
第6章 シミュレーションの検証と考察

6.1 概要

前章まで、本研究における剛体運動シミュレーションのアルゴリズムについて説明した。

本章では、前章までに示したボクセルベース衝突判定法および制約ベース撃力計算法を用いたシミュレーションのアルゴリズムを実装し、さまざまな例に関して以下の3つのことを行った。

- 精度の検証と考察
- 計算時間の検証と考察
- アニメーションの作成

本章の剛体運動シミュレーションに用いたPCの性能については第4章に示したものと同様である。

6.2 ボクセルレベル、タイムステップに関する検証

6.2.1 精度の検証

第4章でも一部示したが、ここであらためて制約ベース剛体運動シミュレーションの精度について検証を行っていく。

精度検証は、Fig. 6-1のような初期条件で5個のスフィアを投射するというシミュレーションで行った。

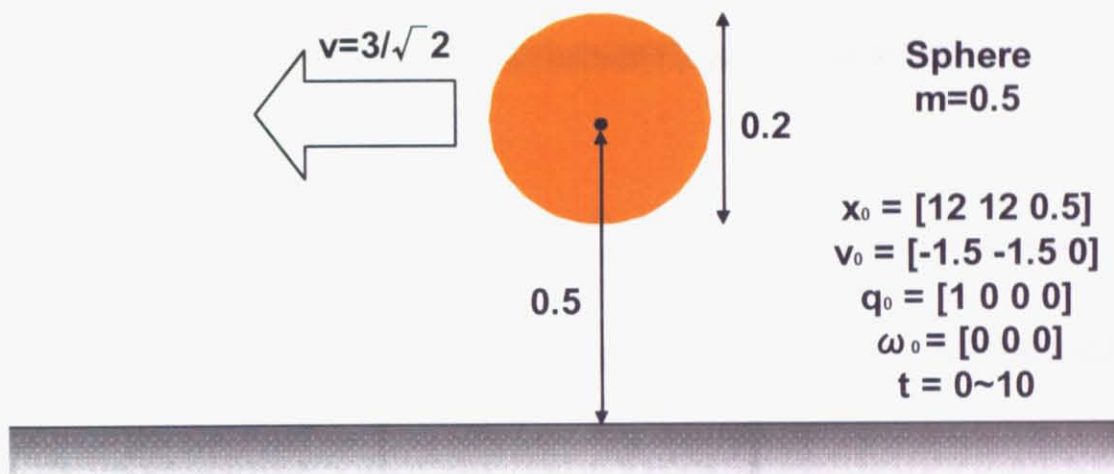


Fig. 6-1 スフィアの投射シミュレーションの初期条件

解析解を計算で求めるのは困難なため、参照解として、ボクセルレベル5、タイムステップ $\Delta t=1/400$ の場合の投射シミュレーションを用いた。比較するのは、ボクセルレベル2~5、タイムステップ $\Delta t=1/100$ という4条件におけるシミュレーションでのスフィアの最終重心位置である。

参照解の条件におけるスフィアの投射シミュレーションから作成されたアニメーションをFig. 6-2に示す。

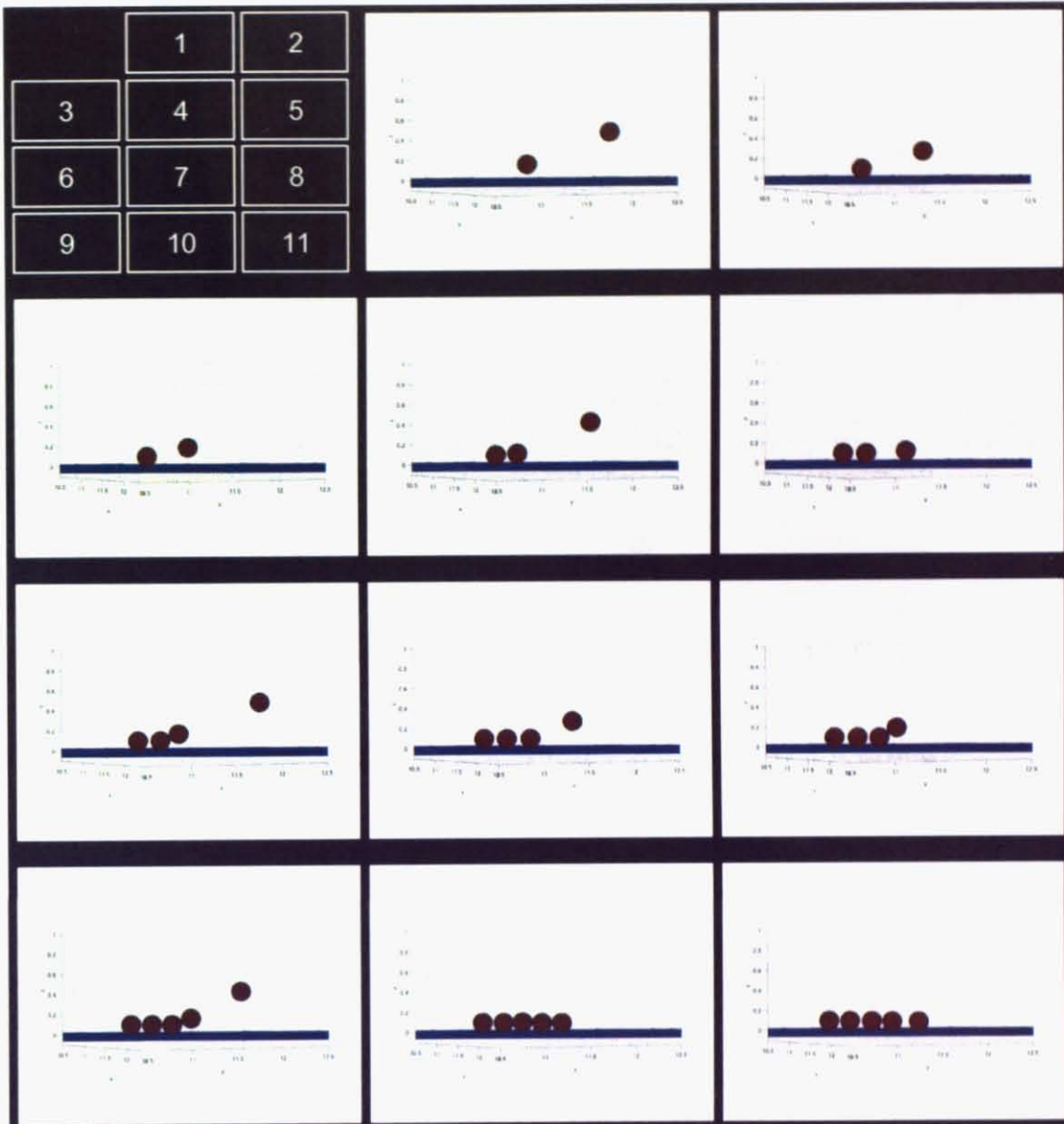


Fig. 6-2 スフィアの投射アニメーション例(レベル 5、 $\Delta t=1/400$)

Fig. 6-3は各ボクセルレベルにおけるスフィアの最終位置誤差をグラフにしたものである。スフィア i の最終位置誤差は以下の式で表す。

$$error_i = \sqrt{(x_i - x_{i,ref})^2 + (y_i - y_{i,ref})^2 + (z_i - z_{i,ref})^2} \quad (6.1)$$

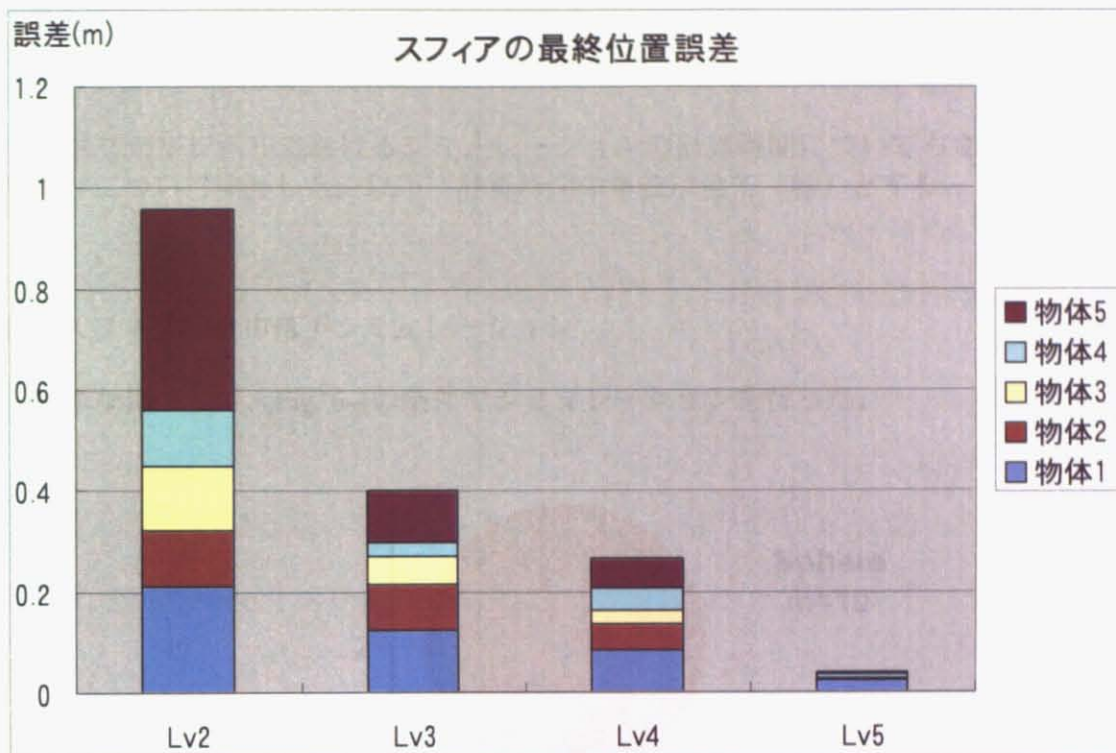


Fig. 6-3 各ボクセルレベルにおけるスフィア最終位置誤差

グラフより、ボクセルレベルが大きくなるほど誤差は小さくなる、つまり参照解の挙動に近づくことが示された。また、グラフのボクセルレベル 5 における誤差を見ると、タイムステップによる影響はボクセルレベルの影響と比較すると微小であることがわかる。

参考までに、計算時間はレベル 2 が 72 秒、レベル 3 が 170 秒、レベル 4 が 258 秒、レベル 5 が 384 秒、参照解シミュレーションが 721 秒であった。

