スプライン関数を用いたデータあてはめ ——遺伝的アルゴリズムによる節点の自動的な決定

吉本 富士市† 森山 真光†

スプライン関数を用いたデータあてはめ問題で良い解（近似関数）を得るためには，節点を変数として扱う必要がある．そのとき，解くべき問題は多変数で多峰性の連続系非線形最適化問題となる．したがって，大域的な最適解を求めることは困難である．本論文では，元の連続系最適化問題を離散系最適化問題へ変換し，それを遺伝的アルゴリズムを用いて解く方法を提案する．離散化した節点の位置の候補を遺伝子と見なして個体を構成することにより，連続系から離散系への変換を行う．また，適応度として赤池の情報量規準 AIC を用いることにより，AIC の意味で統計的に最適なモデルを探索できるようにする．提案する方法は，節点の適切な数と位置を自動的かつ同時に求めることができる特徴がある．また，それらの節点は遺伝的アルゴリズムの特性から大域的な準最適解である可能性が高い．さらに，多次元格子点データのあてはめ問題へ容易に拡張することができる．提案する方法の有効性を示すため，数値実験例をあげている．

Data Fitting with a Spline Function —Automatic Knot Placement by a Genetic Algorithm

FUJIICHI YOSHIMOTO† and MASAMITSU MORIYAMA†

In order to obtain a good result (that is, a good approximation) for data fitting with a spline function, frequently we have to deal with knots as variables. Then, the problem to be solved becomes a continuous nonlinear and multivariate optimization problem with many local optima. Therefore, it is difficult to obtain the global optimum. In this paper, we propose a method that we convert the original problem into a discrete combinatorial optimization problem and solve the converted problem by a genetic algorithm. We construct individuals by considering candidates of the positions of knots as genes, and convert the continuous problem into a discrete problem. We search for the statistically best model among candidate models by using Akaike's Information Criterion (AIC) as a fitness function. Our method can determine appropriate number and positions of knots automatically at the same time. By the characteristics of genetic algorithms, those knots are global suboptimal with a high probability. Moreover, we can easily extend our method for multi-dimensional mesh data. Numerical examples are given to show the effectiveness of our method.

1. まえがき

スプライン関数を用いたデータあてはめは，実験データの処理，形状モデリングなどの重要な要素技術の1つである．よく知られているように，良いスプライン関数（良いモデル）を得るためには，通常は節点の数と位置を適切に決める必要がある．このとき，節点を変数として扱わなければならず，解くべき問題は多変数で**多峰性**の連続系非線形最適化問題となる[1),2)]．したがって，大域的な最適解を求めることはきわめて困難である．

このため，上記の最適化問題をまともに解かない方法（簡便法）がいろいろ提案されてきている[1)~9)]．しかし，これらの方法は，許容誤差または平滑化係数（smoothing factor）が必要であり節点の数も多い[2),6)]，適切な初期節点の配置が容易でない[5)]，など改善の余地があるものが多い．したがって，「自動的に良いモデルを得る手法」の観点から見るとまだ十分とはいえない．ここで「良いモデル」とは，データの元にある関数（underlying function of data）をできるだけよく近似し，しかもモデルのパラメータができるだけ少ないスプライン関数のことを意味する．自動的に良いモデルを得るためには，モデルの良さを評価するための客観的な規準を用いて適切な節点の数と位置を自動的

† 和歌山大学システム工学部情報通信システム学科
Department of Computer and Communication Sciences, Faculty of Systems Engineering, Wakayama University

に決めるアルゴリズムが必要である．しかし，そのための汎用性の高いアルゴリズムは，まだほとんど提案されていない．

ところで，スプライン関数が良い近似関数となるために必要な節点の数と位置は，通常は厳密な意味での最適値でなくてもよく，準最適値であれば十分である[1),2),10)]．このため，上記の解くべき連続系の問題をある程度細かく離散化すれば，離散化された問題の最適解を元の問題の最適解の代わりに用いても十分良い結果を得ることができる．

本論文では，元の連続系の最適化問題を，離散系の組合せ最適化問題へ変換し，それを遺伝的アルゴリズム[11)~14)]を用いて解く方法を提案する[15)]．以下，遺伝的アルゴリズムを簡単のため GA（Genetic Algorithm）と呼ぶことがある．GA を用いることにより，適切な節点の数と位置を自動的かつ同時に決定することが可能となる．また，GA の評価関数として赤池の情報量規準 AIC（Akaike's Information Criterion）[16)]を用いることにより，データの元にある関数を AIC の意味で統計的に最もよく近似するモデルを自動的に選択することができる．提案する手法は，多次元格子点データの問題への拡張が容易である，並列計算への適合性も良い，などいくつかの優れた特徴を持っている．

2. スプライン関数によるデータあてはめ

2.1 1次元問題

あてはめを行うべきデータは，x 軸上の区間 $[a,b]$ 内で与えられ，

$$F_j = f(x_j) + \epsilon_j, \quad (j = 1,2,\cdots,N) \tag{1}$$

と表されるものとする．ここで，$f(x)$ はデータの元にある未知の関数（信号）であり，ϵ_j は平均値 0，分散 σ^2 の正規分布をする互いに独立な誤差であると仮定する．また，σ^2 の大きさは未知である．さらに，$f(x)$ は未知の関数であり，その良い近似関数を作ることがあてはめの目的である．データの横座標 x_j は，等間隔でなくてもよい．

