

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2009-20671
(P2009-20671A)

(43) 公開日 平成21年1月29日(2009.1.29)

(51) Int.Cl.	F I	テーマコード (参考)
G05B 19/4093 (2006.01)	G05B 19/4093 F	3C269
G05B 19/4097 (2006.01)	G05B 19/4097 C	
B23Q 15/00 (2006.01)	B23Q 15/00 3O1K	
	B23Q 15/00 3O1H	

審査請求 未請求 請求項の数 8 O L (全 16 頁)

(21) 出願番号	特願2007-182243 (P2007-182243)	(71) 出願人	504190548 国立大学法人埼玉大学 埼玉県さいたま市桜区下大久保255
(22) 出願日	平成19年7月11日(2007.7.11)	(74) 代理人	100100918 弁理士 大橋 公治
		(74) 代理人	100108729 弁理士 林 紘樹
		(72) 発明者	金子 順一 埼玉県さいたま市桜区下大久保255 国立大学法人埼玉大学内
		(72) 発明者	堀尾 健一郎 埼玉県さいたま市桜区下大久保255 国立大学法人埼玉大学内
		Fターム(参考)	3C269 AB05 BB03 BB08 CC01 EF63 EF71

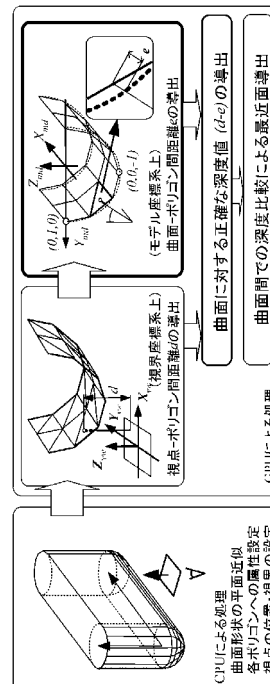
(54) 【発明の名称】 曲面形状と視線との交差座標導出方法

(57) 【要約】

【課題】 曲面と近似多面体との形状の隔たりを補正する曲面形状と視線との交差座標導出方法を提供する。

【解決手段】 モデル座標系上で曲面形状を近似した多面体の各ポリゴンの頂点座標を求め、ポリゴンの各頂点の属性値として、頂点座標と、曲面形状を表すパラメータとを設定する。次いで、視座標系上でポリゴンの各頂点の頂点座標を導出するステップと、視座標系の各視線とポリゴンとの交点座標を導出し、交点座標までの深度値dを算出すると共に、交点座標をポリゴンの各頂点の頂点座標で表すための内挿変数を導出するステップと、内挿変数を用いてモデル座標系上でポリゴンと各視線の交点を導出し、パラメータを用いて交点と曲面形状との視線方向の隔たりeを算出するステップと、距離eに基づいて深度値dを補正するステップとの処理をグラフィックスハードウェア内で行う。視座標系上での曲面と視線との交差座標を正確かつ迅速に求められる。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

曲面形状のトポロジを定義するモデル座標系上で、前記曲面形状を近似した多面体の各ポリゴンの頂点座標を求め、前記ポリゴンの各頂点の属性値として、前記頂点座標と、当該ポリゴンの各頂点に接する曲面形状を表すパラメータとを設定する第 1 のステップと、

基準面上の格子点ごとに設定された垂線を視線とする視界座標系上に前記多面体を移して前記ポリゴンの各頂点の頂点座標を導出する第 2 のステップと、

前記視界座標系の各視線と前記ポリゴンとの交点座標を導出し、前記基準面から前記交点座標までの各視線の距離を示す深度値 d を算出するとともに、前記交点座標を当該ポリゴンの各頂点の頂点座標で表すための内挿変数を導出する第 3 のステップと、

前記内挿変数を用いて前記モデル座標系上の前記ポリゴンにおける各視線の交点を導出する第 4 のステップと、

前記モデル座標系上で、前記パラメータを用いて、前記交点から前記視線の向きにある前記曲面形状までの距離 e を算出する第 5 のステップと、

前記距離 e に基づいて、前記視界座標系上で求めた各視線の前記深度値 d を補正する第 6 のステップと、

を備え、前記第 2 のステップから第 6 のステップまでの処理を、グラフィックハードウェアを用いて行うことを特徴とする曲面形状と視線との交差座標導出方法。

【請求項 2】

請求項 1 に記載の交差座標導出方法であって、前記曲面形状が、数式で定義可能な形状であることを特徴とする曲面形状と視線との交差座標導出方法。

【請求項 3】

請求項 1 に記載の交差座標導出方法であって、前記曲面形状が、円筒形状または球形状であることを特徴とする曲面形状と視線との交差座標導出方法。

【請求項 4】

請求項 1 に記載の交差座標導出方法であって、前記多面体を前記モデル座標系の単位座標系上で定義し、前記単位座標系上で求めた前記距離 e を基に前記視界座標系上の前記深度値 d の誤差を算出して、前記深度値 d を補正することを特徴とする曲面形状と視線との交差座標導出方法。

【請求項 5】

請求項 1 に記載の交差座標導出方法であって、前記第 2 のステップで、拡大・縮小、並行移動、回転または透視変換による変換を与えた前記ポリゴンの各頂点の頂点座標を導出することを特徴とする曲面形状と視線との交差座標導出方法。

