。風上 3 次補間を用いた陽解法 IDO 法による計算結果から得られた数値拡散誤差は風上 3 次補間を用いた陰的 IDO 法のものよりも数倍程度良い結果が得られたが、波数に対する精度は両者ともに  $\Delta x^3$  で見積もられた。また、風上 5 次補間を用いた陰的

IDO 法から得られた数値拡散誤差は風上 3 次補間を用いた陰的 IDO 法のものに比べ  $10^1 \sim 10^2$  程度良いことを示した。数値不安定性が生じない限り陰解法では風上補間に 5 次多項式を用いた方が高精度を示すことを確認した。

### Burgers 方程式の検証

非線形問題に陰的 IDO 法を適用する例題として Burgers 方程式、

$$u_t + uu_x = \kappa u_{xx} \quad (5.18)$$

を検証する。陰的 IDO 法による計算を行うため、Burgers 方程式を式 (5.1)、式 (5.2) に代入し、時間に関する微係数を全て空間に関する微係数に置き換えることで、以下の関係式を得る。

$$u^{n+1} = u^n + u_t^{n+1} \Delta t - \frac{1}{2} u_{tt}^{n+1} \Delta t^2 + \frac{1}{6} u_{ttt}^{n+1} \Delta t^3, \quad (5.19)$$

$$u_x^{n+1} = u_x^n + u_{tx}^{n+1} \Delta t - \frac{1}{2} u_{ttx}^{n+1} \Delta t^2, \quad (5.20)$$

$$\begin{aligned} u_t^{n+1} &= -u^{n+1} u_x^{n+1} + \kappa \tilde{u}_{xx}^{n+1}, \\ u_{tx}^{n+1} &= -u_x^{n+1} u_x^{n+1} - u^{n+1} u_{xx}^{n+1} + \kappa \tilde{u}_{xxx}^{n+1}, \\ u_{ttx}^{n+1} &= -3u_x^{n+1} u_{xx}^{n+1} - u^{n+1} u_{xxx}^{n+1} + \kappa \tilde{u}_{xxxx}^{n+1}, \\ u_{ttx}^{n+1} &= -3u_{xx}^{n+1} u_x^{n+1} - 4u_x^{n+1} u_{xxx}^{n+1} - u^{n+1} u_{xxxx}^{n+1} + \kappa \tilde{u}_{xxxxx}^{n+1}, \\ u_{ttt}^{n+1} &= -10u_{xx}^{n+1} u_{xxx}^{n+1} - 5u_x^{n+1} u_{xxxx}^{n+1} - u^{n+1} u_{xxxxx}^{n+1} + \kappa \tilde{u}_{xxxxxx}^{n+1}, \\ u_{tt}^{n+1} &= -u_t^{n+1} u_x^{n+1} - u^{n+1} u_{tx}^{n+1} + \kappa u_{ttx}^{n+1}, \\ u_{ttx}^{n+1} &= -2u_{tx}^{n+1} u_x^{n+1} - u_t^{n+1} u_{xx}^{n+1} - u^{n+1} u_{ttx}^{n+1} + \kappa u_{tttx}^{n+1}, \\ u_{tttx}^{n+1} &= -3u_{ttx}^{n+1} u_x^{n+1} - 3u_{tx}^{n+1} u_{xx}^{n+1} - u_t^{n+1} u_{xxx}^{n+1} - u^{n+1} u_{tttx}^{n+1} + \kappa u_{ttttx}^{n+1}, \\ u_{ttt}^{n+1} &= -u_{tt}^{n+1} u_x^{n+1} - 2u_t^{n+1} u_{tx}^{n+1} - u^{n+1} u_{ttt}^{n+1} + \kappa u_{tttt}^{n+1}, \end{aligned}$$

ここで、 $\tilde{\quad}$  が付記されている項は非移流相から派生した高次空間微係数であることから、中心補間を微分することで求める。他の項は 3 次精度風上補間を用いて算出した。Burgers 方程式は非線形方程式であるため式 (5.19)、式 (5.20) であらわされるように式 (5.9) に含まれるベクトル  $F^{n+1}$  の乗算により記述される項が含まれることから、線形方程式のように行列表記を行うことができない。そこで、式 (5.19)、式 (5.20) を線形方程式同様、Gauss-Seidel 法で解くために非線形項  $u^{n+1} u_x^{n+1}$  を  $u^m u_x^{n+1}$  と置くことで  $u^{n+1}, u_x^{n+1}$  に対して線形化を行った。 $u^m$  は緩和時間内の最新の情報に緩和ステップ毎に更新され、緩和が進行するに従って  $u^{n+1}$  に漸近する。

空間  $0 \leq x \leq 2, \Delta x = 2/200$  に対して、初期プロファイル  $u = 1.0 (x < 1.0), u = -0.5 (x \geq 1.0)$  を与え、 $CFL = \Delta t / \Delta x = 0.1$  により、固定境界条件のもと  $t = 2$  まで計算した結果を図 5.4 に示す。移流速度が空間で非一様であるため代表速度を 1 として CFL 数を決定した。保存則から求まる弱解から、衝撃波速度は衝撃波前後の移流速度  $a, b$  を用いて  $(a+b)/2$  と求まる。本例題では  $a = 1, b = -0.5$  と与えたことから、衝撃波速度は 0.25 となる。すなわち数値計算で行った  $t = 2$  では初期位置 1 を考慮すると、衝撃波位置は 1.5 と求まる。図 5.4 (a) で、衝撃波位置は陽解法、陰解法ともに弱解と良く一致した結果を得た。次に、拡散係数  $\kappa$  を変化させて陰的 IDO 法により数値計算した結果を図 5.4 (b) に示す。図 5.4 (b) から、衝撃波速度は拡散係数に依存しないことが分かる。拡散係数を  $\kappa = 0.1$  として Burgers 方程式の数値計算を行った場合、陽解法では  $CFL = \Delta t / \Delta x$  を 0.01 以下としなければ数値不安定性が生じたにもかかわらず、陰的 IDO 法を用いた場合、 $CFL = \Delta t / \Delta x$  が 5 でも安定に計算が可能であった。また、IDO 陽解法では拡散係数を大きくするに従って、CFL を小さくしないと数値不安定性が発生したが、陰的 IDO 法では逆に拡散係数を大きくする程 CFL を大きくとれる傾向が見られた。これは陰解法が拡散現象に対して有効である性質によるものである。

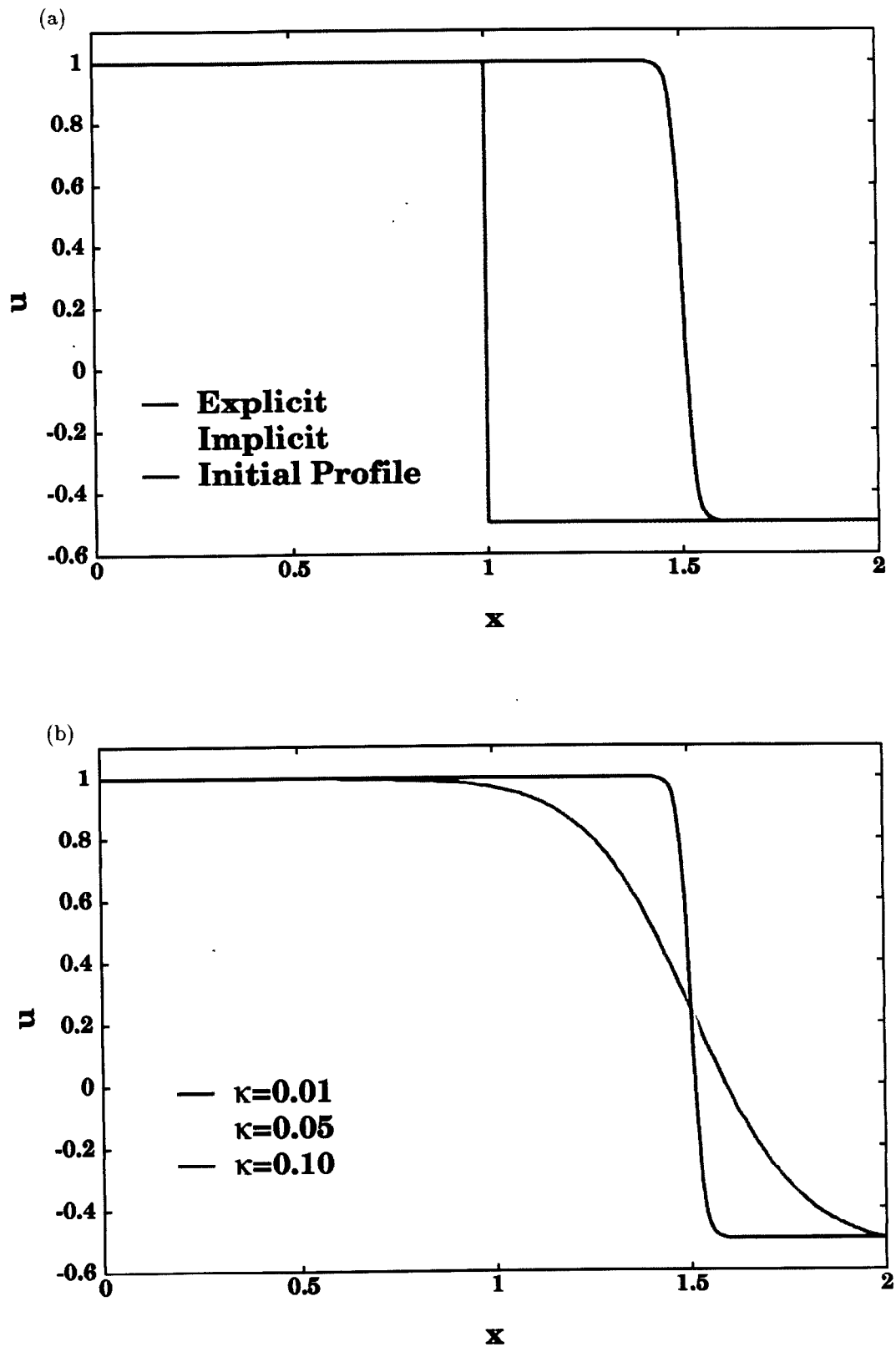


図 5.4: Burgers 方程式の陰的 IDO 法での計算結果

(a) に陽解法と陰解法による Burgers 方程式の数値解を、(b) に陰解法で拡散係数  $\kappa$  を変化した時の数値解を示す。



## Korteweg-de Vries (KdV) 方程式の検証

非移流相を構成する 3 階空間微係数が分散項として働くため高精度で解くことが難しい非線形方程式の一つである KdV 方程式 [44]

$$\partial u / \partial t + u \partial u / \partial x = -\delta^2 \partial u^3 / \partial x^3 \quad (5.21)$$

の数値計算を陰的 IDO 法を用いて行い、数値安定性の評価を行う。離散式は先に挙げた Burgers 方程式の拡散項  $\kappa u_{xx}$  を KdV 方程式の分散項  $-\delta^2 u_{xxx}$  に置き換えることで導くことができるのでここでは省略する。

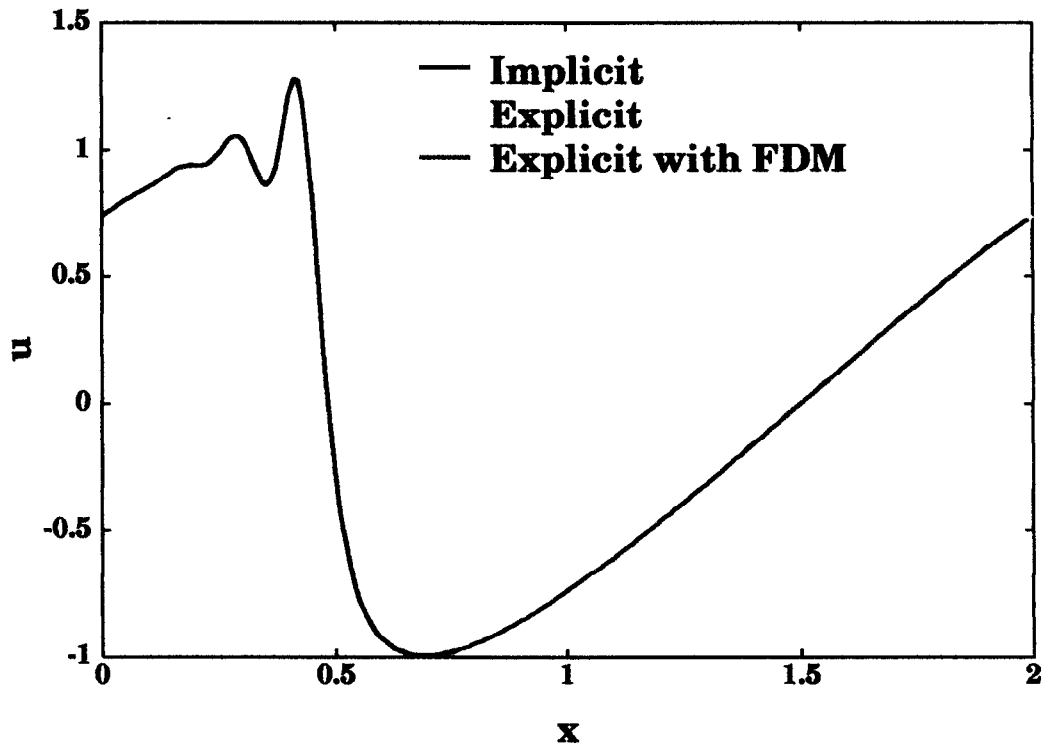
KdV 方程式を陽解法 IDO 法で扱うには物理量の保存誤差の問題から分散項  $-\delta^2 u_{xxx}$  を IDO 法で用いている中心補間式を用いたグリッド点に於ける空間 3 次微係数を与えるのではなく、微係数に対して 2 次中心差分を行い、

$$\partial^3 u / \partial x^3 = (u_x[i+1] - 2u_x[i] + u_x[i-1]) / \Delta x^2 \quad (5.22)$$

として与える必要がある [39]。しかし、分散項を差分形式で扱い陰解法化を行ったところ、CFL 数を大きくすることで数値不安定性が生じたため、中心補間を微分することで分散項を計算し、保存の問題を解消するために計算ステップ毎に保存が成立するように空間全体に線形に補正量を加える計算についても検証した。

空間  $0 \leq x \leq 2$ ,  $\Delta x = 2/200$  に対して初期プロファイルを  $u = \cos(\pi x)$ 、分散係数を  $\delta = 0.022$  とした。陽解法では  $CFL = \Delta t / \Delta x = 2.0 \times 10^{-5}$ 、陰解法では  $CFL = 2.0 \times 10^{-2}$  で  $t = 1/\pi$  まで計算した結果を図 5.5 (a) に示す。ただし、Explicit with FDM では分散項に差分方式を用いているが、他では中心補間から得られる微係数を用いて計算を行った。CFL 数は空間で移流速度が非一様であることから、代表速度を 1 として計算した。

(a)



(b)

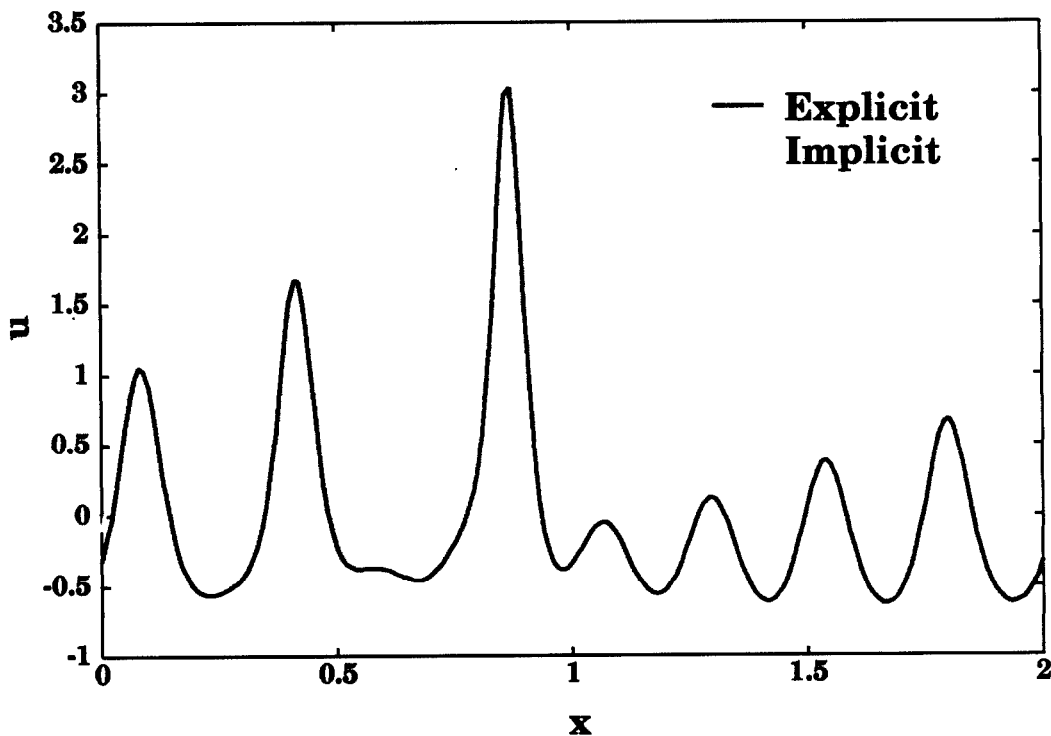


図 5.5: KdV 方程式を陽解法、陰解法 IDO 法で数値計算した結果:

CFL が小さければ陽解法と陰解法から得られる結果は完全に一致する。陰解法で CFL を大きくとっても分散が小さいところでは陽解法から得られる結果とよく一致していることが分かる。

図 5.5 (a) から、陽解法による結果と陰解法による結果が分散項に差分法を用いた陽解法から得られる結果と良く一致した。陰解法と陽解法から求められる数値解の差異を比較するため  $t = 3.6\pi$  まで計算した結果を図 5.5 (b) に示す。陽解法と陰解法で位相誤差が見られたが、陰解法化により波数  $\sqrt{\Delta x / \delta^2 \Delta t}$  より大きな波が  $\Delta t$  を大きくとったことにより表現できなくなったことが原因の一つと考えられる。また、線形補正により保存が成立するように数値計算を行っているが、この手法が現象論的に正しくないことも原因として挙げられる。この問題に関しては今後更なる検討が必要と考えるが、それ以上に陰解法化により CFL 数を 1000 倍以上大きくとっても安定に計算できたことは重要である。

### 1 次元衝撃波管問題の検証

一般的な圧縮性流体問題は密度 ( $\rho$ )、移流速度 ( $u$ )、圧力 ( $P$ ) に関してそれぞれ偏微分方程式で質量保存則、運動量保存則、エネルギー保存則が記述され、それらを連立することで流体運動をとらえることが出来る。1 次元圧縮性流体問題を記述する偏微分方程式群は以下ようになる。

$$\rho_t + u\rho_x + u_x\rho = 0 \quad (5.23)$$

$$u_t + uu_x = -\frac{P_x}{\rho} \quad (5.24)$$

$$P_t + uP_x = -\gamma u_x P \quad (5.25)$$

ここで、 $\gamma (= 1.4)$  は比熱比である。人工粘性を加味しなければ衝撃波位置において数値不安定を生じるため、非移流相に含まれる圧力成分に対し、 $q = \alpha c_s \rho u_x$  で表される粘性を圧縮領域領域 ( $u_x < 0$ ) に対し

て線形に加算した [13]。ここで、 $c_s$  は音速、 $\alpha$  は 1 程度の定数である。上記偏微分方程式に対し、移流相に対しては時間精度 3 次の陰解法を、非移流相に対しては時間精度 1 次の陰解法を適用し、計算を行った。ただし、完全に上記 3 方程式を連立させると IDO 法では物理量に加えてその 1 次空間微係数を従属変数として扱うため 1 次元問題を扱っているにも関わらず  $6 \times 6$  の逆行列計算を緩和ステップ毎に行わなければならないため、計算負荷の増大が考えられる。また、運動量の保存を記述する Navier-Stokes 方程式は非線形方程式であるため、線形化を行わなければ行列式として記述することができない。そこで、移流相に含まれる移流速度は非線形方程式で用いたように、緩和時間毎に更新する値を導入することを考える。これにより各物理量の緩和計算は他の物理量との関連性を持たなくなることから、それぞれに対して  $2 \times 2$  行列の逆行列演算により行うことができる。

$$\rho_t + u^m \rho_x^{n+1} = 0 \quad (5.26)$$

$$u_t + u^m u_x^{n+1} = 0 \quad (5.27)$$

$$P_t + u^m P_x^{n+1} = 0 \quad (5.28)$$

⋮

非移流項は各偏微分方程式を代表する物理量の時刻を  $n+1$  とし、その他の値は下記に示す通り全て緩和時間毎に更新する。

$$\rho_t = -u_x^m \rho^{n+1} \quad (5.29)$$

$$u_t = -\frac{P_x^m}{\rho^m} \quad (5.30)$$

$$P_t = -\gamma u_x^m P^{n+1} \quad (5.31)$$

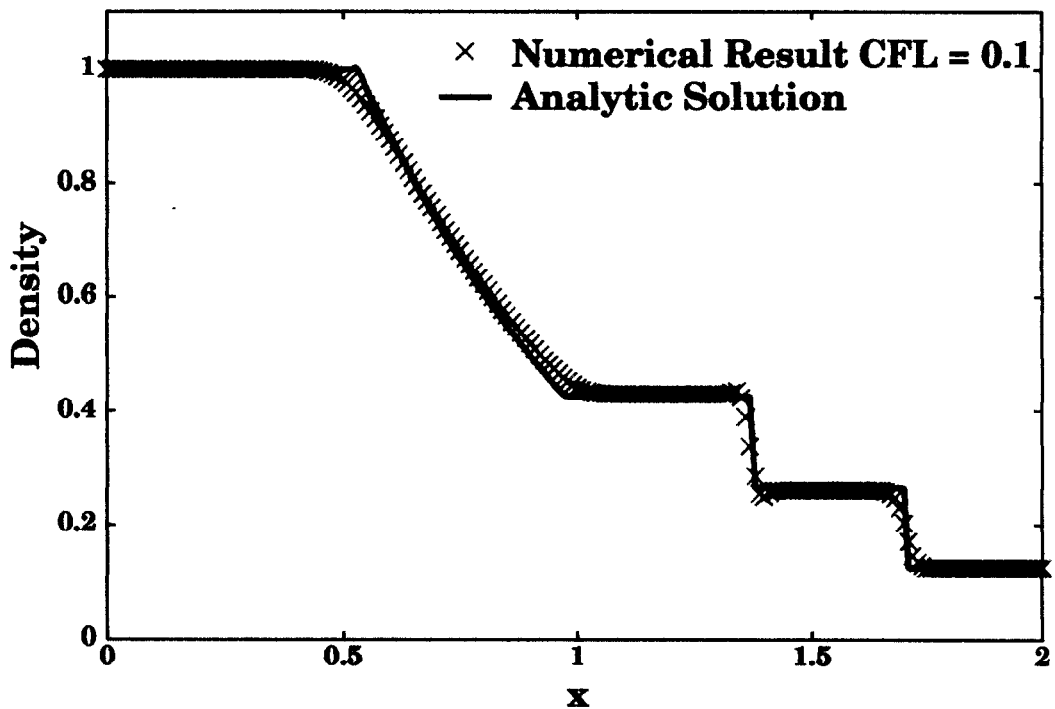


図 5.6: 陰的 IDO 法による 1 次元衝撃波管問題の数値解:

初期プロファイルを  $\rho = 1, P = 1 (x \leq 1), \rho = 0.125,$

$P = 0.1 (x > 1)$ 、全空間に於いて移流速度  $u = 0$  とし、 $t = 0.4$  まで  $CFL = \Delta t / \Delta x = 0.1$  で数値計算した結果

上記手法で数値計算を行ったところ、CFL 数を大きくすることが出来ず、数値不安定が生じる結果となった。更に数値安定性を向上することを目的として、速度、圧力に関する偏微分方程式の非移流相に対して時間 2 次精度まで考慮した場合に派生する空間微係数の内、粘性パラメータとして作用する下記の 2 項の導入を行う。

$$u_{tt} = \gamma \frac{P^m}{\rho^m} u_{xx}^{n+1} \quad (5.32)$$

$$P_{tt} = \gamma \frac{P^m}{\rho^m} P_{xx}^{n+1} \quad (5.33)$$

いずれも 音速<sup>2</sup> × 物理量の空間 2 次微係数となっており、数値粘性として作用することが分かる。

以上をふまえ、空間  $0 \leq x \leq 2, \Delta x = 2/200$  に対し、初期プロファイルを  $\rho = 1, P = 1 (x \leq 1), \rho = 0.125, P = 0.1 (x > 1)$ 、全空間に於いて移流速度  $u = 0, \gamma = 1.4$  とし、 $t = 0.4$  まで  $\text{CFL} = \Delta t / \Delta x = 0.1$  として計算した結果を図 5.6 に、 $\text{CFL} = \Delta t / \Delta t = 0.5$  で計算した結果を図 5.7 に示す。移流速度が空間で非一様であることから移流速度を 1 として CFL 数を表記した。ただし、図 5.6 では、CFL 数が小さいことから非移流相に対して式 (5.32)、式 (5.33) を用いた安定化を施さずに計算を行った。

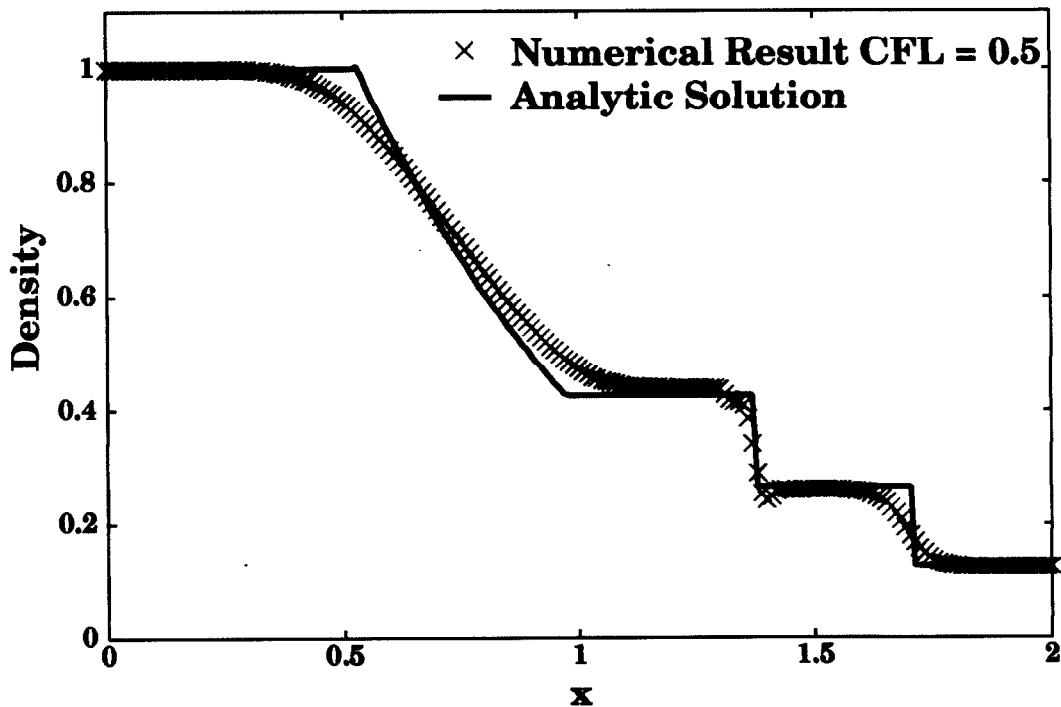


図 5.7: 陰的 IDO 法による 1 次元衝撃波管問題の数値解:

初期プロファイルを  $\rho = 1, P = 1 (x \leq 1), \rho = 0.125,$

$P = 0.1 (x > 1)$ 、全空間に於いて移流速度  $u = 0$  とし、 $t = 0.4$  まで  $\text{CFL} = \Delta t / \Delta x = 0.5$  で数値計算した結果

陽解法 IDO 法では、衝撃波速度が 1 を越えるため  $\text{CFL} = \Delta t / \Delta x \leq 0.1$  でなければ安定に計算できなかったのに対し、陰的 IDO 法では  $\text{CFL} = 0.5$  でも精度良く解けていることが分かる。ただし、衝撃波位置やその形状、膨張波の境界面の数値拡散が顕著に現れている。このような数値拡散は、非移流相計算に於いて時間 2 階微分項のを構成する空間微係数に、拡散項として効果を持つ項を選択的に加えたためと考えられる。時間 2 階微分項の他の項を数値計算に含めることにより厳密な離散化式を形成することでこのような数値拡散を減少させることが可能と考えられるが、本節では CFL 数を大きくとるために意図的に非移流相に対して工夫を行ったため、厳密な解法を行った場合必ずしも CFL 数を大きくとれるとは限ら

ない。比較として、先のような安定化を施さずに陰的 IDO 法を適用し、数値計算を行うと、図 5.6 に示したように少ない数値拡散で衝撃波面をとらえられることを示した。ただし、この手法では CFL 数を大きくとることが出来ず、CFL = 0.2 程度で数値不安定性により計算が破綻した。IDO 法は陽解法、陰解法問わず安定解析が非常に難しい手法であり、大きな CFL であっても常に安定でかつ von Neumann の安定解析から得られる利得係数が 1 に限りなく近い離散化式をつくり出すのは極めて困難であると考えられる。しかし陰的 IDO 法の導入により陽解法 IDO 法に比べて CFL 数に対する安定性の向上を行うことが可能であることを示した。

## 2 次元移流方程式の検証

陰的 IDO 法を 2 次元に拡張し、精度検証ならびに CFL 数に対する安定解析を行う。本節では例題として 2 次元移流方程式

$$f_t + u f_x + v f_y = 0 \quad (5.34)$$

を扱う。空間  $-0.5 \leq x \leq 0.5$ ,  $-0.5 \leq y \leq 0.5$ ,  $\Delta x = 1/100$ ,  $\Delta y = 1/100$  に対して回転速度場  $u(x, y) = y$ ,  $v(x, y) = -x$  を与え、初期プロファイルを  $f = 2(-0.1 \leq x \leq 0.1, 0.15 \leq y \leq 0.35)$ ,  $f = 1$ (その他の点) とした。デカルト座標系に対して回転速度場を与えていることから時刻  $t = 2n\pi + (\pi/2, \pi, 3\pi/2, 2\pi)$  ( $n = 0, 1, 2, \dots$ ) で解析解が求まることを利用して、解析解からの数値拡散誤差を上記時刻に於いて

$$\sum_i^N \sum_j^M |f_{Analytic} - f_{i,j}| / f_{Analytic} / (N \times M) \quad (5.35)$$

により見積もった。ここで、 $f_{Analytic}$  は解析解を示し、 $N, M$  はそれぞれ  $x, y$  方向のグリッド点数とした。 $t = 10\pi$  まで上記手法で数値拡散誤差を計算した結果を図 5.8 に示す。

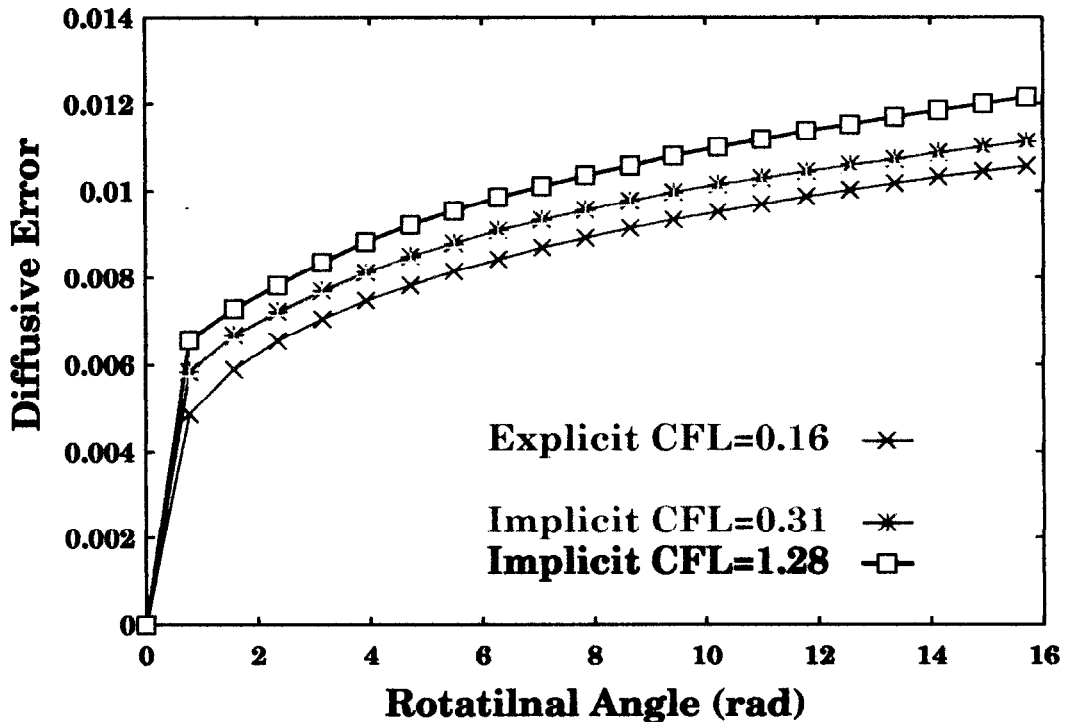


図 5.8: 陰的 IDO 法による 2 次元移流方程式の数値拡散誤差

陽解法 IDO 法から得られた数値拡散誤差に比べ陰解法 IDO 法から得られた数値拡散誤差は相対的にたかだか 20% 大きい程度であることを示した。また、CFL 数による数値拡散誤差の変動も対象とした範囲内では 2 倍と変わらない値を示した。

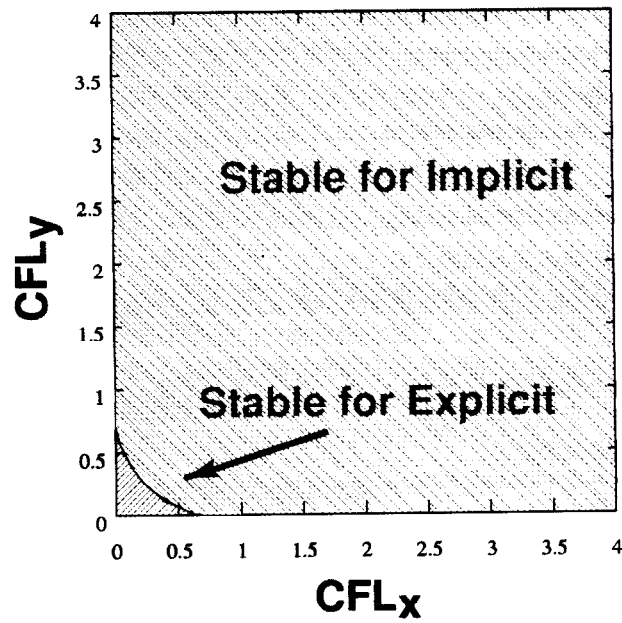


図 5.9: 2次元移流方程式の CFL 数に対する安定領域:

陽解法 IDO 法では高々 CFL 数  $< 1$  でなければ数値不安定性が見られたにも関わらず、陰的 IDO 法を用いることで、 $CFL = \Delta x / \Delta t > 4$  であっても安定にかつ高精度で数値計算を行うことが可能であった (b) .

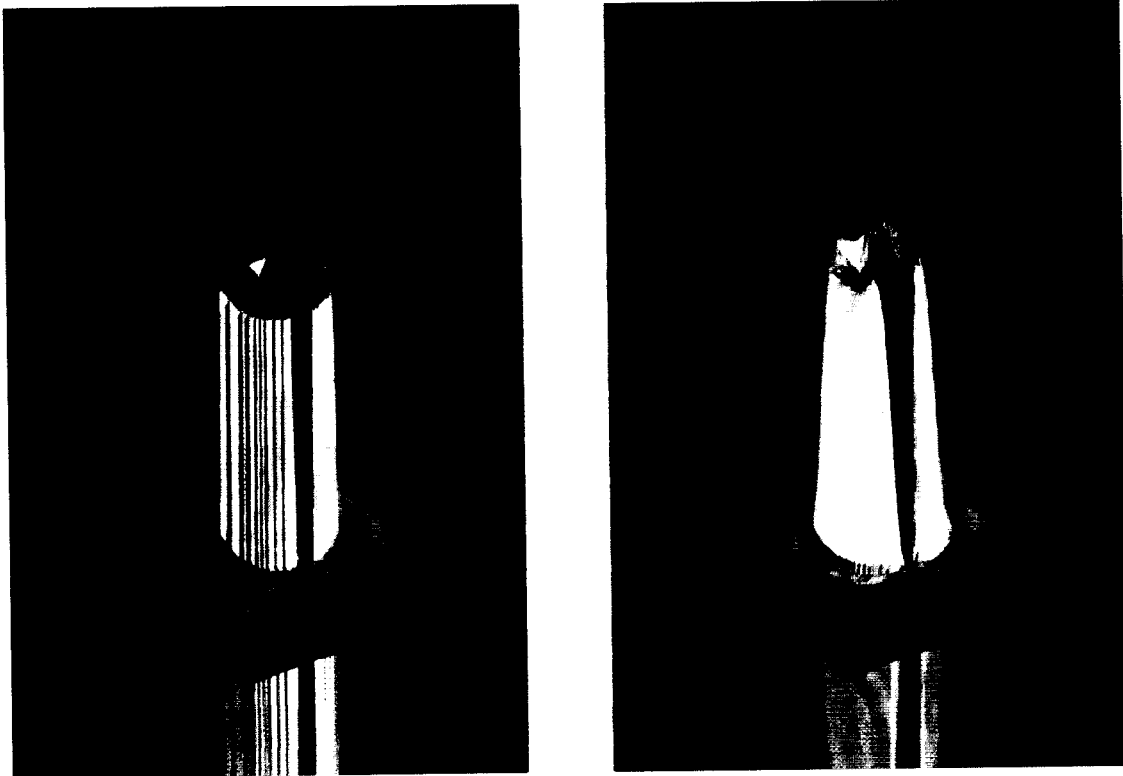


図 5.10: 陰的 IDO 法による Zalesak 例題の数値解:

陰的 IDO 法を用いることで、 $CFL = \Delta x / \Delta t = 1.56$  での計算でも安定にかつ高精度で数値計算を行うことが可能であった

$t = 10\pi$  で計算空間を初期プロファイルは計算空間の原点を中心に 5 回転するが、数値拡散誤差は図 5.8 に示した全ての場合において  $10^{-2}$  程度になることを示した。陽解法による数値計算結果から得られた数値拡散誤差と、陰解法によるそれを比較しても同程度になることを確認した。このことから陰解法化による数値拡散誤差増大は影響が小さいと結論づける。また、陰的 IDO 法では数値拡散誤差に着目すると、CFL 数にほとんど依存しない結果を得た。

2 次元移流方程式に対して、CFL 数による安定性の検証を行った結果を図 5.9 に示す。IDO 陽解法は補間関数の形成に用いる境界条件が CIP 法と比較して少ないことや、その方向が限定されていることから CFL 数に対する安定性が CIP 法よりも制限されている。1 次元方向の移流のみを行った場合では CFL 数 ( $\Delta t/\Delta x$ ) が 1 まで利用できるが、2 次元直交座標系に於いて  $45^\circ$  方向の移流に関しては CFL 数は  $x$  方向、 $y$  方向にせいぜい 0.1 程度までしかとることが出来ない。陰解法化を施すことにより、CFL 数に対する安定性を高めることが可能であり、1 次元方向の移流に関しては CFL 数が 1000 以上でも安定数値計算が可能であることを確認した。また、直交座標に於ける  $45^\circ$  方向の移流では  $\Delta t/\Delta x$  が 4 であっても安定な計算が可能であることを確認した。

次に、2 次元移流問題の例題として Zalesak の例題を扱い、形状の変形についての検証を行う。計算空間  $-0.5 \leq x \leq 0.5, -0.5 \leq y \leq 0.5, \Delta x = 1/100, \Delta y = 1/100$  に対し、 $CFL = \Delta t/\Delta x = 1.56$  で計算を行った結果を図 5.10 に示す。得られた数値解から見積もられた保存誤差、数値拡散誤差はそれぞれ

$$\begin{aligned} \text{保存誤差} & 2.804122 \times 10^{-3} \\ \text{数値拡散誤差} & 1.841643 \times 10^{-2} \end{aligned}$$

と得られた。陽解法では  $CFL \leq 0.12$  程度でなければ安定な計算が出来なかったのに対し、CFL が 1 を越えても安定に、かつ陽解法と比べても遷色の無いほど十分に形状保持をしている結果を得た。

## 5.4 まとめ

大きな CFL 数 ( $> 0.3$ ) に対して IDO 法を安定化するために陰的 IDO 法の開発を行った。様々な偏微分方程式に対して汎用的に適用できるように陰解法での時間前進の際に必要な逆行列算出を緩和法の一つである Gauss-Seidel 法により求め、1 次元移流方程式、Burgers 方程式、KdV 方程式、圧縮性流体方程式系、2 次元移流方程式を一様に扱うことを可能にした。

1 次元移流方程式の検証から、使用する補間関数の次数が同じであれば陽解法の数値拡散誤差と陰解法の数値拡散誤差の差は数倍程度であることを示した。Burgers 方程式、KdV 方程式の例題を検証することで非線形方程式に対しても十分に安定性の改善を行えることを証明した。多次元問題に対しても陽解法と同程度の数値拡散誤差程度で 10 倍程度の CFL 数を用いた計算が可能であることを検証した。更に一般的な問題として、圧縮性流体問題を扱った。偏微分方程式間の情報伝達や非移流相の扱い等にまだ課題が残されているものの、衝撃波管問題のような複雑な衝撃波問題に対しても陰的 IDO 法の効果を得ることが出来た。

現時点で陰解法による計算は陽解法に対して数倍程度の計算時間が必要であるが、より大きな CFL 数をとれるように工夫をすることで逆に陽解法よりも少ない計算時間での計算も可能になると考えられる。また、4.1.2 節で用いるマルチグリッド法を適用することにより、緩和計算の高速化を行うことも可能である。ただし与えられた偏微分方程式の非線形項などに依存して、安定に計算できる CFL 数の範囲が異なるため大きな CFL 数をとること自体は重要とは考えがたい。それ以上に様々な偏微分方程式に対して安定化が図れたことが最大の成果であると考えられる。

## 第 6 章

# 接触不連続面を含む問題への移動カッターセル手法の導入

物質の境界面や、衝撃波面、相界面など自然界には様々な不連続面が存在するが、このような不連続面を数値計算で記述する場合、計算格子による離散化や計算スキームを用いた補間関数の形成により少なからず数値拡散が生じる。TVD スキーム [45, 46] や ENO スキーム [47, 48] に代表される差分系高精度スキームや CIP 法、IDO 法に代表されるエルミート補間を用いた計算スキームを用いたところで、高波数成分を含む不連続面の伝搬を扱う場合は数値拡散により数グリッドに渡る境界面が生じる。このようなオイラー座標系では扱いが困難な不連続面を低数値拡散誤差で解くためにはさまざまな手法を必要とする。

ラグランジュ手法を用いることにより不連続面位置に的確に計算格子点を定義することで不連続面を高精度に記述することが 1 つの解として考えられる。しかし、物体が大変形する現象をラグランジュ手法で扱った場合、物体表面を記述するグリッドも大きく変形するため多次元問題においては計算グリッドの消滅やグリッド間隔の膨張による計算精度低下に伴い計算不能なグリッドが生じる場合がある。そのような場合、計算格子点を再構築する Rezoning や Remeshing と呼ばれる手続きにより計算格子の再構成を行うが、それによって不連続面の表現解像度は大幅に低下することは容易に推測できる。

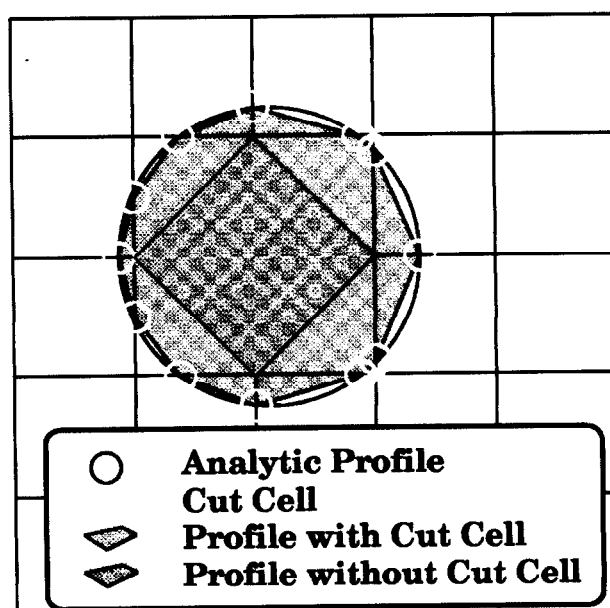


図 6.1: 境界条件として利用されるカッターセル

カッターセルは、オイラーグリッド上にプロファイルの境界をオイラーグリッドの解像度以上の精度で表現するために用いられる。



もう1つのアプローチとしてオイラーグリッド上に不連続面位置を記述することを目的とした識別関数を定義し、境界面を捕獲する手法が挙げられる。Level Set 手法 [49, 50, 51, 52] や、CIP 法で用いられている tangent 変換 [53]、粒子追跡法 (Particle Tracking Method) [54]、Front-Tracking Method [55] で代表されるこれらの手法を用いることでプロファイルの大変形に伴う計算の破綻や境界面記述精度の低下をまぬがれることが出来るが、その記述精度はオイラーグリッドの空間解像度ならびに計算スキームで用いる補間関数の精度に制限される。IDO 法による Level Set 関数の記述を行った場合では境界面記述解像度はほぼグリッド  $1/10$  程度 ( $\Delta x/10$ ) が得られるが、1 グリッド内にプロファイルが存在するような場合は記述することが出来ない。結果としてグリッドサイズ以下に複雑に変形するプロファイルを適切に表現できる手法は無いものと考えてよい。そのような問題に対してはより多くの計算グリッドを利用することで必要解像度を達成することが望ましいと考える。

近年ではカットセル [56, 57, 58] を動的に配置することで非圧縮性流体の境界面を扱う手法 [59, 60, 61] や、境界の変形を関数を用いて表記し、高精度でオイラーグリッド上に境界面を配置する手法 [62] などが登場しているが、本節ではグリッドサイズ以下には変形しないプロファイルに対して Level Set 手法と移動するカットセル手法を用いた不連続面の捕獲ならびに不連続面上での物理量計算手法を提案 [63] するとともに構築、計算検証を行う。

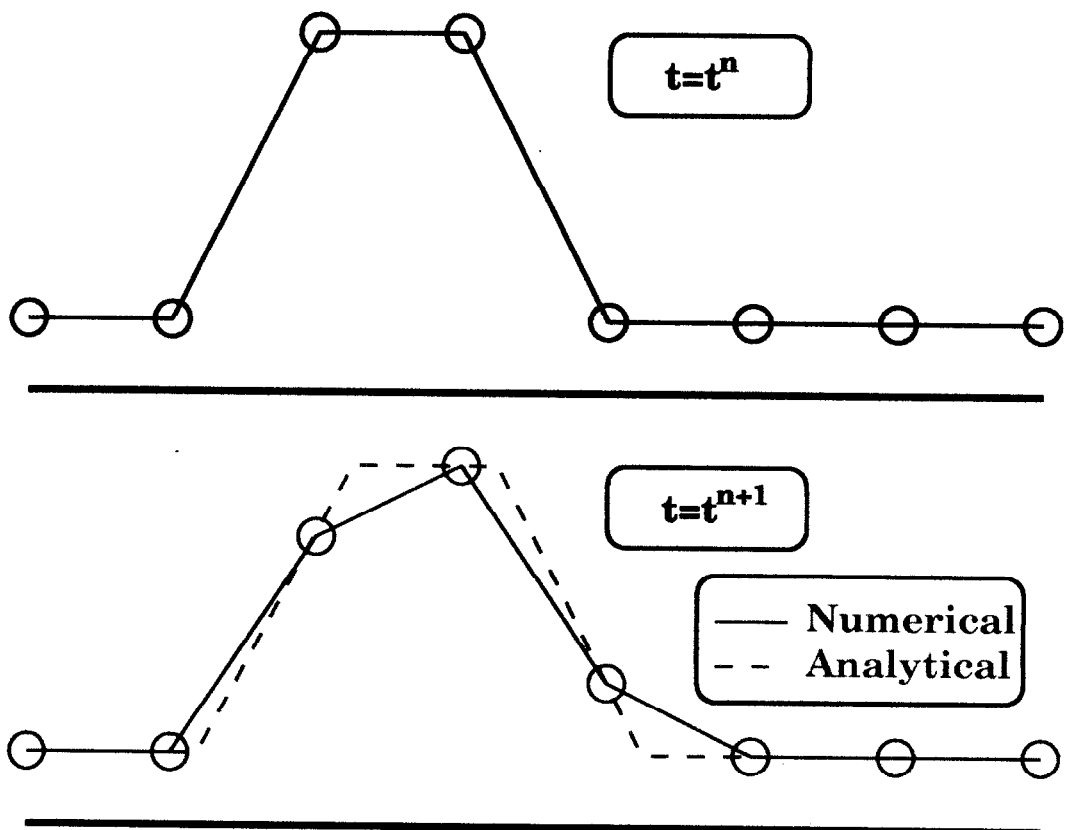


図 6.2: 1 次風上差分法による 1 次元移流方程式の解法

初期プロファイルが正しく移流計算されているにも関わらず、離散グリッド間を直線により補間することで数値拡散が生じる。

## 6.1 移動カットセル手法

カットセル手法は、オイラーグリッド上で物質の境界を高精度に表現するために用いられる手法である。図 6.1 に示すようにオイラーグリッド上に円形で境界条件を与えた場合、カットセルを用いない場合にはオイラーグリッドの解像度による制限から、そのプロファイルは図中の茶色の領域で表される多角形で表現される。その一方で、カットセルをプロファイルとオイラーグリッド軸との交点に定義することでプロファイル青色の領域で表される多角形で表現することができ、より境界条件を高精度で表現できることがわかる。

カットセルは上述のように固定された境界条件として利用されるが、オイラーグリッド上で境界面をオイラーグリッドに比べて高解像度で表現できることと、カットセルグリッド上では異なる領域に対する物理量を多価に定義できることから、移動する不連続面を数値計算で記述できる可能性を考慮し、カットセルがオイラーグリッド上で自由に移動できる移動カットセル手法 (Moving Cut Cell Method) の構築を行う。

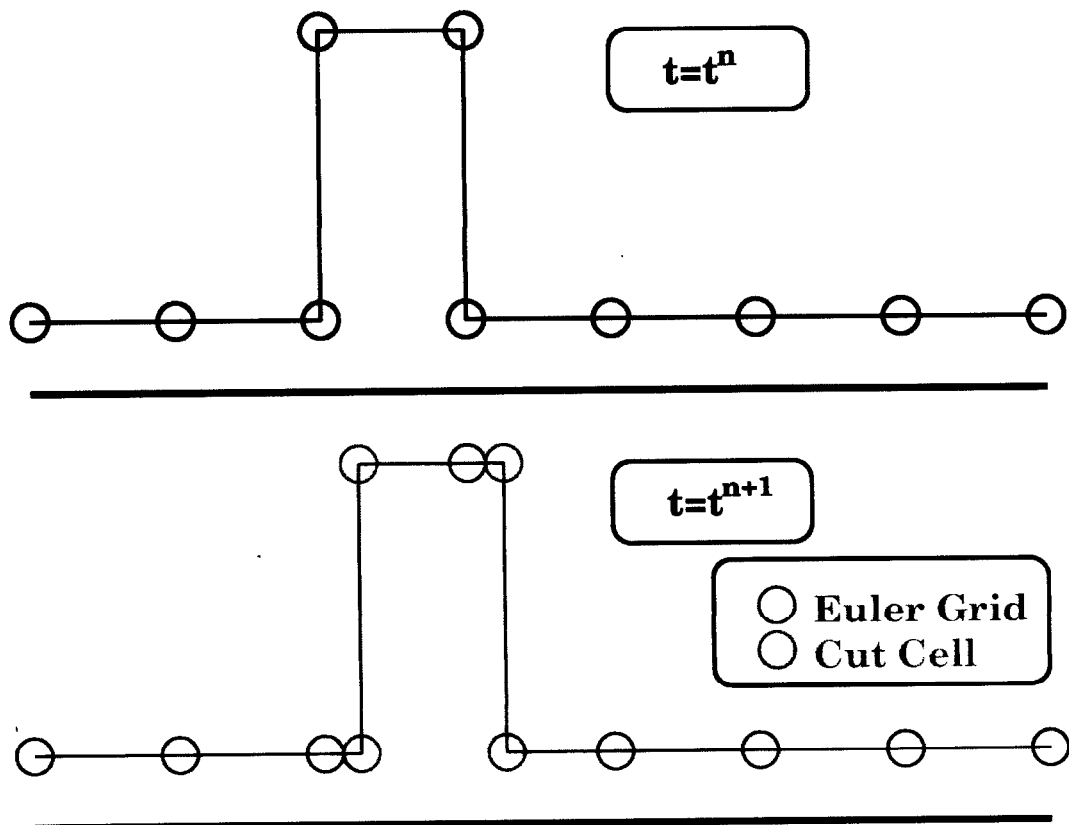


図 6.3: 不連続点にカットセルを適用した 1 次元移流方程式の解法

プロファイルの不連続点にカットセルを適用することにより、オイラーグリッド上であっても適切に不連続面位置を捕らえることが可能になる。また、カットセル上で多価の物理量を定義することでその時間更新も支配方程式に従って行うことが出来る。

例として1次元移流方程式で不連続点のあるプロファイルを解くことを考える。1次精度風上差分法を用いた場合、図6.2の様に計算は進行する。図6.1で説明したように不連続面を持つプロファイルをオイラーグリッド上で表現する場合、図6.3と比較してわかるように初期条件( $t = t^n$ )において少なくとも不連続点を記述できる最大解像度が $\Delta x$ であることからプロファイルの離散化誤差が生じている。1次精度風上差分法を用いて計算を進めるに従い( $t = t^{n+1}$ )プロファイルは正確に移流計算されているにも関わらず離散プロファイルであるためグリッド間情報が欠落してしまい離散情報を補間したプロファイルは更に数値拡散している様子がわかる。更に計算時間を進めるに従いこの様な補間による数値拡散により大幅にプロファイルは減衰する。その一方で、1次風上差分法を用いた場合であってもカットセル手法を不連続面に採り入れると図6.3の様な計算結果を得ることが出来る。

カットセルを不連続点に適用し不連続点上の多価の物理量を同一グリッド上に定義することで不連続点を高精度に表現することが出来る。カットセルの移動はカットセル位置における移流速度を用いた移流計算を行い、またカットセル上に定義された物理量の時間更新はラグランジュ的に記述された支配方程式を解くことで行う。IDO法で支配方程式に従った時間更新を行うため、補間関数を不連続点上で形成する必要がある。風上補間を不連続点上で行うことを考えた場合、カットセル上での移流速度を用いて風上方向を決定することを考えると、多価で定義されている各物理量に対して対象とする物体方向に適切に補間関数を形成しない状況が考えられるため、適切な空間微係数を見積もることが出来ない。そこで、各物理量が属する領域に向かって補間関数を形成し、高次空間微係数を計算する手法を採る。

以上の概念を利用することで、不連続面をオイラーグリッド上で高精度に記述できるほか、不連続点上の物理量更新を支配方程式に従って行うことが可能になる。また不連続点上の物理量に対して異なる物体に向かって急峻な勾配を記述する補間関数を形成しないことから不連続プロファイルをオイラーグリッド上で補間する際に問題となるプロファイルの高波数成分を完全に除去できるので高精度計算スキームにしばしば見られるオーバーシュートやアンダーシュートが低減できると考える。

1次精度風上差分法のような差分スキームでは、カットセル上で物理量のみを考えれば良いが、IDO法は物理量とその空間微係数をグリッド点上に与え、それらの値から高次微係数の計算を行うことから、カットセル上には物理量とその空間微係数を定義しなければならない。また、IDO法の風上補間は通常移流速度に応じて風上方向に対して形成しなければならないにも関わらず、移動カットセル法ではカットセル上の風上補間を領域に応じて形成するため補間関数をの不安定性が生じる可能性がある。以下では1次元移動カットセル手法、2次元カットセル手法をとり上げ、その詳細について説明する。

## 6.2 1次元移動カットセル手法

1次元移動カットセル手法の構築はカットセル上の計算をラグランジュ的手法により行うことで構築することが出来る。ただし、オイラーグリッド上の支配方程式を用いた物理量の時間更新を考えた場合、計算時間刻幅 $\Delta t$ 内にカットセルが通過するオイラーグリッドに対しては $t = t^n$ ,  $t = t^{n+1}$ において属する領域が変化することから数値不安定性が生じることが予想されるため、その扱いに留意しなければならない。

### 6.2.1 カットセルを用いた接触不連続面の定義

本論文で扱う移動カットセルは1つのグリッド上に2個の値を定義できるものとした。カットセルはグリッド線上に定義されることから、小さいグリッド番号から参照される値、大きいグリッド番号から参照される値として区別して定義を行った。多次元問題では各軸方向に対して同様の手続きを行う必要がある。また、カットセル位置を座標値ならびにオイラーグリッドとの相互関係に関しても直接カットセルに定義し、オイラーグリッドにもカットセルとの相互関係を直接記憶させることで検索アルゴリズムの高速化を行っている。

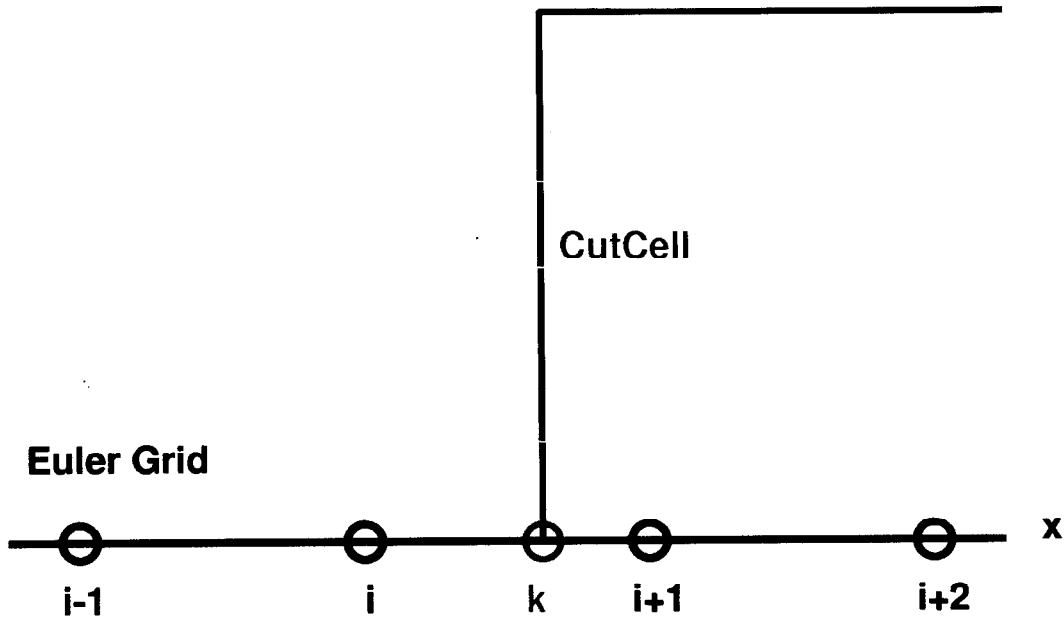


図 6.4: 1次元移動カットセルの構築例

カットセルをオイラーグリッド間に定義することによりオイラーグリッドが持つ空間解像度 ( $\Delta x$ ) 以上の精度でプロファイルを適切に表現することが出来る。また、同じグリッド点上で 2 値の物理量を定義することにより不連続面を高精度に表現することが出来る。

例えば図 6.4 の様なカットセルを含むグリッドでは、カットセルに対して前後のオイラーグリッドが  $i, i+1$  であることとその座標値  $x$  を記憶させると共に、グリッド番号が小さい方の物理量ならびにその空間微係数、グリッド番号が大きい方の物理量ならびにその空間微係数をカットセル上に直接定義する。また、グリッド  $i, i+1$  ではグリッド番号が大きくなる方向に、また小さくなる方向にカットセルが存在することを記憶させるためにフラグを利用し、カットセルが複数存在することを前提にカットセルには連番を用いて識別していることから、カットセルの識別番号を同時にオイラーグリッド上に与える。これらの情報を用いることで、カットセル上の物理量計算は属する領域を新たに識別すること無く、グリッド番号の減少、増大方向を利用して補間関数を形成することにより行うことが可能となる。

