

組立ての容易さと曲げ変形を考慮した3Dポリゴンモデルの展開図生成

和田 弘重[†]

高井 那美[‡]

高井 昌彰[§]

本研究では、計算機で作成される仮想的な3次元ポリゴンモデルを、実際に手で触ることが出来るように実体化することを考える。すなわち、対象となるポリゴンモデルを平面に展開して得られる展開図を設計図とし、これをペーパークラフトの要領で組み立てることで対象を実体化する。この際、展開図の組み立てはユーザがハサミや糊を利用して行うため、組み立ての難易度は展開図の形状に左右される。そこで本論文では、ユーザによる組み立ての容易さを考慮した展開図を自動生成することを目的とする。また紙の曲げ変形特性にも着目し、曲げ変形による滑らかな曲面を含む展開図生成方法を示し、実際のポリゴンモデルへの適用例を示す。

Development Diagram with Curved Surfaces for 3D Polygon Models

Hiroshige Wada[†]

Nami K. Takai[‡]

Yoshiaki Takai[§]

In this paper we focus on a paper craft model which is a way of realization of 3D polygon models defined in a virtual space. People can construct such a paper craft model usually by using scissors and glue. Hence difficulty or complexity in constructing the paper craft model depends on the 2D shape of the development diagram of an objective 3D polygon model to be realized. In this paper we present a heuristic search algorithm for generating the effective development diagram in consideration of easiness of the model's construction work. In addition, we focus on a paper's smoothly curved surfaces, and apply it to the development diagram of 3D polygon models which may originally have some curved surfaces. We show some application examples of the simple paper craft with curved surfaces, and discuss the effectiveness of our approach.

1. はじめに

本研究では3次元ポリゴンモデルを解体して展開図を作成し、それを設計図としてペーパークラフトによるモデルの実体化を行うことを考える [1]。展開図の組み立てはユーザがハサミと糊を使って行うものとする。したがって、展開図の形状に依存して作成に要する時間や難易度が変わってくる。本稿では、どのような展開図が組み立て易いかを考慮し、その展開図を自動生成する手法を提案する。

先行研究 [1] では、展開図を用紙から切り出す長さ、展開図のバウンディングボックスの面積、パーツの数を考慮した展開図生成を行っている。本研究ではこれらに加え用紙の形状及び、余白面積を考慮に入れた展開図生成を考える。

また一般に円錐や円柱など丸みのあるモデルをポリ

ゴンで表現する際には角錐や角柱に近似されるため、その展開図を組み立てても本来の丸みを表現することはできない。本稿では丸みを直接表現するために紙の曲げ変形を考慮にいった展開図生成についても考案する。

2. 展開図

2.1 定義

3次元ポリゴンモデルの各ポリゴンを一平面に移したものを、そのモデルの展開図とする。平面に移したときの各ポリゴンの配置は任意に決めることができ、それにより様々な展開図が可能である。

展開図を作成する際は三角形ポリゴンをベースに考えるため、モデルに多角形ポリゴンが含まれる場合はそのポリゴンを三角形ポリゴンに分割して考える。

2.2 面の隣接関係

展開図を組み立てる最も単純な手法は、全ての三角形を切り出して対応する辺同士を糊で繋げるものである。しかしモデル内で隣接する面同士を別々に切り出

[†]北海道大学大学院工学研究科

[‡]北海道情報大学経済情報学部

[§]北海道大学情報基盤センター

[†]Graduate School of Engineering, Hokkaido University

[‡]Hokkaido Information University

[§]Information Initiative Center, Hokkaido University

してから繋げることは非常に効率が悪い。隣接する面同士は、予め展開図を作成する際に繋げておけば、切り出す手間も繋げる手間も軽減される。このような面の連結関係を考慮して展開図を作成する必要がある。

2.3 連結情報の経路図

経路図とは、ポリゴンモデルのデータを元に面をノード、稜線を枝として作られる無向グラフである(図1右上参照)。この経路図で経路「A → B → C → F → E → D」を通った場合の展開図は図1右下のようになる。つまり全てのノードを通る経路がその元となるモデルの展開図となる。

