

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3950976号

(P3950976)

(45) 発行日 平成19年8月1日(2007.8.1)

(24) 登録日 平成19年5月11日(2007.5.11)

(51) Int. Cl.

G06F 17/50 (2006.01)

F I

G06F 17/50 626C

請求項の数 12 (全 23 頁)

(21) 出願番号	特願2004-4690 (P2004-4690)	(73) 特許権者	301022471
(22) 出願日	平成16年1月9日(2004.1.9)		独立行政法人情報通信研究機構
(65) 公開番号	特開2005-196684 (P2005-196684A)		東京都小金井市貫井北町4-2-1
(43) 公開日	平成17年7月21日(2005.7.21)	(74) 代理人	100085338
審査請求日	平成16年1月9日(2004.1.9)		弁理士 赤澤 一博
前置審査		(74) 代理人	100118245
			弁理士 井上 敬子
		(72) 発明者	荒川 佳樹
			東京都小金井市貫井北町4-2-1 独立行政法人通信総合研究所内
		審査官	加舎 理紅子

最終頁に続く

(54) 【発明の名称】 3次元幾何データの無矛盾化方法及びそのシステム

(57) 【特許請求の範囲】

【請求項1】

3次元幾何データに対して演算処理を行い、前記演算処理過程で生成及び処理された幾何データを記憶する3次元幾何データ処理システムにおいて、処理される3次元幾何データにおける幾何矛盾を検出しかつ除去する方法であって、

3次元幾何図形の表面データである境界面データを、3角形面データまたは3角形の3つの頂点が同一直線上となるゼロ3角形データにより構成し処理する3角形/ゼロ3角形データ処理ステップと、

前記3角形/ゼロ3角形データ処理ステップで処理された3次元幾何データの数値を必要精度に応じて任意に切り捨てる数値データ切り捨て処理ステップと、

前記数値データ切り捨て処理ステップにより、発生する幾何矛盾を検出し除去する幾何無矛盾化処理ステップと、

を有し、

前記幾何無矛盾化処理ステップが、縮退3角形面を検出し除去する縮退3角形面の検出除去処理ステップ、重なり3角形面を検出し除去する面の重なり検出除去処理ステップ、および矛盾立体を検出し除去する矛盾立体の検出除去処理ステップを含むものであることを特徴とする3次元幾何データの無矛盾化方法。

【請求項2】

縮退3角形面の検出除去処理ステップが、2頂点が同一点となるいびつな3角形、3頂点が同一点となるいびつな3角形、および3頂点が同一直線上となるいびつな3角形を検

10

20

出し除去するものであることを特徴とする請求項 1 記載の 3 次元幾何データの無矛盾化方法。

【請求項 3】

面の重なり検出除去処理ステップが、組にした 3 角形面データの同一平面判定および面の向き判定の判定結果に基づき、3 角形面が重なっていると判定された 3 角形面に対してその重なりを除去するようにした処理であることを特徴とする請求項 1 または 2 記載の 3 次元幾何データの無矛盾化方法。

【請求項 4】

矛盾立体の検出除去処理ステップが、
同じ立体を構成する面同士が交差している立体を検出し除去するものであることを特徴とする請求項 1 乃至 3 いずれか記載の 3 次元幾何データの無矛盾化方法。 10

【請求項 5】

矛盾立体の検出除去処理ステップが、
自己干渉立体の分離処理を行うステップ、立体の最上位 3 角形面の探索処理を行うステップ、立体の最近傍上位立体の探索処理を行うステップ、最近傍上位立体順に立体データのソーティングを行うステップ、面の向きによる立体の消去処理を行うステップからなることを特徴とする請求項 1 乃至 4 いずれか記載の 3 次元幾何データの無矛盾化方法。

【請求項 6】

前記各ステップにおいて、
4 次元同次座標系処理と可変長ビットの演算を用いて無誤差演算を行うことを特徴とする請求項 1 乃至 5 いずれか記載の 3 次元幾何データの無矛盾化方法。 20

【請求項 7】

3 次元幾何データに対して演算処理を行い、前記演算処理過程で生成及び処理された幾何データを記憶する 3 次元幾何データ処理システムであって、

3 次元幾何図形の表面データである境界面データを、3 角形面データまたは 3 角形の 3 つの頂点が同一直線上となるゼロ 3 角形データにより構成し処理する 3 角形 / ゼロ 3 角形データ処理部と、

前記 3 角形 / ゼロ 3 角形データ処理部で処理された 3 次元幾何データの数値を必要精度に応じて任意に切り捨てる数値データ切り捨て処理部と、

前記数値データ切り捨て処理部により、発生する幾何矛盾を検出し除去する幾何無矛盾化処理部と、
を有し、 30

前記幾何無矛盾化処理部が、縮退 3 角形面を検出し除去する縮退 3 角形処理部、重なり 3 角形面を検出し除去する重なり 3 角形処理部、および矛盾立体を検出し除去する矛盾立体処理部を含むものであることを特徴とする 3 次元幾何データの無矛盾化システム。

【請求項 8】

縮退 3 角形処理部が、2 頂点が同一点となるいびつな 3 角形、3 頂点が同一点となるいびつな 3 角形、および 3 頂点が同一直線上となるいびつな 3 角形を検出し除去するものであることを特徴とする請求項 7 記載の 3 次元幾何データの無矛盾化システム。

【請求項 9】

重なり 3 角形処理部が、組にした 3 角形面データの同一平面判定および面の向き判定の判定結果に基づき、3 角形面が重なっていると判定された 3 角形面に対してその重なりを除去する処理をすることを特徴とする請求項 7 または 8 記載の 3 次元幾何データの無矛盾化システム。 40

【請求項 10】

矛盾立体処理部が、
同じ立体を構成する面同士が交差している立体を検出し除去する処理を行うことを特徴とする請求項 7 乃至 9 いずれか記載の 3 次元幾何データの無矛盾化システム。

【請求項 11】

矛盾立体処理部が、

自己干渉立体の分離処理と、立体の最上位3三角形面の探索処理と、立体の最近傍上位立体の探索処理と、最近傍上位立体順に立体データのソーティングを行う処理と、面の向きによる立体の消去処理とを行うように構成していることを特徴とする請求項7乃至10いずれか記載の3次元幾何データの無矛盾化システム。

【請求項12】

前記各処理部において、4次元同次座標系処理と可変長ビットの演算を用いて無誤差演算を行うことを特徴とする請求項7乃至11いずれか記載の3次元幾何データの無矛盾化システム。

【発明の詳細な説明】

【技術分野】

10

【0001】

本発明は、3次元幾何データの演算処理方法及びそのシステムに関する。

【背景技術】

【0002】

従来技術では、幾何演算は浮動小数点演算を用いて行うのが一般的である。浮動小数点演算では数値の丸め（切り捨て）が行われるために演算誤差が発生する。そして、このような浮動小数点演算が繰り返されると、誤差が蓄積していく。

【0003】

このような演算誤差のために、浮動小数点演算をベースとした幾何演算処理（図形処理）では、処理が破綻し処理系に暴走（バグ）が生じるという問題点があった。このような処理系の暴走へのこれまでの対処方法は、それぞれの破綻に対応した例外処理ルーチン（アドホック的）を作成することにより対応していた。そして、このような例外処理ルーチンはかなりの量となり、また次第に増加していく。すなわち、根本的に処理系の破綻を回避することが出来なかった。

20

【0004】

この幾何処理の破綻の問題を解決する1つの方法を提案した。この提案では、以下に示す3つの特徴を有し、これにより無誤差形状演算を実現している。そして、計算誤差をまったくなくすことにより、計算誤差から生じる幾何演算の破綻を回避している（例えば特許文献1参照）。

（1）超3三角形（ゼロ3三角形）幾何データ処理

30

（2）4次元同次座標系幾何処理

（3）可変長ビットの整数演算

【特許文献1】特許第3151710号

【発明の開示】

【発明が解決しようとする課題】

【0005】

通常の3三角形面データ処理およびゼロ3三角形を含む超3三角形データ処理は、多角形面データ処理と比較して、より単純なデータ構造及び処理アルゴリズムとなるので、その処理系の高信頼性を確保することが容易である。

【0006】

40

しかしながら、通常の3三角形面データ処理および超3三角形データ処理でも、誤差による破綻は免れない。この幾何演算の誤差による破綻の問題を根本的に解決するために、特許第3151710号では、ゼロ3三角形幾何データ処理および4次元同次座標系幾何処理等を用いて、数値（座標値）の桁数が、ある一定の上限値以上に増大しない効率的な無誤差形状演算方法を実現している。

【0007】

しかし、この特許では、1回限りの形状演算をある上限以下の桁数で効率よく行う方式を提供している。複数回の形状演算を実行すると、やはり桁数が増大していく。この無誤差演算に伴う桁数の増大は数理的に原理的なもので不可避である。

【0008】

50

このような桁数の増大を回避する現実的かつ実用的な方法は、必要精度に応じて数値（座標値）を切り捨てることである。しかしながら、幾何処理において、このような方法を取ると、幾何矛盾が発生する。

【 0 0 0 9 】

本発明は、このような課題に着目してなされたものであって、主たる目的は、上述のような幾何矛盾を検出しかつ除去する汎用的かつ効率的な方法およびその方法を利用したシステムを提供することにある。例えば、3次元幾何データの形状演算において、形状演算を何回も実行しても、演算に破綻が生じない、かつ処理を効率的、高速に実行することが可能な3次元幾何データの演算方法、すなわち幾何無矛盾化処理方法およびその方法を利用したシステムを提供することにある。

10

【課題を解決するための手段】

【 0 0 1 0 】

すなわち、本発明の3次元幾何データの無矛盾化方法は、3次元幾何データに対して演算処理を行い、これらの処理過程で生成及び処理された幾何データを記憶する3次元幾何データ処理システムにおいて、以下により構成される方法を有することを特徴とする。

