

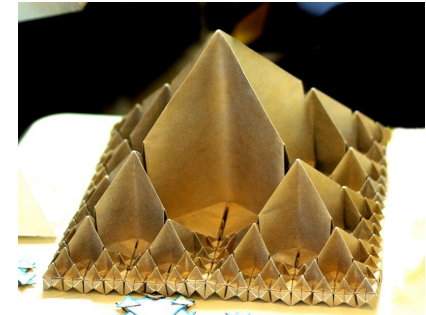
3D折り紙アニメーションのための モデリングソフトウェアの開発

古田 陽介* 三谷 純 福井 幸男

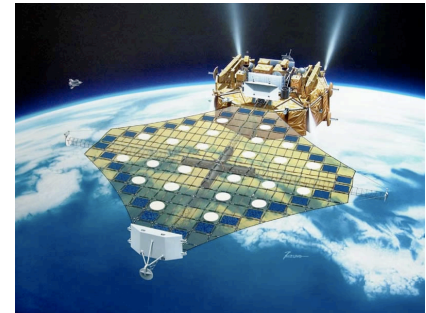
筑波大学 大学院 システム情報工学研究科
コンピュータサイエンス専攻 二年 (200620987)

背景

- 幾何学の世界を中心に研究されている
- 工学的な「折り畳む」用途への応用
宇宙開発、建築、医療...
- 教育に用いる
幾何学の基本的性質の理解
指先を使うことは脳を活性化させる
日本文化の理解



フラクタルピラミッド (池上 2006)



ミウラ-オリ太陽パドル試験衛星(想像図)
©水野哲也イラストレーション事務所



ステント

(Z. You and K. Kuribayashi 2006)

背景

- 幾何学の世界を中心に研究されている
- 工学的な「折り畳む」用途への応用
宇宙開発、建築、医療...

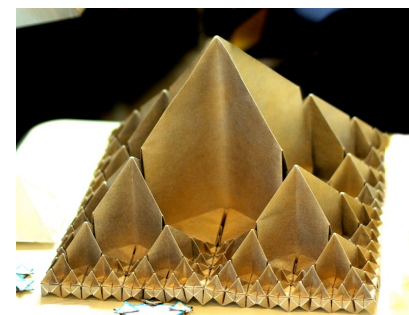
- 教育に用いる

幾何学の基本的性質の理解

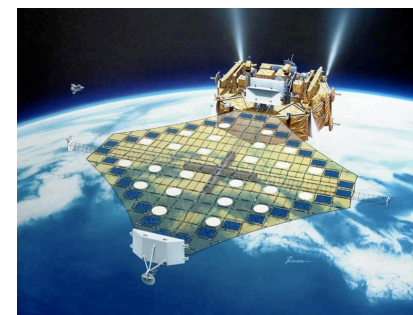
指先を使うことは脳を活性化させる

日本文化の理解

分かりやすく折り手順を
伝える方法が必要



フラクタルピラミッド (池上 2006)



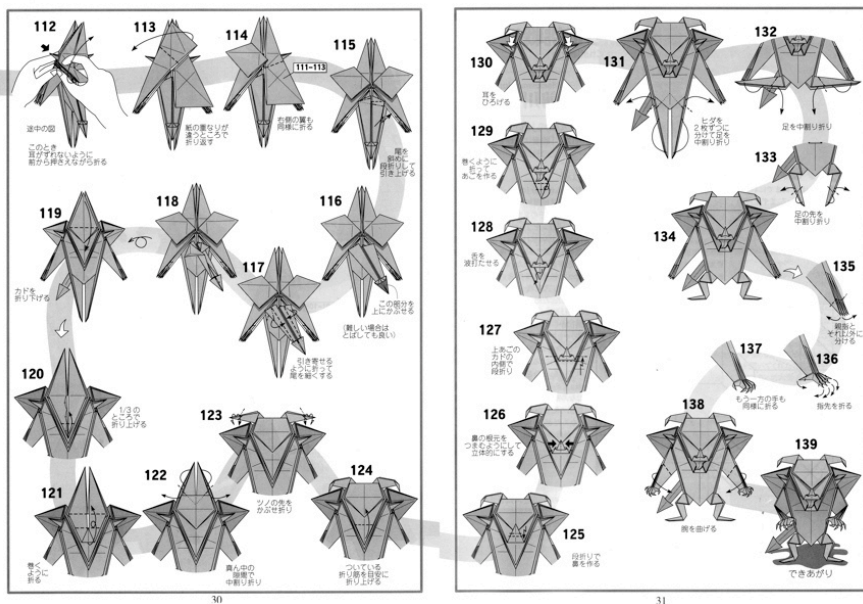
ミウラ-オリ太陽パドル試験衛星(想像図)
©水野哲也イラストレーション事務所



ステント

(Z. You and K. Kuribayashi 2006)

折り図との比較



折り図

日本折紙学会「折紙探偵団 56号」 1999

× 時間が離散的 ×

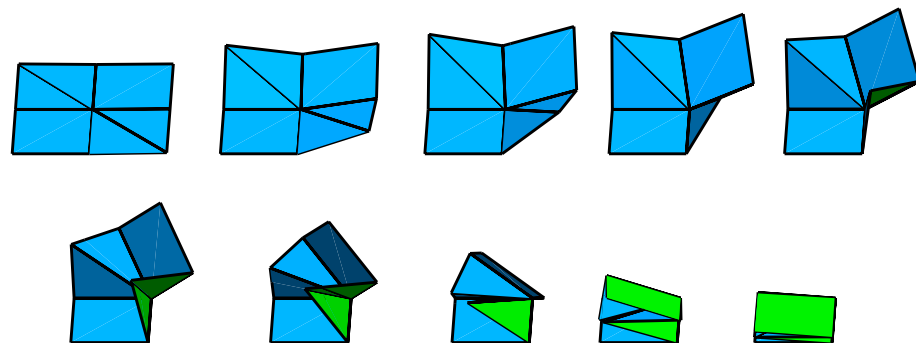
× 二次元 ×

◎三次元◎

◎視点が動かせる◎

◎時間が連続的◎

× 制作に手間 ×



3DCG アニメーション

Devin Balkcom "Robotic Origami Folding"

