

卒業論文

勾配情報を考慮できる粒子群最適化による制約付き数理最適化問題の解の探索

Chemoinformatics Using Feature Selection and Clustering for
Enzyme Commission Number Prediction in Organic Synthesis

富山県立大学 工学部 情報システム工学科

2120019 柴原壮大

指導教員 奥原 浩之 教授

提出年月: 2025 年 2 月

目 次

圖一覽

表一覽

記号一覧

以下に本論文において用いられる用語と記号の対応表を示す.

用語	記号
LiNGAM における i 番目の観測変数	x_i
LiNGAM における j 番目の観測変数から i 番目の観測変数へのパス係数	b_{ij}
LiNGAM における i 番目の観測変数に対する誤差 (非観測変数)	e_i
主問題における各入力に対する重み	v^T
主問題における各出力に対する重み	u^T
主問題における対象 DMU の評価値	z
CCR モデルにおける DMU _o の入力	x_o
CCR モデルにおける DMU _o の出力	y_o
CCR モデルにおける DMU の入力	X
CCR モデルにおける DMU の出力	Y
双対問題における対象 DMU の評価値	w
入力指向モデルにおける対象 DMU の評価値	θ
入力指向モデルにおける各 DMU に対する重み	λ
出力指向モデルにおける対象 DMU の評価値	η
出力指向モデルにおける各 DMU に対する重み	μ
入力指向モデルにおける対象 DMU の i 番目の入力に対する改善案	\hat{x}_i
入力指向モデルにおける参照集合内の k 番目の DMU の i 番目の入力	x_{ik}
入力指向モデルにおける参照集合内の k 番目の DMU に対する重み	λ
出力指向モデルにおける対象 DMU の j 番目の出力に対する改善案	\hat{y}_j
出力指向モデルにおける参照集合内の k 番目の DMU の j 番目の出力	y_j
出力指向モデルにおける参照集合内の k 番目の DMU に対する重み	μ
提案手法における d 番目の市区町村の i 番目の入力	x_{id}
提案手法における d 番目の市区町村の i 番目の出力	y_{id}
提案手法における d 番目の市区町村に対する重み	λ_d
<i>robust Z-score</i> における正規化後の値	ι
<i>robust Z-score</i> を用いて正規化するデータ集合内の値	x
<i>robust Z-score</i> を用いて正規化するデータ集合	X
<i>robust Z-score</i> を用いて正規化するデータ集合の中央値	$median(x)$
<i>robust Z-score</i> を用いて正規化するデータ集合の正規四分位範囲	$NIQR$
0~1 変換の結果の値	ι'
0~1 変換を行うデータ集合内の値の最大値	$max \iota $

はじめに

§ 1.1 本研究の背景

近年、コンピュータサイエンスの発展は、ハードウェアとソフトウェアの進化により、さまざまな分野で顕著な成果を上げている。特に、処理能力の向上やアルゴリズムの革新によって、大規模な最適化問題に対するアプローチが進展し、実世界での適用範囲が拡大している。これに伴い、複雑で高次元な問題を効率的に解く技術の需要が急速に高まっている。

中でも、大規模なデータ処理や最適化問題は、企業や研究機関において重要な課題となっており、解決策としての最適化技術の有効性が強調されている。たとえば、ソーシャルネットワークサービス（SNS）の普及により、膨大な量のログデータや通信パスの解析が求められ、これらのデータの中から有用な情報を抽出するためには、非常に高度で効率的な最適化技術が必要とされる。さらに、IoT（モノのインターネット）やビッグデータの普及により、処理対象となるデータ量やネットワーク規模が爆発的に増加しており、従来のアルゴリズムでは計算時間やメモリの制約が大きな課題となることが多い。

また、科学技術計算やエンジニアリング分野においても、大規模なシミュレーションやデータ分析が求められるシーンが増えており、これらに対する効率的な最適化手法の開発は、ますます重要となっている。特に、エネルギー消費の最適化や製造業における生産計画、交通システムの最適化など、実際の現場における最適化問題では、多くの制約条件や競合する目標が存在し、解法の選定が非常に難しくなっている。

最新のコンピュータ技術においても、これらの問題を解決するためには膨大な計算リソースと時間が必要であり、従来のアルゴリズムでは計算時間の長期化や収束性の低下が問題となる場合が多い。したがって、計算資源を効率的に活用し、短時間で最適解を見つけるための新たな手法が求められている。

このような背景を踏まえ、複雑で高次元な最適化問題を解決するために、粒子群最適化（PSO）などの群知能に基づくアルゴリズムが注目されており、其有効性は多くの実際の問題において証明されている。しかし、PSOのような進化的手法には、局所的最適解に陥るリスクや、計算の収束速度が遅くなる課題が依然として存在するため、これらの問題を改善するためのさらなる研究が必要とされている。

§ 1.2 本研究の目的

本研究の目的は,, 数ステップでもっとも最適な解が見つかる新しいハイブリッド動的システムを実際の多目的最適化問題に適応する. つまり上下制限条件付き最適化問題に対し直接適用可能な勾配情報を追加した粒子群最適化アルゴリズムを活用し, それを拡張することで, 多目的スケジューリング問題へ組み込みを行う

§ 1.3 本論文の概要

本論文は次のように構成される.

第1章 本研究の背景と目的について説明する. 背景では, 昨今における, 教育におけるデジタルトランスフォーメーションの例について説明している. 目的では, 医療分野に対する, マルチエージェントシミュレータを用いたデジタルトランスフォーメーションについて述べている.

第2章

第3章

第4章

第5章

第6章 本論文における前章までの内容をまとめつつ, 本研究で実現できたことと今後の展望について述べる.

多目的最適化の粒子群最適化

§ 2.1 粒子群最適化アルゴリズム

Swarm Intelligence（群知能）は、鳥や魚、アリのコロニーなどのグループの行動に基づく最適化手法である。この技術の一つである粒子群最適化が開発され、様々な研究に応用されている。しかし、粒子群最適化の収束は根拠がない。本研究では、より良い最適解を求めるための群知能とニューラルネットワークダイナミックスの新しいハイブリッド動的システムを提案した。本節では主な結果として、粒子群最適化と勾配法のメカニズムを理論的にどのように組み合わせるかを示し、今後は提案システムが客観的な環境のグローバルな情報に基づいて補間探索を実現できることを確認する。

粒子群最適化 (Particle Swarm Optimization : PSO) は、群の中の固体（粒子）が持つ最良の情報とそのグループの最適値から過去の探索から考慮した確率的最適化手法であり、ケネディ [1] が社会的行動に基づいて開発した並列進化計算技術である。社会的方法と計算方法の両方を扱う PSO に関する標準的な研究がある [2]。本節では数ステップでもっとも最適な解が見つかる新しいハイブリッド動的システムを提供する。そこで連続 PSO アルゴリズムに勾配法を組み込み、定式化を行う。そこで、提案した手法の有効性を示す。その後、提案した PSO の手法を用いて取得したライフログのクラスタリングを行い、その結果を示す。

PSO は群を成して移動する生物の行動を模範したアルゴリズムである。群をなす生物を粒子としてモデル化し、粒子は最適化問題における候補解を示している。PSO は群の中の粒子がもつ最良の情報とその集団の最適値から過去の探索を考慮し、さらにその集団の各粒子の位置および速度を更新することによって計算される。以下に PSO の解説を示す（図 1 参照）。ここで、PSO の探索模式図及び速度と位置の更新式より、各粒子が持つ最良の情報に向かう力、集団の最適値に向かう力、これまでの進行方向へ向かう力の 3 つのベクトルを合成して速度ベクトルを決定し、それを元に次に移動する位置を決定する [3]。

$$x_p^{k+1} = x_p^k + v_p^{k+1} \quad (2.1)$$

$$v_p^{k+1} = wv_p^k + c_1r_1(x_{pbest} - x_p^k) + c_2r_2(x_{gbest} - x_p^k) \quad (2.2)$$

