Crane：プロダクトの意匠性・可折性・可製造性を考慮した折紙の計算設計プラットフォーム

須藤 海 *† 野間 裕太 * 谷道 鼓太朗 † 鳴海 紘也 * 舘 知宏 *

概要. 折紙は，Human-Computer Interaction (HCI) やデジタルファブリケーションなどの学術分野および建築やプロダクトデザインなどの産業分野で活用されつつある．しかし，専門的な知識を持たない人間が (1) プロダクトに求められる意匠・(2) 計算折紙に基づく可折性・(3) 最終的に製造するための厚み処理を同時に考慮して折紙プロダクトを設計するのは難しい．特に，内部頂点を持ち複数のヒンジが連動する折紙についてこれらの問題を考慮することは，専門家にとっても困難である．そこで本稿では，Rhinoceros/Grasshopper 上に実装された折紙の設計プラットフォーム Crane を提案する．ユーザは（a）2D/3D の折りパターンを設計し，(b）その変形をシミュレートし，逆に（c）ユーザの与える制約から新たなパターンを発見することができる．また（d）厚みのない理想的な折りパターンから，ユーザの製造方法に応じて適切なヒンジが付けられた厚みのある構造を出力できる．最後に，本システムの有効性を示すべく 3 つのデザイン事例を示す．

1 はじめに

折紙は機械的な構造を実現する技術として学術と産業の両面で利用されている．例えば，ミウラ折り [19, 20] は人工衛星の太陽光パネル，吉村パターン [38] は飲料缶の耐久性向上，waterbomb pattern（ナマコ）は医療用ステント [16] などに応用されている．最近では，機械的メタマテリアル [5, 17] や自己折り・4D プリント [8, 2] にも折紙が利用されている．また，折紙の意匠を利用したランプシェード・椅子 [13] や，Al Bahr Towers [3] に見られる切紙構造を用いた可動建築，ISSEY MIYAKE の Steam Stretch [36] における複雑なプリーツなど，折紙が応用されるプロダクトも多岐にわたる．そのため，折紙のメカニズムを非専門家にも利用できるようにすることは重要な課題である．

しかし，計算により折紙プロダクトを設計・製造するためには，(1) プロダクトの適切な形状と機能を実現する「意匠性」（大きさ・水平・対称など），(2) 計算折紙の理論に基づく「可折性」（後述する可展性や平坦可折性など），(3) 最終的に現実世界で作るための「可製造性」（製造方法に応じた適切なヒンジ形状や自己交差しない材料の厚みなど）という 3 つの要求を同時に考慮する必要がある．例えば，デザイナが意匠性だけを考えてプロダクトの形状を変更すると，可折性と可製造性は簡単に失われる．このように，適切な折紙プロダクトを作る際にはデザイン・計算折紙・ファブリケーションの知識が要求され，非専門家には敷居が高い．また，折紙の専門家にとっても異なる要求を同時に実現する形状を繰り返し探索するのには多大な労力が必要となる．

本稿では，デザイナが意匠性・可折性・可製造性を同時に考慮することのできる設計プラットフォーム Crane を提案する．本環境の核となるのは，デザイナがインタラクティブに形状を探索する中で自動的に要求を満たせるような制約付きの GUI システムである．これにより，折紙の非専門家および専門家が折紙プロダクトをより簡単に実現することが可能となる．本稿では本システムの機能を示すとともに，有効性を示すデザイン事例（椅子・テーブル・ランプシェード・マスク）を紹介する．なお，本内容の詳細は ACM TOCHI で出版予定 [29] なので，適宜参照してほしい．

2 関連研究

2.1 HCI で用いられる折紙

折紙は，Human-Computer Interaction (HCI) やデジタルファブリケーションの分野においていくつかの目的に利用される．例えば，LaserOrigami はプラスチックの板を選択的に加熱することにより，3 次元の折り曲げ形状を実現する [21]．Thermorph は熱溶解積層（FDM）方式の 3D プリンタから出力される平面状のプラスチックを印刷後の加熱で意図的に歪めることにより，サポート剤を使わず高速に折紙形状を生成する 4D プリントと呼ばれる技術の 1 種である [2]．Pop-up Print は 3D プリント構造物をあらかじめ折り畳んだ状態で造形し造形後に展開することで，短い印刷時間・少ないサポート材・不必要な際の収納性を達成する [22]．その他，折紙構造はインタラクティブで動的な物体 [25, 26, 23, 24]，簡単に作製できる 3 次元の電気回路 [37]，個人の要求に合わせてカスタム可能なハプティクス [4] などに利用されている．

* 東京大学
† Nature Architects 株式会社

しかし，上記の事例に見られる折紙の多くは，内部頂点を持たない比較的単純なパターンか，既に存在する有名なパターンをそのまま利用したものに限られている．意匠性・可折性・可製造性を考慮した上で比較的複雑なパターンをユーザの要求に合わせて改変できるシステムが存在すれば，これらの研究で利用できる折紙の幅は大きく広がる．

