ラディアルベシス関数ネットワークと
領域適応型遺伝的アルゴリズムを用いた最適設計*

(第1報, 制約条件のない場合における検討)

荒川 雅生^{*1}, 中山 弘隆^{*2}, 石川 浩^{*1}Optimum Design Using Radial Basis Function Network
and Adaptive Range Genetic Algorithms
(1st Report, Consideration in Unconstrained Optimization)Masao ARAKAWA^{*3}, Hirotaka NAKAYAMA and Hiroshi ISHIKAWA^{*3}Kagawa University, Dept. Reliability-based Information System Engineering,
2217-20 Hayashicho, Takamatsu, Kagawa, 761-0396 Japan

In this paper, we use the radial basis function network in order to approximate the fitness function of the genetic algorithms and try to obtain the approximate optimum results within the relatively small number of function call. The radial basis function networks (RBF) is a kind of neural network that is composed by the number of radial basis function in Gaussian distribution. RBF has learning system that is composed by additional learning of a basis function and a new data and forgetting of a basis function and an undesirable data. Thus the key issues in RBF are to give new data and to place basis function. So that if we can give these values appropriately, we can carry out approximate optimization even in the case that the optimum solutions are outside the range of the initial settings. Together with the adaptive range genetic algorithms that are proposed to treat mixed variable optimization, we will propose the way to give a new data and basis function. In this study, we have shown the effectiveness of the proposed method through simple numerical examples in unconstrained optimum design case.

Key Words: Design Engineering, Optimization, Adaptive Range Genetic Algorithms, RBF Networks, Neural Network

1. 緒 言

近年, ハードウェア, ソフトウェア両面からの急速な進歩により, 最適設計も実用的な領域に達しつつある. その中心的な役割を演じてきたのが, 商用の最適化パッケージであり, その中では, 解析回数を抑え, 実用的な時間内に結果が得られるように, 様々な近似最適化の工夫がなされている⁽¹⁾. その基礎となっている近似理論はL. A. Schmit⁽²⁾らによって開発されたテイラー展開の利用である. ここでは, テイラー展開を基本としているため, 最適化の途中に得られた点(現在点)まわりの感度情報を基に, 線形性の高い近似式を導入する努力が払われていたと考えよう. しかしながら, 感度情報を基に近似式を生成する方法では, 非線形性の高い問題, 設計変数が多くなってしまった場合, そして, 感度を求めることが基本的に困難, あるいは, 非常に時間がかかってしまうため, 感度を求めることが実用的でない問題に対しては適用が困難であった.

そこで, 考えられたのが, 多点近似であり, なるべ

く数少ないデータを基に, 関数系の応答を近似する応答曲面法であろう. 応答曲面法では, 直交表に基づいて数少ないデータから基の関数系の応答面を簡単な関数に近似し, 最適解を求めていくことを数回繰り返すことにより, 大局的な近似最適解を求めようとする研究である^(3, 4, 5). この手法では, 最初のうちは大まかな把握から大局的な解が得られるであろう近傍を探索し, 徐々に探索領域を狭め, 近似の精度を上げていくことを目的としている. この方法は, 非常に簡潔であり, しかも, 対象とする関数系に関係なく, 数少ない解析回数で有効な近似解を導き出すことが可能である. しかしながら, 近似式の関数系(基底)をあらかじめ決めなければならない. 多点近似に共通した性質であろうが, 簡単な基底を用いれば, 繰り返し回数がいたずらに多くなり, 複雑な基底を用いれば1回の繰り返しに必要なデータ数が多くなってしまいうなど, 適当な基底の選択が容易ではない^(6, 7). 通常, 2次形式が用いられているが, 直交表を用いてパラメータの同定に必要なデータ数を減らすために, 2次形式にも制限が加わっており, その意味において, 有効な適用範囲は非線形性の弱い範囲までと考えるのが妥当であろう. そのため, 多峰性のある問題への適用は一般的には困難であった. さらに, 得られた近似式の有効な

* 原稿受付 2000年6月29日.

^{*1} 正員, 香川大学工学部(〒761-0396 高松市林町2217-20).^{*2} 甲南大学(〒658-8501 神戸市東灘区岡本8-9-1).