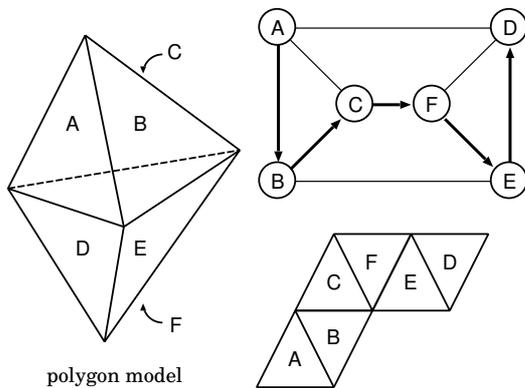


図 1: 経路図の作成

この例のように、全てのノードを一つの経路で通過する必要は無いが、先に述べたように展開図は出来るだけ面を繋げる方が望ましい。しかし、3次元モデルのポリゴン面を2次元平面上に展開する過程で、面同士が平面上で干渉し展開不可能な場合がある。このような場合はノードを結ぶ枝を通過不能経路として処理する。このため最終的に一回の探索のみで通過できないノードが生じる場合がある。その際は、未通過のノードから新たな始点を選び再び探索を行う。なお、このような場合展開図は複数のパーツで構成されることになる。以上の操作を全てのノードを通過するまで繰り返す事で展開図が生成される。

2.4 探索アルゴリズム

前述で述べた経路図を探索するアルゴリズムとして深さ優先探索と幅優先探索を考える [2]。実際に球のポリゴンモデルをこの2種類の探索方法を用いて展開した例を図2に示す。図に示されるように探索の方法を変えるだけで得られる展開図の形状は全く異なってくる。この展開図をユーザの組み立て易さという観点

から比較すると右の展開図の方が良い展開図であるといえる。次の章ではこの比較基準について説明する。

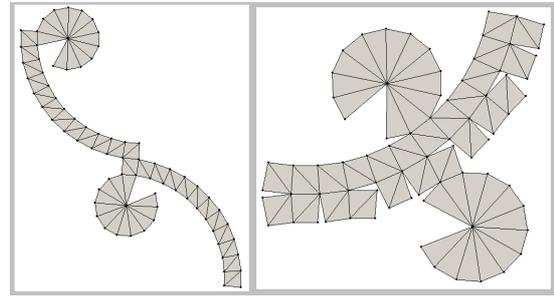


図 2: 深さ優先探索(左)と幅優先探索(右)

3. 展開図の評価

どのような展開図が組み立て易いかを考えるためには、組み立て易さを定量的に評価する必要がある。そして、その評価を元に面同士を繋げていくこと(経路探索)で最終的に組み立て易い展開図が得られる。そこで展開図の評価に用いる要素について説明する。

3.1 切り出す長さ

先ず一つの評価方法として、切り出すために必要な長さを考える。当然、長い方が手間がかかるため作り易さの評価は低い。切り出す長さを減らす事は、面同士を繋げる辺(接続辺)の合計を長くする事と同義である [4]。できるだけ長い接続辺で面を繋げて行けば、切り出す長さをより減少させることができる。また、切り出す長さを減らす事は組み立てる際の糊付けの手間を減らす事にもつながる。

3.2 用紙の余白

次に展開図を印刷した際用の用紙の余白部分の大きさを考える。余白を減らすことは展開図をコンパクトにする事につながる。これにより展開図をより拡大して印刷することも可能になる。また余白を考えるにあたって用紙の形状も考慮に入れ、展開図はバウンディングボックスの縦横比が一般の用紙の比率である $1 : \sqrt{2}$ となるように回転して配置する。これにより用紙の形状を最大限活用することが可能となる [5]。

3.3 経路の分岐

探索の際の経路の分岐も考慮に入れる。分岐が多い経路を辿った展開図は全体的に広がった(まとまった)形状となる(図2右参照)。逆に分岐が少ない経路の場合は、特定の方向に伸びた形状となってしまう(図2左参照)。このような場合は余白も非常に大きくなり

用紙を有効利用できない。

一般に図2左のような形状の場合、糊付けを必要とする面同士の紙面上での距離が大きくなる傾向がある。これは組立て時の切り出し・折り曲げなどによる誤差が蓄積しやすく、不具合の原因となる。それに対して図2右のように分岐が多く全体がまとまった形状の展開図では、糊付けを必要とする面同士が近くに生成されるため誤差が蓄積しにくい。

4. 経路図への評価の導入

4.1 評価の動的変化

前述の評価を探索中の展開図に適用し、それを経路図の重みとして与えて経路を探索していく。しかし接続辺の長さ以外の評価値は動的に変化しており、また重視する評価の方法も経路探索の進行状況に応じて変わってくる。そこで評価を数値化した評価値を次のように定義する。

