

仮想織布を操作するための高速衝突検出法

3P-7

荻野友隆† 亀田能成† 坂口嘉之‡ 美濃導彦† 池田克夫†
†京都大学 ‡東洋紡けいはんな研究室

1 はじめに

仮想空間内で織布を実時間でアニメーションさせ、外部からの操作ができる事を目的とした研究を行っている。

本稿では、高速化に欠かせない自己衝突の高速検出を実現するための一手法として、布の曲率に応じてポリゴン数の削減されたモデルを生成する手法について述べる。

2 織布のモデル

布のモデルとしては、織布が縦糸と横糸から構成されおり、縦/横方向の伸長特性と、斜め方向の剪断特性とが異なる事を考慮し、図1(a)のような正方形のポリゴンで近似する。

ある頂点 P_i から8方向の隣接点までの距離を l_{ik} 、初期長を l_k とする時、エネルギー関数 E を以下のように定義し、外力の影響で P_i の位置が変化した時に、 E の値を最小にするような P_i の位置を収束計算によって求める。

$$E = \sum_i \sum_k^8 (l_{ik} - l_k)^2$$

3 衝突検出

自己衝突を検出する手法として、線分とポリゴンの交差判定を行う方法を採用した。この交差判定の処理は、線分と平面の交差判定と直線とポリゴンの交差判定からなる。

後者の直線とポリゴンの交差判定には、Cyrus-Beck Clipping algorithm [1] を採用した。これは、交点を求める必要がなく、四角形以上の多角形に適用できる反面、凸多面体であるものに限られる。

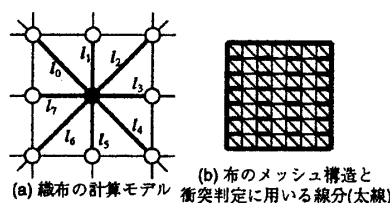


図1: 布の計算モデルと衝突判定用のエッジ

4 衝突検出の高速化手法

織布の自己衝突を検出するためには、前節の交差判定を全ての線分と全てのポリゴンとの総当たりで行う事で実現できる。しかし、一辺 n の頂点からなる正方形の織布の場合、図1(b)のように辺の数は $3(n-1)^2 + 2(n-1)$ 、ポリゴンの数は $2(n-1)^2$ となるため、単純に総当たりで交差判定すると計算量が多くなる。

そこで、以下では線分の数・ポリゴンの数を削減する手法について述べる。これにより衝突計算回数を大幅に抑える事ができる。

4.1 線分の削減

上のモデルの場合、一つの頂点から6本の線分が出ており、その全てについて交差判定を行うのは無駄である。衝突判定の影響を受けるのは頂点であるから、全ての頂点を含む最低限に線分について判定すれば全ての自己衝突を検出する事ができる。ただし輪郭については交差を防ぐため全ての線分について判定する必要がある。

つまり、図1(b)の太線部のように、輪郭は全ての線分について判定し、内部は二つの頂点に対し一つの線分を対応させた部分についてのみ判定すればよい。上の例の場合、辺の数は $n(n-2)/2 + 4(n-1)$ となる。約84%の削減が実現できる。

4.2 ポリゴンの削減

布の曲率が小さい部分において、複数の小さなポリゴンをまとめ一枚の大きなポリゴンで近似するこ

A collision detection method for interacting virtual woven cloth

Tomotaka Ogino, Yoshinari Kameda, Yoshiyuki Sakaguchi, Michihiko Minoh, Katsuo Ikeda

†Kyoto University, ‡Toyobo Keihanna Laboratory

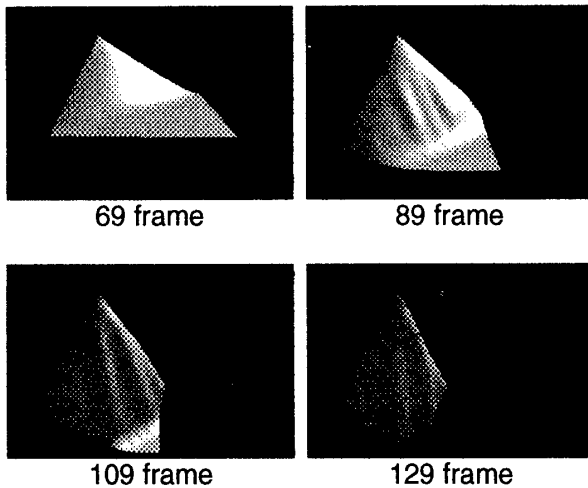


図 2: シミュレーションの様子

とで、全体におけるポリゴンの数を削減することができる。

曲率はマイクロな視点において、あるエッジを共有する二つのポリゴンの曲がりの度合で表わす事ができる。すなわち、エッジを共有するポリゴン P_1, P_2 の法線ベクトルを N_1, N_2 とした時、 $\cos \theta = (N_1 \cdot N_2) / |N_1| |N_2|$ と簡単に表現できる。まったく平坦な場合は最大値 1 をとる。

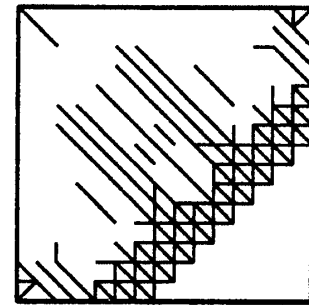
全てのエッジにおいてこの $\cos \theta$ を計算し、ある閾値以上であればそのエッジを消去する。

この情報をもとにポリゴンを削減するが、Cyrus-Beck Clipping が凸多角形にのみ有効である事を考慮し、凸多角形になるように再分割を行う。

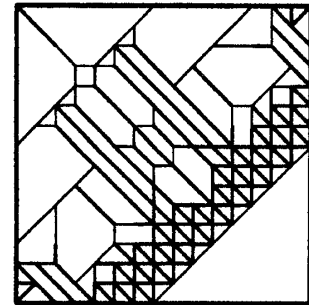
5 実験

16^2 個の頂点からなる正方形の布の端の一点を固定し、全体に重力をかけた環境でシミュレーションを行なった。(図 2)

89 フレーム目において、閾値を 0.99 とした場合のエッジ消去の様子を図 3(a) にしめす。消去できずに残ったエッジは、周囲で大きな曲がりが発生した箇所を表わす。また、再分割の結果を図 3(b) にしめす。このフレームにおいては削減前のポリゴン数 450 に対し削減後は 111 となり、約 75% 削減する事に成功した。ポリゴン削減で 4 倍、エッジ削減と合わせて 25 倍の高速化が期待される。



(a) エッジ消去の様子(89frame)



(b) 分割されたポリゴン(89frame)

図 3: エッジ消去とポリゴン分割

6 おわりに

本稿では、衝突検出の高速化手法の一つとして、布の曲率を用いたポリゴンの削減方法について述べた。

以上の実験は正方形の布に限っていたが、現在任意の形状の布を扱うため輪郭から自動でパッチを生成する処理を実装中である。今後の課題としては、収束計算の高速化と操作環境の構築が挙げられる。

参考文献

- [1] Matthew Moore, Jane Whihelms, "Collision Detection and Response for Computer Animation", Proc. SIGGRAPH'88, Computer Graphics, Vol.22, No.4, pp.289-298, 1988.