必要な節点を $\xi_i\,(i = 1-m, 2-m, \cdots, n+m)$ と書くことにする．ここで，n は区間 $[a,b]$ の内部に配置する節点 $\xi_i\,(i = 1,2,\cdots,n)$ の数である．また m は，式 (3) に示すように，$f(x)$ の近似関数 $S(x)$ を表すために使う B–スプライン $N_{m,i}(x)$ の階数（次数+1）である．両端の m 個の節点はそれぞれ端点 a，b に重ね，

$$\left.\begin{aligned} a &= \xi_{1-m} = \cdots = \xi_0 \\ b &= \xi_{n+1} = \cdots = \xi_{n+m} \end{aligned}\right\} \tag{2}$$

とする．

このとき，近似関数 $S(x)$ は

$$S(x) = \sum_{i=1}^{n+m} c_i N_{m,i}(x) \tag{3}$$

と表すことができる[1)]．ここで，c_i は B–スプライン係数である．

式 (3) に含まれる B–スプラインは，次の漸化式を用いて容易に計算できる[17)]．

$$N_{1,i}(x) = \begin{cases} 1 & (\xi_{i-1} \le x < \xi_i), \\ 0 & (\text{otherwise}), \end{cases} \tag{4}$$

$$N_{r,i}(x) = \frac{(x-\xi_{i-r})N_{r-1,i-1}(x)}{\xi_{i-1}-\xi_{i-r}} + \frac{(\xi_i - x)N_{r-1,i}(x)}{\xi_i - \xi_{i-r+1}}, \quad (r = 2,3,\cdots,m). \tag{5}$$

式 (5) で，節点が分子および分母の両方に入っていることに注意したい．すなわち，節点は B–スプライン $N_{m,i}(x)$ の非線形パラメータである．したがって，式 (3) で与えられるスプライン関数 $S(x)$ は節点の非線形関数である．

最小二乗法を用いて式 (3) を式 (1) で与えられるデータへあてはめるとき，残差の 2 乗和 Q_1 は

$$Q_1 = \sum_{j=1}^{N} w_j \{S(x_j) - F_j\}^2 \tag{6}$$

となる．ここで，w_j はデータの重みであり，$N > n+m$ とする．また，記号 Q の添字 1 はデータの次元を表している．式 (6) を最小にする条件から B–スプライン係数 $c_i\,(i = 1,2,\cdots,n+m)$ を求めることができる．ただし，良い近似を得るためには内部の節点 $\xi_i\,(i = 1,2,\cdots,n)$ の数と位置を適切に決める必要がある．

以上の議論から分かるように，目的関数は式 (6) であり，その変数は B–スプライン係数 $c_i\,(i = 1,2,\cdots,n+m)$ および内部の節点 $\xi_i\,(i = 1,2,\cdots,n)$ である．目的関数に対して，B–スプライン係数は線形パラメータであるが，内部の節点は非線形パラメータであることに注意したい．式 (6) を最小化する問題は，多峰性の最適化問題となることが知られている[2),5)]．

2.2 2次元問題

あてはめを行うべきデータは，$x-y$ 平面上の矩形領域 $D = [a,b] \times [c,d]$ の格子点上で与えられ，

$$F_{i,j} = f(x_i, y_j) + \epsilon_{i,j}, \quad (i = 1,2,\cdots,N_x; j = 1,2,\cdots,N_y) \tag{7}$$

と表されるものとする．ここで，$f(x,y)$ はデータのもとにある未知の関数（信号）であり，$\epsilon_{i,j}$ は平均値 0，分散 σ^2 の正規分布をする互いに独立な誤差であると仮定する．また，σ^2 の大きさは未知である．さらに，式 (7) の N_x と N_y は，それぞれ x 軸方向と y 軸方向のデータ点の数である．

x 軸方向および y 軸方向の必要な節点を，それぞれ $\xi_i\,(i = 1-m_x, 2-m_x, \cdots, n_x+m_x)$ および $\zeta_j\,(j = 1-m_y, 2-m_y, \cdots, n_y+m_y)$ と書くことにする．ここで，n_x は区間 $[a,b]$ の内部に配置する節点 $\xi_i\,(i=1,2,\cdots,n_x)$ の数である．同様に，n_y は区間 $[c,d]$ の内部に配置する節点 $\zeta_j\,(j=1,2,\cdots,n_y)$ の数である．さらに，m_x および m_y は，それぞれ B–スプライン $N_{m_x,i}(x)$ および $N_{m_y,j}(y)$ の階数（次数 $+1$）である．1 次元データの場合と同様に，区間 $[a,b]$ の両端に m_x 重個の節点を置き，区間 $[c,d]$ の両端に m_y 重個の節点を置く．このとき，関数 $f(x,y)$ のモデル関数は

$$S(x,y) = \sum_{i=1}^{n_x+m_x} \sum_{j=1}^{n_y+m_y} c_{i,j} N_{m_x,i}(x) N_{m_y,j}(y) \tag{8}$$