4.2 評価値の定義

探索の進行状況(面の追加数)に応じて、序盤、中盤、終盤の3段階にわけて評価を変える。

序盤に余白を埋めるような探索を行っていくと、面が増えてきた後半に面同士の交差による経路の行き止りが生じてしまう。そこで序盤は余白を埋める事を重視しない評価を行う。同様に分岐をすることも控える。それに対して終盤は前半に生じた余白を埋めていく事を重視し、分岐も多く行うような評価に切替える。

以上の評価戦略に基づき、経路の評価値を次の様に定義する。

$$(\text{評価値}) = (\text{接続辺の長さ}) \times (\text{余白評価値}) \times (\text{分岐評価値})$$

すなわち、常に一定の評価値である接続辺を評価ベースに考え、その値に余白と分岐によるボーナスを乗じる事で評価の動的変化を表現している。ただし $1.0 \leq \text{余白} \cdot \text{分岐評価値} \leq 2.0$ である。なお、本稿では余白と分岐による評価をまとめて形状の評価と呼ぶことにする。

このボーナス値は前述の3段階の評価変化を考慮して探索の進行段階に応じて変化させる。ボーナスの獲得条件は、その経路が余白を埋める経路である場合とする。同様に分岐を行う経路の場合もボーナスを得る。なおその際に得られるボーナス値は探索の進行状況に応じて徐々に大きくなるように設定する(図3参照)。

4.3 多角形ポリゴン

冒頭で述べたとおり、元モデルのポリゴン面が多角形ポリゴンであった場合は三角形ポリゴンに分割して

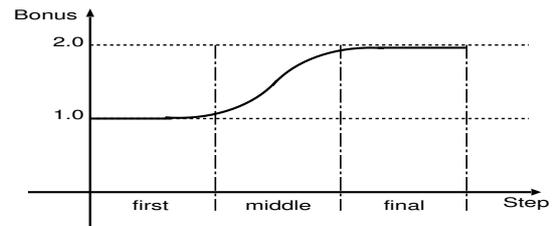


図3: 探索進行に伴うボーナスの変化

考えているが、経路探索の際にはこれらの面を優先的に繋げる探索を行う。またそれ以外の場合も隣接面がほぼ同一平面とみなせる面同士は、組み立てる際に折る必要が無いため、なるべく分割されないようにする。なお、実際に展開図を印刷する際には折る必要の無い稜線は点線等で描くことで明示的に表現する。

5. 結果

5.1 形状評価の適用結果

動的に変化する評価値をもとに実際に球状のポリゴンモデル(図4左上)を展開した結果を図4に示す。実装はJavaを用いて行った[3]。

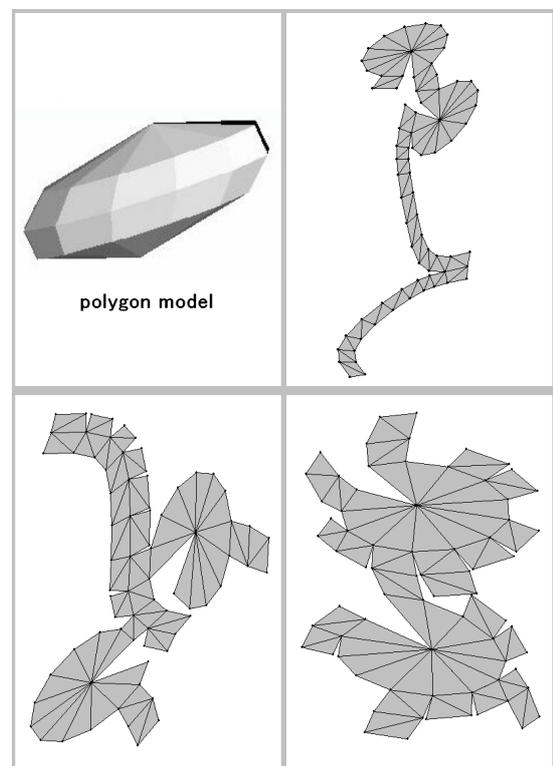


図4: 評価の動的変化を用いた展開図

図4右上 深さ優先探索を用いて生成した展開図である。深さ優先探索は幅優先探索と比べると、切り出す手間、用紙の有効利用のどちらの点でも劣っている。また深さ優先探索のメリットは計算量の少なさであるが、本研究では人間が組み立てることを前提としているためポリゴンモデルのポリゴン数は多くとも300程度と考えている。したがって計算量によるメリットは評価に入れない。

