

3次元物体の衝突面の並列検出実験*

1E-8

二宮 茂樹 渋沢 進†
茨城大学工学部情報工学科‡

1 はじめに

3次元物体の衝突問題は、ロボティクス、コンピュータグラフィックス等多くの分野で中心的な問題の一つである[1][2]。正確に物体間の衝突を検出しようと考えるとき、例えば、作業空間中の物体の全ての面と稜線の組合せの交差を調べるといった方法が考えられる。しかしこの方法では、物体の数や物体の面数が多くなるにつれ、衝突面検出の計算量は大きく増加してしまう。

3次元物体の衝突面を検出する方法としては、離散時間の衝突検査で物体間の衝突面を実際に衝突が起きる直前に特定する衝突面検出アルゴリズムが開発されてきた[3]。また[3]では、共有メモリを用いて、並列アルゴリズムを実現している。

本研究では、3次元物体の衝突面検出を並列化し、分散メモリMIMD型並列計算機を用いて、衝突面検出実験を行った。並列化は、ホストプロセッサから各セルプロセッサに順次一定数の面ペアを送り、各プロセッサで送られてきた面ペアの衝突を並列に検査した。

2 衝突面検出のアルゴリズム

このアルゴリズムでは、3次元空間中の物体の衝突についてその衝突面を検査する。各物体は離散時刻毎に移動し、その移動量はあらかじめ分かっているとする。各stepで、検出すべき領域や面が見つからないときは、次の時刻へ進む。

step1—外接直方体の重なり領域の検出

各物体に対して外接直方体を生成し、各対の物体の重なり領域を検出する。

step2—重なり領域と交差する候補面の抽出

重なり領域を持つ対の物体に対し、各物体を構成している全ての面と重なり領域との交差を調べる。また、このとき交差する面を候補面とする。

step3—8分木を用いた面ペア検出

*Some Parallel Detection Experiments of Colliding Faces among 3D Objects

†Shigeki Nimiya, Susumu Shibusawa

‡Department of Computer and Information Sciences, Ibaraki University

候補面が作業空間中に存在する位置を、深さ l の8分木で表す。8分木のルートノードを作業空間全体とし、順次空間を8分割した木で表現する。また、最下層のノードの大きさは、面が一離散時間内に移動する量よりも十分大きいとする。8分木の構成を図1に示す。

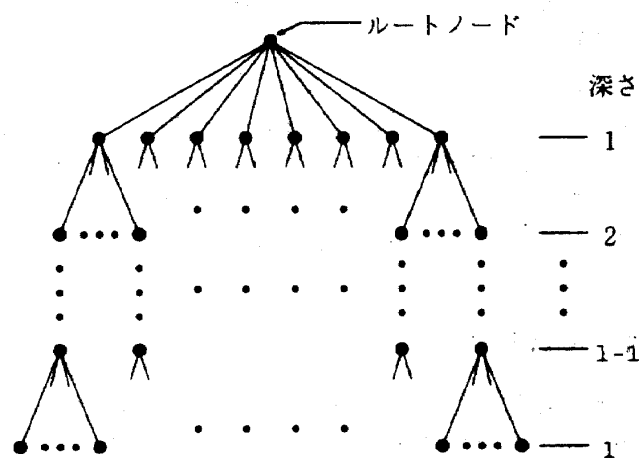


図1: 8分木の構成

8分木を用いて、候補面がどの空間に存在するかを調べる。最下層のノードにおいて、異なる物体からの面の組合せを面ペアとする。

step4—面ペアの衝突検査

各面ペアについて、現在の時刻を t_i とすると、時間 $[t_i, t_{i+1}]$ に各面によって構成される物体が交差していれば、この面ペアは衝突すると判断する。

3 逐次的な衝突面検出

ここでは、三角形で構成された面を持つ二つの球体が衝突する実験1と、二つのある領域内で生成した三角形の集合が衝突する実験2の衝突面検出を行った。球体の面数は162個、領域内の面数は100個である。また実験2については、重なり領域の大きさが異なる実験2(a)と実験2(b)を行った。実験2(a)の方が実験2(b)より重なり領域が大きい。また、今回の実験ではstep4まで処理が進む実験を行った。各実験の実行結果を表1に示す。実行時間は、3回測定した平均値である。表1から、step4が衝突面検出処理の全実行時間中で大きな比重を占めているので、step4を並列化する。

表 1: 逐次処理の実行結果 (実行時間:秒, 比率:%)

	実験1		実験2 (a)		実験2 (b)	
	実行時間	比率	実行時間	比率	実行時間	比率
step1	0.0119	1.3	0.0077	0.2	0.0077	0.4
step2	0.1077	12.2	0.0299	0.7	0.0343	1.6
step3	0.0745	8.5	0.0717	1.8	0.0648	3.1
step4	0.6839	78.0	3.9592	97.3	1.9870	94.9
計	0.8780	100.0	4.0885	100.0	2.0938	100.0

4 並列化

本実験では、分散メモリ MIMD 型並列計算機 AP1000 を用いた。AP1000 はセル数が 16 であり 8×2 の構成になっている。各セルは 25MHz の SPARC プロセッサで、メモリを 16Mbyte 持っている。