【請求項 6】

請求項 1 に記載の交差座標導出方法であって、前記第 1 のステップから第 6 のステップまでの処理を、前記曲面形状を近似する前記多面体の全てのポリゴンと交差する全ての視線に対して行い、各視線の補正後の前記深度値 d で表される面の包絡面形状を導出することを特徴とする曲面形状と視線との交差座標導出方法。

【請求項 7】

請求項 6 に記載の交差座標導出方法であって、前記包絡面形状として、ボールエンドミル工具の掃引形状を導出することを特徴とする曲面形状と視線との交差座標導出方法。

【請求項 8】

請求項 6 に記載の交差座標導出方法であって、前記包絡面形状として、製品形状の上で工具の向きを反転させた逆オフセット形状を導出することを特徴とする曲面形状と視線との交差座標導出方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、視界座標系の上で、例えば、工作機械の工具で削られた被削物の曲面形状と視線とが交差する座標を求める交差座標導出方法に関し、特に、曲面形状を多面体で近似

10

20

30

40

50

したときの誤差が補正できるようにしたものである。

【背景技術】

【0002】

近年、大規模かつ複雑な金型等の製造には、膨大な工具経路情報を伴うNC加工が多用されている。これらの加工の実施に際しては、計算機による被削物形状及びオフセット形状の推定が工具経路の生成・検証に非常に大きな役割を占める。

この場合、図6に示すように、Z-map法では、基準面（Z-map基準面）上の格子点ごとに設定された垂線を視線と見立てた視界座標系の上で、被削物形状等の曲面形状と視線との交点と、基準面との間の距離（深度）を求める処理が行われる。

例えば、被削物の形状を推定して被削物の削り残しや削り過ぎを評価する切削シミュレーションでは、運動ステップごとに工具が通過した領域を工具掃引形状として設定し、それらのうち最も基準面に近い包絡面を求めることによって被削物に与えられる2.5次元形状の推定を実現している。

【0003】

また、逆オフセット法による工具経路の生成においても、図7(a)に示すように、目標形状上に、工具軸の方向を反転させた工具形状を多数配置し、図7(b)に示すように、これらの包絡面をZ-map上の点群として離散的に求め、これらの点群を補間することによって工具中心の移動経路を生成している。

これらの処理では、工具掃引形状や工具反転形状を工具運動ステップごとに定義し、それらの中から最も格子点に近い位置を表面形状として求めている。しかし、これは工具運動ステップ数の増加や格子点間隔の縮小に応じて大規模な幾何計算を必要とするため、処理の高速化が強く求められてきた。

【0004】

そこで、近年、この幾何演算を並行投影されたコンピュータグラフィックスの描画過程に置き換え、描画専用の機器であるグラフィックスハードウェアの能力を利用することで処理を高速化する手法が提案されている（下記特許文献1参照）。

【0005】

図10に示すように、グラフィックスハードウェア44は、3DCGの描画時に行われる幾何演算のうちCPU43ではリアルタイムな処理が難しいとされる頂点座標変換を実施するVertex Shader、及び深度計算・陰面処理等を実施するPixel Shaderを備えており、各Shaderの大規模な並列化により大幅な描画速度の向上が実現されている。

また、近年では特に、グラフィックスハードウェア内の描画機能をユーザーが自作のプログラムから自由に変更し、実行することが可能なプログラマブルシェーダが実用化されつつある。このような最新のグラフィックスハードウェアは描画に関する汎用的な幾何処理が可能であることから特にGPU(Graphics Processing Unit)あるいはVPU(Visual Processing Unit)と呼ばれている。

【0006】

グラフィックスハードウェアを用いた最近点座標の導出では、格子点を出力画面の画素、基準面から交点までの距離を各画素のもつ深度とみなす。評価を行う範囲の変更や描画対象物の変形をポリゴン頂点の座標変換機能を用いて行い、各視線と面との最近交点導出を陰面処理・深度計算の機能を用いて実現している。これらの機能はグラフィックスハードウェア内で多数の実行ユニットによって並列に処理されることから、CPUを用いた従来の計算手法に比べて非常に短時間での処理が可能となっている。

【特許文献1】特開2000-235407号公報

【発明の開示】

【発明が解決しようとする課題】

【0007】

しかし、現状のグラフィックスハードウェアでは、描画の対象を平面からなる多面体に近似する必要があるが、一般に、工具掃引形状及び逆オフセット形状は円筒面と球面等の

10

20

30

40

50

曲面の集合体となるため、これを多面体で近似すると本来の曲面との間にずれが生じ、計算誤差が発生する。

図8は、平面近似で発生する誤差の例を示している。ボールエンドミルの工具掃引形状は、工具経路の始点及び終点に対応する半球状の曲面と、始点・終点間の軌跡に対応する半円筒状の曲面を有する。これらは、いずれも数学的に容易に定義を行うことが可能であるが、多面体によって近似した場合には、内接する位置以外のいずれの箇所においても評価誤差が生じる。

また、工具の逆オフセット形状においては、円周上の工具刃部分が半球状の形状で、側面部の工具刃部分が部分円錐状の形状で表面が構成され、同様に多面体に近似したときに評価誤差が生じる。

10

【0008】

また、図9はX軸に対して並行に配置された半径1の正規化された円筒形状と、それに等角度で内接するように設定された近似多面体とのZ方向の深度値の隔たり e_z のY座標に対する変化を、円筒形への内接点間の角度 n で表したものである。円筒形状の半径に対する誤差の比は、近似に用いる微少多角形の総数に反比例して減少する。そのため、特に円筒の側面部分において誤差を小さくするためには非常に多数のポリゴンが必要となる。