E-mail: arakawa@eng.kagawa-u.ac.jp

範囲は、近似解の近傍に限られ、その上、基底が簡単なものの場合、値そのものは信用できても、感度は信用できるものではないことが多々起こりうるものと考えられる。これに対して、応答曲面法では、近似領域を分割することによって対処することがあるが、今度は、領域の分割における問題点、分割された領域の境界における連続性の問題など、根本的に解決困難な問題が生じてしまい、最適解の値を求めること以外の目的で近似関数を求めようとする場合には根本的な解決は困難であった。

応答曲面法を用いて、元々ある複雑な関数系を近似しながら最適化をはかる際に、設計者が要求することは果たしてどんなことなのであろうか。以下に簡単にまとめてみた。

- (1) 数少ない解析回数（設計現場で耐えうる実用的な回数）で良好な近似大局解を得ること。
- (2) あらかじめ基の関数系に対する知識を必要としないこと。（基底による影響を考えずに済むこと）
- (3) 関数系が複雑でも対応できること。
- (4) 近似解が求まった後に、基の関数系の全容を大まかでもよいので把握できること。

などがあげられよう。(1)は近似最適化ならばどんな方法であっても目的とするところである。(2)は仮に知識があれば、中間変数を用いた近似を用いればよいのでその裏返しでもある。(3)については大局解を得るための障害であらう。(4)は、他の方法では考えられないので、応答曲面法を用いる最大の利点とも考えられる。

本研究では、上記の目的を満足する近似法としてガウス分布を基底とするラディアルベース関数ネットワーク(RBF)を用いることにした^(8,9)。RBFを用いる場合、応答面は多くの場合多峰性を有することになる。そこで、多峰性に強い領域適応型遺伝的アルゴリズムを用いて最適化を行うこととした。本報では、RBFを用いた近似の特性を明らかにするとともに、基礎的な検討として多峰性を有するベンチマーク問題に対して、制約条件の無い最適化問題を設定し、提示した手法の有効性を検討することを目的とした。

2. RBF ネットワーク

RBF ネットワーク^(8,9)は、ニューラルネットワークの一種である。このネットワークでは基底関数として、ガウス分布を考えており(図1a)、その重ね合わせによって応答曲面を生成する(図1b)。すなわち、ネットワークの出力は、

(a) Radial Basis Function (b) RBF output
Fig. 1 RBF network

$$y_i(\mathbf{x}) = \sum_{j=1}^m w_{ij} h(\mathbf{x}, \mathbf{c}_j, r_j) \quad (1)$$

ここで、基底関数は

$$h(\mathbf{x}, \mathbf{c}, r) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}\|^2}{r^2}\right) \quad (2)$$

で与えられる。このとき、 $\mathbf{c}_j$ は基底関数の中心位置であり、 r_j はその半径である。 $\|\cdot\|$ はノルムを表すものとする。さて、既存のデータを基にネットワークのエネルギーを計算する。

$$E_i = \sum_{j=1}^p (y_j - y_i(\mathbf{x}_j))^2 + \sum_{k=1}^m \lambda_{ik} w_{ik} \quad (3)$$

となる。ここでは、出力*i*に関するエネルギーの計算をしている。エネルギーの第2項目は、各基底の重みに対する重みであり、特定の基底に引っ張られ過ぎるのを防ぐ目的があるほか、学習課程において正則性を保証するための項である。ニューラルネットワークの学習とはRBFネットワークの場合、適切な重みを決定する問題に帰着できる。これは、エネルギーの最小化を求めることに他ならない。いま、行列形式に整理して、 $\mathbf{A} = (\mathbf{H}^T \mathbf{H} + \mathbf{\Lambda})$ としたとき、エネルギーの最小化は

$$\mathbf{A} \mathbf{w}_i = \mathbf{H}^T \mathbf{y}_i \quad (4)$$

として表される。ここで、

$$\mathbf{H} = \begin{bmatrix} h(x_1, c_1, r_1) & h(x_1, c_2, r_2) & \dots & h(x_1, c_m, r_m) \\ h(x_2, c_1, r_1) & h(x_2, c_2, r_2) & \dots & h(x_2, c_m, r_m) \\ \vdots & \vdots & \ddots & \vdots \\ h(x_p, c_1, r_1) & h(x_p, c_2, r_2) & \dots & h(x_p, c_m, r_m) \end{bmatrix}$$

$$\mathbf{\Lambda}_i = \begin{bmatrix} \lambda_{i1} & 0 \\ 0 & \lambda_{im} \end{bmatrix}$$

である。以上の結果より、

$$\mathbf{A}^{-1} = (\mathbf{H}^T \mathbf{H} + \mathbf{\Lambda})^{-1} \quad (5)$$