と表すことができる．ここで，$c_{i,j}$ は B–スプライン係数である．

最小二乗法を用いて式 (8) を式 (7) で与えられるデータへあてはめるとき，残差の 2 乗和 Q_2 は

$$Q_2 = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} w_{i,j} \{S(x_i,y_j) - F_{i,j}\}^2 \tag{9}$$

となる．ここで，$w_{i,j}$ はデータの重みである．また，$N_x > n_x + m_x$ および $N_y > n_y + m_y$ とする．さらに，記号 Q の添字 2 はデータの次元を表している．B–スプライン係数 $c_{i,j}\,(i=1,2,\cdots,n_x+m_x;\ j=1,2,\cdots,n_y+m_y)$ は，式 (9) を最小化する条件から求めることができる．

式 (9) を最小化する問題は，節点 $\xi_i\,(i=1,2,\cdots,n_x)$ および $\zeta_j\,(j=1,2,\cdots,n_y)$ を非線形パラメータとする多峰性の最適化問題である．この問題は，データが格子点上にある特徴を利用すると，1 次元データのあてはめ問題の解法を，x 軸方向および y 軸方向に対して繰返し適用することによって解くことができる[18]．

3. 遺伝的アルゴリズムの適用

3.1 遺伝的アルゴリズムを用いる理由

スプライン関数へ GA を応用した研究はすでにあるが[19),20)]，データあてはめ問題の節点の決定へ応用したものはまだ見当らない．一般に，GA は計算量が多いので，従来の解法（傾斜法など）で容易に解ける問題には適用すべきではない．本論文で扱う問題に対して GA を用いる理由は，主として次の 3 つである．

(i) 目的関数は，**多峰性**の多変数関数であり，内部の節点が非線形パラメータとなっている．

(ii) 目的関数は非線形パラメータで微分することが容易でない．

(iii) 通常は，非線形パラメータ（内部の節点）の数の適切な値は未知であるため，その数も変数として最適化問題を解く必要がある．

これらのことから分かるように，解くべき問題は変数の数が未知で複数個の極値を持つ最適化問題である．このため，その大域的な最適解を求めることは，従来の方法ではきわめて困難である．

この困難さを克服するために，本論文では次の方法を提案する：

(a) 元の連続系最適化問題を離散系組合せ最適化問題へと変換する．

(b) その変換された問題を GA を用いて解く．

この方法は，準最適な節点の数と位置を自動的かつ同時に求めることができる特徴がある．

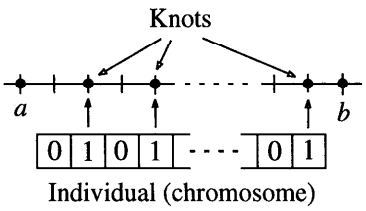


図 1 コード化
Fig. 1 Method of coding.

3.2 コード化

3.1 節の (a) で述べた変換を行うため，連続変数である節点を，次のようなコード化によって離散変数へ変換する．1 次元データの場合には，x 軸上の区間 $[a,b]$ を $L+1$ 個に分割して，その（内部の）分点を個体（染色体）上の遺伝子座に対応させる．ここで，分割は等間隔でなくてもよい．そして，ある遺伝子座の遺伝子が 1 ならその遺伝子座に対応する分点を節点とし，そうでなければ（0 であれば）節点としないことにする（**図 1** 参照）．

このようなコード化を行うと，遺伝子長（遺伝子列の長さ）は L となる．区間 $[a,b]$ の両端にはつねに節点を置くので，その点はコード化しなくてもよい．なお，L の値が大きいほど離散化のための誤差は少な

くなり，節点の位置をより細かく（正確に）決定できる．しかし，計算量はその分だけ多くなることに注意したい．

このことを，区間 $[a,b]$ を L 等分した場合についてやや詳しく述べる．この場合には，節点の位置の精度は $(b-a)/(L+1)$ となるので，遺伝子長 L に逆比例して高精度になる．逆に，交叉や突然変異で扱うべきデータの量は，遺伝子長に比例して多くなる．ところが，本論文で提案するアルゴリズムの計算量は，最小二乗法を用いてあてはめを計算する部分に集中しており，遺伝的操作（4.3 節参照）のための計算量は比較的少ない（6 章参照）．したがって，特に高精度を要求されるのでなければ，遺伝子長を多少長くしても全体的に見たときの計算量の増加は少ない．

2 次元格子点データの場合には，y 軸上についても同様なコード化を行う．

4. 遺伝的アルゴリズムの構成要素

4.1 初期集団

GA では，計算を始めるための初期値として個体（染色体）の初期集団を与える必要がある．1 次元データの場合には，各遺伝子座の遺伝子をランダムに 0 か 1 とした個体を K 個発生させたものを初期集団とする．ここで，K の値は個体数（染色体の数）である．GA では，この個体を解候補の初期集団として最適解を大域的に探索する．

ところで，一般にデータの元にある関数 $f(x)$ が複雑な形をしている場合には，節点を多く必要とする．したがって，1 の多い個体が有利となる．逆に，$f(x)$ が単純な形をしている場合には，節点は少なく，1 の少ない個体が有利となる．

