

3次元アニメーションデータベースシステム MOVE の基本設計とその評価

1 R-2

金子邦彦, 黒木進, 近藤祐介, 牧之内顕文

(九州大学工学部)

1. はじめに

3次元コンピュータアニメーションの基本問題は、アニメーションの登場物に望み通りの動きをさせることである。動きの記述法としては、簡潔な記述で自然な動きが指定できるものが望ましい。

この観点から、従来、さまざまな研究が行われてきた。代表的な手法は、現実世界の物理法則を計算機内でシミュレートして、アニメーションの登場物の動きを求めるものである。登場物の状態変数を支配する何らかの運動法則(としばしば状態変数の初期値)を指定しておけば、物理的に正しくて、現実感を持った自然な動きが、計算機により自動的に計算できるという特徴がある。

物理法則シミュレーションによるアニメーションの代表例が、力学アニメーションである。図1に示した力学振り子は、力学アニメーションの典型的な応用例である。図1の例は簡単なものだが、モーター、バネ、摩擦抵抗などの要素を加えることで、人体、ロボット、各種の機械部品などの幅広い分野への応用が期待されている。

図1では、振り子の各部品は、ニュートンの運動法則、作用反作用の法則、エネルギー保存の法則という力学的法則に従って運動している。これらの法則は状態変数(位置、速度、主軸の向き...)の微分形になるので、これら状態変数の初期値と、外力(この場合、重力)とから、物体の運動が定まることになる。

初期値と外力を指定する方法の欠点は、最終的に物体がどのような動きをするのかを予想するのが困難だということである。従って、思い通りの動きをさせるのが難しい。このための有効な解決法は、物体の運動について何らかの拘束を指定できるようにすることである。図1で言えば、天井は静止しており、天井と振り子は、常に接点で接しながら回転している。拘束の指定により、最終的な動きの指定が容易になる。

拘束を利用した力学アニメーションの概念図を図2に示す。ここで、物体が拘束を満たすことを保証しているのが、物体間の拘束力の存在である。拘束力は、拘束の種類と、物体の状態変数の値とから求めることができる。拘束を利用したアニメーションシステムでは、次のような条件が考えられよう。

- i) 物体の構造について、前もって制限を加えるのは望ましくない。
- ii) ある物体について、時間の経過による拘束の変動や、複数の種類の拘束を指定を可能にする。
- iii) 物体の運動を指定するのに十分な量の拘束が指定されたかどうかを判定する仕組みがあるとよい。
- iv) システム準備の拘束処理ルーチンの他に、利用者が新しい種類の拘束を簡単に実装できるようにする。

すでに、我々は、物体間の拘束から拘束力を求める方式について、オブジェクト指向の観点から考察を行なった^[3]。今回は、上記の4条件を念頭に、MOVEのアニメーションモデルを報告する。我々は、(1)を満足するために、まず、一般的な運動方程式の立式を行なうことにした。次に(2),(3)の条件のため、拘束を表す方程式を、一般的な運動方程式とは独立に立式した。物体の拘束力を求めるには、運動方程式と、拘束式とを連立させれば良い。(4)のためには、オブジェクト指向の概念を利用することにした。

The Design and its Evaluation of 3-Dimensional Animation Database System - MOVE
Kunihiko KANEKO, Susumu KUROKI, Yusuke KONDO, Akifumi MAKINOUCI, Kyushu University

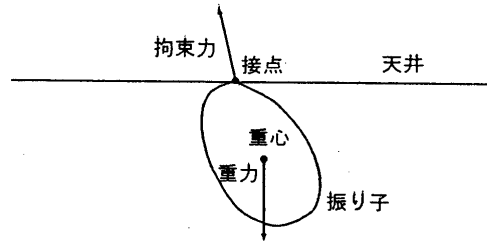


図1: 力学アニメーション

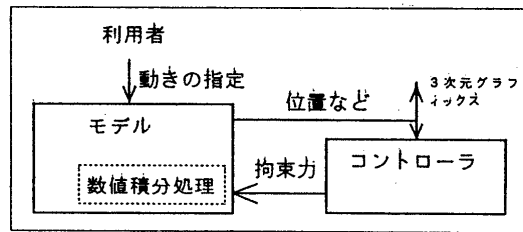


図2: 拘束による力学アニメーション

2. 剛体の力学

我々は、アニメーションの登場物の基本要素として、剛体を選ぶことにした。剛体が互いにつながっているような物体を例として、アニメーションのモデルの検討を行なった。

剛体のパラメータは、図3に示す通りである。質量と慣性モーメントについては、それぞれの剛体に固有の値を持つ。それ以外のパラメータは時間とともに変化する。

剛体の運動は次の2つの方程式に従う。(1)は剛体の重心に関するもので、(2)は回転に関するものである。なお、右上の添字は各剛体の番号に対応している。

$$\left. \begin{aligned} m^i \frac{V_x^i}{dt} &= F_x^i \\ m^i \frac{V_y^i}{dt} &= F_y^i \\ m^i \frac{V_z^i}{dt} &= F_z^i \end{aligned} \right\} (1) \quad \left. \begin{aligned} I_x^i \frac{W_x^i}{dt} - (I_y^i - I_z^i) W_y^i W_z^i &= N_x^i \\ I_y^i \frac{W_y^i}{dt} - (I_z^i - I_x^i) W_x^i W_z^i &= N_y^i \\ I_z^i \frac{W_z^i}{dt} - (I_x^i - I_y^i) W_x^i W_y^i &= N_z^i \end{aligned} \right\} (2)$$

以上の2式から、外力 F^i, N^i をもとに、剛体の運動が求まる。例えば、(2)式により、

$$W_x^i(t + \Delta t) \doteq W_x^i(t) + \Delta t \cdot \{N_x^i(t) + (I_y^i - I_z^i) W_y^i(t) W_z^i(t)\} / I_x^i$$

