

フレームモデルを用いた折紙構造の形状生成法

FORM GENERATION OF RIGID-FOLDABLE ORIGAMI STRUCTURE
USING FRAME MODEL早川 健太郎^{*1}, 大崎 純^{*2}*Kentaro HAYAKAWA and Makoto OHSAKI*

A method is presented for generating a developable and rigid-foldable origami that approximates a curved surface with a simple crease pattern. We use an optimization method to obtain a polyhedron that satisfies geometric conditions for developability. A frame model is developed to use the same variables through all processes including optimization for form generation, evaluation of kinematical indeterminacy, and large-deformation folding analysis using a general finite element analysis software. Numerical examples show a regular grid approximating the target surface can be generated using a rigid-foldable origami, and fold lines can be sequentially fixed to obtain a surface including quadrilateral facets.

Keywords : *Origami, Rigid fold, Deployable structure, Frame model, Degree of kinematical indeterminacy, Large-deformation analysis*

折紙, 剛体折り, 展開構造, フレームモデル, 不安定次数, 大変形解析

1. はじめに

折紙は、数学、情報学、工学、生命科学、芸術などさまざまな分野に関わる領域横断的なテーマであり、近年は、折り動作により平坦な薄い紙から目標とする形状を作り出すという従来の折紙の概念にとどまらない、平坦折り¹⁾や剛体折紙²⁾などの新たな問題や概念が提示されている。折紙の幾何学やメカニズムに関する研究の発展にともなって、折紙に関するさまざまな定理が示され、折紙の設計手法も発展してきている。計算機を用いた折紙の設計も可能となり、Demaine and Demaine³⁾はこれを「計算折紙 (computational origami)」と表現している。また、工学分野においては、折紙の形状やメカニズムの性質を利用した折版構造や展開構造、衝撃吸収機構⁴⁾などへ応用され、ステントグラフトなどの体内で使用する医療機器⁵⁾から建築物⁶⁾までさまざまなスケールのものが提案されている。なかでも、折線部分以外で面内、面外への変形を許容しない剛体折紙は展開過程において面内にひずみや応力が発生しないことから、変形メカニズムが材料に依存しないという特長をもつ。剛体折り可能な折線のパターンとしてミウラ折り⁷⁾や Resch のパターン⁸⁾などが知られている。剛体折紙のもつ特長は展開構造物の構築に有利であり、その応用が期待されている。とくに、建築物における展開構造のデザインではさまざまな設計条件に対応する必要がある、剛体折り可能な多面体の一般的な形状生成法は極めて重要である。

本論文の目的は、建築物への応用を前提とした単純な折線パターンをもつ剛体折り可能な多面体を得ることである。その際、目標曲面を設定し、それを近似する多様な折線パターンの生成を試みる。

一般形状をもつ折紙の設計についてさまざまな手法が提案されているが、必ずしも建築物への応用に適しているとはいえない。

Tachi⁹⁾は紙の一部をひだ状に折りこむことでディスク同相（穴がなくかつ閉じていない）な多面体を1枚の紙を折って生成するアルゴリズムを提示し、Origamizer というソフトウェアとして実装、公開している。Origamizer に実装されているアルゴリズムでは折紙の展開図を作成する際に、折線パターンの解が必ず得られる保証はなかったが、後に、ひだを配置する同様の手法によって必ず解が得られるアルゴリズムが発表された¹⁰⁾。しかし、この手法で得られる折線パターンはしばしば複雑となり、製作が困難な場合がある。また、Tachi¹¹⁾により四角形メッシュ剛体折紙の一般的形状を生成する手法が示されている。この手法では剛体折り機構の幾何学的制約のもとに既存の剛体折り可能な折線パターンを組み合わせ、変形させることで形状のバリエーションを得ているが、曲面を近似する形状を生成するためには試行錯誤が必要となる。曲面の折紙による近似手法としては、Zhao ら¹²⁾による一般化された waterbomb tessellation を用いた手法が挙げられる。ただし、この手法では既存の折線パターンを用いるため、パターンのバリエーションを得ることは難しい。

本論文では、平面上でヒンジ接合によりパネルを組み立て、目的の形状に折り上げることで立体構造物を生成することを想定し、展開可能なディスク同相の多面体の折線パターンを得る手法を検討し、平坦折りについては考慮しない。このとき、平面に展開可能となる幾何学的制約を満たす形状を直接的に得ることは困難なため、最適化手法を用いて近似的に形状を求める。また、剛体折紙のモデル化にフレームモデルを用いることで、最適化手法による形状生成や不安定次数の評価、大変形解析を含む全過程においてモデルの再構築や変数の変換が不要となること、および汎用の有限要素解析ソフトウェアによる大変形解析が可能であることを示す。

^{*1} 京都大学大学院工学研究科建築学専攻 大学院生

^{*2} 京都大学大学院工学研究科建築学専攻 教授・工博

Grad. Student, Dept. of Architecture and Architectural Eng., Kyoto Univ.
Prof., Dept. of Architecture and Architectural Eng., Kyoto Univ., Dr.Eng.

2. 剛体折紙のモデル化

剛体折紙の変形過程は、不安定トラスモデルあるいは回転角モデルを用いてモデル化するのが一般的である¹³⁾。不安定トラスモデルは、線要素と節点により構成され、節点位置を変数とするモデルである。三角形の面の稜線を不変長の3本のバーで、四角形以上の多角形の面は稜線に加えて対角線要素を不変長のバーで構成し、それらを節点においてピン接合する。また、回転角モデルは面要素と回転ヒンジにより構成され、ヒンジの回転角を変数とするモデルであり、剛な平面要素をヒンジで接続することから、各内部頂点まわりに接続する平面の制約が球面上のループする閉じたリンク機構と同等のモデルである¹⁴⁾。

上記のようなモデルでは、剛体折紙の輪郭形状とモデルの形状が一致し、比較的単純にモデル化できる。しかし、不安定トラスモデルでは、四角形以上の多角形をモデル化する場合、各面の面内・面外剛性を保証するために対角線方向に部材を配置するとともに部材を多角錐のように立体的に組む必要があり、部材が非常に多くなることがある。また、回転角モデルでは、面要素を用いるためヒンジ部分のモデル化が煩雑となり、汎用の有限要素解析ソフトウェアで大変形過程を追跡するのが困難である。

そこで、本論文では、剛体折紙の解析を部分剛接合骨組における既存の解析手法と同様に行うため、Fig. 1に示すようなフレームモデルを提案する。灰色の実線はモデル化する多面体の外周、および山折線（ z 軸正方向から見たときの尾根線）、点線は谷折線（ z 軸正方向から見たときの谷線）を表しており、黒色の実線はフレーム要素、折線とフレーム要素の交点は折線まわりに回転するヒンジである。フレームモデルでは、一端がヒンジ、他端が剛接合されるフレーム要素によって剛体折紙をモデル化する。フレームモデルの変数は折線上および多面体外周上の節点座標であり、剛接合される節点の位置は折線で囲まれた面内であれば任意である。

フレームモデルでは最少で辺の数と同数の部材数で各面を定義できるため、四角形以上の多角形を含む場合にも少ない部材で多面体をモデル化できる。また、梁要素を用いた大変形解析が可能であり、汎用有限要素解析ソフトウェアによって容易に解析できる。

