

## 共役勾配法を導入した BP 学習における 安定化に関する研究

後藤 正幸\*, 俵 信彦\*\*

本研究は、ニューラルネットワークの BP 学習に対して、高速で安定性の高い学習アルゴリズムを構築する。共役勾配法の学習速度に対する高速性が明らかとなっているが、学習の初期値に依存するという不安定さを合わせ持つ。そこで、本報では、共役勾配法の学習速度を維持しつつ、安定性を向上させる方法を研究する。具体的には、従来の局所的に二次近似して共役方向を生成する方法に対し、二次近似する範囲を調節できる共役方向生成公式を提案する。本公式を導入した方法を改良共役勾配法と呼ぶ。改良共役勾配法ではより広い範囲で二次近似できるため、局所的情報のみにとらわれず、安定性の高い探索が期待できる。シミュレーション実験の結果、改良共役勾配法は、従来法に比べて、初期値によるばらつきを約半分に抑えることが明らかになった。また、学習速度の面でも高速性を維持している。

### A Study on BP Learning Algorithm with Higher Accuracy based on Conjugate Gradient Method

Masayuki GOTOH\* and Nobuhiko TAWARA\*\*

The objective of the study is to construct a learning algorithm with high speed and accuracy for a neural network model. A learning algorithm from Conjugate Gradient Method is an effective method for high speed learning. However, the Conjugate Gradient Method is unstable in its convergence depending on an initial setting of network parameter values. We, therefore, propose a learning algorithm with higher accuracy based on Conjugate Gradient Method. The method proposed in the study is based on improved formula to create a conjugate direction. We call the proposed method "Improved Conjugate Gradient Method".

#### 1. はじめに

階層型ニューラルネットワークは非線形写像近似手法として提案され〔1〕、経営工学分野においても、パターン認識、経済指標予測、企業の優良性診断、故障診断、気象予測などに広く適用されている〔2〕、〔3〕。

しかし、モデルの構築法である Back Propagation 学習(以下 BP 学習)は、本質的には二乗誤差に対する最急降下法であるため、1) 学習速度が遅い、2) 学習が初期値に依存してしまう(学習の不安定性)という学習の問題点が生じている〔4〕～〔6〕。

学習速度に対しては共役勾配法が提案されており、最急降下法に比べて高速な学習方法であることが明らかになっている〔4〕、〔5〕。共役勾配法は二次関数の最小化問題に対して導出されたが、BP 学習のような

一般の関数の最適化問題に対しても、局所的には二次関数で近似できるという考えからそのまま適用され、シミュレーションによってその有効性が示されている。また、共役方向の生成公式についてはいくつか提案されており〔9〕、〔10〕、二次関数に対してはそれらは等価といえる。しかし、一般の関数に適用した場合には、それらは等価でなくなり、どの公式が良いかについては、実験的に論じられている。

一方、共役勾配法による BP 学習は、学習の初期値によって結果がばらつくという学習の不安定さも合わせ持つ。これは、一般の関数を局所的に二次近似して共役方向を生成することによる歪と考えられる。従って、共役方向生成公式は、一般の関数に拡張した場合の安定性を考慮して導かれるべきである。

そこで本研究では、共役勾配法の不安定性は二次近似の歪が原因と考え、二次近似する範囲を調整することにより、共役勾配法の学習速度を維持しつつ、安定性を向上させる共役方向生成公式を提案する。本公式による方法を改良共役勾配法と呼ぶ。従来の共役方向生成公式より広い範囲で二次近似を行うことで、局所的な情報に捕らわれずに安定した探索を行うことがで

\* 早稲田大学(Waseda University)

\*\* 武蔵工業大学(Musashi Institute of Technology)

受付: 1994年4月6日, 再受付(1回)

受理: 1994年10月20日

きる。そして、二次関数の最小化問題に対して、提案公式が従来の公式と等価であることを示す。さらに、提案公式を導入した改良共役勾配法をBP学習に適用し、シミュレーションを通じてその有効性を示す。

2. 二次近似の範囲を調整する共役方向生成公式

2.1 二次関数最小化問題に対する共役勾配法

まず、二次関数最小化問題に対する共役勾配法の定義を以下に示す。

定義 2-1 正定対称行列  $Q \in R^{n \times n}$  に対して、 $l$  個のベクトル  $d^1, d^2, \dots, d^l \in R^n$  が

$$(d^i)^T Q d^j = 0 \quad i \neq j \text{ かつ } i, j = 1, 2, \dots, l \quad (1)$$

を満たすとき、 $d^1, d^2, \dots, d^l$  は  $Q$  に対して互いに共役であるという。□

定義 2-2  $c \in R^1, b \in R^n$  で  $Q \in R^{n \times n}$  を対称な正定値行列として、二次関数

$$q(x) = c + b^T x + \frac{1}{2} x^T Q x \quad (2)$$

を最小化するために、探索方向が  $Q$  に対して互いに共役になる方向に直線探索を行うアルゴリズムを共役方向法といい、その中でも目的関数の微係数を利用して共役方向を生成する方法を共役勾配法という。□

