

布モデルへの尾根と谷を指定することによるよれの付与手法

史 舜^{1,a)} 齋藤 豪^{1,b)}

概要: 布や服のモデルに現実感を与えるのはよれの表現である。しかし、知識や経験が無いユーザがモデリングツールなどを用いて自然なよれの形状を 3D モデルに付与することは困難である。本研究ではこの困難さを解消するため、ユーザの入力と物理シミュレーションを融合させることで、容易によれを付与することのできる手法を提案する。布はバネ質点モデルを用いてモデル化し、分割に適した形状である三角形を成すようにバネを接続する。提案手法ではユーザは尾根と谷を指定する入力を行い、それに応じてモデルに質点が追加される。その後その質点の周辺部分に平滑処理を施し、最後に物理シミュレーションを行うことでよれの形状を決定する。吊るされて重力が働くことによって生じるよれや、下地との摩擦によって生じるよれの本手法による再現例を示す。

キーワード: モデリング、シミュレーション、細分割

A method for adding wrinkles on cloth models by specifying ridge and valley lines

HATSU SHI^{1,a)} SUGURU SAITO^{1,b)}

1. はじめに

三次元コンピュータグラフィックスを用いた服のモデリングは、映像業界やファッション業界で多用されている。布や服のモデルに現実感を与えるのは、よれの表現である。よれは、折れ曲がる力に対し抵抗する力が強く働かないという、繊維の物理的な特性から生じる現象であり、その形状は布の素材や縫製、織り方などに影響される。そのため、知識や経験に乏しいユーザが既存の 3D モデリングツールなどを用いて、自然なよれをモデルに付与することは困難である。本研究ではこの困難さを解消するために、ユーザの入力と物理シミュレーションを融合させることで、容易によれを付与することのできる手法を提案する。

2. 関連研究

本章では 3D モデルに対しよれを付与する従来の研究を

紹介し、本研究の位置づけと新規性の確認を行う。

Wang ら [1] は 3D モデルに対しユーザに線を入力させ、その線に応じてポリゴンを変形させモデルに凹凸を与える手法を提案した。その際に与えるパラメータによって様々なよれを表現可能であるが、繊維や縫製に関する知識が無ければ自然なよれを付与することは難しい。Li ら [2] は、ユーザのスケッチから複雑な凹凸形状を持つ 3D モデルの生成を行うシステムを開発した。ユーザの入力は輪郭線の決定、また凹凸や曲率の指定に用いる。品質の高い 3D モデルを生成することが可能であるが、自然なよれの再現にはその物体のよれの形状に関する知識が必要となる。我々の手法では物理シミュレーションを行い形状を補正し、ユーザの入力した形状が歪なものであってもシミュレーションによって自然な形状を生成することを目指す。

Reis ら [3] は 3D モデルにシワを付与するために、あらかじめ法線マップ用のテクスチャに発生しうる全てのシワを用意しておき、その中のいくつかを 3D モデルの形状に合わせ描画する手法を提案した。この手法ではシワのテクスチャを用意しなければならないため、事前の準備が必要

¹ 東京工業大学
Tokyo Institute of Technology

a) hatsu@img.cs.titech.ac.jp

b) suguru@c.titech.ac.jp

Algorithm 1 よれの付与

1. ユーザによる線の入力
2. 入力によって選ばれた格子の法線方向に質点を追加し、格子を分割することで単純なよれを想起させる形状を付与 (Algorithm 2)
3. 追加された質点とその近傍を平滑化
4. 各質点の重さの更新
5. 物理シミュレーションの実行

である。Müller ら [4] は少ない粒子数でクロスシミュレーションを実行し、粒子の近傍にシミュレーションでは用いられない描画用の粒子を配置することで、低い計算コストで細かいシワの表現を実現する手法を提案した。高い品質のシワが得られるが、それはシミュレーションによって生じるものでありユーザの望んだものになるとは限らない。Li ら [5] はユーザの入力から服の 3D モデルにシワを与え、そのシワがシミュレーションによって物理的に再現できるように、モデルの型紙の形や硬さのパラメータを自動的に調整する手法を提案した。この手法ではシミュレーションを用いているため、ユーザが入力した線から自然なよれを生成することが出来る。しかし一方でこの手法では布モデルの型紙の用意が必要であり、型紙の形状がわからないような複雑な服によれを付与することが出来ない。一方我々の手法では、3D モデルに直接よれを付与することができるため、型紙を必要としない。

