グラフィクスとCAD 57－3
（1992．7．24）

動力学を応用した
多関節構造体のキーフレームアニメーション

新井　清志

（株）日立製作所　中央研究所

本報告では、人体アニメーションの生成等に用いられる多関節構造体のリアリスティックな動きを得るため、動力学演算を応用した新しい手法を提案する。動力学演算を用いた従来手法では、力やトルクを入力することが多く、これらの物理量はユーザにとって直感的でないという問題があった。本手法では、多関節構造体のキーフレームアニメーションの中で部分的に動力学演算を行ない、直感的な入力だけによってリアリスティックな動作を生成できるようにする。本手法を用いて人体モデルを制御する実験を行なった結果、惰性による動きやジャンプの動きの生成に本手法が有効であることが確認できた。

KEYFRAME ANIMATION OF ARTICULATED FIGURES USING DYNAMICS

Kiyoshi Arai

Central Research Laboratory, Hitachi, Ltd.

1-280 Higashi-Koigakubo, Kokubunji, Tokyo 185, Japan

In this paper, we propose a new method using dynamics to obtain realistic movements of articulated figures used mainly for the generation of human animation. Previous proposed methods using dynamics require value inputs of forces and torques which are difficult to grasp for users. In this method, we use dynamics for the generation of some parts of movements in the keyframe animation to enable usage of simpler user-understandable inputs. Experimental results using a human model show that inertial and jumping motion are generated effectively by this method.

1　はじめに

多数の関節を持ち、その関節角の変化によって動く多関節構造体は、人間の動き等を生成する際に用いられるモデルである。多関節構造体の動作生成のため、動力学を応用した手法の研究が盛んに行なわれており[1]、以下のような手法が提案されている。

（1）力やトルクを与える手法

動力学の最も基本的な応用手法は、物体に働く力やトルクを与えて加速度を求め、動きを生成する手法である。Wilhelms[2]は、人体モデルの大まかな動きをキーフレーム法によって求め、動力学演算の過程で関節に働く力やトルクを対話的に修正することによって、最終的な動きを得ている。力やトルクという直感的でない値を修正する作業は煩雑である。生成する動きの種類を限定すれば、ユーザが力やトルクを与えずに動力学演算を行なうことができる。例えば、Bruderlinら[3]の手法では、人体モデルの歩行動作を、歩行速度や歩幅等の入力により生成できる。Haumannら[4]の手法では、ハミングバードの飛行動作を、羽ばたき方や羽の角度等の入力により自動制御できる。

（2）運動学的な条件を与える手法

関節の角度や目標位置といった運動学的な条件を与えるようにすると、力やトルクを与える手法に比べて、より直感的な制御が可能となる。Isaacsら[5]は、全関節の関節角加速度と関節に働く力との関係を表す行列を求めている。これにより、各関節毎に角加速度と力のうちで与えやすい方を与えるようにしている。関節角加速度は、直接的には関節角を指定することによって与えられる。Barzelら[6]は、物体を目標とする位置や姿勢に近づける力を求めて、動きを生成している。これらの手法においては、目標への到達時刻を調節することが難しい。Leeら[7]は、トルクの許容値を考慮し、逆運動学と逆動力学を平行して用いる手法を提案している。この手法では、手先等に与える質量によって目標点への到達経路をコントロールする。したがって、物を持ち上げる腕の動きや、上半身を持ち上げる足の動き、すなわち立ち上がり動作等の生成には適している。

（3）評価関数を最適化する手法

物体に働くエネルギーや移動距離に関する評価関数を定義し、この関数を最適化する手法も提案されている。Witkinら[8]の報告の中では、この手法を多関節構造体に適用することにより、バネのような関節の動きが得られることが示されている。Kass[9]は、一般的な最適化問題を解くインタフェースを構築している。最適化を行なう手法の問題点は、複雑な動きに対応する評価関数が定義しにくいことと、最適化の計算量が多いことである。

