無誤差 4 次元超 3 角形による幾何無矛盾化法 ——無誤差・無矛盾幾何コンピューティングを目指して

荒 川 佳 樹[†] 山口 富士夫[††]

これまでに提案した**無誤差 4 次元超 3 角形幾何**をベースとした集合演算法は，幾何処理における複雑性と不安定性の問題という二大テーマに対する 1 つの解決策を提示した．しかし，この方法には，数値桁数が際限なく増大するという本質的な問題点があった．そこで，本論文では，要求精度に応じて数値桁数を切り捨てることとし，これによって生じる幾何矛盾を取り除く方法である**幾何無矛盾化アルゴリズム**を提案する．この技術は，無誤差・無矛盾幾何コンピューティングの基盤技術となる可能性を秘めている．

Geometric Normalization Based on 4D Extended Triangulation with Exact Arithmetic — Toward Exact and Robust Geometric Computing

Yoshiki Arakawa[†] and Fujio Yamaguchi[††]

We have proposed a set operation algorithm based on **4D extended triangulation with exact arithmetic**. This method provides a solution to the two major problems of complexity and instability involved in geometric computing. However, it has a fatal flaw in that the number of figures increases infinitely. In this paper, we propose a way to round off numerical data to the required precision. Unfortunately, this approach results in geometric inconsistency. Therefore, we also propose **a geometric normalization algorithm** based on 4D extended triangulation with exact arithmetic. This technology has great potential, promising to become the basis of exact and robust geometric computing.

1. はじめに

幾何モデリングにおいて,「複雑性」と「不安定性」という重要な課題が残されたままである．しかし，その中核機能である集合演算のアルゴリズム（処理系）が複雑かつ大規模になるという「集合演算の複雑性の問題」に関しては，筆者らは 3 次元ユークリッド座標系超 3 角形幾何モデリング技術を確立することにより，1 つの解決策を提示しその有効性を実証した[1)〜3)]．

さらに，数値計算にともなう誤差に起因する「集合演算の不安定性の問題」に関しても，無誤差 4 次元超 3 角形幾何モデリングをこれまでに提案し，この不安定性の問題を根本から解決した[4)]．しかし，この方法は 1 回の集合演算において有効であり，複数回の集合演算を繰り返すと，数値桁数が際限なく増大するという根元的な問題点があった．これでは，実用性はまったくない．

そこで，本論文では，この課題を解決する 1 つの方法として，要求精度に対応して数値の下位桁を切り捨てることを提案する．そして，この切り捨てにより発生する幾何矛盾を検出・除去する方法，すなわち**幾何無矛盾化アルゴリズム**を提案する．また，その有効性を計算機実験により実証したので報告する．

これまでの研究成果として，2 章では完全 4 次元同次幾何処理，3 章では超 3 角形幾何モデリング，に関して説明する．4 章では，本論文の主題である，幾何無矛盾化アルゴリズムを提案し説明する．5 章では，この提案アルゴリズムの計算機による評価実験結果に関して報告する．

2. 完全 4 次元理論と幾何処理の安定性

2.1 幾何処理の安定性

幾何処理の破綻の原因には，処理系が想定されるあ

† ATR 知能ロボティクス研究所
ATR Intelligent Robotics and Communication Laboratories
†† 早稲田大学
Waseda University

らゆる状況に対処しきれていないこと（複雑性）と，数値計算にともなう誤差（不安定性）の 2 点があげられる．現在の浮動小数点計算において本質的に問題なのは，真の誤差管理をしていないこと，またできていないことである．これにより処理系が暴走・破綻する．

特に後者は，アルゴリズムを正常に動作させる努力とは対照的に，比較的最近まで重要な問題であると認識されていたとはいえない．幾何演算の安定性（破綻の防止）を確保するためにこれまでに，大別すると以下の 3 つの手法が提案されてきている[7)～12)]．

（1） 許容誤差法

（2） 位相優先法

（3） 無誤差演算法

許容誤差法は，非常に接近している幾何要素（点，線，面等）は同一であると見なす手法である．また，位相優先法は，数値計算の結果はすでに決定されている位相構造と矛盾しない場合にのみ採用するという方法である[7)～9)]．しかし，これらの方法は，限定された問題に適用されているにすぎず，また完全な安定性はこれまでのところ保証されていない．

無誤差演算法は，誤差がまったくない正確な数値演算を行う．この方法では，誤差の管理が必要でなく完全な安定性が保証される．しかし，無誤差でできる演算の範囲が非常に狭いこと，および数値桁数が際限なく増大する等の本質的な問題点をかかえている．しかしながら，本手法は，多面体ソリッドモデリングの集合演算を含めた様々な幾何処理を，数値的に完全に安定化できる唯一の方法であると，筆者らは考えている．

2.2 完全 4 次元理論

山口らが提案している完全 4 次元同次幾何処理では，4 次元同次座標系 (X, Y, Z, w) において幾何処理を行う[10)～12)]．ここで，完全の意味は，すべての処理を一貫してこの 4 次元同次座標系で行うことにある．3 次元ユークリッド座標系幾何演算では，通常割り算が発生するが，この 4 次元同次処理では，すべ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．そこで，無限桁数の循環小数が発生することはない．

したがって，4 次元同次処理では，可変長ビットの無誤差整数演算と組み合わせることにより，数値的に破綻することのない無誤差幾何アルゴリズムを作成することが可能となる．

以下，本論文の提案手法では，この同次座標系をより一般化したプリュッカー座標系をベースにしているので，この座標系に関して説明する．

（1） **2 点から線の生成**

2 点 $\boldsymbol{V}_0 = (X_0, Y_0, Z_0, w_0)$，$\boldsymbol{V}_1 = (X_1, Y_1, Z_1, w_1)$ を通る直線のプリュッカー座標 $\boldsymbol{L}_{01}$ は次式で与えられる．

$$\boldsymbol{L}_{01} = [P_{01}, Q_{01}, R_{01}, S_{01}, T_{01}, U_{01}] \tag{1}$$

ここで，

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

$$R_{01} = \begin{vmatrix} Z_0 & w_0 \\ Z_1 & w_1 \end{vmatrix}, S_{01} = \begin{vmatrix} Y_0 & Z_0 \\ Y_1 & Z_1 \end{vmatrix},$$

