

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テ-マコード <sup>*</sup> (参考)
G05B 19/4097 19/4069		G05B 19/4097 19/4069	C 5H269

審査請求 未請求 請求項の数12 O L (全32頁)

(21)出願番号	特願2002 - 56858( P 2002 - 56858)	(71)出願人	396020800 科学技術振興事業団 埼玉県川口市本町 4 丁目 1 番 8 号
(22)出願日	平成14年 3 月 4 日(2002.3.4)	(72)発明者	乾 正知 茨城県ひたちなか市東大島 1 - 2 - 21
		(74)代理人	100107010 弁理士 橋爪 健 F タ-ム(参考) 5H269 AB05 AB07 BB03 BB05 QA05

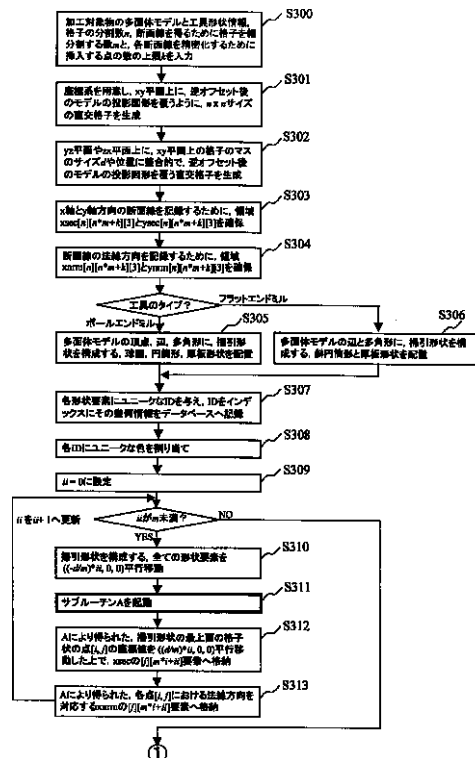
(54) 【発明の名称】 工具経路面計算方法、工具経路面計算プログラム及び工具経路面計算プログラムを記録した記録媒体

(57) 【要約】

【課題】 3次元グラフィックス表示装置を利用し、工具経路面を高速且つ高精度に計算する。

【解決手段】 処理部は、加工対象物の多面体モデル情報と工具形状情報、格子の分割数、断面線を得るために格子を補充する点と、各形状要素を情報化するために挿入する点の座標を入力

【解決手段】 処理部は、加工対象物の多面体モデル情報と工具形状情報とを入力し(S300)、多面体モデルの各要素に対して、逆工具の掃引形状を構成する各形状要素を配置する(S305、S306)。処理部は、配置された各形状要素に対するID番号・色情報・幾何情報を記憶部に記憶し(S307、S308)、逆工具の掃引形状を構成する形状要素に基づき、グラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置をx軸及びy軸方向に所定距離だけずらしつつ、最上面の断面線を細分割した点列データを求める(S310以降)。処理部は、求められた隣接する点列データに基づき、断面線の折れ曲がり点を検出し、断面線の点列データへ挿入する。処理部は、空間を立方体の集積へ分割し、マーチングキューブ法により掃引形状の最上面を求める。



## 【特許請求の範囲】

【請求項 1】数値制御工作機械に取り付けられた工具を経路に沿って移動させて工作物を加工する際に、工具の逆形状分膨らませた工具経路面を生成するための工具経路面計算方法であって、

処理部は、加工対象物の多面体モデル情報と工具形状情報とを入力し、記憶部に記憶するステップと、

処理部は、多面体モデルの各要素に対して、逆工具の掃引形状を構成する各形状要素を配置するステップと、

処理部は、配置された各形状要素に対して、ユニークな ID 番号、ユニークな色情報、形状要素の幾何情報を記憶部に記憶するステップと、

処理部は、入力された逆工具の掃引形状を構成する形状要素に基づき、該掃引形状の最上面を覆う格子状の点群の座標データと、その点における最上面の法線又は接線データを出力するグラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を x 軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を x 軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、

処理部は、グラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を y 軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を y 軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、

処理部は、求められた隣接する点列データの法線又は接線データに基づき、断面線の折れ曲がり点を検出し、検出された折れ曲がり点の点列データを断面線の点列データへ挿入して記憶部に記憶するステップと、

処理部は、格子状の点群に従い、空間を立方体の集積へ分割するステップと、

処理部は、記憶部に記憶された点列データを参照し、逆工具の掃引形状の最上面の x 軸方向の断面線及び y 軸方向の断面線と、分割された各立方体の各辺との交点を算出するステップと、

処理部は、算出された交点の情報に基づいて、各立方体の頂点が工具経路面の上下のどちらに存在するかを求め、マーチングキューブ法により立方体内部に三角形を配置するステップと、

処理部は、各立方体内部に配置された三角形群を、掃引形状の最上面として出力部に出力又は記憶部に記憶するステップと、を含む工具経路面計算方法。

【請求項 2】前記各形状要素を配置するステップは、処理部は、工具形状情報がボールエンドミルの場合には、多角形モデルの要素としての、頂点、辺、多角形のそれぞれに対応して、形状要素として球面、円筒形、厚板形状を配置することを特徴とする請求項 1 に記載の工具経路面計算方法。

【請求項 3】前記各形状要素を配置するステップは、処理部は、工具形状情報がフラットエンドミルの場合には、多角形モデルの要素としての、辺、多角形のそれぞれに対応して、形状要素として斜円筒形と厚板形状を配置することを特徴とする請求項 1 に記載の工具経路面計算方法。

【請求項 4】前記グラフィックスハードウェアの隠面消去処理は、

10 処理部は、逆工具の掃引形状を構成する形状要素から、形状要素の下向きの部分や他の形状要素の内部に含まれている部分等の掃引形状の最上面に關与しない部分を予め除去するステップと、

処理部は、除去されずに残った形状を多面体化するステップと、

処理部は、配置された各形状要素に対する、ユニークな ID 番号、ユニークな色情報、形状要素の幾何情報を記憶部から読み取り、各形状要素を構成する多角形に対する色情報をグラフィックスハードウェアへ渡すステップと、

20 グラフィックスハードウェアは、処理部から受けた色情報に基づき、デプスバッファによる隠面消去処理を用いて、各点における掃引形状の最上面の高さを表すデプス値と、各点における色情報を得て、得られたデプス値と色情報を処理部へ渡すステップと、

処理部は、グラフィックハードウェアから受けた各点の座標データとデプス値に基づいて、掃引形状の最上面を示す工具経路面の点列データを生成するステップと、

30 処理部は、色情報の基づき、記憶部を参照して、最上面の各点に対応する形状要素の ID 及び幾何情報を得て、各点の座標データと幾何情報に従い、各点における法線方向を求めるステップと、

処理部は、工具経路面の点群の座標データとその法線又は接線情報を出力するステップとを含む請求項 1 に記載の工具経路面計算方法。

【請求項 5】前記折れ曲がり点を断面線の点列データへ挿入して記憶部に記憶するステップは、

処理部は、記憶部を参照し、断面線を表す座標データと、各点における工具経路面の法線又は接線データを読み取るステップと、

40 処理部は、各断面線上の連続する 2 点について、ふたつの接線又は法線方向が所定値より大きく異なる場合には、2 点の間で断面線が折れ曲がり点を有すると判断し、

処理部は、新しい折れ曲がり点を計算し、それを点列データに追加するステップと、

処理部は、更新された点列データと、更新後の点列の長さを示すデータを記憶又は出力するステップとを含む請求項 1 又は 4 に記載の工具経路面計算方法。

【請求項 6】前記点列データに追加するステップは、

50 処理部は、求められた新しい折れ曲がり点を示す点列デ

ータを格納する場所を得るために、該当する点列データ以降の点列データと法線又は接線データとを、シフトして格納しなおすステップと、

処理部は、求められた新しい折れ曲がり点を示す点列データをシフトされて空いた領域に格納するステップとを含む請求項 5 に記載の工具経路面計算方法。

【請求項 7】処理部は、得られた工具経路面に基づき、工具経路を求めるステップをさらに備えた請求項 1、4 又 5 のいずれかに記載の工具経路面計算方法。

【請求項 8】処理部は、工具経路面及び / 又は工具経路を表示するステップをさらに備えた請求項 1、4 又 5 のいずれかに記載の工具経路面計算方法。

【請求項 9】数値制御工作機械に取り付けられた工具を経路に沿って移動させて工作物を加工する際に、工具の逆形状分膨らませた工具経路面を生成するための工具経路面計算プログラムであって、

処理部は、加工対象物の多面体モデル情報と工具形状情報とを入力し、記憶部に記憶するステップと、

処理部は、多面体モデルの各要素に対して、逆工具の掃引形状を構成する各形状要素を配置するステップと、

処理部は、配置された各形状要素に対して、ユニークな ID 番号、ユニークな色情報、形状要素の幾何情報を記憶部に記憶するステップと、

処理部は、入力された逆工具の掃引形状を構成する形状要素に基づき、該掃引形状の最上面を覆う格子状の点群の座標データと、その点における最上面の法線又は接線データを出力するグラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を x 軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を x 軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、

処理部は、グラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を y 軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を y 軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、

処理部は、求められた隣接する点列データの法線又は接線データに基づき、断面線の折れ曲がり点を検出し、検出された折れ曲がり点の点列データを断面線の点列データへ挿入して記憶部に記憶するステップと、

処理部は、格子状の点群に従い、空間を立方体の集積へ分割するステップと、

処理部は、記憶部に記憶された点列データを参照し、逆工具の掃引形状の最上面の x 軸方向の断面線及び y 軸方向の断面線と、分割された各立方体の各辺との交点を算出するステップと、

処理部は、算出された交点の情報に基づいて、各立方体の頂点が工具経路面の上下のどちらに存在するかを求め

ることで、マーチングキューブ法により立方体内部に三角形を配置するステップと、

処理部は、各立方体内部に配置された三角形群を、掃引形状の最上面として出力部に出力又は記憶部に記憶するステップとをコンピュータに実行させるための工具経路面計算プログラム。

【請求項 10】前記グラフィックスハードウェアの隠面消去処理は、

処理部は、逆工具の掃引形状を構成する形状要素から、形状要素の下向きの部分や他の形状要素の内部に包含されている部分等の掃引形状の最上面に關与しない部分を予め除去するステップと、

処理部は、除去されずに残った形状を多面体化するステップと、

処理部は、配置された各形状要素に対する、ユニークな ID 番号、ユニークな色情報、形状要素の幾何情報を記憶部から読み取り、各形状要素を構成する多角形に対する色情報をグラフィックスハードウェアへ渡すステップと、

20 グラフィックスハードウェアは、処理部から受けた色情報に基づき、デプスバッファによる隠面消去処理を用いて、各点における掃引形状の最上面の高さを表すデプス値と、各点における色情報を得て、得られたデプス値と色情報を処理部へ渡すステップと、

処理部は、グラフィックハードウェアから受けた各点の座標データとデプス値に基づいて、掃引形状の最上面を示す工具経路面の点列データを生成するステップと、

30 処理部は、色情報の基づき、記憶部を参照して、最上面の各点に対応する形状要素の ID 及び幾何情報を得て、各点の座標データと幾何情報に従い、各点における法線方向を求めるステップと、

処理部は、工具経路面の点群の座標データとその法線又は接線情報を出力するステップをコンピュータに実行させるための請求項 9 に記載の工具経路面計算プログラム。

【請求項 11】前記折れ曲がり点を断面線の点列データへ挿入して記憶部に記憶するステップは、

処理部は、記憶部を参照し、断面線を表す座標データと、各点における工具経路面の法線又は接線データを読み取るステップと、

処理部は、各断面線上の連続する 2 点について、ふたつの接線又は法線方向が所定値より大きく異なる場合には、2 点の間で断面線が折れ曲がり点を有すると判断し、

処理部は、新しい折れ曲がり点を計算し、それを点列データに追加するステップと、

処理部は、更新された点列データと、更新後の点列の長さを示すデータを記憶又は出力するステップとをコンピュータに実行させるための請求項 9 又は 10 に記載の工具経路面計算プログラム。

【請求項 1 2】数値制御工作機械に取り付けられた工具を経路に沿って移動させて工作物を加工する際に、工具の逆形状分膨らませた工具経路面を生成するための工具経路面計算プログラムを記録したコンピュータ読み取り可能な記録媒体であって、

処理部は、加工対象物の多面体モデル情報と工具形状情報とを入力し、記憶部に記憶するステップと、

処理部は、多面体モデルの各要素に対して、逆工具の掃引形状を構成する各形状要素を配置するステップと、

処理部は、配置された各形状要素に対して、ユニークな I D 番号、ユニークな色情報、形状要素の幾何情報を記憶部に記憶するステップと、

処理部は、入力された逆工具の掃引形状を構成する形状要素に基づき、該掃引形状の最上面を覆う格子状の点群の座標データと、その点における最上面の法線又は接線データを出力するグラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を x 軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を x 軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、

処理部は、グラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を y 軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を y 軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、

処理部は、求められた隣接する点列データの法線又は接線データに基づき、断面線の折れ曲がり点を検出し、検出された折れ曲がり点の点列データを断面線の点列データへ挿入して記憶部に記憶するステップと、

処理部は、格子状の点群に従い、空間を立方体の集積へ分割するステップと、

処理部は、記憶部に記憶された点列データを参照し、逆工具の掃引形状の最上面の x 軸方向の断面線及び y 軸方向の断面線と、分割された各立方体の各辺との交点を算出するステップと、

処理部は、算出された交点の情報に基づいて、各立方体の頂点が工具経路面の上下のどちらに存在するかを求め、マーチングキューブ法により立方体内部に三角形を配置するステップと、

処理部は、各立方体内部に配置された三角形群を、掃引形状の最上面として出力部に出力又は記憶部に記憶するステップと、工具経路面計算プログラムを記録したコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、工具経路計算方法、工具経路計算プログラムを記録したコンピュータ読み取り可能な記録媒体及び工具経路計算装置に係り、特

に、数値制御工作機械に取り付けられた工具を経路に沿って移動させて工作物を加工する際に、工具の逆形状分膨らませた工具経路面（工具参照面）を生成する工具経路計算方法、工具経路面計算プログラムを記録したコンピュータ読み取り可能な記録媒体に関する。

【0002】

【従来の技術】プラスチックやダイキャスト製品の製造に用いられる金型は、エンドミル工具を用いた NC ( Numerical Control ) 切削加工により製作される。金型加工では深く彫り込むことが多いため、剛性に優れた 3 軸制御の工作機械の利用が一般的である。NC 切削加工では、切削工具は膨大な回数の微小な直線移動を繰り返しおこない、素材材から少しずつ不要部を除去していく。複雑な工具経路の計算には、高速な計算機を用いても数十分の処理時間を要するため、その高速化が待望されている。工具経路を計算する際には、工具が金型の内部へ削り込む、ガウジ ( G o u g e ) とよばれる問題に注意する必要がある。