（4）グローバルな動力学を用いる手法

個々の関節の動きに動力学を考慮するだけでなく、体のバランスを考慮したり、ジャンプや着地の動作を実現することも重要である。Girard[10]は、体全体の動きに対してのみ簡単な動力学を適用している。具体的には、あるカーブを描いて進む際の体の傾きや、ジャンプする場合の初速度等を、動力学を用いて求めている。ジャンプの初速度と着地までの時間との関係が力学の法則に従っていると、体全体の動きはリアリスティックに見える。しかし、着地の瞬間における足と地面との接触の様子を正確に求めるためには、体と周囲の物体との干渉の有無まで考慮する必要がある。

動力学演算により得られる動きはリアリスティックであるが、力やトルクという直感的でない物理量を入力することが多い。また、動力学演算の中に、ある時刻にある姿勢になる、というキーフレームアニメーションのような条件を細かく与えることは難しい。本報告では、このような問題を解決するため、多関節構造体のキーフレームアニメーションの中で部分的に動力学演算を行ない、直感的な入力だけでリアリスティックな動作を生成する手法を提案する。体全体の位置を決める拘束条件を与える際には、多関節構造体と周囲の物体との干渉チェックを行ない、ジャンプの動き等もリアリスティックに生成できるようにする。

2　動力学演算を用いたキーフレムアニメーション

2．1　多関節構造体の定義

本手法で扱う多関節構造体を図１に示す。関節J_cと、一つ先端側にある関節J_eとを結ぶリンクをL_cとする。また、先端の関節J_{end}の先の部分をリンクL_{end}とする。関節J_cは、J_cを原点とする座標系C_cを持つ。そして、一つベース側にある関節J_bの持つ座標系C_bに対するC_cの相対位置は、４×４の座標変換行列M_cで表される。ベースの関節をJ_{base}とすると、M_{base}はワールド座標系C_{world}に対するC_{base}の相対位置を表す座標変換行列である。これらの座標変換行列により、C_{world}で表したJ_cの位置が求められる。

本手法で扱う多関節構造体の各関節の自由度は１とする。２自由度の関節は、２つの関節と、それらの間を結ぶ長さ０、質量０のリンクに置き換えて考える。同様にして、３自由度の関節は、３つの関節と、それらの間を結ぶ長さ０、質量０の２つのリンクに置き換えて考える。また、ベースから先端に向かって枝分かれを許す。すなわち、あるリンクL_cに対して、一つ先端側のリンクL_eが複数存在してもよい。