PSOの更新式

PSOの更新式

$$x_d^{k+1} = x_d^k + v_d^{k+1}$$

$$v_d^{k+1} = \omega v_d^k + c_1 r_1 (x_{db}^k - x_d^k) + c_2 r_2 (x_{gb}^k - x_d^k)$$

記号

x_d^k :位置

v_d^k :速度

w_d^k :運動量

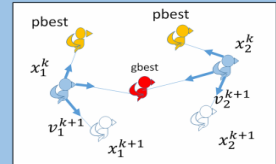
x_{db}^k :p-best(各個体の過去の最良個体)

x_{gb}^k :g-best(集団の中の最良個体)

r_1, r_2 :0から1のパラメータ

c_1, c_2 :調整パラメータ

更新の様子



粒子の探索の様子

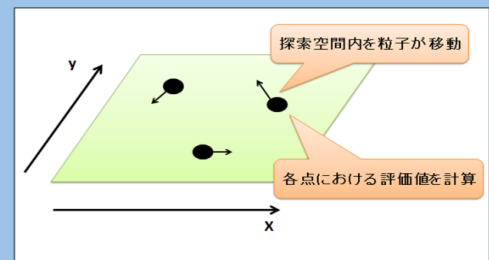


図1 PSOの更新式

ここで x_d^k は粒子の位置, v_d^k は速度, w は慣性係数, c_1 と c_2 は学習係数, r_1 と r_2 はランダムな数値である. PSO の速度ベクトルは, $pbest$ に向かうベクトル, $gbest$ に向かうベクトル, そして過去の進行方向に基づくベクトルの合成によって決定される.

ここで, PSO の探索模式図及び速度と位置の更新式より, p_{best} に向かう $c_1 r_1 (x_k^d - x_d^k)$, g_{best} に向かう $c_2 r_2 (x_k^d - x_d^k)$, これまでの進行方向へ向かう $w v_k^d$ の3つのベクトルを合成して速度ベクトル v_i^{k+1} を決定し, それを元に次に移動する位置 x_d^{k+1} を決定する. PSO の探索式はランダム要素を含み, 同時に最良解情報である p_{best} と g_{best} が探索に伴い変化するという時変性を有している. PSO の探索手順を以下に示す.

PSOの探索手順

STEP1:[群れの初期化]各Particleの位置 x_i と速度 v_i の乱数で初期化する

STEP2:[評価]各Particleに対して最小化したい目的関数値 $f(x_i)$ を評価する

STEP3:[$pbest$ の更新]

Particleのそれまでの最良値 $f(pbest_i)$ と比較する. もし(現在の $f(x_i)$) < $f(pbest_i)$ なら $f(pbest_i) = (現在のf(x_i))$ とし, さらに $pbest_i = x_i$ とする.

STEP4:[$gbest$ の更新]グループ全体の最良値 $f(gbest)$ と比較する. もし(現在の $f(x_i)$) < $f(gbest)$ なら $gbest = x_i$ とする.

STEP5:[速度と位置の更新](1)式で速度を更新し, (2)式で位置を更新する

STEP6:[終了判定]収束条件を満たすまでSTEP2に戻りを繰り返す

図2 PSOの探索手順

このままの形では理論解析が困難であるので、一つの粒子 (Particle) に着目し、一次元の位置 x と速度 v について考え、さらに p_{best} と g_{best} を一つの点に縮約した簡略モデルが提案されている [4]。この簡略モデルは、確定的な線形時不変システムとして表現されており、その安定性を示す。

粒子 i に注目すると速度ベクトル v_{k+1} は以下の式のように変形できる。ステップ幅 φ は二つの一様乱数を足し合わせたものであり、最小値 0、最大値 $C_1 + C_2$ 、平均 $\frac{C_1+C_2}{2}$ の分布に従う。

$$v_{k+1} = wv_k + \varphi(P - x_k) \quad (2.3)$$

ここで、 P は以下のように定義される：

$$P = \frac{\varphi_1 p_{\text{best}}^k + \varphi_2 g_{\text{best}}^k}{\varphi_1 + \varphi_2} \quad (2.4)$$

また、 $\varphi = \varphi_1 + \varphi_2$ 、 $\varphi_1 = C_1 \text{rand}$ 、 $\varphi_2 = C_2 \text{rand}$ とし、さらに $y_k = P - x_k$ とおくと、以下のように表せる。 $\varphi = \frac{C_1+C_2}{2}$ と見なすと固有値 λ は以下のように表せる。

$$\begin{bmatrix} v_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} w & \varphi \\ -w & 1 - \varphi \end{bmatrix} \begin{bmatrix} v_k \\ y_k \end{bmatrix} \quad (2.5)$$

したがって、固有値 λ は次のように表される：

$$\lambda = w + 1 - \varphi \pm \sqrt{(w + 1 - \varphi)^2 - 4w^2}/2 \quad (2.6)$$

よって、 $\lambda = 1$ を境にシステムの特徴が安定・不安定 (収束・発散) に変化することが分かる。

この式より、 λ は (1) 式の定数 w 、 C_1 、 C_2 を変化させることにより変動する。離散線形システムの安定条件より、 λ が 1 を境にシステムの特徴が安定・不安定 (収束・発散) に変化することがわかる。

粒子が漸近安定となる条件は $0 \leq \varphi < 2w + 2$ 、 $0 \leq w < 1$ で与えられる。よって、上記条件を把握することは、探索の軌道とパラメータの関係を把握するうえで重要となる。

ここでは、群れに弱い集中化の特性をもたせるようにパラメータが $w = 0.729$ 、 $C_1 = C_2 = 1.4955$ と設定される。

この解析により群れの収束発散を制御できるため、 V_{max} 設定の困難さが軽減されることになる。

この解析手法の拡張として、多数の粒子での多次元での相互作用を \mathbf{pbest} の更新に取り入れた検討も行われている [5]。

§ 2.2 制約がある場合の PSO

PSO に関する研究は、この手法自体の特性に関するものから応用研究にいたるまで、すでに数多くなされているが、一般に、変数の次元の小さい無制約問題に対しては、更新式

のパラメータを適切に調節すれば、比較的効率よく大域的最適解が探索可能であるのに対して、変数次元の大きな問題に対しては、必ずしも大域的最適化が効率よく求められないことが指摘されている [6]。そこで、変数次元が比較的大きな問題であっても、ある程度狭い有界な領域を制約条件とする問題であるならば、その領域上の大域的最適解を比較的効率よく探索できる制約条件付最適化問題を直接解くことが可能な PSO について解説する。まず、この PSO がもつパラメーターは確定的な値をとるものとした下で、PSO を上記のような力学モデルとして解釈し、まず上下限制約領域内に束縛した PSO モデルを導出する。このために、畳み込み積分を用いて慣性型連続時間系 PSO モデルを表し、それに非線形作用素を施すことによって、上下限制約を全空間に対応づける変数変換を施したモデル、およびそれに対応した内部状態表現モデルを導出する。このモデルにおいて内部状態変数が慣性型連続時間系の PSO モデルで表され、その状態量を出力関数に通すことによって、上下限制約に束縛された挙動を観測することができる。この出力関数は、いわゆるポップフィールド型ニューラルネットワークやカオニューラルネットワークのモデルに用いられるシグモイド関数と同じである。

PSO の更新式を力学系モデルとみなし、その連続化を試みると、

$$\frac{dx_p(t)}{dt} = c \int_0^t e^{-a(t-\tau)} [F_p(x_p(\tau), \tau) + C(x_p(\tau), \tau)] d\tau \quad (2.7)$$

$$\frac{d^2x_p(t)}{dt^2} + a \frac{dx_p(t)}{dt} = c [F_p(x_p(t), t) + C(x_p(t), t)] \quad (2.8)$$

