

ClassA スタイリングデータ作成の 効率化に挑んだ CAD 技術の実装アルゴリズム

Implemented Algorithms of Challenging CAD Technology aimed at Increasing Efficiency
of Class A Surfacing

篠原 智 義*
Tomoyoshi SHINOHARA
勝 部 弘 嗣*
Hirotsugu KATSUBE

春 原 慶 子*
Keiko SUNOHARA
井 出 大 介*
Daisuke IDE

坂 居 三 郎*
Saburo SAKAI

要 旨

本システムの開発にあたっては ClassA サーフェス作成工数50%削減の目標を達成するため、ClassAサーフェス完成までの三つのモデリングフェーズに着目した。各領域での作業効率を革新的に向上させる新技術を開発して、CAD機能としての具現化に成功した。本文では、とりわけ大きな効果を得た代表的な機能についてこれを実現した技術を紹介する。

具体的にはまず、クレイモデルの測定データにフィットする基本曲面の作成において、物理的な板の「しなり」をヒントに、曲面の振る舞いや品質を最適化モデルとして定義し、これを解くことで高品質なフィッティング曲面の作成技術を確立した。

次に ClassA サーフェス数の約 10%を占めるフィレット面の作成において、計算アルゴリズムの根本的な刷新により高速化と高精度化を同時に達成した。

ClassAデータ作成の最終段階においては、デザイナーの審美的視点で目標形状への微修正が可能な TDM(ターゲットドリブンモデリング)機能の実現により、繊細な意匠形状の表現が求められるClassAサーフェスの仕上げ作業を効率化した。

ABSTRACT

In order to satisfy a development target for a 50% reduction in the man hours expended in the generation of Class A surfaces, the development of the system discussed in this paper focused on three phases of modeling leading to the completion of Class A surfaces. New technologies were developed to innovatively increase work efficiency in each area, and these were realized as CAD functions. This paper will discuss the technologies that helped to realize the functions that have had the greatest effect on performance.

First, for the generation of basic surfaces fitted over scanned clay model data, a technology was developed that defines surface behavior and quality as an optimization model based on hints from the stiffness of physical plates. The solution of the optimization model generates high-quality fitting surfaces.

Next, the computational algorithms for the generation of the fillet surfaces that represent approximately 10% of the number of Class A surfaces were fundamentally revised, resulting in a simultaneous increase in speed and accuracy.

For the final phase of Class A data generation, a target-driven modeling (TDM) function was realized that helps enable designers to make minor adjustments in accordance with their aesthetic judgment. This function has increased the efficiency of finishing work for Class A surfaces, which demands the expression of finely detailed design shapes.

1. まえがき

ClassA サーフェス作成工数 50%削減の目標を実現させるために、ClassAサーフェス完成までの三つのモデリングフェーズに着目し、その作業効率を向上できる CAD 機能の実現を目指した⁽¹⁾。三つの作業プロセスを効率化できる

機能要件は以下の(A)から(C)である。

(A)クレイモデルの測定データからフィッティング面を作成するフェーズにおいて、測定データとの精度を維持しながら ClassA データの品質を満たす曲面を効率的に作成できること。

(B)コーナ部や集合部など、複雑な意匠の形状を表現する

* HG 管理室

曲面の作成段階で、高品位な曲面をレスポンス良く作成できること。

- (C)一旦できあがった ClassA サーフェスに対して意匠の傾向の調整やデザイン変更をおこなう場合など、形状目標は明確だが品質を確保しながらの作業に手間がかかる工程の効率化ができること。

我々の求めるサーフェス品質と計算性能を満たす技術は存在せず、本システムの目標達成のためにはこれらの要件を満たす技術の開発が必要不可欠であった。

本システムでは効果的な技術の開発により、上記の機能要件を満たす CAD 機能を実現した。以下の(1)から(3)は本システムで確立した新技術である。本稿ではこれらの CAD 機能実現技術を紹介する。

- (1)大容量の測定データをインタラクティブに操作、参照できるポリゴン演算処理の高速化を図った。そのうえで、物理的な板の「しなり」をヒントに、測定データにフィットする曲面の作成技術を開発した。これは、サーフェスの振る舞いや品質を定義した最適化モデルを解くことで高品質なフィッティング曲面を得る技術である。
- (2)従来システムから培ってきた ClassA データ作成機能のノウハウを継承すると同時に、その高速化、高精度化を進め、185 機能にも及ぶコマンド群の開発をおこなった。とりわけ ClassA サーフェス数の約 10%を占めるフィレット面の作成においては、計算アルゴリズムの根本的な刷新により高速化と高精度化を同時に達成し、複雑なフィレット形状の計算において従来比で 90%以上の高速化を実現した。
- (3)ClassA サーフェスの形状修正作業においては、形状目標は明確な一方で形状完成までに手間がかかる作業工程の効率化を図った。本システムでは CAD 操作の新しいコンセプトである TDM(ターゲットドリブンモデリング)を考案し、これを実現できる技術開発に成功した。TDM を CAD 機能として具現化したことにより、形状修正過程でのモデリング作業の簡略化を図った。

2. ポリゴンカーネル機能概要

ポリゴンメッシュの測定データから ClassA 曲面データを作成する過程では、ポリゴンメッシュの前処理や形状解析が必要となる。本システムで扱う測定データは数百万の頂点を含む大容量データであるが、インタラクティブな操作が要求されるため、ポリゴンカーネルのデータ構造とメモリ管理機構の特性を最大限に引き出すアルゴリズムの構築が必要である⁽¹⁾。主な機能要件とその実現課題を以下に示す。

- (1)測定データのモデル形状を視覚的に確認するために、メッシュの断面線形状をポリラインセクションとして

作成する必要があるが、切断平面との交錯点の計算と、交錯点群のポリライン化での効率化が課題となる。

- (2)ポリゴンメッシュの一部の領域を選択したり、曲面作成の領域設定のためには、3D 曲線をポリゴンメッシュ上に投影する必要がある。しかしポリゴンメッシュ上に穴がある場合や、また投影線が複数存在する場合の対応が課題である。
- (3)測定データの欠落やその他の要因によりポリゴンメッシュにあいた穴を埋める機能が必要であるが、穴の形状や大きさによらず、等方的な穴埋めパッチの高速な作成が課題である。
- (4)ClassA サーフェスと測定データの形状の差異を把握するために、ポリゴンメッシュと ClassA サーフェスの離れ距離を計算する機能が必要とされる。離れ距離を計算するには、直線とポリゴンメッシュの高速な交錯計算が必要である。

本章では、ポリゴンカーネル機能実現のための、(1)から(4)の課題に対する具体的な解決技法を紹介する。

2.1. ポリラインセクション作成手法

本機能は、隣接する交錯点間の相互接続処理を効率的におこなうための技術的課題を解決するため、

- (1)断平面とポリゴンメッシュ位相要素(エッジ、頂点)間の交錯判定計算
 - (2)隣接する交錯点の検索および接続処理
- の二段階の処理で構成した。

交錯判定では、まず断平面と全頂点間との位置関係を調べ、続いて断平面と全エッジとの交錯判定をおこなう。頂点が断平面の表側にある場合は正の符号を、裏側にある場合は負の符号を割り当てる。頂点と断平面間の距離がトランス値以内であれば、交錯頂点とする。次に交錯エッジ判定として、両端の頂点の符号が異なるエッジを、断平面と交錯しているとみなし、エッジ上の交錯点位置を計算する。Fig. 1で、点線は断平面を表し、黒い丸と白い丸はそれぞれ、断平面の表側と、裏側にある頂点を表し、赤い丸は、断平面と交錯頂点および、エッジ上の交錯点を表す。

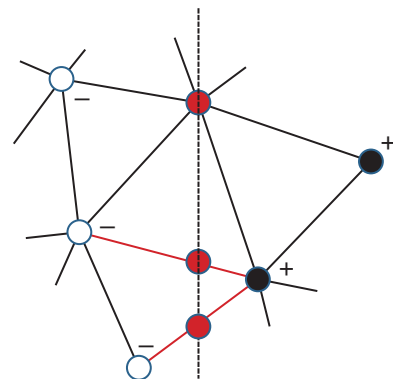


Fig. 1 Polyline-section algorithm

接続処理では、任意の交錯エッジまたは交錯頂点を開始点とし、順次隣接する交錯位相要素を探索して、一つながりの交錯点の配列として再構成する。交錯エッジの場合は、両側のフェイスに属する全位相要素の中から、交錯頂点の場合は、周囲一隣接フェイスに属する全位相要素の中から、それぞれ隣接する交錯位相要素の探索を開始する。隣接する交錯位相要素がすべて検索された時点で、1本のポリラインセクションが完成する。残りの交錯位相要素も同様に再構成し、ポリラインセクションの作成が完了する。

2.2. 曲線投影手法

本来の投影結果のポリラインが、穴をまたぐ場合や、複数本に及ぶ場合にもすべての投影計算を正しく出力する必要がある。本手法では、前項のポリラインセクションを利用して以下の処理ステップで構成した。処理概要を Fig. 2 に示す。

- (1) 投影したい曲線を離散ポリラインに分割する。
- (2) 離散ポリラインのセグメントと投影方向より断平面を定義する。
- (3) 上記(2)の断平面とポリゴンメッシュとの交線ポリラインを求める。
- (4) 交線ポリラインから投影前のセグメントに相当する部分を抽出し、投影セグメントとする。
- (5) ハーフエッジデータ構造を参照しながら隣接するセクションフラグメントの探索をおこない、上記(4)で求めたすべてのセグメント投影形状が相互接続された段階で、曲線形状の抽出が完成する。

