

物理法則に基づく蝶モデルの飛行

北大 ○大江 亮介, 北大 鈴木 育男, 北大 山本 雅人, 北大 古川 正志

A Flight of a Butterfly Model Based on the Physical Law

Hokkaido Univ. Ryosuke OOE, Hokkaido Univ. Ikuo SUZUKI, Hokkaido Univ. Masahito YAMAMOTO,
Hokkaido Univ. Masashi FURUKAWA

To represent a more real butterfly's flight than it was, a butterfly model is designed and simulations based on the physical law are performed. The physical engine library is used for calculation of the rigid-body dynamics, and a fast algorithm on the basis of the drag force is implemented additionally for the fluid dynamics. The wings are designed to be able to do the "flapping" and the "feathering". The flapping is a swing up and down motion of a wing, and the feathering is a twist motion of a wing. A butterfly model flaps according to the data given by measurement for a real butterfly. Feathering angles of wings are controlled by the artificial neural network, and its weights are optimized for a proper flight.

1. はじめに

近年、バーチャルリアリティ (VR) や映像製作において、コンピュータグラフィックス (CG) を用いる機会が増加している。特にポリゴンで構成される 3D モデルを用いた CG 表現は、現実には存在しない仮想的な生物や構造物の描写などに利用されている。リアリティの高い 3D モデルの動作表現には、高精度なモーションキャプチャを用いる場合が多い。しかし、モーションキャプチャで生成された動作は静的なものであり、環境が変化した場合や、外界からの刺激に動的に反応する場合における再利用性が乏しい。

本論文では、リアリティが高く、動的に変化するような 3D モデルの動作生成を目的とし、物理法則に則った蝶の飛行シミュレーションについて述べる。蝶は自然景観の一部として比較的多く用いられる対象であり、映像の背景を彩る要素としての応用が期待できる。

蝶の飛行動作では、流体から受ける流体力を考慮する必要がある。羽ばたきによる力を考慮した研究として、安齋ら⁽¹⁾による蝶の飛行モデルがある。このモデルでは、羽上の代表的な 1 点の速度を用いて羽ばたきによって生じる流体力を計算しているが、蝶の重心位置の移動のみを考慮しており、剛体力学に従うモデルではない。

本研究では、近年開発が進められている物理エンジン PhysX を用いて剛体力学を処理し、羽と胴体を受ける流体力の計算に、物体表面の複数の点を用いる。フラッピング運動については、実際の蝶の測定データを基に与える。一方フェザリング運動は、人工ニューラルネットワーク (ANN) のパラメータを、粒子群最適化 (PSO) に基づいて最適化することで求める。

2. 流体力計算

鳥や航空機の翼が空気から受ける力の簡易的な計算は、揚力と抗力を別々に計算するのが一般的である。しかし本研究では、胴体を受ける力も含めて統一的に扱うために、物体表面の微小面にかかる流体力 ΔF を式 (1) で計算した。

$$\Delta F = \frac{1}{2} \Delta A \rho C_d v^2 \quad (1)$$

ここに、 ρ は流体の密度、 ΔA は微小面の面積である。 \mathbf{v} は微小面の重心の速度のうち、微小面に垂直な成分のみを取り出したベクトルである。また、 C_d は抗力係数であり、本来は物体の形状や寸法によって異なる値をとる。予備実験より、 $C_d = 1.1$ 程度を用いた場合、実際の物体が受ける流体力に近くなることがわかっている。しかし、文献⁽²⁾に示されているように、実際の昆虫の羽ばたきでは、式 (1) のような速度の 2 乗に比例する流体力に対して、約 2 倍の流体力が働いている。よって、本稿で行う実験では、全ての物体について $C_d = 2.2$ を用いた。また、 ρ として、温度が 5.0 [°C] 程度のときの空気の密度 $\rho = 1.28 [\text{kg/m}^3]$ を用いた。

