

非線形歪み誤差評価に基づいた ニューラルネットワークによるブラインド信号源分離 ～模擬生体信号への適用～

陸門 将也^{*} ・ 和久屋 寛^{**}

Blind Source Separation by Neural Networks Based on Nonlinear Distortion Error Criterion — An Application to Biomimetic Signals —

By

Masaya MUTSUKADO and Hiroshi WAKUYA

Abstract : A biological auditory system has the distinct nature for signal separation. A cocktail party effect is one of well-known examples, and blind source separation (BSS) is developed based on its framework to decompose observed signals into the original source signals. Recently, a new technique to solve this tough problem with a cluster of two-layer neural networks is proposed. Its major feature is a step-by-step estimation of source signals through training for reducing nonlinear distortion error. In order to apply this neuro-based BSS technique in the field of biomedical engineering, several computer simulations are carried out with biomimetic signals. As a result, it is found that the objective source signals are separated successfully, even though the observed signals are provided excessively.

Key words : blind source separation, neural network, nonlinear distortion error, biomimetic signal

1 はじめに

ヒトの聴覚系が有する極めて高度な信号分離能力として、カクテルパーティー効果が知られている。これは、不特定多数の会話や音が我々の耳に届いている状況下でも、聞きたい音だけを分離抽出することが出来るというものである。その工学的模倣技術として、独立成分分析 (Independent Component Analysis: ICA)⁽¹⁾ と呼ばれるものがある。これは、源信号がお互いに独立と仮定し、それらを混合した信号から元の信号を復元する手法で、本来は未知である信号混合行列 (源信号→観測信号) の逆行列を、信号分離行列 (観測信号

→復元信号) として統計的手法で推定している。その長所は、高次統計量を用いることで、信号の独立性だけを手掛かりに実行できることである。これに対して短所は、観測信号数が源信号数以上であること、非ガウス分布でなければならないなどの制約が多いところである。

このような ICA は、前提条件や問題設定に応じて BSS (Blind Source Separation)[†] と BSD (Blind Source Deconvolution) に大別できる。BSS は、時間遅れなしで瞬時に混合される信号に対して用いられ、混合画像の分離⁽¹⁾ などへの適用実績がある。これに対して、BSD は時間遅れありで混合される信号に対して用いられる。

本研究では、BSSに入力と出力の2層から成るニュー

平成 21 年 5 月 1 日受理

^{*} 大学院工学系研究科電気電子工学専攻

^{**} 理工学部電気電子工学科

©佐賀大学理工学部

[†]BSS は、「ブラインド信号源分離」と和訳されることが多い。

ラルネットワークを導入し、この問題を解いていく手法について検討する。その目的は、2層構成にすることで従来法より計算負荷を小さくし、アルゴリズムを簡単化することである。そして、生体信号に適用することでその有効性を確認したい。

以下本稿では、まず 2. で BSS 問題の問題設定を示す。さらに非線形関数を用いた本手法の定式化を行い、動作手順を説明する。次に、3. では模擬生体信号を用いた計算機シミュレーションを行う。そして、4. で生体信号処理へ適用する際の課題について考察し、5. で結論を述べる。

2 ブラインド信号源分離問題

2.1 問題設定

本研究では、様々な割合で混合された信号から、計算論的手法によって元の信号への分離を試みる。このとき、時刻 t における I 個の源信号 $s(t)$ 、 J 個の観測信号 $z(t)$ 、 K 個の復元信号 $x(t)$ を、

$$s(t) = (s_1(t), s_2(t), \dots, s_I(t))^T \quad (1)$$

$$z(t) = (z_1(t), z_2(t), \dots, z_J(t))^T \quad (2)$$

$$x(t) = (x_1(t), x_2(t), \dots, x_K(t))^T \quad (3)$$

とおくと、これらの関係は Fig.1 に示すとおりである。ここで、源信号 $s(t)$ から観測信号 $z(t)$ が生成される混合行列 A を、

$$A = (a_{ji}) = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1I} \\ a_{21} & a_{22} & \dots & a_{2I} \\ \vdots & \vdots & \ddots & \vdots \\ a_{J1} & a_{J2} & \dots & a_{JI} \end{pmatrix} \quad (4)$$

とすると、

$$z_j(t) = \sum_{i=1}^I a_{ji} s_i(t) \quad (5)$$

となる。ただし、源信号数 I と観測信号数 J は $I \leq J$ となるように設定する。

さて、ここでやるべきことは、いかにして観測信号 $z(t)$ から復元信号 $x(t)$ を求めるかである。混合行列 A に対応するものとして分離行列 W を考えると、

$$W = (w_{kj}) = \begin{pmatrix} w_{11} & w_{12} & \dots & w_{1J} \\ w_{21} & w_{22} & \dots & w_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ w_{K1} & w_{K2} & \dots & w_{KJ} \end{pmatrix} \quad (6)$$

