

チェビシェフの不等式に基づくロバスト最適化と信頼性の緩和による累積サンプリング

田川 聖治^{1,a)} 藤田 翔平¹

受付日 2016年5月3日, 再受付日 2016年8月18日,
採録日 2016年9月27日

概要: 本論文では, 最悪状況を考慮した最適化手法を提案する. まず, 目的関数や制約条件に含まれる不確実な関数値を確率変数と見なし, チェビシェフの不等式から導出した関数値の予測区間の上限値を解の最悪状況の評価指標として最適化問題を定式化する. 次に, 「最悪状況を考慮した最適化問題」の解を効率的に探索するため, 上限値の計算に必要となる標本数を削減できる3種類の技法(累積サンプリング, 信頼性の緩和, Uカット)を示し, それらを組み込んだ差分進化に基づく最適化アルゴリズムを構築する. 最後に, 複数のテスト問題と工学設計問題を用いた数値実験により, 提案手法の有効性を検証する.

キーワード: ロバスト最適化, チェビシェフの不等式, 標本の節約法, 差分進化

Robust Optimization Based on Chebyshev Inequality and Accumulative Sampling with Reliability Relaxation

KIYOHARU TAGAWA^{1,a)} SHOHEI FUJITA¹

Received: May 3, 2016, Revised: August 18, 2016,
Accepted: September 27, 2016

Abstract: In this paper, a new worst-case optimization method is proposed. In the worst-case optimization problem, each of the uncertain functions' values included in an objective function and constraints is regarded as a random variable. Then, according to the Chebyshev inequality, the prediction interval of the random variable is evaluated from a number of samples and used to estimate the worst-case. Furthermore, for solving the worst-case optimization problem efficiently, an optimization algorithm based on differential evolution is presented with three sample-saving techniques, namely, the accumulative sampling, reliability relaxation, and U-cut. Finally, the usefulness of the proposed worst-case optimization method is demonstrated through the numerical experiment conducted on three test problems and two engineering design problems.

Keywords: robust optimization, Chebyshev inequality, sample-saving technique, differential evolution

1. はじめに

ロバスト最適化問題とは, 広い意味で実世界の不確実性を考慮した最適化問題である. 不確実性のモデル化方法とともに, ロバスト最適化問題の記述方法はいくつもあるが, 非確率論的なものと確率論的なものに大別される [1]. 前者では不確実な変数の領域やいくつかの状況(シナリオ)が与えられ, 後者では不確実な変数の確率分布が与えられる.

また, 実世界では大きなリスクを避けるため, 最悪状況を想定した意思決定や設計が望まれる. 非確率論的なロバスト最適化問題では, 不確実な変数に対する関数値の領域を求めることで最悪状況を判定する [2]. 一方, 確率論的なロバスト最適化問題では最悪状況が自明でないため, シックス・シグマを冠する手法 [3], [4] のように, 関数値の平均や分散から最悪状況の評価指標を定義する. ここで, 関数値が正規分布のような既知の確率分布に従うならば, 最悪状況の程度を確率によって保証できる. しかし, 計算機シミュレーションから関数値を求めるような現実的な最適化問題では, その確率分布を推定することは容易でない.

¹ 近畿大学
Kindai University, Higashi-Osaka 577-8502, Japan
^{a)} tagawa@info.kindai.ac.jp

本論文では、確率論的なロバスト最適化問題に属する「最悪状況を考慮した最適化問題」[5]を定式化する。すなわち、様々な不確実性の影響により、同じ解でも評価するたびに異なる関数値を確率変数と見なし、チェビシェフの不等式から導出した関数値の予測区間の上限値を最悪状況の評価指標とする。チェビシェフの不等式を用いるため、確率変数（関数値）の確率分布が未知であっても、任意の有意水準で精度が保証された予測区間が得られる。

次に、上記のロバスト最適化問題の最適解を効率的に探索するため、差分進化 (DE: Differential Evolution) [6]に基づく最適化アルゴリズムを提案する。関数値の予測区間を算出するためには、1つの解候補を繰り返し評価して関数値の標本を数多く集める必要がある。しかし、計算機シミュレーションによって解候補を評価する場合は、時間や費用が大きな障害となる。そこで、提案する最適化アルゴリズムでは、3種類の標本の節約法（累積サンプリング、信頼性の緩和、Uカット）をDEに組み込んでいる。

累積サンプリングは有望な解候補の標本数を徐々に増やす技法であり、様々なロバスト最適化問題に対する既存の最適化アルゴリズムでも採用されている [7], [8]。本論文では、予測区間の計算に必要な最少の標本数を明らかにするとともに、上限値の収束を数値解析により判定することで、無駄のない累積サンプリングを提案する。ただし、累積サンプリングは予測区間の上限値を過大に評価するため、DEによる最適解の探索を阻害する恐れがある。その欠点を補う技法が「信頼性の緩和」であり、上限値の精度を高く保持するだけでなく、標本数のさらなる削減も可能とする。著者らが考案したUカット [9]は、わずかな標本から有望でない解候補を判別し、それらの評価を中止する技法である。本論文では、提案する累積サンプリングや「信頼性の緩和」と組み合わせた場合におけるUカットの効果を検証する。

以降、2章では標本によるチェビシェフの不等式から、確率変数の予測区間を導出する。3章では最悪状況を考慮した最適化問題を定式化する。4章ではDEに基づく最適化アルゴリズムと、3種類の標本の節約法を説明する。関連研究は5章で紹介する。6章ではテスト問題と工学設計問題を用いた数値実験により、提案手法の有効性を示す。最後に7章において、本論文の結論を述べる。

2. 標本によるチェビシェフの不等式

広く知られたチェビシェフの不等式は、平均 μ 、分散 σ^2 が存在する確率変数 \mathcal{F} と任意の定数 $\lambda > 1$ に対して

$$P_r(|\mathcal{F} - \mu| \geq \lambda\sigma) \leq \frac{1}{\lambda^2} \quad (1)$$

である。ここで、有意水準 α を $\alpha = 1/\lambda^2$ ととると、 \mathcal{F} の値が予測区間 $[\mu - \lambda\sigma, \mu + \lambda\sigma]$ に含まれる確率は

$$P_r([\mu - \lambda\sigma, \mu + \lambda\sigma] \ni \mathcal{F}) \geq 1 - \alpha \quad (2)$$

となる。

チェビシェフの不等式の利点として、確率変数 \mathcal{F} に正規分布のような特定の確率分布を想定しないことがあげられる。しかし、実世界において、大半の確率変数 \mathcal{F} の母平均 μ と母分散 σ^2 は未知である。そこで、Sawら [10]は標本に基づくチェビシェフの不等式を示している。確率変数 \mathcal{F} の N 個の標本 $\mathcal{F}^1, \dots, \mathcal{F}^N$ から、母平均 μ と母分散 σ^2 の推定量である標本平均 \bar{F} と標本不偏分散 s^2 は

$$\bar{F} = \frac{1}{N} \sum_{n=1}^N \mathcal{F}^n \quad (3)$$

$$s^2 = \frac{1}{N-1} \sum_{n=1}^N (\mathcal{F}^n - \bar{F})^2 \quad (4)$$

で与えられる。ここで、チェビシェフの不等式は

$$P_r\left(|\mathcal{F} - \bar{F}| \geq \lambda \sqrt{\frac{N+1}{N}} s\right) \leq \frac{1}{N+1} \left\lfloor \frac{(N+1)(N-1+\lambda^2)}{N\lambda^2} \right\rfloor \quad (5)$$

となる [10]。ただし、 $\lfloor r \rfloor$ は実数 $r \in \mathbb{R}$ の床関数である。

本論文では、式 (5) から確率変数 \mathcal{F} の予測区間を導出する。まず、式 (5) の右辺は $\lfloor r \rfloor \leq r$ であるため、

$$P_r\left(|\mathcal{F} - \bar{F}| \geq \lambda \sqrt{\frac{N+1}{N}} s\right) \leq \frac{N-1+\lambda^2}{N\lambda^2} \quad (6)$$

となる。次に、以下のように係数 κ をおく。

$$\kappa = \lambda \sqrt{\frac{N+1}{N}} \quad (7)$$

式 (7) の係数 κ を式 (6) に代入すると、

$$P_r(|\mathcal{F} - \bar{F}| \geq \kappa s) \leq \frac{N^2 - 1 + N\kappa^2}{N^2\kappa^2} \quad (8)$$

となる。ここで、以下のように有意水準 α をとる。

$$\alpha = \frac{N^2 - 1 + N\kappa^2}{N^2\kappa^2} \quad (9)$$

式 (8) と式 (9) から、確率変数 \mathcal{F} の予測区間は

$$P_r([\bar{F} - \kappa s, \bar{F} + \kappa s] \ni \mathcal{F}) = P_r([\mathcal{F}^L, \mathcal{F}^U] \ni \mathcal{F}) \geq 1 - \alpha \quad (10)$$

となる。ただし、 $\mathcal{F}^L = \bar{F} - \kappa s$ 、 $\mathcal{F}^U = \bar{F} + \kappa s$ とする。

式 (9) から係数 κ は有意水準 α で定義できる。

$$\kappa = \sqrt{\frac{N^2 - 1}{N(\alpha N - 1)}} \quad (11)$$

式 (7) から $\kappa > 1$ であるため、式 (11) により κ の値を決める際に必要となる標本数 N の最小値 N_{\min} は

$$N_{\min} = \left\lfloor \frac{1}{\alpha} + 1 \right\rfloor \quad (12)$$

である。また、式 (11) から κ の極限值は以下となる。

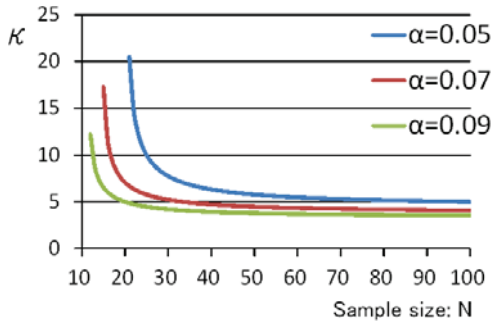


図 1 標本数 N に対する係数 κ の変化
 Fig. 1 Change of κ for the sample size N .