またそれぞれの関数 F_p と C は以下のようになる。

$$F_p(\mathbf{x}_p, k) = c_1 (\mathbf{x}_p(l_p(k)) - \mathbf{x}_p) \quad (2.9)$$

$$C(\mathbf{x}_p, k) = c_2 (\mathbf{x}_Q(k)(l_o(k)) - \mathbf{x}_p) \quad (2.10)$$

ここで、 $l_p(k)$ や $(Q(k), l_o(k))$ は次のように定義される：

$$l_p(k) = \arg \min_l \{E(\mathbf{x}_p(l)) \mid l = 0, 1, \dots, k\} \quad (2.11)$$

$$(Q(k), l_o(k)) = \arg \min_{(q,l)} \{E(\mathbf{x}_q(l)) \mid q = 1, 2, \dots, P, l = 0, 1, \dots, k\} \quad (2.12)$$

また、 $l_p(k)$ は時刻 k までにおいて、個体 p が関数 E の値を最も小さくした時刻を表しており、その状態 $\mathbf{x}_p(l_p(k))$ は PSO における p -best に相当する。さらに、 $(Q(k), l_o(k))$ は時刻 k までにおいて、 P 個の個体の中で関数 E の値を最も小さくした個体番号とその時刻を表しており、その状態 $\mathbf{x}_Q(k)(l_o(k))$ は PSO における g -best に相当する。

ここで、 $l_o(k)$ は個体 p に対応して決まる離散時刻であるため、 $Q(k)$ と $l_o(k)$ との間には次の関係が成り立つ：

$$Q(k) = \arg \min_q \{E(\mathbf{x}_q(l_q(k))) \mid q = 1, 2, \dots, P\} \quad (2.13)$$

$$l_o(k) = l_{Q(k)}(k) \quad (2.14)$$

また、2 階微分方程式で表される連続時間系モデルの状態変数表現を

$$u_p(t) = x_p(t), \quad v_p(t) = \frac{du_p(t)}{dt} + au_p(t) \quad (2.1)$$

といて導入すると、離散時間系に対応した連続系の内部状態表現モデルは、

$$\frac{du_p(t)}{dt} = -au_p(t) + v_p(t) \quad (2.15)$$

$$\frac{dv_p(t)}{dt} = c[F_p(u_p(t), t) + C(u_p(t), t)] \quad (2.16)$$

解説したままのモデルでは無制約なので、制約条件に対応したモデルである上下限制約連続時間 PSO モデルを以下の式に示す。上下限制約付最適化問題

$$\min E(x) \quad (2.16)$$

$$\text{subj: } x_i \leq q_i; \quad i = 1, \dots, n \quad (2.17)$$

を直接解くために、この上下限制約領域内に問題の変数を変換して無制約化した新たな変数空間に無制約 PSO モデルを適用した「変数変換モデル」を導入する。非線形変数変換モデルを作成するために、

$$x_i = f_i(y_i) = \frac{q_i + p_i \exp(-y_i)}{1 + \exp(-y_i)} \quad (2.18)$$

とおく。この変換式を制約条件付き問題に代入して変数 x を消去すると、

$$\min E(f(y)) \quad (2.19)$$

を得ることができる。よって式 (2.8) に対応させると、

$$\frac{d^2 y_p(t)}{dt^2} + a \frac{dy_p(t)}{dt} = c[F_p(y_p(t), t) + C(y_p(t), t)] \quad (2.20)$$

またそれぞれの関数は以下ようになる。

$$F_p(y_p, t) = c_1(y_p(T_p(t)) - y_p) \quad (2.21)$$

$$C_p(y_p, t) = c_2(y_Q(t)(T_o(t) - y_p)) \quad (2.22)$$

次にプログラムへの実装を考えた時に、連続式のままではプログラムに実装することが難しいので、オイラー法を用いて連続式を離散化し非線形変数変換モデルの離散化 PSO を作成する。それぞれに対応する式を以下に示す。

$$\mathbf{u}_p(k+1) = (1 - a\Delta T)\mathbf{u}_p(k) + \Delta T\mathbf{v}_p(k) \quad (2.23)$$

$$\mathbf{v}_p(k+1) = \mathbf{v}_p(k) + c\Delta T [\mathbf{F}_p(\mathbf{u}_p(k); k) + \mathbf{C}(\mathbf{u}_p(k); k)] \quad (2.24)$$

$$\mathbf{F}_p(k; k) = c_1 (\mathbf{u}_p(\mathbf{l}_p(k)) - \mathbf{u}_p(k)) \quad (2.25)$$

$$\mathbf{C}_p(k; k) = c_2 (\mathbf{u}_Q(k)(\mathbf{l}_o(k)) - \mathbf{u}_p(k)) \quad (2.26)$$

$$\mathbf{l}_p(k) = \arg \min (E(\mathbf{x}_p(l)) \mid l = 0, \dots, k) \quad (2.27)$$

$$(\mathbf{Q}(k), \mathbf{l}_o(k)) = \arg \min (E(\mathbf{x}_q(l)) \mid q = 1, 2, \dots, P; l = 0, 1, \dots, k) \quad (2.28)$$

$$\mathbf{x}_p(i)(k) = f_i(\mathbf{u}_p(i)(k)) = \frac{q_i + p_i e^{-\mathbf{u}_p(i)(k)}}{1 + e^{-\mathbf{u}_p(i)(k)}}; \quad i = 1, \dots, n \quad (2.29)$$

PSO では、関数 F_p や C の中に含まれる係数 c_1, c_2 に対して一定区間内の一様乱数により揺らぎを与えて局所的に不安定化させているが、(2.23) 式のモデルにおいて ΔT による分岐特性により不安定化や安定化を調節する新しいモデルを構築することができる。

すなわち、従来の PSO では、**gbest** 情報を探索点群全体で共有し、さらに p -best, g -best への移流の重みに一様乱数を用いた揺らぎを与えることにより局所的最適解からの離脱を期待するのに対して、本提案手法では、 p -best, g -best へ移流する構造を残した上で、さらに比較的大きな ΔT に対する有界領域内全体に渡るカオス的な走査によって、より大域的な探索を期待するのが特徴である。

§ 2.3 多目的最適化のパレート解

近年、多目的ダムや多目的ホールなど、“多目的”という言葉がいろいろと使われるようになってきた。これは一つのシステムあるいは機構をいくつかの異なった目的に対して機能させるという意味であり、この言葉が用いられる背景には、社会的ニーズの多様化および資源の効率的利用などの要求があると考えられる。すなわち、単一の目的関数を用いた最適化問題でカバーできないような多目的最適化問題の重要性が高まってきているのである。

数理計画法や最適制御理論は広い意味で効率の良さを求めるための数学的手段であるが、ほとんどの場合、解空間内で一つの目的関数を最小（または最大）にする最適点を決定することが論じられている。多目的最適化問題に対しても、二つ以上の目的関数の和をとって一つの目的関数にする重みパラメータ法や、一つの目的関数のみを残し、これ以外の目的関数を制約条件に置き換える ϵ 制約法など、従来の数理的手法を応用しようとする方法論がいくつか提案されている [7~10]。多目的最適化の本質は複数の目的関数間でのトレードオフをいかにバランスさせるかという点にあり、従来の方法論では不十分である場合が多い。

一方、多目的最適化の理論について見ると、目的関数間でのトレードオフをバランスさせ得る解に関連して、“パレート最適性”が重要な概念として挙げられている [8]。一般に、このパレート最適性を満足する解（パレート最適解）は複数個あり、いかにして二つの解（選好解）を選び出すかが問題となる。従来、この選好解を決定（多目的意思決定）する過程においては、パラメータを対話的に変更しながら、重みパラメータ法や ϵ 制約法を繰り返

し適用する SWT (Surrogate Worth Trade-off) 法などの対話的手法が提案されている [7]. また, 意思決定者の“あいまいな”選好をファジィ (fuzzy) 理論を用いて表現し, 数理的な手法を適用しようとする報告もなされている [11-12]. しかしながら, パレート最適解のような解候補をあらかじめ集合として求め, これらを提示することができれば, より効率的かつ適切に多目的意思決定を行うことが期待される.

