1L−3

有理ベゼー曲線を応用した、CG アニメーションのための補間アルゴリズム

井川 勝、 中西 正和
慶應義塾大学

1 はじめに

コンピュータでアニメーションを製作する技術の中に、キーフレームアニメーションと呼ばれるものがある。これは、アニメータが製作したいアニメーションの絵の中でキーとなる絵を与えておき、それを計算機が補間することによって製作するものである。この手法を用いる時、補間をどのようなアルゴリズムを用いて行なうかによって生成される軌道が異なり、出来上がるアニメーションが異なってくる。

キーフレームとしてアニメータはある時刻における物体の状態を与えるものとする。それをもとに、物体の特徴的な点の x,y,z 方向の位置と速度、あるいは関節物体ならば関節の角度と角速度が得られるものとする。今、ある2点の時刻、変量、速度が与えられた時、それらを満たすような、時間をパラメータとした変位を求めることに帰着して、その変位を求める方法について述べる。その際に、

- 速度の変化が大き過ぎない
- 冗長な軌道が生まれない

$\cdots\cdots\star$

といった条件を考慮して補間を行ないたい。

Witkin は、物体の軌道として、速度の2乗と加速度の2乗の和として求まる評価関数を与え、それを最も小さくする軌道を採用することを述べている [Witkin 88]。しかしこれは複雑な計算を必要とし、計算量が多くなると思われる。そこで、本報告ではあらかじめ3次の分数式を与え、それに同じような評価関数を与えることによって最適な軌道を求める手法について述べる。

2 3次の有理ベゼー曲線の応用

2点の時刻、位置、速度が与えられた時の変位を表す式を、3次の有理ベゼー曲線を表す式をパラメトリックに用いて表現することにした。ここで有理ベゼー曲線の一般的な形の説明は省略し、それを特殊化した形について説明する。それは次のような式である。

$$R(t) = \frac{B_0^3(t)wP_0 + B_1^3(t)P_1 + B_2^3(t)P_2 + B_3^3(t)wP_3}{B_0^3(t)w + B_1^3(t) + B_2^3(t) + B_3^3(t)w}$$

ただし、$B_i^3(t) = \frac{3!}{i!(3-i)!}t^i(1-t)^{3-i}$ である。パラメータ t によって表される時刻が 0 と 1 の時に位置 P_0, P_3, 速度 V_0, V_1 が与えられた時、その補間は、

$$P_1 = P_0 + \frac{1}{3}wV_0$$

$$P_2 = P_3 - \frac{1}{3}wV_1$$

として $R(t)$ を計算すればよい。なぜならば、各点における速度は次のように、

$$\frac{dR(t)}{dt}|_{t=0} = \frac{2(P_1 - P_0)}{w} = \frac{3(P_0 + \frac{1}{3}wV_0 - P_0)}{w} = V_0$$

$$\frac{dR(t)}{dt}|_{t=1} = \frac{2(P_3 - P_2)}{w} = \frac{3(P_3 - (P_3 - \frac{1}{3}wV_1))}{w} = V_1$$

となるからである。

w を 0.0 から 1.0 の間でいろいろ変えると、異なる補間が行なえる。時刻 0、1 で、位置 P_0, P_3、速度 V_0, V_1 である時の補間で、$\star$ で示された条件を満たす w を求めるために Witkin の考えに従って、

$$f = \int_0^1 (A \times |\frac{d^2R(t)}{dt^2}| + B \times (\frac{dR(t)}{dt})^2)dt,$$

という評価関数を導入する。これは、加速度の絶対値 $|\frac{d^2R(t)}{dt^2}|$ を小さくすることは急激な速度の変化を少なくすることに寄与し、速度の2乗 $(\frac{dR(t)}{dt})^2$ を小さくすることは無駄な動きを少なくすることに寄与する。

一般に、時刻 t_0 、位置 P'_0 、速度 V'_0 の点から時刻 t_1 、位置 P'_3 、速度 V'_1 の点までを補間する時は、

$$P_0 = P'_0$$

$$P_3 = P'_3$$

$$V_0 = (t_1 - t_0)V'_0$$

$$V_1 = (t_1 - t_0)V'_1$$

として、$R(t)$ を $0 \leq t \leq 1$ で計算する。この $R(t)$ を時刻 $(t_1 - t_0)t$ における変位とすればよい。

本報告のために作成したプログラムでは、評価関数の係数は

$$A = 0.6, \quad B = 1$$

Interpolating Method for Computer Animation Using Rational Bezier Curve
Masaru IGAWA, Masakazu NAKANISHI
Keio University