を用いて、求めたい重みは、

$$\mathbf{w}_i = \mathbf{A}^{-1} \mathbf{H}^T \mathbf{y}_i \quad (6)$$

で求まる。RBFネットワークの学習は、 $\mathbf{A}^{-1}$ を求めることに他ならない。

このように、RBFネットワークの学習は逆行列を求める問題に帰着できるので非常に学習が早いという性質を有している。さらに、その性能を特徴づけるも

のとして、追加学習と忘却が存在する。

2. 1 追加学習 階層型のニューラルネットワークで最も有名なバックプロパゲーション法の場合、教師信号として新しいデータが追加されると改めてはじめて学習をやり直さなければならない。これに対して、RBFでは、追加学習則がそれぞれデータの追加、基底関数の追加のために整備されている。

2. 1. 1 データの追加 いま、新しいデータセットが加わった場合を考える。ここで、

$$H_{p+1} = \begin{bmatrix} H_p \\ h_p^T \end{bmatrix} \quad (7)$$

とすると、

$$h_p^T = \{h(x_{p+1}, c_{1p}), h(x_{p+1}, c_{2p}), \dots, h(x_{p+1}, c_{mp})\} \quad (8)$$

このとき、 A^{-1} は

$$A_{p+1}^{-1} = A_p^{-1} - \frac{A_p^{-1} h_p h_p^T A_p^{-1}}{1 + h_p^T A_p^{-1} h_p} \quad (9)$$

として、簡単に学習結果が計算できる。このようなデータの追加則があることは、最適化の過程において新しく得られたデータを加えながら、ネットワークの精度を向上させていけることを意味しており、あらかじめ必要以上に多くのデータを用意する必要が無いことから非常に有用である。

2. 1. 2 基底関数の追加 いま、新しい基底関数が加わった場合を考える。ここで、

$$H_{m+1} = [H_m \quad h_{m+1}] \quad (10)$$

とすると、

$$h_{m+1}^T = \{h(x_{1m}, c_{m+1}), h(x_{2m}, c_{m+1}), \dots, h(x_{pm}, c_{m+1})\} \quad (11)$$

このとき、 A^{-1} は

$$A_{m+1}^{-1} = \begin{bmatrix} A_m^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{1 + h_{m+1}^T P_m h_{m+1}} \begin{bmatrix} A_m^{-1} H_m^T h_{m+1} \\ -1 \end{bmatrix} \begin{bmatrix} A_m^{-1} H_m^T h_{m+1} \\ -1 \end{bmatrix}^T \quad (12)$$

として簡単に学習結果が計算できる。最適化の過程において、初期的に与えた基底関数によって形成される応答曲面では精度を向上できない場合、新しい基底関数を追加しながら、より精度の高い応答曲面を形成することは非常に重要なことである。この性質を用いることによって、初期的によそしていなかったところ大局解がある場合にも、応答曲面の近似を追随させることが可能であり、非常に重要な学習要素である。

2. 2 忘却 人間の学習の過程で不必要な記憶を忘れていくことは非常に重要な要因のようである。一節では、幼児は非常に多くの言語理解能力があり、その一つ一つを忘れていくことによって、母国語に対

する反応を高めていくそうである。実際に近似最適化を考える場合でも、特定のところにデータが偏ってしまったり、全体の傾向を把握する上で不必要になったデータや基底は学習効果を悪くするだけなので、忘却してしまいうことが望ましい。

2. 2. 1 データの忘却 いま、忘却するデータがデータセット内の最後の要素であると考え、このように考えても一般性を失うことはない。

$$H_p = \begin{bmatrix} H_{p-1} \\ h_p^T \end{bmatrix} \quad (13)$$

とすると、

$$h_p^T = \{h(x_p, c_{1p}), h(x_p, c_{2p}), \dots, h(x_p, c_{mp})\} \quad (14)$$

このとき、 A^{-1} は

$$A_{p-1}^{-1} = A_p^{-1} + \frac{A_p^{-1} h_p h_p^T A_p^{-1}}{1 - h_p^T A_p^{-1} h_p} \quad (15)$$

として、簡単に学習結果が計算できる。学習データの密度が必要以上に濃くなる場合、他の部分に比べてその部分だけ、データ過剰の状態になり、全体としての傾向の把握が困難になってしまう。そこで、密度が濃くなった場合、質の高いデータのみ残して、忘却する事が重要である。