これらの問題への対策としては、誤差がある閾値以下となるように、面を非常に微小な多面体の集合で近似する方法がある。しかしながら、高精度な評価を行うために必要となる多面体の総数は閾値に対して指数的に増大する。そのため、大規模なNCプログラムの検証にこの種の対策を取るのには現実的でない。

20

【0009】

本発明は、こうした事情を考慮して創案したものであり、ボールエンドミルを用いた3軸制御加工を対象として、近似時に生じた曲面と多面体との形状の隔たりをグラフィックハードウェア内で補正することができる曲面形状と視線との交差座標導出方法を提供することを目的としている。

【課題を解決するための手段】

【0010】

本発明の曲面形状と視線との交差座標導出方法は、曲面形状のトポロジを定義するモデル座標系上で、前記曲面形状を近似した多面体の各ポリゴンの頂点座標を求め、前記ポリゴンの各頂点の属性値として、前記頂点座標と、当該ポリゴンの各頂点に接する曲面形状を表すパラメータとを設定する第1のステップと、基準面上の格子点ごとに設定された垂線を視線とする視界座標系上に前記多面体を移して前記ポリゴンの各頂点の頂点座標を導出する第2のステップと、前記視界座標系の各視線と前記ポリゴンとの交点座標を導出し、前記基準面から前記交点座標までの各視線の距離を示す深度値 d を算出するとともに、前記交点座標を当該ポリゴンの各頂点の頂点座標で表すための内挿変数を導出する第3のステップと、前記内挿変数を用いて前記モデル座標系上の前記ポリゴンにおける各視線の交点を導出する第4のステップと、前記モデル座標系上で、前記パラメータを用いて、前記交点から前記視線の向きにある前記曲面形状までの距離 e を算出する第5のステップと、前記距離 e に基づいて、前記視界座標系上で求めた各視線の前記深度値 d を補正する第6のステップと、を備え、前記第2のステップから第6のステップまでの処理を、グラフィックハードウェアを用いて行うことを特徴としている。

30

40

この方法では、曲面を平面で近似したことに伴う誤差をグラフィックハードウェア内で補正することができる。

【0011】

また、本発明の交差座標導出方法は、前記曲面形状が、数式で定義可能な形状であれば適用できる。

曲面形状が数式で定義できる場合は、視線方向におけるポリゴンと曲面形状との隔たり e を計算することができるので、深度値 d の補正が可能である。

【0012】

また、本発明の交差座標導出方法では、前記曲面形状を、円筒形状または球形状として

50

も良い。

ボールエンドミル工具の掃引形状や逆オフセット形状は、円筒形状や球形状で表すことができるから、本発明の交差座標導出方法により、ボールエンドミル工具で加工する被削物形状の評価や工具経路の設定を正確に行うことができる。

【0013】

また、本発明の交差座標導出方法では、前記多面体を前記モデル座標系の単位座標系上で定義し、前記単位座標系上で求めた前記距離 e を基に前記視界座標系上の前記深度値 d の誤差を算出して、前記深度値 d を補正する。

ポリゴンを単位座標系上で正規化して定義することにより、深度値 d に対する補正値の算出が簡単になる。

【0014】

また、本発明の交差座標導出方法では、ポリゴンをモデル座標系から視界座標系に移す場合に、拡大・縮小、並行移動、回転または透視変換による変換を与えた前記ポリゴンの各頂点の頂点座標を導出する。

この座標変換により、曲面形状の任意の変形が可能になる。

【0015】

また、本発明の交差座標導出方法では、前記第1のステップから第6のステップまでの処理を、前記曲面形状を近似する前記多面体の全てのポリゴンと交差する全ての視線に対して行い、各視線の補正後の前記深度値 d で表される面の包絡面形状を導出する。

曲面を多面体近似したことに伴う誤差を含まない包絡面形状を得ることができる。

【0016】

また、本発明の交差座標導出方法では、前記包絡面形状として、ボールエンドミル工具の掃引形状を導出する。

正確なボールエンドミル工具の掃引形状が描画されるため、被削物形状の正しい評価が可能になる。

【0017】

また、本発明の交差座標導出方法では、前記包絡面形状として、製品形状の上で工具の向きを反転させた逆オフセット形状を導出する。

正確な逆オフセット形状が描画されるため、工具の正確な移動経路設定することが可能になる。

【発明の効果】

【0018】

本発明によれば、視界座標系上での曲面と視線との交差座標を、正確、かつ、迅速に求めることができる。

【発明を実施するための最良の形態】

【0019】

本発明では、平面近似に伴う誤差を補正する手法を提案する。ここでは、補正の対象となる曲面を球面、円筒面とする。また、これらに対する近似による誤差の補正をグラフィックスハードウェア内部のみで行う。

図1は提案する手法の概念を示す。従来の幾何計算手法では、曲面形状のトポロジを定義するモデル座標系上において平面近似された多面体を、視界座標系に投影し、視点と各平面との距離を深度 d として求め、これを用いて包絡面の推定を行ってきた。今回提案する手法では、この深度計算の過程を近年のGPUが有するプログラマブルシェーダの機能を用いて変更する。

【0020】

具体的には、交点のモデル座標系上での座標値を元に格子点上でのポリゴンと近似前の曲面との間の距離 e を計算する。その上で深度 d と距離 e とから本来の曲面に対して視線が持つべき深度値を推定する。この処理をポリゴンごとにGPUの機能を用いて行い、深度値の比較から最も基準面に近い交点の座標を包絡面の座標として求めることによって正確かつ高速な形状の推定を実現する。