(1) 3次元幾何図形の表面データである境界面データを、3三角形面データまたは3三角形の3つの頂点が同一直線上となるゼロ3三角形データにより構成し処理する。特に、ゼロ3三角形データ形式を用いた幾何処理により、不必要な数値桁数の増大および不必要なデータ量の増大を抑制する。

(2) 3次元幾何データを構成する頂点データの座標値データ等を必要精度に応じて切り捨

20

捨てを行い、数値桁数の増大に対処する。
(3) 前記座標値データ等の切り捨て処理により、発生する幾何矛盾を検出し除去する。この処理は、縮退3三角形面の検出除去処理、面の重なり検出除去処理および矛盾立体の検出除去処理から構成される。

(4) 上記のすべての処理を、4次元同次座標系幾何処理と可変長ビット演算により無誤差演算で行う。4次元同次座標系幾何処理を用いることにより、割り算を排除し、すべての演算を無誤差で行うことができる。

例えば、 $1 \div 3$ のように割り算があると循環小数（無限の桁数）となり、有限の桁数（ビット長）で表現することが出来なくなり、無誤算演算が出来なくなる。これに対して、割り算が必要でない4次元同次座標系幾何処理では、扱う数値が全て有限桁数の有理数であるとすれば、その演算結果も有限桁数の有理数となる。無限桁数の循環小数等になることはない。

30

【発明の効果】

【 0 0 1 1 】

以上に説明したように本発明の3次元幾何データの無矛盾化方法およびその方法を利用したシステムによれば、3次元幾何図形の表面データである境界面データを、3三角形面データまたは3三角形の3つの頂点が同一直線上となるゼロ3三角形データにより構成し処理し、特に、ゼロ3三角形データ形式を用いた幾何処理により、不必要な数値桁数の増大および不必要なデータ量の増大を抑制することができ、また、3次元幾何データを構成する頂点データの座標値データ等を必要精度に応じて切り捨てることを行い、数値桁数の増大に対処することができ、さらに、座標値データ等の切り捨て処理により、発生する幾何矛盾を検出し除去することができるので、効果的に、3次元幾何データにおける幾何矛盾を検出し除去することができる。すなわち、無矛盾な誤りのない3次元幾何データを生成することができる。

40

【発明を実施するための最良の形態】

【 0 0 1 2 】

以下、本発明の一実施形態を、図面を参照して説明するが、まず始めに、簡単に従来の3次元幾何処理、および、本発明で用いる4次元同時座標幾何演算について説明しておく。

【 0 0 1 3 】

50

従来の3次元幾何処理は、3次元のユークリッド座標系（X、Y、Z）を用いて行われてきた。そして、この座標系を用いた幾何演算では、割り算が通常発生する。本発明では、次元を1次元上げた4次元の同次座標系（X、Y、Z、w）を用いて幾何処理を行う。このような4次元処理を幾何演算に用いることにより、全ての演算は加減算とかけ算のみで済み、割り算は必要でなくなる。

【0014】

従って、本発明では、頂点データは、この4次元同次座標（X、Y、Z、w）を用いて表現される。また、本発明では、座標値X、Y、Z、wは、全て可変長ビットの整数表現となる。すなわち、図3（4）に示すように、数値は、そのビット長と、符号（+、-）と、整数表現された数値とから構成される。以下、本発明では、特に断らない限り、座標系は4次元同次座標系での実施である。

10

< 4次元同次座標幾何演算 >

以下、本実施形態で用いる4次元同次座標幾何演算に関して説明する。なお、特に説明なきもので、太文字のものはベクトル、行列を表すものとする。

（1）2点から線の生成

2点 $V_0 = (X_0, Y_0, Z_0, w_0)$ 及び $V_1 = (X_1, Y_1, Z_1, w_1)$ を通る直線 L_{01} は次式（数1）で与えられる。ただし、 V_0 、 V_1 、 L_{01} はベクトル、行列を表している。

【0015】

【数1】

$$L_{01} = [P_{01}, Q_{01}, R_{01}, S_{01}, T_{01}, U_{01}]$$

20

ここで、 P_{01} 等は、下式（数2）で与えられる。

【0016】

【数2】

$$P_{01} = \begin{vmatrix} X_0 & w_0 \\ X_1 & w_1 \end{vmatrix}, \quad Q_{01} = \begin{vmatrix} Y_0 & w_0 \\ Y_1 & w_1 \end{vmatrix}, \quad R_{01} = \begin{vmatrix} Z_0 & w_0 \\ Z_1 & w_1 \end{vmatrix},$$

$$S_{01} = \begin{vmatrix} Y_0 & Z_0 \\ Y_1 & Z_1 \end{vmatrix}, \quad T_{01} = \begin{vmatrix} Z_0 & X_0 \\ Z_1 & X_1 \end{vmatrix}, \quad U_{01} = \begin{vmatrix} X_0 & Y_0 \\ X_1 & Y_1 \end{vmatrix}$$

30

（2）3点から面の生成（平面係数の算出）

3点 $V_0 = (X_0, Y_0, Z_0, w_0)$ 、 $V_1 = (X_1, Y_1, Z_1, w_1)$ 、 $V_2 = (X_2, Y_2, Z_2, w_2)$ を通る面 F_{012} は、次式（数3）で与えられる。ただし、 V_0 、 V_1 、 V_2 、 F_{012} はベクトル、行列を表している。

【0017】

【数3】

$$F_{012} = [A_{012}, B_{012}, C_{012}, D_{012}]$$

ここで、 A_{012} 等は、下式（数4）で与えられる。

40

【0018】

【数4】

$$\mathbf{A}_{012} = \begin{vmatrix} Y_0 & Z_0 & w_0 \\ Y_1 & Z_1 & w_1 \\ Y_2 & Z_2 & w_2 \end{vmatrix}, \quad \mathbf{B}_{012} = \begin{vmatrix} Z_0 & X_0 & w_0 \\ Z_1 & X_1 & w_1 \\ Z_2 & X_2 & w_2 \end{vmatrix},$$

$$\mathbf{C}_{012} = \begin{vmatrix} X_0 & Y_0 & w_0 \\ X_1 & Y_1 & w_1 \\ X_2 & Y_2 & w_2 \end{vmatrix}, \quad \mathbf{D}_{012} = -\begin{vmatrix} X_0 & Y_0 & Z_0 \\ X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \end{vmatrix}$$

10

(3) 線と面の交点計算

2点 $V_a = (X_a, Y_a, Z_a, w_a)$ および $V_b = (X_b, Y_b, Z_b, w_b)$ を通る直線と、面 F_{012} との交点 V は、次式(数5)で与えられる。ただし、 V_a 、 V_b 、 F_{012} はベクトル、行列を表している。

【0019】

【数5】

$$\mathbf{V} = \mathbf{S}_{b012} \mathbf{V}_a - \mathbf{S}_{a012} \mathbf{V}_b$$

20

ここで、 S_{a012} 等は、下式(数6)で与えられる。

【0020】

【数6】

$$\begin{aligned} \mathbf{S}_{a012} &= \mathbf{V}_a \mathbf{F}_{012} \\ &= X_a \mathbf{A}_{012} + Y_a \mathbf{B}_{012} + Z_a \mathbf{C}_{012} + w_a \mathbf{D}_{012} \quad (\leq 0) \end{aligned}$$

$$\begin{aligned} \mathbf{S}_{b012} &= \mathbf{V}_b \mathbf{F}_{012} \\ &= X_b \mathbf{A}_{012} + Y_b \mathbf{B}_{012} + Z_b \mathbf{C}_{012} + w_b \mathbf{D}_{012} \quad (\geq 0) \end{aligned}$$

30

(4) 2点の一致判定

2点 $V_0 = (X_0, Y_0, Z_0, w_0)$ と $V_1 = (X_1, Y_1, Z_1, w_1)$ の一致判定は、前記2点を通る直線 L_{01} において以下(数7)となることである。ただし、 V_0 、 V_1 、 L_{01} はベクトル、行列を表している。

【0021】

【数7】

$$\mathbf{L}_{01} = [\mathbf{P}_{01}, \mathbf{Q}_{01}, \mathbf{R}_{01}, \mathbf{S}_{01}, \mathbf{T}_{01}, \mathbf{U}_{01}] = [0, 0, 0, 0, 0, 0]$$

(5) 2面の一致判定

40

下式(数8)で表される2つの平面 f_0 と f_1 との一致判定は、点と面の双対性により、式(数7)と同じ形式となる。

【0022】

【数8】

$$\begin{aligned} \mathbf{f}_0 &= [a_0, b_0, c_0, d_0]^T \\ \mathbf{f}_1 &= [a_1, b_1, c_1, d_1]^T \end{aligned}$$

(6) 点の面に対する位置判定

点 $V = (X, Y, Z, w)$ が、平面 F_{012} に対してどの位置にあるかの判定は次式(数

50

9) で与えられる。ただし、 V 、 F_{012} はベクトル、行列を表している。

【0023】

【数9】

$$s = \mathbf{V} \mathbf{F}_{012} = X \mathbf{A}_{012} + Y \mathbf{B}_{012} + Z \mathbf{C}_{012} + w \mathbf{D}_{012}$$

式(数9)において、 s の符号 $+$ 、 0 、 $-$ に対応して、点は、面の正側、面上、面の負側となる。

(7) 2線分の向き判定

2線分 L_{01} と L_{23} が同一直線である時には、両者の向きの判定は、次式(数10)で与えられる。ただし、 L_{01} 、 L_{23} はベクトル、行列を表している。

【0024】

【数10】

$$s = \mathbf{L}_{01} \mathbf{L}_{23}$$

上記の式(数10)において、両者の向きは、 s が正の時は同じ方向、負の時は逆向きとなる。

(8) 2面の向き判定

2面 F_{012} と F_{345} が同一平面である時には、両者の向きの判定は次式(数11)で与えられる。ただし、 F_{012} 、 F_{345} はベクトル、行列を表している。