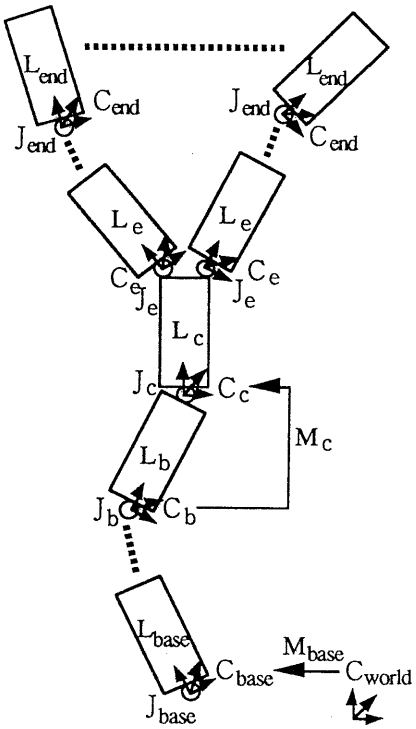


図１　多関節構造体

2．2　逆動力学演算

多関節構造体のすべての関節の、ある時刻における関節角、関節角速度、関節角加速度が与えられると、逆動力学演算により、すべての関節の関節トルクが求められる。本手法ではニュートン・オイラー法を用いたLuh, Walker, Paulによる逆動力学演算[11]を採用する。ｉ番目の成分をｉ番目の関節の関節角θ_iおよび関節トルクτ_iとするＮ×１ベクトル（Ｎは関節の総数）をそれぞれΘ、Ｔとする。また、ベースの加速度から重力加速度を引いた値を表す３×１ベクトルをａとし、L_{end}に外部から働く力およびモーメントの列ベクトルからなる３×Ｅ行列（ＥはJ_{end}の総数）をそれぞれKf、Knとする。’は時間での１次微分、”は時間での２次微分を表す。リンクの慣性テンソルは、リンクを円柱で近似して求める。逆動力学演算によりＴを求める計算を、関数Ｄinvを用いて式(1)のように表す。

$$T = D_{inv}(\Theta,\ \Theta',\ \Theta'',\ a,\ Kf,\ Kn) \qquad (1)$$

2．3　動力学演算

多関節構造体のすべての関節の、ある時刻における関節角、関節角速度、関節トルクが与えられると、動力学演算により、すべての関節の関節角加速度が求められる。動力学演算は、逆動力学演算を利用して行なわれる。

式(1)のΘ”とＴとの間には式(2)の関係がある。

$$H\Theta'' = T - B \qquad (2)$$

Ｈを慣性行列、Ｂをバイアスベクトルという。ＢはＮ×１ベクトルであり、式(3)によって得られる。

$$B = D_{inv}(\Theta,\ \Theta',\ 0,\ a,\ Kf,\ Kn) \qquad (3)$$

ＨはＮ×Ｎ行列であり、そのｉ番目の列ベクトルH_iは式(4)によって得られる（$1 \leqq i \leqq N$）。

$$H_i = D_{inv}(\Theta,\ 0,\ e_i,\ 0,\ 0,\ 0) \qquad (4)$$

式(4)において、e_iは、i番目の成分が1でその他は0であるようなN×1単位ベクトルである。Bのi番目の成分をb_iとすると、b_iは、関節角加速度の影響を無視した場合のJ_iの関節トルクを表す。また、H_iは、関節角速度、重力、ベースの加速度、外力の影響を無視し、J_iの関節角加速度を1、他の関節角加速度を0にした場合のJ_iの関節トルクを表す。

2．4　関節角の範囲と摩擦の効果を考慮した動力学演算

バイアスベクトルBおよび慣性行列Hを求めれば、TからΘ”を求めることができる。しかし、Tは直感的な値ではない。そこで、本手法では、Tをゼロベクトルとして式(2)を解き、得られたΘ”を用いて動きを生成する。このとき得られる動きは、関節に対して能動的な関節トルクを加えないときの動き、すなわち惰性による多関節構造体の動きであり、リアリスティックである。本手法では、関節角の下限値と上限値、および関節に働く摩擦の効果を考慮し、以下に示す手順で動力学演算を行なう。ここで、十分短い一定の時間間隔をΔt、現在着目している時刻をt_c、次の瞬間の時刻を$t_d = t_c + \Delta t$とする。一つの関節角に着目した場合の関節角決定の過程を図2に示す。図2において、θ_c、θ_t'、θ_m'、θ_t、θ_dはそれぞれ以下で定義するΘ_c、Θ_t'、Θ_m'、Θ_t、Θ_dの成分を表す。

（1）時刻t_cにおける関節角Θ_cおよび関節角速度Θ_c'が与えられているものとする。

（2）時刻t_cにおける関節トルクTをゼロベクトルとする。

（3）Θ_c、Θ_c'から、式(3)を用いて時刻t_cにおけるバイアスベクトルBを求める。

（4）Θ_cから、式(4)を用いて時刻t_cにおける慣性行列Hを求める。

（5）T、H、Bを式(2)に代入し、時刻t_cにおける関節角加速度Θ_c''を求める。

（6）Θ_c'、Θ_c''から、式(5)を用いて時刻t_dにおける暫定関節角速度Θ_t'を求める。

$$\Theta_t' = \Theta_c' + \Theta_c'' \cdot \Delta t \qquad (5)$$

（7）Θ_t'の各成分に、摩擦の効果を表す0以上1以下の実数値を乗じて、時刻t_dにおける修正関節角速度Θ_m'を求める。0を乗じる成分に対応する関節は、全く動かない。1を乗じる成分に対応する関節は、摩擦が全くない状態で動く。ただし、手順（11）から手順（6）を経由して手順（7）に戻ってきた場合、Θ_t'の成分の中でΘ_tの修正された成分に対応するものには実数値を乗じない。