目的と概要

計算機上の三次元仮想空間内で紙の形状操作を対話的に行い、それによる形状変化を3DCGアニメーションとして保存

目的と概要

計算機上の三次元仮想空間内で紙の形状操作を対話的に行い、それによる形状変化を3DCGアニメーションとして保存

紙の三次元的な形状変化をリアルタイムに計算するための計算モデル

目的と概要

計算機上の三次元仮想空間内で紙の形状操作を対話的に行い、それによる形状変化を3DCGアニメーションとして保存

紙の三次元的な形状変化をリアルタイムに計算するための計算モデル

マウス、キーボードで対話的に紙を折るためのインタフェース

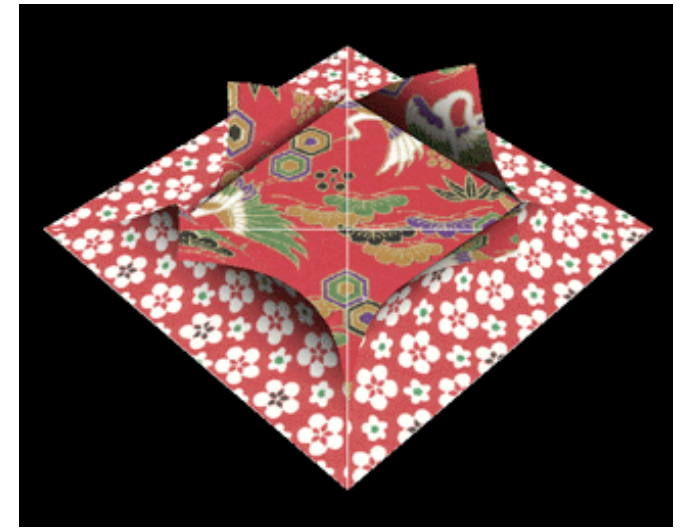
折り紙を表現するためのモデル

重なり順を持った面

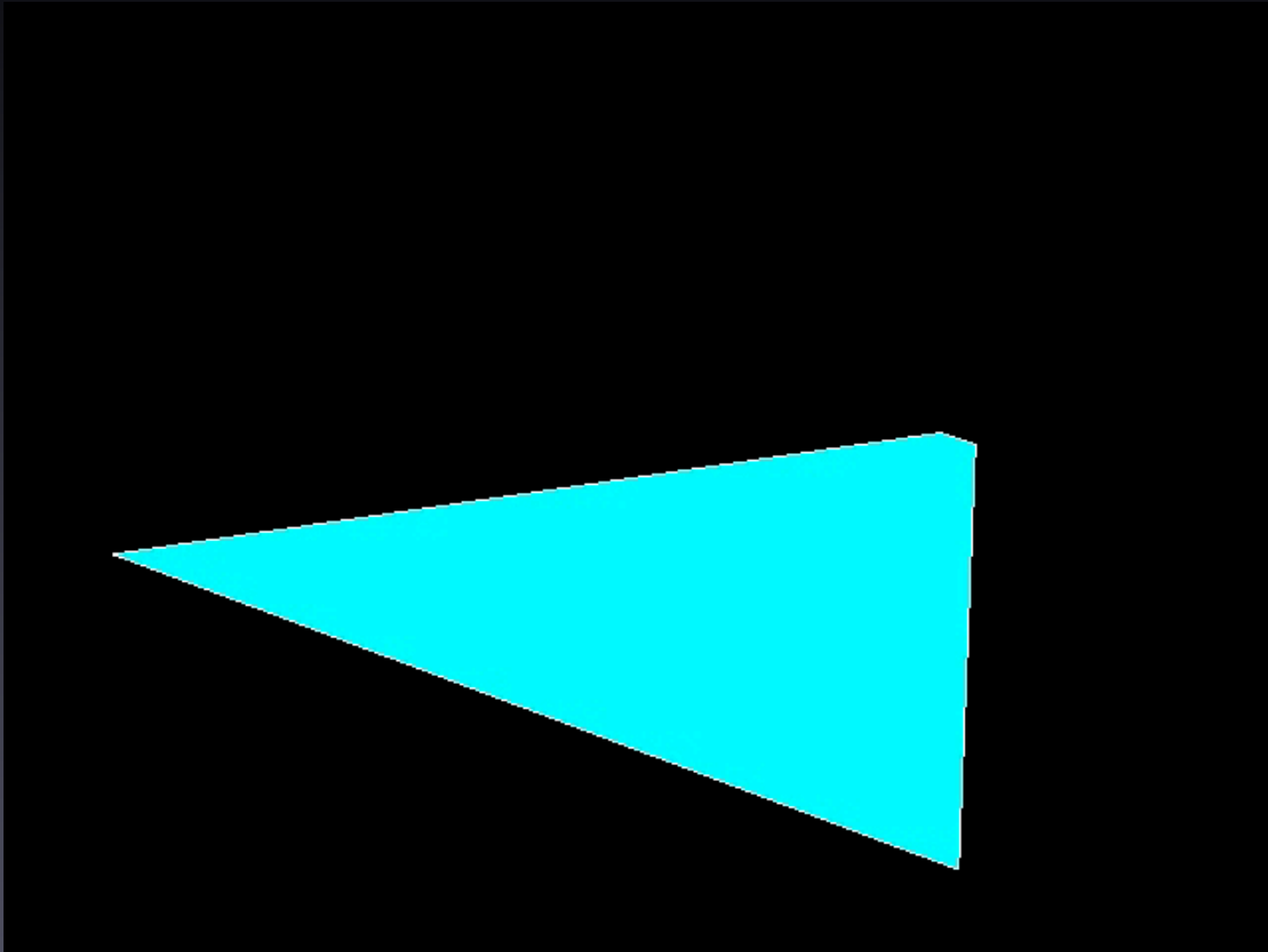
ヒンジによって拘束された剛体面

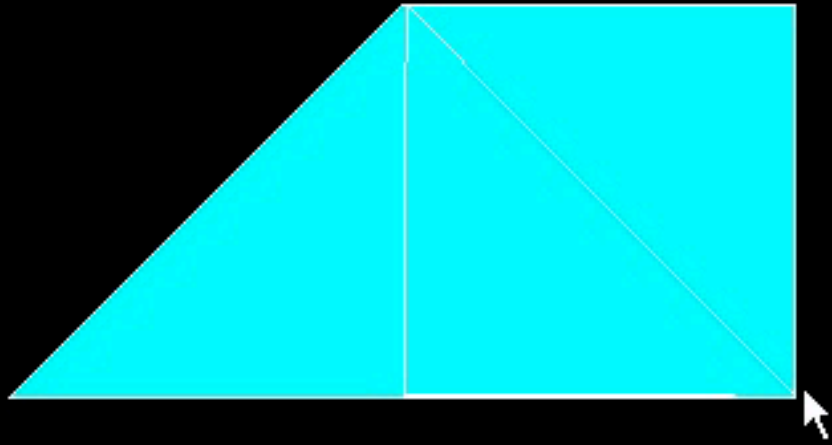
端を共有する複数のバネ

重なり順を持った面

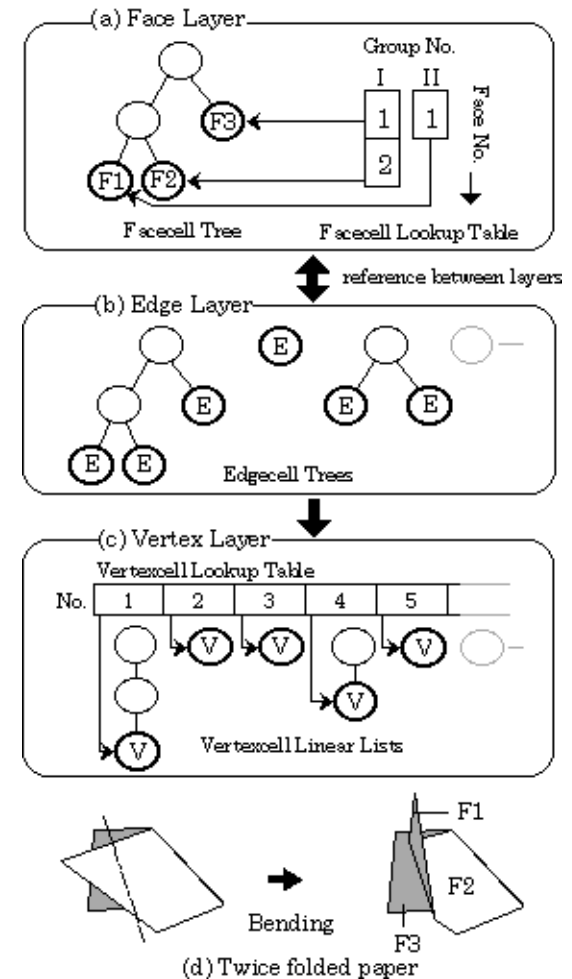
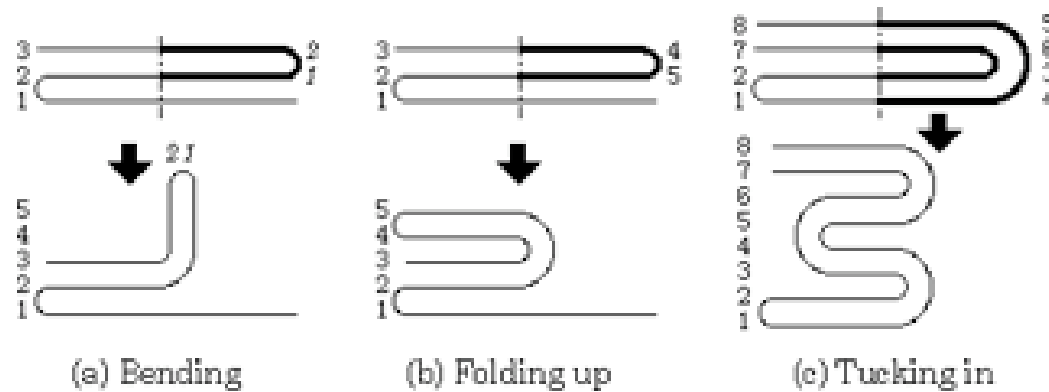


折り紙シミュレーション
宮崎 慎也 (中京大学) 1993.9



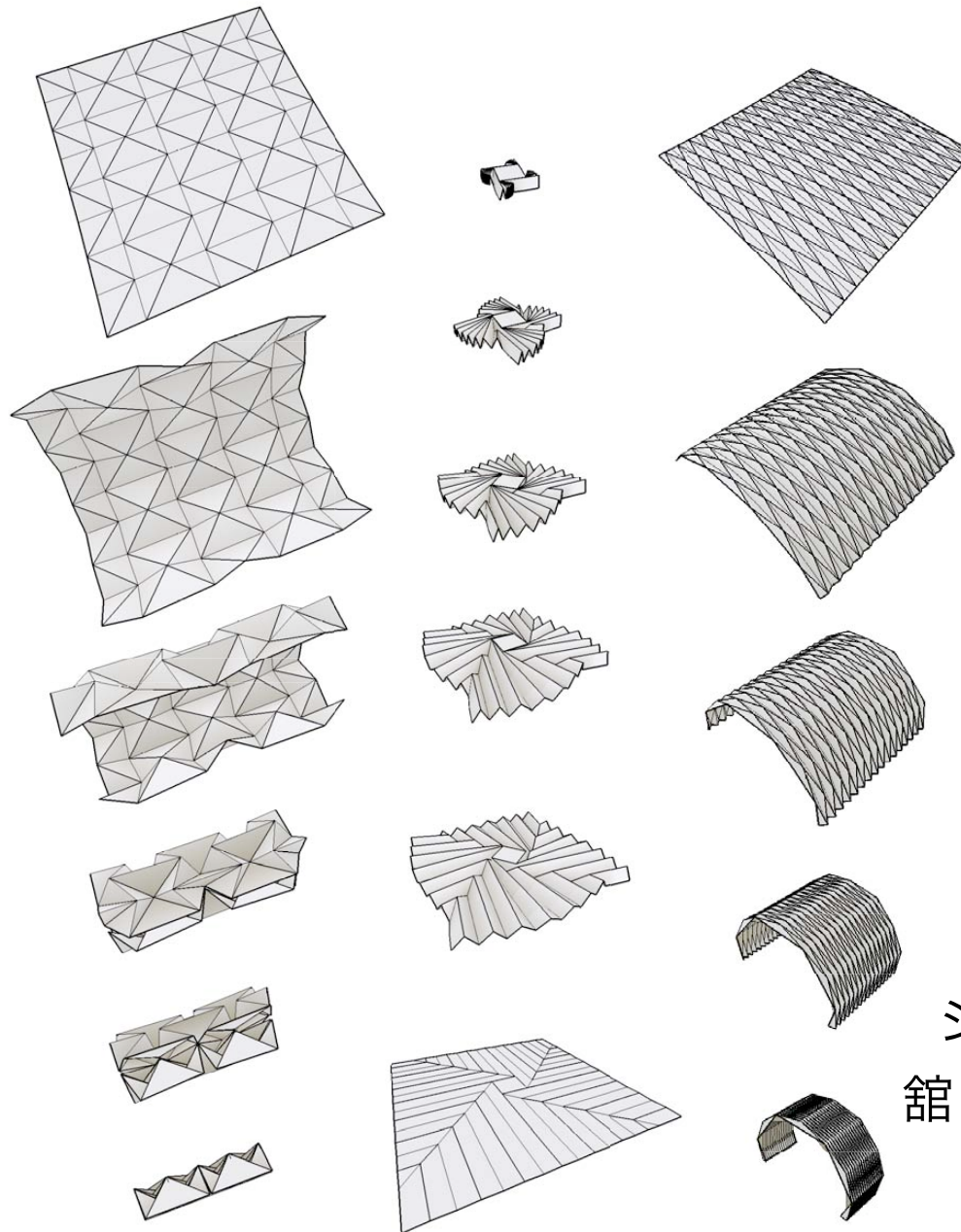


重なり順を持った面



- 面の重なり順を保ったまま面を動かすことができる
- ×複雑な三次元的形状をうまく扱えない

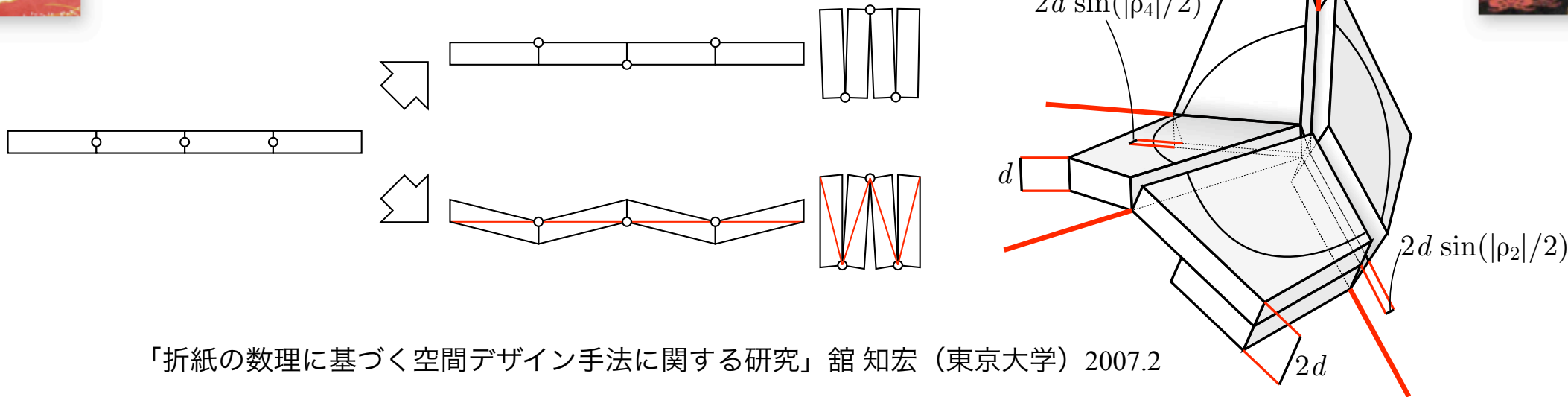
ヒンジによって拘束された剛体面



剛体折り紙
シミュレーション
舘 知宏 (東京大学)
2007.2



ヒンジによって拘束された剛体面

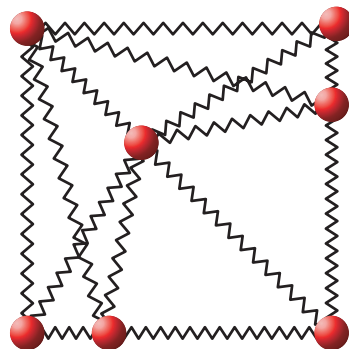
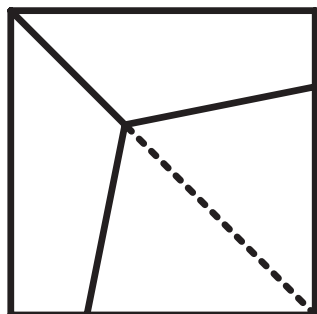
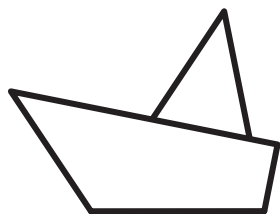