10

20

30

40

50

【 0 0 2 1 】

この深度計算のアルゴリズムでは、パラメトリックに定義された曲面と、近似後のポリゴンと視線との交点の位置関係を把握し、この交点と曲面との距離を求めることが必要となる。そこで、本手法では、ポリゴン頂点の座標定義を正規化されたモデル座標系上でを行い、視座標系での視線とポリゴンとの交点座標をモデル座標系上に投影することによって、GPUのPixel Shaderの内部で曲面と交点との隔たりを計算できるようにしている。

以下ではその内容について述べる。

【 0 0 2 2 】

(モデル座標系における円筒面との交点座標の導出)

10

一般に、コンピュータグラフィックスの描画においては、ポリゴンを設定し描画対象物のトポロジを定義するためのモデル座標系と、視座標系を構成する各視線とポリゴン間の位置関係を正規化した視座標系の二つの座標系が用いられる。

通常の描画処理では、CPUによって与えられたモデル座標系上の各ポリゴンの頂点を、4つの頂点からなる同次座標で表し、これに対して拡大・縮小、並行移動、回転、透視変換からなる座標変換を与えて視座標系上での頂点座標を導出する。全てのポリゴンの頂点を変換し、視座標系上にポリゴンを投影した状態での視線とポリゴンとの交点を元に、各視線に対応する画素の状態を推定し、出力画面の色が決定される。

【 0 0 2 3 】

20

このとき、視線とポリゴンとの交点座標は、GPU内部では対応するポリゴンとそれを構成する視座標系上の頂点座標の内挿として与えられる。そこで、モデル座標系上での各ポリゴンを正規化された形状に対して設定し、交点のモデル座標系上での座標値を視座標系上での各頂点座標から同様に推定する。これと数学的に定義された曲面との隔たりをGPU内部で計算することで深度値の補正を行う。

【 0 0 2 4 】

今、3次元の同次座標変換における拡大・縮小行列 $T(a, b, c)$ を(数1)のように、並行移動行列 $M(a, b, c)$ を(数2)のように、Z軸及びY軸まわりの回転行列 $R_z(\quad)$ 、 $R_y(\quad)$ を(数3)、(数4)のように定義する。

【数1】

30

$$T_{(a,b,c)} = \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

【数2】

40

$$M_{(a,b,c)} = \begin{pmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

【数 3】

$$Rz_{(\theta)} = \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

【数 4】

$$Ry_{(\phi)} = \begin{pmatrix} \cos \phi & 0 & -\sin \phi & 0 \\ 0 & 1 & 0 & 0 \\ \sin \phi & 0 & \cos \phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

10

【0025】

このとき、図 2 に示すような絶対座標系上にある半径 r の円筒形状の同次座標 C_{ab} (x_{ab} 、 y_{ab} 、 z_{ab} 、1) は、モデル座標系上において、Z 軸の方向を中心軸とし半径 1、長さ 1 となる円筒形状の座標 C_{md} (x_{md} 、 y_{md} 、 z_{md} 、1) を用いて次の (数 5) ように設定することができる。

20

【数 5】

$$C_{ab} = M_{(x,y,z)} \cdot Rz_{(\theta)} \cdot Ry_{(\phi)} \cdot T_{(l,r,r)} \cdot C_{md}$$

さらに、絶対座標系上での視界の中心を (x_v 、 y_v 、 z_v)、視界の幅と高さを w 、視界の奥行きを f とする並行投影後の視界座標系における頂点座標 C_{vw} (x_{vw} 、 y_{vw} 、 z_{vw} 、1) は、先の変換に加えて以下の変換式 (数 6) によって C_{md} から設定することができる。

【数 6】

$$\begin{aligned} C_{vw} &= T_{(2/w, 2/w, 1/f)} \cdot M_{(-x_v, -y_v, -z_v)} \cdot C_{ab} \\ &= V \cdot C_{md} \end{aligned}$$

30

C_{md} より C_{vw} を求める頂点座標変換行列 V は、円筒形状の半径及び長さ、位置ごとに異なる行列となるため、工具掃引形状に対して工具の運動ステップごとに設定が繰り返される。

【0026】

ここで、投影後のポリゴンと視線との交点座標 B_{vw} について、そのモデル座標系上での交点座標 B_{md} を求める。今、ポリゴンを三角形と仮定し、その投影後の頂点を A_{vw1} 、 A_{vw2} 、 A_{vw3} とすると、 B_{vw} は内挿変数 s 、 t を用いて、次式 (数 7) のように、

40

【数 7】

$$B_{vw} = s(A_{vw2} - A_{vw1}) + t(A_{vw3} - A_{vw1})$$

と表すことができる。このとき頂点投影行列 V の逆行列 V^{-1} は (数 6) の定義より、

【数 8】

$$V^{-1} = T_{(1/l, 1/r, 1/r)} \cdot Ry_{(-\phi)} \cdot Rz_{(-\theta)} \cdot M_{(-x, -y, -z)} \cdot M_{(x_v, y_v, z_v)} \cdot T_{(0.5w, 0.5w, f)}$$

となる。ここでモデル座標系上での対応するポリゴンの頂点座標を A_{md1} 、 A_{md2} 、 A_{md3} とすると、モデル座標系での交点 B_{md} は、(数 7) に V^{-1} を適用することによって次式 (数 9) のように表すことができる。

