Vol.2022-CG-185 No.8
2022/3/12

3次元メッシュの特異点を与えた沿ベクトル場パラメータ化

野間 裕太[1,a)]　川原 圭博[1,b)]

概要：幾何形状処理において，3 次元メッシュ上にスカラー関数を定義する「パラメータ化」と呼ばれる手法は近年盛んに研究されている．特に，入力されたベクトル場に対し，勾配ベクトルがなるべくベクトル場と等しくなるようなスカラー関数を定義する手法は「沿ベクトル場パラメータ化 (Field-Aligned Parameterization)」と呼ばれ，テクスチャマッピングや多角形メッシュの生成といった多様な応用が存在する．既存手法では，スカラー関数の等高線が分岐する「特異点」が大量に出現することを許すことで，パラメータ化の高速な計算を実現してきた．しかし，特異点は最終形状の美観や使われ方に関わる重要な要素である一方で，既存手法では特異点の発生を制御することはできず，また特異点に対するインタラクティブな操作を十分に高速に行うことができなかった．そこで本稿では，与えた点のみが特異点となりそれ以外は特異点とならないように指定して，沿ベクトル場パラメータ化を高速に計算できる手法を提案する．

1. はじめに

近年幾何形状処理 (Geometry Processing) の技術は，離散微分幾何 (Discrete Differential Geometry) の発展を背景に大いに発展している．その中でも特に，3 次元メッシュ上に定義された接ベクトル場に対し，入力されたベクトル場に対し，勾配ベクトルがなるべくベクトル場と等しくなるようなスカラー関数を定義する手法は「沿ベクトル場パラメータ化 (Field-Aligned Parameterization)」と呼ばれる．このようなスカラー関数は，直感的には**ベクトル場に沿って動くたびにスカラー関数が変化する**ようなものとして理解できる．例えば，図 1a のベクトル場は左から右に向かって流れているが，沿ベクトル場パラメータ化により計算されるスカラー関数の値は，図 1b のように左から右に向かって単調増加している．

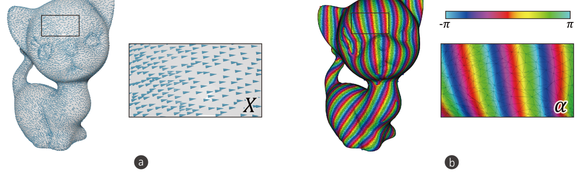

図 1　本研究の手法により生成した沿ベクトル場パラメータ化の例．(a) ベクトル場 X，(b) 沿ベクトル場パラメータ化のスカラー関数 α．

沿ベクトル場パラメータ化は形状の特徴に沿った縞模様を生成できることから，四角形メッシュの生成 [1] やテクスチャマッピング [2]，家具スケールの構造物 [3] など，多様な目的に応用されている．ここで問題になるのが「特異点」と呼ばれる，スカラー関数の等高線を終端したり，等高線の分岐を起こしたりする点が発生することがある (図 4 参照)．このような特異点の発生は，沿ベクトル場パラメータ化の各種応用例の美観や可用性に影響を及ぼす．例えば四角形メッシュの生成では，特異点の配置によっては美観に悪影響を及ぼす（ [4] 図 17 左参照)．また家具スケールの構造物の設計では，特異点付近の強度が弱くなってしまう [3]．このため，沿ベクトル場パラメータ化を用いた多様な形状をユーザが自由に設計できるようにするためには，特異点の発生数を減らしたり，特異点をユーザの好きな位置に移動させせられるアルゴリズムが求められている．特に，ユーザの試行錯誤による探索的な設計プロセスの支援を考えれば，アルゴリズムがインタラクティブに動作するレベルに高速であることが重要である．

しかし既存の沿ベクトル場パラメータ化手法では，特異点の出現数と計算量の多さ・実装の複雑さはトレードオフの関係にあった；沿ベクトル場パラメータ化を高速に計算できる手法 [2], [4] は特異点が大量に出現してしまう一方で，特異点の発生数を抑えられる手法 [1] や特異点の位置を指定できる手法 [5] はいずれも，重く反復的な逆行列計算や実装の複雑な離散アルゴリズムを要する．このため，特異点位置のインタラクティブな編集は既存研究では実質的に不可能であった．

そこで本研究では，**与えた点のみが特異点となりそれ以外は特異点とならないような沿ベクトル場パラメータ化**を

1　東京大学
a)　noma@akg.t.u-tokyo.ac.jp
b)　kawahara@akg.t.u-tokyo.ac.jp

インタラクティブな操作が可能なレベルで十分に高速に行える手法があれば，特異点の発生の防止とインタラクティビティを担保できる高速な計算とを両立できると考えた．これにより，少ない数の特異点を指定すれば特異点の数を抑えたパラメータ化が可能になり，また特異点の数が多かったとしてもユーザが自由に特異点位置を編集できる．

本研究では，特異点の位置を指定し，それ以外は特異点とならないような沿ベクトル場パラメータ化のアルゴリズムを実装した．本研究の最大の特長は，パラメータ化の計算が逆行列計算 1 回分に帰着でき，さらに一度この逆行列計算を計算しておけば再計算が不要な点である．このことから，本研究のアルゴリズムは特異点の移動や対消滅といった操作を，インタラクティブかつ高速に行うことができる．

2. 関連研究

沿ベクトル場パラメータ化はその有用性から，これまで多くの研究者により研究されてきた．沿ベクトル場パラメータ化では，入力されたベクトル場 X がなるべく $\nabla\alpha$ に等しくなるようなスカラー関数 $\alpha \in \mathbb{R}$ を求める．本研究を含め，既存研究では以下の L^2 ノルム ε の最小化問題として定義されてきた：

$$\varepsilon = \int_{\mathcal{M}} |\nabla\alpha - X|^2 dA \tag{1}$$

既存研究は，(1) α を実数で表す方法 [4], [5], [6] と (2) α を $\alpha \in [-\pi, \pi)$ なる角度値で表す方法 [1], [2], [7] の 2 種類に分類できる．

