

境界箱とオフセットを用いたポリゴンメッシュ干渉検出

5U-07

新垣 恒治 松岡 司

株式会社リコー ソフトウェア研究所

1 はじめに

近年、CAD/CG システム等の普及により 3 次元形状データの利用者層が拡大するとともに、シミュレーションや組み立て性の検証など様々な局面で 3 次元形状データを有効活用する機会が増えた。中でも 3 次元モデルのビューワーは、モデルを様々な方向から眺めるといった基本的な機能に加えて、組み立て工程のシミュレーションや検証等の様々な機能を有し、その利便性から需要が高まりつつある。

アセンブリを構成する複数のモデル間の干渉を検出する機能は、設計ミスや入力ミスを検出するために有用な機能であるが、干渉検出機能をビューワーにおいて実現する場合、その検出結果における信頼性を確保するにはデータに含まれる誤差に起因する誤判定が障害となる。ビューワーが利用するデータは、多くの場合 CAD システムの自由曲面データから近似生成されたポリゴンメッシュであるため、誤った判定結果が得られてしまう可能性がある。

また、ポリゴンメッシュで表現される 3 次元形状の干渉を検出する場合、総当たりによる単純な方法では膨大な計算量が必要となり、実用的な処理速度を得るのは困難である。例えばポリゴン数 m のモデル n 個から成るアセンブリの場合、アセンブリから 2 モデルを抽出するすべての組み合わせにおいてポリゴン総当たりの干渉検出を行う回数は ${}_n C_2 \times m^2$ となる。

2 干渉検出における誤判定

自由曲面データから近似生成されたポリゴンメッシュを用いて干渉状態を検出する場合、誤判定が生ずる可能性がある。例えば、凹領域のポリゴンメッシュはモデルの外側にあるため、モデルが干渉していなくてもポリゴンメッシュが干渉することがある (図 1(a))。

また、ビューワーが表示データとして利用するポリゴンメッシュの頂点座標精度は圧縮により低下するため、誤判定の生じる可能性はさらに増加する。この結果、大規模アセンブリでは検出数が膨大となり、設計ミス等による干渉状態との判別には時間がかかる。

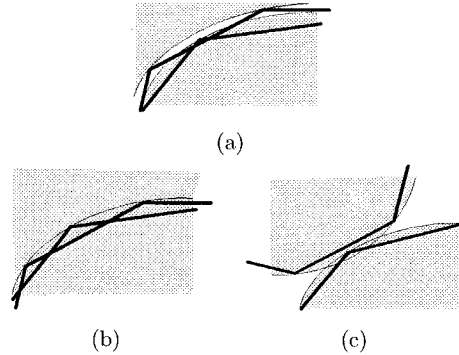


図 1: ポリゴン化誤差による誤判定 (太線はポリゴンメッシュ、細線はモデルの境界を示す)

3 頂点オフセットによる誤判定防止

ポリゴンメッシュを構成する個々のポリゴンの干渉を検出する前処理として、メッシュ頂点を形状の内側、すなわち法線ベクトルの方向とは逆の方向へ移動する処理を導入し、微妙な接触状態 (図 1(b),(c)) を除外して干渉状態のみ検出するように構成する。一般に CAD システムにおける設計ミスや入力ミスは、接触という微妙なケースではなく完全な干渉状態となって現れることが多く、微妙な接触状態及び軽度な干渉状態を除外しても実用的な結果を得ることができる。

移動後の頂点座標は、メッシュの近似誤差 ε と圧縮のための座標誤差 δ 、対応する法線ベクトル \vec{n} から算出する。 ε はメッシュ生成時に指定される近似誤差であり、近似対象とメッシュにより表現される形状との間の最大誤差により定義される。

$$\vec{P}' = \vec{P} - (\varepsilon + \delta)\vec{n}$$

4 境界箱による干渉計算の高速化

ポリゴン間の干渉検出に及ぶ前に計算コストの少ないラフチェックを実施し、干渉の可能性のないケースを可能な限り除外することにより、処理時間の短縮を図る。一般に、ポリゴン間の干渉は以下に示す方法で検出する。

step1 一方のポリゴン (以降 a) を構成する頂点が、他方のポ

リゴン (以降 b) の表側に存在するか、裏側に存在するか調べる。 b を構成する 1 頂点から a の頂点へ向かうベクトルと、 b の法線ベクトルとの内積がすべて正 (表側に存在する) か、もしくはすべて負 (裏側に存在する) であれば、ポリゴン同士は干渉しない。