3. 提案手法

3.1 概要

スカートやシャツなどを想定した単純な形の 3D モデルを、ユーザからのよれの尾根、または谷を想起させる線の入力の後、線の部分がよれの尾根または谷となるような単純な変形を 3D モデルに加え、その後物理シミュレーションを行うことで、その形状を補正した他の剛体との衝突と摩擦なども考慮した自然なよれの生成を実現する。提案手法におけるよれの付与までの流れを Algorithm1 に示す。

3.2 布の物理モデル

布の物理モデルにはバネ質点モデルを採用する。バネ質点モデルは、布を質量が集中した質点群として近似して、それらをバネで接続したものである。バネの接続に関しては、Provot の手法 [6] を参考にする。Provot は、布を構成するバネを構造バネ、せん断バネ、屈曲バネの 3 つに分類した。構造バネは、布としての形を維持しようとするバネであり、3 つのバネの中で最も強いバネ定数を持つ。ここで本稿では、質点と構造バネによって囲まれ定まる形を「格子」と呼ぶ。Provot は上下左右方向に隣接する質点に、四角形型の格子を形成するように構造バネを接続した。またせん断バネは、布をせん断方向に引っ張ろうとする力に対抗するバネであり、四角形格子の対角線上の質点同士を接続する。そして屈曲バネは三次元的に布を折り曲げよう

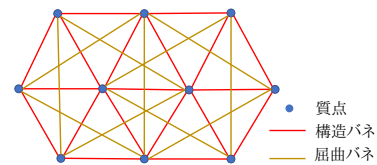


図 1: 質点とバネの接続関係

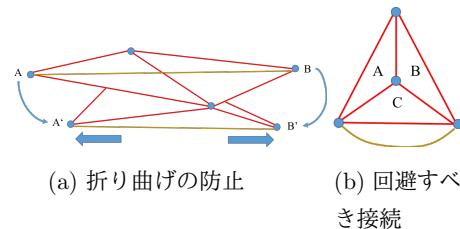


図 2

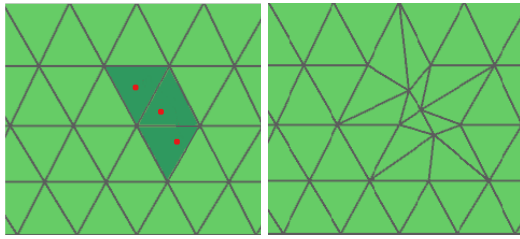
とする力に抵抗するバネであり、上下左右方向の質点に、一つ飛ばしで接続する。

我々はこの手法を基に、バネの接続関係を構築する。しかし我々の手法ではよれの付与のため新たな質点を格子に追加し細分割を行うため、分割によって多角形型の格子を生じる四角形型の格子はモデルとして適していない。そのため、分割に適した形状である三角形型の格子を生成するように構造バネを接続する。

格子の接続関係は図 1 のようになる。青色の点が質点、赤色の線が布を構成する構造バネ、橙色の線が布の折り曲げに対し抵抗する力を模した屈曲バネとなる。構造バネ自体が斜めに接続されているため、せん断バネを必要としない。また折り曲げバネの接続は、ある辺を共有している三角形の組の、その辺に含まれていない質点同士を接続する。屈曲バネは図 2(a) に示すように、A と B に位置する質点が、構造バネの長さを保ったまま A' と B' に移動しようとする時、その三次元的な接近を妨げる役割をする。このように、三角形格子を形成するようにバネを接続しても布の物理シミュレーションが可能となる。しかし通常、布は縦糸と横糸を縫い合わせて作られるため、四角形型の格子では縦方向に接続する構造バネと横方向に接続する構造バネのパラメータを変えることでその再現が容易であるが、三角形格子では構造バネが斜めに接続されるため、縦と横で織り方が違うような布を再現するにはパラメータの調整が困難である。