2. 2. 2 基底関数の忘却 いま、忘却する基底関数が基底関数セット内の最後の要素であると考え、このように考えても一般性を失うことはない。中山は基底関数の忘却に対して、新たに以下のような方法を提案している⁽⁹⁾。

$$H_m = [H_{m-1} \quad h_m] \quad (16)$$

とすると、

$$h_m^T = \{h(x_{1m}, c_m), h(x_{2m}, c_m), \dots, h(x_{pm}, c_m)\} \quad (17)$$

このとき、

$$\begin{bmatrix} A_{m-1}^{-1} & 0 \\ 0 & 0 \end{bmatrix} = A_{m+1}^{-1} + \frac{1}{\lambda_m + h_m^T P_{m-1} h_m} \begin{bmatrix} H_{m-1}^+ (I_p - P_{m-1}) h_m \\ -1 \end{bmatrix} \begin{bmatrix} H_{m-1}^+ (I_p - P_{m-1}) h_m \\ -1 \end{bmatrix}^T \quad (18)$$

として簡単に学習結果が計算できる。ここで、

$$P_{m-1} = P_m + \frac{P_m h_m h_m^T P_m}{\lambda_m - h_m^T P_m h_m} \quad (19)$$

$$\begin{bmatrix} H_{m-1}^+ \\ 0^T \end{bmatrix} = H_m^+ - \frac{H_m^+ h_m h_m^T}{h_m^T h_m} \quad (20)$$

であり、 H_m^+ はMoore-Penroseの一般化逆行列であり、 h_m^+ はそのm行目のベクトルである⁽¹⁰⁾。基底関数は応答曲面を決めるのに非常に重要な要素である。基底関数の形から考えると、必要以上に密集することは効率的でない。できるだけ効率よく、適切な位置、半径を持った基底を設定することが重要であり、そのためには不必要な基底を忘却することが必要である。

Fig. 2 Illustrative sketch of adaptive range expression of continuous variables

3. 領域適応型遺伝的アルゴリズム

紙面の都合上, 詳細は参考文献^(11, 12, 13)に譲るものとし, ここでは, 基本的なところのみを記すものとする。

3. 1 連続変数への対応 本手法では初期世代では従来までの手法と同じである。第2世代目からは, 前世代における遺伝子操作終了後に生き残った個体の各設計変数の平均値 μ_i と標準偏差 σ_i を用いて

$$R(p_i) = \begin{cases} \mu_i - \sqrt{-2\sigma_i^2 \ln \left(LB + \frac{(UB-LB)C(p_i)}{2^{m_i-1}-1} \right)} & \text{for } C(p_i) < 2^{m_i-1} \\ \mu_i + \sqrt{-2\sigma_i^2 \ln \left(LB + \frac{(UB-LB)(C(p_i)-2^{m_i-1})}{2^{m_i-1}-1} \right)} & \text{for } C(p_i) \geq 2^{m_i-1} \end{cases} \quad (21)$$

と表される。この概略は図2に示すとおりであり, システムパラメータLB, UBの間を等間隔に分割し, それぞれのy座標に対応するx座標の値をもって対応する p_i の連続変数とするものである。

3. 2 離散変数に対して 本研究では, 式(2)に示されるように $x_{i,min}$ の値を前世代の整数値の平均値 μ_i を基に操作する手法を提示する。すなわち, 整数においては,

$$DI(p_i) = x_{i,min} + C(p_i) \quad (22)$$

である。ここで,

$$x_{i,min} = \text{Int}(\mu_i + 0.5) - 2^{m_i-1} \quad (23)$$

もし, この値が, 設定した上下限値を越えてしまう場合, 致死遺伝子を生じさせないために,

$$\text{if } x_{i,min} < \text{lower}_i, \text{ then} \\ x_{i,min} = \text{lower}_i \quad (24)$$

$$\text{else if } x_{i,min} + 2^{m_i} - 1 > \text{upper}_i, \text{ then}$$

$$x_{i,min} = \text{upper}_i - 2^{m_i} + 1$$

とすればよいことがわかる。連続変数の場合と同様に最良解を保証する場合,

$$\text{if } x_{i,min} > \text{Max value}_i, \text{ then} \\ x_{i,min} = \text{Max value}_i \quad (25)$$

$$\text{else if } x_{i,min} + 2^{m_i} + 1 < \text{Max value}_i,$$

$$x_{i,min} = \text{Max value}_i - 2^{m_i} + 1.$$

とすればよいことがわかる。

さらに, 所与の離散値を扱う場合, 式(25)で与えられた整数値をさす離散値を与えればよい。すなわち,

$$DC(p_i) = \text{Database}[[DI(p_i)]] \quad (26)$$

となる。

この手法を用いる場合, 従来まで問題になっていた離散点の数とビット数の不適合によって生じる致死遺伝子の問題点が生じないなどの利点があり, これは, 収束性の向上へ向けて大きな寄与をするものと考えられる。

