

多目的非線形最適化手法 Vector Simplex 法による 多目的ファジィ制御ルールの対話的学習

高濱 徹行[†] 阪井 節子^{††}

多目的ファジィ制御ルールの学習は、複数の制御誤差を同時に最小化する、すなわち、微分不可能な複数の目的関数の多目的最適化問題と見なすことができる。しかし、制御ルールの良さは制御誤差のみで決定するのは困難であり、実際の制御過程を観測して総合的に判断する必要がある。このためには、パレート最適解集合全体を近似的に求め、解を総合的に比較して最良の解を対話的に選択することが有効であると考えられる。本研究では、微分可能性を前提としない多目的最適化手法として Vector Simplex 法を提案し、これにより、パレート最適解の近似集合が得られることを示す。さらに、対話的に最適化を行うことにより、大局的な見地から比較を行い、より厳密なパレート最適解を得る対話的方法を提案する。本方法により、多目的制御のためのファジィ制御ルールの学習が可能であり、精度の高いルールが得られることを示す。

Interactive Learning of Multiobjective Fuzzy Control Rules by Multiobjective Nonlinear Optimization Method “Vector Simplex”

TETSUYUKI TAKAHAMA[†] and SETSUKO SAKAI^{††}

Learning of multiobjective fuzzy control rules can be defined as minimizing plural control errors simultaneously. The errors are obtained by the experiments. So, the learning problem is considered as a multiobjective optimization problem, in which the objective functions aren't differentiable. In this research, we propose a new optimization method “Vector Simplex” to obtain a set of Pareto optimal solutions approximately. The method doesn't assume the differentiability of the objective functions. We show that we can obtain the Pareto optimal solutions with considerably high accuracy by Vector Simplex method. Also, we propose an interactive method to obtain more accurate solutions with comparing the solutions globally. We show that we can learn the multiobjective fuzzy control rules by our interactive method.

1. はじめに

ファジィ制御知識は、ファジィ推論ルールの形式で表現され、専門家の経験的知識をルール化することが比較的容易である。このため、ファジィ制御^{1),2)}は、家電製品、車の自動変速器、地下鉄システムなど広範なシステムの制御に利用され、成功を収めてきている^{3),4)}。制御を行う場合には、制御対象をモデル化し、そのモデルあるいはオペレータの経験則に基づき制御ルールを設計し、ルールをチューニングするという手順をたどる。しかし、モデル化が困難な場合や、経験則が存

在しない、あるいは、存在しても個々の経験則に大きな違いがある場合にはこのようなアプローチをとることができない。このような場合に有効な方法は、制御を行いながら制御ルールを自動的に構成する方法、すなわち、制御学習である。

このようなファジィ制御ルールを学習する制御学習に関する研究はかなり早くから行われており、教師データから制御ルールを抽出する研究^{5)~7)}、人間の記述したルールを最適化する研究^{8),9)}、制御を実践してルールを獲得する教師なし制御学習に関する研究がさかんに行われている。しかしながら、制御変数が複数存在する多目的制御問題において誤差 0 に非常に近い状態までの制御を学習できる研究はまだ数少ない。ところが現実の制御問題においては、速度と車間距離の調整を目的とする自動車の制御や 3 次元上の位置を調整するロボットアームの制御のように、多目的制御において精密な制御が求められることも多い。本研究で

[†] 広島市立大学情報科学部
Faculty of Information Sciences, Hiroshima City University

^{††} 広島修道大学商学部
Faculty of Commercial Sciences, Hiroshima Shudo University

は特に、たとえば倒立振子における角度制御と位置制御のように、制御目標が複数存在する多目的ファジィ制御ルールの教師なし学習に着目する。

ファジィ制御ルールの教師なし学習の研究としては、強化学習¹⁰⁾を利用した研究^{11),12)}、ファジィクラシフィアシステム¹³⁾を利用した研究^{14)~16)}、遺伝的アルゴリズム¹⁷⁾を利用した研究^{18)~20)}、非線形最適化手法を用いた研究^{21)~23)}などがある。文献 12) は Bang-Bang 制御のための後件部ファジィラベルの組合せを、文献 14)~16), 18), 19) は後件部あるいは前件部と後件部両方のファジィラベルの組合せを最適化しているが、メンバシップ関数を最適化していないため、それほど精度の高い制御を実現できてはいない。文献 11) は倒立振子制御ルールのメンバシップ関数を最適化しており、角度制御については比較的精度の高い制御を実現しているが、位置制御については十分ではない。文献 21)~23) では、非常に多くの応用例を持つ推論方法である簡略ファジィ推論⁶⁾を採用し、ルールの後件部である実数値を非線形最適化手法で最適化することにより、後件部メンバシップ関数を最適化し、精度の高い制御を実現している。しかし、これらの研究はいずれも制御目的が単一であるか、複数の制御目的をなんらかの尺度で単一化しており、多目的ファジィ制御学習の研究とは考えにくい面がある。

文献 20) は、遺伝的アルゴリズムにより多目的問題を解く方法を提案しており、倒立振子における角度・位置制御に関して高い精度を実現している。しかし、ファジィ変数のレンジをダイナミックにスケールリングするファジィスケールリング制御²⁴⁾を利用しており、通常のファジィ制御ルールの学習という観点からは参考にしにくい面がある。また、最適のルールを選択するために、許容誤差を達成するまでの時間を最小化するという固定的な基準が導入されており、専門家ごとに基準が異なる場合に対応できない。さらに、学習に要する実験回数がかなり多いという問題点も指摘されている。

本研究では、これらの問題を解決するために、多目的非線形最適化手法を利用して対話的に専門家の基準を取り入れる方法を提案する。文献 21)~23) と同様に、簡略ファジィ推論ルールを採用し、後件部実数値を多目的非線形最適化手法で最適化することにより、広く使用されている形式であり、高い精度の制御を実現するファジィ制御ルールを学習できる。また、対話的に学習を進めることによって、専門家などの意思決定者ごとに異なる基準を取り入れることが可能となる。

通常、多目的計画法における対話的手法では、スカ

ラー化手法などで得られたパレート最適解を基準点とし、基準点における目的関数値のトレードオフ比などの局所的な選好情報を参考に対話的に選好解を導出する^{25),26)}。多目的制御学習では、複数の制御目標に対する制御誤差を目的関数とし、多目的最適化を行うことになる。しかし、最良の制御を選択するためには、実際に制御過程を観測し、許容誤差を達成するまでの時間やオーバシュート量などを総合的に考慮する必要があり、制御誤差のトレードオフ比などの局所的な情報のみから、最適解を決定することは困難である。このため、本研究では、以下のような対話的手法を提案する。

- (1) パレート最適解の全体的な近似集合を求める。
- (2) パレート最適解を総合的に比較し、集合中で最良の解を選択する。
- (3) 選択された解の近傍でより詳細なパレート最適解の部分集合を求める。