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

（2） **3 点から面の生成（平面係数の算出）**

3 点 $\boldsymbol{V}_0 = (X_0, Y_0, Z_0, w_0)$，$\boldsymbol{V}_1 = (X_1, Y_1, Z_1, w_1)$，$\boldsymbol{V}_2 = (X_2, Y_2, Z_2, w_2)$ を通る面のプリュッカー座標 $\boldsymbol{F}_{012}$ は次式により与えられる．

$$\boldsymbol{F}_{012} = [A_{012}, B_{012}, C_{012}, D_{012}] \tag{2}$$

ここで，

$$A_{012} = \begin{vmatrix} Y_0 & Z_0 & w_0 \\ Y_1 & Z_1 & w_1 \\ Y_2 & Z_2 & w_2 \end{vmatrix}, B_{012} = \begin{vmatrix} Z_0 & X_0 & w_0 \\ Z_1 & X_1 & w_1 \\ Z_2 & X_2 & w_2 \end{vmatrix},$$

$$C_{012} = \begin{vmatrix} X_0 & Y_0 & w_0 \\ X_1 & Y_1 & w_1 \\ X_2 & Y_2 & w_2 \end{vmatrix}, D_{012} = -\begin{vmatrix} X_0 & Y_0 & Z_0 \\ X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \end{vmatrix}$$

（3） **線と面の交点**

2 点 $\boldsymbol{V}_a = (X_a, Y_a, Z_a, w_a)$，$\boldsymbol{V}_b = (X_b, Y_b, Z_b, w_b)$ を通る直線と，面 $\boldsymbol{F}_{012}$ との交点 $\boldsymbol{V}$ は次式で与えられる．

$$\boldsymbol{V} = S_{b012} \boldsymbol{V}_a - S_{a012} \boldsymbol{V}_b \tag{3}$$

ここで，

$$\begin{aligned} S_{a012} &= \boldsymbol{V}_a \boldsymbol{F}_{012} \\ &= X_a A_{012} + Y_a B_{012} + Z_a C_{012} + w_a D_{012} \quad (\leq 0) \\ S_{b012} &= \boldsymbol{V}_b \boldsymbol{F}_{012} \\ &= X_b A_{012} + Y_b B_{012} + Z_b C_{012} + w_b D_{012} \quad (\geq 0) \end{aligned}$$

3. 超 3 角形幾何モデリング

筆者らは，集合演算の複雑性の問題は,（任意の）多角形をベースとしていることに起因・主因していると考えている．3 次元幾何表現処理の 1 つの方法である BRep（Boundary Representation）では，**図 1** (1) に示すように，通常,（任意の）多角形面が用いられる[5),6)]．しかし，多角形 BRep はデータ構造および処理アルゴリズムともに複雑となり，その処理系も大規

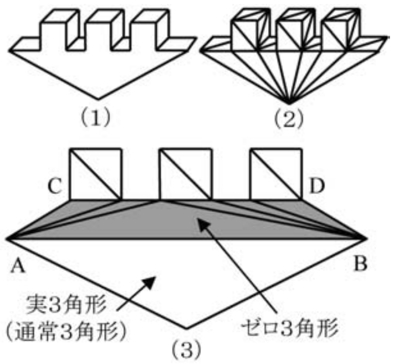

図 1　幾何表現
Fig. 1　Geometric representation.

模なものとなる．

一方，3 角形はこれ以上原理的に分割不可能な究極的に単純な基本図形である．すなわち，幾何学的には，3 角形面は「2 次元単体」と呼ばれる．そこで，3 角形面のみを用いて BRep を構成すると，ある意味で究極的に単純な幾何表現および処理が実現できる（図 1 (2) 参照）．以下，3 角形面を単に 3 角形と呼ぶ．しかし，集合演算等の処理の過程で，3 角形の数（データ量）が指数関数的に急激に増えてしまう等，大きな欠点もあわせ持つ．

3.1　超 3 角形幾何

筆者らは，このような 3 角形幾何が持つ欠点を克服した**超 3 角形幾何モデリング**を開発した[1)~3)]．図 1 (3) では，頂点 A，B，C および D は同一直線上にある．超 3 角形幾何とは，図 1 (3) に示すように，通常の 3 角形の概念を拡張し，3 角形の 3 つの頂点が同一直線上となる縮退した 3 角形を包含する 3 角形幾何表現である．この縮退した 3 角形を面積がゼロになることから「ゼロ 3 角形」，通常の 3 角形を「実 3 角形」と呼んでいる．

図 1 (2) に示す立体の前面と (3) の面の実 3 角形の数を比較すると，従来法では 13 個が必要であるが，超 3 角形法では約半分の 7 個で済む．ゼロ 3 角形は 6 個となるが，3 角形幾何処理からは通常除外することができる．文献 1) では，このゼロ 3 角形の面数抑制効果を定量的に示した．また，文献 2) では，超 3 角形幾何により集合演算が高速化できることを実証した．

このようにゼロ 3 角形により，3 角形幾何の単純性はほとんど損なうことなく，大きな欠点であった 3 角形の数の増大を抑制することができる．

3.2　無誤差 4 次元超 3 角形幾何

上述した超 3 角形幾何では，浮動小数点演算を用いた 3 次元ユークリッド座標系を前提としていた．集合演算の複雑性の問題は解決したが，不安定性の問題は依然として残ったままであった．この集合演算の演算誤差による不安定性を根本的に解決するアプローチとして，筆者らはこれまでに，完全 4 次元処理と超 3 角形幾何処理を融合した無誤差 4 次元超 3 角形幾何モデリングを提案した[4)]．

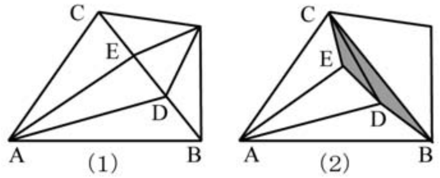

図 2　ゼロ 3 角形による分割
Fig. 2　Triangulation using zero triangles.

