

Adaptive range particle swarm optimization to find multiple optima

メタデータ	言語: jpn 出版者: 公開日: 2017-10-03 キーワード (Ja): キーワード (En): 作成者: メールアドレス: 所属:
URL	https://doi.org/10.24517/00008093

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 International License.



領域適応型Particle Swarm Optimizationによる複数の最適解の探索

北山哲士^{*1} 宮川智栄^{*2}
山崎光悦^{*1} 荒川雅生^{*3}

Adaptive Range Particle Swarm Optimization to Find Multiple Optima

Satoshi Kitayama, Chie Miyakawa, Masao Arakawa, Koetsu Yamazaki
Faculty of Engineering, Kanazawa University,
Kakuma-machi, Kanazawa, 920-1192, Japan

The main objective called as the population-based optimization techniques is to find a global minimum. However, it is sometimes preferable to find multiple optima in the engineering design. The Adaptive Range Particle Swarm Optimization (ARPSO) to find multiple optima is proposed in this paper. The active search domain range of the ARPSO is only one, however, many active search domain ranges to find multiple optima are generated in the proposed approach. At first, the many active search domain ranges are generated by the paired particles, and the procedure of the ARPSO is applied. Then, the paired particles find local or global minimum separately. Many active search domain ranges are sometimes integrated through the search process. The detail procedure to integrate the active search domain ranges is also explained in this paper. It is possible to find multiple optima with high accuracy by the proposed approach. The validity of proposed method is examined through numerical examples.

Key Words : Global Optimization, Adaptive Range Particle Swarm Optimization, Optimum Design, Systems Engineering, Engineering Optimization

1. 緒言

進化的計算手法の一つであるParticle Swarm Optimization(PSO)は、探索点 (Particle)が持つ最良の情報 (p-best) と、探索点から形成されるグループ (Swarm)の最適値 (g-best)から、過去の探索履歴を考慮して連続変数の多峰性関数の大域的最適解もしくは次善の局所的最適解を求める多点同時探索型の手法である⁽¹⁾。また、筆者らは、精度の高い最適解を得るための方法の一つとして、各設計変数の平均・標準偏差を利用して、探索状況に応じて探索領域が適宜変更する領域適応型PSO(ARPSO)⁽²⁾を提案し、数値計算を通じてその有効性を検討した。

多くの最適化手法の目的は、精度の高い大域的最適解を如何にして少ない計算コストで見出すかという点に重点が置かれており、探索過程における複数の局所的最適解の発見に関しては、あまり重点が置かれていないと思われる。しかし、最適化手法を活用して最適設計を行う場合、複数の最適解を得ること

は、次のような利点を含むものと考えられる。

(1) 設計者は多くの設計案を持つことができ、その中から、最終案を決定することもできる。

(2) 設計変数や設計パラメータの変動に対して、ロバストな解、すなわちロバスト最適解を求めることは有用であると考えられ、複数の最適解の中にロバスト最適解が含まれる可能性がある。

(3) 多目的最適化問題において、対話型手法を用いてパレート最適解を求める場合、複数の最適解における目的関数値が希求水準値の候補と成りうる。

複数の最適解を求める方法としては、免疫アルゴリズム⁽³⁾ (IA)による方法⁽⁴⁾、遺伝的アルゴリズム (GA)の適合度を調節する方法⁽⁵⁾、筆者らの方法⁽⁶⁾、シェアリングの導入による方法^{(7),(8)}などが挙げられよう。またPSOでは、文献等の方法^{(9),(10)}がある。シェアリングの導入による複数の最適解探索は、一見すると非常に簡単と思われるが、パラメータの調整に関して、一定の指針を構築することは困難である。また筆者らの方法は、関数の感度を利用するため、関数の感度を用いないPSOに感度計算を含ませることは望ましくない。一方、PSOによる複数の最適解探索法に注目すると、文献(7)の方法は、基本的にはトンネリング・アルゴリズム (TA)の変形であり、さらにパラメータの設定

* 原稿受付 平成??年??月??日

*¹正員, 金沢大学理工学域 (〒920-1192 金沢市角間町)。

*²学生員, 金沢大学大学院 (〒920-1192 金沢市角間町)。

*³正員, 香川大学工学部 (〒761-0396 高松市林町2217-20)

に関する指針が与えられてない。また文献(8)の方法は、探索点がペアを組むことにより、問題に内在する複数の最適解を求める方法であるが、一定の探索回数でペアの解除を行うため、ランダムサーチに近い方法となっており、また新たなパラメータを導入している点、速度の更新式が2章で示す一般的な式とは異なるなど、単一目的の下でのPSOを遵守したものとは言い難い。しかし、探索点がペアを組み、探索を行うという考え自体は、PSOの基本的な性質⁽¹¹⁾を積極的に活用したものであるとも言え、大いに注目すべき点である。

以上をまとめると、文献(1)等で行われている一般的なPSOを遵守しつつ、複数の最適解を求めるアルゴリズム開発に関して、注意事項をまとめると次のようになると思われる。

- (1) パラメータを増加させないこと。仮にパラメータを増加させるのであれば、どのような効果があるのかを示し、さらに設定指針を明確に説明できること。
- (2) 感度(勾配)計算を導入しないこと。
- (3) PSOとGAは根本的に考え方が異なるため、PSOの中にGAのような考え(ルーレット選択や突然変異等)を取り入れるような方法は避けること。
- (4) パラメータ設定に関しては、可能な限り単一目的の下でのPSOのパラメータ設定を用いること。⁽¹²⁾

特に上記(4)に関しては、PSOが離散力学系⁽¹³⁾であることから、その安定性解析を行うことにより、パラメータの設定範囲やその効果が数値実験的に説明されているため、いたずらにパラメータ設定を変更すべきではないとの立場を取るものである。

そこで、本論文では、文献(8)で提案された探索点がペアを組むという考え方と、筆者らが提案したARPSOを併せることにより、問題に内在する複数の局所的最適解と大域的最適解を同時に求めるアルゴリズムを開発し、数値計算を通じて、その有効性を検討する。以降、2章において、PSOと本論文で着目したPSOの基本的性質について述べる。3章で、本論文で提案する方法について記述し、4章にそのアルゴリズムを示す。5章の数値計算例では、探索領域の可視化のため、2変数多峰性関数問題を扱い、本論文で提案する方法を説明し、結言を述べる。

