

複素バックプロパゲーション学習アルゴリズムの学習特性

新田 徹† 古谷 立美†

複素パターンに対する学習アルゴリズムとして提案された複素 BP は、その学習パラメータがすべて複素数となっており、実数値を用いる通常のバックプロパゲーション学習アルゴリズム (BP) を複素数に拡張したものである。本稿では、複素 BP の学習特性を調べた結果、次の特性が明らかになったので報告する。(i) 複素 BP における誤差逆伝播は 2 次元運動に基づいた構造を有している。(ii) 複素 BP は、ネットワークを流れる複素信号を 1 つの処理単位として学習を進めるものである。(iii) 複素 BP では、学習則における実部要因と虚部要因の補完構造により、学習停滞状態の発生が抑えられる。その結果、通常の BP に比べて、複素パターンに対する学習が数倍速くなっている。しかも、そのときに必要となる学習パラメータ数は約半分で済む。これらの意味で、複素 BP は複素パターンの学習に適したアルゴリズムであると言える。また、これらの諸特性は意図されたものではなく、ニューラルネットワークに複素数という数学的構造を導入したことにより自ずと現れてきたものである。複素 BP の基本特性に関しては、重みパラメータの 2 次元運動構造、決定表面の直交性および 2 次元運動学習能力 (図形変換能力) などが既に報告されている。これらは主としてネットワークアーキテクチャからのアプローチにより得られた結果であるのに対し、本稿の結果は学習アルゴリズムの観点から得られたものである。

Characteristics of Learning in the Complex Back-propagation Learning Algorithm

TOHRU NITTA† and TATSUMI FURUYA†

The purpose of this paper is to investigate the characteristics of Complex BP which has been already proposed as a learning algorithm for complex patterns. We obtained the following results. (1) Error back propagation has the structure concerned with two dimensional movement. (2) The unit of learning process is a complex signal flowing in the neural network. (3) The learning rule contains the structure to evade a standstill in learning. In result, learning speed is improved. In addition, the number of parameters (weights and thresholds) needed is almost half that of the standard BP. Thus it seems that Complex BP is an algorithm which is fit for the learning of complex patterns.

1. はじめに

近年、高度な並列分散型の情報処理を特徴とするニューラルネットワークが注目を集めている²⁾。ニューラルネットワークには、大きく分けて、信号の流れが一方向である階層型と、信号がネットワークの中を何度も流れることができる相互結合型が考えられている。層型ニューラルネットワークの学習方式として提案されたバックプロパゲーション学習アルゴリズム¹⁴⁾ (以下、実 BP と略す) は比較的単純なアルゴリズムであり、音声認識や画像処理を含めた様々な分野に応用されつつある^{2), 4)}。

複素バックプロパゲーション学習アルゴリズム⁶⁾ (以下、複素 BP と略す) は、実数値を用いる通常の

実 BP を複素数へ拡張したものであり、複素パターンに対する学習を想定しているものである。複素 BP の重みパラメータには 2 次元運動に関係した制約が課せられており、学習はその制約を常に保ちながら進められる^{7), 9)}。また、その決定表面は基本的に 2 つの互いに直交した超平面から構成されている^{7), 9)}。複素 BP に特徴的なことは、実 BP には見られない 2 次元運動学習能力 (図形変換能力) が備わっていることである^{6), 8), 11)}。

本論文では、学習アルゴリズムの観点から、複素 BP の基本特性を調べた結果、次に示す特性を持つことが明らかになったので報告する。

(i) 複素 BP における誤差逆伝播は (線形代数でいうところの) 2 次元運動に基づいた構造を有している。

(ii) 複素 BP は、ネットワークを流れる複素信号を 1 つの処理単位として学習を進めるものである。

† 電子技術総合研究所情報アーキテクチャ部計算機構研究室
Computational Models Section, Computer Science Division, Electrotechnical Laboratory

(iii) 複素 BP では、学習則における実部要因と虚部要因の補完構造により、学習停滞状態の発生が押えられる。その結果、実 BP に比べて、複素パターンに対する学習が数倍速くなっている（実 BP に複素パターンを学習させる方法については後述する）。しかも、そのときに必要となる領域計算量（重みおよびしきい値の総数）は約半分まで済む。

これらの意味で、複素 BP は複素パターンの学習に適したアルゴリズムであると言える。また、これらの諸特性は意図されたものではなく、ニューラルネットワークに複素数という数学的構造を導入したことにより自ずと現れてきたものであることに注意されたい。

以下では、まず、第2章において、複素 BP の説明を簡単に行う。第3章では、複素 BP の解析を行い、学習の高速性の要因などの基本特性について調べる。それらの2つの例題を用いた計算機実験の結果を第4章で示し、結論と今後の課題については第5章において述べる。

2. 複素 BP の概要

複素 BP⁶⁾について簡単に説明する。ネットワークは多層型であり、複素パターンが入力され、複素パターンが出力されることを想定している。ニューロン間の結合の重みおよび各ニューロンが持つしきい値はすべて複素数であり、各ニューロンの出力関数 $f_C: C^1 \rightarrow C^1$ は、

$$f_C(z) = f_R(x) + if_R(y), \quad z = x + iy, \quad (1)$$

ただし、 $f_R(u) = 1/(1 + \exp(-u))$ である (C は複素数全体の集合)。すなわち、ニューロンの出力値の実部および虚部は、それぞれ、ニューロンへの入力 z の実部 x 、虚部 y に関するシグモイド関数値である。また、学習アルゴリズムは基本的に教師パターンとの誤差を最小(極小)ならしめる方向に学習パラメータ(重みおよびしきい値)の修正を行う最急降下法である。学習収束性については、実 BP の理論的基礎である学習識別理論⁹⁾を複素数に拡張することによって確認されている。