$$\lim_{N \rightarrow \infty} \kappa = \lim_{N \rightarrow \infty} \sqrt{\frac{N^2 - 1}{N(\alpha N - 1)}} = \sqrt{\frac{1}{\alpha}} \quad (13)$$

図 1 に標本数 N に対する係数 κ の変化を示す。標本数 N の増加にとまらぬ κ の値は減少し、式 (13) の $\sqrt{1/\alpha}$ に収束する。また、有意水準 α が小さいほど κ は大きい。

本章の要点をまとめると、任意の有意水準 α に対して、式 (10) の予測区間 $[\mathcal{F}^L, \mathcal{F}^U]$ の上限値 $\mathcal{F}^U = \bar{F} + \kappa s$ は、式 (12) の最小値 N_{\min} 以上の N 個の標本 $\mathcal{F}^1, \mathcal{F}^2, \dots, \mathcal{F}^N$ から、式 (3), (4), (11) によって求められる。

3. ロバスト最適化問題の定式化

制約条件付き最適化問題は式 (14) で記述される。

$$\begin{cases} \min & f(\mathbf{x}) = f(x_1, x_2, \dots, x_D) \\ \text{sub. to} & g_m(\mathbf{x}) \leq 0, m \in I_M \\ & \mathbf{x} = (x_1, x_2, \dots, x_D) \in \mathbf{X} \end{cases} \quad (14)$$

ただし、 $\mathbf{X} = \{\mathbf{x} \in \mathbb{R}^D \mid x_j^L \leq x_j \leq x_j^U, j \in I_D\}$ 、添字集合を $I_M = \{1, \dots, M\}$ 、 $I_D = \{1, \dots, D\}$ とする。

現実的な最適化問題を扱う場合は、様々な不確実性の影響を考慮する必要がある。そこで、各決定変数 $x_j \in \mathbb{R}$ の誤差 $\delta_j \in \mathbb{R}$ や不確実な係数 θ 、ノイズ ξ などを既知の確率分布に従う確率変数とし、式 (14) の最適化問題に含まれる関数 $f(\mathbf{x})$ と $g_m(\mathbf{x})$ 、 $m \in I_M$ を以下のように拡張する。

$$\begin{cases} \mathcal{F}(\mathbf{x}) = f(\mathbf{x} + \boldsymbol{\delta}, \boldsymbol{\theta}) + \xi \\ \mathcal{G}_m(\mathbf{x}) = g_m(\mathbf{x} + \boldsymbol{\delta}, \boldsymbol{\theta}) + \xi, m \in I_M \end{cases} \quad (15)$$

ただし、 $\boldsymbol{\delta} = (\delta_1, \dots, \delta_j, \dots, \delta_D) \in \mathbb{R}^D$ である。

関数値 $\mathcal{F}(\mathbf{x})$ は解 $\mathbf{x} \in \mathbb{R}^D$ に依存するが、解を評価するたびに異なる値が観測される。このため、式 (15) の $\mathcal{F}(\mathbf{x})$ は確率変数と見なせるが、現実的な最適化問題では $f(\mathbf{x})$ の定義式が複雑であるため、誤差 $\boldsymbol{\delta} \in \mathbb{R}^D$ や係数 $\boldsymbol{\theta}$ などの確率分布が既知であっても、 $\mathcal{F}(\mathbf{x})$ の確率分布を推定することは困難である。そこで、有意水準 α で信頼性が保証された解の最悪状況を式 (10) の上限値 \mathcal{F}^U で判定する。すなわち、解 $\mathbf{x} \in \mathbb{R}^D$ を繰り返し評価して得られた標本集合 $\{\mathcal{F}^n(\mathbf{x}) \mid n = 1, \dots, N\}$ から、上限値 $\mathcal{F}^U(\mathbf{x})$ を求める。同様に、制約条件に含まれる各関数値 $\mathcal{G}_m(\mathbf{x})$ 、 $m \in I_M$ の

標本集合から、上限値 $\mathcal{G}_m^U(\mathbf{x})$ 、 $m \in I_M$ を求める。

ここで、本論文で対象とする「最悪状況を考慮した最適化問題」(WCOP: Worst Case Optimization Problem) を、式 (14) を拡張して式 (16) のように定式化する。

$$\begin{cases} \min & \mathcal{F}^U(\mathbf{x}) = \mathcal{F}^U(x_1, x_2, \dots, x_D) \\ \text{sub. to} & \mathcal{G}_m^U(\mathbf{x}) \leq 0, m \in I_M \\ & \mathbf{x} = (x_1, x_2, \dots, x_D) \in \mathbf{X} \end{cases} \quad (16)$$

ただし、標本数 N ($N \geq N_{\min}$) は十分に大きいとする。

4. 差分進化に基づく最適化アルゴリズム

4.1 基本的な DE のアルゴリズム

基本的な DE (DEB と呼ぶ) を式 (16) の WCOP に適用する。DEB では WCOP の解候補を個体と呼び、 N_P 個の個体 $\mathbf{x}_i \in \mathbf{P}$ 、 $i = 1, \dots, N_P$ を集団 \mathbf{P} とする。DEB では各個体 $\mathbf{x}_i \in \mathbf{P}$ を N 回ずつ評価して得られた関数値の標本集合から、上限値 $\mathcal{F}^U(\mathbf{x}_i)$ と $\mathcal{G}_m^U(\mathbf{x}_i)$ 、 $m \in I_M$ を求める。ただし、標本数 N は十分に大きな値に決めておく。

古典的な DE [6] の制御パラメータは、集団サイズ N_P 、スケール係数 S_F 、交叉率 C_R である。DE の探索性能は S_F と C_R の値に大きく左右されるため、DEB では Brest ら [11] による S_F と C_R の自動調整法を採用し、各個体 $\mathbf{x}_i \in \mathbf{P}$ に固有のスケール係数 $S_{F,i}$ と交叉率 $C_{R,i}$ を割り当てる。また、DE の世代交代モデルには同期型と非同期型がある [12]。DEB では非同期型の世代交代モデルを採用し、1 つの集団内の個体 $\mathbf{x}_i \in \mathbf{P}$ を随時更新する。

以下に WCOP に対する DEB のアルゴリズムを示す。

〔基本的な DE (DEB)〕

手順 1 個体 $\mathbf{x}_i \in \mathbf{X}$ 、 $i = 1, \dots, N_P$ をランダムに生成して初期集団 \mathbf{P} とする。全個体のスケール係数と交叉率を $S_{F,i} = 0.5$ 、 $C_{R,i} = 0.9$ と初期化する [11]。

手順 2 各個体 $\mathbf{x}_i \in \mathbf{P}$ 、 $i = 1, \dots, N_P$ を N 回評価し、上限値 $\mathcal{F}^U(\mathbf{x}_i)$ と $\mathcal{G}_m^U(\mathbf{x}_i)$ 、 $m \in I_M$ を求める。

手順 3 終了条件を満たせば、実行可能な個体 $\mathbf{x}_i \in \mathbf{P}$ で目的関数値が最小のものを解とする。実行可能な個体が集団 \mathbf{P} に存在しなければ、探索は失敗である。

手順 4 N_P 個の各個体 $\mathbf{x}_i \in \mathbf{P}$ を順番にターゲット・ベクトルに指定し、手順 4.1 から手順 4.3 を繰り返す。

手順 4.1 新たな個体の候補であるトライアル・ベクトル $\mathbf{u} \in \mathbf{X} \subseteq \mathbb{R}^D$ を生成する。まず、ターゲット・ベクトル $\mathbf{x}_i \in \mathbf{P}$ の $S_{F,i}$ と $C_{R,i}$ から、スケール係数 S_F と交叉率 C_R を以下のように決める [11]。

$$S_F = \begin{cases} 0.1 + \text{rand}_1 \cdot 0.9, & \text{if } \text{rand}_2 < 0.1 \\ S_{F,i}, & \text{otherwise} \end{cases} \quad (17)$$

$$C_R = \begin{cases} \text{rand}_3, & \text{if } \text{rand}_4 < 0.1 \\ C_{R,i}, & \text{otherwise} \end{cases} \quad (18)$$

ただし、 $\text{rand}_k \in [0, 1]$ は一様乱数である。

次に、 $\mathbf{x}_i \in \mathbf{P}$ とは別に集団 \mathbf{P} からランダムに異なる 3 つの個体 $\mathbf{x}_{r1}, \mathbf{x}_{r2}, \mathbf{x}_{r3}$ ($i \neq r1 \neq r2 \neq r3$) を選び、DE の戦略 (DE/rand/1/bin) [6] を用いて、 \mathbf{u} の要素 $u_j \in \mathbb{R}$, $j \in \mathbf{I}_D$ を以下のように決める。

$$u_j = \begin{cases} x_{r1,j} + S_F(x_{r2,j} - x_{r3,j}), & \text{if } (\text{rand}_j < C_R) \vee (j = j_r) \\ x_{i,j}, & \text{otherwise} \end{cases} \quad (19)$$

ただし、添字 $j_r \in [1, D]$ はランダムに選択する。

上記の結果、 \mathbf{u} の要素 u_j が探索範囲 $[x_j^L, x_j^U]$ の外に作られた場合は、以下のように u_j を修正する [6]。

$$u_j = \begin{cases} x_{r1,j} + \text{rand}_j \eta_j^L, & \text{if } u_j < x_j^L \\ x_{r1,j} + \text{rand}_j \eta_j^U, & \text{if } u_j > x_j^U \end{cases} \quad (20)$$

ただし、 $\eta_j^L = x_j^L - x_{r1,j}$, $\eta_j^U = x_j^U - x_{r1,j}$ とする。

手順 4.2 トライアル・ベクトル \mathbf{u} を N 回評価し、上限値 $\mathcal{F}^U(\mathbf{u})$ と $\mathcal{G}_m^U(\mathbf{u})$, $m \in \mathbf{I}_M$ を求める。

