

紙の物理特性を考慮した2次元展開図からの対話的なペーパクラフトシミュレーション

佐藤 崇雄^{††} 藤澤 誠[†] 三浦 憲二郎^{†††}

この論文では、紙の可展面の性質を考慮した3次元形状変形シミュレーション手法と、それによるペーパクラフト組み立てアプリケーションを提案する。提案手法は、紙の引張強さと座屈強さの性質を用いて節点座標を変数とする目的関数を定義し、最小二乗法を用いて最小化することによって、新たな節点位置を算出する。さらに、可展面の中から実際の変形において出現しやすい円柱形状と円錐形状をメッシュ変形の初期形状として用い、紙の性質をより良く再現した。また、開発したアプリケーションは、3次元モデルから作られた展開図の入力とテクスチャ画像をもとにしたユーザによるメッシュ生成の両方に対応することで、ペーパクラフトの作成における手間を軽減する。

Interactive Papercraft Simulation From 2D Papercraft Pattern Based on the Physical Property of Paper

TAKAO SATO,^{††} MAKOTO FUJISAWA[†] and KENJIRO T. MIURA^{†††}

This paper proposes a three dimensional shape deformation simulation method based on a developable surface that is the character of paper and applies the method to a papercraft assembly application. We define an error function in which the node positions are assumed to be variables by using the characters of paper, tensile strength and buckling strength. Updated node positions are calculated by minimizing the error function using the least-square method. Moreover, our application reduces time in making the papercraft by corresponding to both of the input of the papercraft pattern made from the three dimension model and the mesh generation by the user based on the texture image.

1. はじめに

紙を用いて立体形状を表現するペーパクラフトは、その手軽さからホビーを中心として、幅広い分野で用いられている。ペーパクラフトはペーパクラフト作家によって作られた展開図を書籍で購入する、もしくは、展開図データをインターネットからダウンロードし、組み立てるのが一般的である。これは、作成したい3次元形状から展開図を生成することは多くの経験がないと難しいからである。しかし、3次元ポリゴンモデルから展開図を自動的に生成する研究⁸⁾ やアプリケーション¹¹⁾ により、誰でも2次元の展開図データを作成できるようになってきている。しかしながら、2次元の展開図データを実際に印刷し、組み立てた時、思い描いていた形状と異なることがある。これは特に、単純な多面体構造のものではなく、曲面を多く用いたモデルに多い。元の3次元ポリゴンモデルは多面体で近似されており、展開図もまた多面体を仮定して生成されているからである。

3次元モデルの変形・操作を行う研究として、skeletonを用いた変形⁶⁾、Free-Form Deformation(FFD)⁹⁾、表面形状の微分を保存する変形⁷⁾、離散ラプラスアンによって曲率を保存する変形^{1),10)} などがあげられる。これらは、モデルの特徴を保存する変形であり、材質の物理特性に基づいた変形ではない。物理学に基づいた変形手法としては、バネ質点モデルに基づくモデルの操作³⁾、有限要素法(FEM)に基づくもの²⁾ などがあげられる。また、モデルの「硬さ」を表現した2次元の変形手法として、Igarashiらの研究⁵⁾ があげられる。これらの手法もまた、紙の引張強さ、座屈強さ、可展面などの性質を変形に考慮することはできない。

本研究では、紙の物理的な特性に着目し、その特性を考慮した変形シミュレーションにより、コンピュータ内でペーパクラフトを組み立てるアプリケーションを開発することを目的とする。ペーパクラフトは従来の手法と同様に、ポリゴンを用いてモデル化し、ユーザのマウスによる節点操作に基づき、可展面の性質を考慮した変形手法により変形させる。提案する変形手法では、紙の引張強さと座屈強さの性質を用いて節点座標を変数とする目的関数を定義し、最小二乗法を用いてこれを最小化し、新たな節点位置を算出する。さらに、可展面の中から実際の変形において出現しやすい円柱形状と円錐形状をメッシュ変形の初期形状として用い、紙の性質をより良く再現する。また、開発したアプリケーションでは、3次元モデルから作られた展開図の入力とテクスチャ画像をもとにしたユーザによるメッシュ