4. 提示する手法

本研究では, 多点0次感度による近似解析器としてRBFネットワークを用い, 最適化器として領域適応型遺伝的アルゴリズムを用いることとした。以下にその手順を示す。

4. 1 初期データについて

遺伝的アルゴリズムを用いた最適化を考える際に, 初期的に探索領域を設定する。この領域を初期的に関数近似を行う領域と考え, この領域内に初期データを設けることとした。この際に初期探索領域内に偏りなくデータをもうけることが重要である。単純に考えれば, 領域内の中心から境界までを等分割して, その格子点をデータとして採用すればよい。あるいは, 領域内の関数系の対称性を仮定してしまい, 実験計画法に基づいてデータをばらつかせることも考えられる。しかしながら, このような対応では, 確かに万遍なく, しかも, ある程度確実に初期領域内の関数近似を行うことが可能である。しかしながら, 予め用意しなければならないデータ点数が設計変数の増加とともに爆発的に増えてしまったり, 対称性の仮定が不適切である可能性などを考慮すると必ずしも得策ではない。そこで, 以下のようなアルゴリズムを提示し, ここではそれを用いることとした。

[データ分散アルゴリズム]

いま, p 個のデータを用意するものとする。このとき,

- (1) pn 個のデータを探索領域内にランダムに用意する。(以下の例題では $n=3$ とした)
- (2) ランダムに一つを選ぶ。(k=1)
- (3) 選択された集団(k個)と選択されていないデータのユークリッド距離を求め, その最小値が最大のものを選ぶ。k=k+1とする。

$$\text{Select } i \text{ such that } \max_{1 \leq n \leq k} \min_{j \leq k} |x_i - x_j| \quad (27)$$

これを n 個選択されるまで繰り返す。

(4) 選択されたデータについて関数の計算をする。ここでは関数値を求めることを前提としているが、これは実験値でもかまわない。また、関数がブラックボックスであると考え、未知関数と考えることも可能である。

4.2 初期基底関数の配置 基底関数は応答曲面のおおよその形を決定づけるものである。RBFネットワークの場合、他の近似と異なり、関数系を決定づける基底関数の数がデータ点数よりも多くても適切な曲面を張ることが可能である。ここで、 m 個の基底関数を初期的に配置するものとする。本研究では遺伝的アルゴリズムの適応度関数の近似を行うため、応答曲面の最大化を目指しているものと考えられる。そこで、与えられたデータ点の中で適応度関数値が最も大きいものの回り ($[x_{\max} - brxar, x_{\max} + brxar]$) に基底関数をデータ分散アルゴリズムを用いて $m_{\max}$ 個配置することにした。同様に、基の関数系の全容を把握する目的で、適応度が最も小さいものの回りに基底関数を $m_{\min}$ 個配置した。残る $m - m_{\max} - m_{\min}$ 個の基底の配置であるが、これは一端、RBFネットワークを生成した後に、ネットワークの出力とデータ点の適応度の値が最も大きいところの周辺 ($[x - mrxar, x + mrxar]$) に配置するものとした。このような配置方法をここでは、便宜的に基底関数の調整配置と呼ぶことにする。基底関数の調整配置により、ある程度データ点の特徴を生かした応答曲面の生成が可能である。

4.3 データと基底関数の追加 RBFネットワークが生成されると、これを用いて近似最適化を行う。領域適応型遺伝的アルゴリズムを一定の世代 (必ずしも収束させる必要はない) 進めた後に、その最大値を基に、以下のように、データと基底関数の更新を行う。

- (1) 領域適応型遺伝的アルゴリズムで得られた最大値にデータと基底関数を配置する。
- (2) その近傍 ($[x_{\max} - brxar, x_{\max} + brxar]$) にデータ分散アルゴリズムを用いて $mp_{\max}$ 個のデータと基底関数を配置する。
- (3) 基底関数の配置されている領域内 ($c_i \in [c_{imin}, c_{imax}]$) に $m_{\max} - mp_{\max}$ 個のデータを、データ分散アルゴリズムを用いて用意する。この際、既存のデータもユークリッド距離を計算する対象の集合に含める。

- (4) 調整配置によって $p_{\max} - mp_{\max}$ 個の基底関数を配置する。

この際に新しいデータの近傍にあって新しいデータよりも適応度の低いデータは忘却するものとする。

4.4 データと基底関数の忘却 上記の要領でデータと基底関数の追加を続けると、大域的最適解が存在する近傍でのデータと基底関数が必要以上に増えてしまい、この領域のみデータ過剰状態に陥ることが考えられる。そこで、以下のようにデータや基底関数の忘却を行う。いま、データ内の計算結果で評価関数が最大値をとるデータを $x_{\max}$ とし、その最も近傍にある基底関数の中心を $c_{\max}$ とする。(多くの場合、両者は一致する。)

$$\exp \left(\frac{|a_i - c_{\max}|^2}{r_{\max}} \right) \geq \text{threshold} \quad (a = c, x) \quad (28)$$