したがって，計算の収束を早めるためには，データの関数形を見て区間 $[a,b]$ の内部に配置する節点の数 n の初期値（すなわち初期集団に含まれる各個体の 1 の数）を制御できる仕掛けを導入した方がよい．そこで，節点率 λ を設けた．その変域は $0<\lambda<0.5$ である．

ここで，λ が 0 よりも大きい理由は，内部節点の数 n は 1 以上であるためである．また，上限 0.5 は数値実験の結果から決めたものである．すなわち，4.2 節で述べる評価関数が有効に働くためには，内部節点の数 n がデータの数 N に比べて 0.5 未満であればよい．節点が増えすぎるとモデルのパラメータの数が増えすぎ，AIC が有効に働かず適切な解へ収束しない．数値実験では，節点率を以上のように設定したとき，節点数の増殖を回避することができた．

節点率 λ を変化させることによって，内部節点の数 n の初期値の平均を制御できる．たとえば，遺伝子長 $L=99$ のとき，$\lambda=0.3$ にすると，初期集団における各個体の節点数 n の平均は $0.3\times L(\fallingdotseq 30)$ となる．

2 次元格子点データの場合には，両座標軸方向について同様な方法で初期集団を作成する．

4.2 評価関数

評価関数としては，赤池の情報量規準 AIC[16] を用いる．これは，1 次元データのあてはめの場合には，

$$AIC_1 = N\log_e Q_1 + 2(2n+m) \qquad (10)$$

と表現できる[1]．ここで，N はデータの数，Q_1 は式 (6) で表される残差の 2 乗和である．また，AIC の添字 1 はデータの次元を表している．さらに，$(2n+m)$ はモデル関数（近似関数）に含まれるパラメータの数である．この中で，$n+m$ は B-スプライン係数の数，n は内部節点の数である．式 (10) の右辺の，第 1 項はデータに対する適合度，第 2 項はモデルの単純さを表している．

評価関数の値（AIC の値）を適応度と呼び，それを最小にするモデルが候補の中で最も良いモデルであると見なされる．すなわち AIC を用いると，データに対する適合度とモデルの単純さのバランスをとったモデルを自動的に選択できる特徴がある．

評価関数を以上のように決めると，6 章で述べるように節点が適切に配置される場合に対応する個体が生き残り，その結果として良いあてはめを求めることができる．また，AIC を用いることによって，従来の方法[2),6)]で必要とされる許容誤差とか平滑化パラメータは不必要となる．これらの適切な値を設定するためには，経験と試行錯誤を必要とする場合が多いので，AIC を用いる効果は大きい．

なお，GA の文献[11)~14)]では，“適応度が大きいほど最適値に近い” と表現されている場合が多いが，本論文では AIC をそのまま評価関数として用いているため，“適応度が小さいほど最適値に近い” ことになるので注意されたい．

2 次元格子点データの場合の評価関数は，1 次元データの場合のそれを簡単に拡張でき，

$$AIC_2 = N_xN_y\log_e Q_2 + 2\{(n_x+m_x)(n_y+m_y)+n_x+n_y\} \qquad (11)$$

となる．ここで，N_xN_y はデータの数であり，Q_2 は式 (9) で与えられ，$(n_x+m_x)(n_y+m_y)+n_x+n_y$ は式 (8) の中のパラメータの数である．

なお，あてはめの評価規準としては，AIC 以外に

もMDL，BICなどが提案されている[21)~23)]．本論文で提案する方法ではAICを用いたが，その代わりにMDL，BICなどを用いることも可能であろう．ただし，これらの中でどれが自分の望む結果を与えるかは，自分がどの立場に立っているかによって決まるものである[22)]．本論文では，AICを最小化するモデルが最も良いモデルであるという立場に立っている．MDLやBICなどを用いるとどうなるかは興味あるテーマであるが，それについては別の機会に報告したい．

4.3 遺伝的オペレータと制御パラメータ

遺伝的オペレータとしては，選択，交叉，および突然変異の3つがある．本論文では，選択にはトーナメント方式を，交叉には2点交叉を，突然変異には，遺伝子を一定の確率で対立遺伝子に置き換える方法を用いる．

また，GAを実行させるためには，いくつかの制御パラメータが必要である．その主なものは，個体数K，遺伝子長L，交叉の確率C，突然変異の確率Mである．

5. データあてはめのアルゴリズム

GAを用いた節点の決定方法を組み込んだ，スプライン関数による1次元データあてはめのアルゴリズムは，次のようになる．

ステップ1：式(1)で表される，あてはめを行うべきデータを入力する．

ステップ2：制御パラメータ（個体数K，遺伝子長L，交叉率C，突然変異率M，節点率λ）を設定する．

ステップ3：乱数を用いて初期集団を生成する．

ステップ4：各個体ごとに，それに対応する節点を用いてスプライン関数によるあてはめを行い，適応度を計算する．

ステップ5：最終世代まで計算したか？YESのとき計算を終了する．NOのとき次のステップ6へ行く．

ステップ6：各個体の適応度に基づいて個体の選択を行う．

ステップ7：選択された個体に対して交叉を行い，次世代の個体候補を生成する．

ステップ8：個体候補に突然変異を行い，次世代の個体集団を作成し，ステップ4へ戻る．

上記のアルゴリズムの中では，ステップ4に計算負荷が集中しているが，必要であればこの部分は並列計算が可能である．

なお，2次元格子点データの場合には，上記のアルゴリズムを各座標軸方向ごとに適用する．ただし，ステップ1の式(1)を式(7)に換え，ステップ4を「両座標軸方向の対応する個体に対する節点を用いてあてはめを行う」ように変更する．