【0003】図 29 は、金型加工における削り込みの一例を示す説明図であり、金型表面の曲率半径が切削工具（例えば、ボールエンドミル）の半径よりも小さくなる部分で発生する、削り込みの一例を示している。すなわち、製品表面との接触を保ちつつ、図中の左から右へ移動するボールエンドミルは、金型表面の曲率半径が工具半径よりも小さくなる部分（図中、斜線で示したガウジング領域）において、このガウジング領域との接触を見逃すと、そのまま金型製品の内側へ削り込んでしまう。市販の工具経路の計算プログラムは、その処理時間のほとんどを、ガウジ防止のために費やしている。金型の表面を切削工具の逆形状分膨らませた工具経路面（工具参照面）を生成し、工具が常にその上側を移動するように経路を生成すれば、ガウジの問題を完全に解決できる。本発明は、この工具経路面（工具参照面）を精密かつ高速に計算するコンピュータプログラムに関するものである。この技術を用いると、複雑な金型を加工するための工具経路を、ごく短時間で計算することが可能になる。

【0004】

【発明が解決しようとする課題】工具経路面は、金型の表面に沿って工具の逆形状を縦横に滑らせた時の、掃引形状の上面に相当する。CSG や境界表現法などの、立体形状の厳密なモデリング技術では、掃引形状を得る手続きは非常に複雑なものになる。そのため NC 切削加工の分野では、掃引形状の上面を近似的に計算する、逆オフセット法がよく用いられる。この手法では、工具の回転軸方向を z 軸とする座標系を考え、その x y 平面上に十分に細かい直交格子を用意する。そして各格子点から z 軸に平行な直線を伸ばし、掃引形状の最上面との交点を計算することで、掃引形状の上面を覆う稠密な点群を得る。最後に点群の隙間を多角形で補間することで、工具経路面を表す多面体を生成する。逆オフセット法は、

処理が単純なためプログラム化が容易であり、計算も安定している。既にこの技術を利用した工具経路の計算方法が数多く提案されており、実用システムも開発されている。われわれが発明した、グラフィックスハードウェアを用いる計算手法は特に高速であり、複数のベンダーがこの技術の商品化に着手している（特開2001-242919 等参照）。

【0005】一般に、金型には凹凸が複雑に入り組んだ形状が多い。このような形状の壁の部分は、等高線状の経路にそって切削工具を移動させると美しく仕上げることができる。逆オフセット法では、工具経路面を覆う点群を $x$   $y$  平面上の格子に基づいて計算するので、点群は $x$  軸方向と $y$  軸方向に均一な間隔で配置される。そのため、工具経路面が $z$  軸方向に大きく変化する壁形状では、隣接する点間の距離が広がり、点群から多面体を生成すると、計算結果に許容できないほど大きな誤差を生じることがある。また垂直な壁の $x$  軸方向と $y$  軸方向の形状変化は、格子間隔ごとに離散的に記録されるため、素朴な方法で多面体化すると、工具経路面に不自然な折れ曲がりが生じてしまうことがある。

【0006】前者の課題は、格子の間隔を十分に小さいものへ変更すれば解決できる。しかし、間隔を $1/n$ にすると格子の総数が $n^2$  倍に増加するため、今度は処理の手間と記憶容量に負荷を生じる。 $z$  軸方向の変化が激しい部分の格子だけを選択的に精密化する方法もあるが、既存手法は決して高速とはいえない。また格子間隔を小さくしても、縦壁に生じる不自然な折れ曲がりには解消できない。そのため、従来、逆オフセット法は、等高線加工のための工具径路生成には利用することが困難であった。本発明は、以上の点に鑑み、3軸の数値制御

(NC) 工作機械用の工具経路を計算する際に、切削工具の製品への削り込みを防止するために生成された、製品の形状を工具の逆形状分膨らませた曲面である高精度な工具経路面を、グラフィックスハードウェアの機能を用いて高速かつ高精度に計算することを目的とする。

【0007】また、本発明は、空間を微小な立方体の集積へ分割し、各立方体の辺と工具経路面の交差の分類に基づいて、工具経路面を精密に近似する多面体を計算する工具経路計算方法、工具経路計算プログラム及びそのプログラムを記録した記録媒体を提供することを目的とする。また、本発明は、 $x$   $y$  平面上の直交格子を $z$  軸方向へ立ち上げたものに相当する、 $x$   $y$  平面に垂直な平面群で工具経路面を切断し、その断面図に基づいて立方体の辺と工具経路面の交点を効率的に計算する工具経路計算方法、工具経路計算プログラム及びそのプログラムを記録した記録媒体を提供することを目的とする。本発明は、グラフィックスハードウェアの機能を用いて、断面図の輪郭線上の点を高速かつ高密度にサンプリングする工具経路計算方法、工具経路計算プログラム及びそのプログラムを記録した記録媒体を提供することを目的とす

る。さらに、本発明は、グラフィックスハードウェアの機能を用いて、サンプリングされた点における輪郭線の接線方向を正確に計算し、その情報に基づいて断面図を高精度化する工具経路計算方法、工具経路計算プログラム及びそのプログラムを記録した記録媒体を提供することを目的とする。

【0008】

【課題を解決するための手段】本発明の解決手段によると、数値制御工作機械に取り付けられた工具を経路に沿って移動させて工作物を加工する際に、工具の逆形状分膨らませた工具経路面を生成するための工具経路面計算プログラムであって、処理部は、加工対象物の多面体モデル情報と工具形状情報とを入力し、記憶部に記憶するステップと、処理部は、多面体モデルの各要素に対して、逆工具の掃引形状を構成する各形状要素を配置するステップと、処理部は、配置された各形状要素に対して、ユニークなID番号、ユニークな色情報、形状要素の幾何情報を記憶部に記憶するステップと、処理部は、入力された逆工具の掃引形状を構成する形状要素に基づき、該掃引形状の最上面を覆う格子状の点群の座標データと、その点における最上面の法線又は接線データを出力するグラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を $x$  軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を $x$  軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、処理部は、グラフィックスハードウェアの隠面消去処理を用いて、配置された形状要素の位置を $y$  軸方向に所定距離だけずらしつつ該処理を繰り返し、最上面の断面線を $y$  軸方向に細分割した座標データと法線又は接線データとを含む点列データを求め、それを記憶部に記憶するステップと、処理部は、求められた隣接する点列データの法線又は接線データに基づき、断面線の折れ曲がり点を検出し、検出された折れ曲がり点の点列データを断面線の点列データへ挿入して記憶部に記憶するステップと、処理部は、格子状の点群に従い、空間を立方体の集積へ分割するステップと、処理部は、記憶部に記憶された点列データを参照し、逆工具の掃引形状の最上面の $x$  軸方向の断面線及び $y$  軸方向の断面線と、分割された各立方体の各辺との交点を算出するステップと、処理部は、算出された交点の情報に基づいて、各立方体の頂点が工具経路面の上下のどちらに存在するかを求めることで、マーチングキューブ法により立方体内部に三角形を配置するステップと、処理部は、各立方体内部に配置された三角形群を、掃引形状の最上面として出力部に出力又は記憶部に記憶するステップとを含む工具経路面計算方法、これら各ステップをコンピュータに実行させるための工具経路面計算プログラム及び工具経路面計算プログラムを記録したコンピュータ読み取り可能な記録媒体が提供される。

【 0 0 0 9 】また、本発明において、前記グラフィックスハードウェアの隠面消去処理は、処理部は、逆工具の掃引形状を構成する形状要素から、形状要素の下向きの部分や他の形状要素の内部に包含されている部分等の掃引形状の最上面に關与しない部分を予め除去するステップと、処理部は、除去されずに残った形状を多面体化するステップと、処理部は、配置された各形状要素に対する、ユニークな ID 番号、ユニークな色情報、形状要素の幾何情報を記憶部から読み取り、各形状要素を構成する多角形に対する色情報をグラフィックスハードウェアへ渡すステップと、グラフィックスハードウェアは、処理部から受けた色情報に基づき、デプスバッファによる隠面消去処理を用いて、各点における掃引形状の最上面の高さを表すデプス値と、各点における色情報を得て、得られたデプス値と色情報を処理部へ渡すステップと、処理部は、グラフィックスハードウェアから受けた各点の座標データとデプス値に基づいて、掃引形状の最上面を示す工具経路面の点列データを生成するステップと、処理部は、色情報のに基づき、記憶部を参照して、最上面の各点に対応する形状要素の ID 及び幾何情報を得て、各点の座標データと幾何情報に従い、各点における法線方向を求めるステップと、処理部は、工具経路面の点群の座標データとその法線又は接線情報を出力するステップを含むようにすることができる。

【 0 0 1 0 】また、本発明において、前記折れ曲がり点を断面線の点列データへ挿入して記憶部に記憶するステップは、処理部は、記憶部を参照し、断面線を表す座標データと、各点における工具経路面の法線又は接線データを読み取るステップと、処理部は、各断面線上の連続する 2 点について、ふたつの接線又は法線方向が所定値より大きく異なる場合には、2 点の間で断面線が折れ曲がり点を有すると判断し、処理部は、新しい折れ曲がり点を計算し、それを点列データに追加するステップと、処理部は、更新された点列データと、更新後の点列の長さを示すデータを記憶又は出力するステップとを含むようにすることができる。

#### 【 0 0 1 1 】

##### 【発明の実施の形態】 1 . 本発明の関連技術

本実施の形態では、本発明者が以前発明した、「工具参照面計算法、工具参照面計算プログラムを記録したコンピュータ読み取り可能な記録媒体及び工具参照面計算装置（発明者：乾正知）特願 2 0 0 0 - 0 5 6 8 4 3」の機能を用いている。ここでは、本発明の関連する技術について述べる。

##### 1 - 1 . 工具経路面（工具参照面）

図 1 は、金型加工で用いられるエンドミルの種類

（（ a ）ボールエンドミル、（ b ）フラットエンドミル、（ c ）ラウンドエンドミル）と、それらの工具経路点（工具参照点）（Cutter reference point）の位置を示す図である。金型加工では、

先端の切刃が半球形のボールエンドミルを利用することが多いが、作業効率が重視される粗加工ではフラットエンドミルを、またなだらかな曲面形状の加工では、フラットエンドミルの周囲に円環状の切刃が取り付けられた、ラウンドエンドミルを用いることもある。3 軸の NC 切削加工では、これらのエンドミルの位置を代表する点（以後、この点を工具経路点（工具参照点）（Cutter reference point）とよぶ）の移動経路を工作機械の制御装置へ与え、希望する形状を削り出す。ボールエンドミルでは、工具経路点（工具参照点）として切刃の中心点を用いることが一般的である。一方フラットエンドミルでは工具先端の中心点を、またラウンドエンドミルでは工具先端の中心点から、円環状の切刃の半径分内側の点を工具経路点（工具参照点）として用いることが多い。

【 0 0 1 2 】図 2 は、製品表面を滑るボールエンドミルの参照点の描く工具経路面（工具参照面）（ a ）と、工具の逆形状の参照点が製品表面を滑る時の掃引形状（ b ）の幾何学的な関係を示す説明図である。ボールエンドミルが金型製品の表面を縦横に滑るとき、ボールエンドミルの工具経路点（工具参照点）の軌跡の描く面の様子を、図 2（ a ）に太線で示した。この面を工具経路面（工具参照面）（Cutter reference surface）とよぶ。工具経路面（工具参照面）が得られれば、ボールエンドミルを、その参照点が常にこの面の上面に存在するように移動させることで、削り込みを生じない工具経路を高速に計算できる。

【 0 0 1 3 】工具経路面（工具参照面）は以下の手順で計算できる。まず、ボールエンドミルを、工具経路点（工具参照点）を中心に 1 8 0 度回転させた逆形状を得る。次にこの逆形状を、図 2（ b ）に示すように、その参照点が常に製品の表面（Designed surface）に存在するように保ちつつ縦横に移動させ、その掃引形状を生成する。このとき掃引形状の最上面が工具経路面（工具参照面）に対応する。なお、この計算法は、フラットエンドミルやラウンドエンドミルを用いた切削加工にも適用できる。図 3 は、製品表面を滑るフラットエンドミルの参照点の描く工具経路面（工具参照面）（ a ）と、工具の逆形状の参照点が製品表面を滑る時の掃引形状（ b ）の幾何的な関係を示す説明図である。図 3（ a ）には、フラットエンドミルを用いて製品表面を切削するときの、工具経路点（工具参照点）の軌跡の描く面の様子を示した。また同図（ b ）には、フラットエンドミルの逆形状を考え、その参照点が製品表面を縦横に滑るとき、掃引形状の状態を示した。

【 0 0 1 4 】同様に、図 4 は、製品表面を滑るラウンドエンドミルの参照点の描く工具経路面（工具参照面）（ a ）と、工具の逆形状の参照点が製品表面を滑る時の掃引形状（ b ）の幾何的な関係を示す説明図である。いずれの場合も、工具経路面（工具参照面）と掃引形状の

最上面が同一形状になっていることが分かる。したがって工具経路面（工具参照面）を得るためには、工具の逆形状の掃引形状を計算し、その最上面を選択する処理をおこなえばよい。つぎに、図 5 は、半径  $r$  のボールエンドミルの逆形状の表面にそって、半径  $R - r$  のフラットエンドミルの逆形状を滑らせた様子（a）、半径  $r$  のボールエンドミルの逆形状の掃引形状を計算し、さらにその上面にそって半径  $R - r$  のフラットエンドミルの逆形状を滑らせた結果（b）を示す説明図である。軸部の半径  $R$ 、円環状の切削の半径  $r$  のラウンドエンドミルの逆形状は、半径  $R - r$  のフラットエンドミルの逆形状を、半径  $r$  の細いボールエンドミルの表面にそって滑らせたときの、掃引形状と見なすことができる（図 5（a）参照）。したがってラウンドエンドミルの逆形状の掃引形状（図 4（b）参照）は、半径  $r$  のボールエンドミルの逆形状を、金型表面にそって滑らせ掃引形状を生成し、次に得られた形状の最上面にそって、半径  $R - r$  のフラットエンドミルの逆形状を滑らせることで計算できる（図 5（b）参照）。ラウンドエンドミルの逆形状は、半径  $r$  のボールエンドミルの逆形状を、半径  $R - r$  のフラットエンドミルの逆形状にそって滑らせたときの掃引形状とも見なせるので、逆の手順でもラウンドエンドミルの逆形状の掃引形状は計算できる。このような理由から、以後は、ボールエンドミルとフラットエンドミルの逆形状の掃引形状を計算する実施の形態に議論を限定するが、これに限定されず適宜応用することができる。

#### 【0015】1 - 2 . 逆オフセット法

掃引形状の計算は、立体モデリングの分野での古典的な問題であり、既に幾つかの計算法が開発されている。しかしこれらの従来手法は、いずれも処理時間や安定性に課題があり、一般に、金型のような複雑な形状の表面を移動する立体に適用することが難しい。われわれは、逆オフセット法による工具経路面（工具参照面）の近似的な計算技術と、3次元グラフィックスハードウェアを利用した処理技術を組み合わせることで、工具経路面（工具参照面）を安定かつ高速に計算するアルゴリズムを発明し、その特許を出願した。以下の議論では、加工対象の製品の形状が多面体であることを仮定する。ただし、各辺の左右に1枚ずつ多角形が接続している、通常の閉じた多面体だけでなく、辺の左または右のどちらか一方にしか多角形が接続していない、「開いた」形状を許す。任意の曲面形状は、希望する精度でこのような多面体として近似表現できる。曲面形状を多面体化する技術は、コンピュータグラフィックスの分野においてよく研究されており、幾つかの高速なアルゴリズムが既に知られている。

【0016】半径  $r$  のボールエンドミルによる切削加工の場合、工具経路面（工具参照面）にあたる掃引形状の上面は、ボールエンドミルの逆形状の上端の、半球形の切削の移動により生成される。そこで工具の逆形状の代

