Vol. 47 No. 6 情報処理学会論文誌 June 2006

ε 制約遺伝的アルゴリズムによる制約付き最適化

高濱徹行[†] 阪井節子[††]

進化的アルゴリズムに基づいた制約付き最適化に関する研究が活発に行われている．しかし，従来の方法では探索の安定性が低い，制約領域内の局所解からの脱出が不十分である，目的関数の評価回数が多いという問題があった．本研究では，ε 制約法を遺伝的アルゴリズム（GA）に適用した εGA を提案する．εGA は，均等に親を選択し親と子の上位を次世代に残す選択，一様交叉，Gauss 突然変異，Cauchy 突然変異を採用することにより，安定した局所解に陥りにくい効率的な探索を行うことができる．εGA を 13 個の多様な制約付き非線形最適化問題に適用し，他の方法と比較することによりその有効性を示した．

Constrained Optimization by the ε Constrained Genetic Algorithm

TETSUYUKI TAKAHAMA[†] and SETSUKO SAKAI[††]

Researches on constrained optimization using evolutionary algorithms have been actively studied. However, these reseaches have problems that the stability and the efficiency of the search is low and the ability of escaping from local solutions is inadequate. In this study, we propose the εGA, which is defined by applying the ε constrained method to a genetic algorithm. The εGA adopts the selection where parents are chosen equally and next generation is formed by top individuals from parents and children, uniform crossover, Gaussian mutation and Cauchy mutation. The εGA realizes stable and efficient search that can escape local solutions. The advantage of the εGA is shown by applying the εGA to various type of 13 constrained problems and comparing the results to the results by other methods.

1. はじめに

進化的アルゴリズム（Evolutionary Algorithm, EA）は，生物進化の過程をモデル化した最適化アルゴリズムの総称であり，遺伝的アルゴリズム（Genetic Algorithm, GA）や進化的戦略（Evolutionary Strategy, ES）がよく知られている．EA は，最適化の対象である目的関数の値だけを利用して解を求めることができる直接探索法であり，たとえば微分可能性のような目的関数に対する制限がなく，アルゴリズムの実装が容易であることから，様々な最適化問題を解くために利用されるようになってきている．EA は基本的に制約のない最適化問題を解くためのアルゴリズムであるが，実世界の最適化問題の多くは与えられた制約の下で目的関数を最適化する制約付き最適化問題である．このため，EA に基づく制約付き最適化法もさかんに研究され，従来の数理的方法を超える結果が得られることも示されてきている[1),2)]．

目的関数と制約の取扱い方法に着目して，EA による制約付き最適化の研究を分類すると，以下のような 4 種類に大別できる．

(1) 目的関数のみを最適化する方法

制約を満足する 1 つ以上の探索点を初期点として準備し，制約を満足する探索点のみを考慮してゆくことにより，制約の最適化を省略する方法であり，death penalty 法とも呼ばれる．探索の過程で得られた点が制約を満足しない場合には，単純に無視されるか，制約を満足するように修正される．GA において，制約を満足した探索点を参照して制約を満足しない探索点を修正する方法[3)]やパーティクルスウォーム最適化（Particle Swarm Optimization, PSO）[4)]において，制約を満足しない探索点を無視し，既知の制約を満足する探索点に置換する方法[5)]も提案されている．これらの方法は制約領域が比較的広い場合には有効であるが，等式制約など制約の厳しい問題では，初期点を準備したり探索点を修正したりすることは非常に困難である．

(2) 目的関数と制約逸脱度（constraint violation）の荷重和を最適化する方法

† 広島市立大学
Hiroshima City University
†† 広島修道大学
Hiroshima Shudo University

複数の制約条件を組み合わせて制約逸脱度を定義し，目的関数と制約逸脱度の荷重和を求め，その荷重和の一目的最適化問題として解く方法である．制約逸脱度は目的関数に対するペナルティと考えられるため，この方法は一般にペナルティ関数法（penalty function method）と呼ばれる．ペナルティ関数法では，制約逸脱度の強さを調整するための荷重であるペナルティ係数（penalty coefficient）を適切に選択することが困難であるという問題点がある．ペナルティ係数が大きいと，制約を満足する解は得られるが，目的関数の最適化が不十分になり，質の高い解を得ることが困難になる．逆にペナルティ係数が小さいと，目的関数は最適化されるが，制約の最適化が不十分になり，実行可能解を得ることが困難になる．ペナルティ係数を動的に調整する方法[6]，探索の進行状況に応じてペナルティ係数を適応的に制御する方法[7),8]などが提案されている．しかし，これらの方法はペナルティ係数を自動的に調整するために時間がかかり，多くの計算量が必要となる．また，良好な結果を得るためには他のアルゴリズムパラメータを調整する必要があり，同じパラメータ設定で多くの問題に対応するのは困難である．

(3) 目的関数と制約逸脱度を分離して最適化する方法

この範疇における代表的な方法は，制約逸脱度を目的関数より優先する辞書式比較を利用する方法である．GA において，制約を満足しない探索点の適合度を，集団中の制約を満足する探索点における最悪の目的関数値と制約逸脱度との和として与える方法が提案されている[9]．ES において，単に制約逸脱度を優先するのではなく，ある確率で制約逸脱度を無視し目的関数のみで比較を行うという拡張された辞書式比較により最適化を行う stochastic ranking 法[10]や，制約逸脱度を優先するが，制約逸脱度が悪いが目的関数値が良い解も次世代に残す方法[11]が提案されている．より一般的な方法として，直接探索法全般に対して，制約条件を緩和することができる辞書式比較である α レベル比較を使用する α 制約法[12)~15]および ε 比較を使用する ε 制約法[16]が提案されている．α 制約法および ε 制約法は，直接探索法における比較演算子を α レベル比較および ε レベル比較に置換することにより，制約のない問題に対する最適化アルゴリズムを制約付き問題に対する最適化アルゴリズムに変換する方法，すなわちアルゴリズム変換法である．α および ε 制約法は，制約条件を緩和することにより，等式制約を含むような制約条件の厳しい問題に対しても適用することができる．

また，辞書式比較以外の方法として，最初に制約のみを最適化して実行可能解を探索し，次に目的関数を最適化するという 2 段階の最適化法[17]も提案されている．

この範疇の方法は，多様な問題に対して比較的良好な結果を得られることが示されている．

(4) 目的関数と各制約の多目的問題として解く方法

目的関数と一般に複数の制約関数を多目的最適化問題として解く方法である[18),19]．制約が複雑な問題に有効であると期待されるが，多目的最適化問題は一目的問題と比較すると非常に困難な問題であり，一般に多くの計算量を必要とするという問題点がある．