（8）Θ_c、Θ_m'から、式(6)を用いて時刻t_dにおける暫定関節角Θ_tを求める。

$$\Theta_t = \Theta_c + \Theta_m' \cdot \Delta t \qquad (6)$$

（9）Θ_tの全ての成分が、各々の関節角の下限値θ_{min}から上限値θ_{max}までの範囲内にある場合は、Θ_tを時刻t_dにおける関節角Θ_dとして次の時刻における処理に進む。

（10）Θ_tの成分の中で、対応する関節角のθ_{min}からθ_{max}までの範囲外のものがある場合は、対応する値をθ_{min}とθ_{max}のうち近い方に一致させる。修正されたΘ_tから、式(7)を用いてΘ_c''を求め直す。

$$\Theta_c'' = ((\Theta_t - \Theta_c)/\Delta t - \Theta_c')/\Delta t \qquad (7)$$

（11）求め直したΘ_c''の成分の中でΘ_tの修正された成分に対応するものと、手順（2）で与えたTの成分の中でΘ_tの修正された成分に対応しないものとを、式(2)に代入し、あらためて時刻t_cにおける関節角加速度Θ_c''の全ての成分を得て、手順（6）に戻る。

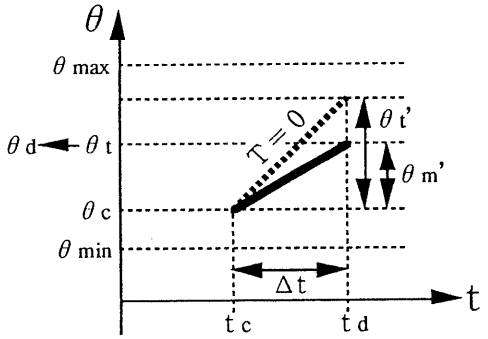


図2　動力学演算を用いた関節角決定の過程

2．5　キーフレームアニメーションへの動力学演算の導入

本手法においては、キーフレームアニメーションの中で部分的に動力学演算を行なう。ある時刻における各々の関節角のデータからなるキーフレームを与える際に、隣接するキーフレームによって定められる区間、および多関節構造体に含まれる部分構造体の中から、特定の区間における特定の部分構造体を選択する。選択された区間における選択された部分構造体に含まれる関節の関節角のデータは、2．4で述べた動力学演算によって求める。それ以外の関節の関節角のデータは、キーフレームにおける関節角を滑らかに内挿することにより求める。

本手法におけるキーフレームアニメーションの具体例を図3に示す。時刻t_1、t_2、t_3、t_4におけるキーフレームが与えられており、区間［t_2, t_3］における部分構造体Pが、動力学演算を行なう部分として選択されている。時刻t_iにおけるJの関節角をθ_i、関節角速度をθ_i'とする（$1 \leq i \leq 4$）。Pに含まれる関節Jの関節角のデータは、以下に示す手順で生成される。

（1）区間［t_1, t_2］のデータは、θ_1、θ_1'、θ_2を用いて2次関数で補間して求める。または、θ_1、θ_1'、θ_2の他にθ_2'を与えて、3次関数で補間して求める。動力学演算を行なう区間の最初の時刻t_2における関節角θ_2および関節角速度θ_2'は、キーフレームの変更によって容易に調整でき、その結果が動力学演算の結果に反映される。