共役勾配法では適当な直線探索を行えば  $n$  回の探索で二次関数の最適解を得ることができる。  $l$  回目の探索点を  $x^l$  とすると、次の探索点  $x^{l+1}$  を以下のように計算する。目的関数を  $q$ 、探索方向ベクトルを  $d^l$  として、

$$x^{l+1} = x^l + \alpha^l d^l \quad (l=1, 2, 3, \dots) \quad (3)$$

$$q(x^{l+1}) = \min_{\alpha > 0} \{q(x^l + \alpha d^l)\} \quad (4)$$

$$d^l = -\nabla^T q(x^l) \quad (5)$$

とすれば、 $l$  回目の共役方向  $d^l$  は、以下の式(6)で生成できる。

$$d^l = -\nabla q(x^l) + \beta^l d^{l-1} \quad (6)$$

式(4)の計算を直線探索といい、 $\alpha^l$  を歩み幅という。 $\beta^l$  ( $l=2, \dots, n$ ) の計算法としては、次のような公式が知られている。

1) Fletcher-Reeves の公式

$$\beta_{FR}^l = \frac{\|\nabla q(x^l)\|^2}{\|\nabla q(x^{l-1})\|^2} \quad (7)$$

2) Polak-Ribiere-Polyak の公式

$$\beta_{PRP}^l = \frac{\nabla^T q(x^l)(\nabla q(x^l) - \nabla q(x^{l-1}))}{\|\nabla q(x^{l-1})\|^2} \quad (8)$$

3) Sorenson-Wolfe の公式

$$\beta_{SW}^l = \frac{\nabla^T q(x^l)(\nabla q(x^l) - \nabla q(x^{l-1}))}{(d^{l-1})^T (\nabla q(x^l) - \nabla q(x^{l-1}))} \quad (9)$$

2.2 近似範囲調整共役方向生成公式の導出

(7)~(9)の公式は、現時点の探索点とその一回前

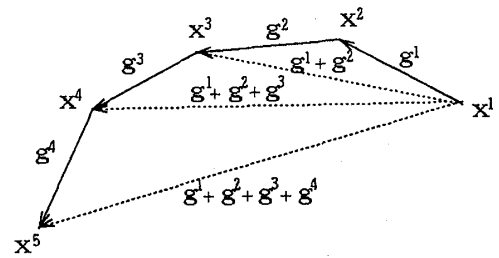


図1 探索ベクトルの概念図

の探索の情報を利用して、共役方向を生成している。一般の関数に適用した場合、この二探索点の範囲を二次関数で近似していることになる。これに対し、本研究では、探索の初期点の微係数も利用して共役方向を生成する公式を導出する。具体的には、更新ベクトルを  $g^l = \alpha^l d^l$  ( $l=1, 2, \dots, n$ ) として、 $l$  回目の探索を全更新ベクトル  $(g^1 + g^2 + \dots + g^{l-1})$  と共役になる方向に行うという考えに基づいた  $\beta^l$  の計算公式である(図1)。

まず、本公式の基礎となる定理を示す。 $l \leq n$  とする。

定理 2-1 正定対称行列  $Q \in R^{n \times n}$  に対して、 $d^1, d^2, \dots, d^{l-1}$  と  $d^l$  は共役であるとする。このとき、全更新ベクトル  $(g^1 + g^2 + \dots + g^{l-1})$  と  $d^l$  も共役な関係にある。

(証明)

$$\begin{aligned} & (g^1 + g^2 + \dots + g^{l-1})^T Q d^l \\ &= (\alpha^1 d^1)^T Q d^l + (\alpha^2 d^2)^T Q d^l + \dots \\ & \dots + (\alpha^{l-1} d^{l-1})^T Q d^l = 0 \quad \square \end{aligned}$$

そこで、逆に  $(g^1 + g^2 + \dots + g^{l-1})$  と共役な方向へ探索するアルゴリズムが考えられる。

定理 2-2 式(2)で示される二次関数を最小化するために  $l$  回目の探索方向ベクトル  $d^l$  を、 $Q$  に関して  $(g^1 + g^2 + \dots + g^{l-1})$  と共役な方向とする探索アルゴリズムを考える。このとき、 $x^{l+1}$  が  $x^1 + \text{span}[d^1, d^2, \dots, d^l]$  上の二次関数  $q(x)$  の最小点である、すなわち、

$$q(x^{l+1}) = \min \{q(x) \mid x \in x^1 + \text{span}[d^1, d^2, \dots, d^l]\} \quad (10)$$

が成り立つためには、 $d^1, \dots, d^{l-1}, d^l$  は  $Q$  に対して互いに共役、つまり、

$$(d^i)^T Q d^j = 0, \quad i \neq j \text{ かつ } i, j = 1, 2, \dots, l \quad (11)$$

が成り立たなければならない。ただし、 $\text{span}[d^1, d^2, \dots, d^l]$  は  $d^1, d^2, \dots, d^l$  の張る部分空間である。

(証明)