2.3. 自動穴埋め処理手法

あらゆる穴輪郭形状において穴埋めパッチ内のポリゴン品質を確保するには、同アルゴリズムの使用が不可欠であるが、大きな穴形状ではパフォーマンスが低下する。この課題を解決するために、本手法では以下の処理ステップで構成した。

- (1) 穴の輪郭形状は、対側の要素を持たないハーフエッジを検出しクラスタリング処理することで得る。穴の輪郭を抽出する段階の図を Fig. 3 (a) に示す。
- (2) 穴の輪郭が 30 頂点未満の場合はパッチの総面積が最小となる条件で Liepa の穴埋めアルゴリズム⁽²⁾を実行し、穴埋めパッチを得る。穴の輪郭形状が 30 頂点以上の場合、下記(3)から(8)の独自アルゴリズムによる初期パッチを作成した後にリメッシュ処理をおこなう。
- (3) 穴の輪郭形状の中心座標を求める。
- (4) 穴の輪郭形状の各頂点から、輪郭形状ポリラインの平均エッジ長だけ中心に向かった場所に新規に頂点を作成し、新規エッジを伸長する。このとき、隣り合う新規頂点間の距離が閾値よりも小さい場合は、一つの頂点に融合する。
- (5) 隣り合う新規エッジ同士を STRIP 状もしくは FAN 状にテセレーションし、穴の外辺を埋める帯状パッチを作成する。
- (6) ここで帯状パッチの内側にできた一回り小さな穴の大きさが、10 頂点よりも小さくなるまで、(1)から(5)の処理をおこなう。Fig. 3 (b) は第二世代目の帯状パッチが作成された様子を示す。
- (7) 穴輪郭形状の大きさが10頂点以下になった時点で、任意の頂点を始点として FAN 状に穴を埋めることで初期パッチが完成する。Fig. 3 (c) はこの段階を示す。
- (8) 穴を埋めた初期パッチに対してリメッシュ処理をおこなう。リメッシュ処理は、穴埋めパッチが周囲形状と同等なポリゴン精度を得るために、目標のエッジ長を輪郭形状ポリラインの平均エッジ長とした等方的なポリゴンで満たされるような位相処理を繰り返す⁽³⁾。(1)から(7)までに作成された初期パッチは、最初から目標のエッジ長を考慮したポリゴン群で構成されており、リメッシュ処理の高速化に有利である。Fig. 3 (d) は穴埋め処理が完了した状態を示す。

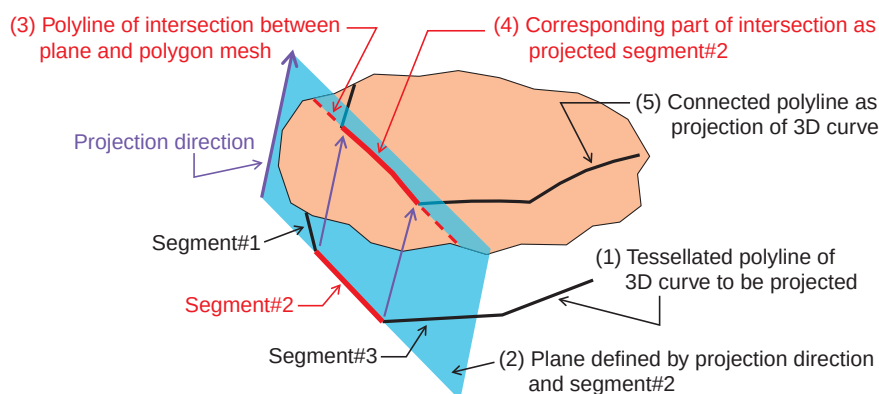


Fig. 2 Algorithm for 3D curve projection on polygon mesh

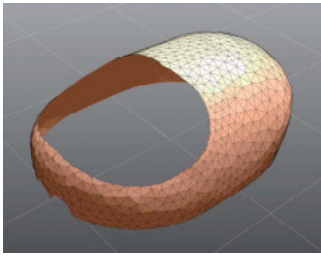


Fig. 3 (a) Hole detection

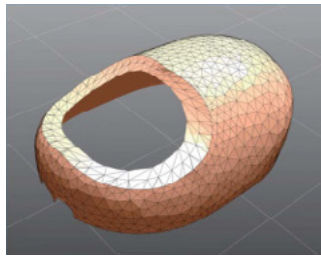


Fig. 3 (b) Initial patch creation

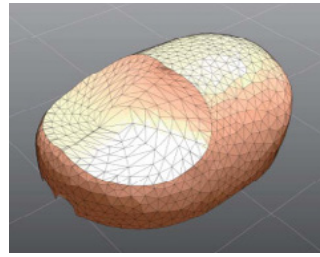


Fig. 3 (c) Completed initial patch

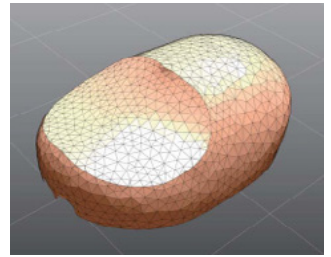


Fig. 3 (d) Hole filled with final patch

2.4. メッシュ - 幾何要素間離れ測定機能

レイとメッシュ間の交錯計算を高速化するためには、効率的な空間分割を構築する必要もあり、本機能では以下の処理ステップでこれを実現した。

(1) 高効率で高速な空間分割

(2) 上記(1)で作成した空間分割を活用した高速交錯計算

空間分割作成は、以下の処理で構成される。

- (i) 任意のポリゴンを選択し、その重心から各軸方向に対して距離が一定距離以内(ターゲットメッシュの長手方向直径の 40 分の 1)にある立方体形状領域を定める。
- (ii) シードを含めこの領域に含まれるすべてのポリゴンをグリッドの構成ポリゴンセットとする。
- (iii) 構成ポリゴンセットの AABB (Axis-Aligned-Bounding-Box) を計算し、これをグリッド形状とする。
- (iv) まだどのグリッドにも属していないポリゴンを対象として、すべてのポリゴンがいずれかのグリッドに分類されるまで(i)から(iii)を繰り返す。

各グリッドは空間配置的に重なりをもっているが、その中に含まれるポリゴンのセットに重複はない。つまり、任意のポリゴンは、いずれかの一つのグリッドにのみ属している。交錯計算処理はまずこのグリッドとの交錯判定を計算し、

その後交錯が確定したグリッド内に存在するすべてのポリゴンに対して交錯を計算する二段階の処理である。

すべてのグリッドは、AABBとして構築してあるので、レイとグリッドの交錯判定は高速におこなうことができる。

Fig. 4 (a) はターゲットメッシュと外部からのレイ 10 本との交錯計算において、レイの集合との交錯が判定されたグリッドの集合を示している。グリッドがその内部に保持しているポリゴンの集合をピンク色で示した。

レイとポリゴンの交錯判定は、このピンク色のポリゴンのセットのみを対象とするので、判定計算回数の縮小、すなわちパフォーマンス向上に大きく寄与する。レイとポリゴンとの交錯判定処理は以下のように構成される。

- (i) レイが X 軸となるように、処理対象メッシュの全頂点セットを座標変換する。
- (ii) 各ポリゴンにおいて、三つの頂点すべての Y 座標の正負がすべて同じ符号の場合、または Z 座標の正負がすべて同じ符号の場合は交錯候補から外す。
- (iii) 上記(ii)のふるいを通過したポリゴンのうち、X 軸との交錯点がポリゴン内部に存在するものすべてが交錯ポリゴンとなる。既に内外判定をおこなっているため、交錯点における 3D 座標を計算することは容易である。

Fig. 4 (b) はターゲットメッシュに向けた複数のレイとの交錯判定結果を示す。



Fig. 4 (a) Ray-grid intersection

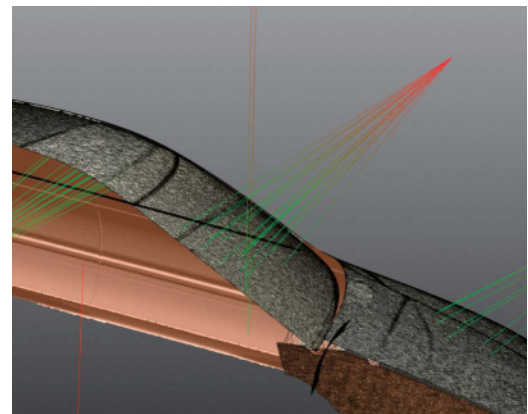


Fig. 4 (b) Ray-mesh intersection

3. フィッティング技術

ClassA データは、クレイモデルの 3D 測定データを入力として作成される。従来この工程では測定データとの傾向や離れを比較し、手動での曲面修正を繰り返していた。この作業は多大な工数を必要とするため、従来より測定データから意匠形状を反映した曲面を自動的に作成する仕組みは数多く考えられてきた。これを「測定フィッティング技術」と定義する。

従来の測定フィッティング技術では、クレイモデルのうねりや測定時の誤差がそのまま出力に反映されるため、ClassA データとしての曲面品質は保てなかった。本システムではデザイン形状を維持しながらも、クレイモデルのうねりや測定誤差を除去し、ClassA 品質を満たした曲面を生成できる以下の技術を開発した。

(1) 単面としての品質を保障する「インテリジェントフィッティング技術」

(2) 複数曲面間の連続性を保障する「マージ技術」

本章ではこれらの技術を確立した具現化手法を述べる。

3.1. インテリジェントフィッティング技術

曲面の品質や、曲面間の連続性を維持するためには、曲面の接線 G1、曲率 G2 をコントロールする必要がある。さらに、曲率のうねりは品質に影響を与えるので、曲率 G2 の変化率 G3 もコントロールする必要がある。