[†] 奈良先端科学技術大学院大学情報科学研究科

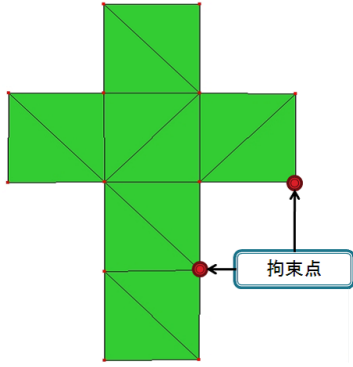
Graduate School of Information Science, Nara Institute of Science and Technology

^{††} スズキ株式会社

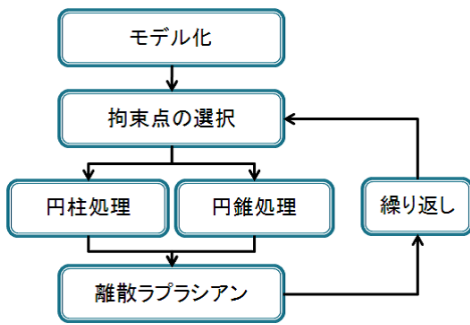
Suzuki Motor Corporation

^{†††} 静岡大学創造科学技術大学院

Graduate School of Science and Technology, Shizuoka University



(a) モデルの操作例



(b) 処理手順

図 1 システム概要

生成の両方に対応することで、ペーパクラフトの作成における手間を軽減する。

2. システム概要

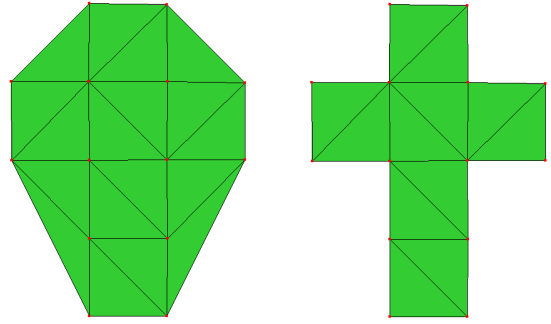
本研究で開発したシステムは、3次元空間上に配置した展開図の頂点をマウスで操作することで、ペーパクラフトの組み立てをシミュレーションする(図1(a))。このとき、マウスで選択された点を拘束点と呼ぶ。

図1(b)に全体の処理手順を示す。まず、紙を三角形メッシュでモデル化する。モデル化は以下の2手順で実行する。

- (1) 画像データをもとにユーザが頂点を指定し、ドローネー三角形分割によりメッシュ化(図2(a))
- (2) メッシュをシミュレーション部分に入力

(1)の手順によるメッシュ化では、全体の形状が凸包となってしまうため、凹形状や穴などに当たるメッシュをユーザが手動で削除する(図2(b))。(2)では、(1)で作成されたメッシュと元の画像情報を読み込み、シミュレーション部分に渡す。このように手順を2つに分けることで、(1)をスキップして、(2)だけ実行することで、あらかじめ作成して保存していたメッシュも同様に読み込むことができる。

シミュレーション部分は三角形メッシュを読み込むと、3章で説明する変形手法を用いて、図1(b)に示したように、拘束点の情報をもとに変形を実行する。変形は、紙の力学特性から引張強さと座屈強さに基づいた変形を行う。さらに、円柱処理と円錐処理の2つでモデル全体の可展面の性



(a) 三角形メッシュ生成

(b) 削除整形後

図 2 モデルの三角形メッシュ化

質を表現する。

3. 変形処理

3次元モデルの変形手法として、メッシュモデルの各頂点において離散ラプラシアンを計算し、その値を保存するようにすることで、元の形状を反映した変形が研究されている¹⁰⁾。本研究で扱うモデルは2次元の展開図を3次元空間に配置したものであり、最終的な組み立て形状はもとの3次元形状を保存したものではないため、我々は紙の力学特性から引張強さと座屈強さに基づいてモデルを変形する。