(1) の実数による表現は，すべての特異点を通るような切り込みを形状表面に入れてディスク同相にした上で，形状表面に実数パラメータ α を配置する方法である．この切り込みは特異点グラフ (Singularity Graph) と呼ばれ，Bommes et al. 2009 [6] によりこの特異点グラフを自動で生成するアルゴリズムが提案された．この方法は形状表面に (u, v) の座標を配置する UV 展開と非常に相性が良く，テクスチャマッピングにおいて頻繁に用いられる表現である．特に Tong et al. 2006 [5] は，指定した箇所にのみ特異点を出現させそれ以外では特異点を出現させないようにするパラメータ化を実現しており，本研究の手法と類似している．

このような切り込みを作る際に注意しなければならないのは，切り込みを跨ぐ時に整数値を指定する必要があることである．形状全体で連続 (Globally Continous) なパラメータ化を実現するためには，特異点グラフの両側でスカラー関数の値を 1 で割った余りが等しくなる必要があり，両側の値の差は必ず整数値でなければならない．しかし，[5] の Integer Rounding や [6] の混合整数ソルバーは準最適な一方で，真に最適な解を求めるためには計算量の膨大な混合整数最適化を解く必要がある．さらに，これらの整数値は特異点グラフに変更が加えられるたびに変更しなければならず，ソルバーの再計算が必要になる．これらの理由から，特異点の位置を変更する操作のたびに重い計算を行わなければならず，特異点のインタラクティブな操作には不向きである．

なお，実数による表現を利用した手法の内，本研究と特に似ている手法として Jakob et al. 2015 [4] が挙げられる．この手法は特異点グラフの代わりに各面の上にローカルなスカラー関数を配置し各面同士の接続関係を最適化することで，実数による表現を利用しながらも，沿ベクトル場パラメータ化の計算をインタラクティブな操作を許すレベルで高速に行うことができる．一方で，本研究とは以下に挙げる違いがある：

- Jakob et al. 2015 は独自実装による Randomized Gauss-Seidel ソルバーを用いている一方，本研究は一般的な逆行列計算ソルバーを用いて計算できる．
- Jakob et al. 2015 は非線形かつ非凸な最適化を行っており，局所最適解に陥る恐れがあるうえそれを防ぐためには複雑な離散アルゴリズムが必要だったが，本研究では凸最適化を行っているためその恐れはない．
- Jakob et al. 2015 は特異点の移動の際に別の箇所に特異点が発生してしまう可能性があるが，本研究では各面が特異点となるかどうかを指定しているためそのようなことは起こりえない．
- Jakob et al. 2015 では大量の特異点が出現し，その特異点の消去も手動で行う必要があるが，本研究ではある点が特異点となるかどうかを事前に指定できる．

一方で (2) の角度値による表現では，上記の実数による表現と異なり，特異点グラフのような切り込みを入れる必要がない．さらに Knöppel et al. 2015 [2] は，$\varphi = e^{\imath\alpha}$ なる複素数による表現を利用することで，最適化関数が凸でかつ高速な特異点の自動配置を実現している．

しかし，Knöppel et al. 2015 [2] は高速な動作と引き換えに，特異点を大量に発生させてしまう問題があった．一方，角度値による表現を採用した手法の中で，特異点の発生を抑える手法はすべて Ginzburg-Landau エネルギー [1] や Principal Matching [7] といった，非凸なうえに計算量の多い最適化を利用している．

本研究では，上記に挙げたどの手法とも異なり，沿ベクトル場パラメータ化にまつわる高速でインタラクティブな操作を，最も簡単な実装により実現している．

3. 連続的な表現

沿ベクトル場パラメータ化は，与えられた曲面 $\mathcal{M}$ とその上に定義されたベクトル場 X に対し，X に沿ったスカラー関数（0-形式）α を求めることである．この時，α の勾配ベクトル $\nabla\alpha$ がなるべく X と一致していれば，α は

X に沿っているといえる．この定義から，求めるスカラー場 α の等高線は X とほぼ直交している（図 1）．

既存の沿ベクトル場パラメータ化手法はすべて，エネルギー関数を 0-形式 α を用いて表し，α を直接エネルギー関数の最適化により求めていた．しかし，α を直接求める方法では，α の等高線が発生する特異点の発生を抑えることができない．そこで本研究では，エネルギー関数を α の外微分 $d\alpha$ を用いて表し，まず $d\alpha$ を計算してから α を計算するアプローチを取る．

本章では，本研究で採用した $d\alpha$ を用いた表現により，エネルギーと特異点の制約を表す．

3.1 表現

形状上のあらゆる箇所で連続なパラメータ化を実現するために重要なのは，適切なパラメータの表現を選ぶことである．本研究では α の範囲を $\alpha \in [-\pi, \pi)$ に制限する表現を採用した．

この角度による表現は [1], [2] でも用いられている一方で，これらの手法では $\alpha \in [-\pi, \pi)$ を直接最適化するのではなく，$\psi = ae^{\imath\alpha}$ なる ψ によりエネルギー関数を表し，まず最適化で ψ を求めてから $\alpha = \arg\psi$ により α を得ている．これは $\alpha \in [-\pi, \pi)$ が $-\pi$ や π 付近の主値の境界で不連続であり，数理最適化において不都合があるからである（図 2 参照）．