（2）区間［t_2, t_3］のデータは、θ_2、θ_2'を初期値とし、2．4で述べた動力学演算によって求める。動力学演算の結果として、θ_3、θ_3'が決定される。

（3）区間［t_3, t_4］のデータは、θ_3、θ_3'、θ_4、θ_4'を用いて3次関数で補間して求める。

このようにして、多関節構造体のキーフレームアニメーションの中で部分的に動力学演算を行なう。キーフレームアニメーションのデータの中で、リアリスティックな動きが必要な部分に本手法を容易に挿入することができる。

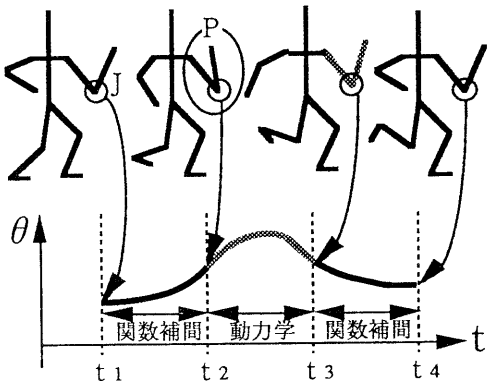


図3　本手法におけるキーフレムアニメーション

3　拘束条件を用いた各フレームでの位置の決定

3．1　関節の位置を固定する拘束

キーフレーム間の関節角を求めるとき、ワールド座標系内での多関節構造体の位置のきめ方が問題になる。多関節構造体と周囲の物体とが接触している場合には、相互に力を及ぼし合うが、本手法ではこの力を扱わず、簡単な拘束条件を与えて、位置を決定する。一つの拘束条件は、関節の位置の固定である。具体的には、あるキーフレームから次のキーフレームへ到達するまでの間、多関節構造体のある関節のワールド座標系内での位置を固定する。例えば、人体モデルの歩行動作を生成する際には、地面に接触している方の足先の関節の位置を固定する拘束を与える。

3．2　重力と干渉を考慮した拘束

もう一つの拘束条件は、多関節構造体に働く重力を考慮したものである。多関節構造体と周囲の物体との関係は、互いに接触しているか、離れているかのいずれかである。前者の場合、ある関節の位置がワールド座標系内で固定されていることが多い。後者の場合、多関節構造体に外から働く力は重力だけである。そこで、重力と干渉を考慮した以下に示すような拘束を与える。ここで、十分短い一定の時間間隔をΔt、現在着目している時刻をt_c、次の瞬間の時刻を$t_d = t_c + \Delta t$とし、時刻t_c、t_dに

おける多関節構造体Fの姿勢をF_c、F_dとする。

（1）F_cが与えられているものとする。

（2）Fに重力だけが働いた場合、時刻t_dにどのような姿勢になるかを、重心の位置に着目して求め、これを時刻t_dにおけるFの暫定姿勢F_tとする。

（3）図4（a）のように、F_cが周囲の物体と干渉（接触）しており、F_tも干渉する場合は、指定された関節の位置が固定されるようにF_tの位置を修正し、これをF_dとする。

（4）F_cが周囲の物体と干渉（接触）しており、F_tは干渉しない場合は、F_tをそのままF_dとする。

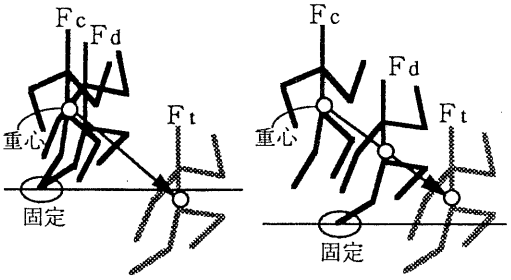
（5）F_cが周囲の物体と干渉しておらず、F_tも干渉しない場合は、F_tをそのままF_dとする。

