

仮想エージェントに対する非剛体運動のモデル化と アニメーションの生成

青木 工太 長橋 宏

東京工業大学 像情報工学研究施設
〒226-8503 横浜市緑区長津田町 4259
E-mail: {aoki,longb}@isl.titech.ac.jp

あらまし 本稿では, Bézier 曲面で構成された 3 次元形状をもつエージェントの非剛体運動のモデル化および, 物理的な環境内に置かれたエージェントの振舞いのシミュレーションとアニメーション生成について述べる. 提案システムでは, エージェントは形状を変化させることで環境から力とトルクを受けて運動する. エージェントの形状は制御点を振り子のように振動させることによって様々に変化する. ユーザは運動の目標となる軌跡をシステムに入力し, システムはその軌跡に沿ってエージェントが運動するように, 最適な制御パラメータを学習する. パラメータの最適な値の系列を用いて 3 次元運動のシミュレーションを行い, アニメーションを生成する.

Non-rigid Motion Modeling and Animation Synthesis for Virtual Agents

Kota Aoki and Hiroshi Nagahashi

Imaging Science and Engineering Laboratory, Tokyo Institute of Technology
4259 Nagatsuta-cho, Yokohama-shi, Kanagawa, 226-8503 Japan
E-mail: {aoki,longb}@isl.titech.ac.jp

Abstract In this paper we propose non-rigid motion modeling of virtual agents composed of several Bézier surfaces and also describe a method for simulation and animation synthesis of their three-dimensional behavior in physical environments. In our system, forces and torques are provided due to mesh deformation of agents, and their resulting motions are simulated. Several control points of each Bézier patch can oscillate, like pendulums, spatially under some constraints, which allows an agent to be deformed. Our system optimizes all the parameters of the agent to move along a desired trajectory given by a user. We simulate and synthesize animation using a series of optimal values of the parameters.

1. はじめに

コンピュータグラフィックスおよび, アニメーションの研究では, 生物・無生物にかかわらず様々な対象のモデルが提案されてきた. 多くの文献は人間を対象にしている (例えば, [1]) が, ヘビ [2], 魚類 [3], 鳥類 [4] あるいは, 仮想生物 [5] などについてもモデル化されている.

本稿では, 「クラゲ」のような骨格を持たない対象を扱う. 一般に, そのような対象 (エージェント) を制御することは難しい. エージェントの位置および, 姿勢を直接与えればその振舞いを制御できるが, 見た目自然な動きを実現することは難しい. また, 力とトルクを与えて物理的にシミュレーションを行えば自然な動きに見えるが, 所望の振舞いを実現することは難しい. どちらの方法も, エージェントの

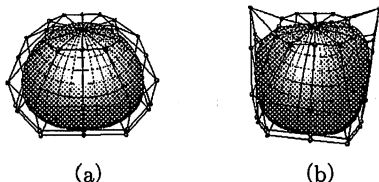


図1 エージェント。(a) 周囲の小球は制御点を表す。(b) 制御点の振動による変形。

形状の変化と振舞いを対応させることは容易でない。

本稿では、エージェントの非剛体の運動モデルを提案する。エージェント表面を構成する制御点の振り子運動として形状変化をモデル化し、変形によってエージェントが物理的な環境から受ける力とトルクを定義することで、形状の変化と振舞いが対応した自然な運動を実現する。また、制御パラメータの最適化について述べる。エージェントにユーザ入力による軌跡に沿って運動させることが可能となり、様々なアニメーションを生成できる。

2. 概要

提案システムにおけるエージェントは図1に示すような、いくつかの Bézier 曲面から構成される仮想的な（生物の形状を模した）物体である。図1(a)に示すエージェントは4×4次 Bézier 曲面を8面用いて構成されている。同図において、制御点はエージェントの周囲に配置された小球として表されている。制御点を動かすことでエージェントの形状が変形し、ある種の動作を生み出す。提案システムでは、それぞれの制御点を「振り子」として、その振動を三角関数によりモデル化する。

