

仮想筋肉による人工生物の遊泳行動の解析

北海道大学 大学院情報科学研究科 ○福本晃宏, 鈴木育男, 山本雅人, 古川正志
要旨

人工生物の行動獲得において従来の研究では関節の角度やトルクを制御するモデルが多い。そこで本研究では、実際の生物のように弾性バネモデルを用いて表現した仮想筋肉を制御して運動するモデルを提案する。自律的な行動を獲得させるために人工生物の制御器には人工ニューラルネットワークを使用し、その結合荷重を進化的計算によって最適化する。そして従来の関節制御の場合と比較して得られる行動の違いを解析する。

1. 結言

人工生物の自律的な行動の獲得に関する研究では K. Sims の研究¹⁾に代表されるように、関節の角度やトルクを制御する方法が多い。しかし実際の生物は筋肉を制御することによって間接的に関節に対して力を加えることで行動している。そこで本研究では弾性バネモデルを筋肉に見立てた仮想筋肉と考え、これをアクチュエータとして遊泳する人工生物をモデル化する。また、関節を直接制御するモデルと得られる行動の違いを解析する。

本稿では、物理現象を高速に計算可能な仮想水中環境での数値シミュレーションを基に実験を行う。この仮想水中環境を利用することによって現実にも似た行動が獲得されることが期待できる。また、自律的な行動を獲得させるためにアクチュエータである関節及び仮想筋肉の制御器には、環境の情報を入力することができ、ある程度の汎化能力が期待できる人工ニューラルネットワーク(Artificial Neural Network, ANN)を使用する。制御器の結合荷重は進化的計算である遺伝的アルゴリズム(Genetic Algorithm, GA)と粒子群最適化(Particle Swarm Optimization, PSO)を用いて最適化する。このとき各個体の評価値は遊泳に関する適応度を用いる。

2. 仮想水中環境

本研究では NVIDIA 社が提供する物理エンジンである PhysX²⁾を利用して物理現象をシミュレートする。PhysX では剛体や弾性体のシミュレーションが容易であり、かつ高速に実行可能である。また、Nakamura らの研究³⁾を参考にして水中における浮力と抗力をモデル化し、高速にシミュレーション可能な仮想水中環境を使用する。

3. 弾性バネモデルによる仮想筋肉

弾性バネモデルによって筋肉の役割をする仮想筋肉を表現する。仮想筋肉では筋肉の伸縮を表現するために自然長を変更する。仮想筋肉 s における力 F_s は式(1)のように表される。

$$F_s = -k_s x - b_s \dot{x} \tag{1}$$

ここで k_s は弾性係数、 b_s は粘性係数、 x は自然長からの変位である。また主動筋と拮抗筋のように互いに機能を補うような制御方法を実現するために、本研究では対となる 2 本の仮想筋肉をそれぞれ式(2)、(3)を用いて自然長の変更を行う。すなわち仮想筋肉の制御では 1 つの制御パラメータ $z(t)$ を用いて 2 本の仮想筋肉 a, b の伸縮を同時に制御する。

$$L_a(t) = (1 + \sigma_a z(t)) L_a(0) \tag{2}$$

$$L_b(t) = (1 - \sigma_b z(t)) L_b(0) \tag{3}$$

ここで $L_a(0), L_b(0)$ は変更する前の基準となる自然長、 $L_a(t), L_b(t)$ は t ステップ時における変更後の自然長、 σ_a, σ_b は仮想筋肉における自然長の最大変化率である。ただし対となる仮想筋肉では弾性係数、粘性係数および自然長の最大変化率は等しい値を用いる。

4. 人工魚モデル

本研究では図 1 に示す単純な人工魚モデルを使用し、制御

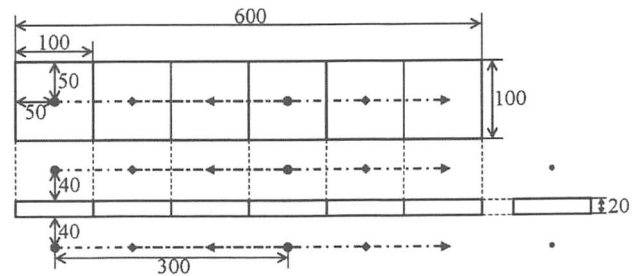


Fig. 1 The artificial fish model. (unit: mm)

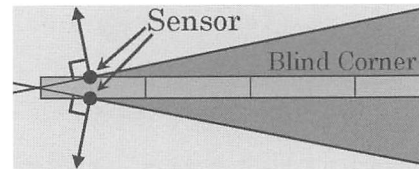


Fig. 2 The sight and the blind corner of the fish model.