「折紙の数理に基づく空間デザイン手法に関する研究」 舘 知宏（東京大学）2007.2

折り紙をヒンジで拘束された剛体パネルと見なし、そのキネマティクスを求めることで折りの遷移状態を計算する

- 全ての折り線が連動して動く様子をシミュレートできる
- × 歪みが発生する折り方が出来ない
- × 対話的な操作が出来ない

端を共有する複数のバネ



$$F_i = \sum_j \left\{ k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} - D v_{ij} \right\} - mg$$

$$v_i(t + \Delta t) = v_i(t) + d \frac{F_i(t)}{m} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t$$

- 隣接する面は複数の頂点を共有する
- 各面の内側には網羅的にバネが張られている
 - 面を構成するすべての頂点間にバネを張る
 - 別の面にある頂点の間にはバネを張らない

個々のバネの弾性力の合計 F から微小時間経過後の頂点位置 r を求める計算を繰り返し行う

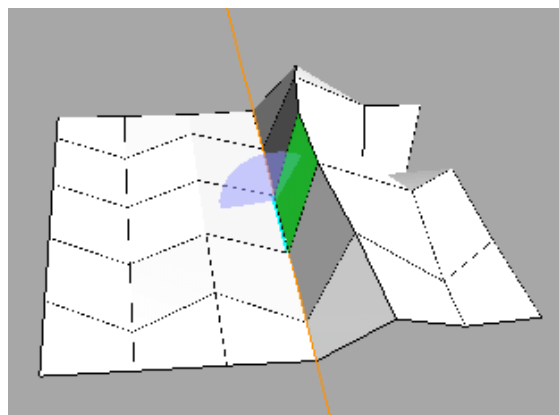
QuickTime ÇΔ

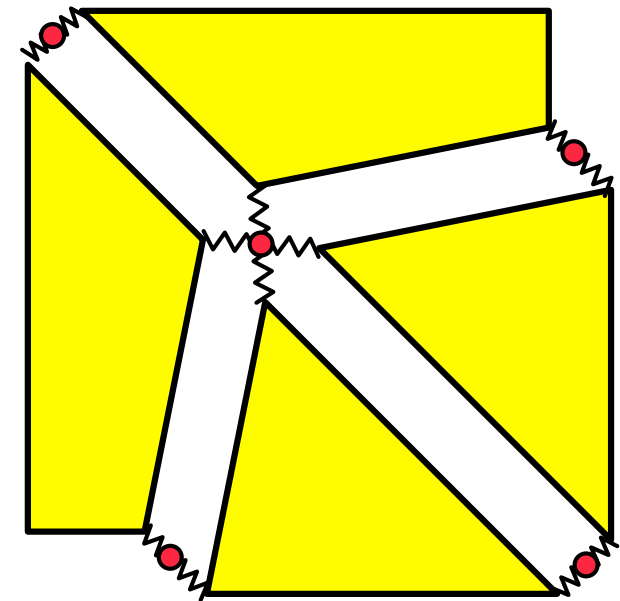
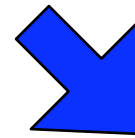
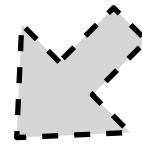
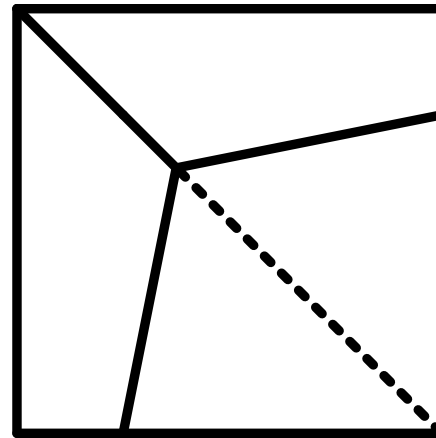
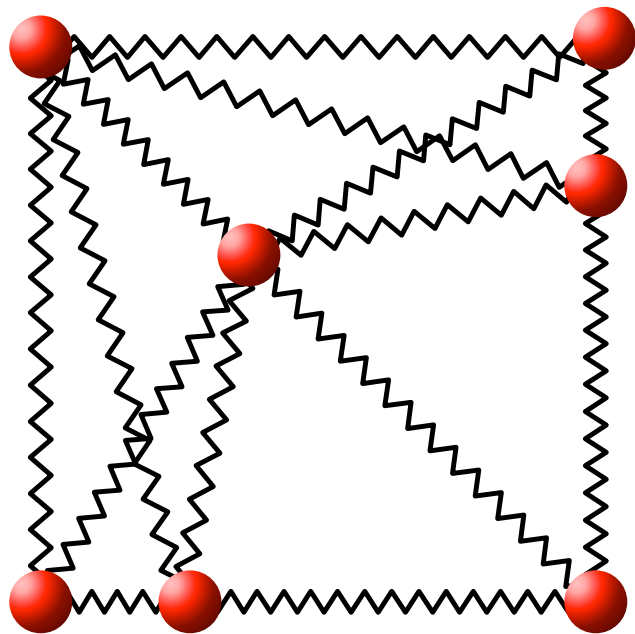
èLí£ÉvÉçÉÓÉâÉÄ

Ç™Ç±ÇÄÉsÉNÉ`ÉÉÇ•â©ÇÉÇΩÇπÇ...ÇÖïKónÇ≈ÇΣÅB

バネモデルの特徴

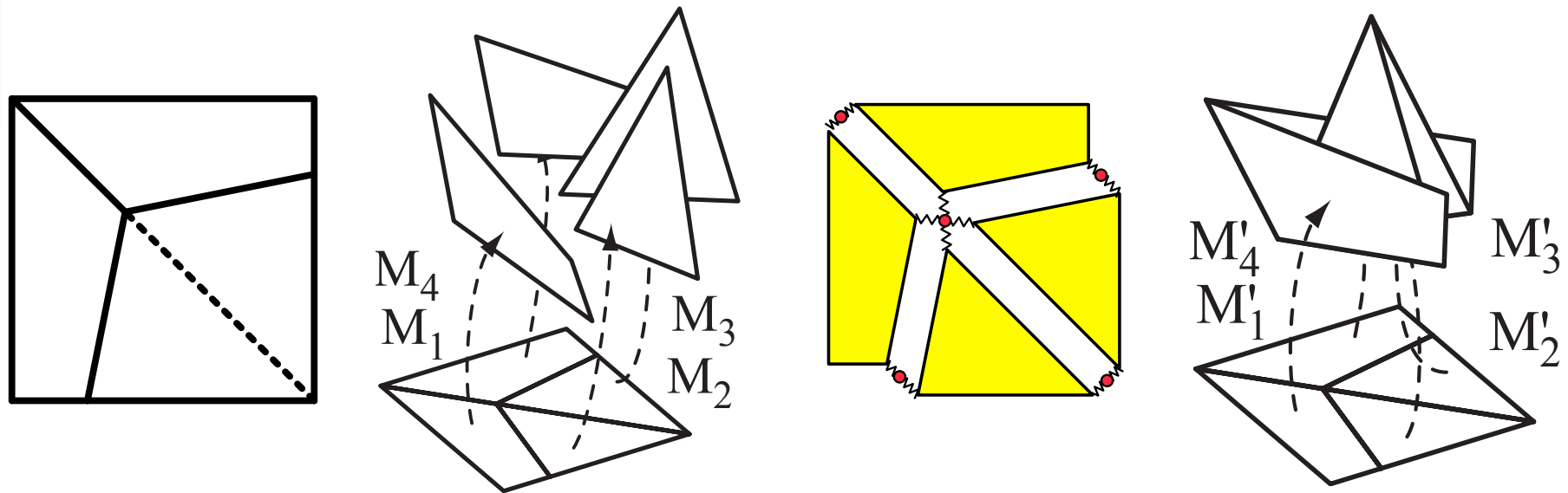
- 多少の歪みを許容できる
- リアルタイム性が高い
- 面の形状に依存しない
- × 多角形だと歪みの収束が遅い
- × 面が多数あると操作しにくい
- × 重なりを考慮することが難しい



