

Development of tool path generation lower calculation cost: Simplifying of 3D model by QEM method

メタデータ	言語: jpn 出版者: 公開日: 2017-10-03 キーワード (Ja): キーワード (En): 作成者: メールアドレス: 所属:
URL	https://doi.org/10.24517/00009108

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 International License.



工具経路生成時間短縮法の開発 — QEM 法による 3D モデルの簡略化 —

金沢大学 ○鬼頭亮太, 金沢工業大学 高杉敬吾, 金沢大学 浅川直紀, 岡田将人

Development of tool path generation lower calculation cost -Simplifying of 3D model by QEM method-

Kanazawa university Ryota KITO, Kanazawa Institute of Technology Keigo TAKASUGI

Kanazawa university Naoki ASAKAWA and Masato OKADA

This study deals with a method to reduce calculation costs of a tool path generation. In the recent years, CAM applications have been becoming more common. Since collision avoidance of a workpiece and a tool is essential in CAM applications, it is an important problem to reduce calculation costs. Therefore, we propose a new method of reducing calculation costs, which translates the workpiece represented by a continuous surface to discrete mesh representation. Moreover, the translated mesh model is simplified with QEM method while keeping the characteristics of the model. In this report, the new collision detection method is implemented and confirmed.

1. はじめに

近年, 航空機や自動車, 家電製品などの加工において, CAM ソフトウェアは欠かせないものになってきている. CAM において, 複雑な 3D モデルの工具経路生成には, 工作物と工具の干渉検出は必須であり, その計算時間の短縮は CAM の開発において重要である. しかし, 既存の CAM では, 干渉検出に依然多くの時間を要しているのが現状である.

そこで本研究では, 干渉検出にかかる計算時間短縮へのアプローチとして, 3D モデルの表現方法に着目した. 一般的に, 3D モデルを構成する自由曲面は B-Spline や NURBS などのパラメトリックな幾何表現手法が用いられているが, 複雑な曲面になればなるほどその曲面を表現するために必要な次数や制御点数は大きくなり, 幾何計算に多くの演算リソースを要することになる. このとき, 加工点の生成に関しては, なるべく誤差の少ない幾何計算が望ましいことから, パラメトリック表現をそのまま用いる必要があるが, 干渉検出に対しては, 加工点生成ほどの厳密さは必要ない. そこで本報では, 3D モデルを可能な限り簡略化することによって干渉検出に必要な幾何計算コストを減らすことを試みた. 具体的には, 次に示す 2 つの方法を提案, 実装し, 有効性の確認を行った.

1 つ目の方法はコントロールポイントの投影による NURBS 曲面の簡略化である. これは NURBS 曲面上にコントロールポイントを投影することによって, 階数を小さくしても, 元の形状に近づけることができる. 2 つ目の方法は, 3D モデルをメッシュを用いて表現する. 大量の点データから構築されたメッシュに対して特徴を維持したままメッシュ数を減らす方法である QEM 法^[1]を適応し, 3D モデルの簡略化を行う.

2. 開発環境

本提案手法の開発には, CPU : Intel Core i5 3570, メモリ : 8GB, OS : Microsoft Windows 7 の PC を用いた. また, 開発環境は, 開発言語 : C/C++, 統合開発環境 : WideStudio v3.98-7, コンパイラ : gcc3.4.5, グラフィックライブラリ : mesa3.1 である.

3. Kodatuno

本提案手法の開発にあたり, オープン CAM カーネル "Kodatuno"^[2]を用いて開発を行う. Kodatuno は本研究室で開発中のオープンソースのサーフェスカーネルである. また, 3D モデルの NURBS 表現, メッシュ表現が可能であり, オープンソースであるため, 3D モデルデータの保存構造がオープンで, 直接保存構造に手を加えることによって, 3D モデルの簡略化を簡単に行うことができる.

4. 3D モデル簡略化方法

4.1 コントロールポイントの投影

まず, 3D モデルの簡略化方法として, NURBS 曲面の階数を下げることを考える. 階数とは, NURBS 曲線, 曲面の方程式の次数に 1 を足したものである. この階数が大きいほど, 曲線, 曲面はより滑らかな表現が可能となるが, 計算時間は大きくなる.

逆に階数を下げると計算コストは減少するものの図 1 に示すように実形状との誤差は大きくなる. そこで, 図 2 に示すように, NURBS 曲面を構成するコントロールポイントをその曲面上に投影することによって, 新しいコントロールポイントを生成し, 階数 2 の NURBS 曲線, 曲面を生成する. 階数が 2 の場合, NURBS 曲面の次数は 1 であるから, 曲面は折れ面となり, 全てのコントロールポイントを通る. 図 3(a) に元の 3D モデル (階数 4), (b) にコントロールポイントを投影し, 階数 2 の NURBS 曲面を生成した結果, (c) にコントロールポイントを投影せずに階数 2 の NURBS 曲面を生成した結果を示す. このように, (b) の結果は (c) の結果に比べ, 元の形状に近いことがわかる.