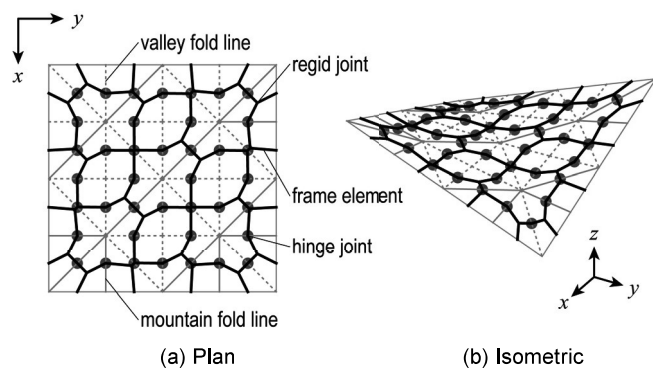


Fig. 1 Example of a frame model

3. 剛体折り可能なモデルの生成手順

Fig. 2に示す Step 1～5のフローにしたがい曲面を近似する展開可能かつ剛体折り可能な多面体を得る。

三角形で構成されるディスク同相な多面体では、微小変形における剛体折りによる変形自由度は適合条件行列がフルランクである一般的な立体状態において「外周の稜線の数-3」である¹⁵⁾。すなわち、1つの頂点に折線が3本のみ接続するなど一部の例外を除き、非平坦時の不安定性は変形に依存しない。したがって、初期の一般形状での微小変形のメカニズムが確認できれば、大変形メカニズムの存在が保証される。

しかし、三角形のみで構成される多面体では、変形自由度が必要以上に大きくなることが多い。そこで、Step 3では平面へ展開可能な多面体形状を得ると同時に、一部の隣接する2つの三角形を1つの平面として統合し、2面の間の折線を削除する（平面内に結合すること）ことで三角形と四角形の混在する多面体を作成し、変形自由度を削減する。このとき、削除する折線の数を変えることで変形自由度の異なる複数のモデルを生成し、それぞれのモデルについて、Step 4および Step 5を実行することで、最も望ましい変形自由度や変形メカニズムをもつ多面体形状を選択する。

Step 4ではフレームモデルの釣り合い行列にヒンジ部分のモーメント解放条件式を加えて拡張した行列の特異値分解によって微小変形の不安定次数を求め^{15, 16)}、モデル全体での剛体変位を除いた不安定な変形モード、すなわち微小変形のメカニズムの数を確認する。Step 3までで得られる多面体について、初期状態から平面までの連続的な展開可能性および展開過程における剛体折り可能性は Step 4まででは保証されないため、Step 5において幾何学的非線形解析によって大変形メカニズムを検証し、展開可能性と剛体折り可能性を確認する。

Step 4および Step 5を実行した結果、展開可能性と剛体折り可能性を満たす結果が得られなかった場合には Step 1に戻り、異なる三角形分割パターンの多面体について上記の作業を繰り返す。

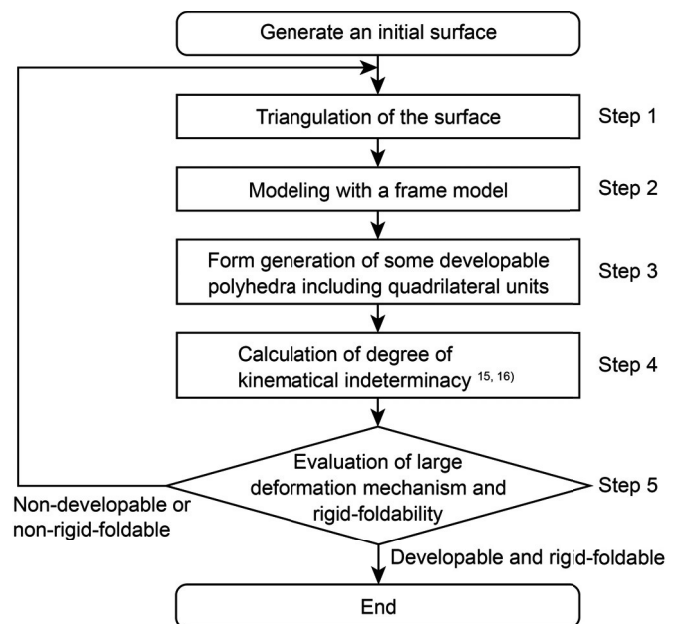


Fig. 2 Flowchart for generating a rigid-foldable polyhedron

4. 展開可能条件・折線削除条件の定式化

ある折り状態（立体状態）から、展開状態（平面に広げた状態）に面の変形をとみなすために到達するために多面体折紙が満たすべき条件を展開可能条件とよぶ。これは、各頂点において接続する折線のなす角の和が 2π となる条件として定式化される¹⁷⁾。ただし、この条件は展開可能性に関する必要条件であることに注意が必要である。

多面体折紙の内部頂点の数を V とし、内部頂点番号を v ($1 \leq v \leq V$) とする。Fig. 3に示すように頂点 v まわりの折線間の k 番目の角度を θ_k^v とすると、展開可能条件は以下のように書ける。

$$F_1 = \sum_{v=1}^V \left(\sum_{k=1}^{f^v} \theta_k^v - 2\pi \right)^2 = 0 \quad (1)$$

ここで、 f^v は頂点 v に接続する折線の数である。

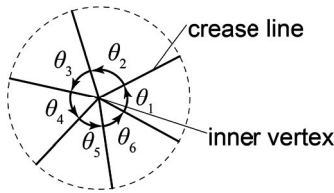


Fig. 3 Angle between crease lines

また、指定された折線まわりの回転を固定し折線を削除することで、四角形の面を生成し変形自由度を削減するため、削除される折線に接続する2平面が平行となるように下記の拘束を課す。

多面体折紙の折線の本数を E とし、折線番号を e ($1 \leq e \leq E$) とする。折線 e に接続する2平面の単位法線ベクトルの x 方向成分および y 方向成分をそれぞれ $n_{e,k}^x$, $n_{e,k}^y$ ($k=1,2$) とし、削除する折線の集合を E_D とすると、2平面が平行になる条件は以下のように書ける。

$$F_2 = \sum_{e \in E_D} \left\{ (n_{e,1}^x - n_{e,2}^x)^2 + (n_{e,1}^y - n_{e,2}^y)^2 \right\} = 0 \quad (2)$$

ただし、各面の法線ベクトルの z 方向成分が十分に大きいことを仮定している。任意の多面体に対して F_1 および F_2 は0以上であることから、多面体の形状が満たすべき条件は以下ようになる。

$$F = F_1 + F_2 = 0 \quad (3)$$

式(3)を満たす解形状を最適化手法を用いて求める。フレームモデルの独立な節点座標のベクトルを $\mathbf{X}$ とし、制約条件を満たす $\mathbf{X}$ の集合を χ とすると、式(3)を満たす解を求める問題は関数 F を最小化する最適化問題として以下のように定式化できる。

$$\begin{aligned} &\text{Minimize: } F(\mathbf{X}) = F_1(\mathbf{X}) + F_2(\mathbf{X}) \\ &\text{subject to: } \mathbf{X} \in \chi \end{aligned} \quad (4)$$

制約条件は多面体の各面の形状や曲面との誤差、モデルの外周の形状などに基いて定める。 $\mathbf{X}$ および χ の具体的な例は次節以降で説明する。最適化問題(4)を解いて目的関数が0となる解が得られたとき、指定された折線に接続する面を1つの四角形とし、展開可能条件を満たす折線パターンを持つ多面体を得られたものとみなす。