ここでも，ゼロ 3 角形は重要な役割を果たしている．たとえば，**図 2** では，3 角形 ABC の辺 BC が，D，E の順に 3 分割される．(1) に示す従来法では，交点 E は 1 次頂点 C（処理前に存在していた頂点）および 2 次頂点 D（処理途上に生成された頂点）から計算されるので 3 次頂点となる．このように，新たに生成される頂点の次数は順次増大し，その座標値桁数も際限なく増大していく．

一方，(2) に示すように，超 3 角形幾何では，実 3 角形 ABC の分割が繰り返されても，ゼロ 3 角形 BCD は分割されることはない．両端点がともに 1 次頂点である 1 次辺 BC は分割されることなく保存される．そこで，交点 E も，1 次辺 BC から求めることができ 2 次頂点となる．ゼロ 3 角形を用いることにより，特別な枠組みを必要とすることなく，1 次辺を保存することができる．これにより，数値桁数の不必要な増大を抑制することができる．

4.　幾何無矛盾化アルゴリズム

上述した無誤差 4 次元超 3 角形幾何を用いた集合演算アルゴリズム[4)] は，1 回限りの集合演算をある上限以下の数値桁数で行う方法を提供しているにすぎない．複数回の集合演算を実行すると，やはり桁数が際限なく増大していく．この無誤差演算にともなう桁数の増大は原理的なもので不可避である．

このような桁数の増大を回避する 1 つの方法は，必要精度に応じて数値桁数を切り捨てることである．しかし，幾何処理において，このような切り捨てを行うと，種々の幾何矛盾が発生する．そこで，ここでは，このような幾何矛盾を検出しかつ修正・除去する方法，すなわち幾何無矛盾化アルゴリズムを提案する．

本論文では，やはり無誤差 4 次元超 3 角形幾何を

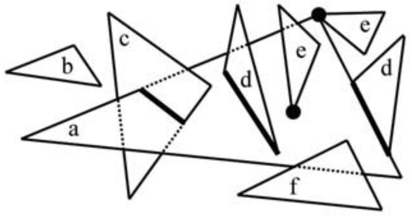

図 3　2 つの 3 角形の位置関係
Fig. 3　Positional relations between two triangles.

ベースとした，かつその特性を最大限利用した幾何無矛盾化アルゴリズムを提案する．また，無誤差数値演算は，可変長ビットの整数演算で行う．

頂点座標値の下位桁の数値の切り捨てを行うと頂点の位置が動く．また，超 3 角形 BRep では，その境界面は（複数の）3 角形のみで構成される．そこで，この頂点の移動により,「3 角形自体の形の変化」と「3 角形相互の相対位置の変化」が発生する．

前者では縮退 3 角形の発生があげられる．すなわち，1）3 頂点が同一直線上，2）2 頂点が同一，3）3 頂点が同一，となる 3 つのタイプの縮退 3 角形，つまりゼロ 3 角形が発生する可能性がある．超 3 角形（ゼロ 3 角形）幾何処理では，これらのゼロ 3 角形を特段消去する必要も，例外処理する必要もない．しかし，今回提案するアルゴリズムでは，2）と 3）のタイプのゼロ 3 角形は，消去するとデータ量（3 角形数）を削減することができるので，検出し消去することとした．この 2 つのタイプのゼロ 3 角形を幾何矛盾として定義した．また，1）のタイプのゼロ 3 角形も消去してもよいが，図 1 (2) と (3) に示すように，ゼロ 3 角形と実 3 角形を合計した全体の 3 角形数は減らないので，そのままとした．

後者に関しては，**図 3** に図示するように，2 つの 3 角形の位置関係をすべて列挙すると以下の 5 通りとなる．

(1)　離れている　(3 角形 a と b)
(2)　交差する　(a と c)
(3)　線（辺）で接する　(a と d)
(4)　点（頂点）で接する　(a と e)
(5)　重なる（同一平面上）　(a と f)

本論文の BRep では,「境界面は交差しない，かつ重ならない」という条件を設けている．これを「境界面条件 1」と呼ぶことにする．そこで, (2), (3) および (5) の場合はこの条件に反する（反する場合がある）ので，幾何矛盾と定義する. (2) および (3) の場合は境界面非交差化処理を行う．また, (5) の場合は重なり 3 角形除去処理を行う．

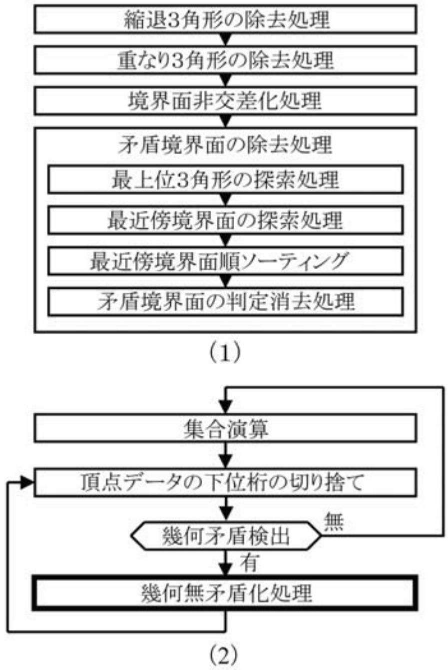

図 4　幾何無矛盾化アルゴリズムの全体
Fig. 4　Overview of geometric normalization algorithm.

さらに,「境界面条件 2」として,「境界面を挟んだ両側の領域は必ず立体の内部と外部の対をなし，無限遠点は外部となる」がある．ここで，内部とは境界面を構成する 3 角形の法線ベクトルの向きと定義する．外部はその逆向きとなる．

そこで，重なり 3 角形除去処理および境界面非交差化処理を行った後の境界面に対して，境界面条件 2 を満たさない矛盾境界面の除去処理を行う．これらの処理を幾何無矛盾化処理と呼ぶことにする．この幾何無矛盾化処理の全体フローは，**図 4** (1) に示すようになる．

また，図 4 (2) には，集合演算を含めた全体の処理の流れを示す．1 回の集合演算（無誤差）が行われるたびに，桁数が増大した「頂点座標値データの下位桁の切り捨て」が行われる．その後，幾何無矛盾化処理が行われる．1 回の幾何無矛盾化処理（無誤差）を行うと数値桁数が増大するので，ここでも頂点データの下位桁の切り捨てを行う．この切り捨てによって，新たな幾何矛盾が発生する場合がある．そこで，幾何無矛盾化処理は，新たな幾何矛盾が発生しなくなるまで繰り返される．

4.1　重なり 3 角形の除去

図 5 の例では，(1) の立体の断面図を (2) に示している．灰色領域が立体の内部である．矢印は境界面を

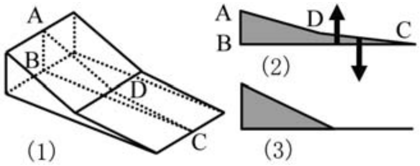

図 5 面の重なり
Fig. 5 Overlap of faces.