【数 9】

$$\begin{aligned} B_{md} &= V^{-1} \cdot B_{vw} \\ &= s(V^{-1} \cdot A_{vw2} - V^{-1} \cdot A_{vw1}) + t(V^{-1} \cdot A_{vw3} - V^{-1} \cdot A_{vw1}) \\ &= s(A_{md2} - A_{md1}) + t(A_{md3} - A_{md1}) \end{aligned}$$

10

【0027】

この内挿変数 s 、 t は、Pixel Shader が視線と各ポリゴンとの交差を判定する際に自動的に計算される。そこで、これを利用してモデル座標系での交点座標を推定する。すなわち、Vertex Shader において行われる各頂点座標の変換に先立って、モデル座標系上での頂点座標を属性値として各頂点に与えておく。Pixel Shader には、 s 、 t を用いて各頂点に与えられた属性値から交点における値を内挿する機能があるため、これを用いて (数 9) を計算する。これにより、逆行列 V^{-1} を個々の交点ごとに明示的に求めることなく、モデル座標系上での交点 B_{md} の座標を推定することが可能となる。

20

【0028】

(モデル座標系での交点座標を用いた誤差補正)

先に述べたように、本発明の手法では曲面と近似多面体との隔たりを、モデル座標系上で正規化された曲面の位置と各ポリゴンへの視線の交点座標を用いて導出する。

3軸制御加工では、絶対座標系の Z 軸と工具の中心軸は常に一致する。このとき、絶対座標系上で $(0, 0, 1)$ の向きに配置された各視線は、モデル座標系上では $(f \cdot \sin \phi, 0, f \cdot \cos \phi)$ の向きに投影される。ここで、モデル座標系上での視線の向きと Z_{md} 軸がなす角を δ とすると、 δ は次式 (数 10) のように求まる。

30

【数 10】

$$\delta = \arctan((r \cdot \sin \phi) / (f \cdot \cos \phi))$$

ここで、 B_{md} を $(x_{md}, y_{md}, z_{md}, 1)$ とおき、パラメータ r 、 f を用いて B_{md} と円筒面と視線との交点間の距離 e_{vw} を視座標系上で表すと次式 (数 11) のようになる。

40

【数 11】

$$e_{vw} = \frac{\left(\sqrt{1 - y_{md}^2} + z_{md} \right) \cdot r}{f \cdot \cos \delta}$$

【0029】

50

さらに、GPUの組み込み機能によって求められる視点と B_{vw} との視座標系上での距離 d_{vw} から e_{vw} を引き、得られた値を改めてグラフィックハードウェアのメモリに書き込むことにより正確な深度の値を各交点に与えることが可能となる。この計算は該当する視線と交差する可能性のある全てのポリゴンとの間で行われ、0から1の間で最も小さい値をとる深度値から包絡面の深度値 d_{true} が決定され、GPUの出力となる。

GPUから出力される画面上での各画素の座標 (u, v) は、それぞれ視座標系における交点の xy 座標 (x_{vw}, y_{vw}) に相当する。そこで、これらの座標値と視点・視座の設定値を用いることによって以下の(数12)(数13)(数14)からZ-map基板面の格子点に対応する各画素における被削物・逆オフセット面の絶対座標系上での座標 $B_{true}(x_{true}, y_{true}, z_{true})$ を求めることが可能となる。

$$(数12) \quad x_{true} = x_v + u \cdot w$$

$$(数13) \quad y_{true} = y_v + v \cdot w$$

$$(数14) \quad z_{true} = z_v + d_{true} \cdot f$$

【0030】

(モデル座標系における球面との交点座標の導出)

球面を対象とした補正においても同様に、正規化されたモデル座標系でポリゴンを定義し、交点座標を推定したのち曲面とポリゴンとの隔たりの計算を行うことによって正確な深度値の計算が可能である。

以前の研究において、球面はボールエンドミルの進行方向が変化する際に現れ、その殆どは前後の工具掃引形状に属する円筒面の内側に存在することが指摘されている。そこでここでは、あらかじめその前後の工具経路から描画の必要のない部分を除いた球面の一部のみを対象としてポリゴンの設定を行う。

【0031】

図3は部分球面の形状を示す。この曲面20は、二つの円筒形状の端面をそれぞれの長辺として持つ三日月状の形状となる。これを本来の球面上に頂点を持ち球面に内接する微小三角形の集合で表し、頂点座標変換を実施する。この頂点座標変換では、Z軸及びY軸まわりの回転変換が発生しないことから、モデル座標系における近似前の球面の半径を1とすると変換行列 V は以下の(数15)のように定義される。

【数15】

$$V = T_{(1/2x_w, 1/2y_w, 1/f)} \cdot M_{(-x_v, -y_v, -z_v)} \cdot M_{(x, y, z)} \cdot T_{(r, r, r)}$$

ここから円筒形状の場合と同様にモデル座標系上での視線とポリゴンとの交点を求め、近似前の球面との隔たりを計算する。

視線の方向はモデル座標系のZ軸方向と一致する。このとき、球面と交点との視座標系上での距離 e_{vw} は、Pixel Shaderに与えられたポリゴンと視線との交点座標 $B_{md}(x_{md}, y_{md}, z_{md}, 1)$ から次式(数16)

【数16】

$$e_{vw} = \frac{\left(\sqrt{1 - x_{md}^2 - y_{md}^2} + z_{md} \right) \cdot r}{f}$$