2.2 折紙設計ソフトウェア

折紙を設計するためのソフトウェアは過去にいくつか提案されている．まず，折紙をパラメトリックな順問題として設計する手法として，Constructive なアプローチが存在する．例えば，ORI-REVO [18] は与えられた輪郭を軸を中心に捻ることで，回転対称な折紙構造を実現する．また，与えられた曲線や断面に沿う剛体折り可能な展開構造を実現する手法も存在する [32, 9]．しかし，これらの方法は形状に何らかの対称性を仮定するため，例えば与えられた任意の 3D 形状を実現することはできない．

その一方で，最適化に基づく手法を用いることで，与えられた多面体に変形する折紙を逆問題として解くことが可能である．例えば Origamizer [31] は任意の多面体を 1 枚の紙から折ることのできるパターンを出力する．同様に，最適化計算により折紙の敷き詰め（テセレーション）を目標形状にフィッティングすることで曲面を近似する手法がいくつか提案されている [30, 7, 39, 10].

特に，Freeform Origami [30] を用いると，形状が平坦な紙から折れることを保証する制約などをかけた上で（つまり可展性などを保証した状態で），2D と 3D の両方の形状をインタラクティブに操作できる．しかし，Freeform Origami で扱う要求は可折性のみであり，プロダクトデザインにおいて重要となる意匠性と可製造性の要求に答えることは容易ではない．また，必要に応じてソフトウェアの機能を組み合わせて新しい制約を定義することができない．さらに，ソフトウェアが独立しているため，必要に応じて入出力をそのまま他のプラグインなどに接続することはできず，デザイナが使い慣れたソフトウェアを使い回すこともできない．

本研究では，上記の最適化と制約に基づくシステムをノードベースのビジュアルプログラミング環境である Grasshopper 上で実現した．これにより，ユーザは可折性だけでなく意匠性や可製造性に関する要求をノードの接続により実現することができ，必要に応じて新たな制約を自分で定義したり，Grasshopper で利用できる様々なプラグインに接続したりできる．なお，Grasshopper は Rhinoceros という 3D CAD の標準機能であり，プロダクトデザイナや建築家に広く利用されている．

3 Crane

3.1 3 つのフェーズ

図 1 に示すように，Crane のワークフローは 3 つのフェーズに分類される：

1. 入力フェーズ (図 1a)
   最初の折りパターンを入力する．本システムは (1)Rhinoceros で作成した 2D/3D のパターン，(2) 現実の折紙の画像認識，(3)FOLD フォーマット [6]，(4) 有名なテセレーション（ミウラ折りなど）のテンプレート，(5) 後述の Generalized Ron Resch パターンを生成するための 3D メッシュなどを受けつける．
2. シミュレーション・形状探索フェーズ (図 1b)
   入力されたパターンに基づき，システムは変形のシミュレーションまたは幾何的な制約に基づく新たな形状探索を行う．
3. ファブリケーションフェーズ (図 1c)
   シミュレーションや形状探索が終わった後で，製造のためのデータを出力する．ユーザの使用するツール（レーザカッター，3D プリンタ，CNC 加工機）などに応じて適切なヒンジを付けた厚みのあるモデルを作る．

ユーザはこれら 3 つのフェーズを行き来して，インタラクティブに試行錯誤を行うことができる．個々のフェーズに関しては 4 章・5 章・6 章で解説するが，次の節からは折紙の椅子をデザインするシナリオを考え，問題解決の手順を示す．

3.2 ターゲットユーザ

本システムは，計算折紙の専門家に限らず折紙の知見を応用したいデザイナ・建築家・研究者などを対象とする．その一方で，Rhinoceros と Grasshopper の基本的な操作および本稿に登場するような計算折紙に関するツールの使い方の習得は前提とする．

3.3 ユースケース：椅子のデザイン

図 1 に示す木製の椅子をデザインする場合を考える．今，ユーザは意匠性に関する複数の要求を持っている．例えば，折れた状態で座面は水平であってほしい．背もたれは快適な角度であってほしい．地面に接する全ての頂点は浮くことなく接地してほしい．ある辺同士は目標の折れ角で互いにぶつかって，それ以上変形が進まないようにしてほしい．椅子は左右対称であってほしい．

一方で，収納のため椅子は展開状態で完全に平坦になってほしい．これが可折性の要求に相当する．また，意匠性と可折性の要求を満たしたとしても，ユーザは厚みがある板で作った際に板同士が自己交差しないようにしなければならない．また，使用するツールで作ることができるヒンジを選ぶ必要がある．これらが可製造性に相当する．

