

# ニューラルネットワークを用いた 風力発電システムのインテリジェント制御

糸井 弦太、 天野 耀鴻

要旨：本論文では、風力発電システムにおけるニューラルネットワークを用いたインテリジェント制御を提案する。風力発電システムには線形的・非線形的なモデル化誤差が存在する。本提案の制御システムでは、風力発電システムの線形的な既知部分に対して適応レギュレータを設計し、非線形なモデル化誤差に対してはニューラルネットワークを用いて同定を行う。それと同時に、このニューラルネットワークにより設計された最適レギュレータの制御入力を適応的に補償する。

キーワード：風力発電、ニューラルネットワーク、非線形のモデル化誤差、パラメータ同定

## 1. はじめに

風のエネルギーを電力に変換する風力発電は、大気は無尽蔵な風を利用して発電するため、火力発電のように有限である化石燃料を必要としない。また排気ガスが少ないため、環境に優しいクリーンなエネルギーを作り出すことが可能である。このような特徴から、近年風力発電はクリーンなエネルギーとして注目を集めている。

この風力発電を高効率化するためには風速の変動に対して、タービンの回転数を最適な回転数に調節する必要がある。このとき回転数を制御するのに、FET(Field Effect Transistor)デバイスを発電ブレーキとして、風速の変動に応じてタービンの回転数を制御する変速制御手法が一つの有効方法である。

しかし、粘性摩擦や発電機の磁石による磁気飽和、回路内部での発熱による抵抗の劣化やコイルの熱膨張など、電力損失を起こす要因によって発電の効率が落ちるといった問題が発生する。これらの要因はすべて非線形なパラメータで表すことができ、風力発電システムの制御性能の向上には、これらの非線形な影響に対する補償が不可欠となる。

本研究では風力発電システムの制御性能向上、及び発電高効率化のために、最適レギュレータによる入力と、非線形パラメータに対してニューラルネットワーク(以下 NN)を用いて補償を行う同定システムを提案する。まず、風力発電システムにおいて非線形要因を含むモデルを構築し、非線形要因について解析する。次に NN を用いた風力発電システムのインテリジェント制御を提案し、制御原理と NN の構成を説明する。最後にシミュレーション実験を行い、最適レギュレータと本提案方法による制御の実験結果を比較し、本提案の有効性と実用性を確認する。

## 2. 風力発電機のモデル

Fig.1 は、本研究で対象とする風力発電機と FET によって構成されたシステムである。このシステムでは、FET によって発電機に流れる電流を変化させ、発電機の回転数の制御を行っている。

FET の等価回路は Fig.2 のようになる。本研究では、FET での電力損失を低減するため、入力信号  $v_g$  に PWM 信号を用いる。この信号は非線形であるが、数学モデルでは線形化されたものとして扱う。

また、Fig.1 の風力発電システムを等価回路に直すと Fig.3 のようになる。この時、 $R_a$  : 電機子抵抗、 $L$

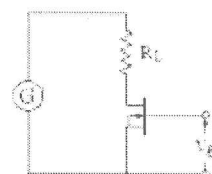


Fig.1 Wind power system

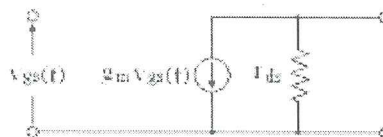


Fig.2 The block diagram of FET

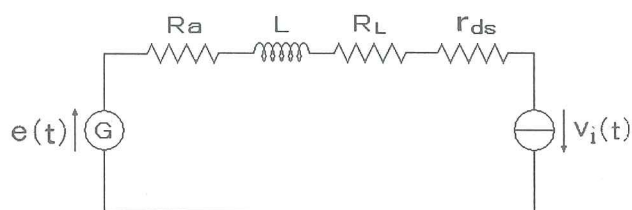


Fig.3 Block diagram of the proposed intelligent adaptive regulator

: 電機子インダクタンス、 $r_{ds}$ : FET の ds 間の抵抗、 $v_i(t)$  は  $g_m r_{ds} v_{gs}$  である。

Fig.3 の等価回路を数式モデルに直すと次のようになる。(R,  $k_t$ ,  $k_b$ , J, D, L=合成抵抗、トルク定数、逆起電圧定数、モータ軸の慣性モーメント、粘性摩擦、インダクタンス)

$$\begin{bmatrix} \dot{\omega} \\ \dot{i} \end{bmatrix} = \begin{bmatrix} -\frac{D}{J} & -\frac{k_t}{J} \\ -\frac{k_b}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \omega \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{L} \end{bmatrix} v_i \quad (1)$$

