

J-31 干渉判定のための多重解像度メッシュモデル生成に関する研究  
 - 局所解像度制御と干渉判定処理への適用 -

A Study of Multiresolution Mesh Model for Collision Detection  
 - Local LOD Control and its Application to Collision Detection -

伊達 宏昭† 金井 理† 岸浪 建史†  
 Hiroaki DATE Satoshi KANAI Takeshi KISHINAMI

1. はじめに

本研究では、干渉判定処理の幾何演算の効率化を目的として、干渉が起りにくい部分を優先的に簡略化し、かつ干渉を過小評価しないよう低解像度形状が高解像度形状を内部に包み込む性質を持つ解像度制御が可能な多重解像度メッシュモデルを生成する方法を提案してきた<sup>[1]</sup>。本報告では、提案した多重解像度メッシュモデルの局所解像度制御法と干渉判定処理への適用法について述べる。まず、効率的なメッシュ簡略化/詳細化(復元)のためのメッシュ階層性のデータ構造を述べ、干渉判定処理の速度、精度を連続的に制御可能にする為の多重解像度メッシュモデルの局所解像度制御手法を述べる。最後に、干渉判定処理への多重解像度メッシュの適用法と多重解像度メッシュを用いる利点を定性的に述べる。本研究の位置付けと役割を図1に示す。

2. 干渉判定用多重解像度メッシュ

解像度制御可能な多重解像度メッシュモデルを得るための、三角形メッシュモデルの多重解像度表現(低解像度メッシュと階層的な失われた情報)生成手法は、1) 解像度制御対象(幾何や属性)、2) 要素の削除・復元順序決定のための解像度制御基準、3) メッシュモデル内の要素削除・復元方法、4) 簡略化前後の形状間の幾何学的関係、5) 失われた情報の解釈の5つの観点から分類できる。本研究では、前報<sup>[1]</sup>において、この分類における表1に示す設定に基づき、頂点の近似曲率と形状膨張評価量を解像度制御基準として、干渉の起りにくい形状の凹部分を優先的に簡略化し、かつ、干渉を過小評価しないよう低解像度形状が高解像度形状を内部に包含する幾何学的関係を導入した三角形メッシュモデルの多重解像度表現生成手法を提案した。提案手法により生成される面分数の少ない低解像度メッシュを干渉判定処理に用いることにより、メッシュモデル間の干渉判定処理効率を向上できるが、更に、ユーザの要求や目的に応じて、干渉判定の精度や処理速度の制御ができることが望ましい。この要求に応えるためには、図2に示すように、干渉検出部の局所的なメッシュ詳細化や面分数の制御が必要である。次節では、これらを実現できる多重解像度メッシュの局所解像度制御手法を述べる。

3. 局所解像度制御手法

3.1 メッシュ階層の表現とデータ構造

低解像度メッシュモデルを得るためのメッシュ簡略化には、図3(a)に示す、メッシュモデルの稜線を構成している2

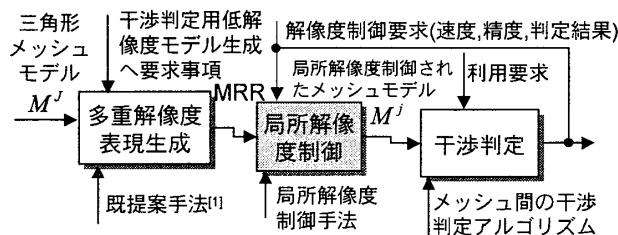


図1 本研究の位置付けと役割

表1 干渉判定用多重解像度メッシュ生成

| 対象 | 解像度制御基準     | 要素削除・復元方法     | 幾何学的関係    | 失われた情報 |
|----|-------------|---------------|-----------|--------|
| 幾何 | 近似曲率, 形状膨張小 | Edge collapse | 低解像度形状を内包 | ---    |

