

物理シミュレータを用いた 衣服移動を抑えた折り畳み軌道の獲得

信州大学大学院 総合理工学研究科 工学専攻 ○津田翔, Solvi ARNOLD, 山崎公俊

本稿では、物理シミュレータを用いた衣服の折り畳み軌道の獲得について報告する。日常生活の中で布製品を扱うとき、それは主に人間の手によって行われ、自動化される例はほとんどない。この理由として、ロボットにとって不定形性を持つ布製品を操作は困難である点が挙げられる。衣服を折り畳むとき、操作軌道が悪ければ隙間やしわが発生する。また、折り畳み動作によって衣服自体が机上を移動することにより、目標の折り畳み位置も移動するため、折り畳み動作自体の破綻を招く可能性がある。そこで本研究では、衣服そのものが移動しないような折り畳み軌道を物理シミュレーションによって獲得することを目指す。

キーワード：物理シミュレータ, 衣服, 折り畳み, 軌道

1. 緒言

日常生活において布製品を取り扱う場合、人の手で行われることがほとんどであり、自動化されている例は少ない。その理由として、不定形物である布製品の操作はロボットにとって困難な場合が多いという点が挙げられる。すなわち、布製品は不定形性を持つため形状予測が難しく、また、センサ情報処理による状態推定にも高い不確実性を伴う。

こうした背景から、我々は双腕型移動マニピュレータによる衣服の展開と折り畳みを行ってきた。しかし、衣服には様々な種類が存在し、形や大きさだけでなく素材による堅さ、曲がりやすさ、しわのできやすさなどの違いがあり、実物の衣服を用いた様々な条件下で実機実験を行うことは時間やコストがかかり効率的ではない。そこで、本研究では物理シミュレータを用いることでこの問題を解決する。

本研究の目的は、双腕型ロボットを活用して衣服を折り畳むことを目指し、衣服移動を抑えた折り畳み軌道を獲得することである。衣服を折り畳む際に、折り畳みの軌道は重要な要素となる。折り畳みの開始および終了位置が同じであったとしても、異なる折り畳みの軌道は異なる折り畳み結果を生む。例えば、折り畳み動作中に衣服が机上を移動すると、目標の折り畳み位置も移動してしまう。それを考慮せずに所定の軌道で折り畳みをおこなっていくと、シワや隙間が生じ、それが蓄積されることで折り畳み作業が破綻する。それを考慮した既存の試みとして、Yinxiao ら[1]の研究がある。ここでは、物理シミュレータと Levenberg-Marquardt 法を用いて軌道の最適化を行い、ロボットによる衣服の折り畳みを達成した。本研究では、Yinxiao らの手法を参考にして、衣服移動を抑えた軌道の獲得を目指す。

2. 衣服のシミュレーション

衣服の動きと変形をシミュレートするために、3次元コンピュータグラフィックスソフトウェアである Blender を使用する。物理シミュレータより作成した衣服のメッシュモデルを Fig. 1 に示す。

実際の折り畳み環境を正確にシミュレートするためには 2 つの重要なパラメータがある。1 つ目は布の特性であり、2 つ目は布と机との間の摩擦力である。

2.1 材料特性

衣服をシミュレーションする際に定めて置くべき重要な特性は、布の伸びである。衣服の折り畳みにおいては、衣服を持ち上げることや引っ張ること、動かすことで衣服が伸びる。しかし、不適切な伸びは現実世界と異なった衣服の形状を作り出し、現実とはかけ離れたシミュレーション結果となる可能性がある。そこで、適切な伸びを設定することで実際の衣服の伸びを再現する。

2.2 摩擦力

衣服と机の間の摩擦力は机の状況によって変化することが考えられる。もし、机がテーブルクロスなどで覆われている場合、表面が粗くて滑りにくくなるが、そうでない場合は滑りやすくなる。すなわち、衣服と机の間の摩擦は衣服の移動に影響を与え、プロセス全体を損なう可能性がある。そこで、机と衣服の間の摩擦を正確に設定することで現実世界に近い作業環境を構築する。

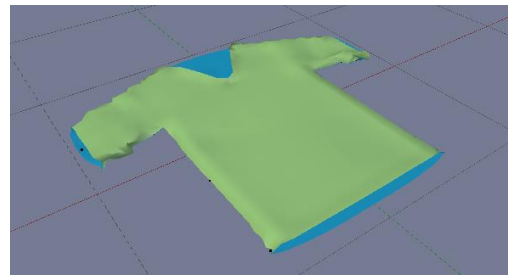


Fig. 1 Clothes model

3. ベジエ曲線

本研究では、ほとんどの曲線を表現できるベジエ曲線を使用して折り畳み軌道を描く。ベジエ曲線 $T(t)$ は $N + 1$ 個の制御点から得られる N 次曲線であり、制御点 $P_k = P_0, P_1, \dots, P_{N-1}$ を用いて以下の式で表される。

$$T(t) = \sum_{k=0}^N B_k^N(t) P_k \quad (1)$$

ここで、 $B_k^n(t) = \binom{n}{k}(1-t)^{n-k}t^k$ はバーンスタイン基底関数である。 t が 0 から 1 まで変化するとき、 P_0 と P_N を両端とするベジェ曲線が得られる。 また、一般には両端以外の制御点は通らない。 本研究では $N = 3$ を使用する。 この具体例を Fig.2 に示す。 把持点となる折り畳み軌道の開始位置 P_0 および目標位置 P_3 を指定する。 ベジェ曲線軌道を黄色の線として示し、軌道に沿って目標位置まで折り畳む。 中間制御点である P_1 、 P_2 を移動させることでベジェ曲線の形を変化させる。