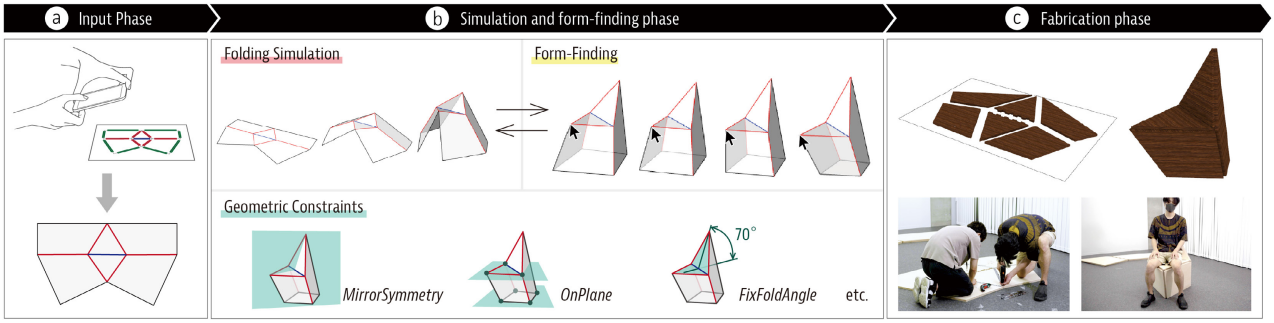

図 1. 設計ワークフロー：(a) 入力フェーズ，(b) シミュレーション・形状探索フェーズ，(c) ファブリケーションフェーズ．

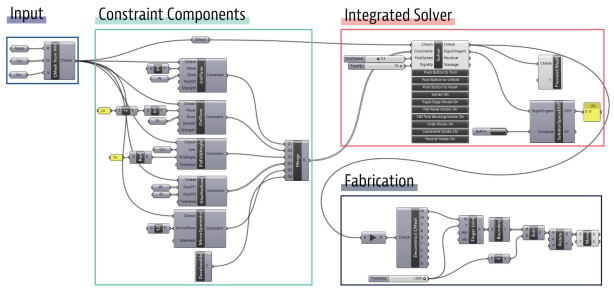

図 2. 椅子の設計に使用した Grasshopper の構造．

これら 3 種類の要求を同時に満たす Grasshopper の構成を図 2 に示す．

3.3.1 椅子 – 入力

ユーザはまず，手作業で実際の紙を折ることで大雑把に椅子のような動きをする折紙をデザインしたと仮定する．このとき，ユーザは図 3 に示すように，最終的な折パターンを赤（山折り）・青（谷折り）・緑（境界）で塗り分け，画像として Grasshopper に取り込む（図 2 の Input）．

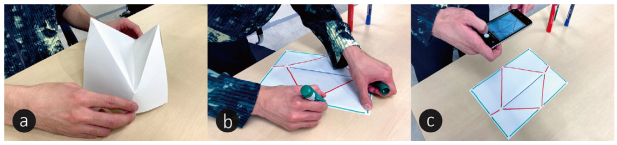

図 3. 画像認識による入力：(a) 手作業で形状を試行錯誤し，(b) 折り目と境界に色を付け，(c) カメラで撮影する．

3.3.2 椅子 – シミュレーションと形状探索

次に，ユーザは得られたパターンの変形をシミュレーションにより確認する．このとき，システムは [30] の手法で剛体折りに基づく変形シミュレーションを行う．Crane 上では，図 2 の Integrated Solver に表示されたソルバー上で「Push Button to Fold」をクリックすることで変形を確認できる．

また，ユーザはいくつかの意匠性に対する要求を制約（図 2 の Constraint Components）として与え，形状探索を行う．ここでは，意匠性に対応する 5 つの制約を与えたとする：

1. *OnPlane Constraint*: 座面の頂点を与えられた水平な平面に固定する
2. *OnPlane Constraint*: 下部の頂点を接地する
3. *FixFoldAngle Constraint*: 背もたれと座面に存在する辺のなす角度を一定の値に固定する
4. *GlueVertices Constraint*: 椅子の背面に存在する 2 つの辺がある折れ角で一致する（くっつく）ように固定する
5. *MirrorSymmetry Constraint*: 鏡映対称にする

同様に，可折性に相当する制約も接続する：

6. *Developable Constraint*: 与えられたパターンを可展（1 枚の平面から折れる状態）にする

図 4 に，上記 6 つ（5 種類）の制約を図解する．これらの制約をソルバーに接続し，ソルバー上の「Solver: ON」というボタンをクリックすることでシステムは新たな形状を探索する．なお，このときユーザはある頂点をドラッグしたり，コンポーネントを追加・削除したりすることで，2D または 3D のモデルの形状をインタラクティブに変更することもできる．その際，システムは変形したモデルに対してリアルタイムで再計算を行い，制約を満たしながら変形を提示する．

3.3.3 椅子 – ファブリケーション

シミュレーションと形状探索で適切な折りパターンを発見した後，ユーザは可製造性の要求を満たすためにコンポーネントを追加する．このとき，ユーザは CNC 加工機によって板を切り出すことを考えているため，ソルバーコンポーネントの後段にフィンガージョイントのコンポーネントを接続し，使用する板材の厚みを入力する（図 2 の Fabrication）．システムは自動的に 2D のパターンに厚み付けをして，自己交差が起きないような矩形のヒンジをつける（図

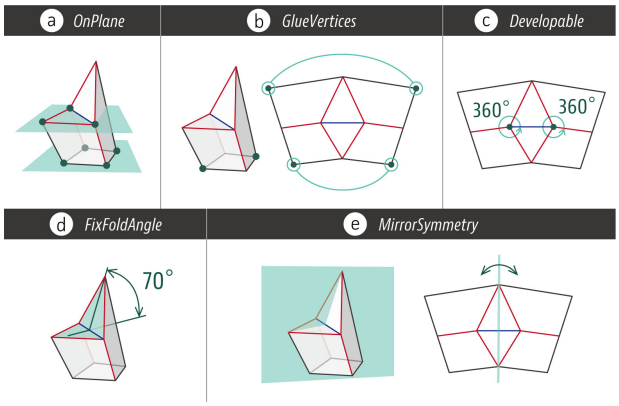

図 4. 椅子の形状探索に使用した制約.