これにより、意思決定者はパレート最適解の全体的な傾向を把握しながら、目的関数値以外の情報も考慮した総合的な比較を行うことができる。さらに、(2)、(3)を繰り返すことにより、より意思決定者の選好に近い、より厳密なパレート最適解を、大局的な見地から対話的に求めることができる。

本対話的手法を実現するためには、パレート最適解集合を近似的に求める多目的非線形最適化手法が必要となる。また、制御学習では、目的関数値が実際の制御実験やシミュレーションによって得られ、目的関数と制御パラメータとの間に明確な関数関係が定義できなかったり、定義できても微分可能性を仮定できない場合が多い。このため、目的関数の微分可能性を前提とせず、目的関数値のみで最適化を行う必要がある。本研究では、このための手法として Vector Simplex 法²⁷⁾を提案する。Vector Simplex 法は、Nelder & Mead により提案された一目的非線形最適化手法 Simplex 法^{28)~30)}を、ベクトル値関数に適用できるように拡張した方法である。Vector Simplex 法では、最良解に対応する「より良いものがない解の集合」、最悪解に対応する「より悪いものがない解の集合」などを定め、後者の集合の要素を、Simplex 法で定義された鏡映、拡張、収縮などの操作により、順次改善して行くことによってパレート最適解集合を探索する。Vector Simplex 法は、目的関数値のみを用いる直接探索法の 1 つであり、目的関数の微分可能性を必要としない。

以下、2 章で多目的最適化問題とパレート最適解を定義し、3 章で Simplex 法について説明する。4 章で

Vector Simplex 法を提案し、5章で対話的最適化について説明する。6章で数値実験を行い、7章で多目的ファジィ制御ルールの学習について述べる。8章はまとめである。

2. 多目的計画問題

2.1 定義

一般に、与えられた制約条件のもとで、複数個の相競合する目的関数を最小化（あるいは最大化）する問題は多目的計画問題（あるいは多目的最適化問題）と呼ばれる。複数個の目的関数をベクトル値関数として形式化すれば、多目的計画問題は、ベクトル値最小化問題として以下のように定式化できる。

$$\begin{aligned} & \text{minimize } \mathbf{f}(\mathbf{x}) \\ & \text{subject to } \mathbf{x} \in X = \{\mathbf{x} \in R^n \mid \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} \end{aligned} \quad (1)$$

ここで、 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ は n 次元決定変数ベクトル、 $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))$ は m 次元ベクトル値目的関数、 $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_k(\mathbf{x}))$ は k 次元ベクトル値制約関数である。このように、多目的計画問題は k 個の不等式制約条件のもとで m 個の目的関数を同時に最小化する n 次元の決定変数ベクトルを求める問題として定式化されるが、本研究では $k = 0$ の制約のない多目的計画問題を対象とする。

多目的計画問題では、目的関数がベクトル値関数であるため、複数の目的関数を同時に最小化する解（完全最適解）が必ずしも存在しない場合がある。このような場合には、ある目的関数の値を改善するためには少なくとも1つは他の目的関数の値を改善するしかないような解であるパレート最適解が重視されている。

パレート最適解 $\mathbf{x}^* (\in X)$ は、以下の条件を満足する解である。

$$\mathbf{f}(\mathbf{x}) < \mathbf{f}(\mathbf{x}^*) \text{ となる } \mathbf{x} \in X \text{ が存在しない。}$$

ここで、ベクトル値の大小関係として、以下の比較演算子を定義しておく。

$$\mathbf{f} < \mathbf{f}^* \Leftrightarrow \forall i f_i \leq f_i^* \text{ and } \exists j f_j < f_j^* \quad (2)$$

$$\mathbf{f} \leq \mathbf{f}^* \Leftrightarrow \forall i f_i \leq f_i^* \quad (3)$$

2.2 従来の研究

本研究が対象とする、目的関数の微分可能性を前提とせずパレート最適解集合を求めるための手法については、遺伝的アルゴリズムを中心に研究が進められている。このような研究として、各目的関数について独立に選択を行う研究^{31),32)}、解の優越関係に基づいて選択を行う研究^{17),33)~35)}、それらを組み合わせた研究³⁶⁾などが提案されている。しかし、遺伝的アルゴ

リズムは通常離散空間を探索するため、実数空間で定義された問題を解く場合に、十分な精度が得られない可能性がある。本研究で提案する Vector Simplex 法は、実数空間上でパレート最適解集合を近似的に求めることができる。

実数空間上で対話的に選好解を求める手法として、Simplex 法を利用した ICS (Interactive Constrained Simplex) 法³⁷⁾が提案されている。ICS 法は、目的関数をスカラー化するためのウェイト空間上で、最良のウェイトを探索することにより、最良の解を選択する方法である。意思決定者は、各ウェイトに対応した解を比較してウェイトを順序付けすることにより選好情報を指定する。Simplex 法の操作により新しいウェイトが求まるたびに、それに対応する解の順序付けを繰り返し行う。各ウェイトに対応した解は別に一目的最適化手法で求めることになる。したがって、本研究と ICS 法とは以下の相違点がある。

- ICS 法は一目的最適化を2段階に組み合わせた方法であり、Vector Simplex 法とは異なり、スカラー化法の一つであると考えられる。
- ICS 法では、新たに求められた解候補が必ずしも意思決定者が選択した解の近傍解とはならない。本研究では、選択した解の近傍で複数の解候補を意思決定者に提示するため、意思決定者が選択した解の近傍に選好解が存在する場合に効率よく探索を進めることができる。
- ICS 法では、すべての解候補間に順序付けを行う必要があるが、本研究では各段階で最良と考える解を指定するだけでよいため、意思決定者の負担が少ない。
- ICS 法ではすべての解を一目的最適化により求める。本研究では、後述のように、大部分の解を Vector Simplex 法で高速に求め、意思決定者が選択した解の精度を向上させるときに一目的最適化を行う。このため、より効率的な探索が可能となり、制御学習のように目的関数の評価回数を少なくすることが重要な問題により適していると考えられる。

3. Simplex 法

本研究で対象とする Simplex 法は、Nelder & Mead により提案された非線形最適化手法である（線形計画問題におけるシンプレックス法とは異なる）。Simplex 法は、 R^n 上にいくつかの点を幾何的に配置し、それらの点での目的関数の値を比較することにより、最適解を探索する方法である。Simplex 法は、制約のない

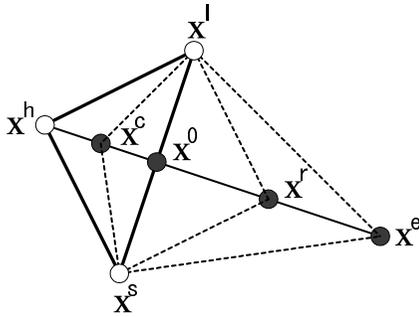


図1 Simplex 法の操作

Fig. 1 Operations in Simplex Method.