となるデータ x_i 、基底関数 c_i を忘却するものとする。 threshold はしきい値であり、 $[0, 1]$ の間の任意の定数とする。

4.5 領域適応型遺伝的アルゴリズムとのリンク

データや基底関数は、あらかじめ決められた世代に決められた数だけ、決められた半径で追加、忘却されるものとする。その際、遺伝的アルゴリズムにとしては関数型が全く異なったものになってしまうので、その世代まで積み上げた情報は基本的には無意味な情報になってしまう。遺伝的アルゴリズムの適応力に依存することも考えられるが、ここでは、領域適応型遺伝的アルゴリズムの特性を生かして、平均値は一つ前の世代の生き残りによって決定し、分散はパラメータで定められた最大値を取るものとし、全個体外来種突然変異を起こすものとした。この場合、遺伝子情報はなくなるものの形質に関する情報は保存されと考えられる。

5. 数値計算例

提示した手法の有効性を検討する目的で、多峰性のある問題として

$$f(x_1, x_2) = 10 \exp(-(x_1 - 10)^2 - (x_2 - 15)^2) \sin(x_1) \quad (29)$$

を最適化設計の対象として考えた。さらに、最適化を困難にする目的で、あらかじめ想定していた探索領域に大域的な解が存在しない場合を想定することにした。ここでは、初期探索領域として $(x_1, x_2) = ([0, 10], [0, 10])$ とした。元々の関数の概要を図3に示す。この関数の大域的な最適解は $(x_1, x_2) = (7.89604,$

Table 1 Setting of each case

case	r	br	threshold
1	0.2	20.0	void
2	2.0	2.00	void
3	1.0	4.00	void
4	5.0	0.80	void
5	2.0	1.00	void
6	2.0	4.00	void
7	control		void
8	control		0.40
9	control		0.70
10	control		0.95

Table 2 Strategy for updating

generation	r	br
50	2.00	4.00
100	2.00	4.00
150	2.00	3.00
200	2.00	3.00
250	2.00	2.00
300	2.00	2.00
400	1.00	3.00
500	1.00	3.00
600	1.00	2.00
700	1.00	2.00
800	1.00	1.00
900	1.00	1.00

15.0000) であり、一例ではあるが、領域適応型GAsを利用することにより、3933回の解析回数でその解を得ている⁽¹¹⁾。

ここで、RBFネットワークのパラメータの差を見るために、表1に示すような各ケースについて、それぞれ5回試行した。表1中のcontrolとは、基底関数の半径を変更したものであり、その変更の状況を表2に示す。その他のパラメータとして、データや基底関数の追加は表2に示す世代に行うものとした。基底関

数の重みに対する重みは便宜的に $\lambda = 0.01$ とし、追加の度に基底関数を12個、データを15個追加候補とすることにした。また、遺伝的アルゴリズムで得られた結果の近傍には5個のデータ、基底関数を追加することとした。そして、調整配置の近傍の判定基準として、 r の0.2倍以内にあるものとし、また、近傍であってデータを削除する際の基準として、距離の2乗が0.1のものとするものとした。細かくは、これらのパラメータも最終結果に多少の影響を与えるが、探索性能を大きく左右するものではないため、本研究では定数として取り扱った。各々最適化の結果をまとめたものを表3に記す。表3には5回の試行うちで最も良い結果を得たもの、その際のデータ数、基底関数の数、5回の結果の平均、平均のデータ数、基底関数の数、5回の試行のうち実際に計算をしたデータで最良のもの、その平均をあげてある。この結果から、半径が小さすぎると(case 1)結果が悪いことが確認できる。これは、図1bに示す山(谷)がとがりすぎており、互いに連携を図れないため図3(a)のような応答曲面を近似することが困難であった