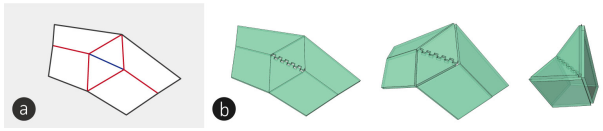

図 5. (a) 厚み付け前と (b) 厚み付け後の椅子.

5)．また，フィンガージョイントの場合，あらかじめシミュレーションした際の目標の角度で板同士がちょうど互いにぶつかり安定するようになっている．厚み付けがなされた状態でも，ユーザはシミュレーションを行い形状変形を確認することができる（図 5b）．

以降の章ではそれぞれのフェーズについて説明していくが，紙面の都合により代表例のみ解説する．

4 入力フェーズ

ここでは 3 種類の入力について述べるが，それぞれの用途は 7 章にて例示する．

4.1 画像認識

椅子の事例で示したように現実の折紙からパターンを入力する場合，ユーザは撮影した画像を PNG ファイルで Grasshopper に与える．システムは複数色の辺を区別した後，一定の距離 ε の範囲に集まる頂点を結合することで折りパターンを生成する．

4.2 テンプレート

また，有名なテセレーションをテンプレートとして呼び出すこともできる．現状では，ミウラ折り [19]，吉村パターン [38]，チキンワイヤーパターン [19]，waterbomb pattern などに対応している．

さらに，テセレーションの基本形状に限らず，図 6 に示すように 2 つのオフセットした NURBS 曲面の間を充填するようなテセレーションを自動生成することもできる（ここでは Fold Mapping [11] や [39] の手法を利用した）．なお，曲面の間を充填したパターンは多くの場合可展ではなくなっているため，1 枚の平面から折る場合は追加で前述の *Developable Constraint* を足す必要がある．

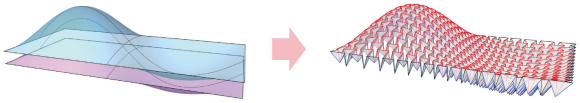

図 6. 2 つのオフセット曲面を充填するテセレーション.

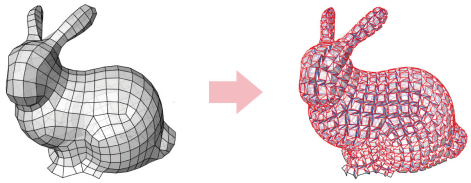

図 7. Generalized Ron Resch パターン.

4.3 Generalized Ron Resch パターン

Crane は入力として 3D メッシュも受け付ける．システムは [34] の手法に基づき入力されたポリゴンの隙間に追加の折り目を挿入し，図 7 に示すような Generalized Ron Resch パターンを生成する．なお，この場合も可展性は保証されないため，多くの場合 *Developable Constraint* をつける必要がある．

5 シミュレーションと形状探索

5.1 ソルバー

ソルバーは，CGNR 法により制約式の最小二乗解を計算する．詳しい計算方法は [30] を参照してほしい．制約のベクトル関数 $\mathbf{C} : \mathbb{R}^n \longmapsto \mathbb{R}^m$ (ただし内部頂点数を V としたとき $n = 3|V|$ で，m は制約の個数) が与えられたとき，ソルバーは方程式 $\mathbf{C}(\mathbf{X}) = \mathbf{0} \in \mathbb{R}^m$ を解く．ここで $\mathbf{C}(\mathbf{X})$ の各行は個々の制約関数に対応する．なお，多くの場合得られる解 $\mathbf{C}(\mathbf{X}) = \mathbf{0}$ は単一ではなくある解空間を持ち，システムは解空間の中から直前の形状に最も近い 1 点を選ぶ．このため制約が不十分な条件でもシステムはロバストに動作する．

5.2 Constraint Components

ここでは意匠性と可折性に対応する制約のコンポーネントについて説明する．現在 Crane には 14 つ（6 種類）の Constraint Components（5 つの *OnGeometry Constraints*，3 つの *Symmetry Constraints*，3 つの *GlueGeometry Constraints*，1 つの *FixFoldAngle Constraint*，1 つの *FlatFoldable Constraint*，1 つの *Develpable Constraint*）が存在するが，ここでは 2 種類だけを取り上げる．それぞれのコンポーネントは制約関数 $c_i(\mathbf{X})$ を出力し，