ユーザはエージェントが運動の目標とする軌道をシステムに入力する。目標軌道は3次 Bézier 曲線 $p(u)$ として表現する。物理的な環境を仮定すると、システムはエージェントが環境から受ける力をシミュレートする。ここでは、水中環境内に置かれたエージェントの運動をモデル化する。システムの目的はエージェントが軌道に沿って移動するように、その最適な動作すなわち、(三角関数でモデル化された) 制御点の振動に対する最適なパラメータを探索することである。目標軌道を指定することによって、ユーザはエージェントに様々な運動を指示できる。

3. 非剛体運動

3.1 制御点の振動

Bézier 曲面で表現されたエージェントは制御点を動かすことで形状を変化させ、それに伴って環境から力を受けて移動する。提案システムでは、エージェントの質量中心を支点とする回転軸を制御点毎に設定して、振り子のように制御点を振動させる。制御点が動き始めてから再び初期位置に戻るまでのステップ数を全ての制御点で同じにして、この一回の振動を「サイクル」と呼ぶことにする。以下では、1サイクルのステップ数を N として、時刻 t を次のように定義する。

$$t = \frac{n}{N}, \quad 0 \leq n \leq N, \quad 0 \leq t \leq 1 \quad (1)$$

ある時刻 t における各制御点の初期位置からの回転角 $\phi(t)$ を次式のように（右手座標系において）、三角関数によってモデル化する。

$$\phi(t) = \begin{cases} a \sin \frac{\pi t}{2b} & t \leq b \\ a \cos \frac{\pi(t-b)}{2(1-b)} & t > b \end{cases} \quad (2)$$

ただし、 $0 \leq b \leq 1$ とする。

どの制御点を動かすかということは曲面の位置と接線における連続性を満たすように、また動かすことによって自然な形状になるかどうかを考慮して、決定する必要がある。図1(a)に示したエージェントに対して、全ての制御点のうち8個の制御点を動かすことにした。また、曲面の連続性を満たすように、近傍の制御点を連動させる。制御点を動かして変形したエージェントの例を図1(b)に示す。また、パラメータ a および、 b の値を変えた式(2)による制御点の振動すなわち、回転角の時間変化を図2に示す。

3.2 エージェントの状態

時刻 t におけるエージェントの状態 $Y(t)$ を次のように定義する。

$$Y(t) = [\bar{x}(t) \ q(t) \ P(t) \ L(t)]^T \quad (3)$$

ここで、 \bar{x} 、 q 、 P および、 L はそれぞれ、エージェントの質量中心、回転を表す四元数、運動量および、

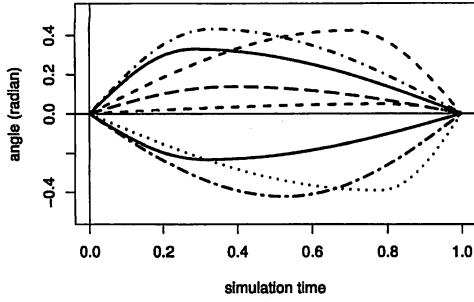


図 2 制御点の振動 (回転角の変動)

角運動量を表す。また、状態の時間微分 $\dot{Y}(t)$ は次のようになる。

$$\dot{Y}(t) = \frac{d}{dt} \begin{bmatrix} \bar{x}(t) \\ q(t) \\ P(t) \\ L(t) \end{bmatrix} = \begin{bmatrix} \bar{v}(t) \\ \frac{1}{2}\omega(t)q(t) \\ F(t) \\ \tau(t) \end{bmatrix} \quad (4)$$

ただし、 \bar{v} および、 ω は質量中心 (重心) の速度および、角速度を表し、ここでは四元数 $[0 \ \omega(t)]'$ と $q(t)$ の積を $\omega(t)q(t)$ と略記した。また、 F および、 τ はエージェントが環境から受けるすべての力および、トルクを表す。