手順 4.3 トライアル・ベクトル \mathbf{u} とターゲット・ベクトル $\mathbf{x}_i \in \mathbf{P}$ を比較し、以下の 3 つの条件のうち、少なくとも 1 つが満たされれば、 \mathbf{u} が $\mathbf{x}_i \in \mathbf{P}$ に勝ると判定し、ただちに $\mathbf{x}_i \in \mathbf{P}$ を \mathbf{u} で置き換える [13]。

- (\mathbf{u} が実行可能) \wedge ($\mathcal{F}^U(\mathbf{u}) \leq \mathcal{F}^U(\mathbf{x}_i)$)
- (\mathbf{u} が実行可能) \wedge (\mathbf{x}_i が実行不可能)
- (\mathbf{u} が実行不可能) \wedge
 $(\forall m \in \mathbf{I}_M; \max\{\mathcal{G}_m^U(\mathbf{u}), 0\} \leq \max\{\mathcal{G}_m^U(\mathbf{x}_i), 0\})$

また、その場合は $S_{F,i} = S_F$, $C_{R,i} = C_R$ とする。

手順 5 手順 3 に戻る。

4.2 累積サンプリングによる標本の節約法

DEB では全個体を N 回ずつ評価して上限値を求める。しかし、現実的な最適化問題ではシミュレーションによる個体の評価に時間や費用を要する。また、最適化の過程では個体間の優劣のみに着目するため、上限値を厳密に求める必要はない。そこで、DEB に累積サンプリングを導入し、有望な個体の標本数を徐々に増やすことで、それらの上限値の精度を高めてゆく。累積サンプリング (Accumulative Sampling) を組み込んだ DEB を DEA と呼ぶ。

DEA は初期集団の個体 $\mathbf{x}_i \in \mathbf{P}$ を N_{ini} 回 ($N_{\text{ini}} \geq N_{\text{min}}$) だけ評価し、上限値 $\mathcal{F}^U(\mathbf{x}_i)$ と $\mathcal{G}_m^U(\mathbf{x}_i)$, $m \in \mathbf{I}_M$ を求める。 N_{min} は式 (12) に示した標本数の最小値である。以降、DEA は 1 世代ごとに集団内の各個体 $\mathbf{x}_i \in \mathbf{P}$ を 1 回だけ再評価し、新たな標本を標本集合に追加して上限値を更新する。ただし、上限値が収束したと見なせる個体の再評価と上限値の更新は行わない。このため、各個体 $\mathbf{x}_i \in \mathbf{P}$ にカウンタ c_i を付与し、 c_i が最大値 \hat{c} ($\hat{c} \geq 1$) に達した時点で \mathbf{x}_i の上限値は収束したと判定する。また、個体ごとに標本数が異なるため、 \mathbf{x}_i の標本数を N_i とする。

以下に WCOP に対する DEA のアルゴリズムを示す。

[累積サンプリングを採用した DE (DEA)]

手順 1 DEB の手順 1 に同じ。 $c_i = 0$ とする。

手順 2 各個体 $\mathbf{x}_i \in \mathbf{P}$, $i = 1, \dots, N_P$ を N_{ini} 回評価し、 $\mathcal{F}^U(\mathbf{x}_i)$ と $\mathcal{G}_m^U(\mathbf{x}_i)$ を求める。 $N_i = N_{\text{ini}}$ とする。

手順 3 終了条件を満たせば、実行可能かつ $c_i = \hat{c}$ である個体 $\mathbf{x}_i \in \mathbf{P}$ で目的関数値が最小のものを解とする。そのような個体がなければ、探索は失敗である。

手順 4 DEB の手順 4 に同じ。

手順 4.1 DEB の手順 4.1 に同じ。

手順 4.2 ターゲット・ベクトル $\mathbf{x}_i \in \mathbf{P}$ の標本数が N_i であるとき、トライアル・ベクトル \mathbf{u} を N_i 回評価し、上限値 $\mathcal{F}^U(\mathbf{u})$ と $\mathcal{G}_m^U(\mathbf{u})$, $m \in \mathbf{I}_M$ を求める。

手順 4.3 DEB の手順 4.3 に同じ。ただし、 $\mathbf{x}_i \in \mathbf{P}$ を \mathbf{u} で置き換えた場合、標本数は N_i , $c_i = 0$ とする。

手順 5 上限値が収束していない各個体 $\mathbf{x}_i \in \mathbf{P}$ ($c_i < \hat{c}$) について、手順 5.1 から手順 5.3 を実行する。

手順 5.1 $y_0^1 = \mathcal{F}^U(\mathbf{x}_i)$, $y_m^1 = \mathcal{G}_m^U(\mathbf{x}_i)$, $m \in \mathbf{I}_M$ とする。

手順 5.2 個体 \mathbf{x}_i を 1 回再評価し、標本数を $N_i = N_i + 1$ として $\mathcal{F}^U(\mathbf{x}_i)$ と $\mathcal{G}_m^U(\mathbf{x}_i)$, $m \in \mathbf{I}_M$ を更新する。次に、 $y_0^2 = \mathcal{F}^U(\mathbf{x}_i)$, $y_m^2 = \mathcal{G}_m^U(\mathbf{x}_i)$, $m \in \mathbf{I}_M$ とする。

手順 5.3 個体 \mathbf{x}_i が式 (21) の条件を満たせば $c_i = c_i + 1$ とし、満たさなければ $c_i = 0$ とする。

$$\forall m' \in \{0\} \cup \mathbf{I}_M; \left| \frac{y_{m'}^2 - y_{m'}^1}{y_{m'}^2} \right| \leq \epsilon \quad (21)$$

手順 6 手順 3 に戻る。

本論文の収束判定では、 $\hat{c} = 3$, $\epsilon = 10^{-3}$ とする。

4.3 信頼性の緩和による標本の節約法

DEA は式 (11) から係数 κ を求めている。図 1 に示したように、標本数が少ないと係数 κ は非常に大きな値となり、DEA は探索の初期段階で上限値を過大に評価する。このため、最適解が狭小な実行可能領域にある場合、それを発見できず、準最適解に初期収束する恐れがある。

「信頼性の緩和」では、標本数が少ない DEA による探索の初期段階で係数 κ を小さな値 $\hat{\kappa}$ に固定する。信頼性の緩和 (Reliability Relaxation) を採用した DEA を DEAR と呼ぶ。要求された有意水準 α に対する標本数の最小値を N_{min} とする。ここで、DEAR では標本数 N_i の個体 ($\mathbf{x}_i \in \mathbf{P}$ および $\mathbf{u} \in \mathbf{X}$) について上限値を計算するとき、式 (11) に代えて、係数 κ を以下の規則により求める。

$$\kappa = \begin{cases} \hat{\kappa}, & \text{if } N_i < N_{\text{min}} \\ \min\{\hat{\kappa}, \kappa_i\}, & \text{if } N_i \geq N_{\text{min}} \end{cases} \quad (22)$$

ただし、

$$\kappa_i = \sqrt{\frac{N_i^2 - 1}{N_i(\alpha N_i - 1)}} \quad (23)$$

とし、式 (13) から $\hat{\kappa}$ は $\hat{\kappa} > \sqrt{1/\alpha}$ の範囲で選択する。

図 2 に標本数 N に対する式 (22) の κ の概念図を示す。

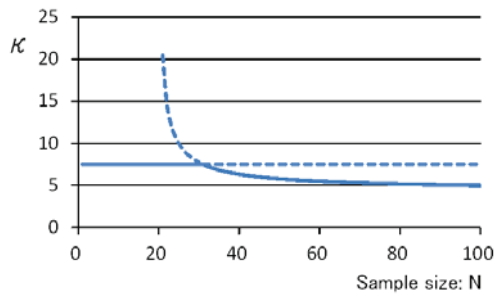


図 2 信頼性の緩和による係数 κ の制御
 Fig. 2 Control of κ with reliability relaxation.

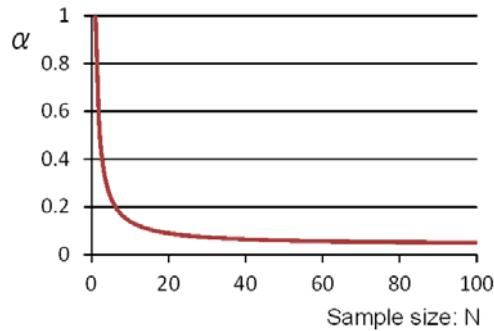


図 3 標本数 N に対する有意水準 α の変化
 Fig. 3 Change of α for the sample size N .

係数 κ を $\hat{\kappa}$ に固定したとき、式 (9) から有意水準 α は

$$\alpha = \frac{N^2 - 1 + N\hat{\kappa}^2}{N^2\hat{\kappa}^2} \quad (24)$$

となり、 $N = 1$ のとき $\alpha = 1$ である。また、標本数 N の増加にともない α は減少し、その極限值は以下となる。

$$\lim_{N \rightarrow \infty} \alpha = \lim_{N \rightarrow \infty} \frac{N^2 - 1 + N\hat{\kappa}^2}{N^2\hat{\kappa}^2} = \frac{1}{\hat{\kappa}^2} \quad (25)$$

図 3 に標本数 N に対する式 (24) の有意水準 α の変化を示す。ただし、図 3 では係数を $\hat{\kappa} = 5.0$ としている。

DEA では有意水準 α を要求された値に固定しているため、標本数により係数 κ の値が変化する。一方、DEAR では実質的な α の値を標本数により変化させることで、標本数に関係なく係数 κ の値を $\hat{\kappa}$ に固定している。すなわち、DEA では α による上限値の信頼性を保証するため、精度 (予測区間幅) を犠牲にしているが、DEAR では信頼性を緩和することで、上限値の精度を高く維持している。

DEA は初期集団の個体 $\mathbf{x}_i \in \mathbf{P}$ を N_{ini} 回 ($N_{\text{ini}} \geq N_{\text{min}}$) 評価する。ここで、有意水準 α が小さな場合、 N_{min} は大きな値となる。一方、DEAR では上限値を式 (3), (4), (22) より求めるため、 N_{ini} は $N_{\text{ini}} \geq 2$ から選択できる。