多目的最適化問題

問題 A (多目的最適化問題): F を \mathbb{R}^n の閉集合とする

$$F \triangleq \{x \mid g_i(x) \leq 0, i = 1, \dots, m\} \quad (2.30)$$

とするとき, $x \in \mathbb{R}^n$ の p 個の関数

$$f_k(x), k = 1, \dots, p \quad (2.31)$$

を, $x \in F$ の範囲でなるべく小さくせよ.

この多目的最適化問題では, 一般にすべての目的関数 $f_k(x), k = 1, \dots, p$ を同時に最小にすることはできない. むしろ, これらの間にトレードオフの関係があることが問題の本質である. したがって, 目的関数間での協調を図って各目的関数をできるだけ小さくするようにしなければならない.

パレート最適解

まず, p 次元定数ベクトル a, b 間の不等関係を,

$$a \leq b \iff a_i \leq b_i \ (i = 1, \dots, p) \quad (2.32)$$

または

$$a < b \iff a_i < b_i \ (i = 1, \dots, p) \quad (2.33)$$

で定義する. ただし, a_i および b_i は, それぞれ a および b の第 i 要素である. ここで,

$$f(x) \triangleq (f_1(x), \dots, f_p(x)) \quad (2.34)$$

とすると, 多目的最適化問題における解の優越関係は次のように定義される.

● 定義 1

$x_1, x_2 \in F$ とする.

- a. $f(x_1) \leq f(x_2)$ のとき, x_1 は x_2 に優越すると言う.
- b. $f(x_1) < f(x_2)$ のとき, x_1 は x_2 に強い意味で優越すると言う.

もし x_1 が x_2 に優越しているならば, x_1 のほうが x_2 より良い解である. したがって, ほかのいかなる解にも優越されない解を選ぶことが合理的な方法であると言える.

● 定義 2 (パレート最適解)

$x_0 \in F$ とする.

- x_0 に強い意味で優越する $x \in F$ が存在しないとき, x_0 を弱パレート最適解という.
- x_0 に優越する $x \in F$ が存在しないとき, x_0 を (強) パレート最適解という.
- 任意の $x \in F$ について $f(x) > f(x_0)$ が成り立つとき, x_0 を (完全) 最適解という.

目的関数が二つ ($p = 2$) の場合のパレート最適解の例を図に示す. 図中, 太線がパレート最適解を, 太い破線が弱パレート最適解をそれぞれ示している.

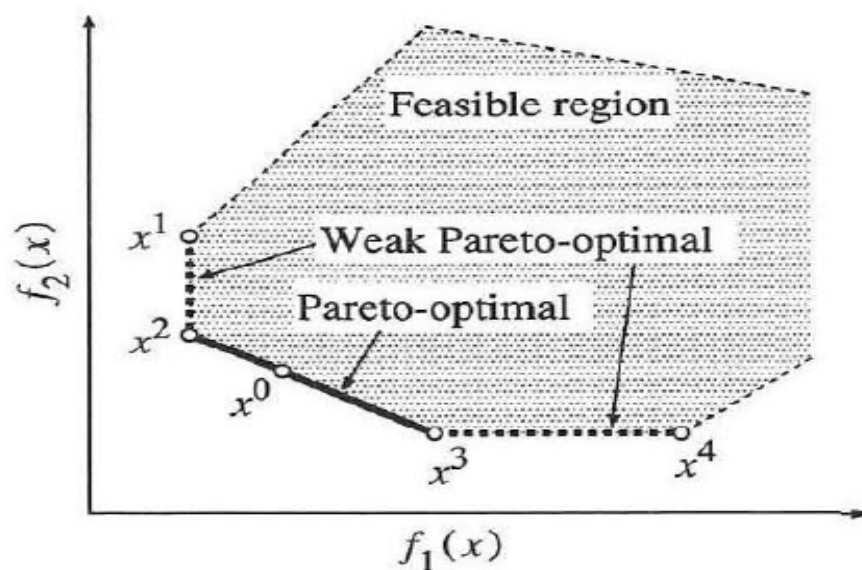


図3 パレート解

定義により, 最適解があればそれはパレート最適解であり, 最適解が存在する時は, それ以外のパレート最適解は存在しない. したがって, パレート最適解は多目的最適化問題に対する最も合理的な解 (の集合) であると言える.

多目的最適化問題

§ 3.1 多目的最適化問題の具体例

代表的な多目的最適化問題として多目的ナップザック問題と多目的巡回セールスマン問題を説明する。

多目的ナップザック問題

多目的ナップザック問題は、要素の集合

$$E = \{e_1, e_2, \dots, e_N\}, \quad (3.1)$$

要素の体積の集合

$$V = \{v_1, v_2, \dots, v_N\}, \quad (3.2)$$

および要素の価値の集合

$$C^{(1)} = \{c_1^{(1)}, c_2^{(1)}, \dots, c_N^{(1)}\}, \dots, C^{(p)} = \{c_1^{(p)}, c_2^{(p)}, \dots, c_N^{(p)}\}$$

が与えられた場合、次の式 (3.3) および (3.4) を満たす要素の部分集合 $E' \subseteq E$ を求める問題として定式化される。

$$\text{maximize} \quad \begin{cases} f_1(x) = \sum_{i=1}^N c_1^{(i)} x_i \\ \vdots \\ f_p(x) = \sum_{i=1}^N c_p^{(i)} x_i \end{cases} \quad (3.3)$$

$$\text{subject to} \quad \sum_{i=1}^N v_i x_i \leq K \quad (3.4)$$

ここで、 K はナップザックの体積サイズであり、変数 x_i は次のように定義される：

$$x_i = \begin{cases} 1 & \text{if } e_i \in E' \\ 0 & \text{if } e_i \notin E' \end{cases} \quad (3.5)$$