エージェントの運動量は環境から力を受けて変動する。運動量から速度が求まり、環境内でエージェントは移動する。エージェントの姿勢 (向き) も同様に、トルクによって角運動量が変動し、角速度から求まる回転角から決まる。それぞれの関係式は以下ようになる。

$$\bar{v}(t) = \frac{1}{M}P(t), \quad \omega(t) = I^{-1}(t)L(t) \quad (5)$$

ここで、 $q(t)$ から求まる回転行列を $R(t)$ および、エージェントの初期形状に対する慣性テンソル [6] を I_0 とすると、 $I^{-1}(t) = R(t)I_0^{-1}R(t)'$ である。エージェントの形状は時刻とともに変化するため、実際には I_0 も時間変化するが、計算コストを削減するため、ここでは定数とする。

3.3 ダイナミクス

提案システムでは、環境から受ける力として抗力および、揚力を以下のようにモデル化する。抗力および、揚力はエージェントを構成する曲面のパッチ

ごとに働くものとする。エージェントの形状変化は微視的に見れば、Bézier 曲面パッチの移動である。パッチ i から見た環境の相対速度 v_i は環境 (水流) の速度 v_0 、パッチ i の単位時間あたりの移動量 Δx_i 、エージェントの速度 \bar{v} および、角速度 ω から、次式のように計算される。

$$v_i = v_0 - \dot{x}_i - \bar{v} - \omega \times x_i \quad (6)$$

ただし、ここでは質量中心を原点とするパッチ i の位置ベクトルを x_i で表す。速度 v_i から迎角 (v_i に対するパッチ i の傾き) θ_i を次式により求める。

$$\theta_i = \tan^{-1} \left(\frac{v_n \cdot n_i}{\|v_t\|} \right) \quad (7)$$

ただし、パッチ i の法線ベクトルを n_i とし、 v_n および、 v_t はそれぞれ、パッチに対する v_i の垂直成分および、水平成分を表す。仮定環境として水中を仮定する場合、 v は水流の相対速度である。定義から、パッチにかかる抗力および、揚力の向きはそれぞれ次のようになる。

$$d = \frac{v}{\|v\|}, \quad l = \frac{d \times n}{\|d \times n\|} \times d \quad (8)$$

抗力係数 $c_d(\theta)$ および、揚力係数 $c_l(\theta)$ は迎角 θ の値によって決まる。文献 [4] を参考に、図 4 のようにいくつかの定数を指定して、それぞれの係数を求める。以上から、 i 番目のパッチが受ける抗力 D_i および、揚力 L_i は次式のようになる。

$$D_i = \frac{1}{2}c_d(\theta_i)\rho s_i\|v_i\|^2 d_i \quad (9)$$

$$L_i = \frac{1}{2}c_l(\theta_i)\rho s_i\|v_i\|^2 l_i \quad (10)$$

ここで、 s_i はパッチ i の面積を表し、 ρ は環境の密度 (海水の場合、 $1020[\text{kg}/\text{m}^3]$) である。エージェントにかかる力 F および、トルク τ はパッチそれぞれにかかる力を合計して求まる。