3.1 引張強さ

図3(a)はモデルの2節点の変形前と変形後の様子を模式的に示したものである。変形前における稜線両端の2つの節点座標を v_0, v_1 とし、変形後の座標をそれぞれ v'_0, v'_1 とすると、変形前後での節点間の距離の変化量は $\| (v'_0 - v'_1) - (v_0 - v_1) \|$ となる。この節点間の距離の変化量をエラーとする。

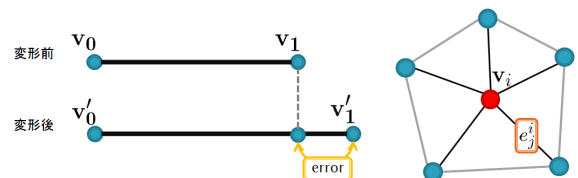
図3(b)より、 i 番目の節点 v_i と稜線を構成する j 番目の節点 v_j との節点間の長さを e_j^i とし、節点 $i-j$ 間の長さを $D(e_j^i)$ とすると、 v_i に関する節点間の長さは、

$$D(e_j^i) = \| v_j - v_i \| \quad (1)$$

で与えられる。硬い物体はこの変形前後において節点間の距離に差がない。そのため、変形後における節点の座標を v' 、稜線を e_j^i 、節点 v_i が持つ稜線の数 n とすると、

$$E(v_i) = \sum_{j=0}^n \| D(e_j^i) - D(e_j^i) \|^2 \quad (2)$$

の節点 v_i に関するエラー関数を定義する。



(a)

(b)

図 3 引張強さのモデル化

3.2 座屈強さ

座屈強さは圧縮力が加わったときの折れ曲がりに対する

強さであり、折り曲げに対し紙は開くように力が作用する。前節では、紙の形状に関して節点間の長さの維持により引張強さの性質を加えたが、それだけでは紙の硬さを表現するには不十分となる。そこで、さらに座屈強さに関する条件を付加する。

紙は折り目の無い状態である方向へ曲げられた時に全体的に曲り、局所的に折り目が付くことはない。そこで、布シミュレーションで用いられる座屈を付加する。引張強さの制約式(式(2))は隣接する節点との関係の式であったが、座屈強さでは図4(a)に示すように、さらに1つ先の接続関係となる節点との長さを見る。図4(b)は横から見た形状である。この制限は節点同士が圧縮する方向に作用する時に働く。

$D(be_j^i)$ を v_i に対し座屈の関係となる v_j との節点間の長さとする。座屈強さに関する制約式は、

$$E(be_i) = \sum_{j=0}^{bn} \| D(be_j^{i'}) - D(be_j^i) \|^2 \quad (3)$$

である。

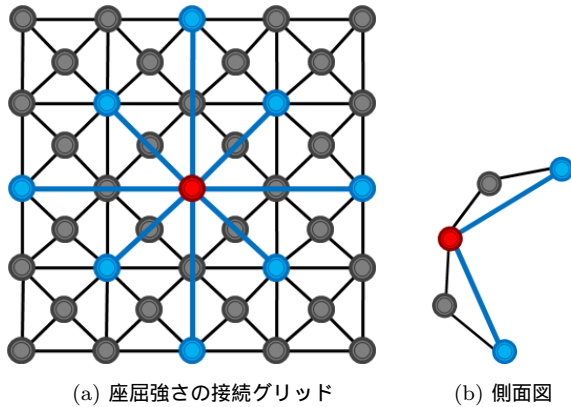


図4 座屈強さに基づく変形のモデル

引張強さ、座屈強さの両方を考慮した最終的な式は、

$$\sum_{j=0}^{bn} \| D(be_j^{i'}) - D(be_j^i) \|^2 + \sum_{j=0}^n \| D(e_j^{i'}) - D(e_j^i) \|^2 \quad (4)$$

となり、式(4)を最小化する接点位置 v_i' を算出することで、引張と座屈強さに基づく変形を行う。

4. 紙の曲面形状化

3章の変形による結果を図11に示す。図11では局所的な可展面の性質は保存されているが、全体的な形状としては代表的な可展面である円柱や円錐とはならない。図11(c)の変形の場合、実際の紙は円柱形状となるべきである。この性質を加えるために、変形時の初期形状に対して、拘束点の移動に合わせた回転などの移動量を加える。可展面が3次元空間で取りうる形状である柱面、錐面、接線曲面の3つのうち、接線曲面は正則曲線から生成されるが、その曲線上で面は正則でなく折れている⁴⁾。そのため実際には