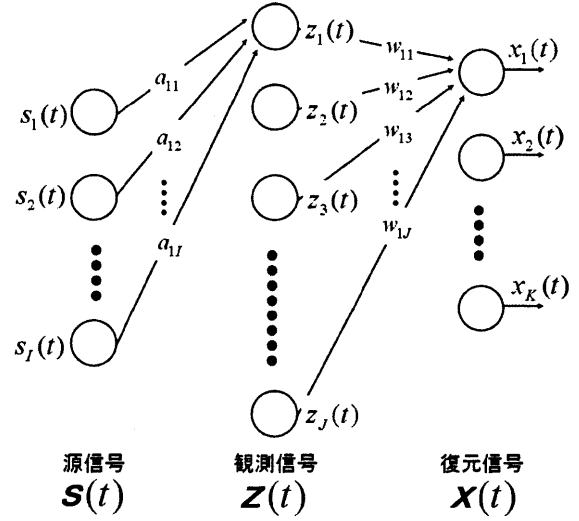


Fig.1 ブラインド信号源分離における源信号、観測信号、復元信号の関係

と表せる。いま、復元信号 $x(t)$ を源信号 $s(t)$ に一致させたいので、

$$W = A^{-1} \quad (7)$$

とすればよい。ただし、上述のとおり、混合行列 A とともに源信号 $s(t)$ が未知であるため、復元信号は順序の入れ替わりと振幅の任意性を許容するのが一般的である。

2.2 定式化

このような問題を解くにあたり、本研究では、観測信号 $z(t)$ から復元信号 $x(t)$ を求める部分を、以下のような階層型ネットワークを用いて実現する⁽²⁾。このとき、観測信号 $z(t)$ を与えるところが入力層、復元信号 $x(t)$ を得られるところが出力層となる。ただし、今回はすべての観測信号 $z(t)$ から復元信号 $x(t)$ を同時に求めるのではなく、 K 個の独立したネットワークを用意して、1つずつ逐次的に求めていく手法を採用する。この様子を表したものが Fig.2 である。このとき k 番目のネットワークの入力信号 $z^{[k]}(t)$ 、重みベクトル $w^{[k]}$ を、