これらをコントロールするために、本システムでは「しなり (Stiffness)」という概念を導入した。ここで「しなり」とは、例えば、実物のモデルに金属板の定規を押し当てた場合、表面の形状傾向には追従するが、細かな凹凸は無視される様子をいう。これは、この定規自体がある一定以上には曲がらない硬さを持つからである。

Fig. 5 は「しなり」のふるまいを模式的に表す。肌色の測定データは細かな段差を伴っている。ここで任意の初期曲面 (水色) から、測定データとの離れを最小化するフィッティングをおこなうと、うねりを伴った紫色の曲面が出力

される。そこで、本システムではフィッティングにより動ける範囲に、曲率変化 G3 の幅として図中の赤矢印で示す制約条件を付加した。これにより、測定データに追従しながらも、曲面のうねりを制御したインテリジェントフィッティング面 (赤色) が得られる。

このアイデアのもと、本システムでは最小二乗法のアプローチに加え、求めるべき形状の性質を条件として付与した条件付き収束演算として定義した。具体的には、下記のような収束演算の目的関数と制約条件を定義した。

目的関数: (測定データとの離れ)² + (各点での G3 の値)

制約条件: (G3 値) < (しなりの上限)

制約条件として G3 値を用いたのは、これが定規の「しなり」のように、うねりを制御することができると考えたためである。ここで、過剰な G3 値制約は測定データに対する追従性を低下させ、過小な G3 値制約は曲面のうねりを生みやすくなる。これを金属板でいう硬さとしてとらえ、押し当てる金属板の厚みを変えるように、「しなり」の調整度合いを制約条件とした。また、G3 値が増大すると曲面全体にうねりを生じやすくなるので、これを収束演算におけるエネルギー関数として、目的関数にも含めた。

上記を具体的に実装する手法について述べる。システムとしての入力および出力は、下記のとおりである。

入力: 3D 測定データ

曲面の領域を定義する 4 本の境界線

NURBS 曲面の制御点 (CP) の数

「しなり」の大きさ

出力: 形状品質を満たした曲面

入力情報から基準となる初期面を生成して、フィッティングをおこなう。曲面の数学的表現では NURBS 曲面を用いた。入力の境界線より定義できる u, v 空間内に、制御点を等間隔に配置して初期面を定義する。NURBS 曲面では、曲面は制御点 (CP) の座標値に対して基底関数 $B(u, v)$ を掛け合わせることで表現される。基底関数は、パラメータ空間 u, v と、空間の節目 (NURBS 曲面の *knot*) を用いて一意に定まる。いま、 n 個の制御点 $a_0, a_1, \dots, a_n$ によるパラメータ

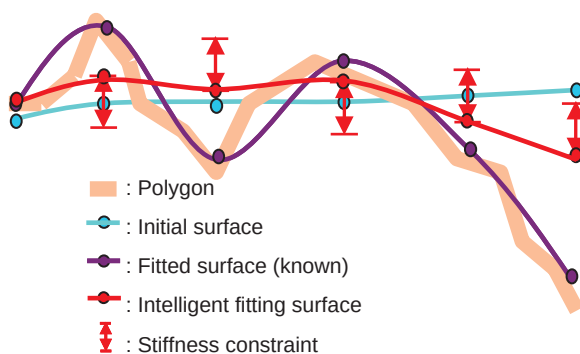
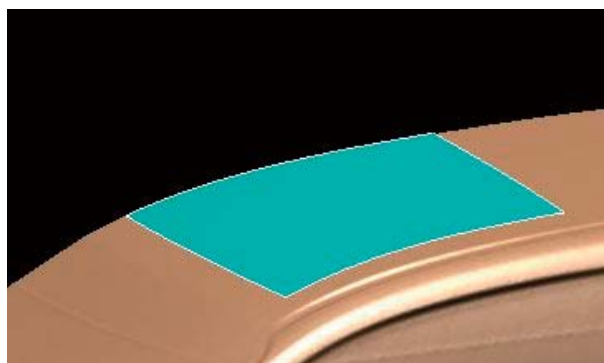


Fig. 5 Stiffness as constraint

空間 u, v での曲面の式 $S(u, v)$ は、以下のように表せる。

$$S(u, v) = (B_0(u, v), B_1(u, v), \dots, B_n(u, v)) \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = [B][a] \quad (1)$$

ここで、基底関数の行列を $[B]$ と記した。

式(1)に示されるように、曲面は線形の演算として書き表せる。フィッティングをおこなう領域や制御点の数はあらかじめ与えられているので、インテリジェントフィッティング曲面生成のために求めるべき未知数は制御点列 $a_0, a_1, \dots, a_n$ である。

$(B_0(u, v), B_1(u, v), \dots, B_n(u, v))$ が定まっているため、 u, v 方向の微分は、 $a_0, a_1, \dots, a_n$ とは独立した演算として定義できる。これを u 方向 n 階目の微分行列 $[D_u^n]$ とあらかじめ定義しておき、後述する式(9)、式(19)での G2 および G3 の計算に用いる。

$$\partial S / \partial u = [D_u^1][B][a], \quad \partial^2 S / \partial u^2 = [D_u^2][B][a], \quad \partial^3 S / \partial u^3 = [D_u^3][B][a] \quad (2)$$

まず、測定点との離れの目的関数 C_{Dist} を求める。 m 個の測定点 $P_0, P_1, \dots, P_m$ に対して、基準面上で最も近いパラメータ地点 $(u_0, v_0), \dots, (u_m, v_m)$ を定義する。

すると、 C_{Dist} は P とパラメータ地点との離れの二乗となるため、

$$C_{Dist} = \begin{pmatrix} P_0 \\ P_1 \\ \vdots \\ P_m \end{pmatrix} - \begin{pmatrix} B_0(u_0, v_0) & B_1(u_0, v_0) & \dots & B_n(u_0, v_0) \\ B_0(u_1, v_1) & B_1(u_1, v_1) & \dots & B_n(u_1, v_1) \\ \vdots & \vdots & \ddots & \vdots \\ B_0(u_m, v_m) & B_1(u_m, v_m) & \dots & B_n(u_m, v_m) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} \quad (3)$$

$$= [P]^T [P] - 2[P][B]^T [a] + [a]^T [B]^T [B][a]$$

$$\therefore C_{Dist} = [f]^T [a] + \frac{1}{2} [a]^T [H][a] \quad (4)$$

$$[f] = -[B]^T [P] : \text{線形成分 } 1 \times n (X \text{ に依存している}) \quad (5)$$

$$[H] = [B]^T [B] : \text{二次成分 } n \times n (X \text{ に依存しない}) \quad (6)$$

となる。ここで、 $[P]^T [P]$ 項は定数になるため除外し、収束演算に用いる項を C_{Dist} とした。

f, H の要素数は測定データの個数 m には依存せず、制御点の個数 n にのみ依存していることがわかる。よって一度初期条件として式(5)、式(6)を計算すると、最適計算の際には膨大な量の測定データを用いずにすみ、計算量を少なくできる。

加えて、G3 値の目的関数 C_{Energy} は、微分行列を用いると下記のように定義できる。

$$C_{Energy} = \frac{\partial^3 S}{\partial u^3} + \frac{\partial^3 S}{\partial v^3} + \frac{\partial^3 S}{\partial u^2 \partial v} + \frac{\partial^3 S}{\partial u \partial v^2} \quad (7)$$

$$= [D_u^3][D_v^3][D_{uv}^3][B][a] + [D_u^3][D_v^3][B][a] + [D_u^2][D_v^3][B][a] + [D_u^3][D_v^2][B][a]$$

最終的な目的関数 C は、下記ようになる、

$$C = C_{Dist} + \lambda C_{Energy} \quad \lambda : \text{定数} \quad (8)$$

次に制約条件を求める。いま uv 方向に対する G3 の値が、 b_L より大きく b_H より小さいとする。この時の各点における制約条件は、微分行列を用いると下記のように決められる。

$$b_L^{u3} < \frac{\partial^3 S}{\partial u^3} < b_H^{u3}, b_L^{v3} < \frac{\partial^3 S}{\partial v^3} < b_H^{v3}, b_L^{uv2} < \frac{\partial^3 S}{\partial u^2 \partial v} < b_H^{uv2}, b_L^{uv2} < \frac{\partial^3 S}{\partial u \partial v^2} < b_H^{uv2} \quad (9)$$

ここで、両端の b^{u3}, b^{v3}, b^{uv2} は u, v 空間で各階数微分した G3 の制約値であり、指定する「しなり」の度合いにより変動する。具体的には、G3 制約の上限値は絶対的、かつ定量的に定義し、上限値での出力結果よりどれだけ硬くするかによって G3 制約の下限値を決定する。

さて、条件付き線形行列の収束演算には「線形計画法 (Linear Programming)」という手法が用いられることが多い。しかし、今回は目的関数内に二次の項目が出るため、二次式に拡張した「二次計画法 (Quadratic Programming)」という手法を用いた。二次計画法を解くアルゴリズムには、OOQP⁽⁴⁾ という手法を用いた。

式(4)、式(9)に示されるように、今回用いる目的関数 C 、制約条件 A は、下記のような一般系として書き表すことができる。以下では OOQP⁽⁴⁾ の表記にあわせ、行列の表現 $[X]$ を簡略化し X と記述する。

$$C = \frac{1}{2} a^T H a + f^T a \quad (10)$$

$$A a > b \quad (11)$$

a : 解

H : 目的関数(二次成分)

f : 目的関数(線形成分)

A : 制約条件(計算式)

b : 制約条件(値)