衝突面検出の step4 は、面ペア検査リストに含まれている面ペア一つ一つに同様の処理を行っている。そこで、並列化の方法として、次のような方法を用いた。まず、ホストが衝突面検出の step1~3 を実行し、step3 で得られた面ペア検査リストの中から、一定数ずつ各セルへ送る。そして、各セルでは送られてきた面ペアについて step4 を実行し、衝突すると判断された面ペアをホストへ返す。

5 実験結果

衝突面検出の step3 で得られた面ペアを $5 \cdot 10 \cdot 15 \cdot 20 \cdot 25 \cdot 30$ ずつ、各セルへ送ったときの実験 1 の実行結果を図 2 に示す。実験 2 の実験結果についても、実験 1 と同様な図が得られた。図中の send5~send30 は各セルへ一度に送る面ペア数が 5~30 のときの結果である。また実行時間は、3 回測定した平均値である。

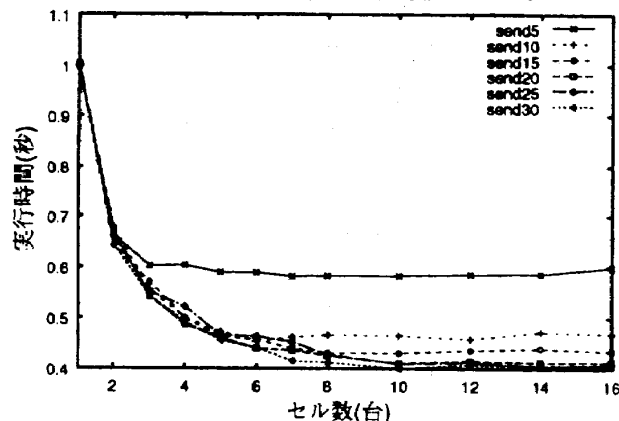


図 2: 実験 1 の実行時間とセル数の関係

図 2 をみると、それぞれセル数が増えると実行時間が減少するので、並列化の効果が現われている。send5 を見ると、セル数が 3 以上では実行時間は一定となる。これは、ホストが各セルへ 5 個ずつの面ペアを送る時間より、セルが 5 個の面ペアを処理する速度のほうが速くなり、セルが何もしない時間が生まれてしまうからだと考えられる。このことから、ある程度多くの面ペアを一度

に送った方がよいことが分かる。

図 2 から send5~send30 の最小の実行時間をみると、送るペア数が増えるにつれて最小の実行時間は減少しているが、その実行時間の減少率は段々小さくなっている。これは、実行時間の減少に限界があることを示す。またこのことは、実験 2 の結果についても言える。

6 考察

まず、並列処理と逐次処理の比較をする。表 2 に最も速い並列処理時間と逐次処理の実行時間の比較を示す。今回の並列実験では、最も効率の良いときで、逐次処理の 21.1% の実行時間で処理できた。

表 2: 逐次処理と並列処理の実行時間の比較

	実験1	実験2 (a)	実験2 (b)
逐次処理 (秒)	0.6839	3.9592	1.9870
並列処理 (秒)	0.4002	0.8374	0.6124
比率 (%)	58.5	21.1	30.8
使用セル数 (台)	10	14	16

次に、並列処理について評価する。並列実験の結果から、セルをある程度増やすと実行時間が一定となることが分かった。実行時間が一定となるときに最小のセル数を p_0 とする。 p_0 より少ないセル数ではプロセッサ数に反比例して実行時間が減少する。これより、実行時間 T には次の関係が成り立つ。

$$T = \begin{cases} c_0 + \frac{c_1}{p} & (p < p_0 \text{ のとき}) \\ c_2 & (p \geq p_0 \text{ のとき}) \end{cases}$$

ここで、 p はプロセッサ数、 c_0, c_1, c_2 は定数である。 $p = p_0$ のときは、一セルで面ペアを処理する時間と次の面ペアが送られるまでの時間が同じとなるときである。これは、最も効率の良い場合だと考えられる。

7 おわりに

本研究では、分散メモリ MIMD 型並列計算機 AP1000 を用いて、3次元物体の衝突面検出アルゴリズムの並列化を行った。今後の課題として、アルゴリズムの高速化、今回の実験結果の定式化などが挙げられる。

謝辞

本研究に際し、ご討論頂いた茨城大学工学部の研究室の皆様へ深く感謝致します。

参考文献

- [1] M. de Berg, M. van Kreveld, M. Overmars and O. Schvargkapf: Computational Geometry - Algorithms and Applications, Springer, 1997.
- [2] F.P. Preparata and M.I. Shamos (著), 浅野孝夫, 浅野哲夫 (訳): 計算幾何学入門, 3章, 総研出版, 1992.
- [3] 北村喜文, Andrew Smith, 竹村治雄, 岸野文郎: 並列計算機による 3次元物体の実時間衝突面検出, 信学論 (D-I), Vol. J78-D-I, No. 8, pp788-797, 1995.