6. 数値実験

6.1 実験結果

前章で述べたアルゴリズムの有効性を調べるため，多くの例題を用いて数値実験を行った．その中から，3つの例題についての結果を報告する．スプライン関数の次数は，最もよく使われている3次または双3次の場合について計算したが，本論文で提案する方法は次数には依存しないことに注意したい．また，最小二乗近似を計算するとき，データの重みwの値はすべて1とした．

以下に示す計算例は，個体数$K=50$，遺伝子長$L=99$，交叉率$C=0.9$，突然変異率$M=0.01$，節点率$\lambda=0.3$（節点数λの初期値の平均が約30）とした場合である．ただし，例3は2次元データであるため，これらの値は両座標軸方向とも同じにした．計算には，シリコングラフィックス製のOnyx（MIPS R4400MC × 4，150 MHz）を用いた．ただし，本例題では並列計算は行っていない．なお，「データあてはめの結果」を示す図のx軸上または$y=-10$の線上にある黒丸は，節点の位置を表している．

例1：ロジスティック曲線データの場合

あてはめるべきデータを次の式

$$F_j = 90/(1+e^{-100(x_j-0.4)}) + \epsilon_j, \quad (j=1,2,\cdots,N) \qquad (12)$$

で作成した．ここで，ϵ_jは期待値0，分散1の正規分布をする誤差である．x_jの値は，$0.0, 0.01, \cdots, 1.0$の101個とした．また，あてはめを行う区間は$[a,b]=[0,1]$にした．

図2は，世代に対する適応度と節点数の変化を示す計算結果である．灰色の線および黒い線は適応度を示している．灰色の線は，初期集団を変えて30回の試行を行い，その平均をとったものである．また黒い線は，最適な適応度を与えた試行結果であり，51世代目で収束している．さらに点線は，そのとき各世代で評価値が最小となる個体について，節点数の変化を示したものである．第1世代では節点数が28であるが，世代が進むにつれて減少していき，最終的には6個になっていることが分かる．ただし，両端の節点はそれぞれ1個として数えているので注意されたい．

図3は，データあてはめの結果である．図3(a)は

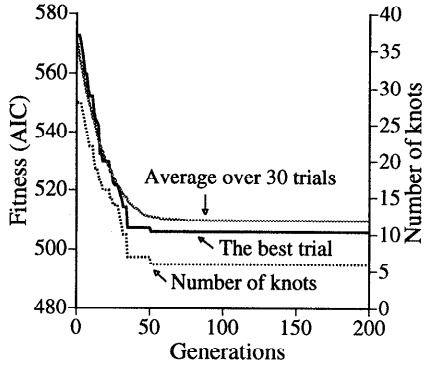


図2 例1の適応度と節点の数
Fig. 2 Fitness and number of knots for Example 1.

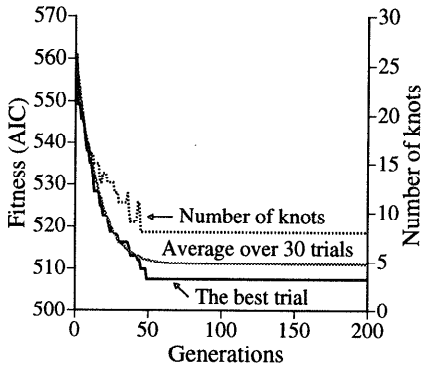
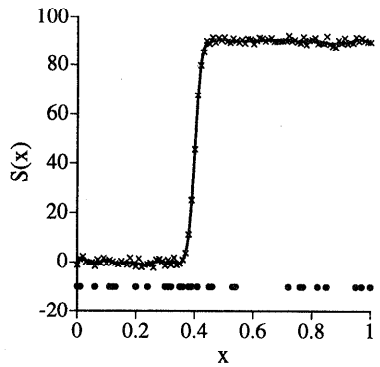
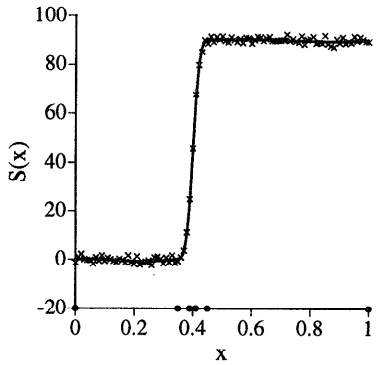


図4 例2の適応度と節点の数
Fig. 4 Fitness and number of knots for Example 2.



(a) 第1世代目の最も良いもの



(b) 収束後

図3 例1のデータあてはめの結果
Fig. 3 Result of data fitting for Example 1.