くわえて、後述する式(19)で示される G2 接続の制約条件などを考えて、等号の制約条件を下記のように定義する。

$$D a = d \quad (12)$$

D : 制約条件(計算式)

d : 制約条件(値)

式(11)、式(12)の条件を満たし、式(10)を最小化する解 a を求める。制約条件がない場合には、最適解 $\underline{a}$ においては、

$$dC / da = 0 \quad \therefore H \underline{a} = -f \quad (13)$$

が成り立つが、制約条件がある場合にはこれを満たすとは限らない。そこで、それぞれの制約条件に対応した新たな未知数 y, z, s を考え、下記のような式 L を定義する。こ

で, s, z は不等号を成立させるため, 正の値である.

$$L = \frac{1}{2} a^T H a + f^T a - y(Da - d) - z(Aa - s - b) \quad (14)$$

$$s \geq 0 \perp z \geq 0$$

ここで, 第一項と第二項が目的関数を, 第三項が等号の制約条件を, 第四項が不等号の制約条件を示す. この時最適解 $\underline{a}$ においては, 下記の条件式が成り立つ.

$$dL/d\underline{a} = 0 \quad \therefore \quad H\underline{a} - D^T y - A^T z = -f \quad (15)$$

$$dL/dy = 0 \quad \therefore \quad A^T \underline{a} = b \quad (16)$$

$$dL/dz = 0 \quad \therefore \quad C^T z - s = d \quad (17)$$

$$dL/ds = 0 \quad \therefore \quad sz = 0 \quad (18)$$

本手法では, これらの条件式を Newton 法により解き, $\underline{a}$ を導出する.

3.2. マージ技術

最終的な ClassA データは, 互いに位置 G0, 接線 G1, 曲率 G2 の連続性を持った複数の面から成り立っている. ここでは, この連続性を自動で保障することを目標とする.

NURBS 式においては, 端末部分での数値を一意に定めることにより, 任意の連続性が確保できる. しかし, この手法では, 曲面の内部で大きな乱れが生じ, ClassA 曲面として不適であった. 本システムでは, インテリジェントフィッティング面の制約条件に加えて, 曲面間の連続性の制約条件を設定した.

式(8)で定義した目的関数は, 単一の曲面だけでなく複数曲面を一度に計算することができる.

いま, 曲面 1 の制御点 $a_0, a_1 \dots a_n$, 曲面 2 の制御点 $b_0, b_1 \dots b_n$ を同時に計算することを考える. ここで, 曲面 1 の (u_0, v_0) で定義される点と, 曲面 2 の制御点 (u_1, v_1) で定義される点にて u 方向での G2 連続を保障するためには,

$$\frac{\partial^2 S_1}{\partial u_1^2} - \frac{\partial^2 S_2}{\partial u_2^2} = 0 \quad (19)$$

が成り立てばよい. そこで, $a_0, a_1 \dots a_n, b_0, b_1 \dots b_n$ を未知数として式(8)を構築し, インテリジェントフィッティング曲面に加えて式(19)の条件式を加えることで, 境界部に曲率連続性を保った二つの曲面を得ることができる.

4. フィレット面張りの高速化と高精度化技術

近年の自動車のデザインでは形状が複雑化し, その曲面数も 5 年前と比較して約 2 倍となっており, ClassA データ

作成工数の増加が課題となっている. ClassA データでは意匠形状の再現性だけではなく, 曲面のデータ精度も重要であるため, データ作成工数の削減には, データ作成の高速化と高精度化の両立が必要である.

Fig. 6 で示すとおり, フィレットとは曲面同士が交わる角を丸めた曲面形状である. 自動車のデザインでは ClassA サーフェス数の約 10% を占めるフィレット曲面について, 計算処理の高速化と, 曲面品質の高精度化をおこなった.

Honda のフィレットは, サポート曲面群とフィレット曲面とが G2 連続をなす滑らかな形状が主流である. そのようなフィレット作成においては, まず Fig. 7 で示すようなサポート曲面に G2 接続するリブと呼ばれる骨組みの曲線群を求め, そのリブ群を用いて曲面を作成する. 本システムでは, リブの作成手法の改善によって高速化, 面張り手法の改善によって高精度化をおこなった.

4.1. 高速化技術

サポート曲面と G2 接続するリブ曲線は, Fig. 8 が示すように, 両端点の 1 階微分ベクトルと 2 階微分ベクトルを入力条件にした, 5 次 1 セグメントの B-Spline 曲線として求める. 1 階微分ベクトルは接線を表し, 2 階微分ベクトルは曲率を表す.

従来の方では, 端点の接線と曲率を求めるために, 稜線の法線断面平面とサポート曲面群との交線から微分値を取得していた. 交線計算は収束計算であるため, 高品質なフィレットでは計算時間が増大する課題がある. 本システ

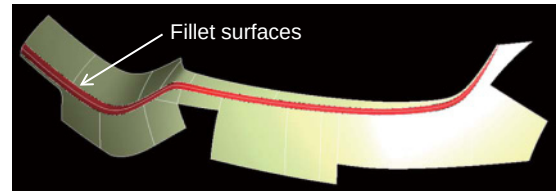


Fig. 6 Sample of fillet

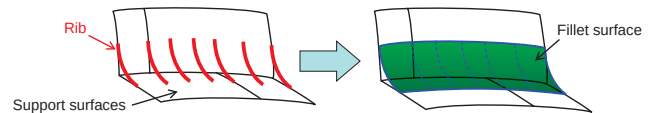


Fig. 7 Sample of rib

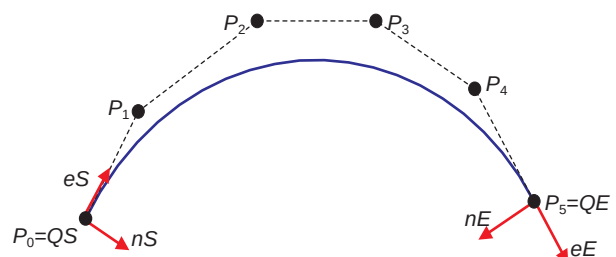


Fig. 8 Control points of 5 degree rib curve

ムでは、交線計算をせずに、サポート曲面上のリブ端点での面単位法線ベクトルと断面平面単位法線ベクトル、さらにサポート曲面の法曲率を用いて、リブ端点での曲率を高速に求める方法を開発した。

曲線 $C(t)$ の単位接線ベクトルを $e(t)$ 、単位主法線ベクトルを $n(t)$ 、曲率関数を $\kappa(t)$ とする。

パラメータ $0 \leq t \leq 1$ 、制御点 $P_0, P_1, P_2, P_3, P_4, P_5$ とすると、5次Bezier曲線 $C(t)$ は次のように表示される。

$$C(t) = (1-t)^5 P_0 + 5(1-t)^4 t P_1 + 10(1-t)^3 t^2 P_2 + 10(1-t)^2 t^3 P_3 + 5(1-t) t^4 P_4 + t^5 P_5 \quad (20)$$

曲線の1階微分ベクトル、2階微分ベクトルは、

$$\Delta P_i = P_{i+1} - P_i \quad (21)$$

$$\Delta^2 P_i = \Delta(\Delta P_i) = \Delta P_{i+1} - \Delta P_i = P_{i+2} - 2P_{i+1} + P_i \quad (22)$$

とする。式(21)、(22)を用いて、

$$\frac{dC}{dt} = 5 \{ (1-t)^4 \Delta P_0 + 4(1-t)^3 t \Delta P_1 + 6(1-t)^2 t^2 \Delta P_2 + 4(1-t) t^3 \Delta P_3 + t^4 \Delta P_4 \} \quad (23)$$

$$\frac{d^2 C}{dt^2} = 20 \{ (1-t)^3 \Delta^2 P_0 + 3(1-t)^2 t \Delta^2 P_1 + 3(1-t) t^2 \Delta^2 P_2 + t^3 \Delta^2 P_3 \} \quad (24)$$

と表示できる。特に始終点での1階微分ベクトル、2階微分ベクトルは

$$\left. \frac{dC}{dt} \right|_{t=0} = 5(\Delta P_0), \quad \left. \frac{dC}{dt} \right|_{t=1} = 5(\Delta P_4) \quad (25)$$

$$\left. \frac{d^2 C}{dt^2} \right|_{t=0} = 20(\Delta P_1 - \Delta P_0), \quad \left. \frac{d^2 C}{dt^2} \right|_{t=1} = 20(\Delta P_4 - \Delta P_3) \quad (26)$$

である。

1階微分ベクトルは、面上リブ端点での接平面と、稜線の法線断面平面から求められ、式(25)より P_0 と P_4 が決まる。2階微分ベクトルは、弧長パラメータを用いて算出されたフレネーの公式⁽⁵⁾から、1階微分ベクトルに直交する⁽⁶⁾ことが示されており、その大きさは、以下の方法から求め、式(26)より P_2 と P_3 が決まる。

Fig. 9 は、以下のリブ端点での曲率計算方法を示す。

n : リブ端点でのサポート曲面単位法線ベクトル

m : 断面平面単位法線ベクトル

e : リブ端点でのリブ単位接線ベクトル(接平面と断面平面との交線から算出)