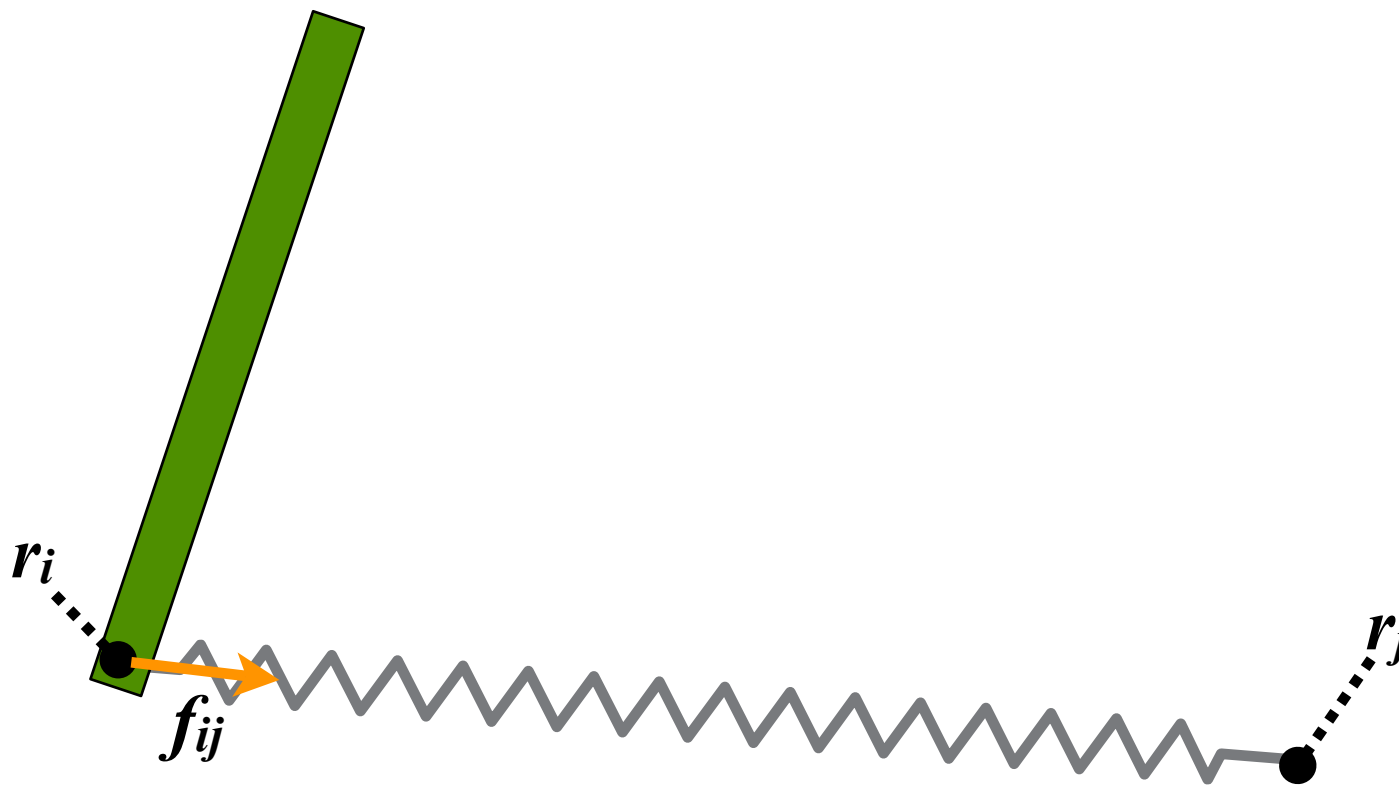


- × 平面の内側に網羅的にバネを張る ×
- 平面の外側の頂点間にバネを張る ○

提案手法



1. 二次元の展開図の各面にアフィン変換の行列を掛けて三次元化
2. 各面の共通する頂点の平均座標に向けてバネを張る
3. バネの力を基にして、離ればなれの面が接続するように行列の値を調整する