システムはそれらを行とする $\mathbf{C}(\mathbf{X}) = \mathbf{0}$ を解く．

5.2.1 OnGeometry Constraints

OnGeometry Constraints はある頂点の位置を目標形状の上に固定するものであり，以下の 5 つのコンポーネントが存在する：

1. *OnPoint*: ある頂点を点上に固定
2. *OnPlane*: ある頂点を平面上に固定
3. *OnMesh*: ある頂点をメッシュ上に固定
4. *OnCurve*: ある頂点を NURBS 曲線上に固定
5. *OnSurface*: ある頂点を NURBS 曲面上に固定

それぞれのコンポーネントは同じ考え方で実装される．制約を受ける頂点を $\mathbf{x} \in \mathbb{R}^3$ としたとき，システムは目標形状の上に存在し $\mathbf{x}$ に最も近い頂点 $\mathbf{x}' \in \mathbb{R}^3$ を計算する．このとき，制約関数 $\mathbf{C}^{\text{ongeo}}$ は

$$\mathbf{C}^{\text{ongeo}}(\mathbf{X}) = \frac{1}{l_{\text{ave}}}(\mathbf{x} - \mathbf{x}')$$

と表される．なお，この関数は初期のパターンの辺長の平均 l_{ave} で割り，スケール非依存にしてある．

5.2.2 Developable Constraint

ある折紙「のような」立体形状が与えられたとしても，それが平面から折れる（可展）かどうかは自明ではない．可展性が満たされる条件は，任意の内部頂点について，頂点周りの角度の和が 2π になることである．各内部頂点とその周りの角度を v および $\theta_{0,v}, \ldots, \theta_{n_v-1,v}$ とすると，制約関数 c_v^{dev} は

$$c_v^{\text{dev}}(\mathbf{X}) = 2\pi - \sum_{i=0}^{n_v-1} \theta_{i,v}$$

と表される．

6 ファブリケーション

図 8 に示すとおり，Crane は (a) オフセットヒンジ [33]，(b) フィンガージョイント [14]，(c) テーパードヒンジ [33]，(d)3D プリンタブルなシングルヒンジ，および (e)3D プリンタブルなダブルヒンジの 5 種類のヒンジ形状を出力できる．

あるヒンジが常に最適ということはなく，それぞれに利点と欠点がある．まず，(a)(b) はレーザカッターおよび 2 軸の CNC 加工機で作製可能であるが，(c) は 3 軸加工機か 3D プリンタでしか作製できない．パネル造形後，(a)(b)(c) はパネルを布に貼り付けたりピアノ丁番で固定したりする後処理が必要になる．一方，(d)(e) は作製後の後処理が不要であるが，3D プリントでしか作れないため他の手法に比べて小さな構造物しか作製できない．その他，(e) を除く全てのヒンジは折れ角に制限がある．(a)(d) は 180° 折りができず，(b) は 90° より大きな谷折り

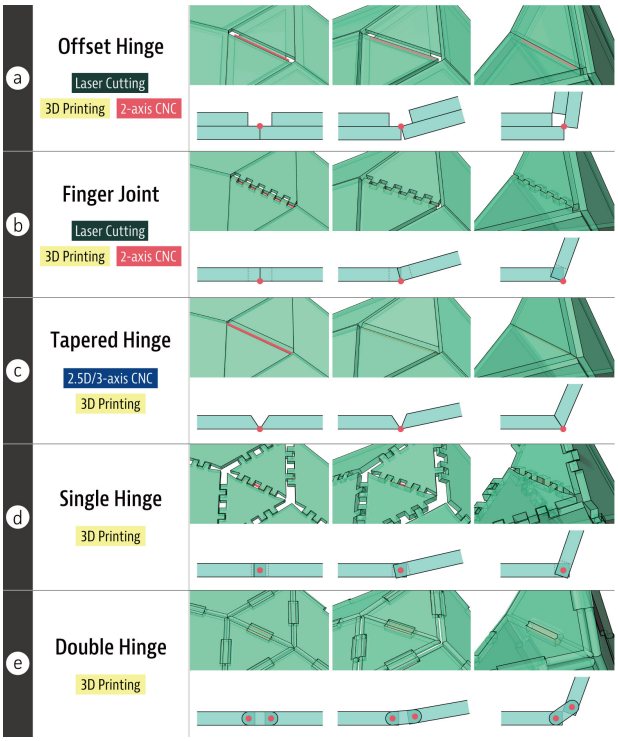

図 8. Crane で利用できるヒンジ：(a) オフセットヒンジ，(b) フィンガージョイント，(c) テーパードヒンジ，(d)3D プリンタブルなシングルヒンジ，(e)3D プリンタブルなダブルヒンジ．

が実現できず，(c) は 180° の谷折りができない．(e) は山折りでも谷折りでも 180° 折りが可能だが，厳密にはヒンジが 1 本ではなくなっているため，元のパターンと同じように扱えるとは限らない [15]．

7 デザイン事例

Crane の使用法と有効性を示すため，テーブル・ランプシェード・マスクのデザイン事例を示す．

7.1 テーブル

図 9 に Crane で作製した折紙テーブルを示す．このテーブルは不必要なときには平面に折り畳める（平坦可折である）ため，収納に便利である（平面「から」折れるではないことに注意）．テーブルを設計するにあたり，我々はまず実際の紙で試作を行い，画像認識を行った．その後，以下の制約で実現される意匠性と可折性の要求を達成した：