本節で示した定式化は、Fig. 2のStep 1やStep 2のアルゴリズムに依存しない。すなわち、さまざまな目標曲面に対して適切な三角形分割の手法を任意に選択し、解析手法などに応じてフレームモデル以外のモデル化の手法も選択することができる。

5. 曲面の三角形分割とフレームモデルにおける変数の縮約

曲面の三角形分割にはドロネー分割¹⁸⁾などさまざまな手法が挙げられるが、本論文ではFig. 4に示すように x 方向 $m+1$ 行、 y 方向 $n+1$ 列の分割点 $P_{i,j}$ ($i=1,2,\dots,m+1; j=1,2,\dots,n+1$)を曲面上にとり、それらを接続することで曲面を三角形分割する。分割点 $P_{i,j}$ の位置ベクトルを以下のように表す。

$$\mathbf{p}_{i,j} = (p_{i,j}^x, p_{i,j}^y, p_{i,j}^z)^T \quad (5)$$

点 $P_{i,j}$ と $P_{i+1,j}$ および点 $P_{i,j}$ と $P_{i,j+1}$ を接続することで曲面を x 方向 m 行、 y 方向 n 列のグリッド状に分割し、各グリッドの対角線にあたる点 $P_{i,j}$ と $P_{i+1,j+1}$ もしくは点 $P_{i+1,j}$ と $P_{i,j+1}$ を接続する。このとき、分割点間の斜め方向の接続条件を以下に定義する m 行 n 列の行列 $\mathbf{T}$ を用いて定式化する。

$$\begin{aligned} \mathbf{T} &= [t_{i,j}] \quad (i=1,\dots,m; j=1,\dots,n) \\ t_{i,j} &= \begin{cases} 0 & (P_{i,j} \text{と} P_{i+1,j+1} \text{を接続}) \\ 1 & (P_{i+1,j} \text{と} P_{i,j+1} \text{を接続}) \end{cases} \end{aligned} \quad (6)$$

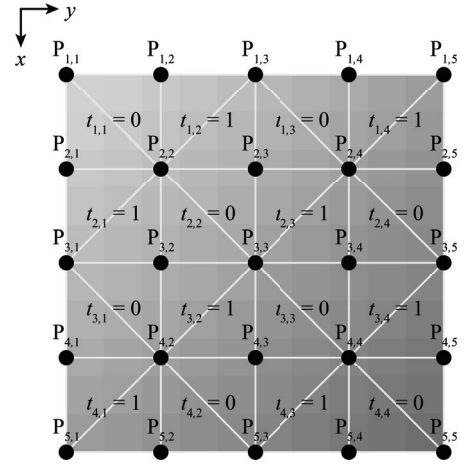


Fig. 4 Triangulation of a curved surface (4x4 grid)

三角形分割した曲面をフレームでモデル化する。フレームの節点を、Fig. 5に示すように $A_{i,j}$, $B_{i,j}$, $C_{i,j}$, $D_{i,j,k}$ の4種に分類する。それぞれの位置ベクトルを $\mathbf{a}_{i,j}$, $\mathbf{b}_{i,j}$, $\mathbf{c}_{i,j}$, $\mathbf{d}_{i,j,k}$ とし、以下のように定義する。ただし、式(10)は右辺に示す三角形内部に節点 $D_{i,j,k}$ が位置することを意味している。

$$\mathbf{a}_{i,j} = (a_{i,j}^x, a_{i,j}^y, a_{i,j}^z)^T = \frac{\mathbf{p}_{i,j} + \mathbf{p}_{i+1,j}}{2} \quad (7)$$

$$\mathbf{b}_{i,j} = (b_{i,j}^x, b_{i,j}^y, b_{i,j}^z)^T = \frac{\mathbf{p}_{i,j} + \mathbf{p}_{i,j+1}}{2} \quad (8)$$

$$\mathbf{c}_{i,j} = (c_{i,j}^x, c_{i,j}^y, c_{i,j}^z)^T = \begin{cases} \frac{\mathbf{p}_{i,j} + \mathbf{p}_{i+1,j+1}}{2} & (t_{i,j} = 0) \\ \frac{\mathbf{p}_{i,j+1} + \mathbf{p}_{i+1,j}}{2} & (t_{i,j} = 1) \end{cases} \quad (9)$$

$$\mathbf{d}_{i,j,1} = (d_{i,j,1}^x, d_{i,j,1}^y, d_{i,j,1}^z)^T \subset \begin{cases} \Delta P_{i,j} P_{i,j+1} P_{i+1,j+1} & (t_{i,j} = 0) \\ \Delta P_{i,j} P_{i,j+1} P_{i+1,j} & (t_{i,j} = 1) \end{cases} \quad (10)$$

$$\mathbf{d}_{i,j,2} = (d_{i,j,2}^x, d_{i,j,2}^y, d_{i,j,2}^z)^T \subset \begin{cases} \Delta P_{i,j} P_{i+1,j} P_{i+1,j+1} & (t_{i,j} = 0) \\ \Delta P_{i,j+1} P_{i+1,j} P_{i+1,j+1} & (t_{i,j} = 1) \end{cases}$$

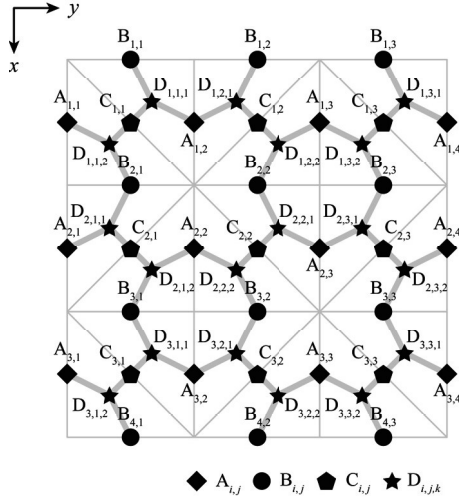


Fig. 5 Classification of nodes (3x3 grid)

モデル化された多面体において隣接する三角形が共有する辺の両端の頂点が一致することから、フレームモデルの各節点の座標をすべて独立に決めることはできない。以下では、4点 $P_{i,j}$, $P_{i+1,j}$, $P_{i,j+1}$, $P_{i+1,j+1}$ に囲まれるグリッドを i 行 j 列のマスと呼ぶこととする。

まず、Fig. 4, 5において x 軸と平行な辺 $\overline{P_{i,j+1}P_{i+1,j+1}}$ を共有する三角形について考える。 i 行 j 列のマスと i 行 $j+1$ 列のマスの三角形が辺 $\overline{P_{i,j+1}P_{i+1,j+1}}$ を共有することから、この2つのマスの斜め方向の接続条件、すなわち、 $t_{i,j}$, $t_{i,j+1}$ の値の組により場合分けができ、節点座標の制約は以下ようになる。

(a) $(t_{i,j}, t_{i,j+1}) = (0, 0)$ のとき

$$\mathbf{c}_{i,j} - \mathbf{b}_{i,j} = \mathbf{b}_{i+1,j+1} - \mathbf{c}_{i,j+1} \Leftrightarrow \mathbf{c}_{i,j+1} = -\mathbf{c}_{i,j} + \mathbf{b}_{i,j} + \mathbf{b}_{i+1,j+1} \quad (11a)$$