方法を変えることによって得られる遊泳行動の違いを解析する。図 1 において破線は仮想筋肉による制御の場合の仮想筋肉の配置を示している。

人工魚モデルにおける剛体の個数は 6 個で、隣り合う剛体同士は関節により結合されている。すべての関節は 1 自由度で変形する最大角度は $\pm 30[\text{deg}]$ とする。剛体の密度は水の密度と等しい値に設定し、重力による力と水中の浮力による力が等しくなるようにする。

4.1 センサ

人工魚モデルには図 2 に示すような視覚センサが付いており、光源の位置を知ることができる。各視覚センサにはそれぞれ視覚ベクトルが設定され、光源方向の角度が認識できる。ただし死角となる位置に光源がある場合は光源の位置が認識できない。

4.2 仮想筋肉の配置

人工魚モデルにおける仮想筋肉は魚の体側筋を模倣して剛体の左右の側面に配置する。このとき結合位置は剛体の重心から 50[mm]の位置とする。また、複数の関節をまたぐように配置することで複数の関節に対して同時に力を加えることができる。そのようにすることで仮想筋肉の総数を減らすことができ、制御パラメータの数を抑えることが可能である。本稿では 3 つの関節をまたぐように配置する。これにより、仮想筋肉の総数は 6 本となる。

5. 人工ニューラルネットワーク

仮想筋肉の制御には ANN を使用する。使用する ANN はフィードフォワードニューラルネットワークであり、中間層と出力層の出力値は式(4)、(5)で計算される。

$$z_j = s \left(\sum_i w_{ij} v_i - a_j \right) \tag{4}$$

$$s(x) = \frac{2}{1 + e^{-px}} - 1 = 2c_p(x) - 1 \tag{5}$$

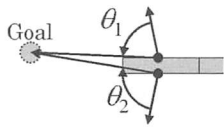


Fig. 3 θ_1 and θ_2 are used for inputs of the ANN.

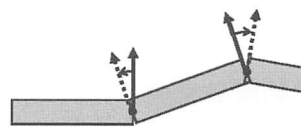


Fig. 4 The angles of the joints.

Table 1 The parameters of GA and PSO.

Common parameters		GA		PSO	
Solution length	135	Elite number	10	φ_w	0.9
Population	50	Crossover	60%	φ_1	2.0
Iteration	500	Mutation	1%	φ_2	2.0

ここで、 z はニューロンの出力値、 w は結合荷重、 y はニューロンへの入力値、 a はニューロンの閾値、 $s(\cdot)$ はゲインが p のシグモイド関数 $\sigma_p(\cdot)$ の値域を(0,1)から(-1,1)にした関数である。入力層と中間層のニューロン数はそれぞれ9個とした。

5.1 入力

入力値は人工魚モデルから見た目的地への情報とモデルの関節の角度情報を使用する。目的地への情報は人工魚のモデルに設定された2つの視覚センサから図3のように角度 θ を求め、連続値で表現できるようにするために $\cos\theta$ と $\sin\theta$ を利用する。モデルの関節の角度情報は図4のように $[-\pi/6, \pi/6]$ の範囲で入力する。

5.2 出力

まず関節を制御する場合の出力値について説明する。関節を制御する場合は、その目標角度を出力値によって指定することで制御を行う。このとき値の範囲は $[-1, 1]$ であり、値が1の場合は正の方向に30[deg]、-1の場合は負の方向に30[deg]となる。次に仮想筋肉によって制御する場合は、出力値を先に説明した仮想筋肉の制御パラメータとして直接使用する。

6. 進化的計算

ANNと仮想筋肉のパラメータを最適化するために進化的計算のGAとPSOを使用する。また事前実験によりGAとPSOをハイブリッドした手法で良い個体が得られることが確認されているため、最適化の実験を行う際にはハイブリッドした手法を使用する。GAとPSOの設定を表1にまとめる。

6.1 遺伝的アルゴリズム

GAでの遺伝操作には、エリート保存、各ニューロンにおけるパラメータ毎に対しての1様交叉及び突然変異を行う。突然変異ではパラメータ毎に突然変異確率に従い1様乱数によって値を変更する。