としている。

与える点を端点の 2 点だけではなく、連続したいくつかの点、例えば、$Q_1, Q_2, \cdots, Q_n$ である時は、$Q_iQ_{i+1}(i = 1, \cdots, n-1)$ のセグメントに分けて補間を行なうという方法が考えられる。Q_1, Q_n での傾きは与えられるものとし、Q_i での傾きは、

$$\frac{Q_{i-1}Q_i\text{の傾き} + Q_iQ_{i+1}\text{の傾き}}{2}$$

とするとよいであろう。

3 結果

いくつかのデータに対して、実際にどのような補間が行なわれるかを以下に示す。

まず、与えたデータを 5 通り示す。表 1 は、与えた時刻、位置、速度と、その時評価関数を最も小さくする w を示している。それぞれのデータについて、補間した結果を図 1 に示す。横軸に時刻を表すパラメータ t をとり、縦軸には変位を表す $R(t)$ をとった。それぞれの曲線はカーブの内側から重み係数 w が 0.1, 0.2, $\cdots$, 1.0 の時のものである。そのうち実線で表してあるものが評価関数を最小にするものである。丸印は P_0, P_1, P_2, P_3 の値を表す。すなわち両端の丸印は $t = 0, 1$ における位置を表し、左の丸印から 2 つ目の丸印への方向が $t = 0$ での速度、4 つ目の丸印から 3 つ目の丸印への方向が $t = 1$ での速度を表すことになる

3 次のベゼー曲線やスプライン曲線といったよく使われる補間は 3 次の多項式からなる関数で表現されるが、これは上の例で重み係数 w が 1.0 の時に相当する。

(a) (e) (f) の場合、選択された w の時と $w = 1.0$ の時とはあまり差異がなかった。

(c), (d) の場合では、3 次関数を使うと無駄な軌道が得られてしまう。本手法ではいずれの場合も適当な重み係数が選択されて、無駄のない軌道が生成された。(b) の場合も、わずかにそれが見られる。

これを応用すると、例えばキーとなる点 $R(t_{i-1})$, $R(t_i)$, $R(t_{i+1})$ が与えられた時に、$R(t_{i-1}) > R(t_i)$ かつ $R(t_i) < R(t_{i+1})$ かつ $\frac{dR(t_i)}{dt} = 0$ ならば、$t = t_i$ のときに必ず最小になるような軌道を生成することができる。これは、制限つきの軌道を得るのに効果があると思われる。

参考文献

[Witkin 88] Andrew Witkin, Michael Kass, "Spacetime Constraints," acm SIGGRAPH, Vol 22, No.4, August 1988, p159-168.

[Toriya 91] 鳥谷浩志, 千代倉弘明編著, 「3 次元 CAD の基礎と応用」, 共立出版, 1991.

表 1: 補間の実験のために与えたデータ

(a)

時刻	0.0	1.0
位置	100	100
速度	-300	300
重み	1.0	

(b)

時刻	0.0	1.0
位置	100	100
速度	-300	100
重み	0.6	

(c)

時刻	0.0	1.0
位置	100	40
速度	-300	0
重み	0.5	

(d)

時刻	0.0	1.0
位置	100	40
速度	-800	0
重み	0.2	

(e)

時刻	0.0	1.0
位置	120	20
速度	0	0
重み	0.3	

(f)

時刻	0.0	1.0
位置	60	60
速度	200	200
重み	0.7	

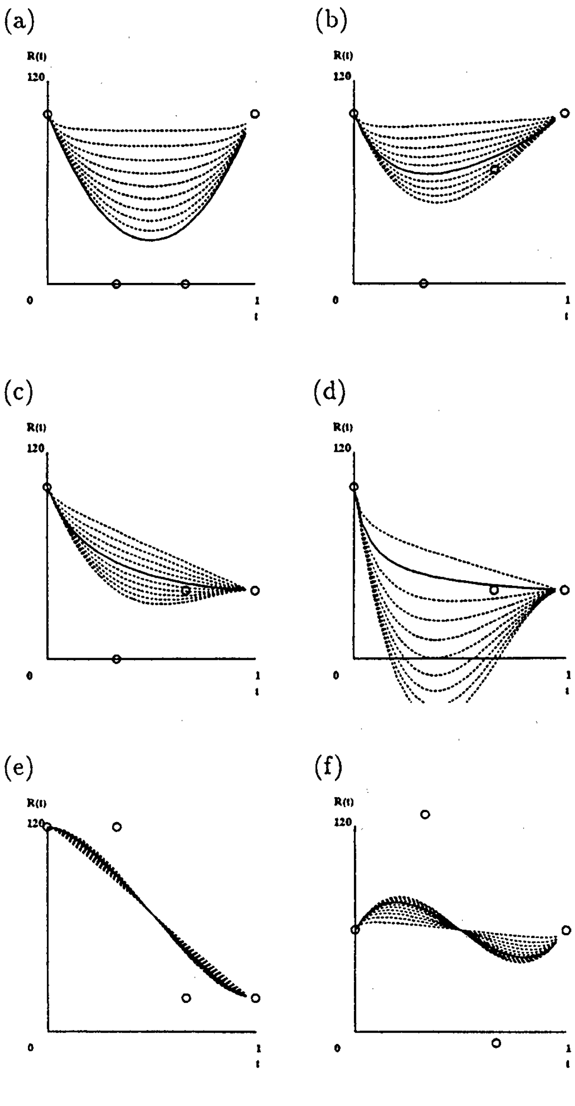

図 1: 軌道の様子