



HOKKAIDO UNIVERSITY

Title	複合的人工ニューラルネットワーク : 教師なし学習を用いた動的制御における選択型ニューラルネットワークアンサンブル
Author(s)	大江, 亮介; 鈴木, 育男; 山本, 雅人 他
Citation	精密工学会誌, 79(6), 552-558 https://doi.org/10.2493/jjspe.79.552
Issue Date	2013
Doc URL	https://hdl.handle.net/2115/64600
Type	journal article
File Information	79_552.pdf



複合的人工ニューラルネットワーク*

—教師なし学習を用いた動的制御における選択型ニューラルネットワークアンサンブル—

大江亮介** 鈴木育男*** 山本雅人† 古川正志†

Composite Artificial Neural Network

- Selection-type Neural Network Ensembles in Dynamic System Control by Unsupervised Learning -

Ryosuke OOE, Ikuo SUZUKI, Masahito YAMAMOTO and Masashi FURUKAWA

This paper describes a novel method of combining artificial neural networks (ANNs), the composite artificial neural network (CANN), to improve performance of evolving ANNs (EANNs) in dynamic system control problems. Methods of combining ANNs by majority voting or averaging are not effective in controlling dynamic systems by EANNs. Unsupervised learning of EANNs is mathematically described, and then it is shown that the reason for ineffectiveness depends on the mechanism of evaluating ANNs indirectly by states. To avoid this problem, the CANN selects the suitable ANN by a high-level ANN to combine some ANNs. In numerical experiments, a flapping flight model is controlled by a common EANN and the CANN. The model motion is calculated by physical engine PhysX, and a common EANN and the CANN are optimized by the particle swarm optimization (PSO) respectively. The experiments show that the average evaluation of the CANN is 6.46% higher than that of a common EANN for the same computational time.

Key words: artificial neural networks, evolving artificial neural networks, unsupervised learning, neural network ensembles, composite artificial neural network, physical engine, particle swarm optimization

1. 緒 言

現代社会において制御は非常に重要な技術のひとつである。その適用範囲はエアコンの温度制御からロボットアームの運動、あるいは複雑な構造を持つ多脚移動型ロボットまで多岐に渡る。有効な制御系の設計を目的として、制御対象であるシステムを数理モデルで記述する制御理論の研究が数多く行われてきた。従来の制御理論では数理モデルのパラメータを正確に求める必要があったが、近年では線形 H_∞ 制御理論¹⁾や非線形 H_∞ 制御理論²⁾⁻³⁾などにより、モデルが多少不確かでも制御が可能である。

制御対象の記述を基に制御系を設計する方法に対して、インテリジェント制御は制御対象の物理的情報を用いずに制御系を設計する方法である。インテリジェント制御には、曖昧さを許容したファジィ集合を用いて制御系を設計するファジィ制御⁴⁾や、脳の神経素子をモデル化した人工ニューラルネットワーク (Artificial neural network, ANN) 制御などがある。特に最適化手法を用いて ANN の学習を行う方法は EANN (Evolving ANN)⁵⁾と呼ばれ、二足歩行ロボットの制御⁶⁾などに用いられている。EANN は生物の学習・進化のように自律的な制御系設計が可能である。しかし一方で、試行錯誤的に学習が進むため制御系の設計に長い時間を要し、必ずしも最適解を発見できるとは限らない。

ANN は制御問題でも用いられるが、与えられた学習データの集合を複数のクラスに分類するクラス分類問題で用いられることが多い。この問題領域では、複数の ANN を組み合わせることによって能力を向上させるアンサンブル学習の研究が行われてきた。アンサンブル学習における ANN の学習方法には、ブートストラ

ップサンプリングによって複数の ANN を独立に学習させるバギング⁷⁾や、学習率を変化させて一括して学習させるブースティング⁸⁾⁻⁹⁾がある。また、複数の ANN を組み合わせる方法として、多数決¹⁰⁾、単純平均¹¹⁾、重み付き平均¹²⁾などが用いられている。

従来のアンサンブル学習は有限個の学習データの分類には効果的だが、EANN による動的なシステムの制御には適していない。これはクラス分類における教師あり学習と EANN による制御における教師なし学習の概念が本質的に異なるためである。そこで本論文では、EANN による制御問題に対して効果的なアンサンブル学習を実現するため、選択型のアンサンブルを行う複合的 ANN (Composite ANN, CANN) を提案する。

著者らの研究グループではすでに、選択型のアンサンブルを用いて円環構造弾性ロボットの行動制御を行っている²²⁾。その際、十分に学習を行った単純な行動を組み合わせることで、未学習の環境における複雑な行動生成に成功した。本研究では、CANN により性能の低い ANN を組み合わせることで、学習した環境での行動が改善可能かを検証する。同等の学習時間で比較して従来の EANN よりも制御能力が向上するかを検証するために、羽ばたき飛翔の制御を対象とした数値計算実験を行う。

以下に本論文の構成を示す。2 章では学習データが与えられる教師あり学習問題と EANN による制御問題を数理的に記述し、両者の差異と問題点を明らかにする。さらに問題点を解決するためのアンサンブル戦略について考察を行う。3 章では導かれた戦略に基づく CANN の学習方法および制御信号の計算手順を示す。また、4 章では CANN の有効性検証のために計算機を用いた数値計算実験を行う。実験結果として、学習時間が等しい場合の従来の EANN と CANN の評価の比較、および導かれた戦略の有効性を示す事例を挙げる。5 章に結論および今後の課題を示す。

* 原稿受付 平成 24 年 10 月 20 日

** 学生会員 北海道大学大学院 (北海道札幌市北区北 14 条西 9 丁目)

*** 正 会 員 北見工業大学 (北海道北見市公園町 165)

† 正 会 員 北海道大学大学院

2. 学習の数理的解析

2.1 人工ニューラルネットワークの教師あり学習

ベクトル $\mathbf{x} \in X$ を変換したときの正解が $\mathbf{y} \in Y$ であるとき、その組 $(\mathbf{x}, \mathbf{y})$ を学習データとする。ANN の教師あり学習では、与えられた Q 個の学習データから正しい写像 $f: X \rightarrow Y$ を学習する。これは式(1)を満たす f を求める問題として定式化できる。

$$\text{Minimize } \sum_{i=1}^Q (f(\mathbf{x}_i) - \mathbf{y}_i)^2 \quad (1)$$

ここに、 $\mathbf{x}_i, \mathbf{y}_i$ は i 番目の学習データである。具体的な学習アルゴリズムとしては、ランダムに選択した学習データ $(\mathbf{x}_i, \mathbf{y}_i)$ に対して二乗誤差 $(f(\mathbf{x}_i) - \mathbf{y}_i)^2$ が小さくなるように学習を進める最急降下法が多く用いられる。

