論文

最急降下法によるファジィ規則の 自動チューニング手法[†]

石 岩^{*1} 水本 雅晴^{*1} 湯場崎 直養^{*2} 大谷 正幸^{*2}

ファジィ規則の自動生成方法として、野村らの提案した三角型メンバーシップ関数を用いたデルタルールによるファジィ推論の自動チューニング手法がある。この方法を用いると、多入力の場合、チューニングの対象となる前件部のメンバーシップ関数のパラメータ数が多くなり、ファジィ規則表の形は崩れてしまい、ファジィ規則がわかりにくくなるという問題点がある。この点を解消するために、本論文では、前件部のメンバーシップ関数を入力変数ごとに設定することにより、多入力の場合でも、各入力次元でのメンバーシップ関数のパラメータ数が急増せずに、ファジィ規則表の形でチューニングできるという学習アルゴリズムを提案する。これにより、汎用性を持って人間にわかりやすい形で表示できるファジィ規則表の自動生成を本学習アルゴリズムの下で構築できる。また、幾つかの数値例によって、従来法との比較・検討を行うと共に、本手法の有効性を示す。

キーワード：簡略化ファジィ推論、三角型メンバーシップ関数、最急降下法、自動チューニング手法

1. はじめに

“IF…THEN…”形式のファジィ推論規則を用いて、ファジィ制御やファジィエキスパートシステムなどの分野向けの応用研究が盛んに行われている。従来は、ファジィモデルを構築する時、専門家や熟練者の持つ知識を基にして、試行錯誤的にそのファジィ規則の決定を行う場合がほとんどであった。しかしながら、応用対象によっては、直接にその推論規則とそのメンバーシップ関数を与えるのが困難な場合がある。このような問題に対して、ニューラルネットワーク学習アルゴリズムによる種々の自動チューニング手法が提案されてきた[1-6]。これらの手法により、与えられた入出力データからそれぞれの学習アルゴリズムの下で自動的にファジィ規則を調整したり、生成することができる。その内、野村ら[4-6]の提案した「デルタルールによる自動チューニング手法」が注目されている。この方法は、前件部のメンバーシップ関数をファジィ規則ごとに独立に設定し、与えられた入出力データから前件部のメンバーシップ関数のパラメータと後件部の実数値を自動的にチューニングし、汎化能力の高い推論規則を短時間で得ようとするものである。

本論文では、多入力の場合に、彼らの手法を用いるとチューニングの対象となる前件部のメンバーシップ

関数のパラメータ数が多くなり、大規模なファジィシステムに適用するのが困難となり、しかも、この手法により自動生成したファジィ推論規則は通常のファジィ規則表の形にならなくて、非発火現象が生じてくることを示す。これらの問題点を改善するために、本論文では、最急降下法によってファジィ規則表をチューニングするという新しい学習アルゴリズムを提案する。

従来法と違って、提案する手法では、前件部のメンバーシップ関数が入力次元ごとに設定され、多入力の場合でも、そのパラメータ数が急増しないという特徴を持ち、ファジィ規則表の形は崩れずに前件部のメンバーシップ関数のパラメータと後件部の実数値のチューニングを行う。これにより、汎用性を持って人間にわかりやすい形で表現できるファジィ規則表の自動チューニングを本学習アルゴリズムの下で構築できる。

ここでは、まず、従来の自動チューニング手法を述べ、その問題点を議論する。次に、提案する手法の学習アルゴリズムを与える。また、関数の同定の例を挙げて、従来の自動チューニング手法との比較・検討を行うと共に、本学習アルゴリズムの有効性を示す。

2. デルタルールによるファジィ推論の 自動チューニング手法

野村らの提案した「デルタルールによるファジィ推論の自動チューニング手法」という従来法を説明し、その問題点を議論する。

2.1 チューニングアルゴリズム

まず、簡略化ファジィ推論を用いて、前件部の二等辺三角型メンバーシップ関数のパラメータと後件部の実数値を自動的にチューニングする従来法[4-6]を簡

[†] A Self-Tuning Method of Fuzzy Rules Based on Gradient Descent Method

Yan SHI, Masaharu MIZUMOTO, Naoyoshi YUBAZAKI and Masayuki OTANI

^{*1} 大阪電気通信大学 情報工学部
Division of Information and Computer Sciences, Osaka Electro-Communication University

^{*2} マイコム株式会社 中央研究部
Department of Research and Development, Mycom Inc.

単に説明すると、以下のようになる。

入力を $x_1, x_2, \dots, x_m$ とし、出力を y とすると、簡略化ファジィ推論[1]のファジィ規則は式(1)のように表わされる。

規則 i :

If x_1 is A_{1i} and x_2 is $A_{2i} \dots$ and x_m is A_{mi}
Then y is y_i $i=1, 2, \dots, n$ (1)

ここで、 $A_{ji} (j=1, 2, \dots, m; i=1, 2, \dots, n)$ は前件部メンバーシップ関数で、 y_i は後件部の実数値を表す。但し、前件部メンバーシップ関数 A_{ji} は、推論規則ごとに独立に設定されている。

推論結果 y は次のように求められる。

$$\mu_i = A_{1i}(x_1) \cdot A_{2i}(x_2) \cdot \dots \cdot A_{mi}(x_m) \quad (2)$$

$$y = \frac{\sum_{i=1}^n \mu_i y_i}{\sum_{i=1}^n \mu_i} \quad (3)$$

但し、式(2)の $\mu_i (i=1, 2, \dots, n)$ は前件部の適合度を表わす。

使用する前件部のメンバーシップ関数 $A_{ji} (j=1, 2, \dots, m; i=1, 2, \dots, n)$ は二等辺三角型であり、中心 a_{ji} および幅 b_{ji} を用いて次のように表わされる。

$$A_{ji}(x_j) = \begin{cases} 1 - \frac{2|x_j - a_{ji}|}{b_{ji}}, & a_{ji} - b_{ji}/2 \leq x_j \leq a_{ji} + b_{ji}/2; \\ 0, & \text{その他} \end{cases} \quad (4)$$