（6）図4（b）のように、F_cが周囲の物体と干渉しておらず、F_tは干渉する場合は、F_cの重心とF_tの重心を結ぶ線分上でF_tの重心の位置をずらし、F_tが周囲の物体とちょうど接触する位置を求め、このときのF_tをF_dとする。ここで接触した部分に最も近い関節が、以後（3）の状態になったときに固定される関節となる。

このような拘束により、多関節構造体のジャンプの動き等が生成できる。

4　実験結果

本手法による人体モデルの動作生成実験の結果を示す。使用した計算機はＨＰ９００００／８３５ＴＳＲＸである。人体モデルの各々の関節は初期状態では３つの回転軸を持ち、それぞれの回転軸まわりの角度に制限を与えることにより、関節の自由度や関節角の範囲を指定した。キーフレーム間の動きは、動力学演算を行なう場合も、関数で補間する場合も、１／３０秒間隔のデータを逐次生成し、１秒分の動きのデータを３０フレームで表現した。以下の説明において、n番目のフレームをF_n、n番目のキーフレームをK_nとする。



（a）干渉有→干渉有　　（b）干渉無→干渉有

図4　重力と干渉を考慮した拘束による位置の決定

図５は、右腕で物を投げる動きを生成した結果である。F_0からF_{30}までの３１フレームからなり、生成に要した時間は約３秒であった。F_0、F_5、F_{15}、F_{30}をそれぞれK_1、K_2、K_3、K_4として与えた。K_1からK_2までの間は、腰から上の部分の動きを全て関数補間で求め、足の部分は逆運動学によって両足先を固定してデータを生成した。K_2からK_3までの間は、左右各々の肩から手先までの部分において動力学演算を行ない、３．１で述べた手法を用いて左足先の関節を固定して全身の位置を決定した。動力学演算において、２．４の手順（7）で述べた摩擦の効果を表す実数値は、右手首では0.4、他の関節では0.8とした。K_3からK_4までの間は、全身の動きを関数補間で求め、左足先の関節を固定した。右腕が前方に投げ出される動き、および左腕が後方に振れる動きが生成できた。

図６は、台の上からジャンプする動きを生成した結果である。F_0からF_{30}までの３１フレームからなり、生成に要した時間は約３秒であった。F_0、F_{10}、F_{30}をそれぞれK_1、K_2、K_3として与えた。３．２で述べた手法を用いて、K_1からK_2までの間は、周囲との干渉があるかぎり右足首の関節を固定し、K_2からK_3までの間は、周囲との干渉が生じたら右足首の関節を固定する拘束条件を与えた。干渉チェックは、円柱の両端に半球を付けた形状で各々のリンクを近似して計算する手法[12]を用いた。人体モデルはF_6において台から離れ、F_{10}において最も高い位置に達し、F_{18}において床の上に着地した。ジャンプの高さ、着地の位置や時刻等が自動的に求められた。