3. 複素 BP の学習特性

本章では、学習アルゴリズムの観点から、学習特性についての解析を行う。

3.1 学習パラメータの修正量の構造

本節では、複素 BP における学習パラメータの修正量の構造を3層ニューラルネットワークを例にとって

調べる。まず、用いる記号を定義しておく。 w_{ji} を入力ニューロン i と中間ニューロン j との間の結合の重み、 v_{kj} を中間ニューロン j と出力ニューロン k との間の結合の重み、 θ_j を中間ニューロン j が持つしきい値、 γ_k を出力ニューロン k が持つしきい値とする。また、 I_i, H_j, O_k を、それぞれ、入力ニューロン i , 中間ニューロン j , 出力ニューロン k の出力値とする(これらがネットワークを流れる信号である)。そして、出力ニューロン k に対する教師パターン T_k と O_k との誤差を $\delta^k = T_k - O_k$ とする。これらはすべて複素数である。また、 $\varepsilon > 0$ を学習率とする。

学習パラメータ x の修正量を Δx 、その実部を Δx^R 、虚部を Δx^I で表すことにすると、各学習パラメータは次のように表現できる。

$$\begin{aligned} \begin{bmatrix} \Delta v_{kj}^R \\ \Delta v_{kj}^I \end{bmatrix} &= \begin{bmatrix} \text{Re}[H_j] & \text{Im}[H_j] \\ -\text{Im}[H_j] & \text{Re}[H_j] \end{bmatrix} \begin{bmatrix} \Delta \gamma_k^R \\ \Delta \gamma_k^I \end{bmatrix} \\ &= |H_j| \begin{bmatrix} \cos \beta_j & \sin \beta_j \\ -\sin \beta_j & \cos \beta_j \end{bmatrix} \begin{bmatrix} \Delta \gamma_k^R \\ \Delta \gamma_k^I \end{bmatrix}, \quad (2) \end{aligned}$$

$$\begin{bmatrix} \Delta \gamma_k^R \\ \Delta \gamma_k^I \end{bmatrix} = \varepsilon \begin{bmatrix} A_k & 0 \\ 0 & B_k \end{bmatrix} \begin{bmatrix} \text{Re}[\delta^k] \\ \text{Im}[\delta^k] \end{bmatrix}, \quad (3)$$

$$\begin{aligned} \begin{bmatrix} \Delta w_{ji}^R \\ \Delta w_{ji}^I \end{bmatrix} &= \begin{bmatrix} \text{Re}[I_i] & \text{Im}[I_i] \\ -\text{Im}[I_i] & \text{Re}[I_i] \end{bmatrix} \begin{bmatrix} \Delta \theta_j^R \\ \Delta \theta_j^I \end{bmatrix} \\ &= |I_i| \begin{bmatrix} \cos \phi_i & \sin \phi_i \\ -\sin \phi_i & \cos \phi_i \end{bmatrix} \begin{bmatrix} \Delta \theta_j^R \\ \Delta \theta_j^I \end{bmatrix}, \quad (4) \end{aligned}$$

$$\begin{aligned} \begin{bmatrix} \Delta \theta_j^R \\ \Delta \theta_j^I \end{bmatrix} &= \begin{bmatrix} C_j & 0 \\ 0 & D_j \end{bmatrix} \sum_k \begin{bmatrix} \text{Re}[v_{kj}] & \text{Im}[v_{kj}] \\ -\text{Im}[v_{kj}] & \text{Re}[v_{kj}] \end{bmatrix} \begin{bmatrix} \Delta \gamma_k^R \\ \Delta \gamma_k^I \end{bmatrix} \\ &= \begin{bmatrix} C_j & 0 \\ 0 & D_j \end{bmatrix} \sum_k |v_{kj}| \begin{bmatrix} \cos \varphi_{kj} & \sin \varphi_{kj} \\ -\sin \varphi_{kj} & \cos \varphi_{kj} \end{bmatrix} \\ &\quad \cdot \begin{bmatrix} \Delta \gamma_k^R \\ \Delta \gamma_k^I \end{bmatrix}, \quad (5) \end{aligned}$$

ただし、 $A_k = (1 - \text{Re}[O_k])\text{Re}[O_k]$ 、 $B_k = (1 - \text{Im}[O_k])\text{Im}[O_k]$ 、 $C_j = (1 - \text{Re}[H_j])\text{Re}[H_j]$ 、 $D_j = (1 - \text{Im}[H_j])\text{Im}[H_j]$ 、 $\beta_j = \arctan(\text{Im}[H_j]/\text{Re}[H_j])$ 、 $\phi_i = \arctan(\text{Im}[I_i]/\text{Re}[I_i])$ 、 $\varphi_{kj} = \arctan(\text{Im}[v_{kj}]/\text{Re}[v_{kj}])$ である ($\text{Re}[z]$ 、 $\text{Im}[z]$ は、それぞれ、複素数 z の実部、虚部)。

まず、 Δv_{kj} について見てみる。式(2)において、 $|H_j|$ は2次元平面上の点の原点との距離に関する縮小・拡大、 $\begin{bmatrix} \cos \beta_j & \sin \beta_j \\ -\sin \beta_j & \cos \beta_j \end{bmatrix}$ は原点を中心とした負の方向(時計回り)への β_j 度の回転を意味する。つまり、式(2)では2次元運動(ただし、平行移動は伴わない)と呼ばれる線形変換が行われていると言える。よって、学習パラメータの修正量(複素数)を2次元平面上の点として表現した場合、中間-出力層の重み

の修正量 ($\Delta v_{kj}^R, \Delta v_{kj}^I$) は、出力層のしきい値の修正量 ($\Delta \gamma_k^R, \Delta \gamma_k^I$) に上記の線形変換 (2次元運動) を施すことによって得られるというわけである (図1)。同様にして、中間層のしきい値の修正量 ($\Delta \theta_j^R, \Delta \theta_j^I$) は、出力層の各しきい値の修正量 ($\Delta \gamma_k^R, \Delta \gamma_k^I$) に中間層-出力層の重み v_{kj} に関する2次元運動を施すことによって得られる値に基づいて定まる (式(5))。また、入力層-中間層の重みの修正量 ($\Delta w_{ji}^R, \Delta w_{ji}^I$) は、中間層のしきい値の修正量 ($\Delta \theta_j^R, \Delta \theta_j^I$) に入力層の出力値 I_i に関する2次元運動を施すことによって得られる (式(4))。要するに、複素BPにおける誤差逆伝播は2次元運動に基づいた構造を有していると言える。