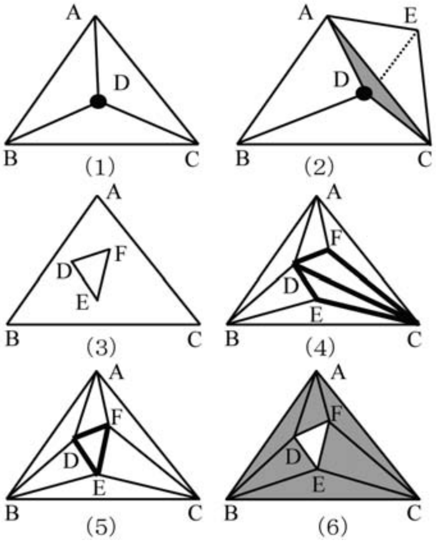

図 6 3 角形分割
Fig. 6 Split of triangular face.

構成する 3 角形の法線ベクトル（立体の内部側）を表している．頂点の桁数の切り捨てを行うと，たとえば (3) に示すように，境界面に重なる部分が発生する．すなわち，重なる 3 角形が発生する．ここでは，このように，両者の面の向きが互いに逆となる場合の重なり 3 角形の除去を行う．もう 1 つの重なる場合として，向きが同じとなる場合があるが，これは，境界面非交差化処理において一括処理する．

重なり 3 角形除去処理では，まず同一平面となり，かつ面の向きが逆となる，重なる 3 角形のペアを求める．そして，このペアにおいて，一方の 3 角形の頂点により他方の 3 角形を分割する．このときのパターンは**図 6** (1) および (2) に示すように，3 角形 ABC が，(1) その内部，(2) 辺上，で分割されるという 2 通りとなる．(2) の場合は，不必要な分割をなくすためにゼロ 3 角形分割を行う．3 角形 ADC がゼロ 3 角形となる．ゼロ 3 角形により，分割履歴の保存が行われ，不必要な数値桁数の増大を抑制することができる（図 2 参照）．一方，通常の分割では，隣接する 3 角形 ACE に，点線で示す分割が発生し（図 6 (2)），実 3 角形の数が増大していく．

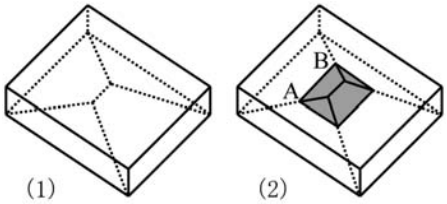

図 7 境界面交差立体
Fig. 7 Intersection of boundaries.

次に，分割された両者の 3 角形において，3 頂点を共有するが，3 辺を共有しない場合は，「位相変形」を行い，3 辺を共有する 3 角形を生成する．そして，3 辺を共有するすべての 3 角形のペアを消去する．

図 6 (3) の例では，3 角形のペア 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，たとえば (4) のように分割される（このケースでは DEF 側は細分割されない）．ここで，3 角形 DEF（図 6 (3)）は，ABC の分割により生成した 3 角形 CDE および CFD （図 6 (4)）とその 3 つの頂点を共有するが，辺 EF は他方に存在しない．

そこで，この 2 つの 3 角形が構成する 4 角形 CFDE（図 6 (4)）において，その対角線となる辺 CD を EF と付け替えることにより，(5) に示すように，新たな 3 角形 DEF と CFE を生成する．この 3 角形の位相変形処理の詳細は文献 1) に記述している．位相変形の結果，同じ 3 辺を共有することとなった両者の 3 角形 DEF を消去する．その結果は (6) となる．灰色領域が，境界面（3 角形）が存在する領域である．

4.2 境界面非交差化処理

図 7 (1) に示す立体では，頂点の桁数を切り捨てると，(2) のように底面が上面を突き抜けることが起こりうる（突き抜けた部分は面の表裏が逆になる）．境界面交差立体とは，このように境界面どうしが交差している立体である．このような場合は，境界面を構成する 3 角形は，交差する，あるいは辺で接する，となる．境界面非交差化処理では，この交差している面（3 角形）をその交線において分割し，それぞれ別の閉じた境界面（立体）とする．

境界面を構成する 3 角形が交線において分割された後の境界面接続切替え処理に関して，**図 8** を用いて説明する．図 8 (1a) は，境界面が完全に交差している場合である．ここで，a，b は面の接続関係を表し，同じ記号どうしの面が接続されている．また，灰色部分は立体の内側を表す．たとえば，図 7 (2) における辺 AB に垂直な断面図がこれにあたる．この場合は当然両者の面の接続関係を，交線において (1b) のように

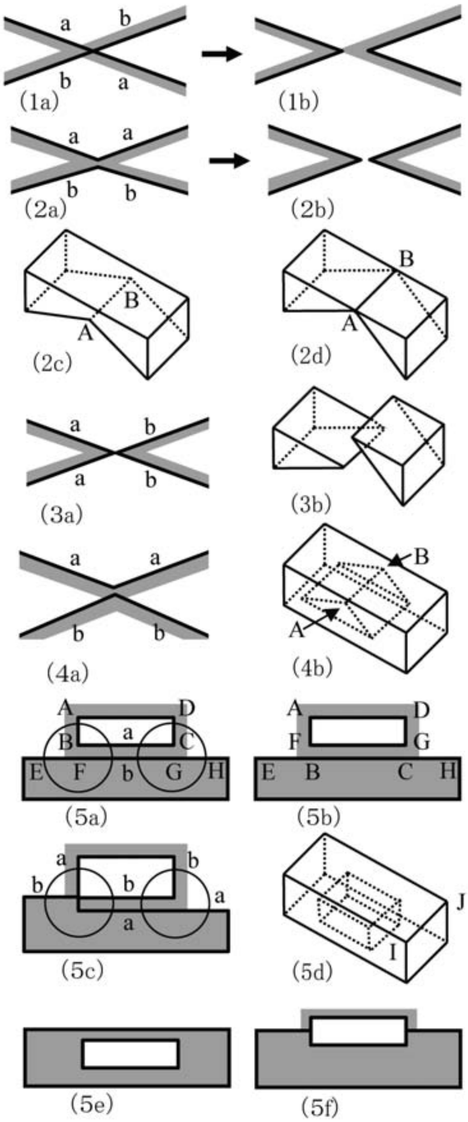

図 8　境界面の交差パターン
Fig. 8　Intersection patterns of boundaries.