図 ?? に、多目的ナップザック問題を簡略に説明したものを示す。

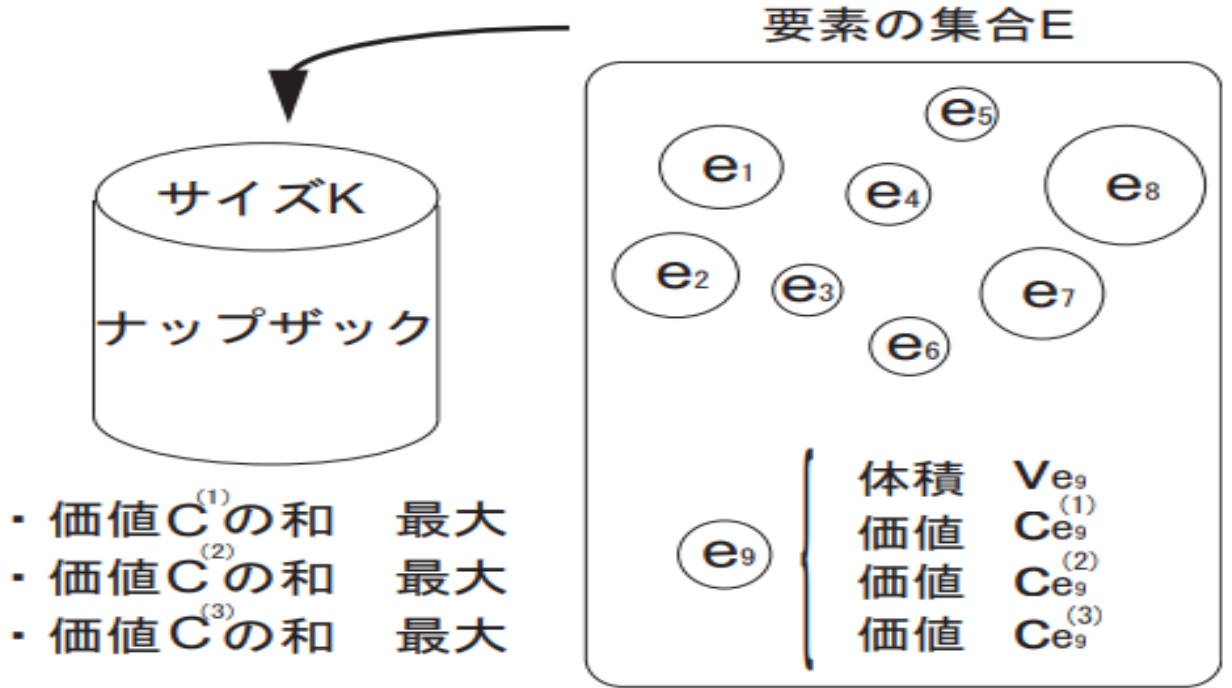


図4 多目的ナップザック問題

図4を用いて多目的ナップザック問題の説明を行う．同図の円柱はナップザックで，その中に詰め込むことのできる要素の総体積を表すサイズは， K で表される．また，円で示した a から i はナップザックに詰め込むべき要素を表しており，各要素には属性として体積と複数の価値が存在する．

多目的ナップザック問題はこれらの要素をある組み合わせでナップザックに詰め込むとき，その体積の和がサイズ K を越えないようにした上で，それぞれの価値の和が最大となるような要素の組み合わせを求める多目的最適化問題といえることができる．

従来よく知られている単目的ナップザック問題と異なる点は，1つの要素に対し2つ以上の価値が存在することである．

多目的巡回セールスマン問題

多目的巡回セールスマン問題は，都市の集合 $U = \{1, 2, \dots, N\}$ とし，都市 (i, j) 間のコストを $c_{ij}^{(1)}, c_{ij}^{(2)}, \dots, c_{ij}^{(p)}$ とすると， N 個の都市すべてをちょうど1度ずつ巡って出発した都市に戻ってくる巡回路の中から，通過した道のコストの総和を最小にする問題とし，以下のように定式化される．

$$\text{minimize} \quad \begin{cases} f_1(x) = \sum_{i \in U} \sum_{j \in U} c_{ij}^{(1)} x_{ij} \\ f_2(x) = \sum_{i \in U} \sum_{j \in U} c_{ij}^{(2)} x_{ij} \\ \vdots \\ f_p(x) = \sum_{i \in U} \sum_{j \in U} c_{ij}^{(p)} x_{ij} \end{cases} \quad (3.6)$$

$$\text{subject to } \begin{cases} \sum_{j \in U} x_{ij} = 1, & \forall i \in U \\ \sum_{i \in U} x_{ij} = 1, & \forall j \in U \\ \sum_{j \in U \setminus V} x_{ij} \geq 1, & \forall V \subset U, V \neq \emptyset, V \neq U \\ x_{ij} \in \{0, 1\}, & \forall i, j \in U \end{cases} \quad (3.7)$$

ここで、 $c_{ii}^{(q)} = +\infty$ ($q = 1, 2, \dots, p$) とした。

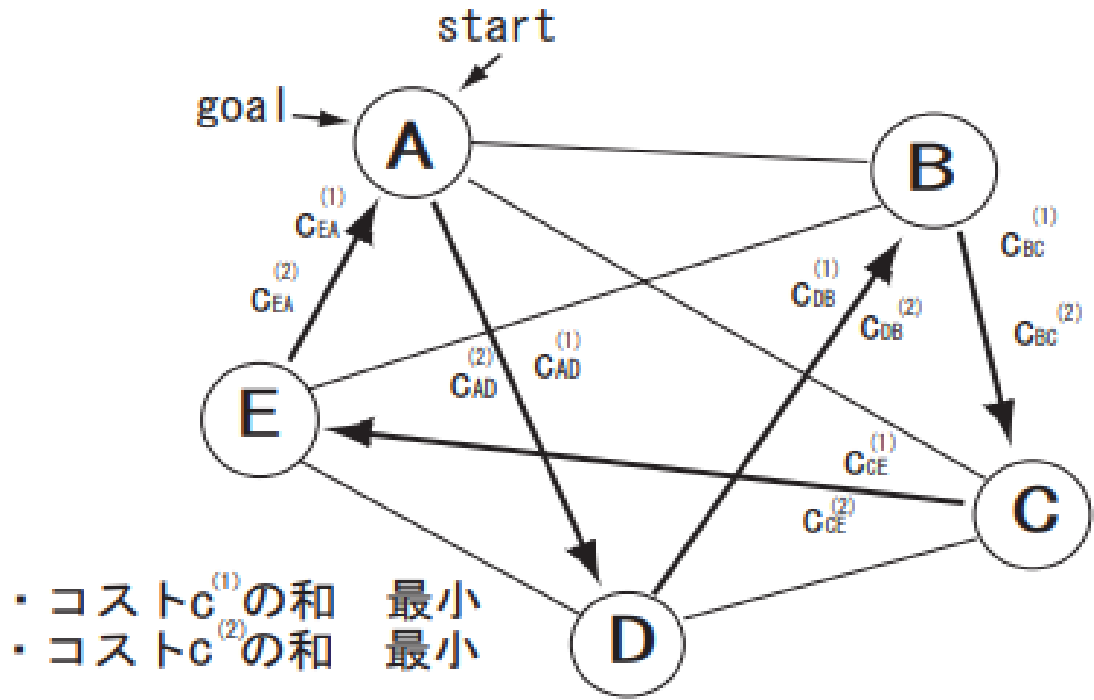


図5 多目的巡回セールスマン問題

図5に多目的巡回セールスマン問題を簡略に説明したものを示す。同図において円で示す都市 A から E は都市 2 を表しており、任意の 2 都市間には複数のコストが存在するものとする。

多目的巡回セールスマン問題は、ある都市を出発し、すべての都市を 1 回ずつまわって最初の都市に戻ってくるような巡回路のうち、その巡回路の複数のコストのそれぞれが最小となるような巡回路を求める多目的最適化問題であるといえる。

同図では、都市 A を出発して再び都市 A に戻ってくるような巡回路の例を示した。従来よく知られている巡回セールスマン問題と異なる点は、任意の 2 都市間に 2 つ以上のコストが定義されていることである。

§ 3.2 実践手法とベンチマークの関係

これまで、工学、医療、経済など多くの実問題において多目的最適化アルゴリズムが優れた性能を示すことが報告されている [13]。一方で、目的数が 4 以上の多数目的最適化問

題 [14,15], 膨大な決定変数を扱う大規模最適化問題 [16], 複雑な多数の制約を持つ最適化問題 [17-21] といった従来の多目的最適化アルゴリズムが苦手とする問題が存在することも報告されている. そのため, 近年の進化計算の分野では, このような困難な特徴を含む実問題を想定した多目的最適化アルゴリズムの開発が高い関心を集めており, 多数目的最適化問題を指向した NSGA-III[22] や大規模最適化問題を指向した WOF[23] のように, 特定の問題領域において高い性能を発揮するアルゴリズムが提案されている. これらの多目的最適化アルゴリズムの性能評価で主に用いられるのがベンチマーク問題である. 特に, 近年の激しい多目的最適化アルゴリズムの開発競争では, DTLZ[24] といった一部の有名なベンチマーク問題群による多目的最適化アルゴリズムの評価が繰り返し行われている. しかし, 現実世界には多様な多目的最適化問題が存在するにも拘らず, これらのベンチマーク問題自体の特徴調査はあまり行われていない. そこで本章では, 1990 年代から 2010 年代までに提案された既存のベンチマーク問題と既存の多目的最適化アルゴリズムについて解説する.