$m+1$ 次元の入出力データ $(x_1, \dots, x_m, y^*)$ が与えられたとする。デルタルール学習は次式の評価関数 E の最小化問題として定式化できることが知られている。

$$E = \frac{1}{2} (y^* - y)^2 \quad (5)$$

ここで、 y^* は理想出力値で、 y はファジィ推論値である。

評価関数 E の値を減少させるために、最急降下法により各パラメータ $a_{ji}, b_{ji}, y_i (j=1, 2, \dots, m; i=1, 2, \dots, n)$ に対する自動チューニングアルゴリズムは次のように与えられる。

$$a_{ji}(t+1) = a_{ji}(t) - K_a \partial E / \partial a_{ji}(t) \quad (6)$$

$$b_{ji}(t+1) = b_{ji}(t) - K_b \partial E / \partial b_{ji}(t) \quad (7)$$

$$y_i(t+1) = y_i(t) - K_y \partial E / \partial y_i(t) \quad (8)$$

ここで、 K_a, K_b, K_y は学習係数であり、 t は学習回数を表わす。

これより、式(1)-(5)によって、野村らの提案したデルタルールによるファジィ推論の自動チューニング手法は次のようになる。

$$a_{ji}(t+1) = a_{ji}(t) + \frac{2 K_a (y^* - y) (y_i(t) - y) \operatorname{sgn}(x_j - a_{ji}(t)) \sum_{k \neq j}^m A_{ki}(x_k)}{b_{ji}(t) \sum_{i=1}^n \mu_i} \quad (9)$$

$$b_{ji}(t+1) = b_{ji}(t) + \frac{2 K_b (y^* - y) (y_i(t) - y) |x_j - a_{ji}(t)| \sum_{k \neq j}^m A_{ki}(x_k)}{b_{ji}(t)^2 \sum_{i=1}^n \mu_i} \quad (10)$$

$$y_i(t+1) = y_i(t) + K_y (y^* - y) \mu_i / \left(\sum_{i=1}^n \mu_i \right) \quad (11)$$

但し、式(9), (10)の $k \neq j$ は $k=1, \dots, j-1, j+1, \dots, m$ を意味し、式(9)の sgn は次の符号関数を表している。

$$\operatorname{sgn}(x) = \begin{cases} -1, & x < 0; \\ 0, & x = 0; \\ 1, & x > 0. \end{cases} \quad (12)$$

この自動チューニング手法は、前件部のメンバーシップ関数を推論規則ごとに独立に設定し、与えられた入出力データに適合するメンバーシップ関数のみをチューニングするという特徴を持つため、適当な数の教師データに対してうまくフィッティングできる。これにより、短い時間で推論規則の構築が可能である。また、この自動チューニング手法を多次元へ拡張するのは簡単であり、複雑な応用に適用できることが考えられる。

この手法を用いて、移動ロボットの動的障害物回避問題に適用した例などが報告されている[4-6]。また、この手法に基づいて種々のファジィ規則のチューニング手法も提案されている[7-8]。

しかしながら、以下に示すように、多入力の場合に、この手法を用いると、前件部のメンバーシップ関数のパラメータの数が急増し、しかも、自動生成されたファジィ推論規則は通常のファジィ規則表の形にならず、非発火現象が生じてくるという問題点がある。

2.2 チューニングアルゴリズムの問題点

問題点1：前件部のメンバーシップ関数は推論規則ごとに独立に設定されているので、各入力次元のメンバーシップ関数の数はファジィ規則の数と等しいことが式(1), (9)-(11)からわかる。これにより、多入力の場合、ファジィ規則の数が増えると、前件部のメンバーシップ関数の数も増加すると共に、チューニングされるパラメータ数が非常に多くなってしまうという問題点を持つ。

例えば、2入力1出力ファジィモデルに対する学習

を始める前に、従来法の初期設定方法[4-6]によって、各入力変数に対して、3個ずつメンバーシップ関数を初期設定し、ファジィ規則の数を合計 $3^2=9$ 個とした場合、アルゴリズム(9)-(11)により学習した後は、各入力次元でのメンバーシップ関数の数は3個から9個になってしまい、ファジィ規則の数と同じとなる。

また、4入力の場合、各入力変数に対して、5個ずつメンバーシップ関数を初期設定し、ファジィ規則の数を合計 $5^4=625$ 個とした場合[4,6]、式(9)-(11)により学習した後は、各入力次元でのメンバーシップ関数の数は625個となり、やはり、ファジィ規則の数と同じとなり、急増する。これは、多入力の場合、従来法を用いるとチューニングの対象となるパラメータの数が急増し(上の例では、パラメータの数は $625 \times 2 \times 4 + 625 = 5625$ 個となる)、計算時間がかかり、学習が非常に困難になることを意味している。

問題点2：前件部のメンバーシップ関数はファジィ規則ごとに独立に存在するので、それらのファジィ規則を規則表の形で表わす場合、通常の規則表のイメージに合わないという問題点がある。例えば、2入力の場合、9個の推論規則を持つファジィモデルに対するアルゴリズム(9)-(11)により学習した後は、そのファジィ規則表は例えば図1のような形になる。ここで、 R_i ($i=1,2,\dots,9$)はファジィ規則での存在範囲を示し、それ以外の箇所にはファジィ規則は存在しないことを表わしている。これにより、従来法では、同じ入力次元においても、メンバーシップ関数が重複的に使えないので、未知入力データによっては非発火現象を生じるという問題点を有している。

例えば、図1で、入力 x_1, x_2 がそれぞれ矢印で示したように与えられているとすると、対応する規則がないので、全ての前件部の適合度 $\mu_i = A_{1i}(x_1) \cdot A_{2i}(x_2) = 0$ ($i=1,2,\dots,9$)であり、非発火の状態になることがわかる。

図1で示したように、 R_i ($i=1,2,\dots,9$)以外の箇所では入力データを与えると、非発火のケースが起こることがわかる。この場合、式(3)の分母が0になり、ファジィ推論結果が求められないことになる。これにより、従来法では、ファジィ規則表を構築する時、設計者のイメージとかなりかけ離れた推論規則となってしまうことが考えられる。

以上の問題点を解決するために、次章では、従来法を改善し、ファジィ規則の調整を行う時、チューニングされるパラメータの数が多くならず、しかも、学習が終わった時点で、自動生成した推論規則を通常のファジィ規則表の形で表現できるという新しいファジィ規則の自動チューニングアルゴリズムを提案する。

図1 従来法の下ファジィ規則表

3. 最急降下法によるファジィ規則の自動チューニング

3.1 ファジィ規則の自動チューニングアルゴリズム

便宜上、2入力1出力の場合に対する最急降下法により新しいファジィ規則の自動チューニングアルゴリズムを導く。多入力多出力の場合には同様な手法で本学習アルゴリズムを拡張することができる(多入力1出力の場合については、付録参照)。

式(4)と同様に、入力 x_1, x_2 の二等辺三角型メンバーシップ関数 A_{1i}, A_{2j} ($i=1,2,\dots,r; j=1,2,\dots,k$) がそれぞれ次のように与えられているとする。

$$A_{1i}(x_1) = \begin{cases} 1 - \frac{2|x_1 - a_{1i}|}{b_{1i}}, & a_{1i} - b_{1i}/2 \leq x_1 \leq a_{1i} + b_{1i}/2; \\ 0, & \text{その他} \end{cases} \quad (13)$$

$$A_{2j}(x_2) = \begin{cases} 1 - \frac{2|x_2 - a_{2j}|}{b_{2j}}, & a_{2j} - b_{2j}/2 \leq x_2 \leq a_{2j} + b_{2j}/2; \\ 0, & \text{その他} \end{cases} \quad (14)$$