3.3 格子の細分割

ユーザの入力は線を想定し、その線がよれの稜線または谷線となる。ユーザから入力された線が通過する三角形格子を分割する格子として選択し、各格子ごとに追加する質点の位置を決定する。ユーザの線は点として離散化され、三角形格子内に複数の点が含まれる場合、その重心座標の



(a) 格子内への質点の追加 (赤い点)
(b) 分割後の格子の形状

図 3: 三角形格子の分割

平均を新たな質点の位置とする。その位置に質点を追加し、追加された質点とその三角形を構成する三点を構造バネで接続する。その後、その三角形のある辺を共有している他の三角形があるならば、その三角形の共通辺を構成しない質点と追加質点を接続し、共有辺の構造バネを削除する。ここで単純に3点のみを接続するだけにしないのは、図2(b)のような接続関係が生じるのを防ぐためである。図2(b)のような接続は、三角形Cを形成する構造バネと、三角形Aと三角形Bの接近を防ぐ屈曲バネが同じ質点同士に接続されるため、物理的に意味を成さない構造となっている。従って、追加質点と周りの三点を接続した後、既存の三角形の各辺と交差するように追加質点と他の点を接続し、その辺の接続を切断することで、その発生を回避している。屈曲バネの最終的な接続も考えた、バネの接続アルゴリズムは Algorithm2 のようになる。実際に Algorithm2 に従って三角形を分割した結果を図3に示す。図3の黒線は構造バネによって形成される三角形格子である。屈曲バネは、3.2節で述べたように、格子の一边を挟んで向かい合う質点同士に再び接続する。

また、追加された質点をよれの尾根、または谷とするため、実際のシステムでは格子内に質点を追加する際にその質点を格子の法線方向に動かす。追加質点の初期位置 $p_{newindex}$ は以下の式で決定される。

$$p_{newindex} = p_{initial} + kn \quad (1)$$

ただし $p_{initial}$ はユーザの入力から定まる三角形格子の面内の一点、 n はその三角形の法線ベクトル、 k は任意のスカラーである。 k はユーザが指定することが可能な変数で、 $k > 0$ でモデルの外側に向かう尾根の線、 $k < 0$ でモデルの内側に向かう谷の線となる。また k の絶対値が大きいほど、その隆起の具合は大きくなる。

三角形格子の分割を行い1に従い法線方向に質点を追加するだけでは、図4(b)、図5(b)のようにポリゴンが尖ったような歪な形を生成してしまう。この問題に対処するため、格子を分割し質点を追加した後に、モデルに局所的に平滑化処理を施す。平滑化処理には各質点に均等な重さを割り当てたラプラシアンスムージングを用いる。ラプラシアンスムージングは新たに追加された質点の第2隣接とな

Algorithm 2 三角形格子の分割アルゴリズム

Require: 分割する格子 m_0, m_1, \dots, m_n と追加する質点の位置 $m_i, (i = 0, 1, \dots, n)$ を追加する質点の位置

```

for  $m_i, (i = 0, 1, \dots, n)$  do
     $m_0, m_1, \dots, m_n$  の指定の場所に質点を追加
    その質点と  $m_i$  を構成する3点を構造バネで接続
end for
for  $m_i, (i = 0, 1, \dots, n)$  do
    if  $m_i$  が他の三角形と辺  $e$  で隣接 then
         $e$  を構成する構造バネを削除
        その隣接三角形の  $e$  に含まれない質点と追加された質点を構造バネで接続
        全ての三角形格子を参照し、屈曲バネを接続
    end if
end for

```

Ensure: 分割した三角形格子

る質点まで適用する。単純にラプラシアンスムージングを適用すると形状が縮小してしまうため、Taubin ら [9] の提案した $\lambda\mu$ アルゴリズムを用いてこの問題に対処する。平滑化は、ユーザが選択した回数だけ行う。

また、上述の平滑化によってユーザが指定したよれの稜線や谷線が消失してしまうことがあるため、ユーザは新たに追加した質点に Taubin らが文献 [9] で提案した階層的制約を課すことができる。この制約は、新たに追加した質点群の平滑化をそれらの中でのみ行う制約であり、既存の質点との接続関係が無視されるため、平滑後に新たに追加した質点が稜線や谷線としての形状を保持する。