4.4 U-カットによる標本の節約法

U-カット [9] はトライアル・ベクトル \mathbf{u} とターゲット・ベクトル $\mathbf{x}_i \in \mathbf{P}$ と比較する際、有望でない \mathbf{u} に対する無駄な評価を中止して標本を節約する。U-カットでは、トライアル・ベクトル \mathbf{u} を 1 回ずつ評価して得られた n 番目

の標本 (関数値) を $\mathcal{F}^n(\mathbf{u})$, $\mathcal{G}_m^n(\mathbf{u})$, $m \in \mathbf{I}_M$ とするとき、式 (10) から $\mathcal{F}^n(\mathbf{u}) < \mathcal{F}^U(\mathbf{u})$ や $\mathcal{G}_m^n(\mathbf{u}) < \mathcal{G}_m^U(\mathbf{u})$ の関係が高い確率 $(1 - \alpha)$ で成り立つことを利用している。

U-カットは DEB, DEA, DEAR のいずれでも採用できる。以下に DEB の場合のアルゴリズム (DEB-U) を示す。
 [U-カットを採用した DEB (DEB-U)]

以下の手順 4.2 以外は、DEB の手順に同じ。

手順 4.2 トライアル・ベクトル \mathbf{u} を 1 回ずつ評価し、標本 $\mathcal{F}^n(\mathbf{u})$ と $\mathcal{G}_m^n(\mathbf{u})$, $n = 1, \dots, N$ を求める。途中で以下の 3 つの条件のうち、少なくとも 1 つが満たされれば、 $\mathbf{x}_i \in \mathbf{P}$ が \mathbf{u} に勝ると判定し、ただちに \mathbf{u} を破棄する。その場合、手順 4.3 は実行しない。

- (\mathbf{x}_i が実行可能) \wedge ($\mathcal{F}^U(\mathbf{x}_i) \leq \mathcal{F}^n(\mathbf{u})$)
- (\mathbf{x}_i が実行可能) \wedge ($\exists m \in \mathbf{I}_M; \mathcal{G}_m^n(\mathbf{u}) > 0$)
- (\mathbf{x}_i が実行不可能) \wedge
 $(\forall m \in \mathbf{I}_M; \max\{\mathcal{G}_m^U(\mathbf{x}_i), 0\} \leq \max\{\mathcal{G}_m^n(\mathbf{u}), 0\})$

一方、 \mathbf{u} を無事に N 回評価できた場合、標本集合から上限値 $\mathcal{F}^U(\mathbf{u})$ と $\mathcal{G}_m^U(\mathbf{u})$, $m \in \mathbf{I}_M$ を求める。

5. 関連研究

5.1 最悪状況を考慮した最適化問題

前述のようにロバスト最適化問題は非確率的なものと同様に確率的なものに大別される [1]。まず、非確率的なロバスト最適化問題では、不確実な変数の領域に対応する関数値の領域を求めるため、おのずと最悪状況を扱うことになる。最適化手法としては、線形計画問題や二次錐計画問題に帰着する手法 [2] や、関数をテイラー展開で線形化した後、不確実な変数の区間から、区間解析によって関数値を求める手法 [14] がある。そのほか、区間解析の効率化を図るため、直交表と DE を組み合わせた手法 [15] も報告されている。

一方、確率的なロバスト最適化問題では、不確実な誤差や係数の確率分布が与えられ、関数値の平均や分散から解の様々な評価指標を定義する。確率的なロバスト最適化問題の記述方法や最適化アルゴリズムは、進化計算の分野でもさかんに研究されているが [16]、解の平均的な性能評価が中心であり、最悪状況を扱ったものはほとんどない。

最悪状況を考慮した確率的な解の評価指標として、シックス・シグマ [3], [4] が広く知られている。ある解に対する不確実な関数値 \mathcal{F} が平均 μ 、分散 σ^2 の正規分布に従うとき、正規分布の上側確率 $\alpha/2$ -点を $z_{\alpha/2}$ として

$$P_r([\mu - z_{\alpha/2}\sigma, \mu + z_{\alpha/2}\sigma] \ni \mathcal{F}) = 1 - \alpha \quad (26)$$

となる。ここで、式 (26) の予測区間の上限値は、有意水準 α によって信頼性が保証された最悪状況と見なせる。

式 (26) で関数値 \mathcal{F} の平均 μ と分散 σ^2 を、式 (3) の標本平均 \bar{F} と式 (4) の標本不偏分散 s^2 で推定する場合は

$$P_r([\bar{F} - \beta s, \bar{F} + \beta s] \ni \mathcal{F}) = 1 - \alpha \quad (27)$$

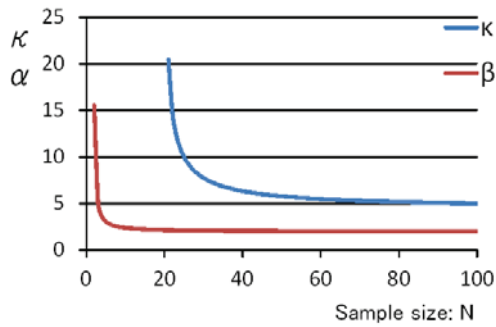


図 4 標本数 N に対する係数 κ と β の比較

Fig. 4 Comparison between κ and β for the sample size N .

となる [17]. ただし, \mathcal{F} の標本数を N , 自由度 $(N - 1)$ の t -分布の上側確率 $\alpha/2$ -点を $t(N - 1, \alpha/2)$ として

$$\beta = t(N - 1, \alpha/2) \sqrt{1 + \frac{1}{N}} \quad (28)$$

とする.

著者らは, 目的関数値を正規分布に従う確率変数とし, 式 (27) の予測区間の上限値を用いて, 最悪状況を考慮した単目的最適化問題 [18] と多目的最適化問題 [9] を定式化し, DE に基づく最適化アルゴリズムを提案している.

図 4 では標本数 N に対する式 (11) の係数 κ と式 (28) の係数 β の変化を比較する. ただし, 有意水準は $\alpha = 0.05$ とした. 図 4 から, 関数値 \mathcal{F} が正規分布に従うならば, その予測区間は式 (10) よりも式 (27) の方が狭く, 本論文で提案した上限値は過大評価となる. しかし, 現実的な最適化問題では関数値に正規分布を仮定できることは稀であり, 保守的でも最悪状況の検討が必要な場合もある.

5.2 機会制約問題との関係

本論文で定式化した式 (16) の WCOP は, 確率計画法 [19] における数理モデルの 1 つである機会制約問題として記述できる. 式 (10) より, 関数値 $\mathcal{G}_m(\mathbf{x})$ の予測区間は

$$Pr([\mathcal{G}_m^L(\mathbf{x}), \mathcal{G}_m^U(\mathbf{x})] \ni \mathcal{G}_m(\mathbf{x})) \geq 1 - \alpha \quad (29)$$

となる. WCOP の制約条件 $\mathcal{G}_m^U(\mathbf{x}) \leq 0$ を考慮すると,

$$\begin{aligned} Pr(\mathcal{G}_m(\mathbf{x}) \leq 0) &= Pr((-\infty, 0] \ni \mathcal{G}_m(\mathbf{x})) \geq \\ &Pr([\mathcal{G}_m^L(\mathbf{x}), 0] \ni \mathcal{G}_m(\mathbf{x})) \geq 1 - \alpha \end{aligned} \quad (30)$$

となる. さらに, WCOP の目的関数 $\mathcal{F}^U(\mathbf{x})$ の上界値を γ として, 関数値 $\mathcal{F}(\mathbf{x})$ の予測区間を式 (30) のように拡張すれば, WCOP は以下のような機会制約問題となる.

$$\left[\begin{array}{ll} \min & \gamma \\ \text{sub. to} & Pr(\mathcal{F}(\mathbf{x}) \leq \gamma) \geq 1 - \alpha \\ & Pr(\mathcal{G}_m(\mathbf{x}) \leq 0) \geq 1 - \alpha, m \in \mathbf{I}_M \\ & \mathbf{x} = (x_1, x_2, \dots, x_D) \in \mathbf{X} \end{array} \right. \quad (31)$$

ランダムイズド・アルゴリズム [20] は, 機会制約問題を数理計画問題の集合に緩和して解く手法である. そこで,

数理計画問題 (緩和問題) の最適解と機会制約問題の実行可能解との関係が精力的に研究されてきた [21], [22].

本論文で提案した DEAR-U は, 機会制約問題に対する新たな解法と考えられる. ランダムイズド・アルゴリズムと比較して, DEAR-U の利点は, 関数に凸性や微分可能の仮定が不要であること, 機会制約問題の近似解ではなく, 実行可能解を直接探索できることである. ただし, 図 4 のようにチェビシェフの不等式による予測区間が過大であることに加えて, 式 (30) のように両側予測区間の上限値で解の実行可能性を判定するため, WCOP の解は式 (31) の機会制約問題の解としては保守的なものとなる.

5.3 制約条件付き最適化問題に対する差分進化

DE は決定変数が実数値をとる最適化問題を対象とした進化計算アルゴリズム (EA: Evolutionary Algorithm) の一種である. DE のアルゴリズムは単純であるが, 様々なテスト問題や現実的な最適化問題において, 優れた探索性能が確認されている [6]. ただし, 基本的に DE を含む EA は制約条件のない最適化問題を解くための最適化アルゴリズムである. そこで, 式 (14) のような制約条件付き最適化問題に EA を適用するため, 多くの EA で制約の取扱い法が長年にわたり研究されてきた [23]. 特に, 近年では DE を対象とした制約の取扱い法の報告が増えている [24].

生存選択で実行可能な個体を優先して選ぶ実行性規則 (Feasibility Rules) [25] は, 妥当な制約の取扱い法である. しかし, 最初に得られた実行可能な個体に集団全体が誘導されて, EA が準最適解に初期収束する恐れがある.