2. Particle Swarm Optimization

PSOでは各探索点が「位置」と「速度」を持ち、集団で探索を行い、各探索点の位置と速度を更新しながら、最適解を探索する。

2.1 位置と速度の更新

k 回目の探索において、

探索点 d の位置 \mathbf{x}_d^k と速度 \mathbf{v}_d^k を用いて、 $k+1$ 回目の位置 \mathbf{x}_d^{k+1} と速度 \mathbf{v}_d^{k+1} は、次の式を用いて更新される。

$$\mathbf{x}_d^{k+1} = \mathbf{x}_d^k + \mathbf{v}_d^{k+1} \quad (1)$$

$$\mathbf{v}_d^{k+1} = w\mathbf{v}_d^k + c_1r_1(\mathbf{p}_d^k - \mathbf{x}_d^k) + c_2r_2(\mathbf{p}_g^k - \mathbf{x}_d^k) \quad (2)$$

式(2)において、 r_1 と r_2 は $[0, 1]$ の乱数である。また c_1 と c_2 はパラメータであり、力学系の安定性解析より

$$c_1 + c_2 \leq 4 \quad (3)$$

となるように、 c_1 と c_2 は決められている。一般的には $c_1 = c_2 = 2$ が用いられている。 w は慣性項と呼ばれるパラメータであり、次式のように線形的に減少することにより、探索点の多様性と集中化を達成している。⁽¹³⁾

$$w = w_{\max} - (w_{\max} - w_{\min})/k_{\max} \times k \quad (4)$$

式(4)において $w_{\max} = 0.9$ 、 $w_{\min} = 0.4$ が一般的には用いられており、 k_{\max} は最大探索回数である。式(2)中の \mathbf{p}_d^k は、探索点 d が k 回目までの探索において、今までで訪れた最良の解(p-best)を表す。一方、 \mathbf{p}_g^k は k 回目の探索における群れ全体の中での最良の解(g-best)を表す。式(1)、(2)を用いて探索点を更新するモデルは通常、g-bestモデルと呼ばれる。また式(2)中の \mathbf{p}_g^k を、 k 回目までの探索で目的関数値を最良にした探索点 \mathbf{p}_g で置き換えたモデル、すなわち \mathbf{p}_d^k の中から目的関数値を最もよくする点をg-bestとするモデルは、最良値保存型モデルと呼ばれる。PSOの基本アルゴリズムは文献(2)等を参照されたい。

2.2 PSOの基礎的性質

PSOでは最低2つの探索点があれば、探索可能である。式(1)、(2)を変形すると次式が得られる。

$$\mathbf{x}_d^{k+1} = \mathbf{x}_d^k + w\mathbf{v}_d^k + \alpha(\mathbf{q} - \mathbf{x}_d^k) \quad (5)$$

$$\alpha = c_1r_1 + c_2r_2 \quad (6)$$

$$\mathbf{q} = \frac{c_1r_1\mathbf{p}_d^k + c_2r_2\mathbf{p}_g^k}{c_1r_1 + c_2r_2} \quad (7)$$

式(5)はPSOが感度を用いた最適化手法との類似構造をもっていることを示しており、探索方向ベクトルは $\mathbf{q} - \mathbf{x}_d^k$ で与えられることを示している。また α は確率的ステップ幅を表していると考えられる。また \mathbf{q} は \mathbf{p}_d^k と \mathbf{p}_g^k を内分する点を表しており、最良値保存モデルでは、必ず

$$f(\mathbf{p}_g) \leq f(\mathbf{p}_d)$$

という関係が成立する。ここで、図1(a)と図1(b)に示すように、二つの探索点が凸空間と非凸空間に存在する場合、すなわち探索点がペアを組んだ場合を考える。探索点がペアを組めば、どちらか一方が必ずg-bestとなり、その結果、図1(a)の場合は目的関数を改善する方向に探索点は移動し、図1(b)の場合は目的関数を改悪する方向への移動を許容することで、局所

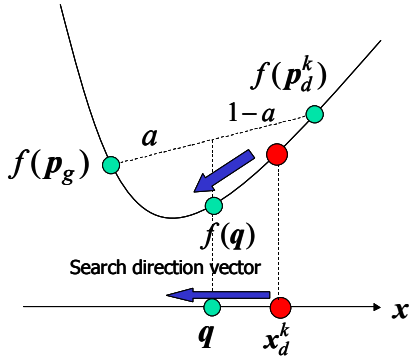


Fig.1(a) Two search points locate in the convex space

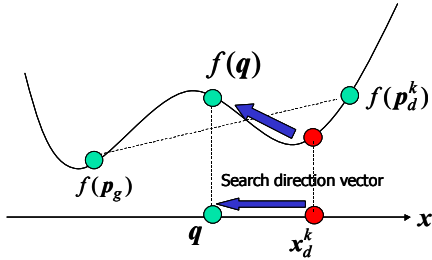


Fig.1(b) Two search points locate in the non-convex space

ある．そのため，過去の探索において目的関数を最良にする探索点の設計変数の情報も失われてしまう．そこで，過去の探索において目的関数を最良にする探索点は必ず有効な探索領域に入るように以下の操作を行う．なお以下の記述において x_i^{best} とは， p_g の i 番目の成分を表す．例えば， x_i^{best} が探索領域の右側に位置するとき，すなわち

$$\mu_i + \sqrt{-2\sigma_i^{R^2} \log a} < x_i^{best} \quad (9)$$

のときは，

$$\sigma_{i,new}^R = \sqrt{\frac{(x_i^{best} - \mu_i)^2}{2 \log a}} \quad (10)$$

として，右側の標準偏差を変更して，新しい有効な探索領域を式 (8) により求める．一方， x_i^{best} が探索領域の左側に位置するときは， $\sigma_{i,new}^L$ を同様に求め，新しい有効な探索領域を決定する．このような操作は，側面制約条件の対処にも用いられる．

2.3.3 探索回数に応じた探索領域の設定

式 (8) 中の a を以下のように更新することにより，有効な探索領域が探索回数に応じて狭くなる．

$$a = a_{\min} + (a_{\max} - a_{\min}) / k_{\max} \times k \quad (11)$$

そのため，精度の高い最適解を得ることが可能となる． a_{\max} と a_{\min} はパラメータであるが，その設定方法については，文献(2)を参照されたい．