### 6.2.2 接触不連続面上の補間関数の形成

カットセルを含むオイラーグリッド上で空間微係数計算のための補間関数を形成する場合、図 6.5 に示すようにカットセル近傍点において非等間隔グリッドとなる。そこで、非等間隔グリッドにおける 5 次中心補間関数形成手法について述べる。

図 6.5 に示すように、カットセルを含む補間関数では、計算対象点の左右でグリッド間隔が異なるため、非等間隔グリッドのための補間関数を形成しなければならない。風上補間では 1 セル内に補間関数を形成するのでこれまでの補間関数と変わらず、3 次多項式で、以下のように表される。

$$F(x) = ax^3 + bx^2 + f_{x,i}x + f_i \quad (6.1)$$

ここで、添字の  $i$  はグリッド番号を表し、 $f_x$  は、物理量  $f$  の 1 階空間微分値を表す ( $f_x = \partial f / \partial x$ )。例えば、グリッド  $i-1$  からグリッド  $i$  にかけて補間関数を形成した場合の係数  $a, b$  はそれぞれ適合条件  $F(-\Delta x) = f_{i-1}, F_x(-\Delta x) = f_{x,i-1}$  を用いることで、以下のように求められる。

$$a = -3 \frac{f_i - f_{i-1}}{\Delta x^2} + \frac{f_{x,i-1} + 2f_{x,i}}{\Delta x}$$

$$b = -2 \frac{f_i - f_{i-1}}{\Delta x^3} + \frac{f_{x,i-1} + f_{x,i}}{\Delta x^2}. \quad (6.2)$$

次に5次多項式を用いた中心補間を非等間隔グリッド上に適用することを考える。補間関数

$$F(x) = ax^5 + bx^4 + cx^3 + dx^2 + f_{x,i}x + f_i \quad (6.3)$$

の未知係数  $a, b, c, d$  はそれぞれ、

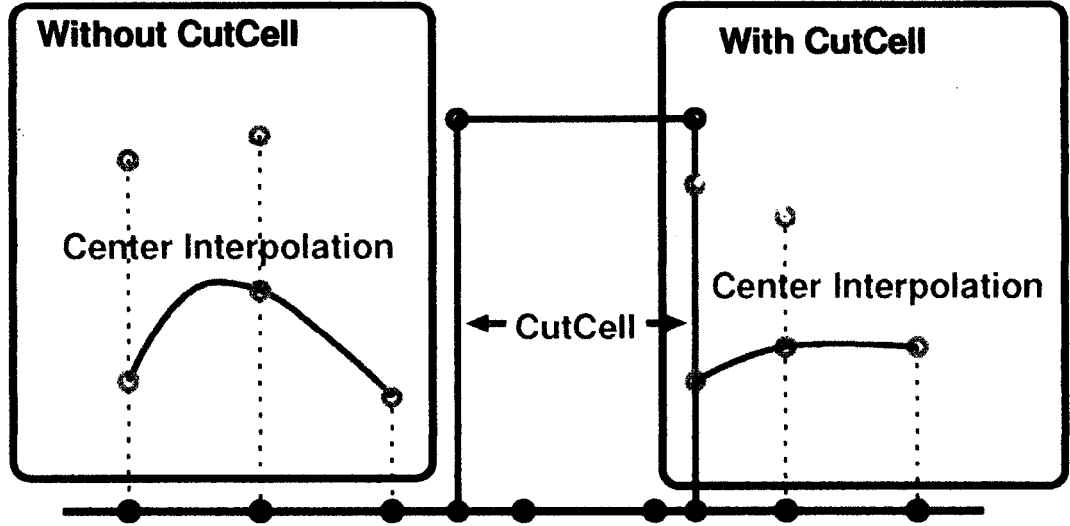


図 6.5: カットセルを含むオイラーグリッド上での補間関数

カットセル近傍のオイラーグリッドならびにカットセル上での補間関数は非等間隔グリッドとなるので非等間隔用の補間関数を形成しなければならない。

$$a = \left[ 2 \left\{ f_{i-1} \Delta x_p^3 (2\Delta x_m + \Delta x_p) + f_i (\Delta x_m - \Delta x_p) \Delta x_m \Delta x_p^3 - f_{i+1} \Delta x_m^3 (\Delta x_m + 2\Delta x_p) \right\} \right. \\ \left. + \left\{ f_{x,i+1} \Delta x_m^3 \Delta x_p (\Delta x_m + \Delta x_p) + f_{x,i} \Delta x_m \Delta x_p (\Delta x_m^3 + 3\Delta x_m^2 \Delta x_p + 3\Delta x_m \Delta x_p^2 + \Delta x_p^3) \right. \right. \\ \left. \left. + f_{x,i-1} \Delta x_m \Delta x_p^3 (\Delta x_m + \Delta x_p) \right\} \right] / \left\{ \Delta x_m^3 \Delta x_p^3 (\Delta x_m + \Delta x_p)^3 \right\} \quad (6.4)$$

$$b = \left\{ f_i \Delta x_m \Delta x_p^3 (4\Delta x_m^2 - 7\Delta x_m \Delta x_p + 4\Delta x_p^2) + f_{i+1} \Delta x_m^3 (-4\Delta x_m^2 - 5\Delta x_m \Delta x_p + 5\Delta x_p^2) \right. \\ \left. + f_{i-1} \Delta x_p^3 (5\Delta x_m^2 - 5\Delta x_m \Delta x_p - 4\Delta x_p^2) + f_{x,i+1} \Delta x_p \Delta x_m^3 (2\Delta x_m^2 + \Delta x_m \Delta x_p - \Delta x_p^2) \right. \\ \left. + 2f_{x,i} \Delta x_p \Delta x_m (\Delta x_m - \Delta x_p) \Delta x_m \Delta x_p^3 + f_{x,i-1} \Delta x_m \Delta x_p^3 (\Delta x_m - 2\Delta x_p) (\Delta x_m + \Delta x_p) \right\} \\ / \left\{ \Delta x_m^3 \Delta x_p^3 (\Delta x_m + \Delta x_p)^3 \right\} \quad (6.5)$$

$$c = \left\{ -2f_{i+1} \Delta x_m^4 (\Delta x_m^3 \Delta x_p - 5\Delta x_p^2) + 2f_i \Delta x_m \Delta x_p^3 (\Delta x_m^3 - 4\Delta x_m^2 \Delta x_p + 4\Delta x_m \Delta x_p^2 - \Delta x_p^3) \right. \\ \left. - 2f_{i-1} \Delta x_p^4 (5\Delta x_m^2 + \Delta x_m \Delta x_p - \Delta x_p^2) - f_{x,i+1} \Delta x_m \Delta x_m^3 \Delta x_p (2\Delta x_p - \Delta x_m) (\Delta x_p + \Delta x_m) \right. \\ \left. + f_{x,i} \Delta x_m^2 \Delta x_p^4 (\Delta x_m^2 - 4\Delta x_m \Delta x_p + \Delta x_p^2) - f_{x,i-1} \Delta x_m \Delta x_p^4 (2\Delta x_m - \Delta x_p) (\Delta x_m + \Delta x_p) \right\} \\ / \left\{ \Delta x_m^3 \Delta x_p^3 (\Delta x_m + \Delta x_p)^3 \right\} \quad (6.6)$$

$$d = (f_{i+1} \Delta x_m^4 (3\Delta x_m + 5\Delta x_p) - f_i \Delta x_m \Delta x_p^3 (3\Delta x_m^2 - 4\Delta x_m \Delta x_p + 3\Delta x_p^2) + f_{i-1} \Delta x_p^4 (5\Delta x_m + 3\Delta x_p) \\ - f_{x,i+1} \Delta x_m^4 \Delta x_p (\Delta x_m + \Delta x_p) - 2f_{x,i} \Delta x_m^2 \Delta x_p^4 (\Delta x_m - \Delta x_p) + f_{x,i-1} \Delta x_m \Delta x_p^4 (\Delta x_m + \Delta x_p)) \Delta x_m \Delta x_p$$

$$\left/ \left\{ \Delta x_m^3 \Delta x_p^3 (\Delta x_m + \Delta x_p)^3 \right\} \right. \quad (6.7)$$

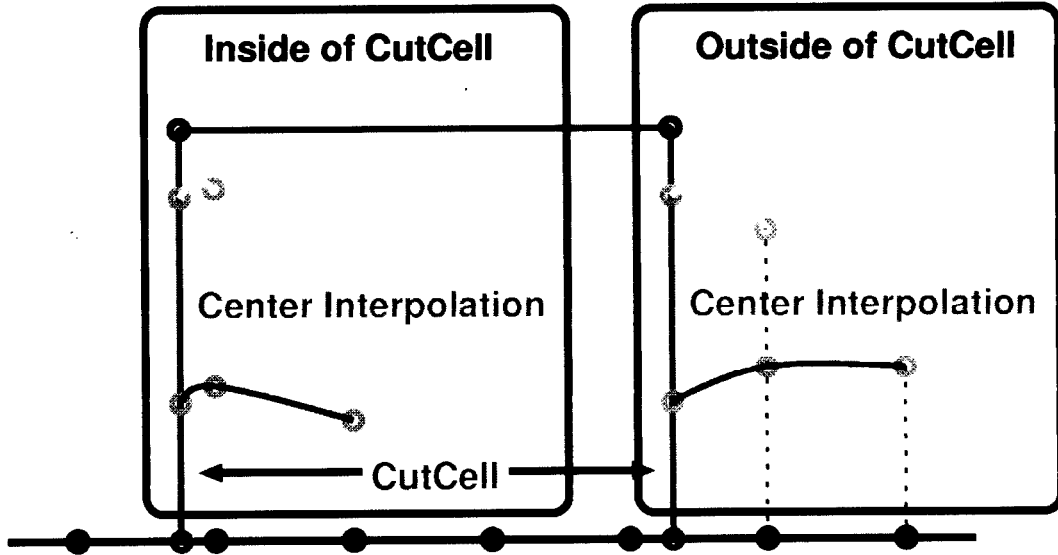


図 6.6: 1次元移動カットセル上での補間関数

カットセル上での補間関数は各領域の方向に向かって補間関数を形成する。カットセルは、オイラーグリッド間に定義されるためカットセルとオイラーグリッドとの間隔は計算ステップ毎に変動することから、カットセル上では非等間隔グリッドに適応した補間関数を用いる必要がある。

と表され、高次空間微係数は

$$\frac{\partial^5 f}{\partial x^5} = 120a, \quad \frac{\partial^4 f}{\partial x^4} = 24b, \quad \frac{\partial^3 f}{\partial x^3} = 6c, \quad \frac{\partial^2 f}{\partial x^2} = 2d \quad (6.8)$$

で見積もられる。ここで、前後のグリッド間隔を

$$\Delta x_m = x_i - x_{i-1}, \quad (6.9)$$

$$\Delta x_p = x_{i+1} - x_i \quad (6.10)$$

とおいた。カットセル上での補間関数は 図 6.6 に示されるようにカットセルによって分割された領域方向に補間関数を形成する。

### 6.2.3 接触不連続面の移動

本論文では対象とする不連続面を接触不連続面 (Contact Discontinuity) とする。このことから、密度プロファイルは不連続を持つが、移流速度は不連続点を持たない。そこで、不連続面 (カットセル) を移動する手法として近傍のオイラーグリッド点上に定義されたい流速度ならびにその空間微係数を用い、カットセル位置における移流速度を 3 次エルミート補間を用いて計算しその値によりラグランジュ手法に基づいた時間前進を行うことを提案する。

時間 1 次精度での移動計算では、次時刻のカットセル位置 ( $x^{n+1}$ ) は現在のカットセル位置 ( $x^n$ ) を用いて

$$x^{n+1} = x^n + u \Delta t \quad (6.11)$$

と求めることができる。時間 2 次精度で次時刻のカッセル位置を計算する場合は

$$x^{n+1} = x^n + u\Delta t + \frac{1}{2} \frac{\partial u}{\partial t} \Delta t^2 \quad (6.12)$$

で表されるテーラー展開を用いる。

$$\frac{\partial u}{\partial t} = \frac{\partial x}{\partial t} \frac{\partial u}{\partial x} = uu_x \quad (6.13)$$

の関係式を用いることで、

$$x^{n+1} = x^n + u\Delta t + \frac{1}{2} uu_x \Delta t^2 \quad (6.14)$$

が導かれる。同様にして、時間 3 次精度では

$$x^{n+1} = x^n + u\Delta t + \frac{1}{2} uu_x \Delta t^2 + \frac{1}{6} \left\{ u \left( \frac{\partial u}{\partial x} \right)^2 + u^2 \frac{\partial^2 u}{\partial x^2} \right\} \Delta t^3 \quad (6.15)$$

と表され、カッセル位置における速度の 2 階空間微係数が必要となることがわかる。

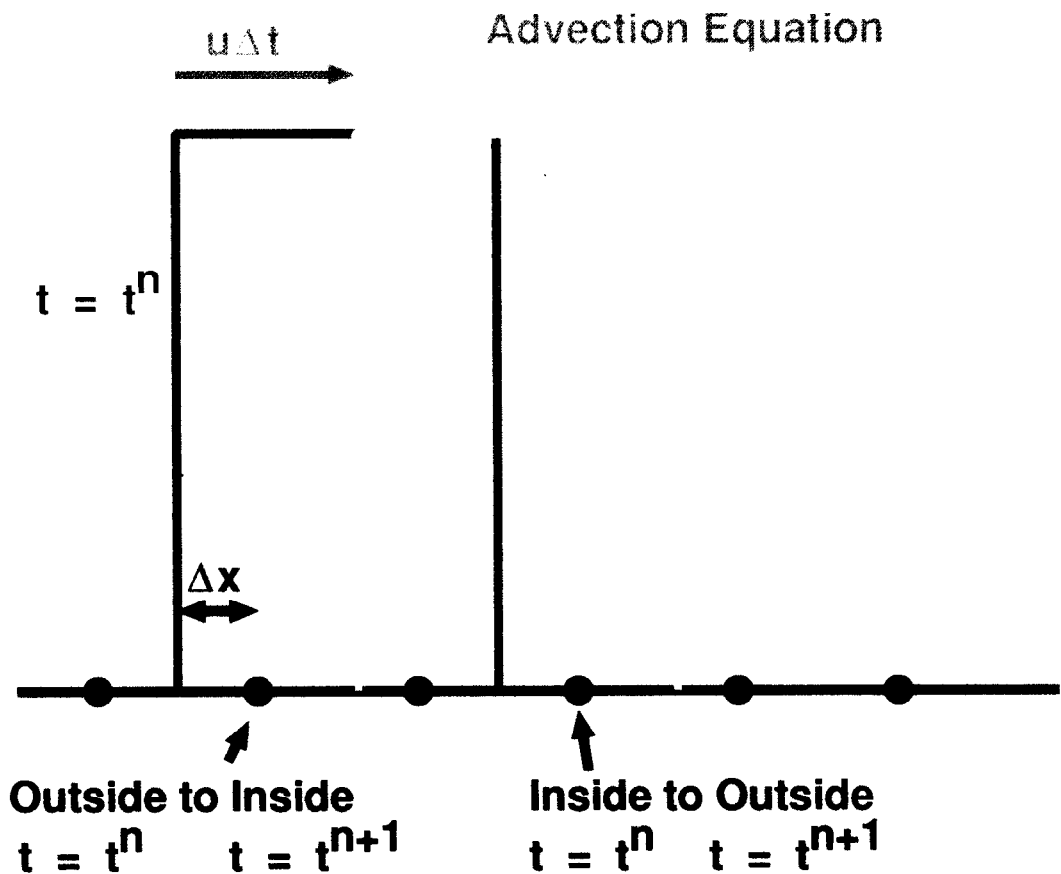


図 6.7: カッセル通過に伴うオイラーグリッド上の数値不安定性

カッセルが通過することでオイラーグリッド上の値が属する領域が変化する場合、 $t = t^n$  において支配方程式を時間積分することによって見積もられた値が  $t = t^{n+1}$  において期待される値とは異なるため、数値不安定性が生じる。

### 6.2.4 局所 CFL 数による数値不安定性

1次元移動カットセルは、オイラーグリッド上をラグランジュ的に移動するため近傍にカットセルが存在するオイラーグリッドでは CFL 数による数値不安定性に注意しなければならない。図 6.7 に示すように、時刻  $t = t^n$  では不連続面に囲まれた高密度領域に属するオイラーグリッド点が  $t = t^{n+1}$  において、境界面の移動により低密度領域に属する場合を考える。陽解法を用いた数値計算では時刻  $t = t^n$  において与えられている物理量を用いて補間関数を形成し、支配方程式に従って次時刻の値を計算させる手法をとることは明らかである。この考えに基づき先のオイラーグリッド点上の値更新を考えた場合、時刻  $t = t^n$  では高密度領域に属しているためそこで形成された補間関数を用いた時間更新から得られる次時刻  $t = t^{n+1}$  における値は高密度領域側の値を示すことは容易に想像できる。しかし、不連続面 (カットセル) の通過により対象オイラーグリッドは時刻  $t = t^{n+1}$  において低密度領域に属するため、得られた値は矛盾を持つ。この問題を解消するために、カットセルがオイラーグリッドに対して

$$u\Delta t > \Delta x \quad (6.16)$$

を満たす場合、オイラーグリッド点では基礎方程式からの時間発展を行わないことにする。ここで、 $\Delta x$  は、オイラーグリッドとカットセルの距離をあらわしている (図 6.7)。計算を行わなかったグリッドの  $t = t^{n+1}$  における値は陰解法的に解を求めることも可能であるが、他の全てのオイラーグリッド上ならびにカットセル上に定義された物理量の時間発展終了後に自身を中心として計算軸の両側の点を用いた 3 次エルミート補間を形成して値を計算する。

### 6.2.5 1次元移動カットセル法のアルゴリズム

以上を元に 1次元移動カットセル法の IDO 法での計算の流れを示す。

1. カットセルの定義 (Initialize)
2. CFL 数による安定性判定
3. オイラーグリッド上の高次微係数の計算
4. オイラーグリッド上の物理量計算と時間更新
5. カットセル上の高次微係数の計算
6. カットセル上の物理量計算 (ラグランジュ法) と時間更新
7. カットセルの移動
8. CFL 条件により除かれたオイラーグリッド上の値を補間

### 6.2.6 1次元計算例題

#### 1次元移流方程式への適用

簡単な例題の 1 つとして 1次元移流方程式

$$f_t = -uf_x \quad (6.17)$$

への適用を考える。移動カットセル上の物理量時間更新は 1次元移流方程式をラグランジュ表記に変換した

$$f_t = 0 \quad (6.18)$$

によって行われる。



計算空間 ( $0 \leq x \leq 2, \Delta x = 2/100$ ) に初期プロファイルを  $f=1(0.4 \leq x \leq 0.6)$ ,  $f=0$  (その他の点) と与え、 $x = 0.4, 0.6$  にそれぞれカットセル定義することで不連続面を移動カットセルにより記述した。移動カットセル上のプロファイルに対する空間微係数は計算アルゴリズムを考慮して、 $f_x = 0$  と定義した。移動カットセルグリッドの移動速度を検証するため、空間で一様な速度場  $u(x) = 1$  を用いる。時刻  $t = 1$  まで、CFL 数 ( $u\Delta t/\Delta x = 0.1$ ) を用いて計算した結果を図 6.8 に示す。

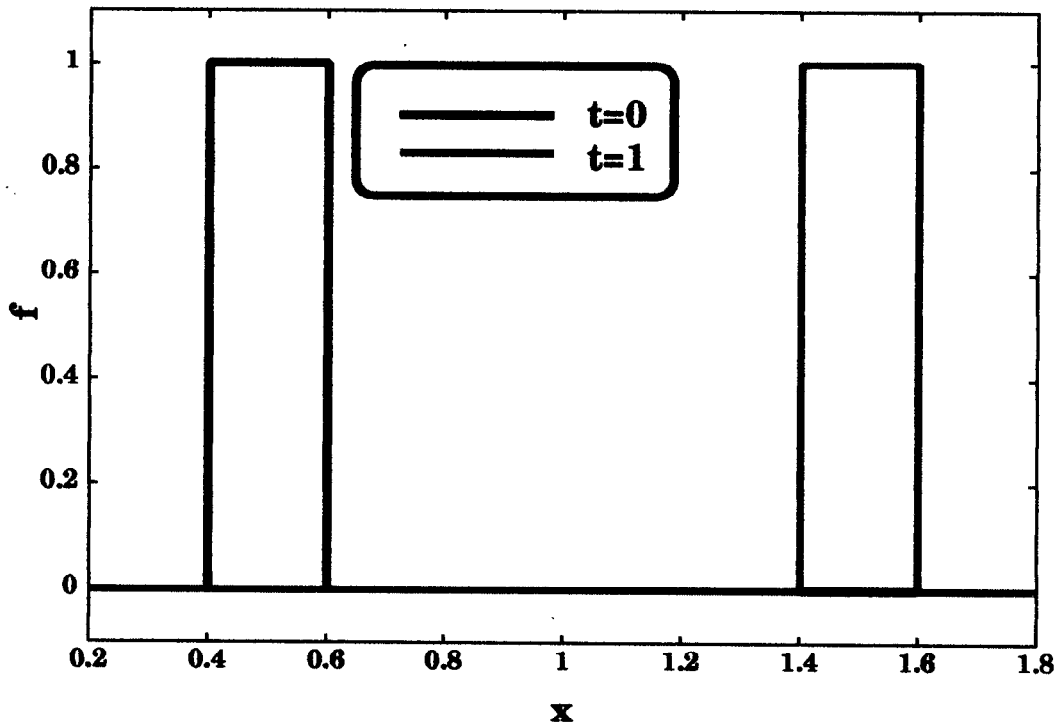


図 6.8: 移動カットセル手法を適用した 1 次元移流方程式の数値計算結果

移動カットセル手法を適用することにより、初期プロファイルに含まれる不連続点を高精度で表現できた。移流速度  $u(x) = 1$  を用いてプロファイルの移流計算を行った結果も不連続点を数値拡散無しで表現できており、またその移流速度も解析解と完全に一致した結果を得た。

図 6.8 の黒の実線で表された初期プロファイルに含まれる不連続点をカットセル手法を用いることにより高精度で表現できている。赤の実線で表された計算結果についても不連続面にいかなる数値拡散誤差も見られず、高精度な不連続面記述を達成している。また、移流速度  $u(x) = 1$  で、プロファイルを  $t = 1$  まで計算したことから、その移動距離は解析的に 1 と求まり、計算結果は解析解と完全に一致した結果であるといえる。以上から、移動カットセル手法は一様定常速度場における 1 次元移流方程式に対して数値拡散誤差 0 で数値解を得ることが出来る手法であるといえる。

次に、物理量の不連続に加え、空間微係数の不連続を持つ Harten により用いられている初期プロファイルについて 1 次元移流方程式に対する有効性を検証する。その初期プロファイルは支配方程式

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \quad (6.19)$$

に対して計算空間  $-1 \leq x < 1$  において

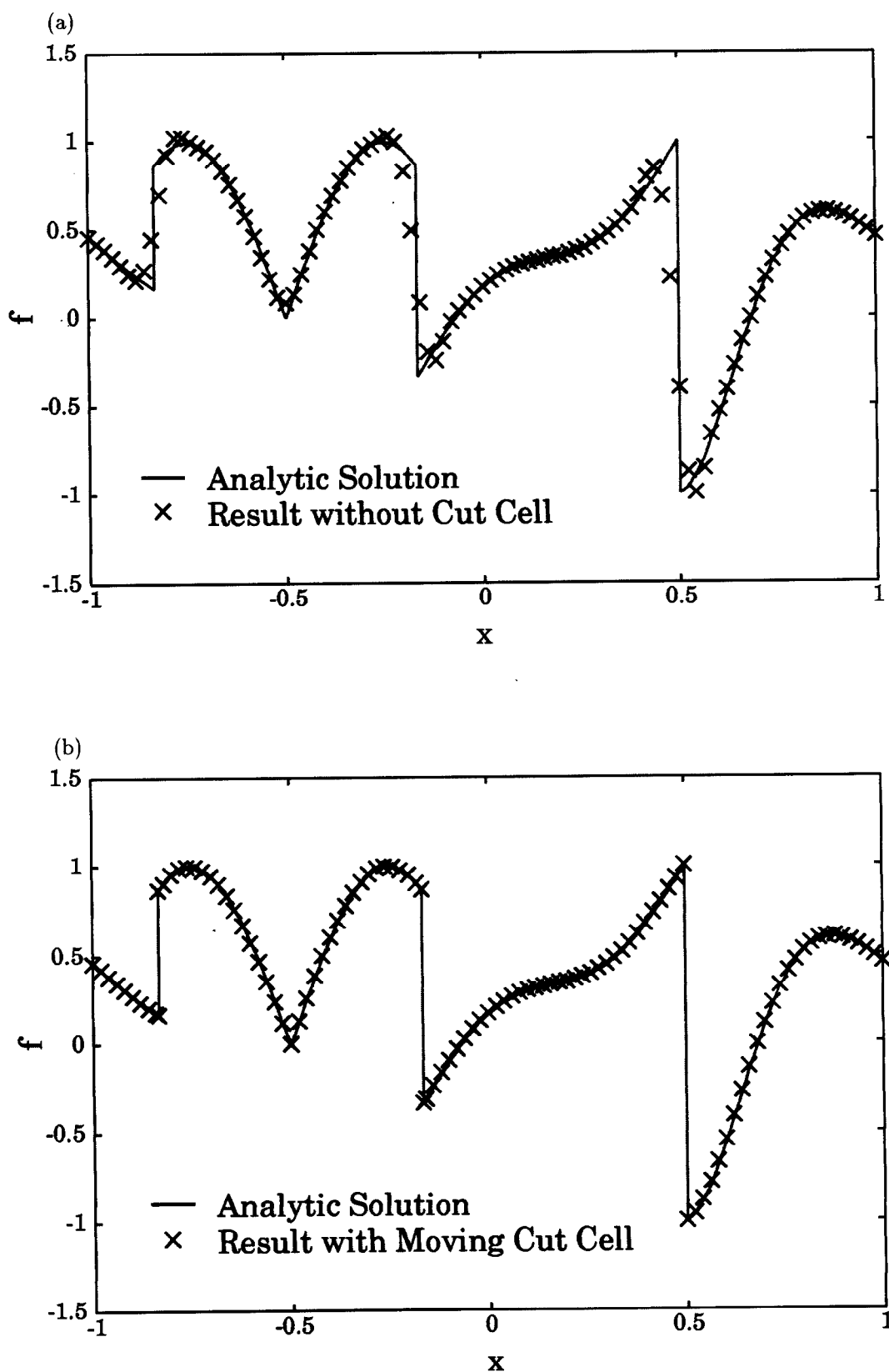


図 6.9: 移動カットセル手法による Harten 例題の検証:

IDO 法は、物理量の空間微係数を物理量とは独立に計算格子点上に与えることができることから、物理量不連続点に加え、空間微係数不連続点を持つ Harten 例題に対して移動カットセル法を導入することで高精度解を与えることができる。

$$u^0(x+0.5) = \begin{cases} -x \sin(\frac{3}{2}\pi x^2) & -1 \leq x < -\frac{1}{3} \\ |\sin(2\pi x)| & |x| < \frac{1}{3} \\ 2x - 1 - \sin(3\pi x)/6 & \frac{1}{3} < x < 1 \end{cases} \quad (6.20)$$

で与えられる。物理量の不連続点を  $x = -1, -1/3, 1/3$  において持ち、その空間微係数の不連続点を  $x = 0$  においてもつ。このことから、移動カットセルを  $x = -1, -1/3, 0, 1/3$  に導入し、周期境界条件下において  $\Delta t = 0.1\Delta x$  を用いて  $t = 2$  まで計算した結果を図 6.9 に示す。対照計算として移動カットセル手法を導入しない IDO 法により計算した結果を併せて示す。移動カットセル手法を用いていない場合でも ENO スキームなどに代表される高精度スキームと同等の計算結果を得ることができた。その一方で、移動カットセル手法を導入した IDO 法による数値計算結果から物理量不連続ならびに空間微係数不連続を正確に移流させることが可能であることを示した。

### 1 次元保存方程式への適用

移動カットセルグリッド上での値が変動する例題として 1 次元保存方程式

$$f_t = -uf_x - u_x f \quad (6.21)$$

への適用を考える。移動カットセル上では空間で非一様な速度場に対しての数値計算を行うため、物理量ならびに空間微係数に対してそれぞれ 1 次元保存方程式をラグランジュ表記に変換した

$$f_t = -u_x f \quad (6.22)$$

$$f_{tx} = -2u_x f_x - u_{xx} f \quad (6.23)$$

を用いて時間更新を行う。

計算空間 ( $0 \leq x \leq 2, \Delta x = 2/100$ ) に初期プロファイルを

$$f^0 = \begin{cases} 0.5 & x \leq 0.4 \\ 1 & 0.4 \leq x \leq 0.6 \\ 0.1 & 0.6 \leq x \leq 2 \end{cases} \quad (6.24)$$

と与え、 $x = 0.4, 0.6$  にそれぞれカットセルを定義し不連続点を移動カットセルにより記述した。移動カットセル上の初期プロファイルに対する空間微係数は計算アルゴリズムを考慮して、 $f_x = 0$  と定義した。解析解が容易に求まるように速度場を  $u(x) = 1/(1 + 0.5 \sin(2\pi x))$  で与えた。解析解は第 1 章で用いた手法と同様に、式 (2.77) ~ 式 (2.86) を使うことで求めた。

移動カットセル手法を適用しない IDO 法による数値計算結果と移動カットセル手法を適用した IDO 法による数値計算結果を図 6.10 に示す。IDO 法による数値計算結果も不連続面をただか数グリッド程度で表現できており、高精度計算を達成しているものの、移動カットセル手法を適用することにより、有限次多項式による不連続面記述により生じる数値拡散誤差を完全に抑制できていることがわかる。カットセル上の物理量計算に関しても風上補間、中心補間を物体の領域を補間関数形成方向の決定要因とし、一方向的に補間関数形成を行っているものの、オーバーシュートやアンダーシュートなどの数値不安定性無しに計算できている様子がわかる。解析解と比較してもわかるように、不連続面記述のために導入された移動カットセル上のラグランジュ表記による物理量計算は高精度で行えている。また、不連続面位置の比較を行うことで、移動カットセルの移流計算が非一様速度場においても高精度に行えていることを示した。

移動カットセル手法を適用した 1 次元保存方程式の計算結果から 1 次元移動カットセル手法は 1 次元プロファイルの持つ不連続点を高精度に記述すると共にその移流速度や、不連続点上の物理量計算を高精度に行える手法であることを明らかにした。

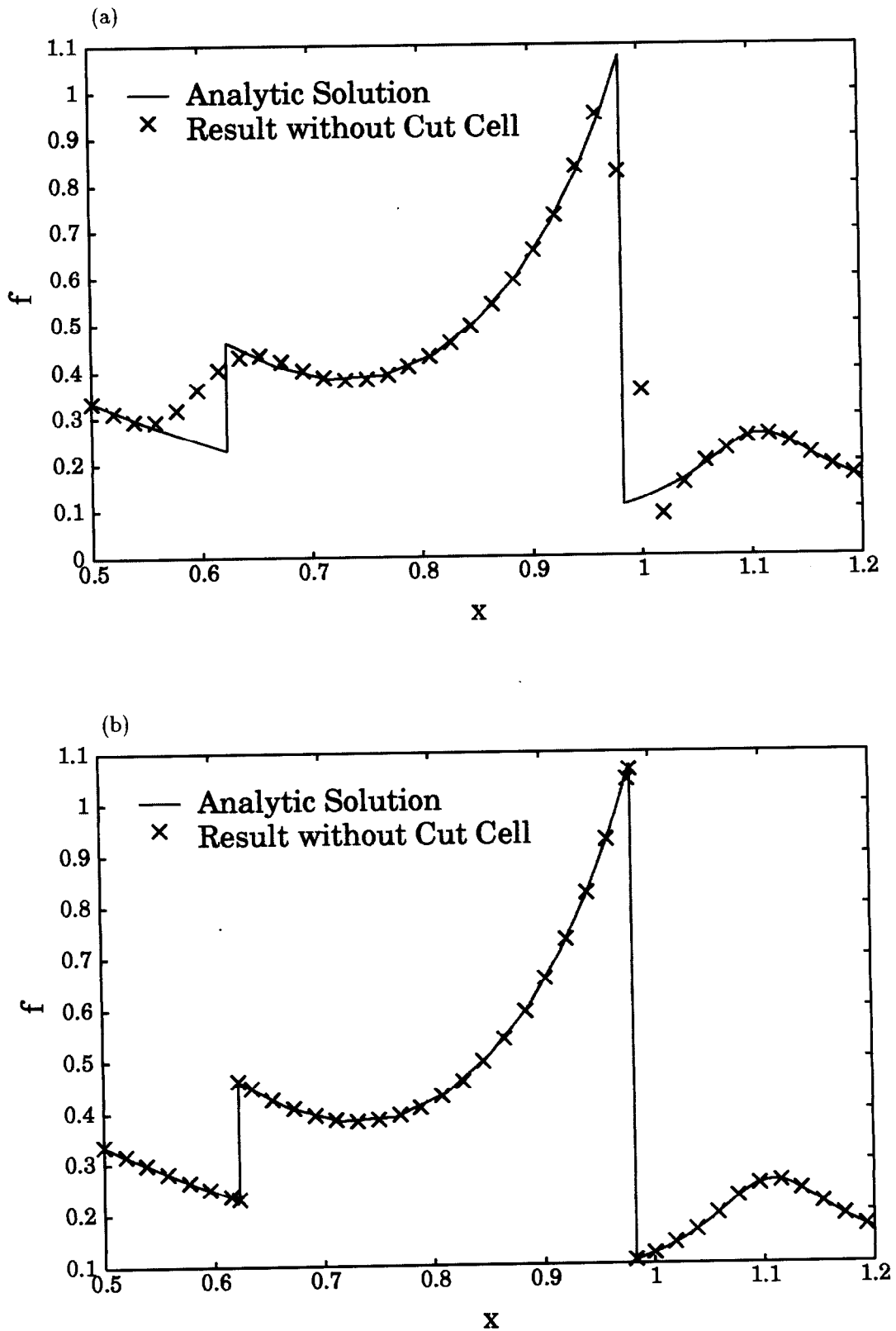


図 6.10: 移動カットセル手法を適用した 1 次元保存方程式の計算結果  
 移動カットセル手法を適用しない IDO 法による数値計算結果も不連続面を数グリッド程度で表現出来ているが、移動カットセル手法を適用することにより不連続面記述のための数値拡散やオーバーシュート、アンダーシュートを完全に抑制した結果を得た。

## 6.3 2次元移動カットセル手法

多次元空間におけるカットセルの導入は、図 6.1 に示すようにベースとなるオイラーグリッド線とプロファイルの不連続面が交わる位置にマーカーとして定義することで行う。このため、移動カットセル手法の多次元化において、1次元移動カットセル手法で用いたラグランジュ的手法を移動カットセルの位置計算ならびに移動カットセル上の物理量の時間発展に用いた場合、次時刻 ( $t = t^{n+1}$ ) におけるカットセル位置がグリッド上にならないため、問題が生じる。そこで、移動カットセルの位置計算を境界面の時間発展ととらえ、Level Set 関数等の境界面捕獲手法を導入することを提案する。境界面捕獲手法を用いて不連続境界面を適切に時間発展し、境界面とオイラーグリッド線との交点にカットセルを再定義する手法を用いることでカットセルの移動を行うことを可能にし、境界面上での物理量の時間発展をセミ・ラグランジュ的に行う手法を探り入れる。

### 6.3.1 2次元移動カットセル導入に関する方針

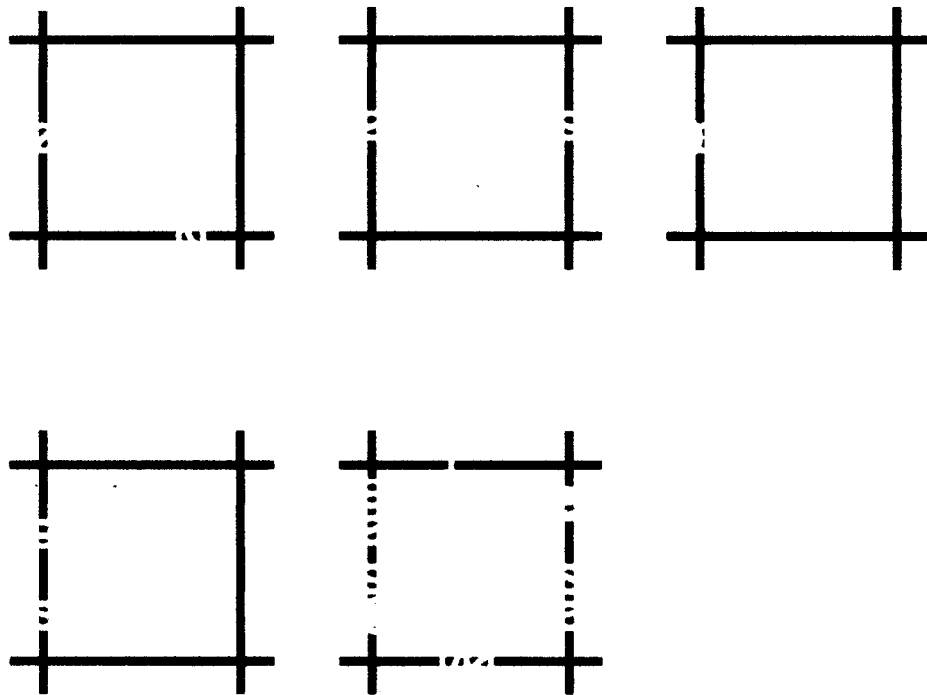


図 6.11: 2次元移動カットセルの定義に関する制限

1 計算セルに含まれる 1 グリッド線上に複数のカットセルを定義することは可能であるが、オイラーグリッド上、カットセル上での物理量計算ならびに LevelSet 関数の構築等を簡略化するため 1 グリッド線上ではカットセルの定義数を 1 とする。

図 6.11 に示すようにカットセルを 1 つのセルグリッド上に多数配置することは可能であるが、グリッド内プロファイルの空間解像度が用いる識別関数の精度に制限され、更にはその形状を的確に把握することは計算負荷などの問題から複数のカットセルの定義が必要となる複雑な境界面への適用手段としてグリッド解像度の増大がより良いアプローチと考え、1 グリッド線上には 1 つ以上のカットセルを定義しないこと

にする。この方針によりカットセルの生成を限定することで LevelSet 関数の初期化や再初期化、カットセルに隣接するオイラーグリッドでの補間関数の形成、カットセル移流によるオイラーグリッド点の数値不安定性判定などの計算アルゴリズム、カットセルの生成時における物理量計算を簡略化することができる。

ただし、この方針を3次元問題に適用した場合、計算機メモリーの状態により2次元問題程度にオイラーグリッド点を用いることができないことから、境界面記述精度が低下する恐れがある。しかしその一方でオイラーグリッドのみを用いて境界形状を扱う限界 ( $\Delta x$ ) に比べ高精度で記述できることは保証される。

### 6.3.2 LevelSet 関数構築のためのアルゴリズム

初期に物体の形状が何らかの関数により記述できる場合、LevelSet 関数はその境界面からの距離を計算することにより容易に形成される。しかし、複雑な境界面の導入を考えた場合 Level Set 関数を構築することが困難になる。そこで、境界面として与えられた離散化されたカットセルから疑似的にその表面形状を3D CG (Computer Graphics) で用いられているポリゴンを形成し、その Level Set 関数を構築することを考える。

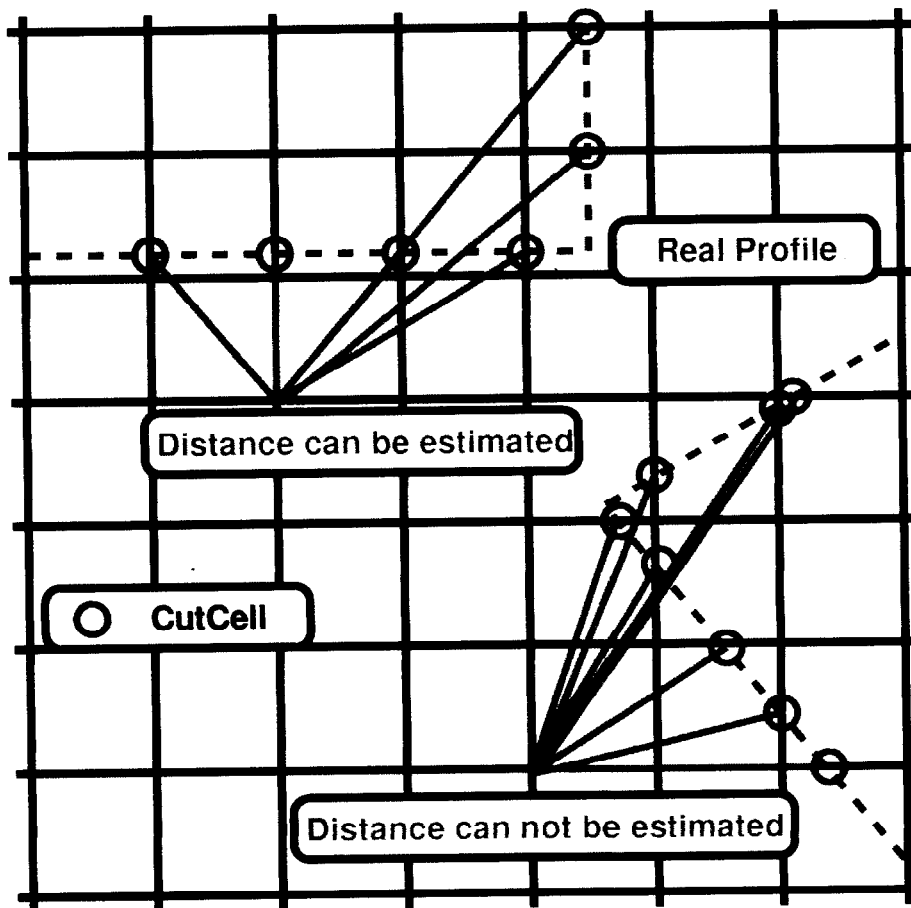


図 6.12: 移動カットセル手法における LevelSet 関数の初期化

移動カットセル手法ではカットセルによって不連続面位置を的確に捕獲していることから、各オイラーグリッド点上において、カットセルによって表現される境界面からの距離を計算することにより疑似的に初期化を行うことが出来る。

LevelSet 関数はそれぞれが属する領域内で負の値を探る。2次元 LevelSet 関数の構築と3次元 LevelSet 関数の構築とはその難しさが大幅に異なるため、2次元のものについて説明する。カットセルは離散的に存在し、それらから距離関数を作成するため、離散点からの距離を計算し最短のものを選択する方法を考えることが出来るが、グリッドに対して直交ないしは平行な形状に対する LevelSet 関数を構築する場合に限りこの手法が正しいといえる(図 6.12)。

しかし、一般的にプロファイルが必ずグリッドに対して平行であるとは限らない。計算対象オイラーグリッド点により近い2ないし3点を求め、何通りかの線分を作成し、それらからの距離の内、最短のものを LevelSet 関数の値として与えることを考えたが、凹型形状を扱う場合、間違った LevelSet 関数を形成する場合があるため、そのような手法を採り入れることが出来ない。そこで、セル毎に距離関数を計算することを考える。先に決めたカットセル導入方針からカットセルを2点以上含むセルのグリッド線上では必ず対する格子点上では異なる物質である。そこで、カットセルを2点以上含むグリッド内でカットセルを各領域を考慮しながら接続することで形成される線分から LevelSet 関数計算点への距離を計算し、その中で最も短いものを LevelSet 関数値として与える手法を採り入れる(図 6.13)。

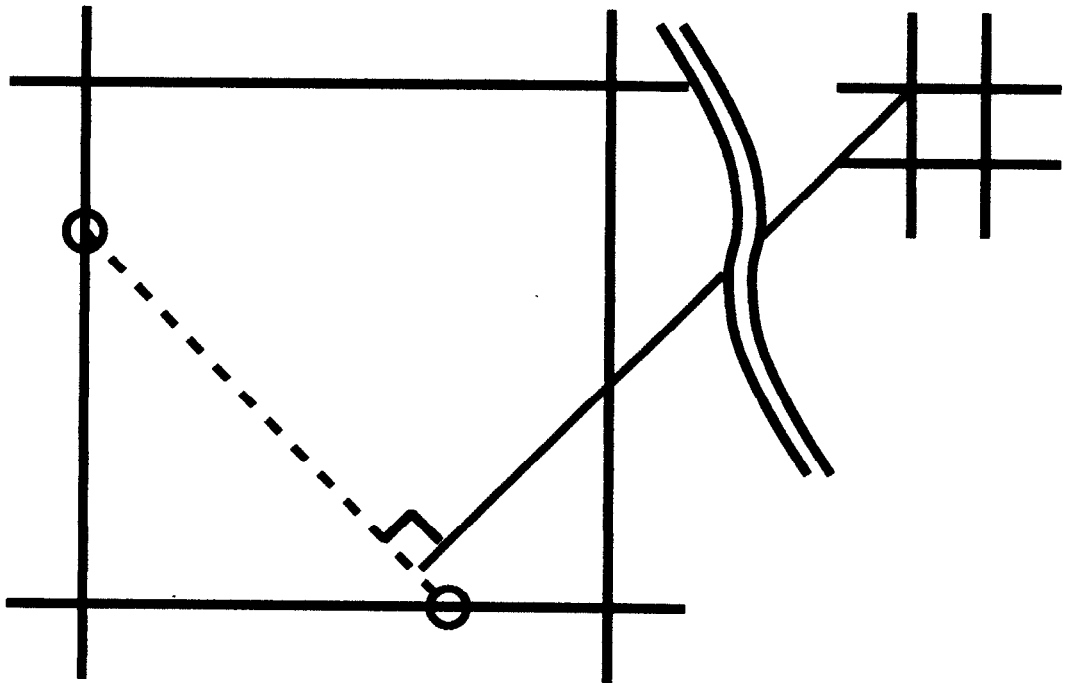


図 6.13: 2次元移動カットセル手法における LevelSet 関数の初期化

移動カットセル手法では不連続面位置にカットセルを定義していることから、その境界面を離散的に把握することが出来る。このメリットを利用して LevelSet 関数の再初期化では各セルに属するカットセルを領域識別に従って接続した線分を用い、オイラーグリッド点からの距離を計算することで Level Set 関数の初期化を行う。

3次元の LevelSet 関数を生成するためにはポリゴン生成アルゴリズムとして有名な Marching Cube 法 [64, 65] を用いることが考えられる。

また、IDO 法で Level Set 関数を扱う場合、不連続面からの距離値だけでなく、その空間微係数を与える必要がある。2次元オイラー座標系では必要となる3空間微係数  $\partial f/\partial x, \partial f/\partial y, \partial^2 f/\partial x \partial y$  の計算が必要となる。そこで、ラプラス方程式

$$\Delta \Phi = 0 \quad (6.25)$$

の性質を利用することを考える。ラプラス方程式は電磁気学や流体力学での等ポテンシャル面を求める際に良く用いられる方程式である。Level Set 関数が境界面からの距離関数であることからポテンシャル面としての性質を持つと考えることができることからラプラス方程式を収束計算させることで空間微係数を適切に求めることができる可能性がある。IDO 法は 4 従属変数を持つことからラプラス方程式を扱う場合、ポアソン方程式を扱う場合と同様に支配方程式をそれぞれ空間変数  $x, y$  について偏微分した以下の 4 式を連立収束計算させる必要がある。

$$\Delta\Phi = 0, \quad \frac{\partial}{\partial x}\Delta\Phi = 0, \quad \frac{\partial}{\partial y}\Delta\Phi = 0, \quad \frac{\partial^2}{\partial x\partial y}\Delta\Phi = 0 \quad (6.26)$$

すでに距離関数として与えられた物理量に関する方程式に関して収束計算させる必要が無いことから、ラプラス方程式を空間微分した 3 式について連立収束計算させることで各空間微係数を計算する。

### 6.3.3 LevelSet 関数再構築 (Reinitialization) のためのアルゴリズム

境界面が時間と共に変形する状況下で Level Set 関数の移流計算を行う場合、数値誤差や計算領域外からの誤差の流入により境界面記述精度が時間更新と共に低下することが容易に予想される。この様な誤差を低減することを目的として LevelSet 手法では Reinitialization と呼ばれる LevelSet 関数の再構築を

$$\phi_t = \text{sign}(\phi)(1 - |\nabla\phi|) \quad (6.27)$$

を収束計算することで行う。ここで、 $\phi$  は LevelSet 関数を表し、添字の  $t$  は、Level Set 関数再構築のための仮想時間による偏微分を表す ( $\phi_t = \partial\phi/\partial t$ )。  $\text{sign}(\phi)$  は  $\phi$  の正負を返す関数である。式 (6.27) を  $\phi = 0$  を挟む両側のグリッド上の識別関数値を固定し、収束させることで  $|\phi| = 1$  を満足するように LevelSet 関数を再構築することが出来る。移動カットセル法では IDO 法をベースとしていることから、式 (6.27) の右辺から生じる空間微係数を 5 次風上補間から計算して収束計算を行う。

### 6.3.4 2次元接触不連続面の位置更新

1次元問題ではカットセルが必ず  $x$  軸上に存在したため、初期に定義したカットセルはラグランジュ手法により計算時間中消滅、生成させることなく移動させることが出来た。しかし多次元移動カットセルは、図 6.14 に示すように必ずしもカットセルの移動先が計算格子線上となるとは限らないため、カットセルの生成、消滅(再構築)を必要とする。そのような多次元移動カットセルの移流を行うため、以下の手続きを提案する。

1. オイラーグリッド点上の移流速度を用いた Level Set 関数の移流
2. Level Set 関数から得られる不連続面位置へカットセルを構築
3. 前時間ステップのカットセルの破棄

Level Set 関数の移流は境界面を適切に移動することができる速度関数  $F$  を用いて

$$\phi_t = -F\nabla\phi \quad (6.28)$$

で表される移流方程式により時間更新される。2次元移動カットセル手法では簡単のために  $F$  を、オイラーグリッド上で定義されている移流速度を用いた。

Level Set 関数からのカットセルの構築は Level Set 関数値が  $\phi = 0$  となる点を含むグリッド線分上で両側のグリッド点上の値を用いた 3 次エルミート補間を形成し、2 分法により収束計算をさせることでカットセル位置を見積もることとした。カットセルは、直交オイラーグリッドの場合  $x$  軸上、 $y$  軸上に定義されるため、 $x$  方向、 $y$  方向それぞれについてカットセルを求めている。



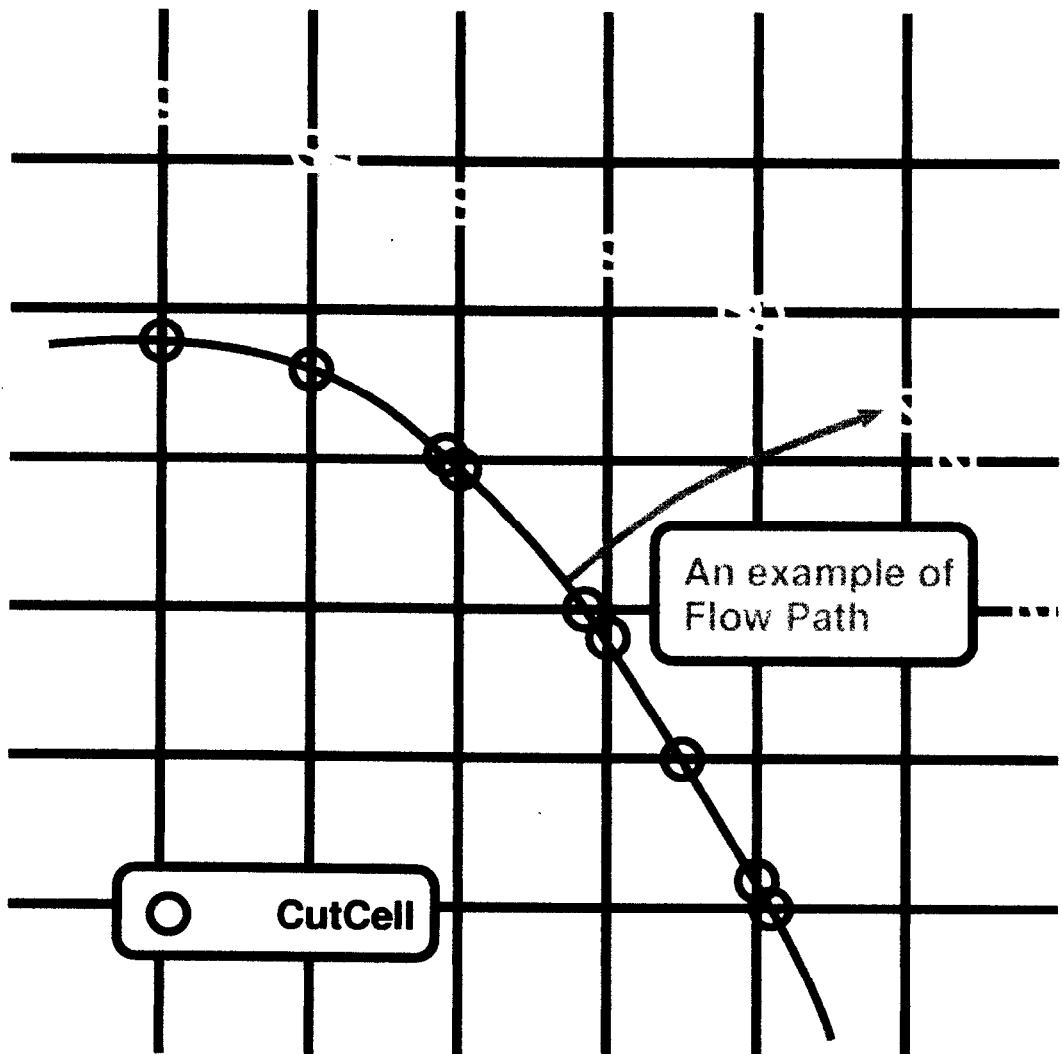


図 6.14: 2次元移動カットセル手法におけるカットセル位置の時間更新

多次元移動カットセル手法では、カットセルの次時刻での位置がグリッド線上に必ずしも位置しないため、1次元移動カットセル手法のようにカットセル位置をラグランジュ的に求めることが出来ないため、Level Set 関数により不連続面を追跡し、Level Set 関数の零点 ( $\phi = 0$ ) と、オイラーグリッド線との交点にカットセルを定義する手法を採る。

### 6.3.5 セミ・ラグランジュ的手法による接触不連続面上の物理量計算

多次元問題で移動カットセル上の物理量計算が困難な理由として時刻  $t = t^{n+1}$  におけるカットセルを時刻  $t = t^n$  におけるカットセルと無関係に作成しなければならないため、 $t = t^{n+1}$  におけるカットセル上の物理量計算のために  $t = t^n$  におけるカットセル上の情報を直接利用できないことが挙げられる。そこで、流体計算をセミ・ラグランジュ的に行うことを提案する。格子線上にカットセルが常に存在する1次元問題では  $t = t^{n+1}$  におけるそれぞれの移動カットセルの移流原点 (Departure point of Flow) が時刻  $t = t^n$  におけるカットセルに一致するため、その値から非移流相積分を行い物理量の時間更新を簡単に行うことが出来る。しかし、多次元問題では  $t = t^{n+1}$  におけるカットセルに対応する移流原点が一般的に格子線上に無いことから、図 6.15 に示すように移流原点位置における物理量を求めることが難しくなる。計算空間中でプロファイルが連続な場合は CIP 法の Type-C の様な補間関数を形成してグリッド間情報を補間す

ることが可能であるが、カットセルを用いた計算空間中ではプロファイルが不連続であるため、周囲の物理量から移流原点位置における物理量を補間するためには各物質中でその境界面を考慮した補間関数を形成する必要があり、座標変換等の問題が伴う。しかし、計算中にカットセルの生成消滅は各時間ステップ毎に不連続領域の大きさに比例して生じる計算であるため、可能な限り低 CPU コストで行えることが望ましい。そこで考える限りシンプルな方法でこの問題にアプローチすることを考える。

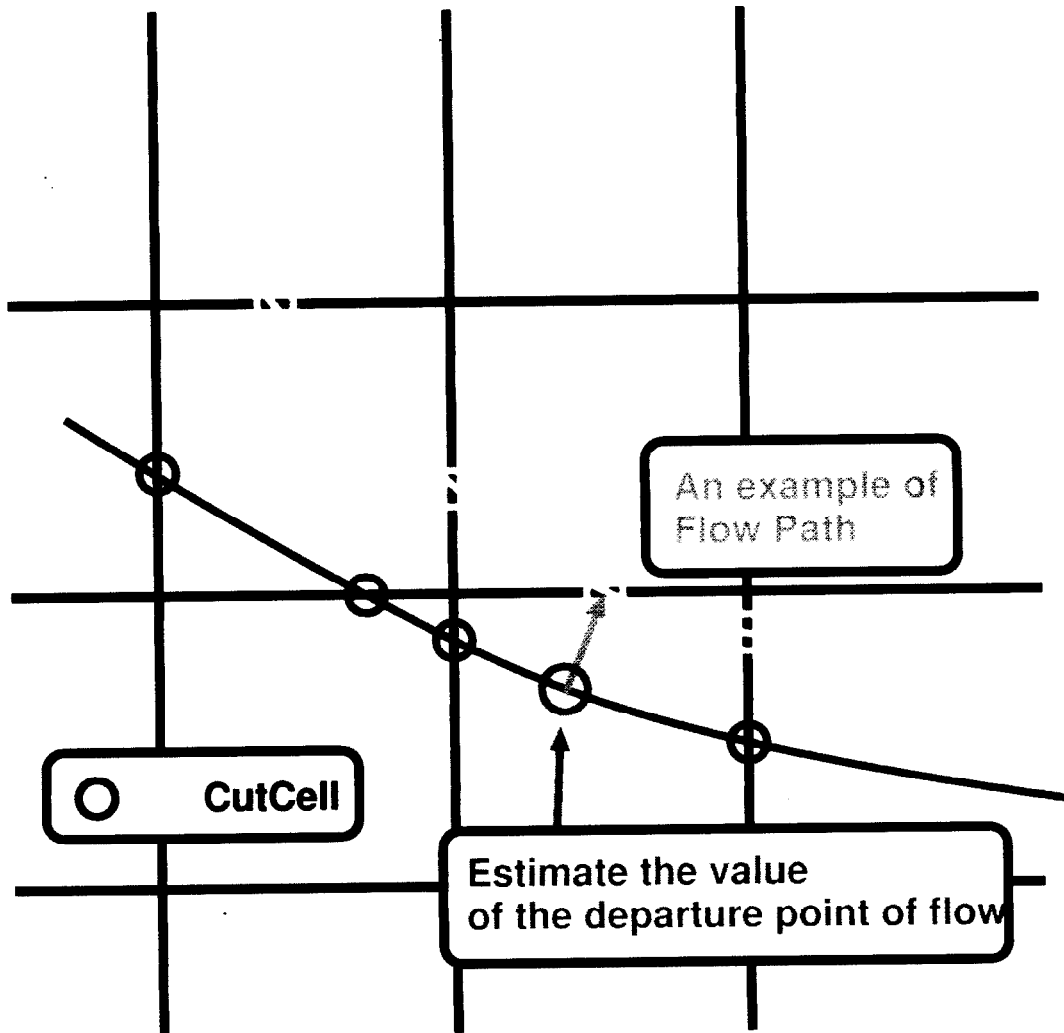


図 6.15: セミ・ラグランジュ的手法による多次元移動カットセル上での物理量計算

移動カットセルの移流原点位置における物理量を周囲の情報から補間し、セミ・ラグランジュ的手法により時間積分することで次時刻のカットセル上の物理量を見積もる。

移流原点位置における物理量の補間精度について考える。LevelSet 関数を用いて不連続面の表現を行うことで高精度に不連続面位置をとらえることが出来ることから、Level Set 関数の境界面記述精度の範囲内で  $t = t^{n+1}$  におけるカットセルに対応する移流原点位置を計算することが可能である。しかし、LevelSet 関数は識別関数としてのみの情報しか持たないため、密度や速度などの物理量を直接 Level Set 関数から得ることが出来ない。そこで、移流原点位置近傍のカットセルないしオイラーグリッド点上の値を用いて移流原点位置の物理量を補間しなければならない。その一方で移動カットセルや、オイラーグリッド上では物理量ならびにオイラーグリッド線との交点位置の情報しか持ち合わせていないため、セル内の不連続面形状プロファイルを高精度で補間することが出来ない。例として図 6.16 の様な状況を考える。