(b) $(t_{i,j}, t_{i,j+1}) = (0, 1)$ のとき

$$\mathbf{c}_{i,j} - \mathbf{b}_{i,j} = \mathbf{c}_{i,j+1} - \mathbf{b}_{i,j+1} \Leftrightarrow \mathbf{c}_{i,j+1} = \mathbf{c}_{i,j} - \mathbf{b}_{i,j} + \mathbf{b}_{i,j+1} \quad (11b)$$

(c) $(t_{i,j}, t_{i,j+1}) = (1, 0)$ のとき

$$\mathbf{c}_{i,j} - \mathbf{b}_{i+1,j} = \mathbf{c}_{i,j+1} - \mathbf{b}_{i+1,j+1} \Leftrightarrow \mathbf{c}_{i,j+1} = \mathbf{c}_{i,j} - \mathbf{b}_{i+1,j} + \mathbf{b}_{i+1,j+1} \quad (11c)$$

(d) $(t_{i,j}, t_{i,j+1}) = (1, 1)$ のとき

$$\mathbf{c}_{i,j} - \mathbf{b}_{i+1,j} = \mathbf{b}_{i,j+1} - \mathbf{c}_{i,j+1} \Leftrightarrow \mathbf{c}_{i,j+1} = -\mathbf{c}_{i,j} + \mathbf{b}_{i+1,j} + \mathbf{b}_{i,j+1} \quad (11d)$$

以上をまとめると次のようになる。

$$\mathbf{c}_{i,j+1} = (-1)^{1-t_{i,j}+t_{i,j+1}} (\mathbf{c}_{i,j} - \mathbf{b}_{i+1,j+1}) + \mathbf{b}_{i+1-t_{i,j},j+1} \quad (12)$$

次に、 y 軸と平行な辺 $\overline{P_{i+1,j}P_{i+1,j+1}}$ は、 i 行 j 列のマスと $i+1$ 行 j 列のマスの三角形によって共有されることから、同様にして以下の制約が得られる。

$$\mathbf{c}_{i+1,j} = (-1)^{1-t_{i,j}+t_{i,j+1}} (\mathbf{c}_{i,j} - \mathbf{a}_{i,j+t_{i,j}}) + \mathbf{a}_{i+1-t_{i,j},j+1-t_{i,j}} \quad (13)$$

最後に、斜め方向の辺 $\overline{P_{i,j}P_{i+1,j+1}}$ または辺 $\overline{P_{i,j+1}P_{i+1,j}}$ を共有する三角形について考える。 i 行 j 列のマスの三角形が斜めの辺を共有することから、以下の制約が得られる。

$$\mathbf{b}_{i+1,j} = -\mathbf{b}_{i,j} + \mathbf{a}_{i,j} + \mathbf{a}_{i,j+1} \quad (14)$$

式(12)–(14)より、曲面を x 方向 m 分割、 y 方向 n 分割したとき、節点 $A_{i,j}$ ($1 \leq i \leq m; 1 \leq j \leq n+1$), $B_{i,j}$ ($1 \leq j \leq n$), $C_{i,1}$ を定めることで残りの節点座標が定まることが分かる。したがって、独立変数の数を $3(m+1)(n+1)$ 個に縮約でき、多面体の頂点の座標を変数とした場合と同じになる。

6. 剛体折り可能混合メッシュ多面体の生成

本節では、5節のモデルを対象として、三角形と四角形で構成される剛体折り紙において、4節で示した最適化問題を定式化する。

曲面を x 方向 m 分割、 y 方向 n 分割してグリッドを構成し、節点 $A_{i,j}$ ($1 \leq i \leq m; 1 \leq j \leq n+1$), $B_{i,j}$ ($1 \leq j \leq n$), $C_{i,1}$ の座標をまとめたベクトル $\mathbf{X}$ を設計変数とする。ここで、 $\mathbf{X}$ は5節で示した方法で縮約されているものとする。

点 $P_{i,j}$ まわりの折線間の k 番目の角度を $\theta_{i,j,k}(\mathbf{X})$ とすると、4節より、 F_1 を $\mathbf{X}$ の関数として展開可能条件は以下のように書ける。

$$F_1(\mathbf{X}) = \sum_{\substack{2 \leq i \leq m \\ 2 \leq j \leq n}} \left(\sum_{k=1}^{f_{i,j}} \theta_{i,j,k}(\mathbf{X}) - 2\pi \right)^2 = 0 \quad (15)$$

ここで、 $f_{i,j}$ は点 $P_{i,j}$ に接続する折線の数である。

また、簡単のため節点 $C_{i,j}$ を通る折線を削除するものとし、 $n_{i,j,k}^x(\mathbf{X})$, $n_{i,j,k}^y(\mathbf{X})$ ($k=1,2$) を $C_{i,j}$ を通る折線に接続する面の単位法線ベクトルの x, y 方向成分とする。削除する折線の位置 (i,j) の集合を E_D とすると、 E_D に含まれる折線に接続する2平面が平行になる条件は F_2 を $\mathbf{X}$ の関数として以下のようにになる。

$$F_2(\mathbf{X}) = \sum_{(i,j) \in E_D} \left\{ \left(n_{i,j,1}^x(\mathbf{X}) - n_{i,j,2}^x(\mathbf{X}) \right)^2 + \left(n_{i,j,1}^y(\mathbf{X}) - n_{i,j,2}^y(\mathbf{X}) \right)^2 \right\} = 0 \quad (16)$$

ベクトル $\mathbf{a}_{i,j}(\mathbf{X})$ の x, y, z 座標成分を $a_{i,j}^x(\mathbf{X})$, $a_{i,j}^y(\mathbf{X})$, $a_{i,j}^z(\mathbf{X})$ とする。ベクトル $\mathbf{b}_{i,j}(\mathbf{X})$, $\mathbf{c}_{i,j}(\mathbf{X})$ についても同様である。節点 $A_{i,j}$, $B_{i,j}$, $C_{i,j}$ を初期形状の曲面に z 軸と平行に投影した点の z 座標と $a_{i,j}^z(\mathbf{X})$, $b_{i,j}^z(\mathbf{X})$, $c_{i,j}^z(\mathbf{X})$ との差 $\Delta a_{i,j}^z(\mathbf{X})$, $\Delta b_{i,j}^z(\mathbf{X})$, $\Delta c_{i,j}^z(\mathbf{X})$ の絶対値の上限値を $\Delta \bar{a}_{i,j}^z$, $\Delta \bar{b}_{i,j}^z$, $\Delta \bar{c}_{i,j}^z$ とする。多面体の4隅の頂点の位置ベクトル $\mathbf{p}_i^c(\mathbf{X})$ は指定値 $\bar{\mathbf{p}}_i^c$ に一致するものとする。また、 xy 投影平面内での4辺上の節点は、最適化の過程でそれぞれの辺上に存在するものとし、座標成分の指定値を $\bar{a}_{i,j}^y$, $\bar{b}_{i,j}^x$ とする。さらに、 $\theta_{i,j,k}(\mathbf{X})$ の上・下限値をそれぞれ $\theta_{\max}$, $\theta_{\min}$ とし、目的関数 $F(\mathbf{X}) = F_1(\mathbf{X}) + F_2(\mathbf{X})$ を最小化する最適化問題を以下のように定式化する。

$$\text{Minimize: } F(\mathbf{X}) = F_1(\mathbf{X}) + F_2(\mathbf{X})$$

Subject to:

$$\begin{aligned} & \theta_{\min} \leq \theta_{i,j,k}(\mathbf{X}) \leq \theta_{\max}, \\ & \quad (i=2, \dots, m; j=2, \dots, n; k=1, \dots, f_{i,j}) \\ & -\Delta \bar{a}_{i,j}^z \leq \Delta a_{i,j}^z(\mathbf{X}) \leq \Delta \bar{a}_{i,j}^z, \quad (i=1, \dots, m; j=1, \dots, n+1) \\ & -\Delta \bar{b}_{i,j}^z \leq \Delta b_{i,j}^z(\mathbf{X}) \leq \Delta \bar{b}_{i,j}^z, \quad (i=1, \dots, m+1; j=1, \dots, n) \\ & -\Delta \bar{c}_{i,j}^z \leq \Delta c_{i,j}^z(\mathbf{X}) \leq \Delta \bar{c}_{i,j}^z, \quad (i=1, \dots, m; j=1, \dots, n) \\ & \mathbf{p}_i^c(\mathbf{X}) = \bar{\mathbf{p}}_i^c, \quad (i=1, 2, 3, 4) \\ & a_{i,1}^y = \bar{a}_{i,1}^y, \quad a_{i,n+1}^y = \bar{a}_{i,n+1}^y, \quad (i=1, \dots, m) \\ & b_{1,j}^x = \bar{b}_{1,j}^x, \quad b_{m+1,j}^x = \bar{b}_{m+1,j}^x, \quad (j=1, \dots, n) \end{aligned} \quad (17)$$

削除する折線の選択には、さまざまな基準が考えられる。ここでは、まず Step 2 で得られた多面体において、節点 $C_{i,j}$ を通る折線のなかから接続する2面のなす角が最も小さいものを E_D に加えて最適化問題(17)を解く。得られた解について同様の操作を行い、以上の手順を繰り返して、不安定次数が0となる、もしくは最適化問題(17)の解が得られなくなる時点で Step 3 を終了する。

Step 4 および Step 5 では、Step 3 で生成された複数のモデルに対して変形メカニズムを確認し、展開可能な剛体折り機構をもつモデルを抽出する。適切なモデルが得られなかった場合には、Step 1 に戻り、分割点 $P_{i,j}$ の接続関係が異なるモデルをあらたに生成して、再度 Step 3 から5を繰り返す。

7. 形状生成例

Step 1~4 を実行するプログラムを Python 3.6 で作成し、最適化には逐次2次計画法をもとにしたアルゴリズムである SLSQP¹⁹⁾ を SciPy よりインポートして用いる。Step 5 の大変形解析には Abaqus 2016²⁰⁾ を用いる。

目標曲面には 4×4 テンソル積ベジエ曲面¹⁹⁾ を用いる。 $M \times N$ 次のテンソル積ベジエ曲面上の点の位置ベクトル $\mathbf{S}_b^{M,N}$ は u, v をパラメータとして以下のように定義される。

$$\mathbf{S}_b^{M,N}(u,v) = \sum_{i=0}^M \sum_{j=0}^N \mathbf{R}_{i,j} B_i^M(u) B_j^N(v) \quad (18)$$

$$B_i^K(t) = \binom{K}{i} t^i (1-t)^{K-i} \quad (K \in \{M, N\}; t \in \{u, v\})$$

ここで、 $\mathbf{R}_{i,j}$ は曲面の制御点 (i, j) の位置ベクトルであり、 $B_i^K(t)$ はバーンスタイン基底関数である。パラメータ u, v の値を $0 \leq u \leq 1$, $0 \leq v \leq 1$ の範囲で1組定めることで曲面上の点が定義される。

7.1. モデル1

モデル1における最適化問題(17)の初期形状には Table 1 に示す双曲放物面状の目標曲面を Fig. 6 のように三角形分割した多面体を用いる。曲面を xy 平面に投影した正方形の1辺の長さは3であり、ライズは1.2である。形状生成の際には単位は重要でないので省略する。

Table 1 Coordinates of control points

(i, j)	Model 1			Model 2		
	x	y	z	x	y	z
(0,0)	0.0	0.0	0.0	0.0	0.0	0.0
(1,0)	1.0	0.0	0.4	1.0	0.0	0.8
(2,0)	2.0	0.0	0.8	2.0	0.0	0.8
(3,0)	3.0	0.0	1.2	3.0	0.0	0.0
(0,1)	0.0	1.0	0.4	0.0	1.0	0.8
(1,1)	1.0	1.0	0.4	1.0	1.0	1.6
(2,1)	2.0	1.0	0.6	2.0	1.0	1.6
(3,1)	3.0	1.0	0.8	3.0	1.0	0.8
(0,2)	0.0	2.0	0.8	0.0	2.0	0.8
(1,2)	1.0	2.0	0.6	1.0	2.0	1.6
(2,2)	2.0	2.0	0.4	2.0	2.0	1.6
(3,2)	3.0	2.0	0.4	3.0	2.0	0.8
(0,3)	0.0	3.0	1.2	0.0	3.0	0.0
(1,3)	1.0	3.0	0.8	1.0	3.0	0.8
(2,3)	2.0	3.0	0.4	2.0	3.0	0.8
(3,3)	3.0	3.0	0.0	3.0	3.0	0.0

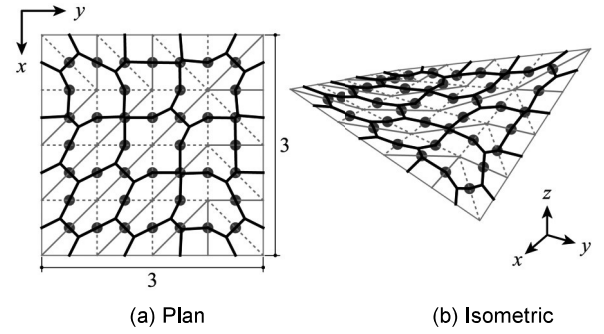


Fig. 6 Initial polyhedron of model 1

曲面を 4×4 グリッドに分割し、分割点の接続状況はランダムに生成した数パターンのなかから式(19)で与えるものを用いる。

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad (19)$$

また、最適化問題のパラメータ値は $\theta_{\min} = \pi/12$, $\theta_{\max} = \pi/3$, $\Delta \bar{a}_{i,j}^z = \Delta \bar{b}_{i,j}^z = \Delta \bar{c}_{i,j}^z = 0.2$ とする。

Step 3 では、6節に示した方法により、最大15本の折線が削除された15個のモデルが得られた。ここでは、折線を7本削除した例について Step 4 および Step 5 の結果を示す。最適化問題(17)の解形状を Fig. 7 に、目的関数を最小化して0にすることを目的としている諸量を Fig. 8, 9 に示す。Fig. 8, 9 および Table 2 より、目的関数値は十分に小さく、得られた解形状は展開可能条件を満たし、かつ削除された折線に接続する2面は十分な精度で同一平面上にあるとみなせる。