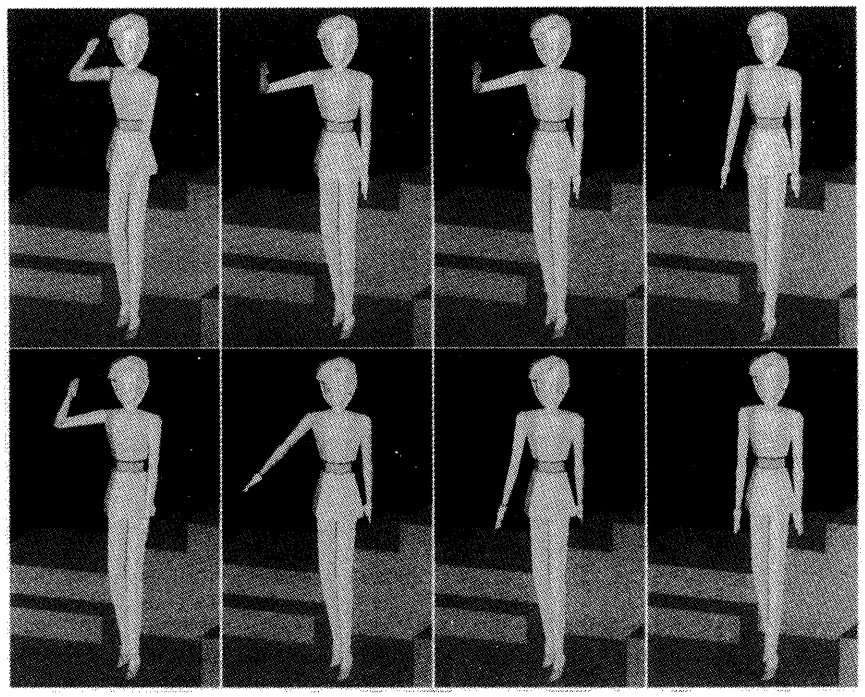


図5　物を投げる動き

K_1 (F_0)	K_2 (F_5)	K_3 (F_{15})	K_4 (F_{30})
F_3	F_{10}	F_{15}	F_{25}

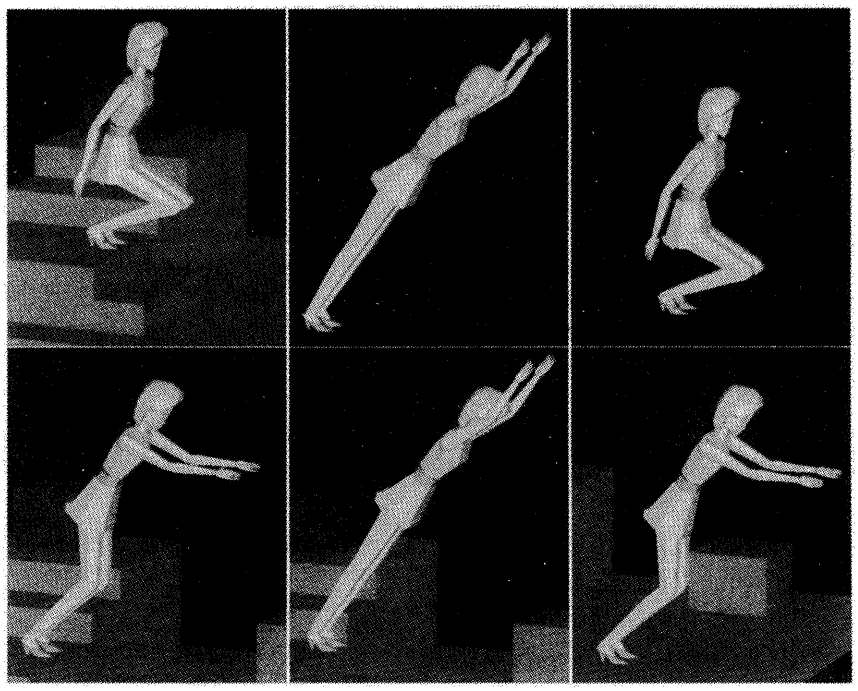


図6　ジャンプする動き

K_1 (F_0)	K_2 (F_{10})	K_3 (F_{30})
F_6	F_{10}	F_{18}

5 考察

本報告で提案した手法において、惰性による動きの生成に要する計算量は、動力学演算の対象となる部分構造体に含まれる関節の数の３乗に比例する。したがって、人体モデル全体を一つの部分構造体とみなして動力学演算を行なうと、計算量は非常に多くなる。しかし、関節の可動範囲が大きく、惰性による動きの視覚的効果が大きい両腕や両足の部分に限定して動力学演算を行なえば、実験で示した程度の少ない計算量で済むので、実用上問題はない。

人体モデルの動作生成実験においては、手首から先の関節の制御を行わなかった。手には多くの関節があり、しかも各々の関節の動きは完全に独立ではなく、物を掴む等、全体で一つの意味を持つ動きをすることが多い。したがって、本手法をそのまま適用して手の動きを生成することは困難であり、リアリスティックな手の動作[13]を生成するためには、新たな機能を追加しなければならない。