ここで、重みパラメータ自身には、2次元運動に関する制約が課せられており、学習はその制約を保ちながら進められる^{7),9)} (ただし、各ニューロンにおいてはシグモイド関数が適用されるので、各ニューロンの出力値に2次元運動に関する制約が厳密に反映されるのは、シグモイド関数が線形変換とみなせるとき、すなわち、ニューロンへの入力の総和 (複素数) の実部および虚部のそれぞれの絶対値が共に小さいときに限る)。よって、複素BPでは、学習パラメータ自身およびその修正量には2次元運動に関係した構造が備わっていると見える。ただし、学習パラメータ自身に課せられている制約は正の方向への回転変換

$$P = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \quad (6)$$

に関係しており、学習パラメータの修正量に課せられている制約は負の方向への回転変換

$$Q = \begin{bmatrix} \cos \beta & \sin \beta \\ -\sin \beta & \cos \beta \end{bmatrix} \quad (7)$$

に関係している。ここで、 P および Q は、共に、直交行列であり、

$$Q^{-1} = {}^tQ = \begin{bmatrix} \cos \beta & -\sin \beta \\ \sin \beta & \cos \beta \end{bmatrix} \quad (8)$$

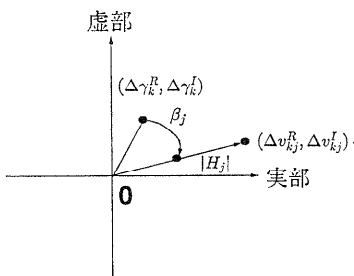


図1 複素BPにおける誤差逆伝播のイメージ
Fig. 1 An image of the error back propagation in complex BP.

を満たし、 Q の逆行列 Q^{-1} および転置行列 tQ は P と同様な形をしている。

ここで、一方の実BPにおける学習パラメータの修正量の構造について確認しておく。実BP¹⁴⁾における学習パラメータの修正量は

$$\Delta v_{kj} = H_j \Delta \gamma_k, \quad (9)$$

$$\Delta \gamma_k = \varepsilon (1 - O_k) O_k \delta^k, \quad (10)$$

$$\Delta w_{ji} = I_i \Delta \theta_j, \quad (11)$$

$$\Delta \theta_j = (1 - H_j) H_j \sum_k v_{kj} \Delta \gamma_k \quad (12)$$

で与えられる (ただし、各パラメータはすべて実数であり、その定義は複素BPのパラメータに対応する)。学習パラメータの修正量 (実数) を1次元直線上の点として表現した場合、中間-出力層の重みの修正量 Δv_{kj} は出力層のしきい値の修正量 $\Delta \gamma_k$ を原点との距離に関して H_j 倍することによって得られる (式(9)、図2)。同様にして、中間層のしきい値 $\Delta \theta_j$ は、出力層の各しきい値の修正量 $\Delta \gamma_k$ をそれぞれ原点との距離に関して v_{kj} 倍した値に基づいて定まる (式(12))。また、入力-中間層の重みの修正量 Δw_{ji} は、中間層のしきい値の修正量 $\Delta \theta_j$ を原点との距離に関して I_i 倍することによって得られる (式(11))。つまり、実BPにおける誤差逆伝播は1次元直線上の点の移動に基づいた構造をしていると言える。このように、実BPが複素数へ拡張されることによって、誤差逆伝播の構造が1次元的なものから2次元的なものへと変質していることがわかった。

また、見方を変えると、誤差逆伝播の2次元的な構造に伴い、複素BPはネットワークを流れる複素信号を1つの処理単位として学習を進めているということがわかる。たとえば、 Δv_{kj}^R および Δv_{kj}^I は、ネットワークを流れる信号の実部 ($\text{Re}[H_j]$, $\text{Re}[O_k]$) および虚部 ($\text{Im}[H_j]$, $\text{Im}[O_k]$) を共に含んでおり、これらに基づいて値が定まる (式(2))。つまり、 Δv_{kj}^R は複素信号の実部 ($\text{Re}[H_j]$, $\text{Re}[O_k]$) だけから、 Δv_{kj}^I は虚部 ($\text{Im}[H_j]$, $\text{Im}[O_k]$) だけから、という具合に互いに独立して決まるのではなく、ネットワークを流れる信号の実部および虚部の両方の値を通じて依存し合い

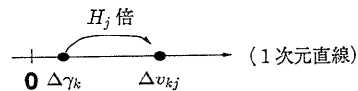


図2 実BPにおける誤差逆伝播のイメージ
Fig. 2 An image of the error back propagation in standard BP.

ながら定まるものであると言える。入力-中間層の結合の重み w_{ji} の実部の修正量 Δw_{ji}^R および虚部の修正量 Δw_{ji}^I 、中間層のしきい値 θ_j の実部の修正量 $\Delta \theta_j^R$ および虚部の修正量 $\Delta \theta_j^I$ についても同様に、ネットワークを流れる信号の実部、虚部の両方の値を含んでおり、それらを通して影響を与えている(式(4)および式(5))。また、出力層のしきい値 γ_k の実部の修正量 $\Delta \gamma_k^R$ は、直接的には、ネットワークを流れる信号(正確には、ネットワークが出力する信号) O_k の実部 ($Re[O_k]$) だけから定まり、虚部の修正量 $\Delta \gamma_k^I$ は虚部 ($Im[O_k]$) だけから決定される(式(3))。ここで、 $Re[O_k]$ についてみると、

$$Re[O_k] = f_R(Re[S_k]) \tag{13}$$

であり、

$$Re[S_k] = \sum_j \{ Re[v_{kj}] Re[H_j] - Im[v_{kj}] Im[H_j] \} + Re[\gamma_k] \tag{14}$$