α : n と m のなす角度

κ_n : リブ端点での e 方向法曲率

κ : リブ端点での断面線曲率

とすると、 κ は断面線を陽に求めなくても、次式で計算される。

$$\kappa = \left| \frac{\kappa_n}{\sin \alpha} \right| \quad (27)$$

ただし、 $\alpha = 0$ の場合は、断面平面と接平面が一致するので、リブは求められない。

また、曲面の面上点での法曲率の算出方法は以下のとおりであり、Fig. 10 に示す。

曲面式を $S(u, v)$ 、単位法線ベクトルを $n(u, v)$ とする。曲面上の点 p で単位接ベクトル e を与えると、 e と n を含む断面平面が定まる。この平面で曲面の断面線を求め、断面線の p での曲率(n の側に曲率中心がある場合を正、 n の逆側に曲率中心がある場合を負とする符号付き曲率を考える)を面上点 p における e 方向の法曲率(normal curvature)という。接ベクトル e を接平面内で1周させると、一般に法曲率も変化する。その最大値と最小値を特に主曲率(principal curvature)という。主曲率を与える接ベクトルは、互いに直交するという性質があるので、特に主方向(principal direction)という。これらの算出は以下のとおりである。 $\langle, \rangle$ はベクトル内積を表し、曲面のパラメータ u, v での偏微分を $S_u, S_v, S_{uu}, S_{uv}, S_{vv}$ で表す。

まず、法曲率を算出したい面上点位置で、第1基本量とよばれる

$$E = \langle S_u, S_u \rangle, F = \langle S_u, S_v \rangle, G = \langle S_v, S_v \rangle \quad (28)$$

および、第2基本量とよばれる

$$L = \langle S_{uu}, n \rangle, M = \langle S_{uv}, n \rangle, N = \langle S_{vv}, n \rangle \quad (29)$$

を算出する。

単位接ベクトル e を

$$e = \xi S_u + \eta S_v \quad (30)$$

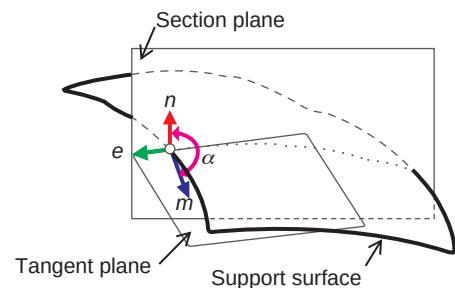


Fig. 9 Method of calculation of curvature at end point of rib

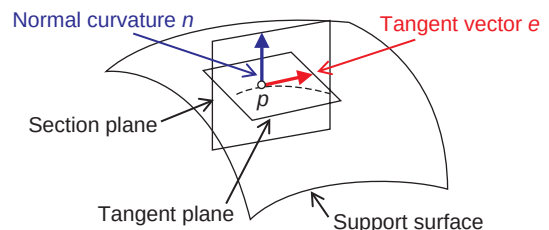


Fig. 10 Normal curvature of surface

と表示する。ただし、

$$\langle e, e \rangle = \langle \xi S_u + \eta S_v, \xi S_u + \eta S_v \rangle = E\xi^2 + 2F\xi\eta + G\eta^2 = 1 \quad (31)$$

である。

主曲率を κ_j 対応する主方向を $e_j = \xi_j S_u + \eta_j S_v$ ($j=1, 2$) とすると、主曲率・主方向は、次の固有値・固有ベクトルを求める問題に帰着する。

$$\frac{1}{EG-F^2} \begin{bmatrix} G, & -F \\ -F, & E \end{bmatrix} \begin{bmatrix} L, & M \\ M, & N \end{bmatrix} \begin{bmatrix} \xi_j \\ \eta_j \end{bmatrix} = \kappa_j \begin{bmatrix} \xi_j \\ \eta_j \end{bmatrix} \quad (32)$$

さて、上記の固有値問題を解いて、主方向と主曲率が得られたとする。二つの主方向は接平面の正規直交基底であるから、

$$e = (\cos\theta)e_1 + (\sin\theta)e_2 \quad (33)$$

と表せるが、このとき、 e 方向の法曲率 κ_n と主曲率の間には、次のオイラーの公式が成立する。

$$\kappa_n = (\cos\theta)^2 \kappa_1 + (\sin\theta)^2 \kappa_2 \quad (34)$$

この公式を用いて、 e 方向の法曲率 κ_n を算出でき、式 (27) よりリブ端点での断面線曲率を求め、5 次 1 セグメントの B-Spline 曲線を求めることによって、交線計算を用いずに高速にリブを求める。

4.2. 高精度化技術

従来の技術では、サポート曲面とフィレット曲面との連続性の保障において、法線断面平面とサポート曲面、フィレットとの交線の接続性を検査していた。交線計算は収束計算であるため、計算時間がかかりながらも、交線位置以外の連続性は保障できなかった。本システムでは、周辺面の微分値情報である Fig. 11 に示す境界横断導関数 (Cross Boundary Derivative: CBD) を用いた段階的な Gordon 面張りによって、サポート曲面とフィレット面の G2 接続を実現した。

フィレット面の作成は、以下のように段階的におこなう。

(1) サポート曲面との G1 接続を保障するために、端リブ

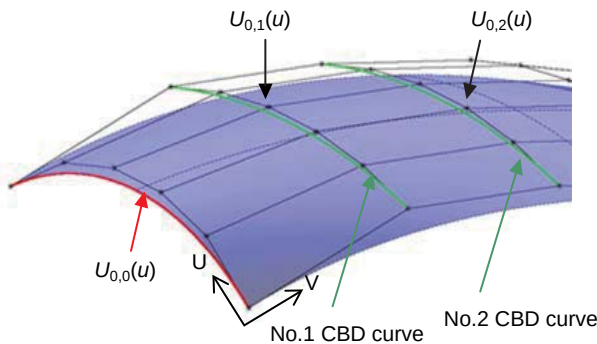


Fig. 11 Cross-boundary derivative curves No.1 and No.2 at $U_0(u)$

$U_{0,0}(u)$ とその近傍のリブ 2 本 $U_{0,1}(u)$, $U_{0,2}(u)$ からなる局所曲面を考え、 v 方向偏微分により 1 階 CBD 曲線を求め、サポート曲面の接線ベクトルをもとに大域変形と呼ばれるツイストベクトル整合をおこなう⁽⁷⁾。

- (2) サポート曲面との G2 接続を保障するため、端リブ $U_{0,0}(u)$ とその一つ内側の参考リブ $U_{0,1}(u)$ と、先に求めた 1 階 CBD 曲線からなる局所曲面を考え、2 階偏微分より 2 階 CBD 曲線を求める。
- (3) 上記 1 階 CBD 曲線と 2 階 CBD 曲線をもとに、サポート曲面との連続性を考えずに G0 Gordon 面を作成する⁽⁷⁾。
- (4) G0 Gordon 面より再度 1 階 CBD 曲線を取り出し、サポート曲面との G2 連続性を確保するための再度大域変形をおこなう。
- (5) 上記の 1 階 CBD 曲線、2 階 CBD 曲線、G0 Gordon 面の 1 階 CBD 曲線、R 止まり線とリブ列で Gordon 面を作成し、出力面とする。

4.2.1. 端リブ沿い 1 階 CBD の決定

始端リブの場合、始端リブ $U_{0,0}(u)$ と、その近傍の参考リブ 2 本 $U_{0,1}(u)$, $U_{0,2}(u)$ の合計 3 リブを通過する二次 Lagrange 補間多項式 $\xi_{0,s}(v)$, $\xi_{1,s}(v)$, $\xi_{2,s}(v)$ を考える。

$$\text{局所曲面 } T(u,v) = U_{0,0}(u)\xi_{0,s}(v) + U_{0,1}(u)\xi_{1,s}(v) + U_{0,2}(u)\xi_{2,s}(v) \text{ から, } (35)$$

$$\text{始端 CBD } \frac{\partial T}{\partial v}(u, b_0) = U_{0,0}(u)\xi'_{0,s}(b_0) + U_{0,1}(u)\xi'_{1,s}(b_0) + U_{0,2}(u)\xi'_{2,s}(b_0) \quad (36)$$

を計算する。このとき、コーナツイストベクトルをサポート曲面から決め、1 階 CBD 制御曲線の端点接線ベクトルをツイスト整合する必要がある。これを、CBD 制御曲線の大域変形と呼ぶ。本システムでは、端リブの端点でサポート曲面との G2 ツイスト整合を考慮するため、1 階 CBD 制御曲線のツイスト整合および、そのツイストベクトルに変更するための大域変形をおこなった⁽⁷⁾。

また、終端リブにおいても同様に 1 階 CBD 制御曲線を算出するが、このとき終端方向から微分する。

4.2.2. 端リブ沿い 2 階 CBD の決定

端リブと、それより一つ内側のリブ、および端リブ沿いの 1 階 CBD 関数から、 1×2 次の Coons 面補間を考える。この Coons 面 $T(u, v)$ で 2 階 CBD 関数を算出する。Coons 面 $T(u, v)$ は

$$T(u,v) = T_1(u,v) + T_2(u,v) - T_3(u,v) \quad (37)$$

ただし、 $\mu_j(u)$ は u 方向の補間関数、 $\lambda_j(v)$ は v 方向の補間関数、 $D_s(u)$: 始端側 1 階 CBD 制御曲線、 u は区間 $[a_v, a_m]$, v は区間 $[b_v, b_m]$ として、

$$T_1(u,v) = [U_0(u), D_s(u), U_1(u)] \begin{bmatrix} \lambda_0(v) \\ \lambda_1(v) \\ \lambda_2(v) \end{bmatrix} \quad (38)$$

$$T_2(u, v) = [\mu_0(u), \mu_1(u)] \begin{bmatrix} V_0(v) \\ V_1(v) \end{bmatrix} \quad (39)$$

$$T_3(u, v) = [\mu_0(u), \mu_1(u)] \begin{bmatrix} U_0(a_0), D_s(a_0), U_1(a_0) \\ U_0(a_m), D_s(a_m), U_1(a_m) \end{bmatrix} \begin{bmatrix} \lambda_0(v) \\ \lambda_1(v) \\ \lambda_2(v) \end{bmatrix} \quad (40)$$