$q$  が狭義凸関数であることを考慮すると、式(10)が成り立つためには、

$$\nabla q(x^{l+1}) \perp \text{span}[d^1, d^2, \dots, d^l] \quad (12)$$

つまり,

$$\nabla^T q(x^{l+1})d^i = 0, \quad i=1, 2, \dots, l \quad (13)$$

が成り立たなければならない。

(1)  $l=2$  のとき,

$$(g^1)^T Q d^2 = a^1 (d^1)^T Q d^2 = 0 \quad (14)$$

となるように  $d^2$  は決定されるから、式(11)は成り立つ。

(2)  $l=k$  のとき、式(10)が成り立つために、式(11)が成り立っていると仮定する。

$$(d^i)^T Q d^j = 0, \quad i \neq j \text{ かつ } i, j=1, 2, \dots, k \quad (15)$$

このとき、式(10)が成り立つという仮定から、式(16)が成り立つ。

$$\nabla q(x^k) d^i = 0, \quad i=1, 2, \dots, k-1 \quad (16)$$

まず、歩み幅  $a^{k+1}$  の最適性から、

$$\nabla^T q(x^{k+2}) d^{k+1} = 0 \quad (17)$$

また、

$$\nabla q(x^{k+2}) = \nabla q(x^{k+1}) + a^{k+1} Q d^{k+1} \quad (18)$$

より、 $i \leq k$  として、

$$\begin{aligned} \nabla^T q(x^{k+2}) d^i &= \nabla^T q(x^{k+1}) d^i + a^{k+1} (d^{k+1})^T Q d^i \\ &= \nabla^T q(x^{k+1}) d^i + a^{k+1} (d^{k+1})^T Q d^i \end{aligned} \quad (19)$$

ここで、式(13)より、

$$\nabla^T q(x^{k+1}) d^i = 0, \quad i=1, 2, \dots, k \quad (20)$$

$$\nabla^T q(x^{k+2}) d^i = 0, \quad i=1, 2, \dots, k+1 \quad (21)$$

従って、

$$(d^{k+1})^T Q d^i = 0, \quad i=1, 2, \dots, k \quad (22)$$

となり、式(15)と合わせると、

$$(d^i)^T Q d^j = 0, \quad i \neq j, i, j=1, 2, \dots, k+1 \quad (23)$$

となり、 $l=k+1$  のときも式(11)は成り立つ。

以上(1)、(2)から、数学的帰納法により、任意の  $l$  に対して式(11)は成り立つ。 □

式(10)は、 $n$ 次元二次関数において高々  $n$ 回の探索で最適点を探索するための条件である。次に、この定理2-2に従って、 $l$ 回目の探索方向ベクトル  $d^l$ を、 $(g^1 + g^2 + \dots + g^{l-1})$ と共役な方向とする探索アルゴリズムを導出する。

定理2-2より  $d^l$ は、式(6)で生成することができる。よって  $l$ 回目の探索では、

$$\{-\nabla q(x^l) + \beta^l d^{l-1}\}^T Q (g^1 + g^2 + \dots + g^{l-1}) = 0 \quad (24)$$

となるように  $\beta^l$ を決定すればよい。ここで、

$$Q (g^1 + g^2 + \dots + g^{l-1}) = \nabla q(x^l) - \nabla q(x^1) \quad (25)$$

という関係と式(6)を用いれば、式(24)の左辺は、次のように展開できる。

$$\{-\nabla q(x^l) + \beta^l d^{l-1}\}^T Q (g^1 + g^2 + \dots + g^{l-1})$$

$$\begin{aligned} &= \{-\nabla q(x^l) + \beta^l d^{l-1}\}^T (\nabla q(x^l) - \nabla q(x^1)) \\ &= -\|\nabla q(x^l)\|^2 + \beta^l (d^{l-1})^T \nabla q(x^1) \\ &= -\|\nabla q(x^l)\|^2 \\ &\quad + \beta^l \{-\nabla q(x^{l-1}) + \beta^{l-1} d^{l-2}\}^T \nabla q(x^1) \\ &= -\|\nabla q(x^l)\|^2 + \beta^l \beta^{l-1} (d^{l-2})^T \nabla q(x^1) \\ &\quad \dots \dots \dots \\ &= -\|\nabla q(x^l)\|^2 + \beta^l \beta^{l-1} \dots \beta^2 \|\nabla q(x^1)\|^2 \end{aligned} \quad (26)$$

ただし、 $\nabla^T q(x^l) d^i = 0$  ( $i=1, 2, \dots, l-1$ )となる性質を利用した。これを0とおけば、次式が得られる。

$$\beta_{\text{Gr}}^l = \frac{1}{\beta^{l-1} \beta^{l-2} \dots \beta^2} \frac{\|\nabla q(x^1)\|^2}{\|\nabla q(x^l)\|^2} \quad (27)$$

$\nabla^T q(x^l) \nabla q(x^{l-1}) = 0$ より、次式も得られる。

$$\beta_{\text{Gr}}^l = \frac{1}{\beta^{l-1} \beta^{l-2} \dots \beta^2} \frac{\nabla^T q(x^l) \{\nabla q(x^l) - \nabla q(x^{l-1})\}}{\|\nabla q(x^1)\|^2} \quad (28)$$