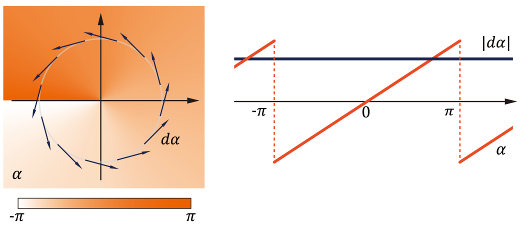

図 2　$\alpha \in [-\pi, \pi)$ と $d\alpha$ の関係．

しかし，α は $-\pi$ や π 付近で必ず不連続になる一方で，α の微分形式である $d\alpha$ は $-\pi$ や π 付近で連続だと仮定できる（図 2）．本研究では，例え α が主値の境界を跨いでも，その微分値である $d\alpha$ は連続であると仮定する．この仮定の下で，$d\alpha$ によりエネルギー関数や制約条件を表し，連続的な最適化を試みる．

3.2 特異点

次に，連続な微分幾何における特異点の特徴を明らかにする．

ここで例として，図 3 に示すような $\mathbb{R}^2$ の場に定義されたスカラー関数 $\beta_0 : \mathbb{R}^2 \longmapsto [-\pi, \pi)$ を考える．簡単のため，このスカラー関数は等高線が分岐したり生起したりするような特異点は存在しないものとする．この場の上に反

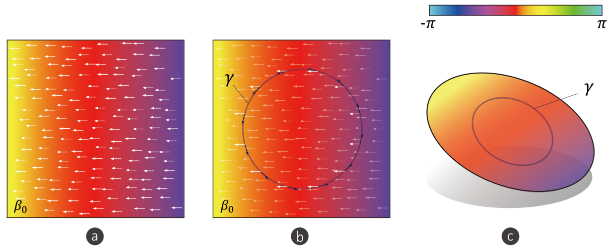

図 3　(a) 特異点のないスカラー関数 β_0 とその勾配ベクトル，(b) β_0 の上に定義した γ，(c) α を立体的に図示したもの．

時計回りの閉曲線 γ を定義すると，以下が成り立つ：

$$\oint_\gamma \nabla\beta_0 \cdot ds = 0 \tag{2}$$

$$\Leftrightarrow \oint_\gamma d\beta_0(ds) \ = 0 \tag{3}$$

これは，閉曲線に沿って移動した時に，スカラー関数の勾配ベクトルに沿って値を変化させていくと，スタート地点に戻ってきた時に値が同じであることを表している（図 3c 参照）．この「閉経路の始点と終点での値が一致する」性質はホロノミー性と呼ばれ，スカラー関数が連続であるためには必ず満たされなければならない．

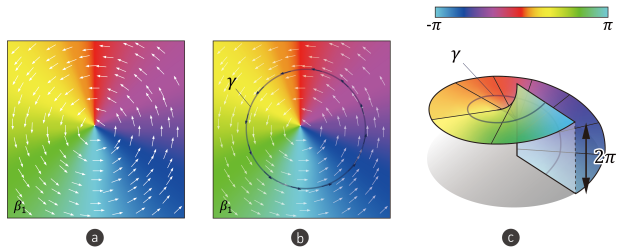

図 4　(a) 特異点のあるスカラー関数 α とその勾配ベクトル，(b) α の上に定義した γ，(c) α を立体的に図示したもの．

次に，図 4 に示すような $\mathbb{R}^2$ の場に定義された，等高線が中心付近から一本生起しているような特異点を持つスカラー関数 $\beta_1 : \mathbb{R}^2 \longmapsto [-\pi, \pi)$ を考える．この場の上に，特異点を囲むような反時計回りの閉曲線 γ を定義すると，以下が成り立つ：

$$\oint_\gamma \nabla\beta_1 \cdot ds = 2\pi \tag{4}$$

$$\Leftrightarrow \oint_\gamma d\beta_1(ds) \ = 2\pi \tag{5}$$

これは，閉曲線に沿って移動した時に，スカラー場の勾配ベクトルに沿って値を変化させていくと，スタート地点に戻ってきた時に値が元の値より 2π 大きくなることを表している（図 4c 参照）．この場合も，β_1 が角度値であることを考えれば，同様にホロノミー性を満たしていると言

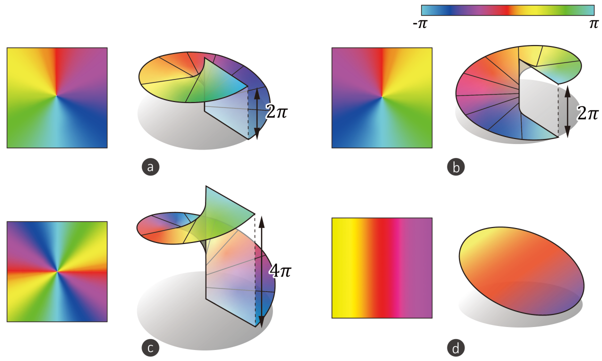

図 5　特異点の指数の例．(a) +1，(b) -1，(c) +2，(d) 0.