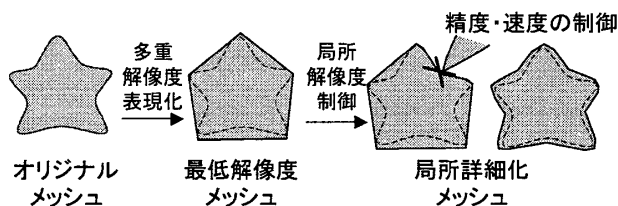


図2 多重解像度表現生成と局所解像度制御

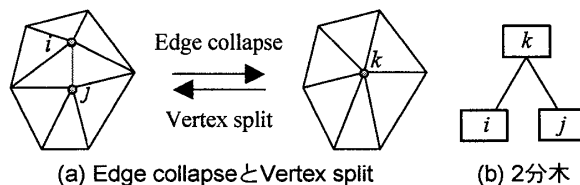


図3 要素削除・復元方法と2分木

頂点を1頂点に統合する edge collapse<sup>[2]</sup>を用いた。Edge collapse 前後の頂点の関係は、削除された頂点对を子ノード、新たな頂点を親ノードとして持つ2分木(図3(b))で表現でき、繰り返しの edge collapse の適用は、図4左に示す頂点の階層を生成する。この頂点の階層を表現するノードのデータ構造を図4右に示す。図4右において Coord は頂点座標値、On\_mesh は現行のメッシュ上の頂点(真)か否(偽)を表し、Parent, Child は木構造の親子、Joint は2つの子ノードの2頂点を含む面分のもう1つの頂点を表す。また、干渉判定精度の制御指標として、全ての子ノードの形状膨張評価量  $d_k^{[1]}$  の総和を属性 Exp\_eval として与えた。これらの構造・属性は多重解像度表現生成時に作成され、現行のメッシュモデルは On\_mesh が真の頂点集合(front ノード集合)から成る。

† 北海道大学大学院工学研究科, Graduate School of Engineering, Hokkaido University

3.2 局所解像度制御手法

高解像度メッシュを得るためのメッシュ詳細化は、front ノード集合に対する、詳細化が指定されたノードの削除と、その子ノードの追加により行われる。この際、子ノードの頂点のメッシュ上での復元には Joint が必要となる。もし Joint の On\_mesh が真でないならば、front ノード集合に含まれる Joint の先祖からノードを復元しながら木を下って Joint を復元し、その後、子ノードの頂点をメッシュ上で復元する。一方、低解像度メッシュを得るためのメッシュ簡略化は、front ノード集合に対して、簡略化が指定された兄弟のノード対を削除し、その親を追加することで行われる。

以上で述べたデータ構造と局所解像度制御手法を用いて以下の多重解像度メッシュの性質の制御が可能となる（ここでは最低解像度メッシュから開始するものと想定するが、簡略化も、ノードの Exp\_eval に対する判定、選択順序の条件を逆にすることにより行える）。

- ・形状近似精度制御： Exp\_eval に対する閾値を設定し、閾値以上の Exp\_eval を持つノードを Exp\_eval の大きな順に詳細化する。
- ・面分数制御： 1 レベルの詳細化は面分数を 1~2 増加させるため、要求する面分数になるまで Exp\_eval の大きなノードから順に詳細化する。

以上の制御は、制御の対象とするノード集合の選択により、大局的または局所的に行える。

3.3 局所解像度制御手法の適用例

図 5(a)に示すメッシュモデルに対し、[1]の多重解像度表現化を行った結果得られた最低解像度メッシュを図 5(b)に示す。図 5(b)中矢印に示す最低解像度メッシュ上の面分に対し干渉が起きたと想定し、その面分の近傍を局所的に詳細化した結果を図 5(c)(d)に示す。図 5(c)では Exp\_eval が最低解像度メッシュ上の頂点の 1/3 になるまで詳細化し、(d)では Exp\_eval が 0 となるまで詳細化した。干渉が起きたと指定された面分近傍のメッシュのみが詳細化されていることが確認できる。図 5(d)の例題において、メッシュ詳細化の処理時間は 1.5msec (P2-450MHz)であった。

4. 多重解像度メッシュの干渉判定処理への適用

本研究で提案する多重解像度メッシュと局所解像度制御手法は、Rigid body simulationなどで要求される動的な干渉判定処理に適用すると有効である。メッシュモデル間の干渉判定処理は、メッシュモデルを直接用いる方法と代表表現を用いて効率的に行う方法が挙げられる。これらの方法に対する多重解像度メッシュの適用法と利点を以下で述べる。