3.4 物理モデルの更新

本手法ではユーザの入力に応じてモデルに質点を追加し、格子の分割を繰り返すため、モデルに自然長の長いバネと短いバネが混在して存在する。これらの自然長が異なるバネに対し共通のバネ定数を用いてシミュレーションを行うのは不都合を生じる可能性があるため、そのため、自然な挙動になるように自然長に応じてバネ定数を変化させる。あるバネ i の自然長が r_i の時、バネ定数 k_i を以下の式で表す。

$$k_i = \frac{r_i}{r_{avg}} k_{std} \quad (2)$$

ここで、 r_{avg} は初期状態でのバネの長さの平均、 k_{std} は基準となるバネ定数である。

また、ユーザの入力によって格子の大きさが変動するため、モデル上の単位面積辺りに含まれる質点の数は場所によって異なる。従って、下記の式によって単位面積あたりに含まれる質点の合計の重さが等しくなるように質点の重さを更新する。

$$m_i = \sum_{j=1}^N \frac{1}{3} A_j m \quad (3)$$

ここで m_i は質点 i の重さ、 N は質点 i が構成に関わっている三角形の数、 A_j は i が構成に関わっている三角形の面積、 m は初期状態でのモデルの単位面積あたりの重さで

ある。

3.5 物理シミュレーションによる形状の最適化

ユーザは最終的に物理シミュレーションを用いて自然なよれを生成する。本システムではユーザは任意に物理シミュレーションの実行と停止を切り替えることができ、モデルに更によれを追加したい場合は Algorithm1 を再び実行することができる。

3.5.1 剛体との衝突

布モデルと剛体との摩擦によるよれを生成するために、剛体との衝突判定を行う。リアルタイムに剛体との接触を計算するため、Fuhrmann らの手法 [7] を元にして、空間内の一点の座標を入力とし、その場所から剛体の最近傍点への符号付き距離を計算する関数 $\Psi(\mathbf{x})$ を事前に計算する。剛体が動かなければ、距離関数は同じものを使えるので計算が高速である。 $\Psi(\mathbf{x})$ を微分して得られる勾配ベクトル $\nabla\Psi(\mathbf{x})$ は、剛体の法線方向のベクトルとなる。この $\nabla\Psi(\mathbf{x})$ を空間内の全ての場所で計算するのは非常に計算負荷が高いため、空間をいくつか立方体に分割し、その中に含まれる点の距離関数の値を立方体を形成する 8 点の距離関数の値をトリリニアに補間することで求める。またその地点の勾配ベクトルも 8 点の勾配ベクトルをトリリニアに補間することで求めることができる。空間の分割は Frisken らの提案した手法 [8] を用いることによって、誤差が少なくなるように適応的に行う。

布を構成する質点 i と剛体との距離 d がしきい値 ϵ 以下になった時、布と剛体が接触したとみなし、質点の位置 $\mathbf{x}_i(t)$ を更新する。更新は Fuhrmann らと同様の手法 [7] で行う。

3.5.2 時間積分

物理シミュレーションを進行させるための最も単純な方法は Euler 法であるが、この手法は誤差が大きく、またシミュレーションが不安定な状態になりやすい。我々は物理シミュレーションを進行させ各時間刻みごとの質点の位置を計算するために、Leapfrog 法を用いる。Leapfrog 法の誤差は二次精度であり、Euler 法よりも安定性が高くまた高次精度の時間積分法よりも計算負荷が低い。そのためリアルタイムの物理シミュレーションに適した方法である。Leapfrog 法を用いた時間積分は以下のように計算される。

$$\mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \frac{1}{2} \frac{\mathbf{F}_i(t)}{m_i} \Delta t^2$$

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \frac{1}{2} \left(\frac{\mathbf{F}_i(t) + \mathbf{F}_i(t + \Delta t)}{m_i} \right) \Delta t$$

$\mathbf{x}_i(t)$ は質点 i の時間 t での位置、 $\mathbf{v}_i(t)$ は速度、 m_i は質点の質量、 $\mathbf{F}_i(t)$ は質点に働く力である。