ペナルティ関数法では, 式 (14) の制約条件付き最適化問題に対する解候補 \mathbf{x} を, 式 (32) のようなペナルティ関数 $p(\mathbf{x})$ によって評価する. ペナルティ関数法の課題は, 適切なペナルティ係数 w_m , $m \in \mathbf{I}_M$ の設定である. そこで, ペナルティ係数を徐々に大きくする動的ペナルティ関数法や, 実行可能な個体の割合に基づきペナルティ係数を調整する適応的ペナルティ関数法が考案されている [24].

$$p(\mathbf{x}) = f(\mathbf{x}) + \sum_{m \in \mathbf{I}_M} w_m \max\{g_m(\mathbf{x}), 0\} \quad (32)$$

目的関数値と制約違反量に分けて個体を評価する制約の取扱い法の 1 つに ϵ 制約法 [26] がある. ϵ 制約法ではパラメータ ϵ により制約条件を緩和することで初期収束を防いでいる. さらに, ϵ 制約法を DE に組み込んだ ϵ -DE [27] では, 実行可能解を探索するために降下法も併用する.

上記の代表的な制約の取扱い法のほか, SR (Stochastic Ranking) 法 [28] など, いくつかの技法がある. しかし, NFL (No Free Lunch) 定理 [29] が示唆しているように, 万能な制約の取扱い法は望めない. そのため, DE の集団を 4 分割し, サブ集団ごとに異なる制約の取扱い法 (実行性規則, 適応的ペナルティ法, ϵ 制約法, SR 法) を採用した最適化アルゴリズム (ECHT-DE) [30] などもある.

本論文では標本の節約法に着目するため、DE の特徴を最も活かした制約の取扱い法 [13] を採用した。すなわち、DEB のアルゴリズムの手順 4.3 に示したように、トライアル・ベクトル \mathbf{u} とターゲット・ベクトル \mathbf{x}_i の目的関数値と制約条件を直接比較する。その利点として、本来の DE に新たな制御パラメータやメカニズムをまったく追加する必要がない。欠点としては、実行性規則の一種であるため、前述のように初期収束を招く恐れがある。また、制約条件の数が増えると、すべての制約条件について $\mathbf{x}_i \in P$ に勝る \mathbf{u} を得ることが難しくなり、集団 P の個体が更新されずに探索が停滞する。ただし、これらの欠点は DE の集団サイズを十分に大きくとることで補うことができる。

6. 数値実験

6.1 実験方法

提案手法の有効性をテスト問題と 2 種類の工学設計問題で検証した。2 種類の工学設計問題は様々な EA の性能評価で広く利用されている [31], [32]。各問題は式 (14) のような制約条件付き最適化問題に変形した後、有意水準を $\alpha = 0.05$ として式 (16) の WCOP に拡張した。以降、不確実性を考慮しない式 (14) の最適化問題に対する解をノミナル解と呼び、WCOP の解をロバスト解と呼ぶ。

各 DE は Java 言語で実装した。表 1 に各 DE の制御パラメータを示す。まず、本論文で採用した制約の取扱い法 [13] を組み込んだ DE により、すべてのテスト問題と工学設計問題に対して既知のノミナル最適解 $\mathbf{x}^* \in \mathbb{R}^D$ が得られることを予備実験で確認し、集団サイズを $N_P = 10D$ とした。終了条件は解の評価回数 (標本数) の合計 (FES : Function Evaluations) で指定した。DEB では各 WCOP で式 (21) を満たす十分に大きな標本数 N を求めた。図 1 に示したように係数 κ は標本数 N に依存する。このため、式 (16) の WCOP で各関数の上限値の精度を確保するためには、標本数 N を大きくとる必要がある。式 (12) から $\alpha = 0.05$ では標本数の最小値が $N_{\min} = 21$ となることを勘案し、DEA の初期集団の標本数 N_{ini} を決めた。DEAR では $\hat{\kappa} > \sqrt{1/\alpha} \approx 4.472$ の範囲から $\hat{\kappa}$ を選択した。

6.2 2変数のロバスト最適化問題

式 (33) の 2 変数のテスト問題において、ノミナル最適解とロバスト最適解の位置関係を考察する。また、累積サンプリングにおける「信頼性の緩和」の効果を検証する。

$$\left[\begin{array}{ll} \min & f_1(\mathbf{x}) = x_1^2 + x_2^2 \\ \text{sub. to} & g_1(\mathbf{x}) = -x_1^2 + x_2 + 4 \leq 0 \\ & g_2(\mathbf{x}) = -x_1 + x_2 - 1 \leq 0 \\ & g_3(\mathbf{x}) = x_1 - 2 \leq 0 \\ & g_4(\mathbf{x}) = -x_2 - 4 \leq 0 \\ & -10 \leq x_j \leq 10, j = 1, 2 \end{array} \right. \quad (33)$$

表 1 DE の制御パラメータ

Table 1 Control parameters of DE.

DE	N_P	FES	N	N_{ini}	$\hat{\kappa}$
DEB	$10D$	$2 \times 10^5 D$	200	—	—
DEA	$10D$	$2 \times 10^5 D$	—	21	—
DEAR	$10D$	$2 \times 10^5 D$	—	6	5.0

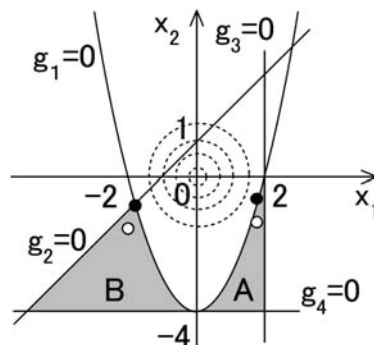


図 5 テスト問題の実行可能領域

Fig. 5 Feasible region of test problem.

表 2 ロバスト解の位置

Table 2 Location of robust solutions.

σ^2	region	DEB	DEA	DEAR
0.01 ²	A	27	20	22
	B	23	30	28
0.02 ²	A	21	9	17
	B	29	41	33
0.03 ²	A	5	3	6
	B	45	47	44
0.04 ²	A	0	0	1
	B	50	50	49
0.05 ²	A	0	0	0
	B	50	50	50

式 (33) のテスト問題の実行可能領域を図 5 に示す。実行可能領域は領域 A と領域 B からなる。破線は目的関数値の等高線である。ノミナル最適解は $\mathbf{x}_A^* = (\sqrt{3.5}, -0.5)$, $f(\mathbf{x}_A^*) = 3.75$ であり、領域 A の境界上にある。また、領域 B の境界上にはノミナル最適解 $\mathbf{x}_B^* \approx (-1.791, -0.791)$, $f(\mathbf{x}_B^*) \approx 3.834$ が存在する。図 5 では各ノミナル解の位置を黒丸「●」で示す。実行可能領域は最適解 \mathbf{x}_A^* の周辺では狭く、準最適解 \mathbf{x}_B^* の周辺では広がっている。

各決定変数 x_j に誤差 $\delta_j \in \mathcal{R}$ を加えて、式 (33) のテスト問題を WCOP に拡張した。誤差 δ_j は平均 0, 分散 σ^2 の正規分布に従う確率変数とした。ロバスト最適解は、誤差の分散 σ^2 が小さいとき、領域 A 内のノミナル最適解 \mathbf{x}_A^* の近傍に位置し、分散 σ^2 が大きくなると、領域 B 内のノミナル最適解 \mathbf{x}_B^* の近傍に移動する。図 5 では予想される各ロバスト最適解の位置を白丸「○」で示す。

誤差の分散 σ^2 を変えて、WCOP に DEB, DEA, DEAR を 50 回ずつ適用して得られた計 50 個のロバスト解が存在した領域を表 2 に示す. 表 2 から, 分散 σ^2 が大きくなると, DEA は狭い領域 A 内のロバスト解を見つけ難くなる. 一方, DEAR は DEB と同程度の数の解を領域 A 内で発見しており「信頼性の緩和」の効果が確認できる.

6.3 標本の節約法の検証

式 (33) のテスト問題のほか, 下記のような標準的なテスト問題 [23] を用いて, 標本の節約法の効果を検証する.

$$\left[\begin{array}{l} \min \quad f_2(\mathbf{x}) = 5.3578547 x_3^2 \\ \quad \quad + 0.8356891 x_1 x_5 \\ \quad \quad + 37.293239 x_1 - 40792.141 \\ \text{sub. to} \quad 0 \leq g_1(\mathbf{x}) \leq 92 \\ \quad \quad 90 \leq g_2(\mathbf{x}) \leq 110 \\ \quad \quad 20 \leq g_3(\mathbf{x}) \leq 25 \\ \quad \quad 78 \leq x_1 \leq 102, 33 \leq x_2 \leq 45 \\ \quad \quad 27 \leq x_j \leq 45, j = 3, 4, 5 \end{array} \right. \quad (34)$$

ただし, $f_2(\mathbf{x}^*) \approx -30665.5$, 各関数は以下のとおりである.

$$\left[\begin{array}{l} g_1(\mathbf{x}) = 85.334407 + 0.0056858 x_2 x_5 \\ \quad \quad + 0.0006262 x_1 x_4 - 0.0022053 x_3 x_5 \\ g_2(\mathbf{x}) = 80.51249 + 0.0071317 x_2 x_5 \\ \quad \quad + 0.0029955 x_1 x_2 + 0.0021813 x_3^2 \\ g_3(\mathbf{x}) = 9.300961 + 0.0047026 x_3 x_5 \\ \quad \quad + 0.0012547 x_1 x_3 + 0.0019085 x_3 x_4 \end{array} \right. \quad (35)$$

$$\left[\begin{array}{l} \min \quad f_3(\mathbf{x}) = (x_1 - 10)^2 + x_3^4 \\ \quad \quad + 5(x_2 - 12)^2 + 3(x_4 - 11)^2 \\ \quad \quad + 10x_5^6 + 7x_6^2 + x_7^4 \\ \quad \quad - 4x_6 x_7 - 10x_6 - 8x_7 \\ \text{sub. to} \quad g_m(\mathbf{x}) \geq 0, m = 1, \dots, 4 \\ \quad \quad -10 \leq x_j \leq 10, j = 1, \dots, 7 \end{array} \right. \quad (36)$$

ただし, $f_3(\mathbf{x}^*) \approx 680.63$, 各関数は以下のとおりである.

$$\left[\begin{array}{l} g_1(\mathbf{x}) = 127 - 2x_1^2 - 3x_2^4 - x_3 - 4x_4^2 - 5x_5 \\ g_2(\mathbf{x}) = 196 - 23x_1 - x_2^2 - 6x_6^2 + 8x_7 \\ g_3(\mathbf{x}) = 282 - 7x_1 - 3x_2 - 10x_3^2 - x_4 + x_5 \\ g_4(\mathbf{x}) = -4x_1^2 - x_2^2 + 3x_1 x_2 - 2x_3^2 \\ \quad \quad - 5x_6 + 11x_7 \end{array} \right. \quad (37)$$

各決定変数 $x_j, j \in I_D$ に誤差 $\delta_j \in \mathcal{R}$ を加え, 式 (33), (34), (36) の各テスト問題を WCOP に拡張し, WCOP1, WCOP2, WCOP3 とする. 誤差 δ_j は平均 0, 分散 σ^2 の正規分布に従う確率変数とした. DE の終了条件が FES で与えられた場合, 1 つの個体の評価に要する標本数を減らすことで, より多くの個体を調べることができ, 最終的に優れたロバスト解が得られることが期待される.