6.2.2 計算時間の検証

各ボクセルレベルにおけるシミュレーションの計算時間についてもさまざまな初期条件について検証した。以下、計算時間の単位は全て「秒」とする。

1. スフィアの自由落下シミュレーション

簡単な例として、Fig. 6-4の条件でシミュレーションを行った。

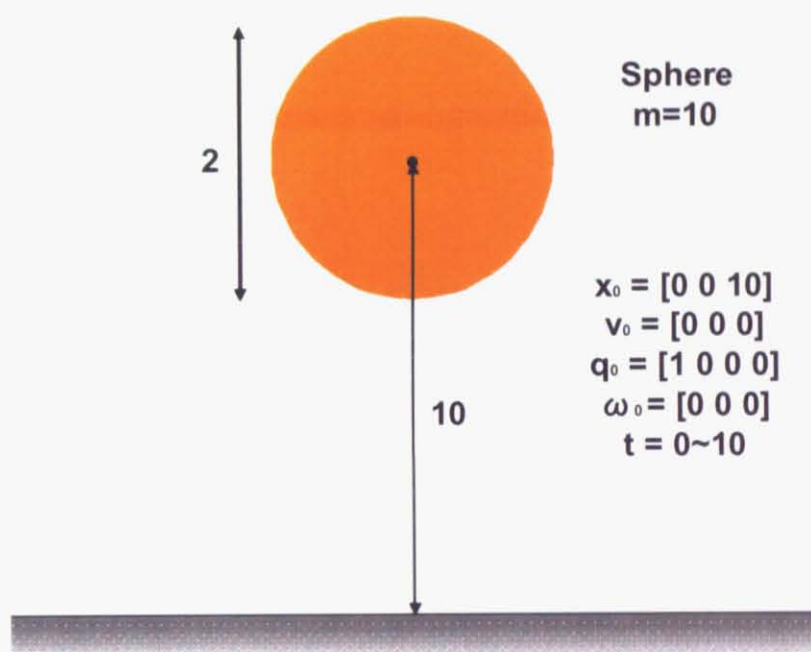


Fig. 6-4 スフィアの自由落下シミュレーションの初期条件

作成されたアニメーションの例をFig. 6-5に示す。Fig. 6-5はボクセルレベル5、タイムステップ $\Delta t=1/200$ でのアニメーション例である。

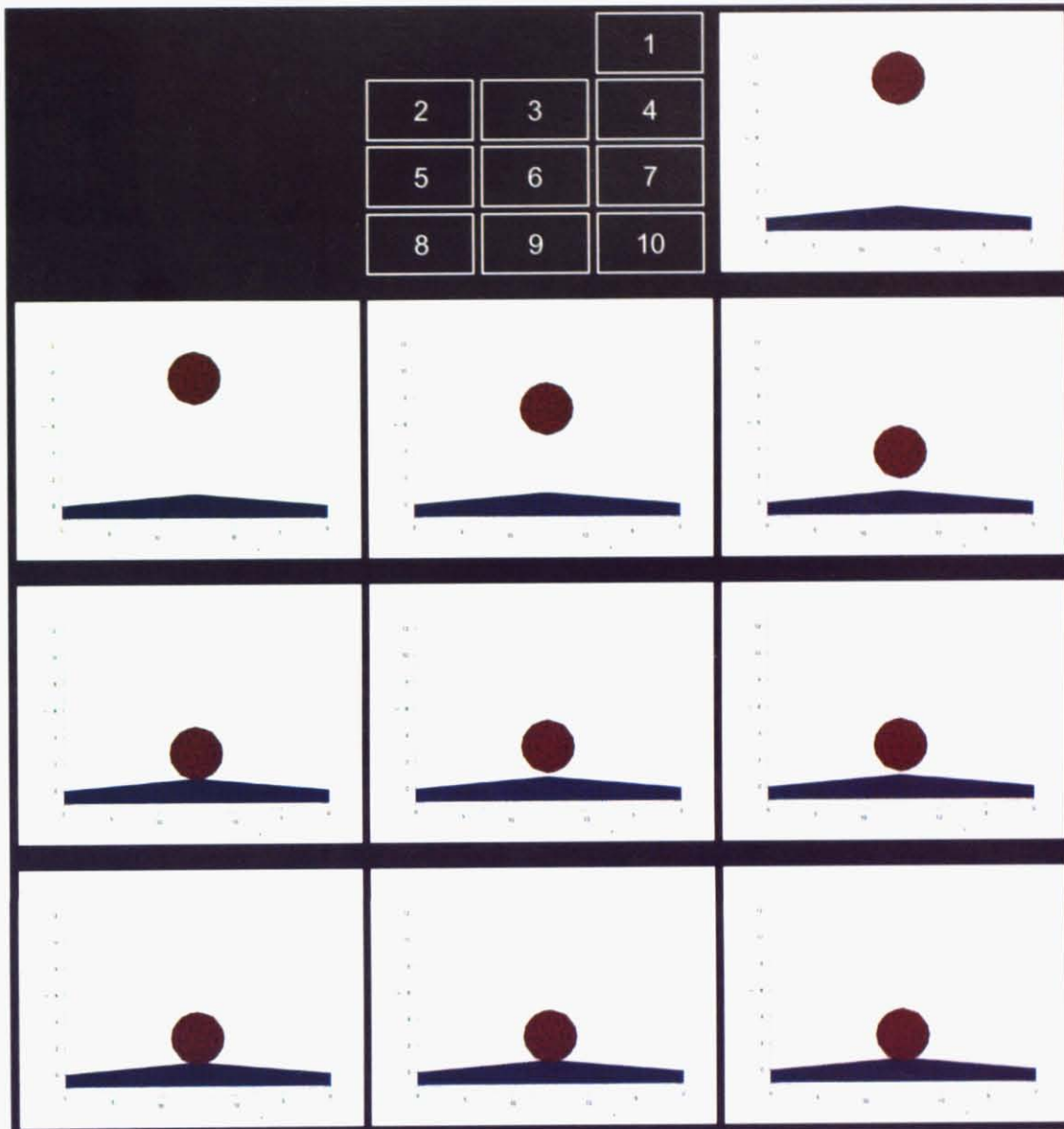


Fig. 6-5 スフィアの自由落下のアニメーション例(レベル 5、 $\Delta t=1/200$)

各ボクセルレベル、各タイムステップにおけるシミュレーション全体の計算時間をTable. 6-1に示す。また、Table. 6-1の計算時間のうち、衝突判定や時間積分などを除いて、純粹に制約ベース法による撃力計算のみにかかった時間をTable. 6-2に示す。

Table. 6-1 スフィアの自由落下シミュレーションの計算時間

z=10からの自由落下	t=0~10	Sphere	m=10		
分割レベル\タイムステップ	1/40	1/80	1/120	1/160	1/200
1	8.5453	10.797	14.449	21.473	41.636
2	25.536	31.942	43.802	63.705	120.03
3	33.37	40.658	54.2	79.593	164.69
4	44.705	57.125	77.558	113.3	226.24
5	30.973	38.709	52.116	76.555	154.34

Table. 6-2 スフィアの自由落下シミュレーションの計算時間(撃力計算のみ)

z=10からの自由落下	t=0~10	Sphere	m=10	C-basedのみ	
分割レベル\タイムステップ	1/40	1/80	1/120	1/160	1/200
1	1.2761	1.5892	1.9585	2.5723	4.9118
2	16.984	21.958	23.527	39.535	67.332
3	22.878	26.381	34.519	50.075	113.26
4	30.609	40.092	57.658	80.724	160.66
5	16.978	20.666	27.517	37.734	82.456

この結果を見ると、レベル3、レベル4の方がレベル5よりも計算時間がかかっていることがわかる。制約ベース法を用いた撃力計算にかかった時間のみを示したTable. 6-2を見ると、計算時間の逆転は制約ベース法撃力計算のアルゴリズムが影響していると考えられる。原因としては、レベル3、レベル4のスフィアモデルでは底面のボクセル外接球が16個であるのに対し、レベル5のスフィアモデルでは底面のボクセル外接球が8個であることが考えられる。したがって、撃力計算において衝突点の数が計算時間に大きく影響していることがわかる。

2. ヘキサポッドの床上滑走シミュレーション

回転運動も含めた例として、Fig. 6-6の条件でシミュレーションを行った。

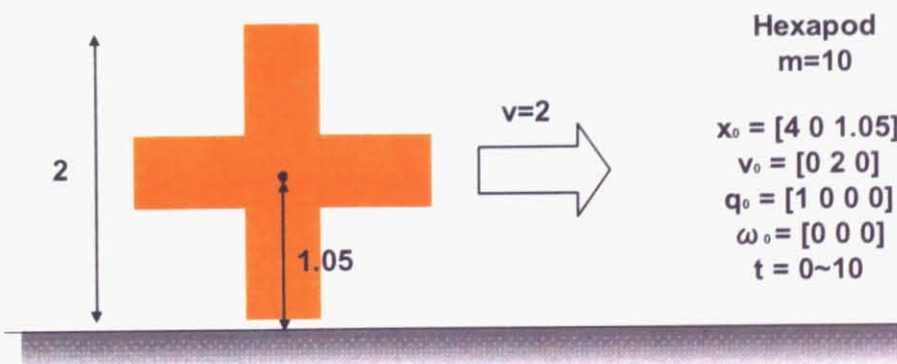


Fig. 6-6 ヘキサポッドの床上滑走シミュレーションの初期条件

ボクセルレベル 5、タイムステップ $\Delta t=1/200$ の条件で作成されたアニメーションの例をFig. 6-7に示す。

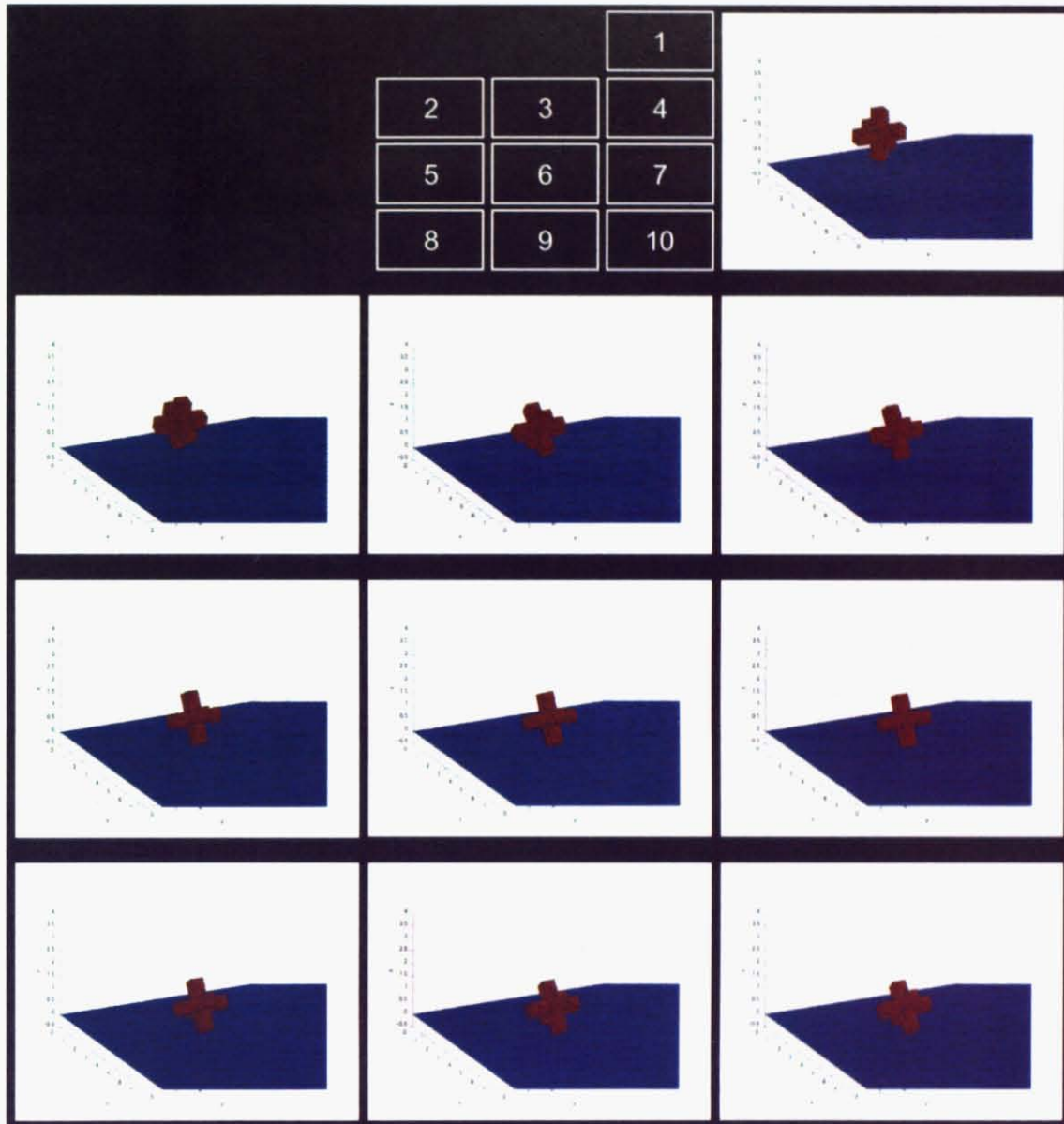


Fig. 6-7 ヘキサポッドの床上滑走のアニメーション例

各ボクセルレベル、各タイムステップにおけるシミュレーション全体の計算時間をTable. 6-3に示す。また、先ほどと同様に制約ベース法による撃力計算のみにかかった時間をTable. 6-4に示す。