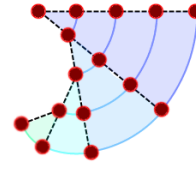


図5 回転処理

あまり利用されていない接線曲面を除き、円柱と円錐形状となるような処理を行う。これらの処理をそれぞれ、円柱処理、円錐処理と呼ぶ。

4.1 円柱処理

円柱処理は、拘束点の位置に基づきモデル全体を回転させたあと、拘束点間の距離に基づき円柱形状化する。紙は拘束点同士が近づくにつれて曲面形状が現れる。この現象を詳しく見ると、紙は拘束点が一点に集まった場合、円形形状を示し、拘束点間の距離が元の半分の場合は、半円の形状を示す。この円柱形状を決定する際に問題となるのは、接続している節点との位置関係である。曲面を滑らかな円柱形状で表現するためには、隣接する節点との角度が適切に計算され処理される必要がある。本研究では図5に示すように、一方向から順番に外角の回転量を与えることで、隣接する節点との位置関係の正誤性を保ちながら、新しい節点位置の計算を行う。

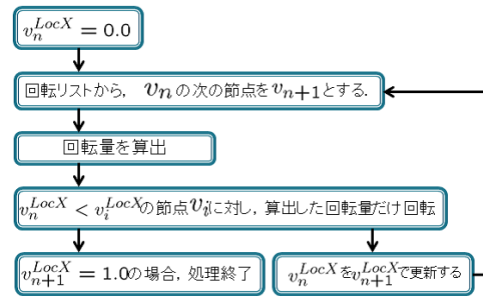


図6 回転処理のアルゴリズム

回転処理のアルゴリズムを図6に示す。はじめに、ローカル座標を定義する。ローカル座標は2点の拘束点で出来る直線をx座標、これに垂直な方向をy座標、拘束点のローカル座標 (x, y) を $(0, 0)$ 、もう片方の拘束点を $(1, 0)$ として定義する。回転はローカルx座標が0から1の方向へ順々に行う。そのため、アルゴリズムに入る前にローカルx座標の値の小さい順に節点をソートしたリストを作成する。 v_n は回転の中心となる節点であり、 v_{n+1} は回転順リストの v_n の次のローカル座標 x を持つ節点である。 $[0, 1]$ の範囲外にはそれ以上の回転が行わないように処理を行う。回転量 Deg は、

$$Deg = 360(v_{n+1}^{LocX} - v_n^{LocX}). \quad (5)$$

v_n^{LocX} は節点 v_n におけるローカルx座標の値である。

非回転領域の抽出

図7(a)のように切れ込みがある場合、ローカル座標 x が $[0, 1]$ にあっても回転の影響を受けない範囲が出てくる。そ

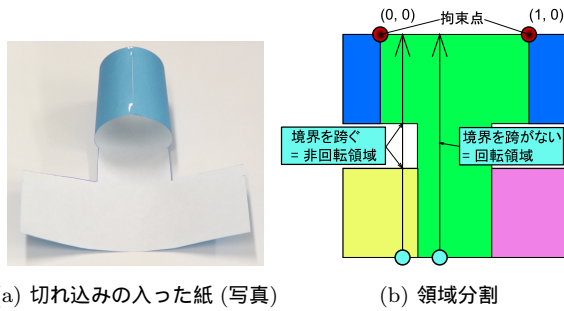


図 7 非回転領域の抽出

のため、これらの回転処理を行う前に、回転領域と非回転領域に分割する(図 7(b))。

回転領域はローカル座標 x の $[0, 1]$ の範囲であるため、その外の領域をまず非回転領域として抽出する(図 7(b) の青領域)。次に、回転範囲の各節点から、ローカル座標において、方向 $(0, -1)$ へ探索し、モデルの境界を跨ぐか跨がないかを調べる。モデルの境界を跨ぐ節点は縦方向の接続が切れるため、非回転領域になる(図 7(b) の黄と紫の領域)。