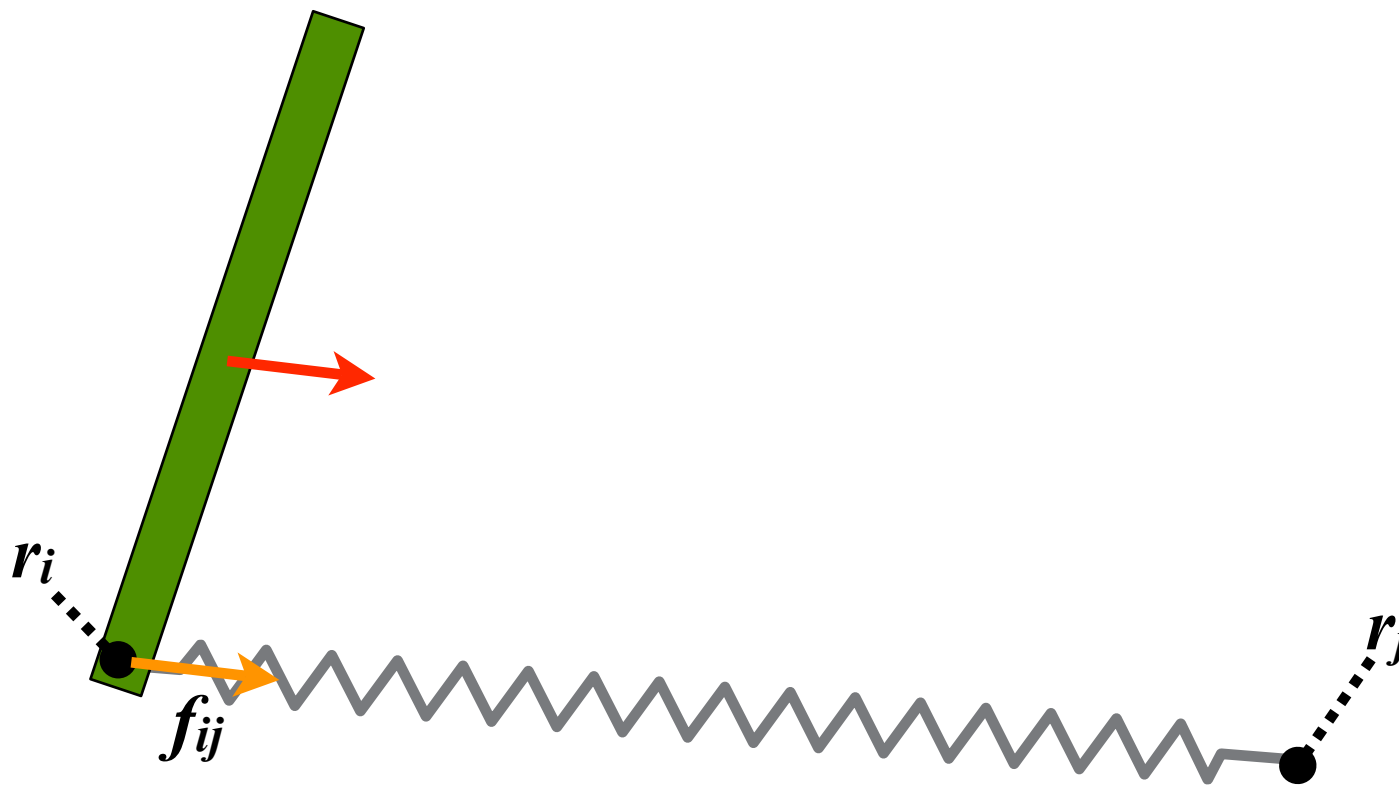


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

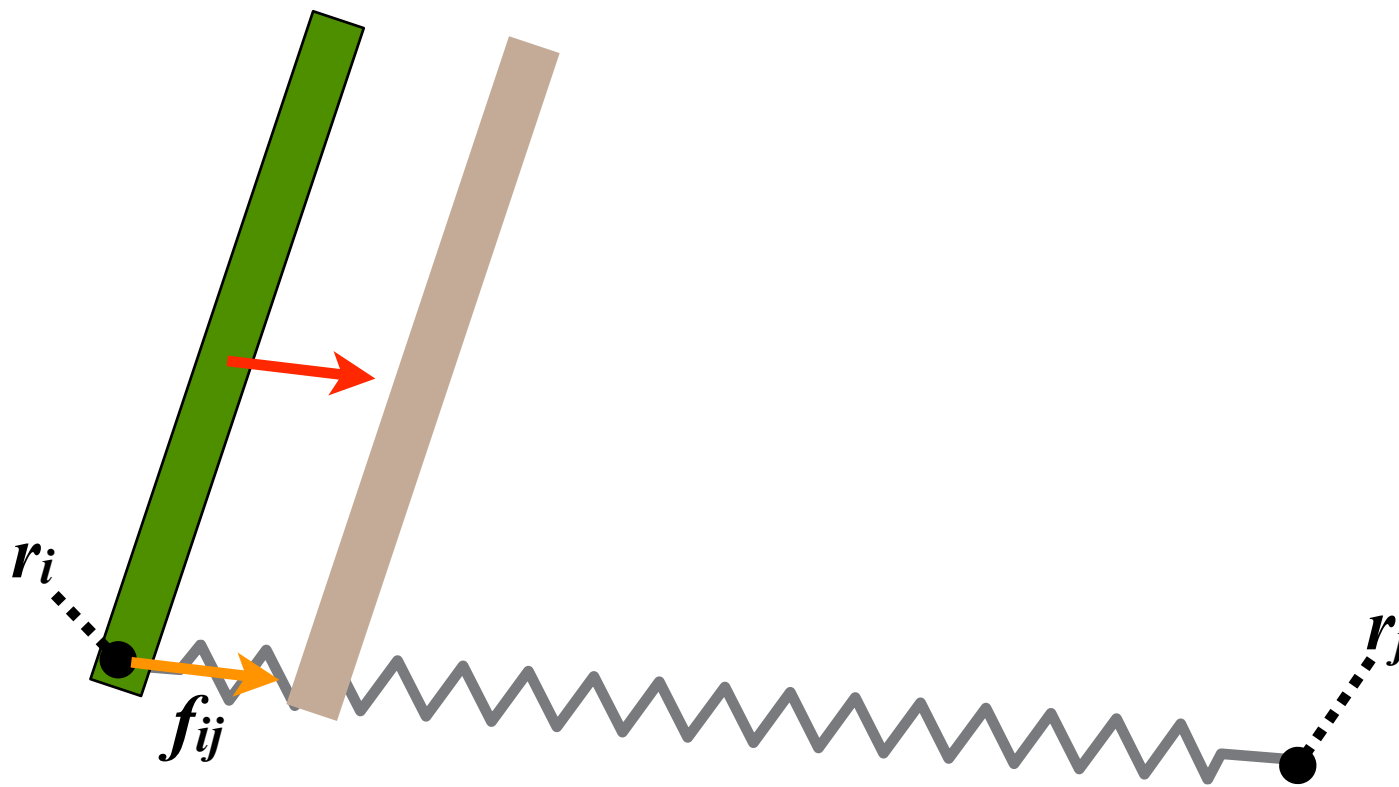


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

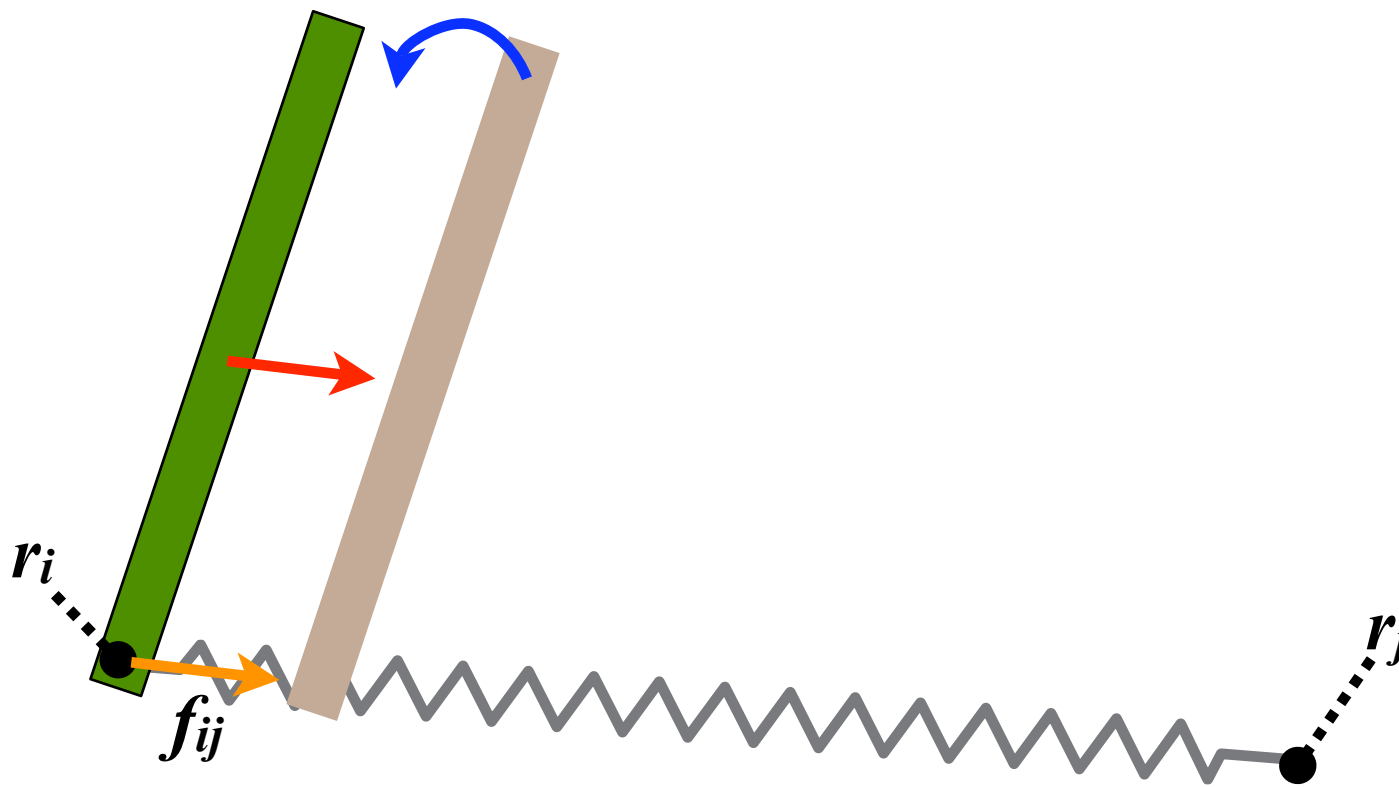


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

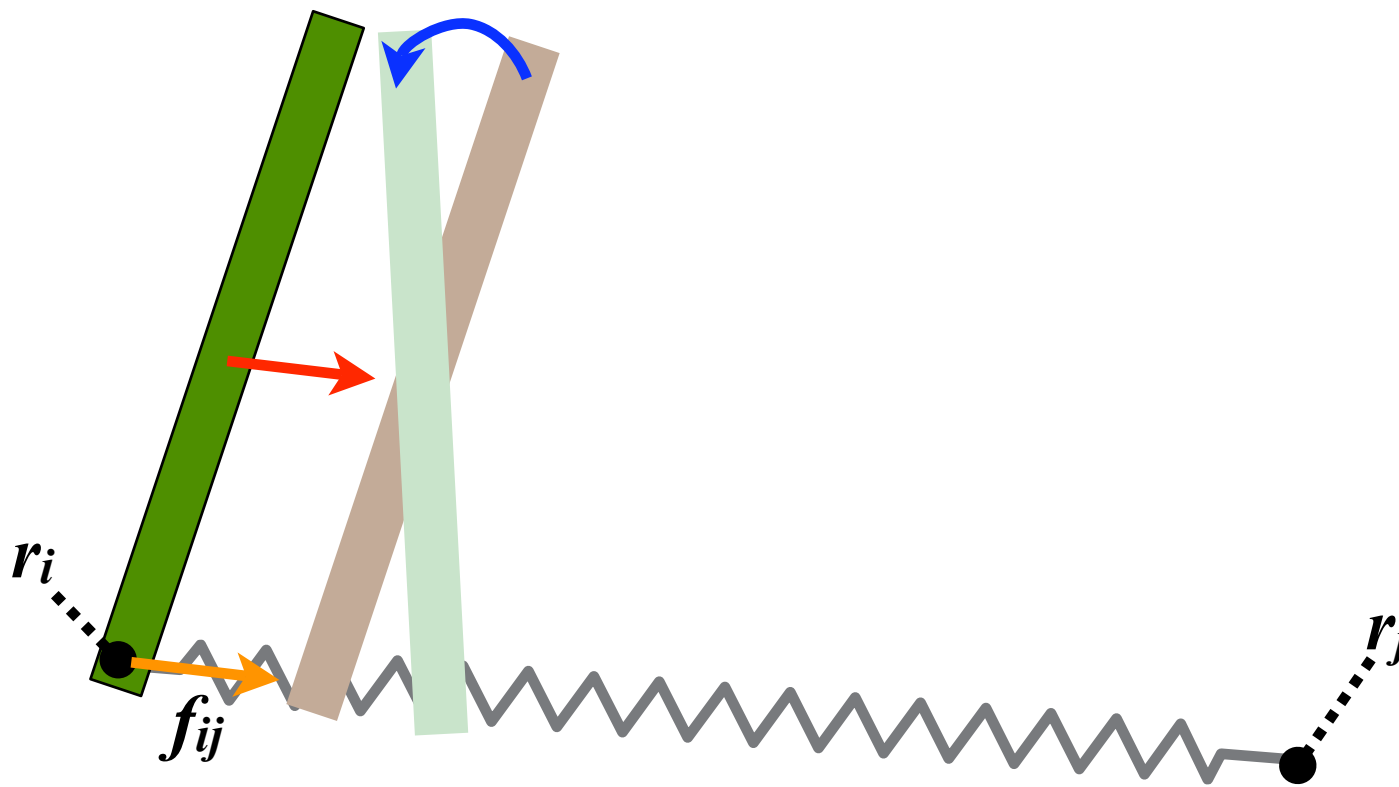


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

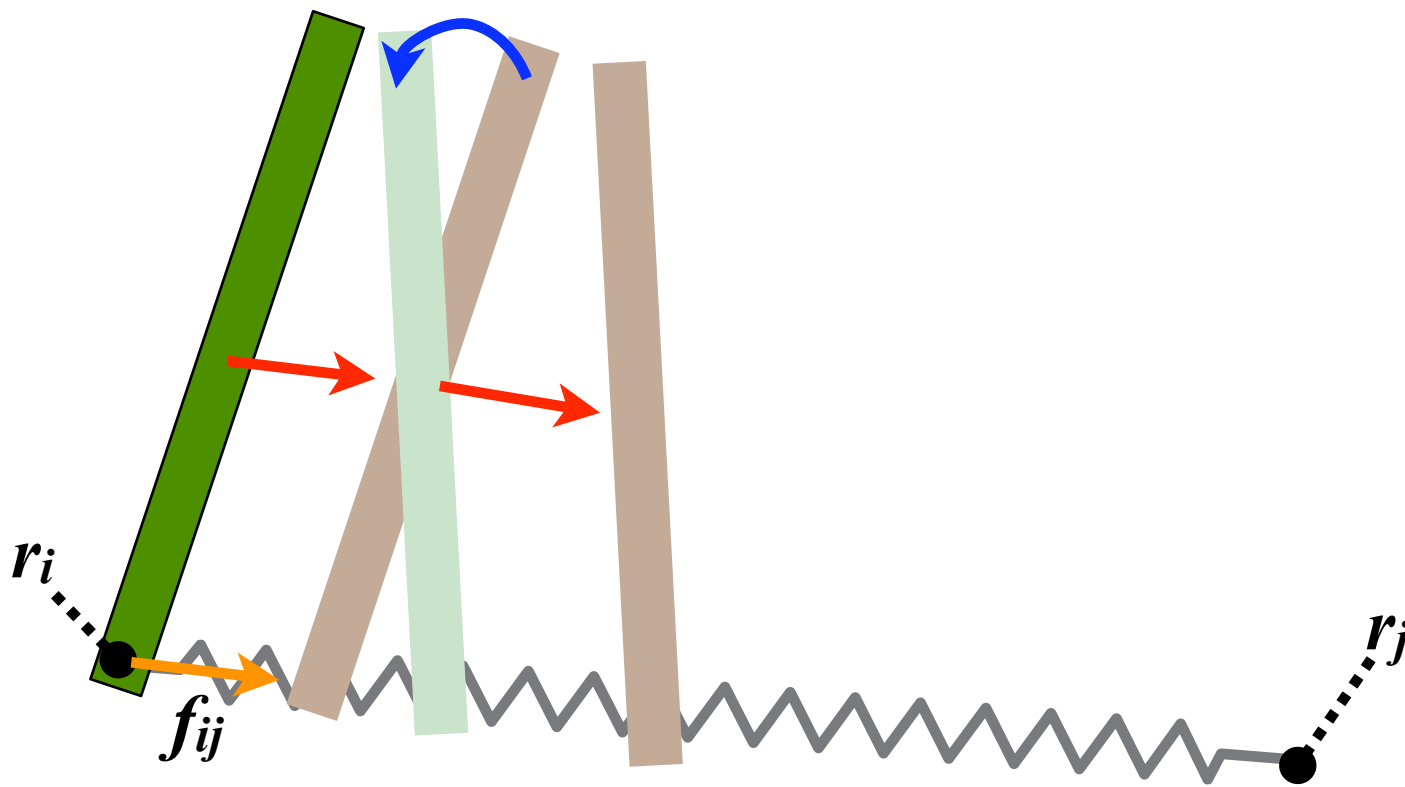