最急降下法は確率的なアルゴリズムであり、学習後の ANN は誤差 $f(\mathbf{x}_i) - \mathbf{y}_i$ を 0 にできるとは限らない。しかしクラス分類問題では、判断材料、たとえば値の正負や要素間の大小関係が満たされていれば分類が可能である。また正解を表す学習データは一意であり、学習後の誤差 $f(\mathbf{x}_i) - \mathbf{y}_i$ は平均 0 の分布に従うと期待できる。これらの性質はアンサンブル学習において単純な多数決や平均が有効に機能する要因である。

2.2 進化人工ニューラルネットワークの教師なし学習

制御対象の現在の状態 $\mathbf{x}_i \in X$ および制御信号 $\mathbf{y}_i \in Y$ から、次の状態 $\mathbf{x}_{i+1} \in X$ を求める写像を $\lambda: (X, Y) \rightarrow X$ とする。EANN の教師なし学習では、制御対象の状態から制御信号を求める写像を学習する。ただし正しい写像に関する学習データは与えられないため、時間変化する連続な状態 Q 個を評価基準として学習を行う。これは評価関数 E が与えられたとき、式(2)を満たす写像 $\mu: X \rightarrow Y$ を求める問題として定式化できる。

$$\begin{aligned} &\text{Maximize } E(s(\mu)) \\ &\text{s.t. } s(\mu) = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_Q), \quad \mathbf{x}_{i+1} = \lambda(\mathbf{x}_i, \mu(\mathbf{x}_i)) \end{aligned} \quad (2)$$

具体的な EANN の学習アルゴリズムとしては、試行錯誤的に解を改善してゆく最適化手法が用いられる。これは状態変換を表す写像 λ をブラックボックスとして扱い、 λ の情報を学習に用いないためである。一般に最適化に長い時間を要するが、 E の設計のみで制御系設計が可能であり汎用性は高い。

EANN の教師なし学習では、評価 E を最大化する状態列 $s(\mu)$ の探索により間接的に μ の学習を行う。この間接的な評価構造から、以下に示す 3 つの問題が生じる。

i) 学習データの局所的分布

最適化を M ステップ行ったとき、写像 $\mu_i (i = 1, 2, \dots, M)$ および μ_i に対応する状態列 $s(\mu_i)$ が得られる。よって状態空間の探索は十分であり、最終的に得られる μ_M は可能な全ての $\mathbf{x}$ に対して正しい応答を行うかという点、そうではない。 μ_M の応答は $s(\mu_M)$ のみで評価されており、最適化過程で得られた状態列 $s(\mu_j) (1 \leq j \leq M-1)$ は μ_M の評価に影響しない。つまり、 $s(\mu_M)$ に含まれる状態を保証状態と定義すると、保証状態以外の状態に対して μ_M が適切に応答可能かどうかは全く保証されない。端的な例を挙げると、目的状態 $\mathbf{x}_g$ に向かう制御を学習した場合、 $\mathbf{x}_g$ までの最短経路以外の状態で制御が失敗する可能性がある。学習後の ANN の写像 μ_M について、保証状態が状態空間内の極めて局所的みに分布する性質を学習データの局所的分布と定義する。

ii) 解の局所収束

今 ANN の写像が $\mu_k (1 \leq k \leq M-1)$ であるとする。最適化は試行

錯誤的に進むため、 $E(s(\mu_{k+1})) > E(s(\mu_k))$ を満たす μ_{k+1} を求めるのは容易ではない。特に、最適ではないが任意の微小変化を加えても評価が向上しない μ で最適化が停滞することがある。これは解の局所収束と呼ばれ、最適化手法を用いた探索においてよく知られている問題である。その解決法としては、解を大きく変化させるメカニズムを最適化手法に実装することが多い。また並列に学習させた複数の ANN 間の多様性を利用し、局所収束の発生を抑える研究もある¹³⁾⁻¹⁴⁾。しかし最適化が確率に左右される以上、局所収束を確実に回避することは不可能である。

iii) 許容可能解の非一意性

クラス分類問題とは異なり、制御問題では $\mu(\mathbf{x})$ の値について評価の大小関係は定義できるが、正しいか誤りかは定義できない。また、 $\mu(\mathbf{x})$ がわずかでも異なれば次の状態は異なる。この性質から、制御問題ではアンサンブル学習において単純な多数決は不可能である。また同様に、高い評価を与える $\mu(\mathbf{x})$ が一意でない可能性を考慮する必要がある。たとえば二足歩行や鳥の周期的な羽ばたきのように、左右対称な動作や逆位相の動作の評価が等しくなる場合がある。制御問題では制御信号に対称性や周期性が現れやすいため、平均によるアンサンブルを用いた場合制御信号が予期せぬ値に収束しやすい。

2.3 アンサンブル戦略についての考察

先に示したように、iii) の性質から制御問題ではアンサンブル方法として多数決を用いることができず、平均も効果を発揮できない。よって EANN で効果的なアンサンブル学習を実現するためには、別の新たなアンサンブル方法を用いる必要がある。

ここで i) の性質より、単一の ANN が持つ保証状態は少ない。よって非常に単純なアイデアとして、相異なる保証状態を持つ複数の ANN を組み合わせる方法が挙げられる。この場合、制御対象の現在の状態を保証状態として持つ ANN を選択して用いることにより、より多くの状態に対して適切な制御信号を計算できる。現在の状態を基にひとつの制御器を選択して用いる方法は、包摂アーキテクチャ¹⁵⁾と類似したアイデアである。ただし、このアンサンブル方法には以下の 2 点の問題がある。

第一の問題は、制御対象の現在の状態が複数の ANN の保証状態である場合、あるいはどの ANN の保証状態でもない場合に、どの ANN を用いれば良いかわからない点である。この解決法のひとつとして、制御対象の現在の状態から適切な ANN を選択する選択器として ANN を用い、EANN の教師なし学習によって適切な選択を学習する方法が挙げられる。