となることから、この式を利用して、 $W^i(t)$ から $W^i(t + \Delta t)$ を求めることができる。従って、数値積分を行なって近似解を求めることが

m	: 剛体の質量
I_x, I_y, I_z	: 剛体の慣性モーメント
V_x, V_y, V_z	: 剛体の重心の速度
W_x, W_y, W_z	: 剛体の角速度
F_x, F_y, F_z	: 剛体に加えられる外力
N_x, N_y, N_z	: 剛体に加えられるトルク

図3: 剛体のパラメータ


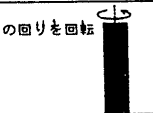

拘束	方程式
固定される 	$V^i + W^i \times R^i = 0$
軸の回りを回転 	$W^i = k^i \cdot \Omega^i$ (k^i は定数)
つながっている 	$V^i + W^i \times R_j^i =$ $V^j + W^j \times R_i^j$

図 4: 力学的拘束の例

できる。また、複数の剛体が連結したような物体の場合、上記の方程式を剛体の個数だけ連立させて、連立の線形微分方程式を立てることができる。

物体の拘束としては、まず、 $f(V^i, W^i, \frac{V^i}{dt}, \frac{W^i}{dt}) = 0$ という形の等式で表現できて、さらに f が $V^i, W^i, \frac{V^i}{dt}, \frac{W^i}{dt}$ に対して線形のもの考えることにした。この場合は、後述するように、拘束式の計算が行列演算に帰着できるので、取り扱いやすい。その代表的な例を図 4 に示す。

3. MOVE アニメーションシステム

我々は、現在 MOVE という新しいシステムを開発中である。MOVE の目的は、3次元コンピュータアニメーションの作成支援である。

MOVE の開発には、オブジェクト指向言語 C++ を用いた。MOVE の全データはオブジェクト指向 DBMS で管理されていて、データの再利用を可能にしている。図 5 に示すように、MOVE には、オブジェクト指向の 3次元クラスライブラリがある。そのクラス階層を図 6 に示す。

我々は、アニメーションモデルを実装において、この 3次元グラフィックスクラスライブラリの、3次元物体の形状モデリングと画像生成の機能を利用することにした。

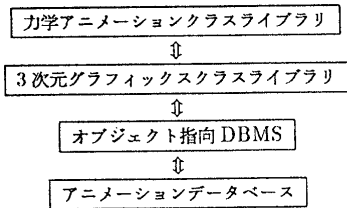


図 5: MOVE のシステム構成

4. 実装

2節で示した剛体の力学に対して、オブジェクト指向でクラス設計を行なった。Body クラス (質点に対応)、RigidBody クラス (剛体に対応)、Constraint クラス (拘束に対応)、Controller クラス (図 2 のコントローラに対応) を設計した (図 6)。

アニメーション作成のときは、適当な RigidBody オブジェクトと Constraint オブジェクトを生成し、RigidBody オブジェクトには、適当な初期値を与える。その後、これらのオブジェクトと、Controller オ

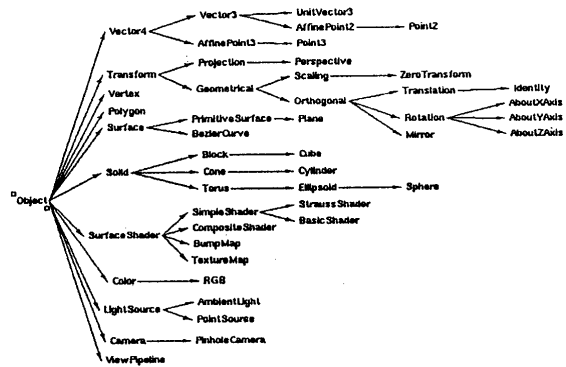


図 6: MOVE の 3次元グラフィックスクラスライブラリ

クラス名	メンバ変数 / メンバ関数
Body	質量 質点の位置、速度、加速度 外力から質点の運動を求める
RigidBody	慣性モーメント 回転角、角速度、角加速度 トルクから角速度などを求める
Constraint	線形方程式の各係数
Controller	拘束力を求める

図 7: 力学アニメーションクラス

プロジェクト間で、適当な情報のやり取りを行なうことで、剛体の運動が求まる。

5. おわりに

我々は、オブジェクト指向にもとづいて、アニメーションモデルの実装を試みている。今後、拘束要素の種類を増やすことと、エネルギー保存の法則を拘束として処理できるようにすることが第 1 の課題である。

3次元アニメーションを、データベースの応用分野の 1つとしてみると、シミュレーション部分の計算負荷も高い上に、1つの処理に必要なオブジェクト数が多いという特徴がある。従って、アニメーションデータベースでは、処理効率の良さが特に重要となる。発表では、具体的な工業製品の 3次元コンピュータアニメーションの例を示して、処理速度の観点から評価を行なう予定である。

参考文献

- James K. Hahn: "Realistic Animation of Rigid Bodies", ACM SIGGRAPH '88, pp. 299-308, 1988.
- Kunihiko Kaeko, Susumu Kuroki and Akifumi Maki-nouchi: "Design of 3D CG Data Model of MOVE Animation Database System", proceedings of the 2nd Far-East Workshop on Future Database Systems, 1992.
- 黒木進, 金子邦彦, 牧之内顕文: "アニメーション用データベースにおける時間的知識の表現", 情報処理学会第 43 回全国大会
- 金子邦彦, 黒木進, 近藤祐介, 牧之内顕文: "3次元 CG モデルの OODBMS による実装", 情報処理学会第 44 回全国大会