単一目的の微分不可能な非線形最適化問題を解く効率の良い方法として知られている。目的関数はスカラー値となるので、ここでは単に $f(x)$ と書くことにする。

R^n 上の $n+1$ 個のアフィン独立な点の凸包を単体という。単体の頂点を $x^i \in R^n$ ($i = 1, 2, \dots, n+1$) とし、頂点全体の集合を $U = \{x^i\}$ とする。 f の最小値を与える最良の頂点 x^l , f の最大値を与える最悪の頂点 x^h , f の 2 番目に大きな値を与える頂点 x^s を以下のように定義する。

$$x^l = \arg \min_i f(x^i) \quad (4)$$

$$x^h = \arg \max_i f(x^i) \quad (5)$$

$$x^s = \arg \max_{i \neq h} f(x^i) \quad (6)$$

さらに、 x^h 以外の頂点から生成される図心 x^0 を以下のように定義する。

$$x^0 = \frac{1}{n} \sum_{i \neq h} x^i \quad (7)$$

これらの点を元にして、以下のような基本的手続きを定義する(図1参照)。

鏡映 (reflection) x^h の x^0 方向への鏡映点 x^r を生成

$$x^r = (1 + \alpha)x^0 - \alpha x^h \quad (\alpha > 0)$$

拡張 (expansion) x^h を x^r 方向に拡張

$$x^e = \gamma x^r + (1 - \gamma)x^0 \quad (\gamma > 1)$$

収縮 (contraction) x^h を x^0 方向に収縮

$$x^c = \beta x^h + (1 - \beta)x^0 \quad (0 < \beta < 1)$$

縮小 (reduction) 全頂点を x^l の方向へ縮小

$$x^i = \frac{1}{2}(x^i + x^l)$$

Simplex 法では、初期単体から探索を開始し、最適解探索の各段階において、最良解、最悪解などを定め、それを基準に鏡映、拡張、収縮などの操作を行い、最悪解を順次改善することにより最適解を求める。通常、初期単体は頂点間の距離がすべて等しい基準形の単体

を用いる。初期単体は大きく取る方がよいとされている。収束条件は、以下のように各頂点における目的関数の値の差が十分に小さくなったときに満たされる。

$$\sqrt{\frac{1}{n+1} \sum_i (f(x^i) - \bar{f})^2} \leq \varepsilon \quad (8)$$

$$\bar{f} = \frac{1}{n+1} \sum_i f(x^i) \quad (9)$$

4. Vector Simplex 法

Simplex 法は基本的に全順序集合に対して適用できる手法である。本研究では、ベクトルの集合のような半順序集合に適用できるように Simplex 法を拡張した Vector Simplex 法を提案する。Vector Simplex 法は、制約のない多目的非線形最適化問題を解く際に利用することができる。

Vector Simplex 法では、頂点の全体集合 U は 1 つの解に収束するのではなく、パレート最適解集合へ収束することになる。このため、全体集合 U は $n+1$ 個の頂点の集合ではなく、最終的に獲得したいパレート最適解の個数 ($\geq n+1$) の点の集合とする。したがって、鏡映、拡張、収縮、縮小などの操作は全体集合 U から選択した $n+1$ 個の頂点に対して行うことになる。なお、初期集合は、ある範囲の値をランダムに発生したり、単体を大きく張るため超球上の点をランダムに発生することにより生成する。

半順序集合では一般に最良解や最悪解を一意的に決定することはできないため、最良解に対応する「より良い頂点のない頂点(極小元)の集合 U^l 」、最悪解に対応する「より悪い頂点のない頂点(極大元)の集合 U^h 」、2 番目に悪い解に対応する「それ以外の頂点の集合 U^s 」を以下のように定め、 U^h の要素を順次改善していくことによってパレート最適解集合を求める。

$$U^l = \{x^l \mid \exists x \in U \text{ s.t. } f(x) < f(x^l), x^l \in U\}$$

$$U^h = \{x^h \mid \exists x \in U - U^l \text{ s.t. } f(x^h) < f(x),$$

$$x^h \in U - U^l\}$$

$$U^s = U - U^l - U^h$$

Vector Simplex 法における処理の区分を図2に示し、アルゴリズムについて以下に説明する。

(1) 部分集合への分割

U の部分集合 U^l, U^h, U^s を求める。もし U^h が空ならば U と U^l が等しいので U をパレート最適解の候補集合として終了する。

(2) 単体の構成

U^h から 1 つ選択した頂点を x^h とし、 $U - \{x^h\}$

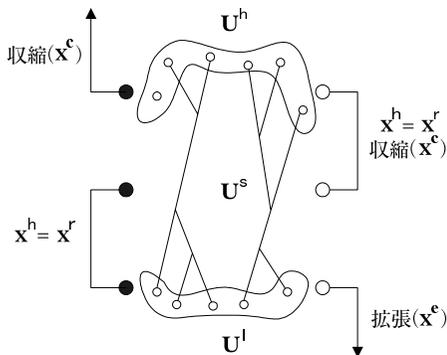


図2 Vector Simplex 法の処理の概要
Fig.2 Flow of Vector Simplex Method.

から選択した n 個の頂点の集合を U^0 とする。両者を合わせた $n+1$ 個の頂点で単体を構成する。

(3) 鏡映

鏡映点 x^r を求める。 $f(x^r)$ との比較により以下のような処理を行う。

(4) 拡張

U^l 中に x^r より悪い頂点が存在すれば、すなわち、 $f(x^r) < f(x)$ となる $x \in U^l$ が存在すれば、拡張を行い、そうでなければ (5) へ進む。拡張点 x^e を求める。 U^l 中に x^e より悪い頂点が存在すれば、 x^h を x^e で置換し、そうでなければ x^h を x^r で置換する。(1) へ戻る。これにより、 x^h は次の段階では U^l の要素となり、収束に近づくことになる。

(5) 置換

x^r が U^l 中の頂点と同等 ($f(x) < f(x^r)$ となる $x \in U^l$ が存在しない)か、あるいは、 U^s 中の頂点と同じかより良い ($f(x^r) \leq f(x)$ となる $x \in U^s$ が存在する)ならば、 x^h を x^r で置換し (1) へ戻る。そうでなければ (6) へ進む。置換により、 x^h は U^l あるいは U^s の要素となる。

(6) 収縮

U^h 中に x^r より悪い頂点が存在していれば、 x^h を x^r で置換する。この場合には、 x^h は U^s の要素に相当する (*)。

収縮点 x^c を求める。 x^c が x^h より改良されていれば、すなわち、 U^l 中の頂点と同等か、あるいは、 U^h 中に x^e より悪い頂点が存在していれば、 x^h を x^c で置換し (1) へ戻る。そうでなければ (7) へ進む。収縮により、 x^h は U^l あるいは U^s の要素となる。