分割の区分に応じて回転処理を行うのだが、これまでの未分割時の回転処理は、回転用のローカル座標 x を 0 から順番に回転を行っていた。しかし、図 7(b) の黄色や紫色で示した領域は、緑との境界部分の回転量に依存して位置が決定される。そのため、 $0 \rightarrow 1$ への回転だけでは、黄領域の処理が適切に行われぬ。この問題を解決するために、 $0 \rightarrow 1$ の回転後に、 $1 \rightarrow 0$ の回転処理も行うことで、黄領域の補正を行う。

4.2 円錐処理

円錐処理も初期形状に対し移動量を加えることで曲面を表現する。円錐形状の処理手順は、

- (1) 円錐頂点の探索
- (2) 円錐形状の推定
- (3) 高さ方向の移動
- (4) 底面方向の移動
- (5) 回転方向の移動

である。以下でそれぞれ詳しく述べる。

円錐頂点の探索

2 拘束点に対して円錐形状の頂点を探索する。円錐の頂点部分は局所的に非可展面であり、ガウス曲率が 0 にならない。紙は可展面の性質を有するため、非可展面である円錐頂点はモデルの面上には出現しないため、円錐頂点が出現する位置はモデルの境界上となる。境界上を次の手順で探索する。(1) 拘束点 2 点間の 2 経路での境界距離を求める。(2) 短い方の経路の中間点に最も近い位置に存在する節点を円錐頂点とする。

円錐形状の推定

2 つの拘束点と探索した円錐頂点の 3 点を用いて、拘束点が結合した際に生じる円錐形状の推定を行う。まず、拘束点と円錐頂点間の長さを円錐の母線 c とし、その間の角度から円錐側面表面積 S を求める。円錐の公式から、底面半径 $r = S/(\pi c)$ である。よって、円錐の高さ $h = \sqrt{c^2 - r^2}$

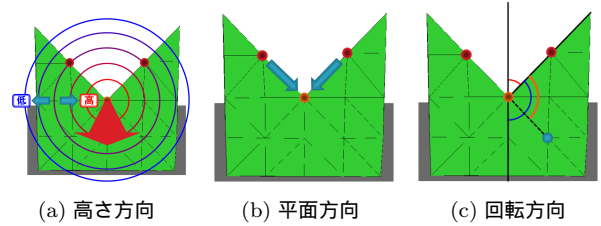


図 8 移動量

となる。これらの円錐パラメータ (c, r, h) をもとに、円錐形状化のための移動量を計算する。移動量は高さ方向、底面方向、回転方向の 3 つに分けて計算する。

高さ方向の移動(図 8(a))

円錐の底面に垂直な円錐頂点を通る軸の方向を高さ方向とし、拘束点を高さ 0、円錐頂点を高さ h とする。円錐頂点からの距離が 0 の時、高さが h であり、軸からの距離が母線 c の時に、高さは 0 となる。そのため円錐頂点の高さ方向を z 軸とすると、 z 軸から l_i の距離にある節点 v_i の高さは

$$Z_i = -\frac{h}{c}l_i + h \quad (6)$$

となる。

底面方向の移動(図 8(b))

高さ方向の処理を行った次は、底面方向に対する処理を行う。底面方向についても、頂点軸を中心座標として、高さ方向と同様に移動量は、

$$L_i = \frac{r}{c}l_i \quad (7)$$

l_i は初期位置での円錐頂点からの距離であり、そこから変形終了後の距離 L_i を計算する。この L_i を用いて円錐軸に接近する方向へ移動させる(図 8(b))。

回転方向の移動(図 8(c))

これまでの処理は、円錐頂点方向へ向って伸びる稜線の長さを維持するための処理である。そのため、円錐頂点方向へ向かわない方向を向いている稜線の長さがこのままでは維持されない。そこで、次に円錐頂点を軸とした回転方向の処理を行う。各点における回転量は、

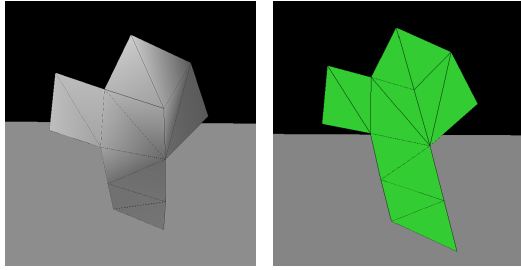
$$D_g^i = S_f \theta_R (1.0 - \frac{\theta_O}{\theta_B}) \quad (8)$$