$$z^{[k]}(t) = (z_1^{[k]}(t), z_2^{[k]}(t), \dots, z_J^{[k]}(t))^T \quad (8)$$

$$w^{[k]} = (w_1^{[k]}, w_2^{[k]}, \dots, w_J^{[k]}) \quad (9)$$

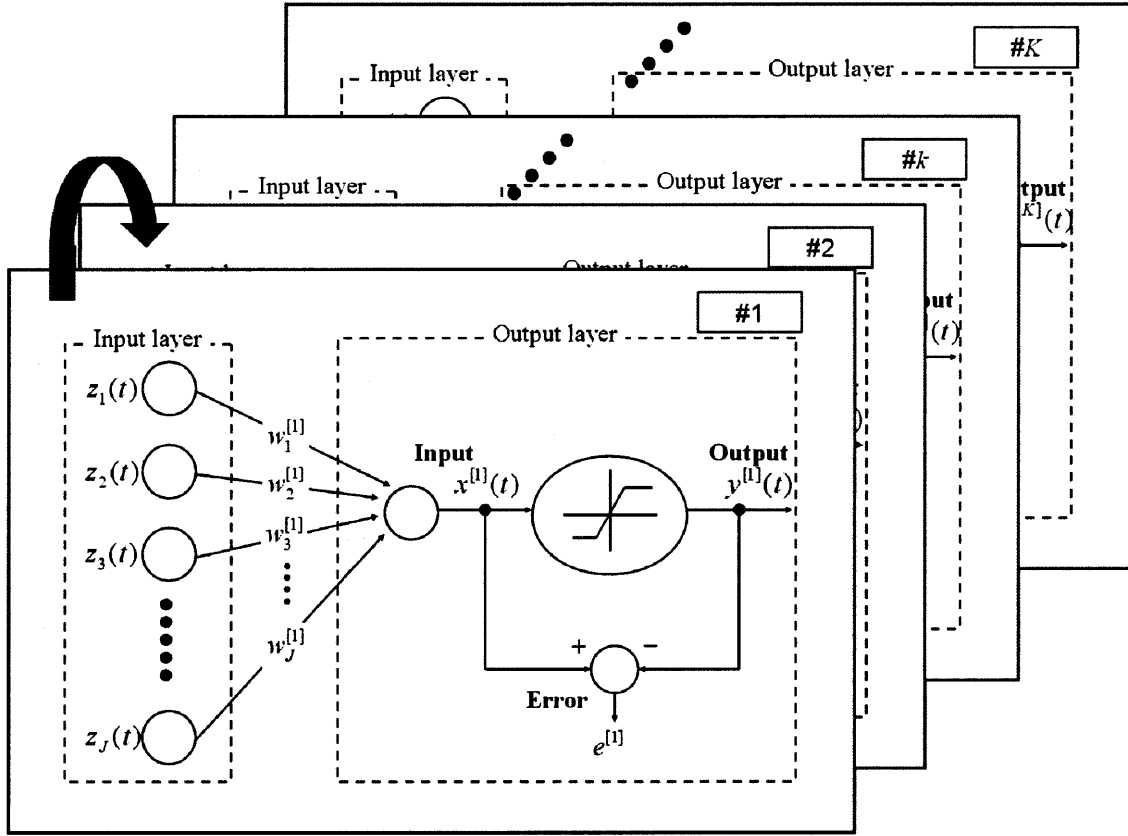


Fig.2 ニューラルネットワークを用いた復元信号の導出過程（概念図）

とすると、出力層ニューロンの活性化レベル $x^{[k]}(t)$ と出力 $y^{[k]}(t)$ は以下の式となる。

$$x^{[k]}(t) = \sum_{j=1}^J w_j^{[k]} z_j^{[k]}(t) \quad (10)$$

$$y^{[k]}(t) = f(x^{[k]}(t)) \quad (11)$$

ここで、 f は非線形関数である。

さて次の問題は、いかにして適切な $w^{[k]}$ を求めるかである。式 (5) によれば、信号混合 A は線形変換であるため、その逆変換である信号分離 W も線形変換のはずである。また、式 (11) の変換関数 f を非線形としているため、これら全ての条件を満足するところは、関数 f を線形近似できる部分、すなわち原点付近ということになる。そこで本研究では、評価尺度として非線形関数によるニューロン出力の歪み誤差を取り上げ、その最小化を行うことで復元信号 $x^{[k]}(t)$ を推定する。これは、出力 $y^{[k]}(t)$ は活性化レベル $x^{[k]}(t)$ が非線形関数 f によって歪められたものと解釈でき、その誤差を最小化することで線形領域に近づけようとするものであ

る。このとき、両者の2乗差をもって歪み誤差を定義し、その期待値を時間平均で置き換えると、

$$\begin{aligned} e^{[k]} &= \frac{1}{2} E \left[\{x^{[k]}(t) - y^{[k]}(t)\}^2 \right] \\ &= \frac{1}{2T} \sum_{t=1}^T \left\{ x^{[k]}(t) - y^{[k]}(t) \right\}^2 \end{aligned} \quad (12)$$

となる。ただし、 T はデータの長さを表す。

次に、式 (12) で定めた評価値が小さくなるように、各重み係数 $w_j^{[k]}$ を最急降下法により更新していく。このときの更新量を $\Delta w_j^{[k]}$ とすると、

$$\Delta w_j^{[k]} = -\eta \frac{\partial e^{[k]}}{\partial w_j^{[k]}} \quad (13)$$

である。ここで、 η は学習率である。ただし、このままでは自由度が大きすぎるため、

$$\overline{\{x^{[k]}(t)\}^2} = \frac{1}{T} \sum_{t=1}^T \left\{ x^{[k]}(t) \right\}^2 = 1 \quad (14)$$

となるような制約条件を $w_j^{[k]}$ に与え、必要に応じて、

$$w_j^{[k]} \leftarrow \frac{w_j^{[k]}}{\sqrt{\sum_{j=1}^J w_j^{[k]} w_j^{[k]}}} \quad (15)$$

という操作によって各重み係数を規格化する。

このように、ネットワーク内で重み係数の更新を続けることで、1つの復元信号 $x^{[k]}(t)$ を導出する。なお、復元信号 $x^{[k]}(t)$ はお互いに独立という制約があるため、 k 番目のネットワークで求める重みベクトル $\mathbf{w}^{[k]}$ は、それ以前の重みベクトル $\mathbf{w}^{[1]} \sim \mathbf{w}^{[k-1]}$ の全てと直交化させながら更新を行うことで、この独立性という制約条件を満足させる。