第二の問題は、複数の ANN をどのように得るかである。上記の方法は、互いに類似した保証状態を持つ ANN の組み合わせでは効果が薄い。よって最も望ましい方法は、ANN の保証状態の類似性が最小となるように、ANN の学習ごとに評価を与える関数 E の設計を変更することである。しかし、そのような関数 E の設計方法を一般化して考えることは極めて難しい。ここでは ii) の性質を逆に利用し、複数回の独立した学習によって目的とする複数の ANN が得られると仮定する。

3. 複合的人工ニューラルネットワーク

3.1 人工ニューラルネットワーク

ANN は生物の脳の神経網をモデル化したものであり、入力ベクトルを出力ベクトルに変換する非線形関数を表現可能である。ANN には大別して階層型と非階層型のものがある。階層型のものでは、入力側の層から出力側の層へと順方向にのみ信号が伝播するフィードフォワード構造や、出力が入力側の層にフィー

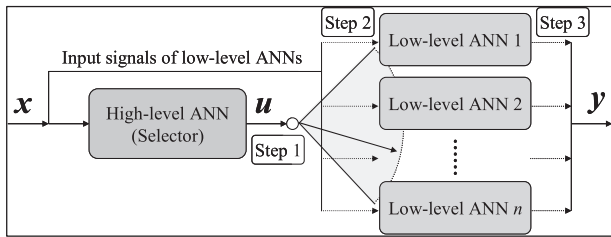


Fig.1 The concept of CANN: the high-level ANN is a selector and low-level ANNs are controllers

ドバックされるリカレント構造などがある。

ANNはニューロンと呼ばれる神経素子が相互結合された構造を持つ。あるニューロン i の出力 y_i は、他のニューロンからの入力を要素とする入力ベクトル $\mathbf{x}_i$ 、ニューロン間の結合重みベクトル $\mathbf{w}_i$ 、ニューロン i の閾値 θ を用いて式(3)のように表される。

$$y_i = \tau(\mathbf{w}_i \cdot \mathbf{x}_i - \theta_i) \quad (3)$$

ここに τ は伝達関数と呼ばれる関数であり、ANNが非線形関数を表現可能とするためには τ に非線形関数を用いる必要がある。たとえば、入力を $[-1, 1]$ の範囲に変換する拡張シグモイド関数などがよく用いられる。これは式(4)のように表される。

$$\tau(z) = 2(1 + e^{-az})^{-1} - 1 \quad (4)$$

ここに a は関数の傾きを変化させるゲインである。本研究では結合重みの値は任意の実数値を取り得る。よって最適化手法により結合重みのスケールが任意に変化するため、 $a=1$ として数値計算実験を行う。

3.2 複合的人工ニューラルネットワーク

提案するCANNは複数のANNモジュールの組み合わせにより構成される。アンサンブルによって組み合わせる複数のANNを低次ANNと定義する。また説明の都合上、 r 番目に評価の高い低次ANNを低次ANN r と定義する。2章の議論より、CANNでは低次ANNを組み合わせるアンサンブル方法として選択方式を用いる。選択を行うためのANNを高次ANNと定義する。

CANNでは従来のアンサンブル学習におけるバギングと同様に、はじめに各低次ANNを独立に学習させる。制御問題におけるEANNの教師なし学習は式(2)に示したとおりであり、低次ANNの全ニューロンの結合重みベクトルと閾値をまとめた解ベクトルを最適化手法によって最適化し、評価関数 E を最大化する学習を行う。次に、低次ANNの選択を高次ANNに学習させる。高次ANNの学習では、最適化を行う解ベクトルは高次ANNの全ニューロンの結合重みベクトルと閾値のみであるが、制御信号の計算はCANN全体で行い評価関数 E を最大化する。

図1にCANNにおける制御信号の計算手順の概念図を示す。CANNに入力される入力ベクトルを $\mathbf{x}$ 、計算される制御信号ベクトルを $\mathbf{y}$ とすると、CANNは以下の手順に従い $\mathbf{y}$ を計算する。

- Step 1 $\mathbf{x}$ を高次ANNに入力し、出力ベクトル $\mathbf{u}$ を計算する。
 $\mathbf{u}$ の次元は組み合わせる低次ANNの個数と等しい。
- Step 2 $\mathbf{u}$ を各低次ANNの利用優先度とみなし、 $\mathbf{u}$ の要素の中で最大の要素番号 $r_{\max}$ を求める。
- Step 3 $\mathbf{x}$ を低次ANN $r_{\max}$ に入力し、 $\mathbf{y}$ を計算する。この $\mathbf{y}$ がCANN全体で計算される制御信号ベクトルとなる。

CANNでは、ANNの出力計算2回、および $\mathbf{u}$ の最大要素を求める計算1回のみで制御信号ベクトルを計算する。従来のアンサンブル方法では組み合わせる複数のANN全ての出力計算が

必要であるため、CANNは従来のアンサンブル方法と比較して制御信号ベクトルの計算に要する計算コストが低い。

3.3 低次ANNの個数と探索ステップの設計

多数決や平均といった従来のアンサンブル方法では、組み合わせる各ANNが等しくアンサンブルに寄与する。ゆえに、性能の低いANNが少ないほどアンサンブルを行ったときの性能も高くなると期待される。一方CANNでは、高次ANNにより選択されない低次ANNはアンサンブルに全く寄与しない。また、性能の低い低次ANNであっても、ある特定の状態に対しては評価の高い出力を行う可能性がある。よって、高次ANNによる選択規則が適切であれば、低次ANNの個数が多いほど可能な低次ANNの組み合わせパターン数が増加し、CANNの性能は向上すると期待される。ただし、低次ANNの個数が多ければ多いほど高次ANNの出力ベクトルの次元が増加し、高次ANNの学習が困難となる。

低次ANNの個数の増加は、上記の組み合わせパターンの増加以外にも利点がある。EANNの教師なし学習は確率的なアルゴリズムにより行われるため、2章で示した解の局所収束などが発生し試行ごとに評価が大きく異なる。よって独立に学習させた低次ANNの個数が多いほど、評価の高い低次ANNを得る確率が高くなる。評価の高い低次ANNは高次ANNの選択の基準となり、状況に応じて他の低次ANNを付随的に選択するように学習が進む。