$$y(t) = [1 \quad 0] \begin{bmatrix} \omega \\ i \end{bmatrix} \quad (2)$$

これらの状態方程式を簡略化すると次のようになる。

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (3)$$

$$y(t) = Cx(t) \quad (4)$$

ここで、 $x(t) \in \mathbb{R}^{2 \times 1}$ ,  $u(t) \in \mathbb{R}^{1 \times 1}$ ,  $y(t) \in \mathbb{R}^{1 \times 1}$  はそれぞれシステムの状態変数、入力、出力であり、また  $A \in \mathbb{R}^{2 \times 2}$ ,  $B \in \mathbb{R}^{2 \times 1}$ ,  $C \in \mathbb{R}^{1 \times 2}$  はそれぞれシステムのパラメータ行列である。しかしながら、実際の風力発電機では動作中に熱が発生する。この熱により、抵抗の劣化による抵抗値の変化や、コイルの銅線の熱膨張によって銅線の長さが増加し、形状の変化が発生する。その結果、インダクタンス値が変化するなど様々な誤差が発生する。それらを加味して(1)式を書き直すと次のようになる。

$$\begin{bmatrix} \dot{\omega} \\ \dot{i} \end{bmatrix} = \begin{bmatrix} -\frac{D + \Delta D}{J} & -\frac{k_t + \Delta k_t}{J} \\ -\frac{k_b}{L} + \Delta \frac{k_b}{L} & -\frac{R}{L} + \Delta \frac{R}{L} \end{bmatrix} \begin{bmatrix} \omega \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{L} \end{bmatrix} v_i \quad (5)$$

$$\begin{bmatrix} \dot{\omega} \\ \dot{i} \end{bmatrix} = \left( \begin{bmatrix} -\frac{D}{J} & -\frac{k_t}{J} \\ -\frac{k_b}{L} & -\frac{R}{L} \end{bmatrix} + \begin{bmatrix} \frac{\Delta D}{J} & \frac{\Delta k_t}{J} \\ \frac{\Delta k_b}{\Delta L} & \frac{\Delta R}{\Delta L} \end{bmatrix} \right) \begin{bmatrix} \omega \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{L} \end{bmatrix} v_i \quad (6)$$

$\Delta$  で表されるパラメータが、発生する誤差である。

このとき、インダクタンスはコイルの巻き数の2乗に比例するため、巻き数が増加するとインダクタンスは Fig.4 のように指数関数的に変化する。そのため、発生する誤差は非線形なものとして表される。

ここで(6)式の状態方程式を簡略化すると次のようになる。

$$\dot{x}(t) = (A_n + \Delta A)x(t) + Bu(t) \quad (7)$$

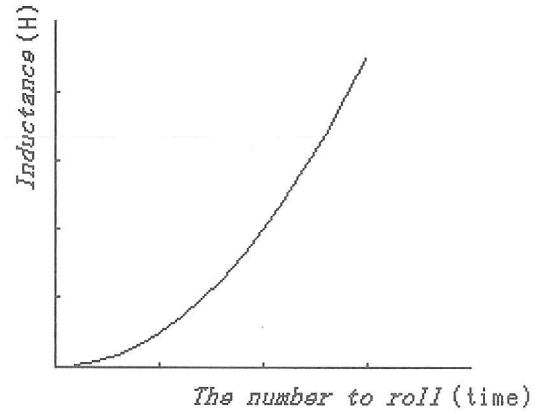


Fig.4 The frequency characteristic of inductance

$$A = A_n + \Delta A \quad (8)$$

$$A_n = \begin{bmatrix} -\frac{D}{J} & -\frac{k_t}{J} \\ -\frac{k_b}{L} & -\frac{R}{L} \end{bmatrix}, \Delta A = \begin{bmatrix} \frac{\Delta D}{J} & \frac{\Delta k_t}{J} \\ \frac{\Delta k_b}{\Delta L} & \frac{\Delta R}{\Delta L} \end{bmatrix} \quad (9)$$

このとき行列  $A_n, \Delta A \in \mathbb{R}^{2 \times 2}$  はそれぞれシステムの既知行列、線形近似の誤差からなるモデル化誤差である。さらに(6)式を書き直すと次のようになる。

$$\dot{x}(t) = Ax(t) + Bu(t) + \Delta_x(t) \quad [\Delta_x(t) = \Delta A x(t)] \quad (10)$$

$\Delta_x$  は非線形な未知パラメータであり、これがモデルと実際のシステムの間で発生する誤差である。

この非線形なパラメータを補償するために、本研究では NN の非線形学習能力によって  $\Delta_x$  を同定する新しいシステムを提案する。

### 3. 提案する制御システム

#### 3.1 システム原理

本研究で提案するこの制御システムは(2), (10)式で表されるシステムの線形部分と未知部分からなるシステムを制御対象とする。制御目的は評価関数

$$J = \int_0^{\infty} [x^T Q x(t) + u^T(t) R u(t)] dt \quad (11)$$

を最小にする最適制御入力を求めるものとする。このとき、

$$R (\in \mathbb{R}^{1 \times 1}) > 0, Q (\in \mathbb{R}^{2 \times 2}) \geq 0 \quad (12)$$