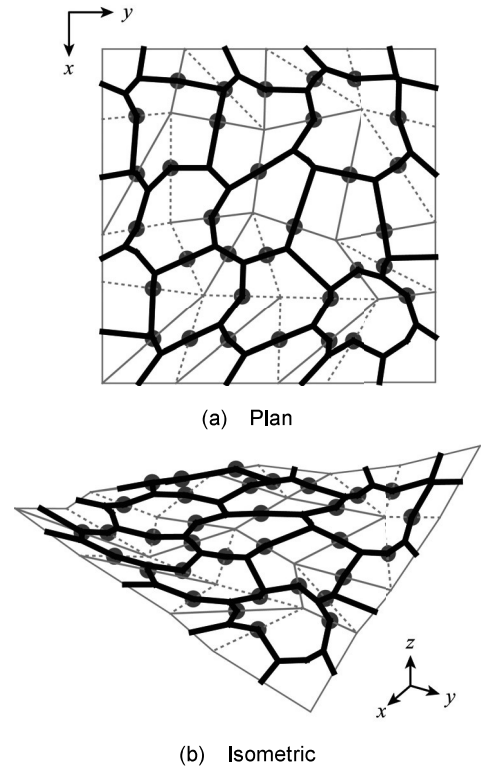


Fig. 7 Optimal shape of model 1

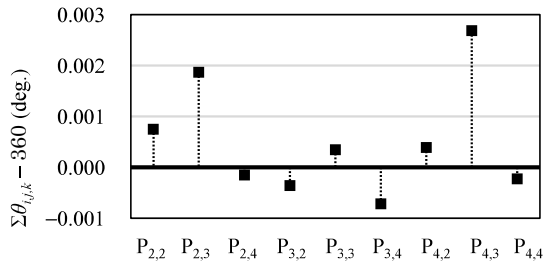


Fig. 8 Sum of angles between crease lines at each vertex

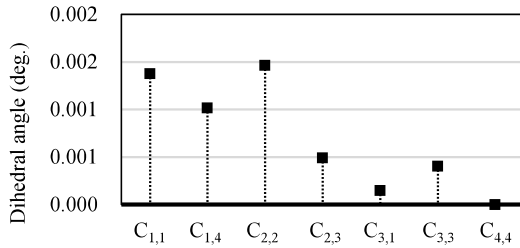


Fig.9 Dihedral angles of facets connected to deleted crease lines

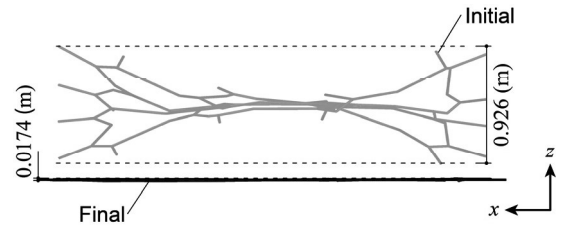
Table 2 Results of optimization

Objective function $F(\mathbf{X})$		5.402×10^{-9}
Error of $\sum \theta_{i,j,k}(\mathbf{X})$ from 360 (deg.)	Max.	2.683×10^{-3}
	Min.	-7.168×10^{-4}
Dihedral angle of assigned facets (deg.)	Max.	1.466×10^{-3}
	Min.	2.832×10^{-6}
DOF		6
Reduction of DOF from initial shape		7

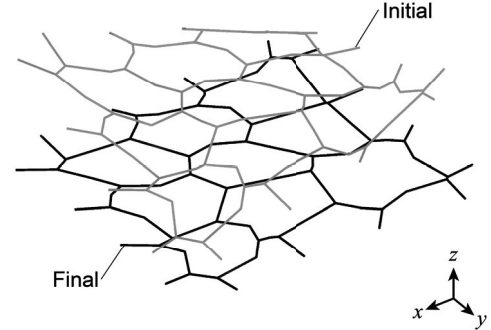
ある多面体が剛体折り可能であるとき、その多面体には変形過程においてひずみが発生しないことから、大変形解析の全過程におけるひずみを確認することにより、剛体折り可能であるかを判定する。

大変形解析では、1つの節点に x, y 方向の拘束を与え、その節点から離れたもう1つの節点に x 方向の拘束を与えたうえで、その2節点に7節点を加えた9節点に z 座標を0とするような強制変位を与えることにより、モデル全体の剛体変位の6自由度と剛体折り機構による変形の6自由度を合わせた計12自由度を拘束した。ここでは、モデル外周の xy 投影平面での1辺の長さを3 mとする。部材断面は外径が0.05 mの円筒であり、厚みは0.002 mである。また、材料のヤング率は $200 \times 10^9 \text{ N/m}^2$ 、ポアソン比は0.3である。ただし、解析の結果ひずみが微小であれば、これらの量の解析結果への影響は微小である。

解析結果を Fig. 10および Table 3に示す。Fig. 10では部材の太さを0.2倍して表示している。Table 3のとおり、大変形解析の全過程でのひずみは正側、負側ともに0.001%未満と小さな値であり、剛体折り可能な機構が得られたとみなすことができる。また、最終状態での節点の z 座標の存在範囲はスパン比で約0.58%、ライズ比で約1.9%とほぼ平面に展開されたとみなすことができる値となった。この誤差の原因は、頂点周りの折線間の角度の和の 360° からの誤差によるものであると考えられる。



(a) Elevation



(b) Isometric

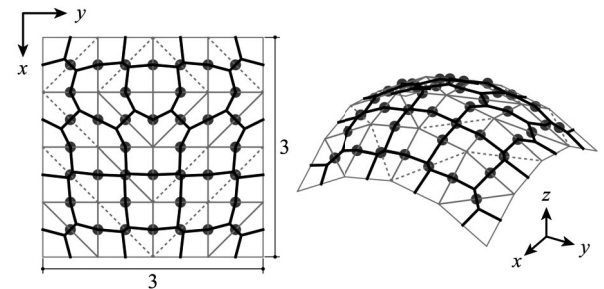
Fig.10 Large deformation analysis of model 1

Table 3 Results of large deformation analysis

Strain through analysis	Max.	6.074×10^{-6}
	Min.	-6.525×10^{-6}
Range of z coordinates of nodes (m)		1.742×10^{-2}
Ave. error of z coordinates of nodes from 0 (m)		2.918×10^{-3}

7.2. モデル2

モデル2における最適化問題(17)の初期形状には Table 1に示すドーム状の目標曲面を Fig. 11のように三角形分割した多面体を用いる。曲面を xy 平面に投影した正方形の1辺の長さは3であり、中央点のライズは1.2である。



(a) Plan

(b) Isometric

Fig. 11 Initial polyhedron of model 2

曲面を 4×4 グリッドに分割し、分割点の接続状況はランダムに生成した数パターンのなかから式(20)で与えるものを用いる。

$$\mathbf{T} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad (20)$$

また、最適化問題のパラメータ値は $\theta_{\min} = \pi/12$, $\theta_{\max} = \pi/3$, $\Delta \bar{a}_{i,j}^z = \Delta \bar{b}_{i,j}^z = \Delta \bar{c}_{i,j}^z = 0.2$ とする。

Step 3では、6節に示した方法により、最大13本の折線が削除された13個のモデルが得られた。ここでは、折線を7本削除した例についてStep 4およびStep 5の結果を示す。最適化問題(17)の解形状をFig. 12に、目的関数を最小化して0にすることを目的としている諸量をFig. 13, 14に示す。Fig. 13, 14およびTable 4より、目的関数値は十分に小さく、得られた解形状は展開可能条件を満たし、かつ削除された折線に接続する2面は十分な精度で同一平面上にあるとみなせる。