切り替える．(1b) では境界面が切り離されて図示されているが，接続関係を分かりやすくするためであり，実際は接したままである．以下，すべての図においてそうである．

(2a) の場合は境界面が立体の内側において接している．これは，たとえば，(2c) の立体において辺 AB が (2d) のように上面に含まれる状態になった場合である．この場合も (2b) のように接続関係を切り替える．(3a) は境界面が立体の外側において接している場合である．これは，たとえば，(3b) のように離れた 2 つの立体が移動し，(2d) のように，辺 AB で接するとなった場合である．この場合は接続関係を切り替えない．(4a) は境界面が面の向きが同じで接する場合である．これは，たとえば (4b) に示すように，1 つの立体の内部に別の立体が完全に入り込み，辺 AB が上面と接するようになった場合である．この場合も接続関係は切り替えない．

(5a) は，境界面において，重なる部分（断面 BC と FG が同一平面上，面の向き同じ）がある場合である．これは，たとえば，(5d) に示す立体のように，内部に空洞があり，この空洞が完全に外にはみ出し接している立体である．(5d) の辺 IJ の中点を通り，かつ IJ に垂直な面による断面図を (5e) に示す．灰色領域が立体の内部であり，ここではその内部に空洞が存在する．頂点データの数値桁数の切り捨てにより，(5f) では空洞が部分的に外部にはみ出している（矛盾立体）．さらに，(5a) では空洞が完全に外部にはみ出し，立体の上面（断面 EFGH）と空洞の底面（断面 BC）が接する状態となっている（矛盾立体）．

このような場合は，重なる 2 つの平面部分をそれぞれ a（断面 BC）と b（断面 FG）とする（図 8 (5a)）．そして，a と接続している面（断面 AB，CD）が，b と接続している面（断面 EF，GH）に対して外側となる場合のみ，接続関係を切り替える．(5a) では，AB，CD ともに，この場合にあてはまるので，(5b) のように接続関係を切り替える（断面 AFGD，EBCH となる）．ただし，ここでは，重なる面が入れ替わるだけで，立体の形はまったく変化しない．

(5c) は (5a) と同じように重なる部分があるが，違いは，(5a) は 2 つの境界面が存在し，(5c) は 1 つの境界面が自己交差している点である．この場合も (5a) の場合と同じルールを適応する．すなわち，(5c) の左側のみを切り替える．結果は (5b) となり，2 つの立体（境界面）に分離される．すなわち，この切替えルールにより，重なる部分がある自己交差立体を切り離すことができる．

ここでは,「外側となる場合」を切り替えるとしたが，この基準を逆にして，内側としてもよい．また，(5a) および (5c) では重なる部分の面の向きが同じであるが，面の向きが逆となる場合も存在する．しかし，このような場合は,「重なり 3 角形の除去処理」（同一平面かつ面の向きが逆）において，検出除去処理されるので，ここでは対象外となる．

4.3　矛盾境界面の除去

4.3.1　最上位 3 角形の探索

最上位 3 角形とは，1 つの境界面を構成する超 3 角形の中で，Z 座標値が最大となる最上位頂点を持ち，

かつ Z 座標軸正方向から見て，他の 3 角形に覆われない部分がある実 3 角形と定義する．4 次元同次座標系において，頂点 $\boldsymbol{V}_0 = (X_0, Y_0, Z_0, w_0)$ の Z 座標値が頂点 $\boldsymbol{V}_1 = (X_1, Y_1, Z_1, w_1)$ より大きいとは，以下の式が成り立つことである．

$$Z_0 w_1 > Z_1 w_0 \qquad (w_0, w_1 > 0) \tag{4}$$

また，3 角形 $\boldsymbol{F}_{012}$ が $\boldsymbol{F}_{345}$ に対して上面であるとは，$\boldsymbol{F}_{012}$ と $\boldsymbol{F}_{345}$ が Z 座標軸方向からみて重なり，かつ 3 角形 $\boldsymbol{F}_{012}$ の 3 つの頂点 $\boldsymbol{V}_0$，$\boldsymbol{V}_1$，$\boldsymbol{V}_2$ において以下の式が成り立つと定義する．ただし，$\boldsymbol{F}_{012}$ と $\boldsymbol{F}_{345}$ は交差しないとする．

$$\begin{array}{ll} (0 < s_0) \vee (0 < s_1) \vee (0 < s_2) & (0 < C_{345}) \\ (s_0 < 0) \vee (s_1 < 0) \vee (s_2 < 0) & (C_{345} < 0) \end{array} \tag{5}$$

ここで，

$$\begin{aligned} s_0 &= \boldsymbol{V}_0 \boldsymbol{F}_{345} \\ &= X_0 A_{345} + Y_0 B_{345} + Z_0 C_{345} + w_0 D_{345} \\ s_1 &= \boldsymbol{V}_1 \boldsymbol{F}_{345} \\ &= X_1 A_{345} + Y_1 B_{345} + Z_1 C_{345} + w_1 D_{345} \\ s_2 &= \boldsymbol{V}_2 \boldsymbol{F}_{345} \\ &= X_2 A_{345} + Y_2 B_{345} + Z_2 C_{345} + w_2 D_{345} \end{aligned}$$

1 つの境界面における最上位 3 角形の探索処理では，まず最上位頂点 $\boldsymbol{V}_{max}$ を求める．そして，この $\boldsymbol{V}_{max}$ を頂点に持つ任意の基準実 3 角形 tbase を選択し，最上位 3 角形 tmax の初期値を tmax=tbase とする．ただし，tbase の法線ベクトルの Z 座標値がゼロでないものを選択する．そして，$\boldsymbol{V}_{max}$ を頂点に持つ実 3 角形 t において，tbase と Z 座標軸方向からみて重なり，かつ t が tmax の上面となる場合は tmax=t とする．この処理をすべての t において繰り返すことにより最上位 3 角形 tmax を求めることができる．