ここで、 r, k はメンバーシップ関数の数を示す。

また、これらのメンバーシップ関数 A_{1i}, A_{2j} を組み合わせることにより、 $r \times k$ 個のファジィ推論規則は次のように与えられるとする(表1参照)[9]。

規則 1 :	$A_{11}, A_{21} \Rightarrow y_1$
規則 2 :	$A_{11}, A_{22} \Rightarrow y_2$
.....	
規則 k :	$A_{11}, A_{2k} \Rightarrow y_k$
規則 $k+1$:	$A_{12}, A_{21} \Rightarrow y_{k+1}$
.....	
規則 $2k$:	$A_{12}, A_{2k} \Rightarrow y_{2k}$
.....	

規則 $(i-1)k+j$: $A_{1i}, A_{2j} \Rightarrow y_{(i-1)k+j}$

.....

規則 $r \times k$: $A_{1r}, A_{2k} \Rightarrow y_{r \times k}$

ここで, $y_{(i-1)k+j}$ ($i=1, 2, \dots, r; j=1, 2, \dots, k$) は後件部の実数値である。

式(15)と表1により, 従来法と違って, 前件部のメンバーシップ関数は入力変数ごとに設定されていることがわかる。これにより, 入力 x_1, x_2 が与えられた時のファジィ規則の前件部 A_{1i}, A_{2j} の適合度は次のようになる。

$$\mu_{(i-1)k+j} = A_{1i}(x_1) A_{2j}(x_2) \quad i=1, 2, \dots, r; j=1, \dots, k \quad (16)$$

したがって, 式(3)の出力 y を式(15), (16)によって, 次のような形に書き直すことができる。

$$y = \frac{\sum_{i=1}^r \sum_{j=1}^k \mu_{(i-1)k+j} y_{(i-1)k+j}}{\sum_{i=1}^r \sum_{j=1}^k \mu_{(i-1)k+j}} = \frac{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2) y_{(i-1)k+j}}{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \quad (17)$$

従来法と同様に, 式(5)の評価関数 E の値を減少させるために, パラメータ $a_{1i}, b_{1i}, a_{2j}, b_{2j}$ および $y_{(i-1)k+j}$ に対して, E が減少する方向ベクトル, つまり, $(-\partial E/\partial a_{1i}, -\partial E/\partial b_{1i}, -\partial E/\partial a_{2j}, -\partial E/\partial b_{2j}, \partial E/\partial y_{(i-1)k+j})$ ($i=1, 2, \dots, r; j=1, 2, \dots, k$) を計算すればよい[6]。

まず, 式(13)-(17)から $\partial E/\partial a_{1i}, \partial E/\partial b_{1i}, \partial E/\partial a_{2j}, \partial E/\partial b_{2j}$ および $\partial E/\partial y_{(i-1)k+j}$ は次のように求められる。

$$\begin{aligned} \partial E/\partial a_{1i} &= (\partial E/\partial y) (\partial y/\partial A_{1i}) (\partial A_{1i}/\partial a_{1i}) \\ &= -(y^* - y) \frac{\sum_{j=1}^k A_{2j}(x_2) (y_{(i-1)k+j} - y)}{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \\ &\quad \frac{2 \operatorname{sgn}(x_1 - a_{1i}(t))}{b_{1i}(t)} \end{aligned} \quad (18)$$

$$\begin{aligned} \partial E/\partial b_{1i} &= (\partial E/\partial y) (\partial y/\partial A_{1i}) (\partial A_{1i}/\partial b_{1i}) \\ &= -(y^* - y) \frac{\sum_{j=1}^k A_{2j}(x_2) (y_{(i-1)k+j} - y)}{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \\ &\quad \frac{2|x_1 - a_{1i}(t)|}{b_{1i}(t)^2} \end{aligned} \quad (19)$$

$$\begin{aligned} \partial E/\partial a_{2j} &= (\partial E/\partial y) (\partial y/\partial A_{2j}) (\partial A_{2j}/\partial a_{2j}) \\ &= -(y^* - y) \frac{\sum_{i=1}^r A_{1i}(x_1) (y_{(i-1)k+j} - y)}{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \\ &\quad \frac{2 \operatorname{sgn}(x_2 - a_{2j}(t))}{b_{2j}(t)} \end{aligned} \quad (20)$$

表1 入力変数ごとに設定されたファジィ規則

$X_1 \backslash X_2$	A_{21}	A_{22}	$\dots$	A_{2j}	$\dots$	A_{2k}
A_{11}	y_1	y_2	$\dots$	y_j	$\dots$	y_k
A_{12}	y_{k+1}	y_{k+2}	$\dots$	y_{k+j}	$\dots$	y_{2k}
$\vdots$						
A_{1i}	$y_{(i-1)k+1}$	$\dots$		$y_{(i-1)k+j}$	$\dots$	y_{ik}
$\vdots$						
A_{1r}	$y_{(r-1)k+1}$	$\dots$				y_{rk}

$$\begin{aligned} \partial E/\partial b_{2j} &= (\partial E/\partial y) (\partial y/\partial A_{2j}) (\partial A_{2j}/\partial b_{2j}) \\ &= -(y^* - y) \frac{\sum_{i=1}^r A_{1i}(x_1) (y_{(i-1)k+j} - y)}{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \\ &\quad \frac{2|x_2 - a_{2j}(t)|}{b_{2j}(t)^2} \end{aligned} \quad (21)$$

$$\begin{aligned} \partial E/\partial y_{(i-1)k+j} &= (\partial E/\partial y) (\partial y/\partial y_{(i-1)k+j}) \\ &= -(y^* - y) \frac{A_{1i}(x_1) A_{2j}(x_2)}{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \end{aligned} \quad (22)$$