以上で述べた EA に基づく制約付き最適化法には一般に以下のような問題がある．

- 探索の安定性が低い
  同じ問題を対象にした場合でも，ランダムに発生した初期探索点集合の状態によって，良好な近似解を発見できる場合もあれば，かなり劣った近似解しか発見できない場合もあることが多い．したがって，平均的に安定して精度の高い近似解を発見できる方法が必要である．
- 制約領域内の局所解からの脱出が不十分
  EA に基づく最適化では，制約条件の最適化と目的関数の最適化のバランスをとることによって発見が困難である制約領域の境界上および境界近傍の解の探索に成功している．しかし，境界内に局所解を多く含むような制約付き最適化問題に対する対処がなされておらず，そのような問題における性能が不十分であるため，局所解から脱出する方法を取り入れる必要がある．
- 目的関数の評価回数が多くなりがち
  EA に基づく最適化法では，異なる初期探索点集合に対して最適化を行うため，ある程度安定した結果を得るためには，目的関数の評価回数が多くなりがちになる．より少ない評価回数で最適化が可能な効率的な方法が望まれる．

本研究では，ε 制約法を GA に適用した εGA を提案する．親の選択方法として，ルーレット選択やランキング選択ではなく，均等に親を選択し，親と子の中で上位の個体を次世代に残すという単純な方法を採用することにより，確率的選択による揺らぎをおさえるとともに，探索をより高速化する．これにより，探索の安定性が高まるとともに，目的関数の評価回数が減少することが期待できる．なお，この選択方法は ES で標準的に用いられている方法である．交叉として，実数値 GA でよく利用されている算術交叉ではなく，より単純な一様交叉を用い積極的な組替えを行う．突

然変異として，Gauss 突然変異に加え，より広範囲の変異を可能にする Cauchy 突然変異[20] を導入する．これらにより，局所解からの脱出が容易になるとともに，安定した探索が行われることが期待できる．なお，従来提案されている αGA[14] は，線形ランク選択，算術交叉，境界突然変異，一様突然変異，ガウス突然変異を採用した GA に α 制約法を適用したものであるが，境界突然変異のために制約の計算量が多大に必要であり，制約領域内の局所解からの脱出が非常に困難であるという問題があった．

εGA では，制約の最適化は ε 制約法により自然に行われるが，特に等式制約のような厳しい制約条件が与えられた場合には，ε レベルの制御を行うことにより，制約を緩和しながら最適化する必要がある．本研究では，等式制約を持つ最適化問題において，より安定した探索を行うために，単純で効果的な新しい ε レベルの制御方法を提案する．

以下，2 章で本論文の対象とする制約付き最適化問題を定義し，ε 制約法を説明する．3 章で εGA を定義し，4 章で他の方法と比較した εGA の性能を示し，5 章でアルゴリズムパラメータに対する εGA の安定性を示す．6 章はまとめである．

2. ε 制約法

ε 制約法が対象とする制約付き最適化問題と ε 制約法の定義および性質について説明する．

2.1 制約付き最適化問題

本論文では，次のような不等式制約，等式制約，上下限制約を持つ最適化問題 (P) を考える．

$$\begin{array}{lll}(P) & \text{minimize} & f(\boldsymbol{x}) \\ & \text{subject to} & g_j(\boldsymbol{x}) \le 0,\ j=1,\ldots,q \\ & & h_j(\boldsymbol{x}) = 0,\ j=q+1,\ldots,m \\ & & l_i \le x_i \le u_i,\ i=1,\ldots,n\end{array} \tag{1}$$

ここで，$\boldsymbol{x}=(x_1,\cdots,x_n)$ は n 次元決定変数ベクトル，$f(\boldsymbol{x})$ は目的関数，$g_j(\boldsymbol{x}) \le 0$ は q 個の不等式制約，$h_j(\boldsymbol{x})=0$ は $m-q$ 個の等式制約であり，f，g，h は線形あるいは非線形の実数値関数である．l_i，u_i はそれぞれ，n 個の決定変数 x_i の下限値，上限値である．さらに，以下ではすべての制約を満足する領域を実行可能領域 $\mathcal{F}$，上下限制約を満足する領域を探索空間 $\mathcal{S}$ と呼ぶことにする．

2.2 制約逸脱度

ε 制約法では，制約をどの程度逸脱しているかを表現するために，制約逸脱度 $\phi(\boldsymbol{x})$ を導入する．制約逸脱度 $\phi(\boldsymbol{x})$ は，以下を満足する関数である．

$$\begin{cases}\phi(\boldsymbol{x}) = 0 \ (\boldsymbol{x} \in \mathcal{F}) \\ \phi(\boldsymbol{x}) > 0 \ (\boldsymbol{x} \notin \mathcal{F})\end{cases} \tag{2}$$

制約逸脱度関数は，ペナルティ関数法におけるペナルティと同様に以下のような定義が可能である．

$$\phi(\boldsymbol{x}) = \max\{\max_j\{0, g_j(\boldsymbol{x})\}, \max_j |h_j(\boldsymbol{x})|\} \tag{3}$$

$$\phi(\boldsymbol{x}) = \sum_j \max\{0, g_j(\boldsymbol{x})\}^p + \sum_j |h_j(\boldsymbol{x})|^p \tag{4}$$

ただし，p は正数である．

2.3 ε レベル比較

関数値と制約逸脱度の組 (f,ϕ) の集合上において，制約逸脱度が ε 以下の場合は目的関数値の大小関係を優先し，それ以外の場合は制約逸脱度の大小関係を優先する比較である ε レベル比較を定義する．

点 $\boldsymbol{x}_1$，$\boldsymbol{x}_2$ における関数値を f_1，f_2，制約逸脱度を ϕ_1，ϕ_2 とすると，通常の大小関係である $<$，$\le$ に対応する関数値と制約逸脱度の組 (f_i,ϕ_i) 間の大小関係である ε レベル比較 $<_\varepsilon$ $(\varepsilon \in (0,\infty))$，$\le_\varepsilon$ $(\varepsilon \in [0,\infty))$ は以下のようになる．

$$(f_1,\phi_1) <_\varepsilon (f_2,\phi_2) \Leftrightarrow \begin{cases} f_1 < f_2, & \text{if } \phi_1,\phi_2 \le \varepsilon \\ f_1 < f_2, & \text{if } \phi_1 = \phi_2 \\ \phi_1 < \phi_2, & \text{otherwise}\end{cases} \tag{5}$$

$$(f_1,\phi_1) \le_\varepsilon (f_2,\phi_2) \Leftrightarrow \begin{cases} f_1 \le f_2, & \text{if } \phi_1,\phi_2 \le \varepsilon \\ f_1 \le f_2, & \text{if } \phi_1 = \phi_2 \\ \phi_1 < \phi_2, & \text{otherwise}\end{cases} \tag{6}$$