図4左下 形状による評価を行わずに、接続辺の長さのみで評価を行った結果である。なお、探索アルゴリズムは幅優先探索を用いた。先の深さ優先探索の展開図に比べて展開図全体がコンパクトになり、用紙も有効利用されていることが確認できる。以降は特に明記しない限り幅優先探索を用いて探索を行う。

図4右下 上記に加えて形状の評価を適用した結果である。左下の図と比べても更に用紙を有効利用しており、組み立て易い形状になっていることがわかる。

5.2 形状評価と余白面積

図5のグラフは前述の図4左下及び右下と同条件の探索を探索開始面を変えて行い、その結果得られた展開図の余白面積を比較したものである。横軸は探索開始面の面番号をとり、縦軸にはその開始面によって得られた展開図の余白の面積を正規化したものをとっている。実線は形状評価を用いた場合の結果で、点線は形状評価を用いなかった場合の結果を示している。グラフに示される通り、どの面から探索を行っても形状評価を用いた展開図の余白は半分程度に減らすことできている。

5.3 評価の動的変化とパーツ数

図6に示すような複雑なモデルの場合、展開図は一般に複数のパーツによって構成される。この例では胴体と両耳の3つのパーツで成り立っている。なお、両耳のパーツは手で紙面余白に配置している。

先に述べた評価の動的変化がパーツの数に与える影響を調べるために、このモデルを用いて比較実験を行った。実験はまず探索開始面をランダムに20個選び、それぞれに動的評価を用いた場合と用いなかった場合(静的評価)の探索を行い展開図を生成した。静的評価では図3で示されるような獲得ボーナスの変化をさせずに、常に2.0とした。得られた展開図のパー

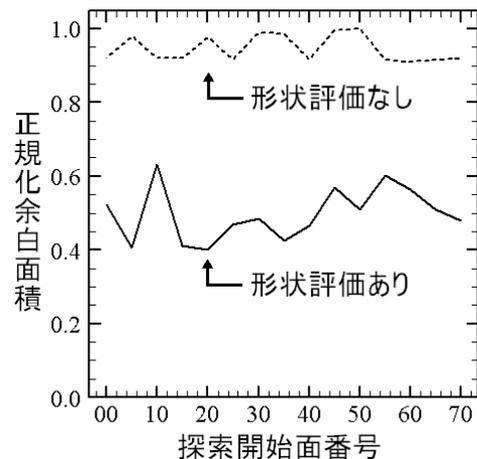


図5: 余白面積の比較

ツ数の最小値・最大値・平均値を調べた結果は以下の表の通りである。

評価	標本数	最小値	最大値	平均値
静的	20	7	11	8.1
動的	20	3	8	6.5

表1: パーツ数の比較

パーツ数の最小値・最大値・平均値のいずれも動的評価を用いた探索の方が少ない値となっており、その有効性が確認できる。

6. 曲げ変形

図7左に示されるような角錐のポリゴンモデルは、多くの場合デザイナーの意図としては円錐を近似表現するために用いたものと考えられる。しかし紙の曲がる特性(曲げ変形)を利用することで、円錐のままの展開図を生成することが可能である(図7右参照)。これにより、デザイナーの意図を反映した実体化が可能となる上、組み立て易さの観点からみても優れた展開図が生成可能となる。そのためには曲げ変形を適用可能な箇所をポリゴンモデルから検出する必要がある [6][7]。