Table. 6-3 ヘキサポッドの床上滑走シミュレーションの計算時間

z=1.05、v0=(4.0.0)の床上滑走	t=0~10	Hexapod	m=10		
分割レベル\タイムステップ	1/40	1/80	1/120	1/160	1/200
1	5.2735	6.7654	8.933	13.634	26.77
2	6.4171	6.2517	8.553	20.522	40.002
3	7.1189	8.5187	12.294	15.369	31.061
4	5.7578	7.3797	11.183	16.336	31.339
5	9.5594	14.805	16.644	24.072	43.525

Table. 6-4 ヘキサポッドの床上滑走シミュレーションの計算時間(撃力計算のみ)

z=1.05、v0=(4.0.0)の床上滑走	t=0~10	Hexapod	m=10	C-basedのみ	
分割レベル\タイムステップ	1/40	1/80	1/120	1/160	1/200
1	1.505	1.878	2.4946	3.6113	6.927
2	2.1464	1.3909	1.9312	8.9296	18.043
3	2.1687	2.2043	4.157	4.3939	9.107
4	1.3171	1.7447	3.3542	4.7821	7.1876
5	4.3362	8.205	8.1812	10.699	15.814

このシミュレーションでも、タイムステップによってはボクセルレベル 2 やレベル 3 でのシミュレーションがレベル 4 を上回っているという現象が生じた。この原因としては、レベル 2 ではボクセルモデルが立方体と同じになってしまい、回転しにくい形状なので、その分底面の部分が衝突している時間が上のレベルのボクセルに比べて長いことが挙げられる(Fig. 6-8)。

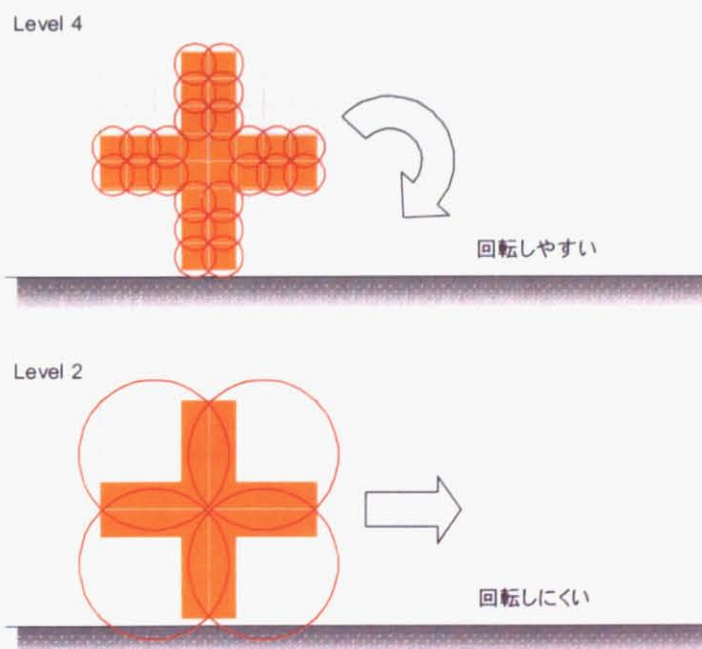


Fig. 6-8 ボクセルレベルと計算時間の逆転の原因

3. 複数のヘキサポッドの落下シミュレーション

物体の個数の影響を検証するために、以下のような条件で複数の物体を落下させるシミュレーションを行った。

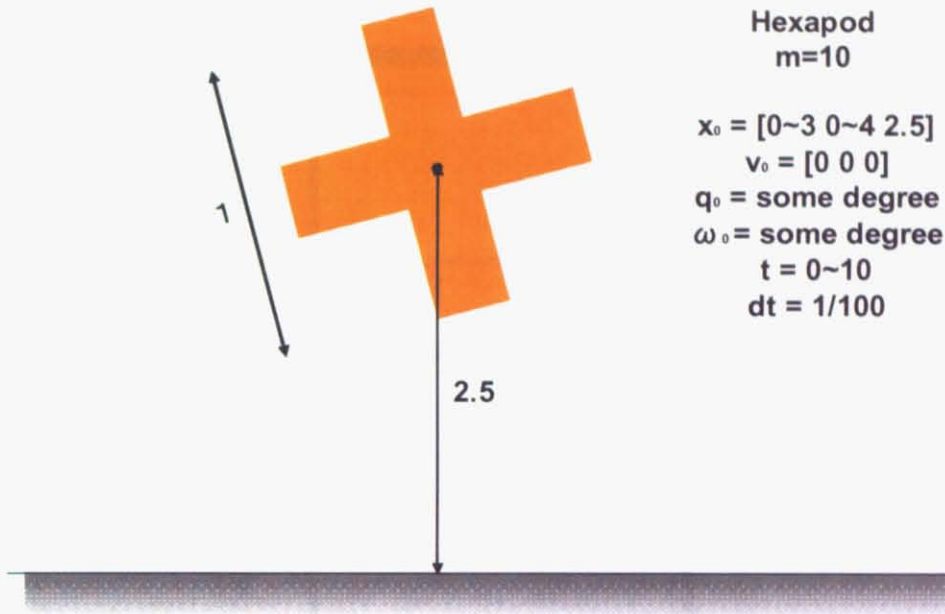


Fig. 6-9 ヘキサポッドの複数落下シミュレーションの初期条件

物体ごとの初期条件は、初期重心位置、初期姿勢、初期回転速度についてはある範囲のランダムな値を設定した。

Fig. 6-10に18個のヘキサポッドをボクセルレベル5、タイムステップ $\Delta t=1/100$ で落下させた場合のアニメーションの例を示す。

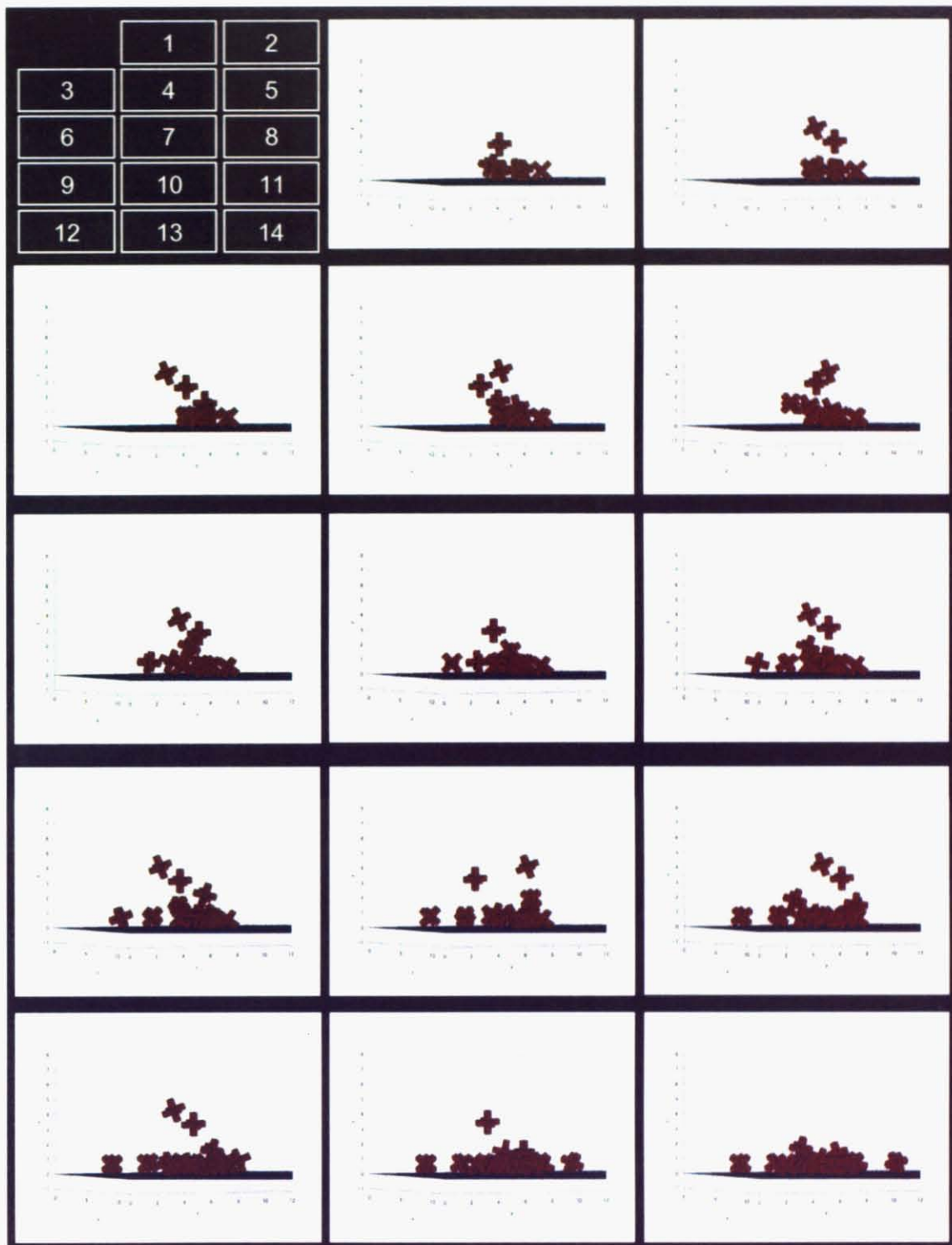


Fig. 6-10 ヘキサポッドの複数落下のアニメーション例

ボクセルレベル 4、5 における物体個数と計算時間の関係を Table. 6-5 に示す。タイムステップは全て $\Delta t = 1/100$ 、シミュレーション時間は 10 秒である。

Table. 6-5 ヘキサポッドの複数落下シミュレーションの計算時間

ボクセルレベル\物体個数	1	5	10	15
4	29.91	180.59	562.53	1527.08
5	38.2	350.91	892.22	3121.09

単位: 秒

Table. 6-5から、ボクセルレベルが大きく、物体個数が多いほど計算時間はかかるということがわかる。

6.2.3 考察

本研究で提案したアルゴリズムをもとに、さまざまな条件でシミュレーションを行った結果、以下のことが判明した。

- 現実的な挙動は確認できる
- ボクセルレベルが大きくなるほど精度は高くなる
- シミュレーション全体としては、ボクセル分割レベルより衝突点の数が計算時間に大きな影響を与える
- 物体個数が多くなるほど計算時間はかかる