える．

本研究では，$\oint_\gamma d\alpha(ds) = 2n\pi (n \in \mathbb{Z})$ とした時に，n を指数 (index) と呼ぶ．図 5 に示すように，指数はその点から新たに生起する等高線の本数を表す．また非特異点を囲む閉曲線における指数は 0 である．

以上をまとめると，$\mathcal{M}$ 上の任意の点を囲む反時計回りの閉曲線 γ に関して以下が成り立つ：

$$\oint_\gamma d\alpha(ds) = \begin{cases} 0 & (\gamma\text{が特異点を囲まない場合}) \\ 2n\pi & (\quad \text{〃} \quad \text{囲む場合}, n \in \mathbb{Z}) \end{cases} \tag{6}$$

3.3　エネルギー

$\nabla\phi = (d\phi)^\sharp$ [8] より，式 1 のエネルギー関数は，$d\alpha$ を用いて以下のように書き換えられる：

$$\begin{aligned} \varepsilon(\alpha) &= \int_\mathcal{M} |d\alpha^\sharp - X|^2 dA \\ &= \int_\mathcal{M} |(d\alpha^\sharp - X)^\flat|^2 dA \\ &= \int_\mathcal{M} |d\alpha - X^\flat|^2 dA \end{aligned} \tag{7}$$

4.　離散的な表現

次に，3 章で述べた連続的な微分幾何における特異点の表現を，離散的な三角形メッシュ上で表現する方法を述べる．

本研究では，入力として三角形メッシュの 2-単体複体 K と，K の上で定義された離散的なベクトル場 $\hat{X}$ を受け取る．以下，各辺と各面は含まれる点の番号により記述する；例えば $ijk \in F$ は点 i, j, k を要素に持つ面を，$ij \in E$ は点 i, j を要素に持つ辺を表す．なお，本稿で用いる表記は，[2] の 3 章の冒頭で説明されているものと同じものである．

4.1　離散的なベクトル場の表現

本研究の離散的なベクトル場の表現には，[2] と同じものを用いた．

離散的なベクトル場 $\hat{X}$ は，各点 $i \in V$ において，基準辺 $ij_0 \in E$ からの角度 ϕ_i とノルム ν_i の 2 つのパラメータで定義される．この時，特に ϕ_i については，点 i に隣接する面の角の合計が 2π となるよう，実際のユークリッド角度にある値をかける．具体的には，点 i に隣接する三角形 ijk の i における頂点の角度を $\tilde{\theta}_i^{jk}$ とすると，

$$s_i := \frac{2\pi}{\sum_{ijk \ni i} \tilde{\theta}_i^{jk}} \tag{8}$$

なる値をおく（なお，メッシュの境界上にある点の場合は $s_i = 1$ とする）．この時 ϕ_i には，基準辺 $ij_0 \in E$ からのユークリッド角度に s_i をかけたものを使用する．

また，以下本章で扱う各面の角度についても，ユークリッド角度を s_i 倍したものを用いる．これにより，各頂点の周辺を平坦なものとして扱えるため，離散的な三次元形状でも接平面の概念を扱える．

4.2　離散的な微分形式の積分形式への変換

スカラー関数（0-形式）の微分形式は 1-形式である．離散的な状況では，0-形式 $\hat{\beta}$ は点ごとの値，1-形式 $\widehat{d\beta}$ は辺ごとの値として表される．この時，$\widehat{d\beta}$ は以下のように表される [8]：

$$\widehat{d\beta}_{ij} = \hat{\beta}_j - \hat{\beta}_i \tag{9}$$

ここで，α を離散化した値を $\hat{\alpha} \in [-\pi, \pi)^{|V|}$，$\alpha$ の微分形式 $d\alpha$ を離散化した値を $\widehat{d\alpha} \in \mathbb{R}^{|E|}$ とおく（本稿では以後，この 2 つの文字は断りなく用いる）．この時，式 9 を用いれば，辺 $ij \in E$ における $\widehat{d\alpha}$ の値 $\widehat{d\alpha}_{ij}$ と，$\hat{\alpha}$ の $i, j \in V$ における値 $\hat{\alpha}_i, \hat{\alpha}_j$ について以下が成り立つ：

$$\widehat{d\alpha}_{ij} = \hat{\alpha}_j - \hat{\alpha}_i + 2n\pi \; (n \in \mathbb{Z}) \tag{10}$$

このことから，$\hat{\alpha}_i$ と $\widehat{d\alpha}_{ij}$ が分かっている場合には，α_j を求めるには $\hat{\alpha}_i + \widehat{d\alpha}_{ij}$ を 2π で割った余りを求めればよいことになる．

4.3　離散的なエネルギー

次に，式 7 を $\widehat{d\alpha}$ を用いて表す．ここで離散的な 1-形式 $\widehat{d\alpha}$ は辺ごとに与えられた値として表せる [8] ことから，本研究ではまず辺ごとにエネルギーを求め，それを全ての辺に対して足し合わせる．本研究では式 7 を以下の L^2 ノルムとして離散化する：

$$\hat{\epsilon}(\alpha) := \sum_{ij \in E} w_{ij} |\widehat{d\alpha}_{ij} - \hat{X}^\flat|^2 \tag{11}$$

この式における w_{ij} はメッシュの切り方による差分を吸

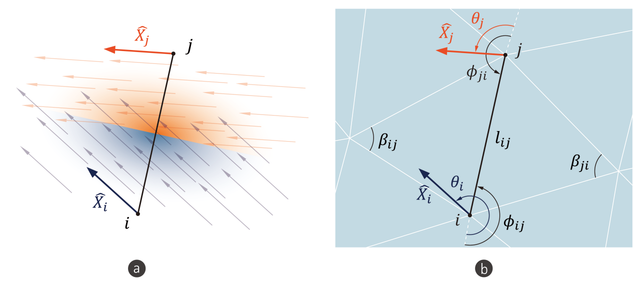

図 6　(a) 辺 ij における $\hat{X}$ の方向，(b) 辺 ij 周辺の様子．

収する Cotangent Weight と呼ばれる量で，離散微分幾何において頻繁に使用される量である．この値は辺 ij の 2 つの対角を β_{ij} と β_{ji} とすれば，$w_{ij} := (\cot\theta_{ij} + \cot\theta_{jk})/2$ とおける（図 6b 参照）．