第1世代で最も良いものであるが，節点が多すぎるうえ，その位置も調整されていないため，曲線に多くの小さな振動が現れている．また図3(b)は収束したときであるが，関数 $S(x)$ はデータをよく近似している．このとき節点は，数が少なくデータの元にある関数の変化が大きいところに集中している．このことは，良い近似を得るための経験的な知識[2),4),9)]とよく一致している．

なお，本例題で100世代の計算を行ったとき，計算時間は約9.87秒であった．また，そのときの遺伝的操作に要した時間とその他の計算に要した時間の割合は6.52%対93.48%であった．

例2：複合有理式曲線データの場合

あてはめるべきデータを次の式

$$F_j = 1.0/\{0.01 + (x_j - 0.3)^2\} + 1.0/\{0.02 + (x_j - 0.6)^2\} + \epsilon_j, \quad (j = 1, 2, \cdots, N) \quad (13)$$

で作成した．ここで，ϵ_j，x_j の値，およびあてはめを行う区間は例1と同じである．

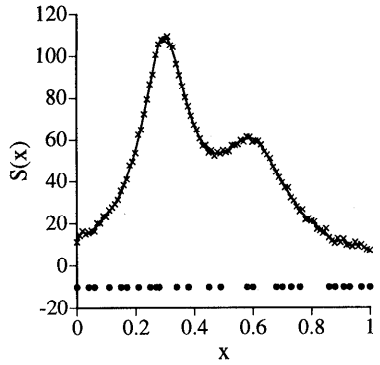
図4，図5 は計算結果である．図4の黒い線は，最適な適応度を与えた試行結果であり，49世代目で収束している．また図4の点線は，そのときの節点数の変化を示している．この例では，適応度の収束が節点数の収束よりも若干遅かった．その理由は，節点数の収束後も節点の位置の調整が若干行われたためである．さらに，図5はデータあてはめの結果である．図5(b)に示すように，収束後のあてはめの結果の節点はデータの元にある関数の変化が大きいところに集中しており，良い近似を得るための経験的な知識[2),4),9)]とよく一致している．

例3：2次元データの場合

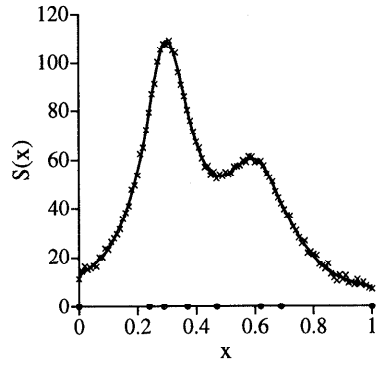
あてはめるべきデータを次の式

$$F_{i,j} = 1.0/\{0.02 + (x_i - 0.4)^2\} + 1.0/\{0.02 + (y_j - 0.2)^2\} + 90/(1 + e^{-100(x_i - 0.8)}) + 80/(1 + e^{-100(y_j - 0.6)}) + \epsilon_{i,j}, \quad (i = 1, 2, \cdots, N_x; j = 1, 2, \cdots, N_y) \quad (14)$$

で作成した．ここで，$\epsilon_{i,j}$ は期待値0，分散1の正規分布をする誤差である．x_i および y_j の値は，ともに



(a) 第 1 世代目の最も良いもの



(b) 収束後

図 5 例 2 のデータあてはめの結果
Fig. 5 Result of data fitting for Example 2.

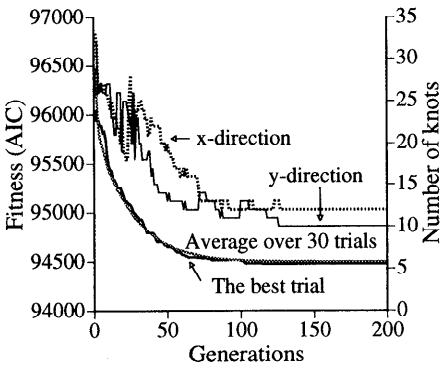
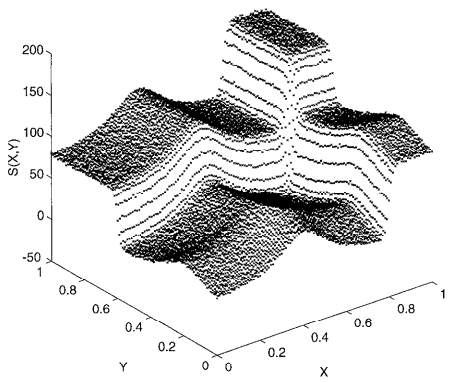
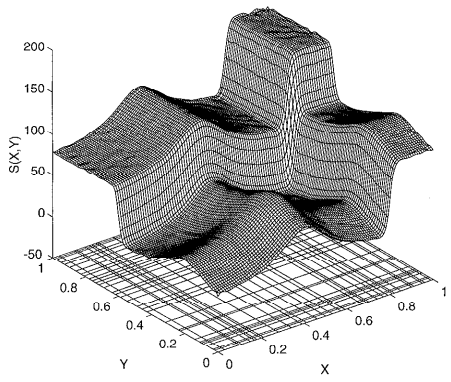


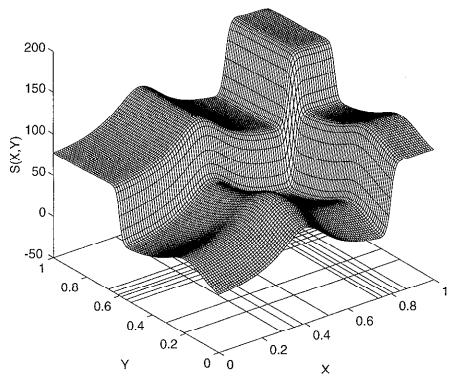
図 6 例 3 の適応度と節点の数
Fig. 6 Fitness and number of knots for Example 3.