なお，$<_0$，$\le_0$ は制約逸脱度を優先する辞書式比較と一致し，$<_\infty$，$\le_\infty$ は目的関数値のみの比較と一致する．

2.4 ε 制約法の性質

ε 制約法は，制約付き最適化問題を直接探索法で解く際に，通常の比較の代わりに ε レベル比較を用いる方法である．通常の大小比較を ε レベル比較に置き換えた最適化問題 $(\mathrm{P}_{\le_\varepsilon})$，すなわち，$\varepsilon$ 制約法による最適化問題は以下のように定義できる．ただし，$\text{minimize}_{\le_\varepsilon}$ は $\le_\varepsilon$ の意味での最小化である．

$$(\mathrm{P}_{\le_\varepsilon}) \quad \text{minimize}_{\le_\varepsilon} \quad f(\boldsymbol{x}) \tag{7}$$

ここで，問題 (P) の制約条件を $\phi(\boldsymbol{x}) \le \varepsilon$ に緩和した問題 (P^ε) を以下のように定義する．なお，(P^0) は問題 (P) と等価である．

$$\begin{array}{lll}(\mathrm{P}^{\varepsilon}) & \text{minimize} & f(\boldsymbol{x}) \\ & \text{subject to} & \phi(\boldsymbol{x}) \le \varepsilon\end{array} \qquad (8)$$

問題 $(\mathrm{P}^{\varepsilon})$ と問題 $(\mathrm{P}_{\le\varepsilon})$，および問題 (P) に関して以下の定理が成り立つ．

定理 1 問題 (P^0) に最適解が存在するならば，問題 $(\mathrm{P}_{\le\varepsilon})$ の最適解は問題 $(\mathrm{P}^{\varepsilon})$ の最適解である．

定理 2 問題 (P) に最適解が存在するならば，問題 $(\mathrm{P}_{\le 0})$ の最適解は，問題 (P) の最適解である．

定理 3 $\{\varepsilon_n\}$ を強い意味で単調減少し 0 に収束する点列とする．$f(\boldsymbol{x})$，$\phi(\boldsymbol{x})$ を連続関数とし，問題 (P^0) の最適解 $\boldsymbol{x}^*$ の存在と任意の ε_n に対する問題 $(\mathrm{P}_{\le\varepsilon_n})$ の最適解 $\hat{\boldsymbol{x}}_n$ の存在を仮定する．このとき，点列 $\{\hat{\boldsymbol{x}}_n\}$ の任意の集積点は問題 (P^0) の最適解である．

定理 1，2 は，ε レベル比較を行うことにより，制約付き問題がそれと等価な制約のない問題に変換されることを示している．したがって，既存の制約のない最適化法に ε レベル比較を導入することにより，制約付きの問題を解くことが可能となる．定理 3 は，ペナルティ法においてペナルティ係数を ∞ まで増加させてゆくのと同様に，ε を 0 まで減少させながら最適化を行っても，最適解が得られることを示している．

α 制約法では，制約逸脱度の代わりに区間 $[0,1]$ の制約満足度を用い，制約満足度が 1 の場合に実行可能解とする．したがって，α 制約法と ε 制約法は理論的には等価である．しかし，計算機上においては，一般に 1 に近い数を表現する場合よりも 0 に近い値を表現する場合の方が表現精度が高いため，α レベルを制御する場合よりも ε レベルを制御する場合の方が制約に対する計算精度が高くなるという利点を持つ．

3. ε 制約遺伝的アルゴリズム εGA

ε 制約法は，目的関数値の大小関係のみに基づく最適化アルゴリズムと組み合わせることができる．本研究では，適合度の大小関係のみに依存する戦略である，均等な親選択および親と子の上位個体を選択する生存者選択を採用することにより，GA に ε 制約法を適用する．

3.1 アルゴリズム

εGA のアルゴリズムを以下に示す．

(1) 初期集団の生成：初期個体をランダムに N 個体生成する．すなわち，探索空間 $\mathcal{S}$ 内に各遺伝子 x_i を区間 $[l_i, u_i]$ の一様乱数で生成する．

(2) 終了判定：本論文では最大世代数 T に達したとき，実行を終了する．

(3) ε レベルの決定：通常は $\varepsilon=0$ として探索すればよい．しかし，等式制約のように実行可能領域が非常に狭い場合には，実行可能領域が狭すぎて有効な最適化が行えない．このため，ε レベルを制御することにより，初期には制約条件を緩和して目的関数の最適化を優先し，次第に制約条件を厳しくしてゆく必要がある（3.3 節参照）．

(4) 親選択（parent selection）：すべての個体を親として選択し，遺伝子プールにランダムに配置する．

(5) 交叉（crossover）：交叉確率 P_c で親を一様交叉し，子を生成する．一様交叉とは，2 個体の親から 2 個体の子を生成する際，ある遺伝子座の遺伝子を子がどちらの親から受け取るかを等確率で決定し，他の子は異なる親から遺伝子を受け取るという交叉である．なお，交叉しない場合には親がそのまま子になったと見なす．

(6) 突然変異（mutation）：広い領域を網羅的に探索するとともに，精度の高い探索を実現するため，変異幅が世代とともに減少する Gauss 突然変異と Cauchy 突然変異を採用する（3.2 節参照）．

(7) 生存者選択（survivor selection）：親と子の $2N$ 個体から ε 比較による上位 N 個体を選択し，次世代に残す．(2) へ戻る．

εGA のアルゴリズムを以下に C 言語風に記述する．

```
εGA()
{
  P(0)=探索空間 S 内にランダムに生成した N 個体;
  for(t=0; t<T; t++) {
    ε=ε(t);
    P'=P(t) のすべての個体をランダムに配置;
    for(each 個体対 x_p, x_q in P') {
      確率 P_c で x_p と x_q を一様交叉;
      for(each 遺伝子 x_i in x_p, x_q) {
        確率 P_G で x_i を Gauss 突然変異;
        確率 P_C で x_i を Cauchy 突然変異;
      }
    }
    P(t+1)=P(t)∪P' の ε レベル比較による
              上位 N 個体;
  }
}
```

ただし，$\varepsilon(t)$ は世代 t における ε レベル，P_c は交叉率，P_G は Gauss 突然変異率である．P_C は Cauchy 突然変異率であり，$1-P_G$ とする．

3.2 突然変異

本論文では，解の近傍を探索するために Gauss 突然変異，広域的な探索や局所解からの脱出を行うために Cauchy 突然変異を採用する．

Gauss 突然変異は，個体のすべての変数について，確率 P_G で以下のような突然変異を起こす．

$$x'_i = x_i + (u_i - l_i) \times \sigma_G(t) N(0,1) \tag{9}$$

ただし，$u_i - l_i$ は探索空間における変数 x_i の範囲の大きさ，$N(0,1)$ は平均 0，分散 1 の正規分布に基づく乱数，$\sigma_G(t)$ は世代 t における Gauss 突然変異のステップ幅である．