で定義され、 $\frac{\partial^2 T}{\partial v^2}(u, b_0)$ を求めることによって、2階 CBD 関数を求める。

また、終端リブにおいても同様に2階 CBD 制御曲線を算出するが、このとき終端方向から微分する。

4.2.3. G0 Gordon 面の作成

R 止まり線、リブ列、端リブ沿い1階2階 CBD 関数から、サポート曲面との連続性を考えないで、Gordon面を張る⁽⁷⁾。このとき、V 方向の補間関数は、5次 C4 スプライン関数とする。

$$\tilde{T}(u, v) = \tilde{T}_1(u, v) + \tilde{T}_2(u, v) - \tilde{T}_3(u, v) \quad (41)$$

ただし、 $D_s(u)$ は終端側1階 CBD 制御曲線、 $E_s(u)$ は始端側2階 CBD 制御曲線、 $E_e(u)$ は終端2階 CBD 制御曲線として、

$$\tilde{T}_1(u, v) = [U_0(u), D_s(u), E_s(u), U_1(u), U_2(u), \dots, U_{n-2}(u), U_{n-1}(u), E_e(u), D_e(u), U_n(u)] \begin{bmatrix} \lambda_0(v) \\ \lambda_1(v) \\ \vdots \\ \lambda_{n+3}(v) \end{bmatrix} \quad (42)$$

$$\tilde{T}_2(u, b_0) = [\mu_0(u), \mu_1(u)] \begin{bmatrix} V_0(v) \\ V_1(v) \end{bmatrix} \quad (43)$$

$$\tilde{T}_3(u, v) = [\mu_0(u), \mu_1(u)] \begin{bmatrix} U_0(a_0), D_s(a_0), E_s(a_0), U_1(a_0), U_2(a_0), \dots, U_{n-2}(a_0), U_{n-1}(a_0), E_s(a_0), D_s(a_0), U_n(a_0) \\ U_0(a_m), D_s(a_m), E_s(a_m), U_1(a_m), U_2(a_m), \dots, U_{n-2}(a_m), U_{n-1}(a_m), E_s(a_m), D_s(a_m), U_n(a_m) \end{bmatrix} \begin{bmatrix} \lambda_0(v) \\ \lambda_1(v) \\ \vdots \\ \lambda_{n+3}(v) \end{bmatrix} \quad (44)$$

である。

4.2.4. G0 Gordon 面より、各内部リブ沿い1階 CBD の決定

G0 Gordon 面から、各内部リブにおいて1階 CBD 制御曲線を取り出し、サポート曲面との G2 接続性を確保するための大域変形をおこなう。

4.2.5. G1 Gordon 面の作成

上述における補正済みの各内部リブ沿い1階 CBD 関数と、R 止まり線、リブ列、端リブ沿い1階、2階 CBD 関数から、5次 C3 補間を使って Gordon 面を張る。この面を G1 Gordon 面とよび、 $\tilde{\tilde{T}}(u, v)$ であらわし、これを出力曲面とする。

$$\tilde{\tilde{T}}(u, v) = \tilde{\tilde{T}}_1(u, v) + \tilde{\tilde{T}}_2(u, v) - \tilde{\tilde{T}}_3(u, v) \quad (45)$$

ただし、 $D_i(u)$ は内部リブ沿い1階 CBD 制御曲線として、

$$\tilde{\tilde{T}}_1(u, v) = [U_0(u), D_i(u), E_i(u), U_1(u), D_i(u), U_2(u), D_i(u), \dots, U_{n-2}(u), D_{n-1}(u), U_{n-1}(u), E_i(u), D_i(u), U_n(u)] \begin{bmatrix} \lambda_0(v) \\ \lambda_1(v) \\ \vdots \\ \lambda_{2n+3}(v) \end{bmatrix} \quad (46)$$

$$\tilde{\tilde{T}}_2(u, b_0) = [\mu_0(u), \mu_1(u)] \begin{bmatrix} V_{10}(v) \\ V_{11}(v) \end{bmatrix} \quad (47)$$

$$\tilde{\tilde{T}}_3(u, v) = [\mu_0(u), \mu_1(u)] \begin{bmatrix} U_0(a_0), D_s(a_0), E_s(a_0), U_1(a_0), D_s(a_0), U_2(a_0), D_s(a_0), \dots, U_{n-2}(a_0), D_{n-1}(a_0), U_{n-1}(a_0), E_s(a_0), D_s(a_0), U_n(a_0) \\ U_0(a_m), D_s(a_m), E_s(a_m), U_1(a_m), D_s(a_m), U_2(a_m), D_s(a_m), \dots, U_{n-2}(a_m), D_{n-1}(a_m), U_{n-1}(a_m), E_s(a_m), D_s(a_m), U_n(a_m) \end{bmatrix} \begin{bmatrix} \lambda_0(v) \\ \lambda_1(v) \\ \vdots \\ \lambda_{2n+3}(v) \end{bmatrix} \quad (48)$$

である。

5. グローバルデフォメーション技術

高品質なスタイリングデザインの追及によってクレイモデルの更新が頻繁になると、ClassA サーフェスを最初から作成し直す従来の方法ではデータの更新が追いつかない。ClassA データ作成の作業を効率化するために、既存の ClassA サーフェスを変形、修正するアプローチに着目した。

従来の曲面修正手法で目標の形状を得るには、まずすべての単曲面で数学パラメータの調節を繰り返して意図する形状を表現し、次にそれらが曲率連続で接続するように曲面同士を微調整する必要がある、時間のかかる作業となっていた。本件システムではデザイナーの審美的視点で形状目標を直接入力することを曲面形状の修正手段とすることで、より直感的で効率のよいデータ修正作業を可能とした。これをターゲットドリブンモデリングと呼ぶ。さらに複数の曲面を一度に入力しながらも、既存の連続性を維持することで出力曲面群の連続性を保証する技術を開発した。これらの技術例としてハイライトコントロールと曲率コントロールを紹介する。

5.1. ハイライトコントロール

ハイライトコントロールとは、ClassA データの品質評価基準のひとつである2次元ハイライトにターゲットドリブンモデリングの考え方を応用したものである。本機能は既存の曲面群に対して目標とするハイライトを指定することで、目標のハイライトを得る形状に変形させる。Fig. 13は上段に入力曲面群とデザイナーの意図である目標のハイライトを示し、下段にその出力結果を示す。図中の青線は曲面群のハイライトを可視化したものであり、その中でも黄線で示したハイライトが、白線で指定した位置を通るように変形をおこなった。

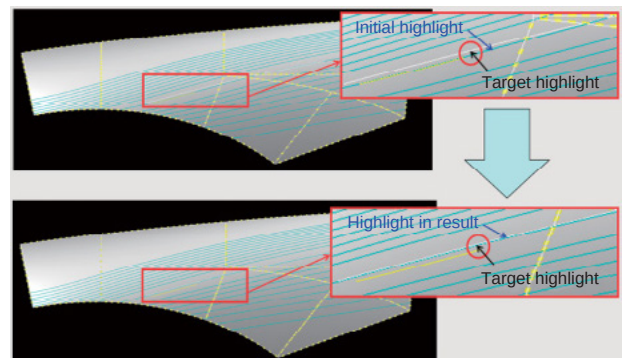


Fig. 13 Input and result of highlight control

5.1.1. 2次元ハイライト

Fig. 14は2次元ハイライトの定義を図示している。2次元ハイライトとは、曲面の意匠形状を評価するために曲面上に発生させるハイライト線である。評価したい曲面に対して一定方向の平面群を定義し、その交線を生成する。生成された交線の接線ベクトルが同じ向きになる点を抽出し、それらを結んで得られる曲面上の曲線として定義する。

5.1.2. ハイライトコントロール手法

ハイライトとは前述のとおり曲面上の同じ接線方向を持つ点を結んだものである。そのため、曲面のハイライトをコントロールするということは、既存のハイライト線から目標のハイライト線へ接線ベクトルを維持しながら移動させることを意味する。しかし、曲面周端部で既存の連続性を固

定しつつハイライト線での接線ベクトルを維持した場合、Fig. 15のように、曲面操作の条件過多により目標のハイライト周辺の曲面形状に逆向きの変化が生じる。

この課題を解決するため、ハイライト線での接線ベクトルを維持するという基本構想に加えて、以下に示す曲面変形の自由度を与えた。

- (1)接線ベクトルは維持しつつ、曲面の法線方向に自由度を持たせる。
- (2)ハイライト操作の視線方向に対して、位置的自由度を持たせる。

これらを Fig. 16 に図示する。本機能で入力する目標ハイライト線は、曲面形状の傾向変移を指定するものであるため、上記の自由度は目標としたハイライトの実現を可能