(7) 縮小

Vector Simplex 法では、1 つの解へ収束するのではないことを考慮して、 x^h のみを U^l の要素に近

づける。 U^h の要素である x^h には、明らかに U^l 中に $f(x^l) < f(x^h)$ となる x^l が存在する。また (*) の場合にも、 x^h は U^s の要素に相当するので、条件を満足する x^l が存在する。このような x^l は 1 つ以上存在するので、その 1 つを選択して縮小を行う。(1) へ戻る。

したがって、 U^h に属していた頂点は、 U^l あるいは U^s の要素となるか、 U^l の要素に近づくことになる。このように、Vector Simplex 法のアルゴリズムは、Simplex 法における頂点 x^l, x^s, x^h を集合 U^l, U^s, U^h に対応させたものとなっている。

5. 対話的最適化

Vector Simplex 法により、近似的にパレート最適解集合を得ることが可能であるが、得られた解の間の中間的な解を求めたり、より厳密なパレート最適解を求めたいことも多い。このような場合には、近似解集合から適当な解を選択することにより、以下のような手順で、対話的に最適化を行うことができる。

(1) パレート最適解近似集合の提示

Vector Simplex 法によりパレート最適解の近似解集合を求め、比較しやすいように意思決定者にグラフで提示したり、目的関数値に基づいてソートして値で提示する。

(2) 意思決定者による選択

意思決定者が近似解集合中の解を詳細に検討しながら、大局的に比較し、対話的に近似解を選択する。

(3) パレート最適解の取得

多目的計画問題を一目的計画問題に変換し、選択された近似解に対応する厳密なパレート最適解を Simplex 法により求める。

(4) 近似解集合の詳細化

(3) で得られたパレート最適解の近傍の頂点を初期集合として Vector Simplex 法によりパレート最適解の近傍の近似解集合を得る。

(5) (2) へ戻る

これにより、局所的な情報に基づいて対話的に最適化を行うのではなく、目的関数値以外の情報も考慮して解を比較することにより、解の全体的な傾向を確認し、少しずつ目的関数値の範囲を狭めながら、最適化を行うことができる。

5.1 一目的計画問題への変換

意思決定者が選択した近似解に対応する厳密なパレート最適解を求めるために、多目的計画問題を一目的計画問題に変換する。変換の方法としては、目標計画法のように意思決定者が目標値を与える方法、妥協

計画法のように各目的関数の最小値からなる理想点を目標値とする方法, 近似解集合の傾向から目標値を推定する方法が考えられる. 制御学習においては, 目的関数値以外の総合的な判断が必要であり, 意思決定者が目標値を与えることは困難であるため, 後者の2つの方法について説明する.

各目的 j に対する目標値を \hat{f}_j とすれば, 目標値との差を最小化する一目的関数 $f_{goal}(x)$ は以下のように定義できる.

$$f_{goal}(x) = \begin{cases} \left(\sum_j w_j (f_j(x) - \hat{f}_j)^p \right)^{1/p} & (1 \leq p < \infty) \\ \max_j \{w_j (f_j(x) - \hat{f}_j)\} & (p = \infty) \end{cases} \quad (10)$$

w_j は, 各目的関数値のスケールを合わせるための定数であり, 各目的ごとの目標値との差が $[0, 1]$ の範囲となるように設定する. 意思決定者が選択した近似解を x^u とすると, 以下のように定義できる.

$$w_j = 1/|f_j(x^u) - \hat{f}_j| \quad (11)$$

本研究では, 意思決定者が選択した近似解よりもすべての目的が良くなる解を探索するため, $p = \infty$ とする.

方法1 理想点を目標値とする方法

各目的関数 f_j について, Simplex 法などの一目的最適化手法により最小値 f_j^* を求め, $f^* = (f_j^*)$ を理想点とする. 式 (10) において, 目標値 $\hat{f}_j = f_j^*$ とすればよい. この方法は, 図3において, 点線の矢印で示したように, $f(x^u)$ から f^* 方向へ最適化を進めることを意味する. なお, 図中の点線で示した曲線がパレート最適解の関数値である.

方法2 目標値を推定する方法

パレート最適解集合は, Vector Simplex 法により得られた近似解集合に対して, 目的関数値が少し小さくなる方向に存在する. 近似解の関数値と近いパレート最適解を求めるために, 近似解集合において各目的関数の最悪値を示す点 f^{max} から $f(x^u)$ への方向へ最適化を行う. このために, f^{max} から $f(x^u)$ へ向かう直線上の点を仮想的な目標値とする.

$$f_j^{max} = \max_{x^i \in U} f_j(x^i) \quad (12)$$

$$\hat{f}_j = f_j(x^u) - \lambda(f_j^{max} - f_j(x^u)) \quad (13)$$

ここで, λ は適当な定数であり, 大きくなるにしたがって $f(x^u)$ から離れた点となる. この方法は, 図3において, 実線の矢印で示したように, $f(x^u)$ から \hat{f} 方向へ最適化を進めることを意味する.

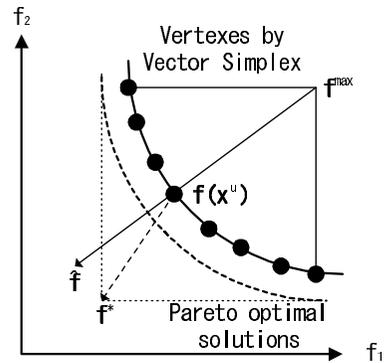


図3 目標値の推定方法
Fig. 3 Estimation of goal values.

5.2 パレート最適解の取得

Simplex 法を用いて一目的計画問題を解くことにより, 意思決定者が選択した近似解に対応するパレート最適解を求める. この際の初期単体は, Simplex 法で標準的に用いられている頂点間の距離 (S) がすべて等しい基準形の単体とし, 意思決定者が選択した頂点 x^u を起点とする $n+1$ 個の頂点で構成する. 距離 S は, 近似解集合の要素間の距離の最大値とする.

$$S = \max_{x^i, x^j \in U} \|x^i - x^j\| \quad (14)$$

この初期単体に対して Simplex 法を実行することにより, 意思決定者が選択した点の近傍のパレート最適解が求まる.

なお, 意思決定者が選択した解に近いパレート最適解を複数求めるために, x^u の近傍の点 x^v を N_v 個選択し, $f(x^v)$ から \hat{f} へ向かう方向にもパレート最適解を求める. したがって, $N_v + 1$ 個のパレート最適解集合 (V) が得られることになる.