これらの結果を最急降下法によって得た学習式(6)-(8)に代入すると, 結局, パラメータ $a_{1i}, b_{1i}, a_{2j}, b_{2j}, y_{(i-1)k+j}$ ($i=1, 2, \dots, r; j=1, 2, \dots, k$) の第 $t+1$ 回の自動チューニングアルゴリズムは次のように得られる。

$$\begin{aligned} a_{1i}(t+1) &= a_{1i}(t) \\ &+ (y^* - y) \frac{2 K_a \operatorname{sgn}(x_1 - a_{1i}(t)) \sum_{j=1}^k A_{2j}(x_2) (y_{(i-1)k+j} - y)}{b_{1i}(t) \sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \end{aligned} \quad (23)$$

$$\begin{aligned} b_{1i}(t+1) &= b_{1i}(t) \\ &+ (y^* - y) \frac{2 K_b |x_1 - a_{1i}(t)| \sum_{j=1}^k A_{2j}(x_2) (y_{(i-1)k+j} - y)}{b_{1i}(t)^2 \sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \end{aligned} \quad (24)$$

$$\begin{aligned} a_{2j}(t+1) &= a_{2j}(t) \\ &+ (y^* - y) \frac{2 K_a \operatorname{sgn}(x_2 - a_{2j}(t)) \sum_{i=1}^r A_{1i}(x_1) (y_{(i-1)k+j} - y)}{b_{2j}(t) \sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \end{aligned} \quad (25)$$

$$\begin{aligned} b_{2j}(t+1) &= b_{2j}(t) \\ &+ (y^* - y) \frac{2 K_b |x_2 - a_{2j}(t)| \sum_{i=1}^r A_{1i}(x_1) (y_{(i-1)k+j} - y)}{b_{2j}(t)^2 \sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \end{aligned} \quad (26)$$

$$\begin{aligned} y_{(i-1)k+j}(t+1) &= y_{(i-1)k+j}(t) \\ &+ \frac{K_y (y^* - y) A_{1i}(x_1) A_{2j}(x_2)}{\sum_{i=1}^r \sum_{j=1}^k A_{1i}(x_1) A_{2j}(x_2)} \end{aligned} \quad (27)$$

明らかに, ファジィ推論規則の設定方法(15)を用いることによって, 得られた学習アルゴリズム(23)-(27)

は従来法の学習式(9)-(11)と違うことがわかる。しかしながら、評価関数 E が最も小さくなる方向を探索しながらパラメータをチューニングすることは従来法と一致する。

3.2 チューニング手順

学習アルゴリズム(23)-(27)を用いて、ファジィ規則を逐次的にチューニングする手順は従来法とほぼ同じで、次のように行える。

ステップ1：推論規則の初期設定を行う。前件部のメンバーシップ関数のパラメータと後件部の実数値の初期値を決める。

例えば、入力変数 x_1 に対するメンバーシップ関数 A_{1i} の初期設定は x_1 の定義域を等分割し、つまり、隣りあうメンバーシップ関数はグレード値 0.5 の所で交差するように設定する。

ステップ2： P 個の学習用の入出力データ対 (x_1^p, x_2^p, y^{*p}) を入力する ($p=1, 2, \dots, P$)。

ステップ3：式(13), (14), (17)を用いて、 p 番目の入力データ x_1^p, x_2^p に対して、ファジィ推論を行い、推論結果 y^p を求める。

ステップ4：推論結果 y^p と理想出力値 y^{*p} を用いて学習アルゴリズム(23)-(27)により、パラメータ $a_{1i}, b_{1i}, a_{2j}, b_{2j}, y_{(i-1)k+j}$ ($i=1, 2, \dots, r; j=1, 2, \dots, k$) を更新する。

ステップ5： $p < P$ ならば p を1増加させてステップ3に戻る。 $p = P$ ならば、次のステップ6に進む。

ステップ6：次式で推論誤差 $E(t)$ を計算する。

$$E(t) = \frac{1}{P} \sum_{p=1}^P (y^{*p} - y^p)^2 \quad (28)$$

$E(t)$ が次式を満足する場合、学習を終了する。そうでない場合、ステップ2に進む。

$$E(t) \leq \delta \quad (29)$$

但し、 δ は学習を終了させるための閾値を示す[6]。

3.3 新手法の特徴

従来の学習アルゴリズムと比べてみると、新手法は次の特徴を持っている。

特徴(1)：ファジィ規則はいわゆる入力1と入力2でのメンバーシップ関数の組み合わせによって構成されていて、つまり、前件部のメンバーシップ関数はファジィ規則毎に独立ではなくて、しかも、その数は学習を行っても変化しないことである。この意味で、同じ初期条件の下では、従来法より新手法の方がチューニングされるパラメータの数が少ないので、計算速度が速いことがわかる。

特徴(2)：新手法はファジィ規則表のチューニング手法と言える。予め与えられたファジィ規則表に対して、新手法では従来法と違って、その形を崩さずにパラメータのチューニングを行うことができる。これはユーザにとって分かりやすい学習アルゴリズムと思われる。

図1のようになる従来法の場合と違って、新手法は、パラメータをチューニングし終わった時点での入出力関係(ファジィ規則表)は図2のように表せる(2入力の場合)。明らかに、この場合は、一般に、非発火ケースを抑制でき、通常ファジィ制御表を用いるファジィ制御の方法と一致する。

以上、従来の「デルタルールによるファジィ推論の自動チューニング手法」の改善として、ここでは最急降下法を用いて、対称三角型メンバーシップ関数の下で新たなファジィ規則の学習アルゴリズムを提案した。

	x_2	A_{21}	A_{22}	A_{23}
x_1				
A_{11}	R_1		R_2	R_3
A_{12}				
A_{13}	R_4		R_5	R_6
	R_7		R_8	R_9

図2 新手法の下でのファジィ規則表

4. 数値例

ここでは、次のような3種類の2入力1出力の非線形関数の同定の例[5-6]を挙げて、従来の学習アルゴリズムとの比較・検討を行うと共に、本手法の有効性を示す。

関数(I)：

$$y = (2x_1 + 4x_2^2 + 0.1)^2 \quad (30)$$

関数(II)：

$$y = 4 \sin(\pi x_1) + 2 \cos(\pi x_2) \quad (31)$$

関数(III)：

$$y = (3 \exp(3x_1) + 2 \exp(-4x_2))^{-0.5} \quad (32)$$

ここで、 $x_1, x_2 \in [-1, 1]$ は入力変数で、 $y \in [0, 1]$ は正規化された出力変数である。

まず、関数(I)を例として、それぞれ新手法と従来法により生成した前件部のメンバーシップ関数を考察してみよう。

4.1 学習により生成したメンバーシップ関数

関数(I)の同定に対して、新手法と従来法の下での

ファジィ規則とそのパラメータの初期値は同じとし、表2のようにファジィ規則表の形で各入力部分に3個ずつのメンバーシップ関数を用いるものとする(図3参照)。表2で、 $-1, 0, 1$ はそれぞれメンバーシップ関数の中心であり、2はメンバーシップ関数の幅で、0.5は後件部の実数値である。

25組のランダムデータ(x_1, x_2)に対して、新手法と従来法を用いて同定を行った。なお、学習係数は $K_a = K_b = K_y = 0.1$ 、学習の終了を判定する閾値(式(29)参照)は $\delta = 0.01$ とした。表3は、25組のランダム入力データ(x_1, x_2)の一例を表すものである。