わりに、切刃と同一半径の球を、その中心が常に製品表面に存在するように滑らせた時の掃引形状の計算を考える。製品が多面体形状のとき、球の掃引形状は、多面体の頂点、辺、多角形を、以下の手順で球面、円筒形、厚板形状に置き換えた時の、これらの和形状となる。図 6 は、球の掃引形状を構成する、球面（a）、円筒形（b）、厚板形状（c）の例を示す説明図である。

[頂点] その頂点  $v$  を中心とする、半径  $r$  の球面に置き換える（図 6（a）参照）。

10 [辺] その辺  $e$  を中心軸とする、半径  $r$  の円筒形に置き換える（同図（b）参照）。

[多角形] その多角形  $f$  を中心とする、厚さ  $2r$  の厚板形状に置き換える（同図（c）参照）。

【0017】一方、半径  $r$  フラットエンドミルによる切削加工の場合、工具経路面（工具参照面）にあたる掃引形状の上面は、フラットエンドミルの逆形状の上端の、円板状の切削の移動により生成される。製品が多面体形状のとき、円板の掃引形状は、多面体の辺と多角形を、以下の手順で斜円筒形と厚板形状に置き換えた時の、これらの和形状となる。図 7 は、円板の掃引形状を構成する、斜円筒形（a）と厚板形状（b）の例を示す説明図である。

[辺] その辺  $e$  の両端の頂点に与えられた、半径  $r$  の 2 枚の水平な円板を結ぶ斜円筒形に置き換える（図 7（a）参照）。

[多角形] 多角形  $f$  の周囲の全頂点に、半径  $r$  の水平な円板を配置し、多角形を水平に移動させ、全ての円板の外周と接するように位置決めする。このような位置は二ヶ所考えられる。そこで、各場所に配置された 2 枚の多角形に挟まれた領域に対応する厚板形状を生成し、多角形  $f$  をこの厚板形状に置き換える（同図（b）参照）。複雑な金型形状を精密に多面体化した場合、その多角形数はしばしば数万から数十万枚となる。辺や頂点の数も多角形数に比例するので、上述した手法による掃引形状の計算では、膨大な数の球面、円筒形、厚板形状などの集合演算を繰り返すことになり、実用的ではない。逆オフセット法は、工具経路面（工具参照面）として最終的に必要となるのが、掃引形状の最上面だけであることに着目し、以下の手順で処理することで、工具経路面（工具参照面）の効率的かつ安定な計算を実現する。

40 50 【0018】図 8 は、ボールエンドミル加工用の工具経路面（工具参照面）を得るために、球の掃引形状を構成する球面、円筒形、厚板形状と、格子の中心を通過する  $z$  軸に平行な直線の交差を調べ、最も上側の交点を選択する様子を示す図である。また、図 9 は、フラットエンドミル加工用の工具経路面（工具参照面）を得るために、円板の掃引形状を構成する斜円筒形や厚板形状と、格子の中心を通過する  $z$  軸に平行な直線の交差を調べ、最も上側の交点を選択する様子を示す図である。まず処

理の基準となる直交座標系を、その  $z$  軸がエンドミルの回転軸の上向き方向と一致するように与える（図 8、図 9 参照）。基準座標系の  $x$   $y$  平面上に、 $x$  軸と  $y$  軸に平行かつ等間隔な直交格子を、これから計算する掃引形状の、 $x$   $y$  平面への投影を覆うように生成する。なお、ボールエンドミル加工の場合も、フラットエンドミル加工の場合も、掃引形状の投影図形は、製品形状の投影図形をエンドミルの半径分膨らませたものに相当するので、この膨らんだ図形を覆うように直交格子を生成すればよい。格子の間隔  $w$  を小さくするほど精密な計算が可能になるが、必要な記憶容量が膨大なものになる。工具経路面（工具参照面）を計算する場合には、格子の総数が  $1,000 \times 1,000$  個程度になるように、格子間隔を調整する場合が多い。

【0019】次に各格子点を通過する  $z$  軸に平行な直線を考え、この直線と掃引形状を構成する全ての形状要素の交差を調べる。例えば、ボールエンドミル加工用の工具経路面（工具参照面）を計算する場合には、各格子点を通過する直線と、頂点に配置された球面、辺に配置された円筒形、多角形に配置された厚板形状の交差を調べる。そして図 8 に示すように、最も上側の交点、すなわち  $z$  座標値が最大の交点を選択する。フラットエンドミル加工用に工具経路面（工具参照面）を計算する場合も、図 9 に示すように、各直線と、辺に配置された斜円筒形および多角形に配置された厚板形状の交差を調べ、最も上側の交点を選択する。これらの処理を全ての格子について繰り返すと、工具経路面（工具参照面）を覆う、格子状に配置された稠密な点群を得ることができる。各格子のマス目の角にあたる隣接する 4 個の点を選び、それらを頂点とする 2 枚の三角形を定義する。この処理を全てのマス目について繰り返すと、隙間無く敷き詰められた工具経路面（工具参照面）を表す三角形群を得ることができる。

【0020】1 - 3 . 3次元グラフィックスハードウェアの利用

逆オフセット法では、各格子点を通過する  $z$  軸に平行な直線と、掃引形状を構成する球面や円筒形などとの交点計算を、格子と形状要素の全ての組み合わせについて繰り返す必要がある。われわれの発明した手法は、3次元グラフィックスハードウェアの隠面消去機能を利用することで、この処理を高速化する。3次元コンピュータグラフィックスでは、他の面に隠されて観察者から見えない隠面を消去した画像を生成するために、デプスバッファを用いる。図 10 は、デプスバッファの利用による、二つの立方体の隠面を消去した画像の生成例を示す図である。図示のように配置された 2 個の立方体を図の左側から観察した画像を、平行投影により画面へ描く場合を例に、この技術を説明する。まず、処理の基準となる座標系を、画面上の任意の位置に、その  $x$  軸と  $y$  軸が画面を構成するピクセルの格子と平行になるように、また  $z$

軸の方向が観察者の視線方向と逆向きになるように与える。また各ピクセルと 1 対 1 に対応する要素からなる、デプスバッファと呼ばれる配列を用意し、その全要素に初期値として十分に小さな数値を与えておく。以上の準備の後、立方体を構成する多角形を、平行投影により順に画面へ描画していく。その際に、画面の各ピクセルへ投影される多角形上の点の  $z$  座標値を計算し、対応するデプスバッファ要素の値（これをデプス値とよぶ）と比較する。もしも  $z$  座標値がデプス値よりも大きいときには、ピクセルを多角形の色で染め、その  $z$  座標値を新しいデプス値として記録する。この処理を全ての多角形について繰り返すと、デプスバッファの各要素には、視点に最も近い多角形上の点の  $z$  座標値が格納される。また各ピクセルには、その点の色が染められることになり、結果として隠面を消去した画像が画面に描かれる。この図の例では、印をつけたピクセルには点  $p_3$  の色が染められ、対応するデプスバッファ要素には、この点の  $z$  座標値がデプス値として格納される。観察者の視線方向を、エンドミルの回転軸の下向き方向に定め、画面のピクセル群を  $x$   $y$  平面上の直交格子と対応付けると、

- ・全ての格子点について、そこを通過する  $z$  軸に平行な直線と、球や円板の掃引形状の最上面の交点を計算すること（図 8、図 9 参照）、及び、
- ・掃引形状を構成する全ての形状要素（球面、円筒形、厚板形状など）を平行投影した画像を、デプスバッファを用いて描くこと（図 10 参照）、は、幾何的に等価な処理となる。全ての形状要素を描き終えたとき、デプスバッファの各要素には、対応する直線と掃引形状の最も上側の交点の  $z$  座標値がデプス値として格納されているので、工具経路面（工具参照面）を表す点群を容易に得ることができる。市販されている多くのグラフィックスボードは、デプスバッファを用いた多面体の隠面消去処理を、ハードウェアで直接実行する機能を備えている。したがって掃引形状を構成する球面や円筒形などを適切に多面体近似し、得られた多角形群を平行投影した画像をハードウェアの機能を用いて描画することで、工具経路面（工具参照面）を高速に計算できる。なお、3次元グラフィックス表示装置は、今後の高速化が最も期待されている半導体デバイスなので、その機能を利用することで処理を将来さらに高速化できる。

【0021】2 . 工具経路計算

本発明は、上述した技術を発展させ、逆オフセット法の弱点とされてきた、工具経路面の壁形状での精度低下を解消することを主な目的としている。

2 - 1 . 逆オフセット法の高精度化

図 11 は、急峻な壁形状における逆オフセット法により得られた点群の分布を示す説明図である。図 11 ( a ) は、A 方向から眺めた様子であり、壁面において理想的な工具経路面と計算結果の間に大きな誤差が生じている。図 11 ( b ) は、B 方向から眺めた様子であり、 $x$



方向のなだらかな変化が表現できず、不自然な折れ曲がりが生じている。逆オフセット法は、直交格子の各マス目をz軸方向に引き伸ばした細長い直方体で空間を分割し、その縦方向の辺と逆工具の掃引形状の最上面の交点を計算し多面体化することで、工具経路面を表現する（後述の図12(a)参照）。この手法では、工具経路面の特に「壁」にあたる形状を正確に表現できない。図11に、壁形状の近傍における、格子から垂直に伸びる直線群と工具経路面の交点の様子を示した。工具経路面が水平かつ滑らかな部分（図中の上側や下側の水平な面）では、交点間の距離は格子間隔程度なので、それらの間を多角形で補間しても、加工に十分な精度が得られる。しかし、工具経路面がz軸方向に大きく変化する急峻な壁形状では、隣接する交点間の距離が広がるため、補間後の形状に大きな誤差を生じてしまう場合がある（図11(a)参照）。また、この方法では、水平方向の形状変化を格子間隔ごとに離散的に記録するので、壁の部分のx軸やy軸方向のなだらかな形状変化を表現できず、不自然な折れ曲がりが生じてしまう場合がある（図11(b)参照）。これらの理由から、逆オフセット法で計算された工具経路面は、壁形状を切削加工するための（例えば等高線状の）工具経路の生成には不適当な場合があるとされてきた。

【0022】そこで、本実施の形態では、工具経路面を、xy平面上の格子だけでなく、yz平面上の格子やzx平面上の格子も利用して計算するように逆オフセット法を拡張することで、これらの課題を解決する。xy平面上の格子と同様に、yz平面上にy軸とz軸に平行かつ等間隔な直交格子を、またzx平面上にz軸とx軸に平行な直交格子を用意する。このとき3つの格子の間で、格子間隔が等しくなるように、また格子線の配置にずれが生じないように注意する。xy平面、yz平面、そしてzx平面上の3つの格子を用いて逆オフセット法を行うことは、空間を各格子のマス目に基づく、互いに直交する3本の直方体の積に相当する微小な立方体で分割し、立方体の12本の辺と掃引形状の最上面の交点を計算し工具経路面を表現することと、幾何的に等価である。図12は、従来の逆オフセット法(a)と、本実施の形態の逆オフセット法(b)の比較図である。ここでは、xy平面上の格子のみに基づく従来の逆オフセット法(a)と、xy平面上の格子以外に、yz平面とzx平面上の二つの格子も利用する新しい逆オフセット法(b)の違いを示した。本実施の形態による新しい逆オフセット法では、従来の方法とは異なり、工具経路面を構成する交点群の隣接する2点は、必ず同じ立方体上に存在するので、それらの間の距離は、工具経路面の形状によらず、立方体の各辺の長さで定められる一定値以下となる。本実施の形態では、工具経路面のx軸方向やy軸方向の形状変化も正確に記録できるので、従来の手法よりもはるかに高精度な計算が可能になる。空間を分割

する微小な立方体の各辺と、掃引形状の最上面の交点が得られれば、一般にマーチングキューブ法とよばれるアルゴリズムにより、立方体の内部に工具経路面の一部を表す三角形群を与えることができる。この処理を全ての立方体について繰り返せば、工具経路面を微小な三角形の集合により近似表現できる。立方体の各辺と掃引形状の最上面がたかだか一点で交差することを仮定すると、その情報に基づいて、各辺の両端の頂点が工具経路面の上下どちらに存在するか決められる。そこで立方体の8個の頂点を、工具経路面の上側に存在する頂点と、下側に存在する頂点とに分類する。

【0023】図13は、微小な立方体と逆工具の掃引形状の最上面の間に生じ得る交差パターンを示す図である。分類パターンは、 $2^8 = 256$ 通り考えられるが、対象性を考慮するとパターンを図に示した20通りに削減できる。なお、マーチングキューブ法の原論文では、例えばパターンDとEを同じと見なすことで15パターンとしているが、本実施の形態では面の向きを考慮して、これらを別パターンとして扱う。この図では、工具経路面の上側に位置する頂点を白色で、また下側に位置する頂点を黒色で示している。マーチングキューブ法は、このパターンにしたがって、立方体の内部に工具経路面を表す三角形を配置する。立方体の辺が、工具経路面と複数回交差する場合には、交差回数が偶数回のときには「交差なし」と考え、また交差回数が奇数回の時には「1点で交差」と見なせば、矛盾なく処理を進められる。なお、本実施の形態と部分的に似た表現を採用している研究に、韓国KAISTのChoiらの提案したEZ-map法がある。彼らの論文(Sculptured Surface Machining, Theory and Applications, B.K. Choi and R.B. Jerard, Kluwer Academic Press, 1998)では、本実施の形態と同様に断面線を計算し、それをNC加工命令生成に利用することを提案しているが、断面線の高速かつ高精度な計算法は議論していない。また、この技術とマーチングキューブ法を結合することで、高精度な工具経路面を計算する手法も提案していない。一方、わが国では、藤尾らがBoundary-Map表現という名称で、マーチングキューブ法を利用した製品形状の表現法を提案しているが、彼らの研究では、逆オフセット法との結合は考えられていない(Boundary-Mapデータ構造に基づくCAD/CAMシステムの開発、藤尾、柳下、鈴木、精密工学会誌、Vol.66、No.7、2000)。

【0024】2-2.断面線に基づく交点の算出  
上述した技術により、工具経路面を精密に多面体化するためには、3つの直交格子に基づいて定義される微小な立方体の12本の辺と、逆工具の掃引形状の最上面の交点を計算する必要がある。その一つの手法として、各辺

を含む直線と最上面の交点を計算し、得られた点の座標から、辺と最上面の交差を判定するアルゴリズムが考えられる。12本の辺のうち、z軸に平行な4本の辺については、各辺を含むz軸に平行な直線と最上面の交点はただか1点であり、その座標値の決定プロセスは従来の逆オフセット法と全く同じなので、グラフィックスハードウェアの隠面消去機能を用いた高速な計算法が利用できる。しかし、残りのx軸やy軸と平行な8本の辺の場合には、これらを含むx軸やy軸に平行な直線と掃引形状の最上面は複数の箇所で見交差する場合があるので、隠面消去機能を用いた計算法は、そのままでは適用できない。そこで、本実施の形態では、yz平面やzx平面に平行な面を用いて掃引形状の最上面を切断し、次にその断面線を利用して、x軸やy軸に平行な辺と最上面の交点を計算する手法を用いる。すなわち、立方体の12本の辺のうち、y軸に平行な4本の辺と最上面の交点は、これらの辺を含むyz平面に平行な面を用いて最上面を切断し、その断面線を得ることができれば、これと辺の交差を調べることによって容易に算出できる。このことを、以下に図を用いて示す。