【0025】

【数11】

$$s = \mathbf{F}_{012} \mathbf{F}_{345}$$

上記の(7)の場合と同様に、両者の向きは、 s が正の時は同じ方向、負の時は逆向きとなる。

<超3角形幾何データ>

本発明では、ゼロ3角形を用いた図形処理方式(超3角形幾何処理方式)を提案している。次に、本発明の特徴であるゼロ3角形幾何とそのデータ構造に関して説明する。

【0026】

従来の3次元形状表現(境界面表現)においては、図2(2)に示すように多角形面表現が用いられてきた。以下、多角形面を単に多角形と呼称する。図2において、(3)は(2)の多角形表現を通常の3角形面表現とした斜視図である。以下、3角形面を単に3角形と呼称する。

【0027】

これに対して、本発明の実施例で例示する超3角形表現は、(4)のようになる。すなわち、図2(4)において、頂点C、D、G、及びHは同一直線上にある。そして、3角形CDH、DGHは頂点が同一直線上となり面積がゼロとなる。これらの3角形をゼロ3角形と呼称している。ゼロ3角形は3角形が完全につぶれた「線形の3角形」とみなすことができる。

【0028】

また、3つの頂点が同一直線上にない通常の3角形を実3角形と呼称している。そして、これらの実3角形とゼロ3角形とを組み合わせた幾何表現形式を拡張された表現形式ということで超3角形幾何表現と呼称している。

【0029】

図2(4)と(3)の3角形の数を比較すると、超3角形表現では4個(ゼロ3角形は2個)、通常3角形表現では6個となる。ゼロ3角形を用いることにより、より少ない実3角形(通常3角形)で面分を表現することが可能となる。このような特徴により、超3角形処理では、3角形幾何の表現性とその処理性が飛躍的に増大する。

【0030】

本発明のデータ構造は、図3に示すように、(1)立体データ、(2)実3角形データ

10

20

30

40

50

、(3)ゼロ3角形データ、(4)頂点データ、そして(5)交線データから構成される。

【0031】

立体データは、実3角形データおよびゼロ3角形データの集合体である。従って、これらのデータへのポイントで構成される。ここで、ポイントとは、記憶装置(メモリ)上において、該当データが格納されている番地(アドレス)を示すデータである。

【0032】

実3角形データ及びゼロ3角形データは、データ構造上は全く同じ形式となり、3角形の3つの頂点データへのポイント(v_0 、 v_1 、 v_2)と、3つの隣接する実3角形データ或いはゼロ3角形データへのポイント(t_0 、 t_1 、 t_2)とから構成される。また、頂点データは、4次元同次座標形式(X 、 Y 、 Z 、 w)で格納される。座標値は、可変長ビットの整数表現となる。すなわち、ビット長、符号(+、-)および数値(ビット列)から構成される。

10

【0033】

交線データは、交線の始点及び終点となる頂点データへのポイント(v_0 、 v_1)と、交線を挟んで隣接する2つの3角形データへのポイント(t_0 、 t_1)とから構成される。

<システム構成>

以下、本発明の一実施形態を、図面を参照して説明する。

【0034】

本発明の3次元幾何データの無矛盾化システムたる3次元幾何データ処理システムSSは、図18に示すように、CPU101、メモリ102、外部記憶装置103、マウスやキーボードなどの入力手段104、ディスプレイなどの出力手段105、外部機器06と通信するための外部機器インタフェース106を主な構成要素とする。

20

【0035】

そして、前記外部記憶装置103やメモリ102に記憶する所定のプログラムを作動させることにより、前記CPU101や周辺機器を作動させ、該3次元幾何データ処理システムSSが、3次元幾何データ処理部01、幾何データ記憶部02、4次元同次座標系幾何処理部(プリユッカー座標演算器)03、可変長ビット整数演算器04、外部機器インタフェース部05等としての機能を発揮するようにしている。

【0036】

以下、各部を説明する。

30

【0037】

3次元幾何データ処理部01は、3角形/ゼロ3角形データ処理部07、数値データ切り捨て処理部08、幾何無矛盾化処理部09としての機能を発揮するものである。

【0038】

より具体的には、3角形/ゼロ3角形データ処理部07は、外部機器インタフェース106により入力された3次元形状データを通常の3角形データ、あるいは超3角形データに変換するものである。

【0039】

数値データ切り捨て処理部08は、頂点データ等の数値データを、必要精度に応じて切り捨てる処理を行うものである。

40

【0040】

幾何無矛盾化処理部09は、前記数値データ切り捨て処理部08において、数値データの切り捨てにより発生する幾何矛盾をなくす処理を行うものである。なお、本実施形態では、この幾何無矛盾化処理部09が、縮退3角形処理部10、重なり3角形処理部11、矛盾立体処理部12としての機能を発揮するように構成している。

【0041】

幾何データ記憶部02は、前記外部記憶装置103およびメモリ102の少なくともいずれか一方の所定領域に形成されるものであって、本実施形態では、さらに、立体データ記憶部13、実3角形データ記憶部14、ゼロ3角形データ記憶部15、頂点データ記憶部16、交線データ記憶部17からなるように構成している。

50

【 0 0 4 2 】

より具体的には、立体データ記憶部 1 3、実 3 角形データ記憶部 1 4、ゼロ 3 角形データ記憶部 1 5、頂点データ記憶部 1 6、交線データ記憶部 1 7 は、前記 3 次元幾何データ処理部 0 1 で生成された超 3 角形データを構成する、立体データ、実 3 角形データ、ゼロ 3 角形データ、頂点データおよび交線データを、それぞれ、記憶するものである。

【 0 0 4 3 】

4 次元同次座標系幾何処理部 0 3 は、全ての幾何演算を 4 次元同次座標系に基づいて行うものである。即ち、プリュッカー座標演算が行われる。

【 0 0 4 4 】

可変長ビット整数演算器 0 4 は、加算器、減算器および乗算器としての機能を有するものである。そして、加算、減算及び乗算が可変長ビットの整数演算を用いて、無誤差で行うようにしている。なお、この可変長ビット整数演算器 0 4 においては、除算器としての機能がないのが特徴である。

10

【 0 0 4 5 】

外部機器インターフェース部 0 5 は、前記 3 次元幾何データ処理部で処理された演算結果などを外部機器 0 6 に対して出力する機能と、前記外部機器 0 6 からのデータを受け付け前記 3 次元幾何データ処理部 0 1 に対して出力する機能とを有するものであって、前記外部機器インターフェース 1 0 6 等を利用して構成している。

【 0 0 4 6 】

次に、本実施形態の 3 次元幾何データ処理システム S S の動作についてフロー図などを用いて説明する。

20

【 0 0 4 7 】

特に、本実施形態の根幹をなす幾何無矛盾化処理部 0 9 における幾何無矛盾化処理について詳細に説明する。頂点データの低位桁の数値の切り捨てを行うと、次の 3 つの幾何矛盾が発生する可能性がある。

- (1) 縮退 3 角形 (いびつな 3 角形)
- (2) 面の重なり (3 角形の重なり)
- (3) 矛盾立体 (立体の自己干渉等)

ここで、このような幾何矛盾を検出し除去する処理を「無矛盾化処理」と呼ぶことにする。この無矛盾化処理の全体フローチャートは図 4 に示すようになる。

30

【 0 0 4 8 】

すなわち、[a] 縮退 3 角形の検出除去処理 (ステップ 4 0 0)、[b] 重なり 3 角形の検出除去処理 (ステップ 4 0 1)、[c] 矛盾立体の検出除去処理 (ステップ 4 0 2) の順で幾何無矛盾化処理が行われる。

【 0 0 4 9 】

なお、[c] 矛盾立体の検出除去処理はさらに、[c 1] 自己干渉立体の分離処理 (ステップ 4 0 3)、[c 2] 立体の最上位 3 角形の探索処理 (ステップ 4 0 4)、[c 3] 立体の最近傍上位立体の探索処理 (ステップ 4 0 5)、[c 4] 最近傍上位立体順に立体データのソーティング (ステップ 4 0 6)、[c 5] 面の向きによる立体の消去処理 (ステップ 4 0 7) の順で行われる。

40

【 0 0 5 0 】

以下、幾何無矛盾化処理部 0 9 における各処理 [a]、[b]、[c 1] ~ [c 5] について説明する。

【 0 0 5 1 】

[a] 縮退 3 角形の検出除去処理

頂点データの低位桁の数値の切り捨てを行うと、立体を構成する 3 角形面において、図 5 (1) ~ (3) に示すように、以下の 3 つの「いびつな」3 角形が発生する可能性がある。このようにいびつな 3 角形を「縮退 3 角形」と呼ぶ。

- (1) 2 頂点が同一点となる 3 角形
- (2) 3 頂点が同一点となる 3 角形

50

(3) 3頂点が同一直線上となる3角形(ゼロ3角形)

縮退3角形の検出とその除去処理について、図6のフローチャートを用いて説明する。まず、3角形データ(実3角形データおよびゼロ3角形データ)を幾何データ記憶部02から順次取り出し、その頂点データ v_0 、 v_1 、 v_2 を頂点データ記憶部16から順次読み込む(ステップ600)。その3つの頂点データ v_0 、 v_1 、 v_2 に関して同一点判定処理を行う。 $v_0 = v_1$ となる場合は(ステップ601)、辺 v_0v_1 を挟んで隣接する2つの3角形データ(図5(4)の例では t_0 と t_1)を消去する。これに伴い、図5(4)に示すように、消去される3角形 t_0 、 t_1 を取り囲む4つの周辺3角形 t_2 、 t_3 、 t_4 、 t_5 の隣接接続情報を変更する。図5(4)では、 t_0 と t_1 の消去に伴い、 t_2 と t_3 、および t_4 と t_5 が隣接するようになる。また、 v_1 を頂点に持つすべての3角形データの頂点 v_1 を v_0 とする。図5(4)では、 v_1 を頂点に持つ3角形面 t_3 、 t_5 、 t_6 の頂点 v_1 を v_0 に変更する。そして、頂点データ v_1 を消去する(ステップ602)。 $v_0 = v_2$ 、 $v_1 = v_2$ となる場合も同様の処理を行う(ステップ603~606)。以上の処理が、(1)2頂点が同一点となる場合、および(2)3頂点が同一点となる場合の縮退3角形データの検出除去処理である。