1990 年代

古典的な多目的最適化アルゴリズムの共通の特徴は, パレート優越関係に基づく個体評価と多様性維持である. パレート優越関係では, 以下のような M 目的最小化問題に対して, (3.9) の関係が成り立つとき解 \mathbf{a} は解 \mathbf{b} を優越すると定義される.

$$\text{Minimize } f_1(x), f_2(x), \dots, f_M(x) \quad \text{subject to } x \in X, \quad (3.8)$$

ここで, $f(x)$ は M 次元目的ベクトル, $f_i(x)$ は最小化される i 番目の目的関数, x は決定変数ベクトル, X は決定変数空間内での実行可能領域を示す.

$$\forall i, \quad f_i(\mathbf{b}) \geq f_i(\mathbf{a}) \text{ and } \exists i, \quad f_i(\mathbf{b}) > f_i(\mathbf{a}) \quad 3.9 \quad (3.1)$$

すなわち, 解 \mathbf{a} は, いずれの目的においても解 \mathbf{b} に劣っておらず, 最低でも一つの目的について解 \mathbf{b} より優れている. このとき, 解 \mathbf{a} は解 \mathbf{b} より明らかに優れている. 一方で, 二つの解の間に上記の関係が成立しない場合, それらは非劣とよばれる.

古典的な多目的最適化アルゴリズムでは, パレート優越関係によりパレートフロントへ解集合を収束させる. しかし, エリート保存戦略が使用されていないため, パレートフロントへの収束が安定しない. ここでは, 1990 年代に頻繁に使用されたベンチマーク問題の特性が, これらの非エリート保存戦略型 EMO アルゴリズムの性質と密接に関連していることを示す. 2つの例を紹介する

• Schaffer

Schaffer のテスト問題は以下の二つの目的関数を最小化する問題である.

$$\text{Minimize } f_1(x) = x^2 \quad \text{and} \quad f_2(x) = (x - 2)^2, \quad (3.10)$$

$$\text{subject to} \quad -20 \leq x \leq 20. \quad (3.11)$$

このテスト問題は、単一の決定変数 x を用いて二つの目的関数の算出を行う単純なベンチマーク問題である。NPGA や NSGA の提案論文など、1990 年代に提案された多目的最適化アルゴリズムの探索性能を評価するために頻繁に使用された。

Schaffer のテスト問題ではパレートフロントを発見するための強い収束性能は必要なく、パレートフロント上に多様な解を分布することが重要となる。このような要素は古典的な非エリート保存戦略型多目的最適化アルゴリズムの特性と一致する。

• F3

F3 は次のように定式化される制約付き 2 目的最小化問題である [25]。

$$\text{Minimize } f_1(x_1, x_2) = x_1^2 - 2x_1 + x_2^2, \quad f_2(x_1, x_2) = 9x_1^2 - (x_1 - 1)^2, \quad (3.12)$$

$$\text{subject to } -20 \leq x_1 \leq 20, \quad -20 \leq x_2 \leq 20, \quad (3.13)$$

$$x_1^2 + x_2^2 \leq 25, \quad 3x_1 - 10 \leq 0, \quad x_1 - x_2 + 2 \leq 0.$$

2000 年代

2000 年頃に提案された SPEA や NSGA-II などのエリート保存戦略を採用した多目的最適化アルゴリズムは、古典的な多目的最適化アルゴリズムに比べ、パレートフロントへの高い収束性を持つことが知られている。これらのアルゴリズムの性能評価に用いられたベンチマーク問題にはパレートフロントへの収束が難しい問題が多く含まれている。ここで、これらのテスト問題の代表例として ZDT テストセットがある。

ZDT テストセットは ZDT1-ZDT6 の 6 種の 2 目的最小化問題から成るテストセットである。各テスト問題は凸状や不連続なパレートフロント、局所最適な疑似パレートフロントの存在といった異なる特徴を持つ。また、決定変数の数 N を自由に変更することが可能であるという特徴を持つ。NSGA-II の提案論文では、決定変数がバイナリ型の ZDT5 を除いた五つのテスト問題がアルゴリズムの性能評価に用いられた。ここでは、ZDT1, ZDT2 について調査を行う。それぞれの問題は以下のように定式化される。

[ZDT1]

Minimize

$$f_1(x) = x_1$$

$$f_2(x) = g(x) \left(1 - \frac{x_1}{g(x)} \right)$$

where

$$g(x) = 1 + \frac{9}{N} \sum_{i=2}^N x_i, \quad (3.15)$$

subject to

$$0 \leq x_i \leq 1, \quad \text{for } i = 1, 2, \dots, N. \quad (3.16)$$

[ZDT2]

Minimize

$$\begin{aligned} f_1(x) &= x_1 \\ f_2(x) &= g(x) \left(1 - \frac{x_1}{g(x)} \right) \end{aligned}$$

where

$$g(x) = 1 + \frac{9}{N} \sum_{i=2}^N x_i, \quad (3.17)$$

subject to

$$0 \leq x_i \leq 1, \quad \text{for } i = 1, 2, \dots, N. \quad (3.18)$$

2010 年代

近年、目的数が 4 以上の多数目的最適化問題で、パレート優越関係ベースの多目的最適化アルゴリズムの探索性能が低下することが報告されている。そのため、目的数を任意に指定可能なテスト問題フレームワークが頻繁にアルゴリズムの性能評価に用いられる。これらは目的数にスケラブルなテスト問題と呼ばれる。ここでは、スケラブルなテスト問題である DTLZ テストセットについて調査する。

DTLZ テストセット

DTLZ テストセットは、目的数を自由に指定可能な 7 個のテスト問題 (DTLZ1-DTLZ7) によって構成される [?]. これらのテスト問題は、パレートの形状と探索領域を別々に設計する *Bottom-Up* アプローチ と呼ばれる方法で作成される。ここでは、DTLZ1 を例として調査を行う。M 目的の DTLZ1 は以下のように定式化される。

$$\begin{aligned} \min \left\{ \begin{aligned} f_1(\mathbf{x}) &= (1 + g(\mathbf{x})) \prod_{h=1}^{M-1} \cos(x_h) \prod_{h=M}^d \cos^2(x_h) \\ f_2(\mathbf{x}) &= (1 + g(\mathbf{x})) \prod_{h=1}^{M-2} \cos(x_h) \prod_{h=M-1}^d \cos^2(x_h) \\ &\vdots \\ f_M(\mathbf{x}) &= (1 + g(\mathbf{x})) \cos(x_1) \prod_{h=2}^M \cos^2(x_h) \end{aligned} \right\} \end{aligned}$$

ここで、 $g(\mathbf{x})$ は以下のように定義される：

$$g(\mathbf{x}) = 100 \sum_{i=M}^d (x_i - 0.5)^2 - \prod_{i=M}^d \cos(20\pi(x_i - 0.5))$$

制約条件：

$$0 \leq x_h \leq 1, \quad \text{for } h = 1, 2, \dots, M-1$$

$$0 \leq x_j \leq 1, \quad \text{for } j = 1, 2, \dots, d$$

Bottom-Up アプローチ では、決定変数は位置変数および距離変数と呼ばれる二つのグループに分類される．ここでは、位置変数を x_h 、距離変数を x_d と表す．それぞれの決定変数の数は位置変数が $M-1$ 、距離変数が N_d となる．パレートフロントは位置変数を使用して設計され、探索領域は距離変数を用いて設計される．近年、DTLZ テストセットは、HypE, NSGA-III, MOEA/DD の提案論文や、多数目的最適化に関する調査などで、多目的最適化アルゴリズムの性能評価に頻繁に使用されている．