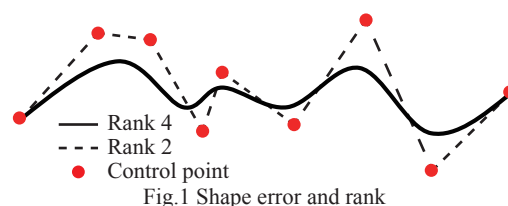


Fig.1 Shape error and rank

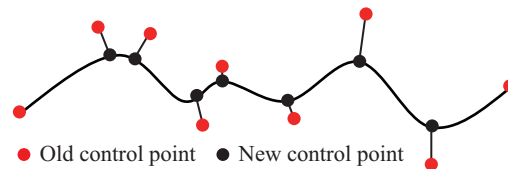


Fig.2 Generation of control points

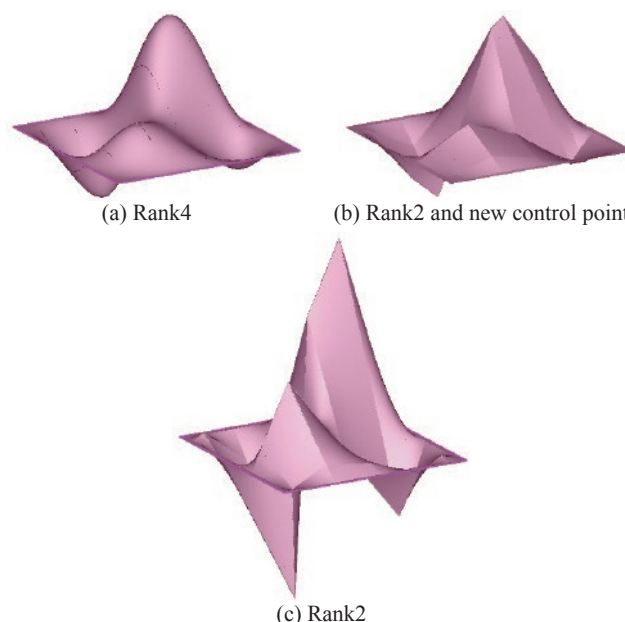


Fig.3 Relation between rank and shape

4.2 QEM 法

1つの三角形を含む平面 f と任意の点 $\mathbf{p}$ の距離は、平面 f の法線ベクトル $\mathbf{n}$ を用いて、 $|\mathbf{n}^T \mathbf{p} + d|$ で表される。頂点 $\mathbf{v}$ 回りの面に対するこの距離の自乗を集めたものをその頂点のQEMと

$$Q^v(\mathbf{v}) = \sum_{f \in \mathbf{v}} \text{area}(f) |\mathbf{n}_f^T \mathbf{v} + d_f|^2$$

$$= \mathbf{v}^T \mathbf{A} \mathbf{v} + 2\mathbf{b}^T \mathbf{v} + c \quad (1)$$

して定義している。よって、

となり、ここで、 $\mathbf{A}$ は 3×3 の対称行列、 $\mathbf{b}$ は列ベクトル、 c はスカラーであり、 $\text{area}(f)$ は三角形の面積による重み付けである。各頂点 $\mathbf{v}$ は $Q^v=(\mathbf{A}, \mathbf{b}, c)$ を構成する $6+3+1=10$ 個の値が保存され、これがメッシュの形状を代わりに表現する^[1]。この Q^v が小さい場合その頂点はモデルの形状に影響が少ない。エッジを消去する場合、エッジの始点の Q^{start} と終点の Q^{finish} を加算したものが小さい順に消去を行い、それを加算したものを新しい頂点に引き継ぐ^[1]。

本研究では、3DモデルをNURBS曲面で表現するのではなく、メッシュを用いて表現を行い、QEM法を用いてエッジを削除し、指定した任意の面数になるまでエッジを消去する。例として、図4(a)に元の3Dモデル（メッシュ数4926）、(b)にQEM法を用いて面数を減らした3Dモデル（メッシュ数100）を示す。このように、任意のメッシュ数に変更することができ、メッシュを多く削除しても、元の3Dモデルの形状を模している。