誤差の分散 σ^2 を変えて, 各 WCOP に DEB, DEB-U,

表 3 目的関数値 ($FES = 2 \times 10^5 D, \sigma^2 = 0.01^2$)

Table 3 Objective function value.

DE	WCOP1	WCOP2	WCOP3
DEB	4.155 ↑	-30626.8 ↑	692.82 ↑
DEB-U	4.169 ↑	-30633.2 ↑	690.50 ↑
DEA	4.162 ↑	-30633.1 ↑	690.58 ↑
DEA-U	4.167 ↑	-30633.4 ↑	690.20 ↑
DEAR	4.129	-30634.3 ↑	690.24 ↑
DEAR-U	4.148 ↔	-30634.8	690.07

表 4 調べた個体の総数 ($FES = 2 \times 10^5 D, \sigma^2 = 0.01^2$)

Table 4 Number of examined individuals.

DE	WCOP1	WCOP2	WCOP3
DEB	2000.0	5000.0	7000.0
DEB-U	2790.4	9117.0	15519.0
DEA	5631.6	13862.0	18012.4
DEA-U	6128.8	16490.0	26196.8
DEAR	7026.8	16581.0	21247.8
DEAR-U	7549.2	18864.0	30100.0

表 5 目的関数値 ($FES = 2 \times 10^5 D, \sigma^2 = 0.05^2$)

Table 5 Objective function value.

DE	WCOP1	WCOP2	WCOP3
DEB	6.089 ↑	-30489.3 ↑	731.45 ↑
DEB-U	6.072 ↑	-30498.4 ↑	728.38 ↑
DEA	6.082 ↑	-30497.2 ↑	728.07 ↑
DEA-U	6.084 ↑	-30498.2 ↑	727.08 ↑
DEAR	6.014	-30508.7 ↔	726.19 ↔
DEAR-U	6.016 ↔	-30509.3	725.99

表 6 調べた個体の総数 ($FES = 2 \times 10^5 D, \sigma^2 = 0.05^2$)

Table 6 Number of examined individuals.

DE	WCOP1	WCOP2	WCOP3
DEB	2000.0	5000.0	7000.0
DEB-U	2520.8	7621.0	11851.0
DEA	4444.8	14831.0	17680.6
DEA-U	4653.2	16589.0	21229.6
DEAR	5273.2	18102.0	21012.6
DEAR-U	5436.8	19802.0	24726.8

DEA, DEA-U, DEAR, DEAR-U を 50 回ずつ適用した. 表 1 の FES の場合, すべての WCOP で各 DE の成功率 (実行可能解を発見した割合) は 100%であった.

誤差の分散が $\sigma^2 = 0.01^2$ の場合, 各 DE で得られたロバスト解の目的関数値の平均を表 3 に示す. 太字は WCOP ごとの最良値である. また, 各 DE が探索過程で調べた個体の総数の平均を表 4 に示す. 同様に, $\sigma^2 = 0.05^2$ の場合の目的関数値と個体数の平均を表 5 と表 6 に示す.

目的関数値は WCOP ごとに太字で示した最良値と他の値をウィルコクソン (Wilcoxon) 検定で比較した. 目的関数値の各表において, 記号「↑」と「↗」を付した値は, それぞれ危険率 5% と 1% で最良値と有意な差がある. 一方,

表 7 目的関数値 ($FES = 2 \times 10^3 D$, $\sigma^2 = 0.01^2$)

Table 7 Objective function value.

DE	WCOP1	WCOP2	WCOP3
DEB	13.920 ↑	-28789.2 ↑	1515175.2 ↑
DEB-U	13.920 ↑	-28789.2 ↑	1515175.2 ↑
DEA	—	-29664.6 ↑	1716.9 ↑
DEA-U	6.726 ↑	-30163.4 ↑	1217.9 ↑
DEAR	7.271 ↑	-30130.7 ↑	2001.3 ↑
DEAR-U	4.555	-30456.8	876.2

表 8 調べた個体の総数 ($FES = 2 \times 10^3 D$, $\sigma^2 = 0.01^2$)

Table 8 Number of examined individuals.

DE	WCOP1	WCOP2	WCOP3
DEB	20.0	50.0	70.0
DEB-U	20.0	50.0	70.0
DEA	180.0	450.0	630.0
DEA-U	448.4	902.0	1449.0
DEAR	328.0	848.0	1120.0
DEAR-U	645.6	1626.0	2118.2

「↔」を付した値は最良値と有意な差が認められない。

表 3 と表 5 の目的関数値と検定結果から、誤差の大きさにかかわらず、すべての WCOP で DEAR-U により最良のロバスト解が得られている。ただし、DEAR-U と DEAR で目的関数値に有意な差がなく、U-カットの導入による DEAR の探索性能の向上が認められない場合も含む。

表 4 と表 6 の調べた個体の総数から、すべての DE (DEB, DEA, DEAR) で U-カットの導入による標本数の削減効果が認められる。また、DEB と DEA の個体数の比較から累積サンプリングによる削減効果が、DEA と DEAR の個体数の比較から「信頼性の緩和」による削減効果が確認できる。ここで、すべての WCOP において DEAR-U が調べた個体の総数が最も多く、DEAR-U によって最良のロバスト解が得られた主要な要因であると考えられる。

現実的な最適化問題では個体の評価回数が厳しく制限される場合も想定される。そこで、表 1 の終了条件のみ $FES = 2 \times 10^3 D$ に変えて、上記の実験を再度実施した。 FES を小さくすると、解の質が低下するほか、実行可能解を発見する前に DE の探索が終了することもある。

誤差の分散が $\sigma^2 = 0.01^2$ の場合、各 DE で得られたロバスト解の目的関数値の平均と検定結果を表 7、調べた個体数の平均を表 8、探索の成功率を表 9 に示す。同様に、 $\sigma^2 = 0.05^2$ の場合の目的関数値の平均と検定結果、個体数の平均、探索の成功率を表 10、表 11、表 12 に示す。

表 7 と表 10 の目的関数値と検定結果から、 FES が小さくても、誤差の大きさにかかわらず、すべての WCOP で DEAR-U により最良のロバスト解が得られている。また、表 8 と表 11 から、DEAR-U は限られた数の標本を無駄なく利用し、他の DE よりも多くの個体を調べている。

表 9 と表 12 から、 FES が少ないと、累積サンプリング

表 9 探索の成功率 [%] ($FES = 2 \times 10^3 D$, $\sigma^2 = 0.01^2$)

Table 9 Success rate for search [%].

DE	WCOP1	WCOP2	WCOP3
DEB	40	100	40
DEB-U	40	100	40
DEA	0	100	6
DEA-U	96	100	100
DEAR	100	100	100
DEAR-U	100	100	100

表 10 目的関数値 ($FES = 2 \times 10^3 D$, $\sigma^2 = 0.05^2$)

Table 10 Objective function value.

DE	WCOP1	WCOP2	WCOP3
DEB	16.574 ↑	-28660.3 ↑	1302029.0 ↑
DEB-U	16.574 ↑	-28660.3 ↑	1302029.0 ↑
DEA	—	-29442.6 ↑	—
DEA-U	17.137 ↑	-29856.9 ↑	1788.9 ↑
DEAR	21.272 ↑	-29982.5 ↑	2512.9 ↑
DEAR-U	12.991	-30273.6	1032.5

表 11 調べた個体の総数 ($FES = 2 \times 10^3 D$, $\sigma^2 = 0.05^2$)

Table 11 Number of examined individuals.

DE	WCOP1	WCOP2	WCOP3
DEB	20.0	50.0	70.0
DEB-U	20.0	50.0	70.0
DEA	180.0	450.0	630.0
DEA-U	443.6	760.0	1387.4
DEAR	320.0	848.0	1120.0
DEAR-U	523.2	1470.0	1933.4

表 12 探索の成功率 [%] ($FES = 2 \times 10^3 D$, $\sigma^2 = 0.05^2$)

Table 12 Success rate for search [%].

DE	WCOP1	WCOP2	WCOP3
DEB	32	100	30
DEB-U	32	100	30
DEA	0	100	0
DEA-U	2	100	34
DEAR	4	100	46
DEAR-U	28	100	100

は探索の成功率を著しく低下させる場合があることが分かる。探索の初期段階で関数の上限値が大きく変動し、DEA の手順 5 で上限値が収束する前に探索が終了するためと思われる。しかし、表 9 と表 12 から、累積サンプリングの欠点を「信頼性の緩和」と U-カットが補うことも確認できる。

6.4 圧力容器のロバスト最適設計

図 6 に圧力容器の構造を示す。筒状の容器の両端に半球型の蓋が付いている。式 (38) の圧力容器の構造設計問題 [33] では、材料、加工、溶接のコストを含む目的関数

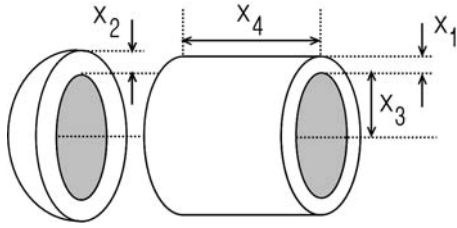


図 6 圧力容器の構造

Fig. 6 Structure of pressure vessel.