【0025】図14は、掃引形状の断面線に基づいて工具経路面と立方体の各辺の交点を計算した例を示す図である。例えばy軸に平行な2本の辺 $e_3$ や $e_{11}$ と最上面の交点は、これらの辺と隣接する立方体の面を含むyz平面に平行な面を用いて最上面を切断し、その断面線(図中の灰色の曲線)と $e_3$ や $e_{11}$ の位置関係を調べることで計算できる(図では、断面線は $e_3$ とのみ交差している)。 $e_3$ も $e_{11}$ もともに水平な線分なので、これらと断面線の交差の確認や交点の計算は簡単である。y軸に平行な残りの二辺 $e_1$ や $e_9$ と最上面の交点も、これらと隣接する面を含む平面を用いて最上面の断面線を計算すれば、その結果から容易に求められる。同様に、x軸に平行な4本の辺 $e_0$ 、 $e_2$ 、 $e_8$ 、 $e_{10}$ と掃引形状の最上面との交点も、これらの辺を含むzx平面に平行な面を用いて最上面を切断し、その断面線(図中の黒色の曲線)と各辺の位置関係を調べることで計算できる。断面線を得ることができれば、z軸に平行な4本の辺と最上面との交点も容易に計算できるので、マーチングキューブ法を適用するために必要な情報は全て得られることになる。

【0026】2-3.グラフィックスハードウェアによる断面線の算出

上述したような、微小な立方体の面のうち、zx平面に平行な面で掃引形状の最上面を切断した場合の断面線は、以下の手順で計算できる。まずxy平面の格子線のうち、切断面のxy平面への投影に対応するx軸に平行な直線を選び、この直線上に十分に稠密な点を等間隔に配置する。次に、配置された点からz軸に平行にのびる半直線群と、逆工具の掃引形状の最上面の交点を計算する。得られた交点を順に接続していくと、それがこの

切断面による最上面の断面線となる。yz平面に平行な面による断面線も、その面のxy平面への投影に対応するy軸に平行な格子線を選択し、その上に稠密に配置された点からz軸に平行にのびる半直線群と最上面の交点を計算し、それらを接続することで計算できる。さらに、精密な断面線を計算するためには、xy平面上のx軸やy軸に平行な直線上に配置される点の間隔を、十分に小さなものにする必要がある。本実施の形態では、断面線計算のために配置される点の間隔を、xy平面上の10倍の直交格子の間隔の $1/m$ (例えば、 $m$ は8~10程度)に設定した。後述する断面線の精密化技術を併用すると、この程度の間隔でz軸に平行な半直線と最上面の交点を計算しプロットすれば、十分な精度の断面線を得ることができる。

【0027】図15は、隠面消去機能を用いた断面線の算出についての説明図である。この図は、xy平面上のx軸やy軸に平行な格子線上に配置された、断面線計算用の点群を見下ろした様子を示したものである。先に我々発明した、前述の隠面消去機能を利用した逆オフセット法の計算技術(例えば、特願2000-056843参照)を用いると、ちょうど格子線上に位置する点(図では白丸で示した)からz軸に平行にのびる半直線と、掃引形状の最上面との交点は高速に計算できる。しかしそれ以外の点(図では黒四角で示した)については、対応する交点をハードウェアで直接計算することはできない。この点は、掃引形状を構成する図形要素(球面、円筒形、厚板形状など)の位置を微小量ずらしてから、逆オフセット処理することで解決できる。ここで、 $[i, j]$ 格子から $(dx, 0, 0)$ ずれた位置にある点(図中に黒い大きな矢印で指示した)について、そこからz軸に平行にのびる半直線と掃引形状の最上面の交点を計算する場合を考える。掃引形状を構成する全ての図形要素を $(-dx, 0, 0)$ 移動させると、この点はちょうど $[i, j]$ 格子の上へ移動するので、この点からのびる半直線と最上面の交点は、隠面消去機能を用いて高速に計算できる。そこで計算後に、得られた交点を移動前の点に対応するものとして記録する。このとき同時に、図中に複数の灰色の他の小さな矢印で指示した点についても交点が計算される。図形の移動や隠面消去の再実行は、グラフィックスハードウェアにより高速に行われるので、これらに伴う処理時間の増加はごくわずかである。同様の処理を、残りの点についても繰り返すことで、全ての断面線についてプロットすべき交点の位置を高速に計算できる。

【0028】2-4.断面線の精度向上

ここでは、前節に示した手法により計算される断面線の精度を、さらに向上させる技術について述べる。図16は、断面線上の隣接した2点を通過する接線に基づく折れ曲がり点の導出についての説明図である。以下に、正しい断面線が、この図に示すような折れ曲がりを持つ場

合を考える。断面線上の点をプロットする際に、折れ曲がりが生じている点とは異なる点を用いると、それらを接続することで得られる断面線と正しい断面線の間、誤差が生じてしまう。プロットされた点における断面線の接線方向を得ることができれば、その情報に基づいて折れ曲がり点を計算し、その点を加えて断面線をプロットしなおすことで、この点を解決できる。断面線上の隣接した二つのプロット点における接線方向が、ある閾値よりも大きく異なるときには、2点の間のどこかで折れ曲がりが生じていると考える。その場合には各プロット点を通過する接線を計算し、それらの交点を正しい折れ曲がり点と考え断面線を補正する。なお、隣接するプロット点間の距離は十分に小さくすることで、このようにして補正した折れ曲がり点は、正確な折れ曲がり点に非常に近くすることができる。断面線上のプロットされた点における接線方向は、この点における掃引形状の最上面の法線方向が得られれば、この法線方向と切断面の法線方向（ $yz$ 平面に平行な切断面の場合 $x$ 軸方向、 $xz$ 平面に平行な切断面の場合 $y$ 軸方向）の外積として容易に計算できる。つぎに、プロットされた点における最上面の法線方向を、グラフィックスハードウェアの機能を用いて計算する手法について述べる。隠面消去機能を用いた逆オフセット法では、例えばボールエンドミル加工の場合、球の掃引形状を構成する球面と円筒形、そして厚板形状を多面体近似し、それらの画像を描画することで工具経路面を得る。そこで球面、円筒形、厚板形状のそれぞれにユニークなID番号を付与し、それをインデックスとして、それぞれの幾何情報（例えば球面の場合、中心の座標と半径）を予め記録しておく。次に、付与された各IDに対してユニークな色を割り当てる。3次元コンピュータグラフィックスでは、例えば、色を24～32ビットで表現するので、100万通り以上の割り当てが可能である。逆オフセット処理の際には、掃引形状を構成する球面や円筒形、厚板形状を、それぞれのIDの対応する色を用いて着色し描画する。これらの図形の隠面を消去した画像を描き終えた時、画面の各ピクセルには、掃引形状の最上面を構成する球や円筒形、そして厚板形状のIDを表す色が染められている。そこで断面線上の点をプロットするために、各ピクセルに対応するデプス値を取得する際、その色も調べ対応する球面、円筒形、厚板形状のIDを取得する。このIDから記録してある幾何情報を検索し、プロットする点の法線方向を計算する。なお、法線方向を計算する代わりに接線方向について計算してもよい。

#### 【0029】3. ハードウェアの構成

図17は、本実施の形態に関するハードウェアの構成図である。このハードウェアは、中央処理装置である処理部（CPU）1と、データを入力する入力部2と、入力されたデータを記憶する記憶部3と、出力部4と、3次元グラフィックス表示部5とを有する。また、CPU

1、入力部2、記憶部3、出力部4及び3次元グラフィックス表示部（グラフィックスハードウェア）5は、スター又はバス等の適宜の接続手段で接続されている。3次元グラフィックス表示部5はデプスバッファを含み、上述の隠面消去の画像をこのデプスバッファを用いることにより生成するハードウェアの装置である。3次元グラフィックス表示部5は、ハードウェア処理により高速にデプスバッファ処理を行うことができる。また、3次元グラフィックス表示部5は、例えば、ボード、カード又は他の装置として構成され、CPU1等と一体に構成されていても、別個の構成であってもよい。出力部4は、3次元グラフィックス表示部5で生成され、フレームバッファへ描画された隠面消去の画像をディスプレイ上に表示する又は他の装置に出力する装置である。なお、出力部4は、バスに直接接続されるように構成することもできる。

【0030】4. 処理アルゴリズムの流れについて  
添付したフローチャートは、われわれの発明した高精度な工具経路面の計算手順をまとめたものである。このフローチャートは、二つのサブルーチン（サブルーチンAとB）とメインルーチンに分けて記述されている。なお、以下のフローチャートでは、法線方向のデータを用いているが、その代わりに接線方向のデータを用いるようにしても良い。

#### 4-1. サブルーチンA

図18に、サブルーチンAについてのフローチャートを示す。このサブルーチンは、逆工具の掃引形状を構成する形状要素群（ボールエンドミル加工の場合には、球面、円筒形、厚板形状、フラットエンドミル加工の場合には、斜円筒形と厚板形状）を入力すると、グラフィックスハードウェア（3次元グラフィックス表示部）5の隠面消去機能を用いて、掃引形状の最上面を格子状に覆う点群の座標値と、それらの点における最上面の法線方向を出力する。このサブルーチンの処理の基本的な部分は、既に出願済みの技術を用いており、特願2000-56843に記載された内容がここに引用され組み込まれることができる。

【0031】以下に、図を参照して各ステップの処理について説明する。

ステップS100： 処理部1は、逆工具の掃引形状を構成する形状要素群の入力を行う。処理部1は、この情報を記憶部3から読み出すことができる。また、これらの情報を、必要に応じて入力部2や他の記録媒体、外部装置等から入力しても良い。

ステップS101： 処理部1は、入力された形状要素から、明らかに掃引形状の最上面に関与しない部分を予め除去する。具体的には、例えば、形状要素の下向きの部分や、他の形状要素の内部に含まれている部分を検出し取り除く。

ステップS102： 処理部1は、除去されずに残った

形状を多面体化する。

ステップ S 103 : 処理部 1 は、予め形状要素には ID 番号が付与されており、グラフィックスハードウェア ( 3 次元グラフィックス表示部 ) 5 の処理に対応して、各 ID にはユニークな色が割り当ててある。また ID をキーとして、各形状要素の幾何情報がデータベースに格納されている。この割り当てられている色を用いて、( 多面体化された ) 各形状要素を構成する多角形を着色する。着色された多角形のデータを、グラフィックスハードウェア 5 へ渡す。

ステップ S 104 : グラフィックスハードウェア 5 は、x y 平面上の直交格子と 1 対 1 に対応する要素からなる、デプスバッファを用意する。着色した多角形群を z 軸の正方向から観察した画像を、平行投影により画面へ描画する。グラフィックスハードウェア 5 の隠面消去機能が、自動的にデプスバッファの内容を更新する。

ステップ S 105 : グラフィックスハードウェア 5 は、全ての多角形を描き終えたとき、デプスバッファの各デプス値は、対応する格子を通過する z 軸に平行な直線と掃引形状の、最も上側の交点の z 座標値を表している。また各ピクセルの色は、これらの点の対応する形状要素の ID を表している。得られた全てのデプス値とピクセルの色データを、処理部 1 側のソフトウェアの処理へ渡す。

ステップ S 106 : 処理部 1 は、グラフィックスハードウェア 5 から直交格子の位置情報 ( 座標データ ) とデプス値を受信して、位置情報 ( 座標データ ) とデプス値に基づいて掃引形状の最上面を覆う稠密な点群を生成する。

ステップ S 107 : 処理部 1 は、色データから最上面を覆う各点の対応する形状要素の ID を決定し、それを検索キーとしてデータベースから形状要素の幾何情報を得る。点の座標値と得られた幾何情報から、各点における正規化された法線方向を決定する。

ステップ S 108 : 処理部 1 は、掃引形状の最上面を格子状に覆う点群の座標値とその法線方向を出力する。

#### 【 0032 】 4 - 2 . サブルーチン B

図 19 に、サブルーチン B についてのフローチャートを示す。このサブルーチンは、逆工具の掃引形状の最上面の、n 列分の断面線 ( 折れ線 ) を表す  $m * n$  個の点列データと、各点における最上面の ( 正規化された ) 法線方向のデータを入力すると、断面線上の折れ曲がり点を検出し、その点を断面線の点列データへ挿入し出力するものである。なお、ここで、折れ曲がり点の個数は最大 k 個を想定している。新たに検出された折れ曲がり点の挿入方法としては、より効率的な技術も考えられるが、ここでは理解しやすさを優先して記述した。以下に、図を参照して各ステップの処理について説明する。

ステップ S 200 : 処理部 1 は、断面線を表す点列データ  $sec [ n ] [ m * n + k ] [ 3 ]$  と、各点に

おける面の正規化された法線方向を表す  $nrm1$

$[ n ] [ m * n + k ] [ 3 ]$  入力する。または、処理部 1 は、これらデータを記憶部 3 から読み出す。この例では、断面線は n 列分が記録されており、各列のデータは  $m * n$  個の点からなる。ただし、以下の処理で、最大 k 個の点が新たに挿入される可能性があるため、その分記憶領域が記憶部 3 に余分に確保されている。法線方向の記憶領域も同様に定義されている。

ステップ S 201 : 処理部 1 は、ローカル変数 i を 0 に初期化する。i が 0 から n - 1 になるまで i を増やしなが、各列の断面線に対して、以下の処理を繰り返す。

ステップ S 202 : 処理部 1 は、ローカル変数 j を 0 に初期化する。j が 0 から  $final - 2$  になるまで j を増やしなが、各折れ線上の連続する 2 点について、以下の処理を繰り返す。final は最初  $m * n$  に設定されている。

ステップ S 203 : 処理部 1 は、座標  $sec [ i ] [ j ]$  と法線方向  $nrm1 [ i ] [ j ]$  から、 $sec [ i ] [ j ]$  における接線  $tan0$  を得る。処理部 1 は、同様に  $sec [ i ] [ j + 1 ]$  と  $nrm1 [ i ] [ j + 1 ]$  から、 $sec [ i ] [ j + 1 ]$  における接線  $tan1$  を得る。

ステップ S 204 : 処理部 1 は、二つの接線  $tan0$  と  $tan1$  の方向が大きく異なる場合には、2点  $sec [ i ] [ j ]$  と  $sec [ i ] [ j + 1 ]$  の間で、断面線が折れ曲がり点を有すると考える。処理部 1 は、この新しい折れ曲がり点を格納する場所を記憶部 3 内に得るために、 $sec [ i ] [ j + 1 ]$  以降の点列データを、 $sec [ i ] [ j + 2 ]$  以降へ格納しなおす。同様に、処理部 1 は、 $nrm1 [ i ] [ j + 1 ]$  以降の法線データを、記憶部 3 の  $nrm1 [ i ] [ j + 2 ]$  へ格納しなおす。

ステップ S 205 : 処理部 1 は、接線  $tan0$  と  $tan1$  の交点として、新しい折れ曲がり点を計算し、これを  $sec [ i ] [ j + 1 ]$  へ格納する。

ステップ S 206 : 処理部 1 は、点列データが 1 個増えたので、その分だけ j と final と増やす。

ステップ S 207 : 処理部 1 は、更新された点列データ  $sec$  と、更新後の点列の長さを示す final を出力する。

#### 【 0033 】 4 - 3 . メインルーチン

図 20 及び図 21 に、メインルーチンについてのフローチャート ( 1 ) 及び ( 2 ) を示す。以下に、図を参照して各ステップの処理について説明する。

ステップ S 300 : 処理部 1 は、入力部 2 からの指示により又は記憶部 3 に記憶されたデータを読み出すことにより、加工対象物の多面体モデルの情報、工具形状の情報 ( 工具のタイプや切刃の形状など )、逆オフセット法のために用意する直交格子の分割数 n を入力する。さ