であるから、 $Re[O_k]$ は $Re[H_j]$ と $Im[H_j]$ の関数である (S_k は出力ニューロン k への入力の総和)。一方、

$$Im[O_k] = f_R(Im[S_k]) \tag{15}$$

であり、

$$Im[S_k] = \sum_j \{ Re[v_{kj}] Im[H_j] + Im[v_{kj}] Re[H_j] \} + Im[\gamma_k] \tag{16}$$

であるので、 $Im[O_k]$ は、やはり、 $Re[H_j]$ と $Im[H_j]$ の関数である。ゆえに、 $\Delta \gamma_k^R$ と $\Delta \gamma_k^I$ は共に $Re[H_j]$ 、 $Im[H_j]$ の関数である。つまり、 $\Delta \gamma_k^R$ と $\Delta \gamma_k^I$ は、 $Re[H_j]$ と $Im[H_j]$ を通じて(間接的にはあるが)依存し合いながら値が定まると言える。

このように、複素BPにおける学習パラメータの実部および虚部の値の修正は、それぞれ、ネットワーク

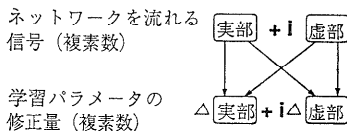


図3 学習パラメータの修正量の決定要因
矢印の始点は、矢印の終点の決定要因である。

Fig. 3 Factors to determine the amount of correction for learning parameters.
The starting point of the arrow is a determination factor of an end point.

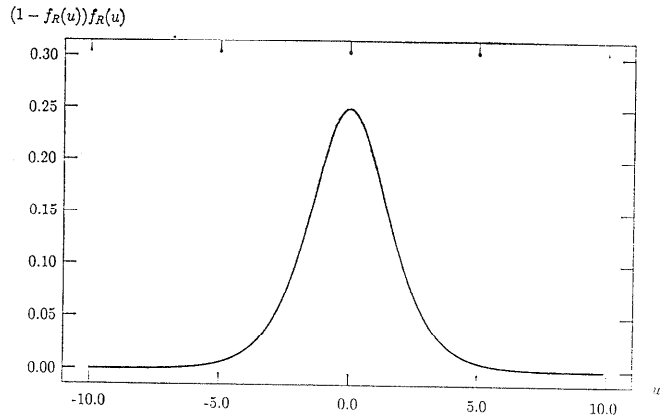


図4 シグモイド関数 $f_R(u)$ の導関数
Fig. 4 Derived function of the sigmoid function $f_R(u)$.

を流れる信号の実部および虚部の両方の値に基づいて、そして、それらを通じて相互に依存し合いながら行われる(図3)。要するに、複素BPは、ネットワークを流れる信号(複素数)を1つの処理単位として学習を進めるものであると言える。

3.2 学習の高速性の要因

前節において見たように、複素BPにおける誤差逆伝播は2次元運動に基づいた構造を有しており、その結果、複素BP学習はネットワークを流れる複素信号を1つの処理の単位としていた。本節では、さらに、このことがその構造において、学習の高速化を促すものであることを示す。

実BP¹⁴⁾の学習パラメータの修正量(式(9)~(12))において、 $(1-H_j)H_j \in \mathbf{R}$ および $(1-O_k)O_k \in \mathbf{R}$ は、各ニューロンの出力関数であるシグモイド関数 $f_R(u) = 1/(1 + \exp(-u))$ の導関数 $(1-f_R(u))f_R(u)$ であり、図4のような曲線を描くものである (\mathbf{R} は実数全体の集合)。つまり、ニューロンへの入力の総和 u の絶対値が大きくなるにつれて0に近づいてゆく関数である。よって、ニューロンの出力値を0もしくは1に少しでも近づけるために、ニューロンへの入力の総和 u の絶対値が大きくなるとシグモイド導関数 $(1-f_R(u))f_R(u)$ は微小な値を取ることになり、その結果、学習は進みにくくなる。その際、特に、誤差が大きな値であり、そのような状態が長く続く場合には、“ローカルミニマムに陥った”と称されることになる。よく知られているように、これが実BPにおける学習停滞のメカニズムである。

一方、複素BPにおいては、学習パラメータの修正

量(式(2)~(5))には2種類のシグモイド導関数が現れる。1つは出力関数の実部の導関数 $(1 - \text{Re}[O_k])\text{Re}[O_k]$ 、 $\text{Re}[O_k]$ 、 $(1 - \text{Re}[H_j])\text{Re}[H_j]$ 、もう1つは虚部の導関数 $(1 - \text{Im}[O_k])\text{Im}[O_k]$ 、 $(1 - \text{Im}[H_j])\text{Im}[H_j]$ であり、複素BPの学習則は基本的にこれらの2つの線形結合

$$\alpha_1(1 - \text{Re}[O_k])\text{Re}[O_k] + \beta_1(1 - \text{Im}[O_k])\text{Im}[O_k], \quad (17)$$

$$\alpha_2(1 - \text{Re}[H_j])\text{Re}[H_j] + \beta_2(1 - \text{Im}[H_j])\text{Im}[H_j] \quad (18)$$

から構成されている $(\alpha_k, \beta_k \in \mathbf{R} (k=1, 2))$ 。ここで、式(17)が微小な値になるのは、 $(1 - \text{Re}[O_k])\text{Re}[O_k]$ と $(1 - \text{Im}[O_k])\text{Im}[O_k]$ の両方の値が微小になるときである(式(18)についても同様)。よって、たとえば、 $(1 - \text{Re}[O_k])\text{Re}[O_k]$ が微小な値をとったとしても、 $(1 - \text{Im}[O_k])\text{Im}[O_k]$ もそうであるとは限らないので、式(17)が微小値をとらない可能性があり、この点で、 $(1 - O_k)O_k \in \mathbf{R}$ が微小値をとれば必ず学習パラメータの修正量が微小になってしまう実BPの学習則(式(9)~(12))とは異なっている。言い換えると、実部要因 $((1 - \text{Re}[O_k])\text{Re}[O_k], (1 - \text{Re}[H_j])\text{Re}[H_j])$ と虚部要因 $((1 - \text{Im}[O_k])\text{Im}[O_k], (1 - \text{Im}[H_j])\text{Im}[H_j])$ は、学習パラメータの修正量が異常に小さくならないようにお互いに補い合っているとも言える。このように、複素BPの学習則は、実BPに比べて、学習の停滞状態が発生しにくい構造を有していると言える(注意:特に、 $(1 - \text{Re}[O_k])\text{Re}[O_k]$ および $(1 - \text{Im}[O_k])\text{Im}[O_k]$ が共に微小な値になることが稀であるときに顕著であると思われる)。この結果、複素BPは、実BPに比べて、学習が高速化されているものと思われる。また、学習曲線の形についても興味のあるところである。これらのことについては、第4章において複素パターンの学習に関する計算機実験を通じて確認する。