(a) あてはめるべきデータ



(b) 第 1 世代目で最も良いもの



(c) 収束後

図 7 例 3 のデータあてはめの結果
Fig. 7 Result of data fitting for Example 3.

$0.0, 0.01, \cdots, 1.0$ の 101 個とした．また，あてはめを行う区間は $D = [0,1] \times [0,1]$ にした．

図 6，**図 7** は計算結果である．図 6 の太くて黒い実線は，最適な適応度を与える試行結果であり，127 世代目で収束している．また，図 6 の点線および細い実線は，そのときの節点数の変化を示している．さらに，図 7 はデータあてはめの結果であるが，図 7 (b) および図 7 (c) の領域 $D = [0,1] \times [0,1]$ の上に書かれている格子は，節点（節線）の位置を表している．図 7 (b) は，第 1 世代で最も良いものであるが，この場合には，節線が多すぎるうえ，その位置も調整されていないため，曲面が波打っている．また，図 7 (c) は収束後のあてはめの結果である．この図の節線は，データの元

にある関数の形が大きく変化している領域に集まっており，良い近似を得るための経験的な知識[2),4),9)]とよく一致している．

なお，本例題で100世代の計算を行ったときの計算時間は約3.28分であった．また，そのときの遺伝的操作に要した時間とその他の計算に要した時間の割合は0.76%対99.24%であった．例1に比べてその差が大きい理由は，多くの世代において本例題の方が節点の数が多いため，全体としてあてはめの計算に要する時間が多かったからである．

6.2 考　察

以上3つの例題について述べた．いずれの例題でも30回の試行中少なくとも数回は良い節点の数と位置へ収束した．しかし，適切な節点の数と位置へ収束しない場合も少なくなかった．その理由の1つは，今回GAとして単純遺伝的アルゴリズム（SGA）を用いたことであろう．SGAは局所解へ収束する確率がかなり高いことが知られている[24)]．また，初期集団の作り方にも改善の余地があるかもしれない．

この収束問題の改善方法については，次の2つが考えられる．

(1) 単純遺伝的アルゴリズム（SGA）を，たとえば適応的な遺伝的アルゴリズム（AGA）[24)]を用いて改良する．

(2) スプラインを用いたデータあてはめの経験的な知識を用いて適切な節点の数と位置にできるだけ近い初期集団を発生させる．すなわち，できるだけ性質の良い親を発生させる．

また，例1および例3で示したとおり，遺伝的操作に要した計算時間は，その他の部分の計算時間に比べて少なかった．この結果は，節点の位置の精度がデータ点 x_j の間隔程度でよければ，遺伝的操作は全体の計算時間に大きな影響を与えないことを示している．ただし，遺伝的操作に要する時間は，遺伝子長にほぼ比例するので，節点の位置を高精度に求めたい場合には，遺伝的操作に要する時間が全体の計算時間に大きな影響を与えるであろう．

7. あとがき

本論文では，スプライン関数を用いたデータあてはめの節点を，GAによって決定する方法を提案した．データあてはめ問題の節点の決定は，多変数で多峰性の非線形最適化問題であり，それをまともに解くことはきわめて困難である．しかし，この問題は実用上十分な精度で離散的な組合せ最適化問題へ簡単に変換できる．また，変換された組合せ最適化問題は，GAを用いて解くことができ，自動的に準最適な節点の数と位置を同時に決めることができる．

提案方法は，評価関数としてAICを用いているので，AICの意味で統計的に最適なモデルを自動的に選択できる特徴がある．すなわち，適切に配置された必要最小限の節点で構成されるスプライン関数を，データ自身の“判断”により選択できる．このため，従来の方法で必要とされることが多い許容誤差とか平滑化パラメータなどの設定は不必要である．ただし，GAは確率的な方法であるので，6章で述べたように，どのような初期集団を与えてもすべて大域的な最適解に収束するというわけではない．

今後の課題としては，計算量の軽減，適切な解への収束率の向上，制御パラメータの自動的な決定方法の開発などがある．これらの課題は，本論文で提案した方法に限らず，GA全般の課題である．この中で，制御パラメータの自動的な決定が可能となれば，ユーザはあてはめを行うべきデータのみをセットすればよくなる．したがって，完全に自動化されたデータあてはめのアルゴリズムが実現する可能性がある．また，他の評価関数を用いた場合との比較や，従来の方法との比較も今後の興味ある課題である．なお，並列計算への適合性についての研究結果は別の機会に報告する．

参考文献

1) 市田浩三，吉本富士市：スプライン関数とその応用，p.220，教育出版，東京 (1979).