らに、処理部 1 は、断面線を得るために格子を再分割する数  $m$  と、各断面線を精密化するために挿入する点の数の上限  $k$  を入力又は読み出す。処理部 1 は、以下のステップのように、断面線を計算する際には、精度を上げるために、格子の分割よりも  $m$  倍細かな分割で、断面線上の点をプロットする。また、処理部 1 は、以下のステップのように、各断面線を精密化するために、最大  $k$  個の断面線の折れ曲がりを検出し、そこに新しい点を挿入する。

ステップ S 3 0 1 : 処理部 1 は、工具の軸方向を  $z$  軸とする座標系を用意し、 $xy$  平面上に、逆オフセット後のモデルの投影図形を覆うように、 $n \times n$  サイズの直交格子を生成する。

ステップ S 3 0 2 : 処理部 1 は、 $yz$  平面や  $zx$  平面上に、 $xy$  平面上の格子のマスサイズ  $d$  や位置に整合的で、逆オフセット後の多面体モデルの投影図形を覆う直交格子を生成する。

ステップ S 3 0 3、S 3 0 4 : 処理部 1 は、記憶部 3 に、 $x$  軸方向と  $y$  軸方向のそれぞれについて断面線上の点列の座標を記録するために、領域  $xsec$  と  $ysec$  を確保する。また、処理部 1 は、記憶部 3 に、各点における逆工具の掃引形状の法線方向を記録するために、領域  $xnrm1$  と  $ynrm1$  を確保する。前述のように、断面線は  $n$  列分が記録され、各列のデータは  $m \times n$  個の点からなる。断面線の精密化処理で、最大  $k$  個の点が新たに挿入される可能性があるため、記憶部 3 にその分の記憶領域が余分に確保されている。

ステップ S 3 0 5、S 3 0 6 : 処理部 1 は、逆工具の掃引形状を定義するために、多面体モデルの各頂点、辺、多角形に形状要素を配置する。具体的には、例えば、工具形状がボールエンドミルの場合には、頂点、辺、多角形のそれぞれに球面、円筒形、厚板形状を配置し、一方、フラットエンドミルの場合には、辺と多角形に斜円筒形と厚板形状を配置する。

ステップ S 3 0 7、S 3 0 8 : 処理部 1 は、配置された各形状要素（球、円筒形、厚板形状などの各要素）にユニークな ID 番号を与える。次にその ID をインデックスとして用いて、形状要素の幾何情報（球面、円筒形、厚板形状、斜円筒形、厚板形状、等）をデータベースへ記録する。さらに各 ID に 1 対 1 に対応する色を割り当てる。

ステップ S 3 0 9 : 処理部 1 は、カウンタ  $ii$  を 0 に初期化し、カウンタ  $ii$  を一つずつ増やしながら、これが  $m$  と等しくなるまで、ステップ S 3 1 0 ~ S 3 1 3 までの処理を繰り返し実行する。この繰り返し処理により、 $x$  軸方向の  $n$  列分の断面線群が得られる。

ステップ S 3 1 0、S 3 1 1、S 3 1 2、S 3 1 3 : 処理部 1 は、配置された形状要素の位置を  $[(-d/m) * ii, 0, 0]$  だけずらした上で、サブルーチン A を起動する。すると、掃引形状の最上面

を覆う格子状の点群が  $[(-d/m) * ii, 0, 0]$  だけずれた位置に得られるので、処理部 1 は、これらの座標を  $[(d/m) * ii, 0, 0]$  ずらし、正しい位置に修正する。その上で、格子  $[i, j]$  に対応する点の座標を、記憶部 3 の  $xsec[j][m * i + ii]$  へ格納する処理を、全ての点について繰り返す。サブルーチン A は、得られた点群における最上面の法線方向もリターンするので、処理部 1 は、座標値と同様に、格子  $[i, j]$  に対応する法線方向を、記憶部 3 の  $xnrm1[j][m * i + ii]$  へ格納する処理を、全ての法線方向について繰り返す。

ステップ S 3 1 4 : 処理部 1 は、カウンタ  $ii$  を 0 に初期化し、カウンタ  $ii$  を一つずつ増やしながら、これが  $m$  と等しくなるまで、ステップ S 3 1 5 ~ S 3 1 8 までの処理を繰り返し実行する。この繰り返し処理により、今度は  $y$  軸方向の  $n$  列分の断面線群が得られる。

ステップ S 3 1 5、S 3 1 6、S 3 1 7、S 3 1 8 : 処理部 1 は、実質的な処理の内容は、ステップ S 3 1 0 ~ S 3 1 3 と同じだが、今度は  $y$  軸にそって図形をずらしてからサブルーチン A を起動する。処理部 1 は、サブルーチン A により得られた、掃引形状の最上面を覆う格子状の点群と法線方向を、 $[i, j]$  格子に対応する点の座標が記憶部 3 の  $ysec[i][m * j + ii]$  に、またその点における法線方向が記憶部 3 の  $ynrm1[i][m * j + ii]$  に格納されるように、処理を繰り返す。

ステップ S 3 1 9、S 3 2 0 : 処理部 1 は、サブルーチン B を起動して、断面線  $xsec$  と  $ysec$  を精密化する。即ち、処理部 1 は、折れ曲がり点を検出し、その点を点列データに挿入するように記憶部 3 に記憶する。

ステップ S 3 2 1 : 処理部 1 は、 $xy$  平面、 $yz$  平面、 $zx$  平面上の三つの格子に基づいて、空間を微小な立方体の集積へ分割する。

ステップ S 3 2 2 : 処理部 1 は、記憶部 3 に記憶された掃引形状の最上面の  $x$  軸方向の断面線  $xsec$  と、 $y$  軸方向の断面線  $ysec$  に基づいて、最上面と各立方体の 12 本の辺との交点を算出する。

ステップ S 3 2 3 : 処理部 1 は、交点の情報に基づいて、マーキングキューブ法により、立方体内部に微小な三角形を配置する。

ステップ S 3 2 4 : 処理部 1 は、全立方体内部に配置された三角形群を、掃引形状の最上面（工具経路面）として出力部 4 に出力する。また、処理部 1 は、このデータを記憶部 3 に記憶する。なお、処理部 1 は、初期データ、処理により求められた中間データまたは最終データ等を適宜記憶部 3 に記憶し、そこから読み出し、出力部 4 に出力するようにしてもよい。また、処理部 1 は、得られた工具経路面に基づき、工具経路を求めるステップをさらに含んでも良い。さらに、処理部 1 は、工具経路面及び / 又は工具経路を表示するステップをさらに含む

ようにしても良い。

#### 【0034】5. 実例と評価

本実施の形態の有効性を検証するために、グラフィック  
 スハードウェアの隠面消去機能を利用して、高精度の工  
 具経路面を計算するプログラムを作成し、計算実験をお  
 こなった。計算には、格子の総数が  $1,000 \times 1,000$  になるように調整した直交格子を用いた。利用した  
 計算機は、Pentium IV (商標) (1.7 GHz) を CPU とする、主記憶 2 GB のパーソナルコンピ  
 ュータである。このコンピュータには、GeForce 10  
 3 Ti500 とよばれる 3 次元グラフィックス表示の  
 ための専用ハードウェアが備えられている。描画する多  
 角形の形状を、OpenGL とよばれるライブラリの仕  
 様にしたがってプログラムに記述すると、デブスバッ  
 ファを用いた隠面消去をハードウェアで実行する機械語コ  
 ードを、コンパイラが自動生成する。図 22 は、30,  
 528 枚の多角形からなる多面体の一例を示す図であ  
 る。図 23 に、図 17 に示した多角形数 30, 528 枚  
 の多面体を、半径 1 mm のボールエンドミルを用いて切  
 削加工するために、従来の逆オフセット法を用いて工具  
 20 経路面を計算した結果を示す。図 24 は、通常の逆オフ  
 セット法の処理結果を部分拡大した画像の一例を示す図  
 である。この図は図 18 の縦壁の部分を拡大表示したも  
 のである。壁に垂直な「折れ曲がり」が生じており、工  
 具経路面の精度が不十分であることが分かる。図 25  
 は、本実施の形態による高精度な逆オフセット法の処理  
 結果を部分拡大した画像の一例を示す図である。この図  
 は、同じ切削工具のための工具経路面を、今回提案した  
 高精度な逆オフセット法を用いて計算した結果を示す。  
 従来の方法による結果とは異なり、縦壁の部分から折れ  
 30 曲がりが消えている。また、この図では、誤差に埋もれ  
 て観察することができなかった微細な凹凸が現れている  
 ことから、はるかに高精度な工具経路面が得られている  
 ことが分かる。図 26 は、高精度な逆オフセット法によ  
 り得られた工具経路面を用いて計算した、等高線状の工  
 具経路の一例を示す図である。図 27 は、高精度な工具  
 経路面に基づいて計算された、等高線状の工具経路を用  
 いた切削加工シミュレーションの結果の一例を示す図で  
 ある。このような加工シミュレーションの画像から、削  
 り込みの生じない適切な工具経路が生成されていること  
 40 が分かる。図 28 は、従来の逆オフセット法により得ら  
 れた工具経路面を用いて工具経路を計算し、切削加工シ  
 ミュレーションを行った結果の一例を示す図である。二  
 つのシミュレーション結果の比較から、本実施の形態に  
 よる新しい逆オフセット法に基づく工具経路のほうが、  
 製品の縦壁部分をはるかに美しく仕上げられることが分  
 かる。以上のように、本実施の形態により得られる高精  
 度な工具経路面を用いて、金型の壁面を等高線状に切削  
 する工具経路を計算する実験的なプログラムを開発し  
 た。等高線状の工具経路は、工具経路面を  $x-y$  平面に平

行な面で切断し断面線を得る処理を、切断面の高さを変  
 えながら繰り返しおこなうことで計算できる。各断面線  
 の計算は、工具経路面を覆う微小な三角形と切断面の交  
 線を得る処理と、得られた微小な交線を接続し 1 本の曲  
 線にまとめる処理に分けられる。われわれの発明した新  
 しい逆オフセット法では、工具経路面を構成する三角形  
 群は、空間を分割する  $x$ 、 $y$ 、 $z$  軸に平行な微小な立方  
 体の内部に定義されている。この立方体の情報を用いる  
 と、以下の理由から断面線の計算を高速化できる。・切  
 10 断面の高さから、この面と交差する立方体を予め絞り込  
 むことができる。絞り込まれた立方体の内部に格納され  
 ている三角形だけが、切断面と交差し得る。・ある立方  
 体の内部の三角形が切断面と交差するとき、立方体間の  
 隣接関係を用いることで、得られた交線と接続し得る交  
 線を含む立方体を絞り込める。なお、本発明では、工具  
 は、ボールエンドミル、フラットエンドミル及びラウン  
 ドエンドミルに限らず、切削工具、型どり用工具等の適  
 宜の工具が使用可能であり、また、掃引形状は球面及び  
 円筒形に限らず工具の先端形状により楕円面、多角形  
 20 面、凹凸面など適宜の形状のものなどにも使用できる。  
 また、工作物は金型に限らず、プラスチック、金属等の  
 各材料の様々な工作物にも本発明を適用することが可能  
 である。上述の説明では、主に、必要な又は全デブスデ  
 ータ、必要な又は全多角形データを処理部と 3 次元グラ  
 フィックス表示部等に入力又は出力しているが、必要な  
 又は全ての各データに限らず、一部の各データをやりと  
 りするようにしても良い。さらには、表面の近似として  
 多面体及び多角形として三角形を例に説明したが、これ  
 30 に限らず適宜の多角形を用いることができる。本発明の  
 工具経路計算方法又は工具経路計算装置・システムは、  
 その各手順をコンピュータに実行させるための工具経路  
 計算プログラム、工具経路計算プログラムを記録したコ  
 ンピュータ読み取り可能な記録媒体、工具経路計算プロ  
 グラムを含みコンピュータの内部メモリにロード可能な  
 プログラム製品、そのプログラムを含むサーバ等のコン  
 ピュータ、等により提供されることができる。

#### 【0035】

【発明の効果】本発明によると、以上のように、3 軸の  
 数値制御 (NC) 工作機械用の工具経路を計算する際  
 40 に、切削工具の製品への削り込みを防止するために生成  
 された、製品の形状を工具の逆形状分膨らませた曲面で  
 ある高精度な工具経路面を、グラフィックスハードウェ  
 アの機能を用いて高速且つ高精度に計算することができ  
 る。また、本発明によると、空間を微小な立方体の集積  
 へ分割し、各立方体の辺と工具経路面の交差の分類に基  
 づいて、工具経路面を精密に近似する多面体を計算する  
 ことができる。また、本発明によると、 $x-y$  平面上の直  
 交格子を  $z$  軸方向へ立ち上げたものに相当する、 $x-y$  平  
 50 面に垂直な平面群で工具経路面を切断し、その断面図に  
 基づいて立方体の辺と工具経路面の交点を効率的に計算

することができる。本発明によると、グラフィックスハードウェアの機能を用いて、断面図の輪郭線上の点を高速かつ高密度にサンプリングすることができる。さらに、本発明によると、グラフィックスハードウェアの機能を用いて、サンプリングされた点における輪郭線の接線方向を正確に計算し、その情報に基づいて断面図を高精度化することができる。

#### 【図面の簡単な説明】

【図 1】(a) は、金型加工で用いられるボールエンドミルとその工具参照点の位置を示し、(b) は、同じくフラットエンドミルとその工具参照点の位置を示し、(c) は、同じくラウンドエンドミルとその工具参照点の位置を示す説明図。

【図 2】(a) は、製品表面を滑るボールエンドミルの参照点の描く工具参照面を示し、(b) は、工具の逆形状の参照点が製品表面を滑る時の掃引形状を示す説明図。

【図 3】(a) は、製品表面を滑るフラットエンドミルの参照点の描く工具参照面を示し、(b) は、工具の逆形状の参照点が製品表面を滑る時の掃引形状を示す説明図。

【図 4】(a) は、製品表面を滑るラウンドエンドミルの参照点の描く工具参照面を示し、(b) は、工具の逆形状の参照点が製品表面を滑る時の掃引形状を示す説明図。

【図 5】(a) は、半径( $r$ )のボールエンドミルの逆形状の表面にそって、半径( $R - r$ )のフラットエンドミルの逆形状を滑らせた様子を示し、(b) は、半径( $r$ )のボールエンドミルの逆形状の掃引形状を計算し、さらにその上面にそって半径( $R - r$ )のフラット

エンドミルの逆形状を滑らせた結果を示す説明図。

【図 6】(a) は、ボールエンドミルによる切削加工での、球の掃引形状を構成する、球面を示し、同じく(b) は、円筒形を示し、同じく(c) は、厚板形状の一例を示す図。

【図 7】(a) は、フラットエンドミルによる切削加工での、円板の掃引形状を構成する、斜円筒形を示し、同じく(b) は、厚板形状の一例を示す図。

【図 8】ボールエンドミル加工用の工具参照面を得るために、球の掃引形状を構成する球面、円筒形、厚板形状と、格子の中心を通過する Z 軸に平行な直線の交差を調べ、最も上側の交点を選択する様子を示す図。