ここで問題になるのは，$\hat{X}^\flat$ を入力された $\hat{X}$ を用いてどう表すかである．連続的な 1-形式を離散化するには，連続的な 1-形式の辺に沿った線積分を求めればよいことが知られている [8]．本研究では $ij \in E$ の中点を境に，$i \in V$ 側の区間におけるベクトル場は $\hat{X}_i$，$j \in V$ 側の区間におけるベクトル場は $\hat{X}_j$ であると仮定する（図 6a）．この時，$ij \in E$ の長さを l_{ij}，$\hat{X}_i, \hat{X}_j$ のベクトル場の基準辺からの角度を θ_i, θ_j，$i, j \in V$ における辺 $ij \in E$ の基準角からの角度をそれぞれ $\phi_{ij}, \phi_{ji} \in [-\pi, \pi)$ とおく（$\theta_i, \theta_j, \phi_{ij}, \phi_{ji}$ はそれぞれユークリッド角度に 4.1 項に述べた s_i, s_j をかけたものである）と，$\hat{X}^\flat$ は以下のように表せる：

$$\begin{aligned}\hat{X}^\flat &= \int_{ij} X^\flat \\ &= \int_{ij} X \cdot ds \\ &= \frac{1}{2} l_{ij}(|\hat{X}_i| \cos(\theta_i - \phi_{ij}) + |\hat{X}_j| \cos(\theta_j - \phi_{ji})) \\ &:= \omega_{ij}\end{aligned} \tag{12}$$

この値 ω_{ij} を用いれば，エネルギー関数は以下のように表せる：

$$\hat{\epsilon}(\alpha) = \sum_{ij \in E} w_{ij} |\widehat{d\alpha}_{ij} - \omega_{ij}|^2 \tag{13}$$

4.4　離散的な特異点制約

本研究の目標である「与えた点が特異点となり，それ以外が特異点とならない」沿ベクトル場パラメータ化を実現するためには， 3.2 項で述べた線積分の制約を離散化し，$\widehat{d\alpha}$ を用いて表す必要がある．

本研究では，ユーザは各 $ijk \in F$ 内の指数 $k_{ijk} \in \mathbf{Z}$ を指定できるようにする．もし三角形 $ijk \in F$ 内に指数 n の特異点が 1 つあれば $k_{ijk} = n$，逆に特異点がなければ $k_{ijk} = 0$ となる．この時，ijk の境界を反時計回りに回るパス γ_{ijk} を考えると，γ_{ijk} は 3 辺 $ij, jk, ki \in E$ をなぞるようなパスになる．この時，以下が成り立つ：

$$\oint_{\gamma_{ijk}} d\alpha(ds) = \sum_{e \in \{ij,jk,ki\} \subset E} \widehat{d\alpha}_e \tag{14}$$

この時，$\sum_{e \in \{ij,jk,ki\}} \widehat{d\alpha}_e = 2\pi k_{ijk}$ が満たされれば，指定した三角形内に指定した指数の特異点が現れる条件を満たすことになる．これがすべての面 $ijk \in F$ で満たされていれば，すべての面の境界におけるホロノミー性が満たされることになる．

4.5　ホロノミー制約

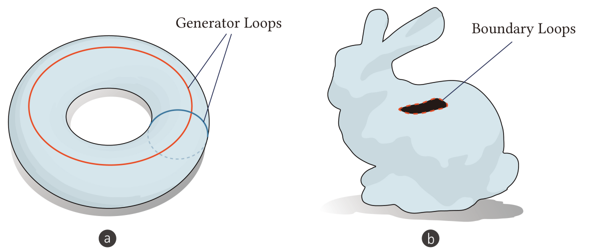

図 7　非可縮なループ．(a) 種数が 0 でない曲面における generator ループ，(b) 境界ループ．

曲面上に配置された閉曲線について，連続的な変形により最終的に点に変形できるものを可縮なループ，点に変形できないものを非可縮なループと呼ぶ（図 7 参照）．もし上記に述べた個々の三角形ループのホロノミー性がすべて満たされていれば可縮なパスにおけるホロノミー性は満たされる一方，非可縮なパスではホロノミー性が満たされるとは限らない（[9] A 参照）．そこで本研究では，非可縮なパスについてもホロノミー性が満たされるよう，非可縮なパスに沿った線積分も 2π の倍数となるように制約を課す．

グラフ構造における非可縮なパスは，(1) generator ループと (2) 境界ループの 2 種類がある．(1) の generator ループは，図 7a に示すような，種数が 0 でない形状における「穴」を横切るパスである．generator ループの数は K の種数を g とすると $2g$ となる．このようなパスは，Tree-Cotree Decomposition [9] により以下の手順により得られる：

- V と E を用いて全域木 T を作る
- F と E の内 T に含まれない辺を用いて全域木 T^* を作る
- T と T^* のどちらにも含まれない辺すべてについて，その辺に隣接する点を T の親までつなぐ

また (2) の境界ループは，ハーフエッジ構造に対する操作により簡単に得られる．

4.6　最適化問題

以上を踏まえ，すべての γ_{ijk} および非可縮なパスからなる集合を G とおくと，本研究における最適化の離散的な表

式は以下のように定式化できる：

$$\begin{array}{lll} \text{min.} & \sum_{ij \in E} w_{ij} |\widehat{d\alpha}_{ij} - \omega_{ij}|^2 & (15) \\ \text{s.t.} & \forall \gamma \in G,\ \sum_{e \in \gamma \subset E} \widehat{d\alpha}_e = 2k\pi (k \in \mathbf{Z}) & (16) \end{array}$$

5. 最適化計算

本章では，4 章で述べた離散的な最適化問題（式 15 参照）を，実際に疎行列ソルバーを用いて求解するアルゴリズムについて述べる．このアルゴリズムは既存の行列計算ライブラリや逆行列ソルバーを利用して容易に実装できる．またこの最適化問題は線形等式制約凸二次最適化問題であり，逆行列計算を 1 回行うだけで高速に計算でき，かつ凸問題のため局所最適解に陥ることがない．さらに，この逆行列は入力された三角形メッシュのグラフ構造のトポロジーが変更されない限り，$\hat{X}$ や各ループの指数ベクトル J の変更時に逆行列の再計算が不要である．このため，これらの変更に対してはさらに高速に動作する．