定理2-3 式(2)で定義された二次関数の最小化問題に関して、式(28)は、(7)~(9)の公式と等価である。

(証明) これまでの議論より明らかである。ただし、式(7)より直接式(27)を得ることもできる。 □

式(28)は、過去の各探索点の情報を利用して共役方向を生成している。一般の関数へ共役勾配法を拡張した場合、従来の式(7)と式(8)による共役勾配法では、局所的な情報で共役方向を近似しているため、グローバルに見た共役方向が生成できない場合も考えられる。つまり、一旦ローカルミニマに陥ると、その近辺を二次近似して探索を続け、脱出が難しくなる。これに対し、式(28)では過去の全ての履歴から共役方向が計算されるため局所的な情報に捕らわれずに探索でき、一般の関数へ拡張した場合に高い頑健性が期待できる。

しかし、この方法では、探索回数が増えるに従って、近似範囲も広がっていく。一般的な関数をあまり広い範囲で二次近似しすぎると、近似精度が悪くなるため、逆に探索効率が低下する。そこで、式(28)を一般化し、近似範囲を調整できる公式に拡張する。

定理2-4 式(2)で定義された二次関数の最小化問題に関して共役勾配法を適用したとき、次式は式(7)~(9)の公式と等価である。

$$\beta_{\text{Gr}}^l = \frac{1}{\beta^{l-1} \beta^{l-2} \dots \beta^{l-j}} \frac{\|\nabla q(x^l)\|^2}{\|\nabla q(x^{l-j-1})\|^2} \quad (29)$$

$$\beta_{\text{Gr}}^l = \frac{1}{\beta^{l-1} \beta^{l-2} \dots \beta^{l-j}} \frac{\nabla^T q(x^l) \{\nabla q(x^l) - \nabla q(x^{l-1})\}}{\|\nabla q(x^{l-j-1})\|^2} \quad (30)$$

ただし、 $j$ は  $j=1, 2, 3, \dots, l-1$ で設定された数。

(証明)

Fletcher-Reeves の公式より,

$$\beta_{FR}^l = \frac{\|\nabla q(x^l)\|^2}{\|\nabla q(x^{l-1})\|^2}$$

および

$$\|\nabla q(x^{l-1})\|^2 = \beta^{l-1} \|\nabla q(x^{l-2})\|^2$$

が成り立つから,

$$\beta^l = \frac{\|\nabla q(x^l)\|^2}{\beta^{l-1} \|\nabla q(x^{l-2})\|^2}$$

となり,  $j=1$  で式(29)は成り立っている.ここで, 任意の  $j$  ( $j=2, 3, 4, \dots, l-1$ ) で式(29)が成り立っているものと仮定すると,

$$\beta^l = \frac{1}{\beta^{l-1} \beta^{l-2} \dots \beta^{l-j}} \frac{\|\nabla q(x^l)\|^2}{\|\nabla q(x^{l-j-1})\|^2}$$

および Fletcher-Reeves の公式,

$$\|\nabla q(x^{l-j-1})\|^2 = \beta^{l-j-1} \|\nabla q(x^{l-j-2})\|^2$$

より,

$$\beta^l = \frac{1}{\beta^{l-1} \beta^{l-2} \dots \beta^{l-j-1}} \frac{\|\nabla q(x^l)\|^2}{\|\nabla q(x^{l-j-2})\|^2}$$

が得られ,  $j+1$  においても式(29)は成り立つ. 従って, 数学的帰納法により任意の  $j$  ( $=1, 2, 3, \dots, l-1$ ) において式(29)は成り立つ. また,  $\nabla q(x^l)$  と  $\nabla q(x^{l-1})$  の直交性から式(30)も成り立つ.  $\square$

ここで示した公式は, ある設定した  $j$  について, 現探索点から数えて  $j+1$  回前の探索点までの情報を利用して共役方向を生成する公式である. 従って,  $j$  を適当に設定することにより, 再出発とは別に, どの範囲を二次近似するかを決定することができる.

しかし, この方法では  $\beta^{l-1}, \beta^{l-2}, \dots, \beta^{l-j}$  も式(30)で計算するため, 再出発からの情報が全て累積して入ってきてしまい, 純粋に  $j+1$  回前の探索点までの情報を利用して共役方向を生成することにはならない. そこで, これを解消するため, 従来の公式により局所的情報で計算した  $\hat{\beta}^{l-1}, \hat{\beta}^{l-2}, \dots, \hat{\beta}^j$  を基にして, 新たに  $\beta^l$  を計算する式(31), 式(32)に変形し, この公式を改良共役勾配法として提案する.

$$\beta_{GT}^l = \frac{1}{\hat{\beta}^{l-1} \hat{\beta}^{l-2} \dots \hat{\beta}^{l-j}} \frac{\nabla^T q(x^l) \{\nabla q(x^l) - \nabla q(x^{l-1})\}}{\|\nabla q(x^{l-j-1})\|^2} \quad (31)$$

$$\hat{\beta}^l = \frac{\nabla^T q(x^l) \{\nabla q(x^l) - \nabla q(x^{l-1})\}}{\|\nabla q(x^{l-1})\|^2} \quad (32)$$

ただし,  $j$  は  $j=1, 2, 3, \dots, l-1$  で設定された数とし,  $j=0$  の場合は式(8)と定義する.