5. 干渉検出計算時間比較

比較を行う方法として、工作物、工具どちらもNURBS曲面で表現されているモデルのある任意の点での干渉検出を行い、この干渉検出の計算時間を計測する。また、コントロールポイントを投影によって3Dモデルを簡略化、QEM法によって3Dモデルを簡略化を行ったモデルに対して、同じ点で干渉検出を行う。この干渉検出にのみ要した時間を計測し、従来の計算時間と比較を行う。干渉検出は、任意の加工点を5度ずつ x 軸回り、 y 軸周りに回転させ、姿勢を変化させる。また、干渉検出はNURBS曲面の場合、工作物と工具に含まれる全ての面同士の干渉検出計算しており、メッシュにおいても、工作物、工具全てのメッシュの組み合わせにおいて、干渉検出を行なっている。また、どちらの場合も1つの姿勢において、干渉が検出されるとその姿勢においては計算を終了する。

今回干渉検出を行う工作物、工具をNURBS曲面で表示したものを図5に示す。工作物は2枚羽のインペラを模した形状であり、工具はボールエンドミル形状である。図6にコントロールポイントの投影を行い階数2のNURBS曲面を生成した工作物を示す。図7にQEM法を用いてメッシュ数を減らした工作物、工具を示す。元のメッシュ数は工作物が3168、工具が770であり、QEM法適応後が工作物が250、工具が25である。

図8に干渉検出を行った結果を示す。この干渉検出に要した時間は、工具、工作物がNURBS曲面の場合1.91秒であり、工作物のみコントロールポイントを投影し、階数2のNURBS曲面にした場合2.07秒、QEM法適応前のモデルでは、7.62秒要したが、QEM適応後では、1.56秒となった。

6. 終わりに

2つの方法を用いて、3Dモデルの形状を簡略化し、干渉検出計算時間の計測を行った。コントロールポイントの投影による簡略化では、計算時間の短縮は見られなかったが、QEM法を用いた簡略化では、計算時間が簡略化前より、18%短縮することができた。しかし、簡略化した3Dモデルは、干渉に関して安全側に変化しているとは言えない。よって、計算時間は短くなったが、干渉検出の精度は不十分であると言える。

7. 今後の展望

今後は、簡略化されたメッシュと元の形状間をメッシュ数で段階的に表現する手法である多重解像度表現を用いることで干渉回避の精度を改善しようと考えている。

参考文献

- [1] 金井 崇, "多重解像度メッシュの生成・編集処理技術", 情報処理学会誌, Vol. 41, No. 10 (2000), pp. 1108-1112
- [2] K. TAKASUGI et al, Development of Platform-Independent Open CAM Kerne, Proc. of the 6th Int. Conf. on Leading Edge Manufacturing in 21th Century, Vol.1, (2011)

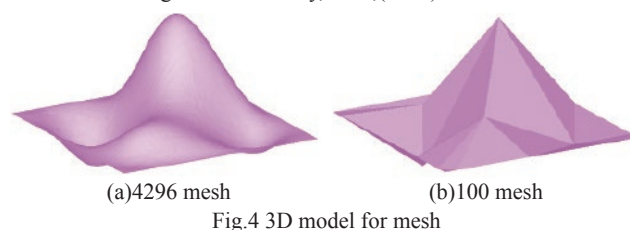


Fig.4 3D model for mesh

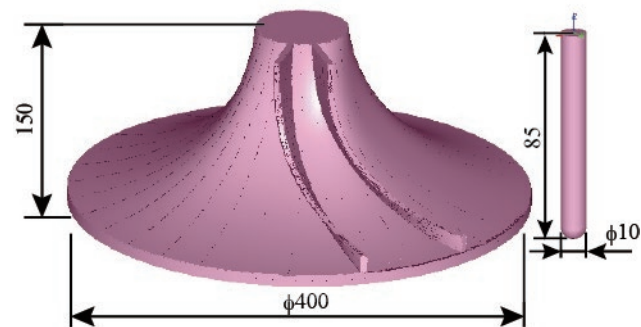


Fig.5 Modeled workpiece and tool



Fig.6 Simplification by control point projection



Fig.7 Simplification by mesh

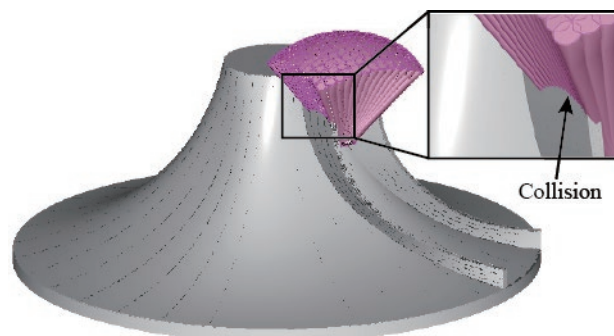


Fig.8 Collision check