5.3 近似解集合の詳細化

近似解集合を詳細化する際には, $N_v + 1$ 個のパレート最適解とその近傍の点により全体集合 (W) の初期集合を構成し, Vector Simplex 法を実行する. 初期集合は, $N_v + 1$ 個のパレート最適解と, x^u から得られたパレート最適解 x^{*u} を中心とするランダムに生成された直径 r の超球上の $|W| - N_v - 1$ 個の点から構成する. 直径 r は, パレート最適解集合 V の要素間の最大距離とする.

$$r = \max_{x^{*i}, x^{*j} \in V} \|x^{*i} - x^{*j}\| \quad (15)$$

6. 数値実験

Vector Simplex 法の性質を調べるために, 多目的最適化の例題としてよく取り上げられている, 2次関数

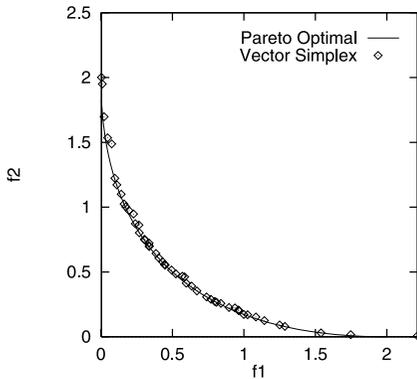


図4 数値実験の結果

Fig. 4 Result of numeric experiment.

を目的関数とする制約のない2目的計画問題に適用し、その結果について検討する。以降、Vector Simplex 法のパラメータについては、 $\alpha = 1$ 、 $\beta = 0.5$ 、 $\gamma = 2$ とする。

例として以下のような2次関数の2目的計画問題を取り上げる。

$$\begin{aligned} & \text{minimize } f_1, f_2 \\ & f_1(\mathbf{x}) = x_1^2 + x_2^2 \\ & f_2(\mathbf{x}) = (x_1 - 1)^2 + (x_2 - 1)^2 \end{aligned}$$

この問題におけるパレート最適解集合は、 $\{(x_1, x_2) | x_1 = x_2, 0 \leq x_1, x_2 \leq 1\}$ である。初期集合を原点を通る直径4の円上の点に設定し、全体集合の要素数 $|U|$ を50としてVector Simplex法を実行した際の目的関数 (f_1, f_2) のグラフを図4に示す。パレート最適解の関数値を線で、Vector Simplex法により求めた近似解の関数値を点で表している。なお、この結果を得るための目的関数の評価回数は388回であった。図から分かるように、かなりよい近似値が得られているため、対話的最適化の際に大局的な判断基準として用いるには十分であると考えられる。

より厳密なパレート最適解を得るため、対話的最適化を行う。対話的最適化のためのパラメータは、パレート最適解を求める際の近傍の頂点数 $N_v = 2$ 、Simplex法の収束精度 $\epsilon = 10^{-5}$ 、詳細化を行うときの全体集合 W の要素数を25、 $\lambda = 0.5$ とした。

意思決定者が大局的な比較から $f_1 = 0.4$ 程度の解が好ましいと判断したと仮定する。意思決定者が随時得られる近似解集合から f_1 が0.4に近い点を選択していく場合について説明する。最初の近似解集合の中でこの条件を満足する初期選択点は、 $(f_1, f_2) = (0.405133, 0.608218)$ である。

方法1では、理想点 $f^* = (0.0, 0.000007)$ を求め、

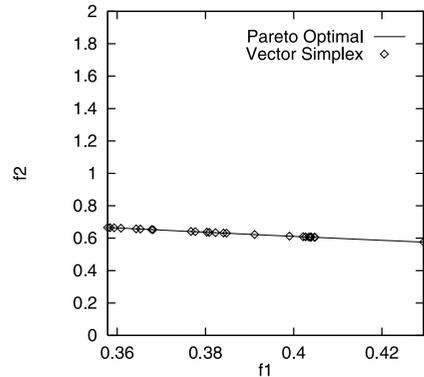


図5 数値実験の結果(1回目の対話後)

Fig. 5 Result of numeric experiment (after one interaction).

一目的最適化問題に変換してパレート最適解を求め、その近傍の点に対して再度Vector Simplex法を実行する。このとき得られた近似解集合を図5に示した。その後 $(0.399040, 0.612645), \dots, (0.400021, 0.611120)$ を選択し、 $(0.400004, 0.611140)$ の値を得た。このとき得られた解は $(x_1, x_2) = (0.447086, 0.447346)$ であり、 x_1 と x_2 の差が0.00026まで達している厳密な解である。なお、目的関数の評価回数は、対話的最適化のために956回、合計1,344回であった。

方法2では、目標値 $\hat{f} = (-0.499531, -0.088251)$ を求め、その後 $(0.400875, 0.610266), \dots, (0.399976, 0.611175)$ を選択し、 $(0.400001, 0.611144)$ の値を得た。このとき得られた解は $(x_1, x_2) = (0.447079, 0.447350)$ であり、差が0.000271まで達している厳密な解である。なお、目的関数の評価回数は、対話的最適化のために641回、合計1,029回であった。

パレート最適解の関数値のグラフ、すなわち図4の f_1 - f_2 グラフにおける $f_1 = 0.4$ での接線の傾きは $1 - \sqrt{5}$ 、それに直交する線分の傾きは $\frac{1}{\sqrt{5}-1} = 0.809$ であるため、 $f_1 = 0.4$ の近傍では傾き0.809に近い方向に解を探索することが望ましい。方法1における初期選択点から理想点への傾きは1.501であり、望ましい方向からずれている。これに対して、方法2における初期選択点から目標値への傾きは0.770であり、方法2の方がより望ましい方向に探索を行っている。なお、最終的に得られた解の質は方法1が、目的関数の評価回数は方法2がやや優れているが、ほぼ同等の結果である。

多目的最適化問題におけるパレート最適解は、目的関数を加重平均した一目的最適化問題の解として得られることが知られている(スカラー化手法)。ここでは比較のために、51通りのウェイトに対して、一目的

化した関数 $\mu f_1 + (1 - \mu)f_2$ ($0 \leq \mu \leq 1$) を Simplex 法により最適化し、51 個のパレート最適解を求める。初期単体を頂点間の距離がすべて 4 の原点を起点とする基準形の単体、収束精度 $\epsilon = 10^{-5}$ とし、区間 $[0, 1]$ を 50 等分した 51 通りの値の μ について実験し、51 個のパレート最適解を求めた。1 つの解を求めるために必要な目的関数の評価回数の平均は 43.82 回であり、Simplex 法では 50 個の解を求めるだけで 2,000 回以上関数を評価する必要がある。また、Simplex 法により得られた解における x_1 と x_2 の差の平均は 0.002501 であり、Vector Simplex 法で最終的に得た解よりもかなり精度が劣っている。したがって、提案した 2 つの方法は、スカラー化手法で 25~30 個程度のパレート最適解を求める評価回数で、意思決定者の選好に基づいたより厳密なパレート最適解を求めることができる効率の良い方法であるといえる。

