

キーフレームを用いた粘弾性体の自由変形アニメーション生成

竹内 亮汰[†] 石川 知一[†] 柿本 正憲[†]

[†] 東京工科大学 メディア学部 メディア学科

1. はじめに

物理的な特徴を考慮して物体の動きを再現したコンピュータグラフィクス (CG) は、映画やゲームのようなエンターテインメント分野において特に需要がある。現実に近い表現を行うことが演出できるが、物理的性質を持ちつつユーザの思ったように動くようなアニメーションの制作は困難である。

本研究では、粘弾性体的な挙動をしつつ、ユーザの入力におおよそ追従するアニメーションの生成を行う。粘弾性体とは、粘性体と弾性体の中間の性質を持つ物質である。力を加えると粘性体のように変形し、加えた力を取り除くと弾性体のようにある程度は形状が元に戻るが、完全には戻らず変形の影響が残る。また、強く引っ張るとちぎれたり、接触させ続けると結合したりといった現象も特徴的である。代表的な例にスライム (図 1) が挙げられる。この粘弾性体をアニメーションに用いることで、液体と個体の両方の性質を持ったようなアニメーションを容易に制作できるようになると期待できる。



図 1 粘弾性体の例 (スライムの写真)

2. 関連研究

本手法では、高松ら[1]が行った粘弾性体のシミュレーションを拡張する。これは、SPH 法 (Smoothed Particle Hydrodynamics Method) と Shape Matching 法で粘性体と弾性体の性質を表現し、組み合わせることでリアルタイムな粘弾性体のシミュレーションを行う。

本研究では、この提案法を改良し、ユーザがコントロール可能な粘弾性体のアニメーション生成手法を提案する。

3. 提案手法

3.1. 処理の流れ

本手法において、ユーザにはキーフレームモデルとキーフレームタイミングの入力を行ってもらう。その後は自動でアニメーションの出力を行い、納得のゆく結果が出るまで入力と出力を行う。

本手法で行う手法の流れを図 2 に示す。

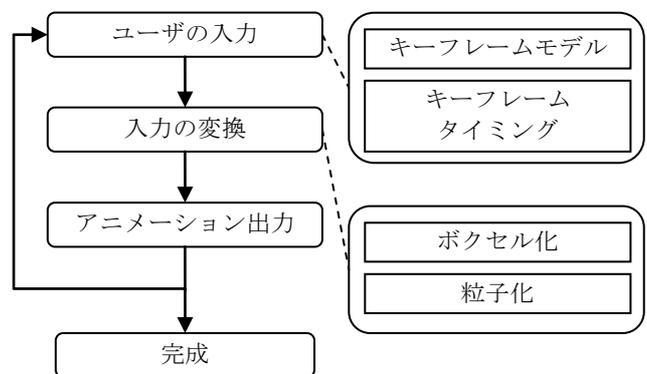


図 2 処理の流れ

3.2. SPH 法と Shape Matching 法の速度合成

高松らの方法[1]では、SPH 法で粘性体的な特徴を、Shape Matching 法によって弾性体的な特徴を表し、それぞれの計算で得られた速度を線形補間することで粘弾性体を表現する。SPH 法による速度を \mathbf{u}_i 、Shape Matching 法による速度を \mathbf{v}_i とした場合、粘弾性体の速度は以下の式で表せられる。

$$\mathbf{v}_i^{final} = \alpha \mathbf{u}_i + (1 - \alpha) \mathbf{v}_i \quad (1)$$

ここで、 α は線形補間用のパラメータで $0 \leq \alpha \leq 1$ の値を取りうる。 α が 0 に近いほど、弾性体のように振る舞い、 α が 1 に近いほど、流体としての挙動を再現できる。各粒子は、式(1)で計算された速度を時間積分することによって位置を更新する。

3.3. キーフレームアニメーションの導入

この節では、我々が検証した 3 つの方法について説明する。

提案法は粒子ベースのシミュレーションであるため、キーフレームとして入力された 3D モデルも粒子データに変換する。変換された粒子データをキーフレーム粒子と呼ぶ。

単純なキーフレームアニメーションの方法と

"Deformable viscoelastic animation using keyframes."