3. 提案する方法

3.1 問題設定 本論文では，側面制約条件の下で単一の目的関数を最小化する問題を対象とする．

$$\text{Find } \mathbf{x} = (x_1, x_2, \dots, x_{ndv})^T \quad (12)$$

Such as

$$f(\mathbf{x}) \rightarrow \min \quad (13)$$

Subject to

$$x_i^L \leq x_i \leq x_i^U \quad i=1,2,\dots,ndv \quad (14)$$

上式において， \mathbf{x} は設計変数ベクトル， ndv は設計変数の数を表す． $f(\mathbf{x})$ は連続変数から成る最小化すべき目的関数であり， x_i^L と x_i^U は i 番目の設計変数に直接課せられる側面制約条件の下限値と上限値である．また最適解探索に必要な探索点の総数を $agent$ と表記する．

3.2 有効な探索領域の設定

初期探索 ($k=1$) では，探索点の位置および速度はランダムに決定される．そこで式 (1)，(2) により探索点の位置および速度を更新し，

$$k = k + 1 \quad (15)$$

として，探索回数 k を増加する．次に $k=2$ において，

的最適解からの脱出が可能となる．上記の点が，GA等の多点同時探索型最適化手法では陽に説明できない点であり，PSOの基礎的性質である．

2.3 領域適応型PSO (ARPSO) 領域適応型PSOは各設計変数の平均と標準偏差を用いて，有効な探索領域を決め，精度の高い最適解を得ることを目的に開発された方法である．以下，その概略についてまとめる．

2.3.1 有効な探索領域の決定 ARPSOでは，図2に示す正規分布を基調とし，この正規分布縦軸の値 a と i 番目の設計変数 x_i の平均 μ_i と標準偏差 σ_i を用いて，有効な探索領域を以下のように決める．

$$\mu_i - \sqrt{-2\sigma_i^{L^2} \log a} \leq x_i \leq \mu_i + \sqrt{-2\sigma_i^{R^2} \log a} \quad (8)$$

式 (8) 中の σ_i^L と σ_i^R は左右別々に与えられる i 番目の設計変数の標準偏差を表す．

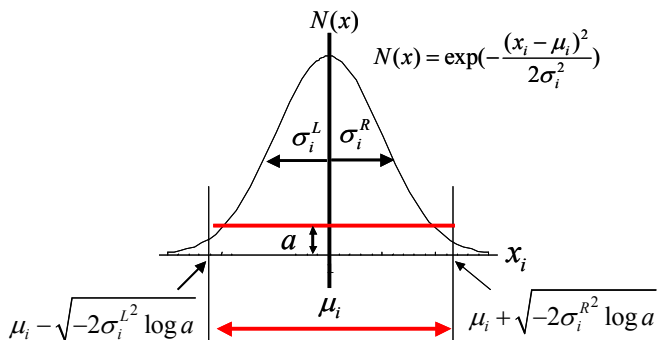


Fig.2 The active search domain of i -th design variable

2.3.2 最良値の保存 ARPSOでは有効な探索領域が直前の探索における探索点の情報を利用して決定され，有効な探索領域が探索回数毎に可変する可能性が

設計変数空間において、最近隣にある探索点同士がペアを組む。そして、ペアを組んだ二つの探索点の各設計変数の平均値と標準偏差を求め、ペアを組んだ探索点同士の有効な探索領域を式(8)を用いて決定する(図3)。

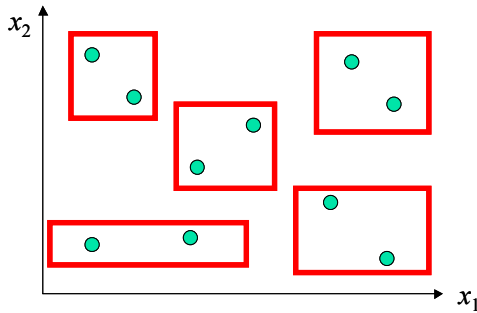


Fig.3 The active search domain ranges by the paired particles

図3中の●は探索点を表しており、四角で囲まれた領域が、ペアを組んだ探索点の有効な探索領域を示している。ペアを組むため、探索点数は偶数でなければならないが、これは2.2節で述べた通り、PSOの基礎的性質を活用するための制約である。 $k=2$ で生成される有効な探索領域 A_i の数は $agent/2$ 個となる。最近隣の点とペアを組むことにより、式(8)中の標準偏差は小さくなることが予想され、探索が進めば有効な探索領域 A_i は小さくなるものと考えられる。そのため、複数の最適解を求めることが期待できる。

3.3 各探索領域内の最良値の保存 本論文では、最適解の探索に対してARPSOを用いるため、探索領域は可変である。有効な探索領域 A_i ($i=1,2,\dots,agent/2$)において、目的関数値を最良にする探索点を $p_{g,i}$ ($i=1,2,\dots,agent/2$)と表記すれば、各探索領域 A_i 内に存在すると考えられる最適解探索の精度の向上を狙うために、2.3.2項に従い、 $p_{g,i}$ を必ず各探索領域 A_i に入れる。

3.4 有効な探索領域の結合 有効な探索領域 A_i 内に含まれる $p_{g,i}$ が図4(a)に示すように、共通な領域(図4(a)中の斜線部)に存在する場合を考える。