2) Dierckx, P.: *Curve and Surface Fitting with Splines*, p.285, Clarendon Press-Oxford (1993).

3) Ichida, K., Yoshimoto, F. and Kiyono, T.: Curve fitting by a piecewise cubic polynomial, *Computing*, Vol.16, No.4, pp.329–338 (1976).

4) Cox, M.G.: A survey of numerical methods for data and function approximation, *The State of the Art in Numerical Analysis*, Jacobs, D.A.H. (Ed.), pp.627–668, Academic Press, New York (1977).

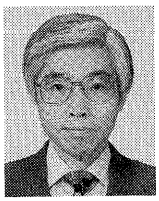
5) Jupp, D.L.B.: Approximation to data by splines with free knots, *SIAM J. Numer. Anal.*, Vol.15, No.2, pp.328–343 (1978).

6) Lyche, T. and Moerken, K: A data-reduction strategy for splines with applications to the approximation of functions and data, *IMA Journal of Numerical Analysis*, Vol.8, No.2, pp.185–208 (1988).

7) Anthony, H.M., Cox, M.G. and Harris, P.M.: The use of local polynomial approximations in a knot-placement strategy for least-squares spline fitting, *NPL Report*, DITC 148/89

(1989).
8) 馬渡鎮夫, 隆 雅久, 豊田吉顯:スプライン平滑化における節点の自動設定に関する一考察, 電子情報通信学会論文誌 (D-II), Vol.J72-D-II, No.11, pp.1816–1823 (1989).
9) 吉本富士市:ファジィ概念を用いたスプライン関数の節点の決定, 情報処理学会論文誌, Vol.35, No.9, pp.1682–1690 (1994).
10) Rice, J.R.: *Numerical Methods, Software, and Analysis*, Second Ed., p.720, Academic Press, San Diego (1993).
11) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, p.412, Addison-Wesley (1989).
12) 樋口哲也, 北野宏明:遺伝的アルゴリズムとその応用, 情報処理, Vol.34, No.7, pp.871–883 (1993).
13) 伊庭斉志:遺伝的アルゴリズムの基礎—GA の謎を解く, p.254, オーム社, 東京 (1994).
14) 坂和正敏, 田中雅博:遺伝的アルゴリズム, p.203, 朝倉書店, 東京 (1995).
15) 吉本富士市, 森山真光:スプライン関数を用いたデータあてはめ—遺伝的アルゴリズムによる節点の自動的な決定, 情報処理学会研究報告, 97-CG-88, pp.1–6 (1997).
16) Akaike, H.: A new look at the statistical model identification, *IEEE Trans. Automatic Control*, Vol.AC-19, No.6, pp.716–723 (1974).
17) de Boor, C.: *A Practical Guide to Splines*, p.392, Springer-Verlag, New York (1978).
18) Dierckx, P.: An algorithm for least-squares fitting of cubic spline surfaces to functions on a rectilinear mesh over a rectangle, *J. of Computational and Applied Mathematics*, Vol.3, No.2, pp.113–129 (1977).
19) Manela, M., Thornhill, N. and Campbell, J.A.: Fitting spline functions to noisy data using a genetic algorithm, *Proc. 5th Int. Conf. on Genetic Algorithms*, pp.549–556, Morgan Kaufmann (1993).
20) Markus, A., Renner, G. and Vancza, J.: Spline interpolation with genetic algorithms, *Proc. 1997 Int. Conf. on Shape Modeling and Applications*, pp.47–54, IEEE Computer Society Press (1997).
21) 小長谷明彦:確率的アプローチによる遺伝子情報処理, 人工知能学会誌, Vol.8, No.4, pp.427–438 (1993).
22) 松嶋敏泰:統計モデル選択の概要, オペレーションズ・リサーチ, Vol.41, No.7, pp.369–374 (1996).
23) 赤池弘次:AIC と MDL と BIC, オペレーションズ・リサーチ, Vol.41, No.7, pp.375–378 (1996).
24) Srinivas, M. and Patnaik, L.M.: Adaptive probabilities of crossover and mutation in genetic algorithms, *IEEE Trans. Systems, Man and Cybernetics*, Vol.24, No.4, pp.656–667 (1994).

(平成 10 年 1 月 8 日受付)
(平成 10 年 7 月 3 日採録)



吉本富士市 (正会員)

昭和 41 年岡山大学工学部電気工学科卒業. 明石工業高等専門学校, 和歌山大学教育学部を経て, 現職は和歌山大学システム工学部教授, システム情報学センター長. 工学博士. 形状モデリング, 遺伝的アルゴリズム, 計算工学, 画像処理等の研究に従事. 共著書「スプライン関数とその応用」(教育出版) 等. IEEE Computer Society, 電子情報通信学会, 日本応用数理学会, 日本計算工学会等各会員.

森山 真光 (正会員)

平成 3 年広島大学総合科学部総合科学科卒業. 平成 5 年同大学大学院工学研究科修士課程修了. 平成 8 年大阪大学大学院基礎工学研究科物理系専攻 (情報工学分野) 博士課程単位取得認定退学. 同年和歌山大学システム工学部助手, 現在に至る. 博士 (工学). コンピュータビジョン, 医用画像処理の研究に従事.