本研究のアルゴリズムは以下の流れにより行われる：

(1) 三角形メッシュ $K = \{V, E, F\}$ と各点上で定義されたベクトル場 $\hat{X}$ から，各種行列を定義する（5.1 項）
(2) 線形ソルバーにより $\widehat{d\alpha}$ を求める（5.2 項）
(3) $\widehat{d\alpha}$（微分形式）を $\hat{\alpha}$（積分形式）に変換する（5.3 項）

本章ではこの流れに沿って，各ステップで行う処理について説明する．

なお，以下では K を有向グラフとして扱い，辺に隣接する 2 点の内，点の番号が小さい方から大きい方へ向かう向きを正とする．

5.1 行列による表現

本章で登場する行列を以下に列挙する：

- $L \in \mathbb{R}^{|F| \times |E|}$: 各三角形ループの表現行列
- $J^L \in \mathbb{R}^{|F| \times |E|}$: 各三角形ループの指数を 2π 倍したもの
- $H \in \mathbb{R}^{(2g+b) \times |E|}$: 非可縮なループの表現行列
- $J^H \in \mathbb{R}^{n_H \times |E|}$: 非可縮なループに沿った回転数
- $O \in \mathbb{R}^{|E|}$: 辺 $e = ij \in E$ に対応する成分 O_e に ω_{ij} を格納した列ベクトル
- $W \in \mathbb{R}^{|E| \times |E|}$: 辺 $e = ij \in E$ に対応する対角成分 W_{ee} に ij の Cotangent Weight（4.3 項参照）を格納した行列
- $x \in \mathbb{R}^{|E|}$: $x_{ij} = \widehat{d\alpha}_{ij} - \omega_{ij}$ なる列ベクトル．本章の最適化は x を求めることが目標

三角形ループ

三角形ループの表現行列 $L \in \mathbb{R}^{|F| \times |E|}$ の各要素は以下のように表せる：

$$L_{ij} = \begin{cases} \pm 1 & (\text{面 } i \text{ に辺 } j \text{ が含まれる時}) \\ 0 & (\text{それ以外}) \end{cases} \tag{17}$$

なお L_{ij} の符号は，面 i の法線に対して右ねじの方向と辺 j の向きが一致すれば正，一致しなければ負となる．

非可縮なループ

非可縮なループの表現行列 $H \in \mathbb{R}^{(2g+b) \times |E|}$ の各要素は以下のように表せる：

$$H_{ij} = \begin{cases} \pm 1 & (\text{ループ } i \text{ に辺 } j \text{ が含まれる時}) \\ 0 & (\text{それ以外}) \end{cases} \tag{18}$$

なお L_{ij} の符号は，各パスの向きと辺 j の向きが一致すれば正，一致しなければ負となる．

また，本稿では証明は省くが，J^L は境界ループに対応する J^H の行の和に一致するようにする必要がある．

最適化問題

以上を踏まえると，式 16 の最適化問題の制約条件は以下のように書き換えられる：

$$\begin{bmatrix} L \\ H \end{bmatrix} (x + O) = 2\pi \begin{bmatrix} J^L \\ J^H \end{bmatrix}$$
$$\Leftrightarrow \begin{bmatrix} L \\ H \end{bmatrix} x = 2\pi \begin{bmatrix} J^L \\ J^H \end{bmatrix} - \begin{bmatrix} LO \\ HO \end{bmatrix}$$

よって，定義した行列を用いた最適化問題は以下のようになる：

$$\begin{array}{ll} \text{min.} & |Wx|^2 \\ \text{s.t.} & \begin{bmatrix} L \\ H \end{bmatrix} x = \begin{bmatrix} J^L \\ J^H \end{bmatrix} - \begin{bmatrix} LO \\ HO \end{bmatrix} \end{array} \tag{19}$$

5.2 線形ソルバーによる求解

一般に，A が横長の（i.e., 行数が列数よりも少ない）行列の場合，線形等式 $Ay = b$ の下で $|y|^2$ を最小化する問題は以下の計算を行うことにより求まる [10]：

$$y = (A^{\mathrm{T}} A)^{-1} A^{\mathrm{T}} b$$

よって，$A = \left[\begin{smallmatrix} L \\ H \end{smallmatrix} \right] W^{-1}$，$y = Wx$，$b = \left[\begin{smallmatrix} J^L \\ J^H \end{smallmatrix} \right] - \left[\begin{smallmatrix} LO \\ HO \end{smallmatrix} \right]$ とおくと，x は以下の手順により求まる：

(1) $y \leftarrow (A^{\mathrm{T}} A)^{-1} A^{\mathrm{T}} b$
(2) $x \leftarrow W^{-1} y$

W は対角行列なので，W^{-1} は W の対角要素の逆数を取ることにより線形時間で求めることができる．それ以外の疎行列の乗算操作は線形時間とみなせるため，上記

の手順の計算量は手順 1 の $(A^{\mathrm{T}}A)^{-1}$ の逆行列計算の計算量に支配される．ここで $A^{\mathrm{T}}A$ は，A のランクが A の行数と等しい限りは必ず半正定値行列になるため，一般の疎行列の逆行列計算ソルバーを利用して計算できる．本研究ではこのアルゴリズムを Eigen [11] に用意されている `SimplicialLDLT` ソルバーにより実装した．