LevelSet 関数で表現される境界面はセル内を複雑に分割しているが、カットセルは、セル内に 2 点しか形成されていない。次時刻のカットセル上の物理量を計算するために移流原点を見積もった結果、図中の青丸のような場所が得られた場合、カットセルならびにオイラーグリッド上の物理量を用いて領域 A、領域 B に対応する物理量を補間により計算するためには、複雑な座標変換 (格子生成) が必要となることは明らかである。また、この様に曲率の大きな境界面形状に対してグリッド内で座標の再構築を行った場合十分な精度が得られるとは考えがたい。そこで、カットセルでの境界面表現限界 (赤の破線) 上に各カットセル位置から移流原点位置までの距離を用いた 1 次元 1 次精度補間を行い内分点として処理する (図 6.17) ことを提案する。ここで、

$$r_a : r_b = R_a : R_b \quad (6.29)$$

である。複雑な補間を行い外挿補間を行うよりも数値安定性に関しては優れていると考える。しかし計算精度が低下することは容易に想像できる。

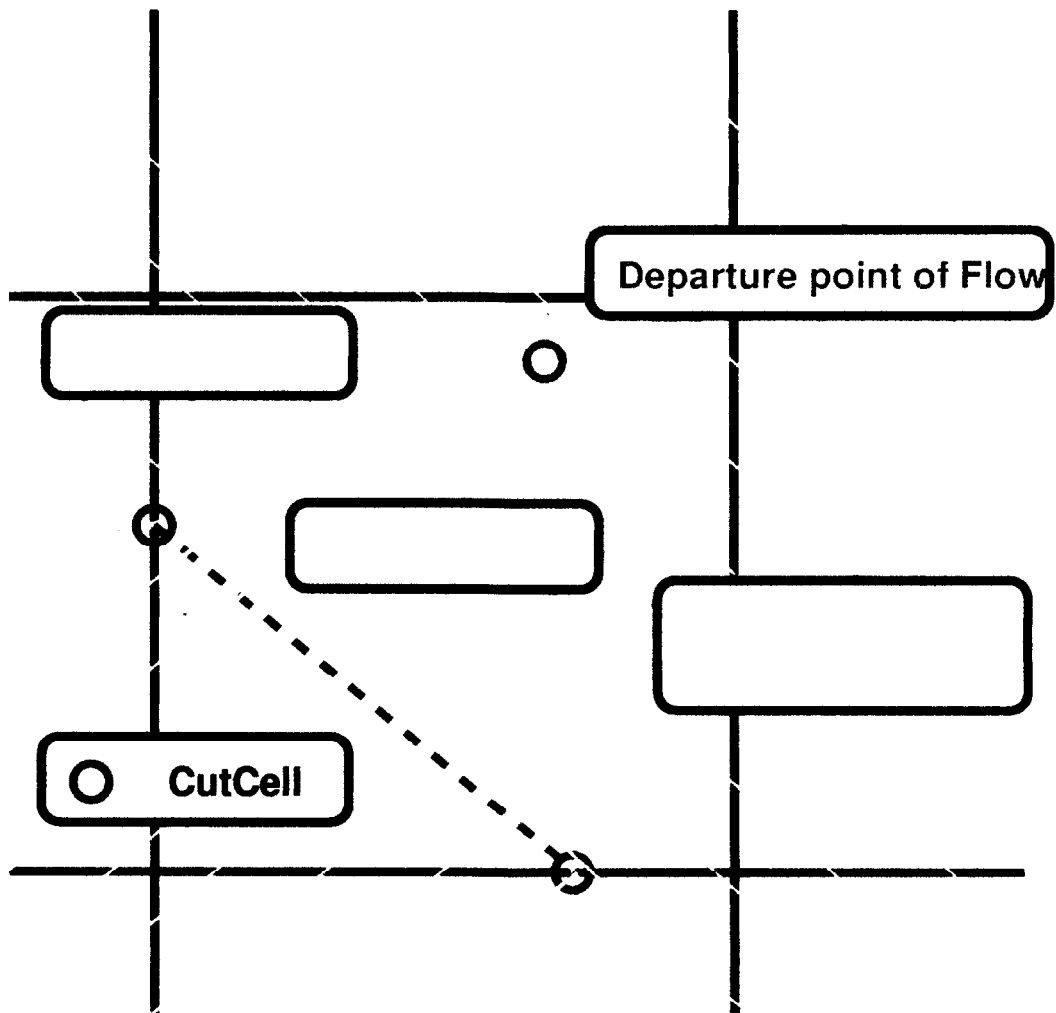


図 6.16: 複雑な境界面上の不連続値の補間

境界面が図に示すように 1 セル内で大きく変化する場合、移流原点位置における不連続な物理量を高精度補間するためには複雑な座標変換を必要とする。

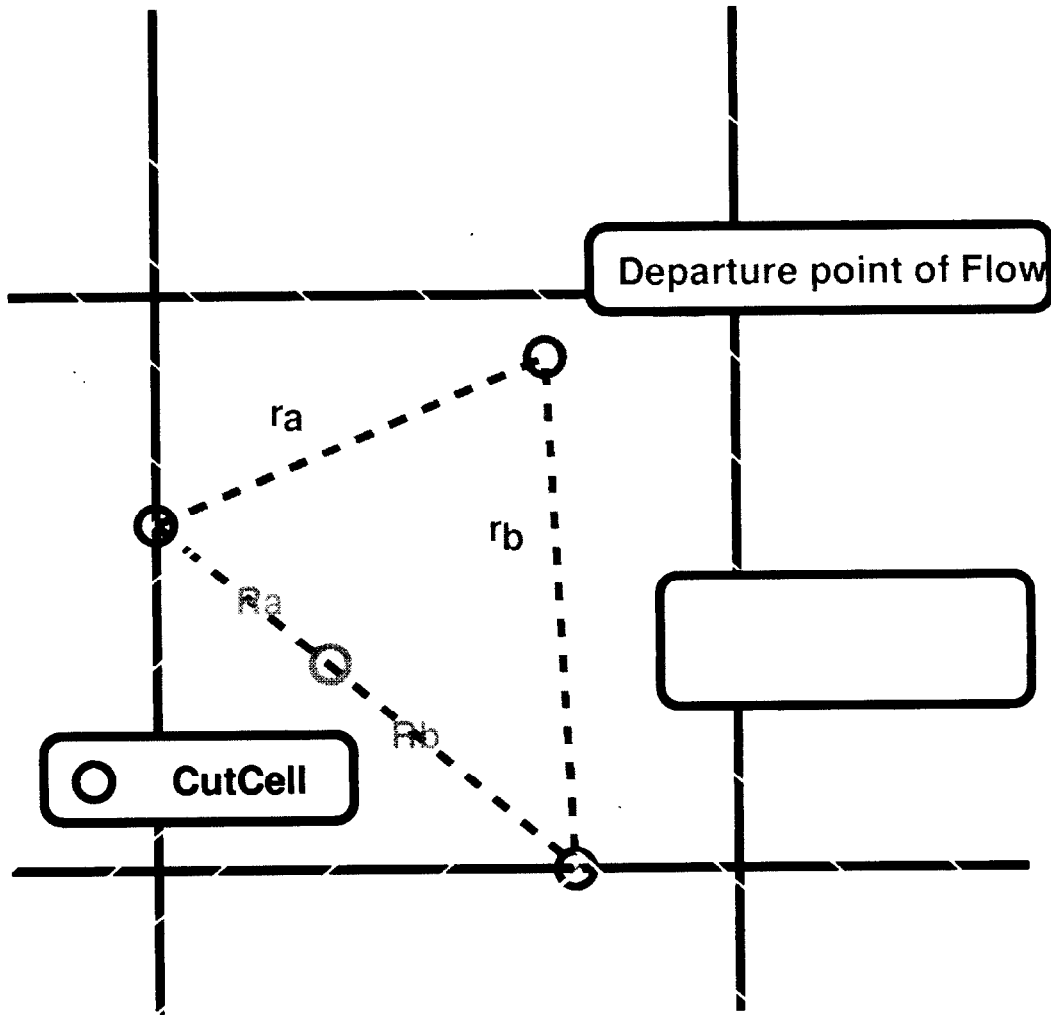


図 6.17: 多次元移動カットセル手法における境界面上の物理量補間手法

複雑な座標変換を行うのではなく、単純に各カットセルから移流原点位置までの距離を用い、1次補間により移流原点位置における物理量を決定する。

1次精度補間による計算精度の低下が見られる場合は不連続面記述が計算で用いているグリッド数で充分に行えていないことが考えられるので、グリッド数を増加させることで対応することが望ましい。

求められた移流原点位置における物理量を用いて一般的な時間発展型偏微分方程式

$$f_t + (\mathbf{u} \cdot \nabla) f = RHS. \quad (6.30)$$

のカットセル上での時間発展方法について示す。式 (6.30) をセミ・ラグランジュ的に時間積分する式は

$$f(x_{\text{Cut Cell}}, t^{n+1}) = f(x_{\text{DOF}}, t^n) + \int_C RHS dt \quad (6.31)$$

で表される。ここで、 $x_{\text{Cut Cell}}$  は、カットセル位置を、 $x_{\text{DOF}}$  はそれぞれのカットセル位置に対応する移流原点位置を表し、 $C$  は、移流原点位置とカットセル位置を繋ぐ移流経路を表している。非移流相積分 ( $\int_C RHS dt$ ) に関しては、IDO 法により行うため、テーラー展開に基づいた時間積分を行う。ただし、多次元移動カットセル手法では凹型形状に対して高次空間微係数を計算する手続きが複雑であることから、以下の例ではテーラー展開の1次展開式、

$$\int_C RHS dt = RHS_{DOF} \Delta t \quad (6.32)$$

により非移流相積分を行う。

次時刻  $t = t^{n+1}$  におけるカッセル位置に対する移流原点位置 (Departure point of Flow; DOF) の計算手法は、必要時間精度に応じ以下に示す 2 通りが容易に導入可能である。1 次精度時間後退法 (Backward-in-time) 法を用いた場合は

$$\mathbf{x}^n = \mathbf{x}^{n+1} - \mathbf{u} \Delta t \quad (6.33)$$

により計算される。時間 2 次精度で移流原点を計算する場合、

$$\mathbf{x}^n = \mathbf{x}^{n+1} - \mathbf{u} \Delta t - \frac{1}{2} \frac{\partial \mathbf{u}}{\partial t} \Delta t^2 \quad (6.34)$$

をもちいる。  $\mathbf{u} = (u, v)$  としてスカラー値で考えると、

$$\begin{pmatrix} x \\ y \end{pmatrix}^n = \begin{pmatrix} x \\ y \end{pmatrix}^{n+1} - \begin{pmatrix} u \\ v \end{pmatrix} \Delta t - \frac{1}{2} \begin{pmatrix} u_t \\ v_t \end{pmatrix} \Delta t^2 \quad (6.35)$$

ここで、

$$u_t = \frac{\partial x}{\partial t} \frac{\partial u}{\partial x} + \frac{\partial y}{\partial t} \frac{\partial u}{\partial y} = uu_x + vu_y \quad (6.36)$$

$$v_t = \frac{\partial x}{\partial t} \frac{\partial v}{\partial x} + \frac{\partial y}{\partial t} \frac{\partial v}{\partial y} = uv_x + vv_y \quad (6.37)$$

を用いると、

$$x^n = x^{n+1} - u \Delta t - (uu_x + vu_y) \Delta t^2 \quad (6.38)$$

$$y^n = y^{n+1} - v \Delta t - (uv_x + vv_y) \Delta t^2 \quad (6.39)$$

から移流原点位置を求めることが出来る。時間精度 3 次以上では 2 階以上の空間微係数を必要とするため、補間関数を利用しなければならないためここでは考慮しない。

### 6.3.6 2次元移動カッセル手法のアルゴリズム

以上をふまえ、2次元移動カッセル手法の計算の流れを以下に示す。

1. カッセルの定義 (Initialize)
2. CFL 数による安定判定
3. オイラーグリッド上の高次微係数の計算
4. オイラーグリッド上の物理量計算と時間更新
5. Level Set 関数の移流計算
6. 次時刻におけるカッセル位置の計算と定義
7. 次時刻のカッセル位置における物理量をセミ・ラグランジュ的手法により計算
8. CFL 条件により除かれたオイラーグリッド上の値を補間

この計算手順は容易に 3 次元化できるものとする。

### 6.3.7 2次元計算例題

#### 2次元移流方程式への適用

2次元移動カットセル手法の有効性を検証するため、カットセル上の物理量が計算時間中に変動しない移流方程式

$$f_t = -uf_x - vf_y \quad (6.40)$$

を扱う。カットセル上の物理量更新は上述の2次元移流方程式をセミ・ラグランジュ形に書き直した

$$f_{\text{Cut Cell}}^{n+1} = f_{\text{DOF}} + \int_C 0 dt \quad (6.41)$$

によって行われる。ここで、 $f_{\text{DOF}}$  は移流原点位置における物理量を表しており、移流原点位置は時間2次精度の時間後退法により計算する。 $C$  は、移流原点位置と計算対象カットセル位置を結ぶ移流経路を表す。

計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, \Delta x = 1/100, \Delta y = 1/100$  において、剛体回転速度場  $u(x, y) = y, v(x, y) = -x$  を与え、図 6.18 に示す初期プロファイル  $f = 1(\sqrt{(x-x_c)^2 + (y-y_c)^2} \leq 15\Delta x, x_c = 0, y_c = 0.5), f = 0$  (その他の点) が移流速度により計算空間を原点を中心に  $CFL = \Delta t/\Delta x = 0.19$  を用いて1回転するまで解いた計算結果を図 6.19 に示す。

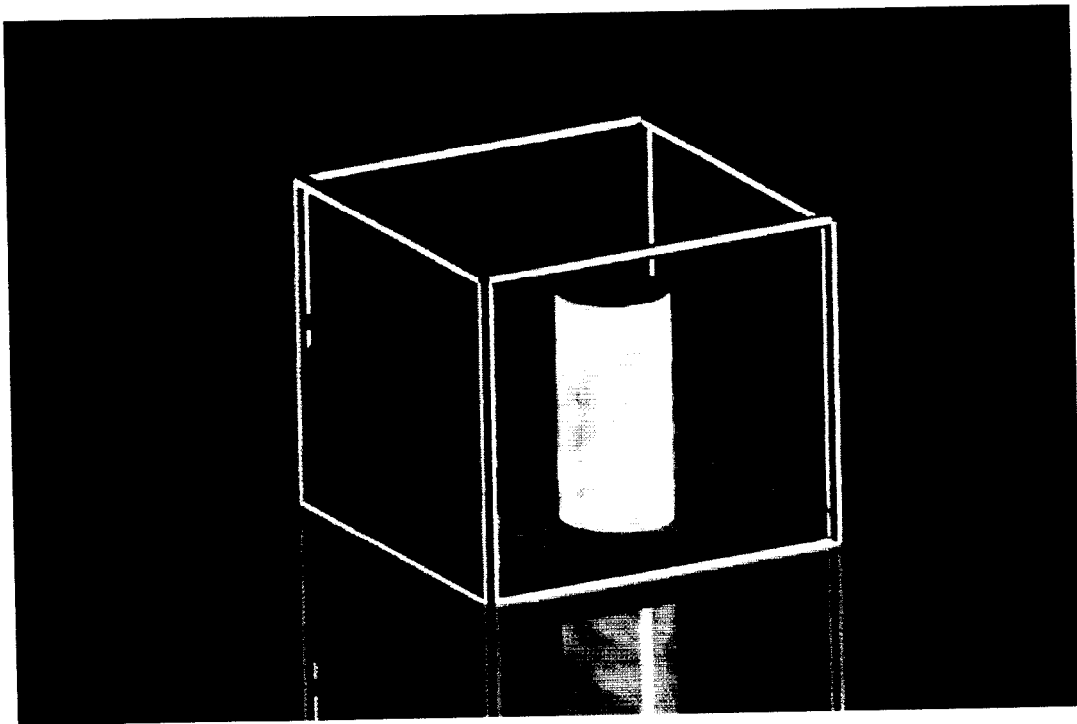


図 6.18: カットセルを適用した IDO 法による 2次元移流方程式のための初期プロファイル  
直交 2次元オイラーグリッド上に円形プロファイルを与えた場合、オイラーグリッド解像度程度でしか円形を表せないが、境界面にカットセルを用い、それを利用した可視化を施すことでより高精度に円形プロファイルを表現できる。

図 6.18 と図 6.19 を比較して分かるように、不連続面からの数値拡散は移動カットセル手法を利用することにより完全に抑制することを達成した。しかし、境界面の移動計算に LevelSet 関数を用いており、その移流計算で生じる数値拡散のために境界面自体が幾分解析解 (初期プロファイルと一致) からずれる結果となった。しかし、移動カットセル手法を適用しない結果と比べると飛躍的に高精度計算を可能にしていることが理解できる。詳細なカットセルの導入結果を示すため  $t = 2\pi$  におけるカットセル位置とその解析

解を図 6.22 に示す。また、計算結果から境界形状の精度を検証することを目的として、移動カットセル位置の、理論解から得られる境界面位置からの誤差を

$$\text{境界位置誤差} = \frac{|\text{C カットセルグリッドの理論解境界面からの距離}|}{\text{カットセルグリッド数}} \quad (6.42)$$

により計算する。その結果、 $1.157082 \times 10^{-5}$  と求められた。プロフィール半径を用いた平均誤差は

$$\frac{1.157082 \times 10^{-5}}{0.15} = 7.71 \times 10^{-3} \quad (6.43)$$

と求められ、プロフィールの変形率はおよそ 0.8% 程度であることを示した。

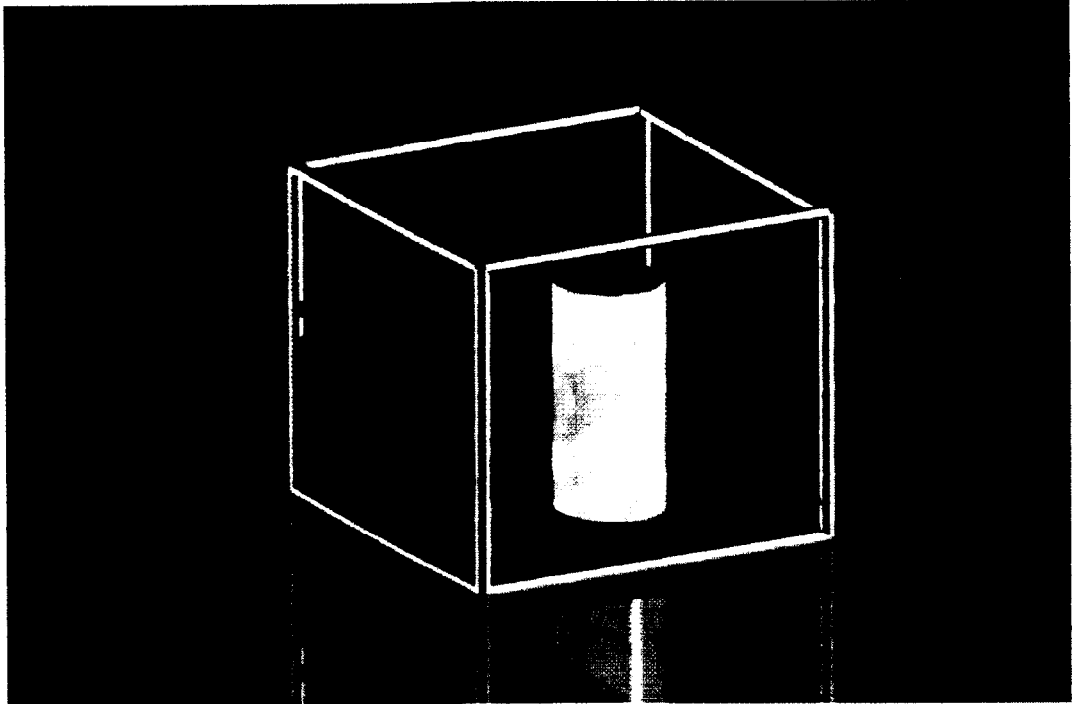


図 6.19: カットセルを適用した IDO 法による 2 次元移流方程式の計算結果

剛体回転速度場において円形プロフィールを計算空間中心点周りに完全に 1 周した後の計算結果を示す。Level Set 関数の数値拡散により若干のプロフィール変形が見られたが、不連続面記述に関しては数値拡散誤差、オーバーシュートならびにアンダーシュート無しで高精度に表現できている。

2 次元移動カットセル手法の特異点における数値拡散誤差を検証することを目的として、正方形を用いた初期プロフィールを与え、同様に剛体回転速度場中での振舞について検証する。初期プロフィールは、 $f = 1$  ( $-0.25 \leq x \leq 0.25, 0.25 \leq y \leq 0.75$ ),  $f = 0$  (その他の点) として与えた (図 6.20)。初期プロフィールは直交格子に適合しているため、カットセルを用いない手法であってもその初期プロフィールを高精度に記述できるが、カットセルを用いることによりその不連続面を完全に同じグリッド上で記述できることから、更に高精度に表現できている。時刻  $t = 2\pi$  における計算結果を図 6.21 に示す。正方形の各頂点 (特異点) では特に数値拡散が顕著に見られたが、正方形の 1 辺の長さを代表長さとして式 (6.42) から計算されたプロフィール変形率は

$$\frac{8.522706 \times 10^{-4}}{0.5} = 1.70 \times 10^{-3} \quad (6.44)$$

となり、0.17% 程度と得られた。特異点近傍の詳細なカットセルの配置を図 6.23 に示す。剛体回転速度場中で 1600 計算ステップによる回転計算であるにもかかわらず特異点の丸めはただか 1 グリッド程度に押えることが可能であることが理解できる。

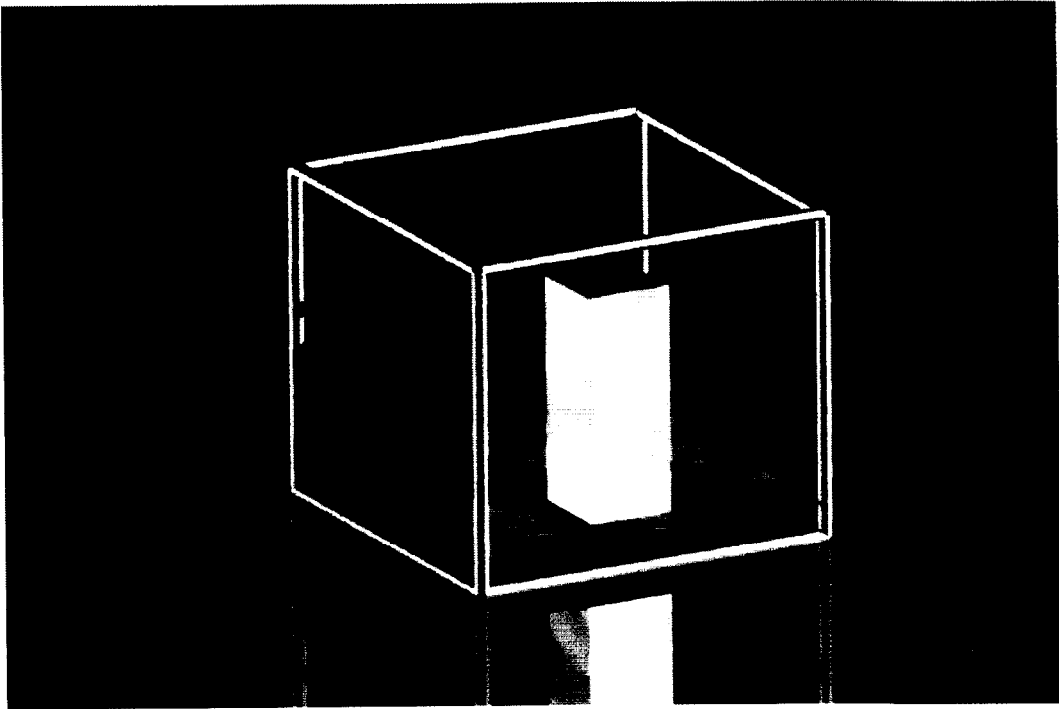


図 6.20: カットセルを適用した IDO 法による 2 次元移流方程式のための特異点を持つ初期プロフィール

特異点の特性を検証することを目的として正方形で初期プロフィールを与えた。

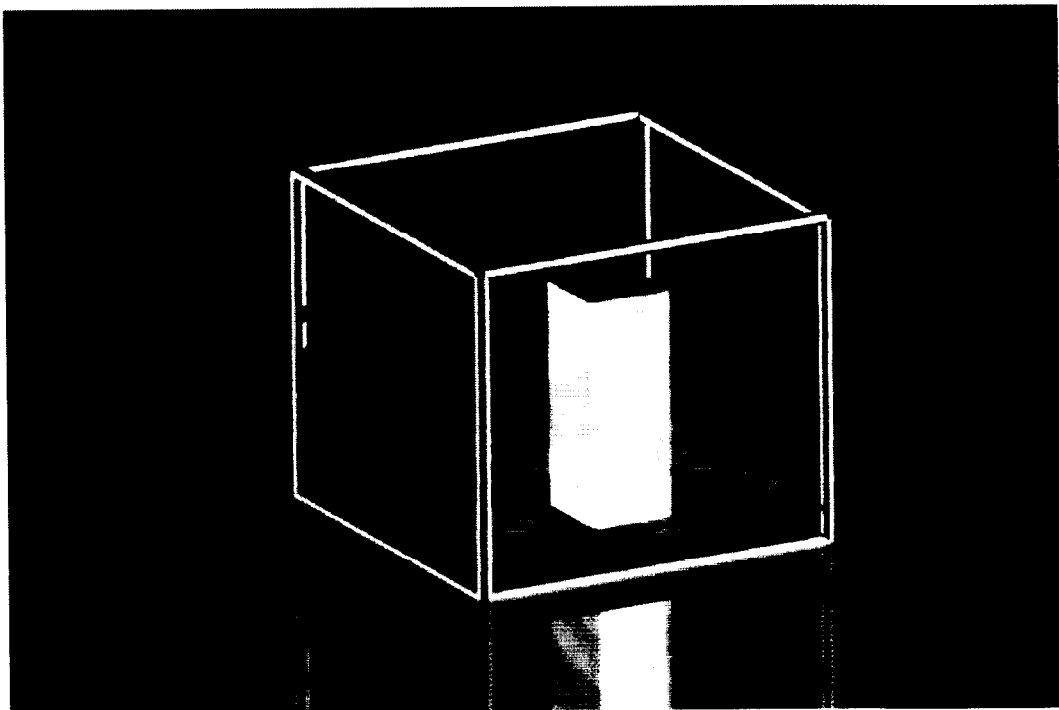


図 6.21: 特異点を持つ初期プロフィールに対する 2 次元移流方程式の計算結果

剛体回転速度場において正方形プロフィールを計算空間中心点周りに完全に 1 周した後の計算結果を示す。Level Set 関数の数値拡散により特異点 (正方形の頂点) において若干の数値拡散が見られたが、補間により発生するアンダーシュートやオーバーシュートを完全に抑制した数値計算結果を得ることに成功した。



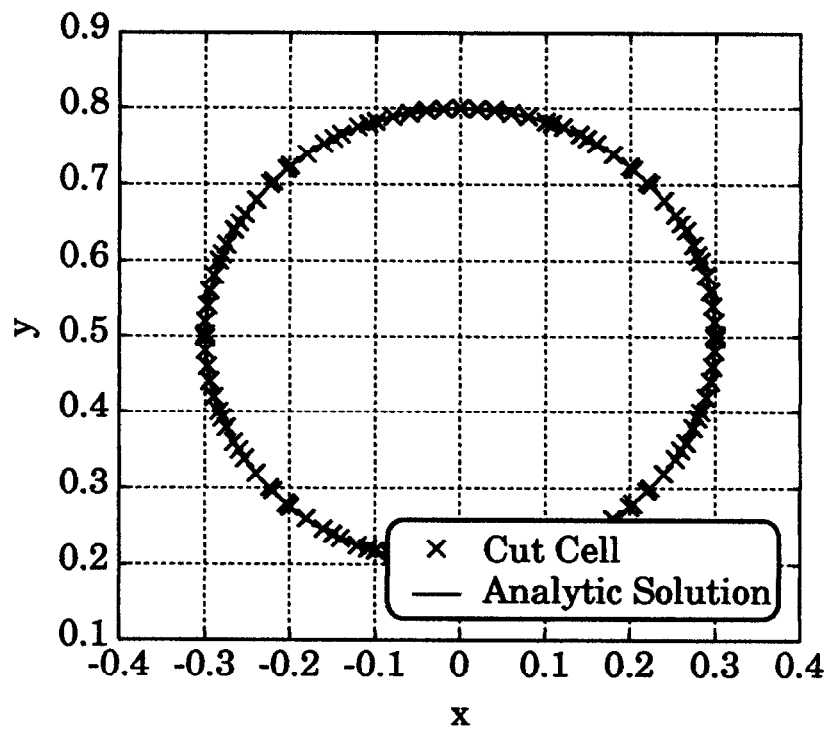


図 6.22: 移動カットセル手法による 2 次元移流方程式の計算結果

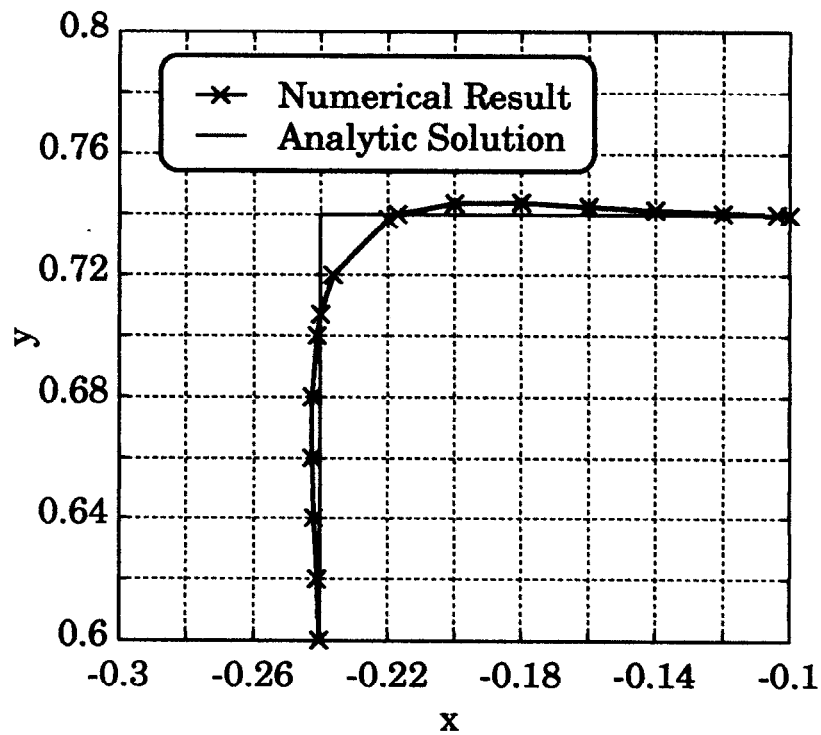


図 6.23: 移動カットセル法による正方形プロファイルの移流計算結果:  
 特異点 (頂点) は数値拡散により丸められる結果となったが、1600 計算ステップでの剛体回転速度場  
 中での回転計算であるにもかかわらずただか 1 グリッド程度の丸めで押えられている。

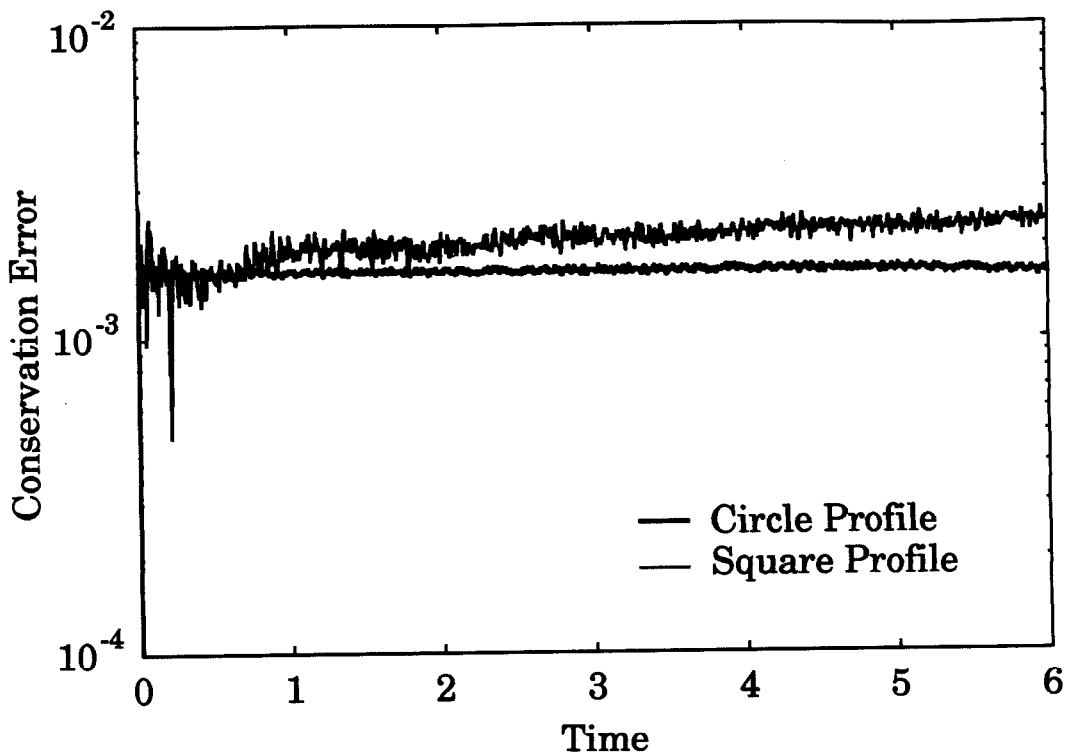


図 6.24: 移動カットセル手法による 2 次元移流方程式の保存誤差:

剛体回転速度場における 2 次元移流方程式の保存誤差は IDO 法による 2 次元 Level Set 法の面積積分誤差を直接示している。完全な保存を保証することは無いが、急激に悪くなる傾向もまた見られない。

円形、正方形初期プロファイルの剛体回転計算における保存誤差の時間変動を図 6.24 に示す。2 次元移流方程式を物体の変形が生じない剛体回転速度場を用いて扱うため、この保存誤差は IDO 法による Level Set 手法の面積保存誤差に当てはまる。円形のようにプロファイル中に特異点の無い形状に対しては完全な保存を得ることはできないが、その値が時間と共に大きく変動することは無かった。正方形でプロファイルを与えた場合は先の計算結果から特異点近傍でプロファイルが鈍化することから保存誤差に影響を及ぼし、誤差が増大する傾向が見られた。

さらにこれまでに CIP 法や IDO 法の基礎精度検証問題として用いて来た Zalesak の剛体回転問題を扱う。計算空間ならびに計算条件はこれまでの円形プロファイル、正方形プロファイルを扱ったものと完全に同じとし、初期プロファイルのみを Zalesak の例題で用いられているものを導入した。その初期プロファイルは図 6.25 (a) に示される形状をとる。剛体回転速度場中で計算空間中心を原点として完全に 1 回転した結果を図 6.25 (b) に示す。計算結果から式 (6.42) を用いて計算したプロファイル変形量は  $1.779914 \times 10^{-3}$  と得られ、プロファイル半径  $15\Delta x$  との相対変形量は

$$\frac{1.779914 \times 10^{-3}}{0.15} = 1.19 \times 10^{-2} \quad (6.45)$$

となり、1% 程度の変形が確認され、これまでの円形プロファイル、正方形プロファイルに比べ最も大きく変形した結果を得た。その理由について考察するため、計算結果を構成するカットセルの位置とその解析解の不連続面位置を図 6.26 に示す。Zalesak 例題の初期プロファイルは円弧ならびに凹型形状の組合せにより表現される。先の円形プロファイルの剛体回転問題から、円形プロファイルに関しては低数値拡散誤差で解くことが可能であることが示されている通り、Zalesak プロファイルの円弧部に関してはおおむね高精度で解を得ることに成功した。その一方で、IDO 法による数値計算結果に比べ十分に高精度で計算で

きているものの、特異点近傍ならびに凹型形状内部では数値拡散により理論解と外れたプロファイルを示していることが分かる。特異点近傍での数値拡散は正方形プロファイルに体する剛体回転問題から鈍化する結果が得られているように Zalesak 例題においても同様の結果を得た。

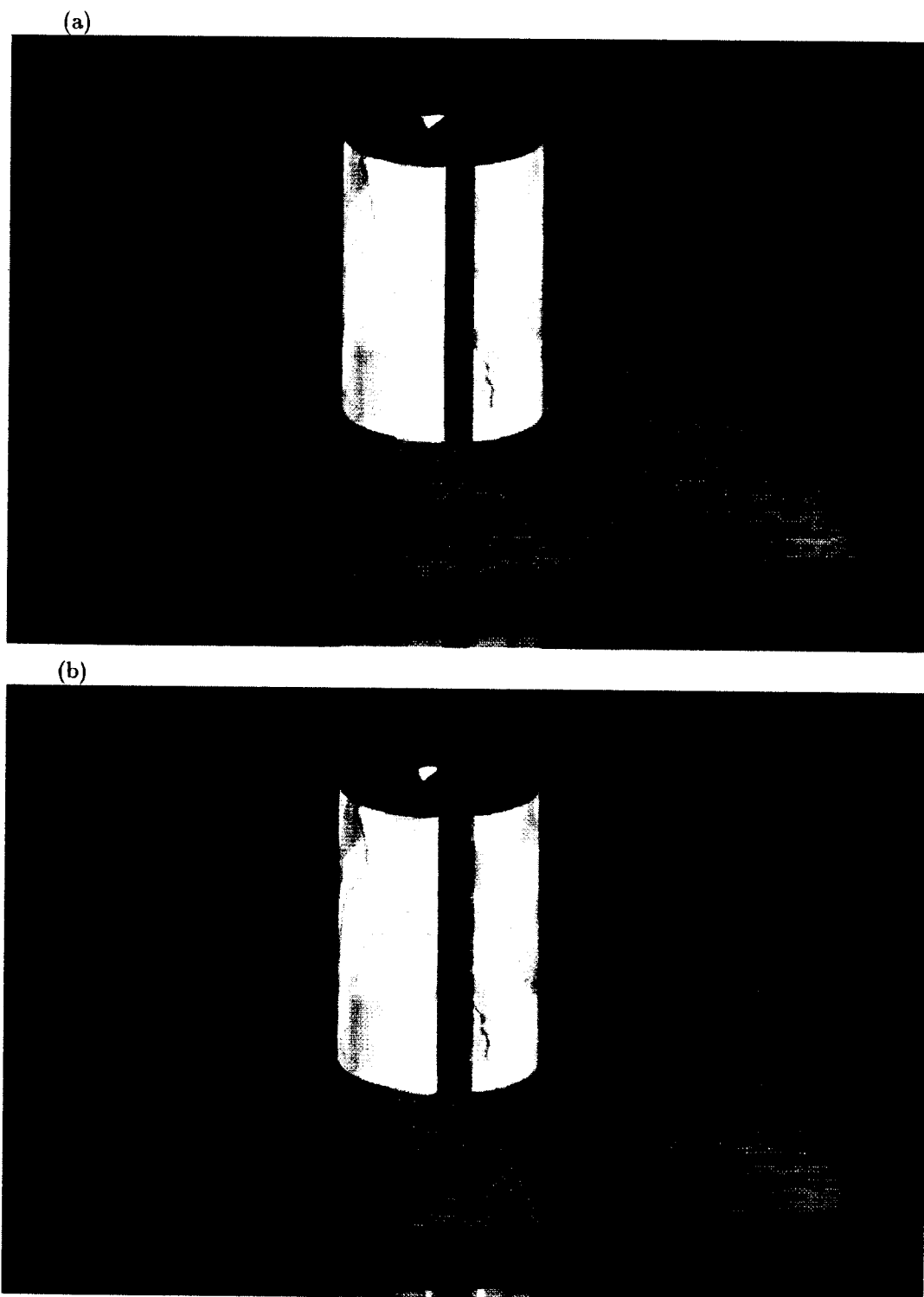


図 6.25: 移動カットセル手法を用いた Zalesak 例題の検証:

(a) 初期プロファイルを剛体回転速度場において 1600 計算時間ステップで計算領域中心を完全に 1 回転させた (b) 数値計算結果。Zalesak 例題の持つ複雑な形状を高精度で移流計算させることに成功した。

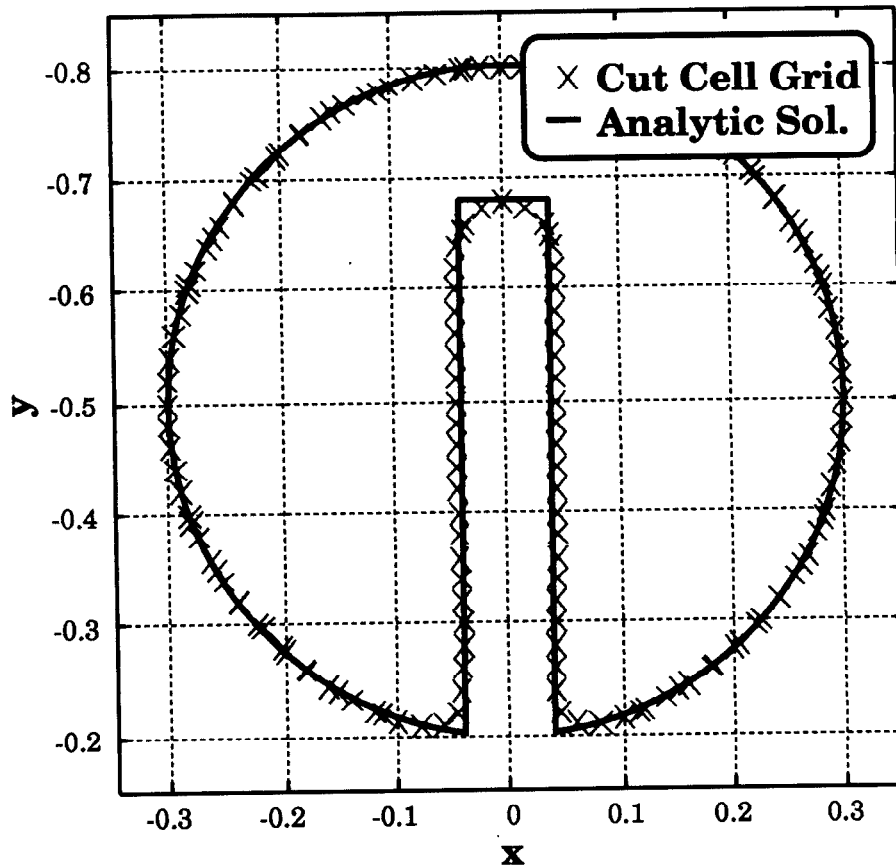


図 6.26: 移動カットセル手法による Zalesak 例題の計算結果:

Zalesak 例題で用いられるプロファイルは円弧と凹型形状を持つ。円弧で表現される界面では数値拡散が比較的少ないが、凹型形状部では特異点が近接することから Level Set 関数の精度が低下するため、界面記述精度が低下する。

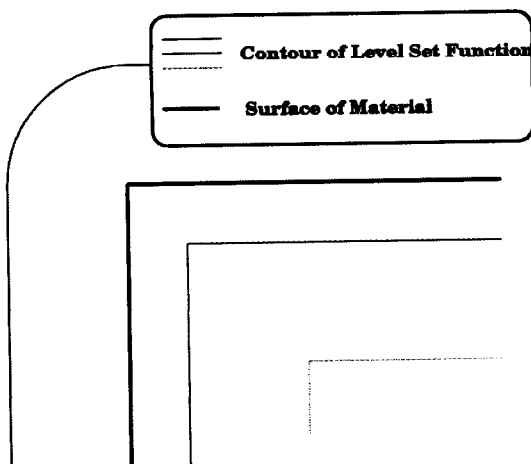


図 6.27: 特異点近傍における Level Set 関数の不連続:  
特異点を持つプロファイルは Level Set 関数の空間微係数に不連続を持つ。

凹型形状内部における数値拡散誤差の原因について考察する。移動カットセル手法では、その設計から界面の移動追跡方法として Level Set 関数を用いている。Level Set 関数は境界面からの距離を物理量として持つ関数であることから多角形頂点では関数値の空間微分に不連続点が生じる。

今回問題となっている凹型形状では、境界面に囲まれた領域の中心で空間微係数の不連続 (微分不可能領域) が存在する。IDO 法は有限差分法などと同様に計算空間全体が微分可能であることを前提としているスキームであることから、このような空間微係数の不連続が生じた場合滑らかな関数で接続するように試みる。その結果空間解像度が低下し、図 6.26 で見られたような誤差が発生したものと考えられる。円形プロファ

イルにおいてもその中心で空間微係数の不連続が生じているが、情報として重要な界面付近までの距離があるため、誤差として顕著に観測されなかったものとする。

## 2 次元保存方程式への適用 (中心へ向かう速度場)

カットセル上の値が時間変動する問題に適用するため、2次元保存方程式

$$f_t = -uf_x - vf_y - (u_x + v_y)f \quad (6.46)$$

を扱う。移動カットセル上の数値計算は上記2次元保存方程式をセミ・ラグランジュ形に書き直した積分方程式

$$f_{\text{Cut Cell}}^{n+1} = f_{\text{DOF}} + \int_C \{-(u_x + v_y)f\} dt \quad (6.47)$$

によって行われる。ここで、 $f_{\text{DOF}}$  は移流原点位置における物理量を表しており、移流原点位置は時間2次精度の時間後退法により計算する。 $C$  は、移流原点位置と計算対象カットセル位置を結ぶ移流経路を表す。

計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, \Delta x = 2/100, \Delta y = 2/100$  において、計算空間の原点に向かう移流速度場  $(u(x, y) = -x, v(x, y) = -y)$  を与えた。図 6.28 に示される初期プロファイル  $f = 1(\sqrt{(x-x_c)^2 + (y-y_c)^2} \leq 15\Delta x, x_c = 0, y_c = 0), f = 0$  (その他の点) に対し  $CFL = \Delta t/\Delta x = 0.19$  を用いて  $t = 0.98$  まで計算を進めた計算結果を図 6.29 に示す。2次元移動カットセル上の物理量に変化する2次元保存方程式を扱った場合でも不連続面記述のために数値拡散誤差が生じることが無いことを数値実験的に明らかにした。また2次元移動カットセル上の物理量計算に伴う数値不安定性も生じていないことがわかる。

移動カットセル手法を構築した際の方針として、各グリッド線にはカットセルを1点以上導入しないことから、各グリッド内ではプロファイルを3角形の組合せで表現することができる。つまり、計算グリッド上に与えられているプロファイルは円形であるが、移動カットセル手法の解像度で多角形により表現することができる。このことを利用してプロファイルの体積を計算し、保存誤差を

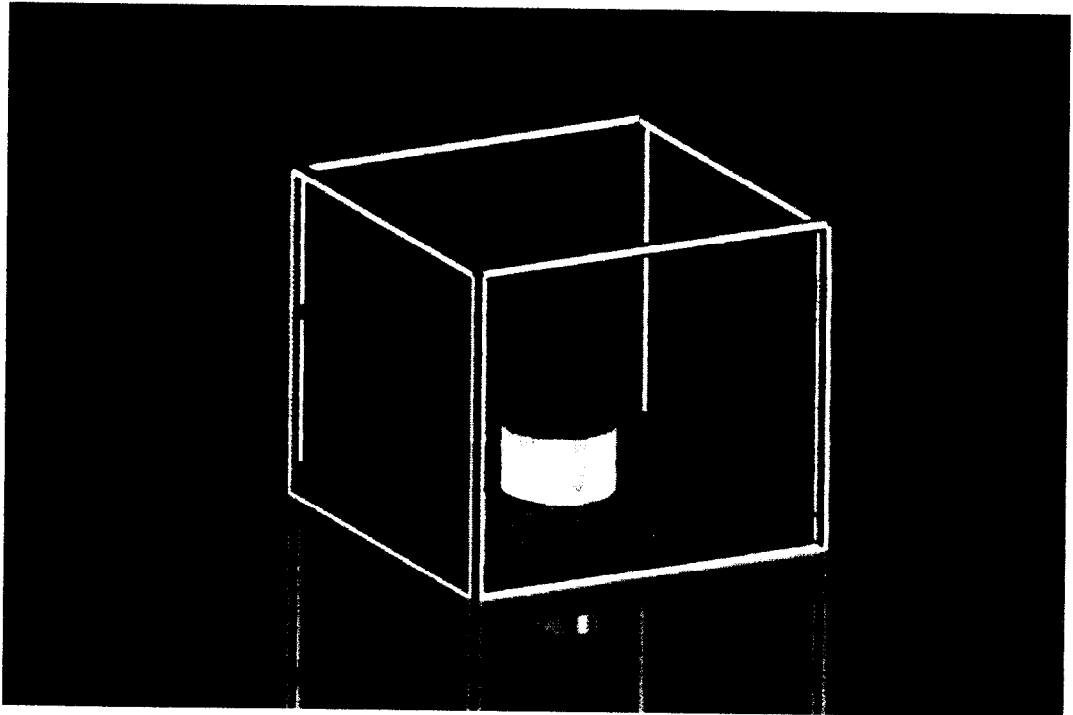


図 6.28: カットセルを適用した IDO 法による 2次元保存方程式のための初期プロファイル  
初期プロファイルを計算空間の原点を中心とした円形で与えている。カットセルを適用することで初期プロファイルをオイラーグリッドの表現精度 ( $\Delta x$ ) 以上の解像度で表現することに成功した。

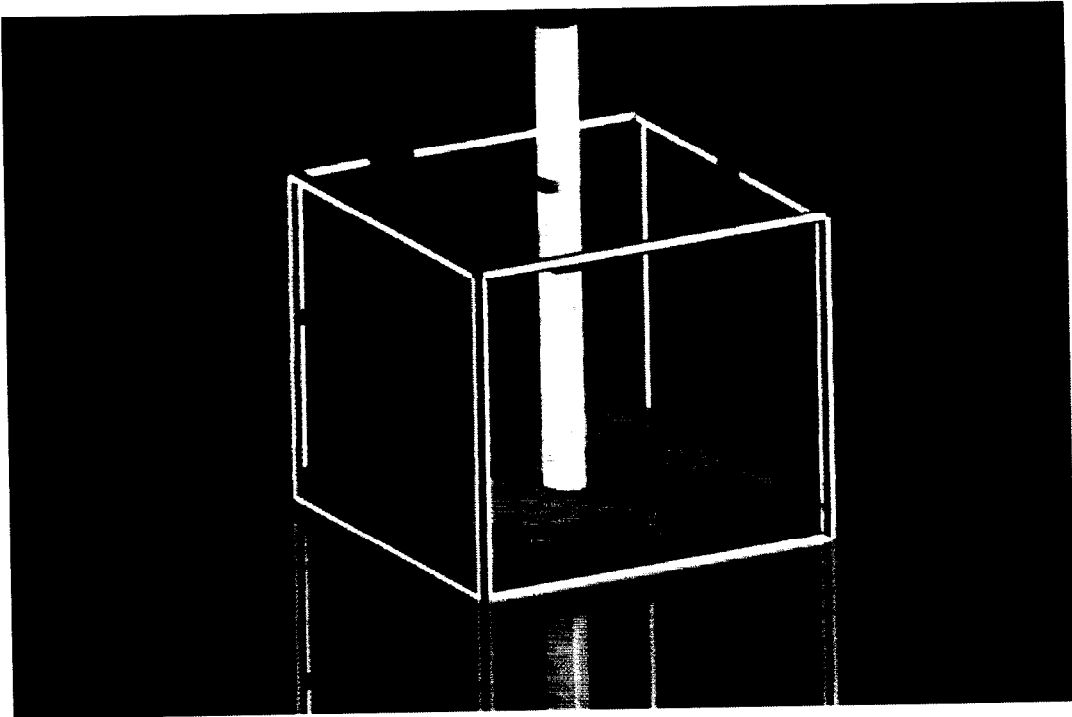


図 6.29: カットセルを適用した IDO 法による 2次元保存方程式の計算結果 (原点への収束)  
 2次元移動カットセル上の物理量に変化する 2次元保存方程式を扱った場合でも不連続面記述のために数値拡散誤差が生じることが無いことが数値実験的に明らかにした。また 2次元移動カットセル上の物理量計算に伴う数値不安定性も生じていないことがわかる。

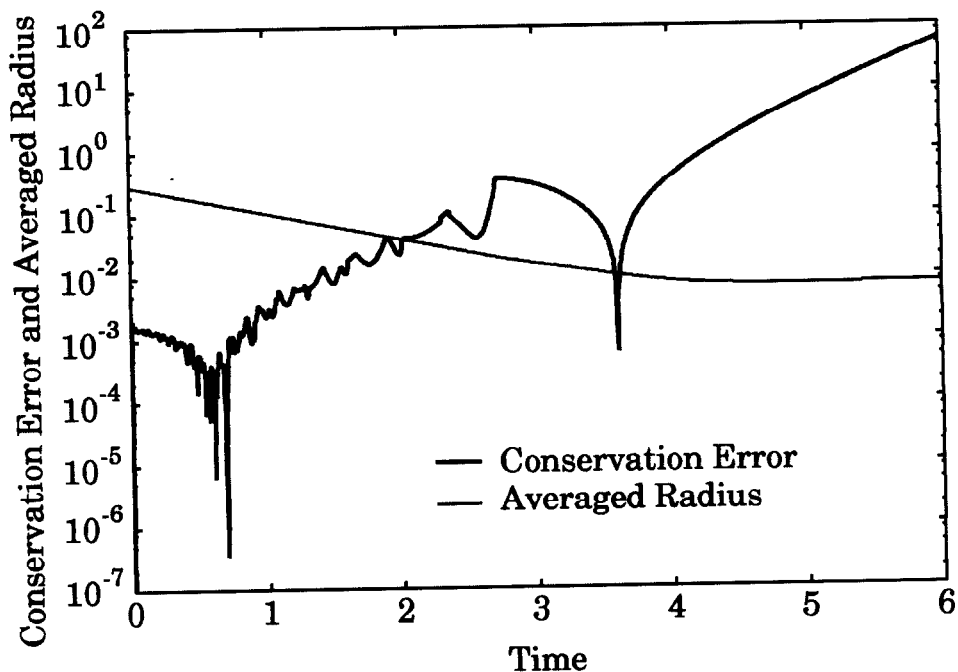


図 6.30: カットセルを適用した IDO 法による 2次元保存方程式の保存誤差 (原点への収束)  
 プロファイル半径が  $\Delta x/2 = 0.01$  程度までの計算が可能であることが計算結果から得られた平均プロファイル半径から明らかにされた。プロファイル半径が  $\Delta x$  までの計算では保存誤差は最大 21.6%で計算することが出来る。

$$\text{Volume} = \sum_p \text{Area}_p \times f_p \quad (6.48)$$

$$\text{Conservation Error} = \left| \frac{\text{Volume}_{\text{Calc}} - \text{Volume}_{\text{Initial}}}{\text{Volume}_{\text{Initial}}} \right| \quad (6.49)$$

により計算した結果を図 6.30 に示す。ここで、添字の  $p$  は、ポリゴン番号をあらわす。プロファイル半径が  $\Delta x/2 = 0.01$  程度までの計算が可能であることが計算結果から得られた平均プロファイル半径から明らかにされた。プロファイル半径が  $\Delta x$  までの計算では保存誤差は最大 36.2% で計算することが出来る。プロファイル半径  $4\Delta x$  までの計算に限定すると、保存誤差は 0.76% 以下で求まることがわかる。

## 2 次元保存方程式への適用 (回転速度場)

より複雑な計算条件下での 2 次元移動カットセル手法の検証を行うため、2 次元保存方程式

$$f_t = -uf_x - vf_y - (u_x + v_y)f \quad (6.50)$$

を空間中で角速度が変動する回転速度場を用いて計算を行った。移流速度は

$$u(x, y) = \frac{y}{1 - 0.2 \cos(\theta)} \quad (6.51)$$

$$v(x, y) = -\frac{x}{1 - 0.2 \cos(\theta)} \quad (6.52)$$

で与えた。ここで  $\theta$  は、座標値  $(x, y)$  を用いて

$$\theta = x/\sqrt{x^2 + y^2} \quad (6.53)$$

とした。この速度場を用いることで初期プロファイルは計算空間を原点中心に  $t = 2\pi$  で 1 回転し、完全に初期プロファイルと一致した形状を探る。また、 $0 < t < 2\pi$  で、プロファイルは移流速度により圧縮、伸長される。計算空間  $-1 \leq x \leq 1, -1 \leq y \leq 1, \Delta x = 2/100, \Delta y = 2/100$  中で、初期プロファイルを図 6.18 示されるように  $f = 1(\sqrt{(x-x_c)^2 + (y-y_c)^2} \leq 15\Delta x, x_c = 0, y_c = 0.5), f = 0$ (その他の点) と与えた。 $CFL = \Delta t/\Delta x = 0.19$  によりプロファイルが計算空間中を 1 回転するまでの計算結果を図 6.31 に示す。不連続面を記述するために生じる補関数による数値拡散を完全に抑制できていることがわかる。ただし、 $t = 2\pi$  における計算結果から、移動カットセル上の物理量計算に数値誤差が確認された。その原因として、2 次元移動カットセル手法ではカットセル上の物理量計算をセミ・ラグランジュ手法を用いて行っており、その移流原点位置の物理量を補関数するために 1 次精度補間を行っていることや、非移流相積分を時間 1 次精度で行っていることを挙げることが出来る。

対照実験として IDO 法による数値計算結果を図 6.32 に示す。低数値拡散で計算できているものの、移動カットセル手法を適用した計算結果と比較した場合、そのアンダーシュートやオーバーシュートの発生ならびに不連続面記述精度低下は明らかである。

具体的に移動カットセル上の補関数精度による数値誤差を確認することを目的として、 $t = 2\pi$  における計算結果のプロファイル中心を通る  $y = 1$  上の不連続面近傍プロファイルを図 6.33 に示す。IDO 法や CIP 法で同様の計算をその基礎精度検証において行っているが、それらから得られたアンダーシュートやオーバーシュートと比較して十分に小さい数値誤差であり、プロファイル中心付近ではそれらに影響される数値振動などが見られないことから、数値不安定性を引き起こす原因としても影響力が小さいと考えられる。

また、保存方程式を扱っていることからその保存誤差を計算時間に対して見積もった結果を図 6.34 に示す。同プロファイルに対して剛体回転速度場中の 2 次元移流方程式を扱った結果では計算時間中でほぼ一定の保存誤差を示したが、本計算ではプロファイルが計算時間中で圧縮、伸長され、またプロファイル値が時間と共に変動することから、計算時間中で保存誤差の変動が生じる結果となった。

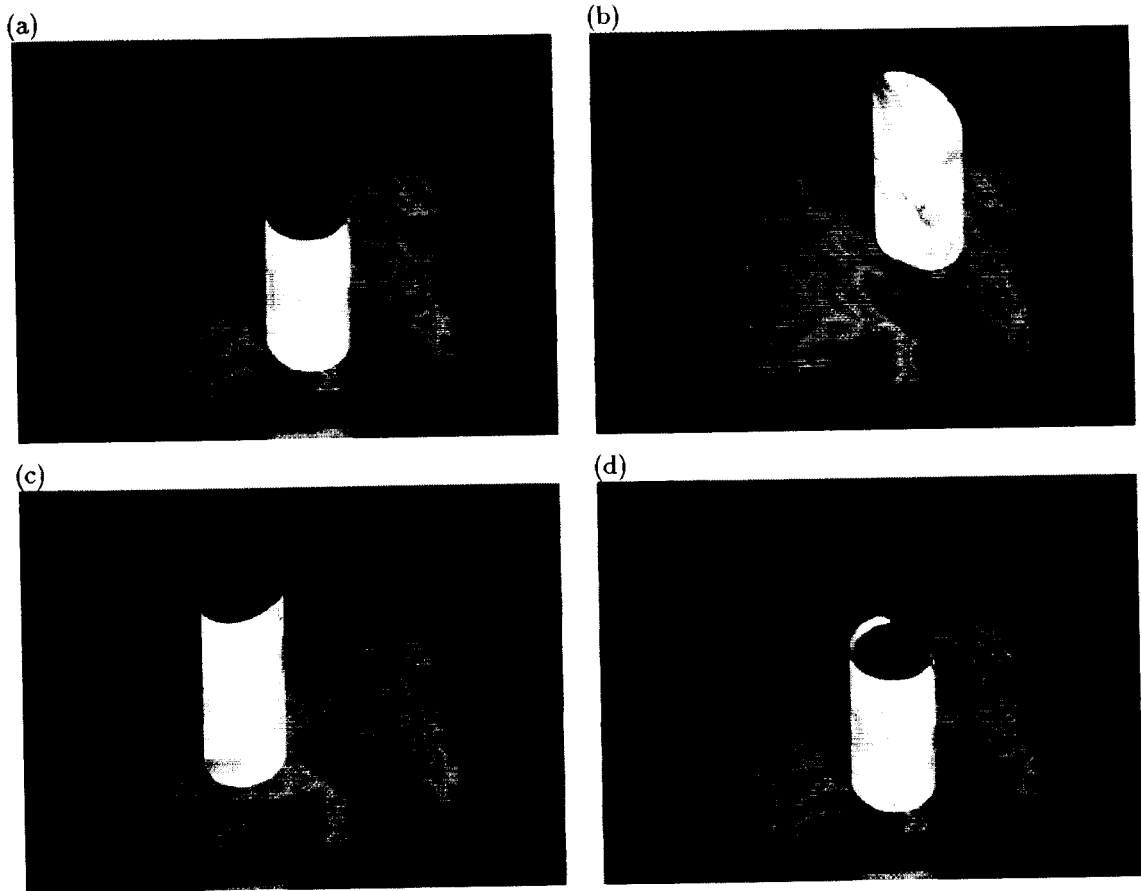


図 6.31: カットセルを適用した IDO 法による 2 次元保存方程式の計算結果 (回転速度場)  
 (a) $t = 0$ , (b) $t = 0.6875\pi$ , (c) $t = 1.5675\pi$ , (d) $t = 2\pi$  におけるプロファイル。  $t = 2\pi$  でのプロファイルには若干の数値誤差が見られたものの、不連続面記述のために生じる数値拡散誤差を完全に抑制すると共に、カットセルを用いることでオイラーグリッドの空間精度 ( $\Delta x, \Delta y$ ) 以上の解像度でプロファイルを表示することに成功した。