上記より、低次ANNの個数が多いほどCANNの潜在的な性能は高くなる。ただし、一般には学習のための計算時間は無限に与えられるものではなく、計算資源などの制限から一定時間で学習を完了させる必要がある。学習のための探索ステップが一定の条件下では、低次ANNの個数の増加するほど十分な学習が困難となる。ゆえに、制御対象が複雑である場合や、与えられた探索ステップが少ない場合は低次ANNの個数を少なくし、逆に制御が容易な場合や長時間の探索が許される場合には低次ANNの個数を多くすべきである。

低次ANNの個数と関連し、高次ANNの学習に割り振る探索ステップもCANNの性能に影響を与える要素のひとつである。高次ANNの学習の難しさは低次ANNの個数に比例するため、低次ANNの個数の増加に従って高次ANNの学習のための探索ステップを増加させる必要がある。ここで、低次ANNの個数が多い場合には、高次ANNの探索ステップを増加させても、低次ANN1個の探索ステップはわずかに減少しない。よって、与えられた探索ステップのうち、高次ANNの学習のための探索ステップに多くを割り振る設計が有効と考えられる。

4. 羽ばたき飛翔制御の数値計算実験

CANNの有効性を検証するために、羽ばたき飛翔を行う制御対象を従来のEANNおよび提案するCANNで制御する数値計算実験を行う。2章で示したように従来のアンサンブル方法は制御問題では有効に機能しないため、比較対象は従来のEANNとした。羽ばたき飛翔は空中に浮き続けるための継続的な翼の羽ばたき運動、および3次元空間を移動する複雑な制御を必要とするため、CANNの有効性検証に適している。ただし本実験の目的は一般的な羽ばたきの制御を通してCANNの学習の特徴を調査することであり、ある特定の羽ばたき飛翔モデルの制御系を設計し、その特性を詳細に解析することではない。また2章で示したように、EANNもCANNも制御対象の物理的情報を必要としない教師なし学習によって制御系を設計する。よって、制

御対象の動力学計算を行うために物理エンジンをうい、空気から受ける力を単純な計算で表現する。

4.1 物理エンジンによる動力学計算

物理エンジンは多体物理を数値積分によって計算するためのソフトウェアライブラリである。多体物理の運動計算では、ラグランジュの運動方程式などにより系全体の物理的特性を導出する解析の手法が多く用いられる。一方、物理エンジンは系を構成する複数の物体の運動を独立に計算する基本的な処理構造を持ち、衝突や摩擦などを外力項として扱うことにより相互作用を表現する。この運動計算方法により、物理エンジンでは拘束ジョイントの破損やバネ・ダンパを持つ拘束ジョイント、ジョイントモータなどの多様な機能を利用可能である。

ある多体物理系が複数の剛体で構成されるとき、各剛体の運動は、式(5)の運動方程式を数値積分することで計算できる。

$$\frac{d\mathbf{P}(t)}{dt} = \mathbf{F}(t), \quad \frac{d\mathbf{L}(t)}{dt} = \mathbf{N}(t) \quad (5)$$

ここに、 t は時間、 $\mathbf{P}$ は剛体の運動量、 $\mathbf{L}$ は角運動量、 $\mathbf{F}$ は剛体に働く力、 $\mathbf{N}$ はトルクである。

式(5)における $\mathbf{F}$ と $\mathbf{N}$ が相互作用の影響を含む外力項である。外力項は重力のような場から受ける力や、各剛体間の制約条件を満たすように計算され、各剛体の数値積分に反映される。制約条件から外力項を求める方法としては、連立一次方程式を数値的に解く方法が定式化されている^{16)~17)}。代表的な物理エンジンとしてBullet Physics、PhysXなどが開発されているが、本研究では多くの機能を持つPhysXを用いて数値計算実験を行う。

4.2 空気から受ける力のモデル化

流体とその内部に存在する物体の非定常的な状態変化を正確に計算するためには、有限要素法や粒子法などの流体解析手法と物体の運動計算を組み合わせた流体—構造連成解析を行う必要がある。しかし、流体—構造連成解析は計算コストが非常に高く、特に解析対象が広大な3次元空間である場合には現実的な時間での計算は難しい。本実験ではCANNの特徴を統計的に調査するために、20秒間の状態変化を60000回計算する。この目的に対して流体—構造連成解析は適当ではないため、計算コストの低い近似的なモデルを用いる必要がある。

流体中の物体が受ける流体力の簡易的なモデルとして準定常的手法がある。準定常的手法では周囲の流体の状態変化を考慮せず、物体の姿勢と流体との相対速度のみから流体力が一意に定まると仮定する。これは渦抵抗などいくつかの流体現象を無視したモデルであるが、流体—構造連成解析と比較して計算コストは非常に低い。

本実験では準定常的手法を基にして、無風状態の空气中で運動する物体上の微小面に働く流体力 D を式(6)で計算する。

$$D = \frac{1}{2} \rho A_p C_d v^2 \cos \theta \quad (6)$$

ここに、 ρ は空気密度、 A_p は微小面の面積、 C_d は係数、 $\mathbf{v}$ は微小面の重心速度、 θ は微小面の外向き法線ベクトル $\mathbf{n}$ と $\mathbf{v}$ のなす角である。流体から受ける力は物体内向きに働くため、 D は微小面の重心位置に $-\mathbf{n}$ と平行に働く。

式(6)をすべての微小面について求めて式(5)の外力項に加えることで、空气中で運動する剛体の運動を近似的に計算できる。物理エンジンと式(6)を組み合わせた具体的な処理手順は文献¹⁸⁾で参照できる。この近似的計算方法を羽ばたき翼に適用した場合、大型鳥類における翼特有の揚抗比の高さや失速現象を表現

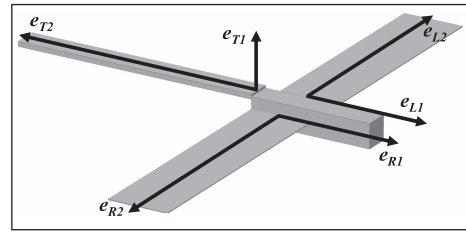


Fig.2 The control subject: it consists of a body, two wings and a tail

できない。しかし羽ばたきによる飛翔が可能であれば本実験の目的は満たされるため、近似による誤差は許容可能である。

4.3 粒子群最適化による最適化

EANNでよく用いられる最適化手法は遺伝的アルゴリズムである。しかし遺伝的アルゴリズムの性能は複数のパラメータに左右されるため、最適なパラメータの組を発見するためには膨大な予備実験が必要である。そこで本実験では、EANNの最適化手法として粒子群最適化(Particle swarm optimization, PSO)¹⁹⁾を用いる。PSOは群知能の一種であり、EANNの最適化にも用いられている²⁰⁾。また、最も単純なPSOでは3つのパラメータのうち2つの値を実質的に固定できるため²¹⁾、適切なパラメータを実験的に発見しやすいという大きな利点がある。