また，一度 $(A^{\mathrm{T}}A)^{-1}$ を計算して結果を保存しておけば，A に変更がない限りは逆行列の再計算が不要である．このため，本アルゴリズムは指数の編集やベクトル場の変更といった A の変更が不要な操作に対しては理論上線形時間で動作する．

上記で求まった x を用いれば，$\widehat{d\alpha}$ は以下の式により求まる：

$$\widehat{d\alpha} = x + O \tag{20}$$

5.3 微分形式の積分形式への変換

離散 1-形式 $\widehat{d\alpha}$ から $\hat{\alpha}$ を求めるには，まずある点における $\hat{\alpha}$ を与えてから，式 10 を隣接する点に対して次々に適用していけばよい．本研究では，この計算を疑似逆行列の計算により実現する．

まず K 上の V と E により，V の番号の最も若い点（$i_0 \in V$ とおく）を頂点とする有向全域木 T を作る．この時，T に含まれる辺の数は $|V|-1$ である．次に，以下の値を持つ行列 $C \in \mathbb{R}^{(|V|-1)\times|V|}$ を作る：

$$C_{ij} = \begin{cases} \pm 1 & (T \text{ に } i \text{ 番目の辺が含まれるとき}) \\ 0 & (\text{それ以外}) \end{cases} \tag{21}$$

C_{ij} の正負は，T の向きと i 番目の辺の向きが一致すれば

次に，C の疑似逆行列 C^{+} を計算する．この時，入力されたメッシュが変更されない限り，C^{+} は一度計算すれば再計算は不要である．

最後に，$\widehat{d\alpha}$ の内 T に含まれる辺の成分のみを抽出した列ベクトル $\widetilde{d\alpha} \in \mathbb{R}^{|V|-1}$ を作れば，$\hat{\alpha}$ は $C^{+}\widetilde{d\alpha}$ の各要素を 2π で割った余りを計算することで得られる．

なお C は行数よりも列数の方が 1 だけ大きい行列なので，任意の $\theta \in \mathbb{R}$ に対して $C^{+}\widetilde{d\alpha} + \theta$ により得られる値もまた，すべての辺において式 10 を満たす．

6. 結果

本研究の高速なアルゴリズムを使えば，特異点位置をインタラクティブに編集することができる．図 8 に，本研究の手法により特異点の位置を移動したり，それぞれを対消滅させたりする様子を示す．このように本研究を使えば，ユーザのにとって好ましい位置に特異点の位置を移動させることができる．

本研究の手法は，図 9b に示すような凹凸のあるメッシュ

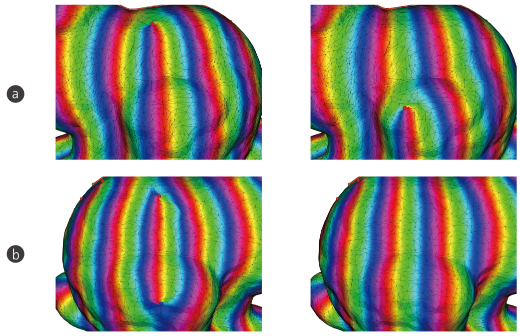

図 8 特異点に対する操作．(a) 移動，(b) 対消滅．

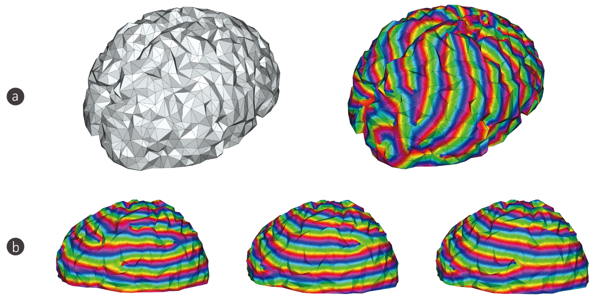

図 9 凹凸のあるメッシュで本研究の手法が動作する様子．(a) 凹凸のあるモデルとその上に [2] を用いて特異点を自動配置した様子，(b) モデルの上の特異点位置を操作する様子．

でも安定して動作し，特異点位置の編集操作も高速に行える．これはメッシュの切り方によるエネルギーの歪みを補正する Cotangent Weight の効果によるものであり，離散微分幾何に基づいた幾何形状処理手法の特徴でもある．

図 10 Zippable 構造 [12] に似た渦巻き模様．

また，渦巻き模様の中心位置を特異点として指定し，それ以外の面を全て非特異点として指定すれば，図 10 に示すような渦巻き模様を形状表面に配置できる．このような紋様は，ジッパーで簡易に形状を組み立てられる Zippable 構造 [12] への応用が期待できる． [12] では，形状に切り込みを入れてパッチに分割し，パッチ間の接続関係を考慮する複雑なアルゴリズムを採用していたのに対し，本研究の手法を用いれば非常に単純なアルゴリズムにより Zippable 構造を設計できる．

7. 評価

手法の有効性を確認するため，本研究では手法の評価として，異なる頂点数の 3 次元メッシュに対する実行時間の測定を行った．このために，ベクトル場 $\hat{X}$ と各ループの

指数 J^L と J^H を与えてから，スカラー場 $\hat{\alpha}$ を得るまでにかかる時間を測定した．また同じ測定を 2 回行い，2 回目の測定では 1 回目に計算した逆行列の計算結果を使い回すことで，1 回目は逆行列の計算も含めた実行時間，2 回目は逆行列の計算を含まない実行時間をそれぞれ測定した．入力には，頂点数の異なる同一の三角形メッシュを利用し，ベクトル場 $\hat{X}$ の計算には [9] を利用した．

測定は筆者のノートパソコン（Intel Core i7-10875H CPU@2.30GHz, 16.0GB RAM）にて行い，処理の並列化は行わず単一のスレッドのみを使用した．