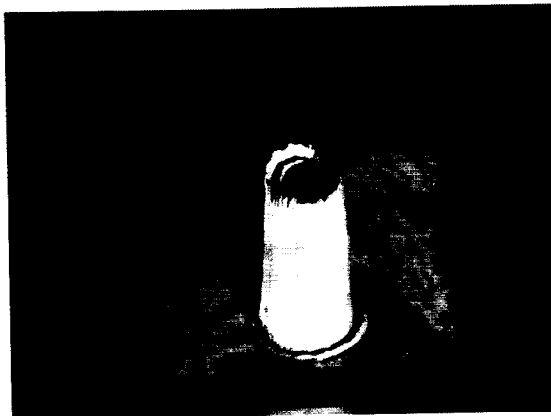


図 6.32: IDO 法による 2 次元保存方程式の数値計算結果

移動カットセル上の物理量計算における数値誤差に関してはセミ・ラグランジュ手法における移流原点位置における物理量計算アルゴリズムの高精度化、ならびに非移流相積分の時間精度の改善を行うことで低減できると考える。ただし、冒頭にも述べたようにそのためには IDO 法を用いていることから高次空間微係数を移動カットセル上で適切に求めることが必要となり、不連続面形状に沿った複雑な座標変換などが要求される。そのための計算コストやプログラミングの複雑化を考えると計算格子の解像度向上、計算時間ステップ数解像度の向上を導入し、それらの問題に当たるとも一つのアプローチであると考えられる。以上から、多次元不連続面計算手法として提案した多次元移動カットセル手法はその精度を解析解と比較した場合若干の遜色が見られるが、数値計算手法としては十分な効果を発揮することを明らかにした。



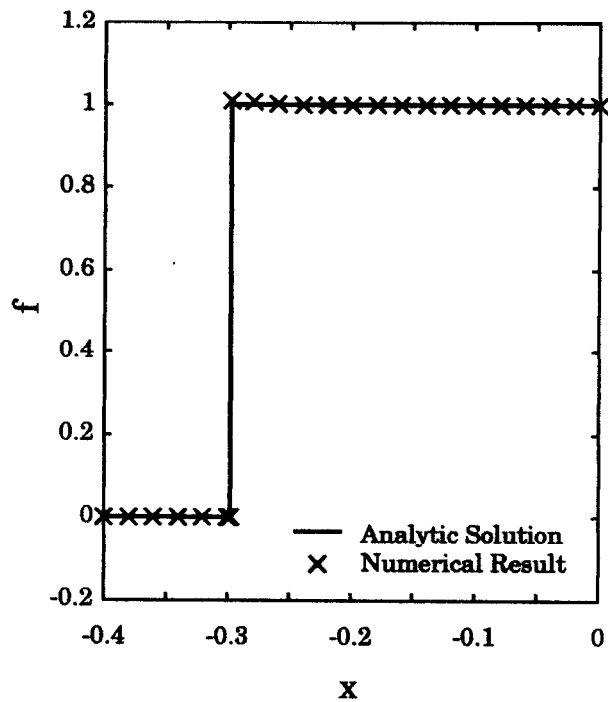


図 6.33: 移動カットセル手法を適用した計算結果のプロファイル切断面:  
 プロファイル中心を通る  $y = 1$  上の不連続面近傍プロファイルならびに解析解を示す。不連続面上の補間関数形成手法において計算精度の制限を行っていることから数値誤差が見られるが、移動カットセル手法を適用しない結果と比較すると明らかに高精度で計算できていることが分かる。

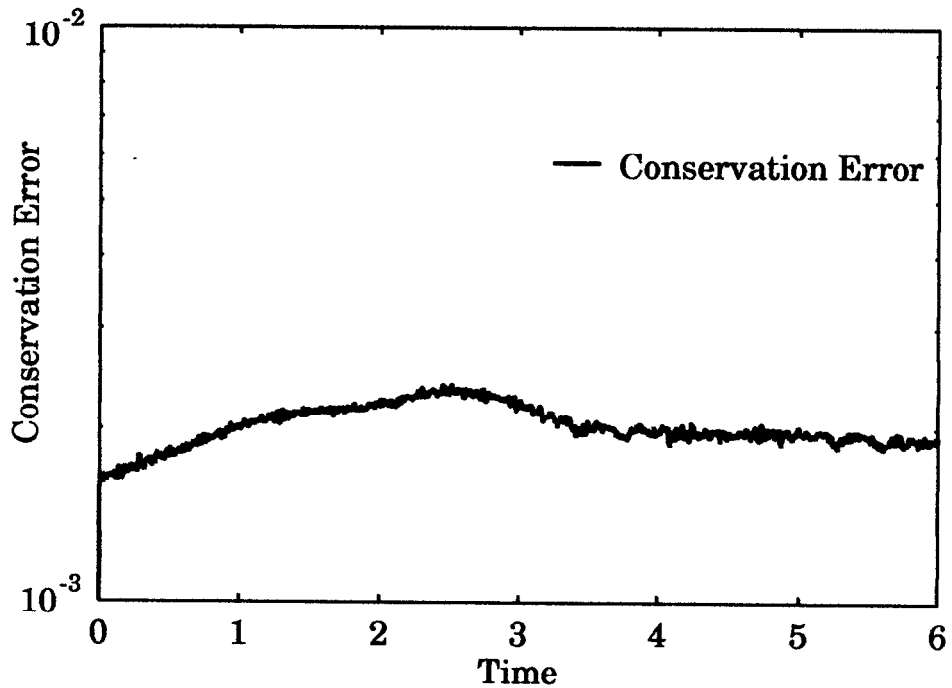


図 6.34: 移動カットセル手法による 2 次元保存方程式の保存誤差:  
 2 次元移流方程式に対する保存誤差ではほぼ一様な値を示したが、保存方程式を扱った場合、その形状の変化ならびにプロファイル値の変動が生じるため計算時間中で保存誤差が変動する結果を得た。

## 6.4 まとめ

オイラーグリッド上に固定境界条件として利用されるカットセル手法により不連続面を高精度で記述できることに着目し、カットセルをオイラーグリッド上で自由に移動することを許可した移動カットセル手法を提案した。1次元移動カットセル手法では、不連続点上の物理量計算をラグランジュ表記に変換された支配方程式をIDO法を用いて時間積分するアルゴリズムを提案した。1次元移動カットセル手法を用い、1次元移流方程式ならびに1次元保存方程式を検証した結果、オイラーグリッドを用いているにも関わらず補間関数による不連続面記述に伴う数値拡散誤差を完全に抑制することを達成した。また、移動カットセル手法を適用することで、オイラーグリッドが持つ空間解像度 ( $\Delta x$ ) 以上の解像度でプロファイルを表現することを可能にした。

多次元問題への適用を行うため、境界面追跡手法の一つであるLevel Set手法を移動カットセルの移流計算のために導入し、不連続面上の計算をセミ・ラグランジュ的手法により行うアルゴリズムを提案した。2次元移流方程式、2次元保存方程式を検証することで、単純な速度場に対しては2次元移動カットセル手法は十分な効果を発揮することを数値実験的に明らかにした。しかしその一方で複雑な速度場に対して2次元保存方程式を扱った際には若干の数値誤差が確認された。その原因として、セミ・ラグランジュ手法を用いてカットセル上の物理量計算を行う際にその移流原点位置の物理量補間を1次精度補間を用いて行っていることや、非移流相積分を時間1次精度で行っていることを挙げることが出来る。これらの数値誤差に関してはセミ・ラグランジュ手法における移流原点位置における物理量計算アルゴリズムの高精度化、ならびに非移流相積分の時間精度の改善を行うことで低減できると考える。しかしその一方で有限差分法やCIP法、移動カットセルを適用しないIDO法の数値計算結果と比較しても十分に高精度数値計算が達成されている。

以上から、2次元不連続面計算手法として提案した2次元移動カットセル手法は複雑な移流速度場に対しては若干の数値誤差を生じるものの、不連続面計算手法としては十分な効果を発揮することを明らかにした。また、識別関数を用いた2次元移動カットセル手法は容易に3次元移動Cut Cell手法に拡張することが可能なアルゴリズムである。

本研究で開発した移動カットセル手法は初期条件として不連続面位置、ならびに不連続面を構成する物理量を与える必要があることから、今後の課題として計算時間中に発生する衝撃波面や接触不連続面などの不連続面を検出し、移動カットセルを導入する手法に付いて検討することが重要であると考えられる。

## 第7章

### 結論

計算機処理能力の向上によりナビエ・ストークス方程式などの非線形偏微分方程式に対する大規模数値流体計算が盛んに行われてきている。近年少ない格子点数で解像度の高いプロファイル表記可能なエルミート補間を利用した数値流体計算手法として CIP 法が提案された。同様にエルミート補間を用いて少ない格子点数で高精度数値計算を可能にする局所補間微分オペレータ法 (Interpolated Differential Operator Scheme; IDO Scheme) が提案され、簡便に有効な結果を示してきている。本論文では IDO 法による数値流体計算への適用性を検討した。

これまでの IDO 法の研究において計算アルゴリズム中で重要となる、エルミート補間を微分して得られる高次空間微係数の精度に関しては十分に検討されていなかった。流体問題を扱う際に IDO 法の基礎精度を考慮することが重要であると考え、エルミート補間を用いた空間微係数の計算精度について検証した。少ない格子点数で高波数プロファイルを扱う場合は差分法と同様に計算精度が低下する一方、エルミート補間の空間精度の高さから、高解像度計算格子を用いた場合でも計算格子中に含まれるプロファイル波数  $k\Delta x$  が 0.01 以下になると丸め誤差の影響により精度向上されなくなることを明らかにした。この結果は、少ない格子点数で空間解像度が十分に得られることを明らかにしている。また、IDO 法による数値流体計算を行う際に、得られる流体構造のスケールに対して適当な計算格子の解像度を見積もることを可能にした。多次元流体問題を扱う場合 IDO 法で利用可能な CFL 数は同じエルミート補間を用いる計算手法である CIP 法に比べ小さな値となる。補間関数の形成手続きからその理由に付いて検討を行った結果、IDO 法で移流計算のために補間関数を形成する手続きが計算軸方向重視であるのに対し、CIP 法では移流原点位置を含む計算格子全体に対して補間関数を形成することから広範囲に渡り空間精度の高い物理量計算が可能であることが要因として挙げられることを明らかにした。従って、IDO 法による多次元移流計算では CFL 数の制限が必然的に生じることを示した。移流方程式、保存方程式を扱い差分法、CIP 法による数値計算結果との比較を行った。IDO 法は非移流相計算を高次精度で行っていることから、CIP 法に比べ高精度数値計算が可能であることを明らかにした。移流相計算を行うために風上 5 次補間を用いることで更に精度向上が行えることを明らかにした。これらの結果から少ない格子点数で複雑な流体問題を扱える可能性を示した。

時間発展する圧縮性流体問題に対する適用性を検討した。1 次元衝撃波管問題を扱い、IDO 法に対する人工粘性の適用性について検討した結果、圧縮領域の判定手法、人工粘性の係数に関して安定に導入することを可能にした。複雑な流体構造を示す前線発達問題を扱い、継続的に高波数構造が発達する問題に対しては移流方程式から見積もられた空間精度  $\Delta x^3$  を得ることが困難であった。その原因として移流方程式の空間精度は、高波数領域において  $\Delta x^3$  を示さず、 $\Delta x^1 \sim \Delta x^2$  まで低下することが挙げられる。このことから高波数プロファイルが誘起される問題に対する計算精度向上が困難であることを示した。その一方で差分法では多くの計算格子点を必要とする複雑な流体構造を示すレーリー・テラー不安定性の発達問題を非常に少ない計算格子点数で行った結果、線形成長時の成長率が高精度で求められ、マッシュルーム構造の発達が得られたことから複雑な流体構造を示す問題への IDO 法による流体計算の有効性を明ら

かにした。超音速混合流問題を扱うため、異なる 2 流体の混合を可能にする手法としてそれぞれの流体に対する識別関数を導入する手法を提案した。得られた計算結果から超音速流混合の主な要因としてノズル先端部の有限厚さに起因する流体擾乱が上流部に生成される衝撃波を誘起し、流体構造を乱すとともに下流域に向かい擾乱が成長することで混合領域における渦構造発達を促進していることを挙げた。実験から観測された微小衝撃波列の生成要因として流体境界領域に発生する Kelvin-Helmholtz 不安定性に起因する周期的渦列が一つとして挙げられることを明らかにした。また、ノズル接合部後流が Kelvin-Helmholtz 不安定性により成長することから、実験で用いたような条件下では流体間速度差は混合を促進するための重要な因子ではないことを明らかにした。

次に、非圧縮性流体問題への適用性を検討した。非圧縮性流体問題を数値計算で扱う場合、圧力波が無限小の時間で伝搬することから計算ステップ毎にポアソン方程式を解く必要がある。そのため、圧縮性流体問題の解法に比べ計算時間が増大することから、非圧縮性流体問題の高速解法を試みた。ポアソン方程式の精度検証から計算空間次元数、境界条件を問わず  $\Delta x^4$  を示すことを明らかにし、差分法に比べ少ない格子点数で高精度にポアソン方程式が解けることを示した。ポアソン方程式の収束計算を高速化するため、従来差分法で緩和計算を高速化する手法として用いられて来たマルチグリッド法を IDO 法に適用した。2 次元ポアソン方程式の検証から SOR 法に比べ 36 倍の高速計算を可能にし、物理量に加え空間微係数を従属変数として持つ IDO 法に対するマルチグリッド法の適用性を明らかにした。マルチグリッド法は計算格子点数が増大するに従いその有効性を示すため、3 次元ポアソン方程式の解法では計算の並列化やベクトル化がより有効であると考え、計算格子点間の情報依存性を排除することを目的として差分法で用いられているレッドブラック法を適用し、複数の従属変数を持つ IDO 法に対してその収束性を数値的に明らかにした。また、並列化を行い 3 次元ポアソン方程式を扱った結果から、計算格子点数が少ない場合でも高いスケラビリティが得られることを明らかにした。非圧縮性流体問題の他の高速化手法として、大きな  $\Delta t$  を用いることにより目的計算時間までにポアソン方程式を解く回数を減らすことにより高速化が可能であると考え、セミ・ラグランジュ手法の導入を行った。CIP 法に対し、CFL 数が 1 を越えても安定に計算が行えるよう、移流経路計算に対して 4 次精度 Runge-Kutta 法ならびに仮想時間ステップを導入し、非一様速度場における計算精度の改善を行った。また、移流原点を含む計算格子に対して補間関数を形成する手法を採り入れることで補間関数の外挿による数値不安定性を改善した。移流方程式による検証からセミ・ラグランジュ CIP 法を用いることで計算時間の短縮を可能にし、計算時間毎に生じる数値拡散を低減することを可能にした。非圧縮性流体問題として 2 次元正方キャビティ流れを扱い、CFL 数を大きくすることで CIP 法に比べ 16 倍の高速計算を可能にした。

IDO 法は、有限差分法に比べ少ない計算格子点数で高精度数値計算を可能にする優れた手法であるが、陽解法を用いた場合には有限差分法同様に CFL (Courant-Friedrich-Levy) 数により計算時間刻幅 ( $\Delta t$ ) が制限される。とりわけ多次元問題を扱う場合には CFL 数は 0.1 程度までしか利用できない。このような問題を回避するため、陰解法化による数値計算の安定化ならびに大きな  $\Delta t$  に対する有効化を図った。その結果、移流方程式に対しては CFL 数が 1 を越えても数値安定性を保証する結果を得た。非線形方程式ならびに流体方程式に対しても陽解法と比較して大きな CFL 数に対する安定計算を可能にし、特に拡散方程式には有効な手段であることを明らかにした。

オイラーグリッドを用いた数値計算では不連続面を扱う場合、不連続面を跨いで有限次数の多項式により補間を行うため、数値拡散が必ず生じる。この問題を解消することを目的としてこれまでに境界条件を記述する手法として用いられて来たカットセル手法を計算空間中を自由に移動可能なものとして扱う移動カットセル手法 (Moving Cut Cell Method) を提案した。不連続面により識別される異なる物体のそれぞれの密度を同じ計算グリッド上に与えることで接触不連続面 (Contact Discontinuity) を扱うことを可能にし、その移動を Level Set 手法により行うことで不連続面を持つ問題を低数値拡散誤差で解くことを可能にした。

IDO 法を用いた数値流体計算の特徴として計算コストが挙げられる。差分法による数値計算に比べ少ない格子点数で高波数プロファイルを高精度で扱えることから、これまで大規模計算機による数値計算が必要とされた問題に対して計算格子点数を大幅に減少させることが可能である。計算格子点数の削減は計算

時間の短縮をもたらし、簡便な高精度計算が行えることを意味している。本論文で行った流体計算の例題からもその有効性は明らかである。さらに、非圧縮性流体解法の高速度化、陰解法化による安定した計算手法の開発、オイラーグリッド上における接触不連続面記述を可能にした新しい計算手法の提案を行い、さまざまな流体問題への適用性を向上した。

本論文で開発した高速かつ高精度数値流体計算手法は、様々な分野における設計や開発、最適化に貢献できる。

## 付録 A

# エルミート補間を用いた計算手法

計算グリッド点上に物理量 ( $f$ ) とその空間微係数 ( $\partial f/\partial x \dots$ ) を物理現象を記述する偏微分方程式 (Partial Differential Equation; PDE) によって関連付けられた従属変数として定義し、局所的にエルミート (Hermite) 補間を形成する数値計算手法の代表的なものとして Cubic-Interpolated Propagation (CIP) 法 [1, 2] や、局所補間微分オペレータ法 (Interpolated Differential Operator; IDO) 法 [3] が挙げられる。エルミート補間を利用することによりグリッド間プロファイルを有限差分法 (FiniteDifference Method; FDM) に比べ少ないグリッド数 (局所的) で高精度で表現することが可能であることから、有限差分法に比べ少ない計算格子点数で精度の高い計算が可能であることがさまざまな論文で示されている。

本章ではエルミート補間を用いた計算手法である CIP 法と IDO 法の概略について説明する。

### A.1 エルミート (Hermite) 補間

エルミート補間とは、離散化された 1 次元空間 ( $x_i$ ) における関数値  $f_i$  とその空間 1 階微分値  $f_{x,i}$  が与えられた場合、その空間上の  $n$  点 ( $x_1, x_2, \dots, x_n$ ) における離散値 ( $f_1, f_2, \dots, f_n$ ) と、その 1 階空間微分値 ( $f'_1, f'_2, \dots, f'_n$ );  $f' = \partial f/\partial x$  を満足するように形成される  $2n-1$  次以下の多項式を指す。CIP 法および IDO 法で主に用いられているエルミート補間は 2 点のグリッドポイント間に形成する 3 次多項式で、以下のように表される。

$$F(x) = ax^3 + bx^2 + f_{x,i}x + f_i \quad (\text{A.1})$$

ここで、添字の  $i$  はグリッド番号を表し、 $f_x$  は、物理量  $f$  の 1 階空間微分値を表す ( $f_x = \partial f/\partial x$ )。例えば、グリッド  $i-1$  からグリッド  $i$  にかけて補間関数を形成した場合の係数  $a, b$  はそれぞれグリッド  $i-1$  における適合条件  $F(-\Delta x) = f_{i-1}$ ,  $F_x(-\Delta x) = f_{x,i-1}$  を用いることで、以下のように求められる。

$$\begin{aligned} a &= -3 \frac{f_i - f_{i-1}}{\Delta x^2} + \frac{f_{x,i-1} + 2f_{x,i}}{\Delta x} \\ b &= -2 \frac{f_i - f_{i-1}}{\Delta x^3} + \frac{f_{x,i-1} + f_{x,i}}{\Delta x^2} \end{aligned} \quad (\text{A.2})$$

補間関数の概念図を図 A.1 に示す。線形補間を行った場合に比べ、グリッド間の情報をより高解像度で表現できることに加え、物理量のみでグリッド間情報を表現する有限差分法とは異なり、局所的に補間を行うことから、より精度の高い数値計算が行える。

### A.2 Cubic-Interpolated Propagation (CIP) 法

CIP 法は圧縮性流体 (compressible fluid)、非圧縮性流体 (incompressible fluid) の統一解法として提案されている。

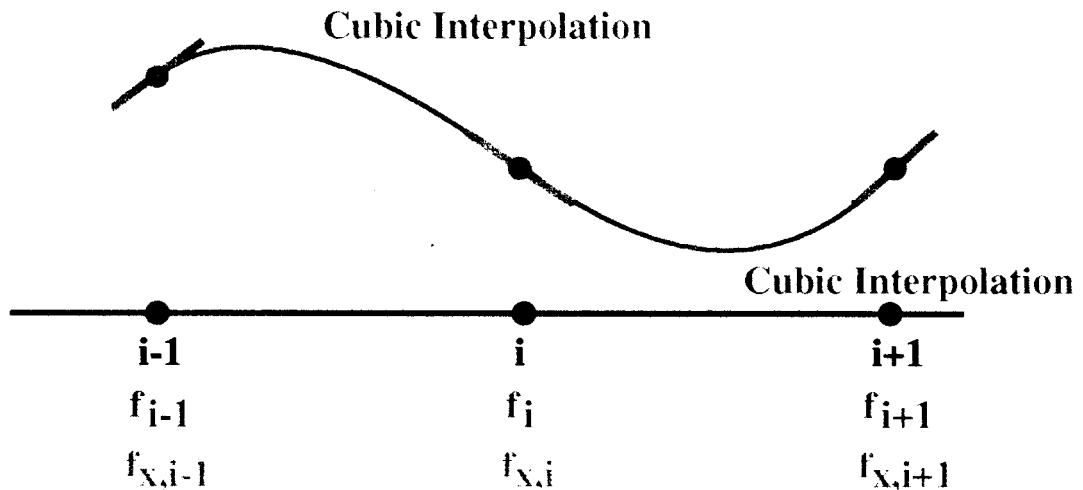


図 A.1: 3 次エルミート補間を用いた場合のグリッド間プロファイル  
 エルミート補間を用いることでグリッド間情報を線形補間に比べ精度良く表現できる。線形補間で同等以上の解像度を出そうとすれば少なくとも 2 倍以上のメッシュ数が必要となる。

CIP 法は当初 Cubic Interpolated Pseudo-particle Method として紹介され [1, 7, 8]、CIP1、CIP2 [4]、Compact CIP [9]、現在の CIP 法と発展してきた。また、補間関数に有理関数を用いる有理関数 CIP 法 [66] や、移流原点位置の計算に 4 次精度 Runge-Kutta 手法を採り入れた Differential Algebraic CIP (DA-CIP) 法 [67, 68] なども提案されている。近年では非保存系による保存を保証した CIP 法 [69, 70] が登場し、さらなる発展が見込まれるスキームである。

CIP 法を用いた数値計算例としてこれまでに固体、液体、気体、プラズマを同時に扱うレーザー加工のシミュレーション、液体の衝突により形成されるミルククラウンのシミュレーション、更に、計算空間に大きな密度差のあるシューメイカー・レビー彗星の衝突シミュレーションなど、様々な成果を挙げている。

### A.2.1 1 次元 CIP 法

CIP 法を考える場合、その大まかな概念は 1 次元移流方程式を持つてとらえると分かりやすい。そこで、以下の偏微分方程式を考える。

$$f_t + \nabla \cdot f u = g \tag{A.3}$$

ここで、添字の  $t$  は、時間  $t$  による物理量  $f$  の微分を表す ( $f_t = \partial f / \partial t$ )。CIP 法では移流相 (advection phase) と、非移流相 (non-advection phase) を分離して考えるので、ここで  $G \equiv g - f \nabla \cdot u$  を導入し、式 (A.3) を次のように書きなおす。

$$f_t + u \nabla \cdot f = G \tag{A.4}$$

式 (A.4) の移流相、非移流相はそれぞれ以下のように表される。

$$f_t + u f_x = 0 \quad (\text{advection phase}) \tag{A.5}$$

$$f_t = G \quad (\text{non-advection phase}) \tag{A.6}$$

簡単のため速度場が時間、空間ともに一定と仮定すると ( $u_t = 0, u_x = 0$ )、移流相は解析的に解くことができ、その解は  $F(x, t) = F(x - ut, 0)$  で与えられる。つまり、速度  $u$  で移流する波を表している。1 次

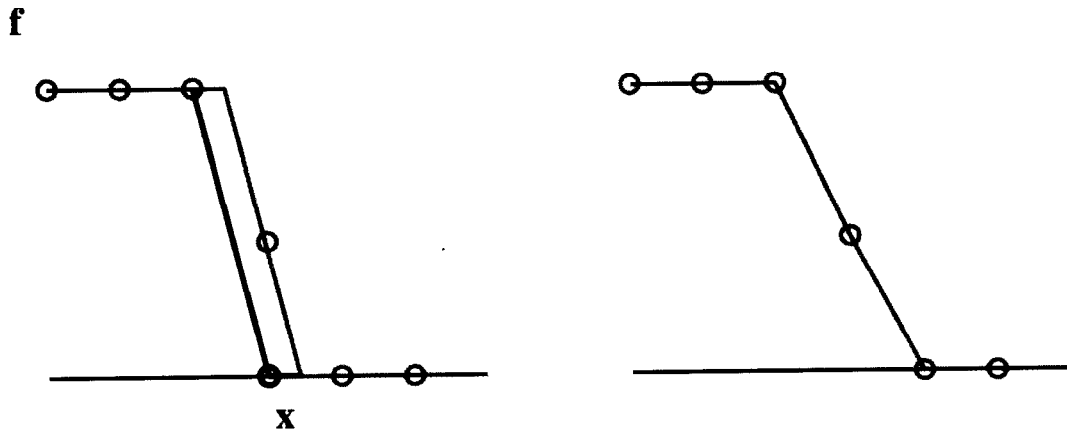


図 A.2: 一様定常速度場における 1 次元移流方程式の 1 次風上差分法による計算結果  
 1 次風上差分法を用いた場合、計算結果が解析解に一致した結果が得られていても離散化されたプロファイル直線近似するため少なからず数値拡散が生じる。

元移流方程式をオイラーグリッド上で 1 次風上差分法により扱う場合、図 A.2 に示すように、不連続面におけるプロファイルは次第に数値拡散する。一方 CIP 法では、プロファイルを記述するために物理量に加え、その空間微係数を用いたプロファイルの記述を行うため 図 A.3 に示すように、有限差分法に比べ補間による数値拡散を大幅に低減することが出来る。CIP 法は 1 階空間微分値を従属変数として扱うため、

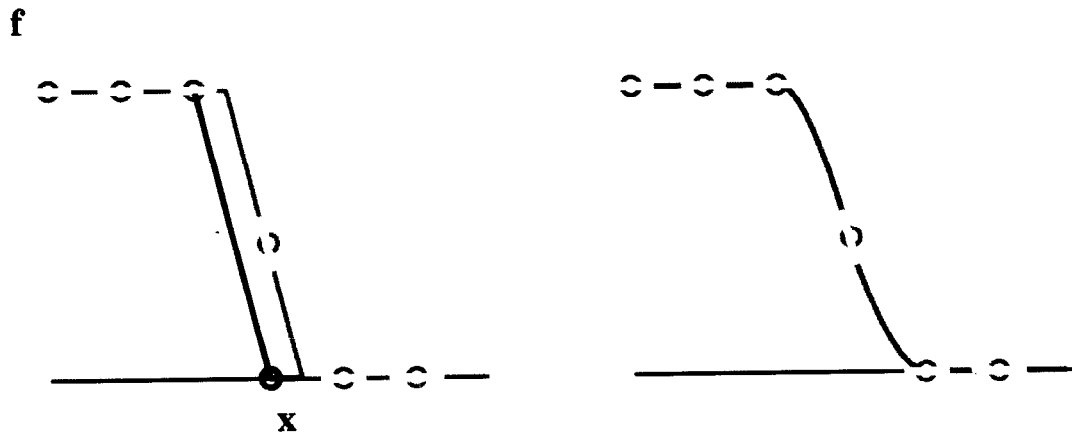


図 A.3: 一様定常速度場における 1 次元移流方程式の CIP 法による計算結果  
 CIP 法では空間微係数を用いてプロファイルを記述するため、有限差分法に比べ補間による数値拡散を低減することが出来る。

式 (A.4) を空間微分し、移流相と非移流相に分離した以下の式も同時に解かなければならない。

$$(f_x)_t + u(f_x)_x = 0 \quad (\text{advection phase}) \tag{A.7}$$

$$(f_x)_t = G_x - u_x f_x \quad (\text{non-advection phase}) \tag{A.8}$$



空間微係数の移流相に関しては式 (A.5) と同形であることから、 $F_x(x, t) = F_x(x - ut, 0)$  と扱えることが容易に推測できる。CIP 法では非移流相を一般的に 2 次精度中心差分法などの有限差分法を用いた手法により積分する。以上をまとめ、CIP 法による数値流体計算手法の具体的な解法は以下ようになる。

移流相の計算では、移流速度の風向きからエルミート補間を行う領域を決定する。グリッド  $i$  における移流速度  $u_i$  が正であれば、情報は風上、つまりグリッド  $i-1$  側から流れて来るので、グリッド  $i \sim i-1$  間にエルミート補間を形成する。移流速度が負であれば逆にグリッド  $i+1 \sim i$  間に補間関数を形成する。用いる補間関数は式 (A.1) で示される 3 次多項式を用いる。

$$\begin{cases} F(-\Delta x) = f_{i-1}, & F_x(-\Delta x) = f_{x,i-1} & \text{if } u_i \geq 0 \\ F(\Delta x) = f_{i+1}, & F_x(\Delta x) = f_{x,i+1} & \text{if } u_i < 0 \end{cases} \quad (\text{A.9})$$

補間関数形成領域が決定されると、上記の適合条件により式 (A.1) の各未知係数  $a, b$  は連立方程式を解くことで以下のように求められる。

$$\begin{cases} a = -3 \frac{f_i - f_{i-1}}{\Delta x^2} + \frac{f_{x,i-1} + 2f_{x,i}}{\Delta x}, \\ b = -2 \frac{f_i - f_{i-1}}{\Delta x^3} + \frac{f_{x,i-1} + f_{x,i}}{\Delta x^2}, & \text{if } u_i \geq 0 \\ a = 3 \frac{f_{i+1} - f_i}{\Delta x^2} - \frac{f_{x,i+1} + 2f_{x,i}}{\Delta x}, \\ b = -2 \frac{f_{i+1} - f_i}{\Delta x^3} + \frac{f_{x,i+1} + f_{x,i}}{\Delta x^2}. & \text{if } u_i < 0 \end{cases}$$

これらの値を用いることで、 $\Delta t$  後の物理量とその 1 階空間微分値は移流方程式の解析解を用いることでそれぞれ以下のように求められる。

$$f_i^* = F(x_i - u_i \Delta t) = a\xi^3 + b\xi^2 + f_{x,i}\xi + f_i \quad (\text{A.10})$$

$$f_{x,i}^* = F_x(x_i - u_i \Delta t) = 3a\xi^2 + 2b\xi + f_{x,i} \quad (\text{A.11})$$

ここで、 $\xi = -u_i \Delta t$  とした。

CIP 法における非移流相計算では、物理量、その 1 階空間微分値について下記のように見積もる。

$$f_i^{n+1} = f_i^* + G_i^* \Delta t \quad (\text{A.12})$$

$$f_{x,i}^{n+1} = f_{x,i}^* + \frac{f_{i+1}^{n+1} - f_{i-1}^{n+1}}{2\Delta x} - \frac{f_{i+1}^* - f_{i-1}^*}{2\Delta x} - u_{x,i} f_{x,i}^* \Delta t \quad (\text{A.13})$$

式 (A.13) の右辺第 2 項、第 3 項は、非移流相における  $G$  の計算を 2 次精度中心差分法で行うことを意味している。

## A.2.2 2 次元 CIP 法

2 次元 CIP 法も移流相を解析的に扱い、非移流相を有限差分法により求めるスキームである。CIP 法で 2 次元問題を扱うには Type-A, Type-B, Type-C の 3 通りの手法が提案されている [71]。本論文中での 2 次元 CIP 法は 1 次元 CIP を重ね合わせることで多次元化を行う Type-C を採用した。Type-A, Type-B, Type-C の大まかな概要を以下に示す。説明を簡単にするため、直交座標系  $x, y$  における以下の 2 次元移流方程式を考える。

$$f_t + u f_x + v f_y = g \quad (\text{A.14})$$

1次元の場合と同様に CIP 法は物理量の微係数も独立変数として扱うことでエルミート補間を形成するので式 (A.14) の他に下記の 3 式も同時に解く必要性が生じる。

$$\begin{aligned} (f_x)_t + u(f_x)_x + v(f_x)_y \\ = g_x - u_x f_x - v_x f_y = R_x \end{aligned} \quad (\text{A.15})$$

$$\begin{aligned} (f_y)_t + u(f_y)_x + v(f_y)_y \\ = g_y - u_y f_x - v_y f_y = R_y \end{aligned} \quad (\text{A.16})$$

$$\begin{aligned} (f_{xy})_t + u(f_{xy})_x + v(f_{xy})_y \\ = g_{xy} - u_x f_{xy} - u_y f_{xx} - u_{xy} f_x \\ - v_x f_{yy} - v_y f_{xy} - v_{xy} f_y \\ = R_{xy} \end{aligned} \quad (\text{A.17})$$

ただし、 $f_{xy}$  は、Type - C でのみ使用される。移流相と非移流相に分離すると下記のように表される。

$$\begin{aligned} \text{移流相} \quad & f_t + u f_x + v f_y = 0 \\ & (f_x)_t + u(f_x)_x + v(f_x)_y = 0 \\ & (f_y)_t + u(f_y)_x + v(f_y)_y = 0 \\ & (f_{xy})_t + u(f_{xy})_x + v(f_{xy})_y = 0 \\ \text{非移流相} \quad & f_t = g \\ & (f_x)_t = R_x \\ & (f_y)_t = R_y \\ & (f_{xy})_t = R_{xy} \end{aligned}$$

移流相に関しては解析解  $f(x, y, t + \Delta t) = f(x - u\Delta t, y - v\Delta t, t)$  を用いることでその解を得ることができ。つまり、CIP 法の補間関数により移流原点  $(-u\Delta t, -v\Delta t)$  に於ける物理量、その 1 階空間微分値等を補間することで解析的に移流相の計算を行うことができる。Type - A, B, C それぞれにおける移流相の計算手法、特に補間関数の形成方法に着目し、以下に述べる。

#### Type - A

Type - A では、2 次元の補間関数を考える。その形は以下の通りである。

$$F(x, y) = C_{3,0}X^3 + C_{2,0}X^2 + \alpha_{i,j}X + f_{i,j} + C_{0,3}Y^3 + C_{0,2}Y^2 + \beta_{i,j}Y + C_{2,1}X^2Y + C_{1,1}XY + C_{1,2}XY \quad (\text{A.18})$$

ここで、 $\alpha = f_x = \partial f / \partial x$ ,  $\beta = f_y = \partial f / \partial y$  と略記した。グリッド  $(i, j)$  における移流速度  $u_{i,j}$  により、補間関数を形成する領域が決定するが、ここでは  $u_{i,j} < 0$ ,  $v_{i,j} < 0$  とし、領域  $(i, j)$ ,  $(i+1, j)$ ,  $(i, j+1)$ ,  $(i+1, j+1)$  で補間関数を形成する (図 A.4)。適合条件  $F(\Delta x, 0) = f_{i+1,j}$ ,  $F_x(\Delta x, 0) = \alpha_{i+1,j}$ ,  $F_y(\Delta x, 0) = \beta_{i+1,j}$ ,  $F(0, \Delta y) = f_{i,j+1}$ ,  $F_x(0, \Delta y) = \alpha_{i,j+1}$ ,  $F_y(0, \Delta y) = \beta_{i,j+1}$ ,  $F(\Delta x, \Delta y) = f_{i+1,j+1}$  を用いることで、各係数  $(C_{3,0}, C_{2,0}, C_{0,3}, C_{0,2}, C_{2,1}, C_{1,1}, C_{1,2})$  を求めることが出来る。

$$\begin{aligned} C_{3,0} &= [-2f_{i+1,j} + 2f_{i,j} + (\alpha_{i+1,j} + \alpha_{i,j})\Delta x] / \Delta x^3, \\ C_{2,1} &= [A - (\alpha_{i+1,j} - \alpha_{i,j})\Delta x] / \Delta x^2 \Delta y, \\ C_{2,0} &= [3f_{i+1,j} - 3f_{i,j} - (\alpha_{i+1,j} + 2\alpha_{i,j})\Delta x] / \Delta x^2, \\ C_{1,1} &= [-A + (\alpha_{i,j+1} - \alpha_{i,j})\Delta x + (\beta_{i+1,j} - \beta_{i,j})\Delta y] / \Delta y^3, \\ C_{0,3} &= [-2f_{i,j+1} + 2f_{i,j} + (\beta_{i,j+1} + \beta_{i,j})\Delta y] / \Delta y^3, \end{aligned}$$

$$C_{1,2} = [A - (\beta_{i+1,j} - \beta_{i,j} \Delta y)] / \Delta x \Delta y^2,$$

$$C_{0,2} = [3f_{i,j+1} - 3f_{i,j} - (\beta_{i,j+1} - \beta_{i,j}) \Delta y] / \Delta y^2$$

ここで、 $A = f_{i,j} - f_{i+1,j} - f_{i,j+1} - f_{i+1,j+1}$  とした。 $C_{3,3}, C_{3,2}, C_{2,3}, C_{3,1}, C_{1,3}, C_{2,2}$  は未知数であるから 0 とした。 $X \equiv x - x_i, Y \equiv y - y_i$  とすることで、式 (A.18) から移流原点における値を得ることができる。

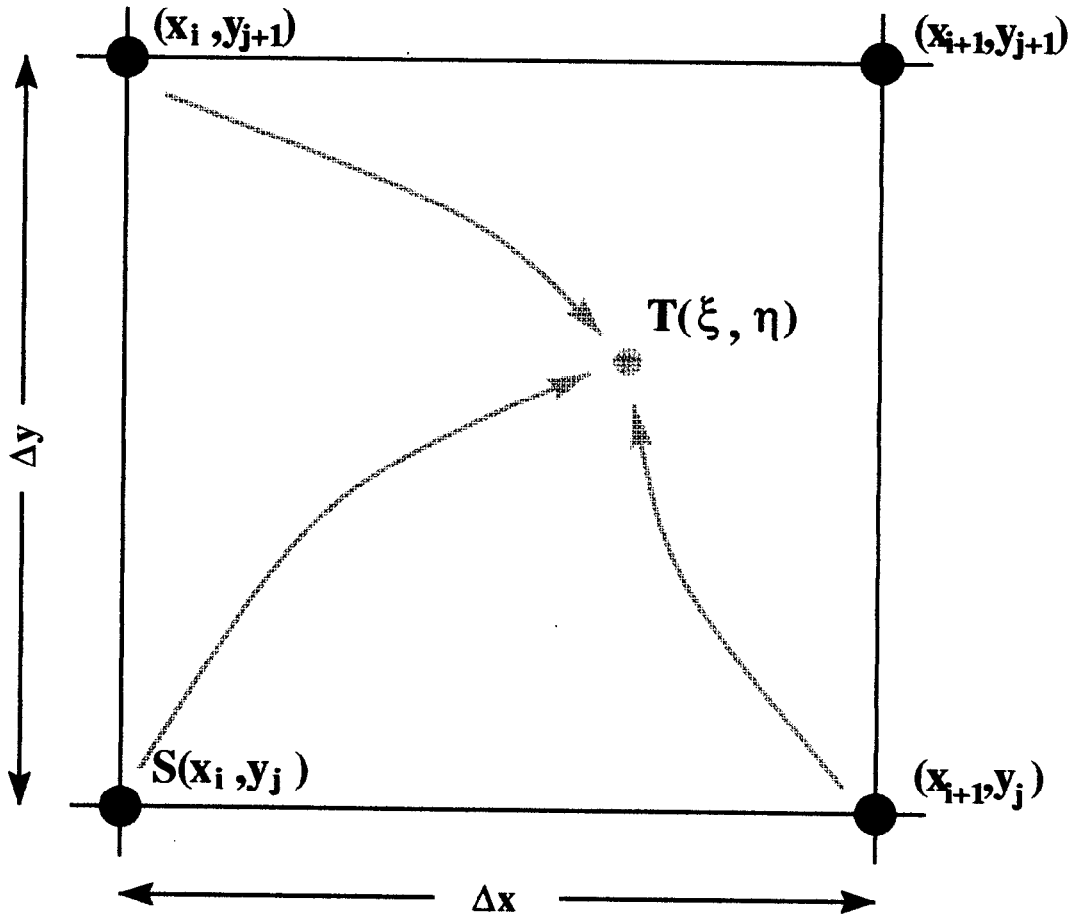


図 A.4: CIP Type-A による 2 次元空間の補間

計算対象グリッド点  $S$  に於ける移流速度  $u_{i,j}$  から風上方向を検知し、2 次元空間に於ける風上方向の 3 点における物理量とその 1 階空間微係数の連続条件、対角点における物理量の連続条件を利用して補間関数を形成することで移流原点の値を見積もる。

#### Type - B

Type - B も Type - A と同様に 2 次元の補関関数

$$\begin{aligned} F(x, y) = & C_{3,0}X^3 + C_{2,0}X^2 + \alpha_{i,j}X + f_{i,j} \\ & + C_{0,3}Y^3 + C_{0,2}Y^2 + \beta_{i,j}Y + C_{3,1}X^3Y \\ & + C_{2,1}X^2Y + C_{1,2}XY^2 + C_{1,3}XY^3 \\ & + C_{1,1}XY \end{aligned} \quad (\text{A.19})$$

を用いる。補間をより正確に行うために  $C_{3,1}X^3Y, C_{1,3}XY^3$  を Type - A の補関式に加えている。グリッド  $(i, j)$  における移流速度  $u_{i,j}$  の風向きによって補関関数が形成される領域が決まるが、ここでは

$u_{i,j} < 0, v_{i,j} < 0$  として、補間関数を領域  $(i, j), (i+1, j), (i, j+1), (i+1, j+1)$  に形成する (図 A.5)。適合条件  $F(\Delta x, 0) = f_{i+1,j}, F_x(\Delta x, 0) = \alpha_{i+1,j}, F_y(\Delta x, 0) = \beta_{i+1,j}, F(0, \Delta y) = f_{i,j+1}, F_x(0, \Delta y) = \alpha_{i,j+1}, F_y(0, \Delta y) = \beta_{i,j+1}, F(\Delta x, \Delta y) = f_{i+1,j+1}, F_x(\Delta x, \Delta y) = \alpha_{i+1,j+1}, F_y(\Delta x, \Delta y) = \beta_{i+1,j+1}$  を用いることで、各係数は以下のように求まる。

$$\begin{aligned} C_{3,0} &= 2(f_{i,j} - f_{i+1,j})/\Delta x^3 + (\alpha_{i,j} + \alpha_{i+1,j})/\Delta x^2, \\ C_{2,0} &= 3(-f_{i,j} + f_{i+1,j})/\Delta x^2 - (2\alpha_{i,j} + \alpha_{i+1,j})/\Delta x, \\ C_{0,3} &= 2(f_{i,j} - f_{i,j+1})/\Delta y^3 + (\beta_{i,j} + \beta_{i,j+1})/\Delta y^2, \\ C_{0,2} &= 3(-f_{i,j} + f_{i,j+1})/\Delta y^2 - (2\beta_{i,j} + \beta_{i,j+1})/\Delta y, \\ C_{3,1} &= -2A/\Delta x^3\Delta y + (-\alpha_{i,j} - \alpha_{i+1,j} + \alpha_{i,j+1} + \alpha_{i+1,j+1})/\Delta x^2\Delta y, \\ C_{2,1} &= 3A/\Delta x^2\Delta y + (2\alpha_{i,j} + \alpha_{i+1,j} - 2\alpha_{i,j+1} - \alpha_{i+1,j+1})/\Delta x\Delta y, \\ C_{1,3} &= -2A/\Delta x\Delta y^3 + (-\beta_{i,j} + \beta_{i+1,j} - \beta_{i,j+1} + \beta_{i+1,j+1})/\Delta x\Delta y^2, \\ C_{1,2} &= 3A/\Delta x\Delta y^2 + (2\beta_{i,j} - 2\beta_{i+1,j} + \beta_{i,j+1} + \beta_{i+1,j+1})/\Delta x\Delta y, \\ C_{1,1} &= -A/\Delta x\Delta y + (-\alpha_{i,j} + \alpha_{i,j+1})/\Delta y + (-\beta_{i,j} + \beta_{i+1,j})/\Delta x\Delta y. \end{aligned}$$

ここで、 $A = f_{i,j} - f_{i+1,j} - f_{i,j+1} - f_{i+1,j+1}, X \equiv x - x_i, Y \equiv y - y_i$  である。 $C_{3,3}, C_{3,2}, C_{2,3}, C_{2,2}$  は未知数であるから 0 とした。Type - B は、Type - A で用いていなかったグリッド  $(i+1, j+1)$  の情報を使うことで Type - A では未知数であった  $C_{3,1}, C_{1,3}$  を求めることを可能にし、より精度の良い補間を行っている。

### Type - C

Type - C は 1 次元 CIP 法を多数回使用することで多次元化を行う方法である。この方法を用いれば 3 次元化も簡単に行うことが出来る。Type - C は、Type - A, Type - B とは異なり、補間関数は 1 次元 3 次多項式を使用する。また必要性から、 $\delta = f_{xy} = \partial^2 f / \partial x \partial y$  も従属変数として各グリッド点毎に定義する。

グリッド点  $(i, j)$  における移流速度  $u_{i,j}$  の成分  $u_{i,j}, v_{i,j}$  がそれぞれ負である場合、補間関数は領域  $(x_i, y_j), (x_{i+1}, y_j), (x_i, y_{j+1}), (x_{i+1}, y_{j+1})$  で形成される。1 次元 CIP 法を多数回使用して  $T(\xi, \eta)$  を見積もる Type-C の方法は  $S \rightarrow A(\xi, y_j) \rightarrow T(\xi, \eta)$  と  $S \rightarrow B(x_i, \eta) \rightarrow T(\xi, \eta)$  の 2 通りあるが、ここでは  $S \rightarrow A \rightarrow T$  のルートを用いる手法を紹介する (図 A.6)。また、 $\xi = -u_{i,j}\Delta t, \eta = -v_{i,j}\Delta t$  とした。 $A(\xi, y_j)$  での見積もりは  $S(x_i, y_j), (x_{i+1}, y_j)$  間に 1 次元補間関数を形成することで、

$$f_{(A)} = a_1\xi^3 + b_1\xi^2 + \alpha_{i,j}\xi + f_{i,j} \quad (\text{A.20})$$

$$\alpha_{(A)} = 3a_1\xi^2 + 2b_1\xi + \alpha_{i,j} \quad (\text{A.21})$$

$$\beta_{(A)} = a_2\xi^3 + b_2\xi^2 + \delta_{i,j}\xi + \beta_{i,j} \quad (\text{A.22})$$

$$\delta_{(A)} = 3a_2\xi^2 + 2b_2\xi + \delta_{i,j} \quad (\text{A.23})$$

$$\begin{aligned} a_1 &= \frac{\alpha_{i,j} + \alpha_{i+1,j}}{\Delta x^2} - 2\frac{f_{i,j} - f_{i+1,j}}{\Delta x^3}, \\ b_1 &= 3\frac{f_{i+1,j} - f_{i,j}}{\Delta x^2} - \frac{\alpha_{i+1,j} + 2\alpha_{i,j}}{\Delta x} \end{aligned} \quad (\text{A.24})$$

$$\begin{aligned} a_2 &= \frac{\delta_{i,j} + \delta_{i+1,j}}{\Delta x^2} - 2\frac{\beta_{i,j} - \beta_{i+1,j}}{\Delta x^3}, \\ b_2 &= 3\frac{\beta_{i+1,j} - \beta_{i,j}}{\Delta x^2} - \frac{\delta_{i+1,j} + 2\delta_{i,j}}{\Delta x} \end{aligned} \quad (\text{A.25})$$

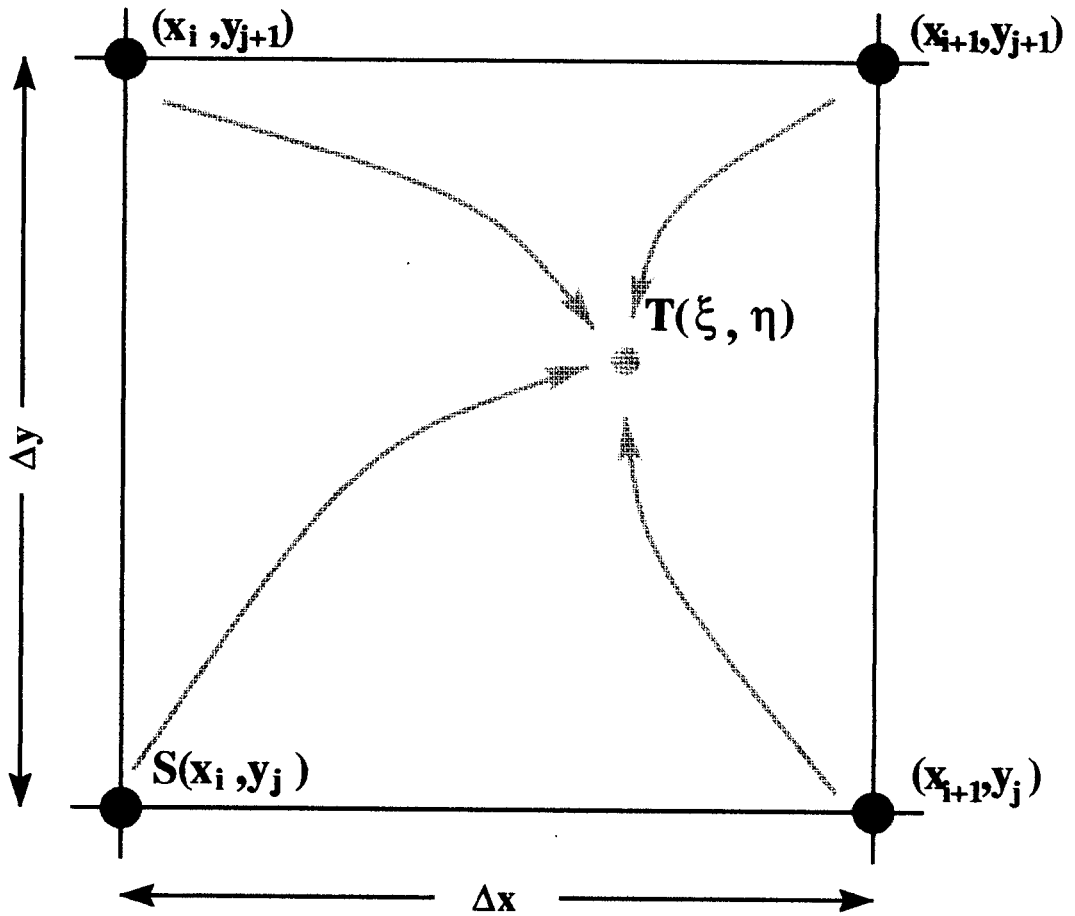


図 A.5: CIP Type-B による 2 次元空間の補間

計算対象グリッド点  $S$  での移流速度  $u_{i,j}$  から風上方向を検知し、移流原点を含む領域を構成するグリッド点全てを使って 2 次元補間関数を形成する。

となる。同様にして  $A'(\xi, y_{j+1})$  を見積もると以下のようなになる。

$$f_{(A')} = a_1 \xi^3 + b_1 \xi^2 + \alpha_{i,j} \xi + f_{i,j} \quad (\text{A.26})$$

$$\alpha_{(A')} = 3a_1 \xi^2 + 2b_1 \xi + \alpha_{i,j} \quad (\text{A.27})$$

$$\beta_{(A')} = a_2 \xi^3 + b_2 \xi^2 + \delta_{i,j} \xi + \beta_{i,j} \quad (\text{A.28})$$

$$\delta_{(A')} = 3a_2 \xi^2 + 2b_2 \xi + \delta_{i,j} \quad (\text{A.29})$$

$$a_1 = \frac{\alpha_{i,j+1} + \alpha_{i+1,j+1}}{\Delta x^2} - 2 \frac{f_{i,j+1} - f_{i+1,j+1}}{\Delta x^3}, \quad (\text{A.30})$$

$$b_1 = 3 \frac{f_{i+1,j+1} - f_{i,j+1}}{\Delta x^2} - \frac{\alpha_{i+1,j+1} + 2\alpha_{i,j+1}}{\Delta x} \quad (\text{A.31})$$

$$a_2 = \frac{\delta_{i,j+1} + \delta_{i+1,j+1}}{\Delta x^2} - 2 \frac{\beta_{i,j+1} - \beta_{i+1,j+1}}{\Delta x^3}, \quad (\text{A.32})$$

$$b_2 = 3 \frac{\beta_{i+1,j+1} - \beta_{i,j+1}}{\Delta x^2} - \frac{\delta_{i+1,j+1} + 2\delta_{i,j+1}}{\Delta x} \quad (\text{A.33})$$

$A, A'$  間で 1 次元補間関数を形成することで、 $T(\xi, \eta)$  を以下のように見積もることができる。

$$f_{i,j}^* = f_{(T)} = a_1 \eta^3 + b_1 \eta^2 + \beta_{(A)} \eta + f_{(A)} \quad (\text{A.34})$$

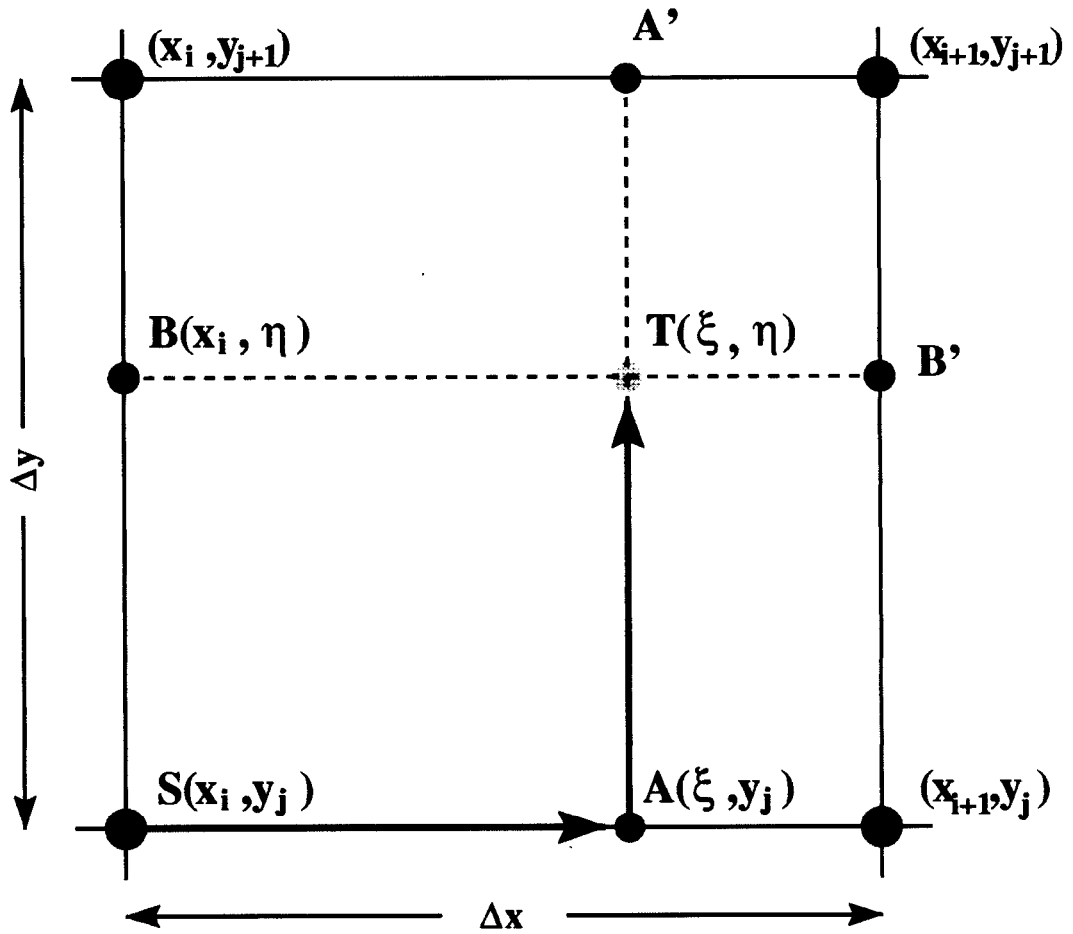


図 A.6: CIP Type-C による 2 次元空間の補間

計算対象グリッド点  $S$  に於ける移流速度  $u_{i,j}$  から、風上方向を検知し、移流原点を含む領域で 1 次元 CIP 法で用いた 3 次精度エルミート補間を組み合わせることで移流原点に於ける物理量を補間する。

$$\alpha_{i,j}^* = \alpha(T) = a_2 \eta^3 + b_2 \eta^2 + \delta_{(A)} \eta + \alpha_{(A)} \quad (\text{A.35})$$

$$\beta_{i,j}^* = \beta(T) = 3a_1 \eta^2 + 2b_1 \eta + \beta_{(A)} \quad (\text{A.36})$$

$$\delta_{i,j}^* = \delta(T) = 3a_2 \eta^2 + 2b_2 \eta + \delta_{(A)} \quad (\text{A.37})$$

$$a_1 = \frac{\beta_{(A)} + \beta_{(A')}}{\Delta y^2} - 2 \frac{f_{(A)} - f_{(A')}}{\Delta y^3}, \quad (\text{A.38})$$

$$b_1 = 3 \frac{f_{(A')} - f_{(A)}}{\Delta y^2} - \frac{\beta_{(A')} + 2\beta_{(A)}}{\Delta y} \quad (\text{A.39})$$

$$a_2 = \frac{\delta_{(A)} + \delta_{(A')}}{\Delta y^2} - 2 \frac{\alpha_{(A)} - \alpha_{(A')}}{\Delta y^3}, \quad (\text{A.40})$$

$$b_2 = 3 \frac{\alpha_{(A')} - \alpha_{(A)}}{\Delta y^2} - \frac{\delta_{(A')} + 2\delta_{(A)}}{\Delta y} \quad (\text{A.41})$$