1. *OnPlane Constraint*: テーブルトップの頂点を与えられた水平面に固定する
2. *OnPlane Constraint*: 下部の頂点を接地する
3. *FlatFoldable Constraint*: 形状を平坦可折にする（紙面の都合上，平坦可折条件は解説しない）
4. *MirrorSymmetry Constraint*: 鏡映対称にする

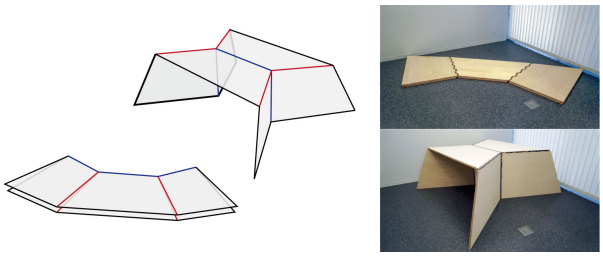

図 9. 平坦可折なテーブル.

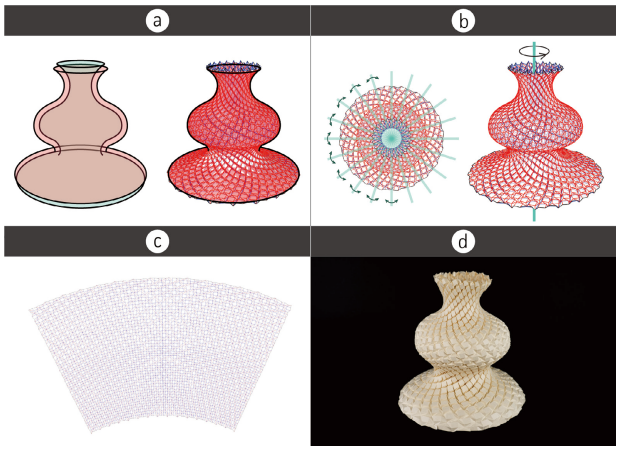

図 10. ランプシェード：(a)2 つのオフセット曲面に充填した waterbomb pattern，(b)*Rotational Symmetry Constraint* による回転対称なモデル，(c) 折紙の展開図，(d) 実物の写真.

ヒンジにはフィンガージョイントを選択し，木材は後述の EMARF [35] により切削し，それぞれのパネルを固定するためにピアノ丁番を使用した.

7.2 ランプシェード

図 10 に Crane で作製したランプシェードを示す. 入力として，2 つのオフセット曲面の間に waterbomb pattern を充填した [39]（第 4.2 節参照）. 制約として，まずランプシェードを 1 枚の紙から折るために *Developable Constraint* をかけた. さらに，形状を回転対称にするために *RotationalSymmetry Constraint* を課した（紙面の都合で説明は省く）. その後，Grasshopper 上で曲面の UV 座標を捻ることによって最終的な形状を得た. 製造の際には，まずカッティングプロッタで折り目の癖を付けた後，約 10 時間手折りすることで実物を得た.

7.3 Mask

図 11 にマスクを示す. 入力として，まず我々は顔の 3D メッシュを用意し，次にその顔を覆うメッシュを作成した. そのメッシュを入力として Generalized Ron Resch パターン（第 4.3 節参照）を生成し，意匠に利用した. また，ランプシェードと同様に *Developable Constraint* を利用して 1 枚の紙から作製できるようにして，カッティングプロッタおよび手折りで作製した.

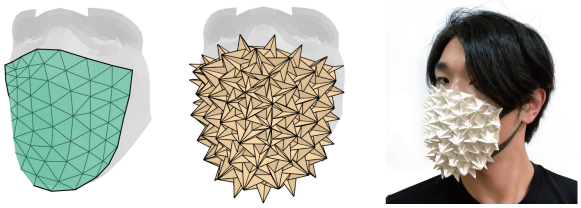

図 11. マスク：(左) 入力したメッシュ，(中) 自動生成されたパターン，(右) 実物の写真.

8 他のプラグインとの接続：EMARF

Crane は Grasshopper 上に実装されているため，プロダクトデザインで有用なプラグインを利用できる. ここでは紙面の都合により 1 つだけ紹介する.

EMARF [35] は Grasshopper で入力したデータに基づいて木材パネルを切削するオンライン製造サービスである. EMARF は Crane から入力した 3D モデルに対して CNC 加工機のツールパスを計算し，そのままデータ送信を行う. なお，本稿で示された椅子とテーブルは実際に EMARF を接続して発注・製造したものである.

9 限界：非剛体折りパターン

本研究で利用したシミュレーションは，折りのプロセスにおいて辺や面が変形しない剛体折りメカニズムに基づく. そのため，例えば折り鶴のような変形の際に歪みが生じるパターンは扱えない. 剛体折りできないパターンをシミュレーションしたい場合は，Origami Simulator [1, 12] など別のシステムを利用したほうが良い. なお，非剛体折りで形状探索ができるツールは現在のところ存在しない.