2.3 前処理

これまでに、BSSの問題設定ならびに本研究で採用するアルゴリズムについて説明してきた。これで全体的な枠組は問題ないが、実際にこの手法を用いる場合には、次の2種類の信号処理を事前に行った方がよい。

- i) 低次元化と直交化
- ii) 中心化と標準化

低次元化と直交化とは、与えられた信号を異なる座標系に変換するものであり、これによって、情報損失を抑えたままで低次元（少ない座標軸から成る座標系）における表現法を求めるものである。しばしば主成分分析（PCA）と呼ばれる手法が用いられている。

これに対して中心化とは、平均が0となるように与えられた信号を線形変換するものである。そして標準化とは、平均や標準偏差が特定の値になるように線形変換することであり、通常は平均を0、標準偏差を1とすることが多い。

3 模擬生体信号を用いた計算機シミュレーション

3.1 実験方法

生体信号とは、生体が様々な活動に伴って発生する信号のことであり、今回は先行研究⁽³⁾と関連づけて筋電信号を採用する。一般に、生体を構成する細胞の内部と外部には異なる種類のイオンが偏って存在しており、そのため、内部は外部に対して-70mV程度の電位となっている。そして、この電位が上昇するとスパイクを発生し、その発生頻度が筋力の大きさに関係することが知られている。

ところで、筋電信号はメキシカンハット型の波形が連続したものとして観測されることが多いため、今回

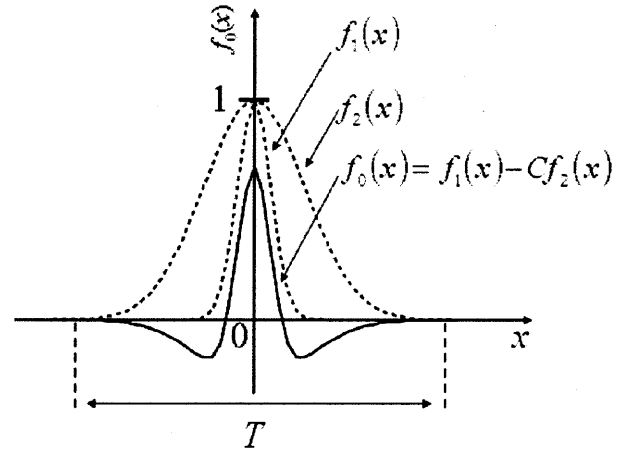


Fig.3 DOG関数による模擬波形の作成法

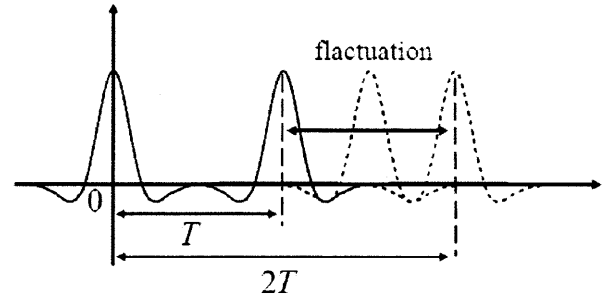


Fig.4 非周期的な模擬波形の作成法

は、DOG (Difference of Gaussian) 関数を用いて模擬信号を作成する。DOG関数 $f_0(x)$ は、2つのガウス関数の差で表されたもので、それぞれを $f_1(x)$ 、 $f_2(x)$ で表すと、

$$f_1(x) = \exp\left(-\frac{x^2}{\sigma_1^2}\right) \quad (16)$$

$$f_2(x) = \exp\left(-\frac{x^2}{\sigma_2^2}\right) \quad (17)$$

$$f_0(x) = f_1(x) - Cf_2(x) \quad (18)$$

となる。ここで、 σ_1 、 σ_2 はそれぞれのガウス関数の広がりを決める定数で、 $\sigma_1 < \sigma_2$ である。 C は1.0未満の任意定数とする。このようにして作成したDOG関数の一例を Fig.3 に示す。また実際の筋電信号に近づけるため、Fig.4 に示すようにDOG関数を連続させ、 $T \sim 2T$ の範囲でばらつかせて模擬波形を作成する。さらに簡単のため、振幅が1となるように事前に規格化しておく。