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

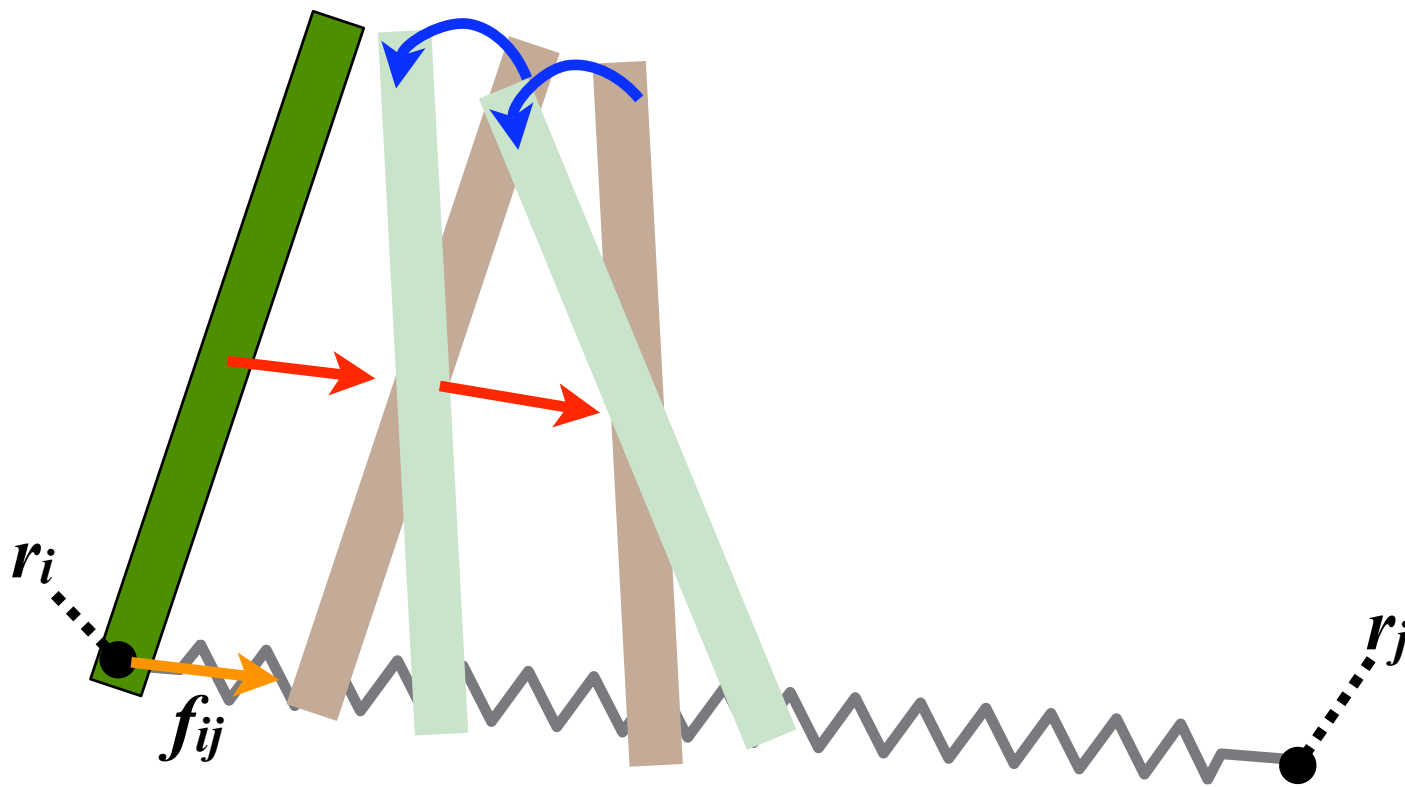


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

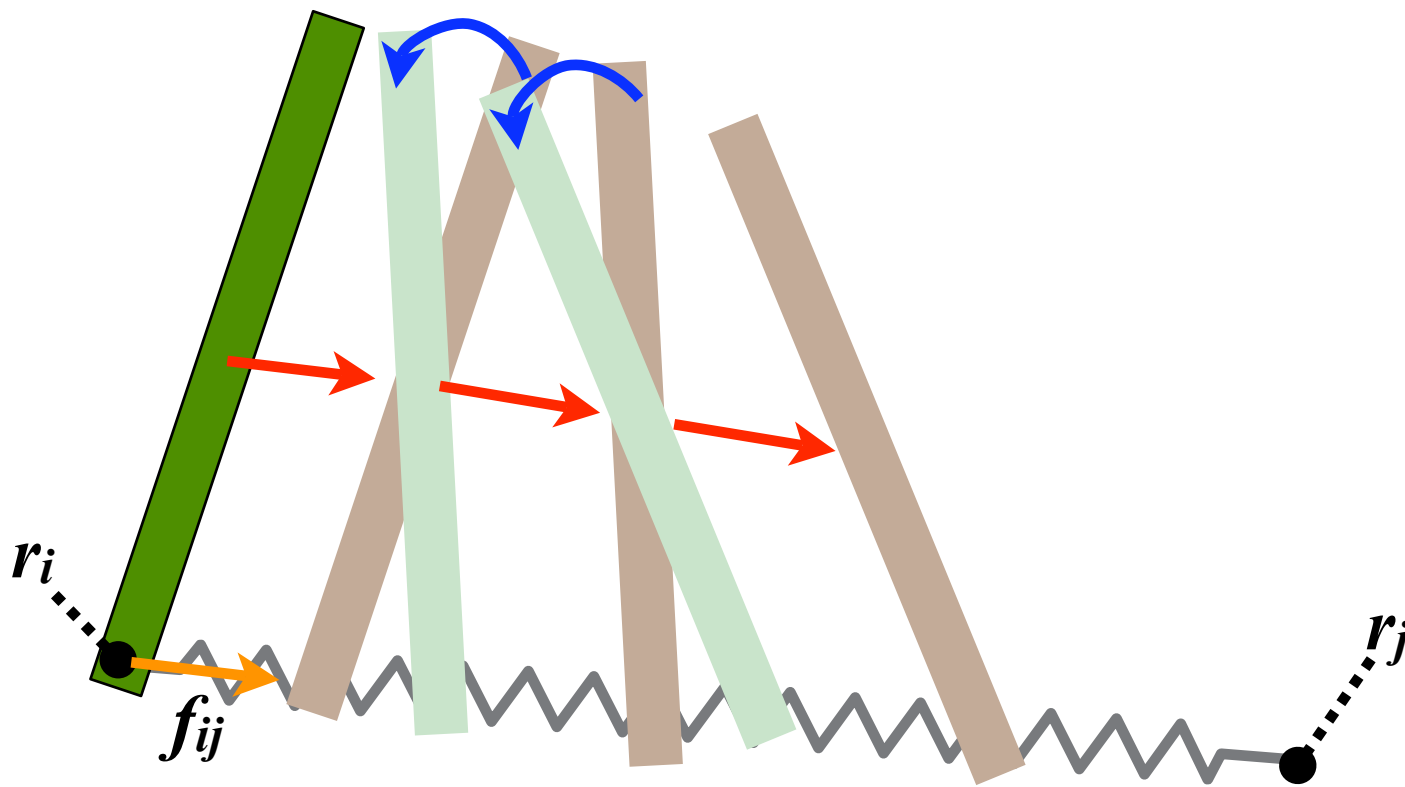


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

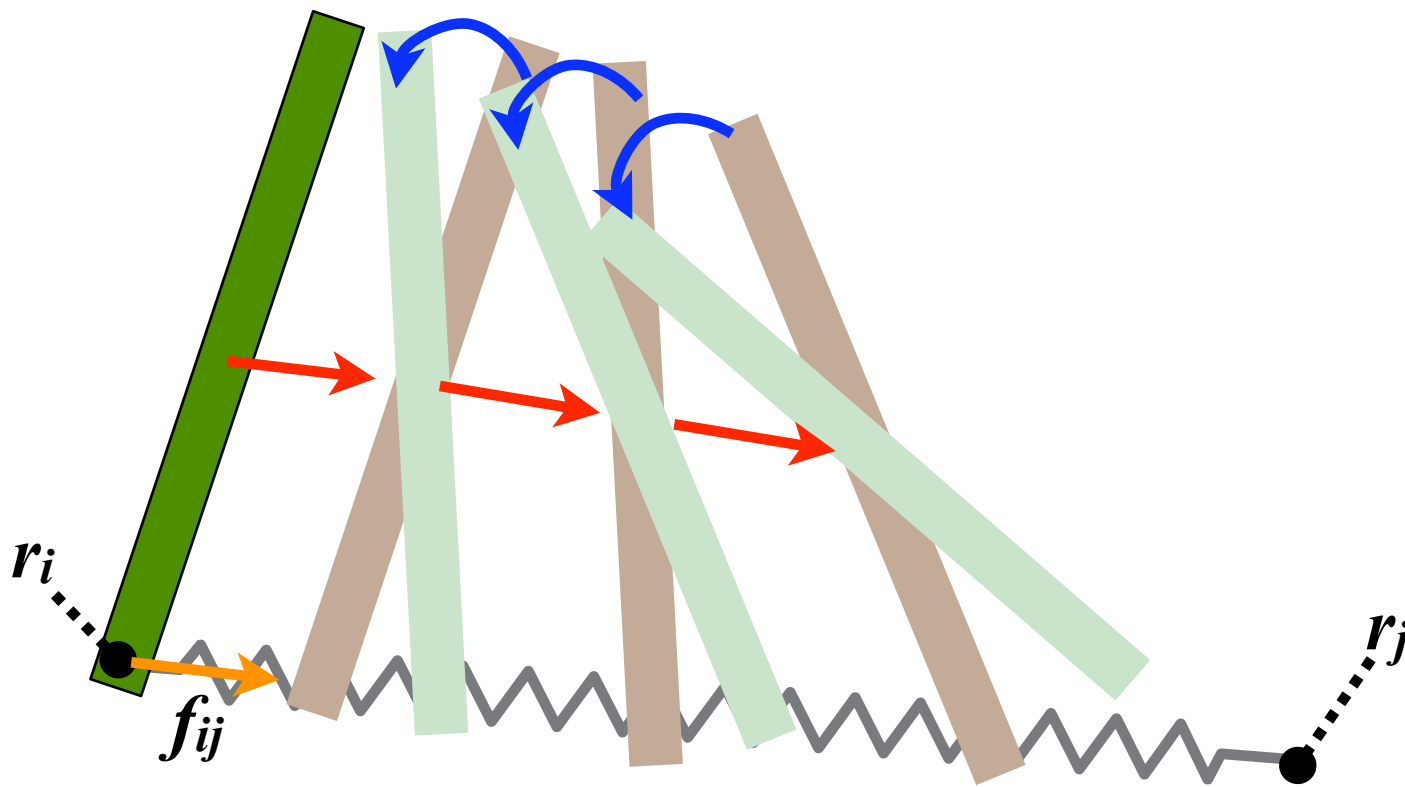


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

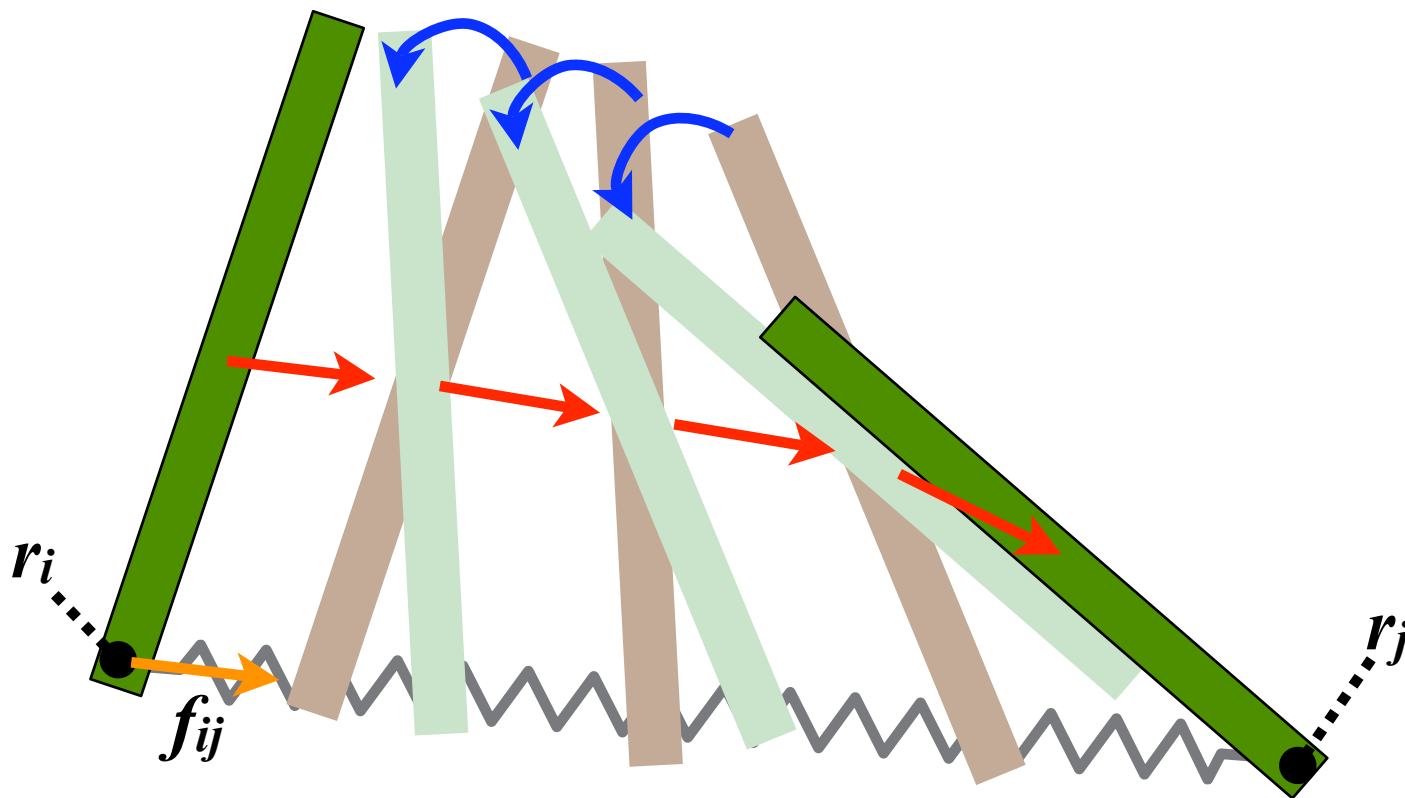


$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$