【0052】

次に、同一頂点を持たない3角形データに対して、その3つの頂点データ v_0 、 v_1 、 v_2 の同一直線判定処理を行う(ステップ607)。これらの3つの頂点点が同一直線上となる場合は、この3角形データをゼロ3角形データとして、ゼロ3角形データ記憶部15に保存する(ステップ608)。同一直線上とならない場合は、通常3角形であるので、実3角形データ記憶部14に書き込む(ステップ609)。以上の処理をすべての3角形データに関して行う。

【0053】

このように、ゼロ3角形幾何処理を行っているので、3角形の3つの頂点点が同一直線上となる場合は、このゼロ3角形を特段消去する必要も、例外処理する必要もない。

【0054】

[b] 重なり3角形の検出除去処理

幾何矛盾の1つとして、図8(8)に示すように、3角形面が重なり「面のしわ」となる場合がある。図8(8)では、形状の断面図を示しており、頂点A、B、C、D、Eは同一平面上にある。そして、3角形面 t_i と t_j が重なっている。このように面が重なる時は、図8(8)に矢印として示すように、面の向きは必ず逆となる。

【0055】

このような重なり3角形データの検出除去処理について、図7のフローチャートを用いて説明する。まず、同一平面上となり、かつ面の向きが逆となる2つの3角形データのペアを探索する。このために、2つの実3角形データ t_i 、 t_j を幾何データ記憶部02から順次読み出し、同一平面判定および面の向き判定を行う(ステップ701~703)。

【0056】

両者が同一平面かつ面の向きが逆となる場合が、重なり面処理の対象となる。同一平面とならない場合または向きが同じとなる場合は次の実3角形データのペアに処理を進める。

【0057】

両者が同一平面かつ面の向きが逆となる場合は、両者 t_i と t_j が重なるかどうかの判定を行う(ステップ704)。重ならない場合は次の実3角形データのペアに処理を進める。

【0058】

両者が重なる場合は、一方の3角形の頂点により他方の3角形を分割する(ステップ705)。この時のパターンは図8(1)~(4)に示すように4通りある。図8では3角形ABCが頂点Dにより分割される。そして、図8(1)と(2)のパターンの場合のみが分割処理の対象となる。本実施形態では、不必要な分割をなくすために「ゼロ3角形分割」を行う(図8(2)の場合)。ここでは、3角形ADCがゼロ3角形となる。

10

20

30

40

50

【 0 0 5 9 】

次に、この分割された3角形が他方の分割される前の3角形の内部にあるかどうかの判定処理を順次行う(ステップ706)。判定される3角形の3つの頂点すべてが、もう一方の3角形の内部(辺上を含む)となる場合は、前者の3角形は後者の内部となる(含まれる)。そして、内部となる場合は、この3角形データを消去する(ステップ708)。

【 0 0 6 0 】

また、判定される3角形の3つの頂点が、他方の内部および外部の両方となる場合(またがる場合)は、「3角形の位相変形」を行い(ステップ707)、判定される3角形が他方の内部あるいは外部のどちらかになるようにする。

【 0 0 6 1 】

図8(5)の例では、分割される前の3角形として、ABCとDEFが重なっている。これらが互いに他を分割しあうと、3角形ABCは、図8(6)のように分割される。ここで、3角形CDEとCFDは、3角形DEFに対して、内部となる部分と外部となる部分の両方がある。そこで、この2つの3角形が構成する4角形CFDEにおいて、その対角線となる辺CDをEFと付け替え、新たな3角形DEFとCFEを生成する(図8(7))。これが3角形の位相変形である。これを繰り返すことにより最終的に、3角形は他方の3角形の内部か外部かのどちらか一方となり、内部および外部の両方に「またがる」場合は存在しなくなる。

【 0 0 6 2 】

分割された3角形が、他方の分割される前の3角形の外部となる場合は、次の分割された3角形データに処理を進める。以上の処理を、すべての分割された3角形データに関して行う。

【 0 0 6 3 】

最後に、消去された3角形に隣接する3角形データを相互に接続する処理(接続情報の更新処理)を行う(ステップ710)。以上の処理を、同一平面上かつ面の向きが逆となる実3角形データのすべてのペアにおいて行う。これにより、すべての重なり3角形(面のしわ)を消去することができる。

【 0 0 6 4 】

[c] 矛盾立体の検出除去処理

[c 1] 自己干渉立体の分離処理

自己干渉立体とは、図10(1)、(2)に示すように、同じ立体を構成する面(3角形面)どうしが交差している立体である。自己干渉立体の分離処理では、図10(3)、(4)に示すように、この交差している面(3角形面)をその交線において分割し、1つの閉じた面(立体)とする処理である。

【 0 0 6 5 】

図10では、立体の断面図を示している。灰色領域が形状の内部である。図10(1)では、立体の一部で表裏が逆転している。矛盾立体が発生している。そこで、図10(2)に示すように、この部分を切り離す処理が分離処理となる。また、図10(3)では、立体の内部にある穴が外部にはみ出している。そこで、図10(4)に示すように、このはみ出し部分を分離する。

【 0 0 6 6 】

自己干渉立体の分離処理の流れを、図9のフローチャートを用いて説明する。2つの実3角形データ t_i 、 t_j を順次幾何データ記憶部02から取り出し、両者の交差判定処理を行う(ステップ901~903)。交差する場合は、両方の3角形 t_i 、 t_j の共通交線 s_k を求めて、この共通交線において両者を分割する(ステップ904)。

【 0 0 6 7 】

図10(5)の例では、交線ABにおいて、3角形 t_i と t_j が交差している。この時、共通交線 s_k はCDとなる。本実施形態では、重なり3角形検出除去処理において説明したように、不必要な分割をなくすために、「ゼロ3角形分割」を行う。ここでの分割パターンは、図10(6)に示すように2通りとなる。共通交線CDにおいて、その端点C

10

20

30

40

50

側は3三角形の辺上となり、ゼロ3三角形分割が行われ、3三角形ECGはゼロ3三角形となる。もう一方の端点Dは、3三角形面内となり、通常の分割が行われる。

【0068】

そして、この分割処理において、共通交線と一致する3三角形の辺が存在しない場合は、重なり3三角形検出除去処理において説明した位相変形を行い、共通交線と一致する3三角形の辺を生成する。以上の処理をすべての実3三角形データのペアに関して繰り返す。

【0069】

次に、この交線 s_k を挟んで隣接する3三角形の隣接情報を、図11、12に示すように変更する(ステップ907~909)。この処理をすべての交線データ s_k に関して行う。これにより、自己干渉立体を分離することができる。

10

【0070】

図11、12は、自己干渉立体の分離処理において求めた共通交線 s_k 近傍の面の交差パターンを示している。これらは立体の断面図であり、黒点は共通交線を示している。また、ハッチングがある側が立体の内部である。

【0071】

図11(1)では、面(3三角形面) t_a と t_b が完全に交差している。この場合は当然両者の面の接続関係を、共通交線において、切り替える。図11(2)の場合は、両者の面が形状の内側(内部)において接している場合である。この場合も、接続関係を切り替える。図11(3)の場合は、両者の面が形状の外側(外部)において接している場合である。この場合は、接続関係は切り替えずそのままとする。図11(4)は、両者の面が

20

【0072】

図12(1)、(2)は、面(3三角形面) t_a と t_b において、同一平面となる部分がある場合である。このような場合は、 t_a が t_b に対して内部となる場合のみ、接続関係を切り替える。従って、図12(1)の場合のみ、切り替える。図12(2)の場合は、 t_a が t_b に対して外部となるので、接続関係は切り替えない。ここでは、内部となる場合を切り替えるとしたが、この基準をまったく逆にして、外部となる場合を切り替えるとしても何ら問題はない。

【0073】

また、図12(1)、(2)では、同一平面となる部分の面の向きが同じとなる場合を

30

【0074】

[c2] 最上位3三角形の探索処理

最上位3三角形とは、1つの立体(1つの閉じた境界面)において、その立体を構成する3三角形の中で、Z座標値が最大となる頂点を持ち、かつ上面が存在しない3三角形である。4次元同次座標系において、頂点 $V_a = (X_a, Y_a, Z_a, w_a)$ のZ座標値が $V_b = (X_b, Y_b, Z_b, w_b)$ より大きいとは以下の式(数12)が成り立つことである。

【0075】

40

【数12】

$$Z_a w_b > Z_b w_a \quad (w_a, w_b \geq 0)$$

また、3三角形 F_{012} が3三角形 F_{345} に対して(Z座標軸に関して)上面となるとは、この2つの3三角形 F_{012} と F_{345} がZ座標軸方向からみて重なり、かつ3三角形 F_{012} の3つの頂点 $V_0 = (X_0, Y_0, Z_0, w_0)$ 、 $V_1 = (X_1, Y_1, Z_1, w_1)$ 、 $V_2 = (X_2, Y_2, Z_2, w_2)$ において、以下の式(数13)が成り立つことである。ただし、3三角形 F_{012} と F_{345} は交差しないとする。