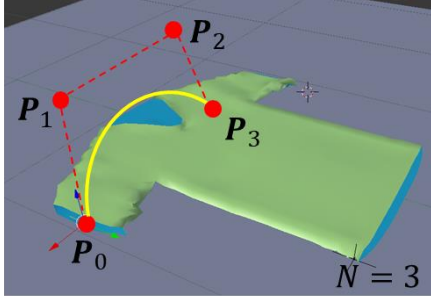


Fig. 2 An example of folding trajectory

4. 軌道の調整方法

P_0 と P_3 は Fig. 2 のように把持点となる折り畳み軌道の開始位置および目標位置に固定される。 次に中間制御点 $x = (P_1, P_2)$ は (2) 式を使用して軌道の調整を行う。

$$C(x) = l_x + \alpha D(S_t, S_x) \quad (2)$$

ここで、 $C(x)$ は 2 項のコスト関数である。 第 1 項は、軌道長さである l_x に対して時間とエネルギーにおける効率的な折り畳み経路を選択するためのものである。 第 2 項は、 x によって得られる折り畳み軌道からなる形状 S_x と、所望の折り畳まれた形状 S_t を比較することで形状のズレを求めるとの項である。 また、重さ α は 2 つの項のバランスを取るためのものである。 本研究では $\alpha = 0.01$ とする。

(3) 式の $D(S_t, S_x)$ を具体的に書くと、次のようになる。

$$D(S_t, S_x) = \sum_i \|q_i - y_i\| \quad (3)$$

ここで、 q_i は Fig.3 に示すようにシミュレーションされた形状 S_x 上の i 番目のメッシュの頂点、 y_i は目標形状 S_t 上の対応する i 番目のメッシュの頂点である。

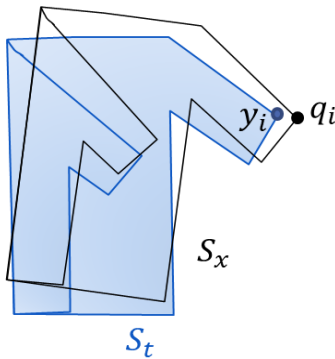


Fig.3 clothing shape dissimilarity

5. 実験

双腕型ロボットで衣服を折り畳むことを想定し、袖と裾の

角を把持位置として衣服の折り畳みを行うこととした (Fig.4)。 中間制御点である P_1 、 P_2 の初期位置を (4)、(5) 式で求めた。

$$P_1 = \frac{2}{3}P_0 + \frac{1}{3}P_3 + h\|P_0 - P_3\|e_v \quad (4)$$

$$P_2 = \frac{1}{3}P_0 + \frac{2}{3}P_3 + h\|P_0 - P_3\|e_v \quad (5)$$

ここで、 e_v は上方向の単位ベクトル、 h は一定値の $1/3$ であり、初期軌道が結び目間で等しい水平度を持つ。 本研究では、Fig.4 に示すように 2 つの軌道があり、それぞれ中間制御点 x によって生成された現在の軌道から僅かに x を移動させ、生成されたそれぞれの軌道における $C(x)$ の値を合計し、より小さな値が得られる x を探索した。

1 つの軌道に対して、中間制御点 x を 1 つの座標軸方向に 1cm 単位で移動させた。 $C(x)$ の値がこれ以上小さくならないと判断した場合、別の座標軸方向に対しても同様に行い $C(x)$ の値を取得した。 これを軌道 1、軌道 2 に対して合計 100 回繰り返し、 $C(x)$ の合計値が最も小さな値となったものが衣服移動を最も抑えた折り畳み軌道とした。 実験結果を Table.1 に示す。 軌道長さと衣服のズレが小さくなる折り畳みを行うことで、効率歴な経路で衣服移動を抑えた折り畳み軌道を求めることができた。

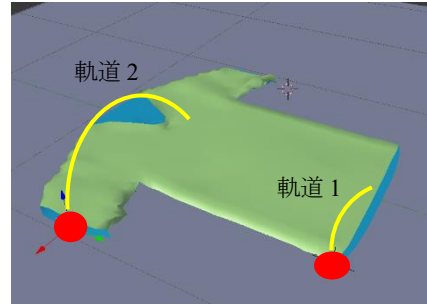


Fig.4 An example of clothing folding

Table.1 recognition result

	軌道 1 の l_x	軌道 2 の l_x	$\alpha D(S_t, S_x)$	$C(x)$ の合計値
初期軌道	13.68	17.68	30.56	92.48
最終軌道	9.70	13.00	6.70	36.10

6. 結言

本稿では、衣服の折り畳み軌道に注目し、衣服の移動を抑えた折り畳み軌道を獲得する手法を提案した。 実物の衣服を用いた実験は効率的ではないことから、物理シミュレータを活用した。 折り畳みにおける軌道の長さから効率的な折り畳み経路を求め、所望の折り畳まれた形状とのズレを取得することで移動を抑えた折り畳み軌道を獲得した。 今後は、軌道獲得の処理を効率化し、軌道獲得を実ロボットでおこなう。

参考資料

- [1] Yinxiao et al.:Folding Deformable Objects using Predictive Simulation and Trajectory Optimization, arXiv:1512.06922v1, 2015 年
- [2] G. Farin. Curves and Surfaces for Computer Aided Geometric Design. Academic Press, 1988
- [3] <https://www.blender.org/>