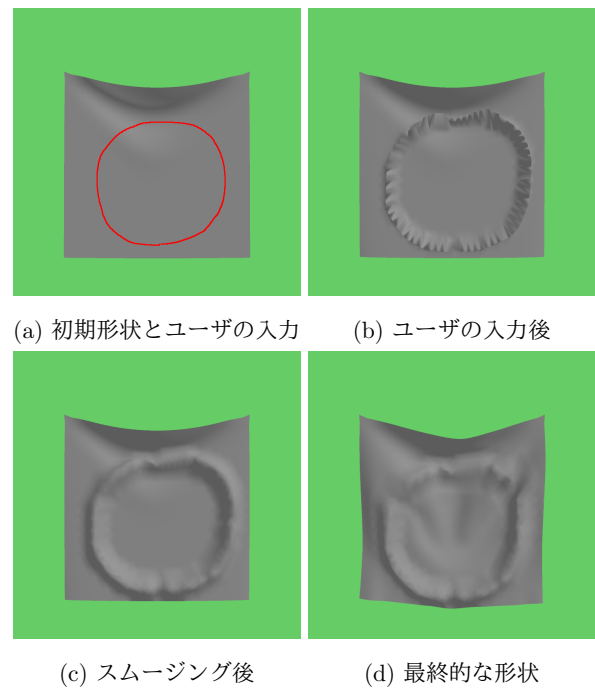


図 4: 提案手法による吊り下げられた布へのよれの付与

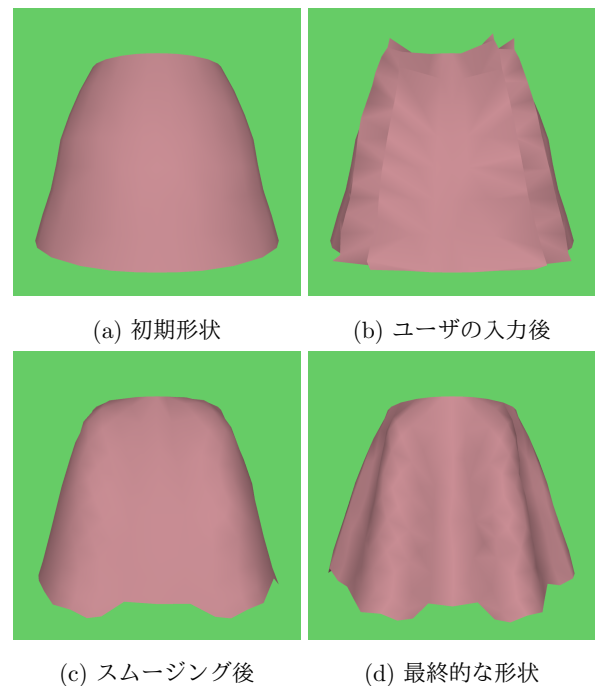


図 5: 提案手法によるスカートの生成

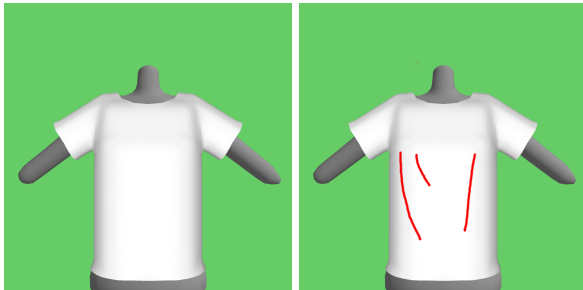
4. 実験結果

4.1 2点で吊り下げられている布に対するよれの付与結果

二点で吊り上げられている単純な形状の布に対し、提案手法でよれが付与できることを確かめる。図 4(a) はモデルの初期形状と、入力を赤の線で表している。(b) は入力後のモデルの形状、(c) は数回ラプラシアンスムージングを行い、円滑化した形状を示している。(d) はシミュレ



図 6: 外力による回転



(a) 初期状態/newline (b) ユーザの入力

図 7: 初期状態とユーザの入力

シヨンをを行った後の、最終的な形状が示されている。布の一部分だけが円状に膨らんでいる現実世界ではありえないようなよれに対しても、物理シミュレーションを用いることによって生成しうる形状を示すことができる。