は設計仕様として与えられる重み行列である。

(10)式に対応した非線形レギュレータを解析的に解くことは一般的に困難である。この問題に対して、一般的に線形近似による線形レギュレータ理論を適用する方法が用いられている。しかし、(10)式には非線形な未知パラメータが含まれており、線形近似のみでは不十分な事が多い。このため、非線形のモ

デル化誤差に対する補償を行う必要がある。そこで本研究では、NN の非線形特性及び学習能力によって非線形補償を行うインテリジェント適応レギュレータを提案する。

### 3.2 線形近似の適応レギュレータ

(3)式で表されるシステムが可制御、可観測であると仮定すると、(11)式の評価関数  $J$  を最小にする最適制御入力  $u^*(t)$  は、

$$u^*(t) = -R^{-1}B^T P x(t) \quad (13)$$

与えられる。この時  $P \in \mathfrak{R}^{n \times n}$  はリカッチ方程式

$$PA + A^T P - PBR^{-1}B^T P + Q = 0 \quad (14)$$

を満たす正定唯一解である。

ここで、このリカッチ方程式の解  $P$  を

$$P = P_n + \Delta_p \quad (15)$$

とおく。このとき、

$$P_n \in \mathfrak{R}^{2 \times 2}, \Delta_p \in \mathfrak{R}^{2 \times 2} \quad (16)$$

はそれぞれシステムの既知部分の解、未知部分の補償解である。(8)式、(15)式を(11)式に代入し、補償解  $\Delta_p$ 、モデル化誤差  $\Delta_A$  に関する2次項  $\Delta_p \Delta_A$ 、 $\Delta_A^T \Delta_p$ 、 $\Delta_p \Delta_p$  などを微小ととれば、リカッチ方程式は次のような二つの式に分解することができる。

$$P_n A_n + A_n^T P_n - P_n B R^{-1} B^T P_n + Q = 0 \quad (17)$$

$$\begin{aligned} \Delta_p A_n + P_n \Delta_A + \Delta_A^T P_n + A_n^T \Delta_p \\ - \Delta_p B R^{-1} B^T P_n - P_n B R^{-1} B^T \Delta_p = 0 \end{aligned} \quad (18)$$

このとき解  $P_n$  は(17)式がリカッチ方程式であるため、正定対称解である。(12)式の最適制御入力  $u^*(t)$  を計算するのに必要な  $P$  のうち、 $P_n$  は(15)式から得ることができる。しかし、補償解  $\Delta_p$  は  $\Delta_A$  が未知であるため直接求めることができない。次節においてこの補償解  $\Delta_p$  の推定法を導出する。

### 3.3 リカッチ方程式の補償解の推定法

(18)式を次のように移項する。

$$\begin{aligned} \Delta_p B R^{-1} B^T P_n - \Delta_p A_n \\ + P_n B R^{-1} B^T \Delta_p - A_n^T \Delta_p = P_n \Delta_A + \Delta_A^T P_n \end{aligned} \quad (19)$$

ここで  $F$  と  $D$  を次のようにおく。

$$D = B R^{-1} B^T P_n - A_n, F = P_n B R^{-1} B^T - A \quad (20)$$

すると、(19)式は次のように変形できる。

$$\Delta_p D + F \Delta_p = \Delta_A^T P_n + P_n \Delta_A \quad (21)$$

この(21)式の両辺に対して  $x(t)$  の2次形式を求めると、

$$\begin{aligned} x^T(t) K x(t) = \Delta_x^T(t) P_n x(t) + x^T(t) P_n \Delta_x(t) \\ \Delta_x(t) = \Delta_A x(t), K = \Delta_p D + F \Delta_p \end{aligned} \quad (22)$$

(22)式を未知パラメータ  $\Delta_x$  に関して展開すると、

$$\omega(t) \theta = \lambda(t) \quad (23)$$

$$\theta = [\vartheta_{11}, \dots, \vartheta_{nn}]^T \in \mathfrak{R}^{4 \times 1} \quad (24)$$

$$\omega(t) = [\omega_{11}(t), \dots, \omega_{nn}(t)] \in \mathfrak{R}^{4 \times 1} \quad (25)$$

$$\begin{aligned} \omega_{kh}(t) = \sum_{i=1}^n x_i(t) d_{hi} x_k(t) + \sum_{j=1}^n x_h(t) f_{jk} x_j(t) \\ (k, h = 1, 2, \dots, n) \end{aligned} \quad (26)$$

$$\lambda(t) = \sum_{j=1}^n \sum_{i=1}^n \Delta_{xj}(t) p_{ji} x_i(t) + \sum_{j=1}^n \sum_{i=1}^n x_j(t) p_{ji} \Delta_{xi}(t) \quad (27)$$