$$f_{ij} = k \left(1 - \frac{L_{ij}}{|r_{ij}|} \right) r_{ij} + Dv_{ij}$$

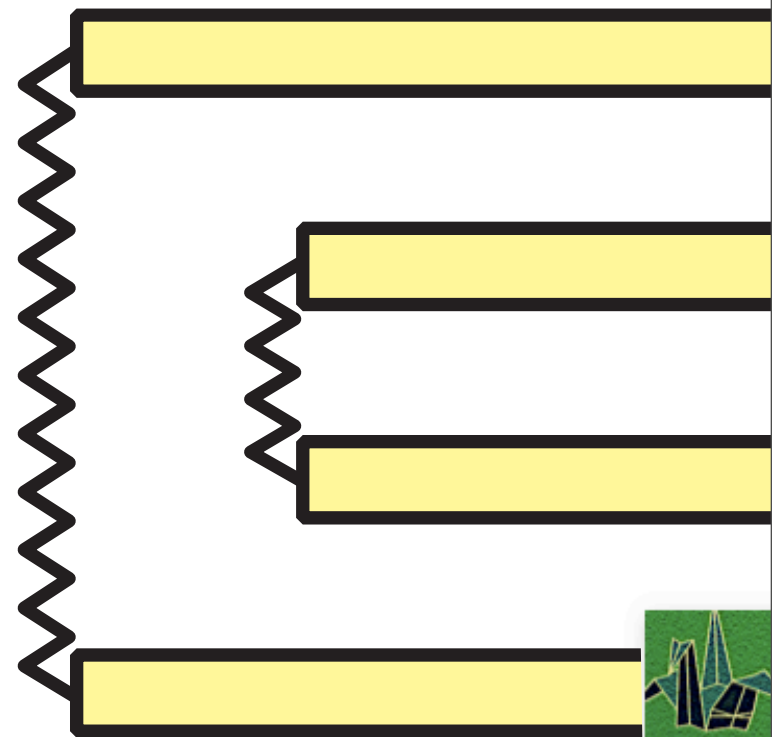
$$\star F_i = \sum f_{ij} - m_i g \quad \star \tau_i = \sum (r_{oi} \times f_{ij}) - r_{oi} \times m_i g$$

$$v_i(t + \Delta t) = v_i(t) + \frac{F_i(t)}{m_i} \Delta t \quad \omega_i(t + \Delta t) = \omega_i(t) + \frac{\tau_i(t)}{I_i} \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + v_i(t) \Delta t \quad \lambda_i(t + \Delta t) = \lambda_i(t) + \omega_i(t) \Delta t$$