キーフレームによる動作の制御は、力やトルクを入力するよりも直感的であり、効率的に動作生成ができる。しかし、アニメーションを制作する際には、ユーザが詳細な動きを指示したい場合と、大まかな指示だけでユーザの意図を反映した動きを得たい場合とがある。本手法は前者の場合には適しているが、後者の場合のユーザの要求を満たす機能は有していない。このような機能を実現するためには、周囲の環境に適応して動作を自動生成する手法[4][14]について検討する必要がある。

6 おわりに

本報告では、多関節構造体の動作生成において、キーフレームアニメーションの中で部分的に動力学演算を行なう手法を提案した。力やトルクという直感的でない値を入力せずに、惰性による動きやジャンプの動きをリアリスティックに生成でた。今後は、手の動作生成機能、および環境に適応した自動的な動作生成機能について検討する予定である。

謝辞

本研究を進めるにあたり、御指導と御助言を賜った（株）日立製作所中央研究所の栗原恒弥研究員、並びに人体モデルの形状データを提供していただいた同研究所の二木誠司研究員に感謝の意を表します。

参考文献

[1]N. Magnenat-Thalmann：
New Trends in the Direction of Synthetic Actors；
CG International '90, pp.17-35（1990）.

[2]J. Wilhelms：Using Dynamic Analysis
for Realistic Animation of Articulated Bodies；
IEEE CG&A, 7, 6, pp. 12-27（1987）.

[3]A. Bruderlin and T. W. Calvert：Goal-Directed,
Dynamic Animation of Human Walking；Computer
Graphics,23,3 (Siggraph '89), pp. 233-242（1989）.

[4]D. R. Haumann and J. K. Hodgins：The Control
of Hovering Flight for Computer Animation；
Computer Animation '92, pp. 3-19（1992）.

[5]P. M. Isaacs and M. F. Cohen：Controlling
Dynamic Simulation with Kinematic Constraints,
Behavior Functions and Inverse Dynamics；Computer
Graphics,21,4 (Siggraph '87), pp. 215-224（1987）.

[6]R. Barzel and A. H. Barr：A Modeling System
Based On Dynamic Constraints；Computer
Graphics,22,4 (Siggraph '88), pp. 179-188（1988）.

[7]P. Lee, S. Wei, J. Zhao and N. I. Badler：
Strength Guided Motion；Computer Graphics,
24,4, (Siggraph '90), pp. 253-262（1990）.

[8]A. Witkin and M. Kass：Spacetime Constraints；
Computer Graphics,22,4 (Siggraph '88),
pp. 159-168（1988）.

[9]M. Kass：Inverse Problems in Computer Graphics；
Computer Animation '92, pp. 21-33（1992）.

[10]M. Girard：Interactive Design of
3D Computer-Animated Legged Animal Motion；
IEEE CG&A, 7, 6, pp. 39-51（1987）.

[11]M. Brady, J. M. Hollerbach, T. L. Johnson,
T. Lozano-Perez and M. T. Mason　編,
高野，古田　監訳：ロボットモーションII；
ＨＢＪ出版局，動力学編, pp. 7-94（1985）.

[12]新井，栗原：多関節構造体の姿勢制御のための干渉チェック手法；1991年電子情報通信学会春季全国大会，D-658（1991）.

[13]N. Magnenat-Thalmann, R. Laperriere, amd
D. Thalmann：Joint-Dependent Local Deformations
for Hand Animation and Object Grasping；
Graphics Interface '88, pp. 26-33（1988）.

[14]M. McKenna, S. Pieper and D. Zeltzer：
Control of a Virtual Actor: The Roach；Computer
Graphics,24,2 (1990 Symposium on Interactive 3D
Graphics), pp. 165-174（1990）.