Fig. 4 Final surface of case 1

Table 3 Comparison of the results with different settings

		case 1	case 2	case 3	case 4	case 5	case 6	case 7	case 8	case 9	case 10
Best results out of 5 trials	x1	8.26734	7.90468	8.02147	7.76843	7.86042	7.8955	7.89885	7.89996	7.90144	7.89606
	x2	14.5062	15.006	15.5351	14.6582	14.8654	14.3511	15.0016	15.0531	14.871	15.0627
	f	8.7787	9.55816	9.45443	9.46834	9.55061	9.51836	9.55849	9.55818	9.5568	9.55815
	error	0.3817	0.00011	0.30206	0.13311	0.01939	0.42107	1E-05	0.00283	0.01667	0.00393
	data	200	200	200	200	200	200	200	175	183	190
	basis	162	161	162	161	147	162	162	146	155	160
Average out of 5 trials	x1	7.95378	7.90644	7.90762	7.79695	7.88792	7.89505	7.89949	7.89902	7.89856	7.89932
	x2	12.3971	15.1234	14.9877	14.5886	15.0141	14.4592	14.9888	14.8418	14.9419	15.4287
	f	8.58324	9.54914	9.4048	9.31263	9.53995	9.48371	9.557	9.48708	9.55366	9.51786
	error	8.99635	0.1116	1.3726	1.61269	0.1712	0.78329	0.01369	0.75441	0.04959	0.41567
	data	200.0	200.0	200.0	200.0	200.0	200.0	200.0	171.0	186.0	189.6
	basis	162.0	161.2	162.0	159.4	156.6	162.0	161.6	140.6	157.6	159.6
Best data out of 5 trials	x1	7.9958	7.90086	7.87216	7.85066	7.88632	7.89684	7.89618	7.90833	7.90972	7.88452
	x2	13.8877	15.1437	15.1582	14.9044	14.5809	15.1591	14.9906	14.7872	14.9537	14.8795
	f	9.15354	9.5481	9.44224	9.51306	9.52147	9.5122	9.55394	9.51526	9.55186	9.54065
Average of best data out of 5 trials	x1	7.92706	7.89675	7.90894	7.91597	7.89753	7.90635	7.9038	7.89708	7.89151	7.88652
	x2	15.5437	14.9844	15.1345	15.0058	14.9415	14.7592	14.9299	15.0331	14.9658	14.9229
	f	9.52562	9.55851	9.55599	9.55659	9.55819	9.55247	9.55777	9.55842	9.55832	9.55752

Fig. 5 Final surface of case 4

ためと考えられる。(図4)一方, case 4のように半径が大きい場合は, 精度は必ずしも高くないものの, 大局的な最適解の位置は大まかな把握ができていることがわかる。しかしながら, 不必要な重なりがあるために, 値そのものは, 大きな誤差を含んでいることが図5からも見て取れる。半径の大きさが不適切でないその他の8つのケースに関しては, 大局的最適解の位置, その応答曲面ともに概ね正確に把握している。case 2とcase 3を比較すると, 問題に応じた半径の設定が精度に与える影響が大きいことがわかる。しかしながら, 本来, 提示した手法を適用するのが妥当であると考えられる対象は関数系が不明である, あるいは, 複雑であり簡単な形式がわからない場合であることを考えると, この程度の誤差はやむをえないものと考えられる。半径の設定については, 今後のさらなる検討が必要となるところである。次に, 探索領域の大きさの問題であるが, case 2, 5, 6を見ると, 大きくとりすぎたcase 6の精度が他のものと比較して大きいことがわかる。この結果, 探索領域は徐々に広がっていくことが妥当であることが確認できる。この結果を基に半径や探索領域を可変するcase 7の設定を行った。その結果を図6に示す。図6に示されるように, 提示した手法では, 初期的に与えられた領域における応答曲面から徐々に大局的最適解の解の近傍の応答曲面を探索し, その後, 全体的な応答曲面を近似していることがわかる。その過程において, 大局的最適解の近傍に多くのデータと基底関数をおいていることがこの図からも見て取れる。そして, 600世代を超えたあたりから, 全体的な傾向を把握するようにデータが分散されていっていることも見て取れる。

このように, RBFでは, 与えられたデータ内の内挿において関数系に無関係にある程度精度よく近似応答曲面を生成していることがわかる。その過程において, 一つの基底関数の半径の設定の問題から, 多峰性

Fig. 6 Growth of response surface in case 7

の激しい応答曲面となっているが, これは, ARange GAs の性質により, 比較的少ない世代数で最大値の探索を行うことができている。その後, 大局的最適解の近傍と探索領域内に均一にデータと基底関数を配置する工夫により, 初期的な領域において大局的最適解が

得られないような問題に対しても対応していることがわかる。表3にあるように最も多い場合でも解析回数は200回である。これは、参考文献(11)の場合に3933回必要であったことを考えると非常に少ない数であり、近似を行うことの意義は十分あるものと考えられる。