【図 9】フラットエンドミル加工用の工具参照面を得るために、円板の掃引形状を構成する斜円筒形や厚板形状と、格子の中心を通過する Z 軸に平行な直線の交差を調べ、最も上側の交点を選択する様子を示す図。

【図 10】デブスバッファの利用による二つの立方体の

隠面消去画像の説明図。

【図 11】急峻な壁形状における逆オフセット法により得られた点群の分布を示す説明図。

【図 12】従来の逆オフセット法(a)と、本実施の形態の逆オフセット法(b)の比較図。

【図 13】微小な立方体と逆工具の掃引形状の最上面の間に生じ得る交差パターンを示す図。

【図 14】掃引形状の断面線に基づいて工具経路面と立方体の各辺の交点を計算した例を示す図。

10 【図 15】隠面消去機能を用いた断面線の算出についての説明図。

【図 16】断面線上の隣接した 2 点を通過する接線に基づく折れ曲がり点の導出についての説明図。

【図 17】本発明に関するハードウェアの構成図。

【図 18】サブルーチン A についてのフローチャート。

【図 19】サブルーチン B についてのフローチャート。

【図 20】メインルーチンについてのフローチャート(1)。

20 【図 21】メインルーチンについてのフローチャート(2)。

【図 22】30, 528 枚の多角形からなる多面体の一例を示す図。

【図 23】前図に示した多面体を、半径 1 mm のボールエンドミルで切削加工する場合の工具経路面の一例を示す図。

【図 24】通常の逆オフセット法の処理結果を部分拡大した画像の一例を示す図。

【図 25】本実施の形態による高精度な逆オフセット法の処理結果を部分拡大した画像の一例を示す図。

30 【図 26】高精度な逆オフセット法により得られた工具経路面を用いて計算した、等高線状の工具経路の一例を示す図。

【図 27】高精度な工具経路面に基づいて計算された、等高線状の工具経路を用いた切削加工シミュレーションの結果の一例を示す図。

【図 28】従来の逆オフセット法により得られた工具経路面を用いて工具経路を計算し、切削加工シミュレーションを行った結果の一例を示す図。

【図 29】金型加工における削り込みの一例を示す図。

40 【符号の説明】

1 処理部

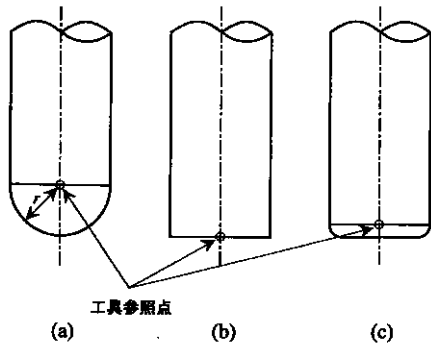
2 入力部

3 記憶部

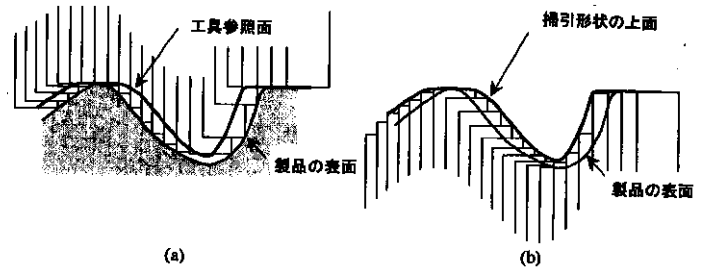
4 出力部

5 3次元グラフィックス表示部(グラフィックスハードウェア)