図4(a)において、●は探索点、▲は各探索領域 A_i 内の $p_{g,i}$ を表している。ここでは簡単のため、 A_i が二つ

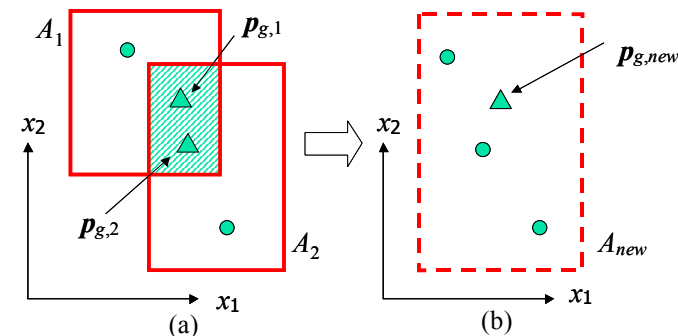


Fig.4 The newly active search domain range by combining A_1 and A_2

(A_1 および A_2)あり、各 A_i 内の $p_{g,i}$ をそれぞれ $p_{g,1}$, $p_{g,2}$ と表記する。また

$$f(p_{g,1}) < f(p_{g,2}) \quad (15)$$

であったとする。

図4に示すように、 $p_{g,1}$ と $p_{g,2}$ が A_1 と A_2 の共通な領域(図4(a)の斜線部)に存在する場合、 A_1 と A_2 を結合して、図4(b)に示す新たに有効な探索領域 A_{new} を作成する。ここで A_{new} を作成する場合、単に A_1 と A_2 を総和した領域を新たな探索領域とするのではなく、必ず A_1 と A_2 に存在する探索点の各設計変数の平均値と標準偏差を計算して、式(8)により A_{new} を決定する。 A_{new} の目的関数値を最良にする探索点に関しては、式(15)の関係から $p_{g,1}$ が A_{new} の最良値となる。また、図4の例において A_{new} に含まれる探索点数は4となる。すなわち、新たに生成される探索領域 A_{new} の探索点数は、結合する二つの探索領域に含まれる探索点数の和となる。また2.2節と3.2節で記述している通り、有効な探索領域に含まれる探索点数は常に偶数となる。

ここで有効な探索領域の結合について、探索初期段階で有効な探索領域の結合が起きた場合と、探索終盤で有効な探索領域の結合が生じた場合について、それぞれ考えてみる。

探索初期段階で有効な探索領域の結合が起きた場合、新たに生成される有効な探索領域は、比較的大きく考えられる。これは、新たに生成される有効な探索領域の設定を式(8)を用いて決めており、新たに生成される有効な探索領域内の探索点間の標準偏差が比較的大きいものと考えられ、結果的には大域的最適解の探索能力を高めるものと考えられる。一方、探索終盤で有効な探索領域が結合する場合は、有効な探索領域内の探索点間の標準偏差は比較的小さいものと考えられ、新たに生成される有効な探索領域は比較的小さくなるものと考えられ、結果的には局所的最適解の探索能力が向上するものと思われる。すなわち、有効な探索領域の結合を導入することにより、大域的最適解および局所的最適解を同時に探索する能力が向上するものと期待できる。

4. アルゴリズム

本論文で提案する複数に最適解を求めるためのARPSOを以下に記述する。

(STEP1) 探索点数 $agent$ 、最大探索回数 k_{max} を決定し、探索点の位置および速度を、側面制約条件内にランダムに決める。探索回数 $k=1$ とする。式(1), (2)を用いて探索点の位置および速度を更新する。 $k=k+1$

とする。

(STEP2)設計変数空間において、探索点間の距離を計算．最近隣に位置する探索点同士がペアを組む．

(STEP3)ペアを組んだ探索点に対し、各設計変数の平均と標準偏差を計算し、式(8)によりペア毎の有効な探索領域 A_I ($I=1,2,\dots,agent/2$) を決定．

(STEP4)各設計変数の標準偏差をチェックする．もし標準偏差の最小値を下回っている場合は、標準偏差の最小値とする．

$$\sigma_i^L < \sigma_{i,\min} \Rightarrow \sigma_i^L = \sigma_{i,\min} \quad (16)$$

$$\sigma_i^R < \sigma_{i,\min} \Rightarrow \sigma_i^R = \sigma_{i,\min} \quad (17)$$

なお、標準偏差の最小値は各設計変数の側面制約条件(2)を利用して決められる．

(STEP5)有効な探索領域 A_I 内におけるp-best $p_{d,I}^t$ と A_I 内において目的関数値を最良にした $p_{g,I}$ を求める．

(STEP6)ペア毎の有効な探索領域 A_I に、共通した領域があり、さらに $p_{g,I}$ が共通した領域内に存在する場合は領域を結合し、式(8)により新たに生成される有効な探索領域を設定する．また有効な探索領域の数を $I=I-1$ とする．

(STEP7) A_I 内の探索点の位置と速度を式(1),(2)により更新．

(STEP8)慣性項 w を式(4)により更新．また式(8)中の a を式(11)により更新．

(STEP9)最大探索回数 k_{\max} 以下なら、 $k=k+1$ としてSTEP3へ戻る．そうでなければ、探索終了する．

4.1 本論文で提案する方法の特徴 本論文で提案する複数の局所的最適解と大域的最適解を同時に求めるためのARPSOでは、以下のような特徴を持つ．

(1)パラメータの増加がない： シェアリングパラメータを別途設定する必要なく、単一目的関数下におけるPSOを遵守しており、複数の最適解を求めることが可能である．例えば文献(6)では、GAで用いられるシェアリングパラメータ σ_{share} の推奨式として、

$$\sigma_{share} = \frac{\sqrt{\sum_{i=1}^{ndv} (x_i^U - x_i^L)^2}}{2ndv\sqrt{q}} \quad (18)$$

が挙げられている．ここで、 q は求めたい最適解の数を表している．この方法では、 q 値を事前に設定しなければならない．一方でPSOの探索点数について、文献(14)によれば20~30がよいとされていることを踏まえれば、本論文で提案する方法では、例えば探索点数が20の場合、最大で10個の最適解を発見することが初めから制約されてしまい、結果的には式(18)中の q 値を用いていることと同じであると思われる．しかし、本論文では2.2節で述べたPSOの性質を活用したも

のであり、本質的なパラメータの増加がない．

(2) PSOの基礎的性質を遵守している： 2.2節で記述した通り、PSOは最低二つの探索点があれば、探索可能であり、探索点同士がペアを組むことにより生成された有効な探索領域内に存在すると考えられる局所的もしくは大域的最適解の探索を可能としている．

(3) 有効な探索領域の結合： 本論文で提案するアルゴリズムでは、複数の有効な探索領域が生成されるが、3.4節で述べた方法により、有効な探索領域が結合することもある．有効な探索領域が結合しなければ、多峰性関数の最適解を複数見つけることが期待でき、一方で凸関数に対しては、各有効な探索領域が結合することにより、最適解を求めることができる．さらに探索初期段階で有効な探索領域の結合が起きた場合、幅広い探索が可能となることもあり、大域的最適解の探索能力の向上が可能となると考えられる．一方、探索終盤で探索領域が結合する場合は、局所的最適解の探索能力が向上するものと考えられる．すなわち、進化的計算法で重要とされている探索点の多様性と集中化が行われている．