Cauchy 突然変異は，個体のすべての変数について，確率 $P_C = 1 - P_G$ で以下のような突然変異を起こす．

$$x'_i = x_i + (u_i - l_i) \times \sigma_C(t) C(0,1) \tag{10}$$

ただし，$C(0,1)$ は Cauchy 分布に基づく乱数，$\sigma_C(t)$ は世代 t における Cauchy 突然変異のステップ幅である．Cauchy 分布 $C(\mu, \sigma)$ の密度関数 $f_C(x)$ は以下のように定義され，正規分布よりも裾野が広い形状をしているため，より広範囲の探索が可能となる．

$$f_C(x) = \frac{1}{\pi} \frac{\sigma}{(x-\mu)^2 + \sigma^2}, -\infty < x < \infty \tag{11}$$

なお，x が $(-\pi/2, \pi/2)$ 上の一様分布に従うとき，$\tan x$ は Cauchy 分布 $C(0,1)$ となる．

ステップ幅は，初期には大域的な探索を行うため大きく設定し，次第に小さくすることにより精密な解を探索することが望ましい．本研究では，ステップ幅 $\sigma_G(t)$，$\sigma_C(t)$ を以下のようなベキ乗関数で設定した．

$$\sigma_{rnd}(0) = 0.5 \times \frac{1}{\sqrt{n}} \tag{12}$$

$$\sigma_{rnd}(t) = \sigma_{rnd}(0) \left(\sqrt[T]{\sigma_f} \right)^t \tag{13}$$

ただし，σ_{rnd} は σ_G および σ_C を意味する．σ_f は最終的なステップ幅を指定するパラメータである．なお，ES ではステップ幅の初期値を $(u_i - l_i)/\sqrt{n}$ にとるが，小さめがよいという研究[11] があるため本研究では 0.5 倍とした．

3.3 ε レベルの制御

本論文では，等式制約を含む最適化問題に対して，次のような制御方法を採用する．ε レベルを制御する際には，そのレベル以下の個体がつねにある程度含まれるように 0 に減少し，0 になった後もある程度の最適化を行うことが望ましい．そこで，初期値 $\varepsilon(0)$ を初期集団の中で制約逸脱度が小さい上位 20%番目の個体の制約逸脱度とし，最大世代数 T の 80%以降はつねに 0 となる，以下のような世代 t のベキ乗関数による制御を提案する．

$$\varepsilon(0) = \phi(\boldsymbol{x}_\theta) \tag{14}$$

$$\varepsilon(t) = \begin{cases} \varepsilon(0)(1 - \frac{t}{T_c})^{cp}, & 0 < t < T_c, \\ 0, & t \geq T_c \end{cases}$$

ただし，$\boldsymbol{x}_\theta$ は ϕ の良い個体の上位 20%番目の個体，$T_c = 0.8T$ である．ベキ乗の係数 cp は通常 5 とする．

4. εGA による制約付き最適化

本論文では，多くの研究で取り上げられてきた 13 個の制約付き非線形最適化問題を取り上げ，εGA による結果と他の方法による結果を比較する．

4.1 テスト問題と実験条件

13 個の制約付き非線形最適化問題 g01～g13 は，文献 10)，11) をはじめとするいくつかの研究で取り上げられている．g02，g03，g08，g12 は最大化問題であり，その他は最小化問題である．g03，g05，g11，g13 は等式制約を含む問題である．特に g02 は制約領域内に多数の局所解を持つ多峰性の問題であり，従来の方法では安定した結果が得られていない．

表 1 に各問題の概略を示す[11),21)]．各問題について，決定変数の数（n），目的関数の形式，線形不等式制約（LI），非線形不等式制約（NI），線形等式制約（LE），非線形等式制約（NE）の数，制約式の値が 0 となる有効制約の数（active），および探索空間における実行可能領域の割合 $\rho = \frac{|\mathcal{F}|}{|\mathcal{S}|}$ を示した．ただし，ρ は探索空間中に 10,000,000 点をランダムに生成することによって本研究で求めたものである．

このように，これらの問題は，線形式，多項式，指数関数式などの多様な目的関数および制約式を含む問題となっている．また，制約領域の形状や制約の厳しさについても多様な問題となっている．たとえば，g12 は半径 0.25 の 9^3 個の円形制約領域を持つ，すなわち分離された制約領域を持つ問題である．制約の厳しさ

表 1 テスト問題の概略
Table 1 Summary of test problems.

prob.	n	Form of f	LI	NI	LE	NE	active	ρ(%)
g01	13	quadratic	9	0	0	0	6	0.00022
g02	20	nonlinear	1	1	0	0	1	99.99639
g03	10	polynomial	0	0	0	1	1	0.00000
g04	5	quadratic	0	6	0	0	2	26.95954
g05	4	cubic	2	0	0	3	3	0.00000
g06	2	cubic	0	2	0	0	2	0.00655
g07	10	quadratic	3	5	0	0	6	0.00009
g08	2	nonlinear	0	2	0	0	0	0.86109
g09	7	polynomial	0	4	0	0	2	0.52759
g10	8	linear	3	3	0	0	6	0.00063
g11	2	quadratic	0	0	0	1	1	0.00000
g12	3	quadratic	0	9^3	0	0	0	4.76560
g13	5	nonlinear	0	0	1	2	3	0.00000

表 2 標準設定における実験結果
Table 2 Result of independent 30 runs under standard settings.

prob.	optimal	best	median	average	worst	stddev	violation
g01	−15.000	−14.999996	−14.999987	−14.999987	−14.999980	3.127e-06	0
g02	0.803619	0.803617	0.803610	0.798846	0.786157	0.00569	0
g03	1.000	0.999983	0.999940	0.999932	0.999808	3.713e-05	1.453e-10
g04	−30,665.539	−30,665.538660	−30,665.538610	−30,665.538608	−30,665.538540	3.078e-05	0
g05	5,126.498	5,126.502474	5,126.829000	5,127.702549	5,136.358674	1.934	4.117e-05
g06	−6,961.814	−6,961.813120	−6,961.807625	−6,961.806695	−6,961.798004	0.003321	0
g07	24.306	24.310091	24.325065	24.335327	24.394265	0.0232	0
g08	0.095825	0.095825	0.095825	0.095825	0.095825	0	0
g09	680.63	680.630274	680.631639	680.631915	680.635284	0.001422	0
g10	7,049.25	7,221.224236	7,313.839923	7,329.004713	7,464.261373	68.17	0
g11	0.75	0.750000	0.750000	0.750001	0.750005	1.06e-06	1.75e-10
g12	1.000	1.000000	1.000000	1.000000	1.000000	0	0
g13	0.053950	0.053951	0.053955	0.053960	0.054003	1.263e-05	8.485e-07