### 非移流相

2 次元 CIP 法のための非移流相の計算について述べる。Type - A, B, C に関わらず、非移流相は 2 次精度中心差分法を用いて次のように計算される。

$$f_{i,j}^{n+1} = f_{i,j}^* + g\Delta t \quad (\text{A.42})$$

$$\alpha_{i,j}^{n+1} = \alpha_{i,j}^* + \frac{f_{i+1,j}^{n+1} - f_{i-1,j}^{n+1}}{2\Delta x} - \frac{f_{i+1,j}^* - f_{i-1,j}^*}{2\Delta x} - \frac{\partial u}{\partial x} \Big|_{i,j} \alpha_{i,j}^* \Delta t - \frac{\partial v}{\partial x} \Big|_{i,j} \beta_{i,j}^* \Delta t \quad (\text{A.43})$$

$$\beta_{i,j}^{n+1} = \beta_{i,j}^* + \frac{f_{i,j+1}^{n+1} - f_{i,j-1}^{n+1}}{2\Delta y} - \frac{f_{i,j+1}^* - f_{i,j-1}^*}{2\Delta y} - \frac{\partial u}{\partial y} \Big|_{i,j} \alpha_{i,j}^* \Delta t - \frac{\partial v}{\partial y} \Big|_{i,j} \beta_{i,j}^* \Delta t \quad (\text{A.44})$$

$$\delta_{i,j}^{n+1} = \delta_{i,j}^* + \frac{f_{i+1,j+1}^{n+1} - f_{i-1,j+1}^{n+1} - f_{i+1,j-1}^{n+1} + f_{i-1,j-1}^{n+1}}{4\Delta x\Delta y} - \frac{f_{i+1,j+1}^* - f_{i-1,j+1}^* - f_{i+1,j-1}^* + f_{i-1,j-1}^*}{4\Delta x\Delta y} - \frac{\partial^2 u}{\partial x\partial y} \Big|_{i,j} \alpha_{i,j}^* - \frac{\partial^2 v}{\partial x\partial y} \Big|_{i,j} \beta_{i,j}^* \quad (\text{A.45})$$

### A.2.3 3次元 CIP 法

3次元 CIP 法も 1次元 CIP 法の特徴を受け継ぎ、計算空間を補間関数により Remapping し、計算対象グリッド点に対する移流原点位置の物理量ならびに空間微係数を移流方程式の解析解に従いセミラグランジュ的に移流させる計算スキームである。現在の所 3次元 CIP 法では Type-A, Type-C の補間手法が提案されている。

#### Type-A

3次元 CIP 法 Type-A で用いる補間関数は

$$\begin{aligned} F(x, y, z) = & [(B1\eta + B2\mu + B3\zeta + B4)\eta + B5\mu + \partial_x f_{i,j,k}] \eta \\ & + [(B6\mu + B7\zeta + B8\eta + B9)\mu + B10\zeta + \partial_y f_{i,j,k}] \mu \\ & + B16\eta\mu\zeta + f_{i,j,k} \end{aligned} \quad (\text{A.46})$$

であらわされる。ここで、 $\eta = -u_{i,j,k}\Delta t$ ,  $\mu = -v_{i,j,k}\Delta t$ ,  $\zeta = -w_{i,j,k}\Delta t$  とした。 $u_{i,j,k} < 0$ ,  $v_{i,j,k} < 0$ ,  $w_{i,j,k} < 0$  である場合、補間関数は計算領域  $(i, j, k) - (i+1, j, k) - (i, j+1, k) - (i+1, j+1, k) - (i, j, k+1) - (i+1, j, k+1) - (i, j+1, k+1) - (i+1, j+1, k+1)$  に形成され、グリッド  $(i+1, j, k)$ ,  $(i, j+1, k)$ ,  $(i, j, k+1)$  における物理量  $f$  とその空間微係数  $\partial_x f$ ,  $\partial_y f$ ,  $\partial_z f$  の連続条件とグリッド  $(i+1, j+1, k)$ ,  $(i, j+1, k+1)$ ,  $(i+1, j, k+1)$ ,  $(i+1, j+1, k+1)$  における物理量  $f$  の連続条件から未知係数  $B1 \sim B16$  は次のように決定される。

$$B1 = [-2D_i + \partial_x(f_{i+1,j,k} + f_{i,j,k})\Delta x] / \Delta x^3 \quad (\text{A.47})$$

$$B2 = -[B17 + \partial_x D_j \Delta x] / \Delta x^2 \Delta y \quad (\text{A.48})$$

$$B3 = -[B18 + \partial_x D_k \Delta x] / \Delta x^2 \Delta y \quad (\text{A.49})$$

$$B4 = [3D_i - \partial_x(f_{i+1,j,k} + 2f_{i,j,k})\Delta x] / \Delta x^2 \quad (\text{A.50})$$

$$B5 = [B17 + \partial_x D_j \Delta x + \partial_y D_i \Delta y] / \Delta x \Delta y \quad (\text{A.51})$$

$$B6 = [-2D_j + \partial_y(f_{i,j+1,k} + f_{i,j,k})\Delta y] / \Delta y^3 \quad (\text{A.52})$$

$$B7 = -[B19 + \partial_y D_k \Delta y] / \Delta y^2 \Delta z \quad (\text{A.53})$$

$$B8 = -[B17 + \partial_y D_i \Delta y] / \Delta x \Delta y^2 \quad (\text{A.54})$$

$$B9 = [3D_j - \partial_y (f_{i,j+1,k} + 2f_{i,j,k}) \Delta y] / \Delta y^2 \quad (\text{A.55})$$

$$B10 = [B19 + \partial_y D_k \Delta y + \partial_z D_j \Delta z] / \Delta y \Delta z \quad (\text{A.56})$$

$$B11 = [-2D_k + \partial_z (f_{i,j,k+1} + f_{i,j,k}) \Delta z] / \Delta z^3 \quad (\text{A.57})$$

$$B12 = -[B18 + \partial_z D_i \Delta z] / \Delta x \Delta z^2 \quad (\text{A.58})$$

$$B13 = -[B19 + \partial_z D_j \Delta z] / \Delta y \Delta z^2 \quad (\text{A.59})$$

$$B14 = [3D_k - \partial_z D_i \Delta z + \partial_z (f_{i,j,k+1} + 2f_{i,j,k}) \Delta z] / \Delta z^2 \quad (\text{A.60})$$

$$B15 = [B18 + \partial_z D_i \Delta z + \partial_x D_k \Delta x] / \Delta x \Delta z \quad (\text{A.61})$$

$$B16 = [B17 + f_{i,j,k+1} - (f_{i,j+1,k+1} + f_{i+1,j,k+1}) + f_{i+1,j+1,k+1}] / \Delta x \Delta y \Delta z \quad (\text{A.62})$$

$$B17 = -f_{i,j,k} + f_{i+1,j,k} + f_{i,j+1,k} - f_{i+1,j+1,k} \quad (\text{A.63})$$

$$B18 = -f_{i,j,k} + f_{i+1,j,k} + f_{i,j,k+1} - f_{i+1,j,k+1} \quad (\text{A.64})$$

$$B19 = -f_{i,j,k} + f_{i,j+1,k} + f_{i,j,k+1} - f_{i,j+1,k+1} \quad (\text{A.65})$$

$$D_i = f_{i+1,j,k} - f_{i,j,k} \quad (\text{A.66})$$

$$D_j = f_{i,j+1,k} - f_{i,j,k} \quad (\text{A.67})$$

$$D_k = f_{i,j,k+1} - f_{i,j,k} \quad (\text{A.68})$$

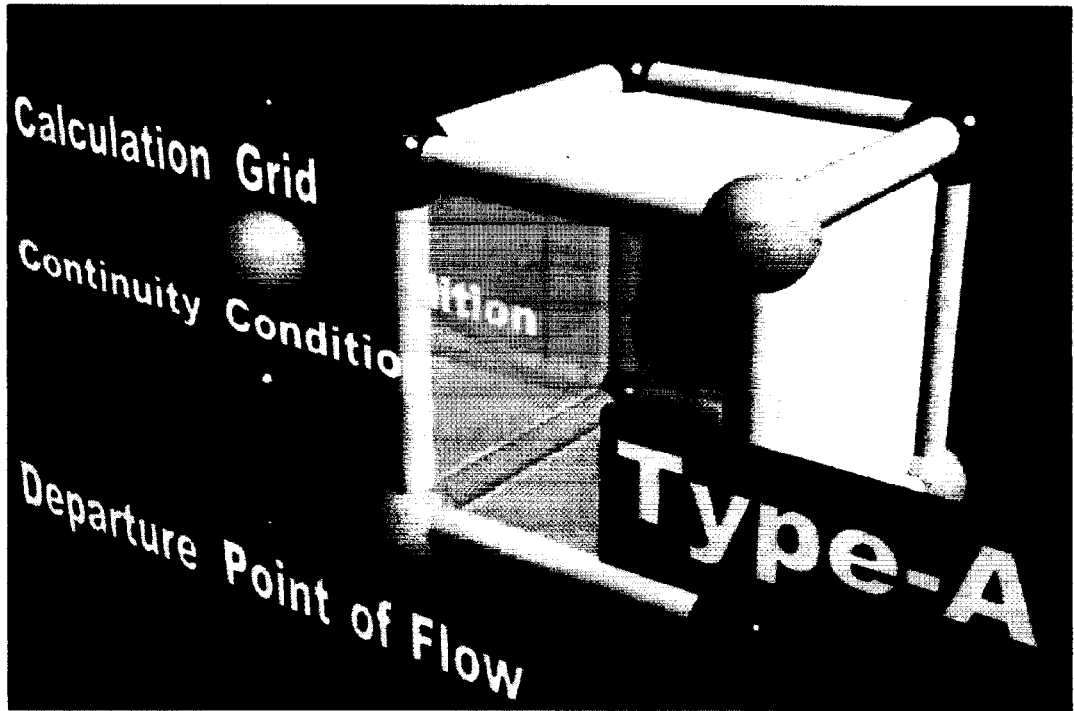


図 A.7: 3次元 CIP 法 Type-A 型補間

3次元 Type-A では、計算対象点 (赤点) 近傍の 3 点 (橙点) における物理量とその一階空間微係数の連続条件ならびにその他の 4 点 (緑点) における物理量の連続条件を用い、補間関数を形成する。

### Type-C

3次元 Type-C は、3次元 Type-A で用いられる物理量  $f$ 、1階空間微係数  $\partial f / \partial x, \partial f / \partial y, \partial f / \partial z$  に加え 2階、3階の空間微係数  $\partial^2 f / \partial x \partial y, \partial^2 f / \partial x \partial z, \partial^2 f / \partial y \partial z, \partial^3 f / \partial x \partial y \partial z$  を用いて移流原点位置における



物理量を補間する。

3次元 CIP 法における Type-C の Remapping 計算手順は、2次元 CIP 法 Type-C と同様に、1次元 CIP 法で用いられる補間関数を多数回使用することにより表記される。その補間経路の採り方は様々であるが、ここでは代表的に図 A.8 に示す手順で移流原点位置における物理量ならびにその空間微係数の計算方法を示す。

2次元 CIP 法 Type-C で用いた計算手順を用いて  $B(\xi, \eta, z_k), B'(\xi, \eta, z_{k+1})$  における物理量ならびに空間微係数を決定する。点 B における物理量ならびに微係数を決定するために点 A、A' における物理量、空間微係数を決定する必要がある。点 A における物理量は次の補間により決定される。

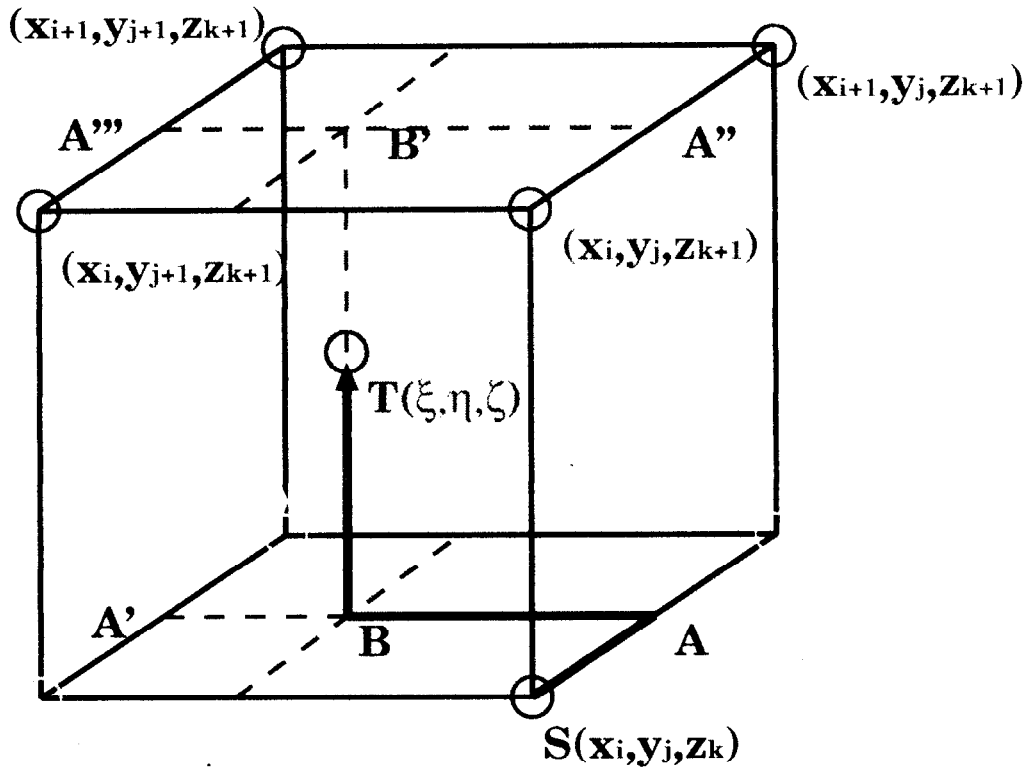


図 A.8: 3次元 CIP 法 Type-C の補間手順

1次元 CIP 法で用いられる補間関数を多数回使用することにより移流原点位置における物理量ならびに空間微係数を補間する。

$$f(A) = a_1 \xi^3 + b_1 \xi^2 + \partial_x f_{i,j,k} \xi + f_{i,j,k}, \tag{A.69}$$

$$\partial_x f(A) = 3a_1 \xi^2 + 2b_1 \xi + \partial_x f_{i,j,k}, \tag{A.70}$$

$$\partial_y f(A) = a_2 \xi^3 + b_2 \xi^2 + \partial_{xy} f_{i,j,k} \xi + \partial_y f_{i,j,k}, \tag{A.71}$$

$$\partial_z f(A) = a_3 \xi^3 + b_3 \xi^2 + \partial_{xz} f_{i,j,k} \xi + \partial_z f_{i,j,k}, \tag{A.72}$$

$$\partial_{xy} f(A) = 3a_2 \xi^2 + 2b_2 \xi + \partial_{xy} f_{i,j,k}, \tag{A.73}$$

$$\partial_{xz} f(A) = 3a_3 \xi^2 + 2b_3 \xi + \partial_{xz} f_{i,j,k}, \tag{A.74}$$

$$\partial_{yz} f(A) = a_4 \xi^3 + b_4 \xi^2 + \partial_{xyz} f_{i,j,k} \xi + \partial_{yz} f_{i,j,k}, \tag{A.75}$$

$$\partial_{xyz} f(A) = 3a_4 \xi^2 + 2b_4 \xi + \partial_{xyz} f_{i,j,k}, \tag{A.76}$$

$$a_1 = \frac{\partial_x f_{i,j,k} + \partial_x f_{i+1,j,k}}{\Delta x^2} - 2 \frac{f_{i,j,k} - f_{i+1,j,k}}{\Delta x^3}, \quad (\text{A.77})$$

$$b_1 = 3 \frac{f_{i+1,j,k} - f_{i,j,k}}{\Delta x^2} - \frac{\partial_x f_{i+1,j,k} + 2\partial_x f_{i,j,k}}{\Delta x}, \quad (\text{A.78})$$

$$a_2 = \frac{\partial_{xy} f_{i,j,k} + \partial_{xy} f_{i+1,j,k}}{\Delta x^2} - 2 \frac{\partial_y f_{i,j,k} - \partial_y f_{i+1,j,k}}{\Delta x^3}, \quad (\text{A.79})$$

$$b_2 = 3 \frac{\partial_y f_{i+1,j,k} - \partial_y f_{i,j,k}}{\Delta x^2} - \frac{\partial_{xy} f_{i+1,j,k} + 2\partial_{xy} f_{i,j,k}}{\Delta x}, \quad (\text{A.80})$$

$$a_3 = \frac{\partial_{xz} f_{i,j,k} + \partial_{xz} f_{i+1,j,k}}{\Delta x^2} - 2 \frac{\partial_z f_{i,j,k} - \partial_z f_{i+1,j,k}}{\Delta x^3}, \quad (\text{A.81})$$

$$b_3 = 3 \frac{\partial_z f_{i+1,j,k} - \partial_z f_{i,j,k}}{\Delta x^2} - \frac{\partial_{xz} f_{i+1,j,k} + 2\partial_{xz} f_{i,j,k}}{\Delta x}, \quad (\text{A.82})$$

$$a_4 = \frac{\partial_{xyz} f_{i,j,k} + \partial_{xyz} f_{i+1,j,k}}{\Delta x^2} - 2 \frac{\partial_{yz} f_{i,j,k} - \partial_{yz} f_{i+1,j,k}}{\Delta x^3}, \quad (\text{A.83})$$

$$b_4 = 3 \frac{\partial_{yz} f_{i+1,j,k} - \partial_{yz} f_{i,j,k}}{\Delta x^2} - \frac{\partial_{xyz} f_{i+1,j,k} + 2\partial_{xyz} f_{i,j,k}}{\Delta x}. \quad (\text{A.84})$$

同様にして  $A'(\xi, y_{j+1}, z_k)$  における諸量を見積もると以下ようになる。

$$f(A') = a_1 \xi^3 + b_1 \xi^2 + \partial_x f_{i,j,k} \xi + f_{i,j,k}, \quad (\text{A.85})$$

$$\partial_x f(A') = 3a_1 \xi^2 + 2b_1 \xi + \partial_x f_{i,j,k}, \quad (\text{A.86})$$

$$\partial_y f(A') = a_2 \xi^3 + b_2 \xi^2 + \partial_{xy} f_{i,j,k} \xi + \partial_y f_{i,j,k}, \quad (\text{A.87})$$

$$\partial_z f(A') = a_3 \xi^3 + b_3 \xi^2 + \partial_{xz} f_{i,j,k} \xi + \partial_z f_{i,j,k}, \quad (\text{A.88})$$

$$\partial_{xy} f(A') = 3a_2 \xi^2 + 2b_2 \xi + \partial_{xy} f_{i,j,k}, \quad (\text{A.89})$$

$$\partial_{xz} f(A') = 3a_3 \xi^2 + 2b_3 \xi + \partial_{xz} f_{i,j,k}, \quad (\text{A.90})$$

$$\partial_{yz} f(A') = a_4 \xi^3 + b_4 \xi^2 + \partial_{xyz} f_{i,j,k} \xi + \partial_{yz} f_{i,j,k}, \quad (\text{A.91})$$

$$\partial_{xyz} f(A') = 3a_4 \xi^2 + 2b_4 \xi + \partial_{xyz} f_{i,j,k}, \quad (\text{A.92})$$

$$a_1 = \frac{\partial_x f_{i,j+1,k} + \partial_x f_{i+1,j+1,k}}{\Delta x^2} - 2 \frac{f_{i,j+1,k} - f_{i+1,j+1,k}}{\Delta x^3}, \quad (\text{A.93})$$

$$b_1 = 3 \frac{f_{i+1,j+1,k} - f_{i,j+1,k}}{\Delta x^2} - \frac{\partial_x f_{i+1,j+1,k} + 2\partial_x f_{i,j+1,k}}{\Delta x}, \quad (\text{A.94})$$

$$a_2 = \frac{\partial_{xy} f_{i,j+1,k} + \partial_{xy} f_{i+1,j+1,k}}{\Delta x^2} - 2 \frac{\partial_y f_{i,j+1,k} - \partial_y f_{i+1,j+1,k}}{\Delta x^3}, \quad (\text{A.95})$$

$$b_2 = 3 \frac{\partial_y f_{i+1,j+1,k} - \partial_y f_{i,j+1,k}}{\Delta x^2} - \frac{\partial_{xy} f_{i+1,j+1,k} + 2\partial_{xy} f_{i,j+1,k}}{\Delta x}, \quad (\text{A.96})$$

$$a_3 = \frac{\partial_{xz} f_{i,j+1,k} + \partial_{xz} f_{i+1,j+1,k}}{\Delta x^2} - 2 \frac{\partial_z f_{i,j+1,k} - \partial_z f_{i+1,j+1,k}}{\Delta x^3}, \quad (\text{A.97})$$

$$b_3 = 3 \frac{\partial_z f_{i+1,j+1,k} - \partial_z f_{i,j+1,k}}{\Delta x^2} - \frac{\partial_{xz} f_{i+1,j+1,k} + 2\partial_{xz} f_{i,j+1,k}}{\Delta x}, \quad (\text{A.98})$$

$$a_4 = \frac{\partial_{xyz} f_{i,j+1,k} + \partial_{xyz} f_{i+1,j+1,k}}{\Delta x^2} - 2 \frac{\partial_{yz} f_{i,j+1,k} - \partial_{yz} f_{i+1,j+1,k}}{\Delta x^3}, \quad (\text{A.99})$$

$$b_4 = 3 \frac{\partial_{yz} f_{i+1,j+1,k} - \partial_{yz} f_{i,j+1,k}}{\Delta x^2} - \frac{\partial_{xyz} f_{i+1,j+1,k} + 2\partial_{xyz} f_{i,j+1,k}}{\Delta x}. \quad (\text{A.100})$$

$A, A'$  間で 1 次元補間関数を形成することで、 $B(\xi, \eta, z_k)$  における諸量を以下のように見積もることができる。

$$f(B) = a_1 \eta^3 + b_1 \eta^2 + \partial_y f(A) \eta + f(A), \quad (\text{A.101})$$

$$\partial_x f(B) = a_2 \eta^3 + b_2 \eta^2 + \partial_{xy} f(A) \eta + \partial_x f(A), \quad (\text{A.102})$$

$$\partial_y f(B) = 3a_1 \eta^2 + 2b_1 \eta + \partial_y f(A), \quad (\text{A.103})$$

$$\partial_z f(B) = a_3 \eta^3 + b_3 \eta^2 + \partial_{yz} f(A) \eta + \partial_z f(A), \quad (\text{A.104})$$

$$\partial_{xy} f(B) = 3a_2 \eta^2 + 2b_2 \eta + \partial_{xy} f(A), \quad (\text{A.105})$$

$$\partial_{xz} f(B) = a_4 \eta^3 + b_4 \eta^2 + \partial_{xyz} f(A) \eta + \partial_{xz} f(A), \quad (\text{A.106})$$

$$\partial_{yz} f(B) = 3a_3 \eta^2 + 2b_3 \eta + \partial_{yz} f(A), \quad (\text{A.107})$$

$$\partial_{xyz} f(B) = 3a_4 \eta^2 + 2b_4 \eta + \partial_{xyz} f(A), \quad (\text{A.108})$$

$$a_1 = \frac{\partial_y f(A) + \partial_y f(A')}{\Delta y^2} - 2 \frac{f(A) - f(A')}{\Delta y^3}, \quad (\text{A.109})$$

$$b_1 = 3 \frac{f(A') - f(A)}{\Delta y^2} - \frac{\partial_y f(A') + 2\partial_y f(A)}{\Delta y}, \quad (\text{A.110})$$

$$a_2 = \frac{\partial_{xy} f(A) + \partial_{xy} f(A')}{\Delta y^2} - 2 \frac{\partial_x f(A) - \partial_x f(A')}{\Delta y^3}, \quad (\text{A.111})$$

$$b_2 = 3 \frac{\partial_x f(A') - \partial_x f(A)}{\Delta y^2} - \frac{\partial_{xy} f(A') + 2\partial_{xy} f(A)}{\Delta y}, \quad (\text{A.112})$$

$$a_3 = \frac{\partial_{yz} f(A) + \partial_{yz} f(A')}{\Delta y^2} - 2 \frac{\partial_z f(A) - \partial_z f(A')}{\Delta y^3}, \quad (\text{A.113})$$

$$b_3 = 3 \frac{\partial_z f(A') - \partial_z f(A)}{\Delta y^2} - \frac{\partial_{yz} f(A') + 2\partial_{yz} f(A)}{\Delta y}, \quad (\text{A.114})$$

$$a_4 = \frac{\partial_{xyz} f(A) + \partial_{xyz} f(A')}{\Delta y^2} - 2 \frac{\partial_{xz} f(A) - \partial_{xz} f(A')}{\Delta y^3}, \quad (\text{A.115})$$

$$b_4 = 3 \frac{\partial_{xz} f(A') - \partial_{xz} f(A)}{\Delta y^2} - \frac{\partial_{xyz} f(A') + 2\partial_{xyz} f(A)}{\Delta y}. \quad (\text{A.116})$$

同様にして A'', A''' を 1 次元補間により計算し、B' における諸量を補間する手続きは次の通りである。

$$f(A'') = a_1 \xi^3 + b_1 \xi^2 + \partial_x f_{i,j,k+1} \xi + f_{i,j,k+1}, \quad (\text{A.117})$$

$$\partial_x f(A'') = 3a_1 \xi^2 + 2b_1 \xi + \partial_x f_{i,j,k+1}, \quad (\text{A.118})$$

$$\partial_y f(A'') = a_2 \xi^3 + b_2 \xi^2 + \partial_{xy} f_{i,j,k+1} \xi + \partial_y f_{i,j,k+1}, \quad (\text{A.119})$$

$$\partial_z f(A'') = a_3 \xi^3 + b_3 \xi^2 + \partial_{xz} f_{i,j,k+1} \xi + \partial_z f_{i,j,k+1}, \quad (\text{A.120})$$

$$\partial_{xy} f(A'') = 3a_2 \xi^2 + 2b_2 \xi + \partial_{xy} f_{i,j,k+1}, \quad (\text{A.121})$$

$$\partial_{xz} f(A'') = 3a_3 \xi^2 + 2b_3 \xi + \partial_{xz} f_{i,j,k+1}, \quad (\text{A.122})$$

$$\partial_{yz} f(A'') = a_4 \xi^3 + b_4 \xi^2 + \partial_{xyz} f_{i,j,k+1} \xi + \partial_{yz} f_{i,j,k+1}, \quad (\text{A.123})$$

$$\partial_{xyz} f(A'') = 3a_4 \xi^2 + 2b_4 \xi + \partial_{xyz} f_{i,j,k+1}, \quad (\text{A.124})$$

$$a_1 = \frac{\partial_x f_{i,j,k+1} + \partial_x f_{i+1,j,k+1}}{\Delta x^2} - 2 \frac{f_{i,j,k+1} - f_{i+1,j,k+1}}{\Delta x^3}, \quad (\text{A.125})$$

$$b_1 = 3 \frac{f_{i+1,j,k+1} - f_{i,j,k+1}}{\Delta x^2} - \frac{\partial_x f_{i+1,j,k+1} + 2\partial_x f_{i,j,k+1}}{\Delta x}, \quad (\text{A.126})$$

$$a_2 = \frac{\partial_{xy} f_{i,j,k+1} + \partial_{xy} f_{i+1,j,k+1}}{\Delta x^2} - 2 \frac{\partial_y f_{i,j,k+1} - \partial_y f_{i+1,j,k+1}}{\Delta x^3}, \quad (\text{A.127})$$

$$b_2 = 3 \frac{\partial_y f_{i+1,j,k+1} - \partial_y f_{i,j,k+1}}{\Delta x^2} - \frac{\partial_{xy} f_{i+1,j,k+1} + 2\partial_{xy} f_{i,j,k+1}}{\Delta x}, \quad (\text{A.128})$$

$$a_3 = \frac{\partial_{xz} f_{i,j,k+1} + \partial_{xz} f_{i+1,j,k+1}}{\Delta x^2} - 2 \frac{\partial_z f_{i,j,k+1} - \partial_z f_{i+1,j,k+1}}{\Delta x^3}, \quad (\text{A.129})$$

$$b_3 = 3 \frac{\partial_z f_{i+1,j,k+1} - \partial_z f_{i,j,k+1}}{\Delta x^2} - \frac{\partial_{xz} f_{i+1,j,k+1} + 2\partial_{xz} f_{i,j,k+1}}{\Delta x}, \quad (\text{A.130})$$

$$a_4 = \frac{\partial_{xyz} f_{i,j,k+1} + \partial_{xyz} f_{i+1,j,k+1}}{\Delta x^2} - 2 \frac{\partial_{yz} f_{i,j,k+1} - \partial_{yz} f_{i+1,j,k+1}}{\Delta x^3}, \quad (\text{A.131})$$

$$b_4 = 3 \frac{\partial_{yz} f_{i+1,j,k+1} - \partial_{yz} f_{i,j,k+1}}{\Delta x^2} - \frac{\partial_{xyz} f_{i+1,j,k+1} + 2\partial_{xyz} f_{i,j,k+1}}{\Delta x}, \quad (\text{A.132})$$

$$f_{(A''')} = a_1 \xi^3 + b_1 \xi^2 + \partial_x f_{i,j,k+1} \xi + f_{i,j,k+1}, \quad (\text{A.133})$$

$$\partial_x f_{(A''')} = 3a_1 \xi^2 + 2b_1 \xi + \partial_x f_{i,j,k+1}, \quad (\text{A.134})$$

$$\partial_y f_{(A''')} = a_2 \xi^3 + b_2 \xi^2 + \partial_{xy} f_{i,j,k+1} \xi + \partial_y f_{i,j,k+1}, \quad (\text{A.135})$$

$$\partial_z f_{(A''')} = a_3 \xi^3 + b_3 \xi^2 + \partial_{xz} f_{i,j,k+1} \xi + \partial_z f_{i,j,k+1}, \quad (\text{A.136})$$

$$\partial_{xy} f_{(A''')} = 3a_2 \xi^2 + 2b_2 \xi + \partial_{xy} f_{i,j,k+1}, \quad (\text{A.137})$$

$$\partial_{xz} f_{(A''')} = 3a_3 \xi^2 + 2b_3 \xi + \partial_{xz} f_{i,j,k+1}, \quad (\text{A.138})$$

$$\partial_{yz} f_{(A''')} = a_4 \xi^3 + b_4 \xi^2 + \partial_{xyz} f_{i,j,k+1} \xi + \partial_{yz} f_{i,j,k+1}, \quad (\text{A.139})$$

$$\partial_{xyz} f_{(A''')} = 3a_4 \xi^2 + 2b_4 \xi + \partial_{xyz} f_{i,j,k+1}, \quad (\text{A.140})$$

$$a_1 = \frac{\partial_x f_{i,j+1,k+1} + \partial_x f_{i+1,j+1,k+1}}{\Delta x^2} - 2 \frac{f_{i,j+1,k+1} - f_{i+1,j+1,k+1}}{\Delta x^3}, \quad (\text{A.141})$$

$$b_1 = 3 \frac{f_{i+1,j+1,k+1} - f_{i,j+1,k+1}}{\Delta x^2} - \frac{\partial_x f_{i+1,j+1,k+1} + 2\partial_x f_{i,j+1,k+1}}{\Delta x}, \quad (\text{A.142})$$

$$a_2 = \frac{\partial_{xy} f_{i,j+1,k+1} + \partial_{xy} f_{i+1,j+1,k+1}}{\Delta x^2} - 2 \frac{\partial_y f_{i,j+1,k+1} - \partial_y f_{i+1,j+1,k+1}}{\Delta x^3}, \quad (\text{A.143})$$

$$b_2 = 3 \frac{\partial_y f_{i+1,j+1,k+1} - \partial_y f_{i,j+1,k+1}}{\Delta x^2} - \frac{\partial_{xy} f_{i+1,j+1,k+1} + 2\partial_{xy} f_{i,j+1,k+1}}{\Delta x}, \quad (\text{A.144})$$

$$a_3 = \frac{\partial_{xz} f_{i,j+1,k+1} + \partial_{xz} f_{i+1,j+1,k+1}}{\Delta x^2} - 2 \frac{\partial_z f_{i,j+1,k+1} - \partial_z f_{i+1,j+1,k+1}}{\Delta x^3}, \quad (\text{A.145})$$

$$b_3 = 3 \frac{\partial_z f_{i+1,j+1,k+1} - \partial_z f_{i,j+1,k+1}}{\Delta x^2} - \frac{\partial_{xz} f_{i+1,j+1,k+1} + 2\partial_{xz} f_{i,j+1,k+1}}{\Delta x}, \quad (\text{A.146})$$

$$a_4 = \frac{\partial_{xyz} f_{i,j+1,k+1} + \partial_{xyz} f_{i+1,j+1,k+1}}{\Delta x^2} - 2 \frac{\partial_{yz} f_{i,j+1,k+1} - \partial_{yz} f_{i+1,j+1,k+1}}{\Delta x^3}, \quad (\text{A.147})$$

$$b_4 = 3 \frac{\partial_{yz} f_{i+1,j+1,k+1} - \partial_{yz} f_{i,j+1,k+1}}{\Delta x^2} - \frac{\partial_{xyz} f_{i+1,j+1,k+1} + 2\partial_{xyz} f_{i,j+1,k+1}}{\Delta x}. \quad (\text{A.148})$$

$A''$ ,  $A'''$  間で 1 次元補間関数を形成することで、 $B'(\xi, \eta, z_{k+1})$  における諸量を以下のように見積もることができる。

$$f_{(B')} = a_1 \eta^3 + b_1 \eta^2 + \partial_y f_{(A'')} \eta + f_{(A'')}, \quad (\text{A.149})$$

$$\partial_x f_{(B')} = a_2 \eta^3 + b_2 \eta^2 + \partial_{xy} f_{(A'')} \eta + \partial_x f_{(A'')}, \quad (\text{A.150})$$

$$\partial_y f_{(B')} = 3a_1 \eta^2 + 2b_1 \eta + \partial_y f_{(A'')}, \quad (\text{A.151})$$

$$\partial_z f_{(B')} = a_3 \eta^3 + b_3 \eta^2 + \partial_{yz} f_{(A'')} \eta + \partial_z f_{(A'')}, \quad (\text{A.152})$$

$$\partial_{xy} f_{(B')} = 3a_2 \eta^2 + 2b_2 \eta + \partial_{xy} f_{(A'')}, \quad (\text{A.153})$$

$$\partial_{xz} f_{(B')} = a_4 \eta^3 + b_4 \eta^2 + \partial_{xyz} f_{(A'')} \eta + \partial_{xz} f_{(A'')}, \quad (\text{A.154})$$

$$\partial_{yz} f_{(B')} = 3a_3 \eta^2 + 2b_3 \eta + \partial_{yz} f_{(A'')}, \quad (\text{A.155})$$

$$\partial_{xyz} f_{(B')} = 3a_4 \eta^2 + 2b_4 \eta + \partial_{xyz} f_{(A'')}, \quad (\text{A.156})$$

$$a_1 = \frac{\partial_y f_{(A'')} + \partial_y f_{(A''')}}{\Delta y^2} - 2 \frac{f_{(A'')} - f_{(A''')}}{\Delta y^3}, \quad (\text{A.157})$$

$$b_1 = 3 \frac{f_{(A''')} - f_{(A'')}}{\Delta y^2} - \frac{\partial_y f_{(A''')} + 2\partial_y f_{(A'')}}{\Delta y}, \quad (\text{A.158})$$

$$a_2 = \frac{\partial_{xy} f_{(A'')} + \partial_{xy} f_{(A''')}}{\Delta y^2} - 2 \frac{\partial_x f_{(A'')} - \partial_x f_{(A''')}}{\Delta y^3}, \quad (\text{A.159})$$

$$b_2 = 3 \frac{\partial_x f(A''') - \partial_x f(A'')}{\Delta y^2} - \frac{\partial_{xy} f(A''') + 2\partial_{xy} f(A'')}{\Delta y}, \quad (\text{A.160})$$

$$a_3 = \frac{\partial_{yz} f(A'') + \partial_{yz} f(A''')}{\Delta y^2} - 2 \frac{\partial_z f(A'') - \partial_z f(A''')}{\Delta y^3}, \quad (\text{A.161})$$

$$b_3 = 3 \frac{\partial_z f(A''') - \partial_z f(A'')}{\Delta y^2} - \frac{\partial_{yz} f(A''') + 2\partial_{yz} f(A'')}{\Delta y}, \quad (\text{A.162})$$

$$a_4 = \frac{\partial_{xyz} f(A'') + \partial_{xyz} f(A''')}{\Delta y^2} - 2 \frac{\partial_{xz} f(A'') - \partial_{xz} f(A''')}{\Delta y^3}, \quad (\text{A.163})$$

$$b_4 = 3 \frac{\partial_{xz} f(A''') - \partial_{xz} f(A'')}{\Delta y^2} - \frac{\partial_{xyz} f(A''') + 2\partial_{xyz} f(A'')}{\Delta y}. \quad (\text{A.164})$$

点 B、B' における物理量を補間することにより移流原点位置 (T) における物理量ならびに空間微係数を計算する。

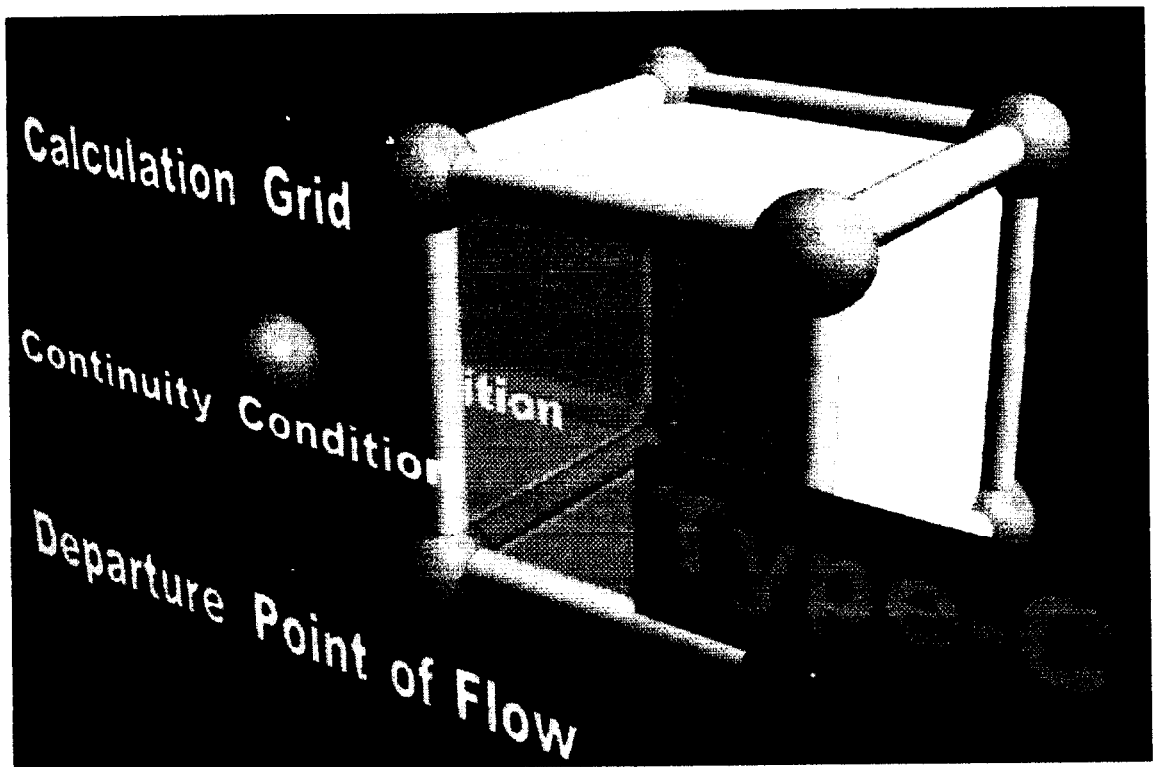


図 A.9: 3次元 CIP 法 Type-C 型補間

3次元 Type-C では、計算対象点 (赤点) とセルに含まれるグリッド (橙点) における物理量とその定義空間微係数の連続条件を用いて補間関数を形成する。

$$f(T) = a_1 \zeta^3 + b_1 \zeta^2 + \partial_z f(B) \zeta + f(B), \quad (\text{A.165})$$

$$\partial_x f(T) = a_2 \zeta^3 + b_2 \zeta^2 + \partial_{xz} f(B) \zeta + \partial_x f(B), \quad (\text{A.166})$$

$$\partial_y f(T) = a_3 \zeta^3 + b_3 \zeta^2 + \partial_{yz} f(B) \zeta + \partial_y f(B), \quad (\text{A.167})$$

$$\partial_z f(T) = 3a_1 \zeta^2 + 2b_1 \zeta + \partial_z f(B), \quad (\text{A.168})$$

$$\partial_{xy} f(T) = a_4 \zeta^3 + b_4 \zeta^2 + \partial_{xyz} f(B) \zeta + \partial_{xy} f(B), \quad (\text{A.169})$$

$$\partial_{xz} f(T) = 3a_2 \zeta^2 + 2b_2 \zeta + \partial_{xz} f(B), \quad (\text{A.170})$$

$$\partial_{yz} f(T) = 3a_3 \zeta^2 + 2b_3 \zeta + \partial_{yz} f(B), \quad (\text{A.171})$$

$$\partial_{xyz} f(T) = 3a_4 \zeta^2 + 2b_4 \zeta + \partial_{xyz} f(B), \quad (\text{A.172})$$

$$a_1 = \frac{\partial_z f(B) + \partial_z f(B')}{\Delta z^2} - 2 \frac{f(B) - f(B')}{\Delta z^3}, \quad (\text{A.173})$$

$$b_1 = 3 \frac{f(B') - f(B)}{\Delta z^2} - \frac{\partial_z f(B') + 2\partial_z f(B)}{\Delta z}, \quad (\text{A.174})$$

$$a_2 = \frac{\partial_{xz} f(B) + \partial_{xz} f(B')}{\Delta z^2} - 2 \frac{\partial_x f(B) - \partial_x f(B')}{\Delta z^3}, \quad (\text{A.175})$$

$$b_2 = 3 \frac{\partial_x f(B') - \partial_x f(B)}{\Delta z^2} - \frac{\partial_{xz} f(B') + 2\partial_{xz} f(B)}{\Delta z}, \quad (\text{A.176})$$

$$a_3 = \frac{\partial_{yz} f(B) + \partial_{yz} f(B')}{\Delta z^2} - 2 \frac{\partial_y f(B) - \partial_y f(B')}{\Delta z^3}, \quad (\text{A.177})$$

$$b_3 = 3 \frac{\partial_y f(B') - \partial_y f(B)}{\Delta z^2} - \frac{\partial_{yz} f(B') + 2\partial_{yz} f(B)}{\Delta z}, \quad (\text{A.178})$$

$$a_4 = \frac{\partial_{xyz} f(B) + \partial_{xyz} f(B')}{\Delta z^2} - 2 \frac{\partial_{xy} f(B) - \partial_{xy} f(B')}{\Delta z^3}, \quad (\text{A.179})$$

$$b_4 = 3 \frac{\partial_{xy} f(B') - \partial_{xy} f(B)}{\Delta z^2} - \frac{\partial_{xyz} f(B') + 2\partial_{xyz} f(B)}{\Delta z}. \quad (\text{A.180})$$

以上により得られた移流原点位置における物理量ならびに低次空間微係数をセミ・ラグランジュ的に移流することで移流相の計算を行う。

### 非移流相

3次元 CIP 法のための非移流相の計算について述べる。Type - A, C に関わらず、非移流相は 2 次精度中心差分法を用いて次のように計算される。

$$f_{i,j,k}^{n+1} = f_{i,j,k}^* + g \Delta t \quad (\text{A.181})$$

$$\begin{aligned} \partial_x f_{i,j,k}^{n+1} &= \partial_x f_{i,j,k}^* + \frac{f_{i+1,j,k}^{n+1} - f_{i-1,j,k}^{n+1}}{2\Delta x} - \frac{f_{i+1,j,k}^* - f_{i-1,j,k}^*}{2\Delta x} \\ &\quad - \frac{\partial u}{\partial x} \Big|_{i,j,k} \partial_x f_{i,j,k}^* \Delta t - \frac{\partial v}{\partial x} \Big|_{i,j,k} \partial_y f_{i,j,k}^* \Delta t - \frac{\partial w}{\partial x} \Big|_{i,j,k} \partial_z f_{i,j,k}^* \Delta t \end{aligned} \quad (\text{A.182})$$

$$\begin{aligned} \partial_y f_{i,j,k}^{n+1} &= \partial_y f_{i,j,k}^* + \frac{f_{i,j+1,k}^{n+1} - f_{i,j-1,k}^{n+1}}{2\Delta y} - \frac{f_{i,j+1,k}^* - f_{i,j-1,k}^*}{2\Delta y} \\ &\quad - \frac{\partial u}{\partial y} \Big|_{i,j,k} \partial_x f_{i,j,k}^* \Delta t - \frac{\partial v}{\partial y} \Big|_{i,j,k} \partial_y f_{i,j,k}^* \Delta t - \frac{\partial w}{\partial y} \Big|_{i,j,k} \partial_z f_{i,j,k}^* \Delta t \end{aligned} \quad (\text{A.183})$$

$$\begin{aligned} \partial_z f_{i,j,k}^{n+1} &= \partial_z f_{i,j,k}^* + \frac{f_{i,j,k+1}^{n+1} - f_{i,j,k-1}^{n+1}}{2\Delta z} - \frac{f_{i,j,k+1}^* - f_{i,j,k-1}^*}{2\Delta z} \\ &\quad - \frac{\partial u}{\partial z} \Big|_{i,j,k} \partial_x f_{i,j,k}^* \Delta t - \frac{\partial v}{\partial z} \Big|_{i,j,k} \partial_y f_{i,j,k}^* \Delta t - \frac{\partial w}{\partial z} \Big|_{i,j,k} \partial_z f_{i,j,k}^* \Delta t \end{aligned} \quad (\text{A.184})$$

$$\begin{aligned} \partial_{xy} f_{i,j,k}^{n+1} &= \partial_{xy} f_{i,j,k}^* + \frac{f_{i+1,j+1,k}^{n+1} - f_{i-1,j+1,k}^{n+1} - f_{i+1,j-1,k}^{n+1} + f_{i-1,j-1,k}^{n+1}}{4\Delta x \Delta y} \\ &\quad - \frac{f_{i+1,j+1,k}^* - f_{i-1,j+1,k}^* - f_{i+1,j-1,k}^* + f_{i-1,j-1,k}^*}{4\Delta x \Delta y} \\ &\quad - \frac{\partial^2 u}{\partial x \partial y} \Big|_{i,j,k} \partial_x f_{i,j,k}^* \Delta t - \frac{\partial^2 v}{\partial x \partial y} \Big|_{i,j,k} \partial_y f_{i,j,k}^* \Delta t - \frac{\partial^2 w}{\partial x \partial y} \Big|_{i,j,k} \partial_z f_{i,j,k}^* \Delta t \end{aligned} \quad (\text{A.185})$$

$$\partial_{xz} f_{i,j,k}^{n+1} = \partial_{xz} f_{i,j,k}^* + \frac{f_{i+1,j,k+1}^{n+1} - f_{i-1,j,k+1}^{n+1} - f_{i+1,j,k-1}^{n+1} + f_{i-1,j,k-1}^{n+1}}{4\Delta x \Delta z}$$

$$\begin{aligned} & \frac{f_{i+1,j,k+1}^* - f_{i-1,j,k+1}^* - f_{i+1,j,k-1}^* + f_{i-1,j,k-1}^*}{4\Delta x\Delta z} \\ & - \frac{\partial^2 u}{\partial x\partial z} \Big|_{i,j,k} \partial_x f_{i,j,k}^* \Delta t - \frac{\partial^2 v}{\partial x\partial z} \Big|_{i,j,k} \partial_y f_{i,j,k}^* \Delta t - \frac{\partial^2 w}{\partial x\partial z} \Big|_{i,j,k} \partial_z f_{i,j,k}^* \Delta t \end{aligned} \quad (\text{A.186})$$

$$\begin{aligned} \partial_{yz} f_{i,j,k}^{n+1} &= \partial_{yz} f_{i,j,k}^* + \frac{f_{i,j+1,k+1}^{n+1} - f_{i,j+1,k-1}^{n+1} - f_{i,j-1,k+1}^{n+1} + f_{i,j-1,k-1}^{n+1}}{4\Delta y\Delta z} \\ & - \frac{f_{i,j+1,k+1}^* - f_{i,j+1,k-1}^* - f_{i,j-1,k+1}^* + f_{i,j-1,k-1}^*}{4\Delta y\Delta z} \\ & - \frac{\partial^2 u}{\partial y\partial z} \Big|_{i,j,k} \partial_x f_{i,j,k}^* \Delta t - \frac{\partial^2 v}{\partial y\partial z} \Big|_{i,j,k} \partial_y f_{i,j,k}^* \Delta t - \frac{\partial^2 w}{\partial y\partial z} \Big|_{i,j,k} \partial_z f_{i,j,k}^* \Delta t \end{aligned} \quad (\text{A.187})$$

$$\begin{aligned} \partial_{xyz} f_{i,j,k}^{n+1} &= \partial_{xyz} f_{i,j,k}^* \\ & + (f_{i+1,j+1,k+1}^{n+1} - f_{i-1,j+1,k+1}^{n+1} - f_{i+1,j-1,k+1}^{n+1} + f_{i-1,j-1,k+1}^{n+1} \\ & - f_{i+1,j+1,k-1}^{n+1} + f_{i-1,j+1,k-1}^{n+1} + f_{i+1,j-1,k-1}^{n+1} - f_{i-1,j-1,k-1}^{n+1}) / (8\Delta x\Delta y\Delta z) \\ & - (f_{i+1,j+1,k+1}^* - f_{i-1,j+1,k+1}^* - f_{i+1,j-1,k+1}^* + f_{i-1,j-1,k+1}^* \\ & - f_{i+1,j+1,k-1}^* + f_{i-1,j+1,k-1}^* + f_{i+1,j-1,k-1}^* - f_{i-1,j-1,k-1}^*) / (8\Delta x\Delta y\Delta z) \\ & - \frac{\partial^3 u}{\partial x\partial y\partial z} \Big|_{i,j,k} \partial_x f_{i,j,k}^* \Delta t - \frac{\partial^3 v}{\partial x\partial y\partial z} \Big|_{i,j,k} \partial_y f_{i,j,k}^* \Delta t - \frac{\partial^3 w}{\partial x\partial y\partial z} \Big|_{i,j,k} \partial_z f_{i,j,k}^* \Delta t \end{aligned} \quad (\text{A.188})$$

### A.3 局所補間微分オペレータ法

局所補間微分オペレータ (Interpolated Differential Operator; IDO) 法は様々な偏微分方程式を高次精度で解くことを目的として提案されたスキームである。その基本概念は偏微分方程式に含まれる空間微係数を全て局所的に形成された高次のエルミート補間を微分することで求め、より解像度の高い数値計算を行うことにある。時間前進はテーラー展開を用いて行われるが、計算手法の特徴から補関関数の次数により時間精度は制限される。

#### A.3.1 補関関数

IDO 法では移流相、非移流相の区別無く時間前進を行う。ただし、それぞれに対する補関関数の形成方法に若干の違いがある。移流相に対しては CIP 法のようにグリッド点における移流速度から風上方向を検出し、風上方向に対して補関関数を形成する。これを風上補関 (Upwind Interpolation) と呼んでいる。拡散項、分散項のような非移流相に対してはグリッド点を中心にした補関関数を形成する。これを中心補関 (Center Interpolation) と呼んでいる。以下にこれらの補関関数について説明する。

##### 風上補関 (Upwind Interpolation)

IDO 法では風上補関を 3 次多項式、5 次多項式と、必要精度に応じて選択が可能である。3 次多項式を用いた場合の風上補関では、補関関数は下記に示す 3 次多項式を用いる。

$$F(x) = ax^3 + bx^2 + f_{x,i}x + f_i \quad (\text{A.189})$$

グリッド  $i$  における移流速度  $u_i$  が正であれば、風上方向  $i-1$  に向かって補関関数を形成するのでグリッド  $i-1$  における適合条件:  $F(-\Delta x) = f_{i-1}$ ,  $F_x(-\Delta x) = f_{x,i-1}$  を補関式に代入することで各係数  $a, b$  が次のように求まる。

$$a = -3 \frac{f_i - f_{i-1}}{\Delta x^2} + \frac{f_{x,i-1} + 2f_{x,i}}{\Delta x} \quad (\text{A.190})$$

$$b = -2 \frac{f_i - f_{i-1}}{\Delta x^3} + \frac{f_{x,i-1} + f_{x,i}}{\Delta x^2} \quad (\text{A.191})$$

IDO 法ではグリッド点における高次空間微係数を見積もる必要があるため、計算の対象となっているグリッド  $i$  における空間 2 次、空間 3 次の微係数を補間関数を微分することで以下のように見積もる。

$$\begin{aligned}\frac{\partial^2 f}{\partial x^2}\Big|_i &= F''(0) = 2a, \\ \frac{\partial^3 f}{\partial x^3}\Big|_i &= F'''(0) = 6b\end{aligned}\quad (\text{A.192})$$

風上補間として 5 次多項式を選択した場合、補間関数は次式で示される 5 次多項式を 4 グリッド点に渡り形成する。

$$F(x) = ax^5 + bx^4 + cx^3 + dx^2 + f_{x,i}x + f_i, \quad (\text{A.193})$$

$$F_x(x) = 5ax^4 + 4bx^3 + 3cx^2 + 2dx + f_{x,i} \quad (\text{A.194})$$

グリッド点  $i$  における移流速度  $u_i$  が負であれば、補間関数は領域  $x_{i-1}$  から  $x_{i+2}$  に形成され、適合条件  $F(-\Delta x) = f_{i-1}$ ,  $F(\Delta x) = f_{i+1}$ ,  $F_x(\Delta x) = f_{x,i+1}$ ,  $F(2\Delta x) = f_{i+2}$  を用いることで、各係数が以下のように求められる。

$$a = -\frac{1}{12\Delta x^5}(f_{i-1} + 9f_i - 9f_{i+1} - f_{i+2}) - \frac{1}{2\Delta x^4}(f_{x,i} + f_{x,i+1}), \quad (\text{A.195})$$

$$b = \frac{1}{12\Delta x^4}(4f_{i-1} + 21f_i - 24f_{i+1} - f_{i+2}) + \frac{1}{2\Delta x^3}(3f_{x,i} + 2f_{x,i+1}), \quad (\text{A.196})$$

$$c = -\frac{1}{12\Delta x^3}(5f_{i-1} - 9f_i + 3f_{i+1} + f_{i+2}) - \frac{1}{2\Delta x^2}(f_{x,i} - f_{x,i+1}), \quad (\text{A.197})$$

$$d = \frac{1}{12\Delta x^2}(2f_{i-1} - 33f_i + 30f_{i+1} + f_{i+2}) - \frac{1}{2\Delta x}(3f_{x,i} + 2f_{x,i+1}). \quad (\text{A.198})$$

また、グリッド点  $i$  における移流速度が正であれば、補間関数は計算領域  $x_{i-2}$  から  $x_{i+1}$  に形成される。適合条件  $F(\Delta x) = f_{i+1}$ ,  $F(-\Delta x) = f_{i-1}$ ,  $F_x(-\Delta x) = f_{x,i-1}$ ,  $F(-2\Delta x) = f_{i-2}$  を用いることで、各係数は以下のように求められる。

$$a = \frac{1}{12\Delta x^5}(f_{i+1} + 9f_i - 9f_{i-1} - f_{i-2}) - \frac{1}{2\Delta x^4}(f_{x,i} + f_{x,i-1}), \quad (\text{A.199})$$

$$b = \frac{1}{12\Delta x^4}(4f_{i+1} + 21f_i - 24f_{i-1} - f_{i-2}) - \frac{1}{2\Delta x^3}(3f_{x,i} + 2f_{x,i-1}), \quad (\text{A.200})$$

$$c = \frac{1}{12\Delta x^3}(5f_{i+1} - 9f_i + 3f_{i-1} + f_{i-2}) - \frac{1}{2\Delta x^2}(f_{x,i} - f_{x,i-1}), \quad (\text{A.201})$$

$$d = \frac{1}{12\Delta x^2}(2f_{i+1} - 33f_i + 30f_{i-1} + f_{i-2}) + \frac{1}{2\Delta x}(3f_{x,i} + 2f_{x,i-1}). \quad (\text{A.202})$$

風上 5 次エルミート補間の概念図を図 A.10 に示す。風上 5 次エルミート補間では風上の情報ばかりではなく、風下の情報も用いて補間関数を形成する。また、グリッド点における高次空間微係数は補間関数を微分することで下記のように求められる。

$$\begin{aligned}\frac{\partial^5 f}{\partial x^5}\Big|_i &= F''''''(0) = 120a, \quad \frac{\partial^4 f}{\partial x^4}\Big|_i = F''''(0) = 24b, \\ \frac{\partial^3 f}{\partial x^3}\Big|_i &= F'''(0) = 6c, \quad \frac{\partial^2 f}{\partial x^2}\Big|_i = F''(0) = 2d.\end{aligned}\quad (\text{A.203})$$

### 中心補間 (Center Interpolation)

拡散項や分散項を含む偏微分方程式の特性方程式の数は 1 を越えるので対象グリッド ( $x_i$ ) を中心に両側 ( $x_{i-1} \sim x_{i+1}$ ) に補間関数を張る中心補間を用いる必要がある (図 A.11)。補間関数は式 (A.193) 同様に



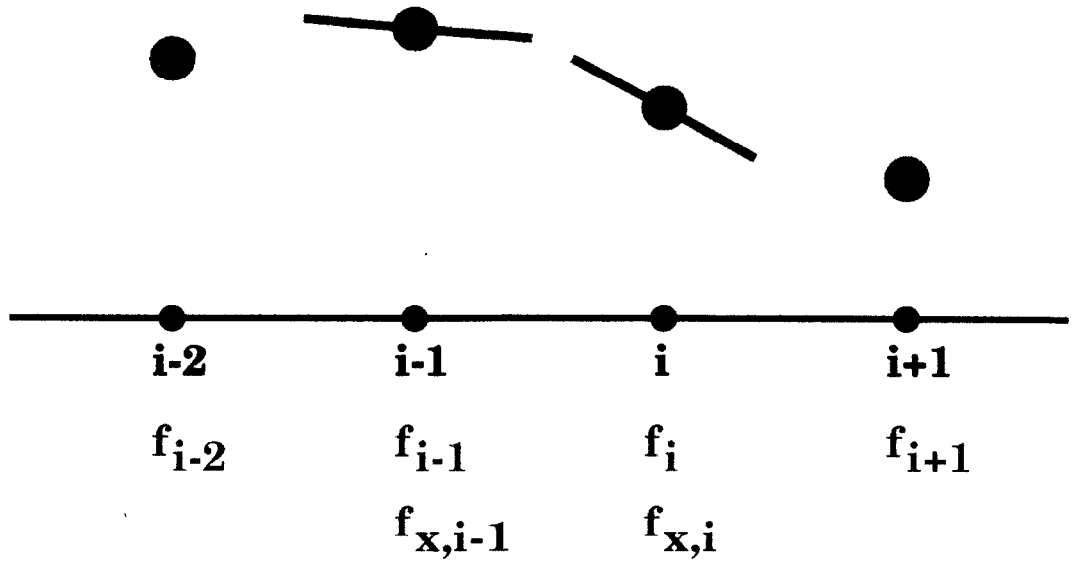


図 A.10: 5 次エルミート補間を用いた場合のグリッド間プロファイル  
グリッド  $i$  から、風上補間を負の方向に 5 次補間を形成する場合、グリッド  $i-2, i-1, i, i+1$  に於ける物理量とグリッド  $i-1, i$  に於ける空間 1 次微分値を用いてエルミート補間を形成する。

5 次多項式を用いる。適合条件  $F(-\Delta x) = f_{i-1}$ ,  $F_x(-\Delta x) = f_{x,i-1}$ ,  $F(\Delta x) = f_{i+1}$ ,  $F_x(\Delta x) = f_{x,i+1}$  より、各係数は以下のように求められる。

$$a = -\frac{3}{4\Delta x^5} (f_{i+1} - f_{i-1}) + \frac{1}{4\Delta x^4} (f_{x,i+1} + 4f_{x,i} + f_{x,i-1}), \quad (\text{A.204})$$

$$b = -\frac{1}{2\Delta x^4} (f_{i+1} - 2f_i + f_{i-1}) + \frac{1}{4\Delta x^3} (f_{x,i+1} - f_{x,i-1}), \quad (\text{A.205})$$

$$c = \frac{5}{4\Delta x^3} (f_{i+1} - f_{i-1}) - \frac{1}{4\Delta x^2} (f_{x,i+1} + 8f_{x,i} + f_{x,i-1}), \quad (\text{A.206})$$

$$d = \frac{1}{\Delta x^2} (f_{i+1} - 2f_i + f_{i-1}) - \frac{1}{4\Delta x} (f_{x,i+1} - f_{x,i-1}). \quad (\text{A.207})$$