7. 多目的ファジィ制御ルールの対話的学習

多目的ファジィ制御ルールの対話的学習の対象として、倒立振り子^{9),20)~23)}を取り上げる。倒立振り子の制御における状態変数は、鉛直方向に対する振り子の角度 θ 、振り子の角速度 ω 、台車の中央位置からの距離 x 、台車の速度 v である。操作変数は、台車を押す力 F である。倒立振り子の制御目標は、中央位置で直立させることであり、角度と位置に関する制御誤差を最小化するという以下のような 2 目的非線形計画問題として定義できる。

$$\text{minimize } f_\theta, f_x \tag{16}$$

$$f_\theta = \int_0^T |\dot{\theta}_t| dt / (\theta_{max} T)$$

$$f_x = \int_0^T |x_t| dt / (x_{max} T)$$

ただし、 θ_{max} 、 x_{max} は、それぞれ角度と位置の最大値であり、 f_θ 、 f_x を $[0, 1]$ の範囲に規格化するための定数である。

ファジィ制御ルールとしては、以下のように前件部変数を (θ, ω) および (x, v) の組合せとし、後件部変数を F としたものを採用した。

$$\text{if } \theta \text{ is } A_i^\theta \text{ and } \omega \text{ is } A_j^\omega \text{ then } F \text{ is } C_{ij}^{\theta\omega} \tag{17}$$

$$\text{if } x \text{ is } A_i^x \text{ and } v \text{ is } A_j^v \text{ then } F \text{ is } C_{ij}^{xv} \tag{18}$$

ただし、 $A_i^\theta, A_j^\omega, A_i^x, A_j^v$ はそれぞれ θ, ω, x, v に関する前件部ファジィ集合、 $C_{ij}^{\theta\omega}$ および C_{ij}^{xv} は F に関する後件部ファジィ集合である。

ファジィ推論には、and を代数積とする簡略ファジィ推論を採用し、後件部ファジィ集合を 1 つの実

表 1 制御ルール表

Table 1 Table of fuzzy control rule.

θ/ω	NB	NS	ZO	PS	PB
NB	p_1	p_2	p_3	p_4	p_5
NS	p_6	p_7	p_8	p_9	p_{10}
ZO	p_{11}	p_{12}	0.0	$-p_{12}$	$-p_{11}$
PS	$-p_{10}$	$-p_9$	$-p_8$	$-p_7$	$-p_6$
PB	$-p_5$	$-p_4$	$-p_3$	$-p_2$	$-p_1$
x/v	NB	NS	ZO	PS	PB
NB	p_{13}	p_{14}	p_{15}	p_{16}	p_{17}
NS	p_{18}	p_{19}	p_{20}	p_{21}	p_{22}
ZO	p_{23}	p_{24}	0.0	$-p_{24}$	$-p_{23}$
PS	$-p_{22}$	$-p_{21}$	$-p_{20}$	$-p_{19}$	$-p_{18}$
PB	$-p_{17}$	$-p_{16}$	$-p_{15}$	$-p_{14}$	$-p_{13}$

数値で表現した。前件部ファジィ集合は、5 種類のファジィラベル NB, NS, ZO, PS, PB を三角型メンバシップ関数で表現し、 (θ, ω, x, v) の範囲である ($\pm 15^\circ, \pm 120^\circ/s, \pm 3.0\text{ m}, \pm 3.0\text{ m/s}$) の両端を NB, PB とし、NS, ZO, PS をその間に均等に配置した。

時刻 t における前件部入力データ $(\theta_t, \omega_t, x_t, v_t)$ に対して、各ルールの前件部適合度を以下のように求める。

$$\mu_{ij}^{\theta\omega}(\theta_t, \omega_t) = \mu_{A_i^\theta}(\theta_t) \cdot \mu_{A_j^\omega}(\omega_t) \tag{19}$$

$$\mu_{ij}^{xv}(x_t, v_t) = \mu_{A_i^x}(x_t) \cdot \mu_{A_j^v}(v_t) \tag{20}$$

これらすべてのルールから得られた推論結果から以下の出力変数値が得られる。

$$F = \frac{\sum \mu_{ij}^{\theta\omega}(\theta_t, \omega_t) \cdot C_{ij}^{\theta\omega} + \sum \mu_{ij}^{xv}(x_t, v_t) \cdot C_{ij}^{xv}}{\sum \mu_{ij}^{\theta\omega}(\theta_t, \omega_t) + \sum \mu_{ij}^{xv}(x_t, v_t)} \tag{21}$$

倒立振り子の対称性を利用し、制御目標を達成した場合に $F = 0$ とすることを考慮し、これらのルールを制御ルール表で表現すれば、表 1 のような θ/ω 表と x/v 表となり、学習対象のパラメータは 24 個である。なお、 p_i は $C_{ij}^{\theta\omega}$ および C_{ij}^{xv} に対応する。

実験条件は、次のとおりである。倒立振り子の台車の質量を 1.0 kg、振り子の質量を 0.1 kg、振り子の長さを 0.5 m、レールの長さを $\pm 3\text{ m}$ とする。 F の最大値を 10 N、 $(15, 0, 0, 0)$ 、 $(-15, 0, 3, 0)$ という 2 つの初期状態から実験を行い、各目的関数の平均値を対象に最適化を行う。タイムメッシュ 0.02 s で離散化した Euler 法によりシミュレーションを行い、各初期状態から 500 ステップ (10 秒間) 行ったところで終了する。なお、これらの実験条件は、文献 9), 20)~23) と同一の条件である。

このような制御ルールの最適化では、目的関数の評価値は実際のシステムやシミュレータを動作させ一連の制御を行って得ることになる。したがって、できる

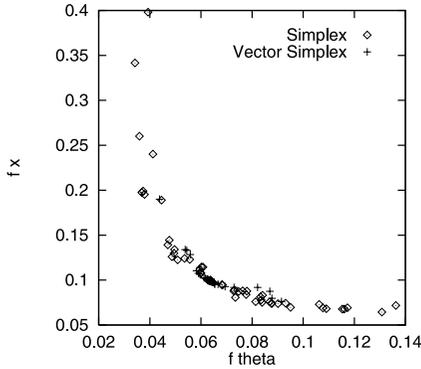


図 6 制御学習の結果 (対話前)

Fig. 6 Result of control rule learning (before interaction).

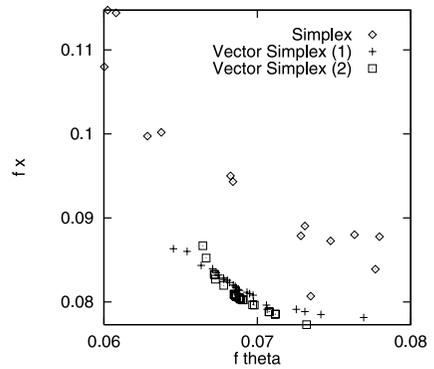


図 7 制御学習の結果 (対話後)

Fig. 7 Result of control rule learning (after interaction).