6.3 他手法との比較検証

6.3.1 精度の検証

ここでは、撃力ベース法および見かけの衝突点を用いた制約ベース法シミュレーションとの精度の比較検証を行った。

1. ヘキサポッドの複数落下シミュレーション

まずは、以下のような条件で各手法についてシミュレーションを行った。

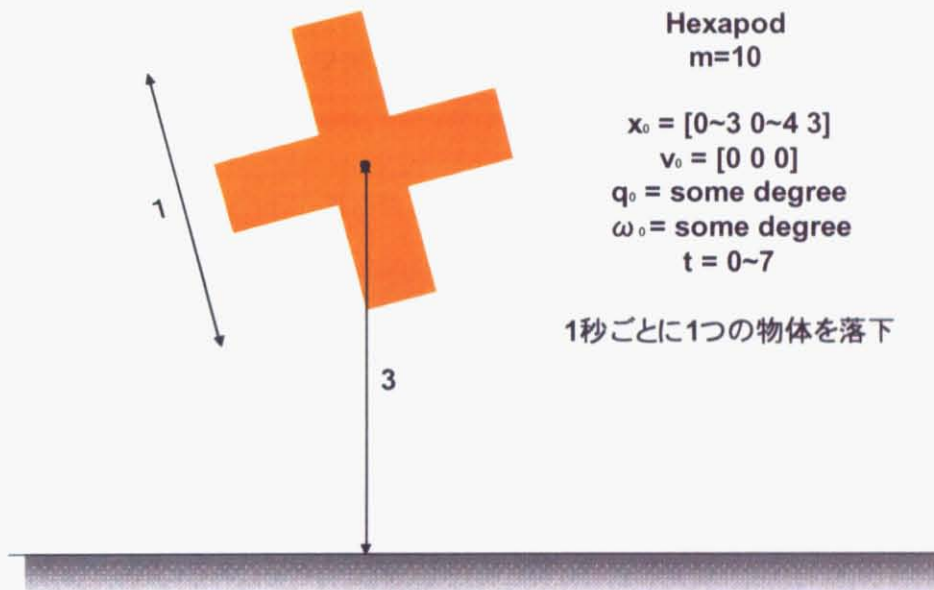


Fig. 6-11 ヘキサポッドの複数落下シミュレーションの初期条件

各手法は、衝突判定部分のアルゴリズムは全く同じであり、衝突時の撃力計算部分のアルゴリズムのみが異なる。そこで、精度検証として、参照解との重心位置の誤差を衝突回数で割り、その値を時系列で表現することとした。式で表すと以下のようになる。

$$error_i(t) = \frac{\sqrt{(\mathbf{x}_i(t) - \mathbf{x}_{ref}(t))^2}}{ncol_i(t)} \quad (6.2)$$

ここで、 $ncol_i(t)$ は物体 i の時刻 t までの衝突回数である。

アニメーション例は、Fig. 6-10と似たようなアニメーションとなるのでここでは省略する。

参照解は、ボクセルレベル 5、タイムステップ $\Delta t=1/500$ でのシミュレーションにおける各物体の重心位置とした。比較対象は、以下の 5 つである。

- i. ボクセルレベル 5、タイムステップ $\Delta t=1/100$ 、制約ベース
- ii. ボクセルレベル 4、タイムステップ $\Delta t=1/100$ 、制約ベース
- iii. ボクセルレベル 3、タイムステップ $\Delta t=1/100$ 、制約ベース
- iv. ボクセルレベル 5、タイムステップ $\Delta t=1/100$ 、撃力ベース
- v. ボクセルレベル 5、タイムステップ $\Delta t=1/100$ 、制約ベース+見かけの衝突点