また、グリッド点における高次空間微係数は補間関数を微分することで式 (A.203) と同型を得る。

### A.3.2 時間前進

IDO 法の時間前進は時間に関するテーラー展開を用いて行う。陽解法では、時間  $t = t^n$  において  $\Delta t$  進行するテーラー展開を用いる。また、IDO スキームは従属変数として物理量に加え、1 階空間微分値も各グリッド点毎に定義しているので、空間微係数のためのテーラー展開も必要となる。以上から、時間前進のために必要な式は以下の通りである。

$$f^{n+1} = f^n + f_t^n \Delta t + \frac{1}{2} f_{tt}^n \Delta t^2 + \frac{1}{6} f_{ttt}^n \Delta t^3 \dots, \quad (\text{A.208})$$

$$f_x^{n+1} = f_x^n + f_{tx}^n \Delta t + \frac{1}{2} f_{ttx}^n \Delta t^2 + \frac{1}{6} f_{tttx}^n \Delta t^3 \dots, \quad (\text{A.209})$$

ここで、上付きの  $n$  と  $n+1$  はそれぞれ時刻  $t, t + \Delta t$  における値を示している。また、上式中に含まれる時間微分項 ( $f_t, f_{tx} \dots$ ) は対象としている偏微分方程式を時間に関する微分オペレータとみなし空間微分項に変換して計算を行う。

例として時間、空間において一定な速度場における 1 次元移流方程式 ( $f_t + u f_x = 0$ ) について考える。1 次元移流方程式を時間、空間に関してそれぞれ微分して、式 (A.208)、式 (A.209) を全て空間微分項で

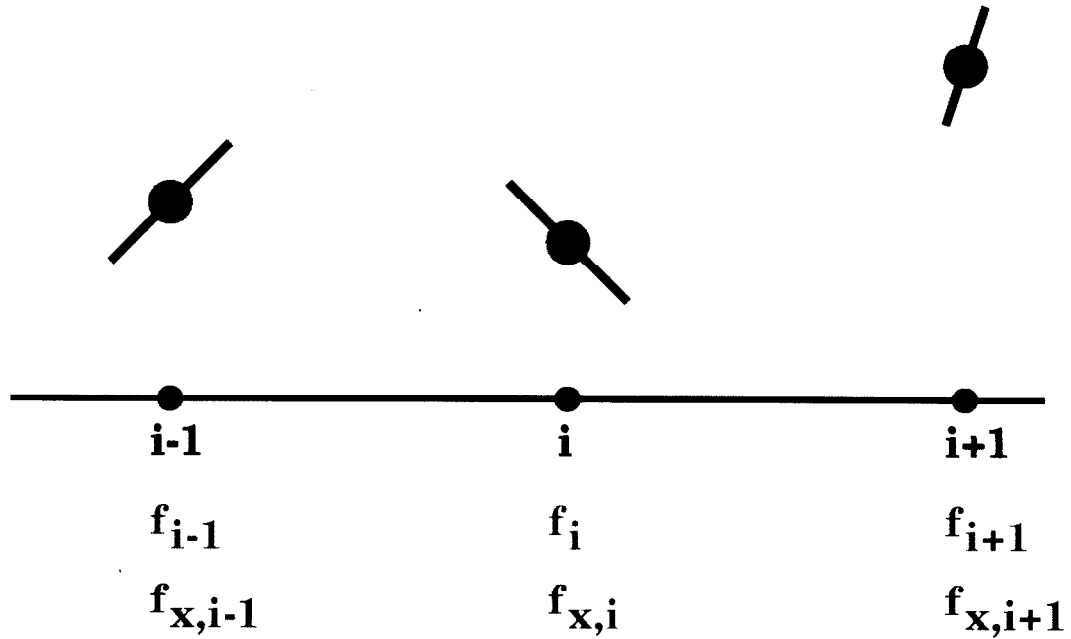


図 A.11: 中心補間に 5 次エルミート補間を用いた場合のグリッド間プロファイル

グリッド  $i$  で、中心補間を形成する場合、グリッド  $i-1, i, i+1$  に於ける物理量とその 1 階微分値を用いる。

表記する。次の関係;  $f_{tx} = -uf_{xx}, f_{txx} = -uf_{xxx}, \dots; f_{tt} = -uf_{tx}, f_{ttt} = -uf_{ttx}$  を利用することにより、式 (A.208)、式 (A.209) は、

$$f^{n+1} = f^n + f_x^n(-u\Delta t) + \frac{1}{2}f_{xx}^n(-u\Delta t)^2 + \frac{1}{6}f_{xxx}^n(-u\Delta t)^3, \quad (\text{A.210})$$

$$f_x^{n+1} = f_x^n + f_{xx}^n(-u\Delta t) + \frac{1}{2}f_{xxx}^n(-u\Delta t)^2, \quad (\text{A.211})$$

と表すことが出来る。ただし、空間分解性能の制限から物理量、1 階空間微分値をそれぞれテーラー展開の 3 次、2 次の項までとした。ここで、式 (A.192) を式 (A.210)、式 (A.211) に代入すると、CIP 法の離散式に完全に一致することが分かる。

### A.3.3 多次元 IDO 法

IDO 法を多次元化問題へ適用する場合、CIP 法のように多次元補間関数を利用するのではなく、1 次元補間関数を次元数に応じて各軸方向にオペレートすることで高次元の空間高次微係数を計算する手法を採用するため、その手続きは非常に単純になる。説明を簡略化するため、以下ではデカルト座標系で話を進める。基本概念は 1 次元 IDO 法と変わらず、補間関数を微分することでグリッドポイント毎に高次の空間微係数を求め、元の偏微分方程式をテーラー展開した時間前進に関する方程式を離散形で記述することにある。例えば 2 次元問題として以下の方程式を考える。

$$f_t = \mathfrak{D}_x f + \mathfrak{D}_y f \quad (\text{A.212})$$

ここで、 $\mathfrak{D}_x, \mathfrak{D}_y$  はそれぞれ  $x$  方向、 $y$  方向への 1 階微分演算子とした。1 次元 IDO でも行ったように、従属変数として定義した 1 階空間微係数の時間前進を行うため、空間微係数に関する支配方程式が必要となる。偏微分方程式を空間で微分することで不足分の方程式を生成すると以下ようになる。

$$f_{tx} = \mathfrak{D}_{xx} f + \mathfrak{D}_{xy} f \quad (\text{A.213})$$

$$f_{ty} = \mathfrak{D}_{xy} f + \mathfrak{D}_{yy} f \quad (\text{A.214})$$

$\mathfrak{S}_{xx}f, \mathfrak{S}_{yy}f$  は 1 次元補間関数を  $x$  方向ならびに  $y$  方向に関して形成することで容易に求めることが可能であるが、 $\mathfrak{S}_{xy}f$  は、補間関数を作ることが困難であるため、求めることが難しい。そこで、多次元 IDO 法では  $f_{xy}$  もまた従属変数として各グリッドポイント毎に定義する。従属変数が増加したことで方程式が不足するため次の偏微分方程式を更に導入する。

$$f_{txy} = \mathfrak{S}_{xxy}f + \mathfrak{S}_{xyy}f \quad (\text{A.215})$$

$f_{xx}$  が、 $f$  と  $f_x$  を用いて  $x$  方向に補間関数  $F(x)$  を形成し、補間関数を直接微分することにより求められるのと同様に、 $f_{xxy}$  は、 $f_y$  と  $f_{xy}$  を用いての  $x$  方向に補間関数  $F_y(x)$  を形成し、補間関数を  $x$  に関して微分することで求めることが出来るので方程式系は閉じ、解を得ることが出来る。つまり 1 次元のエルミート補間関数を物理量  $f$  に対して  $x, y$  方向に、空間微係数  $f_x$  に対して  $y$  方向に、そして空間微係数  $f_y$  に対して  $x$  方向に形成することで必要な微係数はほぼ得ることが出来る。5 次の補間関数を用いた場合の対応は以下の様になる。

$$\begin{aligned} f, f_x &\Rightarrow f_{xx}, f_{xxx}, f_{xxxx}, f_{xxxxx} \\ f, f_y &\Rightarrow f_{yy}, f_{yyy}, f_{yyyy}, f_{yyyyy} \\ f_y, f_{xy} &\Rightarrow f_{xxy}, f_{xxyy}, f_{xxxxy}, f_{xxxxyy} \\ f_x, f_{xy} &\Rightarrow f_{xyy}, f_{xyyy}, f_{xyyyy}, f_{xyyyyy} \end{aligned}$$

$f_{xxyy}$  のような高次空間微係数を求めることは不可能ではないが計算精度にほとんど影響しないので通常 0 とみなして計算を行う。必要な場合は同様の手法で  $f_{xx}, f_{xxy}$  を用いて補間を行って求めることが可能であり、また差分形式で用いて求めることも可能である。

次に 3 次元化について考える。2 次元問題と同様に簡単のためデカルト座標系で話を進める。3 次元化でも 2 次元化と同様な手法を使うことが出来る。例として以下のような方程式系を考える。

$$f_t = \mathfrak{S}_x f + \mathfrak{S}_y f + \mathfrak{S}_z f \quad (\text{A.216})$$

先と同様にこの偏微分方程式を解くためには従属変数  $f, f_x, f_y, f_z, f_{xy}, f_{yz}, f_{xz}$  の数だけ偏微分方程式を空間微分した式を用いる必要があり、以下の方程式系が要求される。

$$f_t = \mathfrak{S}_x f + \mathfrak{S}_y f + \mathfrak{S}_z f \quad (\text{A.217})$$

$$f_{tx} = \mathfrak{S}_{xx}f + \mathfrak{S}_{xy}f + \mathfrak{S}_{xz}f \quad (\text{A.218})$$

$$f_{ty} = \mathfrak{S}_{xy}f + \mathfrak{S}_{yy}f + \mathfrak{S}_{yz}f \quad (\text{A.219})$$

$$f_{tz} = \mathfrak{S}_{xz}f + \mathfrak{S}_{yz}f + \mathfrak{S}_{zz}f \quad (\text{A.220})$$

$$f_{txy} = \mathfrak{S}_{xxy}f + \mathfrak{S}_{xyy}f + \mathfrak{S}_{xyz}f \quad (\text{A.221})$$

$$f_{tyz} = \mathfrak{S}_{xyz}f + \mathfrak{S}_{yyz}f + \mathfrak{S}_{yzz}f \quad (\text{A.222})$$

$$f_{txz} = \mathfrak{S}_{xxz}f + \mathfrak{S}_{xyz}f + \mathfrak{S}_{zzz}f \quad (\text{A.223})$$

2 次元 IDO 法の場合と同様に  $f_{xyz}$  を従属変数  $f, f_x, f_y, f_z, f_{xy}, f_{yz}, f_{xz}$  から補間により求めることは困難であるため、新たに従属変数として導入することを考える。また、2 次元 IDO 法の場合と同様に  $f_{xyz}$  を導入することで以下の微分方程式を導入する必要性が発生する。

$$f_{txyz} = \mathfrak{S}_{xxyz}f + \mathfrak{S}_{xyyz}f + \mathfrak{S}_{xyzz}f \quad (\text{A.224})$$

先と同様に補間式から求める微係数と補間式に必要な独立変数の対応をとると以下のようになる。

$$f, f_x \Rightarrow f_{xx}, f_{xxx}, f_{xxxx}, f_{xxxxx}$$

$$f, f_y \Rightarrow f_{yy}, f_{yyy}, f_{yyyy}, f_{yyyyy}$$

$$\begin{aligned}
f, f_z &\Rightarrow f_{zz}, f_{zzz}, f_{zzzz}, f_{zzzzz} \\
f_y, f_{xy} &\Rightarrow f_{xy}, f_{xxy}, f_{xxyy}, f_{xxyyy} \\
f_x, f_{xy} &\Rightarrow f_{xy}, f_{xyy}, f_{xyyy}, f_{xyyyy} \\
f_z, f_{xz} &\Rightarrow f_{xz}, f_{xzz}, f_{xzzz}, f_{xzzzz} \\
f_x, f_{xz} &\Rightarrow f_{xz}, f_{xzz}, f_{xzzz}, f_{xzzzz} \\
f_y, f_{yz} &\Rightarrow f_{yz}, f_{yzz}, f_{yzzz}, f_{yzzzz} \\
f_z, f_{yz} &\Rightarrow f_{yz}, f_{yyz}, f_{yyyz}, f_{yyyyz} \\
f_{yz}, f_{xyz} &\Rightarrow f_{xyz}, f_{xxyz}, f_{xxyyz}, f_{xxxyyz} \\
f_{xz}, f_{xyz} &\Rightarrow f_{xyz}, f_{xyyz}, f_{xyyyz}, f_{xyyyyz} \\
f_{xy}, f_{xyz} &\Rightarrow f_{xyz}, f_{xyzz}, f_{xyzzz}, f_{xyzzzz}
\end{aligned}$$

偏微分方程式を解くのに必要な微係数は以上からほぼ求めることができるのでこの方程式系は閉じる。3次元の場合も2次元の場合と同様に  $f_{xxyyz}$  等の微係数を求めるのは不可能でないにしろ繁雑で、計算量の増加が伴う。しかしこれらの係数も他の低次の微係数と比べると方程式系への寄与が小さいので通常は0とみなす。

## 付録 B

# 計算手法

### B.1 4 次精度 Runge-Kutta 法

セミ・ラグランジュ CIP 法における移流経路の計算に 4 次精度 Runge - Kutta 法を用いた。その際に利用した微分方程式は下記の通りであり、式 (B.1) を積分することで移流経路を求めた。

$$\frac{\partial \mathbf{x}}{\partial t} = U(\mathbf{x}, t) \quad (\text{B.1})$$

ここで、 $U$  は速度場を表す。ただし、離散化されているので  $U$  は各グリッド点上の値か、もしくはグリッド毎に定義されている速度  $u$  と、その空間一次微分値  $u_x$  を利用して形成された 3 次多項式によって補間された値である。Runge - Kutta 法により式 (B.1) は下記のようにして積分される。

$$\mathbf{x}_1 = \Delta t \times U(t, \mathbf{x}), \quad (\text{B.2})$$

$$\mathbf{x}_2 = \Delta t \times U(t - \Delta t/2, \mathbf{x} - \mathbf{x}_1/2), \quad (\text{B.3})$$

$$\mathbf{x}_3 = \Delta t \times U(t - \Delta t/2, \mathbf{x} - \mathbf{x}_2/2), \quad (\text{B.4})$$

$$\mathbf{x}_4 = \Delta t \times U(t - \Delta t, \mathbf{x} - \mathbf{x}_3), \quad (\text{B.5})$$

$$\mathbf{x}(t - \Delta t) = \mathbf{x}(t) - \frac{\mathbf{x}_1 + 2\mathbf{x}_2 + 2\mathbf{x}_3 + \mathbf{x}_4}{6} \quad (\text{B.6})$$

式 (4.60) で示したように、各時間ステップ毎に  $\Delta t = \sum_i \lambda_i$  と、仮想時間ステップに区切って計算を行う。つまり、上記 Runge - Kutta 法による積分は、仮想時間  $\lambda_i$  に対して行われる。速度場の時間変化が十分に小さい ( $u_t \simeq 0$ ) と仮定すると、Runge - Kutta 法での計算に必要な時刻  $t - \Delta t/2, t - \Delta t$  における速度プロファイルを全て時刻  $t$  における速度プロファイルに置き換えることが出来るので、計算は非常に簡単になる。

### B.2 シンプソン則

セミ・ラグランジュ CIP 法は偏微分方程式 (Partial Differential Equation; PDE) の非移流相の計算をシンプソン則によって行う。シンプソン則は次のように書き表される。

$$I(f) = \int_a^b f(t) dt \simeq \frac{b-a}{6} \left[ f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right]. \quad (\text{B.7})$$

例としてセミ・ラグランジュ CIP 法を用いて一次元移流方程式の空間微分に関する時間前進に含まれる、非移流相の数値積分について考える ( $(f_x)_t = -u_x f_x$ )。セミ・ラグランジュ CIP 法での時間前進は仮想時間ステップ  $\lambda$  を用いて行う  $\sum_i \lambda_i = \Delta t$ 。つまり、非移流相 ( $-u_x f_x$ ) はシンプソン則を用いて次のよ

うに積分される。

$$\begin{aligned} \int_t^{t+\Delta t} (-u_x f_x) dt &= \sum_i \int_{\lambda_i}^{\lambda_{i+1}} (-u_x f_x) dt \\ &= \sum_i -\frac{\Delta\lambda}{6} \left[ u_x(\lambda_i) f_x(\lambda_i) + 4u_x\left(\frac{\lambda_i + \lambda_{i+1}}{2}\right) f_x\left(\frac{\lambda_i + \lambda_{i+1}}{2}\right) \right. \\ &\quad \left. + u_x(\lambda_{i+1}) f_x(\lambda_{i+1}) \right]. \end{aligned} \quad (\text{B.8})$$

各仮想時刻  $\lambda$  に於ける位置は 4 次精度 Runge-Kutta 法を用いた積分  $\mathbf{x}_{tmp} = \int_t^{t+\Delta t} \mathbf{u}(\mathbf{x}, t) dt$  から得ることが出来る。仮想時刻に於ける位置が分かれば  $u_x(\mathbf{x}_{tmp}, \lambda)$ 、 $f_x(\mathbf{x}_{tmp}, \lambda)$  は多次元 CIP 法を用いることで求めることが出来る。

また、微係数を用いるシンプソン則として、以下の関係式も利用した。

$$I(f) = \frac{b-a}{2} [f(a) + f(b)] + \frac{(b-a)^2}{12} [f'(a) - f'(b)] \quad (\text{B.9})$$

ただし、非移流項に対して時間に関する積分を行うため、各微係数は時間に関するものとなる。ここで、対象となる PDE をシンプソン則の微係数として与えることで時間微係数を空間微係数に置換し、数値計算を行った。

## 付録 C

# 人工粘性

### C.1 1次元の人工粘性

衝撃波前面、背面の密度、粒子速度、圧力、内部エネルギーをそれぞれ図 C.1 に示すように  $(\rho_0, U_0, P_0, E_0)$ ,  $(\rho_1, U_1, P_1, E_1)$  と定め、衝撃波伝搬速度を  $S$  とおく。

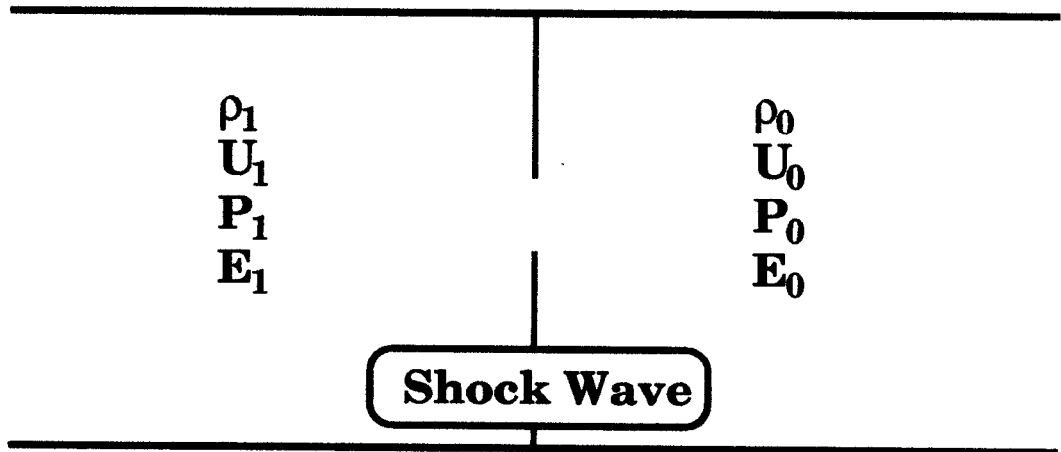


図 C.1: Rankine-Hugoniot の関係

衝撃波前後の物理量を用いて Rankine-Hugoniot の関係式を記述すると、

$$S = \frac{(U_1 - U_0) \rho_1}{\rho_1 - \rho_0}, \quad (\text{C.1})$$

$$P_1 - P_0 = \rho_0 S (U_1 - U_0), \quad (\text{C.2})$$

$$E_1 - E_0 = \frac{P_1 + P_0}{2} \left( \frac{1}{\rho_0} - \frac{1}{\rho_1} \right), \quad (\text{C.3})$$

$$P = P(E, \rho) \quad (\text{C.4})$$

となる。理想気体の状態方程式

$$P = (\gamma - 1) \rho E \quad (\text{C.5})$$

を式 (C.3) に作用することで、

$$\frac{1}{\gamma - 1} \left( \frac{P_1}{\rho_1} - \frac{P_0}{\rho_0} \right) = \frac{P_1 + P_0}{2} \left( \frac{1}{\rho_0} - \frac{1}{\rho_1} \right) \quad (\text{C.6})$$

が得られる。上式を密度に関して整理することで

$$\frac{\rho_0}{\rho_1} = \frac{(\gamma-1)P_1 + (\gamma+1)P_0}{(\gamma+1)P_1 + (\gamma-1)P_0} \quad (\text{C.7})$$

を得る。また、式 (C.1)、C.2から、

$$\frac{1}{\rho_0} - \frac{1}{\rho_1} = \frac{\Delta U^2}{P_1 - P_0} \quad (\text{C.8})$$

の関係式が得られる。ここで、 $\Delta U = U_1 - U_0$  とおいた。式 (C.8)、C.5を式 (C.3) に代入し、式 (C.7) を用いてまとめると

$$\frac{1}{\gamma-1} \left( \frac{P_1}{\rho_1} - \frac{P_0}{\rho_0} \right) = \frac{P_1 + P_0}{2} \frac{\Delta U^2}{P_1 - P_0} \quad (\text{C.9})$$

$$\frac{\rho_0}{\rho_1} P_1 - P_0 = \frac{(\gamma-1)\Delta U^2 \rho_0}{2} \frac{P_1 + P_0}{P_1 - P_0} \quad (\text{C.10})$$

↓

$$(P_1 + P_0)(P_1 - P_0)^2 = \frac{\Delta U^2 \rho_0}{2} (P_1 + P_0) \{(\gamma+1)P_1 + (\gamma-1)P_0\} \quad (\text{C.11})$$

と変形できる。 $P_1 > P_0 > 0$  より、 $P_0 + P_1 \neq 0$  であるから、

$$P_1^2 - \left( 2P_0 + \frac{(\gamma+1)\Delta U^2 \rho_0}{2} \right) P_1 + P_0^2 - \frac{(\gamma-1)\Delta U^2 \rho_0}{2} P_0 = 0 \quad (\text{C.12})$$

と展開される。2次方程式の解の公式より、

$$\begin{aligned} P_1 &= \frac{2P_0 + \frac{(\gamma+1)\Delta U^2 \rho_0}{2} \pm \sqrt{\left( 2P_0 + \frac{(\gamma+1)\Delta U^2 \rho_0}{2} \right)^2 - 4 \left( P_0^2 - \frac{(\gamma-1)\Delta U^2 \rho_0}{2} P_0 \right)}}{2} \\ &= P_0 + \frac{(\gamma+1)\Delta U^2 \rho_0}{4} \pm \sqrt{\gamma \Delta U^2 \rho_0 P_0 + \left( \frac{(\gamma+1)\Delta U^2 \rho_0}{4} \right)^2} \end{aligned} \quad (\text{C.13})$$

が得られる。ここで、 $P_1 > P_0$ 、 $\rho_0 > 0$  であることから、

$$P_1 = P_0 + \frac{(\gamma+1)\rho_0 \Delta U^2}{4} + |\Delta U| \rho_0 \sqrt{\left( \frac{\gamma+1}{4} \right)^2 (\Delta U)^2 + \frac{\gamma P_0}{\rho_0}} \quad (\text{C.14})$$

によって、衝撃波前後の圧力  $P_0, P_1$  は関係付けられる。局所的な音速

$$a_0 = \sqrt{\frac{\gamma P_0}{\rho_0}} \quad (\text{C.15})$$

を用いることで式 (C.14) は、

$$P_1 = P_0 + \frac{(\gamma+1)\rho_0 \Delta U^2}{4} + |\Delta U| \rho_0 \sqrt{\left( \frac{\gamma+1}{4} \right)^2 (\Delta U)^2 + a_0^2} \quad (\text{C.16})$$

と書き直すことが出来る。 $\Delta U \gg a_0$  の場合、人工粘性  $q$  は

$$q = \frac{(\gamma+1)}{2} \rho_0 \Delta U^2 \quad (\text{C.17})$$

の形で与えられ、その逆であれば

$$q = \rho_0 |\Delta U| a_0 \quad (\text{C.18})$$



で与えられることが分かる。前者は von Neumann 型の人工粘性

$$q_1 = c_0^2 \rho (\Delta x)^2 \left( \frac{\partial \dot{x}}{\partial x} \right)^2 \quad (\text{C.19})$$

に対応し、後者は Landshoff 型の人工粘性

$$q_2 = c_L \rho \Delta x a \left| \frac{\partial \dot{x}}{\partial x} \right| \quad (\text{C.20})$$

に対応することがわかる。

## C.2 多次元の人工粘性

多次元問題に対して人工粘性を提供するためには2つの問題がある。1つは速度の空間勾配で表される歪み率の算出方法であり、1つはグリッドの代表長さである。この問題を解決する手法として速度のダイバージェンスを計算するかわりに保存方程式を扱い、体積変化率を用いた手法が提案されている。しかし、流れが収束する場においては衝撃波が存在しない場合でもこの条件に当てはまる。そこで、ある面が他の面を追い越すことにより生じる体積変化率を知る必要がある。この問題は流体が加速される方向に対して歪み率を計算することで解決でき、式 (C.21) を用いて表現される。

$$\frac{ds}{dt} = \frac{\partial \dot{x}}{\partial x} \cos^2 \alpha + \frac{\partial \dot{y}}{\partial y} \sin^2 \alpha + \left( \frac{\partial \dot{x}}{\partial y} + \frac{\partial \dot{y}}{\partial x} \right) \cos \alpha \sin \alpha \quad (\text{C.21})$$

ここで、 $\alpha$  は 図 C.2 に示されるように加速度の方向を示している。

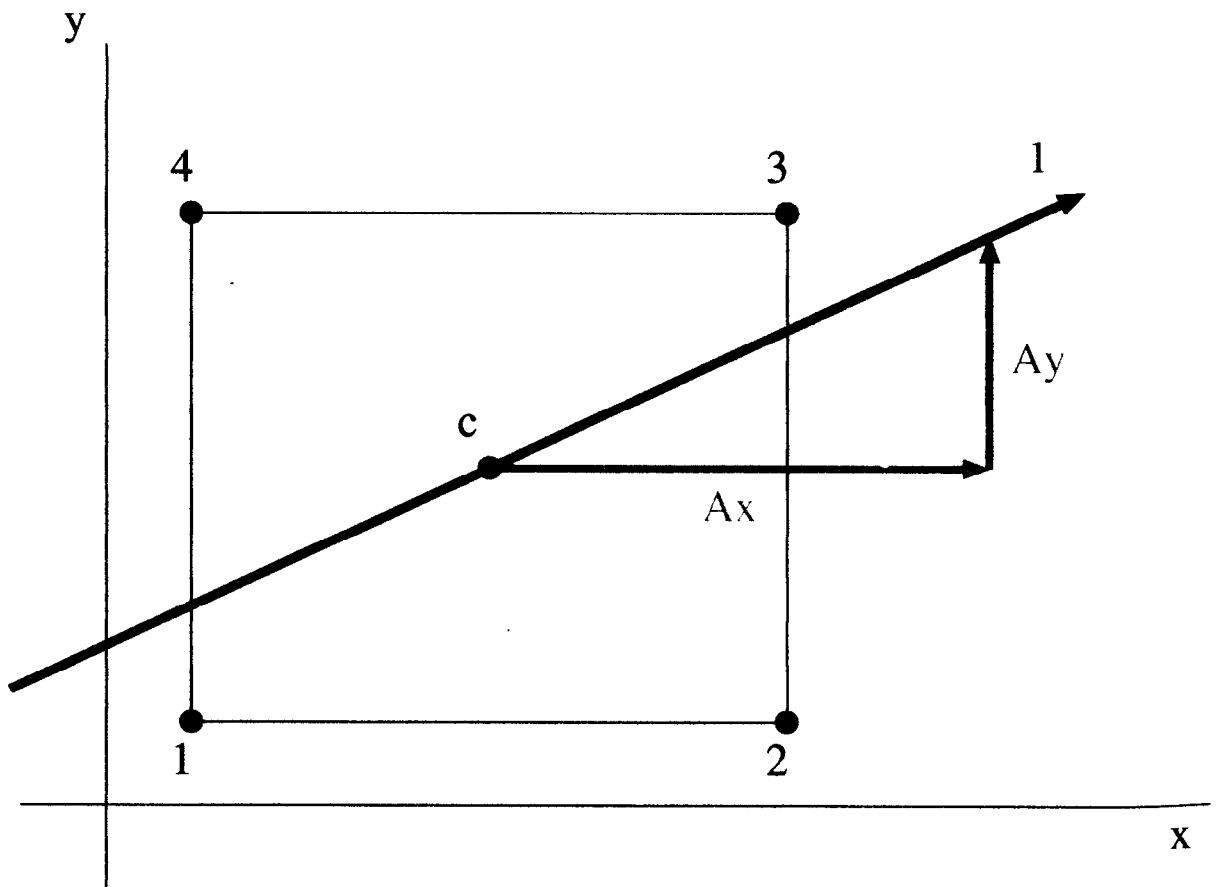


図 C.2: 2次元人工粘性のための加速度の検出

また、 $x, y$  方向の加速度  $A_x, A_y$  を用いることで、 $\alpha$  は次のように表される。

$$\cos \alpha = \frac{A_x}{\sqrt{A_x^2 + A_y^2}} \quad (\text{C.22})$$

$$\sin \alpha = \frac{A_y}{\sqrt{A_x^2 + A_y^2}} \quad (\text{C.23})$$

3次元問題のための人工粘性計算では、 $x, y, z$  方向に対する加速度  $A_x, A_y, A_z$  により記述される下記に示す歪み率を用いる。

$$\frac{ds}{dt} = \left[ \frac{\partial \dot{x}}{\partial x} A_x^2 + \frac{\partial \dot{y}}{\partial y} A_y^2 + \frac{\partial \dot{z}}{\partial z} A_z^2 + \left( \frac{\partial \dot{x}}{\partial y} + \frac{\partial \dot{y}}{\partial x} \right) A_x A_y + \left( \frac{\partial \dot{y}}{\partial z} + \frac{\partial \dot{z}}{\partial y} \right) A_y A_z \right] \cdot \left[ \frac{1}{A_x^2 + A_y^2 + A_z^2} \right] \quad (\text{C.24})$$

次に、グリッドの代表長さについて考える。

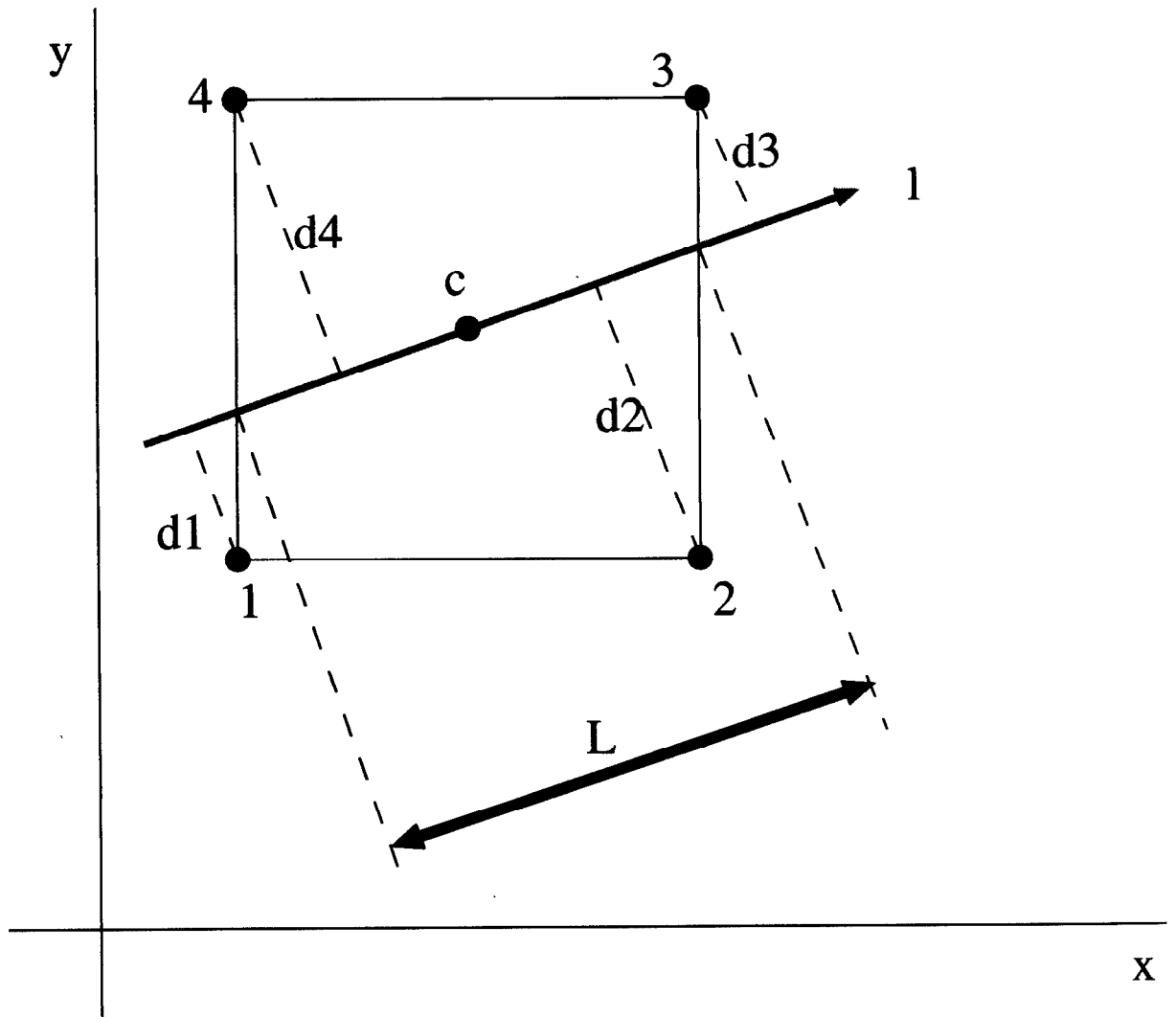


図 C.3: 2次元人工粘性のための代表グリッド長さ

グリッドの代表長さを簡単に決定することを目的として、流体の加速度方向に対するグリッド長さを採用する。図 C.3 に示すようにセルに含まれるグリッド点からの距離  $d_1, d_2, d_3, d_4$  と、セルの面積  $A$  を用

いることで、その長さ  $L$  を

$$L = \frac{2A}{d_1 + d_2 + d_3 + d_4} \quad (\text{C.25})$$

で表記することが出来る。

以上の多次元歪み率、グリッド長さをを用いることで一般化された人工粘性  $q$  は

$$q = c_0 \rho L^2 \left( \frac{ds}{dt} \right)^2 + c_L \rho L a \left| \frac{ds}{dt} \right| \quad (\text{C.26})$$

により表される。ここで、

$$q = 0 \quad \text{for} \quad \frac{ds}{dt} \geq 0 \quad (\text{C.27})$$

$$a = \sqrt{\frac{P}{\rho}} \quad (\text{C.28})$$

$$c_0 \simeq 2, \quad c_L \simeq 1 \quad (\text{C.29})$$

である。

## 付録 D

# マルチグリッド法

### D.1 CS 法 (Correction Storage Scheme)

多重格子法では図 D.1 に示すように格子間隔の異なる複数の階層的な格子  $G^k$  を用いる。上付きの  $k$  は格子の細かさを表し、 $k$  が小さい程格子は粗くなる。 $G^k$  の格子間隔を  $h^k$  とすると、 $h^{k-1} = 2h^k$  である。通常マルチグリッド法は多数のグリッドを用いるが、ここでは簡単のため格子間隔の細かい  $G^k$  とその2倍の格子間隔を持つ  $G^{k-1}$  の2格子を用いる手法を考える。

基礎方程式を  $G^k$  上で離散化し、下記の式を得る。

$$L^k W^k = F^k \quad (D.1)$$

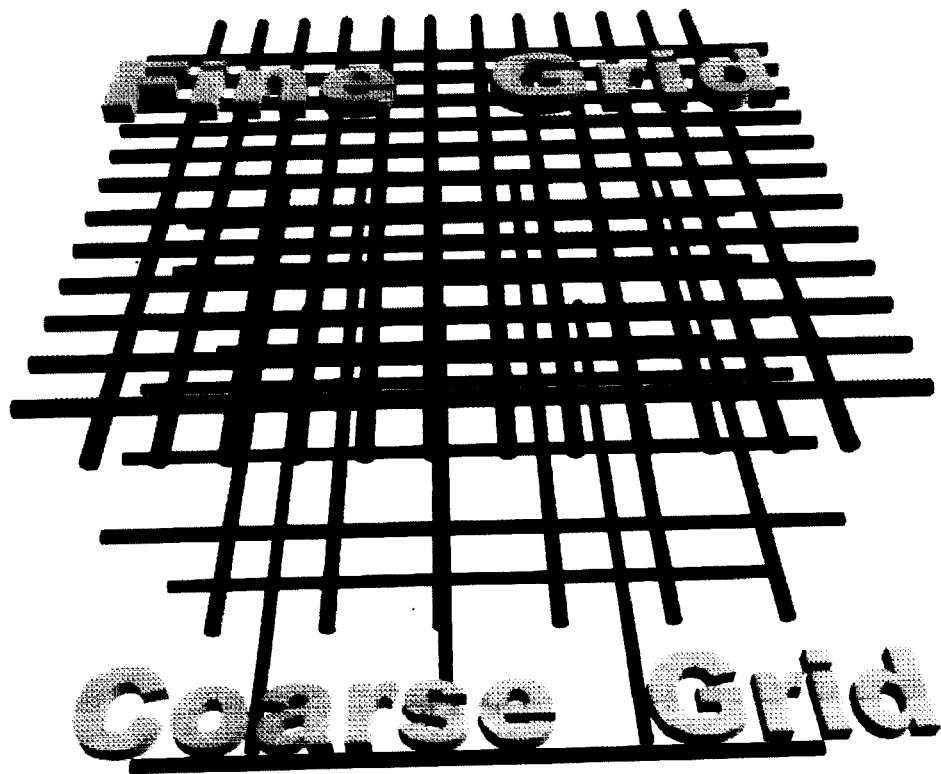


図 D.1: マルチグリッド法で用いられるグリッド構造:  
格子間隔の異なるグリッドを複数用意し、それぞれの格子上で適切に誤差を収束させることで高速な緩和計算を行う。

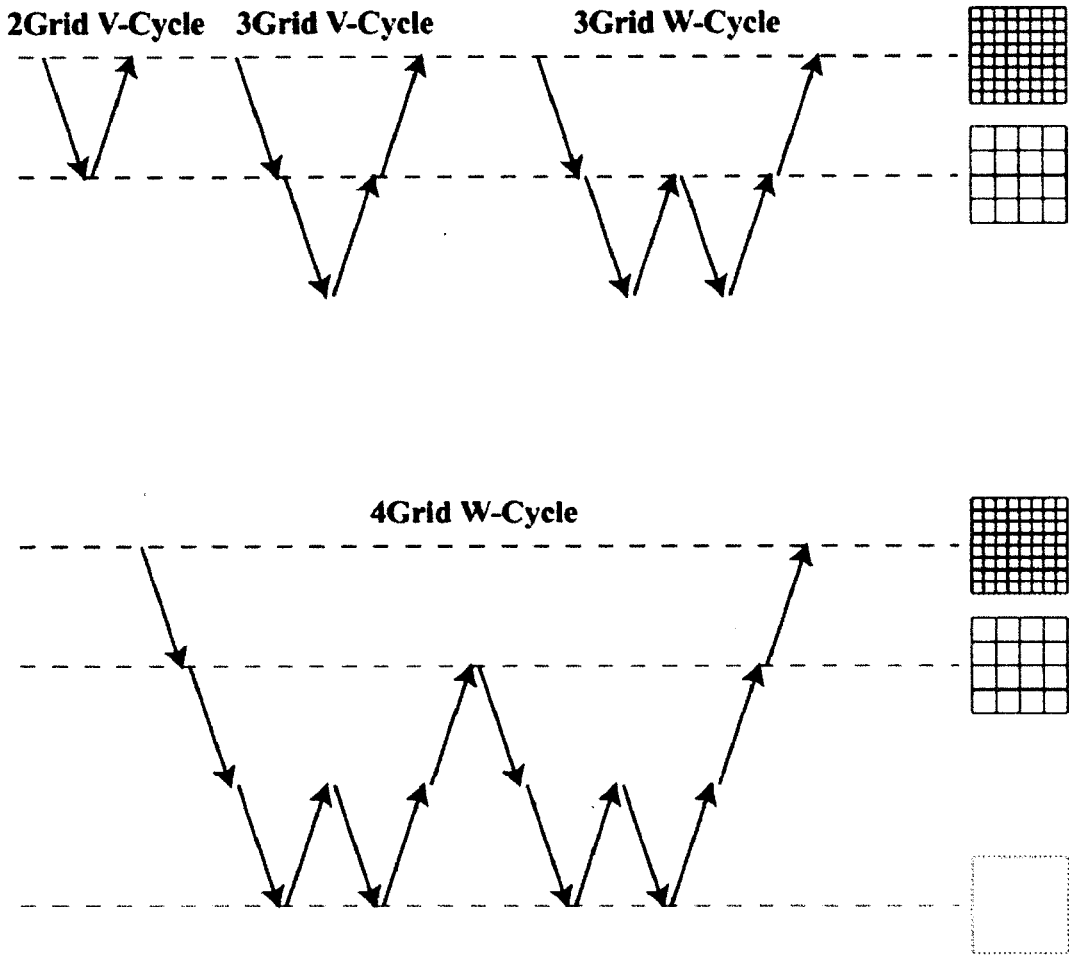


図 D.2: マルチグリッド法で用いるグリッド数とサイクルの模式図  
 マルチグリッド法では格子数が増加するに従い、様々な緩和経路を選択できる

ここで、 $L^k$ 、 $W^k$ 、 $F^k$ 、はそれぞれ  $G^k$  上での演算子、厳密解、定数項を示す。適当な初期値  $w_0^k$  を仮定し、緩和法で  $\nu_1$  回反復計算して近似解  $w_1^k$  を得る。この過程を以下のように記述する。

$$w_1^k = \text{Relax}(L^k, F^k, w_0^k, \nu_1) \tag{D.2}$$

この段階での誤差の残差を次のように定義する。

$$R^k = F^k - L^k w_1^k \tag{D.3}$$

近似解の修正量を  $v^k = W^k - w_1^k$  の様に定義すると、細かい格子上での修正方程式は以下のように求められる。

$$L^k v^k = R^k \tag{D.4}$$

$G^k$  上で減衰しにくい長波長の誤差を含む上式を長波長の誤差を減衰しやすいグリッド  $G^{k-1}$  に制限補間 ( Restriction ) する。

$$R^{k-1} = I_k^{k-1} R^k \tag{D.5}$$

$$L^{k-1} v^{k-1} = R^{k-1} \tag{D.6}$$

ここで、 $I_k^{k-1}$  は  $G^k$  から  $G^{k-1}$  への変換を示す演算子である。 $G^{k-1}$  上で修正方程式を解くことで修正値  $v^{k-1}$  を得る。

$G^k$  上の修正量  $v^k$  は、 $v^{k-1}$  を延長補間 (Prolongation) することで求められる。

$$v^k = I_{k-1}^k v^{k-1} \quad (\text{D.7})$$

ここで、 $I_{k-1}^k$  は、 $I_k^{k-1}$  の逆向きの変換を行う。次に  $w_1^k$  に修正量  $v^k$  を加えて修正を行い、さらに  $\nu_2$  回の緩和計算を行うことで、最終的な解  $w_3^k$  を得る。

$$w_2^k = w_1^k + v^k w_3^k = \text{Relax}(L^k, F^k, w_2^k, \nu_2) \quad (\text{D.8})$$

以上を 2 グリッドサイクルと呼び、このサイクルを何度か繰り返すことで目的の方程式を高速に解くことができる。

簡単のため、2 グリッドでの説明を行ったが、さらに粗い格子を用いて長波長誤差を高速に緩和することができる。単純に粗い格子と細かい格子を連続して利用するサイクルを V サイクルと呼ぶ。また、途中の粗い格子で再び粗い格子に戻って長波長誤差を減衰させる手法もあり、このサイクルを W サイクルと呼んでいる。こういったサイクルのとり方は様々であるが、このように何度も修正方程式を構成して行く手法をマルチグリッド法と呼んでいる (図 D.2)。

## 付録 E

# PC クラスターの製作ならびに MPI を用いた並列プログラミング

近年、パーソナルコンピュータ (Personal Computer; PC) の低価格化、高性能化に伴い、比較的安価でギガフロップス程度のベンチマーク値を弾き出す PC クラスターを組み上げることが可能になってきた。PC クラスターのメリットとして、まずその利便性が挙げられる。大規模計算機では仕様変更を行うのにさまざまな問題が付きまとい、更にハードウェアの拡張に至っては専用品を使用する必要があることからそのメンテナンスに要する費用が非常に高価になる。それに比べ PC クラスターで用いられるハードウェアはその市場効果による大量生産化に伴い年々低価格化が進行しており、ハードウェア仕様変更に関しては PC の普及とともに、より簡易に行えるようになると言える。また、PC クラスターは分散メモリ型システムであり、通信トラフィックが無視出来ると考えた場合 PC を増加させるに従い、単純に並列計算機としての処理能力を向上させることができるメリットもある。

これまでは各 PC にオペレーションシステム (Operation System; OS) をインストールしなければならなかったため、並列計算に利用するコンピューターの台数分だけシステム構築、またメンテナンスの面で手間が生じていたが、近年ネットワークブート機能を持ち合わせた BIOS が登場しており、これを利用することでシステム管理の面でもその利便性は向上するものと思われる。

スーパーコンピュータでもシステムの複雑化に伴い一般的に OS やコンパイラ等の仕様変更で様々なコストが必要とされる。PC クラスターでも同様に接続台数の増加に伴い OS やコンパイラなどのライセンスコストも無視できなくなる。幸いなことに、PC 用無償 UNIX OS として近年 Linux が注目されており、最新ハードウェアへの対応も世界中の多くの有能な開発者たちの手によっていち早く行われるようになって来た。コンパイラなどのアプリケーションも GNU Project (<http://www.gnu.org/>) から無償で提供されており、ソフトウェア面でのコストもほとんど必要ないと言える状況にある。

組み上げた PC クラスター上で並列プログラムを作成するためにはネットワーク間通信を考慮しなければならない。UNIX OS に標準ライブラリとして付属している pipe を用いた通信によりそのような機能を実装可能であるが、その理解と実装までには相当の時間が必要になるのが一般的である。そこで、MPI (Message Passing Interface) ライブラリを用いた簡単な並列プログラミングを使用することを考える。

MPI ライブラリは現在 MPI2 が開発段階にあり有償で配布されているが、MPI1 に関しては無償で提供されている。本章では MPI1 を用いた開発について述べる。

## E.1 PC クラスターシステムの構築

### E.1.1 ハードウェア構成

本論文中の並列計算は東京工業大学情報処理センター所有の Origin 2000 システムを用いて行ったが、並列プログラムの開発環境としてパーソナルコンピュータを用いた PC クラスターシステムを利用した。

その概念図を図 E.1 に示す。8 台の計算用コンピューターを 100M ベースのスイッチングハブで接続することにより通信経路を確保し、並列計算を行う。それらのコントロール用サーバーとして他に 1 台の PC を利用した。また、データ保存のためネットワークファイルシステム (Network File System; NFS) を利用している。各ハードウェアの構成内容は次のようになる。

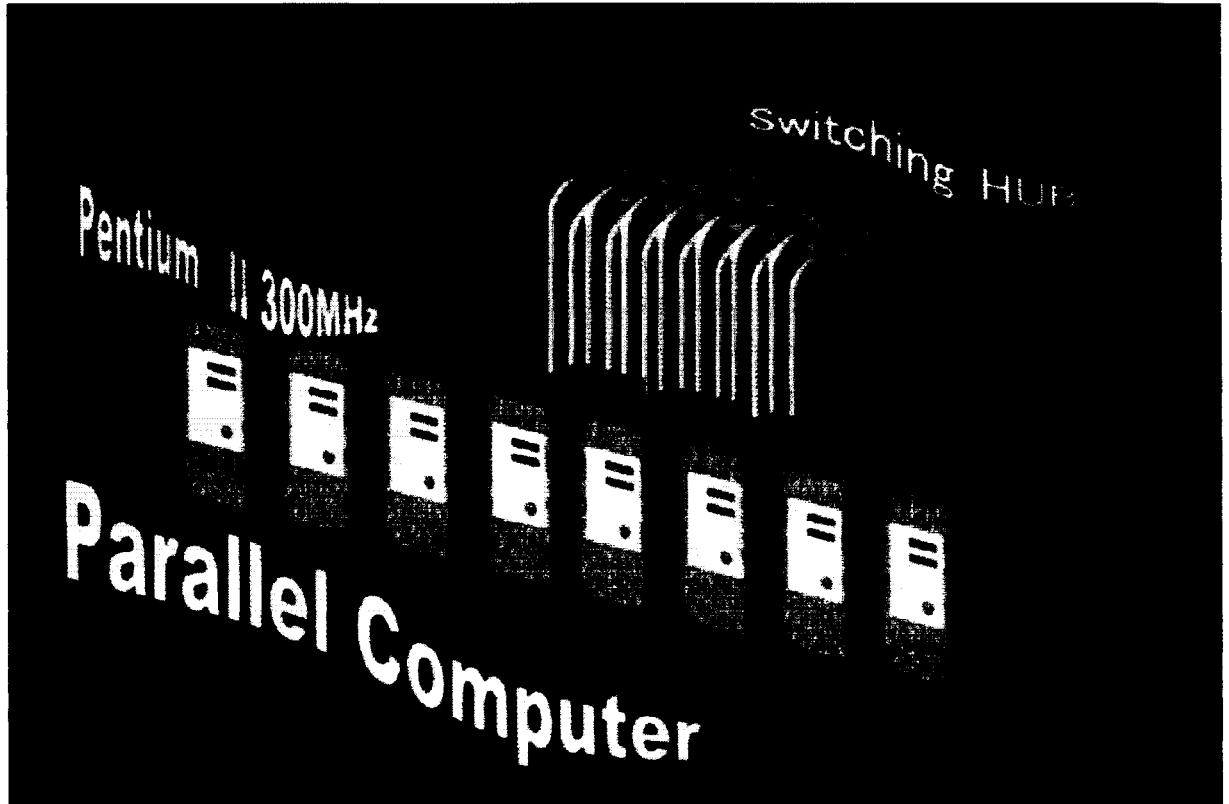


図 E.1: PC クラスタシステムの概念図

8 台のパーソナルコンピューターをスイッチングハブで接続することにより通信経路を確保する。

- パーソナルコンピューター (PC)

サーバー用 ×1

CPU: PentiumII 450MHz

Mem: 128MByte

HDD: IDE 6.4GByte

NIC: Digital 21140A Fast Ethernet 100BASE-T

Graphics: Riva TNT2

計算用 ×8

CPU: PentiumII 300MHz

Mem: 64MByte

HDD: IDE 4.2GByte

NIC: 3Com 3c905-TX Fast Etherlink XL 100BASE-T

ファイルサーバー

CPU: PentiumPro 200MHz

Mem: 64MByte



HDD: IDE 8.7GByte + 1.0GByte & SCSI 3.0GByte

NIC: Digital 21140A Fast Ethernet 100BASE-T

- ハブ  
16Port 100BASE-T Switching HUB

高速で安価な通信経路確保を目的として 100BASE ネットワークカードを利用し、スイッチングハブを用いることで通信パケットの衝突 (Collision) を減らしている。また、NFS を利用することによりデータ転送の簡便化を行った。

### E.1.2 OS 並びに MPI ライブラリの導入

パーソナルコンピューター用の UNIX OS (Operation System) として Linux, FreeBSD, OpenBSD, NetBSD, BeOS, Solaris, JAVA OS など様々なものが提供されている。並列計算機での計算を行うことを考えると、簡便なアプリケーションと比較して大容量メモリーを必要とする高負荷なものがその対象として考えられることから PC クラスターシステムに要求されることとして長期的な安定運用が挙げられる。また、常にネットワークトラフィックが発生することから通信に対する実績も必要とされる。上記に挙げた OS は様々な分野で利用されており実績もあることからいずれの OS を選択しても安定な PC クラスターシステムを構築することは可能であると考えられる。

しかしそのメンテナンスを考えた場合、利用者がもっとも熟知した OS を利用することが望ましいと考え、ここでは FreeBSD を用いることを考える。また以下の機能を付加した。

- Soft Updates の有効化
- NFS server の導入
- samba, netatalk の導入

Soft Updates とは、FreeBSD 特有のファイルシステムの形態で、非同期 (async) ファイルシステムと同期 (sync) ファイルシステムの長所を利用した特徴を持っている。非同期ファイルシステムであればファイルシステムの全ての I/O を非同期に行うためシステムクラッシュによるリポートが生じた場合はファイルシステムの再構築を行うことが非常に困難になる。その一方で同期ファイルシステムではファイルの書換えが生じる毎に i-node 情報の更新を行うことからファイルの I/O が一般的に遅くなる。Soft Updates では、安全性を確保しながらディスク I/O を高速化する方法である。

また、NFS サーバー導入を行うことによりファイルの共有を簡便にするとともに、Windows, Machintosh とのファイル共有を行うため、samba, netatalk を導入した。

### E.1.3 各種設定

並列計算を行う上でシステム上必要であると思われる設定ファイルを示す。

- /etc/hosts
- /etc/hosts.equiv
- /etc/rhosts

/etc/hosts を用いることで DNS (Domain Name Service) サーバーを参照すること無く相手の IP アドレスを知ることが出来ることからわずかであるがネットワークトラフィックを改善できる。hosts ファイルでは IP アドレスとその名前を列挙型で表 E.1 のように記述する。PC クラスターシステムにおける計算用コンピューターの名前を登録しておくことが望ましい。

次に `hosts.equiv` について説明する。今回並列計算を行うためのコンピューター間通信を MPI ライブラリを用いて実装する。MPI ライブラリでは他のコンピューター上でプロセスの起動を行うために “`rsh`” もしくは “`ssh`” を利用する。これらのコマンドは共にパスワードを必要とせず他のコンピューターへログインするために用いられるものである。

表 E.1: `hosts` ファイルの書式

`hosts` ファイルを利用することで DNS サーバーへの問い合わせを減らすことが出来る。列挙型で IP アドレスとその名前を記述する。

#	IP Address	Host Name	
	192.168.0.1	hostname001	hostname001.domain
	192.168.0.2	hostname002	hostname002.domain
	192.168.0.3	hostname003	hostname003.domain
	192.168.0.4	hostname004	hostname004.domain
	192.168.0.5	hostname005	hostname005.domain
	192.168.0.6	hostname006	hostname006.domain
	⋮	⋮	⋮

しかし、デフォルトの状態ではこれらのコマンドを用いたとしてもパスワードの問い合わせが必要となるため、MPI プログラムを実行するためのスクリプト “`mpirun`” によるホストコンピューター上のプロセス起動が不可能である。“`rsh`”、“`ssh`” をパスワード無しで起動するために必要となるのが “`hosts.equiv`” ならびに “`/.rhosts`” ファイルである。これらのファイルに `hosts` ファイル同様、IP アドレスとホスト名を列挙型で記述することにより、その中に書かれた計算機とはパスワード無しで “`rsh`”、“`ssh`” により接続 (`login`) できるようになる。ただし、“`hosts.equiv`” は管理者 (`root`) が設定し、“`/.rhosts`” は各ユーザーのホームディレクトリで定義する点異なる。

## E.2 MPI を用いた並列コードの開発

並列計算コード開発支援ライブラリとして MPI (Message Passing Interface) ライブラリを用いることとした。同様の並列計算用通信ライブラリとして PVM (Parallel Virtual Machine) ライブラリが有名である。両ライブラリともにサーバクライアント型のインターフェースを提供するが、MPI では各コンピュータを同等に扱い、同じ内容の計算をさせることを主流としているのに対し、PVM ではジョブ毎に様々な CPU に振り分けると考えることが出来る。ただし、近年ではバージョンアップされる毎に両者の間に差がほとんど見られなくなっていることからどちらを利用しても問題は無いと考えるが、ここでは MPI ライブラリを用いた並列コードの開発について説明する。

### E.2.1 MPI (Message Passing Interface) ライブラリ

MPI ライブラリは情報 (Message) の伝達 (Passing) を行うプログラムのための標準規格として開発されて来ている。メッセージ・パッシング (message Passing) 技術は主に分散メモリ型の並列計算機で用いられおり、その標準化を行うことで異なる環境への委嘱が効果的に行えるようになり、又ハードウェアの構成を替えることでスケラビリティを増大させることが出来るようになる。MPI は標準化を達成するために目的として以下の項目を掲げている。

- コンパイラやシステムが装備しているライブラリに依存しないプログラミングインターフェースの構築。
- 効果的な情報通信：メモリのコピーを避け、計算と通信の多重処理を許可し、可能であれば通信負荷を通信専用のコ・プロセッサに任せる。
- 異種環境間での実装を認める。
- C や Fortran 77 からの呼び出しを可能にする。
- 信頼性の高い通信インターフェースの確立：ユーザー側で通信の成功、失敗に対する処理をしなくて良い様にする。通信の不成功は MPI の通信サブシステム下で対処する。
- 現在並列計算用に用いられている PVM, NX, Express, p4 等とあまり変わらないインターフェースを定義し、より自由度の高い拡張性を提供する。
- 通信、システムソフトウェア内で手を加えることなく、さまざまなプラットフォーム上で実装可能なインターフェースを定義する。
- インターフェース関数名の意味はプログラミング言語依存とする。
- 並列実行下でも安全なインターフェースを提供する。

### E.2.2 MPI で利用できるデータ型

MPI により送受信できるデータ型とそれに対応する C 言語でのデータ型を表 E.2 に示す。

データ型 MPLBYTE と MPLPACKED は C 言語のデータ型に対応するものが存在しない。MPLBYTE は、1 バイトで構成されるデータを表す。char 型で表される文字は異なるマシンでは異なる文字表現になることがあり、また文字を表現する時に複数バイト使用することもあるが、バイト型は全てのマシンで同じ 2 進値を持つ。MPLPACKED はユーザー定義により MPLPACKED 以外の型を複合して新たな型として利用する際に用いられる。これによりさまざまな型の変数を一度の通信で送信、受信できるようになる。

表 E.2: MPI で利用できる変数の型と対応する C 言語での型

MPI datatype	C datatype
MPLCHAR	signed char
MPLSHORT	signed short int
MPLINT	signed int
MPLLONG	signed long int
MPLUNSIGNED_CHAR	unsigned char
MPLUNSIGNED_SHORT	unsigned short int
MPLUNSIGNED	unsigned int
MPLUNSIGNED_LONG	unsigned long int
MPLFLOAT	float
MPLDOUBLE	double
MPLLONG_DOUBLE	long double
MPLBYTE	
MPLPACKED	

### E.2.3 グループ、コミュニケーター の概念

MPI を用いて並列プログラムを作成すると、起動された各ホスト上のプロセスは下記に述べるグループ、コミュニケーター の概念に従って情報ネットワークに属する。グループ、コミュニケーター を用いることで並列プログラミングが飛躍的に簡単になることから、その概念の理解は有効である。

#### MPLCOMM\_WORLD

MPI を用いて書かれたプログラムが mpirun により起動されると、その引数に応じたプロセスが計算機上に生成され、それら全てのプロセスを含む MPLCOMM\_WORLD と呼ばれる並列計算空間中の仮想世界が生成される。

図 E.2 に示すように MPLCOMM\_WORLD 中では全てのプロセスに rank と呼ばれる識別番号が与えられている。ただし、rank の大小によりプロセスの優劣が決まることは無く、全てのプロセスは平等に扱われる。

MPI 以前の並列プログラミングは、おおよそこの様な状態から利用者 (プログラマー) がそれぞれのプロセスの役割を決め、どのプロセスとどのプロセスが通信するかなどのトポロジーを決めることから始まった。MPI ではこういった手間を省くためにグループやコミュニケーター と呼ばれるプロセスの集合体を作る支援関数が用意されており、比較的簡単に通信トポロジーの作成が可能になっている。またそれに伴った集団通信関数も用意されているので開発が簡便になっている。

#### Group, Communicator

MPLCOMM\_WORLD の概念のみで通信を行うプログラムを書くことは可能であるが、通信相手の決定、集団通信のルーチンなど全ての手続きをユーザーが実装しなくてはならなくなる。幸いなことに MPI