【0076】

【数 1 3】

$$(0 < s_0) \vee (0 < s_1) \vee (0 < s_2) \quad (0 < C_{345})$$

$$(s_0 < 0) \vee (s_1 < 0) \vee (s_2 < 0) \quad (C_{345} < 0)$$

ここで、 s_0 等は、下式（数 1 4）で与えられる。

【0 0 7 7】

【数 1 4】

$$s_0 = \mathbf{V}_0 \mathbf{F}_{345} = X_0 \mathbf{A}_{345} + Y_0 \mathbf{B}_{345} + Z_0 \mathbf{C}_{345} + w_0 \mathbf{D}_{345}$$

$$s_1 = \mathbf{V}_1 \mathbf{F}_{345} = X_1 \mathbf{A}_{345} + Y_1 \mathbf{B}_{345} + Z_1 \mathbf{C}_{345} + w_1 \mathbf{D}_{345}$$

$$s_2 = \mathbf{V}_2 \mathbf{F}_{345} = X_2 \mathbf{A}_{345} + Y_2 \mathbf{B}_{345} + Z_2 \mathbf{C}_{345} + w_2 \mathbf{D}_{345}$$

10

さらに、2つの3三角形が、Z座標軸方向からみて、重なるかどうかの判定処理に関して説明する。4次元同次座標系（X、Y、Z、w）において、Z座標を無視した3次元同次座標系（X、Y、w）を想定する。この座標系における3三角形の重なり判定処理に帰着することができる。この処理は、3三角形の頂点と、方向を持った辺との位置関係判定処理（領域判定処理）に帰着する。

20

【0 0 7 8】

最上位3三角形を求める処理を、図13のフローチャートを用いて説明する。まず最上位頂点Vmaxに適当な初期値（例えば存在する頂点データ）を代入する（ステップ1300）。次に、処理対象となる1つの立体を構成する実3三角形データを幾何データ記憶部02から順次読み出し、さらにその実3三角形を構成する3つの頂点データv0i、v1i、v2iを幾何データ記憶部02から読み出し比較し、Z座標値が最大となる頂点データVを求める（ステップ1301）。

【0 0 7 9】

次に、この頂点データVと最上位頂点データVmaxのZ座標値を比較する（ステップ1302）。Vが大きい場合は、Vmax = Vとする（ステップ1303）。以上の処理を、処理対象立体のすべての実3三角形データに関して行う。以上の処理により、処理対象立体において、Z座標値が最大となる最上位頂点Vmaxが求まる。

30

【0 0 8 0】

次に、最上位3三角形Tmaxと基準3三角形Tbaseに適当な初期値を代入する。例えば処理対象立体において、最上位頂点Vmaxを頂点に持つ適当な実3三角形データを両者に代入する（ステップ1305）。そして、処理対象立体において、この最上位頂点Vmaxを頂点に持つ実3三角形データtiを順次幾何データ記憶部02から取り出す（ステップ1306、1307）。

【0 0 8 1】

次に、この3三角形tiとTbaseがZ座標軸方向からみて重なるかどうかの判定を行う（ステップ1308）。重なる場合は、この3三角形tiとTmaxがZ座標軸方向からみて重なるかどうかの判定を行う（ステップ1309）。重なる場合は、tiとTmaxの位置関係をチェックする（ステップ1310）。tiがTmaxの上面となる場合（Tmax < ti）はTmax = tiとする（ステップ1311）。以上の処理をすべての実3三角形データtiに関して行う。これにより、最上位3三角形Tmaxが求まる。

40

【0 0 8 2】

[c 3] 最近傍上位立体を求める処理

ある立体biの最近傍上位立体bnearとは、以下の条件を満たす立体のことである（図15参照）。すなわち、最近傍上位立体とは、ある立体（1つの閉じた境界面）に、Z座標軸正方向に最も近い立体のことである。

50

(1) 立体 b_i の最上位頂点 $V_{max} = (X_{max}, Y_{max}, Z_{max}, W_{max})$ に Z 座標軸 と平行に正方向に立てた半直線と交差あるいは接する立体。

(2) この交点あるいは接点を $V_{near} = (X_{near}, Y_{near}, Z_{near}, W_{near})$ とすると、 V_{near} は頂点 $V_{max} = (X_{max}, Y_{max}, Z_{max}, W_{max})$ に最近傍となる立体。ここで、 V_{near} を最近傍上位点と呼ぶことにする。

(3) Z 座標軸方向から見て、立体 b_i の最上位 3 角形 T_{max} と重なる上面 T_{near} を持つ立体。

(4) 上記上面 T_{near} が複数個ある場合は、その中で最も下面となる面を持つ立体。ここで、下面とは、3 角形 t_2 が 3 角形 t_3 の上面となる場合、逆に t_3 は t_2 に対して下面となる (図 15 参照)。また、この最も下面となる面 T_{near} を最近傍上位 3 角形と呼ぶことにする。

10

【0083】

図 15 の例では、立体の断面図を示している。ここでは、立体 b_i の最上位頂点は $V_{max} = v_0$ 、最上位 3 角形は $T_{max} = t_1$ である。立体 $b(j+1)$ は、半直線 L と接し、かつ最近傍頂点 v_1 を持つが、 T_{max} と重なる面が存在しない。そこで、立体 $b(j+1)$ は、立体 b_i の最近傍上位立体ではない。

【0084】

一方、立体 b_j は、半直線 L と接し、かつ (立体 $b(j+1)$ は対象外となるので) 最近傍上位点 (頂点) v_2 を持ち、かつ最上位 3 角形 T_{max} と重なる面 t_3 が存在する。そこで、立体 b_j が、立体 b_i の最近傍上位立体となる。ここで、最上位 3 角形 T_{max} と重なる面は t_2 と t_3 の 2 つが存在するが、両者を比較すると、 t_3 が t_2 の下面となるので、 t_3 が最近傍上位 3 角形 T_{near} となる。

20

【0085】

最近傍上位立体を求める処理に関して、図 14 のフローチャートを用いて説明する。ここでは、立体 b_i の最近傍上位立体 b_{near} を求める。まず、立体 b_i の最上位頂点 V_{max} を通る Z 座標軸と平行な直線 L を求める。そして、最近傍上位頂点 V_{near} 等を初期設定する (ステップ 1400)。例えば、 V_{near} には、Z 座標値に適当な大きな値、X、Y、w 座標値に V_{max} と同じ値を代入しておけばよい。また、NULL は「値なし」(該当データなし) を意味する。

【0086】

次に、立体 b_i とは異なる他の立体 b_j の実 3 角形データ t_{jk} を幾何データ記憶部 02 から順次読み込む (ステップ 1402)。そして、この t_{jk} と直線 L との交点 V_s が存在するかどうかを判定し、存在する場合は V_s を求める (ステップ 1403)。

30

【0087】

交点 V_s が存在する場合は、 V_{max} と V_s の Z 座標値を比較する (ステップ 1404)。 V_s の Z 座標値が V_{max} の Z 座標値より小さい場合 ($V_s < V_{max}$) は、次の実 3 角形データに処理を移す。逆に、 $V_{max} \leq V_s$ となる場合は、この 3 角形 t_{jk} と T_{max} が Z 座標軸方向からみて重なるかどうかの判定 (ステップ 1405)、および頂点データを比較して、 t_{jk} が T_{max} の上面となるかどうか ($T_{max} \leq t_{jk}$) の判定処理を行う (ステップ 1406)。

40

【0088】

t_{jk} と T_{max} が Z 座標軸方向からみて重なり、かつ t_{jk} が T_{max} の上面となる場合 ($T_{max} \leq t_{jk}$) は、 V_s と V_{near} を比較する (ステップ 1407)。 $V_{near} < V_s$ となる場合は、次の実 3 角形データに処理を移す。 $V_s < V_{near}$ となる場合はステップ 1410 に処理を進める。

【0089】

$V_s = V_{near}$ となる場合は、3 角形 t_{jk} と T_{near} が Z 座標軸方向からみて重なるかどうかの判定 (ステップ 1408)、および頂点データを比較して、 t_{jk} が T_{near} の下面となるかどうか ($t_{jk} \leq T_{near}$) の判定処理を行う (ステップ 1409)。

50

【0090】

$t_{j k}$ と T_{near} がZ座標軸方向からみて重なり、かつ $t_{j k}$ が T_{near} の下面となる場合($t_{j k} \leq T_{near}$)は、 $V_{near} = V_s$ 、 $T_{near} = t_{j k}$ 、 $b_{near} = b_j$ とする(ステップ1410)。

【0091】

以上の処理を、立体 b_i 以外のすべての実3角形データに関して行う。以上の処理により、立体 b_i の最近傍上位立体 b_{near} 、最近傍上位頂点 V_{near} 、最近傍上位3角形 T_{near} が求まる。

【0092】

[c4] 最近傍上位立体順に立体データのソーティング

10

最近傍上位立体順に立体データ b_i をソーティング処理するとは、立体データ(幾何データの集合体(実3角形データまたはゼロ3角形データの集合体)) b_i と b_j ($i < j$)において、 b_i の最近傍上位立体 b_{near} が b_j となる場合は、両者のデータの順番を入れ替えることである。すなわち、ここでは b_i が b_j よりもデータの先頭にある($i < j$)ので、 b_i と b_j のデータ順を入れ替え、 b_j が b_i よりもデータの先頭となるようにする。

【0093】

この処理により、ある立体 b_i の最近傍上位立体データ b_{near} は、必ず立体データ b_i よりも上位となる(先頭となる)。図15の例では、立体データの順番は上位から、 $b(j+2)$ 、 b_j 、 $b(j+1)$ 、 b_i の順番となる。

20

【0094】

[c5] 面の向きによる立体の消去処理

面の向きによる立体の消去処理とは、ある立体の面の表裏が最近傍上位立体との関係で矛盾する場合は、この矛盾する立体データの消去を行う処理である。

【0095】

図17を用いて、面の向きと矛盾立体との関係を説明する。図17は立体の断面図を表している。また、ハッチングのある側が立体(形状)の内部、ない側が立体の外部を表現している。言い換えれば、矢印の向きが立体の内部を表している。