$$F = \sum_i (D_i + L_i), \quad (11)$$

$$\tau = \sum_i x_i \times (D_i + L_i) \quad (12)$$

4. 制御点パラメータの最適化

エージェントは制御点を振り子のように動かして運動する。式 (2) に従って振動させる制御点の数を

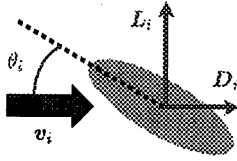


図 3 パッチにかかる抗力および、揚力

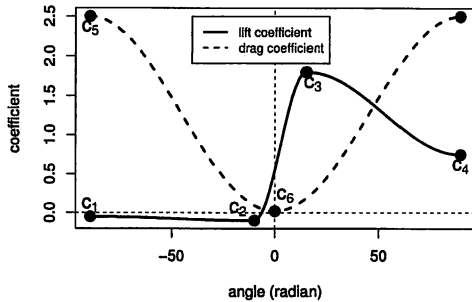


図 4 抗力係数および、揚力係数

K 個とすると、1 サイクルのシミュレーションに対応するパラメータは $\{a_i, b_i | i = 1, \dots, K\}$ の $2K$ 個になる。エージェントがユーザによって入力された目標軌道 $p(u)$ に沿って運動するように、システムはこれらのパラメータを最適化しなければならない。変形によってエージェントにかかる力は非線形に変化するため、あるいはエージェントが異なる環境に置かれる場合を想定すると、最適なパラメータを予測することは困難である。ここでは、遺伝的アルゴリズム (GA)、とくにパラメータの値を (二進数に変換せず) 実数のまま扱う実数値 GA によってパラメータの最適化を行う。

4.1 実数値 GA による最適化

実数値 GA では、図 5 に示すように、解の候補である各個体はパラメータ (変数) の値を実数値配列として持つ染色体によって表現される。また、各個体はある評価基準に従って最適解に近い個体ほど高い得点 (低い罰金) を与えられる。高い得点 (低い罰金) を持つ個体ほど、次世代の母集団 (解の候補) を構成する個体として選択される (生存する) 確率が高くなる。交叉あるいは、突然変異といった遺伝的操作を行い、母集団の多様性を保ちながら世代交

	制御点 1		制御点 2		制御点 K	
	a_1	b_1	a_2	b_2	a_K	b_K
個体 1	0.28	0.33	0.70	0.43	0.80	0.05
個体 P	0.24	0.41	0.55	0.05	0.43	0.38

図 5 実数値 GA の母集団 (解の候補)

代を繰り返すことによって、局所解への収束を回避して、最適解を探索できる。

実数値配列で表現された個体に対する交叉法はいくつか提案されている [7] が、ここでは比較的単純な方法として BLX- α [8] を用いる。これは親となる二つの個体を囲む超直方体のある比率 α だけ拡張した領域から、一様分布に従って子個体を生成する方法であり、解探索の初期段階では比較的、大域的な探索が行われ、解空間において親個体同士が近くなると局所的な探索となる。

突然変異は突然変異率に従って遺伝子座ごとに、ある範囲の一様乱数により新しいパラメータの値に置き換える (一様突然変異)。

世代交代は Minimal Generation Gap (MGG) モデル [9] により行う。最適解探索の具体的な処理は以下ようになる。

- (1) ある範囲の一様乱数により母集団を生成する。
- (2) 母集団からランダムに二つの親個体を選ぶ。
- (3) 選ばれた親個体に交叉あるいは、突然変異操作を行い、子個体を生成する。
- (4) 選ばれた親個体および、生成された子個体をそれぞれ評価し、得点 (罰金) を与える。
- (5) 最も高い得点 (低い罰金) を得た個体および、ルーレット選択によって選ばれた二つの個体を親個体と入れ替える。
- (6) (2) から (5) まで (世代交代) を既定の回数だけ繰り返す。

これは 1 サイクルのシミュレーションに対応するパラメータの最適化である。最適なパラメータの値を使ってシミュレーションを行い、もしエージェントが目標軌道の終端点付近に到達していなければ、引

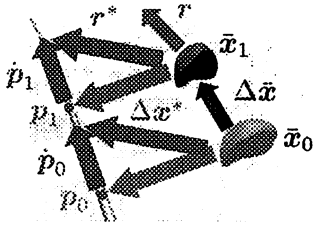


図6 エージェント運動の評価

き続き次のサイクルに対する最適なパラメータ値の探索を行う。

4.2 パラメータ値の評価

母集団の個体すなわち、パラメータの値を評価するためには、その値を使って実際にシミュレーションを行う必要がある。図6に示すように、シミュレーション前のエージェントの位置 \bar{x}_0 および、シミュレーション後の位置 \bar{x}_1 と主軸の方向 r から以下のように、四つの評価値を重みづけして加算した値を使って個体を評価する。