ではそれらを行うためのいくつかの支援関数を準備していることから、その利用を考える。そのためにはまずプロセスの群 (group) の概念を理解する必要がある。

group に分割されたプロセスはそのグループ内でユニークな rank を新たに取得することができる。ただし、MPI\_COMM\_WORLD の世界から参照した場合、MPI 通信世界を生成した時点で割り振られた、MPI\_COMM\_WORLD 内でユニークな rank を返すこともできる (図 E.3)。

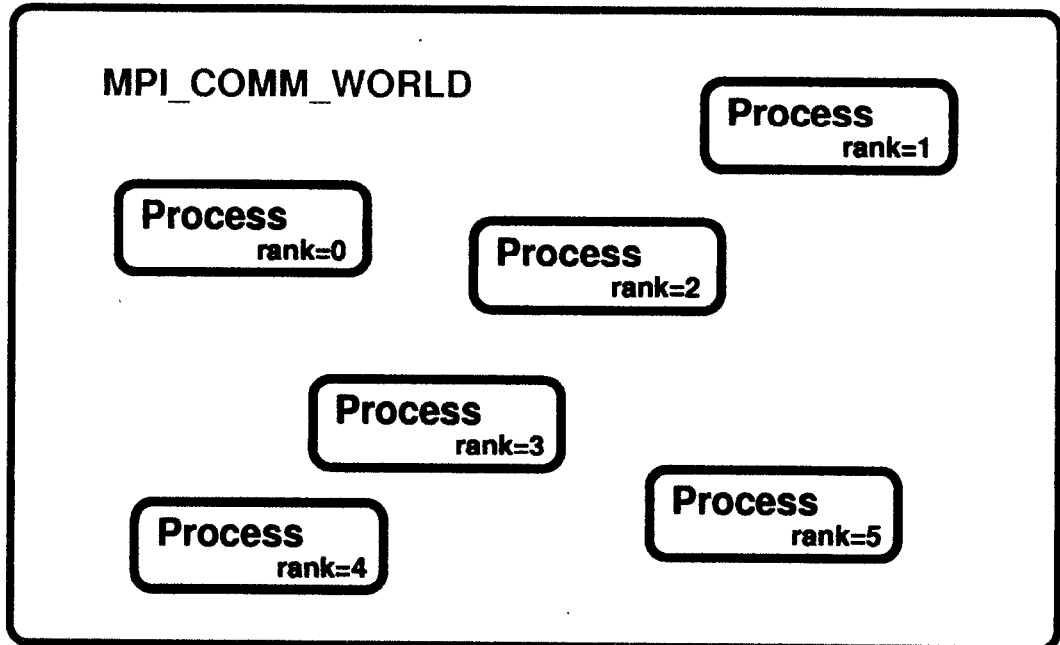


図 E.2: MPI によって作成された仮想並列空間は MPI\_COMM\_WORLD と呼ばれる全プロセスを含む集合体により包括されている。

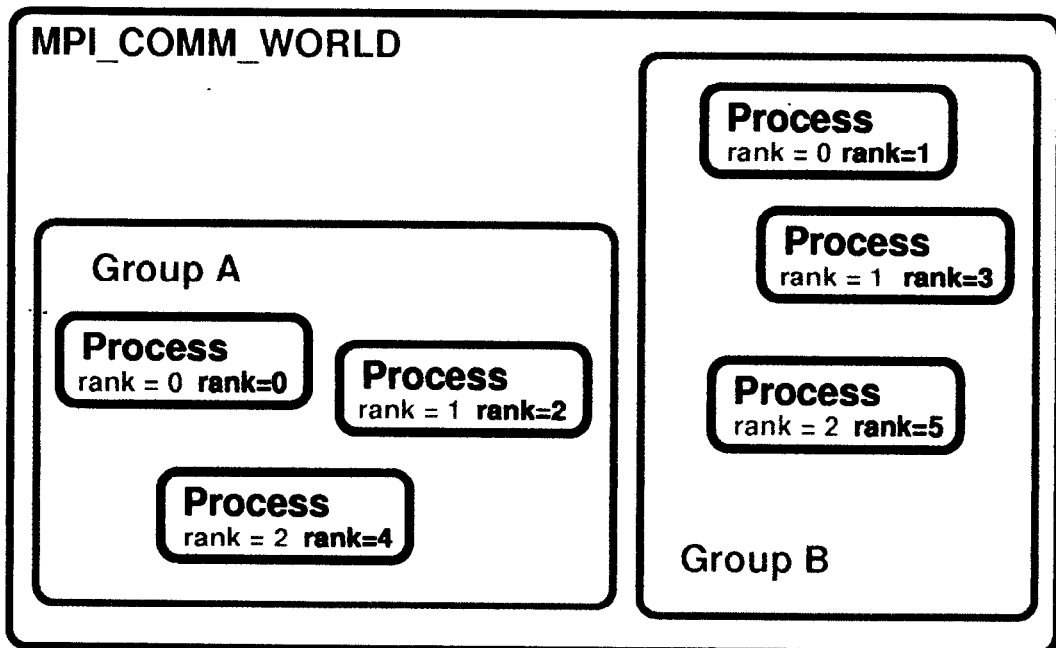


図 E.3: MPI における group の概念

group を形成し、プロセスをいくつかの群に分けることでプロセス管理を簡便にすることが出来る。

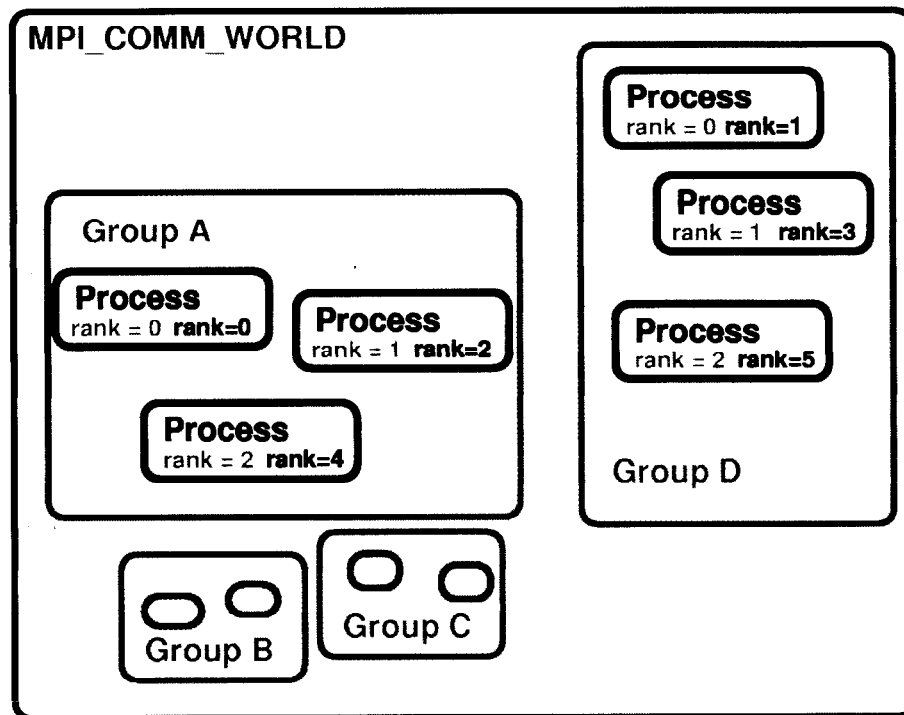


図 E.4: コミュニケーター (Communicator) の概念  
 コミュニケーターはいくつかのグループを含むことができる。

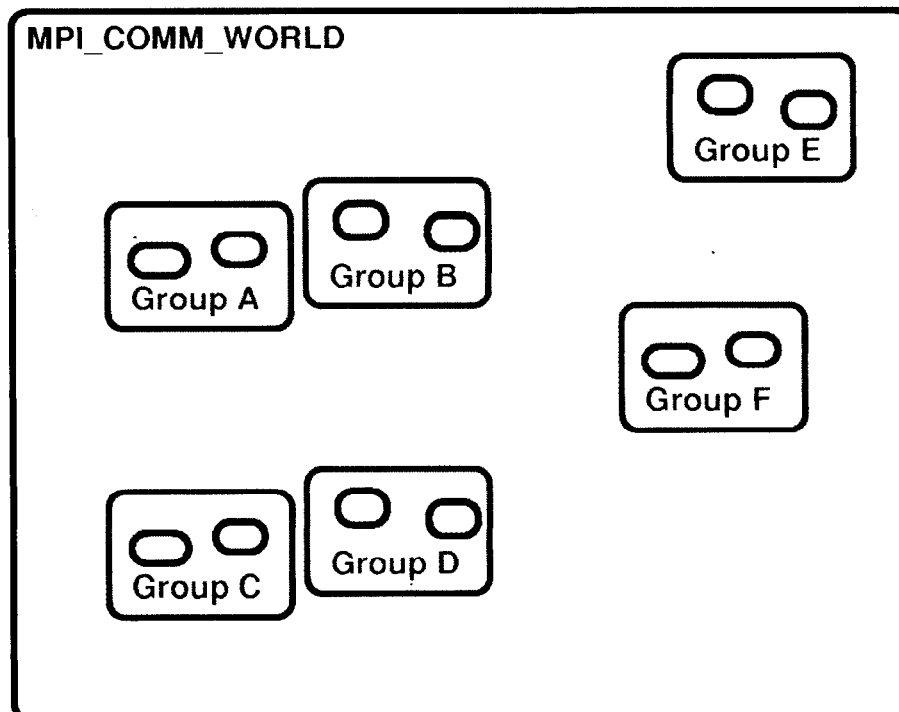


図 E.5: コミュニケーター間通信  
 コミュニケーター間の通信を行うにはそれらのコミュニケーターを含む新しいコミュニケーターが必要になる。

group をベースとした通信を行うと group 内のみの通信となる (他の group のプロセスを参照できないため) が保証される。group 間で通信を行う場合、コミュニケーター (Communicator) を用いなければならない。MPI\_COMM\_WORLD もコミュニケーターの一つであることから、MPI\_COMM\_WORLD を利用した通信も可能であるが、グループ毎に新しくコミュニケーターを作成し、概念的に分かりやすい通信経路の確保を行う。

コミュニケーター概念はそもそもグループ概念から生成されているので基本的な考え方は同じであるが、相違点として集団通信の関数がコミュニケーター内で用いられることと、内包するプロセスをより細かな集団 (group) に分割できることが挙げられる (図 E.4)。グループ内通信同様に、コミュニケーター内での通信は他のコミュニケーターに属するプロセスとの通信に影響しないことが保証されている。つまり、コミュニケーター間の通信を行うためには通信を行いたい目的のコミュニケーターを全て含む新しいコミュニケーター (inter communicator) を作成する必要がある (図 E.5)。

## E.2.4 グループ管理

グループを使用する時に必要となる関数、およびその使用方法について説明する。その実行にはプロセス間通信を必要としない。

### グループ参照関数

- int MPI\_Group\_size(MPI\_Group group, int \*size)

入力 group グループ (ハンドル)

出力 size グループ中のプロセス数 (整数型)

この関数は、group で指定するグループに含まれるプロセス数を size に返す。

- int MPI\_Group\_rank(MPI\_Group group, int \*rank)

入力 group グループ (ハンドル)

出力 rank 呼び出しプロセスのグループ内のランク

この関数は呼び出したプロセスの group 内におけるランクを rank に返す。プロセスがグループのメンバーでない場合 MPI\_UNDEFINED を返す。

### グループコンストラクタ

- int MPI\_Group\_union(MPI\_Group group1, MPI\_Group group2, MPI\_Group \*newgroup)

入力 group1 第一グループ (ハンドル)

入力 group2 第二グループ (ハンドル)

出力 newgroup 和グループ (ハンドル)

第一グループ (group1) の全ての要素の後に、第一グループに無い第二グループ (group2) の全ての要素を続け、和グループ (newgroup) を生成する。

- int MPI\_Group\_intersection(MPI\_Group group1, MPI\_Group group2, MPI\_Group \*newgroup)

入力 group1 第一グループ (ハンドル)

入力 group2 第二グループ (ハンドル)

出力 newgroup 積グループ (ハンドル)

第一グループの要素で第二グループの要素でもある全ての要素を積グループ (newgroup) に抽出する。順序は第一グループの通りとなる。

- `int MPI_Group_difference(MPI_Group group1, MPI_Group group2, MPI_Group *newgroup)`

入力	group1	第一グループ (ハンドル)
入力	group2	第二グループ (ハンドル)
出力	newgroup	差グループ (ハンドル)

第二グループに無い第一グループの要素で差グループを構成する。順序は第一グループの通りとなる。上記操作により新しいグループが空集合になる場合は、新規グループは `MPI_GROUP_EMPTY` に等しくなる。

- `int MPI_Group_incl(MPI_Group group, int n, int *ranks, MPI_Group *newgroup)`

入力	group	グループ (ハンドル)
入力	n	配列 ranks の要素数
入力	ranks	newgroup に出力される group 中のランクの配列
出力	newgroup	生成されるグループ (ハンドル)

関数 `MPI_GROUP_INCL` は、グループ group 中のランク `rank[0], …, rank[n-1]` の n 個のプロセスからなるグループ newgroup を生成する。

- `int MPI_Group_excl(MPI_Group group, int n, int *ranks, MPI_Group *newgroup)`

入力	group	グループ (ハンドル)
入力	n	配列 ranks の要素数
入力	ranks	newgroup に出力さない group 中のランクの配列
出力	newgroup	生成されるグループ (ハンドル)

関数 `MPI_GROUP_EXCL` は、グループ group 中からランク `rank[0], …, rank[n-1]` のプロセスを削除することで得られるプロセスのグループ newgroup を作成する。

### グループデストラクタ

- `int MPI_Group_free(MPI_Group group)`

入出力	group	グループ (ハンドル)
-----	-------	-------------

この関数は、グループ・オブジェクトに解放マークをつける。ハンドル group は、呼び出しにより `MPI_GROUP_NULL` に設定される。このグループを使用する実行中の操作は全て正常に完了する。

他にもグループ操作に関する関数はいくつかあるが、ここでは省略させて頂く。



## E.2.5 コミュニケーター管理

### コミュニケーター操作

- `int MPI_Comm_size(MPI_Comm comm, int *size)`

入力 `comm` コミュニケーター (ハンドル)  
出力 `size` コミュニケーター中のプロセス数

`MPI_Comm_size` はコミュニケーターにかかわるプロセスの数を返す。コミュニケーターとして並列計算プロセス起動時に生成される `MPI_COMM_WORLD` を選択すると、利用可能なプロセスの総数をあらわす。

- `int MPI_Comm_rank(MPI_Comm comm, int *rank)`

入力 `comm` コミュニケーター (ハンドル)  
出力 `rank` 呼び出しプロセスのコミュニケーター内のランク

`MPI_Comm_rank` は、特定のコミュニケーター内の自プロセスのランクを与える。

### コミュニケーターコンストラクタ

- `int MPI_Comm_create(MPI_Comm comm, MPI_Group group, MPI_Comm *newcomm)`

入力 `comm` コミュニケーター (ハンドル)  
入力 `group` グループ (ハンドル)  
出力 `newcomm` 新しいコミュニケーター (ハンドル)

この関数は、`group` によって定義された通信グループと新しいコンテキストを持つ新規コミュニケーター `newcomm` を生成する。この呼び出しは、新規グループに属す属さないにかかわらず `comm` 野全てのプロセスによって実行されるべきである。またこの呼び出しはグループ内コミュニケーターにのみ適用される。

- `int MPI_Comm_split(MPI_Comm comm, int color, int key, MPI_Comm *newcomm)`

入力 `comm` コミュニケーター (ハンドル)  
入力 `color` サブセット割当の制御  
入力 `key` ランク割当の制御  
出力 `newcomm` 新しいコミュニケーター (ハンドル)

この関数は、`comm` に付随するグループを `color` の値に対応してそれぞれ重ならないサブグループに分割する。各サブグループは同一 `color` のプロセスを全て含む。各サブグループ内で、プロセスは引数 `key` の値の順序でランクが付けられ、同一の `key` のプロセスは旧グループのランク順に従ってランクが決められる。

### コミュニケーターストラクタ

- `int MPI_Comm_free(MPI_Comm *comm)`

入出力 `comm` 壊されるコミュニケーターストラクタ

この関数は、通信オブジェクトに解放のマークを付ける。ハンドル `comm` は `MPI_COMM_NULL` にセットされる。このコミュニケーターストラクタを使用中の保留通信操作は全て正常終了する。

### E.2.6 プロセストポロジー

実際にコミュニケーターストラクタを利用するにあたり、さまざまな支援関数が MPI では用意されている。MPI のプロセスを  $n$  個の集まりとすると、グループ内の各プロセスには 0 から  $n-1$  までのランクがそれぞれに割り当てられる。多くの並列アプリケーションでは、プロセス群への線形でのランク付けはプロセスの論理的通信パターンを充分には反映していない。多くの場合、プロセス群は二次元的、あるいは三次元的なトポジカルパターンに配置されるべきである。より一般的には論理的过程配置はグラフで表現されるであろう。この章では、この論理的过程配置を『仮想トポロジー』と呼ぶことにする。

仮想トポロジーは、元になる物理的ハードウェアのトポロジーとは全く異なるものである。

### E.2.7 カルテシアン構造

並列アプリケーションの大部分は、環、二次あるいは高次元の格子、またはトーラスのようなプロセストポロジーを使用する。これらは次元数と各座標方向におけるプロセスの数を指定すれば完全に定義できる、デカルト座標であらわされる直積構造である。このように直積であらわされるものをカルテシアンと呼ぶことにする。

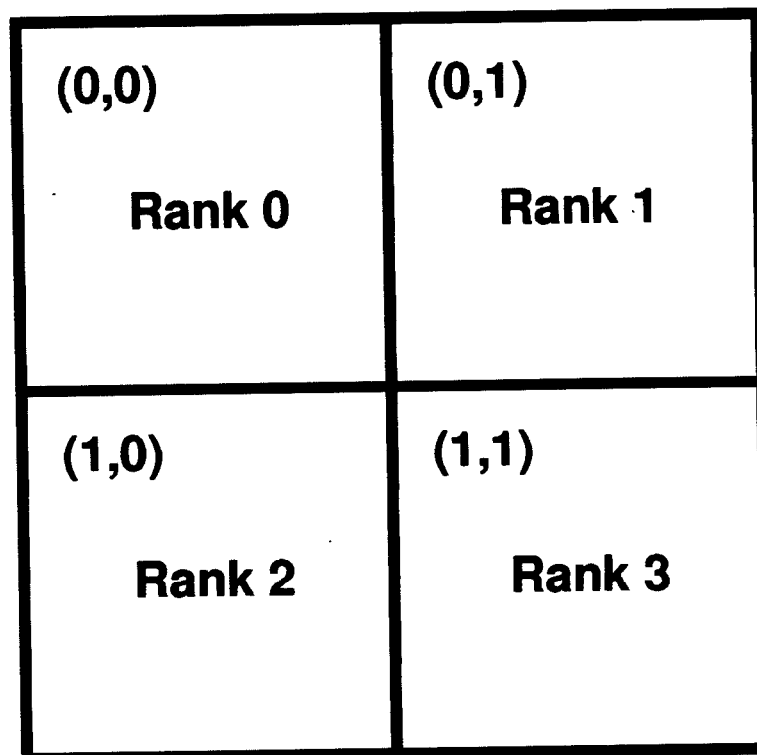


図 E.6: カルテシアン構造を持つプロセス群

仮想トポロジーとして 2 次元配列形式をとるカルテシアン構造を用いることでプロセス間の相互関係がより明確になる。

カルテシアン構造のプロセス座標には、0 から番号をつける。カルテシアン構造を持つプロセス群には常に行優先番号付けを使用する。例えば図 E.6 に示すような (2 × 2) 格子における 4 個のプロセスのグループランクと座標には次の関係があることを意味する。

#### カルテシアンコンストラクタ

- `int MPI_Cart_create(MPI_Comm comm_old, int ndims, int *dims, int *periods, int reorder, MPI_Comm *comm_cart)`

入力	<code>comm_old</code>	入力コミュニケーター
入力	<code>ndims</code>	カルテシアン格子の次元数
入力	<code>dims</code>	各次元に対しそのプロセス数を指定した <code>ndims</code> 次元の整数型配列
入力	<code>periods</code>	各次元に対しその格子が周期的か否かを指定する論理型配列
入力	<code>reorder</code>	ランク付けを変更するか否か
出力	<code>comm_cart</code>	カルテシアントポロジーを持つコミュニケーター

`MPI_CART_CREATE` は、カルテシアン・トポロジー情報を付加した新しいコミュニケーターのハンドルを返す。`reorder = false` であれば、新規グループの中の各プロセスのランクは旧グループのそれと同じになる。

#### カルテシアン支援関数

カルテシアン・トポロジーでは関数 `MPI_DIMS_CREATE` を使用して、バランスをとるべきグループ内のプロセスの個数とユーザーが指定できるオプションな制約に応じて、座標方交互にバランスのとれたプロセス配置を選ぶことが出来る。これによって全てのプロセスを `n` 次元トポロジーに分割することが出来る。

- `int MPI_Dims_create(int nnodes, int ndims, int *dims)`

入力	<code>nnodes</code>	格子中のノード数
入力	<code>ndims</code>	カルテシアントポロジーの次元数
出力	<code>dims</code>	各次元に対しそのノード数を指定したサイズの配列

配列 `dims` のエンタリは、`ndims` 次元でかつ全部で `nnodes` 個のノードのカルテシアン格子を記述するように設定される。次元は適切な分割アルゴリズムを使用して、出来るだけ互いに近い値になるように設定される。呼び出し側は更に配列 `dims` の要素を指定することで、このルーチンの操作を制限することが出来る。`dims[i]` が正数に指定されている場合、このルーチンは次元 `i` のノードの数を変更しない。

#### 例

- 呼び出し前の `dims` (0,0)  
関数呼び出し `MPI_DIMS_CREATE(6,2,dims)`  
戻り値 (3,2)
- 呼び出し前の `dims` (0,0)  
関数呼び出し `MPI_DIMS_CREATE(7,2,dims)`  
戻り値 (7,1)

- 呼び出し前の dims (0,3,0)  
関数呼び出し MPI\_DIMS\_CREATE(6,3,dims)  
戻り値 (2,3,1)
- 呼び出し前の dims (0,3,0)  
関数呼び出し MPI\_DIMS\_CREATE(7,3,dims)  
間違った呼び出し

### カルテシアントポロジーへの問い合わせ関数

カルテシアントポロジーを利用するためにいくつかの問い合わせ関数が MPI 側で用意されているが、ここでは良く利用するもののみについて説明する。

- `int MPI_Cart_rank(MPI_Comm comm, int *coords, int *rank);`

入力	comm	カルテシアン構造を持つコミュニケータ
入力	coords	プロセスのカルテシアン座標を指定した配列
出力	rank	指定したプロセスのランク

関数 `MPI_CART_RANK` は、カルテシアン座標値から、それに対応するプロセスのランクに変換する。この変換は 1 対 1 対応する。周期的なカルテシアン構造では範囲外の座標を指定すると自動的にシフトされて値が返される。これの逆変換 (ランク → カルテシアン座標値) を行うのは次に挙げる `MPI_CART_COORDS` である。

- `int MPI_Cart_coords(MPI_Comm comm, int rank, int maxdims, int *coords);`

入力	comm	カルテシアン構造を持つコミュニケータ
入力	rank	comm の中でのプロセスのランク入力
入力	maxdims	呼び出し側プログラムの coords のサイズ
出力	coords	プロセスのカルテシアン座標値

関数 `MPI_CART_COORDS` は `MPI_CART_RANK` とは逆にプロセスのランク値から対応するカルテシアン座標値に変換する。

### E.2.8 グラフ構造

より一般的なトポロジーの生成のために MPI ライブラリではグラフ構造トポロジーの生成支援関数を用意してある。グラフトポロジーを用いることでカルテシアントポロジーを含め、さまざまなトポロジーの生成が可能である。

数値流体計算では境界条件が複雑になると、計算領域を分割した方が計算効率が良い場合が多い。例えば飛行機周りの流れを解く場合、胴体部、主翼部を分割して計算することが可能であり、地下鉄の駅構内流れの計算では、プラットフォーム、階段を独立した計算領域にすることが出来る。そういった場合にはカルテシアン構造よりもグラフ構造を用いた計算領域分割を行うことが望ましい。

グラフコンストラクタにより計算領域を分割して新しくコミュニケータを用意おけば、計算中は計算領域の順序を意識せずに通信を行うことが出来ることから、トポロジーの利用を推奨する。

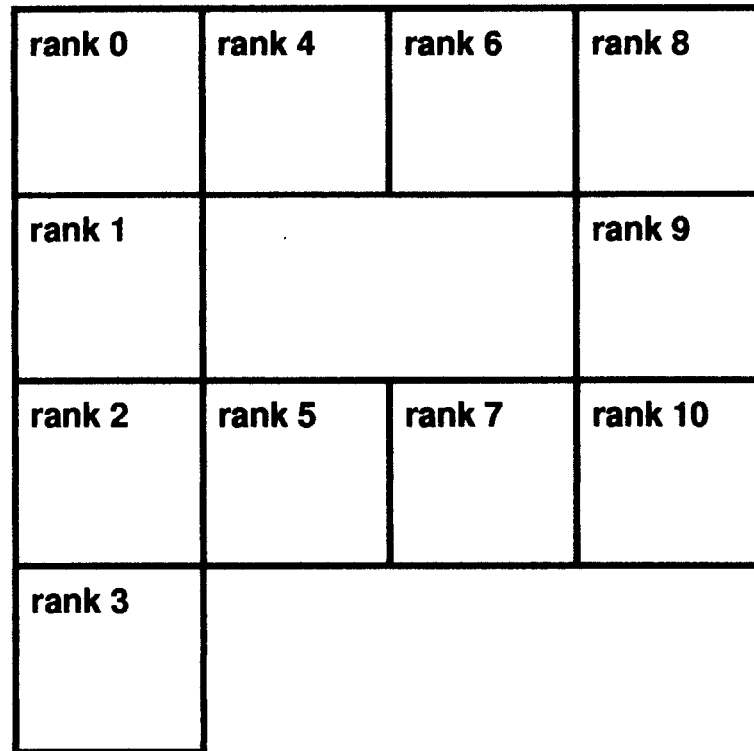


図 E.7: グラフ構造を持つプロセス群

仮想トポロジーとしてグラフ構造を扱うことでより複雑なプロセス間相互関係を表現することが可能になる。

#### グラフコンストラクタ

- `int MPI_Graph_create(MPI_Comm comm, int nnodes, int *index, int *edges, int reorder, MPI_Comm *comm_graph);`

入力	<code>comm</code>	入力コミュニケータ
入力	<code>rank</code>	グラフのノードの数
入力	<code>index</code>	ノードの次数をあらわす配列
入力	<code>edges</code>	グラフのエッジをあらわす配列
入力	<code>reorder</code>	ランク付けを変更するか否か論理型
出力	<code>comm_graph</code>	グラフトポロジーを持つコミュニケータ

`MPI_GRAPH_CREATE` は、グラフトポロジー情報が付加された新しいコミュニケータのハンドルを返す。`reorder = false` であれば、新規グループの中の各プロセスのランクは旧グループのものと同じになる。

グラフのサイズ `nnodes` が、`comm` のグループのサイズよりも小さければ、`MPI_CART_CREATE` と同様に、いくつかのプロセスには `MPI_COMM_NULL` を返す。つまり、この関数を入力コミュニケータのグループサイズよりも大きなグラフを指定して呼び出すことは誤りである。

3 つのパラメータ `nnodes`、`index` および `edges` でグラフ構造を定義する。`nnodes` はグラフのノードの個数である。ノードには 0 から `nnodes-1` までの番号が付けられる。配列 `index` の `i` 番目のエン

トリーには、最初の  $i$  個のグラフのノードの近傍の総数が格納される。ノード  $0, 1, \dots, \text{nnodes}-1$  の近傍のリストには、配列 `edges` 中の連続した位置に格納される。それゆえに、配列 `edges` はエッジリストを平坦化した表現となる。index のエントリの総数は `nnodes` で、`edges` のエントリの総数はグラフのエッジの本数に等しい。

例として、図 E.8 ようなグラフトポロジーの生成を考える。

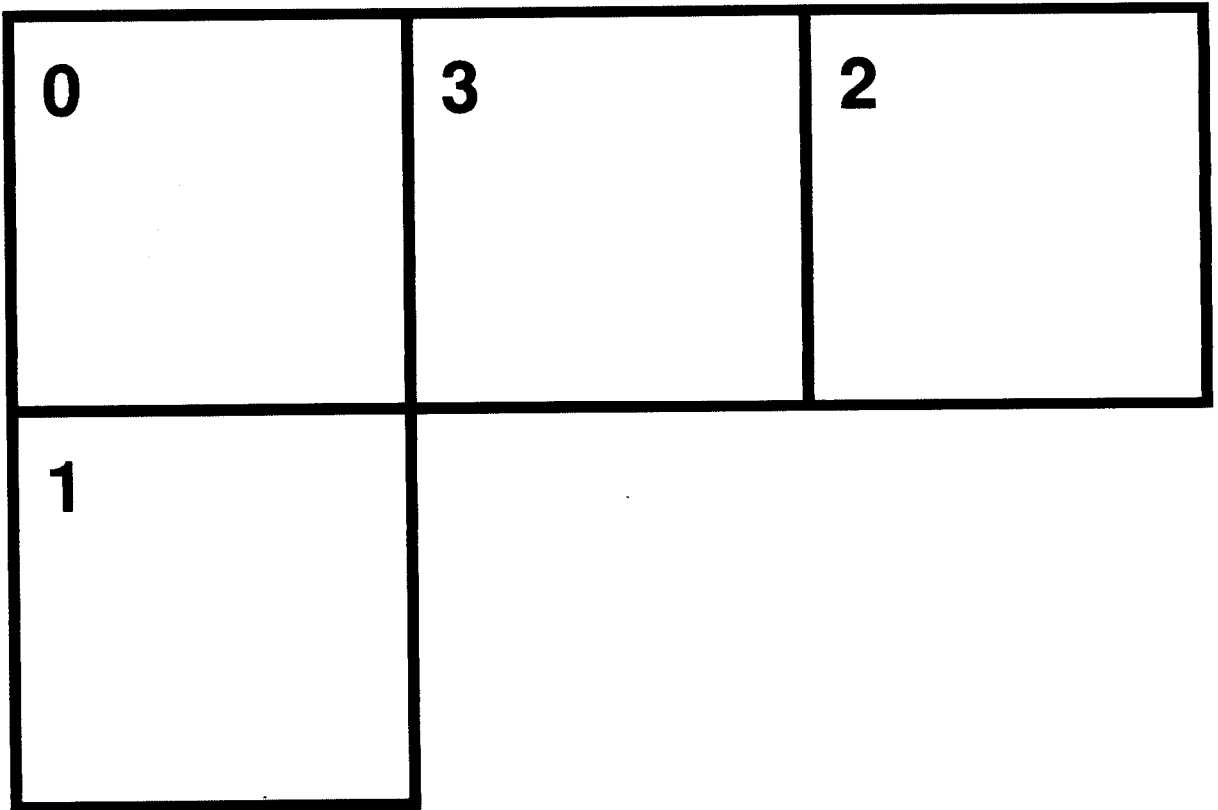


図 E.8: 仮想トポロジーとしてのグラフ構造の例

隣接関係を簡単にまとめると、

プロセス	近傍
0	1,3
1	0
2	3
3	0, 2

となる。`nnodes` は全体のプロセス数をあらわしているので、4。`index` には、引数までのランクを持つプロセスが隣接するプロセスの総数をあらわしているため、それぞれのプロセスが隣接するプロセスの数から、

rank	隣接数	index
0	2	2
1	1	3
2	1	4
3	2	6

が求まる。また、各プロセスが隣接するプロセスの rank から、edges が、

rank	隣接するプロセスの rank	edges
0	1,3	1,3
1	0	1,3,0
2	3	1,3,0,3
3	0,2	1,3,0,3,0,2

となるので、各入力変数は、

nnodes = 4,      index = 2,3,4,6,      edges = 1,3,0,3,0,2

と最終的に求まる。

#### グラフトポロジへの問い合わせ関数

MPL\_GRAPH\_NEIGHBOERS\_COUNT および MPL\_GRAPH\_NEIGHBOERS は一般的なグラフトポロジの隣接情報を提供する。

- int MPI\_Graph\_neighbors\_count(MPI\_Comm comm, int rank, int \*nneighbors);

入力 comm      グラフトポロジを持つコミュニケータ  
 入力 rank      comm 内でのプロセスのランク  
 出力 nneighbors      指定したプロセスの近傍の数

- int MPI\_Graph\_neighbors(MPI\_Comm comm, int rank, int maxneighbors, int \*neighbors);

入力 comm      グラフトポロジを持つコミュニケータ  
 入力 rank      comm 内でのプロセスのランク  
 入力 maxneighbors      neighbors のサイズ  
 出力 nneighbors      指定したプロセスの近傍にあたるプロセスのランク

#### E.2.9 ブロッキング通信、ノンブロッキング通信

通信方法により、ノンブロッキング通信 (図 E.9) とブロッキング通信 (図 E.10) に分類できる。

ブロッキング通信ではデータ通信中は操作が完了するまで通信に利用しているリソースを再使用することが出来ない。つまりデータ転送が完了するのを必ず待たなければならない。しかし、ブロッキング通信を用いている限りはデータが不当に書き換えられたりしない (情報の安全性) ことが保証されている。

これに比べノンブロッキング通信では、データ転送が完了する前で、なおかつ呼び出しにおいて指定されたリソースをユーザーが再使用できるようになる前に手続きから戻り得る。つまりデータ転送を行って

いる最中に計算可能になる(手続きの完了を待たなくて良い)可能性がある。並列計算の高速化を考える場合、データ転送の問題が無視できなくなる場合が多い。そのような状況下ではデータ転送と計算を同時に行うことが非常に有利である。ただしユーザーが責任を持ってデータの整合性を確保するプログラミングに努めなければならない。

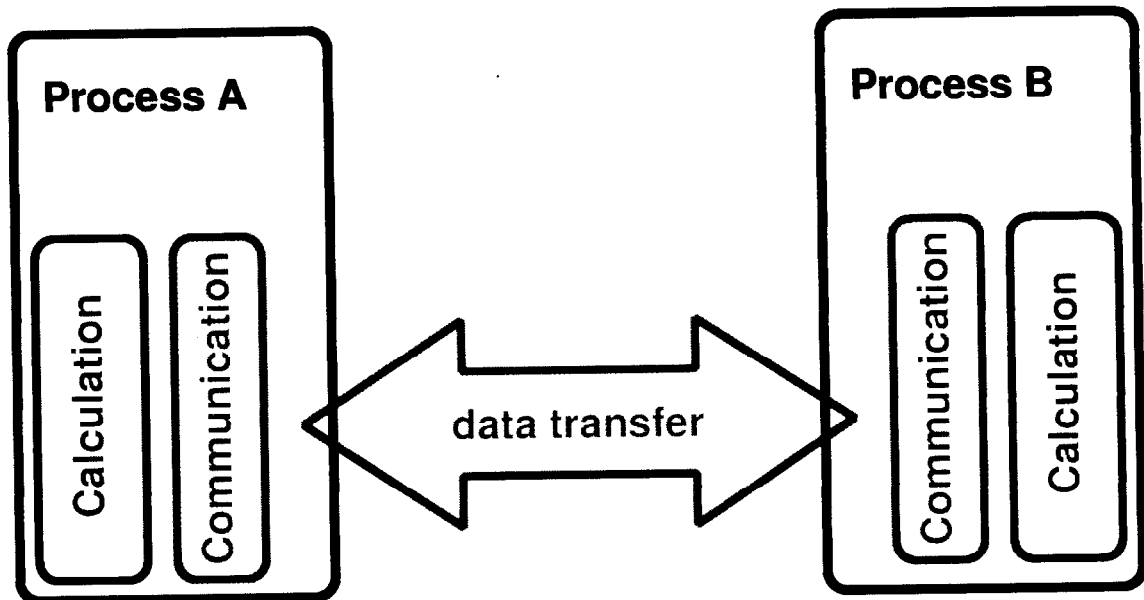


図 E.9: ノンブロッキング通信では、データ転送と同時に演算を行える可能性がある。

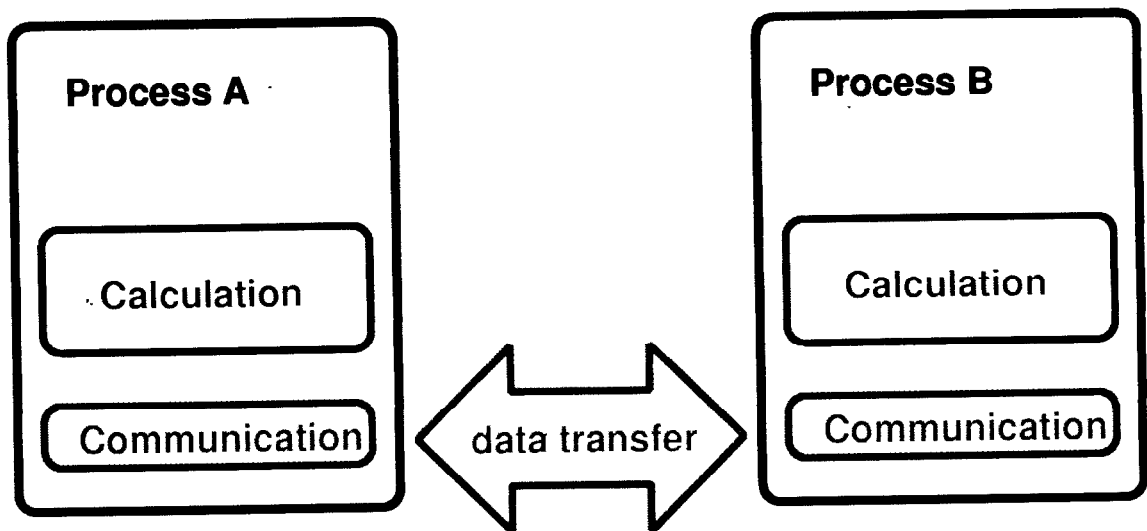


図 E.10: ブロッキング通信ではデータ転送の完了を待つ。



## E.2.10 1 対 1 通信

MPI における基本的な通信手段として 1 対 1 通信が準備されている。実際に通信可能な次のプログラムを用いて説明を行う。

```
#include "mpi.h"
int main(argc, argv)
    int argc;
    char **argv;
{
    char message[20];
    MPI_Status status;
    MPI_Init(&argc, &argv);
    MPI_Comm_rank( MPI_COMM_WORLD, &myrank );
    if (myrank == 0) /* プロセス番号 = 0 */
    {
        strcpy(message, "Hello, there");
        MPI_Send(message, strlen(message),
                 MPI_CHAR, 1, 99, MPI_COMM_WORLD);
    }
    else /* プロセス番号 = 1 */
    {
        MPI_Recv(message, 20, MPI_CHAR, 0, 99,
                 MPI_COMM_WORLD, &status);
        printf("received :%s:\n", message);
    }
    MPI_Finalize();

    return 0;
}
```

このプログラムでは、MPI\_Init によりプロセスを初期化し、MPI\_Comm\_rank により、自身のランクを取得するとともに rank = 0 であればメッセージの送信を、rank = 1 であればメッセージの受信を行っている。最後に MPI\_Finalize() によりプログラムを終了する。MPI における 1 対 1 通信はプログラム中で使用されている MPI 関数 MPI\_Send(), MPI\_Recv() によって基本的に行われる。

## ブロッキング通信のための MPI 関数

- int MPI\_Send(void\* buf, int count, MPI\_Datatype datatype, int dest, int tag, MPI\_Comm comm)

入力	buf	送信バッファの先頭アドレス
入力	count	送信バッファ内の要素数
入力	datatype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター

MPI における通信では、データ部分の他にメッセージを区別したり、選択的に受信する時に使用する情報を付加して伝える。これらをメッセージ・エンベロープと呼び、次のもので構成される。

- 送信元
- 送信先
- タグ
- コミュニケーター

送信元の情報は、メッセージの送信側の ID によって暗黙的に決定される。メッセージの送信先は dest 引数で指定する。タグは整数値であり、これによって情報の区別が可能になる。comm 引数により、送信関数で使用するコミュニケーターが指定される。

ここで用いた MPI\_Send は、ブロッキング通信の関数であり、バッファを利用することが出来ない。そのため、同時にいくつかの通信を行う際に、通信のブロックが生じるとデッドロックが発生する危険性がある。

### 通信の種類

MPI ライブラリでは、標準通信以外に次の通信関数を利用することが出来る。

- バッファモード

対応する受信が発行されているかどうかにかかわらず、開始することが出来る。これは、対応する受信が発行される前に完了することも想定している。つまり、標準送信とは異なり、この操作はローカルであり、他のプロセスに影響しない。そして、その完了は対応する受信が発生するかどうか依存しない。したがって、送信が実行され、対応する受信が発行されない場合には MPI は、送信呼び出しが完了できるように送出メッセージをバッファリングしなければならない。バッファ領域が不十分な場合はエラーとなる。利用可能なバッファ領域の容量はユーザーが制御しなければならず、バッファモードを有効に機能させるためにはユーザーがバッファアロケーションを行わなければならない。

- 同期モード

対応する受信が発行されているかどうかにかかわらず、開始することが出来る。しかしながら、送信は対応する受信が発行され、受信操作が同期送信によって送信されたメッセージを受信開始した場合にのみ成功完了する。つまり、同期送信の完了は送信バッファが再利用できることを示すだけでなく、受信側が実行におけるあるポイントに達したこと、つまり対応する受信の実行が開始されたことを示す。送信および受信の両操作がブロッキング操作の場合、同期モードの使用は同期通信を意味する。通信は両方のプロセスが通信位置で通信を取り交わす前にはどちら側も完了しない。このモードで実行される送信はノンローカルである。

- レディ通信モード

対応する受信が既に発行される場合にのみ開始する。それ以外の場合はこの操作はエラーになり、その結果は保証されない。システムによってはレディ通信は他の通信に必要なハンドシェイク操作を無くすことが可能になり、性能改善をもたらす。送信操作の完了は、対応する受信の状態には依存せず、単に送信バッファが再使用できることを意味する。レディモードを使用する通信は、標準送信操作あるいは同期送信操作と意味的には同じである。単に送信側が (対応する受信がすでに発行されていると言う) 追加情報をシステムに提供し、ある種のオーバーヘッドを押えらる言うことに過ぎない。したがって、正しいプログラムにおいては、性能を除くプログラムの動作に影響を与えることなくレディ通信を標準送信で置き換えることが出来る。

これら 3 つの追加的な通信モードに対して、3 つの追加的な送信関数が提供されている。通信モードは 1 文字の接頭語 (B,S,R) であらわされる。B はバッファ、S は同期、R はレディである。

- int MPI\_Bsend(void\* buf, int count, MPI\_Datatype datatype, int dest, int tag, MPI\_Comm comm)

入力	buf	送信バッファの先頭アドレス
入力	count	送信バッファ内の要素数
入力	datatype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター

#### バッファモードの送信

- int MPI\_Ssend(void\* buf, int count, MPI\_Datatype datatype, int dest, int tag, MPI\_Comm comm)

入力	buf	送信バッファの先頭アドレス
入力	count	送信バッファ内の要素数
入力	datatype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター

#### 同期モードの送信

- int MPI\_Rsend(void\* buf, int count, MPI\_Datatype datatype, int dest, int tag, MPI\_Comm comm)

入力	buf	送信バッファの先頭アドレス
入力	count	送信バッファ内の要素数
入力	datatype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター

#### レディモードの送信

受信操作は MPI\_Recv のみであり、どの送信モードにも対応できる。

- int MPI\_Recv(void\* buf, int count, MPI\_Datatype datatype, int source, int tag, MPI\_Comm comm, MPI\_Status \*status)

入力	buf	受信バッファの先頭アドレス
入力	count	受信バッファ内の要素数
入力	datatype	受信バッファの各要素のデータ型
入力	dest	送信元のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター
出力	status	ステータスオブジェクト

受信バッファは、datatype で指定される型の連続した count 個の要素を含む記憶領域によって構成される。この記憶領域の開始アドレスは buf である。受信されるメッセージの長さは、受信バッファの長さ以下でなければならない。オーバーフローエラーは、送られてくるデータが切り詰めなくては受信バッファに収まらない場合に発生する。

受信関数は、メッセージエンベロープの値によってメッセージを選択する。そのエンベロープが受信関数によって指定された source、tag、comm のそれぞれの値と一致する場合にメッセージは受信関数によって受信される。受信側は、source の値としてワイルドカード MPLANY\_SOURCE、および、tag の値として MPLANY\_TAG を指定することが出来、これによって任意の送信元、およびタグが受け入れ可能であることを示す。comm の値にはワイルドカードを指定することが出来ない。このように、メッセージが受信プロセスに送られ、コミュニケーターが一致し、送信元が一致し (source ≠ MPLANY\_SOURCE)、タグが一致 (tag ≠ MPLANY\_TAG) する場合に限り、メッセージは受信関数で受信される。

送信関数と受信関数の次の違いに注意しなければならない：

受信関数では、任意の送信元からメッセージを受け付けるが、送信関数では特定の受信先を指定しなければならない。このことは、データ転送が送信側によってなされる”プッシュ型”通信メカニズムに対応していることを示す (データ転送が通信側によってなされる”プル型”メカニズムとは異なる)。また、送信元と送信先の一致は許される。つまりプロセスがメッセージを自分自信に送ることは可能である。ただし、ブロッキング送受信操作の場合、デッドロックを引き起こす可能性があるため安全ではない。

## バッファ操作

バッファモードで送られるメッセージのバッファリングに使用されるバッファを指定することが出来る。

- int MPI\_Buffer\_attach(void\* buf, int size)

入力 buf バッファの先頭アドレス

入力 size バッファサイズ

送出メッセージのバッファリングに使用するユーザーメモリ内のバッファを MPI に与える。このバッファは、バッファモードで送られるメッセージによってのみ使用される。同時には、ただ 1 つのバッファのみが、1 プロセスに貼り付けられる。

int MPI\_Buffer\_detach(void\*\* buf, int \*size)

出力 buf バッファの先頭アドレス

出力 size バッファサイズ

現在 MPI に付随しているバッファを切り離す。この呼び出しは切り離されたバッファのアドレスとサイズを返す。現在バッファにある全てのメッセージが転送されるまで、この操作はブロックする。この関数が戻った上で、ユーザーはこのバッファにより占められた領域を再利用したり、非アロケートしたり出来る。

MPI は、送信メッセージデータが、送信プロセスにより指定されたバッファ領域へ巡回的な連続領域アロケーションの原則にしたがってバッファリングされるために必要な量のバッファを送出メッセージに対して、提供しなければならない。

MPI は、標準モード送信で実行されるバッファリングについて問い合わせや制御のためのメカニズムを提供しない。そのような情報は、ベンダーによる実装で提供されることが望ましい。

## ノンブロッキング通信のための MPI 関数

多くのシステムにおいて、通信と計算を重ね合わせることによって性能を向上させることが出来る。通信が知的な通信コントローラーによって自動的に実行できるようなシステムにこのことは特に当てはまる。軽負荷スレッドはこのような重ね合わせを達成するための 1 つの機構である。これより良い性能が実現されるもう 1 つの機構としてノンブロッキング通信が挙げられる。ノンブロッキング通信開始が呼ばれると、送信操作が起動されるが、完了はしない。送信バッファからメッセージがコピーされる前に送信開始の呼び出しは戻ってくる。通信を完了させるため、すなわちデータが送信バッファからコピーされたことを確認するためには、別の送信完了の呼び出しが必要となる。適当なハードウェアを使えば、送信が初期化されたが完了はしていない時点で、送信側での計算と並行させて送信側のメモリーからのデータ転送を行うことが出来る可能性がある。同様に、ノンブロッキング受信開始呼び出しは受信作業を起動するが完了はしない。この呼び出しはメッセージが受信バッファに格納される前に戻される。受信操作を完了させ、データが受信バッファに受け取られたことを確認するためには、別の受信完了の呼び出しが必要である。適当なハードウェアを使えば、受信が起動され終了する以前に、受信側のメモリーへのデータ転送を計算と並行させて行える可能性がある。

ノンブロッキング送信開始の呼び出しは、ブロック送信と同じ標準、バッファ、同期、レディの 4 モードを使用することが出来る。これらはブロック通信の時と同じ意味を持っている。レディ以外の全てのモードでは、対応する受信が発行されたかどうかにかかわらず送信を開始することが出来る。ノンブロッキングのレディモードでの送信は対応する受信が発行された時にのみ開始することが出来る。

## 通信の起動

ノンブロッキングモードでの送受信関数は、ブロッキング通信に対応する名前付けの規約を使用する。さらに、ノンブロッキングモードでの関数には I(即座 Immediate) と言う接頭辞が付記される。

- `int MPI_Isend(void* buf, int count, MPI_Datatype datatype, int dest, int tag, MPI_Comm comm, MPI_Request *request)`

入力	<code>buf</code>	送信バッファの先頭アドレス
入力	<code>count</code>	送信バッファ内の要素数
入力	<code>datatype</code>	送信バッファの各要素のデータ型
入力	<code>dest</code>	送信先のランク
入力	<code>tag</code>	メッセージタグ
入力	<code>comm</code>	コミュニケーター
出力	<code>request</code>	通信要求

標準モードのノンブロッキング送信を開始する。

- `int MPI_Ibsend(void* buf, int count, MPI_Datatype datatype, int dest, int tag, MPI_Comm comm, MPI_Request *request)`

入力	buf	送信バッファの先頭アドレス
入力	count	送信バッファ内の要素数
入力	datatype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター
出力	request	通信要求

#### バッファモードのノンブロッキング送信

- `int MPI_Issend(void* buf, int count, MPI_Datatype datatype, int dest, int tag, MPI_Comm comm, MPI_Request *request)`

入力	buf	送信バッファの先頭アドレス
入力	count	送信バッファ内の要素数
入力	datatype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター
出力	request	通信要求

#### 同期モードのノンブロッキング送信

- `int MPI_Irsend(void* buf, int count, MPI_Datatype datatype, int dest, int tag, MPI_Comm comm, MPI_Request *request)`

入力	buf	送信バッファの先頭アドレス
入力	count	送信バッファ内の要素数
入力	datatype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	tag	メッセージタグ
入力	comm	コミュニケーター
出力	request	通信要求

#### レディモードのノンブロッキング送信

受信操作は `MPI_Irecv` のみであり、どの送信モードにも対応できる。

- `int MPI_Irecv(void* buf, int count, MPI_Datatype datatype, int source, int tag, MPI_Comm comm, MPI_Request *request)`

入力	buf	受信バッファの先頭アドレス
入力	count	受信バッファ内の要素数
入力	datatype	受信バッファの各要素のデータ型
入力	dest	送信元のランク
入力	tag	メッセージタグ
入力	comm.	コミュニケーター
出力	request	通信要求

これらの呼び出しは通信要求オブジェクトを割り当て、要求ハンドル(引数 request) と対応付ける。この要求は後で通信の状態を問い合わせたり、その完了を待つために使用することが出来る。ノンブロッキング送信呼び出しは、システムが送信バッファからデータをコピーすることを開始しても良いとすることを意味する。

ノンブロッキング通信についてこれまで説明してきたが、PC クラスターシステムでは通信用に特別な演算装置を持ち合わせていないことから有効な手段として効果的に機能する可能性は低い。しかし、今後の PC アーキテクチャーの改善や大型計算機への移植を考えた場合、通信負荷の減少に充分役割を果たす可能性がある。

## 送受信

送受信操作は、特定の送信先に対するメッセージの送信操作と、他のプロセスからの別のメッセージを受信する動作を 1 つに組み合わせたものである。なお、送信側と受信側の両者は、同一にすることも可能である。この送受信操作は、プロセスの連鎖にしたがって連続したシフト操作を実行する時に大変有効である。ブロック化された送信や受信が、このようなシフト操作に利用される場合、デッドロックを引き起こすような周期的な依存関係を避けるためにも、送信と受信の順序を正しく保つ必要がある。

- int MPI\_Sendrecv(void\* sendbuf, int sendcount, MPI\_Datatype sendtype, int dest, int sendtag, void\* recvbuf, int recvcount, MPI\_Datatype recvtype, int source, int tag, MPI\_Comm comm, MPI\_Status \*status)

入力	sendbuf	送信バッファの先頭アドレス
入力	sendcount	送信バッファ内の要素数
入力	sendtype	送信バッファの各要素のデータ型
入力	dest	送信先のランク
入力	sendtag	送信タグ
出力	recvdbuf	受信バッファの先頭アドレス
入力	recvcount	受信バッファ内の要素数
入力	recvtype	受信バッファの各要素のデータ型
入力	source	送信元のランク
入力	recvtag	受信タグ
入力	comm	コミュニケーター
出力	status	ステータスオブジェクト

上記操作により、ブロック化された送受信操作が実行される。なお、送信と受信の両方で、同じコミュニケーターが使われているが、タグは異なったものにすることが望ましい。ただし、送信バッファと受信バッファは独立したものであり、異なった長さやデータ型を持つことが出来る。

- `int MPI_Sendrecv_replace(void* buf, int count, MPI_Datatype datatype, int dest, int sendtag, int source, int recvtag, MPI_Comm comm, MPI_Status *status)`

入出力	<code>buf</code>	バッファの先頭アドレス
入力	<code>count</code>	バッファ内の要素数
入力	<code>datatype</code>	バッファの各要素のデータ型
入力	<code>dest</code>	送信先のランク
入力	<code>sendtag</code>	送信タグ
入力	<code>source</code>	送信元のランク
入力	<code>recvtag</code>	受信タグ
入力	<code>comm</code>	コミュニケーター
出力	<code>status</code>	ステータスオブジェクト

上記の操作により、ブロック化された送受信が実行される。ここでは、送信操作と受信操作の両方に対して同じバッファが利用される。そのため、送信されたメッセージは受信されたメッセージに置換されてしまう。

送受信操作の機能は、同時に動作している送信を実行するスレッドと受信を実行スレッドの3つが呼び出されるような場合に対応して、これら2つのスレッドを結合した場合に同じような情報を受け取ることである。

## その他参考事項

### マルチプロセス

多くの場合、通信を行う場合に、“ダミー”の送信元と送信先を指定することが有効になる。これによって、非循環型のシフト操作を送受信操作の呼び出しで実行する場合などにおいて、境界を取り扱うためのプログラムを単純にすることが出来る。

呼び出しにおいて、送信元と送信先の引数が必要な場合には、どのような場合があっても特別な値 `MPI_PROC_NULL` をランクの変わりに用いることが可能である。プロセス `MPI_PROC_NULL` による通信では何も実行されない。この場合、`MPI_PROC_NULL` に対する送信は成功し、ただちに戻る。また、`MPI_PROC_NULL` による受信も成功し、受信バッファに対しての修正が行われることなく、ただちに戻る。

### 派生データ型

さまざまな種類のデータ (`int`, `double`, `char`) を一度の通信で転送するために、MPI ではこれらを組み合わせた派生データ型を作成することが出来る。行列やテンソルも同様にして派生データ型として登録できるので非常に便利であるが、ここでは詳細については省略する。

## E.2.11 集団通信

集団通信は、グループ内の全プロセスが同一の引数を持って通信ルーチンを呼び出すことで実行される。集団通信の構文および意味は1対1通信のそれと同じように定義されるので、一般のデータタイプが利用



可能であり、このタイプは送信側プロセスと受信側プロセスとで一致しなければならない。また、集団通信はあるコミュニケーター内でしか行われぬ。しかし、集団通信呼び出しで用いるコミュニケーターは、1対1通信で用いたコミュニケーターと同じものを使用することが出来る。MPI では、集団通信呼び出しで作られたメッセージが1対1通信で作られたメッセージと混同されないことを保証している。

### バリア同期

- `int MPI_Barrier(MPI_Comm comm)`

入力 `comm` コミュニケーター

`MPI_BARRIER` は全グループメンバーが呼び出すまで呼び出し側をブロックする。この呼び出しは、全グループメンバーがその呼び出しを行った後でなければ戻らない。

### ブロードキャスト

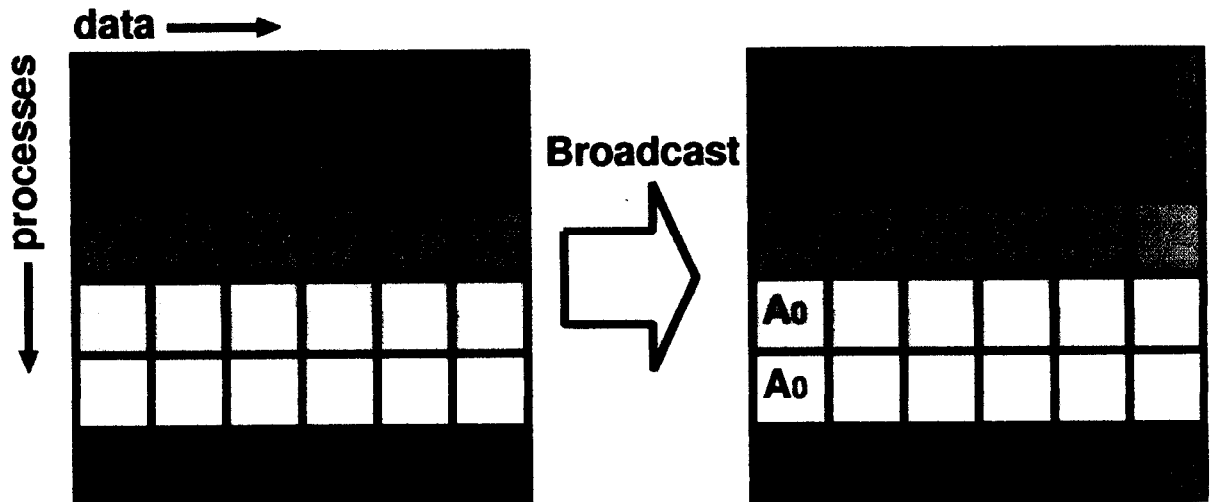


図 E.11: 集団通信によるブロードキャスト

- `int MPI_Bcast(void* buf, int count, MPI_Datatype datatype, int root, MPI_Comm comm)`

入出力	<code>buf</code>	バッファの先頭アドレス
入力	<code>count</code>	バッファ内の要素数
入力	<code>datatype</code>	バッファの各要素のデータ型
入力	<code>root</code>	ルートのランク
入力	<code>comm</code>	コミュニケーター

ルートとは、ブロードキャストするメッセージを保有するプロセスをさしている。`MPI_BCAST` は、ランクが `root` であるプロセスからそのプロセスを含むグループ内の全プロセスへメッセージをブロードキャストする。`comm` および `root` として同じ引数を使って全グループメンバーにより呼び出される。戻った時点で `root` の通信バッファの内容が全プロセスへコピーされている。

## スキッター、ギャザー

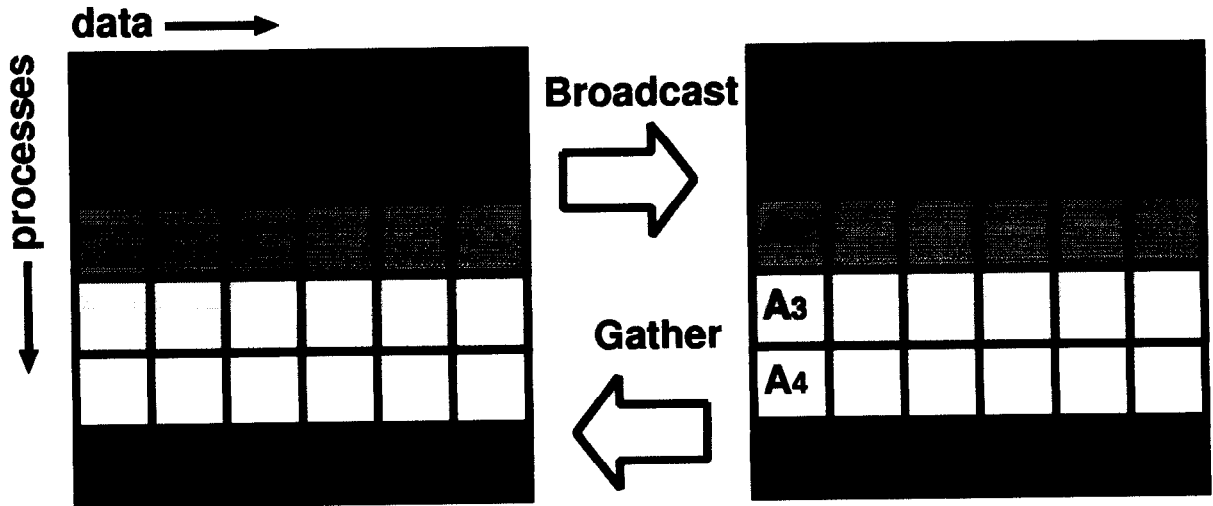


図 E.12: 集団通信を用いたスキッター、ギャザー

- `int MPI_Scatter(void* sendbuf, int sendcount, MPI_Datatype sendtype, void* recvbuf, int recvcount, MPI_Datatype recvtype, int root, MPI_Comm comm)`