表 13 圧力容器の設計問題における決定変数

Table 13 Decision variables for pressure vessel problem.

σ^2	x_1	x_2	x_3	x_4
0.01 ²	0.838	0.444	41.493	185.107
0.03 ²	0.929	0.563	42.350	176.357
0.05 ²	1.079	0.703	46.495	135.481
—	0.778	0.384	40.321	199.980

$f(\mathbf{x})$ が最小となるように、容器の厚さ x_1 、蓋の厚さ x_2 、容器の内径 x_3 、蓋を除く容器の長さ x_4 を決める。

$$\begin{cases} \min & f(\mathbf{x}) = 0.6224 x_1 x_3 x_4 \\ & + 1.7781 x_2 x_3^2 + 19.84 x_1^2 x_3 \\ & + 3.1661 x_1^2 x_4 \\ \text{sub. to} & g_1(\mathbf{x}) = -x_1 + 0.0193 x_3 \leq 0 \\ & g_2(\mathbf{x}) = -x_2 + 0.00954 x_3 \leq 0 \\ & g_3(\mathbf{x}) = -\pi x_3^2 x_4 - \frac{4}{3} \pi x_3^3 \\ & \quad + 1296000 \leq 0 \\ & g_4(\mathbf{x}) = x_4 - 240 \leq 0 \\ & 0.0625 \leq x_j \leq 6.1875, j = 1, 2 \\ & 10 \leq x_j \leq 200, j = 3, 4 \end{cases} \quad (38)$$

各決定変数 x_j に誤差 δ_j を加え、式 (38) の工学設計問題を WCOP に拡張した。誤差 δ_j は平均 0、分散 σ^2 の正規分布に従う確率変数とした。式 (38) の工学設計問題のノミナル解 $\mathbf{x}^* \in \mathbb{R}^4$ [32] と DEAR-U で求めた分散 σ^2 が異なる WCOP のロバスト解 $\mathbf{x}^\circ \in \mathbb{R}^4$ を表 13 に示す。表 13 の再下段がノミナル解 \mathbf{x}^* である。また、各ロバスト解 \mathbf{x}° とノミナル解 \mathbf{x}^* による関数値を表 14 で比較する。

表 13 から、誤差の分散 σ^2 が大きくなるほど、容器の厚さ x_1 と蓋の厚さ x_2 が増して、容器の長さ x_4 が短くなり、ロバスト解 \mathbf{x}° はノミナル解 \mathbf{x}^* から離れてゆくことが分かる。

表 14 から、ノミナル解 \mathbf{x}^* は WCOP の実行可能解ではない。特に、 $G_3^U(\mathbf{x}^*)$ の制約違反量は大きく、誤差の分散 σ^2 に比例して増大する。一方、 $G_4^U(\mathbf{x}) \leq 0$ は誤差の有無にかかわらず不活性な制約条件である。ロバスト解 \mathbf{x}° の目的関数値 $\mathcal{F}^U(\mathbf{x}^\circ)$ も σ^2 に比例して増大する。分散 σ^2 が大きくなると、関数値のバラツキ (標本不偏分散 s^2) も大きくなり、式 (10) の予測区間が広がるためと考えられる。

表 14 圧力容器の設計問題における関数値

Table 14 Function values for pressure vessel problem.

σ^2	0.01 ²	0.03 ²	0.05 ²
$\mathcal{F}^U(\mathbf{x}^*)$	6295.5	7102.3	7774.1
$\mathcal{F}^U(\mathbf{x}^\circ)$	6651.9	8114.8	9972.7
$G_1^U(\mathbf{x}^*)$	0.050	0.149	0.251
$G_1^U(\mathbf{x}^\circ)$	-0.001	-0.003	-0.015
$G_2^U(\mathbf{x}^*)$	0.048	0.143	0.235
$G_2^U(\mathbf{x}^\circ)$	-0.001	-0.023	-0.037
$G_3^U(\mathbf{x}^*)$	3172.3	9311.1	15764.7
$G_3^U(\mathbf{x}^\circ)$	-1041.0	-5157.5	-28625.1
$G_4^U(\mathbf{x}^*)$	-39.97	-39.87	-39.78
$G_4^U(\mathbf{x}^\circ)$	-54.84	-63.49	-104.28

6.5 溶接はりのロバスト最適設計

図 7 に溶接はりの構造を示す。式 (39) の溶接はりの構造設計問題 [34] では、片持ちはりの先端に作用する荷重 ρ によるせん断応力 $\tau(\mathbf{x})$ 、曲げ応力 $h(\mathbf{x})$ 、たわみ $q(\mathbf{x})$ 、座屈応力 $\rho_c(\mathbf{x})$ などに関する制約条件の下で、材料と溶接のコストを含む目的関数 $f(\mathbf{x})$ が最小となるように、溶接部の厚さ x_1 と長さ x_2 、はりの縦幅 x_3 と横幅 x_4 を決める。

$$\begin{cases} \min & f(\mathbf{x}) = 1.10471 x_1^2 x_2 \\ & + 0.04811 x_3 x_4 (L + x_2) \\ \text{sub. to} & g_1(\mathbf{x}) = \tau(\mathbf{x}) - 13600 \leq 0 \\ & g_2(\mathbf{x}) = h(\mathbf{x}) - 3 \times 10^4 \leq 0 \\ & g_3(\mathbf{x}) = x_1 - x_4 \leq 0 \\ & g_4(\mathbf{x}) = 0.125 - x_1 \leq 0 \\ & g_5(\mathbf{x}) = q(\mathbf{x}) - 0.25 \leq 0 \\ & g_6(\mathbf{x}) = \rho - 1000 \rho_c(\mathbf{x}) \leq 0 \\ & 0.1 \leq x_1 \leq 2, j = 1, 4 \\ & 0.1 \leq x_j \leq 10, j = 2, 3 \end{cases} \quad (39)$$

$$\begin{cases} J(\mathbf{x}) = \sqrt{2} x_1 x_2 \left(\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2} \right)^2 \right) \\ R(\mathbf{x}) = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2} \right)^2} \\ \tau_1(\mathbf{x}) = \frac{\rho}{\sqrt{2} x_1 x_2}, \quad q(\mathbf{x}) = \frac{\rho L^3 \times 10^{-5}}{75 x_3^3 x_4} \\ \tau_2(\mathbf{x}) = \rho \left(L + \frac{x_2}{2} \right) \frac{R(\mathbf{x})}{J(\mathbf{x})}, \quad h(\mathbf{x}) = \frac{6 \rho L}{x_3^2 x_4} \\ \tau(\mathbf{x}) = \sqrt{\tau_1(\mathbf{x})^2 + \frac{\tau_1(\mathbf{x}) \tau_2(\mathbf{x}) x_2}{R(\mathbf{x})} + \tau_2(\mathbf{x})^2} \\ \rho_c(\mathbf{x}) = \frac{4013 \sqrt{10} x_3 x_4^3}{L^2} \left(1 - \frac{\sqrt{0.625} x_3}{2L} \right) \end{cases} \quad (40)$$

設計段階で荷重 ρ の正確な値は不明である。そこで、荷重 ρ を式 (15) の θ に相当する不確実な係数とし、式 (39) の工学設計問題を WCOP に拡張した。荷重 ρ は平均 6000 [lb]、分散 σ^2 の正規分布に従う確率変数とした。はりの長さ $L = 14$ [in] と各決定変数の誤差は考えない。式 (39) の工

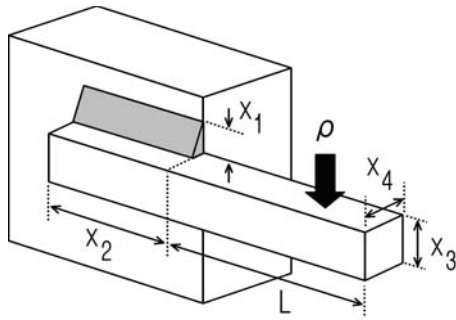


図 7 溶接はりの構造

Fig. 7 Structure of welded beam.

表 15 溶接はりの設計問題における決定変数

Table 15 Decision variables for welded beam problem.

σ^2	x_1	x_2	x_3	x_4
100^2	0.246	6.446	8.495	0.247
300^2	0.253	6.772	8.823	0.254
500^2	0.263	6.340	9.454	0.267
—	0.244	6.217	8.291	0.244

表 16 溶接はりの設計問題における関数値

Table 16 Function values for welded beam problem.

σ^2	100^2	300^2	500^2
$f(\mathbf{x}^*)$	2.380	2.380	2.380
$f(\mathbf{x}^\circ)$	2.502	2.723	2.958
$G_1^U(\mathbf{x}^*)$	1220.0	3619.8	6168.3
$G_1^U(\mathbf{x}^\circ)$	-50.25	-53.36	-55.44
$G_2^U(\mathbf{x}^*)$	2690.2	7983.8	13605.3
$G_2^U(\mathbf{x}^\circ)$	-209.9	-259.9	-3958.3
$g_3(\mathbf{x}^*)$	-6.8E-6	-6.8E-6	-6.8E-6
$g_3(\mathbf{x}^\circ)$	-0.001	-4.664	-0.003
$g_4(\mathbf{x}^*)$	-0.119	-0.119	-0.119
$g_4(\mathbf{x}^\circ)$	-0.121	-0.128	-0.138
$G_5^U(\mathbf{x}^*)$	-0.232	-0.230	-0.227
$G_5^U(\mathbf{x}^\circ)$	-0.234	-0.235	-0.238
$G_6^U(\mathbf{x}^*)$	538.2	1596.9	2721.3
$G_6^U(\mathbf{x}^\circ)$	-13.84	-40.72	-1154.4

学設計問題のノミナル解 $\mathbf{x}^* \in \mathcal{R}^4$ [32] と DEAR-U で求めた分散 σ^2 が異なる WCOP のロバスト解 $\mathbf{x}^\circ \in \mathcal{R}^4$ を表 15 に示す. 表 15 の再下段がノミナル解 \mathbf{x}^* である. また, それぞれの解による関数値を表 16 で比較する. 式 (39) の目的関数 $f(\mathbf{x})$ と制約条件の $g_3(\mathbf{x})$ と $g_4(\mathbf{x})$ に荷重 ρ は影響しないため, これらの関数値は確率変数とはならない.