従来、学習の高速化手法には、出力関数側からのアプローチと損失関数側からのアプローチが提案されており、それぞれ、効果をあげている。前者には、学習停滞状態に陥ったならば、シグモイド関数の傾きを強制的に修正することにより学習パラメータの修正量を増やすという手法¹⁵⁾、あるいは出力関数を区分的線形関数とし、その傾きを学習の対象として取り込む方法¹³⁾などがある。後者(損失関数側からのアプローチ)には、損失関数にKullback Divergenceを用いる方法⁵⁾、損失関数にエントロピー項を付加する方法¹¹⁾な

どがある。複素BPは、これらのように学習の高速化を意図されたものではなかったが、結果的に高速化を促進する構造が自然に備わっているのである。その構造は出力関数側からのアプローチに属するものであり、出力関数の実部と虚部の導関数の線形結合性に基づいている。よって、上述の従来の高速化手法のうち、損失関数側からのアプローチのものとは観点が異なっており、複素BPと組み合わせることで適用することが可能である。また、複素BPの出力関数の実部と虚部は、実BPの出力関数と同じシグモイド関数である(式(1)参照)。よって、上述の従来の高速化手法のうち、出力関数側からのアプローチのものを複素BPの出力関数の実部と虚部に対して、それぞれ、適用することにより、更なる高速化が可能であると思われる。

4. シミュレーション

本章では、複素パターンを使った計算機実験を通じて、3章において示した複素BPの学習特性の確認を行う。

一般に、学習速度の評価は単純に学習回数だけからではなく、計算量¹²⁾の観点からトータルに行う必要がある。計算量には、時間計算量(time complexity)と領域計算量(space complexity)の2種類がある。時間計算量はアルゴリズムの中の特定作業の実行回数のことであり、消費時間の目安にされるものである。一方、領域計算量はアルゴリズムの中で使用される変数の個数のことであり、消費記憶容量の目安にされるものである。

本稿では、アルゴリズムの中で行われる(実数の)四則演算の総回数を時間計算量、学習パラメータ数を領域計算量とする。ただし、時間計算量において、(1回の加算の計算量)と(1回の減算の計算量)は等しいものとする。また、一般に、(1回の乗算の計算量) \geq (1回の加算の計算量)であるが、本稿では念のため、(1回の加算の計算量) = (1回の乗算の計算量)および(1回の加算の計算量) = 0という極端な2つのケースを考える。なお、除算は実BPおよび複素BPでは使用されないで常に0回である。

次の4.1節および4.2節では、1回の学習の時間計算量が実BPと複素BPとで等しくなるようにネットワークの構造を調節したうえで、それぞれの収束するまでに必要な平均学習回数を基準にして学習速度の評価を行う。加えて、そのときの領域計算量も評価する。

実験において、学習パラメータの初期値は、本章を通して、 -0.3 から $+0.3$ の間の乱数により設定する。学習は誤差

$$\sqrt{\sum_{k=1}^N |T_k - O_k|^2} \quad (19)$$

が 0.10 となったときに収束（終了）したもののみなし、収束の規範とする（ただし、 $O_k, T_k \in \mathbf{C}$ は、それぞれ、出力ニューロン k の出力値およびその教師信号であり、 N は出力ニューロンの個数を表す）。また、学習パターンは、通常、数種類のパターンから構成されているが、その数種類のパターンをひと通り提示することで1回の学習と数えることとする。

4.1 例題 1

本節では、複素パターンから構成される単純な学習パターン（表1）を使った実験の結果を示す。

複素 BP には 1-3-1 ネットワーク、実 BP には 2-6-2 ネットワークおよび 2-7-2 ネットワークを使った。複素 BP および実 BP をこれらのネットワークに適用するときのそれぞれの計算量を表2に示す。学習1回あたりの時間計算量に関して、複素 BP 1-3-1 ネットワークは、（1回の加算の計算量）=0 のとき、

実 BP 2-6-2 ネットワークに等しく、（1回の加算の計算量）=（1回の乗算の計算量）のとき、実 BP 2-7-2 ネットワークにほぼ等しい。

実 BP に複素パターンを学習させる方法の1つは、入出力層において実部、虚部を担当させるニューロンをあらかじめ決めておくことである。ここでは、入力される複素数の実部を入出力ニューロン1、虚部を入出力ニューロン2に担当させた。

学習率をパラメータとして 0.1 から 0.6 まで 0.1 刻みで変化させた。各学習率について、それぞれ 50 回の試行を行い、それらの収束に要した学習回数の平均を評価の基準とした。学習は 5 万回で打ち切りとしたが、すべての試行は収束した。図5がその実験結果である。また、学習曲線（学習率は 0.1 ）の一例を図6に示す（実 BP は 2-7-2 ネットワーク）。

さらに、上述の実験における学習パラメータの修正量の絶対値の平均値を表3に示す。複素 BP、実 BP 共に学習率 0.5 の場合について、学習回数が 750 回の時点での値を調べた（ただし、パターン1に対応するデータに基づいている）。なお、実 BP ネットワークの代表としては 2-7-2 ネットワークを選んだ。複素 BP については、出力ニューロン1の出力関数の導関数値の実部 $(1 - \text{Re}[O_1])\text{Re}[O_1]$ が 0.10 以上、かつ、虚部 $(1 - \text{Im}[O_1])\text{Im}[O_1]$ が 0.05 以下であるような

表 1 学習パターン [例題 1]
Table 1 Learning patterns [Example 1].