【0096】

図17(1)および(2)の場合は、立体Aにおいて、最近傍上位立体がない場合である。このような場合は、立体Aの最上位面(最上位3角形)のZ軸方向の向きを用いて判定を行う。図17では、立体Aにおける矢印はすべて、立体Aの最上位3角形(面)のZ軸方向の向きを表している。

30

【0097】

図17(1)では、立体Aの最上位3角形のZ軸方向の向きはマイナスとなっている。このような場合は、立体Aは矛盾立体ではないので、消去されない。一方、図17(2)では、立体Aの最上位3角形のZ軸方向の向きはプラスとなっている。このような場合は、立体Aは矛盾立体となる(空間全体に立体の内部が広がっている)ので、消去される。

【0098】

図17(3)~(6)の場合は、立体Aの最近傍上位立体が存在する場合である。ここでは、立体Bが立体Aの最近傍上位立体となる。また、立体Bにおける矢印はすべて、立体Aに対する最近傍上位3角形(面)のZ軸方向の向きを表している。

40

【0099】

図17(3)~(6)の場合は、立体Aの最上位3角形と、立体Aに対する(立体Bに属する)最近傍上位3角形のZ軸方向の向きの組み合わせをチェックする。図17(3)と(6)に示すように、両者のZ軸方向の向きが逆となる場合は、立体Aは立体Bに対して「無矛盾立体」となる。そこで、このような場合は、立体Aは消去されない。

【0100】

一方、図17(4)と(5)の場合は、両者のZ軸方向の向きが同じとなる。このような場合は、立体Aは立体Bに対して「矛盾立体」となる。そこで、このような場合は、立

50

体 A は消去される。

【 0 1 0 1 】

次に、面の向きによる立体の消去処理に関して、図 1 6 のフローチャートを用いて説明する。ここでは、立体データ b_i の幾何矛盾チェックを、データ順に（データの先頭から）行う。まず、初期処理として、すべての立体データの消去フラグをすべて OFF とする（ステップ 1 6 0 0）。ここで、OFF は消去しない（無矛盾立体である）を意味する。そして、立体 $b = b_i$ とする（ステップ 1 6 0 1）。次に、立体 b の最近傍上位立体 b_{near} が存在するかをチェックする（ステップ 1 6 0 2）。

【 0 1 0 2 】

存在しない場合は、立体 b_i の最上位 3 角形 T_{max} の Z 座標軸方向の向きをチェックする（ステップ 1 6 0 3）。正の場合は幾何矛盾となるので、立体データ b_i の消去フラグを ON（消去を意味する）とする（ステップ 1 6 0 7）。負の場合は幾何矛盾とならないので、立体データ b_i の消去フラグを OFF のままとする。ここでの処理は、図 1 7（1）と（2）の場合に対応している。

【 0 1 0 3 】

立体 b の最近傍上位立体 b_{near} が存在する場合は、 b_{near} の消去フラグが ON か OFF かのチェックを行う（ステップ 1 6 0 4）。消去フラグが ON の場合は、この最近傍上位立体 b_{near} は矛盾立体であり消去されるので、 $b = b_{near}$ とし（ステップ 1 6 0 5）、さらに上位の消去フラグが OFF となっている無矛盾立体である最近傍上位立体を探す。

【 0 1 0 4 】

消去フラグが OFF となっている最近傍上位立体 b_{near} が存在する場合は、この b_{near} の最近傍上位 3 角形 T_{near} と、立体 b_i の最上位 3 角形 T_{max} の Z 座標軸方向の向きの組み合わせをチェックする（ステップ 1 6 0 6）。 T_{near} と T_{max} の Z 座標軸方向の面の向きの組み合わせが同じとなる場合は（図 1 7（4）と（5）の場合）、立体 b_i は矛盾立体となるので、立体 b_i の消去フラグを ON とする（ステップ 1 6 0 7）。面の向きの組み合わせが逆となる場合は（図 1 7（3）と（6）の場合）、立体 b_i は矛盾しないので、消去フラグは OFF のままとする。そして、次の立体に処理を移す。以上の処理を、すべての立体データに関して行うことにより、すべての立体の幾何矛盾チェックが完了し、その結果として、消去フラグの値（ON または OFF）が確定する。

【 0 1 0 5 】

最後に、消去フラグが ON となっているすべての立体データを幾何データ記憶部 0 2 から消去する（ステップ 1 6 0 9）。これにより、すべての矛盾立体が消去され、無矛盾立体だけが保存される。

【 0 1 0 6 】

以上に詳述したように、本実施形態の 3 次元幾何データ処理システム S S によれば、3 次元幾何図形の表面データである境界面データを、3 角形面データまたは 3 角形の 3 つの頂点が同一直線上となるゼロ 3 角形データにより構成し処理する 3 角形 / ゼロ 3 角形データ処理部 0 7 と、前記 3 次元幾何データの数値を必要精度に応じて任意に切り捨てる数値データ切り捨て処理部 0 8 と、前記数値データ切り捨て処理部 0 8 により、発生する幾何矛盾を検出し除去する幾何無矛盾化処理部 0 9 とを具備しているので、3 次元幾何図形の表面データである境界面データを、3 角形面データまたは 3 角形の 3 つの頂点が同一直線上となるゼロ 3 角形データにより構成し処理し、特に、ゼロ 3 角形データ形式を用いた幾何処理により、不必要な桁数の増大および不必要なデータ量の増大を抑制することができ、また、3 次元幾何データを構成する頂点データの座標値データ等を必要精度に応じて切り捨てを行い、数値桁数の増大に対処することができ、さらに、座標値データ等の切り捨て処理により、発生する幾何矛盾を検出し除去することができるので、効果的に、3 次元幾何データにおける幾何矛盾を検出し除去することができる。すなわち、無矛盾な誤りのない 3 次元幾何データを生成することができる。

10

20

30

40

50

【 0 1 0 7 】

また、前記幾何無矛盾化処理部 0 9 が、縮退 3 角形を検出し除去する縮退 3 角形処理部 1 0、重なり 3 角形を検出し除去する重なり 3 角形処理部 1 1、および自己干渉立体等を検出し除去する矛盾立体処理部 1 2 を含むようにしているので、幾何学的な矛盾にすべて対応することができる。

【 0 1 0 8 】

特に、縮退 3 角形処理部 1 0 によって、2 頂点が同一点となるいびつな 3 角形、3 頂点が同一点となるいびつな 3 角形、および 3 頂点が同一直線上となるいびつな 3 角形を、検出し除去することが確実にできる。

【 0 1 0 9 】

また、重なり 3 角形処理部 1 1 によって、3 角形面が重なっていると判定された 3 角形に対してその重なりを除去することを確実にできる。

【 0 1 1 0 】

また、矛盾立体処理部 1 2 が、自己干渉立体の分離処理と、立体の最上位 3 角形の探索処理と、立体の最近傍上位立体の探索処理と、最近傍上位立体順に立体データのソーティングを行う処理と、面の向きによる立体の消去処理とを行うように構成しているので、立体に矛盾のあるものを、確実に除去することができる。

【 0 1 1 1 】

加えて、各処理部において、4 次元同次座標系処理と可変長ビットの演算を用いて無誤差演算を行うようにしているので、演算を何回実行してもその演算に破綻が生じず、システムを安定して作動させることができる。

【 0 1 1 2 】

なお、本発明は、以上に詳述した実施形態に限られるものではない。

【 0 1 1 3 】

例えば、本実施形態では、3 次元幾何データ処理システム S S を、スタンドアローン型で動作させているが、インターネット等の通信回線網を介して接続される端末装置およびサーバ装置からなるものとしてもよい。

【 0 1 1 4 】

その他、各部の具体的構成についても上記実施形態に限られるものではなく、本発明の趣旨を逸脱しない範囲で種々変形が可能である。

【 図面の簡単な説明 】

【 0 1 1 5 】

【 図 1 】本発明の一実施形態における 3 次元幾何データの無矛盾化処理システムの機能構成図。

【 図 2 】同実施形態における従来と本発明の形状表現法の違いを示す図。

【 図 3 】同実施形態におけるデータ構造を示す図。

【 図 4 】同実施形態における幾何無矛盾化処理の全体フローを示す図。

【 図 5 】同実施形態における縮退 3 角形およびその除去処理に関する説明図。

【 図 6 】同実施形態における縮退 3 角形の検出除去処理のフローチャート。

【 図 7 】同実施形態における重なり 3 角形の検出除去処理のフローチャート。

【 図 8 】同実施形態における重なり 3 角形の検出除去処理に関する説明図。

【 図 9 】同実施形態における自己干渉立体の分離処理のフローチャート。

【 図 1 0 】同実施形態における自己干渉立体の分離処理に関する説明図。

【 図 1 1 】同実施形態における自己干渉立体の面の交差パターンに関する説明図。

【 図 1 2 】同実施形態における自己干渉立体の面の交差パターンに関する説明図。

【 図 1 3 】同実施形態における立体の最上位 3 角形探索処理のフローチャート。

【 図 1 4 】同実施形態における最近傍上位立体探索処理のフローチャート。

【 図 1 5 】同実施形態における最近傍上位立体に関する説明図。

【 図 1 6 】同実施形態における幾何矛盾立体の消去処理のフローチャート。

【 図 1 7 】同実施形態における幾何矛盾立体に関する説明図。

10

20

30

40

50

【図18】同実施形態における3次元幾何データの無矛盾化処理システムの機器構成図。

【符号の説明】

【0116】

SS・・・3次元幾何データの無矛盾化システム(3次元幾何データ処理システム)