以下は、参照解と i~v の各シミュレーションとにおける各物体の衝突毎平均誤差を示した 7 秒間の時系列グラフである。

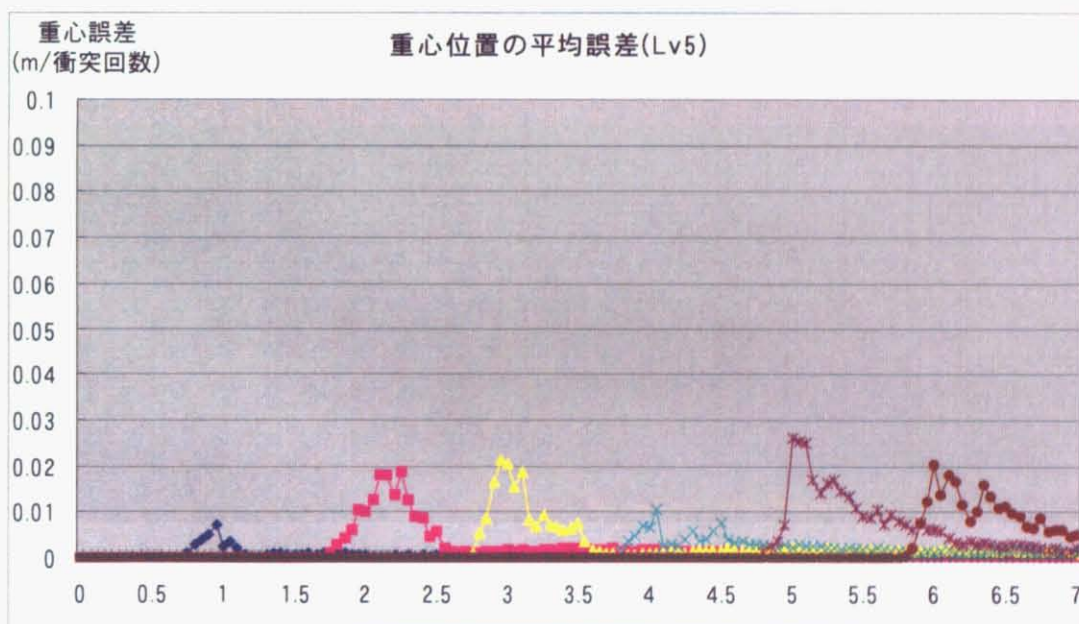


Fig. 6-12 参照解と i との重心位置平均誤差

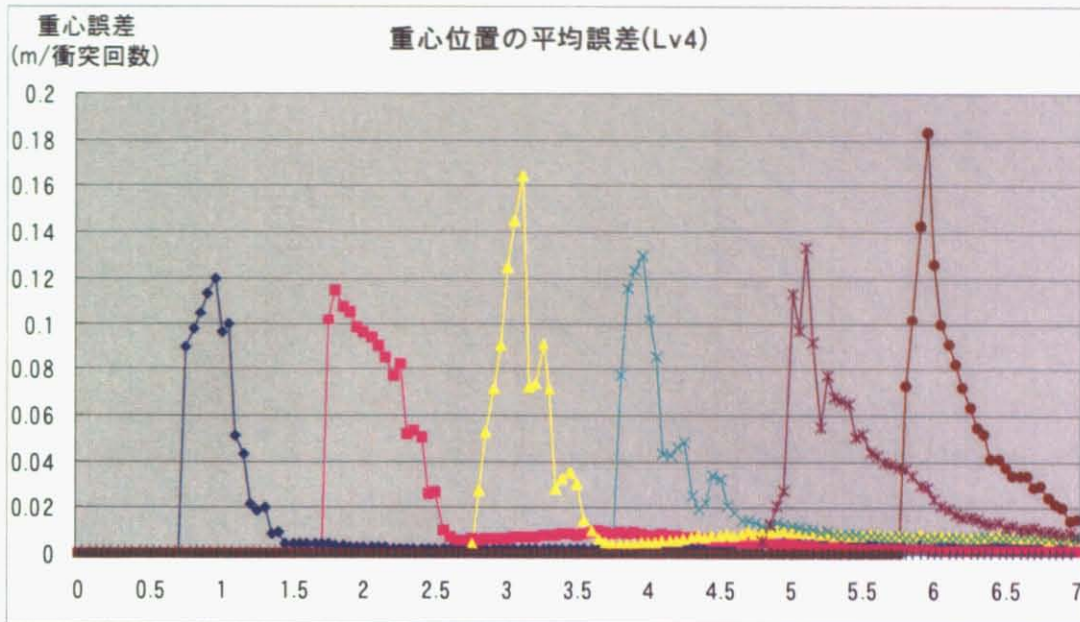


Fig. 6-13 参照解と ii との重心位置平均誤差

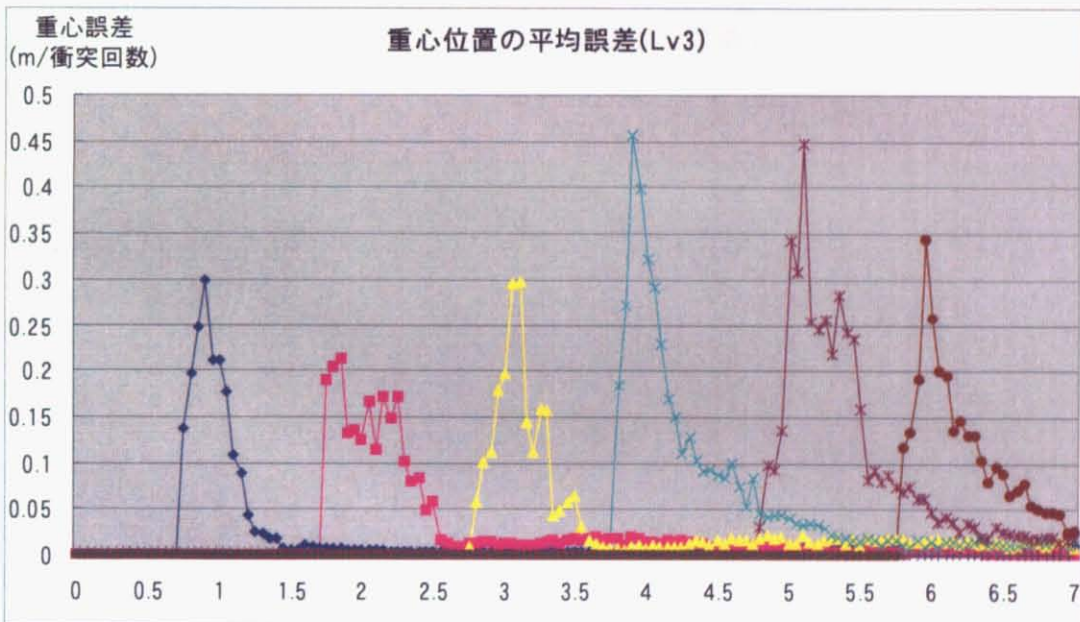


Fig. 6-14 参照解と iii との重心位置平均誤差

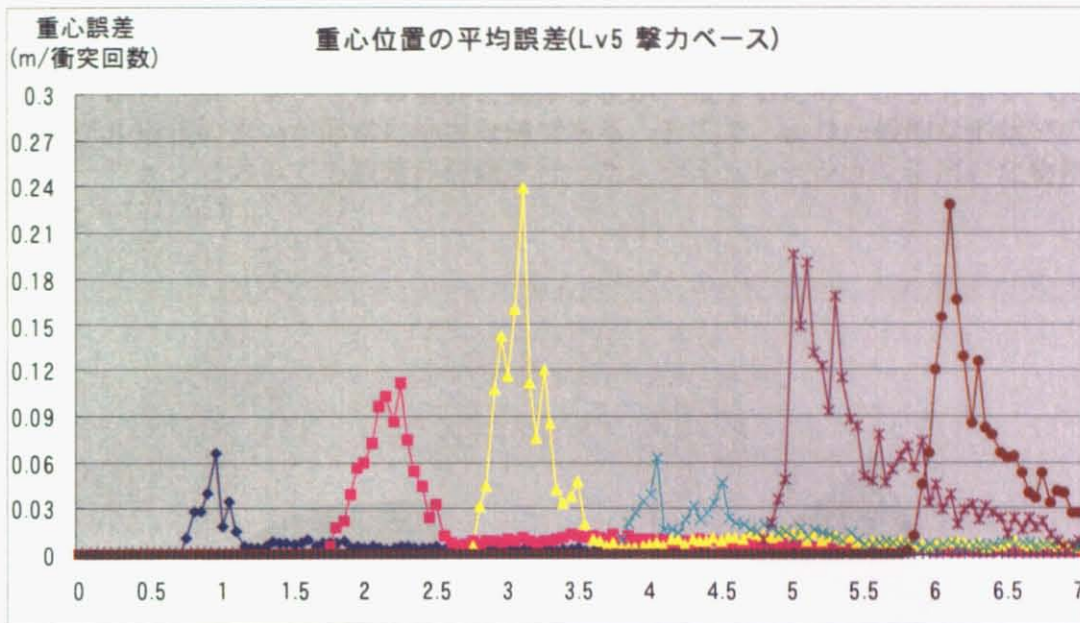


Fig. 6-15 参照解とivとの重心位置平均誤差

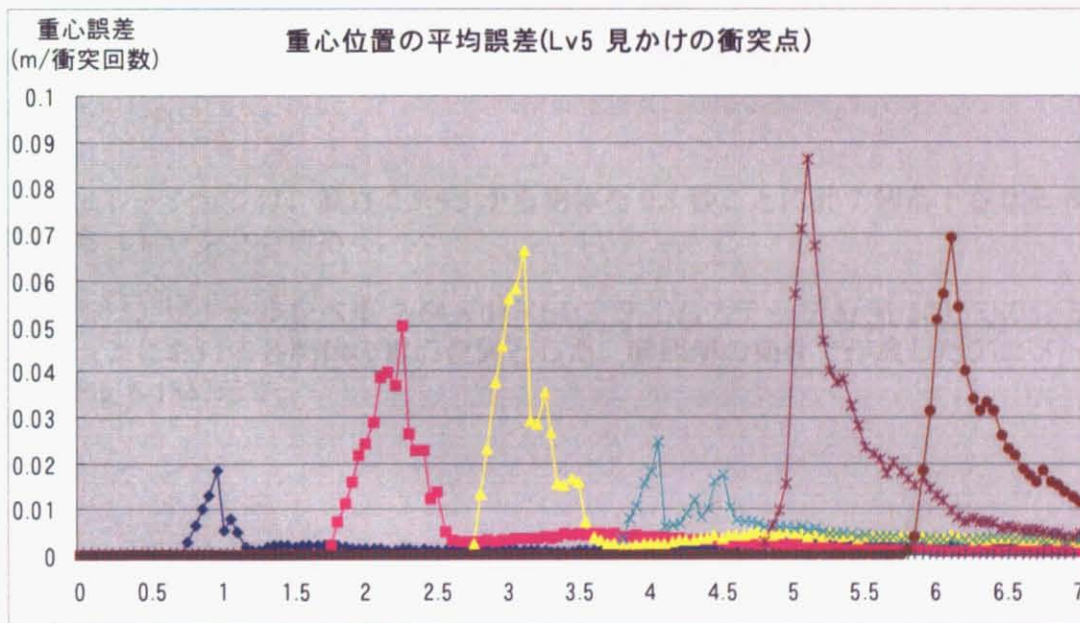


Fig. 6-16 参照解とvとの重心位置平均誤差

以上の結果から、タイムステップによる誤差の影響はボクセルレベルによる誤差の影響に比べて非常に小さいこと、見かけの衝突点を用いたシミュレーションは撃力ベース法によるシミュレーションと比較して精度が高いことが示された。

2. 岩状物体の複数落下シミュレーション

ヘキサポッドは、ボクセルの境界と物体形状が一致するため、ボクセルレベルによるモデル形状の違いが非常に小さな例である。そこで、より一般的な形状でのシミュレーションにおいても誤差の評価を行った。シミュレーションに用いた物体形状をFig. 6-17に示す。

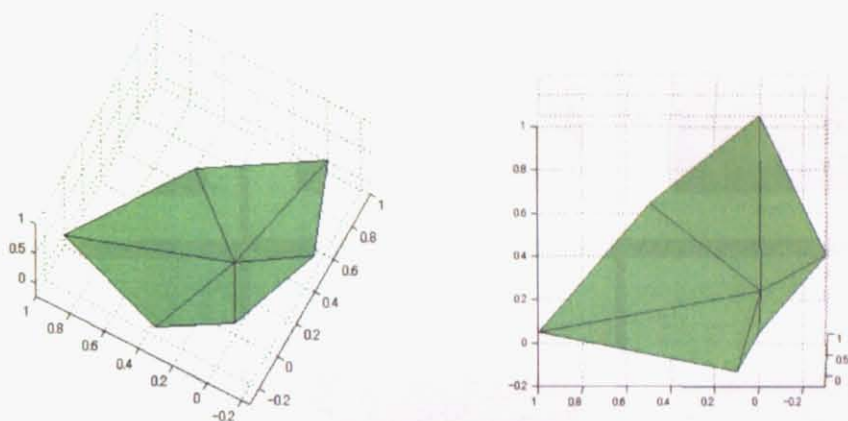


Fig. 6-17 シミュレーションに用いた物体の形状

シミュレーションは、高さ 2.5~3 から物体を 0.8 秒ごとに計 7 個落下させる 6 秒間のシミュレーションである。

参照解は、ボクセルレベル 7(64×64×64)、タイムステップ $\Delta t=1/500$ でのシミュレーションにおける各物体の重心位置とした。参照解の条件で作成したアニメーションをFig. 6-18に示す。

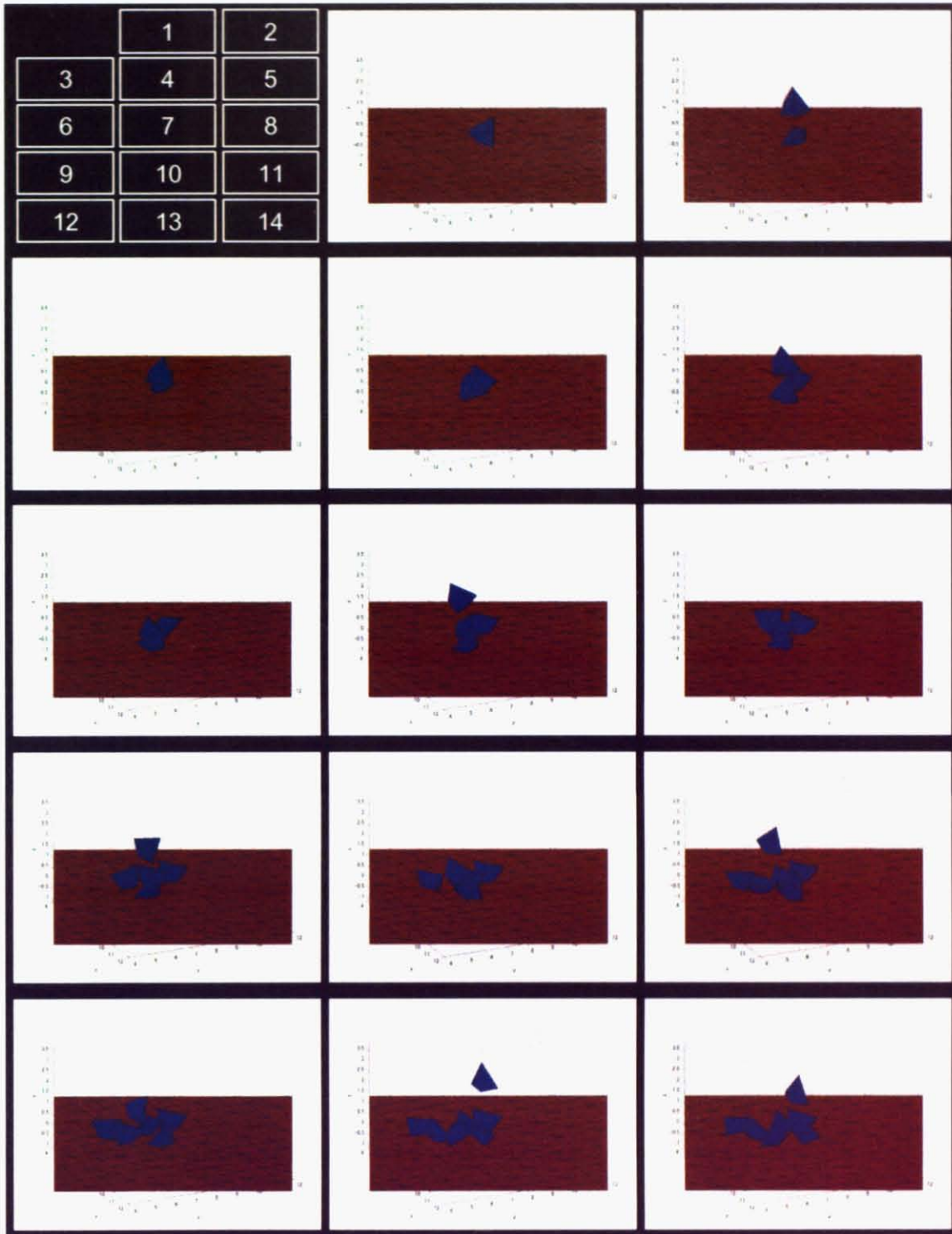


Fig. 6-18 岩のような物体の落下のアニメーション例

また、参照解との比較対象は、以下の7つである。

- i. ボクセルレベル 6、タイムステップ $\Delta t=1/200$ 、制約ベース
- ii. ボクセルレベル 5、タイムステップ $\Delta t=1/200$ 、制約ベース
- iii. ボクセルレベル 4、タイムステップ $\Delta t=1/200$ 、制約ベース
- iv. ボクセルレベル 6、タイムステップ $\Delta t=1/200$ 、撃力ベース
- v. ボクセルレベル 5、タイムステップ $\Delta t=1/200$ 、撃力ベース
- vi. ボクセルレベル 6、タイムステップ $\Delta t=1/200$ 、制約ベース+見かけの衝突点
- vii. ボクセルレベル 5、タイムステップ $\Delta t=1/200$ 、制約ベース+見かけの衝突点