この方法により,  $j+1$  回前までの  $j+2$  回分の探索点の範囲についての情報を利用して, 共役方向を生成できる. 一般の関数に適用した場合, この  $j+2$  回の探索点の範囲を二次近似して共役方向を生成している

ため,  $j$  を適切に設定することにより, 局所的情報にとらわれない安定した探索が期待できる.

$j=0$  の場合に式(8)になるとした定義は, 式(31)による本探索法が  $j+2$  回分の探索点の情報を利用して共役方向を生成することを考えれば, 極めて自然な定義といえる. つまり, 式(8)は1回前の探索点を合わせた2探索点の情報から共役方向を生成するため, 提案公式の  $j=0$  の場合に相当するためである.

また, 本提案方式は非線形最適化手法の枠組みの中での拡張であり, その後, BP 学習へ適用される. 従って, ニューラルネットワークの BP 学習にのみ適用が限定された学習法ではない.

### 3. 共役勾配法による BP 学習への適用

実際にニューラルネットワークに学習させる際の共役勾配法アルゴリズムを以下に示す.

$$\Delta w(t) = \alpha(t) \left\{ -\nabla E(t) + \beta(t) \frac{\Delta w(t-1)}{\alpha(t-1)} \right\} \quad (33)$$

ただし,  $\alpha(t)$  は直線探索により求める歩み幅(ステップ幅),  $-\nabla E(t)$  は各学習パターン之二乗誤差に対する最急降下方向の和とし, BP アルゴリズムで生成する.  $t$  は再出発 ( $t=1$ ) からの更新回数である.

$\beta^l$  は以下の公式で計算する. 式(34)は Polak-Ribiere-Polyak の公式であり, 式(35), 式(36)は本研究で提案した公式である.

1) Polak-Ribiere-Polyak の共役勾配法

$$\beta(t)_{PRP} = \frac{\nabla^T E(t) \{\nabla E(t) - \nabla E(t-1)\}}{\|\nabla E(t-1)\|^2} \quad (34)$$

2) 改良共役勾配法

$$\beta(t)_{GT} = \frac{1}{\hat{\beta}(t-1) \hat{\beta}(t-2) \dots \hat{\beta}(t-j)} \frac{\nabla^T E(t) \{\nabla E(t) - \nabla E(t-1)\}}{\|\nabla E(t-j-1)\|^2} \quad (35)$$

$$\hat{\beta}(t) = \frac{\nabla^T E(t) \{\nabla E(t) - \nabla E(t-1)\}}{\|\nabla E(t-1)\|^2} \quad (36)$$

ただし,  $j$  は  $j=1, 2, 3, \dots, l-1$  で設定された数であり,  $j=0$  のときは式(34)と定義する.

ここで, 一般の関数(二乗誤差関数)を二次近似しているため, 探索曲面の性質によっては, 式(33)で計算した探索方向が降下方向になっていない場合も考えられる. そこで, 計算した探索方向が降下方向にならない場合には, 最急降下方向へセットしなおして探索を行うものとする.

また, 本研究では, 共役方向への歩み幅を決める直線探索として, 次の囲い込みと二次補間法を使用する.

(囲い込み-二次補間法)

十分小さな値から、次第に  $\alpha$  を大きくしていき、

$$\begin{cases} E(\alpha_1) \geq E(\alpha_2) \leq E(\alpha_3) \\ \alpha_1 < \alpha_2 < \alpha_3 \end{cases} \quad (37)$$

となる三点  $\alpha_1, \alpha_2, \alpha_3$  を見いだす。この三点において  $E_1 = E(\alpha_1), E_2 = E(\alpha_2), E_3 = E(\alpha_3)$  を計算すれば最小値の推定値として  $\hat{a}$  を次式により得ることができる。

$$\hat{a} = \frac{\{\alpha_2^2 - \alpha_3^2\}E_1 + \{\alpha_3^2 - \alpha_1^2\}E_2 + \{\alpha_1^2 - \alpha_2^2\}E_3}{2\{(\alpha_2 - \alpha_3)E_1 + (\alpha_1 - \alpha_2)E_3 + (\alpha_3 - \alpha_1)E_2\}} \quad (38)$$

#### 4. 改良共役勾配法の計算量に関する考察

本節では、Polak-Ribiere-Polyak の共役勾配法と改良共役勾配法に関して、学習 1 回当たりの計算量の面から、計算の複雑さについて考察する。本研究では、計算量の基本単位を次のように表記する。

$$u \leftarrow v \Delta w \quad (39)$$

ここで、 $u$  は変数であり、 $v$  と  $w$  は変数または定数である。記号  $\Delta$  は、演算記号  $+$ ,  $-$ ,  $\times$  のうちのどれかである。この代入文を 1 単位として、以下に学習 1 回当たりの計算量を示す。