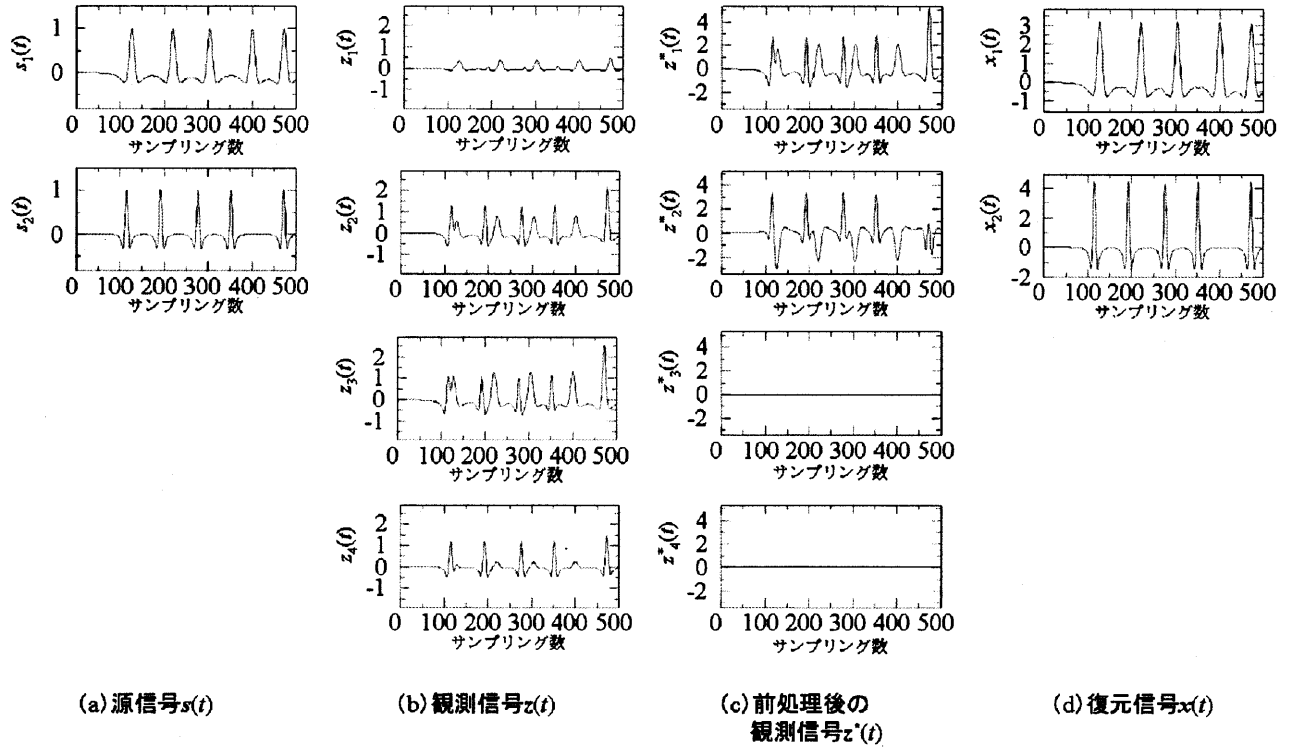


Fig.5 計算機シミュレーションの結果

3.2 実験結果

まず, 3.1 で言及した方法に基づいて, 2 種類の源信号 $s(t)$ を作成する。そして, 事前に用意した混合係数

$$A = (a_{ji}) = \begin{pmatrix} 0.38 & 0.11 \\ 0.81 & 1.39 \\ 1.33 & 1.24 \\ 0.25 & 1.25 \end{pmatrix} \quad (19)$$

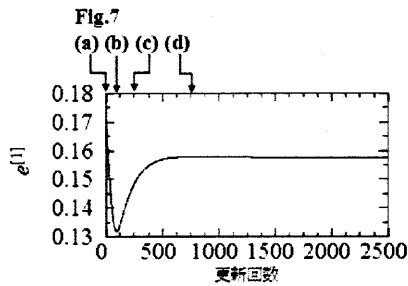
によって 4 種類の観測信号 $z(t)$ を作る。その後, PCA による低次元化と直交化を行い, 続いて中心化, 標準化を行う。このような前処理を施した観測信号 $z^*(t)$ を入力とし, 2.2 で説明したアルゴリズムに基づいて復元信号 $x^{[k]}(t)$ を求める。ただし, 式 (11) の非線形関数 f は \tanh 関数 (双曲線正接関数) とする。このときの結果を示したものが Fig.5 であり, 左から順に, 源信号 $s(t)$, 観測信号 $z(t)$, 前処理後の観測信号 $z^*(t)$, 復元信号 $x(t)$ と配置している。

これによれば, 前処理後の観測信号 $z^*(t)$ の第 3・4 成分はほぼ 0 と見なすことができる。そのため, 当初は 4 種類用意していた観測信号 $z(t)$ は低次元化によって 2 つの成分のみとなり, 冗長な成分が除去されて, 正しく信号の分離が出来ていることが分かる。また, 復