表 15 から, 荷重 ρ の分散 σ^2 に比例して, すべての決定変数 x_j の値が大きくなり, 溶接はりは大型化することが分かる.

表 16 から, ノミナル解 \mathbf{x}^* は WCOP の実行可能解ではない. また, $G_1^U(\mathbf{x}^*)$, $G_2^U(\mathbf{x}^*)$, $G_6^U(\mathbf{x}^*)$ の制約違反量は大きく, 荷重の分散 σ^2 に比例して増大する. ノミナル解 \mathbf{x}^* の $f(\mathbf{x}^*)$ に荷重 ρ の分散 σ^2 は影響しないが, ロバスト解

\mathbf{x}° の $f(\mathbf{x}^\circ)$ は σ^2 に比例して大きくなる. これは, 表 15 に示した決定変数 x_j の値の増加によるものである.

7. おわりに

本論文では, 最悪状況を考慮した最適化問題 (WCOP) を定式化した. まず, 標本によるチェビシェフの不等式 [10] から, 確率分布が未知である確率変数 (関数値) の予測区間を導出し, その上限値を最悪状況の評価指標とした. また, 予測区間と標本数の関係を明らかにするとともに, 任意の有意水準に対して必要な標本数の最小値を示した.

次に, WCOP の最適解 (ロバスト最適解) を効率的に探索するため, 3 種類の標本の節約法 (累積サンプリング, 信頼性の緩和, U-カット) を採用した DE (DEAR-U) を構築した. また, 提案した「信頼性の緩和」が累積サンプリングを補完することを理論的に解説した. さらに, 3 種類の標本の節約法を組み合わせて, DE が探索の過程で調べる個体数が最も増えること, その結果として優れたロバスト解が得られることを数値実験で確認した.

最後に, 2 種類の現実的な工学設計問題を WCOP に拡張し, DEAR-U で得られたロバスト解を解析することで, 提案したロバスト最適化手法の有用性を実証した.

今後の課題として, チェビシェフの不等式の多次元分布への拡張があげられる [35]. 本論文では関数値の予測区間を独立に求めているが, 複数の関数値を多次元分布の標本ととらえることで, 最悪状況をより厳密に評価できる可能性がある. また, 数多くの制約条件を含む大規模な WCOP にも DEAR-U を有効に適用するため, ϵ 制約法 [26] など, 新たな制約の取扱い法を採用することも考えられる.

参考文献

- [1] Parkinson, A., Sorensen, C. and Pourhassan, N.: A general approach for robust optimal design, *Journal of Mechanical Design*, Vol.115, No.1, pp.74-80 (1993).
- [2] 武田朗子: ロバスト最適化法とその動向, 電気学会論文誌 C, Vol.134, No.6, pp.760-764 (2014).
- [3] Koch, P.N. and Yang, R.-J.: Design for six sigma through robust optimization, *Structural and Multidisciplinary Optimization*, Vol.26, No.3, pp.235-248 (2004).
- [4] Liu, X., Wang, S., Qiu, J., Zhu, J.G., Guo, Y. and Lin, Z.W.: Robust optimization in HTS cable based on design for six sigma, *IEEE Trans. Magnetics*, Vol.44, No.6, pp.978-981 (2008).
- [5] 田川聖治, 藤田翔平: チェビシェフの不等式を用いた予測区間に基づくロバスト最適化, 情報処理学会研究報告, Vol.2016-MPS-107, No.5, pp.1-4, 情報処理学会 (2016).
- [6] Price, K.V., Storn, R.M. and Lampinen, J.A.: *Differential Evolution — A Practical Approach to Global Optimization*, Springer (2005).
- [7] Park, T. and Ryu, K.R.: Accumulative sampling for noisy evolutionary multi-objective optimization, *Proc. GECCO'11*, Dublin, Ireland, pp.793-800, ACM (2011).
- [8] Fieldsend, J.E. and Everson, R.M.: The rolling tide evolutionary algorithm: a multi-objective optimizer for noisy optimization problems, *IEEE Trans. Evolutionary*

- Computation*, Vol.19, No.1, pp.103–117 (2015).
- [9] 田川聖治, 原田翔一: ノイズを含む多目的最適化問題に対する最悪状況の予測に基づく差分進化の適用, 電気学会論文誌 C, Vol.136, No.2, pp.189–198 (2016).
- [10] Saw, J.G., Yang, M.C. and Mo, T.C.: Chebyshev inequality with estimated mean and variance, *The American Statistician*, Vol.38, No.2, pp.130–132 (1984).
- [11] Brest, J., Greiner, S., Bošković, B., Merink, M. and Žumer, V.: Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, *IEEE Trans. Evolutionary Computation*, Vol.10, No.6, pp.646–657 (2006).
- [12] 田川聖治: (解説) 差分進化の基礎と並行プログラミング, システム/制御/情報, Vol.59, No.2, pp.47–52 (2015).
- [13] Lampinen, J.: A constraint handling approach for the differential evolution algorithm, *Proc. CEC'2002*, Hawaii, US, pp.1468–1473, IEEE (2002).
- [14] Ren, Z., Pham, M.T. and Koh, C.S.: Robust global optimization of electromagnetic devices with uncertain design parameters: comparison of the worst case optimization methods and multiobjective optimization approach using gradient index, *IEEE Trans. Magnetics*, Vol.49, No.2, pp.851–859 (2013).
- [15] Tsai, J.-T.: An evolutionary approach for worst-case tolerance design, *Engineering Applications of Artificial Intelligence*, Vol.25, pp.917–925 (2012).
- [16] Jin, Y. and Branke, J.: Evolutionary optimization in uncertain environments – a survey, *IEEE Trans. Evolutionary Computation*, Vol.9, No.3, pp.303–317 (2005).
- [17] Wackerly, D.D., Mendenhall, W. and Scheaffer, R.L.: *Mathematical Statistics with Applications, 7th Edition*, Thomson Learning, Inc. (2008).
- [18] Tagawa, K. and Suenaga, T.: Extended differential evolution algorithm for worst-case value minimization problems, *International Journal of Mathematical Models and Methods in Applied Sciences*, Vol.8, pp.262–272 (2014).
- [19] Prékopa, A.: *Stochastic Programming*, Kluwer Academic Publishers (1995).
- [20] Tempo, R., Calafiore, G. and Dabbene, F.: *Randomized Algorithms for Analysis and Control of Uncertain Systems: With Applications*, Springer (2012).
- [21] Alamo, T., Tempo, R. and Camacho, E.F.: Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems, *IEEE Trans. Automatic Control*, Vol.54, No.11, pp.2545–2559 (2009).
- [22] 藤崎泰正, 和田孝之: (解説) ロバスト凸最適化のためのランダムイズドアルゴリズム, 計測と制御, Vol.50, No.11, pp.950–955 (2011).
- [23] Michalewicz, Z. and Schoenauer, M.: Evolutionary algorithms for constrained parameter optimization problems, *Evolutionary Computation*, Vol.4, No.1, pp.1–32 (1996).
- [24] Montes, E.M. and Coello Coello, C.A.: Constraint-handling in nature-inspired numerical optimization: past, present and future, *Swarm and Evolutionary Computation*, Vol.1, pp.173–194 (2011).
- [25] Deb, K.: An efficient constraint handling method for genetic algorithms, *Computer Methods in Applied Mechanics and Engineering*, Vol.186, pp.311–338 (2000).
- [26] 高濱徹行, 阪井節子: ε 制約遺伝的アルゴリズムによる制約付き最適化, 情報処理学会論文誌, Vol.47, No.6, pp.1861–1871 (2006).
- [27] Takahama, T. and Sakai, S.: Constrained optimization by the ε constrained differential evolution with gradient-based mutation and feasible elites, *Proc. CEC'2006*, Vancouver, BC, Canada, pp.308–315, IEEE (2006).
- [28] Runarsson, T.P. and Yao, X.: Stochastic ranking for constrained evolutionary optimization, *IEEE Trans. Evolutionary Computation*, Vol.4, No.3, pp.284–294 (2000).
- [29] Wolpert, D.H. and Macready, W.G.: No free lunch theorems for optimization, *IEEE Trans. Evolutionary Computation*, Vol.1, No.1, pp.67–82 (1997).
- [30] Mallipeddi, R. and Suganthan, P.N.: Differential evolution with ensemble of constraint handling techniques for solving CEC 2010 benchmark problems, *Proc. CEC'2010*, Barcelona, Spain, pp.1–8, IEEE (2010).
- [31] Coello Coello, C.A. and Montes, E.M.: Constraint-handling in genetic algorithms through the use of dominance-based tournament selection, *Advanced Engineering Informatics*, Vol.16, No.3, pp.193–203 (2002).
- [32] Garg, H.: Solving structural engineering design optimization problems using an artificial bee colony algorithm, *Journal of Industrial and Management Optimization*, Vol.10, No.3, pp.777–794 (2014).
- [33] Kannan, B.K. and Kramer, S.N.: An augmented Lagrange multiple based method for mixed integer discrete continuous optimization and its applications to mechanical design, *Journal of Mechanical Design*, Vol.116, No.2, pp.405–411 (1994).
- [34] Rao, S.S.: *Engineering Optimization, 3rd Edition*, John Wiley and Sons (1996).
- [35] Stellato, B., Parys, B.V. and Goulart, P.J.: Multivariable Chebyshev inequality with estimated mean and variance, Cornell University Library (online), available from (<http://arxiv.org/abs/1509.08398>) (accessed 2016-03-30).



田川 聖治 (正会員)

計測自動制御学会, 進化計算学会, IEEE 各会員.

1991年神戸大学大学院工学研究科修了。1993年神戸大学工学部助手。2005年同助教授。2007年近畿大学理工学部教授となり、現在に至る。進化計算アルゴリズムとその応用に関する研究に従事。博士(工学)。電気学会,



藤田 翔平

2016年近畿大学理工学部情報学科卒業。同年同大学院総合理工学研究科に入学し、現在に至る。ロバスト最適化問題に対する差分進化の適用法に関する研究に従事。進化計算学会会員。