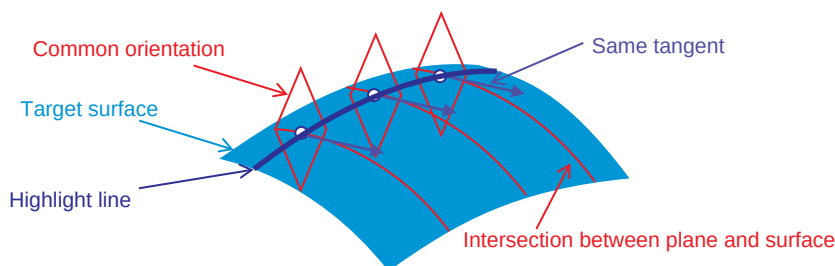


Fig. 14 Definition of 2D highlight

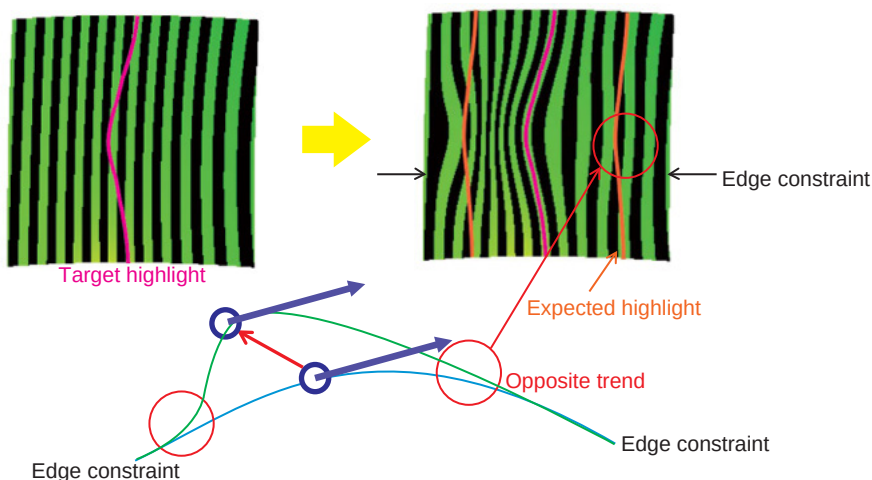


Fig. 15 Issue from over-constraints

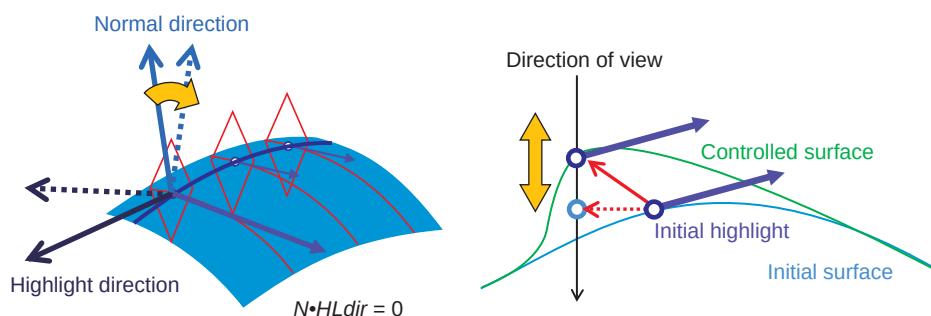


Fig. 16 With given freedom

とし、かつ悪影響を及ぼさない。以上の手法により、ハイライトコントロールにおける技術的課題を解決した。

5.2. 曲率コントロール

曲率コントロールとは、既存の曲線に対して曲率プロットの目標線を入力することで既存曲線の形状を変形する機能である。従来システムでは、目標の曲線を得るために、曲線のコントロールポイントや通過点の位置をひとつずつ繰り返し調整していた。本機能は、ターゲットドリブンの考えに基づき、Fig. 17 に示すように既存の曲線 C と目標の曲率プロット TF を入力することで、目標の曲率プロットを得る曲線形状へダイレクトに変形させることを目的とする。

5.2.1. 曲率コントロール手法

Fig. 17 中の曲線 C 上の任意の点 P において、目標の曲率プロット TF を満たす曲率 κ の値は、点 P の法線ベクトルと TF との交点を点 P_f としたとき、

$$\kappa = (P - P_f) / \text{曲率描画スケール} \quad (49)$$

として表現される。これを曲線 C のパラメータ s を用いて曲線全体で表現すると、

$$\kappa(s) = (C(s) - TF) / \text{曲率描画スケール} \quad (50)$$

となる。このとき、目標の曲率を満たす結果の曲線 CT は、既存の曲線 C の始点位置を P_0 、始点ベクトルを T_0 とした

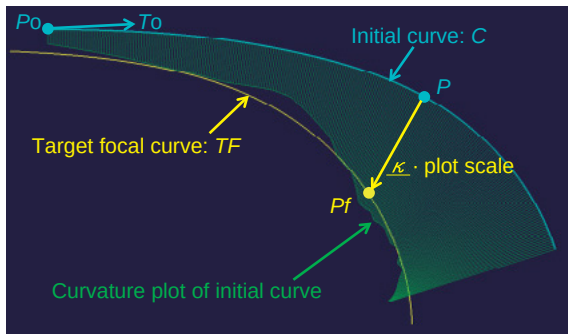


Fig. 17 Input for curvature control

とき、

$$\theta(s) = \text{Angle}[T_0] + \int_{[0,s]} \kappa(u) \frac{du}{ds} ds \quad (51)$$

$$CT(s) = P_0 + \int_{[0,s]} \{\cos[\theta(u)], \sin[\theta(u)]\} \frac{du}{ds} ds \quad (52)$$

として求めることができる。

5.2.2. 始終点の拘束条件

しかし、式(52)では結果として得られる曲線 CT の終点の位置を指定することができない。Class A モデリングでは、既存の曲線の始終点の位置だけでなく、そのベクトルや曲率を維持することが重要となる。

例えば、得られる曲線 CT の曲率を κ_c とすると、始終点のベクトルを維持するためには、

$$\int_0^u \kappa_c(u) du = \int_0^u \kappa(u) du \quad (53)$$

を満たす必要がある。さらに、始終点の曲率を維持するためには、

$$\kappa_c(u_a) = \kappa(u_a) \quad (54)$$

$$\kappa_c(u_b) = \kappa(u_b) \quad (55)$$

を満たす必要がある。Fig. 18 は式(53)から式(55)を概念的に表している。既存の曲線の始終点のベクトルが維持されるためには、曲線の位置パラメータ $u[u_a, u_b]$ とその位置での曲率 $\kappa(u)$ で、既存の曲線と結果の曲線ではその積分された面積が同じになる。

しかし、目標の曲率プロット TF はユーザが描画し入力するため、前述の拘束条件を完全に満たすことは難しい。そのため、式(49)より得られる曲率 $\kappa(u)$ には、式(53)から式(55)の拘束条件を満たすような補正が必要である。本件システムではシステムが動的に定めるスカラー関数 $\varphi(u)$ を用いて、

$$\kappa_m(u) = \kappa(u) \times \varphi(u) \quad (56)$$

とする補正をおこない、式(49)より得られる曲率 $\kappa_m(u)$ に置

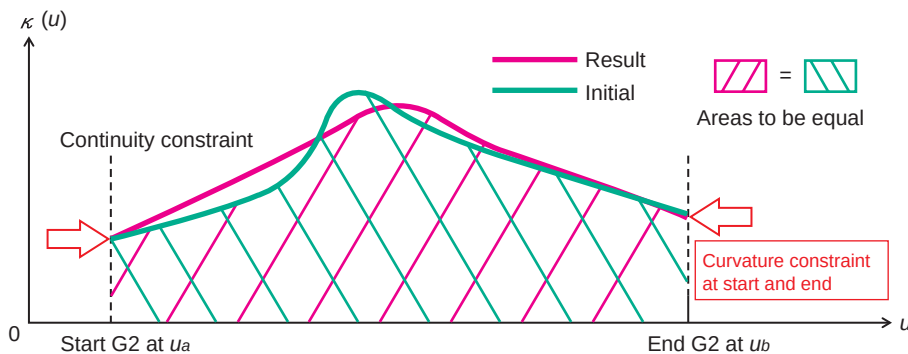


Fig. 18 Constraint of integrated area

き換えた。これより、始終点の拘束条件を満たすことができ、ClassA モデリングに要求される形状連続での曲率コントロールを実現した。

6. 新技術の効果

6.1. ポリゴン処理技術

本件システムのポリゴン処理技術において達成された内容は以下のとおりである。

- (1) ハーフエッジデータ構造を利用した効率的なポリラインセクション計算を実現した。計算結果を Fig. 19 に示す。
- (2) 投影曲線をセグメントに分割することでメッシュ上への正確な投影計算を可能とした。計算結果を Fig. 20 に示す。
- (3) 独自に考案した初期パッチの効率的な生成手法による、ロバストで高速な自動穴埋め処理を実現した。計算過程と結果を Fig. 3 に示した。
- (4) メッシュのフェイスセットを高速に空間分割する独自の交錯計算手法により、効率的なポリゴンメッシュ - 幾何形状間の離れ距離計算を実現した。Fig. 21は計算結果を可視化したものを示す。

6.2. フィッティング技術

フィッティング技術によって達成された内容を整理する⁽¹⁾。

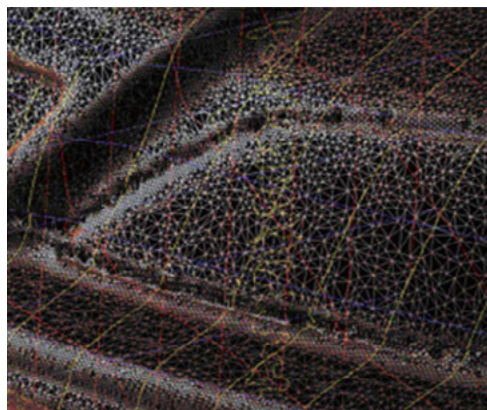


Fig. 19 Creation of polyline-section on scanned data

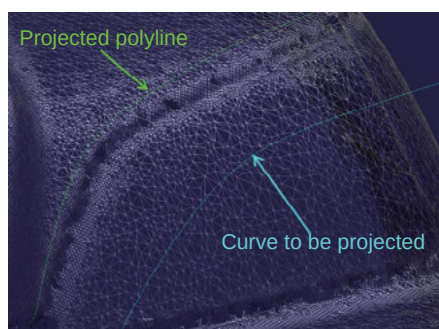


Fig. 20 Projection curve on scanned data

Table 1 Fillet performance

	Previous system	New system
Sample of fillet 1	27 min.	15 sec.
Sample of fillet 2 (Fig. 22)	52 sec.	5 sec.
Sample of fillet 3	34 sec.	3 sec.