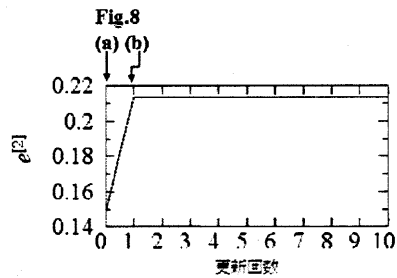
元された信号波形に着目すると, $s_1(t)$ と $x_1(t)$, $s_2(t)$ と $x_2(t)$ がそれぞれ対応している。今回の場合, 復元後の順番に入れ替わりはなく, 復元信号の振幅は $s_1(t)$ と $x_1(t)$ の間で約 3 倍, $s_2(t)$ と $x_2(t)$ の間で約 4 倍に大きくなっている。

次に, このような結果が得られる過程について詳しく調べてみよう。ネットワークの初期結合荷重 $w_j^{[k]}$ は, $[0.00, 1.00]$ の区間でランダムに設定し, $\Delta w_j^{[k]} = 0$ になるまで更新していく。Fig.6 は, この時の各ネットワークの誤差変化を表したものである。なお, 学習率 η は 0.01 と設定する。Fig.6(a) によれば, 1 番目のネットワークの誤差 $e^{[1]}$ は, 約 100 回の学習までは減少していくが, その後一転して増大し, 2669 回の学習によって 0.156 に収束している。また Fig.6(b) によれば, 2 番目のネットワークの誤差 $e^{[2]}$ も増大し, 2 回の学習で一定の値になっている。このように, 歪み誤差の時間変化は奇妙な過程を経ているが, これは式 (15) に相当する規格化の影響であることを, 別途確認済みである。

このとき, 更新回数 N に応じて復元波形がどのように変化するかをまとめたものが Figs.7~8 である。Fig.7 に示す 1 番目の復元信号 $x_1(t)$ は, 学習回数が 0, 100,



(a) 1番目のネットワークの場合



(b) 2番目のネットワークの場合

Fig. 6 結合荷重の更新に伴う歪み誤差の時間変化

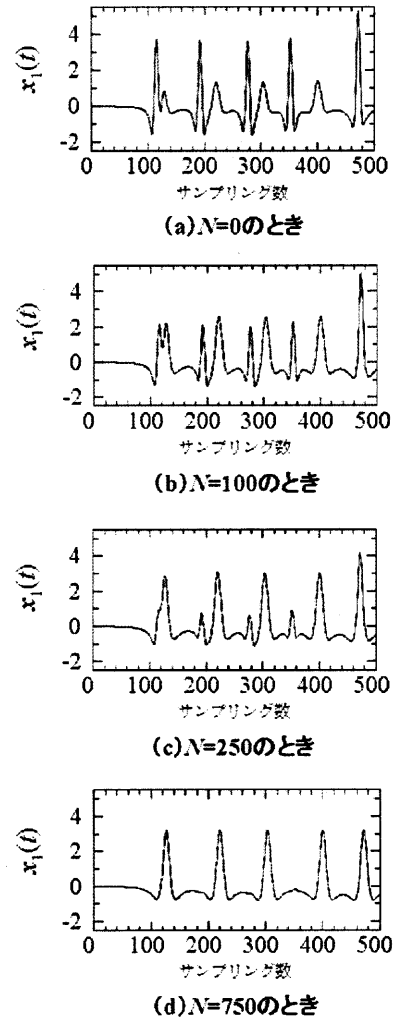
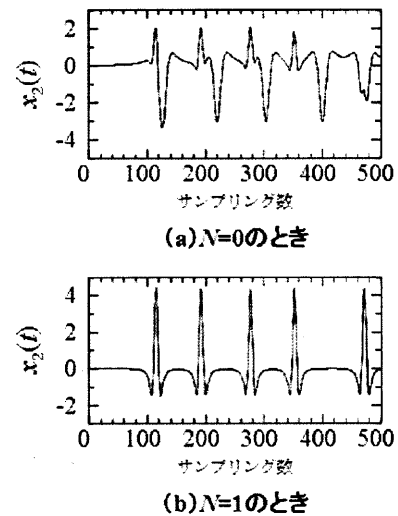
250, 750 回の場合である。連続する 2 つのピークのうち、最初は前のほうが大きいものの、学習が進むとともに抑制されていき、これに代わって後のほうが徐々に目立つようになる。そして、 $N = 750$ 付近で $s_1(t)$ と波形がほぼ一致し、Fig. 6(a) にもあるとおり、歪み誤差の変化が収束している。また、Fig. 8 に示す 2 番目の復元信号 $x_2(t)$ は、学習回数が 0, 1 回の場合である。こちらは、1 回の重み更新で源信号 $s_2(t)$ に似てくるが、これは制約条件で復元信号 $x_1(t)$ と直交するようにしているためだと考えられる。また、最終的に求められる分離行列 W は、

$$W = \begin{pmatrix} w_1^{[1]} & w_1^{[2]} \\ w_2^{[1]} & w_2^{[2]} \end{pmatrix} = \begin{pmatrix} -0.693 & -0.720 \\ 0.720 & -0.693 \end{pmatrix} \quad (20)$$