5. 数値計算例

数値計算例を通じて、本論文で提案する方法の有効性を検討する．式(11)中のパラメータ a_{\max} と a_{\min} は、それぞれ次のように設定した．

$$a_{\min} = 1.0 \times 10^{-5} \quad (19)$$

$$a_{\max} = 0.882 \quad (20)$$

5.1 2変数多峰性関数 有効な探索領域の変動を可視化するため、次の簡単な多峰性関数の最適解を求める問題を考える⁽¹⁵⁾．

$$f(x) = -0.5 \exp\left[\frac{-(x_1+2)^2 - (x_2+2)^2}{2}\right] - 1.0 \exp\left[\frac{-(x_1+2)^2 - (x_2-2)^2}{2}\right] - 1.5 \exp\left[\frac{-(x_1-2)^2 - (x_2+2)^2}{2}\right] - 2.0 \exp\left[\frac{-(x_1-2)^2 - (x_2-2)^2}{2}\right] \rightarrow \min \quad (21)$$

$$-5 \leq x \leq 5 \quad (22)$$

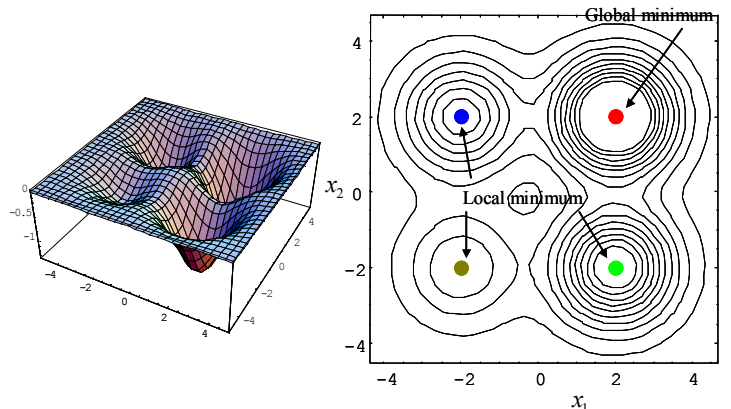


Fig.5 Behavior and contour of objective function

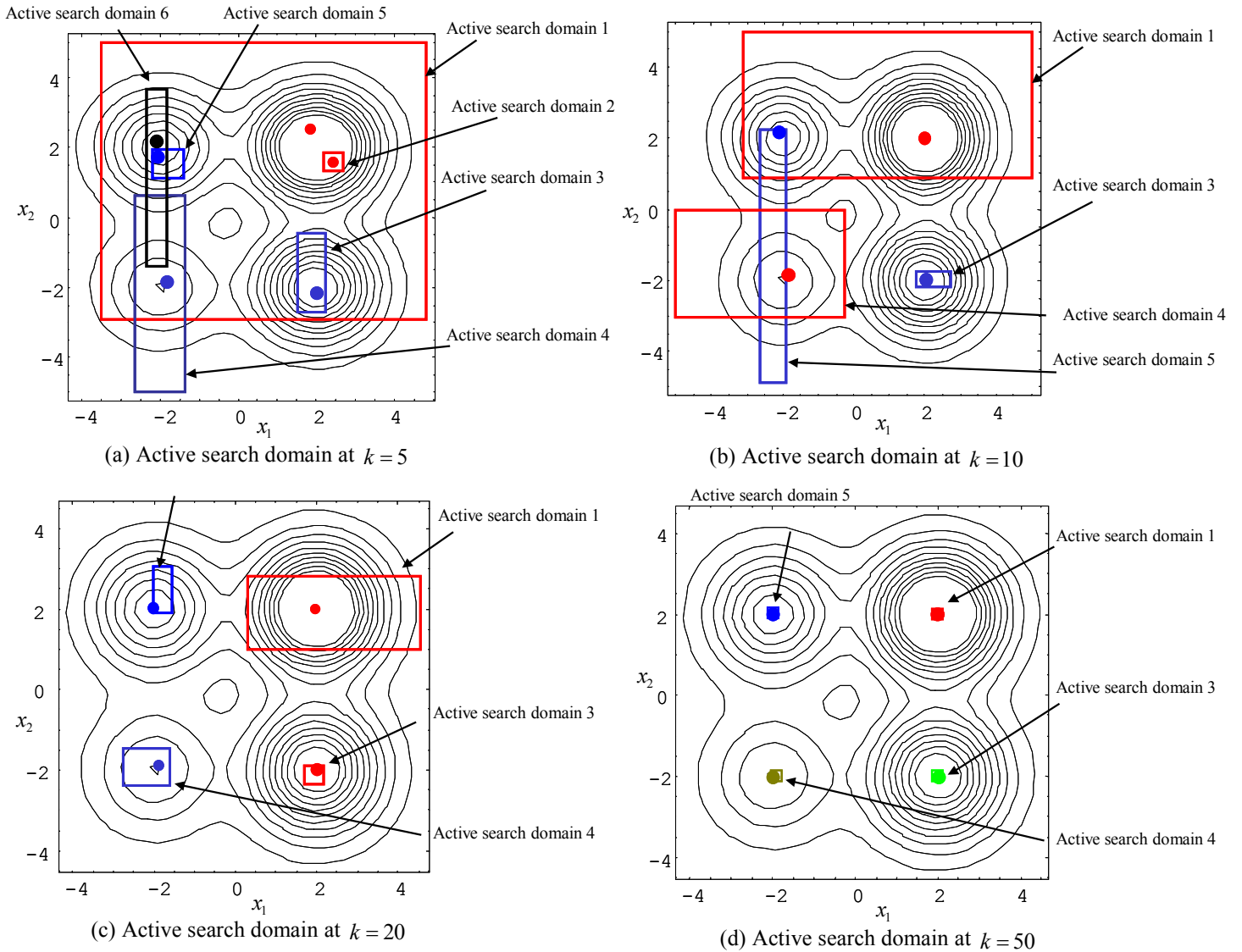


Fig. 6 Search process to find multiple optima

この関数の大域的最適解 \mathbf{x}_G および局所的最適解 \mathbf{x}_L はそれぞれ、次のようになる.

$$\mathbf{x}_G = (2, 2)^T \quad f(\mathbf{x}_G) = -2.0 \quad (23)$$

$$\mathbf{x}_L = (2, -2)^T \quad f(\mathbf{x}_L) = -1.5 \quad (24)$$

$$\mathbf{x}_L = (-2, 2)^T \quad f(\mathbf{x}_L) = -1.0 \quad (25)$$

$$\mathbf{x}_L = (-2, -2)^T \quad f(\mathbf{x}_L) = -0.5 \quad (26)$$

関数の様相と等高線を図5に示す. なお図5右の●は大域的最適解および局所的最適解を表している.