図 9 の境界面 b0 において，$\boldsymbol{V}_{max}$ は A，B，C，D の 4 頂点となる（どれでもよい）．また，tmax は 3 角形 ABC または ADB となる（$\boldsymbol{V}_{max}$ を持てばどちらでもよい）．ここでは，$\boldsymbol{V}_{max}$ =C，tmax=ABC とした．

4.3.2 最近傍境界面の探索

1 つの境界面 b の最近傍境界面 bnear とは，以下の条件を満たす境界面と定義する．

（1）b の最上位頂点に Z 座標軸と平行に正方向に立てた半直線 L と交差あるいは接する．この交点あるいは接点を，$\boldsymbol{V}_{near}$ とすると，$\boldsymbol{V}_{near}$ と $\boldsymbol{V}_{max}$ の距離は最小となる．ここで，$\boldsymbol{V}_{near}$ を最近傍頂点と呼ぶ．

（2）b の最上位 3 角形 tmax の上面 tnear を持つ．tnear が複数ある場合は，その中で最も下面となる面を持つ．この面を最近傍 3 角形と呼ぶ．ここで，下面とは，3 角形 t1 が 3 角形 t0 の上面となる場合，逆に t0 は t1 に対して下面と呼ぶ．

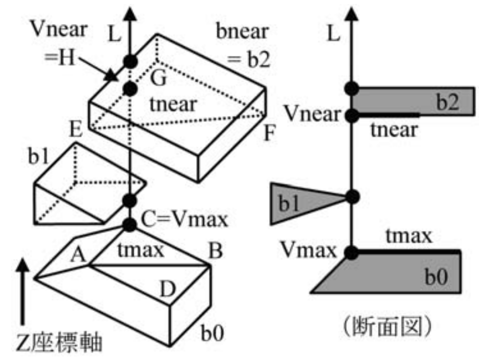

図 9 境界面間の位置関係
Fig. 9 Positional relations among boundaries.

図 9 の例では，境界面 b1 は L と接するが，tmax=ABC と重なる面（上面）が存在しない．そこで，b1 は b0 の最近傍境界面ではない．一方，b2 は L と接し（$\boldsymbol{V}_{near}$ =H），かつ tmax と重なる tnear=EFG が存在する．そこで，b2 が b0 の最近傍境界面となる．

4.3.3 最近傍境界面順ソーティング

幾何無矛盾化処理では，矛盾境界面と無矛盾境界面が混在しているので，ある 1 つの境界面が他の境界面との関係で，矛盾しているかどうかをいきなり判定することはできない．唯一できる場合は，最近傍境界面が存在しない境界面に対してである．ここに，境界面データをソーティングする理由がある．

最近傍境界面順ソーティングでは，境界面 bi の bnear=bj となり，かつ bi が bj よりもデータの上位にある場合は，両者のデータの順番を入れ替える．この処理により，ある境界面 bi の bnear は，必ず bi よりも上位データとなり，かつデータ先頭には必ず最近傍境界面が存在しない境界面がくる．そこで，先頭の境界面は矛盾境界面かどうかを必ず確定することができる．

この結果をもとにデータの上位から順次，境界面の矛盾/無矛盾判定を行うことができる．図 9 の例では，b1 および b2 の最近傍境界面は存在しないので，ソーティング後の境界面データの順番は上位から（b1, b2, b0），(b2, b1, b0)，(b2, b0, b1）のいずれかとなる（いずれでもよい）．

4.3.4 矛盾境界面の判定消去

「境界面条件 2」に基づいて，ある境界面の表裏が最近傍境界面との関係で矛盾する場合は，この境界面データを消去する．**図 10** を用いて，矛盾/無矛盾境

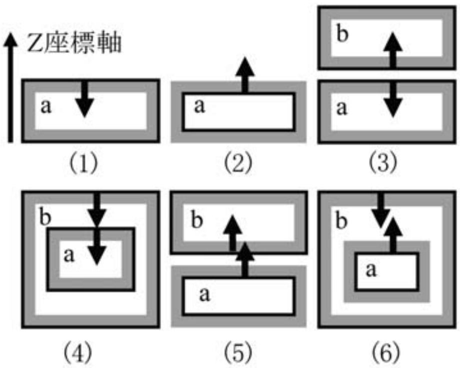

図 10 2 つの境界面の組合せ
Fig. 10 Combinations of two boundaries.

界面判定処理を説明する．図 10 は立体の断面図を表している．灰色側は立体の内部を表している．境界面 a の矢印は，最上位 3 角形の法線ベクトルの Z 座標の向き（符号）を表している．以下，これを a の向きと呼ぶ．また，b における矢印はすべて，a に対する最近傍 3 角形の法線ベクトルの Z 座標の向き（符号）を表している．以下，これを b の向きと呼ぶ．

(1) および (2) は，境界面 a の最近傍境界面が存在しない場合である．この場合は，a の向きを用いて判定を行う．(1) では，a の向きは負であり，a は空間に対して閉じている（無限遠点が外部となる）ので，a は無矛盾境界面となる．一方，(2) では，正であり，空間に対して開いている（無限遠点が内部となる）ので，矛盾境界面となる．境界面の先頭データは，最近傍境界面が存在しないので，必ずこの (1) あるいは (2) のどちらかとなる．そして，矛盾/無矛盾を確定することができる．

(3)〜(6) の場合は，a の最近傍境界面 b が存在する場合である．(3)〜(6) の場合は，a の向きと，b の向きの組合せを，「境界面条件 2」に基づいてチェックする．両者の向きが逆となる (3) と (6) の場合は，a は b に対して無矛盾境界面となり，両者の向きが同じとなる (4) と (5) の場合は，a は b に対して矛盾境界面となる．そして，矛盾境界面と判定された面はすべて消去する．