となった。

4 考察

本研究では、筋電信号を模擬したスパイク列を作成し、これを用いてニューラルネットワークによる信号分離の可能性について検討を行った。源信号より観測信号が多い場合でも、低次元化という前処理によって、適切に対応できることがわかった。一般に、生体信号処理では源信号の個数が未知であることが多く、事前

Fig. 7 1 番目のネットワークの結合荷重更新に伴う復元信号 $x_1(t)$ の変化Fig. 8 2 番目のネットワークの結合荷重更新に伴う復元信号 $x_2(t)$ の変化

に設定しておくことは困難である。したがって、源信号の個数に関する制約を受けないというのは非常に魅力的である。ただし、今回は源信号が2つで、しかも模擬信号を用いている場合に分離可能性が確認されただけである。さらに、条件も極めて緩やかなものとなっており、今後はより現実的な問題に近づけていく努力が必要である。

5 おわりに

本稿では、模擬生体信号を取り上げ、ニューラルネットワークを用いた信号分離を試みた。その結果、本手法の適用可能性が確認できた。ただし、今回は基本的な波形を対象としているため、今後は実際の生体信号で試みるなど、さらなる検討が必要である。

参考文献

- [1] A. ヒバリネン, J. カルネン, E. オヤ : “詳解 独立成分分析 —信号解析の新しい世界”, 東京電機大学出版局, 2005
- [2] 湊原哲也, 古川徹生, 安井湘三 : “非線形歪み誤差の最小化による BSS”, 電子情報通信学会技術研究報告 (非線形問題研究会), NLP2003-122, 2003
- [3] 上川 諒, 和久屋 寛, 斎藤健治 : “独立成分分析を用いた表面筋電信号から運動単位活動電位への分離 ~模擬信号を用いた分離可能性の検討~”, 佐賀大学理工学部集報, Vol.37, No.1, pp.5-15, 2008

付 録

A 動作アルゴリズム

ここでは、2.2 で説明したアルゴリズムについて、より詳細な説明を行う。まずは観測信号 $z(t)$ の荷重和を求め、そして出力信号の歪み誤差が最小となるように学習していけばよい。その時のフローチャートが Fig.9 である。以下、各手順について説明する。

[手順 1]

復元信号を1つずつ逐次的に求めていくため、 $k=1$ とおき、1個目の復元信号 $x^{[1]}(t)$ を導出するネットワークについて考える。

[手順 2]

k 番目のネットワークの重みベクトル

$$w^{[k]} = (w_1^{[k]}, w_2^{[k]}, \dots, w_J^{[k]}) \quad (21)$$

を乱数で決定し、

$$w_j^{[k]} \leftarrow \frac{w_j^{[k]}}{\sqrt{\sum_{j=1}^J w_j^{[k]} w_j^{[k]}}} \quad (22)$$