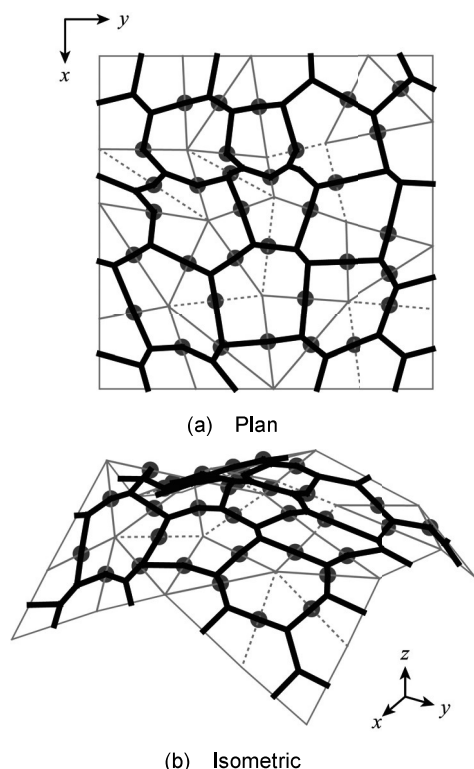


Fig. 12 Optimal shape of model 2

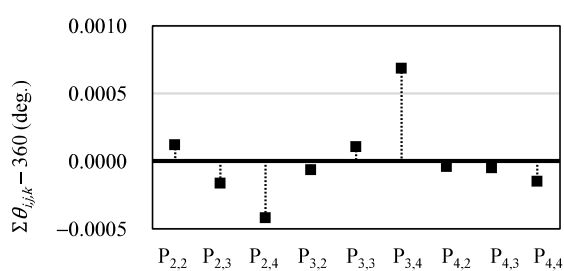


Fig. 13 Sum of angles between crease lines at each vertex

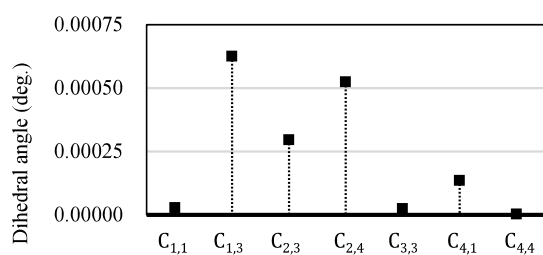


Fig. 14 Dihedral angle of facets connected to deleted crease lines

Table 4 Results of optimization

Objective function $F(\mathbf{X})$		4.438×10^{-10}
Error of $\sum \theta_{i,j,k}(\mathbf{X})$ from 360 (deg.)	Max.	6.872×10^{-4}
	Min.	-4.173×10^{-4}
Dihedral angle of assigned facets (deg.)	Max.	5.256×10^{-4}
	Min.	4.005×10^{-6}
DOF		6
Reduction of DOF from initial shape		7

大変形解析の諸条件はモデル1と同様である。解析結果を Fig. 15 および Table 5に示す。Fig. 15では部材の太さを0.2倍して表示している。Table 5のとおり、大変形解析全過程でのひずみは正側、負側ともに0.001%未満と小さな値であり、剛体折り可能な機構が得られたとみなせる結果が得られた。また、最終状態での節点の z 座標の存在範囲はスパン比で約1.0%, ライズ比で約3.4%とほぼ平面に展開されたとみなすことができる値となった。

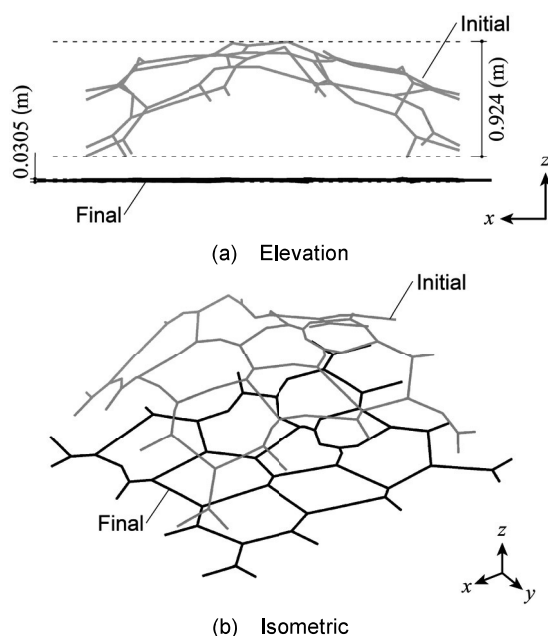


Fig.15 Large deformation analysis of model 2

Table 5 Results of large deformation analysis

Strain through analysis	Max.	9.142×10^{-6}
	Min.	-9.411×10^{-6}
Range of z coordinates of nodes (m)		3.051×10^{-2}
Ave. error of z coordinates of nodes from 0 (m)		4.019×10^{-3}

8. まとめ

本論で得られた成果は、以下のようにまとめられる。

1. 平面に展開可能な剛体折紙の形状生成と解析のためのフレームモデルを提案した。フレームモデルを用いることで、最適化、不安定次数の計算と大変形解析の全過程を同一モデル、同一変数で行うことが可能である。
2. 三角形分割した曲面を初期形状として、平面に展開可能となる

幾何学的制約を満たす多面体形状を、最適化問題を解くことにより得る手法を提案した。その際、最適化問題の目的関数に、指定した隣り合う2つの三角形平面が平行となる条件を表す項を加えることで、平行となった2面の間の折線の回転を固定し1つの四角形として剛体折りの変形自由度を低減できる。

3. 上記2の定式化は、曲面の形状や三角形分割の方法、多面体のモデル化の方法に依存しないため、さまざまな曲面に対して適切な三角形分割、モデル化の手法を選択できる。1つの例として、曲面をグリッド状に分割した際の定式化を示した。ここで、フレームモデルの節点位置に関する幾何学的制約を考慮すると、独立な変数の数は、多面体の頂点座標を変数とする場合と同じである。
4. 形状生成例として、複曲率と単曲率の目標曲面をもつ2種類の曲面に対して、最適化問題を解いて剛体折り条件と展開可能条件を満たす解を求めて、大変形解析によって平面に展開できることを確認した。本論文における手法により、単純な折線パターンと小さな折角の折線をもつ多面体が得られることを計算例で示した。これらの計算例は、斜め方向の折線の配置をランダムに発生させているため、本手法を用いてさまざまな形状を生成することが可能である。
5. 複曲率の曲面を平面に展開するには面を伸縮させる必要があるため、形状生成例では一部にしわが寄ったような形状が得られた。そのため、ガウス曲率の絶対値の大きな目標曲面に対しては、目標曲面に対する誤差制約のなかで最適化問題の解が収束しない場合があった。