4.1 メッシュモデルを直接用いた干渉判定処理

メッシュモデル間の干渉判定処理の最も単純な方法は、メッシュモデル間の面分に対し総当り的に干渉を検出する方法である。この場合、干渉判定処理は各モデルの面分数に比例した時間がかかる。提案手法を用いることにより、まず、低解像度メッシュを用いて初期の干渉判定を行い、干渉が検出された場合に、必要に応じて、局所解像度制御を用いた段階的な干渉判定を行うことが可能となる。面分数が減少した低解像度メッシュの利用は、干渉判定の処理時間を短縮でき、更に、処理時間は、面分数制御法により制御できる。低解像度メッシュで干渉が検出された場合、高解像度のメッシュ上における干渉箇所は、干渉が検出された面分を構成する頂点

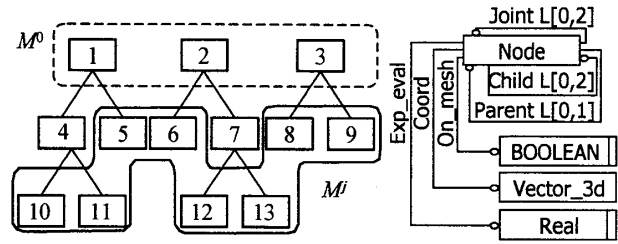
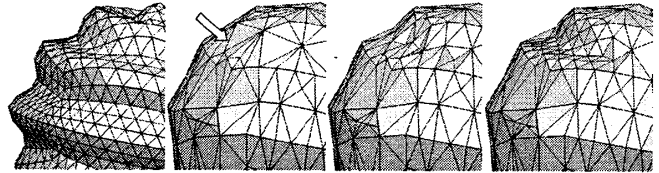


図4 頂点の階層とノードのデータ構造



(a) オリジナルメッシュ (#T=2048) (b) 最低解像度メッシュ (#T=596) (c) 局所解像度制御例1 (#T=618) (d) 局所解像度制御例2 (#T=646)

図5 局所解像度制御の例 (#T: 面分数)

の子孫からなる面分に限定される。よって、干渉が検出された面分の頂点とその子孫に対する形状近似精度制御法の適用と、詳細化のレベルの制限により、精度と速度を制御可能な面分を絞り込んだ段階的な干渉判定ができる。

4.2 代表表現を用いた干渉判定処理

代表表現を用いた干渉判定方法では、まず、高速な干渉判定処理が可能な Bounding Box<sup>[3]</sup>や球<sup>[4]</sup>等の代表表現を形状モデルに基づいて階層的に生成する。干渉判定は、最初、代表表現を用いて行い、最終的には、絞り込まれた代表表現に含まれる面分を用いて干渉判定を行う。代表表現を用いた干渉判定の処理時間Tは次式で評価される<sup>[3]</sup>。

$$T = t_v \times C_v + t_p \times C_p \tag{1}$$

ここで、 $t_v$ と $C_v$ は、各々、代表表現間の干渉の1検査にかかる時間と検査回数、 $t_p$ と $C_p$ は、各々、面分間の干渉の1検査にかかる時間と検査回数である。この方法においては、低解像度メッシュを入力として用いることで次の利点を得られる：代表表現内の要素数を一定にした場合の代表表現数の削減( $C_v$ の減少)、代表表現内の面分数削減( $C_p$ の減少)、代表表現作成時間の短縮。また、最終的な干渉判定処理は面分間で行われるため、4.1節で述べた利点も得られる。

5. おわりに

本報告では、干渉判定用多重解像度メッシュに対する階層的データ構造と局所解像度制御手法を提案し、提案法の干渉判定処理への適用法と定性的な利点を述べた。提案手法に基づく干渉判定システムの開発と評価が課題である。

【参考文献】

[1] 伊達他: 干渉判定のための多重解像度メッシュモデル生成に関する研究, 情報処理学会第 64 回全国大会講演論文集, n.4, pp. 33-34, 2002  
 [2] H. Hoppe: Progressive Meshes, Computer Graphics (SIGGRAPH'96), pp. 98-108, 1996  
 [3] S. Gottschalk et al.: OBBTree: A hierarchical structure for rapid interference detection, Computer Graphics (SIGGRAPH'96), pp.171-180, 1996  
 [4] P. Francis et al.: Automatic Generation of Sphere Hierarchies from CAD Data, Proc. of International Conference on Robotics and Automation 1998