表3のデータを用いて、新手法と従来法により生成されたファジィ規則はそれぞれ表4および式(33)のようになっている。

従来法によるファジィ規則：

$$\begin{aligned}
 &1: A_{11}(-0.41, 1.70), A_{21}(-0.48, 1.55) \Rightarrow 0.0122 \\
 &2: A_{12}(0.88, 1.26), A_{22}(-1.29, 1.83) \Rightarrow 0.4018 \\
 &3: A_{13}(-0.37, 1.26), A_{23}(-0.79, 1.19) \Rightarrow 0.2432 \\
 &4: A_{14}(0.72, 1.81), A_{24}(1.49, 1.96) \Rightarrow 0.3623 \\
 &5: A_{15}(-0.50, 1.77), A_{25}(0.65, 1.79) \Rightarrow 0.0150 \\
 &6: A_{16}(-0.05, 1.49), A_{26}(0.68, 1.36) \Rightarrow 0.0947 \\
 &7: A_{17}(1.47, 1.95), A_{27}(-0.51, 1.88) \Rightarrow 0.2112 \\
 &8: A_{18}(0.57, 0.73), A_{28}(0.83, 0.56) \Rightarrow 0.4477 \\
 &9: A_{19}(1.23, 1.47), A_{29}(0.89, 0.96) \Rightarrow 0.7103
 \end{aligned} \quad (33)$$

(a) x_1 におけるメンバーシップ関数

(b) x_2 におけるメンバーシップ関数

図3 初期メンバーシップ関数

表2 ファジィ規則の初期設定(1)

A_{1j} \ A_{2j}	$(-1, 2)$	$(0, 2)$	$(1, 2)$
$(-1, 2)$	0.5	0.5	0.5
$(0, 2)$	0.5	0.5	0.5
$(1, 2)$	0.5	0.5	0.5

図4は、式(33)の前件部、つまり、従来法による学習後の x_1, x_2 に関するメンバーシップ関数を表している。各入力次元でのメンバーシップ関数の数は3個から9個に増え、生成されたファジィ規則はファジィ規則表の形(表2参照)を崩してしまっている。

また、従来法では、前件部のメンバーシップ関数は推論規則毎に独立に存在しているので、すなわち、図1のような関係が成り立つので、入力データによっては非発火現象が生じることがある。例えば、図4において、 $x_1 (\in [0.45, 0.49])$ が $\uparrow$ の付近の入力であり、 $x_2 (\in [-0.38, 0.01])$ が $\uparrow$ の付近の入力である場合を考えてみよう。

この時、 $\uparrow$ で示したように、 x_1 に対して発火するメンバーシップ関数は $A_{12}, A_{14}, A_{16}, A_{18}$ であり、 x_2 に対して発火するメンバーシップ関数は A_{21}, A_{25}, A_{27} であるので、結局、この場合、全ての前件部の適合度は $\mu_i = A_{1i}(x_1) \cdot A_{2i}(x_2) = 0 (i=1, 2, \dots, 9)$ となり、非発火の状態となることがわかる。

表3 入力データ (x_1, x_2)の一例

No.	x_1	x_2	No.	x_1	x_2	No.	x_1	x_2
1	0.20	0.36	10	-0.80	-0.04	19	0.40	-0.78
2	0.82	0.96	11	-0.44	0.12	20	-0.58	-1.00
3	0.24	-0.18	12	0.58	-0.84	21	-0.10	0.64
4	0.34	-0.64	13	-0.44	0.68	22	0.20	-0.64
5	0.32	0.80	14	0.74	0.66	23	-0.14	0.50
6	-0.02	-0.20	15	-0.96	-0.94	24	-0.90	0.36
7	-0.64	0.86	16	-0.54	-0.34	25	0.02	-0.60
8	-0.68	-0.16	17	-0.52	-0.12			
9	0.90	-0.26	18	0.58	-0.92			

(a) x_1 に関するメンバーシップ関数

(b) x_2 に関するメンバーシップ関数

図4 従来法による学習後のメンバーシップ関数

一方、新手法の場合、表4と図5で示したように、ファジィ規則表の形は崩さずにそのパラメータのチューニングが行われ、学習が終わった後も、パラメータの数は増加せず、しかも、非発火現象を抑制できる。

表4と式(33)のファジィ規則を用いて、関数(I)に対する2601個の評価データ (x_1, x_2) (ここで、 x_1, x_2 はそれぞれ-1.0から1.0まで0.04きざみとした)を入力して、新手法と従来法によって得たファジィ推論値と関数(I)の理想出力値との絶対誤差を表すと、図6のようになる。

また、理想出力値とファジィ推論値との評価誤差(平均二乗誤差) D を次のように求める。

$$D = \frac{1}{2601} \sum_{d=1}^{2601} (y_d^* - y_d)^2 \quad (34)$$

ここで、 $y_d^* (d=1, 2, \dots, 2601)$ は理想出力値で、 y_d はファジィ推論値である。

この例には、それぞれの評価誤差と最大誤差は、新手法の場合、0.0104, 0.4606であり、従来法の場合、0.0175, 0.6572である。

この例から、従来法の場合、先ほど述べたように入力部分空間 $[0.45, 0.49] \times [-0.38, 0.01]$ において、ファジィ規則が存在しないので、非発火の状態になっていて、その評価誤差が大きくなることがわかる。但し、非発火の場合、通常の推論値は存在しないが、図6の

表4 新手法によるファジィ規則

$A_{1i} \backslash A_{2j}$	$(-1.28, 1.82)$	$(-0.01, 1.98)$	$(1.28, 1.84)$
$(-1.14, 1.92)$	0.2055	0.0625	0.2919
$(-0.07, 1.99)$	0.3738	0.0000	0.2451
$(1.33, 1.67)$	0.5394	0.2720	0.6667

(a) x_1 に関するメンバーシップ関数

(b) x_2 に関するメンバーシップ関数

図5 新手法による学習後のメンバーシップ関数

場合には、強制的に0.5にしている。

次に、関数(I)-(III)に対して、新手法と従来法との学習回数、評価誤差の比較を行おう。

4.2 学習回数・評価誤差の比較

式(30)-(32)の3つの関数に対して、最初に、各入力次元を前件部のメンバーシップ関数で5等分割し、ファジィ規則の総数を25個として、新手法と従来法のパラメータの初期値は同じで、表5のように設定する。

ここで、学習時に、チューニングされるパラメータの数は従来法の場合、 $25 \times 2 \times 2 + 25 = 125$ 個であり、新手法の場合、 $5 \times 2 \times 2 + 25 = 45$ 個であることに注意されたい。

表5 ファジィ規則の初期設定(2)