07・・・3角形/ゼロ3角形データ処理部

08・・・数値データ切り捨て処理部

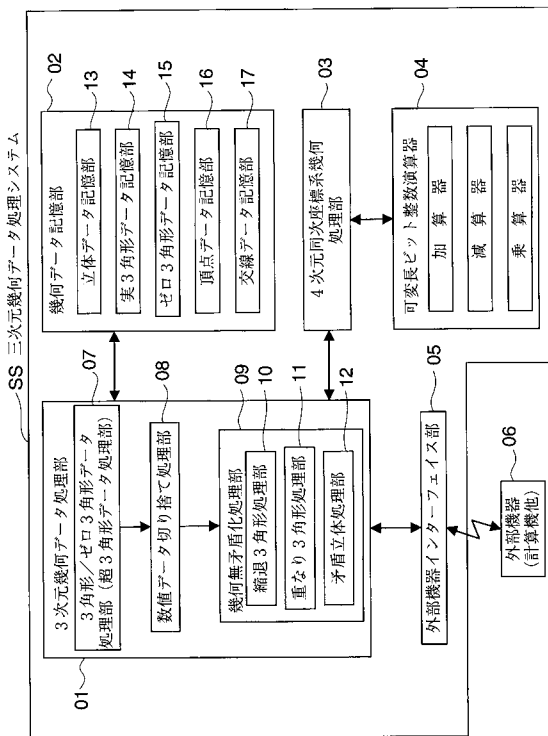
09・・・幾何無矛盾化処理部

10・・・縮退3角形処理部

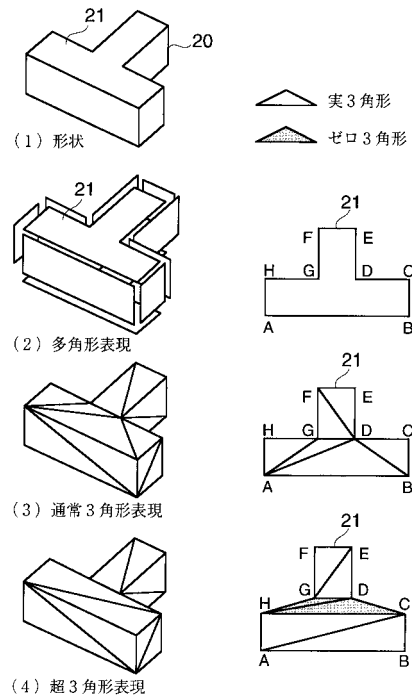
11・・・重なり3角形処理部

12・・・矛盾立体処理部

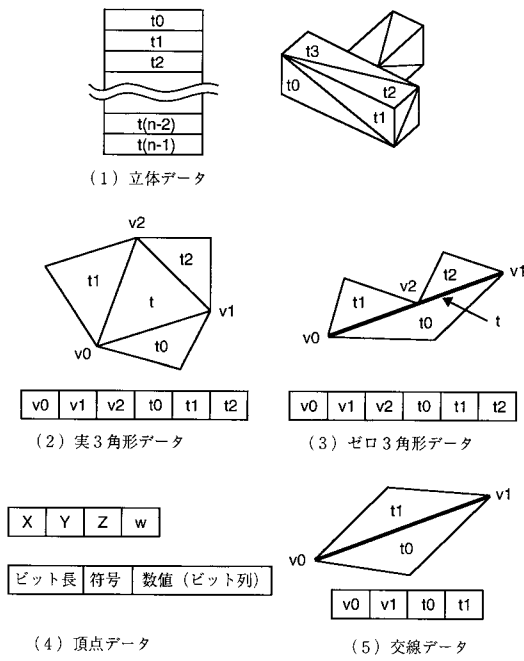
【図1】



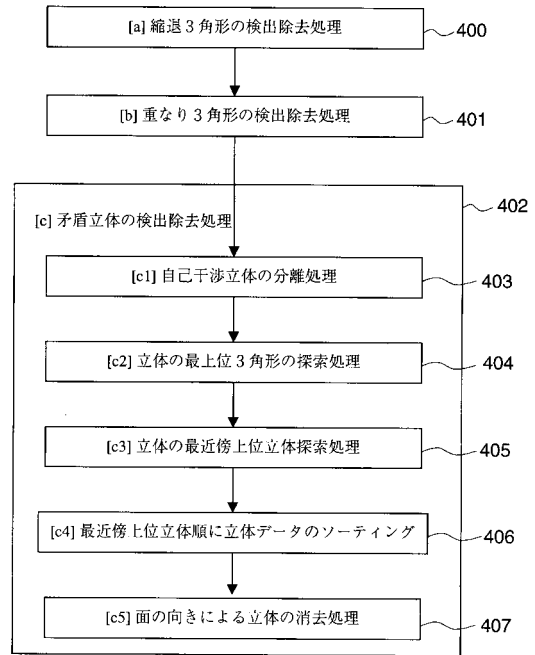
【図2】



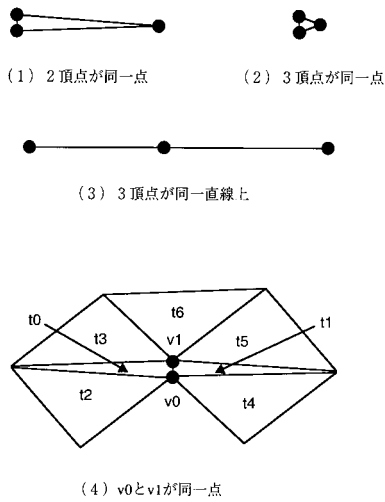
【 図 3 】



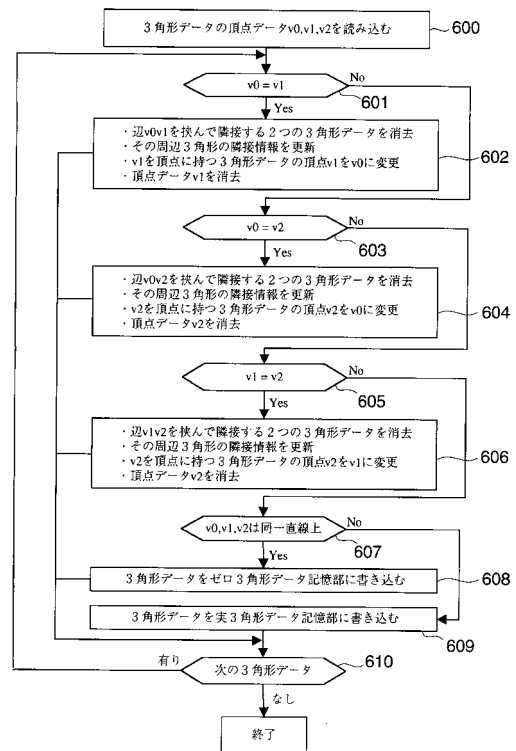
【 図 4 】



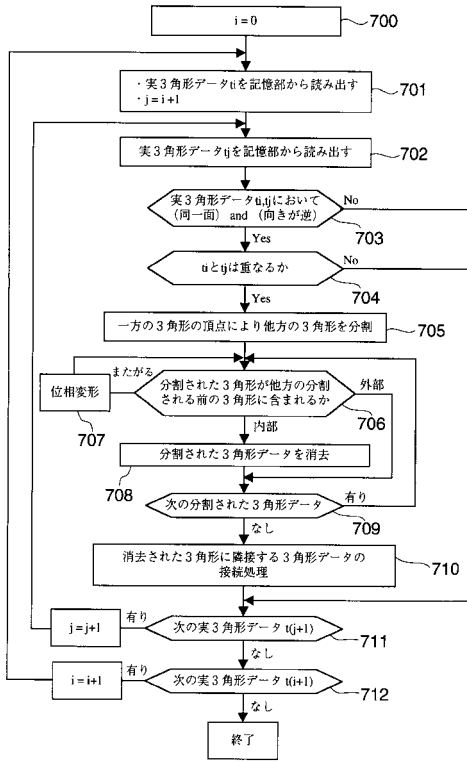
【 図 5 】



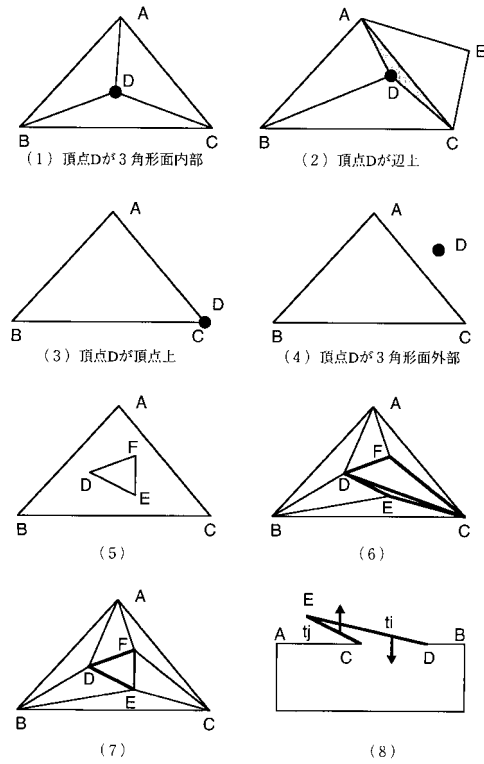
【 図 6 】



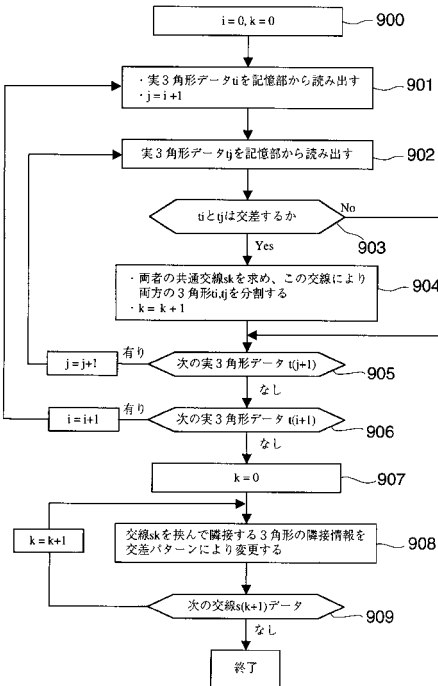
【 図 7 】



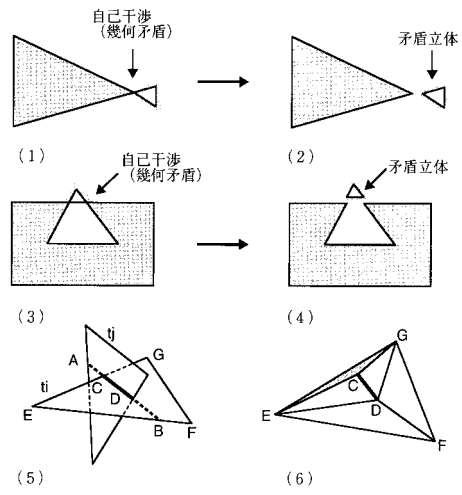
【 図 8 】



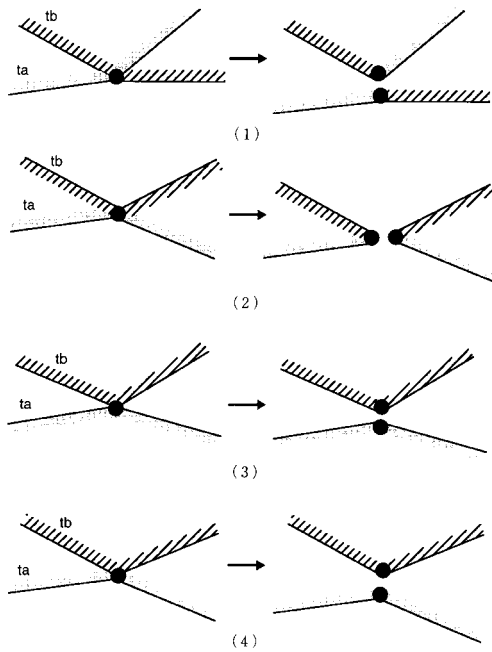