以下に、参照解と i ~ vii に示した条件でのシミュレーションとの重心位置平均誤差を時系列グラフにしたものを示す。

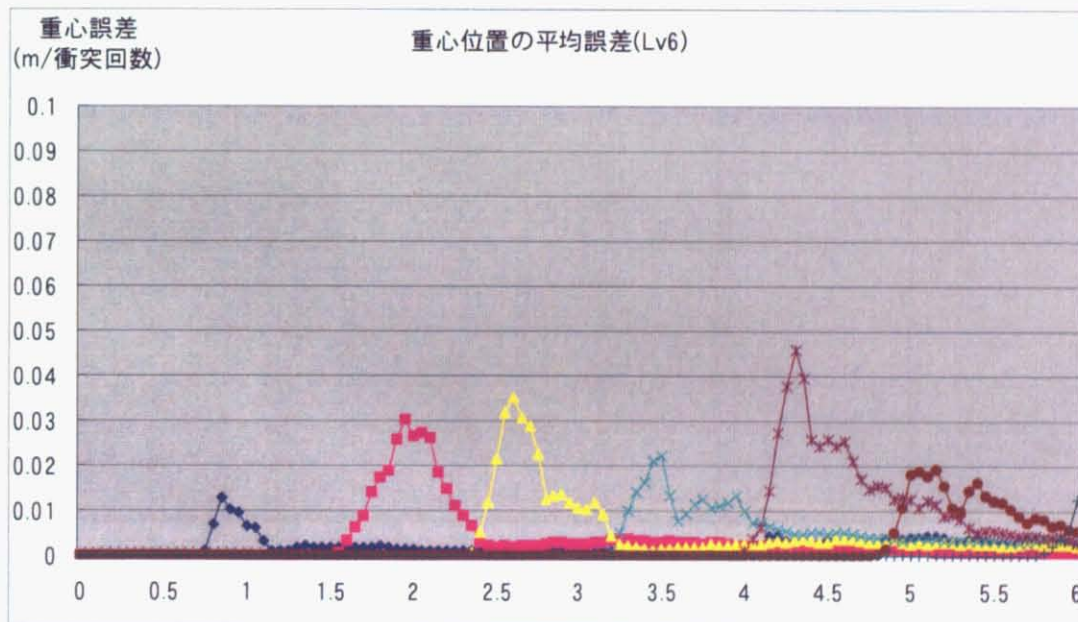


Fig. 6-19 参照解と i との重心位置平均誤差

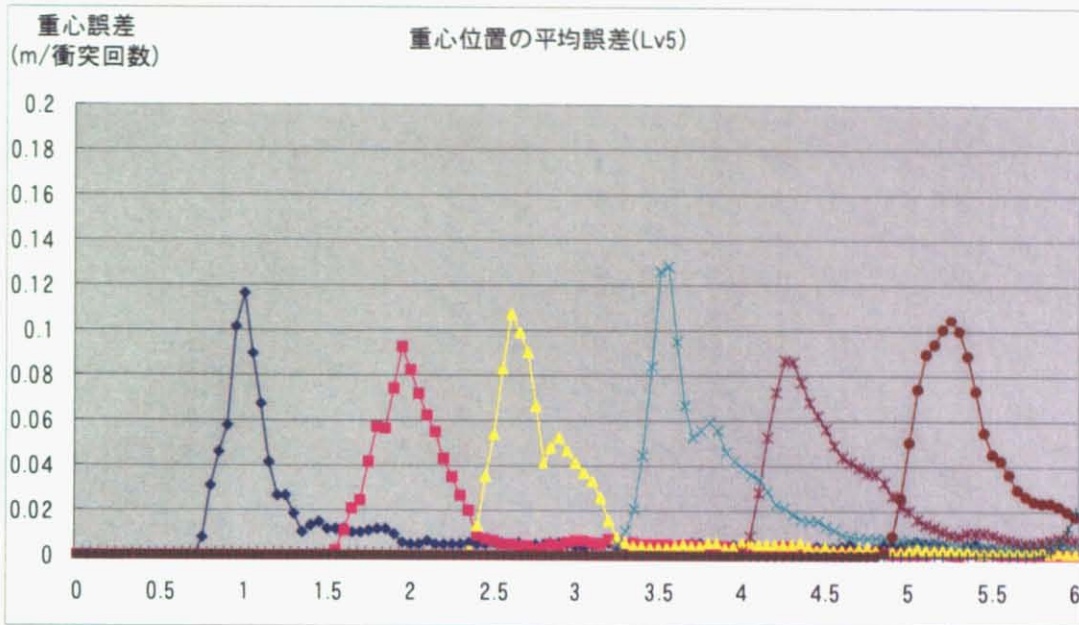


Fig. 6-20 参照解とiiとの重心位置平均誤差

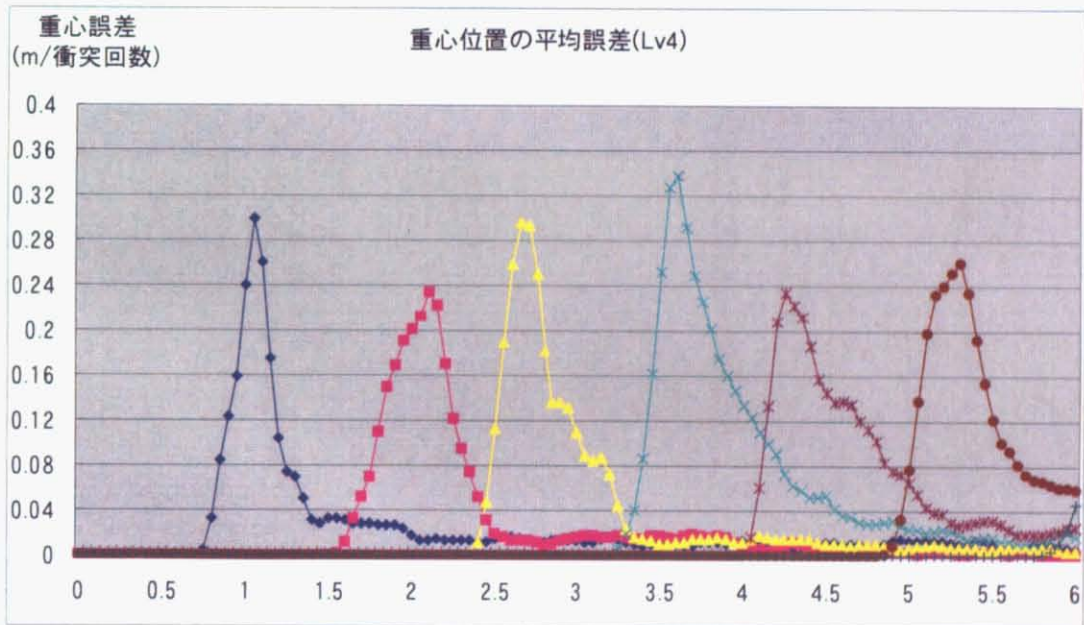


Fig. 6-21 参照解とiiiとの重心位置平均誤差

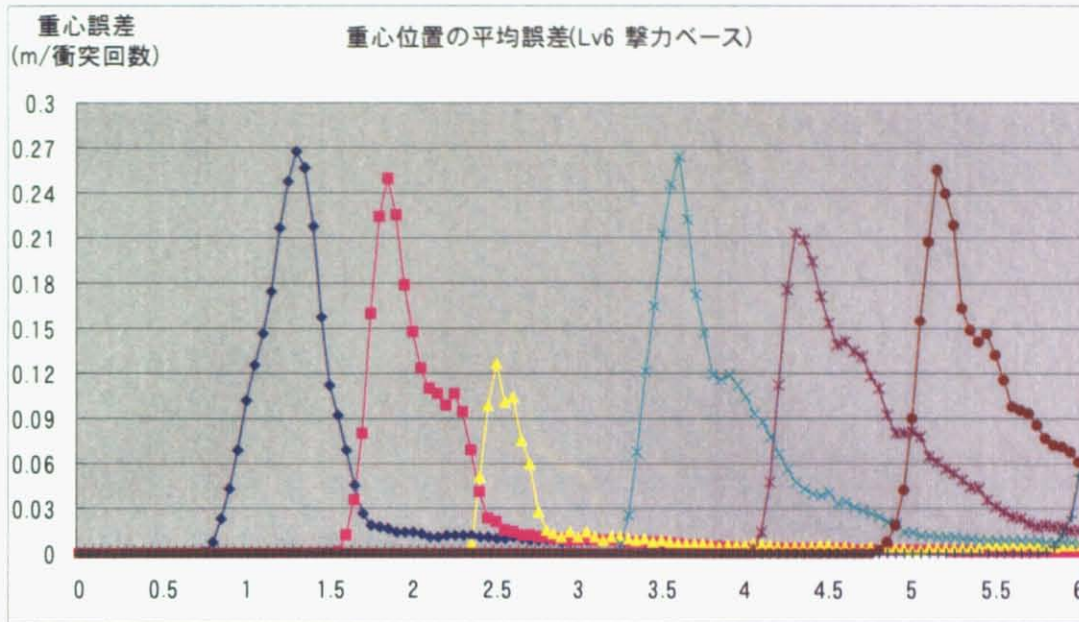


Fig. 6-22 参照解とivとの重心位置平均誤差

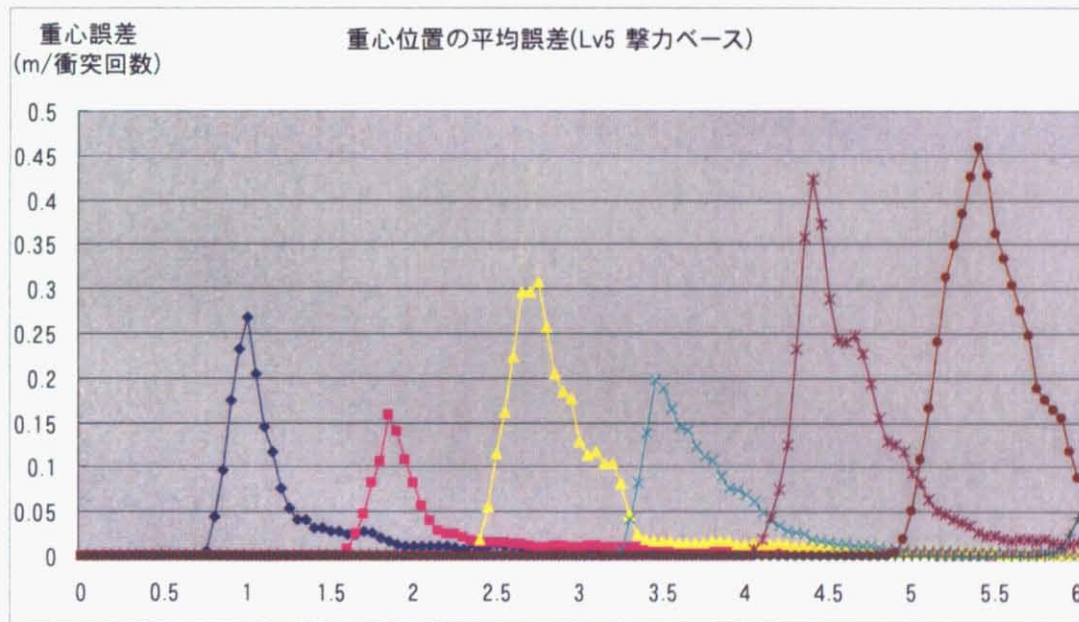


Fig. 6-23 参照解とvとの重心位置平均誤差

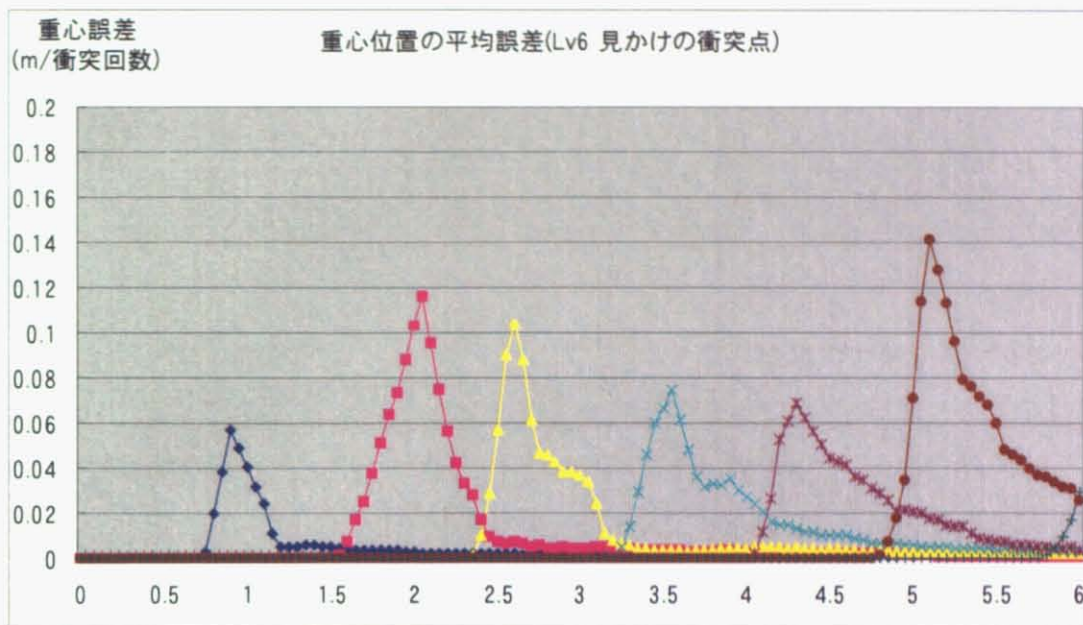


Fig. 6-24 参照解とviとの重心位置平均誤差

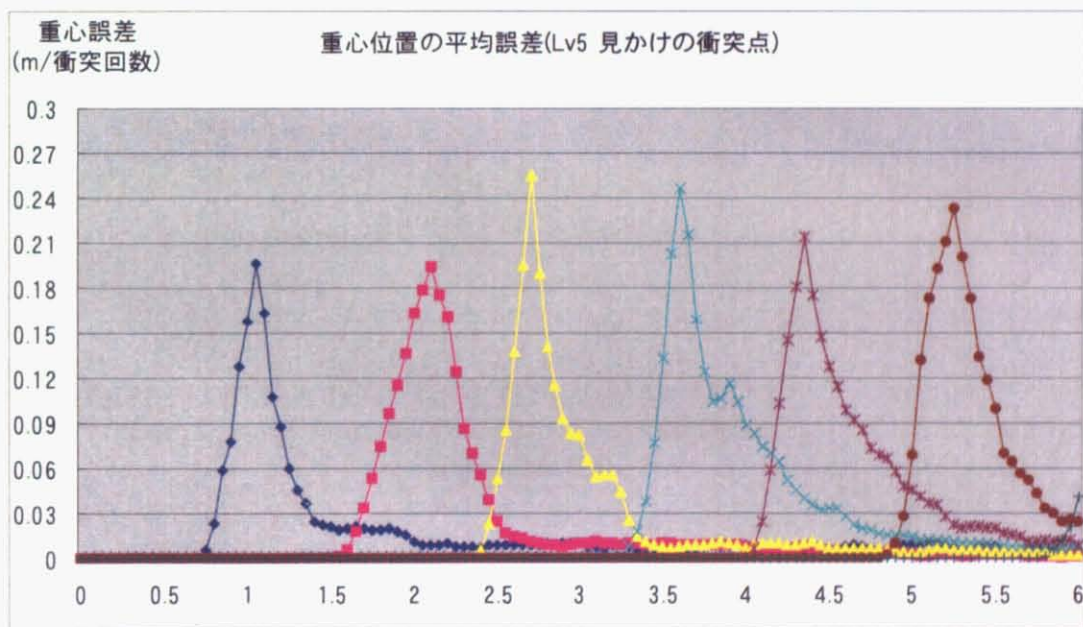


Fig. 6-25 参照解とviiとの重心位置平均誤差

Table. 6-6に各ボクセルレベル、各手法に対する平均誤差を物体のスケールで無次元化した値を示す。

Table. 6-6 同一タイムステップにおける各ボクセルレベル、各手法の平均誤差

ボクセルレベル\手法	撃力ベース	制約ベース	見かけ+制約
4	43.75	35	41.25
5	36.25	13.75	25.625
6	26.25	3.125	11.75
			単位: %

また、同一タイムステップではなく、ボクセルレベルに応じてタイムステップを変更した場合の平均誤差についてもTable. 6-7に示す。

Table. 6-7 ボクセルレベルに応じたタイムステップを設定した場合の平均誤差

ボクセルレベル\手法	撃力ベース	制約ベース	見かけ+制約
4 (dt=1/50)	48	34.75	44.25
5 (dt=1/100)	40.5	15.25	26.125
6 (dt=1/200)	26.25	3.125	11.75
			単位: %

このシミュレーションにおいても、撃力ベース法と比較して見かけの衝突点を用いたシミュレーションの精度が高いことを確認できた。

6.3.2 計算時間の検証

6.3.1の「1. ヘキサポッドの複数落下シミュレーション」および「2. 岩状物体の複数落下シミュレーション」における各手法のシミュレーションにかかった計算時間をボクセルレベルごとにTable. 6-8およびTable. 6-9(ボクセルレベルに応じたタイムステップを設定した場合は)に示す。

Table. 6-8 ヘキサポッドの複数落下シミュレーションの計算時間($\Delta t = 1/100$)