PSOでは解ベクトルを粒子とみなし、複数の粒子が多次元解空間を移動しながら探索を行う。各粒子は現在までに自身が発見した最良評価を与える解 $\mathbf{p}_d$ を持つ。また群全体が現在までに発見した最良評価を与える解 $\mathbf{p}_g$ を全粒子が共有する。探索ステップ k における粒子の解を $\mathbf{x}_k$ 、速度を $\mathbf{v}_k$ とおくと、探索ステップ $k+1$ における粒子の解 $\mathbf{x}_{k+1}$ 、速度 $\mathbf{v}_{k+1}$ は式(7)により更新される。

$$\begin{aligned} \mathbf{v}_{k+1} &= w \mathbf{v}_k + c_1 r_1 (\mathbf{p}_d - \mathbf{x}_k) + c_2 r_2 (\mathbf{p}_g - \mathbf{x}_k) \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{v}_{k+1} \end{aligned} \quad (7)$$

ここに、 r_1, r_2 は $[0, 1]$ の一樣乱数、 w は慣性係数である。 c_1, c_2 はそれぞれ $\mathbf{p}_d, \mathbf{p}_g$ に近づくための係数であり、経験的に $c_1 = c_2 = 2$ とすればよいことが知られている²¹⁾。

4.4 実験方法

4.4.1 制御対象

制御対象は図2に示す構造物である。羽ばたき飛翔を行う鳥類や昆虫の一部の構造を一般化したものであり、胴体部に2枚の翼および尾が接続されている。左右の翼は軸 $\mathbf{e}_{L1}, \mathbf{e}_{R1}$ 回りの羽ばたき動作と、軸 $\mathbf{e}_{L2}, \mathbf{e}_{R2}$ 回りのひねり動作が可能である。同様に尾も軸 $\mathbf{e}_{T1}$ 回り、軸 $\mathbf{e}_{T2}$ 回りの2軸に関して動作が可能である。翼および尾の関節の可動範囲は軸 $\mathbf{e}_{L1}, \mathbf{e}_{R1}$ 回りが $\pm 80^\circ$ 、軸 $\mathbf{e}_{L2}, \mathbf{e}_{R2}$ 回りが $\pm 40^\circ$ 、軸 $\mathbf{e}_{T1}$ 回りが $\pm 20^\circ$ 、軸 $\mathbf{e}_{T2}$ 回りが $\pm 40^\circ$ である。制御対象の動力学、すなわち式(2)における状態変換写像 λ は物理エンジンと式(6)の流体力により与えられ、数値積分のタイムステップを1/100秒として状態変化が計算される。

4.4.2 制御方法

制御対象は提案するCANNあるいは一般的なANNで制御され、3次元空間上に設置された目標点へ向かって飛翔を行う。ここで羽ばたき飛翔に関する一般的な知見から、左右の翼は対称的な動作を行う。このとき、翼の運動は前進や上昇・下降に寄与し、尾の運動は左右への旋回やロール運動に寄与する。制御信号ベクトルは軸 $\mathbf{e}_{L1}, \mathbf{e}_{R1}$ 回りの角度、軸 $\mathbf{e}_{L2}, \mathbf{e}_{R2}$ 回りの角度、軸 $\mathbf{e}_{T1}$ 回りの角度、軸 $\mathbf{e}_{T2}$ 回りの角度からなる4次元ベクトルである。また、CANNあるいはANNに入力する状態ベクトルは、軸 $\mathbf{e}_{L1}$

回りの翼の角度と回転速度, 軸 e_{L2} 回りの翼の角度, 制御対象のピッチ角とロール角, 前進速度, 目標地点に対する垂直・水平方向の相対角度からなる 8 次元ベクトルである。軸 e_{L1} 回りの回転速度は, フィードフォワード型 ANN で周期的な羽ばたきを生成するために用いられる。

性能を比較するために用いるのは, 低次 ANN の個数が 2 個の CANN (CANN 2) と 5 個の CANN (CANN 5), および一般的な ANN の 3 種類である。高次 ANN, 低次 ANN および一般的な ANN として 3 層フィードフォワード型 ANN を用いる。3 層フィードフォワード型 ANN は中間層ニューロン数が多いほど非線形関数の表現能力が高くなるが, その分結合重みの総数が増えるため学習に長い時間を要する。そこで, 設定した探索ステップを考慮して中間層ニューロン数は 20 とする。また, 伝達関数として式(4)の拡張シグモイド関数を用いる。ただし出力の幅は $[-1, 1]$ であるため, 制御信号を各関節の可動範囲にスケールングして角度制御を行う。

4.4.3 評価方法

制御対象は y 軸が鉛直上向きとなる 3 次元空間内の座標 $(0, 100, 0)$ に, z 軸が前向きとなるように初期配置される。制御対象が向かう目標点 p_g は座標 $(100, 150, 100)$ であり, 制御対象の初期位置から見て左斜め上に位置する。制御対象の胴体の重心位置を p_b としたとき, 式(2)における評価関数 E を式(8)で与える。

$$E = \sum_{t=1}^N e(t), \quad e(t) = \begin{cases} \frac{10}{10 + |p_g - p_b(t)|} & : t < T \\ 1 & : t \geq T \end{cases} \quad (8)$$

ここに, $e(t)$ はタイムステップ t で加算される評価, T は制御対象が目標点の周囲 1m 以内に最初に到達したタイムステップである。目標点到達までの行動のみを評価するため, 目標点到達の前後で評価が異なっている。評価する状態数 N を 2000 とし, 20 秒間の評価の総和を最大化するように学習を行う。評価が全ての状態についての総和であるため, この評価に対して期待される最適行動は最短時間で目標点に到達する行動となる。