$A_{1i} \backslash A_{2j}$	$(-1, 1)$	$(-0.5, 1)$	$(0, 1)$	$(0.5, 1)$	$(1, 1)$
$(-1, 1)$	0	0	0	0	0
$(-0.5, 1)$	0	0	0	0	0
$(0, 1)$	0	0	0	0	0
$(0.5, 1)$	0	0	0	0	0
$(1, 1)$	0	0	0	0	0

(a) 新手法による絶対誤差

(b) 従来法による絶対誤差

図6 ファジィ推論値と理想出力値との絶対誤差

まず、各々の関数(I)-(III)に対して、49組のランダムデータ(x_1, x_2)を入力して、式(30)-(32)と各学習法によるそれぞれの理想出力値とファジィ推論値との誤差を式(28)で定める。

式(28)の推論誤差 E が $E < \delta$ を満たすまでそれぞれ新手法と従来法の下で学習を行う。但し、学習係数は $K_a = K_b = 0.05$, $K_y = 0.65$ とし、 δ の設定は関数の順番に $\delta = 0.002, \delta = 0.005, \delta = 0.001$ とする。

そして、新手法と従来法により生成したファジィ規則に対して、関数(I)-(III)に対する2601個の評価データ(x_1, x_2) (ここで、 x_1, x_2 はそれぞれ-1.0から1.0まで0.04きざみとした)を入力して、式(34)を用いて関数(I)-(III)の理想出力値とファジィ推論値との評価誤差、最大絶対誤差を求める。

表6-8は、関数(I)-(III)に対して、それぞれ新手法と従来法を用いることにより学習回数と評価誤差の比較(10通り)を示すものである。

表6-8を考察すると、次のことが考えられる。

1) : 学習回数についてはほぼ同じであるが、新手法より従来法の方がかなり多い場合があった。これは学習途中で非発火ケースが生じてきたからだと考えられる。新手法の学習回数が多かったのは、前件部のメンバーシップ関数を重複的に使っているため、パラメータの変化の自由度が小さく、与えられた教師データに対して速くフィッティングできないからと考えられる。総合的に言えば、3つの関数に対する学習回数は2つの手法ではほぼ同じであるが、新手法の前件部メンバーシップ関数のパラメータ数は従来法の場合の5分の1であることを注意すると、結局、新手法は従来法より学習時間が速いことがわかる。例えば、関数(I)に対して、一回の学習時間は、新手法の約0.38秒に対して、従来法の場合、0.81秒かかっている。これにより、新手法はチューニングされるパラメータの数が少なく、多入力ファジィシステムモデルに適用でき、従来法と同じく高速な学習が可能であると考えられる。

2) : 新手法は、前件部のメンバーシップ関数の数は少ない(従来法の場合の5分の1)という条件の下で、従来法と同精度の結果(或いは、従来法よりやや良い結果)が得られることが示された。また、新手法はファジィ規則表の形が崩されないという特徴を持ち、非発火を抑制できることから、実应用的な応用に対して汎用的なチューニング技法と言えるであろう。

表6 関数Iに対する従来法との比較

No.	学習回数		評価誤差		最大絶対誤差	
	従来法	新手法	従来法	新手法	従来法	新手法
①	301	52	0.004621	0.003389	0.425821	0.317746
②	24	26	0.004817	0.005232	0.444861	0.570375
③	14	18	0.013321	0.016190	0.746787	0.813321
④	39	55	0.005486	0.002782	0.419521	0.315845
⑤	55	57	0.011930	0.005951	0.645300	0.537174
⑥	926	7	0.005327	0.014572	0.728291	0.929757
⑦	34	89	0.010243	0.009044	0.696054	0.615451
⑧	86	64	0.005367	0.005436	0.464406	0.487588
⑨	32	21	0.002888	0.003690	0.280864	0.314910
⑩	35	109	0.006644	0.004028	0.423014	0.346637

表7 関数IIに対する従来法との比較

No.	学習回数		評価誤差		最大絶対誤差	
	従来法	新手法	従来法	新手法	従来法	新手法
①	19	164	0.009578	0.006853	0.378469	0.307029
②	98	40	0.006577	0.007160	0.304451	0.315427
③	55	12	0.006333	0.005763	0.279614	0.202603
④	132	83	0.009581	0.007992	0.393263	0.319067
⑤	23	19	0.010069	0.007704	0.387836	0.338116
⑥	20	49	0.005857	0.008522	0.272872	0.330197
⑦	87	28	0.007646	0.004413	0.353949	0.159031
⑧	39	225	0.008693	0.007639	0.413483	0.333493
⑨	93	49	0.009575	0.010005	0.355557	0.293072
⑩	18	251	0.005857	0.005654	0.314790	0.308485

表8 関数IIIに対する従来法との比較

No.	学習回数		評価誤差		最大絶対誤差	
	従来法	新手法	従来法	新手法	従来法	新手法
①	24	12	0.001339	0.000760	0.222732	0.247734
②	16	46	0.001067	0.000804	0.258798	0.225256
③	27	14	0.002816	0.002012	0.415812	0.356186
④	20	12	0.001119	0.001146	0.252151	0.112236
⑤	109	5	0.001114	0.001794	0.112429	0.187353
⑥	39	22	0.002087	0.001259	0.278152	0.314454
⑦	33	25	0.001242	0.000859	0.320178	0.277709
⑧	13	23	0.001489	0.000667	0.268116	0.101592
⑨	35	209	0.001048	0.001451	0.183283	0.193183
⑩	8	64	0.011437	0.004349	0.699883	0.478412

5. 結論

本論文では、従来の「デルタルールによるファジィ推論の自動チューニング手法」に対する検討を行い、その改善法として、ファジィ規則表のチューニングを目指す学習アルゴリズムを提案した。また、3つの関数の同定を行い、従来法と比較すると共に、新手法の有効性を示した。

しかしながら、新手法では、前件部のメンバーシップ関数は、入力次元ごとに設定されているので、相互的に依存関係が強くパラメータの変化が拘束され、場合によっては同定精度に到達しにくいという問題が存在する(表6-8参照)。これに対して、各入力次元での分割数やトレーニングデータの数を調整することにより解消できるものと思われる。