探索点数を20, すなわち最大で10個の最適解を求めることを目的とし, 最大探索回数を $k_{\max} = 100$ とした. 単一目的のPSOやARPSOでは, 2000回の計算コスト (ファンクションコール: 一般には探索点数と最大探索回数の積で与えられる) で一つの大域的最適解を求めることが目的となるが, 本論文で提示する手法では, 同じ計算コストで複数の最適解を求めることが目的となる.

この問題において, 複数の最適解の探索過程の一例を図6に示す. 探索初期段階では10個の有効な探索領域が生成されるが, いくつかの有効な探索領域が重な

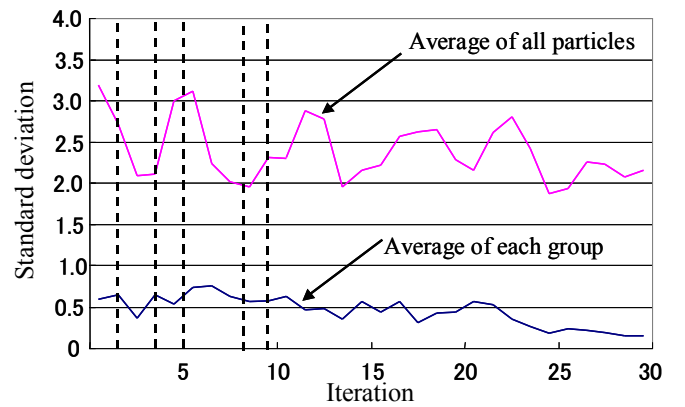


Fig.7 Standard deviation of x_1

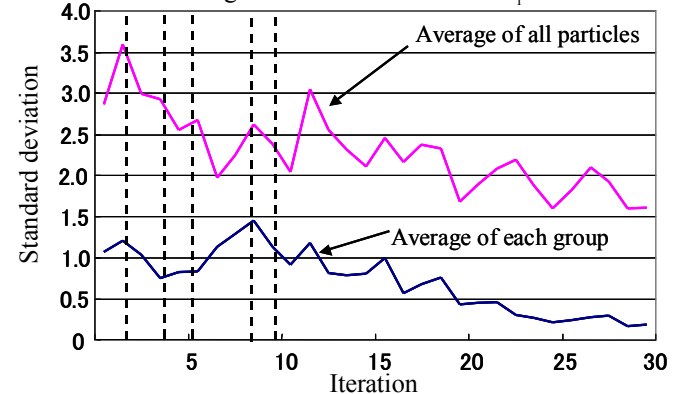


Fig.8 Standard deviation of x_2

り、見づらいため、探索回数 $k=5$ 以降の様子を図6(a)～(d)に示す。図6中の四角で囲まれた領域が有効な探索領域であり、●は各有効な探索領域内における目的関数値を最良にする探索点を表している。

図6(a)では、有効な探索領域が6個生成されている。(つまりこの時点で、4個の有効な探索領域が結合されていることになる。)次に図6(b)において、図6(a)と比較すれば、図6(a)の有効な探索領域1と2が結合し、図6(b)中の新しい有効な探索領域1が生成されている。同様に、図6(a)の有効な探索領域5と6が結合し、図6(b)中の新しい有効な探索領域5が生成されている。探索回数を重ねる毎に、有効な探索領域の結合と生成が繰り返し行われ、図6(c)では、局所的最適解付近に有効な探索領域が生成されていることがわかる。そして図6(d)において、有効な探索領域が徐々に探索領域が狭くなっていることがわかる。最終的には図5中の●で示す大域のおよび局所的最適解に探索点が集中した。

次に、すべての探索点間の多様性の評価の一つの指標として、 $k=30$ までの各設計変数の標準偏差の履歴を図7、図8に示す。両図中にある縦の破線は、有効な探索領域が結合したときの探索回数を表している。両図において、各グループにおける有効な探索領域の標準偏差の平均は、探索が進めば徐々に小さくなっており、これは、各グループが精度の高い最適解を見つけるように動いていることを示す結果である。また、有効な探索領域が結合した箇所では、各設計変数の標準偏差の履歴において、凸凹が生じていることがわかる。一方で、図6に示したように、最適解の存在箇所がばらついているため、すべての探索点の標準偏差の平均は、ある一定値に収束しており、探索点の多様性が維持されていることがわかる。すなわち、多点同時探索型最適化手法に望まれる探索点の多様性維持と集中化が、本論文で提案した手法では同時に達成されていることがわかる。

5.2 ベンチマーク問題への適用例 本論文で提案する方法を表1に示すベンチマーク問題へ適用した。設計変数の数はすべて10とし、最大探索回数を $k_{\max}=500$ とした。

すべての数値計算例において、探索点の数を20と30にして、それぞれ10回の試行を行い、最終的に見つかった最適解の数を表2から表4に示す。各表において、「Best」、「Worst」はそれぞれ、10回の試行中で得られた最大、最小の最適解の数を表しており、「Average」は得られた最適解の平均数を表している。これらの結果から、複数の最適解が得られていることがわかる。なお、得られた最適解に関しては、PSOが関数の感度を用いていないため、解の最適性が不明で

Table 2 Result of 2n minma function

Number of agent		Number of Optima
20	Best	10
	Worst	8
	Average	9
30	Best	14
	Worst	9
	Average	12

Table 3 Result of griewank function

Number of agent		Number of Optima
20	Best	8
	Worst	4
	Average	6
30	Best	10
	Worst	5
	Average	7

Table 4 Result of rastrigin function

Number of agent		Number of Optima
20	Best	8
	Worst	4
	Average	7
30	Best	12
	Worst	5
	Average	9