となる。この時、 $p_{ij}$ 、 $d_{hi}$ 、 $f_{jk}$  はそれぞれ既知行列  $P_n$ 、 $D$ 、 $F$  の要素、 $\vartheta_{ij}$  は未知のベクトル  $\theta$  の要素で、

$$\Delta_p = [\vartheta_{ij}] \in \mathfrak{R}^{n \times n} \quad (28)$$

である。

いま、サンプリング時間  $\Delta_t$  ごとに  $s$  個のデータを得たとすると、(23)式より

$$\Omega(t) \theta = \Lambda(t) \quad (29)$$

$$\Omega(t) = [\omega(t - \Delta t), \dots, \omega(t - s \Delta t)]^T \in \mathfrak{R}^{s \times n^2} \quad (30)$$

$$\Lambda(t) = [\lambda(t - \Delta t), \dots, \lambda(t - s \Delta t)]^T \in \mathfrak{R}^{s \times 1} \quad (31)$$

を得る。(26)式の  $\omega_{kh}(t)$  は計算で求めることができるため、(27)式の  $\lambda(t)$  が分かれば最小二乗法を用いて  $\theta$  を求めることが可能となる。その場合、最適解は、

$$\theta = \Omega^+(t) \Lambda(t) \quad (32)$$

与えられる。ここで  $\Omega^+(t)$  は擬似逆行列を表す。

しかし、ここで扱われる行列  $\Omega^+(t)$  は特殊な形をしているため、通常の計算では擬似逆行列を求めることができない。この擬似逆行列の計算法を次節において示す。

### 3.4 擬似逆行列の計算法

風力発電システムの状態方程式(1)式を扱う場合、リカッチ方程式の解  $P$  は  $2 \times 2$  行列となるため、 $\omega(t)$  と  $\theta$  はそれぞれ4つの要素を持つことになる。また、リカッチ方程式の解  $P$  は正定対称行列であるため、 $\omega_{12}(t) = \omega_{21}(t)$ 、 $\theta_{12} = \theta_{21}$  となる。すなわち、(29)式は次のような形になる。

$$\Omega(t) \quad \theta = \Lambda(t) \quad (33)$$

$$\begin{bmatrix} a_1 & b_1 & b_1 & c_1 \\ a_2 & b_2 & b_2 & c_2 \\ \vdots & \vdots & \vdots & \vdots \\ a_n & b_n & b_n & c_n \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix}$$

この式において行列 $\omega(t)^T \omega(t)$ を計算すると、2列目と3列目が完全に一致してしまい、逆行列の計算ができなくなる。この問題を解決するために、上式を次のように変形する。

$$\begin{bmatrix} a_1\theta_1 & b_1\theta_2 & b_1\theta_2 & c_1\theta_3 \\ a_2\theta_1 & b_2\theta_2 & b_2\theta_2 & c_2\theta_3 \\ \vdots & \vdots & \vdots & \vdots \\ a_n\theta_1 & b_n\theta_2 & b_n\theta_2 & c_n\theta_3 \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} \quad (34)$$

$$\begin{bmatrix} a_1\theta_1 & 2b_1\theta_2 & c_1\theta_3 \\ a_2\theta_1 & 2b_2\theta_2 & c_2\theta_3 \\ \vdots & \vdots & \vdots \\ a_n\theta_1 & 2b_n\theta_2 & c_n\theta_3 \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} \quad (35)$$

$$\begin{bmatrix} a_1 & 2b_1 & c_1 \\ a_2 & 2b_2 & c_2 \\ \vdots & \vdots & \vdots \\ a_n & 2b_n & c_n \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} \quad (36)$$

(33)式をこのような形に変形することにより、(32)式の擬似逆行列の計算が可能となる。つまり、(27)式の $\lambda(t)$ が計算できれば、最小二乗法によって行列 $\Delta_p$ を推定することが可能となり、それによって制御入力 $u(t)$ を最適入力 $u^*(t)$ に近づけることができる。しかし、モデル化誤差 $\Delta_A$ が未知であるため、(27)式に含まれる $\Delta_x(t)$ を計算することができない。この問題を解決するために、本研究ではニューラルネットワークを導入する。

### 3.5 NNを用いた同定システム

Fig.5 に本研究で提案するインテリジェント適応レギュレータシステムの構成を示す。この Fig.5 におけるパラメータの同定システムを、

$$\dot{\hat{x}}(t) = Ax(t) + Bu(t) + x_{NN}(t) \quad (37)$$

とする。ただし、

$$\dot{\hat{x}}(t) \in \mathfrak{R}^{n \times 1}, x_{NN}(t) \in \mathfrak{R}^{n \times 1} \quad (38)$$