限り目的関数の評価回数を少なくする必要がある。このため、ここでは収束精度 $\epsilon = 10^{-3}$ とし、その他のパラメータは例 1 と同じものを用いた。また、初期集合は原点を中心とする直径 100 の超球上の点をランダムに 50 個発生させたものを用いた。

比較のために、一目的化した関数 $\mu f_{\theta} + (1 - \mu) f_x$ ($0 \leq \mu \leq 1$) を Simplex 法により最適化する。初期単体を頂点間の距離がすべて 100 の図心を原点とする基準形の単体、収束精度 $\epsilon = 10^{-3}$ とし、区間 $[0, 1]$ を 50 等分した 51 通りの値の μ について実験した。

図 6 に、スカラー化手法を用いた Simplex 法による解を菱形で、Vector Simplex 法による解を十字で示す。Vector Simplex 法の解は Simplex 法の解に比べて特定の目的関数が最小となる極端解の部分での収束が不十分であるが、多目的計画問題で重要視される妥協解の部分では同程度の解まで収束している。これらの解を得るための関数の評価回数は、Vector Simplex 法が 50 個の解を得るために 926 回、Simplex 法が 1 個の解を得るために平均 331 回である。したがって、Vector Simplex 法では、Simplex 法で 3 つの解を求める程度の評価回数で、かなり広範囲に精度の高い解が得られており、実際的な問題についても高速な近似解法となっている。

シミュレーション結果を観測して総合的な比較を行い、 $f_{\theta} = 0.07$ 程度の解が良いと判断したと仮定する。方法 1 では、理想点 $f^* = (0.034196, 0.071809)$ を求めた後、対話的に最適化を行う。最終的に得られた結果を図 7 中に十字で示した。評価回数 7,741 回で、 $(f_{\theta}, f_x) = (0.069758, 0.080053)$, $(0.070600, 0.079622)$ などの解を得た。方法 2 では、目標値 $\hat{f} = (0.063683, 0.042798)$ を求めた後、評価回数 7,742 回で、 $(0.069805, 0.079627)$, $(0.070770, 0.078824)$ などの解を得た。最終的に得た結果を図 7 中に四角で示

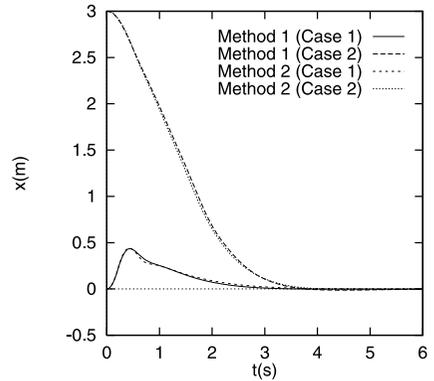
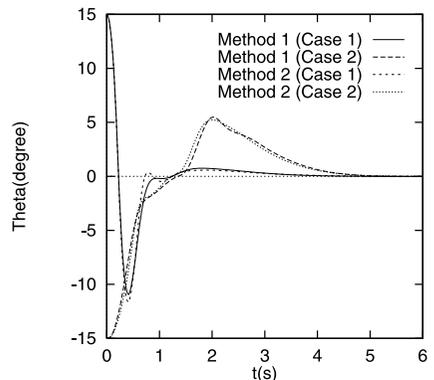


図 8 制御結果

Fig. 8 Result of control.

した。図から明らかなように、これらの解はすべてスカラー化手法で得られた解よりも優れた解となっている。また、 $f_1 = 0.7$ の近傍では、方法 2 が方法 1 よりも関数値がわずかに小さく、やや優れた解が得られている。

得られたファジィ制御ルールを用いて、各方法 (Method 1, 2) について学習に用いた 2 つの初期値 (Case 1, 2) から制御を行った結果を図 8 に示す。上が角度制御、下が位置制御の結果である。どちらの方

法においても両方の初期値について、角度・位置ともに急速に目標値に近づいており、明らかに精密な制御に成功している。

8. おわりに

非線形最適化手法である Simplex 法を半順序集合に適用できるように拡張し、制約のない多目的非線形最適化問題においてパレート最適解の近似集合を直接求めることが可能な Vector Simplex 法を提案した。さらに、近似解集合から意思決定者が適切な解を選択し、Vector Simplex 法を繰り返し実行するという、大局的な観点からの対話的最適化手法を提案した。

数値実験および多目的ファジィ制御ルールの学習において、微分可能性を仮定しない直接探索法の中で効率の良い方法として知られている Simplex 法をスカラー化手法と組み合わせた場合と比較することにより、Vector Simplex 法は、良い精度で非常に高速にパレート最適解の近似集合を得ることができる方法であることを示した。また、対話的最適化を行うことにより、全体的な解の様子を比較しながら非常に厳密なパレート最適解を得ることができることも示した。さらに、対話的最適化の際の一目的計画問題への変換方法として、理想点に基づく方法と Vector Simplex 法で得られた近似解集合の傾向から目標値を推定する方法を提案し、どちらの方法も有効であることを示した。

今後は以下のような領域で Vector Simplex 法について研究を進めていく予定である。

(1) 収束性の向上

Vector Simplex 法は高速性を重視し、パレート最適解集合への収束性をある程度犠牲にした手法である。しかし、最初からより厳密なパレート最適解集合を得たい場合もあるので、高速性を維持しながら収束性を向上させたい。

(2) 網羅性の向上

Vector Simplex 法は、極端解より妥協解を探索しやすいという望ましい性質を持つ。しかし、すべての解を網羅的に得たい場合もあるため、探索の広がりやを制御する方法も必要である。