4.4.4 学習方法

PSO のパラメータを探索ステップ 1000, 粒子数 20, $c_1 = c_2 = 2$ とした。はじめに予備実験として, 文献²¹⁾を参考に w を 0.9 から 0.4 まで線形に減少させる設定で一般的な ANN を学習させた。また同じく予備実験として, w を固定する設定で一般的な ANN を学習させた。固定した w の値は 0.1, 0.2, ..., 0.9 の 9 通りである。合計 10 通りの w について各 10 回の数値計算実験を行った結果, $w = 0.4$ に固定した場合が最も良い平均評価を得た。 w が 0.3 以下の場合探索ステップの初期で低い評価に収束し, w が 0.5 以上の場合および w を変化させた場合は評価の増加が遅く, 探索ステップ 1000 では不足であった。この予備実験より, 実際の数値計算実験では $w = 0.4$ に固定する設定を用いた。

一般的な ANN の学習は, 予備実験で最も良い平均評価を得た探索ステップ 1000, $w = 0.4$ の設定で行う。一方 CANN は, はじめに低次 ANN の学習をそれぞれ独立に行い, その後高次 ANN の学習を行う必要がある。本実験では一般的な ANN と CANN の学習で動力学計算に要する計算量を等しくして性能を比較するため, 複数の低次 ANN および高次 ANN の学習に要する探索ステップの総和を 1000 とする。ここで 3.3 節より, 本来であれば低次 ANN の個数によって高次 ANN の学習のための探索ステップを変化させることが望ましい。ただし本実験では, 3.3 節の

設計指針が有効であるかを確認するため, 高次 ANN の学習の探索ステップは CANN 2, CANN 5 とともに 400 探索ステップとする。具体的には, CANN 2 の場合, 2 個の低次 ANN の学習をそれぞれ 300 探索ステップずつ行い, 高次 ANN の学習を 400 探索ステップ行う。同様に, CANN 5 の場合, 5 個の低次 ANN の学習をそれぞれ 120 探索ステップずつ行い, 高次 ANN の学習を 400 探索ステップ行う。CANN の学習では最適な w の値についての予備実験は行わず, 一般的な ANN の学習と同様に $w = 0.4$ を用いる。一般的な ANN および 2 種類の CANN について独立な数値計算実験を各 100 回ずつ行い, 統計的に結果を比較する。

4.5 実験結果

数値計算実験を行った結果, 一般的な ANN と 2 種類の CANN に関わらずほぼ全ての場合で, 周期的な羽ばたきを行う飛翔行動が得られた。ただし, 羽ばたきの周期や振幅, 目標点に到達できたかどうかに関してばらつきが大きい結果となった。個々の羽ばたきに関するデータを示すことは冗長であり, その解析は本実験の目的に寄与しないため省略する。

一般的な ANN および 2 種類の CANN について, 探索ステップごとの評価の推移がそれぞれ 100 個ずつ得られた。各探索ステップについて 100 個の評価を平均し, 横軸を探索ステップ, 縦軸を評価としたグラフが図 3 である。ただし CANN の場合は, 2 個あるいは 5 個の低次 ANN の評価を順に並べた後, 600 探索ステップ以降に高次 ANN の評価を示している。CANN の低次 ANN の学習はそれぞれ独立に行ったため, 平均の計算を行う際の低次 ANN の並べ方はランダムとした。学習が終了した 1000 探索ステップでの平均評価は, 一般的な ANN, CANN 2, CANN 5 の場合でそれぞれ 1094.14, 1146.84, 1164.81 であった。

CANN 5 の場合, 100 回の数値計算実験において合計 500 個の学習済み低次 ANN が得られた。それらの評価を昇順に並べたグラフが図 4 である。また図 5 は, 5 個の低次 ANN のうち最大の評価を与える低次 ANN 1, CANN 5, および一般的な ANN の学習後の評価を昇順に並べたグラフである。図 6 は, 100 回の数値計算実験で得られた各 CANN 5 について, 低次 ANN 1 の評価に対する CANN 5 の評価の増加率を求め, 昇順に並べたグラフである。図 6 の増加率の最大値, 平均値, 最小値はそれぞれ 4.93707, 1.75738, 1.00044 であった。なお CANN 2 の場合, 学習が終了した CANN 2 の評価は CANN 5 とほぼ一致したが, 低次 ANN 1 の評価はやや高く, 評価の増加率はやや低い傾向が見られた。

CANN 5 の 100 回の数値計算実験のうち, 多くの場合で 2 個あるいは 3 個といった少数の低次 ANN のみを用いられた。また比較的長時間同一の低次 ANN が選択され続けるパターンと, 選択が細かく切り替わるパターンの混在が多く見られた。そこで CANN の評価の増加率が高い事例のうち, 上記の特徴が最も顕著なものを代表的事例として挙げる。この事例では低次 ANN 1 の評価は 463.08 であり, 5 個の低次 ANN のいずれを用いても目標点に到達できない。しかし高次 ANN の学習の結果, 3 番目に高い評価 444.56 を与える低次 ANN 3 とアンサンブルを行うことにより, 7.75 秒で目標点に到達可能となった。このときの CANN 5 の評価は 1331.72 であり, 増加率は 2.87581 であった。図 7 は横軸を飛翔の経過時間, 縦軸を高次 ANN によって選択された低次 ANN としたグラフである。また図 8 は, 低次 ANN 1, 低次 ANN 3, および CANN 5 によって制御された飛翔の軌跡を, それぞれ細線, 太線, 点線で示した図である。細線は大きく上昇した後に下降する飛翔軌跡, 太線は左に蛇行しながらゆっくりと上昇する飛翔軌跡である。