参考文献

- 1) Y. Matsuoka and J. Mitani: Mathematics of flat origami, Transactions of the Japan Society for Industrial and Applied Mathematics, Vol. 27, No. 4, pp. 333–353, 2017.12 (in Japanese)
松川剛久, 三谷 純: 平坦折り紙の数理, 日本応用数理学会論文誌, Vol. 27, No. 4, pp. 333–353, 2017.12
- 2) T. Tachi: Rigid origami mechanisms, Journal of the Robotics Society of Japan, Vol. 34, No. 3, pp. 184–191, 2016.4 (in Japanese)
舘 知宏: 剛体折紙メカニズム, 日本ロボット学会誌, Vol. 34, No. 3, pp. 184–191, 2016.4
- 3) E. D. Demaine and M. L. Demaine: Recent results in computational origami, Proc. 3rd Int. Meeting of Origami Science, Mathematics, and Education (Origami3), A K Peters, pp. 3–16, 2002
- 4) J. Ma and Z. You: The origami crash box, Proc. 5th Int. Conf. on Origami in Science, Mathematics, and Education (Origami5), CRC Press, pp. 277–289, 2011
- 5) K. Shigetomi: Medical applications of origami engineering: Applications of origami folding techniques for medical devices and regenerative medicine, Journal of the Visualization Society of Japan, Vol. 33, No. 131, pp. 29–34, 2013.10 (in Japanese)
繁富(栗林)香織: 折紙工学の医療分野への応用: 折り紙の折畳み技術を用いた医療器具と再生医療への応用, 可視化情報学会誌, Vol. 33, No. 131, pp. 29–34, 2013.10
- 6) P. M. Reis, F. L. Jimenez and J. Marthelot: Transforming architectures inspired by origami, Proc. National Academy of Sciences of the United States of America, Vol. 112, No. 40, pp. 12234–12335, 2015.10
- 7) K. Miura: Method of packaging and deployment of large membranes in Space, 31st Congress of International Astronautical Federation, Paper No. IAF 80–A31, 1980
- 8) R. D. Resch: Self-supporting structural unit having a series of repetitious geometrical modules, United States Patent No. 3,407,558, 1968.
- 9) T. Tachi: Origamizing polyhedral surfaces, IEEE Transactions on Visualization and Computer Graphics, Vol. 16, No. 2, pp. 298–311, 2010
- 10) E. D. Demaine and T. Tachi: Origamizer: A practical algorithm for folding any polyhedron, Proc. 33rd Int. Symposium on Computational Geometry (SoCG 2017), pp. 34:1–34:15, 2017
- 11) T. Tachi: Generalization of rigid-foldable quadrilateral-mesh origami, Journal of the International Association for Shell and Spatial Structures, Vol. 50, No. 3, pp. 173–179, 2009
- 12) Y. Zhao, Y. Endo, Y. Kanamori and J. Mitani: Approximating 3D surfaces using generalized waterbomb tessellations, Journal of Computational Design and Engineering, Vol. 5, pp. 442–448, 2018.10
- 13) T. Nozima and I. Hagiwara: Geometrical Mathematics for Origami and Its Industrial Application, Kyoritsusyuppan, 2012 (in Japanese)
野島武敏, 萩原一郎: 折紙の数理とその応用, 共立出版, 2012
- 14) E. D. Demaine and J. O'Rourke: Geometric folding algorithms: Linkages, Origami, Polyhedra, R. Uehara (Trans.), Kindaigakakusya, 2009 (in Japanese)
E. D. ドメイン, J. オルーク: 幾何的な折りアルゴリズム リンケージ, 折り紙, 多面体, 上原隆平 訳, 近代科学社, 2009
- 15) S. Tsuda, M. Ohsaki, S. Kikugawa and Y. Kannno: Analysis of stability and mechanism of frames with partially rigid connections, Journal of Structural and Construction Engineering (Transactions of AIJ), Vol. 78, No. 686, pp. 791–798, 2013.4 (in Japanese)
津田勢太, 大崎 純, 菊川翔平, 寒野義博: 部分剛接合骨組の安定性評価とメカニズムの解析, 日本建築学会構造系論文集, Vol. 78, No. 686, pp. 791–798, 2013.4
- 16) M. Ohsaki, S. Tsuda, and Y. Miyazu: Design of linkage mechanisms of partially rigid frames using limit analysis with quadratic yield functions, International Journal of Solids and Structures, Vol. 88–89, pp. 68–78, 2016.6
- 17) J. O'Rourke: How to Fold It: The Mathematics of Linkages, Origami, and Polyhedra, R. Uehara (Trans.), Kindaigakakusya, 2012 (in Japanese)
J. オルーク: 折り紙のすうり リンケージ・折り紙・多面体の数学, 上原隆平 訳, 近代科学社, 2012
- 18) D. F. Watson: Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes, The Computer Journal, Vol. 24, pp. 167–172, 1981.1
- 19) Architectural Institute of Japan (Ed.): Introduction to Design Computing: Python Programming for Generation, Simulation and Optimization of Architectural Form and Function, Koronasya, 2017 (in Japanese)
日本建築学会 編: デザイン・コンピューティング入門 Python による建築の形態と機能の生成・分析・最適化, コロナ社, 2017
- 20) ABAQUS 2016 Documentation, SIMULIA, 2015

FORM GENERATION OF RIGID-FOLDABLE ORIGAMI STRUCTURE USING FRAME MODEL

Kentaro HAYAKAWA^{*1} and *Makoto OHSAKI*^{*2}

^{*1} Grad. Student, Dept. of Architecture and Architectural Eng., Kyoto Univ.

^{*2} Prof., Dept. of Architecture and Architectural Eng., Kyoto Univ., Dr.Eng.

Various methods exist for analysis and design of rigid-foldable origami. However, the crease pattern generated by an existing general method is often too complicated to apply to deployable structures in architecture. Furthermore, it is difficult to obtain various crease patterns using typical crease patterns such as generalized Miura-ori or waterbomb tessellations. Thus, we present a method for generating a rigid-foldable polyhedron that approximates a curved surface with a simple crease pattern. We consider developability and rigid-foldability of polyhedron, but we do not consider flat foldability.

Sums of angles between adjacent crease lines around each interior vertex of a developable polyhedral origami are equal to 2π . However, it is difficult to directly obtain a shape of the polyhedron which satisfies this condition. Therefore, we formulate an optimization problem that minimize the sum of errors of angles at vertices. Optimization starts from a triangulated curved surface to be approximated by a polyhedral origami, and constraints are sequentially assigned to reduce the degree of freedom (DOF). A polyhedron with quadrilateral flat facets is generated by assigning condition so that the specified pair of adjacent triangle facets have parallel normal vectors, and by removing the crease lines between them. The existence of infinitesimal folding mechanism is confirmed by singular value decomposition, and large-deformation analysis is carried out to confirm that the polyhedron can be continuously developed to a plane and rigid-foldable.

A frame model enables to use the same variables in form generation, evaluation of kinematic indeterminacy, and large-deformation analysis using a general finite element analysis software. In a frame model, a polyhedral origami is modeled with frame elements connecting the node on a crease line or an outer edge and the node in a facet. Frame elements are connected by hinges on crease lines and rigidly connected in facets. Variables of a frame model are coordinates of nodes on crease lines and outer edges, and we can arbitrarily define coordinates of nodes in facets. In fact, coordinates of nodes on crease lines and outer edges are not independent of each other. Constraints should be satisfied such that both ends of the edge shared by adjacent triangle facet meets at the same points. Therefore, we can reduce the number of variables, and it is proved that the reduced number of variables is the same as the number of coordinates of the polyhedron's vertices.

The optimization problem for form generation formulated above is independent of a method for triangulation of the target surface or a method for modeling a polyhedron. As an example, we show a method to triangulate a curved surface in a form of grid. In this method, we arrange division points in grid pattern and connect them vertically, horizontally, and diagonally. We applied the optimization method and this triangulation method to two kinds of curved surface with positive and negative Gaussian curvature, respectively. We triangulated them into 4x4 grids and obtained several optimized shapes which have different DOFs according to the number of removed crease lines. It has been confirmed by a large-deformation analysis that the 6-DOF polyhedron whose seven crease lines are removed is developable and rigid-foldable.

It has been concluded from the numerical examples that polyhedra with various shapes can be generated using the proposed optimization. The DOF of mechanism can be easily reduced by assigning constraint to remove the crease lines to generate quadrilateral faces. The frame model is very useful to use the same variable through the all process of obtaining a developable and rigid-foldable polyhedron.

(2018年12月9日原稿受理, 2019年2月25日採用決定)