$$E = w_1 E_1 + w_2 E_2 + w_3 E_3 + w_4 E_4 \quad (13)$$

なお、ここでは、最適解に近い個体ほど低い罰金を与えられ、生存する確率が高くなるものとする。

エージェントの位置 \bar{x}_0 および、 \bar{x}_1 に最も近い目標軌道上の点の座標をそれぞれ $p_0, p_1 \in p(u)$ とする。シミュレーション後のエージェントの位置と目標軌道の距離の差 E_1 を次式により求める。

$$E_1 = \|\bar{x}_1 - p_1\|^2 \quad (14)$$

エージェントは点 p_0 における目標軌道の接線方向に移動することが望ましい。しかし、エージェントが目標軌道から離れた場合、再び軌道に戻る ($p_0 - \bar{x}_0$) ことと、接線方向に移動する (\dot{p}_0) ことは対立する要求となる。そこで、二つの移動方向を合成して、目標とする移動ベクトル Δx^* を次のように定める。

$$\Delta x^* = \frac{\dot{p}_0}{\|\dot{p}_0\|} + \frac{p_0 - \bar{x}_0}{\|p_0 - \bar{x}_0\|} \quad (15)$$

エージェントの移動ベクトル $\Delta \bar{x} = \bar{x}_1 - \bar{x}_0$ から、誤差 E_2 を次のように求める。

$$E_2 = \|\Delta \bar{x} - \Delta x^*\|^2 \quad (16)$$

表1 最適値探索 (GA) で用いたパラメータ

交叉確率	0.85
突然変異率	0.05
BLX- α [8] の α	0.10
(w_1, w_2, w_3, w_4)	(0.3, 0.3, 0.1, 0.3)

ただし、 $\Delta \bar{x}$ および、 Δx^* は単位ベクトル化されているものとする。

エージェントが目標軌道の終端点付近に到達するまで、シミュレーションは繰り返される。次のサイクルにおいて目標となる移動方向 r^* を Δx^* と同様にして、次式のように定める。

$$r^* = \frac{\dot{p}_1}{\|\dot{p}_1\|} + \frac{p_1 - \bar{x}_1}{\|p_1 - \bar{x}_1\|} \quad (17)$$

1 サイクル後のエージェントの主軸方向 r は r^* に近いほど良い。誤差 E_3 を次のように求める。

$$E_3 = \|r - r^*\|^2 \quad (18)$$

ただし、 r および、 r^* は単位ベクトル化されているものとする。

最後に、移動距離が大きいほど評価が高くなるように、次式のように評価値 E_4 を求める。

$$E_4 = e^{-\|\Delta \bar{x}\|} \quad (19)$$

5. 結果

提案システムによるエージェントの制御パラメータの最適化に関して、以下の条件で実験を行った。エージェントの初期位置および、姿勢はユーザ入力による目標軌跡の始点座標および、始点座標における接線方向に設定する。エージェントの質量は $M = 100[\text{kg}]$ とした。物理シミュレーションにおける単位時間は $\Delta t = 0.0333$ とし、1 サイクルに対するステップ数は $N = 60$ とした。制御パラメータの値はそれぞれ、 $-\pi/6 \leq a \leq \pi/6$ 、 $0.2 \leq b \leq 0.8$ の範囲に制限し、遺伝的アルゴリズムによる最適値探索では、世代交代を 500 回、母集団のサイズを 50、MGG モデルにおける世代交代の候補個体数を 40 とした。また、その他のパラメータに関しては表1の値を用いた。