6.2 粒子群最適化

PSOにおける解群の更新は次の式(6)及び(7)を使用する。

$$v_i^{t+1} = \varphi_w v_i^t + \varphi_1 R_1 (x_i^b - x_i^t) + \varphi_2 R_2 (x_g - x_i^t) \quad (6)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (7)$$

ここで粒子 i について、 v_i は速度、 x_i は解群、 φ_w は慣性係数、 φ_1 は各粒子における今までの最良解 x_i^b に関する係数、 φ_2 は粒子全体の中での最良解 x_g に関する係数である。また R_1 と R_2 は対角成分が $[0, 1]$ の実数値乱数である対角行列である。

6.3 適応度関数

GAとPSOで使用する適応度関数 F は、より速く目的地に到着することを評価するために式(8),(9)のように設定する。

$$F = \sum_i D_i \quad (8)$$

$$D_i = \begin{cases} \max\left(0, 1 - \frac{d_i}{d_0}\right) & \text{before goal} \\ 1 & \text{after goal} \end{cases} \quad (9)$$

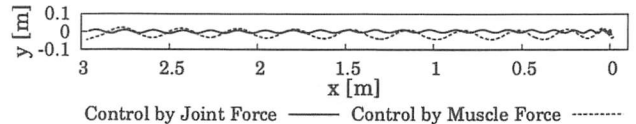


Fig. 5 The swimming trajectory of the best individual.

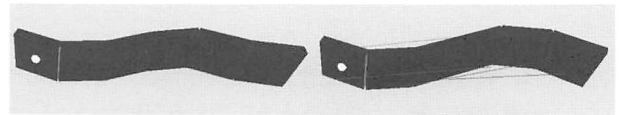


Fig. 6 Left side fish is the control by joint force model and right side one is the control by muscle force model.

ただし、 D_i は t ステップ時の距離に関する適応度、 d_i は t ステップ時の目的地までの距離、 d_0 は目的地までの初期距離である。目的地に到着したあとの適応度は最大値(=1)に固定する。

7. 遊泳獲得実験

作成した人工魚モデルでそれぞれの制御方法を用いて遊泳行動の獲得実験を行い、獲得した遊泳行動の違いを比較する。実験条件は目的地を人工魚モデルの正面に配置し、初期距離は3[m]とする。また数値シミュレーションは1ステップを1/60[sec]とし、計600ステップの10秒間行う。

7.1 実験結果

それぞれの実験によって得られた最良個体同士の遊泳行動を比較する。まず、得られた遊泳行動の軌跡の比較を図5に示す。これはモデルの先頭と末尾の剛体における重心の軌跡をプロットしたものである。この遊泳の軌跡から関節を制御する方法より仮想筋肉によって制御する方法の方が大きな動きになっており、ストローク回数が少ないことが確認できる。次に遊泳時の人工魚モデルの姿勢を図6に示す。この図を観察すると関節を制御する方法の場合は身体の波が細かく、仮想筋肉によって制御する場合は波が大きく滑らかなことが分かる。これは仮想筋肉が複数の関節に対して同時に力を加えることが可能で、大きな動きの運動を制御しやすいためであると思われる。

8. 結言

本研究では弾性バネモデルを仮想筋肉として制御する方法を提案し、関節を制御する方法の遊泳行動と比較し特徴を解析した。その結果、仮想筋肉を使用して制御した場合に大きな波の振動を伝播させることができ、滑らかな遊泳が可能であることが分かった。

本稿では魚やうなぎのような関節が明確に分かれていない生物を、関節のある人工魚モデルを用いて仮想筋肉による制御方法の解析を行った一方、関節が明確な生物に対して仮想筋肉を適用した場合にどのような特徴の行動が得られるか解析を行うことも必要である。また、仮想筋肉を用いて行動の制御を行う際に人工生物の設計者の負担を軽減するために、どこに配置するかなども含めた最適化手法の開発についても今後の課題である。

参考文献

- 1) K. Sims, "Evolving 3D morphology and behavior competition", *Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems, MIT, Cambridge, MA, USA*, pp.28-39, 1994.
- 2) NVIDIA PhysX, http://www.nvidia.com/object/physx_new.html
- 3) K. Nakamura, M. Morinaga, K. Iwadate, K. Yoneda, I. Suzuki, M. Yamamoto, M. Furukawa, "Studies on Modeling Environment for Physics Modeling", *Conference Abstracts of Fourth International Symposium on Adaptive Motion of Animals and Machines*, pp.123-124, 2008