入力	<code>sendbuf</code>	送信バッファの先頭アドレス
入力	<code>sendcount</code>	送信バッファ内の要素数
入力	<code>sendtype</code>	送信バッファの各要素のデータ型
出力	<code>recvbuf</code>	受信バッファの先頭アドレス
入力	<code>recvcount</code>	受信バッファ内の要素数
入力	<code>recvtype</code>	受信バッファの各要素のデータ型
入力	<code>root</code>	ルートのランク
入力	<code>comm</code>	コミュニケーター

`MPI_SCATTER` は、ルートが `MPI_Send()` によってメッセージを送ることであるとも言い換えられる。このメッセージは  $n$  個の等しいセグメントに分割されて  $i$  番目のセグメントが  $i$  番目のプロセスに送られる。

- `int MPI_Gather(void* sendbuf, int sendcount, MPI_Datatype sendtype, void* recvbuf, int recvcount, MPI_Datatype recvtype, int root, MPI_Comm comm)`

入力	sendbuf	送信バッファの先頭アドレス
入力	sendcount	送信バッファ内の要素数
入力	sendtype	送信バッファの各要素のデータ型
出力	recvbuf	受信バッファの先頭アドレス
入力	recvcount	受信バッファ内の要素数
入力	recvtype	受信バッファの各要素のデータ型
入力	root	ルートのランク
入力	comm	コミュニケーター

各プロセス (ルートプロセスを含む) は、送信バッファの内容をルートプロセスへ送信する。ルートプロセスはメッセージを受信し、ランクの順番に格納する。

その他の集団通信

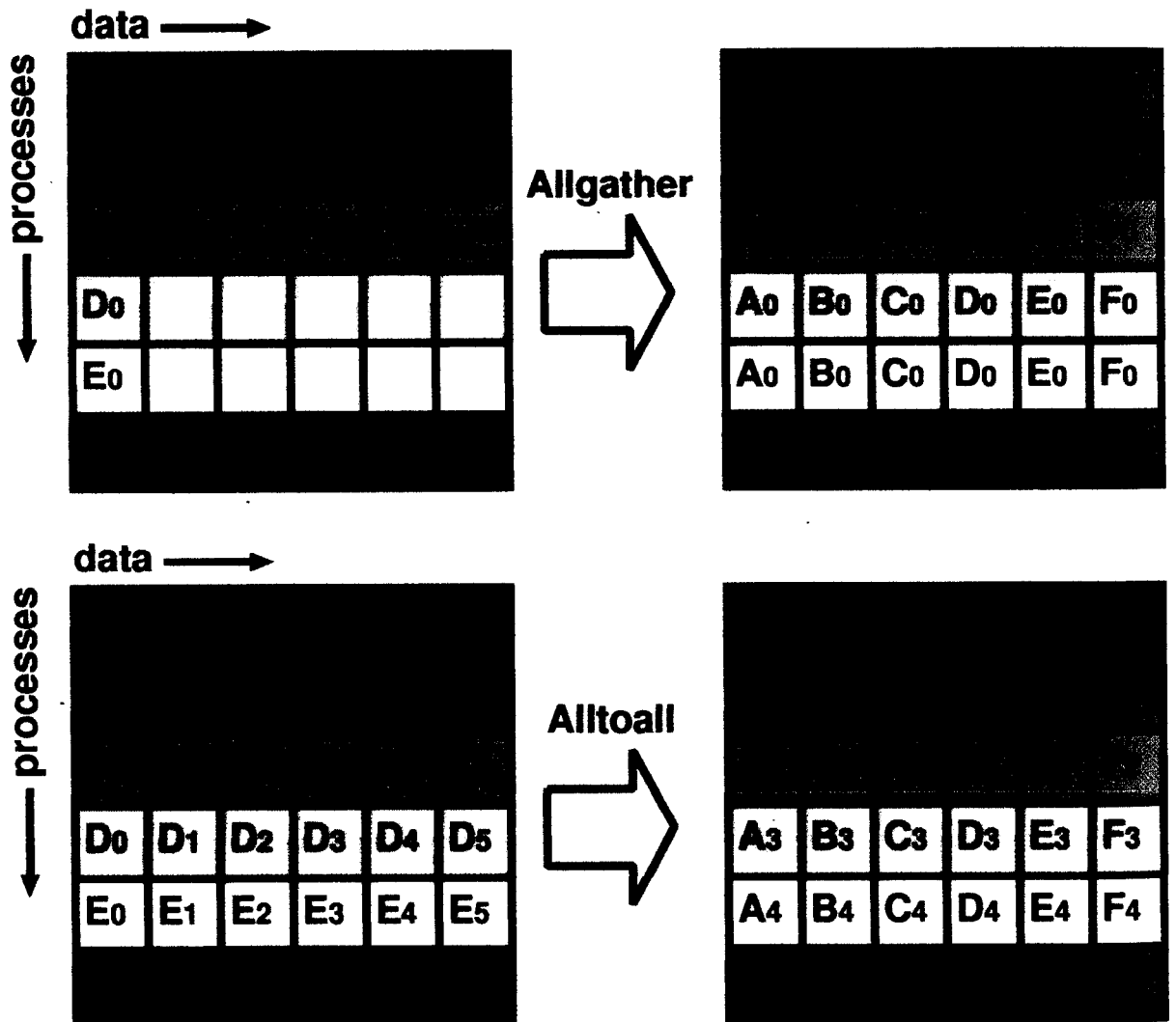


図 E.13: 集団通信を用いた全対全スキャッタ/ギャザー

MPI では、ブロードキャスト、スキッター、ギャザーの他に、全プロセスへのギャザー (MPIALLGATHER)、全対全スキッタ/ギャザー (MPIALLTOALL) が用意されている。これらの機能を簡単に図示すると図 E.13 のようになる。MPIALLGATHER は、ルートプロセスの代わりに全てのプロセスが結果を受け取る MPIGATHER と考えて良い。j 番目のプロセスから送信されるデータのブロックが全てのプロセスで受信され、バッファの第 j 番目のブロックとして格納される。

MPIALLTOALL は、MPIALLGATHER の機能を、各プロセスが異なるデータを受信プロセスに送ることの出来るように拡張したものである。プロセス i から送られる j 番目のブロックは、プロセス j の受信バッファの i 番目のブロックに格納される。

### 大域的なリダクション操作

MPI は、各プロセス間で良く使用される大域的なリデュース操作 (sum, max, ...) を標準で提供する。また、標準で組み込まれている操作以外でもユーザーがオペレーションを定義することで使用可能である。大域的なリダクション関数にはいくつかバリエーションがある。ルートプロセスに結果を返すリデュース、全プロセスに結果を返すオールリデュース、およびスキャン操作である。

### リデュース

```
int MPI_Reduce(void* sendbuf, void* recvbuf, int count, MPI_Datatype datatype, MPI_Op op, int root, MPI_Comm comm)
```

入力	sendbuf	送信バッファの先頭アドレス
出力	recvbuf	受信バッファの先頭アドレス
入力	count	受信バッファ内の要素数
入力	datatype	受信バッファの各要素のデータ型
入力	op	演算
入力	root	ルートのランク
入力	comm	コミュニケーター

MPIREDUCE は、演算 op を使用してグループ内の各プロセスの入力バッファの中に与えられる要素に対して通信を行い、その結果をランク root のプロセスの出力バッファの中に返す。演算 op として利用できる定義済のリデュース操作は次の通りである。

表 E.3: 大域的なリダクション操作の種類

op	機能	op	機能
MPI_MAX	最大値	MPI_MIN	最小値
MPI_SUM	和	MPI_PROD	積
MPI_LAND	論理積	MPI_BAND	ビット演算の積
MPI_LOR	論理和	MPI_BOR	ビット演算の和
MPI_LXOR	排他的論理和	MPI_BXOR	ビット演算の排他的論理和
MPI_MAXLOC	最大値と位置	MPI_MINLOC	最小値と位置

これらの詳細とユーザー定義演算に関してはここでは省略する。

## オールリデュース

MPI では、演算結果がグループの全てのプロセスへ返されるリデュース操作のバリエーションが用意されている。MPI では、これらの操作に関わる全てのプロセスが同一の結果を受け取ることが要求されている。

- `int MPI_Allreduce(void* sendbuf, void* recvbuf, int count, MPI_Datatype datatype, MPI_Op op, MPI_Comm comm)`

入力	<code>sendbuf</code>	送信バッファの先頭アドレス
出力	<code>recvbuf</code>	受信バッファの先頭アドレス
入力	<code>count</code>	受信バッファ内の要素数
入力	<code>datatype</code>	受信バッファの各要素のデータ型
入力	<code>op</code>	演算
入力	<code>comm</code>	コミュニケーター

結果が全グループメンバーの受信バッファに格納される点を除き、`MPI_REDUCE` と機能は同じである。

## リデューススキャッター通信

MPI では、リデュース演算のバリエーションが用意されている。そこでは、戻る際に演算結果がグループ内の全プロセスにスキャッターされる。

- `int MPI_Reduce_scatter(void* sendbuf, void* recvbuf, int* recvcount, MPI_Datatype datatype, MPI_Op op, MPI_Comm comm)`

入力	<code>sendbuf</code>	送信バッファの先頭アドレス
出力	<code>recvbuf</code>	受信バッファの先頭アドレス
入力	<code>recvcount</code>	各々のプロセスに分配される結果の要素数
入力	<code>datatype</code>	受信バッファの各要素のデータ型
入力	<code>op</code>	演算
入力	<code>comm</code>	コミュニケーター

`MPI_REDUCE_SCATTER` は、まず、`sendbuf`、`count`、および `datatype` で定義される送信バッファ内の  $count = \sum_i recvcounts[i]$  個の要素からなるベクトルに対し要素毎のリダクションを実行する。次に、得られた結果のベクトルを  $n$  個の非連結セグメントに分割する。ここで  $n$  はグループ内の総メンバー数である。セグメント  $i$  には、`recvcounts[i]` 個の要素が入る。第  $i$  セグメントがプロセス  $i$  に送信され、`recvbuf`、`recvcounts[i]`、および `datatype` で定義される受信バッファの中に格納される。

## スキャン

- `int MPI_Scan(void* sendbuf, void* recvbuf, int count, MPI_Datatype datatype, MPI_Op op, MPI_Comm comm)`

入力	sendbuf	送信バッファの先頭アドレス
出力	recvbuf	受信バッファの先頭アドレス
入力	count	入力バッファの要素数
入力	datatype	受信バッファの各要素のデータ型
入力	op	演算
入力	comm	コミュニケーター

MPLSCAN は、グループ中に分配されたデータに対しプレフィックスリダクションを実行する場合に使用する。この演算は、ランク  $i$  のプロセスの受信バッファの中に、ランク  $0, 1, \dots, i$  までのプロセスの送信バッファ中の値のリダクションを返す。サポートされている演算の種類や意味、および送信、受信バッファに対する制約は MPLREDUCE の場合と同様である。

### E.2.12 実装時における注意

MPI を用いて並列計算を行う場合、主としてカルテシアントポロジーを用いて計算空間の領域分割を行うことにより並列化を達成することが多い。その場合、計算空間の分割方法により計算ステップあたりに生じる情報転送量に変動が生じるため、可能な限り情報転送量の少ない領域分割手法をとることが並列計算の高速化につながる。

ノンブロッキング通信を行うことで通信量の大小に関わらず高速な数値計算を行うことは可能であるが通信時間の見積りと、更新順序の最適化を行う必要があるため、通常はブロッキング通信を行うことが考えられるので、情報通信量を少なくする領域分割手法について簡単に述べる。

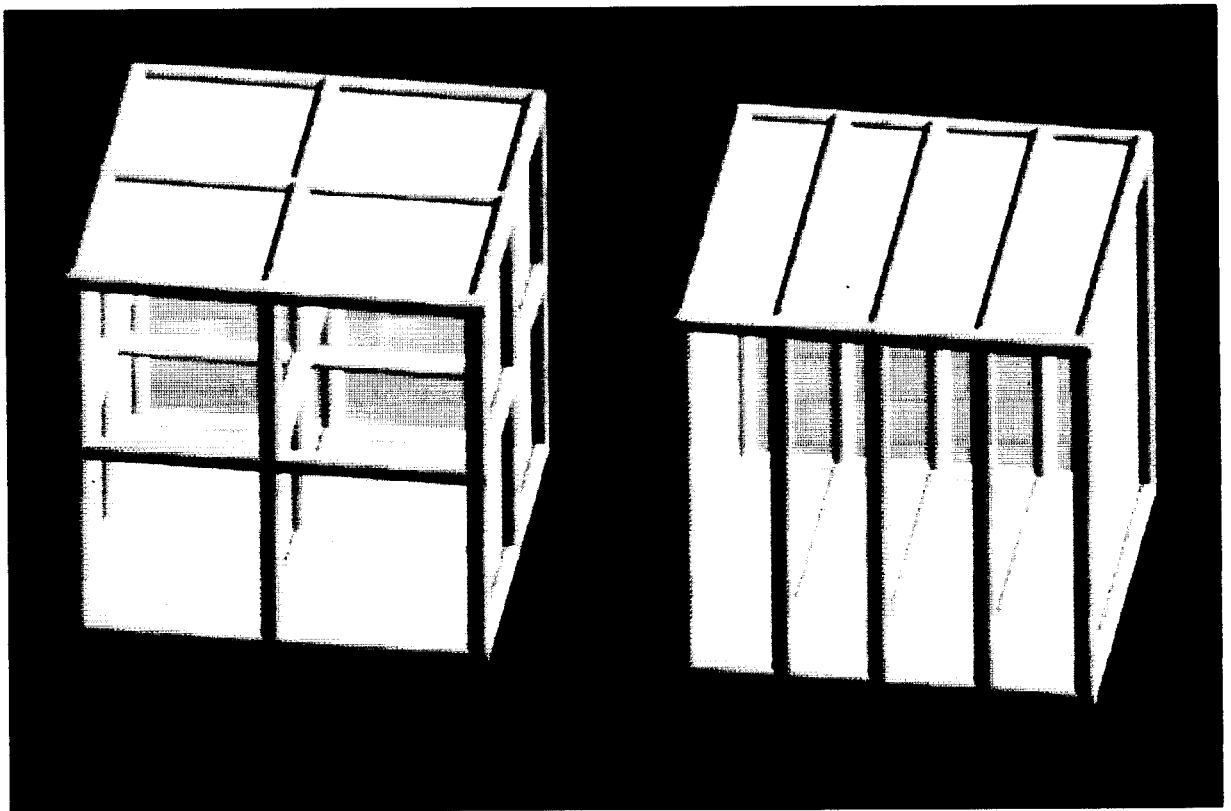


図 E.14: 3次元計算領域の分割例

分割した際に発生する断面積を減少させることで通信量を減らすことが出来る。

3次元計算領域の単純な領域分割例として図 E.14 に示す分割方法について考える。計算ステップ毎に生じる情報量は単純に分割された領域の表面積に比例することが容易に理解できる。計算領域の1辺の長さを  $L$  とすると、図 E.14 左図の計算通信量は切断面が  $3L^2$  あり、周期境界条件を考えると総面積は  $11L^2$  となる。その一方で右図では領域を4分割しているので総面積は  $11L^2$  となるが、左図と同様に8分割すると  $15L^2$  となるため、通信量が  $4L^2$  面積分多くなることになる。通信用の計算手順を単純化するためには右図のような領域分割を用いると非常に簡単にプログラムを作成することが出来るが、並列計算の高速化といった観点からは適していないことは明らかである。

次に計算負荷の均等分配について考える。一般的に PC クラスターを作成する場合、各計算機の演算能力は等しいものと思われる。そのため、計算領域の分割を行う際にその体積を等しくすることが並列計算の高速化に寄与すると考えられる。計算領域中で著しく他の領域に比べて計算負荷の高い領域を含む特殊な例を除くと、例えば陽解法による流体計算を行う場合では全ての計算領域において同等の計算負荷が各計算領域で必要となる。この様な場合図 E.15 右図に示すように不均一な計算領域分割を行うと、右図と左図の計算領域分割手法で情報の転送量は等しいにも関わらず右図の分割法では左図の分割法に比べて充分なスケーラビリティが得られない。

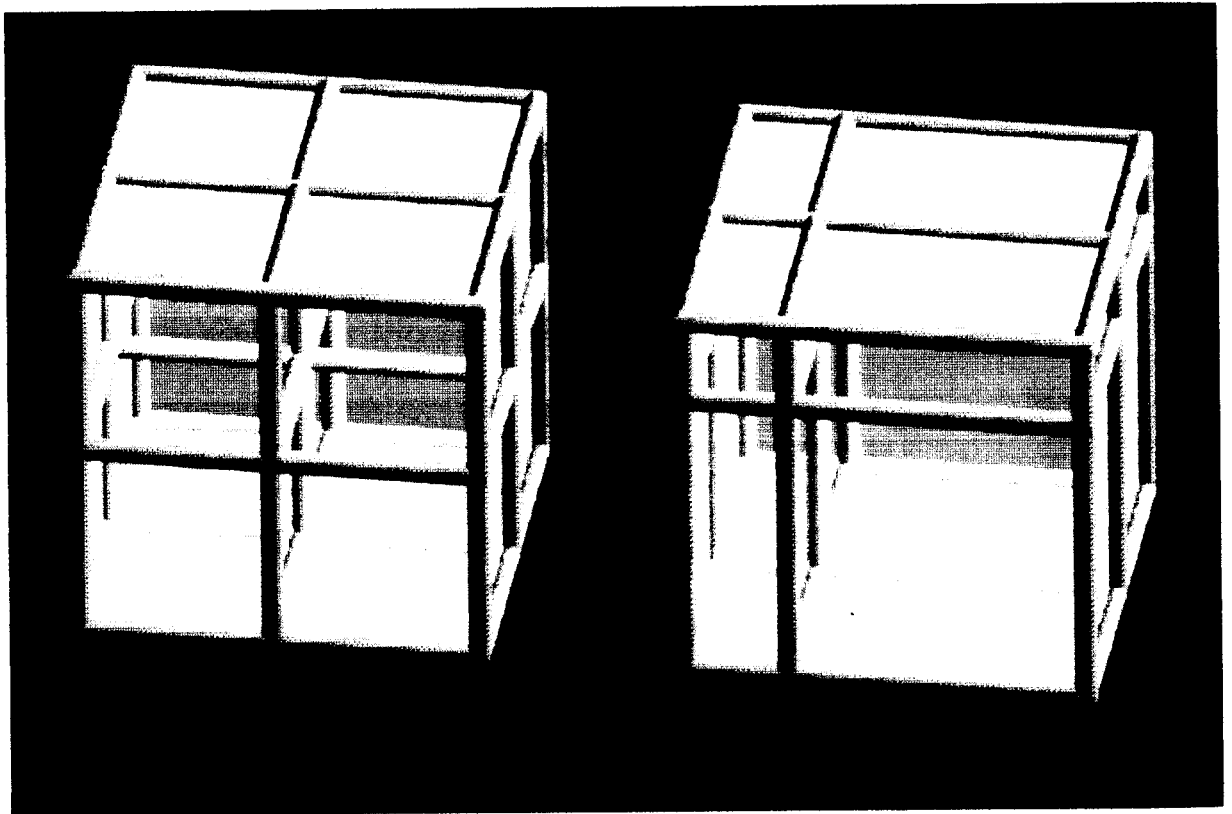


図 E.15: 不均一な計算領域分割を行うことである計算機のみが負荷がかかる問題が生じるため、期待される並列度 (スケーラビリティ) を得ることが出来ない。

計算領域を不均一に分割することで各計算ノードの負荷に差が生じてしまう。その結果 図 E.16 に示すようにある計算ノードのみが計算していて、他の計算ノードは通信待ち状態になってしまうため、計算機能力を十分に発揮できないことにその問題がある。そこで、左図に示すように均一に計算領域分割を行い、図 E.17 に示すように全計算ノードの演算が同時に終了し、一斉に通信を開始することが並列計算の高速化に望ましいと言える。

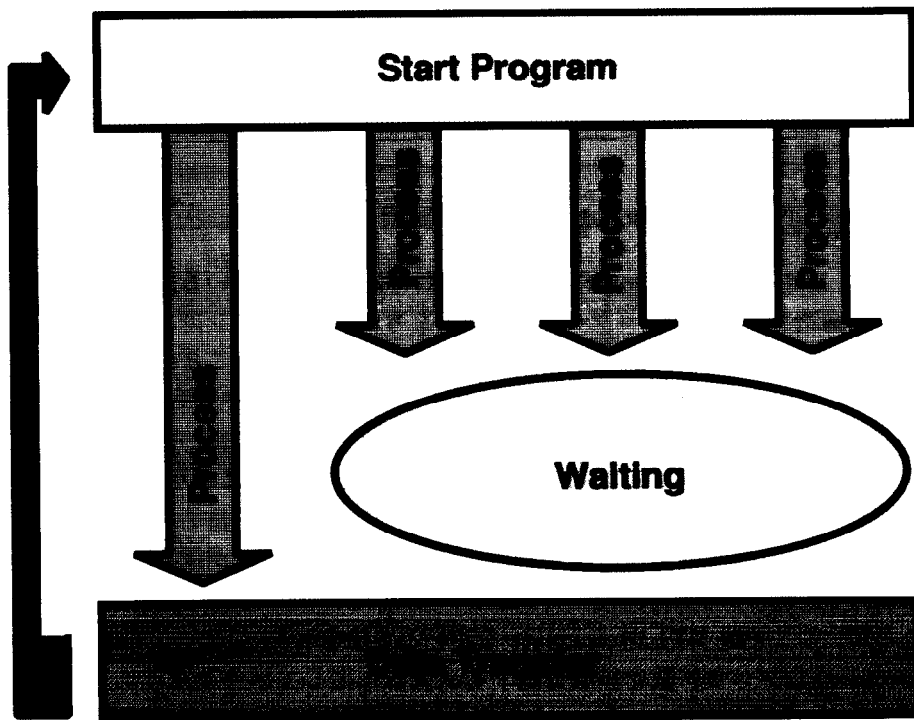


図 E.16: 計算負荷の不均一による低速化

ある計算ノードが他の計算ノードに比べ計算負荷が高い場合、他の計算ノードの能力が無駄になるため期待された計算速度を得ることが出来ない。

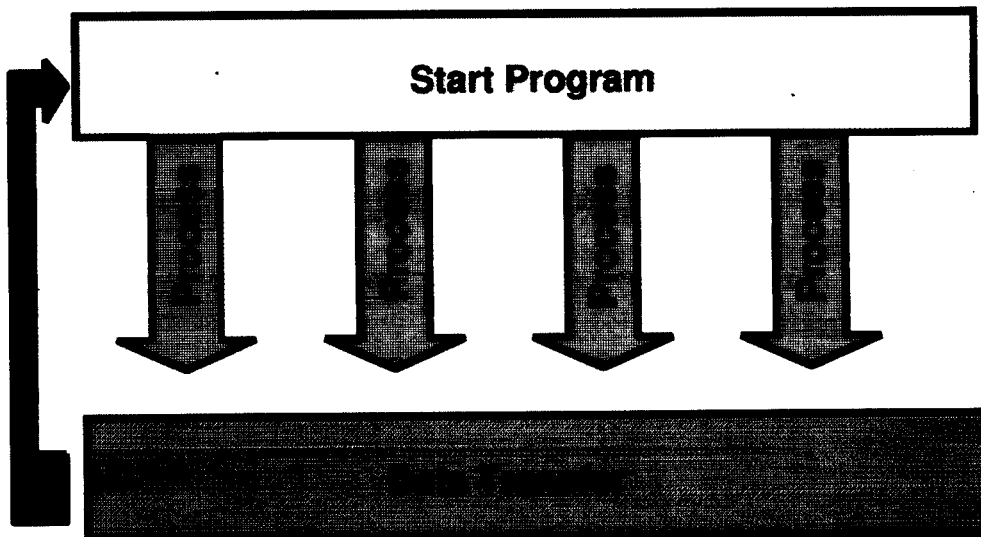


図 E.17: 計算負荷の均一化

計算負荷を全ての計算ノードに対して均一化することで効率の良い並列計算が達成される。



## E.3 実装例

MPI (Message Passing Interface) ライブラリを実装し、並列計算を行うための手続きを具体的に示す。緩和計算の例題として 3 次元ポアソン方程式を扱い、また陽解法の計算手続きの例として 3 次元流体計算を考える。

### E.3.1 計算空間へのカルテシアン構造の導入

IDO (Interpolated Differential Operator) 法は主に直行等間隔格子を用いることから、通信を簡単化するためカルテシアン構造を持つコミュニケーターの利用を考える。カルテシアン構造を持たせることにより、計算空間の分割を簡便化でき、また周期境界条件の導入を容易にすることが出来る。

計算領域にカルテシアン構造を持ったコミュニケーターを適用することで、図 E.6 に示したようにグリッドの特性を活かした領域分割を行うことが出来る。また、隣接する分割領域を所有する計算ノードの rank を実装者が設計時に指定する必要は無く、必要に応じて `MPI_Cart_rank` により得ることが出来る。

以上の特性を活かし、カルテシアン構造を採り入れた計算空間の導入方法を示す。まず始めに領域分割方法について考える。図 E.15 に示したように均一に領域分割を行うことが並列計算を高速化させることに一般的に有効である。計算格子点数に応じて実装者が各軸方向に対する領域分割数を決めることは当然有効な手段であるが、ここでは `MPI_Dims_create` を用いる手法を示す。

`MPI_Dims_create` は全プロセス数を領域分割する次元数に応じて出来るだけ均等に各次元方向にプロセスを分配する。全プロセス数はコミュニケーター支援関数の一つである `MPI_Comm_size` により、`MPI_COMM_WORLD` に対して

```
MPI_Comm_size(MPI_COMM_WORLD, &allproc);
```

と呼び出すことで `allproc` に返される。この値を利用し、

```
int dims[3];
MPI_Dims_create(allproc,3,dims);
```

として呼び出すことで、全プロセスを 3 次元方向に分配した次元情報 `dims` を得ることが出来る。カルテシアン構造を持つコミュニケーターは `MPI_COMM_WORLD` から生成されるものとして

```
int periods[3];
MPI_Cart_create(MPI_COMM_WORLD,3,dims,periods,FALSE,&Cart_COMM);
```

として生成する。ここで `periods` は、各軸方向に対して周期境界条件を適用する (`true`) か否 (`false`) かを与える (図 E.18)。

たとえば、`periods[0] = true;` とした場合、 $x$  方向に関して周期境界条件が適用される。全方向に対して周期境界条件を適用する場合は併せて `periods[1]=periods[2]=true;` とする必要がある。ただし、参照関係に関してのみ周期境界条件が適用される。つまり、通信の実装を行うことで始めて周期境界条件が達成される。

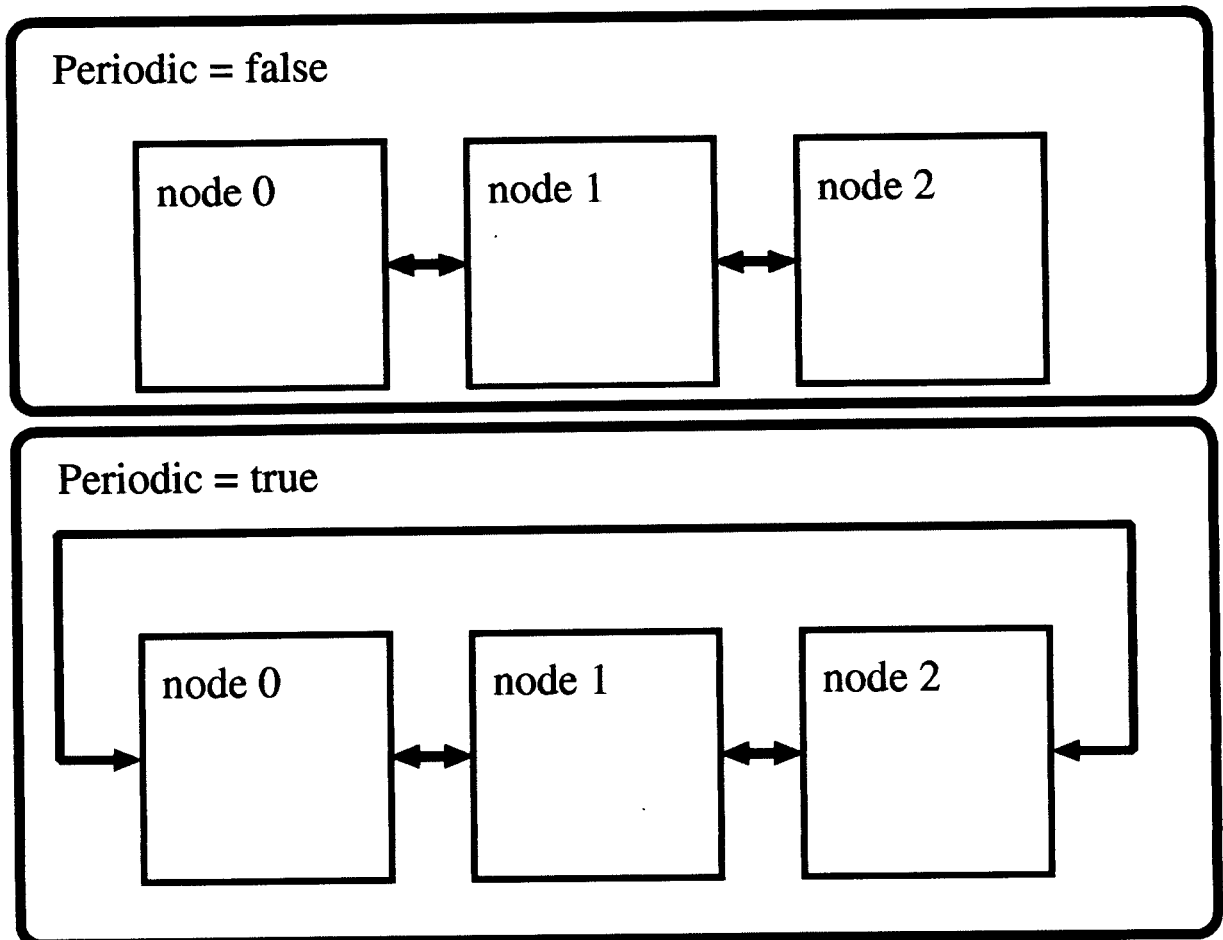


図 E.18: カルテシアン構造の周期境界条件

カルテシアン構造生成時に周期境界条件であることを明示的に与えると、参照関係が周期的になる。

### E.3.2 並列計算のための通信の実装

ノンブロッキング通信を行うことで計算と平行した通信の実装が可能であり、それにより並列計算の高速化を行うことが可能である。しかし、ノンブロッキング通信による高速化は一般的に並列計算機のアーキテクチャに依存すると考えられる。通信を行うための cpu コストを考えると、通信専用の cpu を持つシステムを用いない限り、そのコストを削減することは出来ない。ただし、近年では完全に cpu がその通信のための計算を行うものではなく、チップセットと呼ばれるネットワークカード上の演算系により処理されるものが増えて来た。そういった理由から、ノンブロッキング通信を実装することにも十分な意味があると考えられるが、計算の安全性を確保することを第一条件として考えると、ノンブロッキング通信の実装は困難を伴うことが容易に予想できる。このような理由から、ここではブロッキング通信を実装することを考える。

また、通信を実装する際にはデッドロックに注意する必要がある。MPI.Send()、MPI.Recv() を用いた場合、通信の手順を吟味しなければデッドロックを引き起こす可能性があることから、MPI.Sendrecv\_replace() を用いた実装を行った。通信の順序は各々の自由で設定できるが、ここでは単純に  $x, y, z$  方向の順に通信を行う。

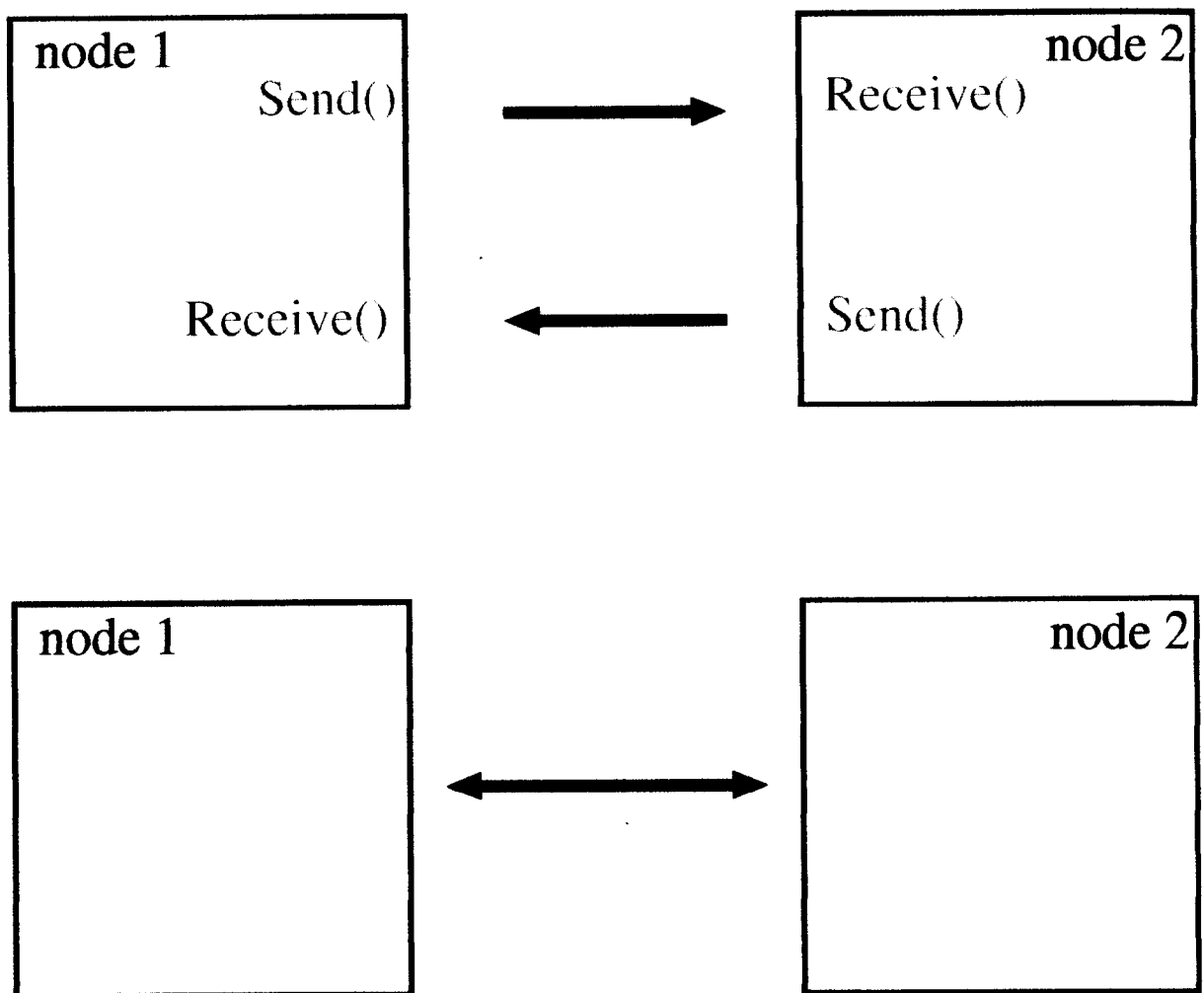


図 E.19: SendRecv() を用いた通信

同じ型の変数を同数双方向通信を行う場合、SendRecv() 関数を用いることで、プログラムが簡単になる。

MPLPACKED や、新しいデータタイプを宣言することで多次元配列による通信を行うことが出来るが、プログラムの簡素化のため一旦通信する情報を 1 次元配列に格納し、データ転送を行う。以上のポリシーにより通信アルゴリズムは図 E.20 に示す手順をとる。

はじめにカルテシアン構造を持つコミュニケーターに対する支援関数として準備されている MPI\_Cart\_rank() を用い、各計算ノードの通信相手の rank を得る。次に通信領域のデータを 1 次元配列に格納し、SendRecv() によりその情報の交換を行う。通信により得られた情報を的確な場所に格納することで各計算ノードから見た境界条件の更新を行うことが出来る。この一連の操作を  $x, y, z$  軸方向に対して行うことで通信は完了する。

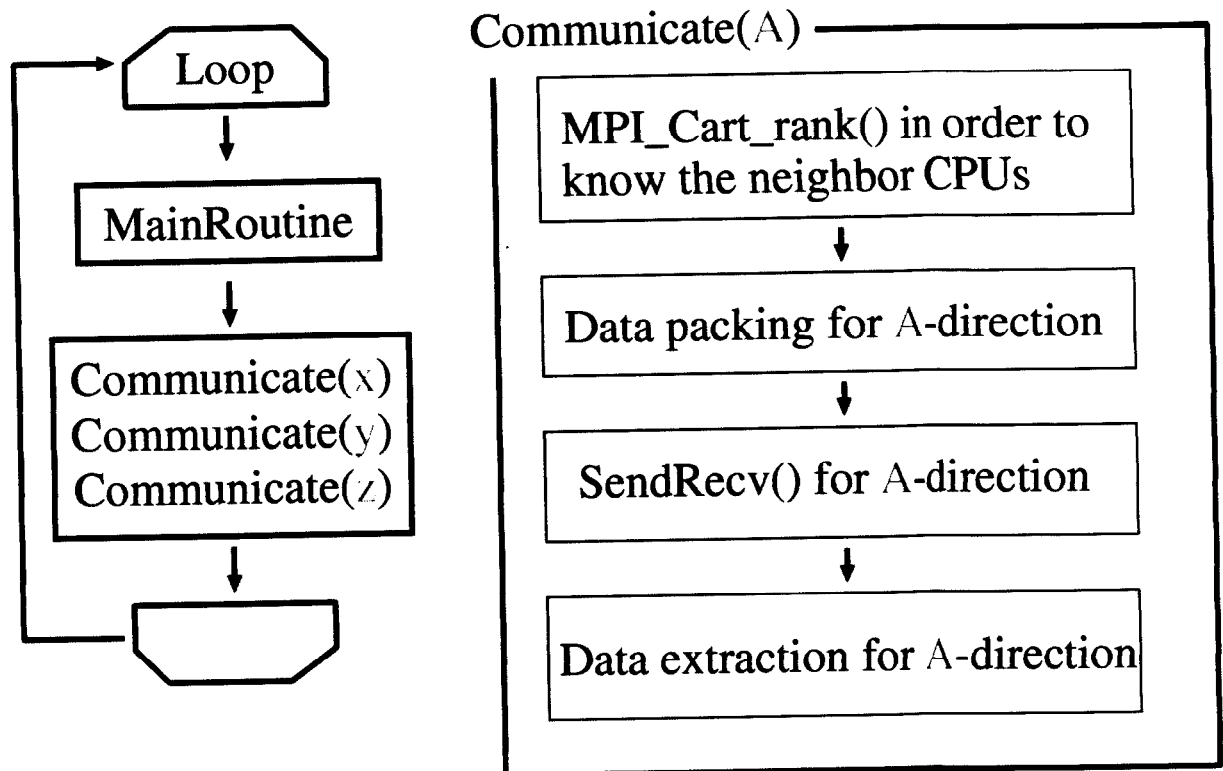


図 E.20: MPI によるカルテシアントポロジーを用いた通信アルゴリズム  
カルテシアン構造を用いることで通信相手を容易に検索することができ、また SendRecv() 関数を用いることでプログラムを簡略化できる。

### E.3.3 ポアソン方程式の並列解法

緩和計算への適用例として 3 次元ポアソン方程式の並列計算手順を示す。緩和計算を行う際に生じる特徴的な問題として、まず各格子点上での情報の依存関係が挙げられる。これは 4.1.2 節で示したようにレッドブラックアルゴリズムを適用することで回避することが可能である。つぎにその収束判定が考えられる。並列計算を行うことにより、各ノードでの収束誤差を見積もることは容易に可能であるが、計算領域全体での収束判定を行うためには全ノードとの通信が必要となる。このアルゴリズムを集団通信の支援関数 MPI\_Allreduce() を用いることで行う。その際の演算は MPLMAX ないしは MPLSUM となることが予想されるが、本論文では MPLSUM を用いた。

計算の流れを図 E.21 に示す。MainRoutine において、レッドブラックアルゴリズムを適用した緩和法により緩和ステップ内の更新ならびに各ノードが持つ収束誤差の総和演算を行う。各ノードの境界情報を近傍の計算領域を所有するノードとの通信により更新を行い、MPI ライブラリによって提供されている集団通信支援関数の一つである MPI\_Allreduce() を用いることで各ノードが持つ誤差の総和を各ノードに対して伝達することにより全てのノードに計算空間全体の収束誤差を与える。得られた空間全体の収束誤差により計算の収束判定を行い、全体のループの条件分岐を行う。

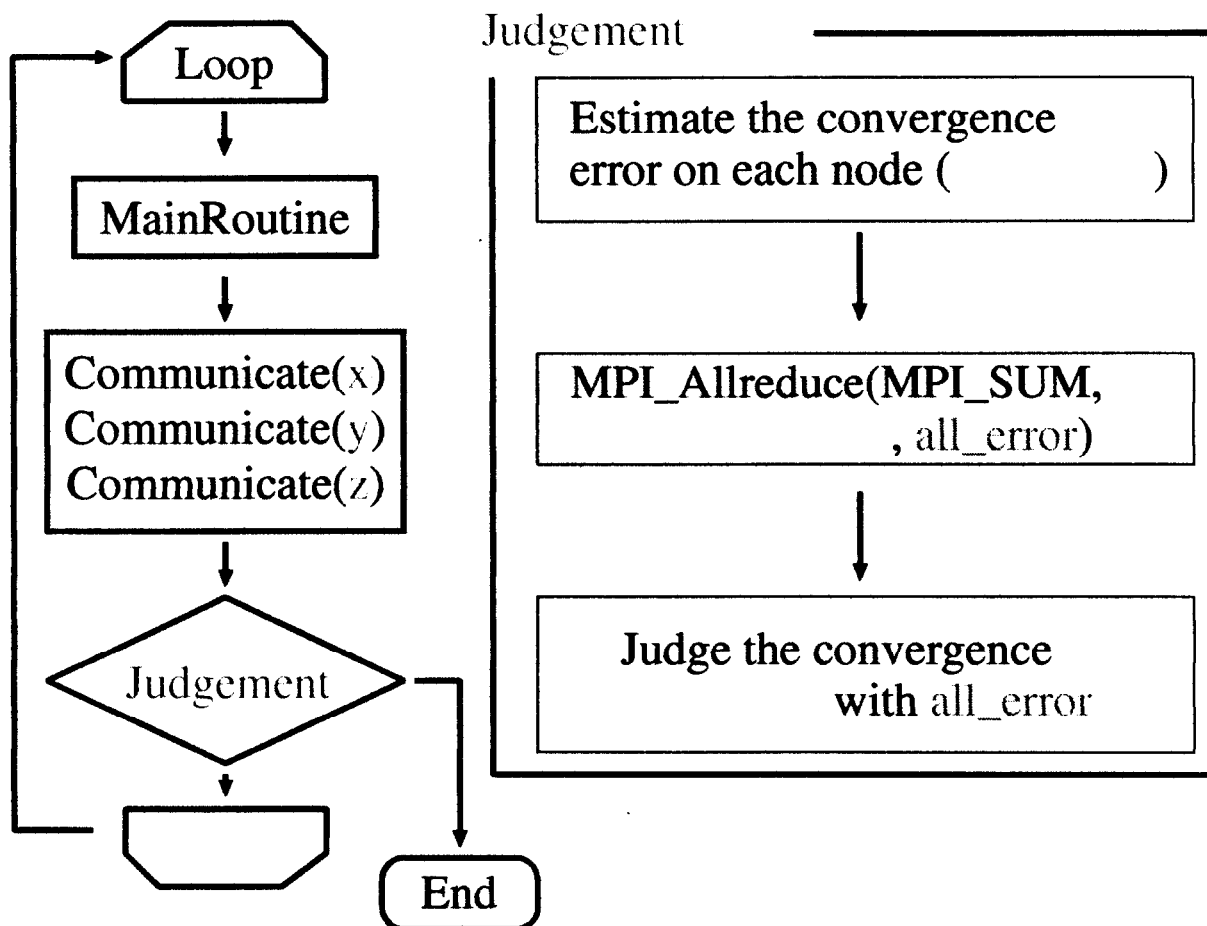


図 E.21: ポアソン方程式の並列解法における計算の流れ

収束判定を行うため、全ノードが持つ誤差の和をとる必要がある。その作業を MPI の集団通信支援関数の一つである `MPI_Allreduce()` を用いて行う。

#### E.3.4 流体計算の並列解法

陽解法による圧縮性流体問題の解法を考える場合、その計算手続きは図 E.20 に示すものと一致する。また、陰解法化や非圧縮性流体問題への適用を考えた場合、図 E.21 に示す計算手順を踏まえることで対処できる。

このように、直交等間隔計算格子点を用いる場合はその性質を良く反映しているカルテシアントポロジーを実装したコミュニケーターならびにその支援関数を用いることで比較的容易に空間分割による並列計算のためのアルゴリズムを実装することが出来る。通信の安全性を強調する場合は `MPI_Barrier()` を使い、全プロセスの完全な同期をとることで行うことが出来る。また、通信の高速化を求める場合はノンブロッキング通信を行い、計算中に通信を行うことで通信によるスケーラビリティ低減を抑制することが出来る。

# 参考文献

---

## 第 1 章 序論

---

- [1] H. Takewaki, A. Nishiguchi, and T. Yabe (1985) Cubic interpolated pseudo-particle (CIP) method for solving hyperbolic-type equations; J. Comput. Phys., vol. 60 pp.261-268
- [2] T. Yabe and E. Takei (1988) A new higher-order Godunov method for general hyperbolic equation; J. Phys. Soc. Jpn., vol.57 pp.2598-2601
- [3] T. Aoki(1996) Interpolated Differential Operator (IDO) scheme for solving partial differential equations; Comp.Phys.Comm.,vol.102,pp.132-146
- [4] T.Yabe and T.Aoki(1991) A universal solver for hyperbolic equations by cubic-interpolation - I. One-dimensional solver; Comp. Phys. Comm. ,vol.66 pp.219-232
- [5] T.Yabe,T.Ishiwaka,P.Y.Wang,T.Aoki,Y.Kadota and F.Ikeda(1991) A universal solver for hyperbolic equations by cubic-interpolation-II. Two- and three-dimensional solver; Comp.Phys.Comm.,vol.66 pp.233-242
- [6] T. Yabe, P. Y. Wang (1991) Unified Numerical Procedure for Compressible and Incompressible Fluid; J. Phys. Soc. Jpn., vol.60, pp.2105-2108
- [7] H. Takewaki and T. Yabe (1987) The Cubic-Interpolated Pseudo Particle (CIP) Method: Application to Nonlinear and Multi-dimensional Hyperbolic Equations; J. Comput. Phys., vol. 70, pp.355-372
- [8] T. Yabe, T. Ishikawa, Y. Kadota and F. Ikeda (1990) A Multidimensional Cubic-Interpolated Pseudoparticle (CIP) Method without Time Splitting Technique for Hyperbolic Equations; J. of the Phys. Soc. of Japan, vol.59, no.7, pp.2301-2304
- [9] T. Yabe, T. Aoki, G. Sakaguchi, P. Y. Wang and T. Ishikawa (1991) The Compact CIP (Cubic-Interpolated Pseudo-Particle) Method as a General Hyperbolic Solver; J. Phys. Soc. Jpn., vol. 59, pp.2301-2304
- [10] T. Aoki (1998) Development of High Accurate CFD Solver with Boundaries given by CAD Geometry Data (DXF); 第 12 回 数値流体力学シンポジウム講演論文集, pp. 321-322

---

## 第 2 章 IDO 法の計算精度の評価

---

- [11] T. Utsumi, T. Kunugi, T. Aoki (1997) Stability and accuracy of the Cubic Interpolated propagation scheme; J. Comput. Phys. Comm., vol.101, pp.9-20
- [12] S.T.Zalesak(1979) Fully multidimension flux-corrected transport algorithms for fluids; J.Comput. Phys.,vol.31,pp 335-362

---

 第 3 章 圧縮性流体問題への適用
 

---

 1 次元衝撃波管問題
 

---

- [13] J. von Neumann and R. D. Richtmyer (1983) A Method for the Numerical Calculation of Hydrodynamic Shocks; J. Appl. Phys. vol. 21, pp.232-237

 1 次元乱流 Burgers 方程式
 

---

- [14] S. N. GURBATOV, S. I. SIMDYANKIN, E. AURELL, U. FRISCH and G. TÓTH (1997) On the decay of Burgers turbulence; J. Fluid Mech., vol.344, pp.339-374
- [15] H. P. Breuer and F. Petruccione (1992) Burgers's turbulence model as a stochastic dynamical system: Master equation and simulation; Physical Rev. E, vol.47, no.3, pp.1803-1814
- [16] Kunio Kuwahara (1999) UNSTEADY FLOW SIMULATION AND ITS VISUALIZATION; AIAA, 99-3045

 前線発達問題への適用
 

---

- [17] Doswell, C. A., II (1984) A Kinematic Analysis of Frontogenesis Associated with a Nondivergent Vortex; J. Atmos. Sci, vol 41, pp 1242-1248
- [18] M. Rančić (1991) Semi-Lagrangian Piecewise Biparabolic Scheme for Two-Dimensional Horizontal Advection of a Passive Scalar; Month. Weather Rev. vol. 120, pp 1394-1406

 レーリー・テラー不安定性への適用
 

---

- [19] S. Chandrasekhar (1961) dynamic and Hydromagnetic Stability; Oxford University Press
- [20] H. J. KULL and R. C. ARNOLD (1989) Rayleigh-Taylor Instability: Design of an Experiment for a Heavy - Ion - Heated Plasma at GSI; Nuc. Inst. and Methods in Phys. Research, A278, pp. 135 - 137
- [21] D. Shavarts *et. al.* (2000) Scaling Laws of Nonlinear Rayleigh-Taylor and Richtmyer-Meshkov Instabilities in Two and Three Dimensions; C. R. Acad. Sci. Paris, t. 1, Série IV, pp.719-726
- [22] T. Yabe, H. Hoshino and T. Tsuchiya (1991) Two- and three-dimensional behavior of Rayleigh-Taylor and Kelvin-Helmholtz instabilities; Phys. Rev. A, vol. 44, num. 4, pp. 2756-2758

 超音速混合流問題への適用
 

---

- [23] 佐竹常久 (1999) 超音速自由渦型空力レーザー窓の高性能化に関する研究; 博士論文
- [24] N. D. Sandham and W. C. Reynolds (1991) Three-dimensional Simulations of Large Eddies in the Compressible Mixing Layer; J. Fluid Mech., vol. 224, pp. 133-158
- [25] C. Staquet (2000) Mixing in a Stably Stratified Shear Layer: Two- and Three-dimensional Numerical Experiments; Fluid Dynamics Research, vol.27, pp.367-404
- [26] 松尾 正毅、桜井 浩一、森瀬 隆司、佐竹 常久、中島 充夫、堀岡 一彦, 超音速平交流の混合層に誘起される微小衝撃波列;

- [27] R. Zhang, X. He, G. Doolen, S. Chen (2001) Surface tension effects on Two-dimensional Two-phase Kelvin-Helmholtz Instabilities; *Advances in Water Resources*, vol.24, pp.46-478
- [28] W. raig, M. D. Groves (2000) Normal Forms for Wave Motion in Fluid Interfaces; *Wave Motion*, vol.31, pp.21-41
- [29] K. Schneider, M. Farge (2000) Numerical Simulation of a Mixing layer in an Adaptive Wavelet Basis; *C. R. Acad. Sci. Paris*, t. 328, Série II b, pp.263-269
- [30] S. Chandrasekhar (1961) *Hydrodynamic and Hydromagnetic Stability*; Oxford University Press, ISBN 0-486-64071-X
- [31] Garry L. Brown and Anatol Roshko (1974) On Density Effects and Large Structure in Turbulent Mixing Layers; *J. Fluid Mech.*, vol. 64, part 4, pp. 775-816

---

#### 第 4 章 非圧縮性流体解法の高速化

---

##### ポアソン方程式解法の高速度化

---

- [32] K. Sakurai, T. Aoki, L. W. Han and K. Kato Poisson Equation Solver with 4th-order Accuracy by using Interpolated Differential Operator Scheme; Submitted to *App. Math. Phys.*
- [33] A. Brandt (1977) Multi-Level Adaptive Solutions to Boundary-Value Problems; *Math. Comput.*, vol. 31, pp 333-390
- [34] C. Qianshun (1987) Use of the Splitting Scheme and Multigrid Method to Compute Flow Separation; *Int. J. Num. Meth. Fluids*, vol.7, pp 719-731

---

##### 移流計算の高速化

---

- [35] M. Tanguay, A. Robert and R. Laprise (1990) A Semi-implicit Semi-Lagrangian Fully Compressible Regional Forecast Model; *Monthly Weather Rev.*, vol.118, pp.1970-1980
- [36] K. Sakurai, T. Aoki and T. Yabe (2001) A Semi-Lagrangian Cubic-Interpolated Propagation (CIP) Scheme for Long Parcel Trajectories; *Comp. Fluids Dynamics Journal*, vol. 10, no.1, pp.76-84
- [37] P. K. Smolarkiewics and J. A. Pudykiewicz (1992) A Class of Semi-Lagrangian Approximations for Fluids; *J. Atmospheric Sciences*, vol.49, no.22, pp.2082-2096
- [38] U. Ghia, K.N.Ghia and C.T.Shin (1982) High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method; *J. Comp. Phys.*, vol.48, pp 387-411

---

#### 第 5 章 陰的 IDO 法による CFL 数に対する安定化

---

- [39] K. Sakurai and T. Aoki(1999) Implicit IDO (Interpolated Differential Operator) Scheme; *Comp. Fluids Dynamics Journal*, vol.8, no.1, pp 6-12
- [40] M.Ida and T.Yabe(1995) Implicit CIP(Cubic-Interpolated Propagation) method in one dimension; *Comp.Phys.Comm.*, vol.92, pp 21-26
- [41] H. Makuuchi, T. Aoki and T. Yabe(1997) Implicit CIP(Cubic-Interpolated Propagation) Method in Two Dimensions; *JSME Inter.J.*, vol.40, pp 365-376



- [42] M.H.Gutknecht(1991) Variants of Bi-CGSTAB for matrices with complex spectrum; Tech.Report 91-14, IPS Research Report
- [43] Y.Saad and M.Schultz(1985) Conjugate gradient-like algorithms for solving nonsymmetric linear systems; Math. Comp., vol.44 pp.417-424
- [44] J. Zabusky and M.D.Kruskal(1965) Interaction of "SOLITONS" in a Collisionless Plasma and the Interface of Initial States; Phys.Rev.Lett.,vol.15,pp 240-243

#### 第6章 接触不連続面を含む問題への移動カットセル手法の導入

- [45] A. Harten (1983) High Resolution Schemes for Hyperbolic Conservation Laws; J. Comp. Phys. vol.49, pp.357-393
- [46] A. Harten (1984) On a Class of High Resolution Total-Variation-Stable Finite Difference Schemes; SIAM J. on numerical analysis, vol.21, pp 1-22
- [47] A. Harten, B. Engquist, S. Osher and S. Chakravarthy(1986) Uniformly high order essentially non-oscillatory schemes, III; J. Comp. Phys., vol. 71, pp.231-303
- [48] Chi-Wang Shu and Stanley Osher (1989) Efficient Implementation of Essentially Non-oscillatory Shock-Capturing Schemes II; J. Comp. Phys., vol.83, pp.32-78
- [49] M. Sussman, P. Smereka and S. Osher (1994) A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow; J. Comput. Phys., vol.114, pp. 146-159
- [50] Y. C. Chang, T. Y. Hou, B. Merriman and S. Osher (1996) A Level Set Formulation of Eulerian Interface Capturing Methods for Incompressible Fluid Flows; J. Comput. Phys. vol.124, pp.448-464
- [51] M. Sussmann and E. G. Puckett (2000) A Coupled Level Set and Volume-of-Fluid Method for Computing 3D and Axisymmetric Incompressible Two-Phase Flows; J. Comput. Phys., vol.162, pp.301-337
- [52] Takayuki AOKI, Seiji NISHITA and Kouichi SAKURAI (2001) Interpolated Differential Operator (IDO) Scheme and Application to Level Set Method; Comput. Fluid Dynamics Journal, vol.9, no.4, pp.406-417
- [53] T. Yabe and F. Xiao (1995) Description of Complex and Sharp Interface with Fixed Grids in Incompressible and Compressible Fluid; Computers Math. Applic., vol.29, no.1, pp 15-25
- [54] J.U. Brackbill and H.M. Ruppel, FLIP (1986) A method for adaptively zoned particle-in-cell calculations of fluid flows in two dimensions; J. Comput. Phys., vol.65, pp.314-343
- [55] De-Kang MAO (1993) A Treatment of Discontinuities for Finite Difference Methods in the Two-Dimensional Case; J. Comput. Phys., vol.104, pp.377-397
- [56] J.J.Quirk(1994) An Alternative to Unstructured Grids for Computing Gas Dynamic Flows around Arbitrarily Complex Two - Dimensional Bodies; Comp. Fluids, vol.23, no.1, pp.125-142
- [57] R. B. Pember and J. B. Bell (1995) An Adaptive Cartesian Grid Method for Unsteady Compressible Flow in Irregular Regions; J. Comput. Phys., vol.120, pp.278-304
- [58] H. Forrer and R. Jeltsch (1998) A Higher-Order Boundary Treatment for Cartesian-Grid Methods; J. Comput. Phys., vol.140, pp.259-277

- [59] James A. Viecelli (1971) A Computing Method for Incompressible Flows Bounded by Moving Walls; J. Comp. Phys., vol.8, pp.119-143
- [60] H. S. Udaykumar, H. C Kan, W. Shyy, and R. Tran-Son-Tay (1997) Multiphase Dynamics in Arbitrary Geometries on Fixed Cartesian Grids; J. Comp. Phys., vol.137, pp.366-405
- [61] G. Yang, D. M. Causon, D. M. Ingram, R. Saunders and P. Batten (1997) A Cartesian Cut Cell Method for Compressible Flows Part B: Moving Body Problems; The Aero. J. vol
- [62] E. M. Saiki and S. Biringen (1996) Numerical Simulation of a Cylinder in Uniform Flow: Application of a Virtual Boundary Method; J. Comput. Phys., vol.123, pp.450-465
- [63] K. Sakurai and T. Aoki (2001) A Numerical Procedure to Evolving Contact Discontinuity by using Moving Cut-Cell Method; Compt. Fluid Dynamics Journal, vol.10, no.1, pp.85-95
- [64] W.E.Lorensen and H.E.Cline (1987) Marching Cubes: A High Resolution 3D Surface Construction Algorithm; CG, vol.21,pp 163-169
- [65] G.M.Nielson, B.Hamann (1991) The Asymptotic Decider: Resolving the Ambiguity in Marching Cubes; Proc. IEEE Visualization, pp.83-91

---

Appendix A エルミート補間を用いた計算手法

---

- [66] F. Xiao, T. Yabe, T. Ito (1996) Constructing oscillation preventing scheme for advection equation by rational function; Comp. Phys. Comm., vol.93, pp.1-12
- [67] T. Utsumi (1995) Differential Algebraic Hydrodynamics Solver with Cubic-Polynomial Interpolation; Comput. Fluid Dynamics J., vol 4, pp.225-238
- [68] T. Utsumi, T. Kunugi, J. Koga (1998) A numerical Method for Solving the One-dimensional Vlasov-Poisson Equation in Phase Space; Comp. Phys. Comm., vol 108, no. 2-3, pp.159-179
- [69] R. Tanaka, T. Yabe and H. Wu (1999) A Class of Conservative Formulation of the CIP Method; Comp.Fluids Dynamics Journal in press
- [70] R. Tanaka, T. Nakamura, T. Yabe (2000) Constructing Exactly Conservative Scheme in a Non-conservative form; Comp. Phys. Comm., vol.126, pp.232-243
- [71] T. Aoki(1995) Multi-Dimensional Advection of CIP(Cubic-Interpolated Propagation) Scheme; Comp. Fluids Dynamics Journal, vol.4, no.3, pp 279-291