パターン番号	入力パターン	出力パターン
1	0	0
2	i	1
3	1	$1+i$
4	$1+i$	i

表 2 複素 BP および実 BP の計算量 [例題 1]
時間計算量は 1 回の学習で行われる四則演算の総数、領域計算量は学習パラメータ数を意味する。

Table 2 Computational complexity of complex BP and standard BP [Example 1].
Time complexity means the sum of the four operations performed per learning cycle. Space complexity means the sum of the parameters (weights and thresholds).

ネットワーク	時間計算量			領域計算量		
	乗除算	加減算	計	重み	しきい値	計
複素 BP 1-3-1	78	52	130	12	8	20
実 BP 2-6-2	78	40	118	24	8	32
実 BP 2-7-2	90	46	136	28	9	37

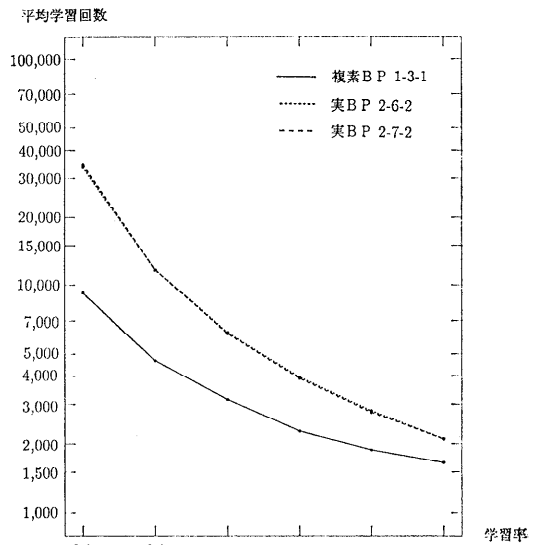


図 5 平均学習速度（複素 BP と実 BP の比較） [例題 1]
Fig. 5 Average of learning speed (a comparison between complex BP and standard BP) [Example 1].

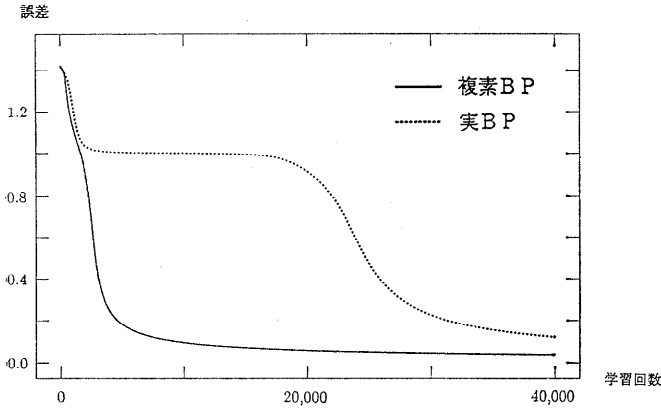


図 6 学習曲線 (複素 BP と実 BP の比較) [例題 1]

Fig. 6 Learning curve (a comparison between complex BP and standard BP) [Example 1].

表 3 学習パラメータの修正量の絶対値の平均値 [例題 1]

Table 3 Expectation of the absolute value of the amount of correction for the parameters (weights and thresholds) [Example 1].

ネットワーク	学習パラメータ	修正量の絶対値の平均値
複素 BP 1-3-1	$Re[w_{12}]$	0.00615
	$Im[w_{12}]$	0.00521
実 BP 2-7-2	w'_{12}	0.02330
	w'_{22}	0.00056

試行についての平均をとった。そのような試行は 50 回のうち 37 回あった。対象とした学習パラメータは中間ニューロン 2 と出力ニューロン 1 との間の重みパラメータ w_{12} である。実 BP については、出力ニューロン 1 の出力関数の導関数値 $(1-O_1)O_1$ が 0.10 以上、かつ、出力ニューロン 2 の出力関数の導関数値 $(1-O_2)O_2$ が 0.05 以下であるような試行についての平均をとった (O_k は出力ニューロン k の出力値を表す)。そのような試行は 50 回のうち 31 回あった。対象とした学習パラメータは中間ニューロン 2 と出力ニューロン 1 との間の重みパラメータ w'_{12} および中間ニューロン 2 と出力ニューロン 2 との間の重みパラメータ w'_{22} である。表 3 からわかるように、実 BP の場合、出力ニューロンの出力関数の導関数値が小さな値をとるとそれに対応する学習パラメータの修正量はそれに応じて小さな値をとってしまう。これに対して、複素 BP では出力関数の導関数値の実部、虚部のどちらか一方が小さな値をとった場合でも、それらに対応する学習パラメータの修正量は、(実 BP に比べて収束がある程度進んでいるため値はいくぶん小さいが) 実部、

虚部共に同程度の値をとっており、学習停滞状態は発生しにくくなっていることがうかがえる。

4.2 例題 2

次に、表 4 に示す学習パターンを使った実験を行った。この学習パターンは複素パターンから構成されており、次の 2 つの規則に従っている。

- (i) 入力される複素数 1 と複素数 2 が等しいならば、出力される複素数 3 の実部は 1、そうでなければ 0。
- (ii) 入力される複素数 2 が 1 または i ならば、出力される複素数 3 の虚部は 1、そうでなければ 0。

例題 1 と異なるのは、複素 BP には 2-4-1 ネットワーク、実 BP には 4-8-2 ネットワークおよび 4-9-2 ネットワークを使った点である。これら

表 4 学習パターン [例題 2]

Table 4 Learning patterns [Example 2].

パターン番号	入力パターン		出力パターン
	複素数 1	複素数 2	複素数 3
1	0	0	1
2	0	i	i
3	i	i	$1+i$
4	i	1	i
5	1	1	$1+i$
6	i	0	0
7	$1+i$	$1+i$	1
8	$1+i$	i	i

表 5 複素 BP および実 BP の計算量 [例題 2]

時間計算量は 1 回の学習で行われる四則演算の総数、領域計算量は学習パラメータ数を意味する。

Table 5 Computational complexity of complex BP and standard BP [Example 2].