Table 1 Test functions

No.	Name	Objective function	Side constraints
1	2^n minima	$f(\mathbf{x}) = \sum_{i=1}^{ndv} (x_i^4 - 16x_i^2 + 5x_i)$	$-10 \leq \mathbf{x} \leq 10$
2	Griewank	$f(\mathbf{x}) = 1 + \frac{1}{400} \sum_{i=1}^{ndv} x_i^2 - \prod_{i=1}^{ndv} \cos\left(\frac{x_i}{\sqrt{i}}\right)$	$-10 \leq \mathbf{x} \leq 10$
3	Rastrigin	$f(\mathbf{x}) = \sum_{i=1}^{ndv} (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$-5.12 \leq \mathbf{x} \leq 5.12$

ある．そこで，解の最適性を検討するため，得られたすべての最適解に対して，その点を初期点として，汎用最適設計ソフトウェアDOTを用いて，再度最適解の探索を計算し，その最適性を確認している．また，表1に示したベンチマーク問題においては，すべての試行で大域的最適解を見つけることができた．

5.3 重量最小化設計への適用 文献(16)のI字断面はりの重量最小化設計を考える．設計変数は図9(b)に示す $t_1(=x_1)$ ， $t_2(=x_2)$ ， $b(=x_3)$ ， $h(=x_4)$ であり，すべて連続変数である．最適設計問題は次のように定式化される．

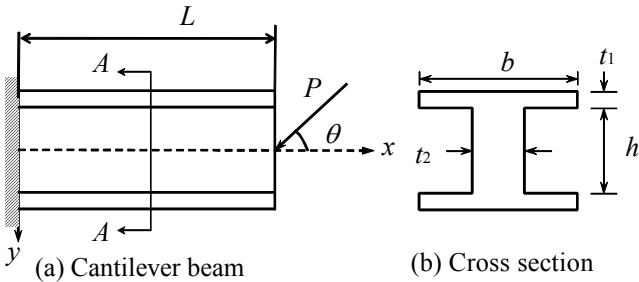


Fig.9 Minimum weight design of cantilever beam

$$f(\mathbf{x}) = (2x_1x_3 + x_2x_4)L \rightarrow \min \quad (27)$$

$$g_1(\mathbf{x}) = \frac{P \cos \theta}{2x_1x_3 + x_2x_4} + \frac{PL \sin \theta}{Z} - 10^8 \leq 0 \quad (28)$$

$$g_2(\mathbf{x}) = \frac{3P \cos \theta}{2} \frac{x_3(2x_1 + x_4)^2 - x_4^2(x_3 - x_2)}{x_3(2x_1 + x_4)^3 - x_4^3(x_3 - x_2)} - 6.00 \times 10^7 \leq 0 \quad (29)$$

$$g_3(\mathbf{x}) = PL^3 \sin \theta / 3EI - 0.015 \leq 0 \quad (30)$$

$$g_4(\mathbf{x}) = P \cos \theta - \pi^2 EI / 4L \leq 0 \quad (31)$$

$$0.10 \leq x_1 \leq 1.00 \quad (32)$$

$$0.10 \leq x_2 \leq 1.50 \quad (33)$$

$$10.0 \leq x_3 \leq 100 \quad (34)$$

$$0.10 \leq x_4 \leq 15.0 \quad (35)$$

ただし $P = 70[kN]$ ， $L = 150[cm]$ ， $\theta = \pi/4$ ， $E = 200[GPa]$ であり，またI字断面はりの断面二次モーメント I と断面係数 Z は

$$I = \frac{x_3(2x_1 + x_4)^3 - x_4^3(x_3 - x_2)}{12} \quad (36)$$

$$Z = \frac{x_3(2x_1 + x_4)^3 - x_4^3(x_3 - x_2)}{6(2x_1 + x_4)} \quad (37)$$

で与えられる．大域的最適解 \mathbf{x}_G と目的関数値は

$$\mathbf{x}_G = (1.00, 0.5218, 50.4437, 15.0)^T [cm] \quad (38)$$

$$f(\mathbf{x}_G) = 16307.20 [cm^3] \quad (39)$$

である．

この問題は，実行可能領域内に複数の最適解が存在する多峰性設計問題である．探索点数を30，最大探索回数を500とし，10回の試行を行った．制約条件である式(28)～(31)は，文献(2)の方法を用い，ペナルティ関数として扱った．結果の一例を表5に示す．また，各試行において得られた最適解の中で，最も目的関数値のよかったものを表6に示す．

表5より，大域的最適解と複数の次善の局所的最適解が得られており，また表6より，すべての試行において，精度の高い最適解が複数得られていることがわかる．

5.4 今後の検討課題

本論文で示した数値計算例に限れば，大域的最適解と複数の局所的最適解を見つけることができたが，例えば表1に示したベンチマーク問題において，設計変数の数がさらに増加した場合は大域的最適解を見つけることができない試行もあつ

Table 5 Objective and constraints at some optimal solutions

Number of optima	$x_1(t_1)$	$x_2(t_2)$	$x_3(b)$	$x_4(h)$	$g_1(\mathbf{x})$	$g_2(\mathbf{x})$	$g_3(\mathbf{x})$	$g_4(\mathbf{x})$	obj.
1	1.00000	0.52181	50.44377	15.00000	0.00000	-0.00012	-1.06915	-14452173.29	16307.20
2	0.50514	0.53938	100.00000	15.00000	0.00000	-4.18957	-1.04308	-13605493.43	16367.95
3	1.00000	0.81605	49.73285	15.00000	-0.10797	-2134.76246	-1.06888	-14433833.94	16755.96
4	1.00000	1.50000	48.09136	15.00000	0.00000	-3860.84467	-1.06829	-14394305.61	17802.41