角度 θ の添え字の R, O, B は、図 8(c) で赤、橙、青で示した角度に対応する。拘束点側に近い節点ほど回転し、そこから離れるほど回転しない。こうすることにより、円錐頂点へ向かわない方向の節点間の長さを維持する。

これまでの移動量を初期形状へ加えることで円錐処理を行う。

4.3 複数の円錐処理

2 次元可展面の展開図を用いて行う円錐形状の推定は、1 回の円錐処理を行うことにより、3 次元非可展面からの形状推定へと変化する。円錐処理では 2 つの拘束点から円錐形状の推定を行い、それに依りて移動量を加えていた。しかし、処理を 1 度行うことでモデルが 3 次元形状へ変形し、



(a) 円錐処理後の形状 (b) 仮の 2D モデル

図 9 複数の円錐処理：推定用モデルの構築

初期状態では存在しない非可展面を含むモデルとなる。前節で説明した手法は、2次元の可展面を基に円錐形状を推定しているため、同じ手法の重ね合わせを行っても、結果として推定される形状が歪む。そこで、本研究では推定用モデルを構築し、パネ-質点系を用いて歪み補正を行う。

4.3.1 推定用モデルの構築と適用

本研究では円錐処理ごとに変形に応じて仮の2次元モデルを生成し、それを元に足し合わせる円錐形状を推定する。仮の2次元モデルは、高さ方向・平面方向・回転方向の移動量推定の内、平面方向と回転方向の処理を行うことで構築する。ここで、実際の3次元モデルと仮の2次元モデルの間で、稜線の長さや拘束点と円錐頂点の間の角度が異なるため、稜線の長さは実際の長さである3次元モデルを用い、拘束点と円錐頂点の間の角度は仮の2次元モデルを用いて、次の円錐処理後の形状の推定を行う。移動量は、図9(b)の仮のモデルより算出された角度から図9(a)の実際の角度まで既に移動量を足したとみなし、それに依って移動量を加えていく。

推定用2次元モデルを用いて移動させる方向は、ペーパクラフトのデザインが描かれている方の、周囲のポリゴンの法線を平均して求めた接点の法線方向である。

4.3.2 パネ-質点系による歪み補正

推定モデルを用いても、本来の展開図を用いて円錐推定を行っていないため、そのまま重ね合わせを行っても歪みを含む形状になってしまう。そこでパネ-質点系をモデルへ適用し、歪みの補正を行う。

パネ-質点系による歪み補正は、以下の式を用いて行う。

$$\Delta P = kde \quad (9)$$

ここで e は、節点とそれに接続する節点の正規化ベクトルであり、 k は節点間に働くパネ定数、 d はその節点間のばねの伸びであり、 ΔP は微小変位である。これを3章と同様に、すべての隣接する接続と接点を1つ超えた接続に対し、求めた微小変位を足し合わせ、それを位置 P に加えることにより最終的な近隣節点との関係により位置を更新する。

4.3.3 円錐影響範囲

単一の円錐処理においては、影響範囲を考慮に入れずとも形状にほとんど歪みが生じないため問題とはならなかった。しかし複数の処理を足し合わせる場合は、歪みが生じる必要のない部分や、変形部分からより遠い位置にある変形の影響が小さい節点にも歪みが生じ、パネ-質点系にお

ける収束計算が遅くなる問題が生じる。そこで、変形による形状変形の影響範囲を検出・考慮することで、変形の影響を受けない範囲の歪み量を減らし、パネ-質点系における収束を速くする。

円錐の影響範囲は、その変形により満たされるべき形状要件が強く出る範囲である。円錐処理において満たされるべき形状要件が強く出る範囲は、「円錐頂点と拘束点間を母線とする円錐形状範囲」である。本研究では、円錐頂点と各節点を直線で結んだ時に、その直線がモデルの境界を跨ぐかどうかで、影響範囲を計算する(図9(d))。そして、円錐を足し合わせる際は影響範囲のみに移動量を加え、収束計算を行うことで形状を表現する。