はそれぞれ同定システムの状態変数、NN の出力である。

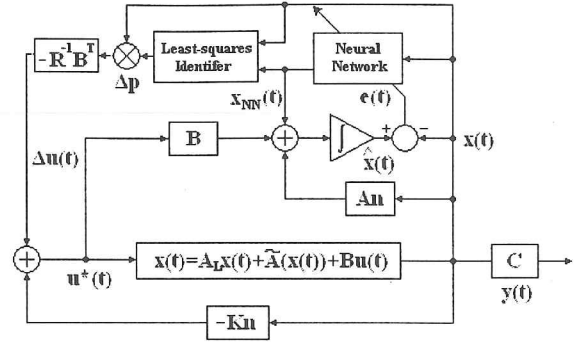


Fig.5 Block diagram of the proposed intelligent adaptive regulator

一方、(15)式を(13)式に代入すると、最適入力 $u^*(t)$ は

$$u^*(t) = -R^{-1}B^T P(P_n + \Delta_p)x(t) = u_n(t) + \Delta_u(t) \quad (39)$$

となる。ただし、 $u_n(t) = -K_n x(t)$ ,  $K_n = R^{-1}B^T P_n$ ,  $\Delta_u(t) = -R^{-1}B^T \Delta_p x(t)$ であり、 $K_n \in \mathfrak{R}^{n \times n}$ は最適レギュレータの状態フィードバックゲインに相当する。このとき(3)式と(37)式より、同定システムの状態変数 $\hat{x}(t)$ と $x(t)$ との同定誤差 $e(t)$ は、

$$e(t) = \hat{x}(t) - x(t) = \int_0^t \zeta(\tau) d\tau \in \mathfrak{R}^{n \times 1} \quad (40)$$

となる。ここで

$$\zeta(\tau) = [x_{NN}(\tau) - \Delta_x(\tau)] \quad (41)$$

は、制御対象のモデル化誤差 $\Delta_A$ に起因する状態変化量 $\Delta_x(t)$ とNNの出力との誤差である。状態変数の同定誤差 $e(t)$ が0の場合でも(38)式の被積分項が0にならないケースが存在することを考慮して、本研究ではNNの学習のための誤差関数 $E(t)$ を

$$E(t) = \frac{1}{2} \dot{e}^T(t) \dot{e}(t) + \frac{1}{2} e^T(t) e(t) = E^{(1)}(t) + E^{(2)}(t) \quad (42)$$

$$E^{(1)}(t) = \frac{1}{2} \dot{\zeta}^T(t) \dot{\zeta}(t), \quad E^{(2)}(t) = \frac{1}{2} e^T(t) e(t) \quad (43)$$

と定義し、この誤差関数が最小になるようにNNの学習を行う。そして、NNの学習が進んで同定誤差が0になったとき、同定システムの状態変数 $\hat{x}(t)$ は制御対象の状態変数 $x(t)$ と一致し、NNの出力 $x_{NN}(t)$ は状態変化量 $\Delta_x(t)$ に一致する。すなわち、

$$x_{NN}(t) = \Delta_x(t) \quad (44)$$

となる。これによりNNの学習に伴って、本手法で構成される制御対象への入力が最適入力(39)式に

徐々に近づいていくのである。

### 3.6 NNの構成

本研究で使用する三層の階層型NNの構成をFig.6に示す。中間層の重み行列と出力層の重み行列はそれぞれ、

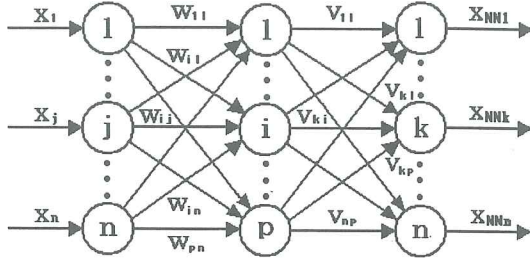


Fig.6 Structure of the used neural network

$$W(t) \in \mathfrak{R}^{p \times n}, \quad V(t) \in \mathfrak{R}^{n \times p} \quad (45)$$

と書くことにする。また Fig.5 に対応して、NNの入力、出力ベクトルはそれぞれ  $x(t)$ 、 $x_{NN}$  となる。入力

層の  $j$  番目のユニットの出力  $I_j$  は

$$I_j = x_j \quad (j=1, \dots, n) \quad (46)$$

とする。中間層の  $i$  番目のユニットの出力  $H_i$  は

$$H_i = \sigma(s_i), \quad s_i = \sum_{j=1}^n w_{ij} I_j \quad (47)$$

で与えられる。また、 $\sigma(\cdot)$  はシグモイド関数である。同様に。出力層のユニット  $O_k$  は

$$O_k = \sum_{i=1}^p v_{ki} h_i \quad (48)$$

である。

学習は(42)式の誤差関数  $E(t)$  を重み  $w_{ij}$  および  $v_{ki}$  に関して最小化するように行う。誤差逆伝播学習法により重みの更新即は

$$v_{kj}(t + \Delta t) = v_{kj} - \eta \left[ \frac{\partial E^{(1)}(t)}{\partial v_{kj}(t)} + \frac{\partial E^{(2)}(t)}{\partial v_{kj}(t)} \right] \quad (49)$$

$$w_{kj}(t + \Delta t) = w_{kj} - \eta \left[ \frac{\partial E^{(1)}(t)}{\partial w_{kj}(t)} + \frac{\partial E^{(2)}(t)}{\partial w_{kj}(t)} \right] \quad (50)$$