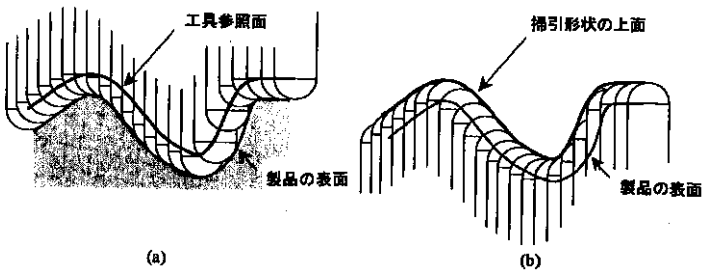
【 図 1 】



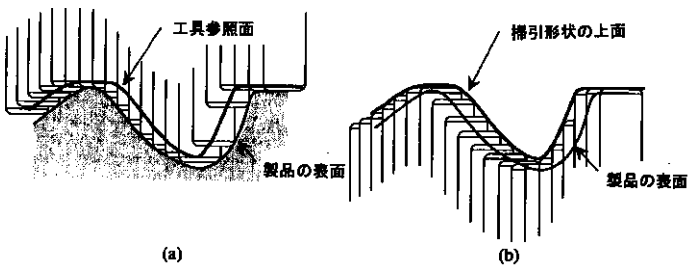
【 図 3 】



【 図 2 】

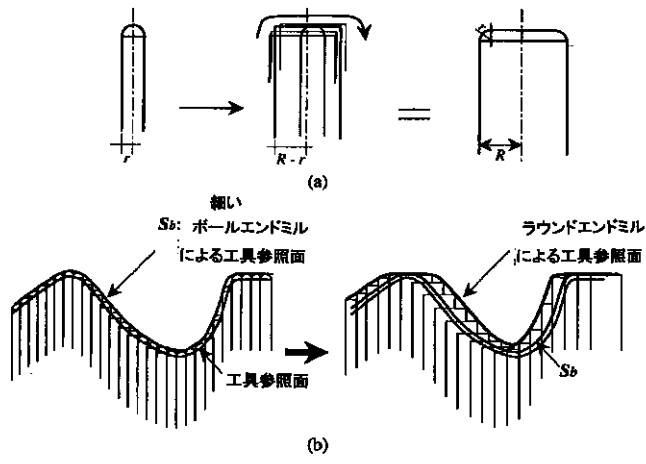


【 図 4 】

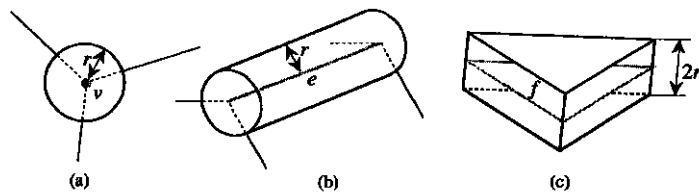




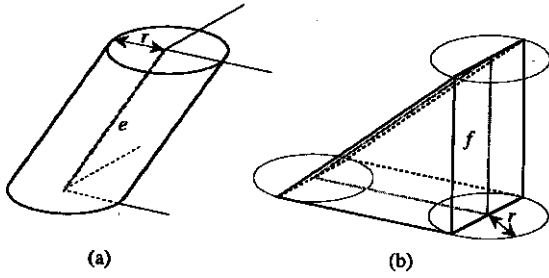
【 図 5 】



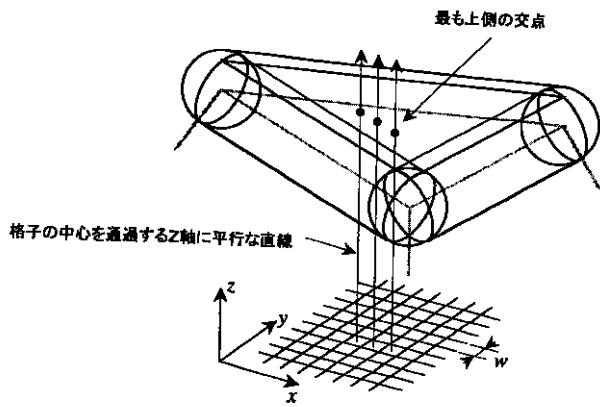
【 図 6 】



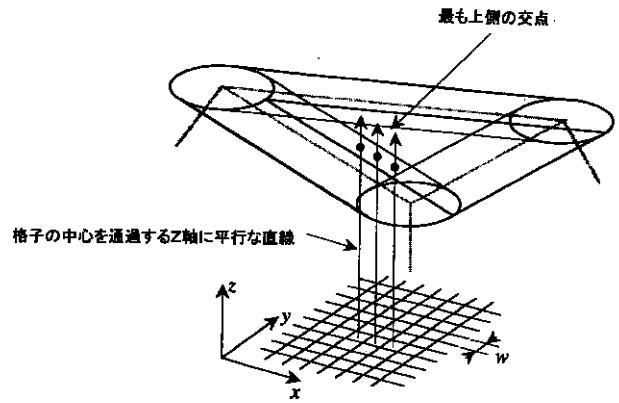
【図7】



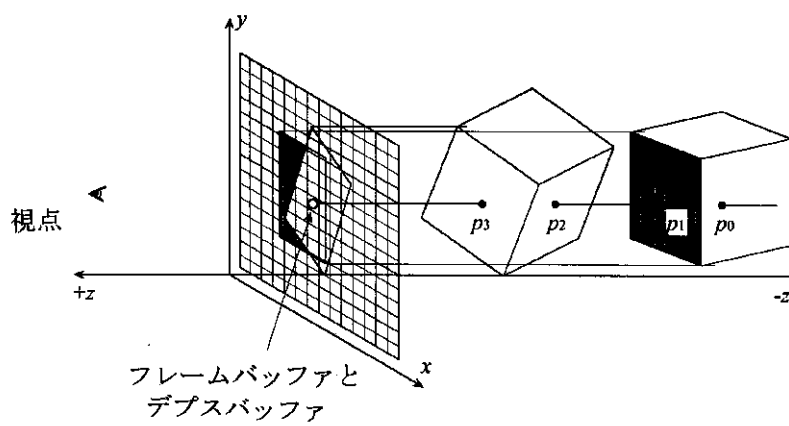
【図8】



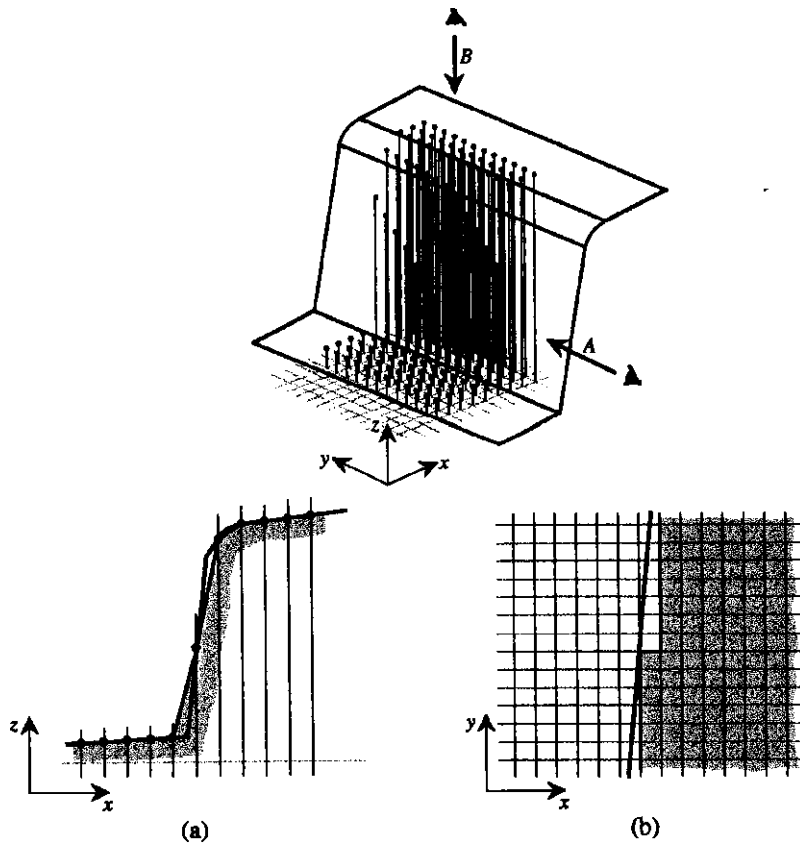
【図9】



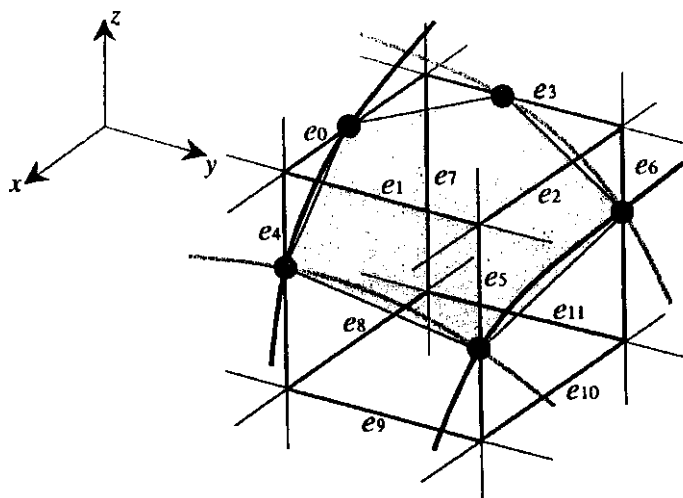
【図10】



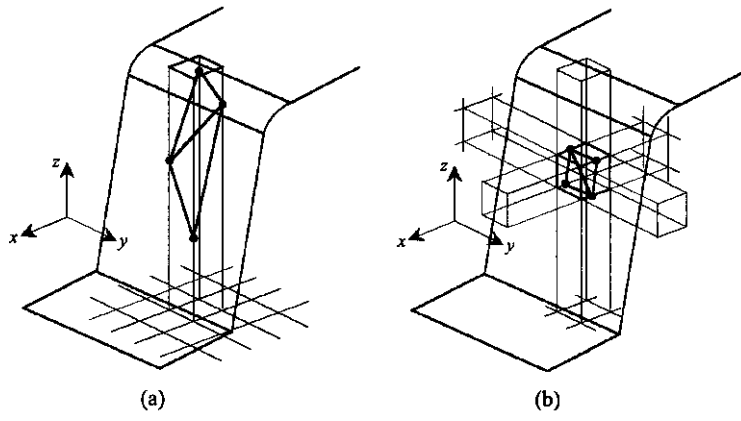
【図11】



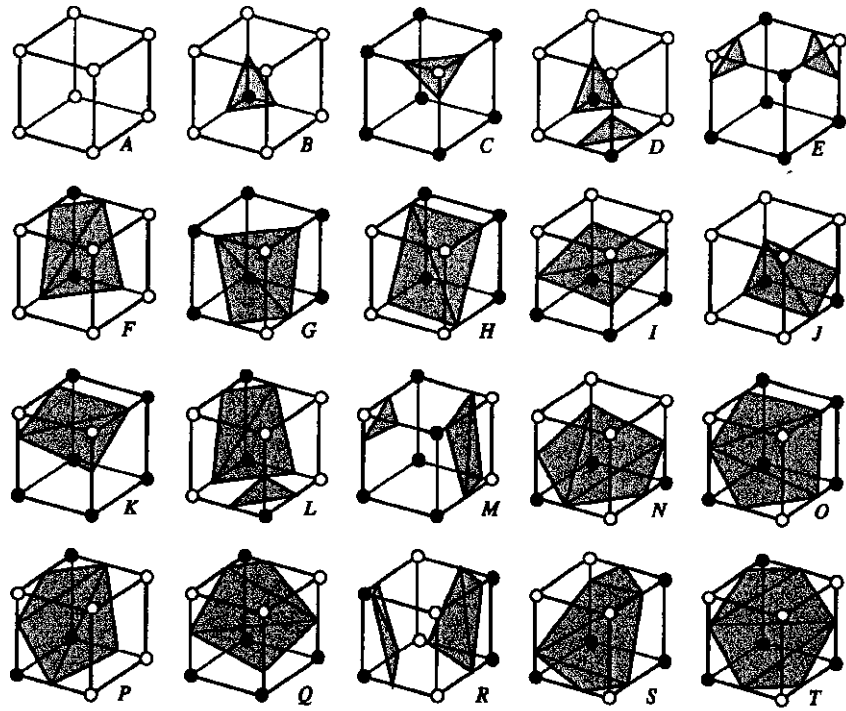
【図14】



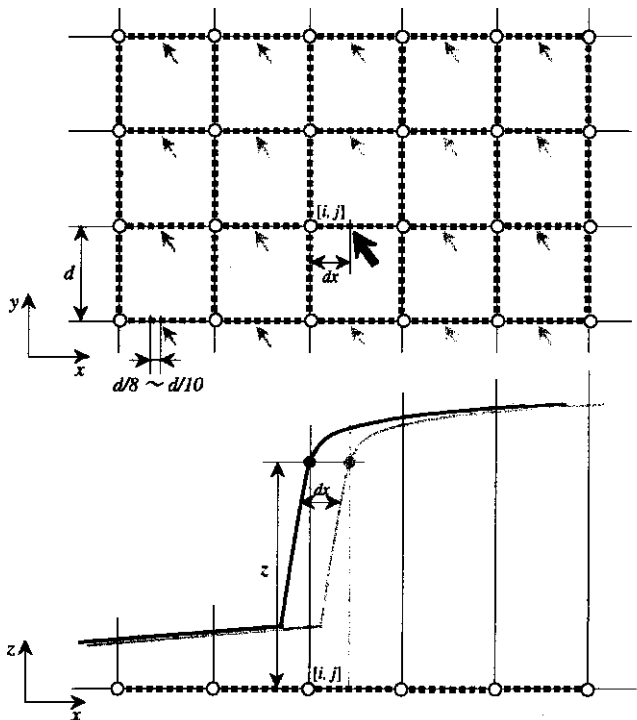
【図12】



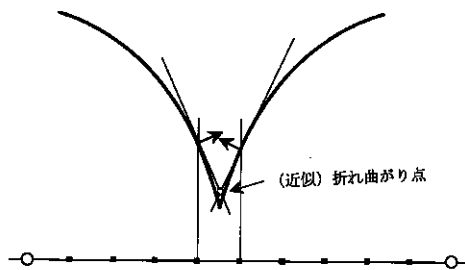