##### 4.1 計算量算出のための仮定

ニューラルネットワークは、入力層素子数  $m$ 、中間層素子数  $n$ 、出力層素子数 1 の 3 層ネットワークとし、学習サンプルの数を  $p$  とする。また、囲い込みでは、 $\alpha$  を設定値  $h (> 1)$  に従って、

$$\alpha = ah \quad (40)$$

のように次第に大きくしていくものとする。ここで、式(40)の計算を  $s$  回計算したときに、式(37)の三点を見いだしたものと仮定する。

##### 4.2 学習 1 回当たりの計算量

4.1 の仮定のもとで、学習 1 回当たりの計算量を示すと以下ようになる。

(1) ニューラルネットワークの出力計算

$$m(m-1) + n(n-1) + 4 \quad (41)$$

(2) 最急降下ベクトルの計算

$$p\{(4n+1)(n+1) + \{mn(2n+1) + 1\}(mn+n)\} + p - 1 \quad (42)$$

(3)  $\beta^l$  の生成の計算

1) Polak-Ribiere-Polyak の共役勾配法

$$5n(m+2) + 4 \quad (43)$$

2) 改良共役勾配法

$$t > j \text{ のとき,} \\ 10n(m+2) + 8 + j \quad (44)$$

$$t \leq j \text{ のとき,}$$

$$10n(m+2) + 7 + t \quad (45)$$

(4) 共役方向生成の計算

$$2n(m+2) + 2 \quad (46)$$

(5) 直線探索の計算

$$s[p\{m(m-1) + n(n-1) + 4\} + p + 2] + s + 19 \quad (47)$$

#### 4.3 計算量の面からの考察

まず、本研究で提案した共役方向生成公式と Polak-Ribiere-Polyak の公式の計算量を比較すると、式(43)~(45)に示したように、改良共役勾配法では計算量はほぼ倍増している。しかし、 $\beta^l$  以外の計算過程の計算量をみると、 $\beta^l$  以外の計算に必要な計算量の方がはるかに大きいことがわかる。とくに、直線探索部分の計算量は、 $s$  によって変動はするが、 $\beta^l$  に比べて非常に大きい。 $s$  については、本報の 6. のシミュレーション実験では、歩み幅の平均値を探索するためには、 $s$  は 29 としなくてはならなかった。また、パターン数が増えれば、直線探索、最急降下方向の計算量は増加していくが、 $\beta^l$  の計算は増えないこともわかる。

従って、本研究で提案した公式は  $\beta^l$  の計算が複雑になっているが、全体の計算量からすればそれほど複雑化していないといえる。 $\beta^l$  の計算量を議論するよりは、直線探索部分の計算量をどのように削減するかを議論する方が重要である〔6〕~〔8〕。

#### 5. 乱数データの学習シミュレーション実験

##### 5.1 シミュレーション条件

本節では、予備実験として乱数で発生させたデータを学習させた結果を示す。データは入力 5 変数、出力 1 変数とし、全て  $[0, 1]$  で 20 サンプルを一様乱数で発生させた。同様にして乱数発生を数度繰り返し、10 種類の学習データのセットを作成し、それぞれの学習データセットに対して実験を行った。

ニューラルネットワークは 3 層とし、ウェイトの初期値は  $-0.10 \sim 0.10$  の範囲で 30 通り乱数発生させて実験を行った。学習回数は 300 回とし、学習終了時の二乗誤差の初期値による平均値、標準偏差で評価を行う。改良共役勾配法においては、 $j=2$  とした。

また、従来の Polak-Ribiere-Polyak の共役勾配法は、以下では単に共役勾配法と示してある。

##### 5.2 実験結果および考察

学習終了時の二乗誤差の平均値、標準偏差を比較した結果を表 1 に示す。

二乗誤差の平均値と標準偏差は、全体的に改良共役

表1 乱数データのシミュレーション結果

データ	共役勾配法		改良共役勾配法	
	平均値	標準偏差	平均値	標準偏差
1	0.0210	0.04909	0.0172	0.01802
2	0.0243	0.05301	0.0194	0.03044
3	0.0198	0.02841	0.0192	0.01965
4	0.0238	0.04605	0.0230	0.02534
5	0.0186	0.03704	0.0183	0.02649
6	0.0267	0.04834	0.0227	0.01903
7	0.0381	0.06318	0.0304	0.02942
8	0.0293	0.05196	0.0189	0.02415
9	0.0326	0.05614	0.0205	0.02906
10	0.0235	0.04998	0.0171	0.01862

勾配法の方が小さい。二乗誤差の平均値が小さいということは、改良共役勾配法がローカルミニマを回避していることを示している。また、二乗誤差の平均値に大きな差がない場合でも、標準偏差は約半分に抑えられている。標準偏差が小さいということは、探索の初期値に大きく影響されず、どのような初期値からも安定して小さな二乗誤差を探索できているといえる。

## 6. 化学プロセスの制御介入データに対する学習シミュレーション実験

### 6.1 学習対象データ

5.で提案法の有効性が示された。そこで、本章では化学プロセスの制御介入モデル構築に適用することにより、提案法の有効性を検証する。

このプロセスでは、熟練オペレータがプロセスの状態を監視し、経験と勘により操作(制御介入)を加えている。本研究では、このオペレータの知識をニューラルネットワークモデルで構築することを試みた。入力変数としてはプロセスの状態を表す13変数(各設置点での温度の変化率など)を取り、出力は熟練オペレータが実際に入力した制御介入量とした。入力変数は(-1,1)に変数変換して使用し、出力は(0,1)に変数変換した。学習用データには350サンプルを用いた。図2に、構築したニューラルネットワークモデルを示す。