と求まる。以降は円筒形状と同様に深度値の書き換えを行い、各ポリゴンの深度値を比較する。これにより正確な包絡面形状の推定が実現する。

【0032】

Pixel Shader内部では、上記のようにポリゴンに対応する曲面の種類によっ

10

20

30

40

50

て深度値の補正式が切り替えられる。また、補正に必要な情報が近似前の形状の種類ごとに参照される。例えば、近似前の曲面が球であるポリゴンでは r 及び f が、円筒であるポリゴンではそれに加えて d がそれぞれ `Pixel Shader` 内部で参照される。本発明では、これらのパラメータの値を CPU による平面近似の実施時に、ポリゴン各頂点の属性値として設定し、これを GPU に備えられたメモリに予め転送しておくことによって処理の分岐と補正情報の参照とを可能としている。

【0033】

(評価システムのアルゴリズム)

この手法の有効性を検証し、高速かつ高精度な形状評価が可能かどうかを確認するため、3軸加工を対象としたプロトタイプの切削シミュレーションシステムを開発した。

図4は、開発したシステムのアルゴリズムを表す。このアルゴリズムにおいては、まず NC プログラムからデータが読み込まれ(ステップ1)、NC プログラム内の各工具運動ステップに対応した円筒面と、その間に存在する部分球面とがポリゴンの集合としてそれぞれ設定され、各ポリゴンの頂点座標が算出される。このとき、各ポリゴンの頂点情報(属性値)には、頂点座標とともに、近似前の曲面の種類と、対応する工具掃引形状の半径 r 及び視界座標系での視線の Z 軸に対する角度 θ がパラメータとして与えられる(ステップ2)。

次に、Z-map 形式で形状を出力する際の評価範囲と格子点間隔とが設定され、これに対応した視点座標が与えられる(ステップ3)。

【0034】

これらの情報をもとに、`Vertex Shader` によって各頂点の座標と視界の状態を反映した頂点座標変換行列 V が設定され、全てのポリゴンが視界座標系に投影される(ステップ4)。`Pixel Shader` においては、視線とポリゴンとの視界座標系上での交点座標 B_{vw} が導出され(ステップ5)、交点座標 B_{vw} での視線に対するポリゴンの深度値と内挿変数 s 、 t とが計算され(ステップ6)、内挿変数 s 、 t を用いてモデル座標系上での交点座標が導出され(ステップ7)、パラメータを用いて近似前の曲面とポリゴン上の交点との距離 e_{vw} が計算され(ステップ8)、基準面に対する正確な深度値が導出される(ステップ9)。これらの深度値を視線と交差する全てのポリゴンに対して求め、それらと比較して、視点と最も近い位置での交点 B_{true} の深度 d_{true} が推定され(ステップ10)、ディスプレイに出力されて(ステップ11)、形状の評価が実現する。

【0035】

このとき、ステップ4からステップ10の処理(図4中(a)の範囲)は、GPU 内で行われる。深度値の計算を行うステップ6からステップ9の処理(図4中(b)の範囲)は、全てのポリゴンと全ての視線に対して繰り返し処理を行うことが必要となるが、これらは全て GPU 内で計算が行われる。そのため、平面及び円筒面、球面の包絡面に対する Z-map 形式での形状の推定は短時間で処理を終えることが可能になる。

【0036】

(計算機実験による検証)

本評価システムの実装にあたっては、現在、CAD/ CAM/ CAE ソフトウェアにおいて広く用いられている `OpenGL` グラフィックライブラリと、そのプログラマブルシェーダーカスタマイズ用ライブラリである `GLSL` (`OpenGL Shading Language`) を用いた

。検証に用いた工具経路は6種類の工具による工程からなり、総数19万2千 `Step` の工具運動によって構成されている。これに対して本計算機実験では、部分球面及び円筒面を各辺の長さが球面・円筒面上で10度以下となるように平面近似し、形状評価に用いている。

【0037】

図5は10万 `Step` 目の加工の途中において、工具刃が進行方向で被削物を除去する部分を GPU の機能を用いて特定した状態を表す。描画面の縦横の長さは10万 `step` 目に加工を行っている工具の径に相当し、中央の黒い部分が工具進行方向上で工具刃が除

10

20

30

40

50

去する部分、その他の色の部分がそれ以前の工具経路によって既に除去され、工具刃よりも視点に近い、すなわちZ方向の高さが低い部分を表す。

本発明を適用した場合を図5(a)に、ポリゴンと視線との交点座標の深度のみを用いて計算を行った場合を図5(b)に示している。これらと比較すると、特に(b)で除去部分の境界部分で誤差が発生していることが分かる。これは、曲面とポリゴンとの間の近似誤差が視線の位置ごとに変化し、その結果各ポリゴンに対する視線上での前後関係の評価が近似前の曲面の前後関係と異なる状態になったのが原因であると考えられる。

【0038】

この計算には、CPUにCore 2 Duo E6600 2400MHz、GPUにGeForce 7800 GTX 430MHzを持つ主記憶容量2048MBのPCを用いた。試作したシステムでは、一画面を構成する262,144個の格子点群に対して10万step目までの間で工具一箇所あたり平均0.057秒で描画範囲に入る全ての工具掃引形状間の上下関係を判定し、任意の瞬間での被削物形状に対する工具切込みの状態を導出することが可能であった。従来曲面に対する補正を実施せずに、工具掃引形状を多面体近似したポリゴン群との深度値を求めた場合の平均所要時間は平均0.021秒であり、本発明の手法は、従来の手法と同様に短時間で形状の推定を行うことが可能である。