【図13】



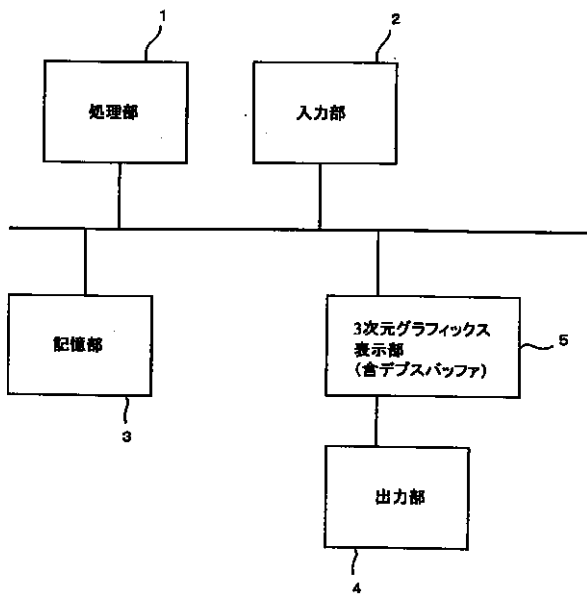
【図15】



【図16】

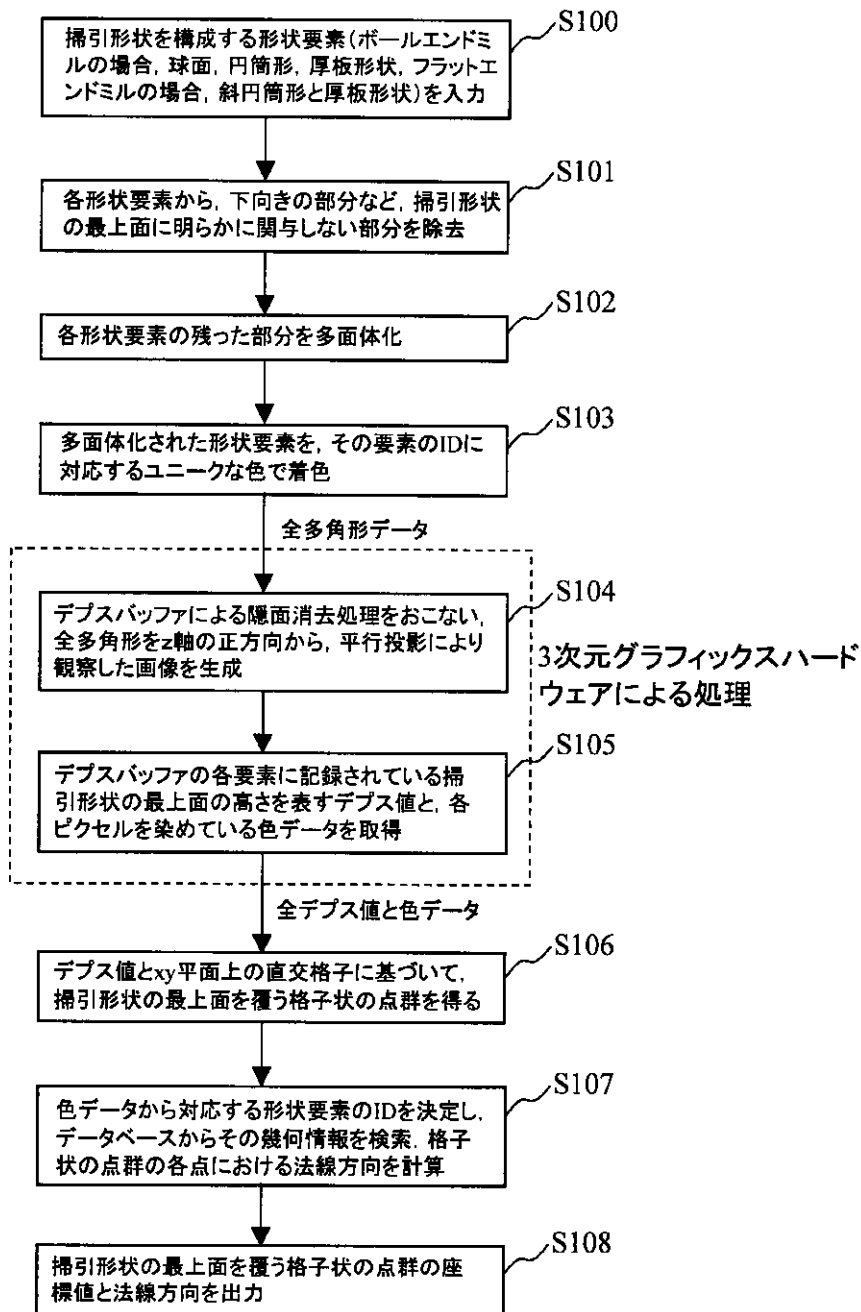


【図17】



【図18】

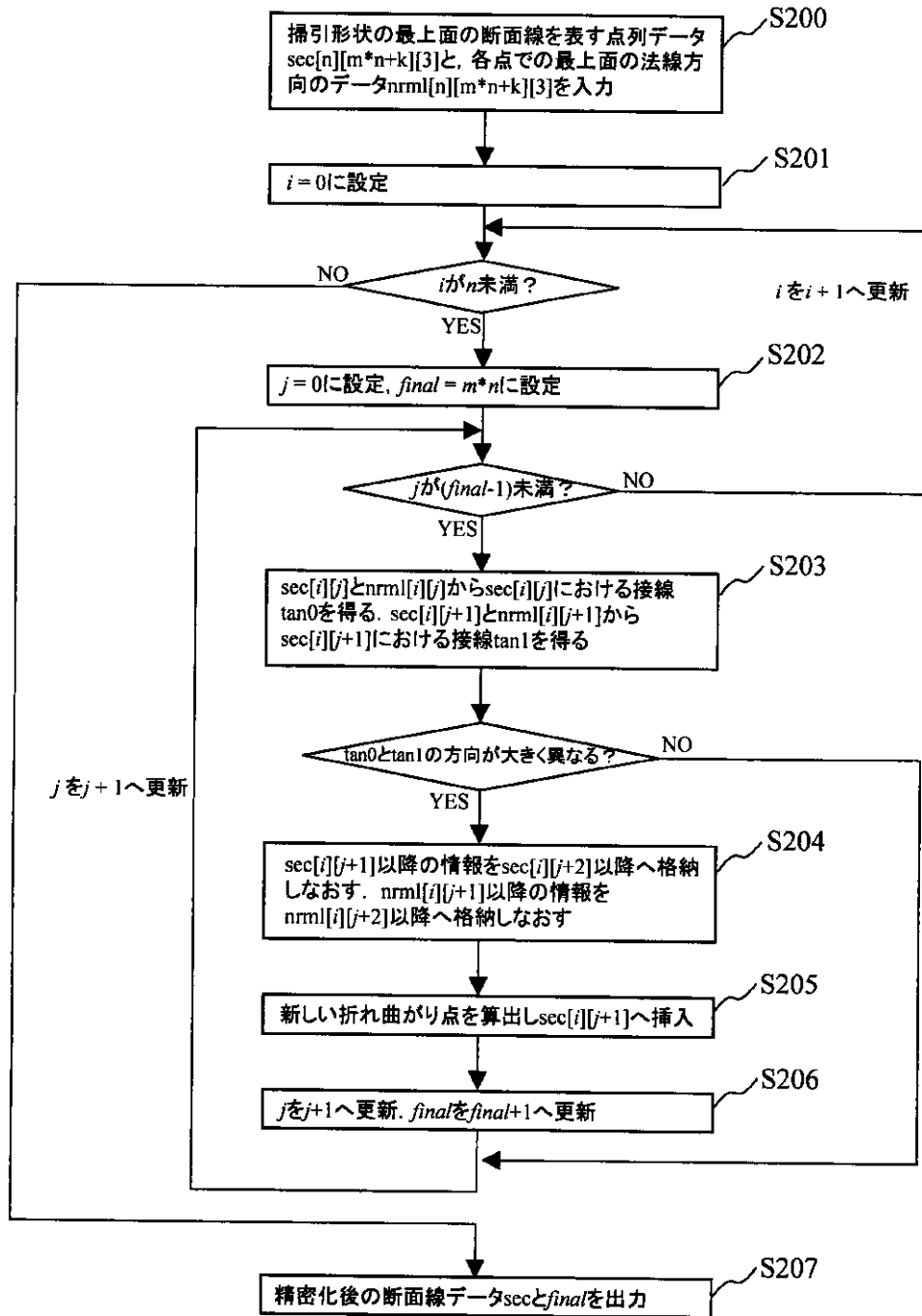
### サブルーチンA



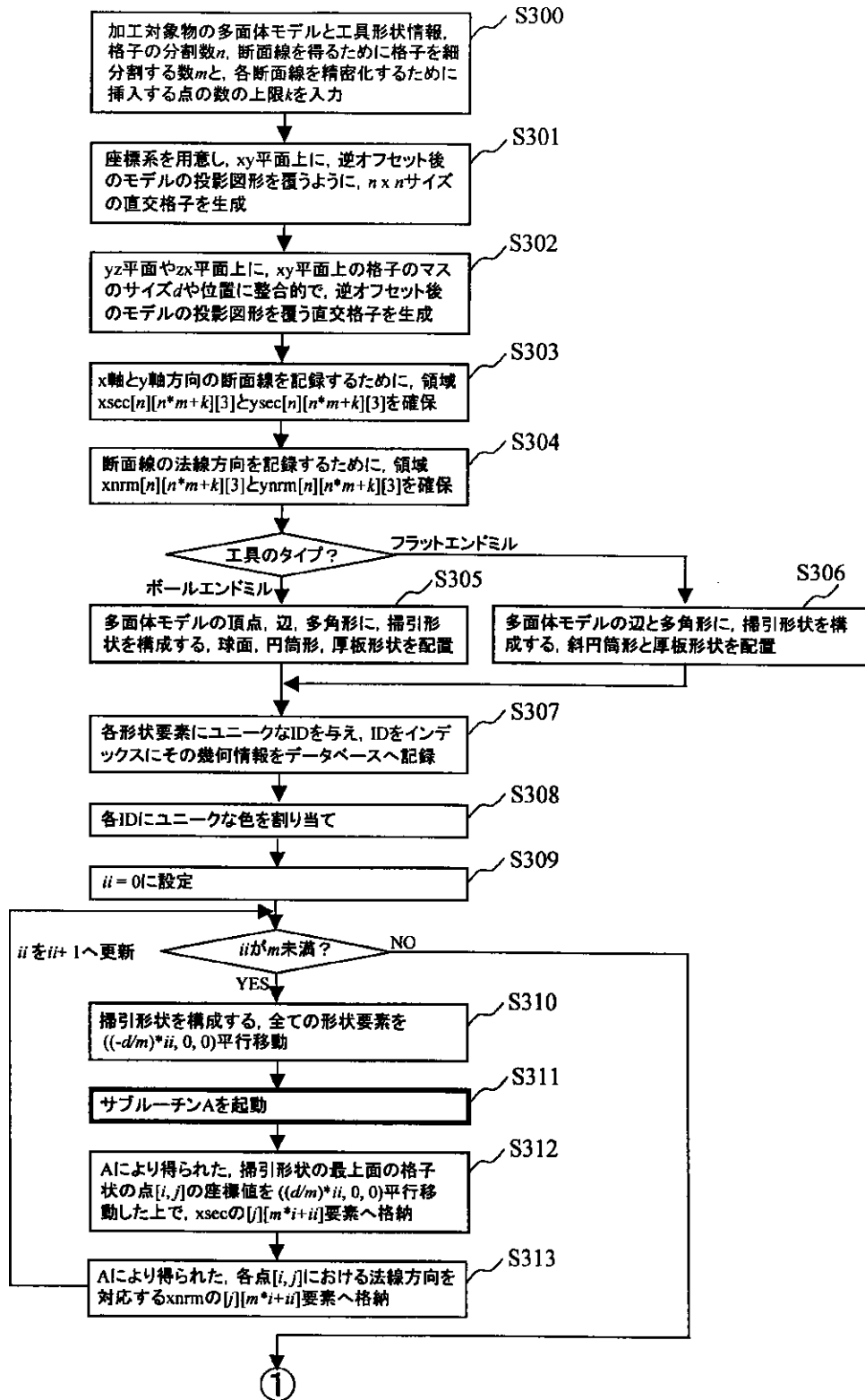


【図 19】

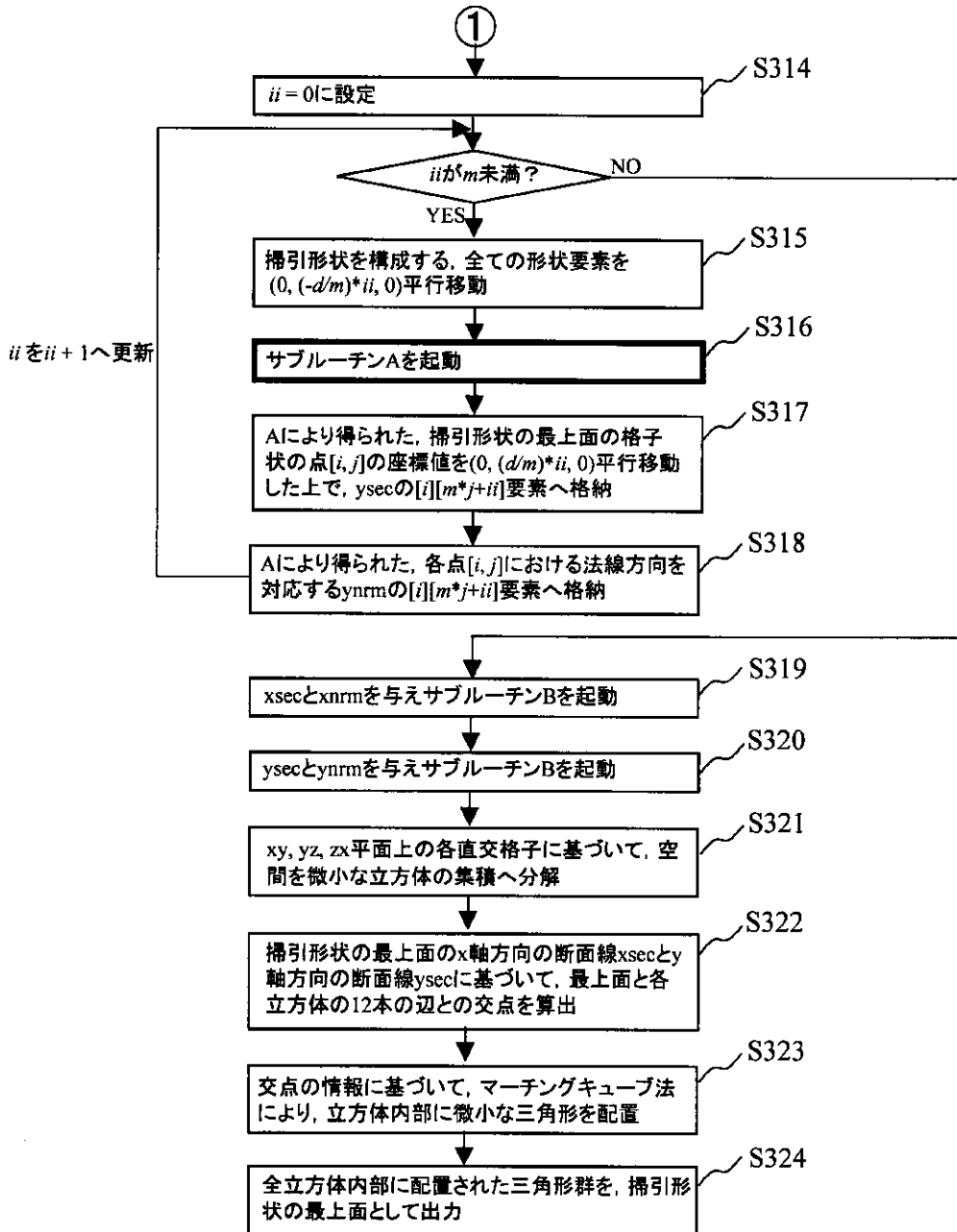
### サブルーチンB



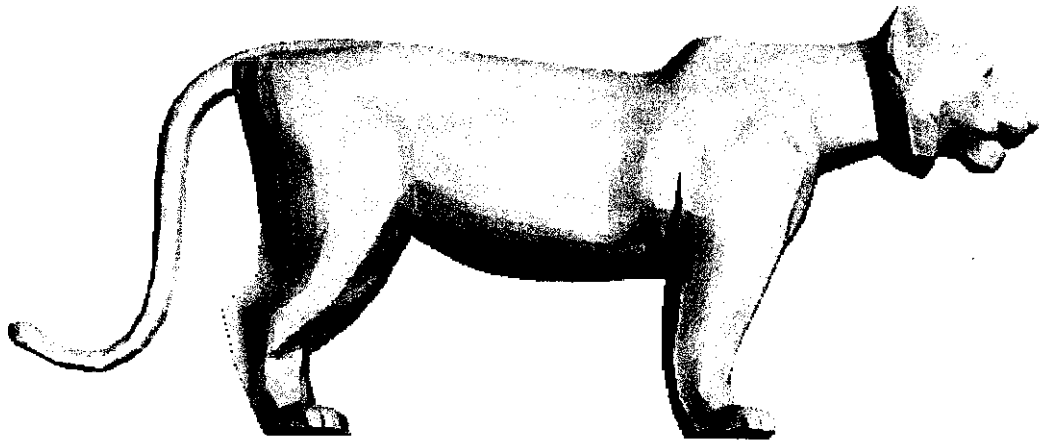
【図 20】



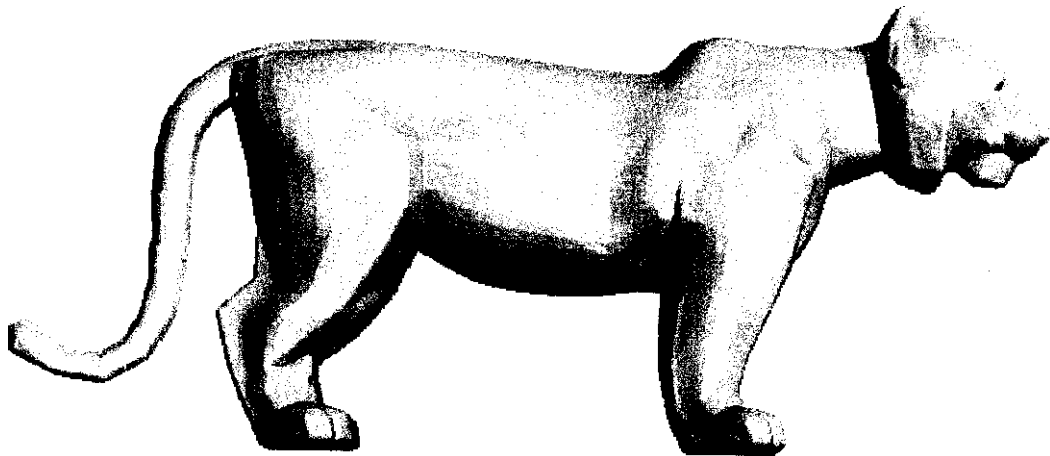
【図 2 1】



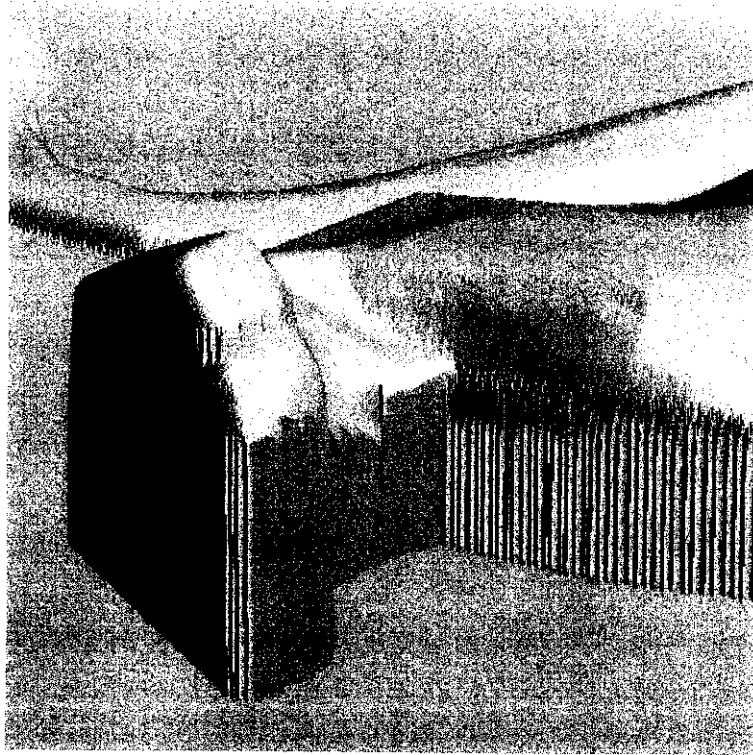
【図22】



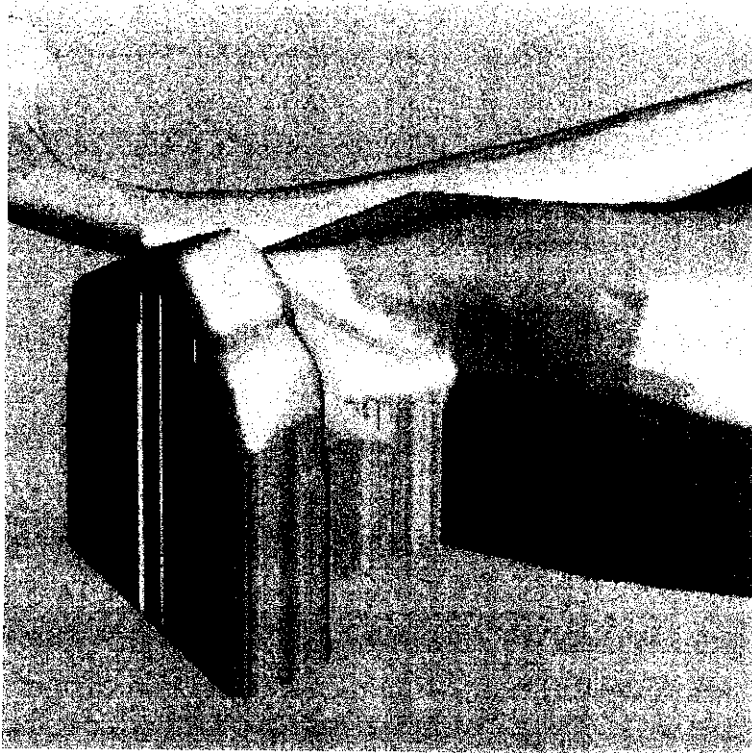
【図23】



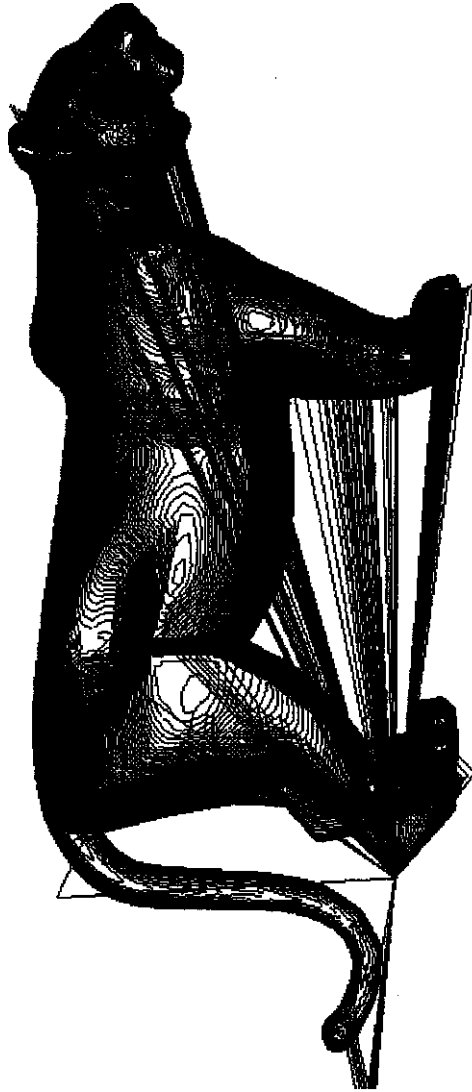
【図24】



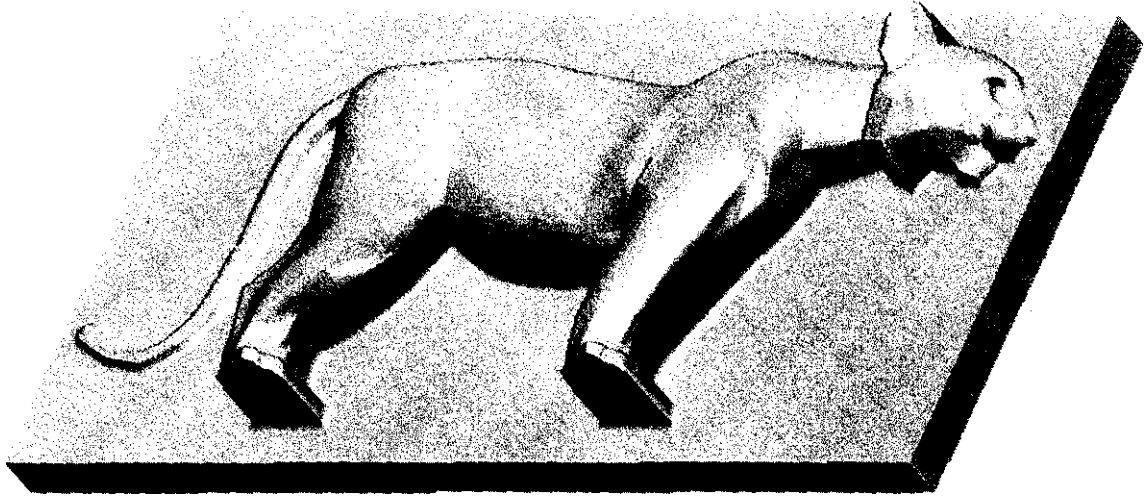
【図25】



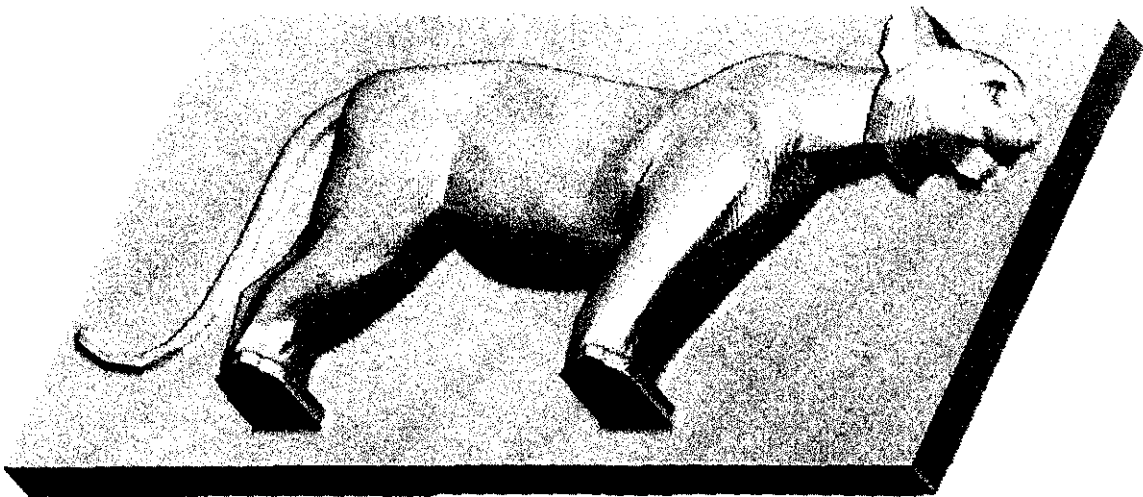
【図26】



【図27】



【図28】



【図29】