今後の課題として、新手法を用いて、実際のファジィ制御対象向けの応用研究を行い、より良い結果を上げることが考えられる。

謝 辞 この研究に貴重な意見を頂いた査読の方々に厚く感謝の意を表す。

参 考 文 献

1. 市橋・渡辺, 簡略ファジィ推論を用いたファジィモデルによる学習型制御, 日本ファジィ学会誌, Vol.2, No.3, 429-437(1990)
2. 林・野村・若見, ニューラルネット駆動型ファジィ推論による推論ルールの獲得, 日本ファジィ学会誌, Vol.2, No.4, 585-597(1990)
3. 堀川・古橋・大熊・内川, ニューラルネットワークによる学習型ファジィ制御器, 計測自動制御学会論文集, Vol.27, No.2, 208-215(1991)
4. 野村・林・若見, 最急降下法によるファジィ推論の自動チューニングと障害物回避への応用, 第6回「ファジィシステムシンポジウム」講演論文集, 423-426(1990)
5. H.Nomura, I.Hayashi and N.Wakami, A learning method of fuzzy inference rules by descent method, Proc. of the 4th IFSA Congress, Vol.Engineering, Brussels, 155-158(1991)
6. 野村・林・若見, デルタルールによるファジィ推論の自動チューニング手法と障害物回避への応用, 日本ファジィ学会誌, Vol.4, No.2, 379-388(1992)
7. 増田・夜久, ユニット生成機能をもつニューラルネットワークによるファジィ推論ルールの獲得手法, 日本ファジィ学会誌, Vol.5, No.2, 348-357(1993)
8. 岸田・宮島・福元・村島, 学習アルゴリズムによるファジィモデルの構成, 日本ファジィ学会誌, Vol.7, No.3, 585-593(1995)
9. 石・水本・湯場崎・大谷, ニューロ・ファジィ学習アルゴリズムによるファジィ規則の一生成法, 日本ファジィ学会誌, Vol.8, No.4, 695-705(1996)

[付録]

多入力1出力の場合の学習アルゴリズムの導出

まず, 入力 $x_1, x_2, \dots, x_n$ の二等辺三角型メンバーシップ関数 A_{ij} (記法上, A_i^j と示す)($i=1, 2, \dots, n; j=$

$1, 2, \dots, k$)は次のように与えられているとする。

$$A_i^j(x_i) = \begin{cases} 1 - \frac{2|x_i - a_i^j|}{b_i^j} & a_i^j - b_i^j/2 \leq x_i \leq a_i^j + b_i^j/2; \\ 0, & \text{その他} \end{cases} \quad (\text{A-1})$$

また, ファジィ規則は次のようである。

$$\begin{aligned} \text{規則1:} & A_1^1, \dots, A_{n-1}^1, A_n^1 \Rightarrow y_1 \\ \text{規則2:} & A_1^1, \dots, A_{n-1}^1, A_n^2 \Rightarrow y_2 \\ & \dots\dots\dots \\ \text{規則}k: & A_1^1, \dots, A_{n-1}^1, A_n^k \Rightarrow y_k \\ \text{規則}k+1: & A_1^1, \dots, A_{n-1}^2, A_n^1 \Rightarrow y_{k+1} \\ & \dots\dots\dots \\ \text{規則}2k: & A_1^1, \dots, A_{n-1}^2, A_n^k \Rightarrow y_{2k} \\ & \dots\dots\dots \\ \text{規則}(j_1-1): & k^{n-1} + \dots + (j_{(n-1)}-1)k + j_n: \\ & A_1^{j_1}, \dots, A_{n-1}^{j_{(n-1)}}, A_n^{j_n} \\ & \Rightarrow y_{(j_1-1)k^{n-1} + \dots + (j_{(n-1)}-1)k + j_n} \\ & \dots\dots\dots \\ \text{規則}k^n: & A_1^k, \dots, A_{n-1}^k, A_n^k \Rightarrow y_{k^n} \end{aligned} \quad (\text{A-2})$$

ここで, y_m ($m=1, 2, \dots, k^n$)は後件部の実数値で, 添字は $j_1, \dots, j_{(n-1)}, j_n \in \{1, 2, \dots, k\}$ である。

入力 $x_1, x_2, \dots, x_n$ が与えられた時, 式(16), (17)の拡張として, 出力 y は次のように求められる。

$$y = \frac{\sum_{j_1=1}^k \dots \sum_{j_{i-1}=1}^k \sum_{j_i=1}^k \dots \sum_{j_n=1}^k A_1^{j_1}(x_1) \dots A_i^{j_i}(x_i) \dots A_n^{j_n}(x_n) y_m}{\sum_{j_1=1}^k \dots \sum_{j_{i-1}=1}^k \sum_{j_i=1}^k \dots \sum_{j_n=1}^k A_1^{j_1}(x_1) \dots A_i^{j_i}(x_i) \dots A_n^{j_n}(x_n)} \quad (\text{A-3})$$

但し, $y_m = y_{(j_1-1)k^{n-1} + \dots + (j_{(n-1)}-1)k + j_n}$ ($j_1, \dots, j_i, \dots, j_n \in \{1, 2, \dots, k\}$)である。

式(A-3)により, $A_i^{j_i}$ に対して, その微分は次のように求められる。

$$\begin{aligned} \partial y / \partial A_i^{j_i} &= \\ & \frac{\sum_{j_1=1}^k \dots \sum_{j_{i-1}=1}^k \sum_{j_{i+1}=1}^k \dots \sum_{j_n=1}^k A_1^{j_1}(x_1) \dots A_{i-1}^{j_{i-1}}(x_{i-1}) \dots A_{i+1}^{j_{i+1}}(x_{i+1}) \dots A_n^{j_n}(x_n) (y_m - y)}{\sum_{j_1=1}^k \dots \sum_{j_{i-1}=1}^k \sum_{j_i=1}^k \dots \sum_{j_n=1}^k A_1^{j_1}(x_1) \dots A_i^{j_i}(x_i) \dots A_n^{j_n}(x_n)} \\ & = D_i^{j_i} \end{aligned} \quad (\text{A-4})$$

但し, $i=1, 2, \dots, n; j_i \in \{1, 2, \dots, k\}$ である。

したがって, 式(5)および式(A-1), (A-4)によって, 学習式(6)-(8)における $\partial E / \partial a_i^{j_i}, \partial E / \partial b_i^{j_i}$ および $\partial E / \partial y_m$ は次のように求められる。

$$\begin{aligned} \frac{\partial E}{\partial a_i^{j_i}} &= \frac{\partial E}{\partial y} \frac{\partial y}{\partial A_i^{j_i}} \frac{\partial A_i^{j_i}}{\partial a_i^{j_i}} \\ &= (y - y^*) D_i^{j_i} \frac{2 \operatorname{sgn}(x_i - a_i^{j_i})}{b_i^{j_i}} \end{aligned} \quad (\text{A-5})$$

$$\frac{\partial E}{\partial b_i^{ji}} = \frac{\partial E}{\partial y} \frac{\partial y}{\partial A_i^{ji}} \frac{\partial A_i^{ji}}{\partial b_i^{ji}} = (y - y^*) D_i^{ji} \frac{2|x_i - a_i^{ji}|}{(b_i^{ji})^2} \quad (\text{A-6})$$

$$\begin{aligned} \frac{\partial E}{\partial y_m} &= (\frac{\partial E}{\partial y}) (\frac{\partial y}{\partial y_m}) \\ &= (y - y^*) \frac{A_1^{j_1}(x_1) \cdots A_n^{j_n}(x_n)}{\sum_{j_1=1}^k \cdots \sum_{j_n=1}^k A_1^{j_1}(x_1) \cdots A_n^{j_n}(x_n)} \end{aligned} \quad (\text{A-7})$$