については，等式制約を含む問題が最も厳しい制約を持つが，g01，g06，g07，g10 もかなり厳しい制約を持つ．逆に，g02，g04，g12 はかなり緩やかな制約を持っている．このような多様な制約付き問題に対して安定した最適化が可能な方法は，他の問題に対しても有効であることが期待できる．

εGA における標準的な設定を説明する．制約逸脱度は，式 (4) において $p=1$ とする単純和を用いる．GA に関するパラメータは，個体数 $N=40$，交叉率 $P_c=0.8$，Gauss 突然変異率 $P_G=0.75$，Cauchy 突然変異率 $P_C=1-P_G=0.25$，$\sigma_f=1\times10^{-6}$ とした．ε 制約法に関するパラメータは，等式制約を含まない問題については $\varepsilon(t)=0$ に固定し，等式制約を含む問題については式 (14) により ε レベルを制御し，ベキ乗の係数 $cp=5$ とする．実験では，g12 を除く問題について最大世代数 $T=4{,}999$ まで，問題 g12 は最大世代数 $T=499$ まで 30 回の試行を行った．このため目的関数の最大評価回数は，$N*(T+1)=200{,}000$ 回（g12 は 20,000 回）となる．

4.2 実験結果

標準設定における実験結果を**表 2** に示す．表において，optimal は各問題における最適値を示し，best, median, average, worst, stddev はそれぞれ εGA による各試行で得られた最良解の 30 回の試行における最良値，中央値，平均値，標準偏差を示している．violation は最良解における式 (3) で定義される制約逸脱の最大値の 30 回の試行における平均値である．

g01，g03，g04，g06，g08，g09，g11，g12，g13 については，最良値，中央値，平均値，最悪値とも非常に最適解に近い値を探索しており，安定した性能を示している．制約領域内に多数の局所解を持つ g02 については，最良値および中央値が最適解に非常に近い値となっており，局所解からの脱出が有効に働いていると判断できる．g05，g07 についても最良値および中央値は最適解と非常に近い値となっている．g10 については良い結果が得られなかったのは，g10 ではすべての制約が有効制約となる狭い領域に最適解が存在するが，εGA では局所解からの脱出を行う効果が強いため，その領域にうまく収束できなかったと考えられる．また，等式制約を含まない問題においてはすべて実行可能解を発見しており，等式制約を持つ g03，g05，g11，g13 においても，制約の逸脱は 10^{-10}～10^{-5} 程度と非常に小さく，かなり厳密な実行可能解が得られている．

表 3 標準設定における評価回数
Table 3 Number of evaluations in independent 30 runs under standard settings.

prob.	eval	const	CPU(s)
g01	43,046.4	197,721.2	2.10
g02	116,727.0	197,874.9	3.64
g03	76,209.3	184,865.8	1.72
g04	55,244.1	194,074.6	1.02
g05	8,623.1	196,980.3	0.98
g06	8,353.8	193,980.9	0.55
g07	24,715.9	197,893.2	1.70
g08	155,274.3	191,833.1	0.71
g09	59,328.9	195,503.2	1.51
g10	10,225.8	199,253.3	1.46
g11	48,793.6	186,265.6	0.59
g12	15,206.8	19,650.9	0.49
g13	31,727.2	194,596.6	1.05

表 3 に評価回数および実行時間を示す．eval，const は最良解を発見したときの目的関数および制約逸脱度の評価回数の平均値である．CPU(s) は Mobile Pentium III 1.3 GHz を搭載したノート型パーソナルコンピュータにおける実行時間の平均値であり，参考のために示した．

ε 制約法では，目的関数の評価と制約条件の評価を分離して行っており，制約逸脱度の比較のみで大小関

係が定まる場合には目的関数の評価を省略することができる．このため，目的関数の評価回数は，かなり少なくなっている．また，εGA では探索の終盤に非常に小さなステップ幅の突然変異を行うため，最終世代の近くにおいても解が更新されており，制約条件の評価回数は最大評価回数 200,000 回（g12 は 20,000 回）に近い値となっている．

4.3 他の方法との比較

13 種類の問題について，十分な統計データが示されており，その他の方法[22),23)] と比較して良好な結果を残している研究として，Runarsson らの Stochastic Ranking（SR）法[10)] と Mezura-Montes らの Simple Multimembered Evolution Strategy（SMES）法[11)] がある．SR，SMES および ASCHEA[23)] は，基本的な最適化アルゴリズムとして進化的戦略（ES）を採用しており，従来の GA を採用した方法[22)] よりも優れた結果を残している．ES は，決定変数に加えて突然変異のステップ幅も個体にコーディングし，ステップ幅も同時に最適化することにより精度の高い探索を行うが，最適化する項目が増えるため，計算量が多くなるとともに，最適化が不安定になる場合がある．これに対して εGA は GA を基本とするアルゴリズムであり，ステップ幅を世代とともに低減させている．これにより，ES よりも計算量が減少し，安定した最適化が可能となると考えられる．

SR および SMES は，基本的には目的関数よりも制約逸脱度を優先するという ε 制約法と同様の方法を用いているが，ある確率で目的関数を優先することを提案している．SR では，ある確率で目的関数を優先するという確率的比較を使用し，バブルソート的に個体をランク付けするという方法，SMES では，ある確率で目的関数が最良の個体を選択するという方法をとっている．すなわち，SR および SMES では，確率的に目的関数を優先することにより，制約領域の近傍でより良好な解を探索することを期待していると考えられる．これに対して，εGA では，ε レベル比較を導入することにより，等式制約のように制約の厳しい問題では ε レベルを緩和し制約領域を広げることにより，制約領域の近傍を探索するという方法をとっている．さらに，Cauchy 突然変異を導入することにより，局所解を多く含む多峰性の問題に対する探索能力を向上させている．

εGA を SR および SMES と比較するために実験を行い，**表 4**，**表 5**，**表 6**，**表 7**，**表 8** に各方法で得られた最良値，中央値，平均値，最悪値，標準偏差を示した．

表 4 最良値の比較
Table 4 Comparison of best values.

prob.	optimal	εGA	SR	SMES
g01	−15.000	**−15.000**	**−15.000**	**−15.000**
g02	0.803619	**0.803617**	0.803515	0.803601
g03	1.000	**1.000**	**1.000**	**1.000**
g04	−30,665.539	**−30,665.539**	**−30,665.539**	**−30,665.539**
g05	5,126.498	5,126.502	**5,126.497**	5,126.599
g06	−6,961.814	−6,961.813	**−6,961.814**	**−6,961.814**
g07	24.306	24.310	**24.307**	24.327
g08	0.095825	**0.095825**	**0.095825**	**0.095825**
g09	680.63	**680.630**	**680.630**	680.632
g10	7,049.25	7,221.224	7,054.316	**7,051.903**
g11	0.75	**0.750**	**0.750**	**0.75**
g12	1.000	**1.000000**	**1.000000**	**1.000**
g13	0.053950	**0.053951**	0.053957	0.053986