Table 6 The results through 10 trials and best solution

Trial	Number of Optima	$x_1(t_1)$	$x_2(t_2)$	$x_3(b)$	$x_4(h)$	$g_1(\mathbf{x})$	$g_2(\mathbf{x})$	$g_3(\mathbf{x})$	$g_4(\mathbf{x})$	obj.
1	2	1.00000	0.52181	50.44377	15.00000	0.00000	0.00000	-1.06915	-14452173.29	16307.20
2	4	1.00000	0.52181	50.44377	15.00000	0.00000	-0.00012	-1.06915	-14452173.29	16307.20
3	5	1.00000	0.52181	50.44377	15.00000	0.00000	-0.00042	-1.06915	-14452173.29	16307.20
4	4	1.00000	0.52181	50.44377	15.00000	-0.00003	-0.00054	-1.06915	-14452173.32	16307.20
5	2	1.00000	0.52181	50.44377	15.00000	-0.00029	-0.00904	-1.06915	-14452173.65	16307.20
6	4	1.00000	0.52181	50.44377	15.00000	0.00000	-0.01305	-1.06915	-14452173.22	16307.20
7	5	1.00000	0.52181	50.44378	15.00000	-0.00249	-0.00042	-1.06915	-14452176.90	16307.21
8	5	1.00000	0.52181	50.44378	15.00000	-0.00267	-0.00378	-1.06915	-14452177.15	16307.21
9	3	1.00000	0.52181	50.44378	15.00000	-0.00196	-0.01687	-1.06915	-14452175.84	16307.21
10	5	1.00000	0.52191	50.44446	15.00000	-0.17985	-1.11548	-1.06916	-14452428.37	16307.63

た。これは、探索点がペアを組んで生成する有効な探索領域が設計変数空間において非常に離れていて有効な探索領域が全く結合しなかったり、有効な探索領域が設計変数空間において非常に近くにあり、探索の早期段階で結合してしまったためである。このような状況を避けるために、有効な探索領域の結合もしくは分解というような仕組みを取り入れ、大域的最適解探索能力の向上を図る必要があると考えている。

6. 結言

本論文では、PSOが2つの探索点が存在すれば最適解の探索が可能であるとの点に着目し、筆者らが開発した領域適応型Particle Swarm Optimization(ARPSO)を基調とし、複数の最適解を同時に求めるアルゴリズムを提案した。提案したアルゴリズムでは、探索のはじめに、設計空間内において、最近隣にある探索点同士がペアを組むことにより、有効な探索領域を決定し、各有効な探索領域に含まれる探索点が最適解を探索する方法である。本論文で提案した方法は、従来の多点同時探索型最適化手法において一つの最適解を求めるために要する計算コスト（一般には探索点数と最大探索回数の積）で、複数の最適解が得られるという特徴がある。また各ペアの探索領域が重なり、かつ目的関数値を最良とする探索点が重なった領域内に存在する場合は、探索領域を結合し、新たな探索領域を生成する方法である。提案した方法は、文献(14)で報告されている探索点数の制約等を考慮すれば、必ずしも問題に内在する最適解をすべて求めることはできないが、いくつかの局所的小および大域的最適解を同時に求めることができる。数値計算例を通じて、本論文で提案するアルゴリズムの有効性の一端を確認した。

本研究を行うにあたり、山川宏先生（早稲田大学）、杉本博之先生（北海学園大学）、中山弘隆先生（甲南大学）に貴重なご助言をいただいた。ここに感謝の意を表したい。

参考文献

- (1) Parsopoulos, K.E., Vrahatis M.N., Recent approaches to global optimization problems through Particle Swarm Optimization, *Natural Computing*, **1**, (2002), pp.235-306.
- (2) Kitayama, S., Arakawa, M., Yamazaki, K., Proposal of Adaptive Range Particle Swarm Optimization, *Trans. of the Japan Society of Mechanical Engineers, Series C*, **73**-725, (2007), pp.280-287.
- (3) Mori, K., Tsukiyama, M., Fukuda, T., Immune Algorithm with Searching Diversity and its Application to Resource Allo-

- cation Problem, *Tran. of IEE Japan*, Vol.113-C, No.10, (1993), pp.872-878.
- (4) Arakawa, M., Miyashita, T., Ishikawa, H., Development of Genetic Range Genetic Algorithms (3rd report), *Trans. of the Japan Society of Mechanical Engineers, Series C*, **70**-695, (2004), pp.2062-2069. (in Japanese).
- (5) Kitayama, S., Yamazaki, K., Global Optimization by Generalized Random Tunneling Algorithm (3rd report), *Trans. of the Japan Society of Mechanical Engineers, Series A*, **70**-695, (2006), pp.970-977. (in Japanese)
- (6) Deb, K., *Multi-Objective Optimization using Evolutionary Algorithms*, (2001), Wiley.
- (7) Parsopoulos, K.E., Vrahatis M.N., On the computation of all global minimizers through particle swarm optimization, *IEEE Trans. on Evolutionary Computation*, Vol.8, No.3, (2004), pp.211-224.
- (8) Ogata, H., Ono, S., Nakayama, S., Fundamental Consideration on Multiple-Solution Search by Pair Particle Swarm Optimization, *System, Control, and Information Engineers*, Vol.19, No.8, (2006), pp.330-332. (in Japanese)
- (9) Ge, R.P., A Filled Function Method for finding a Global Minimizer of a Function of Several Variables, *Mathematical Programming*, Vol.46, (1990), pp.191-204.
- (10) Hadjisavvas, N., Pardalos, P.M.(eds.), *Advances in Convex Analysis and Global Optimization*, Kluwer Academic Publishers, (2001), pp.553-567.
- (11) Kitayama, S., Arakawa, M., Yamazaki, K., Penalty Function Approach for the Mixed Discrete Non-Linear Problems by Particle Swarm Optimization, *Structural and Multidisciplinary Optimization*, Vol.32, No.3, (2006), pp.191-202.
- (12) Brandstatter, B., Baumgartner, U., Particle Swarm Optimization - Mass-Spring System Analogon, *IEEE Trans. on Magnetics*, Vol.38, No.2, (2002), pp.997-1000.
- (13) Yasuda, K., Ishigame, A., Nonlinear Programming Algorithm - from the practical point of view-, *System, Control, and Information Engineers*, Vol.50, No.9, (2006), pp.1-7. (in Japanese)
- (14) Schutte, F., Groenwold, A., A., A Study of Global Optimization Using Particle Swarm, *Journal of Global Optimization*, Vol.31, (2005), pp.93-108.
- (15) Shin, Y.S., Grandhi, R.V., A Global Structural Optimization Technique Using an Interval Method, *Structural and Multidisciplinary Optimization*, Vol. 22, (2001), pp.351-363.
- (16) Arora, S., Introduction to Optimum Design, Elsevier Academic Press, (2004), p.430.