で与えられる。ここで  $\eta > 0$  は学習係数、 $\Delta t$  は学習を行う時間間隔である。

同定誤差の微分

$$\zeta(t) = \dot{e}(t) \quad (51)$$

を用いた誤差関数  $E^{(1)}(t)$  は

$$E^{(1)}(t) = \frac{1}{2} \zeta^T(t) \zeta(t) = \frac{1}{2} \sum_{q=1}^n [x_{NNq}(t) - \Delta_{x_q}(t)]^2 \quad (52)$$

と書くことができる。ここで  $x_{NNq}(t)$ 、 $\Delta_{x_q}(t)$  はそれぞれ NN の出力  $x_{NN}(t)$  と状態変化量  $\Delta_x(t)$  の要素である。

(40)式、(52)式より  $\partial E^{(1)}(t) / \partial v_{ki}(t)$  は、

$$\frac{\partial E^{(1)}(t)}{\partial v_{ki}(t)} = \dot{e}(t) \frac{\partial x_{NNk}(t)}{\partial v_{ki}(t)} = \dot{e}(t) \sigma'(\kappa_k) H_i \quad (53)$$

となる。ただし、 $\dot{e}_k(t)$  はベクトル  $\dot{e}(t)$  の要素、 $\sigma'(\cdot)$  は  $\sigma(\cdot)$  の導関数を表す。同様に  $E^{(1)}(t) / \partial w_{ki}(t)$  は次式となる。

$$\frac{\partial E^{(1)}(t)}{\partial w_{ki}(t)} = \sum_{q=1}^m \dot{e}_q(t) \sigma'(\kappa_q) \frac{\partial}{\partial w_{ij}(t)} \left[ \sum_{i=1}^p v_{qi} H_i \right] \quad (54)$$

$$= \sum_{q=1}^m \dot{e}_q(t) \sigma'(\kappa_q) v_{qi} \sigma'(s_i) x_j(t) \quad (55)$$

一方、同定誤差  $e(t)$  を用いた誤差関数  $E^{(2)}(t)$  は

$$E^{(2)}(t) = \frac{1}{2} e^T(t) e(t) = \frac{1}{2} \sum_{k=1}^n [\hat{x}_k(t) - x_{kq}(t)]^2 \quad (56)$$

と展開できるから、 $\partial E^{(2)}(t) / \partial v_{ki}(t)$  は、

$$\frac{\partial E^{(2)}(t)}{\partial v_{ki}(t)} = e_k(t) \frac{\partial e_{k_k}(t)}{\partial x_{NNk}(t)} \frac{\partial x_{NNk}(t)}{\partial v_{ki}(t)} \quad (57)$$

となる。ただし  $e_k(t)$  はベクトル  $e(t)$  の要素を表す。ここで、 $\partial e_k(t) / \partial x_{NNq}(t)$  を微小な変化量を用いて

$$\frac{\partial e_k(t)}{\partial x_{NNk}(t)} \approx \frac{\Delta \partial e_k(t)}{\Delta x_{NNk}(t)} \quad (58)$$

のように近似する。 $x_{NNk}(t)$  を微小量  $\Delta x_{NNk}(t)$  だけ変化させたとき、 $e_k(t)$  の変化  $\Delta e_k(t)$  は

$$\begin{aligned} \Delta e_k(t) &\approx \sum_{j=0}^{N_t} [x_{NNk}(j\Delta t_s) - \Delta x_k(j\Delta t_s) + \Delta x_{NNk}] \Delta t_s \\ &\quad - \sum_{j=0}^{N_t} [x_{NNk}(j\Delta t_s) - \Delta x_k(j\Delta t_s)] \Delta t_s \\ &= \Delta x_{NNk}(t) \Delta t_s \end{aligned} \quad (59)$$

となるから、結局

$$\frac{\partial e_k(t)}{\partial x_{NNk}(t)} \approx \Delta t_s \quad (60)$$

と近似することができる。ただし  $\Delta t_s$  は十分小さなサンプリングタイムで、 $t = N_t \Delta t_s$  である。上式を(57)式に代入すると、

$$\frac{\partial E^{(2)}(t)}{\partial v_{ki}(t)} \approx e_k(t) \Delta t_s \frac{\partial x_{NN_k}(t)}{\partial v_{ki}(t)} \quad (61)$$

となる。同様に  $\partial E^{(2)}(t)/\partial w_{ki}(t)$  は、

$$\frac{\partial E^{(2)}(t)}{\partial w_{ki}(t)} \approx \sum_{q=1}^n e_q(t) \frac{\partial x_{NN_q}(t)}{\partial w_{ij}(t)} \quad (62)$$