さらに、この関数の設定は応答曲面法を用いて解く場合、設定にある種の偶然性がない限り大局解にはいたらず、さらに全体的な様相を捉えることも不可能な設定である。それにもかかわらず、提示した手法で大局的最適解、及び、全体的な様相を捉えることができたのは、1つの基底関数の影響が局所的であり、広範に渡って悪影響を与えないためである。(釣鐘上の関数のすそのでは0に漸近する。)その結果、応答局面法における探索領域の分割を自動的に内在していることによるものと考えられる。

表3のCase7,8,9,10の結果の比較を見てみたい。データと基底関数の忘却を入れることにより、確実に基底関数やデータの数は減っている。基底関数やデータの数が多くなると近似曲面を生成するのに時間がかかってしまうためこの数は、なるべく少ないほうが都合がよい。そのことと最終結果における精度を見てみると、数を減らしても精度の減り方は誤差範囲であると考えられる。Orrによると、投影マトリクスPの計算は提示した手法を用いてもデータ数の2乗の関数で計算量が増加しており、その数が多い場合、わずかな差でも最終的な計算時間の差は大きいので重要である。

6. 結 言

本論文では、未知関数の最適化へ向けた近似最適化手法の開発の一環として、制約条件のない場合の近似最適化手法を提示した。提示した手法は、

- ・ RBFによる関数近似
 - ・ 領域適応型GAsによる応答曲面の最適化
 - ・ RBFの近似にあわせたデータ、基底関数の配置の3つから成り立っている。提示した手法を多峰性のあるベンチマーク問題に適用した。その際に、予め大局的な最適解が初期設定外にある近似最適化にとっては比較的悪い設定から最適化を行った。その結果、
 - ・ 大局的最適解の良好な近似解が得られた
 - ・ 応答曲面は最適解近傍だけでなく全体に渡って比較的良好な曲面が得られた。
 - ・ 最適化の結果、解析回数は大幅に軽減できた。
- などの利点が確認された。特に第2点目は他の近似手法では得られない特性であり、本手法の有益性を示すものである。このような結果が得られれば、最適設計

の設定の変更にも対応できるほか、得られた解に不都合が生じた際に、次善の解を用意する際に、元の解の近傍で選ぶだけでなく、他の局所解を提示できる事などさまざまな利点が考えられる。一方で、

- ・ 基底関数の半径の与え方
- ・ データ、基底関数の数の減らし方

といった点に更なる工夫や改良が必要であることが判明した。この問題は今後検討を加えていく予定である。

謝 辞

本研究は南海育英会、及び、穴吹工務店の研究助成を受けて行った研究の一部である。ここに関係各位に謝意を表する。また、研究の遂行に当たって有益な助言をいただいた、山口大学の古川浩平教授、北海学園大学の杉本博之教授、早稲田大学の山川宏教授に謝意を表する。

参考文献

- (1) Vanderplaet, G., "Two Programs for Design Optimization", (1997), Proc. of OPID97, in CD-ROM
- (2) L.A. Schmit Jr., B. Farshi, "Some Approximation Concepts Concepts for Structural Synthesis", AIAA J., Vol.12, No.5, (1974), 692.
- (3) 荒川雅生, 深沢博樹, 山川宏, 機構論 [95-8(I)], (1995), 216.
- (4) 山崎, 韓, 機構論 [98-14], (1998), 165.
- (5) 矢島, 柏村, 干, 白鳥, "統計的設計支援システムによる設計最適化(第1回)", 計算工学, 4-3, (1999), 39-46.
- (6) 荒川, 山川, 機構論 [96-05(II)], (1996), 511.
- (7) 荒川, 山川, 機構論 [96-05(II)], (1996), 514.
- (8) Orr, M.J.L., Introduction to Radial Basis Function Networks, <http://www.cns.ed.ac.uk/people/mark/intro/intro.html>, (1996)
- (9) Nakayama, H., Yaganiuchi, S., Furukawa, K., Araki, Y., Suzuki, S., Nakata, M. "Additional Learning and Forgetting by RBF Networks and its Application to Design of Support Structures in Tunnel Construction", Proc. International ICSC/IFAC Symposium on Neural Computation (NC'98), (1998), 544-550.
- (10) Ben Israel, A., Greville T.N.E., Generalized Inverse: Theory and Application, (1976), John Wiley.
- (11) 荒川, 萩原, 機論, 63-616C, (1997), 4216-4223.
- (12) 荒川, 萩原, 機論, 64-621C, (1998), 1626-1635.
- (13) 荒川, 萩原, 機論, 65-638C, (1999), 4156-4163.