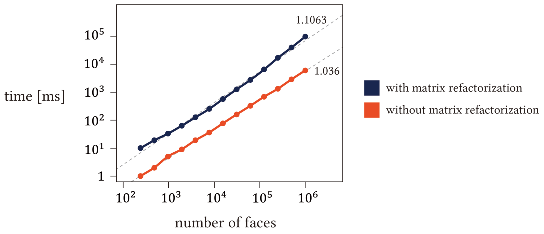

図 11 実行時間の計測結果．近似曲線は累乗近似を用いており，近似曲線近傍の数値は近似曲線の指数を示す．

図 11 に，実行時間の測定結果を示す．グラフに示すように，1k, 8k, 63k の面数の形状の沿ベクトル場パラメータ化の計算にはそれぞれ，逆行列計算を行った場合は 41 ms, 256 ms, 3209 ms，また逆行列計算を行わなかった場合は 6 ms, 38 ms, 312 ms を要した．このように面数が〜10k 程度であれば，逆行列が不要な特異点の操作は，リアルタイムのアニメーションにすら応用できるレベルで高速な計算を実現している．逆行列計算を行った場合と行わなかった場合は計算時間のオーダーが概ね 1 桁異なり，本研究のアルゴリズムの計算量が逆行列計算に支配されることを示唆している．また，逆行列計算を行わなかった場合の近似曲線の指数は 1.036 であり，理論通り面数と計算時間が概ね線形に比例した．さらに，本研究の測定では並列化を一切行っていないことを考えれば，行列計算等の並列化によりさらなる高速化が期待できる．

8. おわりに

本研究は，ある点が特異点となりそれ以外は特異点とならないような沿ベクトル場パラメータ化を，インタラクティビティを担保できるレベルで高速に計算する手法を実現した．この技術の鍵となるのは，各点におけるスカラー関数 $\hat{\alpha}$ ではなく，その微分形式 $\widehat{d\alpha}$ により問題を定式化した点にある．このために本研究ではまず連続微分幾何における定式化を再定義し，それを離散微分幾何を利用して離散化し，既存の疎行列ソルバーのライブラリを利用して簡単に実装できるアルゴリズムにまで落とし込んだ．この結果できたアルゴリズムは非常にシンプルであり，実装も単純で，かつ高速に動作する．

本研究は形状のインタラクティブな設計との相性が非常に高く，本稿で紹介した実例にとどまらない多様な応用が期待できる．特に，本研究では触れていないベクトル場生成やパラメータ化の手法と本研究の手法を組み合わせれば，全く新しいインタラクティブな設計体験を探求できると考えている．

謝辞

本研究は JST ACT-X (JPMJAX200L) および JST ERATO 川原万有情報網プロジェクト (JPMJER1501) の助成を受けたものである．

参考文献

[1] Ray, N., Li, W. C., Lévy, B., Sheffer, A. and Alliez, P.: Periodic Global Parameterization, *ACM Trans. Graph.*, Vol. 25, No. 4, p. 1460–1485 (online), DOI: 10.1145/1183287.1183297 (2006).

[2] Knöppel, F., Crane, K., Pinkall, U. and Schröder, P.: Stripe Patterns on Surfaces, *ACM Trans. Graph.*, Vol. 34, No. 4 (online), DOI: 10.1145/2767000 (2015).

[3] Ma, Z., Walzer, A., Schumacher, C., Rust, R., Gramazio, F., Kohler, M. and Bächer, M.: Designing Robotically-Constructed Metal Frame Structures, *Computer Graphics Forum*, Vol. 39, No. 2, pp. 411–422 (online), DOI: https://doi.org/10.1111/cgf.13940 (2020).

[4] Jakob, W., Tarini, M., Panozzo, D. and Sorkine-Hornung, O.: Instant Field-Aligned Meshes, *ACM Trans. Graph.*, Vol. 34, No. 6 (online), DOI: 10.1145/2816795.2818078 (2015).

[5] Tong, Y., Alliez, P., Cohen-Steiner, D. and Desbrun, M.: Designing Quadrangulations with Discrete Harmonic Forms, *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, SGP '06, Goslar, DEU, Eurographics Association, p. 201–210 (2006).

[6] Bommes, D., Zimmer, H. and Kobbelt, L.: Mixed-Integer Quadrangulation, *ACM Trans. Graph.*, Vol. 28, No. 3 (online), DOI: 10.1145/1531326.1531383 (2009).

[7] Meekes, M. and Vaxman, A.: Unconventional Patterns on Surfaces, *ACM Trans. Graph.*, Vol. 40, No. 4 (online), DOI: 10.1145/3450626.3459933 (2021).

[8] Crane, K., de Goes, F., Desbrun, M. and Schröder, P.: Digital Geometry Processing with Discrete Exterior Calculus, *ACM SIGGRAPH 2013 Courses*, SIGGRAPH '13, New York, NY, USA, Association for Computing Machinery, (online), DOI: 10.1145/2504435.2504442 (2013).

[9] Crane, K., Desbrun, M. and Schröder, P.: Trivial Connections on Discrete Surfaces, *Computer Graphics Forum*, Vol. 29, No. 5, pp. 1525–1533 (online), DOI: https://doi.org/10.1111/j.1467-8659.2010.01761.x (2010).

[10] Rencher, A. C. and Christensen, W. F.: *Methods of Multivariate Analysis (Wiley Series in Probability and Statistics)*, Wiley (2012).

[11] Guennebaud, G., Jacob, B. et al.: Eigen v3, http://eigen.tuxfamily.org (2010).

[12] Schüller, C., Poranne, R. and Sorkine-Hornung, O.: Shape Representation by Zippables, *ACM Trans. Graph.*, Vol. 37, No. 4 (online), DOI: 10.1145/3197517.3201347 (2018).