10 結論

本稿では，折紙を利用したプロダクトの設計プラットフォームである Crane を提案した. このシステムは入力・シミュレーション・形状探索・厚み付けに対応しており，ユーザの要求する意匠性・可折性・可製造性を制約ベースのシステムでインタラクティブに実現する. Grasshopper 上に実装されているため，必要に応じて他のプラグインを利用できる.

最後に，本システムは GitHub と Food4Rhino で公開されている [28, 27]. 我々は，HCI・ファブリケーション・建築・デザイン・ユビキタスコンピューティングなどに関わる様々な分野の人々が，Crane を通じて新たな驚くべき方法で折紙を活用することを期待している.

謝辞

本研究は情報処理推進機構 (IPA) の未踏ソフトウェア創造事業および JST ACT-I（JP18070086）の支援を受けて実施された．また，椅子の設計例は下田悠太氏によるものである．

参考文献

[1] A. Ghassaei et al., Origami Simulator, 2017, https://origamisimulator.org/.

[2] B. An, Y. Tao, J. Gu, T. Cheng, X. A. Chen, X. Zhang, W. Zhao, Y. Do, S. Takahashi, H.-Y. Wu, T. Zhang, and L. Yao. Thermorph: Democratizing 4D Printing of Self-Folding Materials and Interfaces. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI '18, p. 1–12, New York, NY, USA, 2018. Association for Computing Machinery.

[3] ARUP, Al Bahr Towers, 2012, https://www.arup.com/projects/al-bahr-towers.

[4] Z. Chang, T. D. Ta, K. Narumi, H. Kim, F. Okuya, D. Li, K. Kato, J. Qi, Y. Miyamoto, K. Saito, and Y. Kawahara. Kirigami Haptic Swatches: Design Methods for Cut-and-Fold Haptic Feedback Mechanisms. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, CHI '20, p. 1–12, New York, NY, USA, 2020. Association for Computing Machinery.

[5] K. C. Cheung, T. Tachi, S. Calisch, and K. Miura. Origami interleaved tube cellular materials. Smart Materials and Structures, 23(9):094012, aug 2014.

[6] E. Demaine, FOLD file format, 2017, https://github.com/edemaine/fold.

[7] L. H. Dudte, E. Vouga, T. Tachi, and L. Mahadevan. Programming curvature using origami tessellations. Nature materials, 15(5):583–588, 2016.

[8] S. Felton, M. Tolley, E. Demaine, D. Rus, and R. Wood. A method for building self-folding machines. Science, 345(6197):644–646, 2014.

[9] E. T. Filipov, G. H. Paulino, and T. Tachi. Origami tubes with reconfigurable polygonal cross-sections. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 472(2185):20150607, 2016.

[10] M. Gardiner, R. Aigner, H. Ogawa, and R. Hanlon. Fold mapping: parametric design of origami surfaces with periodic tessellations. Origami, 7:105–118, 2018.

[11] M. Gardiner, R. Aigner, H. Ogawa, and R. Hanlon. Fold Mapping: Parametric Design of Origami Surfaces with Periodic Tessellations. 09 2018.

[12] A. Ghassaei, E. D. Demaine, and N. Gershenfeld. Fast, interactive origami simulation using GPU computation. Origami, 7:1151–1166, 2018.

[13] ITOKI, Flip Flap Chair, 2017, https://www.itoki-global.com/products/flip_flap.html.

[14] J. S. Ku. Folding thick materials using axially varying volume trimming. In ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2017.

[15] J. S. Ku and E. D. Demaine. Folding flat crease patterns with thick materials. Journal of Mechanisms and Robotics, 8(3), 2016.

[16] K. Kuribayashi, K. Tsuchiya, Z. You, D. Tomus, M. Umemoto, T. Ito, and M. Sasaki. Self-deployable origami stent grafts as a biomedical application of Ni-rich TiNi shape memory alloy foil. Materials Science and Engineering: A, 419(1):131–137, 2006.

[17] B. Liu, J. L. Silverberg, A. A. Evans, C. D. Santangelo, R. J. Lang, T. C. Hull, and I. Cohen. Topological kinematics of origami metamaterials. Nature Physics, 14(8):811–815, 2018.

[18] J. Mitani. A Design Method for 3D Origami Based on Rotational Sweep. Computer-Aided Design and Applications, 6(1):69–79, 2009.

[19] K. Miura. Proposition of pseudo-cylindrical concave polyhedral shells. In IASS Symposium on Folded Plates and Prismatic Structures, 1970.

[20] K. Miura and R. J. Lang. The science of Miura-ori: A review. Origami, 4:87–99, 2009.

[21] S. Mueller, B. Kruck, and P. Baudisch. LaserOrigami: Laser-Cutting 3D Objects. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13, p. 2585–2592, New York, NY, USA, 2013. Association for Computing Machinery.