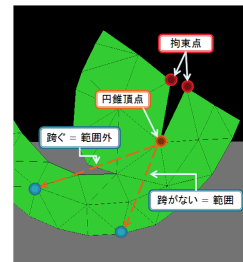
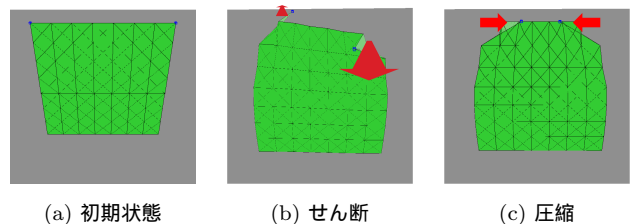


図 10 影響範囲の計算方法

5. 結果

これまでの処理を用いて、ペーパクラフト組み立てアプリケーションを作成し、コンピュータ内で展開図から組み立てた結果を示す。

まず、円柱処理および円錐処理を行った結果をそれぞれ、図12、図13に示す。これらは、各処理を1回行った際に得られた結果であるが、図12(a)は切り込みなしの正方形を斜めにねじったときの結果、図12(b)は切り込みがある形状を曲げた時の結果である。円錐処理の結果として、図13(a)の形状を上部の両端点を拘束点として接合した時の結果が図13(b)である。図13(a)において手前を向いている方を表面として処理した。これらの結果から、紙が持つ可展面の性質の柱面および錐面の形状がうまく表現されている。また、図14は、Mitaniらのbunnyの頭の部分の展開図⁸⁾をシミュレーションした結果である。円錐処理を繰り返すにより得られた結果であるが、実際のペーパクラフトと同様の形状が得られることが確認できた。



(a) 初期状態 (b) せん断 (c) 圧縮

図 11 円柱、円錐処理なしの変形

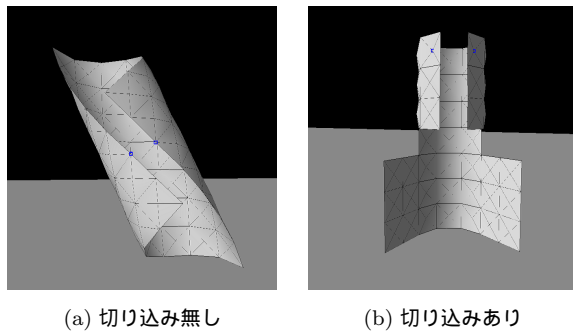


図 12 円柱処理結果

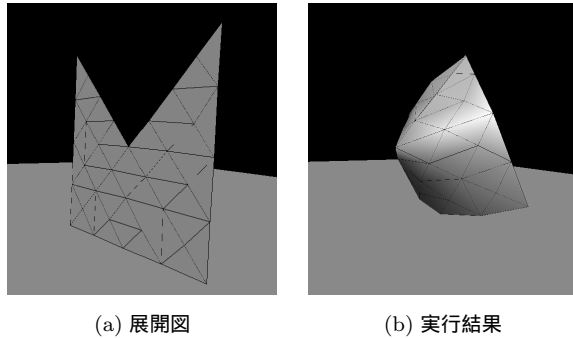


図 13 円錐処理結果

6. おわりに

本研究では、紙を三角形メッシュでモデル化し、紙の引張強さ、座屈強さ、可展面という物理的な性質に着目し、それらを考慮した変形シミュレーションにより、ユーザのマウスによる節点操作に基づくペーパクラフト組立てアプリケーションを開発した。提案手法では、紙の引張強さと座屈強さの性質を用いて節点座標を変数とする目的関数を定義し、最小二乗法を用いてこれを最小化し、新たな節点位置を算出することで変形操作を行い、可展面である円柱形状と円錐形状をメッシュ変形の初期形状として用いることで、紙の性質を再現した。そして、開発したアプリケーションを用いてペーパクラフトの組立てを行い、展開図より紙特有の曲面形状を持つ立体模型をシミュレートできた。