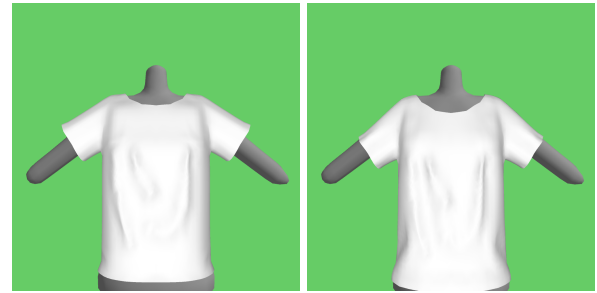
4.2 単純な形のスカートに対しよれの付与結果

より実用的に使われることを想定し、スカートを想定した単純な形の 3D モデルから、ブルームスカートのようなよれを生じさせた結果を示す。ユーザの入力は図 5(a) に表されているような、スカートのよれを想定したものである。結果として図 5(d) に示されたすその広がりおよびよれを付与できることが確認できた。また、外力により回転を加えることで、付与したよれが広がることを確認できた。

4.3 マネキンにかけられているシャツに対してのよれの付与

女性型のマネキンにシャツをかけた際のよれを再現する実験を行った。初期状態とユーザの入力は図 7 に示すとおりである。

線上のよれを付与した結果を図 8 に示す。初期状態におけるユーザの入力から得られた形状は図??に示す通りである。左胸部のよれはシミュレーションが進行すると細くなり、また右胸部のよれはシミュレーションが進行するにつれ消失していくことがわかる。ユーザはシステムが理想の形状を出力した際にシミュレーションを停止することで望んだ形状のよれを得ることができる。



(a) ユーザの入力後に得られ (b) シミュレーションにより
た初期形状 補正された結果

図 8: よれの付与された結果

5. まとめと今後の課題

本稿では、3D モデリングや衣服に関する知識や経験が乏しいユーザでも、手軽に自然なよれを 3D モデルに付与することのできる手法を提案した。提案手法は、まずユーザの入力に合わせて 3D モデルを構成する格子を再分割し、面積を編集することで単純な形のよれを付与する。その後物理シミュレーションを行うことによって、自然な形状へと変形させる。そして実験を行い、本手法によって、吊るされて重力が働くことによって生じるよれや、下地との摩擦によって生じるよれを再現した。

本手法ではよれの生成に際し、物理シミュレーションによってそのよれが消失したり、違った形状を生成することがあり、それが必ずしもユーザの望んだ結果であるとは限らない。今後の課題として、意図したよれの形状を生成できるように新たなユーザの入力方法を導入する必要がある。

参考文献

- [1] Yu Wang, Charlie C. L. Wang, and Matthew M. F. Yuen. Technical section: Fast energy-based surface wrinkle modeling. *Comput. Graph.*, 30(1):111–125, February 2006.
- [2] Changjian Li, Hao Pan, Yang Liu, Xin Tong, Alla Sheffer, and Wenping Wang. Bendsketch: Modeling freeform surfaces through 2d sketching. *ACM Trans. Graph.*, 36(4):125:1–125:14, July 2017.
- [3] Clausius Duque G. Reis, José Mario De Martino, and Harlen Costa Batagelo. Real-time simulation of wrinkles. 2007.
- [4] Matthias Müller and Nuttapon Chentanez. Wrinkle meshes. In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA '10*, pages 85–92, Aire-la-Ville, Switzerland, Switzerland, 2010. Eurographics Association.
- [5] Minchen Li, Alla Sheffer, Eitan Grinspun, and Nicholas Vining. FoldsSketch: Enriching garments with physically reproducible folds. *ACM Transaction on Graphics*, 37(4), 2018.
- [6] Xavier Provot. Deformation constraints in a mass-spring model to describe rigid cloth behaviour. In *Proceedings of Graphics Interface '95*, pages 147–154, 1995.
- [7] Arnulph Fuhrmann, Gerrit Alexander Sobottka, and UNI-BONN Clemens Groß. Distance fields for rapid collision

detection in physically based modeling. 2003.

- [8] Sarah F. Frisken, Ronald N. Perry, Alyn P. Rockwood, and Thouis R. Jones. Adaptively sampled distance fields: A general representation of shape for computer graphics. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 249–254, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.
- [9] Gabriel Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, pages 351–358, New York, NY, USA, 1995. ACM.