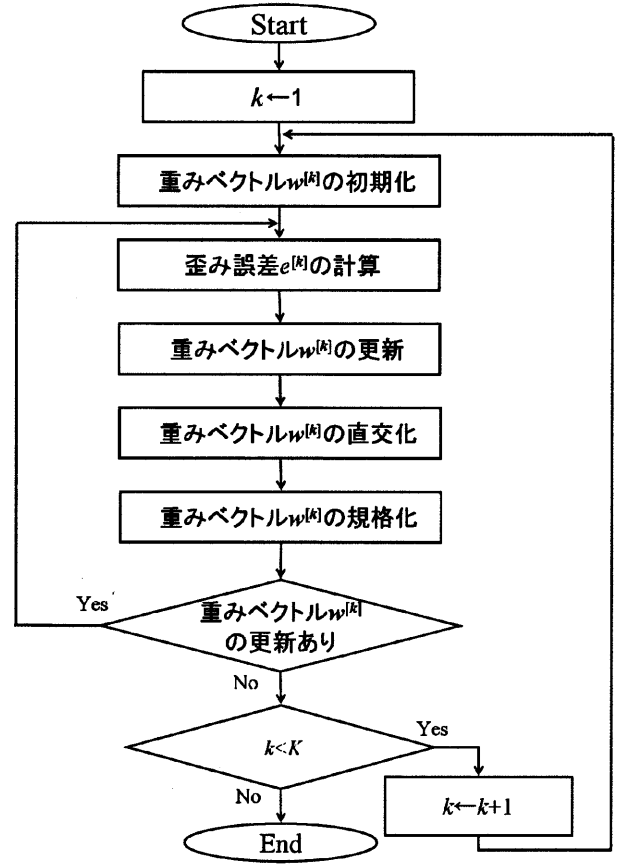


Fig.9 本研究で採用した BSS アルゴリズムのフローチャート

によって大きさが1となるように初期値 $w_j^{[k]}$ を設定する。

[手順 3]

式 (10)~(12) を用いて、全データに対する出力信号の歪み誤差 $e^{[k]}$ を求める。このとき、用いる非線形関数 f は \tanh 関数 (双曲線正接関数) である。

$$\begin{aligned}
 e^{[k]} &= \frac{1}{2} E \left[\left\{ x^{[k]}(t) - y^{[k]}(t) \right\}^2 \right] \\
 &= \frac{1}{2T} \sum_{t=1}^T \left\{ x^{[k]}(t) - f \left(x^{[k]}(t) \right) \right\}^2 \\
 &= \frac{1}{2T} \sum_{t=1}^T \left\{ \sum_{j=1}^J w_j^{[k]} z_j(t) - \tanh \left(\sum_{j=1}^J w_j^{[k]} z_j(t) \right) \right\}^2 \quad (23)
 \end{aligned}$$

[手順 4]

式 (13) で定義した更新量に基づいて、以下のように

重み係数を更新する。

$$\begin{aligned}
 \Delta w_j^{[k]} &= -\eta \frac{\partial e^{[k]}}{\partial w_j^{[k]}} \\
 &= -\eta \frac{1}{T} \sum_{t=1}^T \left\{ \sum_{n=1}^J w_n^{[k]} z_n(t) \right. \\
 &\quad \left. - \tanh \left(\sum_{n=1}^J w_n^{[k]} z_n(t) \right) \right\} \\
 &\quad \times z_j(t) \left\{ 1 - \frac{\partial}{\partial w_j^{[k]}} \tanh \left(\sum_{n=1}^J w_n^{[k]} z_n(t) \right) \right\} \\
 &= -\eta \frac{1}{T} \sum_{t=1}^T \left\{ \sum_{n=1}^J w_n^{[k]} z_n(t) \right. \\
 &\quad \left. - \tanh \left(\sum_{n=1}^J w_n^{[k]} z_n(t) \right) \right\} \\
 &\quad \times z_j(t) \left\{ 1 - \frac{1}{\cosh^2 \left(\sum_{n=1}^J w_n^{[k]} z_n(t) \right)} \right\} \quad (24)
 \end{aligned}$$

【手順 5】

ここまで求められた全ての復元信号 $x_k(t)$ がお互いに独立である為に、シュミットの直交化を用いて、重みベクトル $w^{[k]}$ の逐次直交化を行う。

$$w_n^{[k]} \leftarrow w_n^{[k]} - \sum_{i=1}^{k-1} w_n^{[i]} \left\{ \sum_{j=1}^J w_j^{[k]} w_j^{[i]} \right\} \quad (25)$$

このとき、 $w_j^{[i]}$ に含まれる添字 i は、復元信号の導出に用いられているネットワークの番号 $1 \sim (k-1)$ に対応している。

【手順 6】

さらに、自由度の問題から、

$$w_j^{[k]} \leftarrow \frac{w_j^{[k]}}{\sqrt{\sum_{j=1}^J w_j^{[k]} w_j^{[k]}}} \quad (26)$$

によって制約を課す。

【手順 7】

上述のように、重みベクトル $w^{[k]}$ を出力 $y^{[k]}(t)$ と活性化レベル $x^{[k]}(t)$ の誤差が最小化するまで更新を続ける。このときの判断基準は、重みベクトル $w^{[k]}$ の変化の有無である。つまり、重みベクトルを $w^{[k]}$ が更新されている間は、【手順 3】へ戻る。

【手順 8】

k を 1 つ増やし、 k 番目のネットワークから $(k+1)$ 番目のネットワークへと移る。このとき $k \leq K$ であれ

ば【手順 2】へ戻り、 $k = K$ であれば終了する。

以上のようなアルゴリズムのうち【手順 2】から【手順 8】までを繰り返すことで、所望の信号が得られるように信号分離を試みる。