Lv\撃力計算法	撃力ベース	制約ベース	制約+見かけ
3	41	224	93
4	79	391	155
5	132	615	229
参照5(dt=1/500)		2890	
			単位: 秒

Table. 6-9 岩状物体の複数落下シミュレーションの計算時間($\Delta t=1/200$)

Lv\撃力計算法	撃力ベース	制約ベース	制約+見かけ
4	124	676	274
5	202	1108	396
6	570	3230	844
参照7(dt=1/500)		27901	
			単位:秒

Table. 6-10 岩状物体の複数落下シミュレーションの計算時間(Δt =ボクセルレベルに応じた値)

Lv\撃力計算法	撃力ベース	制約ベース	制約+見かけ
4(dt=1/50)	40	178	81
5(dt=1/100)	114	537	221
6(dt=1/200)	570	3230	844
参照7(dt=1/500)		27901	
			単位:秒

6.3.3 考察

これまでのさまざまな検証の結果から、計算時間に関しては撃力ベース法に一日の長があるが、精度に関して見かけの衝突点を用いた制約ベース法シミュレーションの優位性を確認できた。

また、タイムステップの影響は、ボクセルレベルに応じて貫通しない値を取ることによって計算時間を短縮することができ、精度にはボクセルレベルほど大きな影響は与えないことがわかった。

6.4 作成したアニメーション例

6.4.1 MATLAB 上でのアニメーション例

ボクセルベース法は物体形状の複雑さに依存しない衝突判定であることを第3章で述べた。そこで、表面ポリゴンが非常に多い物体でのシミュレーションおよびアニメーション作成を行った。シミュレーションに用いた物体形状をFig. 6-26に示す。



Fig. 6-26 表面のポリゴンが多い物体の例

Fig. 6-26の物体情報は以下の通りである。

- 接点数：1,530
- 表面ポリゴン数：2,558

Fig. 6-27に作成したアニメーションを示す。シミュレーションの条件は、ボクセルレベル 5、タイムステップ $\Delta t=1/500$ 、6 秒間のシミュレーションで、物体個数は 20 個である。計算時間は 9688 秒だった。

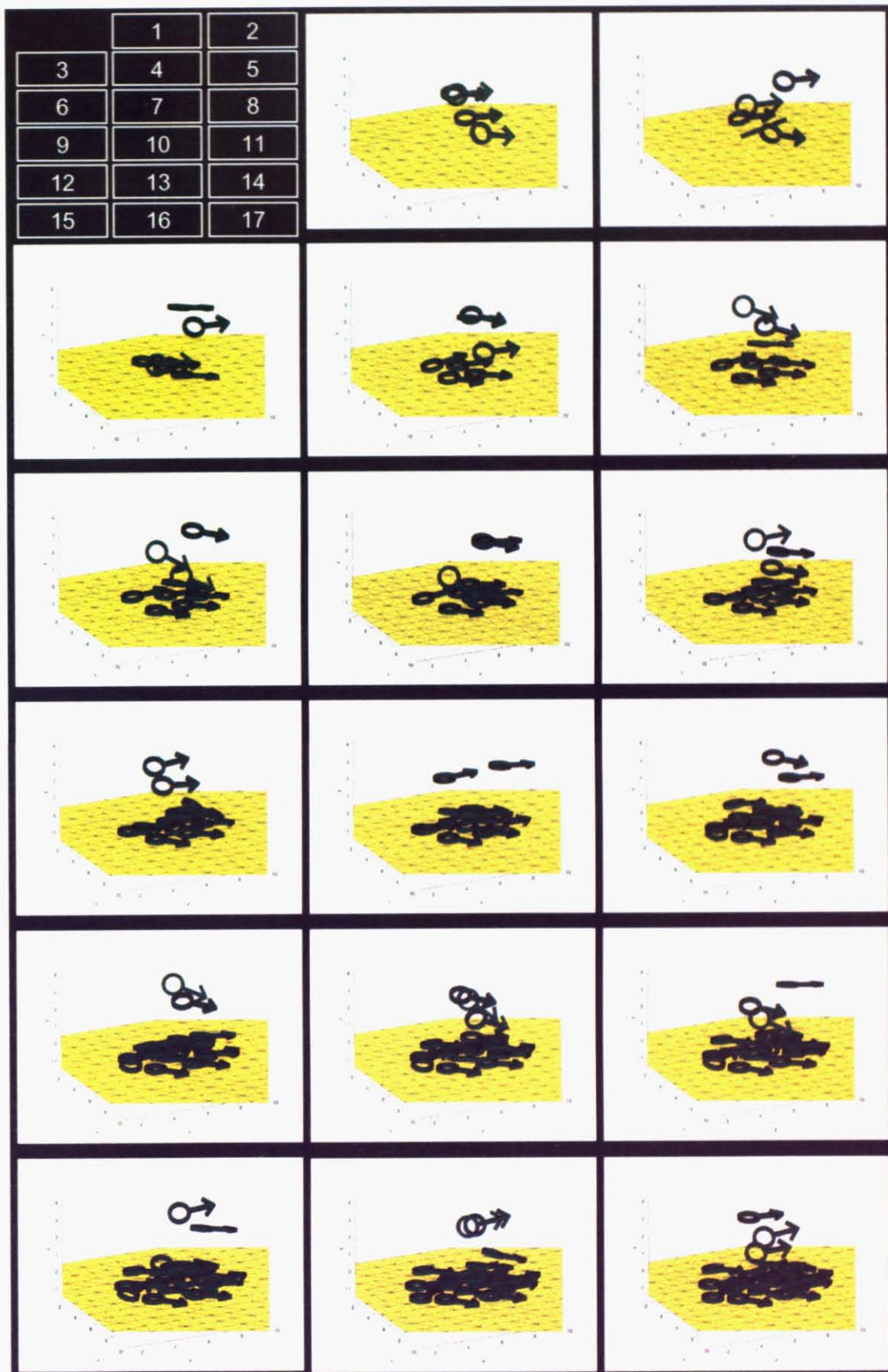


Fig. 6-27 ポリゴン数が多い物体の落下アニメーション

6.4.2 Cell 上でのアニメーション例

株式会社フィックスターズ(<http://www.fixstars.com/>)によって、本研究で提案および実装したシミュレーションアルゴリズムを用いた物理エンジンの開発が行われている。特に、Cell と呼ばれる高速な CPU 上でのリアルタイムシミュレーションを目的としている。