表 5 中央値の比較
Table 5 Comparison of median values.

prob.	optimal	εGA	SR	SMES
g01	−15.000	**−15.000**	**−15.000**	**−15.000**
g02	0.803619	**0.803610**	0.785800	0.792549
g03	1.000	**1.000**	**1.000**	**1.000**
g04	−30,665.539	**−30,665.539**	**−30,665.539**	**−30,665.539**
g05	5,126.498	**5,126.829**	5,127.372	5,160.198
g06	−6,961.814	−6,961.808	**−6,961.814**	**−6,961.814**
g07	24.306	**24.325**	24.357	24.426
g08	0.095825	**0.095825**	**0.095825**	**0.095825**
g09	680.630	**680.632**	680.641	680.642
g10	7,049.25	7,313.840	7,372.613	**7,253.603**
g11	0.75	**0.75**	**0.75**	**0.75**
g12	1.000	**1.000000**	**1.000000**	**1.000**
g13	0.053950	**0.053955**	0.057006	0.061873

表 6 平均値の比較
Table 6 Comparison of average values.

prob.	optimal	εGA	SR	SMES
g01	−15.000	**−15.000**	**−15.000**	**−15.000**
g02	0.803619	**0.798846**	0.781975	0.785238
g03	1.000	**1.000**	**1.000**	**1.000**
g04	−30,665.539	**−30,665.539**	**−30,665.539**	**−30,665.539**
g05	5,126.498	**5,127.703**	5,128.881	5,174.492
g06	−6,961.814	**−6,961.807**	−6,875.940	−6,961.284
g07	24.306	**24.335**	24.374	24.475
g08	0.095825	**0.095825**	**0.095825**	**0.095825**
g09	680.63	**680.632**	680.656	680.643
g10	7,049.25	7,329.005	7,559.192	**7,253.047**
g11	0.75	**0.750**	**0.750**	**0.750**
g12	1.000	**1.000000**	**1.000000**	**1.000**
g13	0.053950	**0.053960**	0.067543	0.166385

まず，各方法の探索能力について比較する．最良値については，最も良い値を示した問題数は，εGA が 9，SR が 10，SMES が 8 であり，あまり大きな差はない．中央値については εGA が 11，SR が 7，SMES が 8，平均値については εGA が 12，SR が 6，SMES が 7，最悪値については εGA が 13，SR が 6，SMES が 6，標準偏差については εGA が 11，SR が 2，SMES が 4 であり，いずれについても εGA がかなり優れている．また，εGA が最適値を発見できなかった g10 についても，最良値は劣っているが，中央値，平均値については SR より優れており，最悪値，標準偏差については SR と SMES よりも優れている．特に SR は g06 において，SMES は g05 においてかなり不安定な結果

表 7 最悪値の比較
Table 7 Comparison of worst values.

prob.	optimal	εGA	SR	SMES
g01	−15.000	**−15.000**	**−15.000**	**−15.000**
g02	0.803619	**0.786157**	0.726288	0.751322
g03	1.000	**1.000**	**1.000**	**1.000**
g04	−30,665.539	**−30,665.539**	**−30,665.539**	**−30,665.539**
g05	5,126.498	**5,136.359**	5,142.472	5,304.167
g06	−6,961.814	**−6,961.798**	−6,350.262	−6,952.482
g07	24.306	**24.394**	24.642	24.843
g08	0.095825	**0.095825**	**0.095825**	**0.095825**
g09	680.63	**680.635**	680.763	680.719
g10	7,049.25	**7,464.261**	8,835.655	7,638.366
g11	0.75	**0.750**	**0.750**	**0.750**
g12	1.000	**1.000000**	**1.000000**	**1.000**
g13	0.053950	**0.054003**	0.216915	0.468294

表 8 標準偏差の比較
Table 8 Comparison of standard deviations.

prob.	εGA	SR	SMES
g01	3.13e-06	**0**	**0**
g02	**5.69e-03**	2.0e-02	1.67e-02
g03	**3.71e-05**	1.9e-04	2.09e-04
g04	3.08e-05	2.0e-05	**0**
g05	**1.93**	3.5	50.06
g06	**3.32e-03**	1.6e+02	1.85
g07	**2.32e-02**	6.6e-02	1.32e-01
g08	**0**	2.6e-17	**0**
g09	**1.42e-03**	3.4e-02	1.55e-02
g10	**68.17**	5.3e+02	136.02
g11	**1.06e-06**	8.0e-05	1.52e-04
g12	**0**	**0**	**0**
g13	**1.26e-05**	3.1e-02	1.77e-01

となっているが，εGA は安定した結果を得ている．また，多峰性の問題である g02 においても，εGA は他の方法に比べて非常に良好な結果を得ており，Cauchy 突然変異による局所解からの脱出の効果が現れている．したがって，全体的には εGA が最も安定した結果が得られる方法であることは明らかである．

次に，目的関数の評価回数について比較する．εGA は最大 200,000 回の評価回数であるが，実質的には 8,000 回〜150,000 回程度の評価回数となっている．これに対して，SR は 350,000 回の評価，SMES は 240,000 回の評価を行っている．したがって，εGA は少ない評価回数で安定した結果が得られる方法である．

最後に，等式制約問題に対する探索能力について比較する．εGA では ε レベルを制御し，最終的には ε レベルを 0 に設定することにより，できる限り実行可能解に近い解を探索しており，制約からの逸脱が 10^{-10}〜10^{-5} という非常に高精度の解を得ている．これに対して，SR では等式制約を含む問題を直接解くのではなく，等式制約を緩和して $|h_j(\boldsymbol{x})| \leq 10^{-4}$ の不等式制約に変換して問題を解いているため，εGA よりも大きな 10^{-4} 程度の制約逸脱度となっていると考えられる．SMES では，εGA と同様の制約の緩和を行っているが，g03，g05，g11 では最終的に 4×10^{-4} まで，g13 では 3×10^{-5} までの緩和を行っており，問題ごとに緩和方法を調整しなければならず，最終的な緩和の程度も εGA よりかなり大きいという問題がある．等式制約問題における目的関数値を比較すると，g13 の最良値，g05，g13 の中央値，平均値および最悪値は εGA が最も優れており，標準偏差についてはすべての問題で εGA が最も優れている．g05 の最良値のみがわずかに SR の結果より劣っているが，εGA の解の制約逸脱度は約 4×10^{-5} であり，SR の解よりも制約逸脱度が小さい解となっている．すなわち，εGA はより単純かつ効果的な方法で ε レベルを制御することにより，非常に制約逸脱度が小さく，目的関数値も良好な解を安定して得ている．したがって，εGA は最も等式制約問題に対する探索能力が高い方法である．