6.1 円錐候補の検出

ポリゴンモデルから円錐近似可能箇所を自動で検出する手順について説明する。ユーザは閾値として角錐の角数 N を与える。

手順1 ポリゴンモデルの全ての頂点を調べ、その中で共有する面の数が N 以上のものを円錐候補とする。

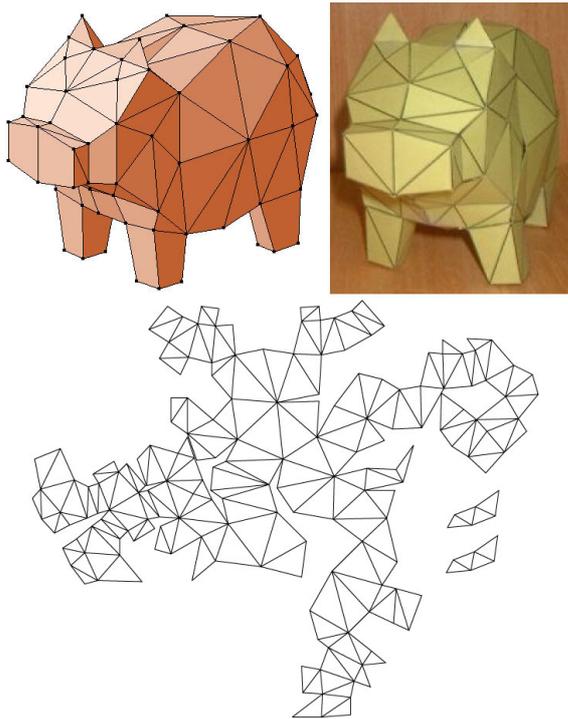


図 6: ポリゴンモデルの展開図と組み立て例

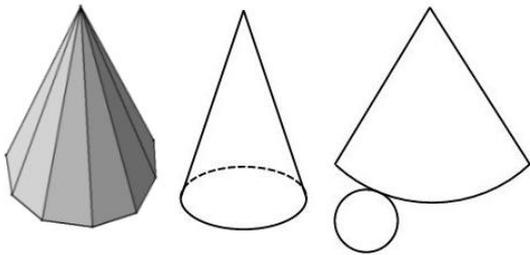


図 7: 角錐モデルと円錐の展開図

手順 2 共有面が環状でない候補を除外する。

手順 3 円錐の底面を構成することになる頂点が、ある許容範囲の同一平面上に無い候補を除外する。

以上の手順により検出されたものを円錐候補として次のステップである円錐近似に進む。

6.2 円錐近似

実際のモデルでは底面を構成することになる頂点が完全に同一平面上に存在する事は少ない(手順 3)。したがって検出時には多少の誤差を考慮して検出を行い、近似時に同一平面上に並ぶようにモデルデータを修正する。この際元モデルの他の部分になるべく影響

が及ばないように修正を行う。

次に、底面の多角形の重心を計算し、それを円錐底面の円の中心とする。ここで重心が底面の外であった場合は近似不可能として処理する。

円錐近似された部分の展開図は本体と切り離して生成する(図 8 下参照)。別パーツとして生成された本体と円錐部の展開図はそれぞれ組み立て後に接続面同士を繋げる。なお、この時に必要となる接続面(角錐の多角形)を本体側に補完しておく。

したがって底面の円は元の多角形をはみ出さない最大の円が望まれる。こうして算出された「底面の円の中心・半径・底面の法線ベクトル・円錐の頂点」から円錐を生成することができる。

6.3 円錐近似適用結果

実際に円錐近似を適用した例を図 8 に示す。図左上は元のポリゴンモデルあり、図右上はそのモデルに円錐検出&近似を適用した後である。このモデルを展開した結果は図 8 下となる。

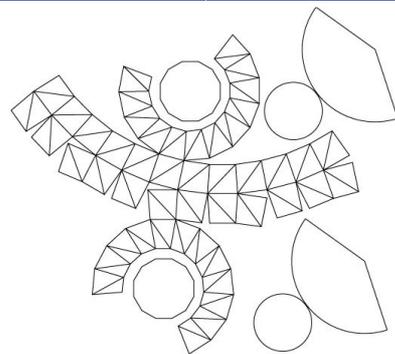
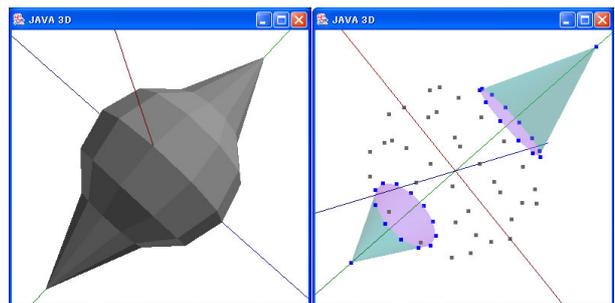


図 8: 円錐近似モデル (1) と展開図

先に示した条件のみで円錐候補を検出し、その全てに近似を適用すると、結果として意図しない形状を形成する事も考えられる。この問題は検出条件をユーザに設定させる事と、検出結果から近似適用箇所を選択させることで解決できる。検出条件としては前述の角数 N 以外に、同一平面の誤差範囲、適用部分の大きさ、