で与えられる。

以上より、(49),(50)式の重みの更新則は

$$v_{kj}(t + \Delta t) \approx v_{kj} - \eta \left[ \frac{\partial E^{(2)}(t)}{\partial v_{ki}(t)} + e_k(t) \Delta t_s \right] \frac{\partial x_{NN_k}(t)}{\partial v_{ki}(t)} \quad (63)$$

$$w_{kj}(t + \Delta t) \approx w_{kj} - \eta \sum_{q=1}^n \left[ \frac{\partial E^{(2)}(t)}{\partial w_{ij}(t)} + e_q(t) \Delta t_s \right] \frac{\partial x_{NN_q}(t)}{\partial w_{ij}(t)} \quad (64)$$

となる。

#### 4. MATLAB によるシミュレーション

本研究で提案した制御システムの有用性を確認するため、MATLAB でのシミュレーション実験を行った。実験内容は、ある風力の最適回転数が 300rpm であると仮定して、最適レギュレータと本提案のシステムによる追従制御の比較を行うものである。

実験でのインテリジェント適応レギュレータの構成図を Fig.7 に示す。このとき実験のパラメータ設定は目標回転数を 300rpm、三層の NN においては、NN ブロック内にある S-Function のパラメータ設定において、入力層と出力層のユニット数をそれぞれ 1 個、中間層のユニット数を 10 個、サンプリング時間  $\Delta t_s = 0.1s$  とした。学習係数  $\eta$  については、いくつかのパターンを比較した結果、 $\eta = 0.025$  が最適であると判断した。また、与えられた数学モデルに対する非線形誤差は前述したコイルの熱特性などを考慮し、次式のような指数関数的なものとした。このとき、 $M$  は定数、 $t$  は時間である。

$$\Delta_A = A_n \times (M \cdot t)^2 \quad (65)$$

$$\Delta_A = A_n \times (M \cdot t)^3 \quad (66)$$

$$\Delta_A = A_n \times M \cdot e^{nt} \quad (67)$$

まず、(65)式の非線形誤差を与えたときのシミュレーション結果を Fig.8 に示す。グラフにおいて、実線がインテリジェント適応レギュレータによる制御出力、点線が従来の最適レギュレータによる制御出力である。これを見ると、最適レギュレータによる制御では、実験開始から 5 秒後に一度目標回転数 300rpm に収束したように見えるが、実験開始から約 30 秒後に発散しており、目標回転数で安定していない。それに対して、インテリジェント適応レギュレ

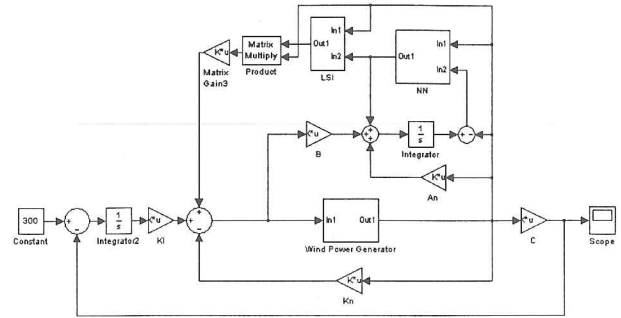


Fig.7 The simulink model for simulation

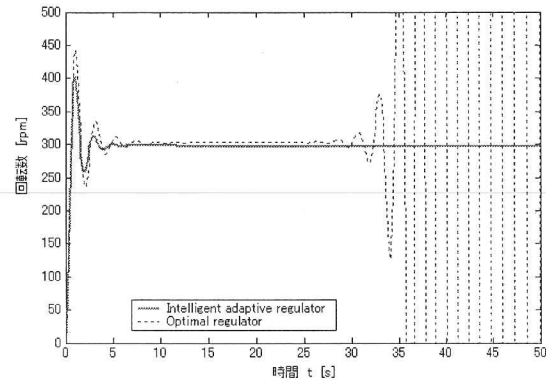


Fig.8 The simulation result of (a) [ $M=0.02$ ]

ータによる制御では、実験開始から約 5 秒後に目標回転数に収束した後、発散を起こさずに安定しているのがわかる。このとき、実験開始から収束するまでの約 5 秒間の振動は非線形パラメータに対する NN の同定において、同定誤差が 0 になっていないためである。これが時間経過による NN の学習によって同定誤差が 0 に近づいていくため、出力は目標回転数に収束して安定する。

(66)式におけるシミュレーション結果 Fig.9 においても同様に、最適レギュレータによる制御では実験開始から約 40 秒で発散を起こしているのに対し、インテリジェント適応レギュレータによる制御では制御出力は目標回転数に収束した後、発散せずに安定しているのがわかる。

また、(67)式の非線形誤差を与えたときのシミュレーション結果を Fig.10 に示す。Fig.10 では、最適レギュレータの制御出力は常に発散しているが、インテリジェント適応レギュレータの制御出力は目標回転数に収束して安定している。