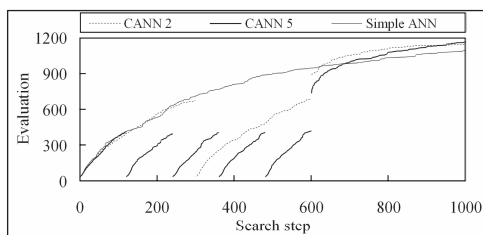


Fig.3 Average evaluation in one hundred trials

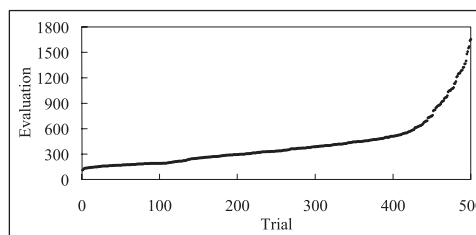


Fig.4 Evaluation of low-level ANNs learned for 120 steps

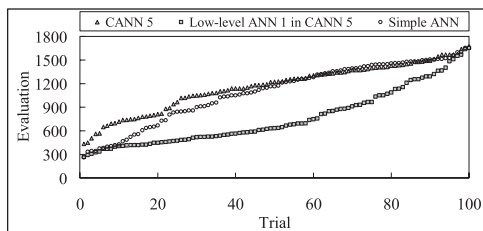


Fig.5 Final evaluation of each controller in one hundred trials

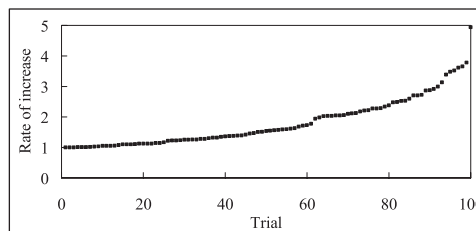


Fig.6 Rate of increase from only low-level ANNs to whole CANN 5

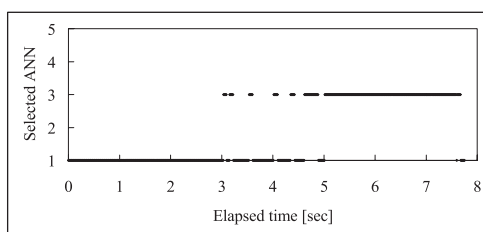


Fig.7 Selected low-level ANNs by a high-level ANN in a typical CANN 5

4.6 考 察

図3より、高次ANNの学習が始まった直後の600探索ステップでは、CANNの平均評価は一般的なANNと比較して低い。ここで、600探索ステップでのCANNの平均評価が2個または5個の低次ANNの平均評価より高いのは、学習後の低次ANNの評価にばらつきがあるためである。高次ANNの学習開始直後には、2個または5個の低次ANNのうち最大の評価を与える低次ANN 1を長く選択し続けるパターンが高い評価を得る。これは3.3節で示したとおりの現象である。一方、学習が終了した1000探索ステップでの平均評価はCANN 5、CANN 2、一般的なANNの順であり、600探索ステップでの順序が逆転している。これはCANN 2よりもCANN 5の方がより多くの相異なる保証状態を持つためであり、選択によるアンサンブル方法の有効性を示す。ただし低次ANNがわずか2個のCANN 2でさえ、同一計算量の学習において一般的なANNよりも優れた性能を示した。CANNでは高次ANNと各低次ANNの探索ステップ数が少ないにも関わらず、PSOのパラメータ w は一般的なANNの学習により調整された値を用いている。よってパラメータ w を探索ステップ数が少ない学習で調整すれば、CANNの評価はさらに向上すると期待される。

2章のアンサンブル方法の第二の問題において、独立した学習により保証状態の類似性が低い低次ANNが得られると仮定した。保証状態の比較は行っておらず完全な証明ではないが、図4より、少なくとも全ての低次ANNの評価が類似する確率は極めて低い。実際のデータでも、低次ANN間で評価の差が全て100以下となる場合はCANN 2で16%、CANN 5で0%であった。よ

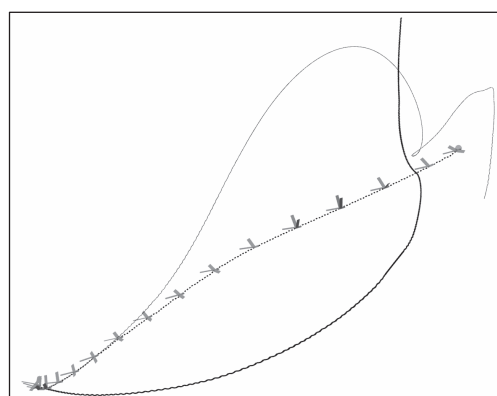


Fig.8 Trajectory: the thin line is given by low-level ANN 1, the thick is given by low-level ANN 3 and the dotted is given by CANN 5

って、組み合わせる低次ANNの個数が増加するほど、独立した学習により類似性が低い低次ANNを得る方法の有効になるといえる。

図4と同様に、図5にも解の局所収束性が現れている。低次ANN 1と一般的なANNは、いずれも評価分布の最大値と最小値が等しいが、一般的なANNの方が全体的に高い評価域に分布している。ところが学習が終了したCANN 5では、特に評価が1200以下の領域において一般的なANNよりも高い評価となった。一方、評価が1200以上の領域では一般的なANNと同等、あるいはわずかに低い評価である。これは、アンサンブルによって組み合わせた5個の低次ANNが十分な学習を行ったものではないためと考えられる。しかし図6より、十分な学習を行っていない低次ANN群を組み合わせることで評価は約1.76倍増加した。この結果から、平均的に評価が低いANNでも選択型のアンサンブルを行うことで評価が向上し得るといえる。

代表的な事例として挙げた図7の選択例は、状態ごとに低次ANNを選択して用いるアンサンブル方法が有効に機能することを示す。経過時間3秒から5秒にかけて2つの低次ANNが交互に選択されているのは、それぞれが有効に機能する状態間の状態遷移を表していると考えられる。ただし図8より、低次ANN 3による飛翔軌跡とCANNによる飛翔軌跡には重複部分が存在

せず、低次 ANN 3 が選択されている状態は低次 ANN 3 の保証状態ではない。これは学習後の低次 ANN の制御能力が、学習データの局所的分布から推測されるよりも高いことを示す。全ての飛翔パターンから定量的評価を行うことは困難であり断言はできないが、こうした低次 ANN の能力の高さは、CANN の多くの場合で少数の低次 ANN のみが用いられたこととも関連があると考えられる。保証状態以外の状態に対してどのような応答を行うかは CANN に限らず EANN 全般に関わる重要な問題であり、今後明らかにすべきである。