- (1) インテリジェントフィッティング技術により、単面のフィッティングサーフェスの形状品質を確保できた。
- (2) フィッティング技術を応用したマージ技術により、複数のフィッティングサーフェス間の連続性を保障した接続を可能とした。

6.3. フィレット技術

本件システムのフィレット技術では計算処理の高速化と出力形状の高精度化を同時に達成した⁽¹⁾。

- (1) リブ導出計算のアルゴリズムを刷新することで計算処理速度を向上させることができた。Table 1でフィレット作成に要する時間を従来システムと比較した。すべてのサンプルで 90%以上の高速化を確認できた。Fig. 22 はもっとも複雑な形状サンプルである。
- (2) フィレットサーフェス作成の最終処理においては、段階的 Gordon 面張りを適用することでサポートサーフェスと G2 接続するフィレット面の作成に成功した。また、段階的 Gordon 面張りにより、フィレット面自

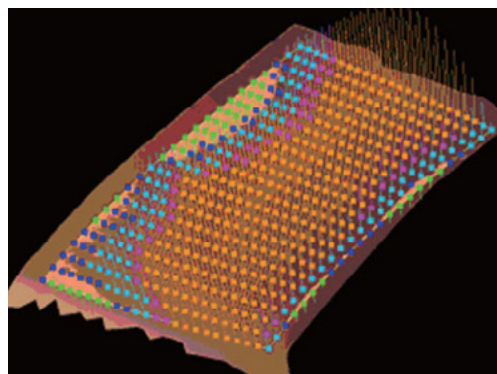


Fig. 21 Mesh-surface distance map

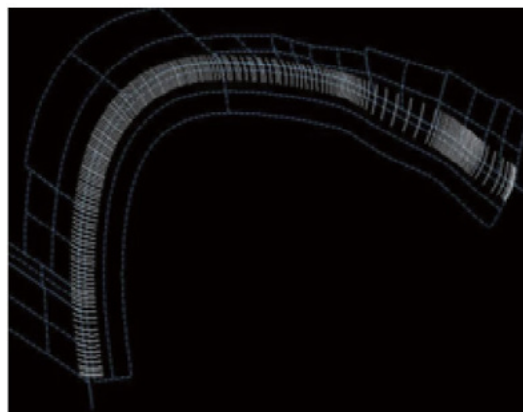


Fig. 22 Sample of fillet 2

体の形状品質を向上させた。Fig. 23は新旧のシステムで同一の入力から得られたフィレット形状のガウス曲率を可視化したものである。従来システムのフィレット面ではガウス曲率の急激な変化が確認できるが、新システムでは高い形状品質を確保している。

6.4. グローバルデフォメーション技術

ハイライトコントロールと曲率コントロールは、グローバルデフォメーション技術の適用で TDM を実現した一例である⁽¹⁾。

(1)ハイライトコントロール機能では、複数の曲面で構成される意匠形状に対して目標とするハイライトの指示をするだけで、短時間で目標形状への修正を可能にした。Fig. 13 のケースでは従来の修正作業には 60 min. を要していたが、本件システムではこれを43 sec.に短縮した。

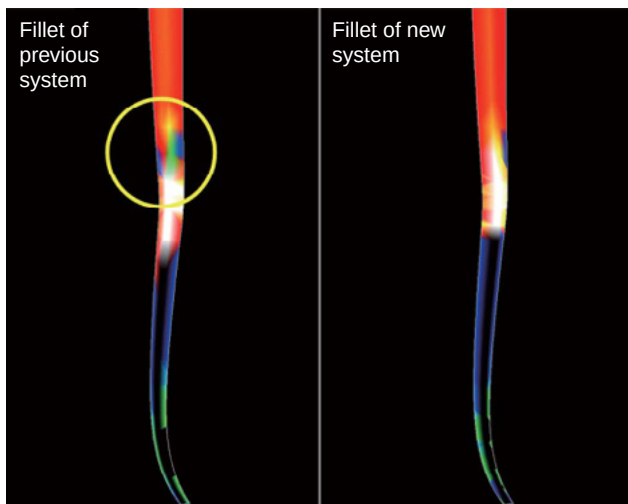


Fig. 23 Visualization of gauss curvature

(2)曲率コントロール機能では、既存の曲線に対して曲率プロットの目標線、および始終点の連続性を指示するだけで曲線の変形操作を可能とした。Fig. 24は同一の入力曲線と目標の曲率プロットに対して、始終点の連続性条件だけを変えた出力結果を示す。本技術により従来手法に要した時間比で最大 17%までに短縮した。

7. まとめ

ClassA サーフェス作成工数 50%削減の目標を実現させるために、ClassA サーフェス完成までの三つのモデリングフェーズに着目し、その作業効率を格段に向上できるCAD機能の実現を目指した。

- (1)クレイモデルの測定データからフィッティング面を作成するフェーズにおいて、独自のポリゴンワークベンチによって大容量の測定ポリゴンデータの加工をロバストかつ高速に可能とすることで、測定データへの曲面フィッティング作業の基盤を構築した。そのうえで曲面フィッティング機能では、ClassA サーフェスの基本曲面となる品質要件を満たしつつ、測定データとの精度を維持した曲面生成を実現し、ClassA データ作成の初期段階での効率化に成功した。
- (2)ClassA サーフェス作成機能の高速、高精度化においては、使用頻度が高いフィレット機能で独自のアルゴリズムにより計算処理の高速化と高精度化を実現するなど、ClassA データ作成の中核機能における効率化とデータ品質の向上を実現した。
- (3)ClassA データ修正フェーズにおいては、新しいコンセプトである TDM の具現化に成功し、意匠傾向の調整やデザイン変更への対応など、手間がかかる作業工程で効率化を実現した。

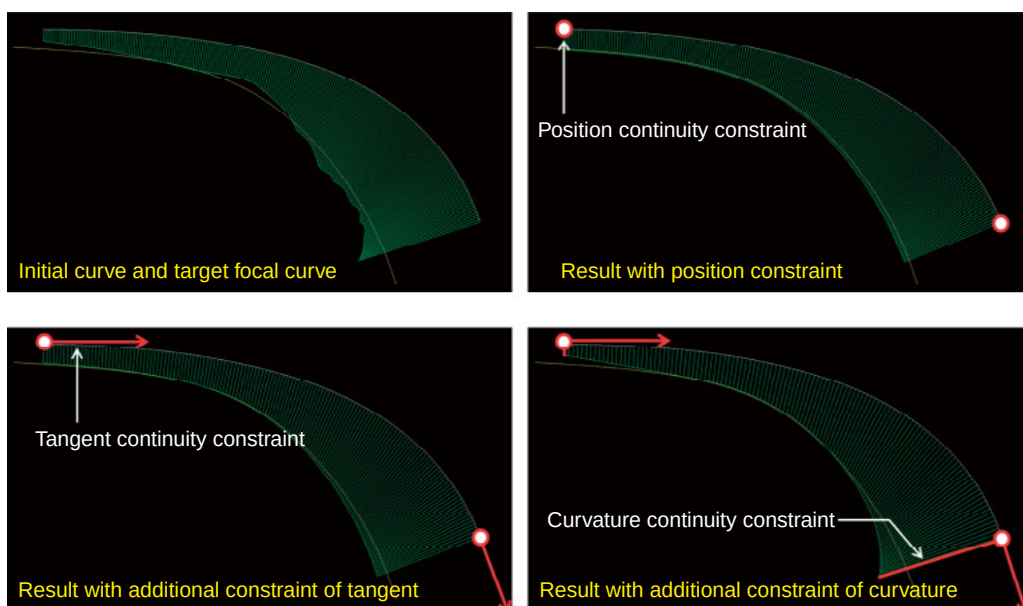


Fig. 24 Input and results of curvature control with various continuity constraints

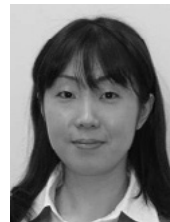
参考文献

- (1) 篠原智義, 春原慶子, 坂居三郎, 勝部弘嗣, 井出大介, 原宗生, 金田健: ClassA スタイリングデータ作成 CAD システムの開発, Honda R&D Technical Review, Vol.22, No.1, p.88-98
- (2) Liepa, P.: Filling Holes in Meshes., Proc. Eurographics/ACM SIGGRAPH Symp. Geometry Processing, p.200-205 (2003)
- (3) Botsch, M., Kobbelt, L.: A Remeshing Approach to Multiresolution Modeling, Proc. Eurographics/ACM SIGGRAPH Symp. Geometry Processing, p.189-196(2004)
- (4) Edward, M. G., Stephan, J. W.: Object-oriented software for quadratic programming, ACM Transactions on Mathematical Software, Vol.29, Issue 1, p.58-81(2003)
- (5) Gerald, E. F.: Curves and Surfaces for CAGD, The Frenet Frame, p.191-194(1993)
- (6) 清水保弘: 意匠面創成のための曲率単調曲線生成技術, UNISYS 技報, Vol.21, No.3, (No.71), p.85-104(2001)
- (7) 清水保弘, 関戸勝己: 周辺面群との接続機能付きメッシュ面創成コマンドの開発, UNISYS 技報, Vol.19, No.4, (No.64), p.1-26(2000)

■ 著 者 ■



篠原 智義



春原 慶子



坂居 三郎



勝部 弘嗣



井出 大介