§ 3.3 種々のパレート解の導出法

パレート解の導出法の中から、いくつかの手法を取り上げ、その概要を紹介する．

重みパラメータ法

目的関数の加重和を最小にする問題は次のように表される：

$$\min_{\mathbf{x}} \sum_{k=1}^p \omega_k f_k(\mathbf{x}) \quad (3.21)$$

ここで、 $\mathbf{x} \in \mathcal{F}$ であり、重みパラメータ ω_k は次の条件を満たす：

$$\omega_k \geq 0, \quad \sum_{k=1}^p \omega_k = 1 \quad (3.22)$$

この問題を解くと、もし $f_k(\mathbf{x})$ と $g(\mathbf{x})$ が凸関数であれば、パレート最適解の一つが得られる．

ε 制約法

ε 制約法では、一つの目的関数 $f_l(\mathbf{x})$ のみを最小化し、他の目的関数 $f_k(\mathbf{x})$ ($k \neq l$) を定数 ϵ_k で制約したスカラー最小化問題として定式化する．

$$\min_{\mathbf{x}} f_l(\mathbf{x}) \quad (3.33)$$

制約条件は次のように表される．

$$\mathbf{x} \in \mathcal{F} \cap \delta, \quad \delta = \{\mathbf{x} \mid f_k(\mathbf{x}) \leq \epsilon_k, k = 1, \dots, l-1, l+1, \dots, p\}$$

このようにして、 ϵ_k を変化させることにより、パレート最適解の一つを求めることができる。一般に、パレート最適解は唯一解ではなく、集合（以下、パレート最適集合と呼ぶ）を形成する。

計算の効率化と遺伝アルゴリズムの利用

上述の重みパラメータ法および ε 制約法の方法では、パラメータを変えながら繰り返し問題を解くことにより、パレート最適集合を求めることができるが、これには膨大な計算時間が必要である。そこで、進化的アルゴリズムを適用することで、パレート最適集合をより効率的に求める方法を考える。進化的アルゴリズムでは、集団（解候補の集合）を用いて探索が進められるため、パレート最適集合を直接的かつ効率的に求めることが期待される。

進化的アルゴリズム

進化的アルゴリズムは、問題の特定の構造に依存せず、広範囲な探索を通じて近似解を得る方法として、最適化問題の解法に広く用いられている。ここでは、代表的な進化的アルゴリズムであるシミュレーティッド・アニーリング (Simulated Annealing: SA)、遺伝的アルゴリズム (Genetic Algorithm: GA)、免疫的アルゴリズム (Immune Algorithms: IA)、および粒子群最適化 (Particle Swarm Optimization: PSO) について解説する。

シミュレーティッド・アニーリング (Simulated Annealing: SA)

シミュレーティッド・アニーリング (SA) は、物理学におけるアニーリング（焼きなまし）の過程を模倣した最適化手法であり、エネルギーが最小となる状態（最適解）を探索することを目的としている。SA では、解候補間のエネルギー差に基づいて新しい解の受け入れ確率が決まる。具体的には、現在の解を x 、新しい解を x' 、それらのエネルギー差を $\Delta E = E(x') - E(x)$ としたとき、受け入れ確率 $P(\text{accept})$ は次のように表される。

$$P(\text{accept}) = \begin{cases} 1 & \text{if } \Delta E \leq 0, \\ \exp\left(-\frac{\Delta E}{T}\right) & \text{if } \Delta E > 0. \end{cases} \quad (3.2)$$

ここで、 T は温度を表し、温度は徐々に低下させる。温度の更新は以下のように行われる。

$$T_{k+1} = \alpha T_k, \quad (3.34)$$

ここで、 α は冷却率であり、 $0 < \alpha < 1$ の範囲にある。この冷却率により、温度の低下速度が決まる。

遺伝的アルゴリズム (Genetic Algorithm: GA)

遺伝的アルゴリズム (GA) は、自然選択と遺伝の原理を模倣した進化的アルゴリズムであり、個体群 (解の集まり) を進化させることによって最適解を探索する。GA の主要な操作は、選択、交叉 (交配)、および突然変異である。

選択

個体はその適応度 $f(x)$ に基づいて選択される。選択方法には、ルーレット選択やトーナメント選択などが使用される。

交叉

交叉は、2つの親個体 x_1 と x_2 から新しい個体 x' を生成するために行われる。交叉操作は通常、遺伝子 (解の各部分) の一部を交換する形式で実行される。例えば、一点交叉では次のように交叉が行われる。

$$x' = \text{crossover}(x_1, x_2). \quad (3.35)$$

交叉によって得られる子個体 x' は、親の情報を組み合わせた新しい解候補となる。

突然変異

突然変異は、個体の遺伝子をランダムに変更する操作である。この操作により、探索空間の多様性を保ちつつ、新たな解を探索することが可能となる。遺伝子 x_i を突然変異させる操作は次のように表される。

$$x'_i = \text{mutate}(x_i). \quad (3.36)$$

免疫的アルゴリズム (Immune Algorithms: IA)

免疫的アルゴリズム (IA) は、生物の免疫システムを模倣して、最適化問題を解く方法である。IA では、解を「抗体」として表現し、その適応度を評価することで最適解を探索する。

抗体と適応度

解は「抗体」と呼ばれ、その適応度は「抗体の品質」として評価される。抗体の適応度 $A(x)$ は、解 x の適応度関数 $f(x)$ に基づいて次のように定義される。

$$A(x) = f(x). \quad (3.37)$$

免疫の選択と記憶

免疫システムは「記憶細胞」を持ち、これにより良い解を記憶し、後の探索で再利用する。適応度が高い解は選ばれる確率が高くなり、最適解に向かって進化していく。

交叉と突然変異

免疫的アルゴリズムでは、GA と同様に交叉と突然変異が使用される。交叉は解の組み合わせを、突然変異は解の変化を導き、探索の多様性を維持する。

粒子群最適化 (Particle Swarm Optimization: PSO)

粒子群最適化 (PSO) は、群れの個々の粒子が協調的に最適解を探索する手法である。各粒子は位置 x_i と速度 v_i を持ち、その位置と速度は次の式で更新される。

粒子の位置と速度の更新

粒子の位置・速度は次のように更新される。

$$x_p^{k+1} = x_p^k + v_p^{k+1} \quad (3.38)$$

$$v_p^{k+1} = wv_p^k + c_1r_1(x_{pbest} - x_p^k) + c_2r_2(x_{gbest} - x_p^k) \quad (3.39)$$

ここで、 w は慣性重み（過去の速度の影響）、 c_1 と c_2 は学習係数（個体のベスト解と全体のベスト解への影響）、 r_1 と r_2 はランダムな数（ $0 \leq r_1, r_2 \leq 1$ ）、 p_i は粒子 i の個人ベスト位置、 g は全体のベスト位置である。この更新式により、粒子は局所最適解に引き寄せられつつ、全体の最適解に向かって進化していく。

これらのアルゴリズムは、複雑な最適化問題を解決するための基盤を提供しており、実世界のさまざまなシナリオに適応可能である。この中でも、PSO はそのシンプルな構造と高い収束速度から注目されている。さらに、他の最適化手法（例えば、遺伝的アルゴリズムや進化戦略）と組み合わせやすく、ハイブリッド手法を構築することで、さらなる性能向上が期待できる。

これにより、局所解に陥りやすいといったデメリットの改善もしやすいため、PSO は多目的最適化の分野において非常に魅力的な選択肢となっている。

提案手法

§ 4.1 適応問題

§ 4.2 MOPSO

適切なフィードバックを行うには、データの分析が必要になるため、収集されたデータの傾向と理解度の可視化が重要になる。そこで、それぞれの方法について説明する。

データの傾向を可視化する方法としては、ソートによるブロック表示法というものがある。まず、横軸を受験者、縦軸を問題とし、右から点数の高い順として並べる。正解を白色とし、不正解の場合は、それぞれの選択肢ごとに色を決め、その色とする。このままでは、全体の傾向をつかむことは困難であるため、ソートを行い、全体のデータの傾向を見やすくするというものである。□

理解度を可視化する方法としては、

§ 4.3 多目的勾配MOPSO(問題への適応?)