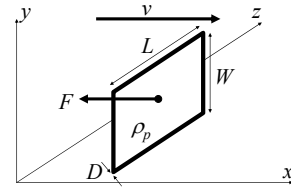


Fig. 1 Moving plane in fluid

3. シミュレーションの安定性

式 (1) の計算を物理エンジンに導入した場合、タイムステップ Δt の条件について述べる。今、図 1 のような長辺 $L [\text{m}]$ 、短辺 $W [\text{m}]$ 、厚さ $D [\text{m}]$ 、密度 $\rho_p [\text{kg/m}^3]$ の平板が、 x 軸の正の方向に速度 $v [\text{m/s}]$ で移動しているとすると、このとき平板には、 x 軸の負の方向に流体力 $F [\text{N}]$ が働き、その大きさは式 (2) で計算される。

$$F = -\frac{1}{2} L W \rho C_d v^2 \quad (2)$$

F によって、 Δt [s] 後には平板の位置・速度は変化している。このときの速度 v' は、質量 $m = W L D \rho_p$ [kg]、運動方程式 $F = ma$ より、単純なオイラー法を用いた場合、式 (3) のように求められる。

$$v' = v - \frac{\rho C_d v^2}{2 D \rho_p} \Delta t \quad (3)$$

運動する物体が流体から受ける力は、運動を妨げる向きにのみ作用すべきである。従って、 $v' > 0$ が必要となる。ただし、より正確な計算を行うためには、 v' は v に近いほど良い。また、 $v' < -v$ のときには、 v' は時間の経過とともに発散し、エラーを引き起こす。よって、安定にシミュレーションを行うためには、最悪でも $v' > -v$ を満たさなければならない。 $v' > -v$ となるためのタイムステップ Δt の条件は、式 (4) で与えられる。

$$\Delta t < \frac{4 D \rho_p}{\rho C_d v} \quad (4)$$

式 (4) より、物体のスケールや密度が小さいほど、また速度が大きいくほど、 Δt を小さくする必要があることがわかる。ただし、実際の運動方程式では、重力や羽ばたきを行うための内力を考慮する必要があり、回転の式についても同様の条件を求める必要がある。

4. 蝶モデルのパラメータ

蝶モデルは図 2 のように、胴体、左羽、右羽の 3 つの剛体から構成されている。羽の形状は実際のアゲハを基に作成しているが、胴体部分の構造はやや異なる。黒線で囲まれた各領域が、流体力計算に用いる微小面である。各パラメータは文献⁽¹⁾と等しく設定して

おり、総重量は0.319[g]、羽の最大幅は5.5[cm]、フラッピング角の範囲は[-75, 90]、フェザリング角の範囲は[-5, 20]とした。ただし、フラッピング角は羽を振り上げたときを正、フェザリング角は羽前部が上方に向くときを正としている。また、胴体と両方の羽の重量の比は5:1に設定しており、胴体の重量は0.266[g]、羽の重量はそれぞれ0.0265[g]となっている。

5. フラッピングのみの飛行

文献⁽¹⁾に、実際の蝶の測定データを基に作成されたフラッピング角度関数が示されている。これを参考に、図3(a)を1周期のフラッピング角度関数とし、フェザリング角度を常に0としたときの蝶モデルの飛行を観察した。

設定したパラメータ及びフラッピング角度で、タイムステップ Δt を数パターン変化させてシミュレーションを行ったところ、 Δt はおおよそ 1.0×10^{-4} 程度が必要であり、これよりも大きいとエラーが発生した。一方、蝶モデルのサイズのみを100倍してシミュレーションを行ったところ、 Δt が 1.0×10^{-2} 程度でもエラーは発生しなかった。このことから、物体のスケールが小さくなるほど、 Δt を小さくする必要があることが実験的にわかった。よって、以降の実験は全て $\Delta t = 1.2^{-1} \times 10^{-4}$ として行った。このとき、CPUの周波数が1.66[GHz]の計算機で10[s]、すなわち 1.2×10^5 ステップのシミュレーションを行うのに、約66秒の計算時間を要した。

図3(b)に、フラッピングのみの飛行時の8.7[s]までの胴体の重心位置の推移を示す。蝶の飛行の特徴である胴体の上下動が確認できるが、前後方向にも振動しており、ノコギリの刃のような飛行軌跡となっている。また、徐々に前傾しながら墜落しており、フェザリングなしでは安定した飛行は不可能であることがわかった。

6. フェザリング角度の最適化

安齋⁽¹⁾の研究では、フェザリング角度関数はシミュレーションの繰り返しによって試行錯誤で求められており、パラメータ調整の自動化が課題として挙げられている。そこで本研究では、墜落せずに飛行可能なフェザリング角度を自動的に求めることを試みた。