以上の考察より、CANN の有効性として 2 つの利点が示された。1 つは、独立な学習を複数行うことにより、性能の異なる低次 ANN を得られる点である。図 4 に示されるように、1 回の学習における評価は特定の分布に従う。しかし、図 5 に示されるように、学習を複数行ったときの最大評価は、1 回の学習における評価の分布よりも平均評価が高い分布に従う。これは解の局所収束による影響を回避するのに有効であり、一般的な ANN と比較して、CANN は評価が極めて低くなる事例が少ない。この利点は、EANN の教師なし学習が確率的なアルゴリズムにより行われることに起因するものであり、他の制御対象に対して適用する場合にも同様の結果が期待される。2 つ目の利点は、選択型のアンサンブルによる性能の向上である。CANN では、組み合わせる低次 ANN のうち最大の性能を持つものを基に学習が進むため、CANN 全体の性能は低次 ANN のみの性能より低くなることはない。これは高次 ANN の学習の特徴であり、制御対象に依存しない。一方、低次 ANN を複数組み合わせることで性能がどの程度増加するか、あるいは図 7 に示されたような滑らかな切り替えが必ず現れるかについては、制御対象によって異なる可能性がある。羽ばたき飛翔とは異なる特性を持つ制御対象に CANN を適用した場合も、本実験と同様の性能向上が見られるかを検証する必要がある。

5. 結 言

本論文では、制御問題における EANN の教師なし学習で効果的なアンサンブルの実現を目的とし、以下の結果を得た。

- (1) 制御問題における EANN の教師なし学習を数理的に記述し、その問題点として学習データの局所的分布、解の局所収束、許容可能解の非一意性を導いた。
- (2) EANN に対して効果的なアンサンブルとして、独立に学習した低次 ANN 群を選択型アンサンブルで組み合わせ、その選択に高次 ANN を用いる CANN を提案した。
- (3) 羽ばたき飛翔の制御で、CANN は同一計算量の学習において従来の EANN よりも平均 6.46%優れた評価を得た。
- (4) 低次 ANN が 5 個の CANN 5 では、CANN を用いることで低次 ANN 群のみの評価が平均 75.74%向上した。
- (5) CANN による制御の代表的事例から、選択型のアンサンブル方法が有効に機能することを示した。

数値計算実験によって CANN の有効性が示された一方で、低

次 ANN が保証状態以外の状態に対しても適切な応答が可能である結果も得られた。この結果は、保証状態のみならず、適切な応答が可能な全ての状態の数を基準に低次 ANN 群を得る必要があることを示す。よって CANN の効果をより高めるためには、より多くの状態に対して最適応答が可能な ANN の学習方法、および低次 ANN 間の類似性を最小化する方法が必要である。

なお、本研究の一部は JSPS 科研費 24・2124 の助成を受けて行われたものであり、ここに謝意を表する。

参 考 文 献

- 1) R.A. Nichols, R.T. Reichert and W.J. Rugh: Gain scheduling for H-infinity controllers: A flight control example, *IEEE Trans. Control Systems Technology*, **1**, 2, (1993) 69.
- 2) J.W. Helton and M.R. James: Extending H-infinity Control to Nonlinear Systems, SIAM, Philadelphia, (1999).
- 3) A.J. van der Schaft: L_2 -gain and passivity techniques in nonlinear control, Springer, Berlin, (1996).
- 4) C. Phillips, C.L. Karr and G. Walker: Helicopter flight control with fuzzy logic and genetic algorithms, *Engineering Applications of Artificial Intelligence*, **9**, 2, (1996) 175.
- 5) X. Yao: Evolving Artificial Neural Networks, *Proc. IEEE*, **87**, 9, (1999) 1423.
- 6) R. Ghorbani, Q. Wu and G.G. Wang: Nearly optimal neural network stabilization of bipedal standing using genetic algorithm, *Engineering Applications of Artificial Intelligence*, **20**, 4, (2007) 473.
- 7) L. Breiman: Bagging predictors, *Machine Learning*, **24**, 2 (1996) 123.
- 8) R.E. Schapire: The strength of weak learnability, *Machine Learning* **5**, 2, (1990) 197.
- 9) Y. Freund and R.E. Schapire: A decision-theoretic generalization of on-line learning and an application to boosting, *Proc. Second European Conf. Computational Learning Theory*, (1995) 23.
- 10) L.K. Hansen and P. Salamon: Neural network ensembles, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **12**, 10, (1990) 993.
- 11) D.W. Opitz and J.W. Shavlik: Actively searching for an effective neural network ensemble, *Connection Science*, **8**, 3, (1996) 337.
- 12) M.P. Perrone and L.N. Cooper: When networks disagree: ensemble method for neural networks, *Artificial Neural Networks for Speech and Vision*, Chapman & Hall, New York, (1993) 126.
- 13) B. Sareni and L. Krahenbuhl: Fitness sharing and niching methods revisited, *IEEE Trans. Evolutionary Computation*, **2**, 3, (1998) 97.
- 14) F. Gomez: Sustaining diversity using behavioral information distance, *Proc. 11th Ann. Conf. Genetic and Evolutionary Computation*, (2009) 113.
- 15) R.A. Brooks: A Robot that Walks; Emergent Behaviors from a Carefully Evolved Network, *Neural Computation*, **1**, 2, (1989) 253.
- 16) D. Baraff: Analytical methods for dynamic simulation of non-penetrating rigid bodies, *Proc. 16th Ann. Conf. on Computer Graphics and Interactive Techniques*, **23**, 3, (1989) 223.
- 17) D.E. Stewart and J.C. Trinkle: An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and Coulomb friction, *Int. J. Numerical Methods in Engineering*, New Jersey, **39**, 15 (1996) 2673.
- 18) R. Ooe, I. Suzuki, M. Yamamoto and M. Furukawa: Physical Simulation Environment for Motion Computation in Fluid —Deciding a Shape of da Vinci's Spiral Wing by Particle Swarm Optimization—, *J. Jpn. Soc. Precis. Eng.*, **78**, 4, (2012) 348 (in Japanese).
- 19) J. Kennedy and R. Eberhart: Particle Swarm Optimization, *Proc. IEEE Int. Conf. Neural Networks*, **4**, (1995) 1942.
- 20) R.C. Eberhart and X. Hu: Human tremor analysis using particle swarm optimization, *Proc. Congress on Evolutionary Computation*, **3**, (1999) 1927.
- 21) R.C. Eberhart and Y. Shi: Particle swarm optimization: developments, applications and resources, *Proc. 2001 Congress on Evolutionary Computation*, **1**, (2001) 81.
- 22) K. Yoneda, I. Suzuki, M. Yamamoto and M. Furukawa: Behavioral Acquisition for Elastic Circular Robot by using Composite Behavior, *J. Jpn. Soc. Precis. Eng.*, **77**, 5, (2011) 476 (in Japanese).