5. 計算機実験と考察

今回提案した幾何無矛盾化アルゴリズムを実証評価するために，このアルゴリズムを計算機に実装し，実証評価実験を行った．計算機は，DELL DIMENSION 8250 （Pentium4 3.06 GHz，1.5 GB RAM，WindowsXP）を使用した．当然のことながら，この PC は可変長整数演算をサポートしていないので，可変長整数演算ソフトウェアを開発し，これを用いて無誤差演算を行った．評価形状として，以下の 2 形状を採用した．

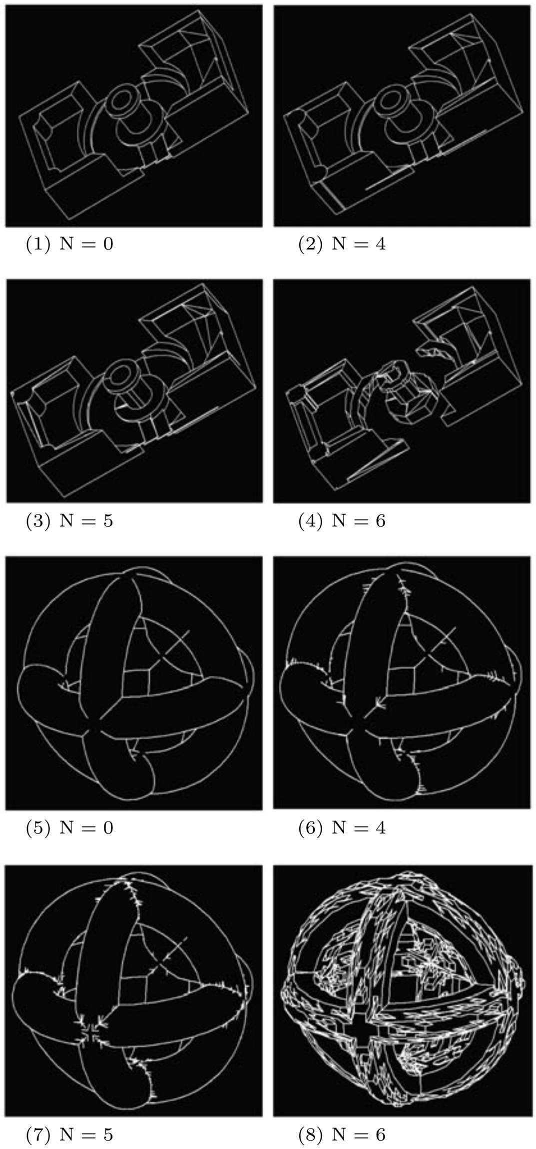

図 11 生成形状
Fig. 11 Generated shapes.

（1）部品

プリミティブ 18 個の和差（17 回の集合演算）により生成される形状（**図 11** (1)〜(4) 参照）．この評価形状は，平面はもちろん，斜面，円筒面，穴，薄版等の種々の幾何的特徴をあわせ持つ．

（2）トーラス

曲面形状の代表例として 3 個のトーラスの形状和（2 回の集合演算，図 11 (5)〜(8) 参照）．

表 1 計算機実験結果（部品）
Table 1 Experimental results (mechanical part).

N	縮退	重なり	矛盾面	データ量
0	8	0	0	2,732
1	364	696	4	2,206
2	268	694	0	2,536
3	290	683	0	2,360
4	230	693	1	2,510
5	270	560	2	2,078
6	708	185	0	876
7	1,244	4	0	0

表 2 計算機実験結果（トーラス）
Table 2 Experimental results (torus).

N	縮退	重なり	矛盾面	データ量
0	0	0	0	18,720
1	126	0	7	19,434
2	106	0	25	19,514
3	52	0	2	19,450
4	286	0	4	19,202
5	560	17	13	17,724
6	2,654	93	1	9,150
7	18,848	96	0	0

集合演算を行う前の頂点データ（整数表現）の最大ビット長 $B = 32$ とした．このとき，1 回の集合演算を行った後の頂点データ（2 次頂点）の最大桁数は $5B + 9 = 169$ となる．この詳細は文献 4) に記述している．1 回の集合演算後（集合演算には文献 4) で開発した処理系を使用），4 次元同次座標で表現されている頂点データ (X, Y, Z, w) を 3 次元ユークリッド座標 (X/w, Y/w, Z/w) に変換し，小数点以下を切り捨て整数化した．さらに，この各整数座標値の下位 N 桁（10 進）を切り捨て，4 次元同次座標 (X’，Y’，Z’，1) に戻した．このとき，発生した幾何矛盾の種類とその数をモニタリングした．そして，本提案の無矛盾化処理を行い，次の集合演算を行った．N は，0 から 7 まで変化させて計算機実験を行った．

計算機実験の結果は，**表 1**，**表 2** および図 11 となった．表 1 および表 2 の「縮退」,「重なり」および「矛盾面」の各項目はそれぞれ，すべての頂点データの切り捨て過程において発生した，縮退 3 角形の数（単位：個），重なり 3 角形面のペアの数（単位：対），矛盾境界面の数（単位：面）の総和である．また，データ量はすべての処理を終了した時点での 3 角形面の数（単位：個）である．

今回の 2 つの形状例において，N を 0 から 7 まで変化させたすべて場合において，提案した幾何無矛盾化アルゴリズムは破綻することなく安定して動作し，幾何矛盾を除去した．表 1，表 2 および図 11 に示すように，どちらの形状においても，$N = 6$ を境にして，最終形状データ量（3 角形面数）は極端に減少し，形状は極端につぶれていった．そして，$N = 7$ の場合は，データ量が 0 となり，形状は存在しなくなった．形状は徐々につぶれていくのではなく，ある点（$N = 6$）から，加速度的に大きくつぶれた．

形状がつぶれる過程をより詳細に観察すると，部品では，$N = 5$ までは「微細構造」が現れるだけで，全体形状に大きな変化は現れない．$N = 6$ のときに，薄板が完全に消滅し，円筒面が崩れる等の大きな変化が現れた．また，トーラスにおいても同様で，$N = 5$ までは交差部分（交線付近）に微細構造が現れるだけである．やはり $N = 6$ のときに，表面（曲面）全体が完全に崩れた．