二つの異なる目標軌跡をそれぞれ入力して得られたパラメータ値の系列を用いて、シミュレーション

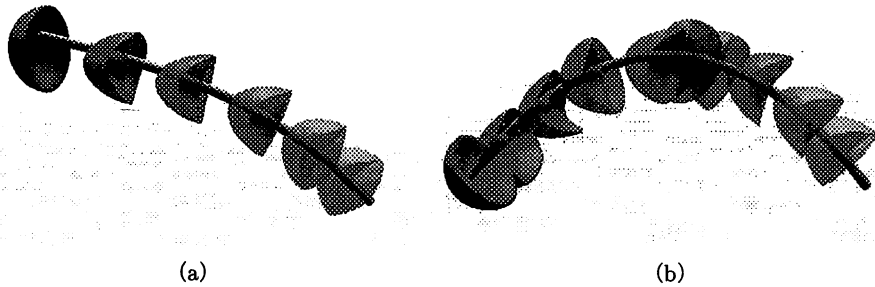


図 7 最適化されたパラメータ値系列による運動

を行った結果を図7に示す。エージェントが環境から受ける力とトルクに基づいてシミュレートすることにより、またその力とトルクがエージェントの形状変化に対応しているため、見た目に自然な運動を生成できる。目標軌道の曲がり方が急であっても（図7 (b)）、軌道に沿った自然な運動を達成した。

6. おわりに

本稿では、仮想的なエージェントの非剛体運動モデルについて述べ、物理的な環境内に置かれたエージェントの振舞いを最適化するシステムを提案した。

最適値探索および、アニメーション生成において、エージェントは同じ環境に置かれている。アニメーション生成時の環境が例えば、最適値探索のときよりも流れ（水流あるいは、気流）の速い環境になれば当然、エージェントは目標とした軌道に沿って運動できない。このような環境の動的な変化に対応するために、異なる環境あるいは、多くの目標軌道に対して最適なパラメータの値をあらかじめ獲得しておいて、それらを適応的に組み合わせることによって、制御パラメータの新たな最適値系列を構築する方法が必要である。

参考文献

- [1] Hodgins, J. K., Wooten, W. L., Brogan, D. C. and O'Brien, J. F.: Animating Human Athletics, *Proc. Ann. Conf. Comput. Graphics and Interactive Tech.*, ACM Press, pp. 71-78 (1995).
- [2] Miller, G. S. P.: The Motion Dynamics of Snakes and Worms, *Proc. Ann. Conf. Comput. Graphics and Interactive Tech.*, ACM Press, pp. 169-173 (1988).
- [3] Tu, X. and Terzopoulos, D.: Artificial Fishes: Physics, Locomotion, Perception, Behavior, *Proc. Ann. Conf. Comput. Graphics and Interactive Tech.*, ACM Press, pp. 43-50 (1994).
- [4] Wu, J. and Popović, Z.: Realistic Modeling of Bird Flight Animations, *ACM Trans. Graph.*, Vol. 22, No. 3, pp. 888-895 (2003).
- [5] Sims, K.: Evolving Virtual Creatures, *Proc. Ann. Conf. Comput. Graphics and Interactive Tech.*, ACM Press, pp. 15-22 (1994).
- [6] Mirtich, B.: Fast and Accurate Computation of Polyhedral Mass Properties, *J. Graphics Tools*, Vol. 1, No. 2, pp. 31-50 (1996).
- [7] 小野 功, 佐藤 浩, 小林重信: 単峰性正規分布交叉 UNDX を用いた実数値 GA による関数最適化, *人工知能学会誌*, Vol. 14, No. 6, pp. 1146-1155 (1999).
- [8] Eshelman, L. J. and Schaffer, J. D.: Real-Coded Genetic Algorithms and Interval-Schemata, *Foundations of Genetic Algorithms* (Rawlins, G. J. E., ed.), Morgan Kaufmann, second edition, pp. 187-202 (1993).
- [9] 佐藤 浩, 小野 功, 小林重信: 遺伝的アルゴリズムにおける世代交代モデルの提案と評価, *人工知能学会誌*, Vol. 12, No. 5, pp. 734-744 (1997).