- step2 step1 にて a と b の関係を入れ替えて判定を行う。
- step3 ポリゴンの表側と裏側に存在する頂点について、その頂点間の線分と平面との干渉点を求める。
- step4 step3 にて求めた干渉点がポリゴン内に存在するか否かを判定する。干渉点からポリゴンの各頂点へ向かうそれぞれのベクトルと、ポリゴンの法線ベクトルとの内積により判定される。

この計算回数を、事前のラフチェックにより減らすことにより、全体の処理速度を低減する。ラフチェックは、モデル → 面 → ポリゴン の順に、その境界箱の干渉の検出を行うことにより実現する。形状を構成する単位の大きい順にラフチェックを行うことにより、モデルが干渉しない場合を早い段階で除外できる。なお、多くの CAD システムではモデルや面といった形状要素単位の境界箱を取得する API が提供されるので、これらの境界箱をビューワーが利用するデータに含めて利用できる。

5 結果

規模の異なる 2 つのアセンブリ A (ポリゴン数 1,175,476、モデル数 1,112)、 B (ポリゴン数 119,893、モデル数 75) について、頂点オフセット導入前及び導入後の干渉しているモデルペアの検出数を表 1 に示す。なお、表中の CAD の項は、CAD システムにて提供されるモデル間の干渉検出機能による結果である。

アセンブリ A における干渉検出数は 1017 件から 318 件へ減少しており、699 件の誤判定が除外される効果が見られた。また、アセンブリ B における導入後の検出数は CAD システム上の検出数を下回るが、その差分は頂点オフセットにより除外した微妙な接触状態及び軽度の干渉状態であることを確認した。

表 1: 干渉モデルペア検出数

	導入前	導入後	CAD
A	1017	318	266
B	61	36	50

表 2 は干渉しているモデルペアの検出に要した時間 (単位: 秒) を計測した結果である。また、表 3 は表 2 の B のケースにおいてポリゴン間の干渉検出が行われた回数を示す。境界箱を用いたラフチェックにより、ポリゴン間干渉検出の計算回数が削減され、ラフチェックを全く実施しない場合に比べて約 $\frac{1}{100}$ 以下に処理時間が短縮された。

表 2: 処理時間

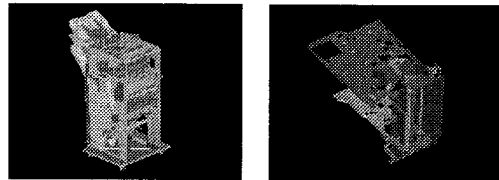
	(1)	(2)	(3)	(4)	CAD
A	28,851	2,875	736	21	218
A'	32,027	3,283	985	28	—
B	159	117	38	1.6	52
B'	171	128	47	1.8	—

$A', B' \dots A, B$ それぞれに対し、頂点オフセットを導入した場合

- (1)... 境界箱によるラフチェック未実施
 - (2)... モデル境界箱によるラフチェックのみ実施
 - (3)... モデル及び面の境界箱によるラフチェックを実施
 - (4)... モデル、面及びポリゴンの境界箱によるラフチェックを実施
- CAD... CAD システムにてモデル間干渉検出を実施

表 3: ポリゴン間干渉検出の計算回数

	(1)	(2)	(3)	(4)
B	81,100,723	62,952,912	22,787,321	8,106



(A) (B)

図 2: サンプルデータ

6 おわりに

頂点オフセットにより誤判定の問題を解決し、さらにポリゴン間の干渉検出を実施する前に境界箱を用いたラフチェックを行うことにより全体の処理を高速化したポリゴンモデル間の干渉検出方法を提案した。

本報告で提案した方法はモデル間の干渉をポリゴンメッシュの干渉により検出するものであり、モデルが他のモデルの内部に内包されるような干渉状態を検出できない。今後の課題としては、モデルの完全な包含関係も含めた干渉状態を検出可能にする拡張が考えられる。

参考文献

- [1] 鳥谷 浩志 and 千代倉弘明. 3次元 CAD の基礎と応用. 共立出版株式会社, 1991.