【 図 9 】



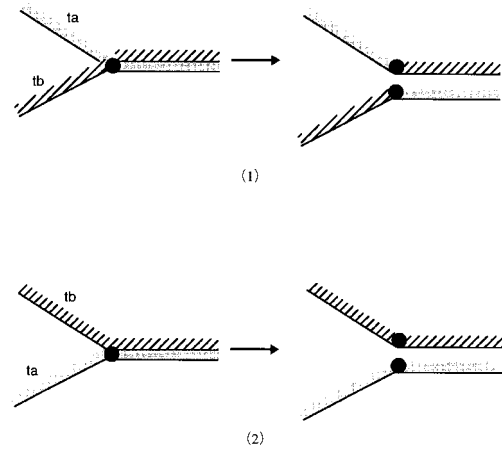
【 図 10 】



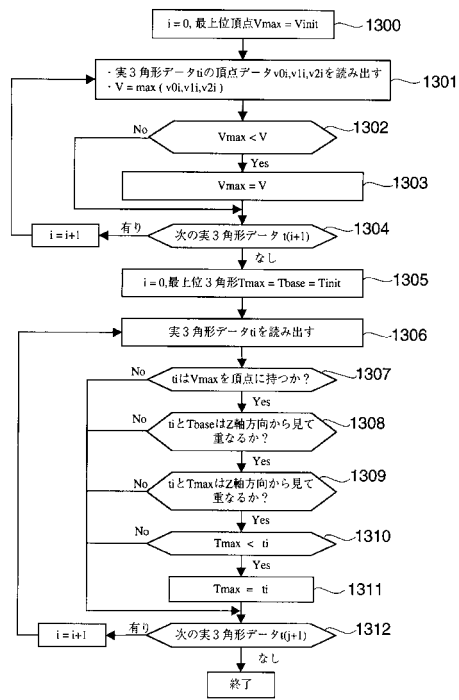
【 図 1 1 】



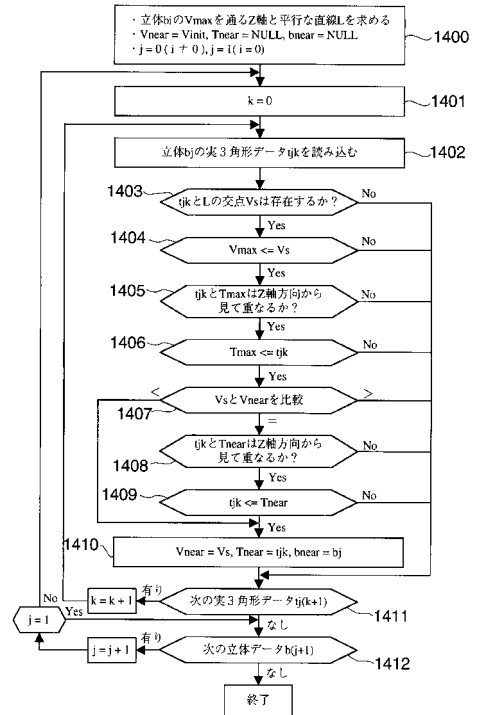
【 図 1 2 】



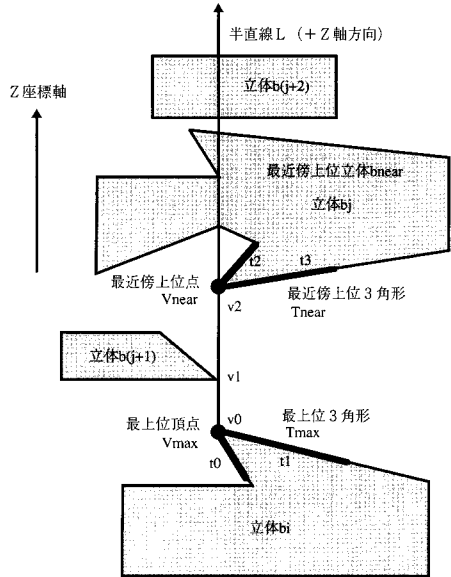
【 図 1 3 】



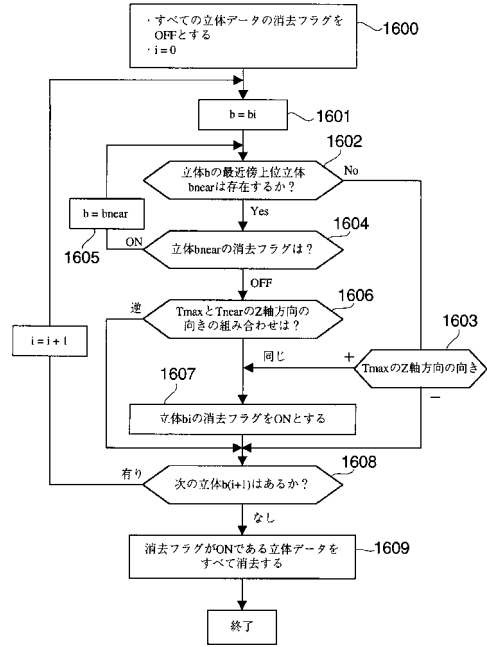
【 図 1 4 】



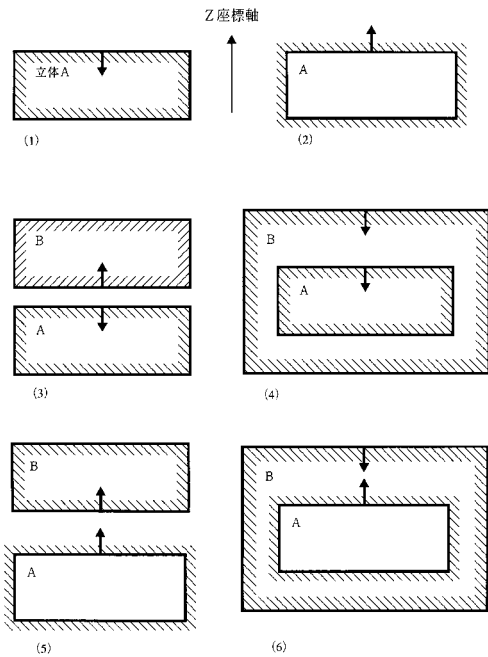
【 図 15 】



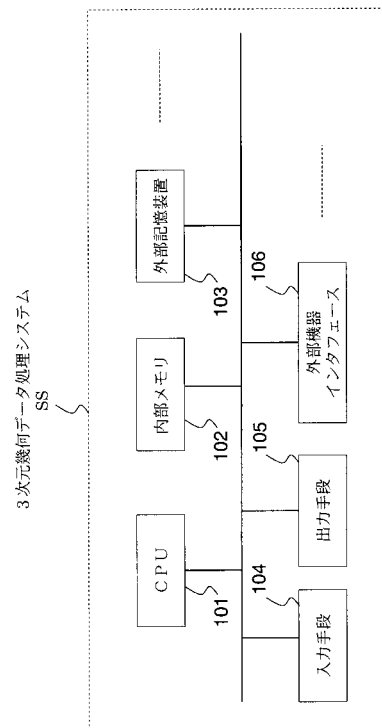
【 図 16 】



【 図 17 】



【 図 18 】



フロントページの続き

(56)参考文献 特開平01-286081(JP,A)

特開平06-124324(JP,A)

荒川佳樹 他, 超3角形BRepにおける無誤差完全4次元処理を用いた形状演算アルゴリズム
 , 情報処理学会論文誌, 1999年 9月15日, 第40巻, 第9号, p. 3471-3482

(58)調査した分野(Int.Cl., DB名)

G06F 17/50

G06T 11/20

G06T 17/10