5. εGA に関する検討

εGA が他の方法と比較して安定した効率的なアルゴリズムであることを示した．ここでは，εGA におけるパラメータを標準設定から変化させることにより，パラメータ設定が εGA に及ぼす影響について検討する．

表 9 に標準設定から交叉率のみを $P_c = 0$，0.2，0.5，0.8，1 と変化させた場合，**表 10** に Gauss 突然変異率のみを $P_G = 0$，0.25，0.5，0.75，1 と変化させた場合，**表 11** に最終ステップ幅のみを $\sigma_f = 10^{-4}$，10^{-5}，10^{-6}，10^{-7}，10^{-8} と変化させた場合の各試行で得られた最良値の 30 回の試行における平均値を示す．

交叉率については，g02 と g10 以外の問題では，交叉率を変化させても結果に大きな違いはなく，つねに最適値に近い解が得られている．したがって，εGA は交叉率の変化に対して安定している．g02 については $P_c = 0.5$ 以上，g10 については $P_c = 0.5$ 以下の場合に良好な結果が得られている．したがって，本研究では局所解からの積極的な脱出を強調するため $P_c = 0.8$ を標準としたが，$P_c = 0.5$ も有力な設定であると考えられる．

突然変異率については，g02 と g05 以外の問題では，突然変異率を変化させても結果に大きな違いはなく，つねに良好な結果が得られている．したがって，εGA は突然変異率の変化に対しても安定している．g02 については，$P_G = 0.75$ 以下（$P_C = 0.25$ 以上）の場合に良好な結果が得られており，Cauchy 突然変異が局所解からの脱出に有効に働いていることが分かる．g05 については，$P_G = 0.5$ 以上（$P_C = 0.5$ 以下）の場合に良好な結果が得られている．したがって，全体的には小さな Cauchy 突然変異率を設定することが有

表 9 交叉率 P_c の変化による平均値の変化
Table 9 Average Results of 30 independent runs varying crossover rate P_c from standard settings.

prob.	$P_c = 0$	$P_c = 0.2$	$P_c = 0.5$	$P_c = 0.8$	$P_c = 1$
g01	−14.933307	−14.999981	−14.999984	**−14.999987**	−14.999986
g02	0.772402	0.784425	0.792270	**0.798846**	0.797715
g03	0.999967	**0.999968**	0.999955	0.999932	0.999900
g04	**−30,665.538611**	−30,665.538601	**−30,665.538611**	−30,665.538608	−30,665.538608
g05	**5,126.532890**	5,126.646844	5,126.872891	5,127.702549	5,128.461099
g06	**−6,961.807068**	−6,961.805464	−6,961.805703	−6,961.806695	−6,961.806508
g07	24.331700	24.328290	**24.324660**	24.335327	24.328523
g08	**0.095825**	**0.095825**	**0.095825**	**0.095825**	**0.095825**
g09	680.633165	680.633253	680.632335	**680.631915**	680.632570
g10	**7,161.444291**	7,207.526095	7,253.289514	7,329.004713	7,354.693071
g11	**0.750000**	**0.750000**	0.750001	0.750001	0.750001
g12	**1.000000**	**1.000000**	**1.000000**	**1.000000**	**1.000000**
g13	**0.053953**	0.053954	0.053956	0.053960	0.053976

表 10 突然変異率 P_G の変化による平均値の変化
Table 10 Average Results of 30 independent runs varying mutation rate P_G from standard settings.

prob.	$P_G = 0$	$P_G = 0.25$	$P_G = 0.5$	$P_G = 0.75$	$P_G = 1$
g01	−14.999982	−14.999983	−14.999986	**−14.999987**	−14.508879
g02	0.797919	0.796021	0.796428	**0.798846**	0.784705
g03	**0.999952**	0.999948	0.999943	0.999932	0.999933
g04	−30,665.538594	−30,665.538603	−30,665.538606	**−30,665.538608**	−30,665.538604
g05	5,144.390980	5,133.030069	5,128.453920	5,127.702549	**5,126.990121**
g06	−6,961.801848	−6,961.805483	−6,961.805758	**−6,961.806695**	−6,961.806544
g07	24.349245	24.348889	24.333226	24.335327	**24.326504**
g08	**0.095825**	**0.095825**	**0.095825**	**0.095825**	**0.095825**
g09	680.632531	680.632789	680.632708	680.631915	**680.631818**
g10	7,351.353514	7,354.974442	7,345.019838	7,329.004713	**7,294.224518**
g11	0.750001	0.750001	0.750001	0.750001	**0.750000**
g12	**1.000000**	**1.000000**	**1.000000**	**1.000000**	**1.000000**
g13	0.053968	0.053968	0.053963	**0.053960**	0.053966

表 11 最終ステップ幅 σ_f の変化による平均値の変化
Table 11 Average Results of 30 independent runs varying final step size σ_f from standard settings.

prob.	$\sigma_f = 10^{-4}$	$\sigma_f = 10^{-5}$	$\sigma_f = 10^{-6}$	$\sigma_f = 10^{-7}$	$\sigma_f = 10^{-8}$
g01	−14.998842	−14.999872	−14.999987	−14.999999	**−15.000000**
g02	**0.799888**	0.798037	0.798846	0.794767	0.792829
g03	0.999875	**0.999960**	0.999932	0.999853	0.999650
g04	−30,665.533365	−30,665.538036	−30,665.538608	−30,665.538664	**−30,665.538670**
g05	5,145.586802	**5,127.339046**	5,127.702549	5,127.375611	5,127.448421
g06	−6,961.182384	−6,961.751551	−6,961.806695	−6,961.812940	**−6,961.813764**
g07	24.334077	**24.329334**	24.335327	24.334473	24.333728
g08	**0.095825**	**0.095825**	**0.095825**	**0.095825**	**0.095825**
g09	680.633605	**680.631586**	680.631915	680.632798	680.633577
g10	**7,280.608990**	7,312.619099	7,329.004713	7,356.787844	7,341.971058
g11	0.758341	**0.750001**	**0.750001**	0.750003	0.750015
g12	**1.000000**	**1.000000**	**1.000000**	**1.000000**	**1.000000**
g13	0.054071	0.053969	**0.053960**	0.053979	0.053995

効であると考えられる．

最終ステップ幅については，g02，g05，g11 以外の問題では，ステップ幅を変化させても結果に大きな違いはなく，良好な結果が得られている．したがって，εGA はステップ幅の変化に対して安定している．g02 については $\sigma_f = 10^{-6}$ 以上の場合，g05 と g11 については $\sigma_f = 10^{-5}$ 以下の場合に良好な結果が得られている．したがって，$\sigma_f = 10^{-6}$～10^{-5} 程度に設定