今回測定した「縮退 3 角形数」,「重なり 3 角形数」および「矛盾境界面数」は，モデリング精度の評価にも使うことができる．すなわち，これらの値がゼロあるいはゼロに近ければ，モデリング精度は十分に確保されているといえる．また，処理系が安定している（破綻しない）ので，今回の実験のように精度を連続的に変化させて，生成形状を表示しその状態を視覚的に確認しながら，モデリング精度を決定することも可能となった．

6. おわりに

本論文では，以下の特徴を持つ幾何無矛盾化アルゴリズムを提案した．さらに，計算機実験により，この提案アルゴリズムが破綻することなく安定して動作することを実証した．

（1）必要精度に応じて，頂点の数値データの下位桁を切り捨てることにより，無誤差演算に付随する数値桁数の増大という問題点を解消し，連続した無誤差幾何演算（集合演算）に対応した．

（2）この数値の切り捨てにより発生する幾何矛盾を検出し除去する方法を提案した．このアルゴリズムは，縮退 3 角形の除去処理，重なり 3 角形面の除去処理，境界面非交差化処理および矛盾境界面の除去処理から構成される．

（3）この幾何無矛盾化アルゴリズムは，完全 4 次元処理と超 3 角形幾何を融合した無誤差 4 次元超 3 角形幾何をベースとしたものである．特に，超 3 角形（ゼロ 3 角形）幾何により，アルゴリズムが単純化され，かつ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る．また，4 次元同次処理を用いることにより，割り算を排除した無誤差整数幾何演算を実現した．これらにより，破綻が生じない安定な幾何無矛盾化処理系を実現した．

今回提案した幾何無矛盾化アルゴリズムは，任意精度の幾何矛盾を安定して取り除くことができる．そこで，この技術は，無誤差・無矛盾幾何コンピューティングの１つの基盤技術となる可能性がある．また，この技術をベースにして発展展開することにより，可変精度（多重精度）幾何モデリングを実現することができる．

今後の課題としては，本方法の正当性をさらに多方面から検証することである．今回の成果は，どちらかといえば，計算機実験を主体とした実証的側面から導出されたものである．したがって，すべての幾何矛盾が完全に取り除けることを理論的に保証しているとはいえない．そこで，証明等を含む数学的・幾何学的な立場からの理論検証が必要不可欠である．

また，今回は無誤差演算（長ビット整数演算）をソフトウェアでシミュレーションしているために，処理時間が非常に遅くなっている．処理時間に言及すると，「部品」の例では最大 30 分,「トーラス」の例では最大 20 時間を要している．そこで，もう１つの課題は，実用性を確保するために，数千ビットクラスの高速長ビット整数演算ハードウェアを研究開発することである．

今回提案した幾何無矛盾化アルゴリズムは，すべての演算において，無誤差演算（多倍長整数演算）を用いている．すなわち，必ずしも無誤差演算を必要としない処理，および計算結果が必要ではなくその符号が判定できればよい処理に対しても，無誤差演算を用いている．そこで，前者には浮動小数点演算を，後者には適応的符号判定法等を導入することにより，ハードウェアに頼らない高速化の余地も残されている．このようなソフトウェア（アルゴリズム）面の高速化の研究も有意義である．

謝辞 本研究において，いろいろとご指導と協力・支援をいただいた，ATR 知能ロボティクス研究所の萩田所長，小暮室長，そして研究者各位に，感謝の意を表する．

参 考 文 献

1) 荒川佳樹：面積ゼロ３角形を用いた３角形 BRep，情報処理学会論文誌，Vol.36, No.2, pp.362–373 (1995).
2) 荒川佳樹：超３角形 BRep における高速形状演算アルゴリズム，情報処理学会論文誌，Vol.37, No.4, pp.624–634 (1996).
3) 荒川佳樹，山口富士夫：超３角形 BRep における Edge-based データ構造と形状演算アルゴリズム，情報処理学会論文誌，Vol.39, No.1, pp.39–49 (1998).
4) 荒川佳樹，山口富士夫：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，Vol.40, No.9, pp.3471–3482 (1999).
5) Requicha, A.A.G. and Voelcker, H.B.: Solid Modeling: A Historical Summary and Contemporary Assessment, *IEEE Computer Graphics and Applications*, Vol.2, No.2, pp.9–24 (1982).
6) Requicha, A.A.G. and Voelcker, H.B.: Solid Modeling: Current Status and Research Directions, *IEEE Computer Graphics and Applications*, Vol.3, No.7, pp.25–37 (1983).
7) 杉原厚吉：数値誤差と幾何学的整合性，電子情報通信学会誌，Vol.76, No.6, pp.618–625 (1993).
8) Sugihara, K.: A Robust and Consistent Algorithm for Intersecting Convex Polyhedra, *EUROGRAPHICS '94*, Vol.13, No.3, pp.C45-C54 (1994).
9) 杉原厚吉：計算幾何工学，培風館 (1994).
10) 山口富士夫：４次元理論による図形・形状処理工学，日刊工業新聞社 (1996).
11) Yamaguchi, F.: *Computer-Aided Geometric Design*, Springer-Verlag (2002).
12) 吉田典正：完全４次元処理に基づくソリッドモデリング技術に関する研究，早稲田大学博士論文 (1997).

(平成 16 年 1 月 29 日受付)
(平成 16 年 5 月 21 日採録)

荒川　佳樹（正会員）

1954 年生．1978 年早稲田大学理工学部工業経営学科卒業．1980 年同大学院理工学研究科修士課程機械工学専攻修了．同年松下電器産業（株）入社．1990 年郵政省通信総合研究所入所．2003 年（株）国際電気通信基礎技術研究所（ATR）入所．現在，同知能ロボティクス研究所主幹研究員．超３角形/超４面体幾何による図形・画像処理通信の研究に従事．電子情報通信学会，精密工学会各会員．

山口富士夫（正会員）

1935 年生．1959 年早稲田大学第一理工学部機械工学専攻卒業．同年横河電機製作所入社．1967 年機械振興協会技術研究所入所．1978 年九州芸術工科大学工業設計学科助教授．1979 年〜1980 年米国ユタ大学コンピュータサイエンス学科客員準教授．1986 年早稲田大学理工学部機械工学科教授．コンピュータグラフィックスと曲面の自動設計に関する研究に従事．著書『4 次元理論による図形・形状処理工学』（日刊工業新聞社）ほか．精密工学会会員．