Cell とは、ソニー、ソニー・コンピュータ・エンターテインメント、IBM、東芝が共同で開発したマイクロプロセッサで、1つの CPU の中に9個のプロセッサコアを持つマルチコア CPU である。1個の汎用プロセッサコア(PPE)と8個のシンプルなプロセッサコア(SPE)から構成される。この8個の SPE が並列演算に強みを発揮し、物理演算やシミュレーション、音声処理などに対して非常に強力な性能を発揮させることができると期待されている。現在はソニー・コンピュータ・エンターテインメントが発売したプレイステーション3に搭載されている。

今回の物理エンジン開発の手順としては、本研究で実装された MATLAB のプログラムを一度 C++に書き直し、その上で Cell 用に C++のプログラムを最適化し、Cell 上にポーティングするというものである。

開発中の物理エンジンの例をFig. 6-28、Fig. 6-29に示す。Fig. 6-28でのシミュレーションの条件はボクセルレベル4のキューブ20個の落下シミュレーションであり、描画も含めてリアルタイムで動いている。Fig. 6-29でのシミュレーションの条件はボクセルレベル2の物体200個以上が描画も含めてリアルタイムで動いている。

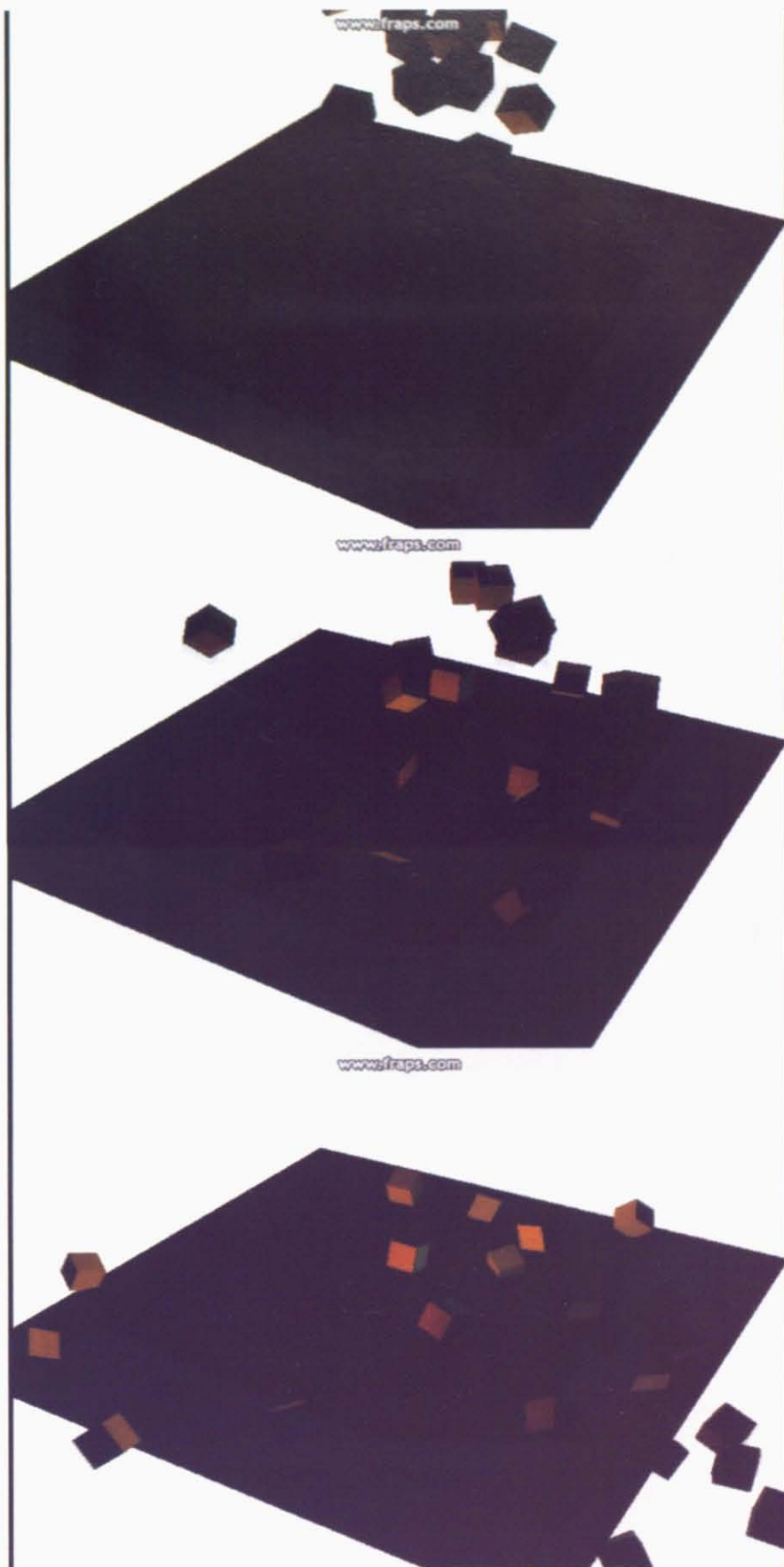


Fig. 6-28 Cell 上での制約ベース物理シミュレーションの例



Fig. 6-29 Cell 上での制約ベース物理シミュレーションの例 2

第7章 結論

本研究では、衝突判定法としてボクセルベース法を採用し、その上で各撃力計算手法の検証・考察などを行い、以下の結論を得た。

撃力計算手法に関しては、正確なシミュレーションという点に重きをおき、ペナルティ法、SEM 法、ベクトル場を用いる手法を除外した。また、実際にプログラムを作成して比較検証した上で、多点衝突時における撃力ベース法の問題点を指摘し、制約ベース法の優位を示すことができた。

また、計算コストの軽減手法として見かけの衝突点を用いた制約ベース法アルゴリズムを提案した。撃力の仮定より見かけの衝突点位置、見かけの衝突点における法線方向ベクトル、時間積分の式を求めた。さらにシミュレーションを実行し、撃力ベース法と比較して見かけの衝突点を用いた手法の精度に関する優位性を示すことができた。

最後にボクセルベース衝突判定法および制約ベース撃力計算法を用いた本研究のアルゴリズムを用いて、MATLAB 上および企業の協力による Cell 上でのアニメーションの作成例を示した。

参考文献

- [1] 久保田純: ボクセルベース衝突シミュレーションによる CG 制作法, 東京大学大学院工学系研究科環境海洋工学専攻 修士論文, 2004
- [2] P. Jimenez et al.: 3D collision detection: a survey, *Computers & Graphics* 25, pp. 269-285, 2001
- [3] S.Ding and M.A.Mannan and A.N.Poo: Oriented bounding box and octree based global interference detection in 5-axis machining of free-form surfaces, *Computer-Aided Design*, In Press, Uncorrected Proof, 2003
- [4] V.J.Milenkovic and H.Schmidl: Optimization-Based-Animation, *ACM SIGGRAPH 2001* pp.37-46, 2001
- [5] Dingliana, J and O'Sullivan, C.: Graceful Degradation of Collision Handling in Physically Based Animation, *Computer Graphics Forum Vol.19 Issue3*, pp. 239-248, 2000
- [6] Gino van den Bergen: *Collision Detection in Inractive 3D Environments*, Morgan Kaufmann Publishers, 2003
- [7] Brian Mirtich and John Canny: Impulse-based simulation of rigid bodies. In *Proceedings of Symposium on Interactive 3D Graphics*, 1995
- [8] David Baraff: *Physically Based Modeling*, ACM SIGGRAPH '99 Course Notes, 1999
- [9] David Baraff: Analytical methods for dynamic simulation of non-penetrating rigid bodies. In *Computer Graphics Proceedings*, Vol. 23, pp. 223-232. ACM SIGGRAPH, 1989
- [10] David Baraff: Fast contact force computation for non-penetrating rigid bodies. In *Computer Graphics Proceedings, Annual Conference Series*, pp. 23-34. ACM SIGGRAPH, 1994
- [11] 川地克明: 機械機構の挙動の直観的な把握を目的とした剛体シミュレーション手法, 東京大学工学部精密機械工学科 修士論文, 2001
- [12] 藤岡奨, 牛島省: LBM と DEM による任意形状流体を含む固液混相流れの数値計算法, 第 19 回数値流体力学シンポジウム講演論文集, CD-ROM, B4-4, 2005

[13] Parris K. Egbert and Scott H. Winkler: Collisionfree object movement using vector fields. *IEEE Computer Graphics and Application*, pp. 18–24, July 1996.

[14] 杉本剛志: 物理ベースCGにおける割れの表現手法, 東京大学大学院新領域創成科学研究科環境学専攻 修士論文, 2005

謝辞

本論文を作成するにあたり、ご指導いただいた東京大学大学院新領域創成科学研究科 鈴木克幸助教授に心から感謝いたします。毎週の輪講では数々の示唆および叱咤激励をいただき、適切な研究方針を持つことができました。

また、研究室の様々な仕事をサポートしてくださり、研究環境を整えていただいた技官の吉田二郎さん、秘書の大嶋さやかさんに心から感謝いたします。ありがとうございました。

博士課程3年の張愛年さん、環境海洋工学科博士課程3年の Shi Jiangshui さん、博士課程2年の Kirkayak Levent さんには研究室の大黒柱として様々なサポートをしていただきました。心から感謝いたします。

修士課程2年の岩田隆道君、中川玲君、高明さん、De Souza Vinicius Aguiar さんとは研究室の同僚として共に励ましあい、多くの時間を楽しく過ごせました。本当に心強く、楽しい仲間達でした。心から感謝いたします。これからの社会人生活もお互いがんばりましょう。Vinicius さんはこれからの研究がんばってください。

修士課程1年の有木俊博君、市川幸太君、伊藤陽介君には研究における様々な助言、サポートをいただいただけでなく、研究室のムードメーカーとして研究室全体を明るくしていただきました。心から感謝いたします。これからの研究および研究室生活の充実を祈っております。いつか研究室にお邪魔することもあるかと思いますが、こころよく迎えていただければ幸いです。

工学部システム創成学科4年の磯部信君、橘田尚明君、呉琛曦君とは研究に関する議論をするだけでなく、研究室での多くの時間を楽しく過ごすことができました。磯部君は本研究室での修士研究、橘田君、呉君は他の研究室での修士研究および院生生活の充実を祈っております。

元助手の澤村淳二さん、元学術研究支援員の中住昭吾さん、すでに研究室を卒業された嶋村寿文さんには研究に関しての助言ばかりでなく、研究室の先輩として数々のアドバイスをいただきました。心から感謝すると共にこれからのより一層のご活躍を願っております。

また、株式会社フィックスターズの皆様には宮本圭様をはじめとして研究に関しての数え切れないほどのサポートをいただきました。深く感謝いたします。

非常に不出来な学生で多くの方々に迷惑をかけたことと思いますが、皆様の支えのおかげで本論文を提出することができました。辛いと思ったことも多々ありますが、この経験は社会人になってもきっと活けると信じています。ありがとうございました。

最後に、陰ながら精神的に私を支えてくれた家族に感謝して、筆を置きたいと思います。