することが有効であると考えられる．

以上のように εGA はパラメータの変化に非常に強く，パラメータをそれほど調整しなくてもつねに安定した良好な結果が得られる方法であるといえる．

6. む す び

局所解からの脱出を積極的に行う仕組みを導入した遺伝的アルゴリズムに ε 制約法を適用した εGA を提案した．様々なタイプの 13 種類の制約付き最適化問題に εGA を適用し，ほとんどの問題で最適値に近い解が得られ，すべての問題で安定した近似解が得られることを計算機実験により確認し，εGA が精度の高い安定したアルゴリズムであることを示した．さらに，進化的アルゴリズムを利用した制約付き最適化方法の中で有効性がよく知られている SR 法および SMES 法と比較することにより，εGA が特に安定性，探索の効率，実行可能解の探索能力についてこれらの方法よりもさらに優れた方法であることを示した．

εGA はパラメータ変化に対して安定であるが，問題によっては交叉率を適切に選択することでさらに良好な結果が得られる場合がある．たとえば，g10 において $P_c = 0.0$ を選択すれば，最良値 7,051.260 という最適値に近い解を得ることができる．したがって，交叉率を適応的に変化させる方法について検討する価値があると考えられる．今後は，εGA をより大規模な問題に適用するために，εGA をさらに改良する予定である．このため，Particle Swarm Optimization など他の最適化方法とのハイブリッドにより，より多くの問題に対応できる安定したアルゴリズムを提案してゆきたいと考えている．

謝辞　本研究の一部は，日本学術振興会科学研究費補助金基盤研究（C）（課題番号 16500083，17510139）による補助のもとで行われた．

参 考 文 献

1) Michalewicz, Z.: *Genetic algorithm + data structures = evolution programs, 3rd ed.*, Springer-Verlag, Berlin (1996).

2) Coello, C.A.C.: Theoretical and Numerical Constraint-handling Techniques Used with Evolutionary Algorithms: A Survey of the State of the Art, *Computer Methods in Applied Mechanics and Engineering*, Vol.191, pp.1245–1287 (2002).

3) Michalewicz, Z. and Schoenauer, M.: Evolutionary Algorithms for Constrained Parameter Optimization Problems, *Evolutionary Computation*, Vol.4, No.1, pp.1–32 (1996).

4) Kennedy, J. and Eberhart, R.C.: *Swarm Intelligence*, Morgan Kaufmann, San Francisco (2001).

5) Parsopoulos, K.E. and Vrahatis, M.N.: Particle Swarm Optimization Method for Constrained Optimization Problems, *Intelligent Technologies — Theory and Application: New Trends in Intelligent Technologies*, Sincak, P., Vascak, J., et al. (Eds.), Frontiers in Artificial Intelligence and Applications, Vol.76, pp.214–220, IOS Press (2002).

6) Joines, J. and Houck, C.: On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GAs, *Proc. 1st IEEE Conference on Evolutionary Computation*, Fogel, D. (Ed.), Orlando, Florida, pp.579–584, IEEE Press (1994).

7) Coit, D., Smith, A.E. and Tate, D.M.: Adaptive Penalty Methods for Genetic Optimization of Constrained Combinatorial Problems, *INFORMS Journal on Computing*, Vol.8, No.2, pp.173–182 (1996).

8) Hamida, S.B. and Schoenauer, M.: An Adaptive Algorithm for constrained optimization problems, *Parallel Problem Solving from Nature — PPSN VI*, Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J.J. and Schwefel, H.-P. (Eds.), pp.529–538, Springer, Berlin (2000).

9) Deb, K.: An efficient constraint handling method for genetic algorithms, *Computer Methods in Applied Mechanics and Engineering*, Vol.186, No.2/4, pp.311–338 (2000).

10) Runarsson, T.P. and Yao, X.: Stochastic ranking for constrained evolutionary optimization, *IEEE Trans. Evolutionary Computation*, Vol.4, No.3, pp.284–294 (2000).

11) Mezura-Montes, E. and Coello, C.A.C.: A Simple Multimembered Evolution Strategy to Solve Constrained Optimization Problems, *IEEE Trans. Evolutionary Computation*, Vol.9, No.1, pp.1–17 (2005).

12) 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ー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).

13) 高濱徹行，阪井節子：α 制約 Simplex 法によるファジー制御ルールの学習，電子情報通信学会論文誌，Vol.J83-D-I, No.7, pp.770–779 (2000).

14) 高濱徹行，阪井節子：α 制約遺伝的アルゴリズム αGA による制約付き最適化，電子情報通信学会論文誌，Vol.J86-D-I, No.4, pp.198–207 (2003).

15) Takahama, T. and Sakai, S.: Constrained Optimization by the α Constrained Particle

Swarm Optimizer, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol.9, No.3, pp.282–289 (2005).

16) Takahama, T. and Sakai, S.: Constrained Optimization by ε Constrained Particle Swarm Optimizer with ε-level Control, *Proc. 4th IEEE International Workshop on Soft Computing as Transdisciplinary Science and Technology (WSTST '05)*, Muroran, Japan, pp.1019–1029 (2005).

17) Venkatraman, S. and Yen, G.G.: A Generic Framework for Constrained Optimization Using Genetic Algorithms, *IEEE Trans. Evolutionary Computation*, Vol.9, No.4, pp.424–435 (2005).

18) Camponogara, E. and Talukdar, S.N.: A genetic algorithm for constrained and multiobjective optimization, *3rd Nordic Workshop on Genetic Algorithms and Their Applications (3NWGA)*, Alander, J.T. (Ed.), Vaasa, Finland, University of Vaasa, pp.49–62 (1997).

19) Surry, P.D. and Radcliffe, N.J.: The COMOGA Method: Constrained optimisation by multiobjective genetic algorithms, *Control and Cybernetics*, Vol.26, No.3, pp.391–412 (1997).

20) Yao, X., Liu, Y. and Liu, G.: Evolutionary Programming Made Faster, *IEEE Trans. Evolutionary Computation*, Vol.3, No.2, pp.82–102 (1999).

21) Farmani, R. and Wright, J.A.: Self-adaptive fitness formulation for constrained optimization, *IEEE Trans. Evolutionary Computation*, Vol.7, No.5, pp.445–455 (2003).

22) Koziel, S. and Michalewicz, Z.: Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization, *Evolutionary Computation*, Vol.7, No.1, pp.19–44 (1999).

23) Hamida, S.B. and Schoenauer, M.: ASCHEA: New Results Using Adaptive Segregational Constraint Handling, *Proc. 2002 Congress on Evolutionary Computation*, pp.884–889, IEEE Press (2002).

(平成 17 年 10 月 6 日受付)
(平成 18 年 4 月 4 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 年同大学講師．1998 年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．2005 年より同大学教授．Natual Computing，遺伝的アルゴリズム，非線形最適化，機械学習，ファジィ推論，CAI，自然言語処理等に関する研究に従事．IEEE，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，意志決定，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．IEEE，日本 OR 学会，日本知能情報ファジィ学会，日本生産管理学会各会員．工学博士．