フェザリング角度を制御するための制御器として、3層フィードフォワード型のANNを用いた。入力層のニューロン数は4個とし、胴体を横に貫く軸回りの角度(ピッチ角)、高度、フラッピング角度、フラッピング状態を入力信号とした。ただし、フラッピング状態は、羽を振り上げているときに1、振り下ろしているときに-1となる信号である。中間層のニューロン数は4個、出力層のニューロン数は1個で、左右の羽共通のフェザリング角度を出力する。

ANNの重みパラメータを最適化するために、PSOを用いた。ANNを遺伝的アルゴリズム(GA)やPSOで最適化する制御手法は、制御対象の目的関数に関する知識などを必要とせずに最適化を行うことが可能であり、未知の制御問題に対しても適用が可能である。ANNの重み配列を解ベクトルとし、粒子数20個、繰り返し探索回数100回として解の探索を行った。

評価関数 H は式(5)で与えた。

$$H = \sum_{t=1}^{120000} h(t) \quad (5)$$

$$h(t) = \begin{cases} x & : |y| \leq 0.05 \\ |1-y|x & : |y| \leq 0.5 \end{cases}$$

ここに、 x [m] はステップ t における水平方向の距離、 y [m] は初期高度を0としたときの高度である。 1.2×10^5 ステップ経過後か、あるいは $|y| > 0.5$ となった時点までを1回のシミュレーションとした。また、エラーが発生した場合は評価を負の無限大とした。



Fig. 2 Shape of the butterfly model

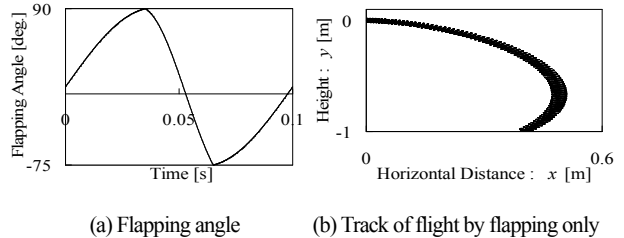


Fig. 3 Flapping angle and track of flight

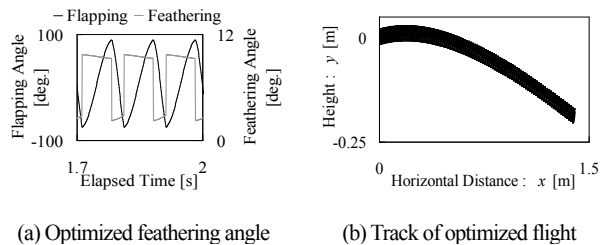


Fig. 4 Optimized feathering angle and track of flight

最適化の結果、PSOの探索回数1回の際の最大評価値が 2.7×10^4 であったのに対し、最適化後に得られた最大評価値は 7.6×10^4 であった。図4(a)に最適化されたフェザリング角度を示す。フェザリング角度の範囲は[-5, 20]だが、ほぼ[2, 10]の範囲しか出力されておらず、フラッピング状態のみに大きく反応している。ただし、全体的な波形は安齋⁽¹⁾が求めたフェザリング角度を単純化したような形となっている。

図4(b)に、最適化された飛行時の10[s]までの胴体位置の推移を示す。フラッピングのみの飛行と比較して、高度維持と前進の性能が向上しているが、初期高度を維持することはできなかった。また、安齋⁽¹⁾のシミュレーションでは、フェザリングを用いた場合3.6[m/s]の前進速度が得られているが、本研究では10[s]での前進速度が約0.13[m/s]であり、大きく異なっている。これについては、探索中にエラーの発生が何度か見られたことから、設定したタイムステップが3.6[m/s]での前進には十分でなかった可能性や、胴体に働く流体力の影響などが考えられるが、明確な原因は不明である。

7. おわりに

本論文では、物理法則を考慮した蝶モデルを作成し、ANNとPSOを用いて自動的に適切なフェザリング角度を求めて飛行させた。その結果、前進速度が実際の蝶よりも遅くなるという大きな問題が生じた。この問題の原因解明・解決と、シミュレーション・最適化における計算量の低減が今後の課題である。

参考文献

- (1) 安齋祐一, 村岡一信, 千葉則茂, 齊藤伸自: “翅のはばたきによる力を考慮した蝶の飛行モデル”, 情報処理学会論文誌, Vol.41, No.3 (2000) 658-667
- (2) 河内啓二: “昆虫の飛行メカニズム(流体力学的視点から)”, 生物物理, Vol.39, No.5 (1999) 279-284