近似による誤差(角錐と円錐の体積差)などが考えられる。

またポリゴンモデル作成時に検出可能箇所を意識したモデリングを行うことで、意図的に検出させることも考えられる。例えば図9のようなモデルでは、歯の部分のポリゴン数を意図的に増やすことで角数Nのみで自動検出できるようにしている。これを実際に組み立てた結果は図10である。

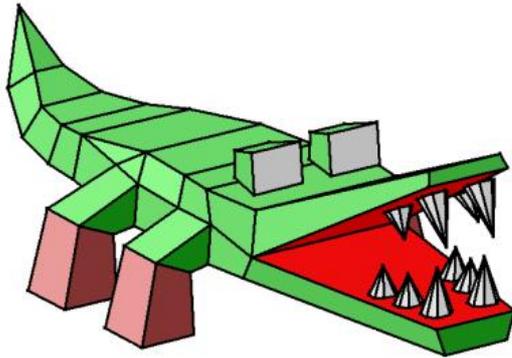


図 9: 円錐近似モデル (2)

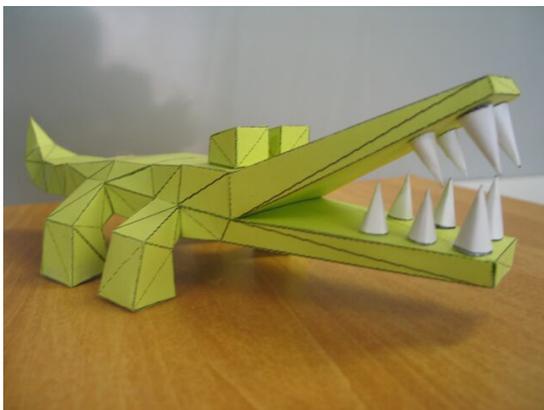


図 10: 円錐近似モデル (2) の組み立て

7. まとめと今後の展望

与えられた3次元ポリゴンモデルデータから、組み立ての容易さを考慮した動的な評価方法をもとに展開図を自動生成できることを示した。また曲げ変形に関してはモデルデータから自動的に円錐近似適用箇所を検出し、曲げ変形を利用した展開図の生成が可能であることを示した。

今後は円錐以外の曲げ変形適用可能箇所の自動検出

について検討を進めていきたい。同じ錐であっても底面の形状を円以外のもの、例えば楕円や半円などについても取り入れることで曲げ変形の適用可能箇所の幅が広がると考えている。半円などは図9のモデルの目の部分に利用することで、より効果的な実体化ができる。また、図11に示すようなモデルは円錐を多段に組み合わせることで近似可能である。これは図9の尻尾の部分などに利用ができる。更に円柱近似を用いれば足の部分の近似も可能であると考えている。

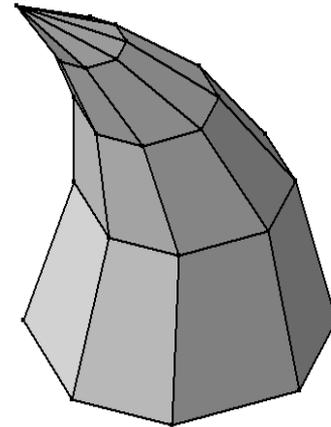


図 11: 曲がった錘

参考文献

- [1] 三谷, 鈴木, 木村: "3次元ポリゴンモデルの展開図作成", 情処技報 Vol.96-3, pp.13-18.(1999)
- [2] 茨木, 福島: "最適化の手法", 共立出版 (1993)
- [3] Henry Sowizral, Kevin Rushforth, Michael Deering: "The Java 3D API仕様", アスキー (1999)
- [4] 和田, 高井: "組み立ての容易さを考慮したポリゴンモデルの展開図作成", 電気関係学会北海道支部連合大会, p.383(2002)
- [5] 和田, 高井: "曲げを考慮したポリゴンモデルの展開図作成", 情処北海道支部シンポジウム 2003, pp.25-26(2003)
- [6] 和田, 高井: "紙の曲げを考慮した3次元ポリゴンモデルの展開図生成", FIT 2003, pp.251-252(2003)
- [7] 和田, 高井: "丸みを表す曲げ変形を考慮した展開図作成", 電気関係学会北海道支部連合大会, p.347(2003)