### 6.2 シミュレーション条件

使用したネットワークモデルは、入力層素子数13、中間層素子数3、出力層素子数1の3層ネットワークである。ウェイトの初期値は-0.10~0.10の範囲で10通り乱数発生させた。学習回数は300回とし、学習終了時の二乗誤差の初期値による平均値、標準偏差で評価を行う。

また、改良共役勾配法では、 $j=2$ とした。

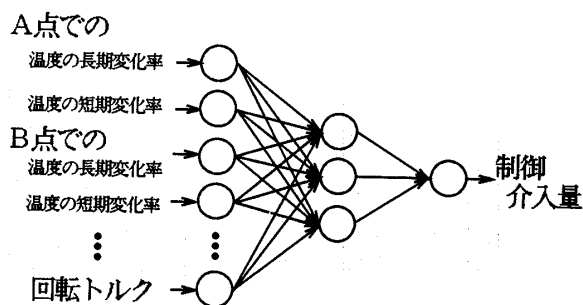


図2 制御介入ニューラルネットワークモデル

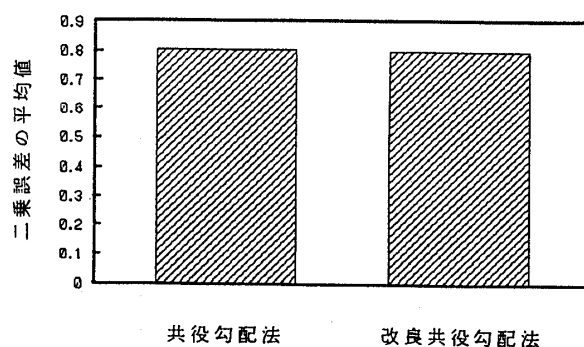


図3 学習終了時二乗誤差の平均値比較

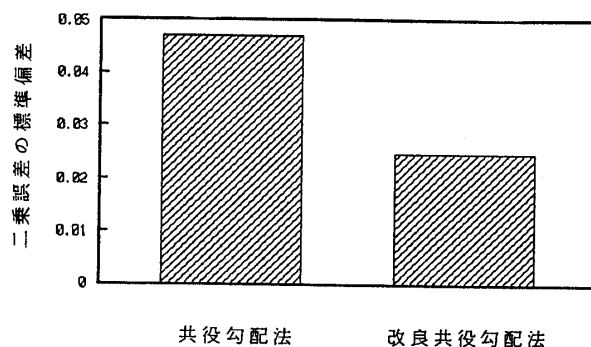


図4 学習終了時二乗誤差の標準偏差比較

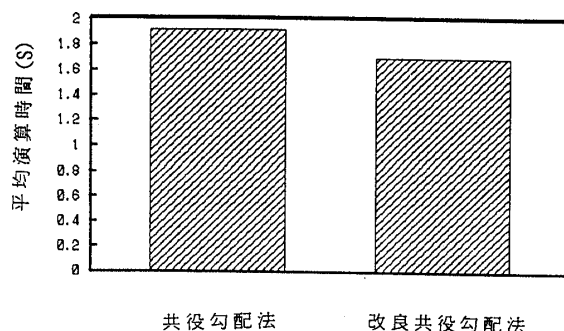


図5 平均演算時間(学習300回)の比較

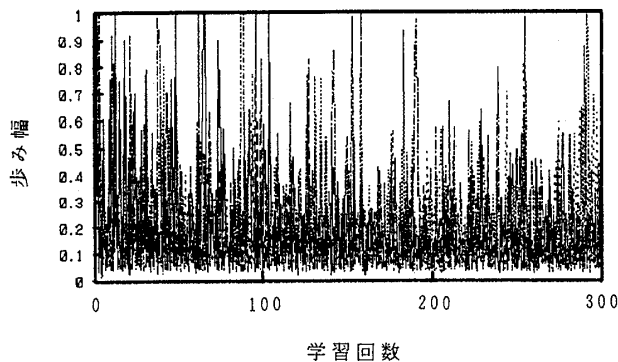


図6 歩み幅の挙動(共役勾配法)

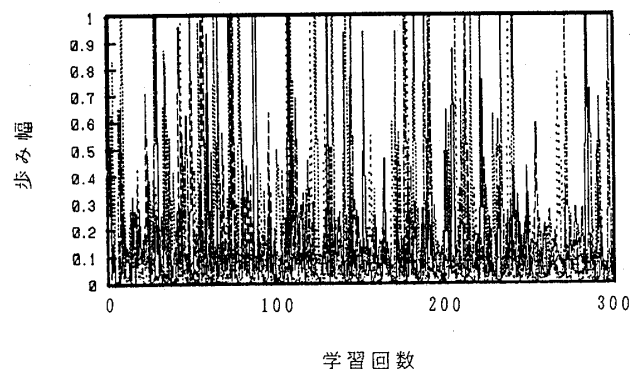


図7 歩み幅の挙動(改良共役勾配法)