今後の課題としては、折り目や面同士の拘束の表現、円錐処理における凹凸方向のユーザによる選択、多数の展開図接合することによるより複雑な形状のペーパクラフトの再現などがあげられる。また、現在の手法では円柱処理と円錐処理はユーザがどちらかを選択する様になっているが、これらの自動的な判別や、円柱処理はモデル全体に対して一度しか実行できない制限があり、より複雑なモデルを表現するためにはその制限を解決する必要がある。

参考文献

- 1) Alexa, M.: Mesh editing based on discrete Laplace and Poisson models, *SIGGRAPH 2006 Courses*, pp. 51–59 (2006).
- 2) Celniker, G. and Gossard, D.: Deformable curve and surface finite-elements for free-form shape de-

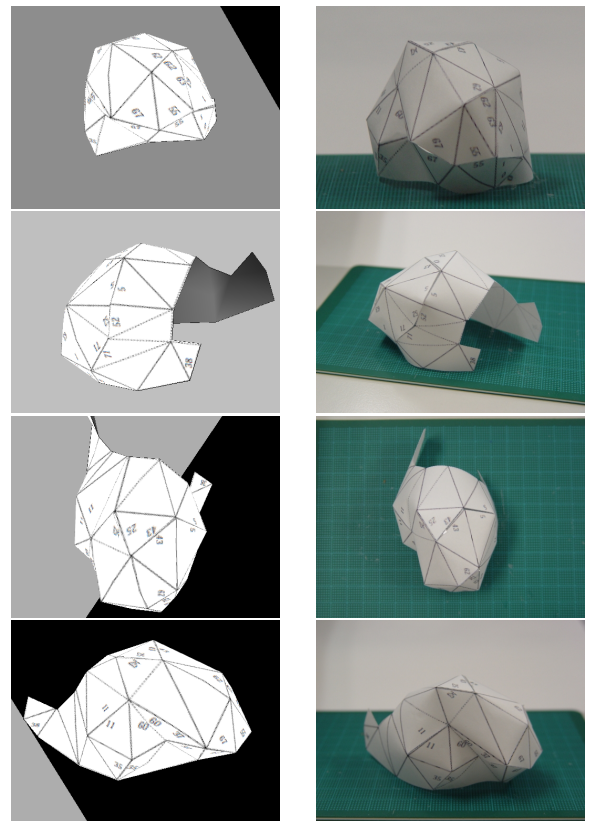


図 14 ペーパクラフトの実行結果

- sign, *SIGGRAPH1991*, pp.257–266 (1991).
- 3) Christensen, J., Marks, J. and Ngo, J.T.: Automatic Motion Synthesis for 3D Mass-Spring Models, *The Visual Computer*, Vol.13, No.1, pp.20–28 (1997).
- 4) do Carmo, M.: *Differential Geometry of Curves and Surfaces*, Prentice Hall (1976).
- 5) Igarashi, T., Moscovich, T. and Hughes, J. F.: As-Rigid-As-Possible Shape Manipulation, *SIGGRAPH 2005*, pp.1134–1141 (2005).
- 6) Lewis, J. P., Cordner, M. and Fong, N.: Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation, *SIGGRAPH2000*, pp.165–172 (2000).
- 7) Lipman, Y., Sorkine, O., Cohen-Or, D., Levin, D., Rössl, C. and Seidel, H.-P.: Differential Coordinates for Interactive Mesh Editing, *Proc. SMI2004*, pp. 181–190 (2004).
- 8) Mitani, J. and Suzuki, H.: Making papercraft toys from meshes using strip-based approximate unfolding, *SIGGRAPH 2004*, pp.259–263 (2004).
- 9) Sederberg, T.W. and Parry, S.R.: Free-form deformation of solid geometric models, *SIGGRAPH1986*, pp.151–160 (1986).
- 10) Sorkine, O., Cohen-Or, D., Lipman, Y., Alexa, M., Rössl, C. and Seidel, H.-P.: Laplacian surface editing, *Proc. SGP2004*, ACM, pp.175–184 (2004).
- 11) Tama Software Ltd.: ペーパクラデザイナー, <http://www.tamasoft.co.jp/pepakura/>.