(3) 制約付き問題への対応

多目的計画問題の多くは制約付き問題である。ペナルティ法などを用いることにより対処可能であるが、Vector Simplex 法自体で対処すべきである。

謝辞 本研究の一部は、広島市立大学特定研究費および日本学術振興会科学研究費補助金基盤研究(C)(2)(課題番号 11680462)による補助をいただいた。

参考文献

- 1) 菅野道夫：ファジィ制御，日刊工業新聞社(1988)。
- 2) 田中一男：応用をめざす人のためのファジィ理論入門—ファジィ集合からファジィ制御まで，ラッセル社(1991)。
- 3) Yasunobu, S. and Miyamoto, S.: *Automatic train operation by predictive fuzzy control*, North-Holland, pp.1-18 (1985)。
- 4) 長町三生(編)：ファジィ化製品開発の基礎と実際，海文堂出版(1991)。
- 5) Takagi, T. and Sugeno, M.: Fuzzy Identification of Systems and Its Application to Modeling and Control, *IEEE Trans. Systems, Man, and Cybernetics*, Vol.SMC-15, No.1, pp.116-132 (1985)。
- 6) 市橋秀友，渡辺俊彦：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, pp.429-437 (1990)。
- 7) 岸田一也，宮島廣美，福元伸也，村島定行：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, pp.585-593 (1995)。
- 8) 大西 颯：都市ごみ焼却プラントの学習型ファジィ制御システム，日本ファジィ学会誌，Vol.3, No.2, pp.357-365 (1991)。
- 9) 高濱徹行，阪井節子：制約付き非線形最適化手法 α 制約法によるファジィ制御ルールの最適化，電子情報通信学会論文誌，Vol.J82-A, No.5, pp.658-668 (1999)。
- 10) Sutton, R.S.: Learning to Predict by the Methods of Temporal Differences, *Machine Learning*, Vol.3, pp.9-44 (1988)。
- 11) 唐 政，小森雅和，石塚興彦，淡野公一：増強学習に基づく適応性をもつ ULR ファジーコントローラ，電子情報通信学会論文誌，Vol.J78-A, No.8, pp.1051-1058 (1995)。
- 12) 高濱徹行，阪井節子，小倉久和，中村正郎：強化学習法による離散値制御のためのファジィ制御規則の学習，日本ファジィ学会誌，Vol.8, No.1, pp.115-122 (1996)。
- 13) Rendon, M.V.: The Fuzzy Classifier System: A Classifier System for Continuously Varying Variables, *Proc. 4th ICGA*, pp.346-353 (1991)。
- 14) 古橋 武，中岡 兼，森川幸治，内川嘉樹：多段ファジィクラシファイアシステムによる制御知識獲得，日本ファジィ学会誌，Vol.6, No.3, pp.603-609 (1994)。
- 15) 古橋 武，中岡 兼，森川幸治，内川嘉樹，前田 宏：ファジィクラシファイアシステムによる知識発見に関する一考察，日本ファジィ学会誌，Vol.7, No.4, pp.839-848 (1995)。
- 16) 中岡 兼，古橋 武，内川嘉樹，前田 宏：ファジィクラシファイアシステムの報酬と信頼度割り当てに関する一提案，日本ファジィ学会誌，Vol.8,

- No.1, pp.65–71 (1995).
- 17) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley (1989).
 - 18) 高濱徹行, 宮本誠司, 小倉久和, 中村正郎: 遺伝的アルゴリズムによるファジィ制御規則の学習, 第8回ファジィシンポジウム講演論文集, pp.241–244 (1992).
 - 19) Chin, T.C. and Qi, X.M.: Genetic algorithms for learning the rule base of fuzzy logic controller, *Fuzzy Sets and Systems*, Vol.97, No.1, pp.1–7 (1998).
 - 20) 高濱徹行, 阪井節子: 遺伝的アルゴリズムによる多目的ファジィスケールリング制御規則の学習, 電子情報通信学会論文誌, Vol.J81-D-II, No.1, pp.119–126 (1998).
 - 21) 阪井節子, 高濱徹行: 非線形最適化手法を用いた倒立振り子ファジィ制御規則の学習, 第13回ファジィシステムシンポジウム講演論文集, pp.61–64 (1997).
 - 22) Sakai, S. and Takahama, T.: Learning fuzzy control rules by α constrained Simplex method, *Proc. 4th Asian/Pacific International Symposium on Instrumentation, Measurement and Automatic Control*, pp.240–246 (1999).
 - 23) 高濱徹行, 阪井節子: α 制約 Simplex 法によるファジィ制御ルールの学習, 電子情報通信学会論文誌, Vol.J83-D-I, No.7, pp.770–779 (1999).
 - 24) ムハマド ロムジ, 高濱徹行, 小高知宏, 小倉久和: 倒立二重振り子系に対するファジィ制御知識の表現とスケールリングによる適応制御, 日本ファジィ学会誌, Vol.8, No.3, pp.576–585 (1996).
 - 25) 坂和正敏: 非線形システムの最適化—目的から多目的へ, 森北出版 (1986).
 - 26) 中山弘隆: 対話型多目的計画法とその応用, オペレーションズ・リサーチ, Vol.36, No.9, pp.435–439 (1991).
 - 27) 高濱徹行, 阪井節子: 多目的最適化手法とファジィ制御規則の学習について, 第56回情報処理学会全国大会講演論文集(2), pp.72–73 (1998).
 - 28) Nelder, J. and Mead, R.: A Simplex Method for Function Minimization, *Computer J.*, Vol.7, No.4, pp.308–313 (1965).
 - 29) Kowalik, J. and Osborne, M.R.: *Methods for Unconstrained Optimization Problems*, American Elsevier Publishing Company (1968).
 - 30) 今野 浩, 山下 浩: 非線形計画法, 日科技連出版社 (1978).
 - 31) Schaffer, J.D.: Multiple Objective Optimization with Vector Evaluated Genetic Algorithms, *Proc. 1st ICGA*, pp.93–100 (1985).
 - 32) 村田忠彦, 石淵久生, 田中英夫: 遺伝的アルゴリズムによるフローショップ・スケジューリングと多目的最適化問題への応用, 計測自動制御学会論文誌, Vol.31, No.5, pp.583–590 (1995).
 - 33) Fonseca, C.M. and Fleming, P.J.: Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization, *Proc. 5th ICGA*, pp.416–423 (1993).
 - 34) Louis, S.J. and Rawlins, G.J.E.: Pareto Optimality, GA-Easiness and Deception, *Proc. 5th ICGA*, pp.118–123 (1993).
 - 35) 村川正宏, 吉澤修治: 近傍モデル遺伝的アルゴリズムによる多目的最適化, 情報処理学会論文誌, Vol.40, No.4, pp.1792–1800 (1999).
 - 36) 北野宏明(編): 遺伝的アルゴリズム2, 産業図書 (1995).
 - 37) 市川惇信(編): 多目的決定の理論と方法, 計測自動制御学会 (1980).

(平成 11 年 8 月 19 日受付)

(平成 13 年 9 月 12 日採録)



高濱 徹行 (正会員)

1982 年京都大学工学部電気第二工学科卒業。1987 年同大学院博士課程研究認定退学。同年福井大学工学部助手。1994 年同大講師。1998 年より広島市立大学情報科学部知能情報システム工学科助教授。非線形最適化, ファジィ制御を対象とした学習, 機械学習, 推論, CAI, 自然言語処理等に関する研究に従事。電子情報通信学会, 人工知能学会, 教育システム情報学会, 言語処理学会, IEEE 各会員。工学博士。



阪井 節子

1979 年福井大学教育学部卒業。1984 年大阪大学大学院基礎工学研究科数理系後期課程単位取得退学。1986 年甲子園大学経営情報学部講師。1990 年福井大学教育学部助教授。1998 年より広島修道大学商学部経営学科教授。ゲーム理論, 意思決定, ファジィ数理計画, GA によるファジィ制御, CAI 等に関する研究に従事。日本 OR 学会, 日本ファジィ学会, 日本生産管理学会各会員。工学博士。