Time complexity means the sum of the four operations performed per learning cycle. Space complexity means the sum of the parameters (weights and thresholds).

ネットワーク	時間計算量			領域計算量		
	乗除算	加減算	計	重み	しきい値	計
複素 BP 2-4-1	134	92	226	24	10	34
実 BP 4-8-2	134	68	202	48	10	58
実 BP 4-9-2	150	76	226	54	11	65

表 6 収束率 [例題 2]
Table 6 Rate of convergence [Example 2].

ネットワーク	学 習 率					
	0.1	0.2	0.3	0.4	0.5	0.6
複素 BP 2-4-1	100	96	88	92	90	98
実 BP 4-8-2	0	48	70	84	94	94
実 BP 4-9-2	0	22	64	78	90	100

表 7 学習パラメータの修正量の絶対値の平均値 [例題 2]
Table 7 Expectation of the absolute value of the amount of correction for the parameters (weights and thresholds) [Example 2].

ネットワーク	学習パラメータ	修正量の絶対値の平均値
複素 BP 2-4-1	$Re[w_{12}]$	0.00331
	$Im[w_{12}]$	0.00513
実 BP 4-9-2	w'_{12}	0.00038
	w'_{22}	0.00906

平均学習回数

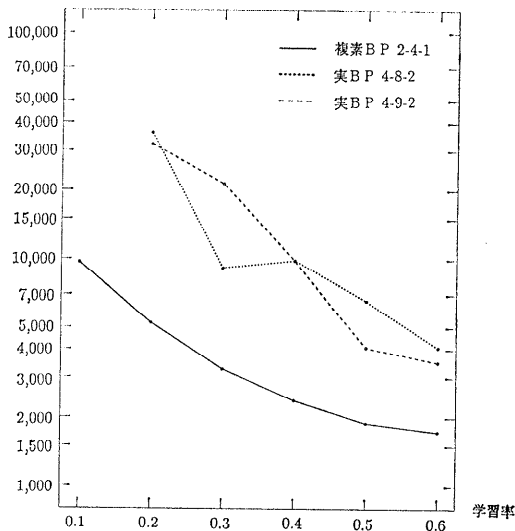


図 7 平均学習速度 (複素 BP と実 BP の比較) [例題 2]
Fig. 7 Average of learning speed (a comparison between complex BP and standard BP) [Example 2].

誤差

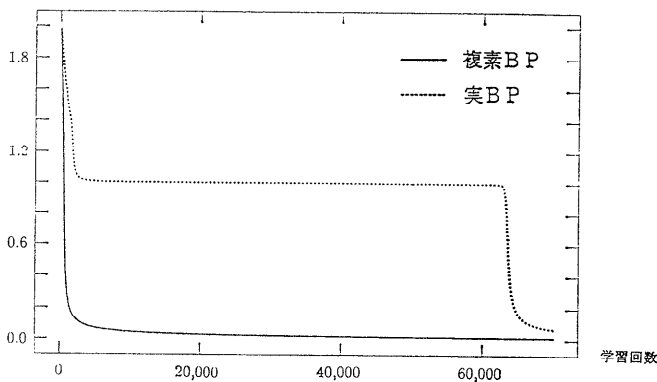


図 8 学習曲線 (複素 BP と実 BP の比較) [例題 2]
Fig. 8 Learning curve (a comparison between complex BP and standard BP) [Example 2].

のネットワークが使われる際のそれぞれの計算量を表 5 に示す。学習 1 回あたりの時間計算量に関して、複素 BP 2-4-1 ネットワークは、(1 回の加算の計算量) = 0 のとき、実 BP 4-8-2 ネットワークに等しく、(1 回の加算の計算量) = (1 回の乗算の計算量) のとき、実 BP 4-9-2 ネットワークに等しい。

実 BP に上述の学習パターンを与える際には、例題 1 と同様にして、入出力層において実部、虚部を担当させるニューロンをあらかじめ決めておいた。すなわち、入力される複素数 1 の実部、虚部、複素数 2 の実部、虚部を、この順番で、入力ニューロン 1, 2, 3 そして 4 に担当させた。また、出力される複素数 3 の実部は出力ニューロン 1、虚部は出力ニューロン 2 に担当させた。

学習は 10 万回で打ち切りとした。図 7 に実験結果を示す。参考のために、各ネットワークでの収束率 (50 回の試行のうち収束した試行の割合) を表 6 に示しておく (注意: この場合、収束率に関してみると学習率を 0.6 以上に選ぶと実 BP の方が良い結果が得られる可能性があるように思われる)。

また、学習曲線 (学習率は 0.3) の一例を図 8 に示す (実 BP は 4-9-2 ネットワーク)。

さらに、例題 1 と同様に、上述の実験における学習パラメータの修正量の絶対値の平均値を表 7 に示す。複素 BP, 実 BP 共に学習率 0.5 の場合について、学習回数が 450 回の時点での値を調べた (ただし、パターン 6 に対応するデータに基づいている)。なお、実 BP ネットワークの代表としては 4-9-2 ネットワークを選んだ。複素 BP については、出力ニューロン 1 の出力関数の導関数値の実部 $(1 - Re[O_1])Re$

$[O_i]$ が 0.05 以下、かつ、虚部 $(1 - \text{Im}[O_i])\text{Im}[O_i]$ が 0.10 以上であるような試行についての平均をとった。そのような試行は 50 回のうち 17 回あった。対象とした学習パラメータは中間ニューロン 2 と出力ニューロン 1 との間の重みパラメータ w_{12} である。実 BP については、出力ニューロン 1 の出力関数の導関数値 $(1 - O_1)O_1$ が 0.05 以下、かつ、出力ニューロン 2 の出力関数の導関数値 $(1 - O_2)O_2$ が 0.10 以上であるような試行についての平均をとった (O_k は出力ニューロン k の出力値を表す)。そのような試行は 50 回のうち 21 回あった。対象とした学習パラメータは中間ニューロン 2 と出力ニューロン 1 との間の重みパラメータ w_{12} および中間ニューロン 2 と出力ニューロン 2 との間の重みパラメータ w_{22} である。表 7 からわかるように、複素 BP では実 BP と異なり、出力関数の導関数値の実部、虚部のどちらか一方が小さな値をとった場合でも、それらに対応する学習パラメータの修正量は、例題 1 と同様に、実部、虚部共に同程度の値をとっており、学習停滞状態は発生しにくくなっていることがうかがえる。