但し, $i=1, 2, \dots, n$; $j_i \in \{1, 2, \dots, k\}$; $m=1, 2, \dots, k^n$ である。

結局, パラメータ a_i^{ji}, b_i^{ji}, y_m の第 $t+1$ 回の自動チューニングアルゴリズムは次のように得られる。

$$\begin{aligned} a_i^{ji}(t+1) &= a_i^{ji}(t) - \frac{\partial E}{\partial a_i^{ji}}(t) \\ &= a_i^{ji}(t) - (y - y^*) D_i^{ji} \frac{2K_a \text{sgn}(x_i - a_i^{ji}(t))}{b_i^{ji}(t)} \end{aligned} \quad (\text{A-8})$$

$$\begin{aligned} b_i^{ji}(t+1) &= b_i^{ji}(t) - \frac{\partial E}{\partial b_i^{ji}}(t) \\ &= b_i^{ji}(t) - (y - y^*) D_i^{ji} \frac{2K_b |x_i - a_i^{ji}(t)|}{(b_i^{ji}(t))^2} \end{aligned} \quad (\text{A-9})$$

$$y_m(t+1) = y_m(t) - \frac{\partial E}{\partial y_m}(t)$$

—— 著 者 紹 介 ——

石 岩 (SHI Yan)

大阪電気通信大学大学院工学研究科博士後期課程

1982年, 中国東北重型機械学院基礎工学科卒業。同年, 同大学応用数学科助手。1988年, 中国大連海事大学大学院修士課程応用数学科専攻終了。同年, 東北重型機械学院応用数学科講師。1991年4月来日, 山形大学工学部客員研究員, 大阪電気通信大学工学部客員研究員を経て, 1994年4月より, 大阪電気通信大学大学院博士後期課程情報工学専攻に在学中。ファジィ推論, ファジィモデリングに関する研究に従事。

水本 雅晴 (みずもと まさはる)

大阪電気通信大学 情報工学部情報工学科

1966年, 大阪大学基礎工学部電気工学科卒業。1971年, 同大学院博士課程修了。工学博士。同年, 同大学情報工学科助手。1978年, 大阪電気通信大学工学部経営工学科助教授。1980年, 同大学教授となり, 現在に至る。1980~81年, フンボルト奨学生としてドイツ・アーヘン工科大学に滞在。現日本ファジィ学会理事, バイオメディカルファジィ学会副会長, 前日本ファジィ学会誌編集委員長。ファジィ理論とその応用, 特に, ファジィ推論, ファジィ制御法, ファジィ・ニューロ, ファジィ演算, ファジィ数などの研究に従事。著書に「ファジィ理論とその応用」(サイエンス社)など。日本ファジィ学会, 国際ファジィシステム学会, 計測自動制御学会, システム制御情報学会, 人工知能学会, 電子情報通信学会, 情報処理学会の会員。

$$y_m(t) = \frac{K_y(y - y^*) A_1^{j_1}(x_1) \cdots A_n^{j_n}(x_n)}{\sum_{j_1=1}^k \cdots \sum_{j_n=1}^k A_1^{j_1}(x_1) \cdots A_n^{j_n}(x_n)} \quad (\text{A-10})$$

ここで, K_a, K_b, K_y は学習係数であり, t は学習回数を示す。

(1995年9月28日 受 付)

(1996年4月9日 再受付)

[問い合わせ先]

〒572 大阪府寝屋川市初町 18-8

大阪電気通信大学情報工学部

情報工学科水本研究室

石 岩 (SHI Yan)

TEL : 0720-20-4569

FAX : 0720-24-0014

E-mail : shi@mzlab.osakac.ac.jp

湯場崎 直養 (ゆばざき なおよし)

マイコム株式会社取締役社長

1964年, 京都市立洛陽工業高等学校電気通信科卒業。1968年, マイコム株式会社の前身, 松本電機株式会社を創業。1970年, 大阪電気通信大学工学部通信工学科卒業。現在, マイコム株式会社取締役社長。(米国) Nyden Corporation, (シンガポール) Mycom Technology (S) Pte. Ltd. 各社長を兼務。(台湾) Mycom Technology Inc. と (韓国) Mycom Korea Inc. の取締役最高顧問。日本ファジィ学会評議委員。制御概念の研究, とくにモータ制御とファジィ制御の融合に関する研究に従事。日本ファジィ学会会員。

大谷 正幸 (おおたに まさゆき)

マイコム株式会社 中央研究部

1972年, 千代田電子専門学校電子工学科卒業。同年松本電機株式会社(現マイコム株式会社)へ入社。現在, マイコム株式会社中央研究部部長, マイコム・グローバル研究機構ジェネラルマネージャ兼務。ファジィ推論チップ, 学習アルゴリズム, インテリジェンスモーションコントロール等に関する研究開発に従事。日本ファジィ学会会員。

A Self-Tuning Method of Fuzzy Rules Based on Gradient Descent Method

by

Yan SHI, Masaharu MIZUMOTO, Naoyoshi YUBAZAKI and Masayuki OTANI

Abstract :

As a kind of learning approach for a fuzzy system model, so-called the self-tuning method of fuzzy rules by Delta rule proposed by H. Nomura et al. is well known. While the method has high generalization capability, the number of tuning parameters increases quickly for fuzzy system models with multiple-input variables and the shape of the fuzzy rule table is destroyed after the learning. Moreover, sometimes the case of a non-firing occurs for evaluating input data. To improve the above problems, we suggest a new tuning algorithm for learning fuzzy models based on the gradient descent method. In this learning algorithm, the number of tuning parameters is fixed in the learning process and the shape of fuzzy rule table never changes even after the learning. Furthermore, we compare the suggested method with the self-tuning method, and illustrate the efficiency of the suggested method for tuning fuzzy rules by means of several numerical examples.

Keywords : Simplified fuzzy reasoning, Triangular-type membership function, Gradient descent method, Self-tuning method

Contact Address : **Yan SHI**

*Division of Information and Computer Sciences,
Osaka Electro-Communication University
18-8, Hatsu-cho, Neyagawa, Osaka, 572, Japan
TEL : +81-720-20-4569
FAX : +81-720-24-0014
E-mail : shi@mzlab.osakac.ac.jp*