[†] Ryota TAKEUCHI, [†] Tomokazu ISHIKAWA,

[†] Masanori KAKIMOTO

Tyokyo University of Technology

して、ある時刻におけるシミュレーション粒子から、キーフレーム粒子に直接移動させる方法が考えられる。すなわち、ある時刻 t における i 番目の粒子位置を \mathbf{x}_i^t とし、対応するキーフレーム粒子の位置を \mathbf{r}_i とすると、速度ベクトル \mathbf{v}_i^t を以下の式で決定する。

$$\mathbf{v}_i^t = k \frac{\mathbf{r}_i - \mathbf{x}_i^t}{|\mathbf{r}_i - \mathbf{x}_i^t|} \quad (2)$$

ここで、 k は速度ベクトルの大きさを決めるためのパラメータである。この方法では、粘弾性体の特徴を損なうことになるため、我々は以下の2つの方法を実装し、検証した。

- i) 式(1)と式(2)を補間して速度を決定する
- ii) 参照形状を時間的に変化させる

i)の方法では、次の式によって最終的な更新速度を計算する。

$$\mathbf{v}_i^{final} = \beta(\alpha \mathbf{u}_i + (1-\alpha)\mathbf{v}_i) + (1-\beta)k \frac{\mathbf{r}_i - \mathbf{x}_i^t}{|\mathbf{r}_i - \mathbf{x}_i^t|} \quad (3)$$

ここで、 β は線形補間用のパラメータで $0 \leq \beta \leq 1$ の値を取る。 β が 0 に近いほど直接的にキーフレーム粒子に向かうことになるが、粘弾性体の動きは損なわれる。 β が 1 に近いほど、粘弾性体としての挙動を残すが、キーフレームには追従しなくなる。本研究では、 β は 0.1~0.7 の値を使用した。

ii)の方法では、最終的な更新速度を(1)によって決定する。参照形状を数タイムステップ前の形状から、キーフレーム形状による参照形状へと徐々に変化させる。粘弾性体シミュレーションによる参照形状の粒子位置を \mathbf{x}_i^{ref} 、目標のタイムステップ t_{key} におけるキーフレーム粒子の位置を \mathbf{k}_i とした場合、あるタイムステップ t において参照する粒子の位置 $\mathbf{x}_i^{final-ref}$ を以下の式で計算する。

$$\mathbf{x}_i^{final-ref} = \frac{t_{key} - t}{t_{key} - t_0} \mathbf{x}_i^{ref} + \frac{t - t_0}{t_{key} - t_0} \mathbf{k}_i \quad (4)$$

ここで t_0 はシミュレーションの開始時か、直前にキーフレームを指定したタイムステップとする。

4. 結果

まず、3.3 節で説明した2つの手法についてキーフレームとして入力した粒子群と、シミュレーション粒子との距離差を測定した。計測するタイムステップはユーザがキーフレーム指定

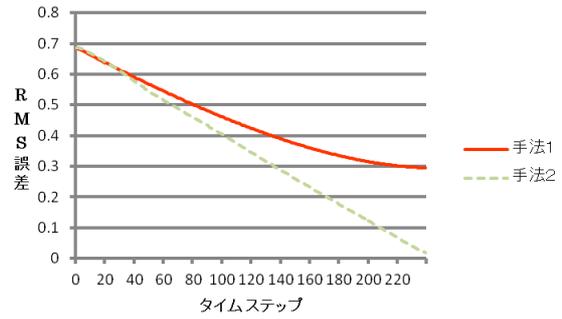


図3 キーフレーム粒子とシミュレーション粒子のRMS 誤差の時間変化

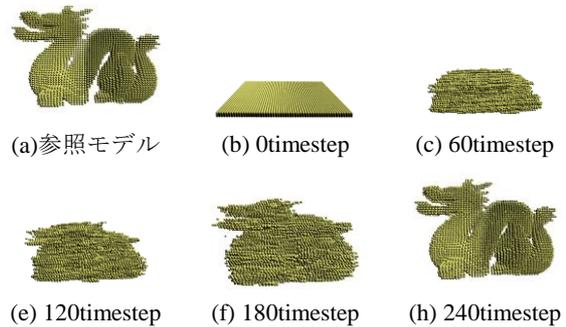


図4 結果画像

を行ったタイムステップとし、距離差については RMS 誤差を用いた。RMS 誤差の時間経過のグラフを図3に示す。この結果、参照形状を時間的に変化させる方法が有用であることがわかった。図4(a)にキーフレームとして用いた3Dモデルを示す。このキーフレームを240タイムステップとした場合のシミュレーション結果を図4(b)~(h)に示す。指定した時間に希望の粘弾性体の形状に変形していく様子が見える。また、計算時間についても、従来の粘弾性体シミュレーションの高速性を損なうことなく計算できた。

5. おわりに

粘弾性体のシミュレーションにキーフレームを導入することによって、ユーザの希望に沿うようなアニメーションを生成することができた。Shape Matching 法と SPH 法の組み合わせによって粘弾性体を表現している手法において、Shape Matching 法の参照形状を時間的に変更していく方法をとることで、粘弾性体の動きを損なうことなくキーフレームに則した動きを再現することができた。

参考文献

[1] 高松 賢二, 金井 崇. 高速かつ頑健な点群ベースの粘弾性体アニメーション, グラフィクスとCAD/Visual Computing 合同シンポジウム, No.16, 2011.