なお、ここでは、補正対象の曲面が球面及び円筒面の場合について説明したが、それ以外の曲面であっても、数式で定義可能な形状であれば、本発明を適用することができる。

【産業上の利用可能性】

【0039】

本発明は、工作機械でNCプログラムを実行したときの被削物の削り残しや削り過ぎを事前に検証する切削シミュレーション、あるいは、NCプログラムを作成する際の製品モデルから工具経路を生成する処理や、製品形状と工具軸との干渉を判定する処理などに広く適用することができ、それらの処理の高精度化を実現し、NCプログラムの最適化を図ることができる。

【図面の簡単な説明】

【0040】

【図1】本発明の曲面形状と視線との交差座標導出方法を概略的に示す図

【図2】絶対座標系上での円筒形状を示す図

【図3】工具掃引形状の間隙における部分球面形状を示す図

【図4】本発明の交差座標導出方法を採用入れた切削シミュレーションの手順を示すフロー図

【図5】本発明の交差座標導出方法を採用入れた切削シミュレーションの計算結果(a)と、従来の切削シミュレーションの計算結果(b)とを示す図

【図6】被削物形状の推定方法を説明する図

【図7】逆オフセット法による工具経路生成方法を説明する図

【図8】工具掃引形状の平面近似に伴う誤差を説明する図