[22] Y. Noma, K. Narumi, F. Okuya, and Y. Kawahara. Pop-up Print: Rapidly 3D Printing Mechanically Reversible Objects in the Folded State. UIST '20, p. 58–70, New York, NY, USA, 2020. Association for Computing Machinery.

[23] S. Olberding, S. Soto Ortega, K. Hildebrandt, and J. Steimle. Foldio: Digital Fabrication of Interactive and Shape-Changing Objects With Foldable Printed Electronics. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, UIST '15, p. 223–232, New York, NY, USA, 2015. Association for Computing Machinery.

[24] J. Ou, M. Skouras, N. Vlavianos, F. Heibeck, C.-Y. Cheng, J. Peters, and H. Ishii. AeroMorph - Heat-Sealing Inflatable Shape-Change Materials for Interaction Design. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology, UIST '16, p. 121–132,

New York, NY, USA, 2016. Association for Computing Machinery.

[25] J. Qi and L. Buechley. Animating Paper Using Shape Memory Alloys. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12, p. 749–752, New York, NY, USA, 2012. Association for Computing Machinery.

[26] G. Saul, C. Xu, and M. D. Gross. Interactive Paper Devices: End-User Design & Fabrication. In Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction, TEI '10, p. 205–212, New York, NY, USA, 2010. Association for Computing Machinery.

[27] K. Suto, CRANE, 2019, https://www.food4rhino.com/en/app/crane.

[28] K. Suto, KaiSut0/Crane, 2021, https://github.com/KaiSut0/Crane.

[29] K. Suto, Y. Noma, K. Tanimichi, K. Narumi, and T. Tachi. Crane: An Integrated Computation Design Platform for Functional, Foldable, and Fabricable Origami Products. ACM Transactions on Computer-Human Interaction (TOCHI), to appear in 2022.

[30] T. Tachi. Freeform variations of origami. J. Geom. Graph, 14(2):203–215, 2010.

[31] T. Tachi. Origamizing Polyhedral Surfaces. IEEE Transactions on Visualization and Computer Graphics, 16(2):298–311, 2010.

[32] T. Tachi. One-DOF rigid foldable structures from space curves. In Proceedings of the IABSE-IASS Symposium, 2011.

[33] T. Tachi. Rigid-foldable thick origami. pp. 253–264, 2011.

[34] T. Tachi. Designing Freeform Origami Tessellations by Generalizing Resch's Patterns. Journal of Mechanical Design, 135(11), 10 2013. 111006.

[35] VUILD, EMARF, 2019, https://www.food4rhino.com/app/emarf.

[36] WOW, ISSEY MIYAKE "Steam Stretch" Concept Movie, 2014, https://vimeo.com/108092253.

[37] J. Yamaoka, M. D. Dogan, K. Bulovic, K. Saito, Y. Kawahara, Y. Kakehi, and S. Mueller. FoldTronics: Creating 3D Objects with Integrated Electronics Using Foldable Honeycomb Structures. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI '19, p. 1–14, New York, NY, USA, 2019. Association for Computing Machinery.

[38] Y. Yoshimura. On the mechanism of buckling of a circular cylindrical shell under axial compression. 1955.

[39] Y. Zhao, Y. Endo, Y. Kanamori, and J. Mitani. Approximating 3D surfaces using generalized waterbomb tessellations. Journal of Computational Design and Engineering, 5(4):442–448, 01 2018.

[40] 鳴海紘也, 中原健一, 佐藤宏樹, 新山龍馬, 川原圭博. Liquid Pouch Motors: 紙のインタフェースのための薄く軽く柔軟なアクチュエータ. 第 26 回インタラクティブシステムとソフトウェアに関するワークショップ (WISS 2018), No. 83, p. 1–6, 2018.

[41] 野間裕太, 鳴海紘也, 奥谷文徳, 川原圭博. Pop-up Print: 双安定な折りたたみ 3D プリント. 第 28 回インタラクティブシステムとソフトウェアに関するワークショップ (WISS 2020), No. 89, p. 43–48, 2020.

未来ビジョン

我々はこれまでの WISS で，折紙のような薄い物体を変形させることのできるアクチュエータ Liquid Pouch Motors [40] や，3D 構造物をあらかじめ畳んだ状態で印刷し，印刷後に展開する手法 Pop-up Print [41] など，アクチュエーション・ファブリケーションに関する技術を紹介してきた．

本研究は折紙の設計に立ち返り，これまで専門家にしか扱えなかった計算折紙の理論を他分野のユーザにも利用できることを目指したものである．また，理想的な折紙をシミュレーション上で再現するにとどまらず，実際にプロダクトとして作ることのできる環境を整えた．元々記載したかった内容が長大であったため，様々な機能や評価をカットしなるべくシステム全体の理解が可能となるように努めた．発表では，原稿で触れられなかった部分も紹介したい．

また，我々は計算折紙の理論で作成された複雑なパターンをより簡単に製造する技術の開発も進めている．我々の最終目標は，折紙の分野に見られる機能的で美しい形状を，計算設計と計算製造の知見を活用することで実世界で簡単に再現することである．本研究はそのための重要な第一歩であり，本研究の成果を利用して様々な応用が発生することも期待している．