これらの実験結果より、NN による非線形誤差の同定システムが有効であること、また従来の最適レギュレータに比べ制御性能が向上していることが確認できる。

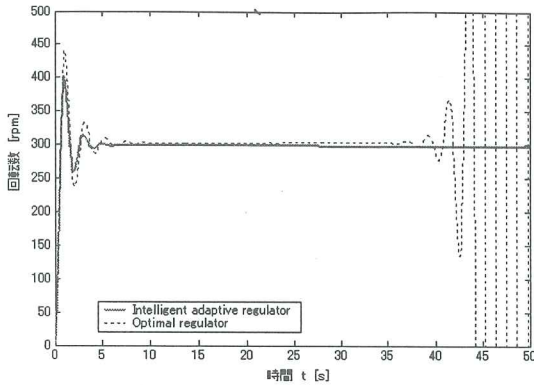


Fig.9 The simulation result of (b) [ $M=0.01$ ]

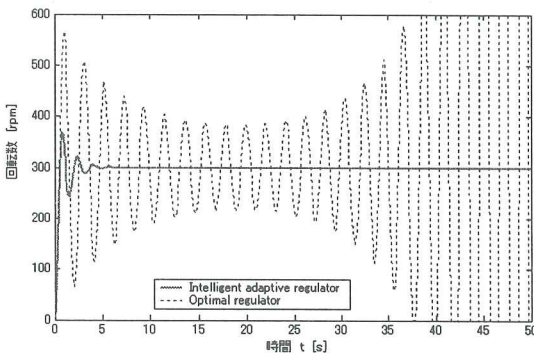


Fig.10 The simulation result of (c) [ $M=0.7, n=0.01$ ]

## 5. おわりに

本研究では風力発電システムの高効率化のため、NN の非線形に対する学習能力を生かしたインテリジェント適応レギュレータシステムを提案し、実際に MATLAB によるシミュレーション実験を行うことで、指数関数的な非線形誤差を含むシステムの制御における、インテリジェント適応レギュレータシステムの有用性を確認した。シミュレーション実験では、非線形誤差に対する NN を用いた同定システムが有効であることを確認したが、この NN のパラメータは任意に設定することが可能であるため、ユニットの重みの初期値、NN のユニット数、学習係数などのパラメータを変更することで、より完成度の高い同定システムを構築することが可能であると考えられる。また、学習係数を可変にしたり、慣性項を考慮するなどのアルゴリズムの改良による学習の高速化についても検討する必要がある。

しかし、実際の制御における効果については実験によってさらなる検証が必要である。以上のことを踏まえて、今後は実際に風力発電機を用いての実機実験により、本研究の実用性を検証する予定である。

## 参考文献：

- 1) 徐 炳鴻, 辻 敏夫, 浜津 恭治, 飯島 雅彦, 「ニューラルネットワークを用いた適応レギュレータの構成法」, 電気学会論文誌 D, Vol.119-6, 809-815, (1999)
- 2) 吉見 浩司, 「高効率の風力発電システムに関する研究」, 平成 16 年度電気学会全国大会, Vol.16, pp20, (2004)
- 4) David Pleter Molmaar, “Cost-Effective Design and Operation of Variable speed Wind turbines; closing the Gap Between the Control Engineering and the Wind Engineering Community”, Delt Univ1.Press (2003)
- 5) 平野 広美, 「C でつくるニューラルネットワーク」, パーソナルメディア株式会社(1991)
- 6) 中野 馨, 飯沼 一元, ニューロンネットワークグループ, 桐谷 滋, 「入門と実習 ニューロコンピュータ」, 株式会社技術評論社(1991)
- 7) 松本 元, 大津 展之, 「脳とコンピュータ 2 ニューロコンピューティングの周辺」, 株式会社培風館(1991)
- 8) 牛山 泉, 「風車工学入門 基礎理論から風力発電技術まで」, 森北出版株式会社(2002)
- 9) 小玉 成人, 松坂 知行, 猪股 登, 「風力発電機に作用する風速の推定法」, 電気学会論文誌 B, Vol.120-7, PP1018~1019 (2000)
- 10) 片山 徹, システム同定入門, 朝倉書店(1994)
- 11) 徐 炳鴻, 辻 敏夫, 金子 真, ニューラルネットワークを利用したモデル化誤差を含むプラントの同定と制御. 計測自動制御学会論文集, Vol32, No.4, pp510-516(1996)
- 12) 志水 清孝, ニューラルネットワークと制御, コロナ社

## AN INTELLIGENT CONTROL USING A NEURAL NETWORK

Genta ITOI, Yoko AMANO

**ABSTRACT** : This paper proposes an intelligent adaptive regulator using a neural network for wind power system. There are linear and nonlinear modeling errors in wind powers systems, the proposed method, the conventional optimal regulator is designed based on a known linear part of the wind power system, and the modeling errors included in the wind power system are identified using the neural network. At the same time, the neural network adaptively compensates a control input computed by the predesigned optimal regulator.

*Key Words* : *Wind power system, Neural network, Nonlinear parameter, Parameter identification*

---