【図9】円筒面の多角形近似による深度値誤差の変化を説明する図

【図10】グラフィックスハードウェアの構成を説明する図

【符号の説明】

【0041】

20	部分球面
43	CPU
44	GPU

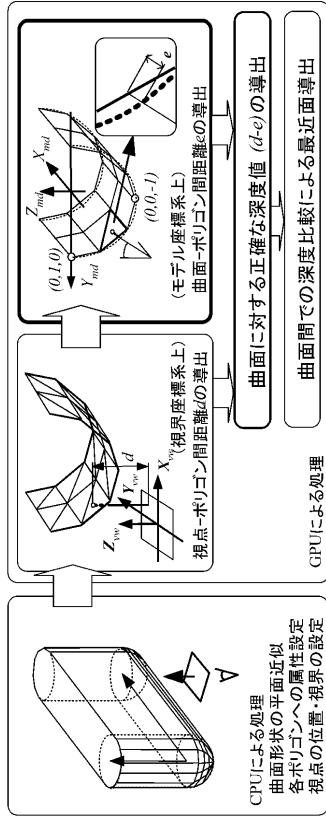
10

20

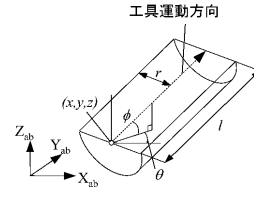
30

40

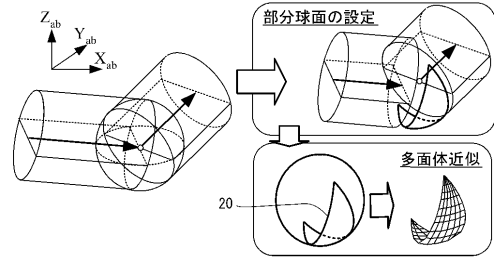
【 図 1 】



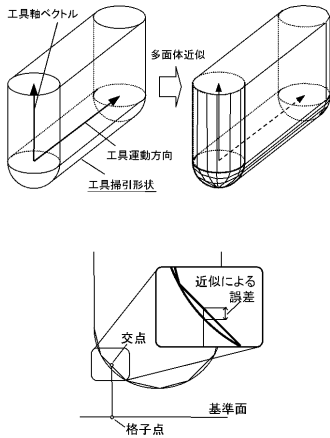
【 図 2 】



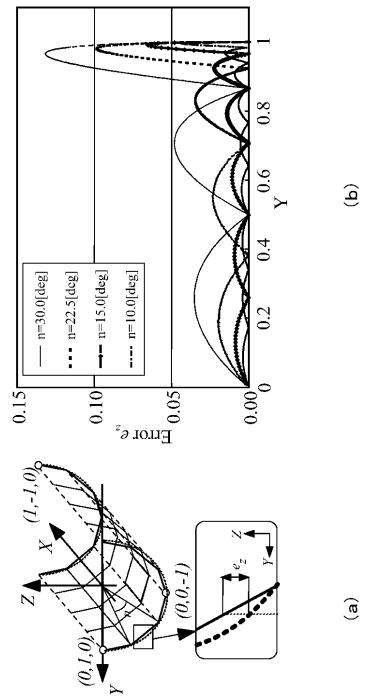
【 図 3 】



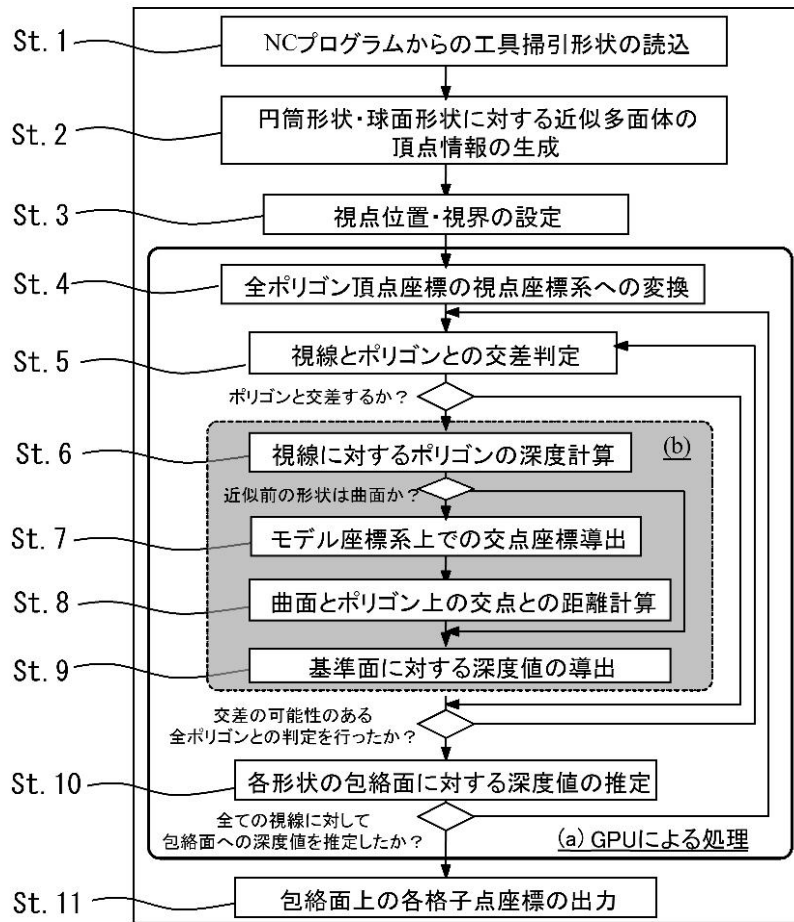
【 図 8 】



【 図 9 】



【 図 4 】



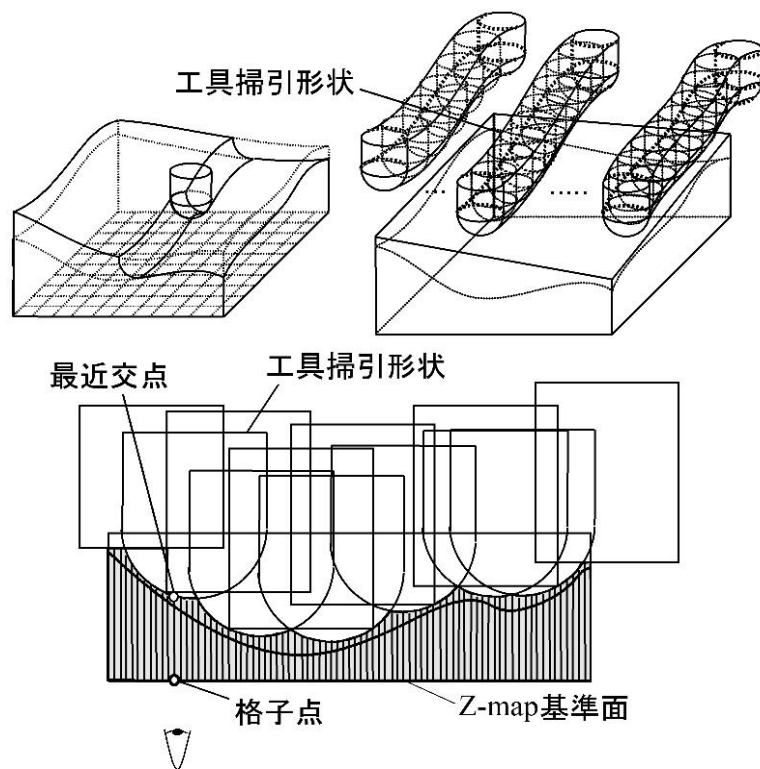
【 图 5 】



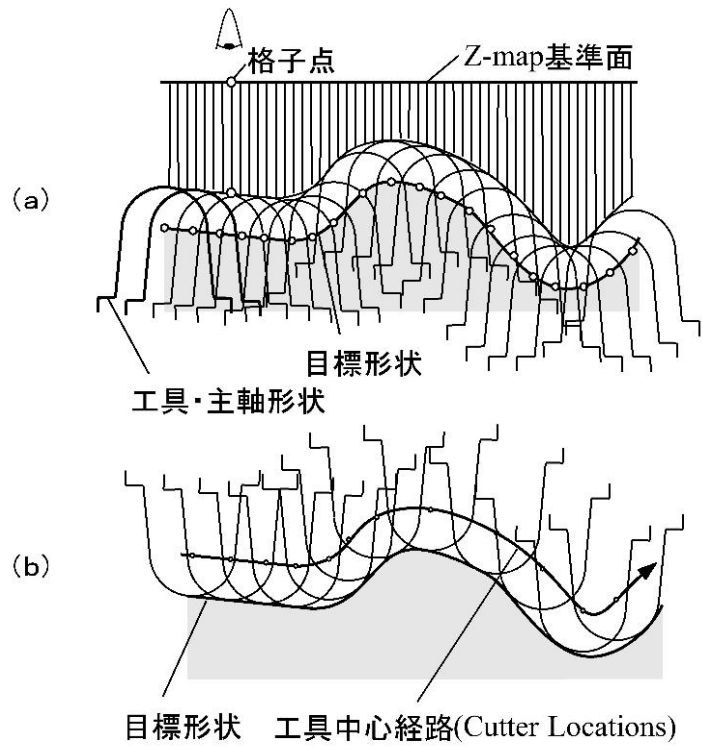
(a)

(b)

【 图 6 】



【图 7】



【 図 1 0 】