4.3 考察

4.1 節および 4.2 節の実験結果により、複素 BP には複素パターンの学習に関する以下の特徴が備わっていることがわかる。

(i) 複素 BP は、実 BP に比べて、学習が数倍速くなっている (図 5, 図 7)。しかも、そのときの領域計算量は約半分で済む (表 2, 表 5)。

(ii) 複素 BP の学習曲線には、実 BP とは異なり、学習の停滞状態 (誤差が大きく、かつ、 x 軸にはほぼ平行な部分) が見当たらない (図 6, 図 8)。

これらは 3.2 節で述べた出力関数の導関数の実部と虚部の (学習則における) 線形結合性 (式 (17) および式 (18)) による学習停滞状態の解消構造が、学習速度および学習曲線に現れたものであると考えられる。

5. おわりに

複素 BP の基本特性に対して、学習アルゴリズムの側面からアプローチし、次の結果を得た。

複素数への拡張は複素 BP に次のような性質をもたらしている。複素 BP における誤差逆伝播は、(線形代数でいうところの) 2 次元運動に関係した構造を有している。複素 BP は、ネットワークを流れる複素信号を 1 つの処理単位として学習を進める。複素 BP の学習則には、学習停滞状態の発生を抑えるメカ

ニズム (実部要因と虚部要因の補充機構) が含まれている。これにより、複素パターンに対する学習が実 BP に比べて数倍速くなっている。しかも、そのときに必要となる領域計算量は約半分で済む。

これらの意味で複素 BP は複素パターンを自然に学習することができるものであり、信号処理、電気工学、制御工学などの複素パターンを操作する分野において有効に利用され得るものと期待される。

今後は、画像処理や音声認識などへの本格的な応用を通じた複素 BP の評価を進めるとともに、複素関数の近似能力の解析などの理論的なアプローチを目指してゆく予定である。また、複素 BP をさらに高次元化した場合に、本論文で示したような特性は基本的にはそのまま引き継がれていくものと予想している。

謝辞 本研究の機会を与えてくださった情報アーキテクチャ部 弓場部長、貴重なコメントをいただいた情報科学部情報数理研究室 赤穂研究官、有益なコメントをくださった査読者の方々に感謝いたします。

参考文献

- 1) 秋山, 古谷: 損失関数にエントロピー項を導入したバックプロパゲーション学習則, 信学技報, NC 91-6, pp. 39-46 (1991).
- 2) 麻生: ニューラルネットワーク情報処理, p. 198, 産業図書, 東京 (1988).
- 3) 甘利: 学習識別の理論, 電子通信学会誌, Vol. 50, No. 7, pp. 1272-1279 (1967).
- 4) 佐竹, 古谷, 下原: ニューラルネットワークによる米の品質判定, 農業施設学会研究発表会講演要旨, pp. 3-4 (1990).
- 5) 丹, 加藤, 江島: 誤差評価関数による PDP モデルの高速化, 信学論, Vol. J 73-D-II, No. 12, pp. 2022-2028 (1990).
- 6) 新田, 古谷: 複素バックプロパゲーション学習, 情報処理学会論文誌, Vol. 32, No. 10, pp. 1319-1329 (1991).
- 7) 新田, 赤穂, 秋山, 古谷: 複素 BP の基本的性質, 第 43 回情報処理学会全国大会論文集 (分冊 2), 6G-8, pp. 223-224 (1991).
- 8) 新田, 古谷: 複素 BP の汎化能力, 第 43 回情報処理学会全国大会論文集 (分冊 2), 6G-9, pp. 225-226 (1991).
- 9) 新田, 赤穂, 秋山, 古谷: 複素バックプロパゲーション・ネットワークにおける重みパラメータと決定表面の構造, 情報処理学会論文誌, Vol. 33, No. 11, pp. 1306-1313 (1992).
- 10) 新田, 古谷: 複素 BP 学習の高速性, 信学技法, NC 91-127, pp. 215-221 (1992).
- 11) 新田: 回転を学習した複素 BP ネットワークのふるまい, 情報処理学会論文誌, Vol. 34, No. 1, pp. 39-51 (1993).

- 12) 野崎: アルゴリズムと計算量, p. 220, 共立出版, 東京 (1987).
- 13) Rezgui, A. et al.: The Effect of the Slope of the Activation Function on the Back Propagation Algorithm, *Proc. IJCNN 1990*, Vol. 1, pp. 707-710 (Jan. 1990).
- 14) Rumelhart, D. E. et al.: *Parallel Distributed Processing*, Vol. 1, p. 547, MIT Press (1986).
- 15) Yamada, K. et al.: Handwritten Numeral Recognition by Multi-layered Neural Network with Improved Learning Algorithm, *Proc. IJCNN 1989*, Vol. 2, pp. 259-266 (June 1989).

(平成4年2月12日受付)

(平成4年11月12日採録)



新田 徹 (正会員)

1960年生. 1983年筑波大学第1学群自然科学類卒業. 1985年同大学院修士課程修了. 同年日本電気(株)入社. 1990年同社退社. 同年電子技術総合研究所入所. 現在, 同所情報アーキテクチャ部計算機構研究室主任研究官. ニューラルネットワークの研究に従事. 日本OR学会会員.



古谷 立美 (正会員)

昭和22年生. 昭和48年成蹊大学大学院修士課程修了. 同年電子技術総合研究所入所. 現在同所計算機構研究室長. 工学博士. IEEE 会員.