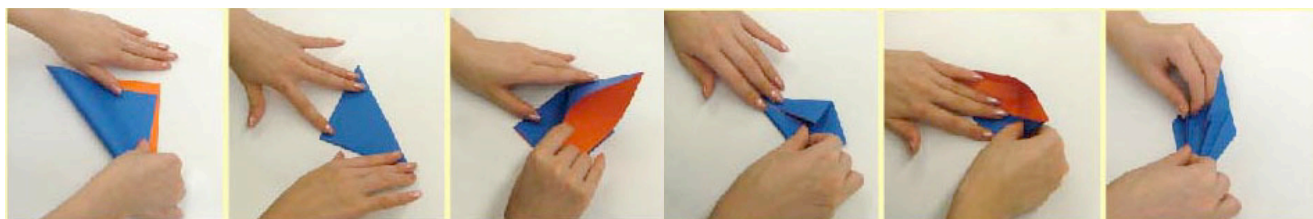
提案手法の特徴

- 多少の歪みを許容できる
- リアルタイム性が高い
- 面の形状に依存しない
- ◎面に歪みが発生しない
- ◎重なりを表現しやすい
- ◎厚みを表現しやすい



紙を折るインタフェース

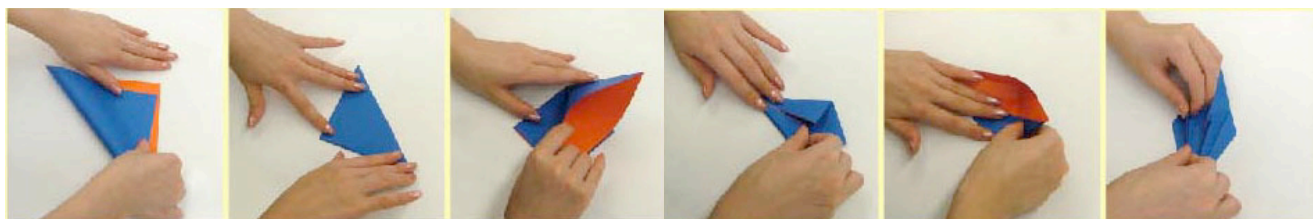
両手で行われる折り操作をマウスやキーボードをつかって再現しなければならない



株式会社スタジオスピーク「まねっこおりがみ おれたくんDVD」

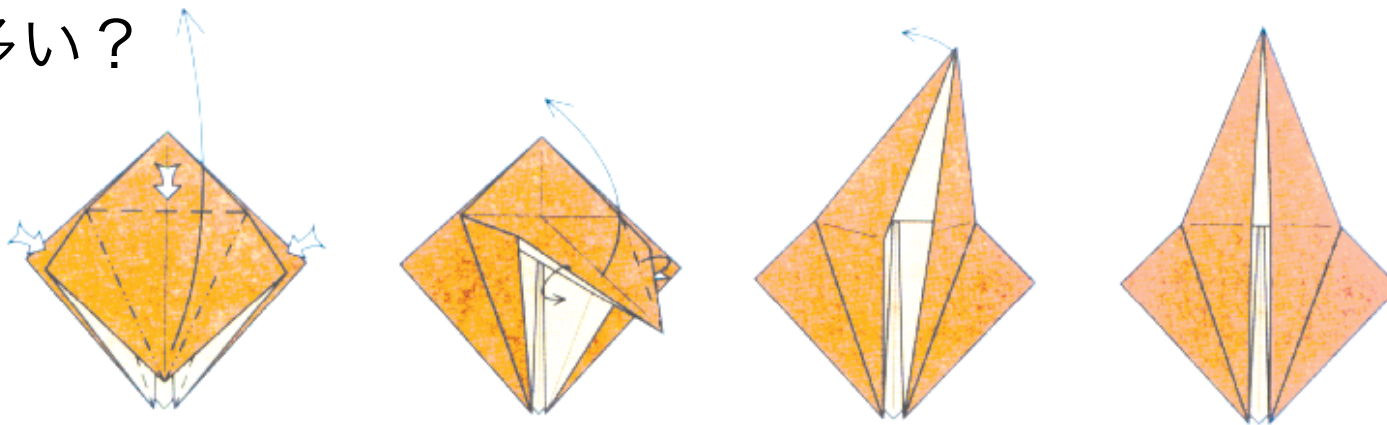
紙を折るインタフェース

両手で行われる折り操作をマウスやキーボードをつかって再現しなければならない



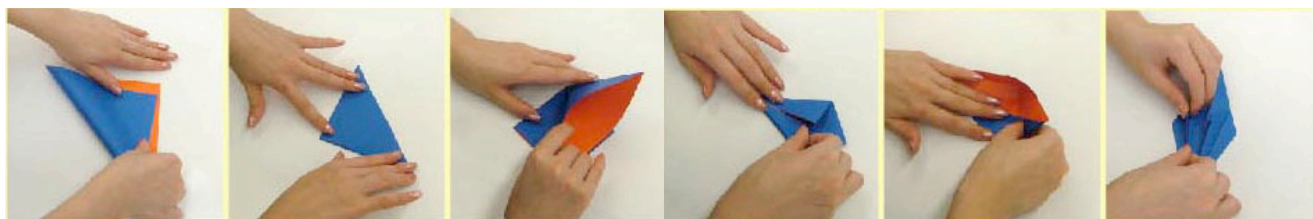
株式会社スタジオスピーク「まねっこおりがみ おれたくんDVD」

あらかじめ折り線をつけておけば、多くの折り操作は面の回転とそれに付随する隣接面の動きで表現できる場合が多い？



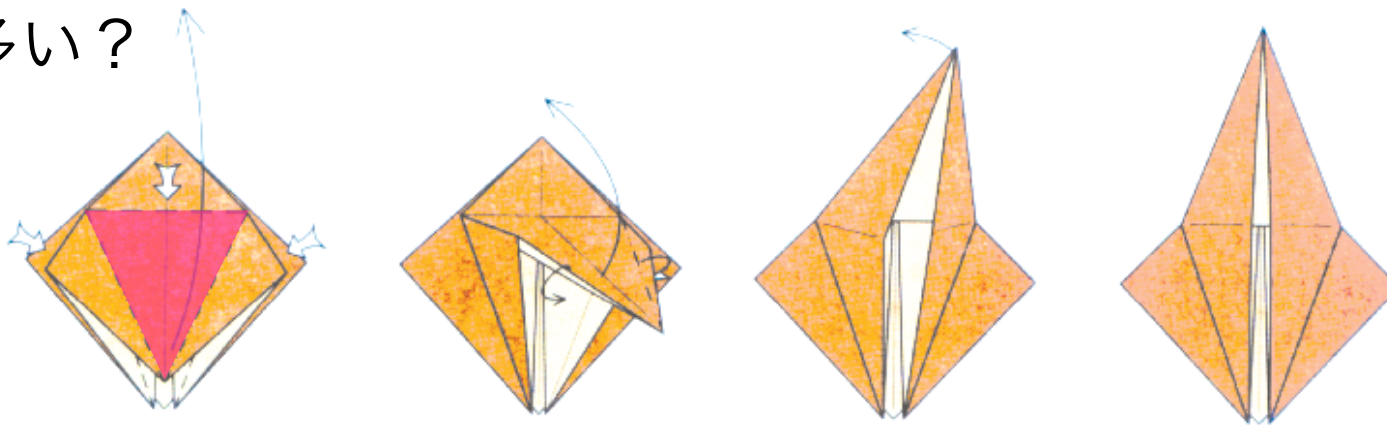
紙を折るインタフェース

両手で行われる折り操作をマウスやキーボードをつかって再現しなければならない



株式会社スタジオスピーク「まねっこおりがみ おれたくんDVD」

あらかじめ折り線をつけておけば、多くの折り操作は面の回転とそれに付随する隣接面の動きで表現できる場合が多い？



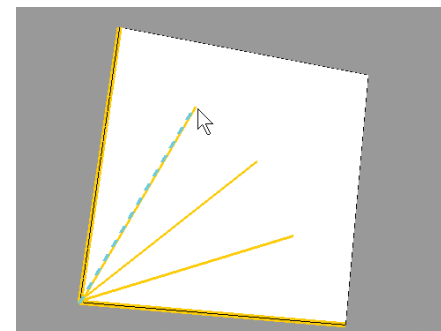
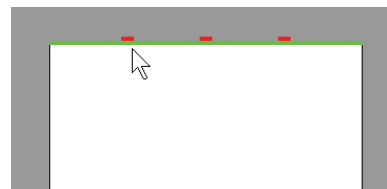
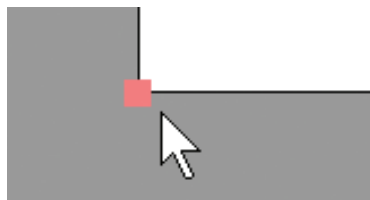
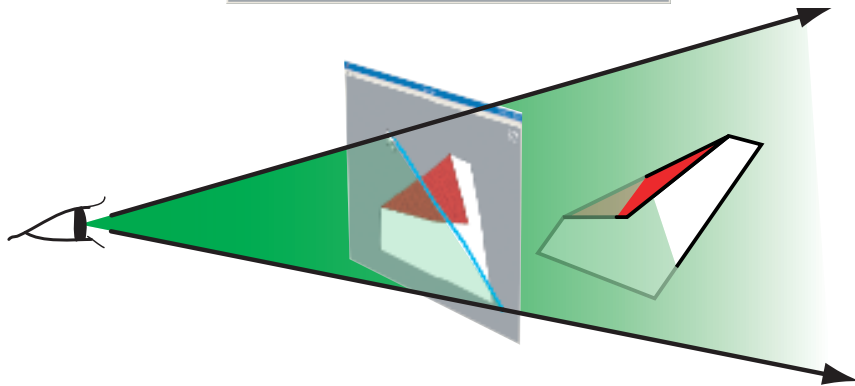
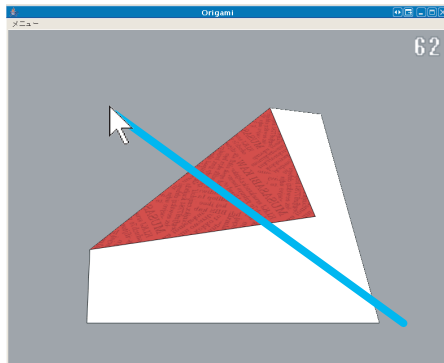
折れ線の入力

2Dドローツールを参考

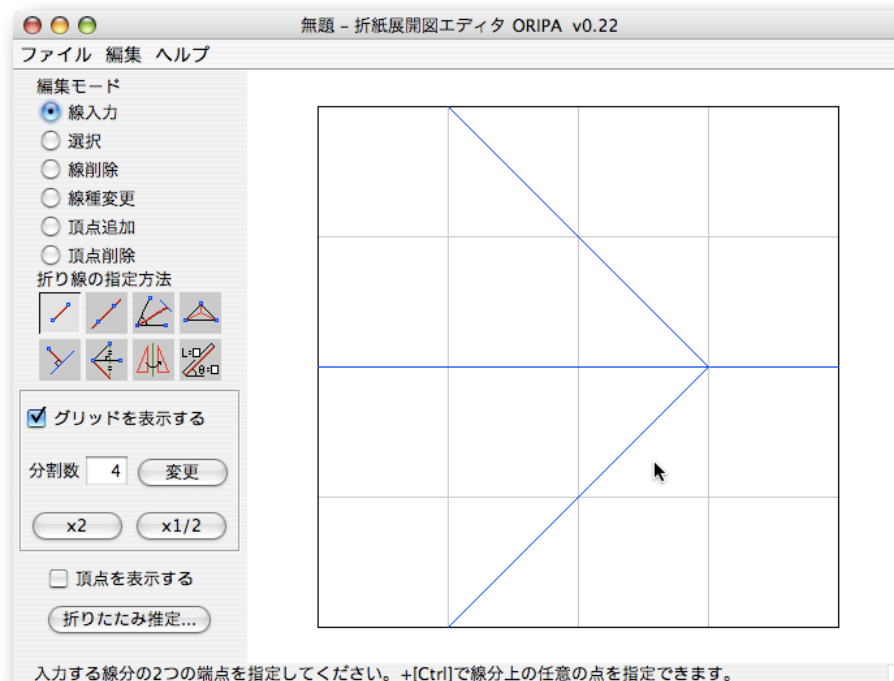
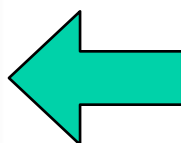
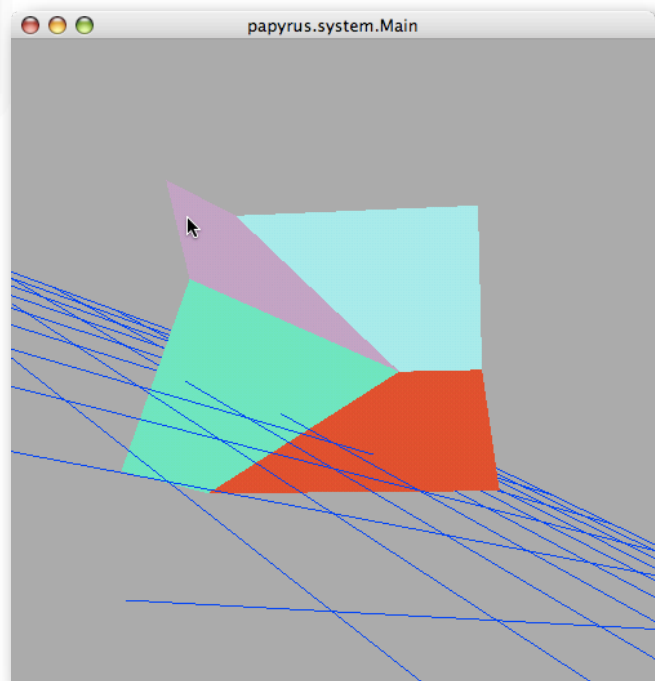
画面上の任意の場所をドラッグすることで自由に折れ線を「描く」

頂点付近や辺、角を等分する場所の付近ではマウス座標を自動的にスナップ

△ 重なっている下の面にも折れ線が引かれてしまう △



展開図エディタとの連携

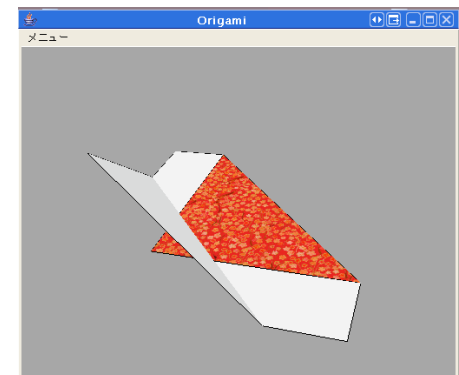
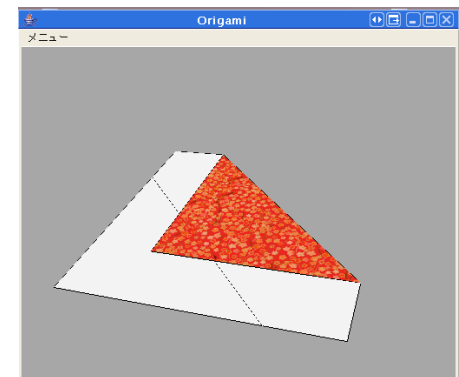
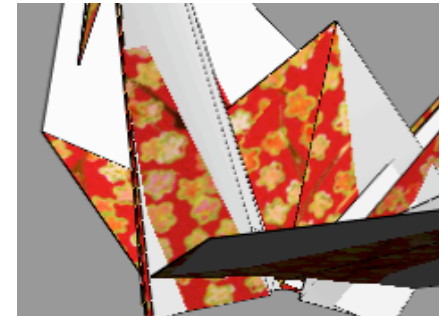


折紙展開図エディタ ORIPA

- 2Dエディタでの変更を監視し、3Dモデルに反映させる
- 変更前の面の三次元的な位置情報を変更後の面に継承
 - ▶ 二次元の交差判定で変更のあった面を判別

課題

- **面同士の衝突を考慮していない**
 - レンダリング結果が乱れる
 - 重なり合った面を同時に動かすことができない
 - ▶ 衝突判定
 - ▶ 角度に制約
- **3Dから2Dへの変更を反映できない**
- **面の接続が切れたように表示される**
 - ▶ レンダリングの際に隙間を埋める
- **他の手法との定量的な比較**



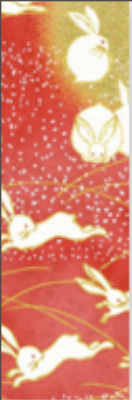
まとめ

依然のモデルより折り紙の形状を表現するのに適した新しい計算モデルを考案

従来の紙に折り目をつけるインタフェースを補うための、二次元CADツールと連携したインタフェースの提案

提案手法の実装

課題について考察



ご静聴ありがとうございました