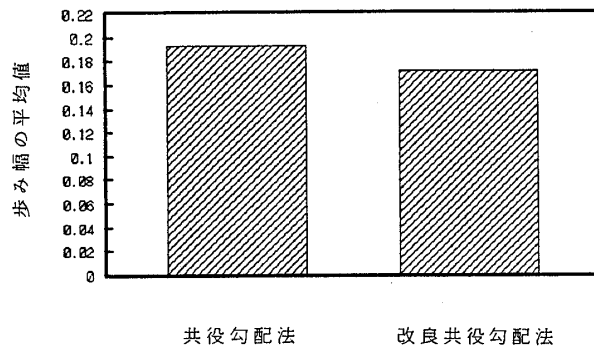


図8 歩み幅の平均値比較

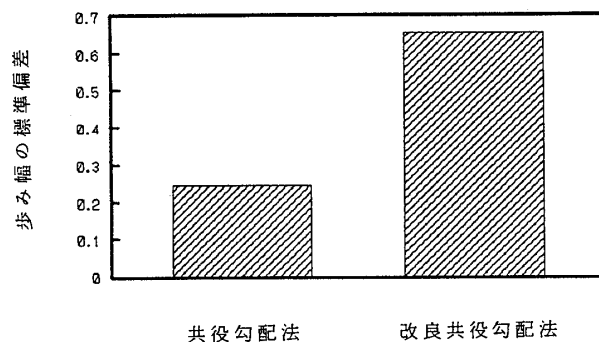


図9 歩み幅の標準偏差

### 6.3 シミュレーション結果および考察

以下に、学習 300 回終了時の二乗誤差の初期値による平均値(図3)と標準偏差(図4)、学習 300 回に要した演算時間の初期値による平均値(図5)、学習回数に対する歩み幅の挙動(図6、図7)、および、その歩み幅の平均値(図8、図9)を示す。

二乗誤差の平均値は Polak-Ribiere-Polyak の共役勾配法と改良共役勾配法でほぼ同じである(図3)のに対し、二乗誤差のばらつきは改良共役勾配法の方が小さい(図4)。これより、Polak-Ribiere-Polyak の共役勾配法は、小さい二乗誤差を得ることもあるが、ローカルミニマにも陥りやすいといえる。改良共役勾配法では、初期値によるばらつきが半分程度に抑えられており、初期値の影響が少ないといえる。以上より、改良共役勾配法では、どのような初期値からも安定して小さな誤差が得られており、学習の不安定さが解消されていることがわかる。

改良共役勾配法は共役方向生成の計算が複雑化しているにもかかわらず、学習時間が少ない(図5)。これは、歩み幅の挙動が影響している。Polak-Ribiere-Polyak の共役勾配法では歩み幅は安定して大きいですが、改良共役勾配法の歩み幅は平均的に小さい(図6～図8)。歩み幅が小さいことは、直線探索にかかる

計算量が少なくなり、これが改良共役勾配法の演算時間の短縮化につながったといえる。

この結果は、共役方向生成の計算量に比べ、直線探索部分の計算量の方が大きく、共役方向生成のための計算の複雑化が全体の計算量増加にはつながっていないことを示している。これは、4.における計算量からの考察の正当性を裏付けるものである。

## 7. ま と め

より広範囲で二次近似するという考えに基づいた共役方向生成公式を提案し、この公式を導入した共役勾配法 BP 学習アルゴリズムを提案した。この方法により、共役勾配法の不安定さが克服でき、さらに学習時間の面でも低減が可能であることが明らかになった。本研究では、この学習の高速化は歩み幅が小さいことに起因すると考えている。

二次近似する範囲を決定するパラメータ  $j$  の設定法を示すことが今後の課題である。

## 参 考 文 献

- [1] Rumelhart, D. E., McClelland, J. L., Hinton, G. E. and PDP・Research Group, 甘利俊一監訳:「PDP モデル」, 産業図書, (1989)

- [2] Kar Yan Tan and Melody Kiang: "Predicting Bank Failures a Neural Network Approach", *Appl. Artif. Intell.*, pp. 265-282, Vol. 4, (1990)
- [3] Timo Sorsa, Heikki N. Koivo and Hannu Koivisto: "Neural Networks in Process Fault Diagnosis", *IEEE Trans. Syst. Man. Cybern.*, pp. 815-825, Vol. 21, No. 4, (1991)
- [4] Charalambous, C.: "Conjugate Gradient Algorithm for Efficient Training of Neural Networks", *IEE, Proc.*, pp. 301-310, Vol. 139, No. 3, (1992)
- [5] 吉田利信: "2次元探索共役勾配法を用いた多層神経回路網の高速学習法", *情報処理学会論文誌*, pp. 988-995, Vol. 31, No. 7, (1990)
- [6] 後藤正幸, 開沼泰隆, 俵 信彦: "共役勾配法によるBP学習について", *日本経営工学会平成5年度秋季全国大会予稿集*, pp. 138-139, (1993)
- [7] Moller, M. F.: "A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning", *Neural Networks*, pp. 525-533, Vol. 6, (1993)
- [8] 齊藤和巳, 中野良平: "3層ニューラルネットワークにおける2階導関数を用いた学習アルゴリズムの高速化", *信学技報*, NC94-7, (1994)
- [9] 板和正敏: 「非線形システムの最適化」, 森北出版, (1986)
- [10] 今野 浩, 山下 浩: 「非線形計画法」, 日科技連, (1978)