本研究で提案するシステムの概要について説明する。初めに FlexSim を用いて、処方箋問題を提示するシステムを作る。このシステムを用いて、問題を解いてもらうことによって、正誤のデータを取得し、CSV として蓄積する。そのデータを Python を用いて適切に処理することによって明らかになった、回答の特徴をもとに、適切に回答者にフィードバックを行い、学習の効率化を図るというものである。

数値実験並びに考察

§ 5.1 数値実験の概要

§ 5.2 実験結果と考察

おわりに

謝辞

本研究を遂行するにあたり，多大なご指導と終始懇切丁寧なご鞭撻を賜った富山県立大学工学部電子・情報工学科情報基盤工学講座の奥原浩之教授，António Oliveira Nzinga René 講師に深甚な謝意を表します．また，システム開発および数値実験にあたり，ご助力いただいた富山県立大学電子・情報工学科3年生の島部達哉氏に感謝の意を表します．最後になりましたが，多大な協力をしていただいた研究室の同輩諸氏に感謝致します．

2022 年 2 月

長瀬 永遠

参考文献

- [1] J. Kennedy and R.C. Eberhart, “Particle swarm optimization,” *IEEE Conf. on Neural Networks*, vol. IV, Piscataway, NJ, pp. 1942-1948, 1995.
- [2] J. Kennedy, R.C. Eberhart, and Y. Shi, *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, CA, pp. 1942-1948, 2001.
- [3] 石亀敦司, 安田恵一郎, “群れの知能: Particle Swarm Optimization,” 知能と情報 (日本知能情報ファジィ学会誌), vol. 20, no. 6, pp. 829-839, 2008.
- [4] J. Kennedy, R. C. Eberhart, Y. Shi, *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, CA, pp. 1942-1948, 2001.
- [5] T. M. Blackwell, “Particle Swarm and Population Diversity,” *Soft Computing*, vol. 9, pp. 793-802, 2005.
- [6] 岩崎信弘・安田恵一郎・井出東, “Particle Swarm Optimization の高次元問題への適用に関する検討,” 電学研資, 産業計測制御研究会, IIC-04-47~66, pp. 87-92, 2004.
- [7] 市川惇信 編, 多目的意思決定の理論と方法, 計測自動制御学会, 1980.
- [8] 西川, 三宮, 茨木, 最適化, 4 章, 岩波書店, 1982.
- [9] 伊理, 今野 編, 数理計画法の応用〈理論編〉, 5 章, 産業図書, 1982.
- [10] 坂和正敏, 非線形システムの最適化〈一目的から多目的へ〉, 7 章, 森北出版, 1986.
- [11] M. Sakawa, *Fuzzy Sets and Interactive Multiobjective Optimization*, Plenum Press, 1993.
- [12] 坂和, 乾口, 砂田, 澤田, 改良型遺伝的アルゴリズムによるファジィ多目的組合せ最適化, 日本ファジィ学会誌, 6-1, 177/186, 1994.
- [13] C. A. C. Coello and G. B. Lamont, *Applications of Multi-objective Evolutionary Algorithms*, vol. 1, World Scientific, 2004.
- [14] R. C. Purshouse and P. J. Fleming, “Evolutionary many-objective optimisation: An exploratory analysis,” In *Proc. of 2003 IEEE Congress on Evolutionary Computation*, pp. 2066-2073, 2003.
- [15] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, “Evolutionary many-objective optimization: A short review,” In *Proc. of 2008 IEEE Congress on Evolutionary Computation*, pp. 2419-2426, 2008.
- [16] M. Sakawa, “Optimal reliability-design of a series-parallel system by a large-scale multiobjective optimization method,” *IEEE Trans. on Reliability*, vol. 30, no. 2, pp. 173-174, 1981.

- [17] R. R. de Lucena, B. S. L. P. de Lima, B. P. Jacob, and D. M. Rocha, “Optimization of pipeline routes using an AIS/adaptive penalty method,” In *Proc. of Eighth International Conference on Engineering Computational Technology*, Paper 66, 2012.
- [18] M. Liao, Y. Zhou, Y. Su, Z. Lian, and H. Jiang, “Dynamic analysis and multi-objective optimization of an offshore drilling tube system with pipe-in-pipe structure,” *Applied Ocean Research*, vol. 75, pp. 85–99, 2018.
- [19] G. K. W. Kenway and J. R. R. A. Martins, “Multipoint high-fidelity aerostructural optimization of a transport aircraft configuration,” *Journal of Aircraft*, vol. 51, no. 1, pp. 144–160, 2014.
- [20] K. S. Zhang, Z. H. Han, Z. J. Gao, and Y. Wang, “Constraint aggregation for large number of constraints in wing surrogate-based optimization,” *Structural and Multidisciplinary Optimization*, 18 pages, 2018.
- [21] T. Kohira, H. Kemmotsu, A. Oyama, and T. Tatsukawa, “Proposal of benchmark problem based on real-world car structure design optimization,” In *Companion of the Genetic and Evolutionary Computation Conference (GECCO) 2018*, pp. 183–184, 2018.
- [22] K. Deb and H. Jain, “An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part I: Solving problems with box constraints,” *IEEE Trans. on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.
- [23] H. Zille, H. Ishibuchi, S. Mostaghim, and Y. Nojima, “A framework for large-scale multiobjective optimization based on problem transformation,” *IEEE Trans. on Evolutionary Computation*, vol. 22, no. 2, pp. 260–275, 2018.
- [24] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, “Scalable multi-objective optimization test problems,” In *Proc. of 2002 IEEE Congress on Evolutionary Computation*, pp. 825–830, 2002.
- [25] N. Srinivas and K. Deb, “Multiobjective optimization using nondominated sorting in genetic algorithms,” *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1995.
- [26] 坪井利樹, 西田佳史, 持丸正明, 河内まき子, 山中龍宏, 溝口博, “身体地図情報システム”, 日本知能情報フレンジイ学会誌, Vol. 20, No. 2, pp. 155-163, 2008.
- [27] 杉原豪, 塚井誠人, “統計的因果探索による社会基盤整備のストック効果の検証”, 土木学会論文集 D3 (土木計画学), Vol. 75, no.6, pp. 583-589, 2020.
- [28] Dentsu Digital Tech Blog, “Google Colab で統計的因果探索手法 LiNGAM を動かしてみた”, 閲覧日 2022-02-08, https://note.com/dd_techblog/n/nc8302f55c775.

- [29] 藤井秀幸, 傅靖, 小林里佳子, ”データ包絡分析を用いたふるさと納税の戦略提案-K市のふるさと納税への適用事例-”, 日本経営工学会論文誌, Vol. 71, No. 4, pp. 149-172, 2021.
- [30] 刀根薫, ”包絡分析法 DEA”, 日本フuzzy学会誌, Vol. 8, No. 1, pp. 11-14, 1996.
- [31] 金成賢作, 篠原正明, ”DEA における入力指向と出力指向の比較 (その 1) ”, 日本大学生産工学部第 42 回学術講演会, 2009.
- [32] 日本オペレーション・リサーチ, ”第 4 章 包絡分析-入力と出力と”, 閲覧日 2022-02-08, <http://www2.econ.tohoku.ac.jp/ksuzuki/teaching/2006/ch4>.
- [33] pork_steak, ”folium 事始め”, 閲覧日 2022-02-08, https://qiita.com/pork_steak/items/f551fa09794831100faa.
- [34] 保母敏行ほか, ”日本分析学会における標準物質の開発”, 日本分析化学会誌, vol. 57, No. 6, pp. 363-392, 2008.
- [35] 射水市役所, ”総合戦略-射水市”, 閲覧日 2022-02-08, <https://www.city.imizu.toyama.jp/appupload/EDIT/054/054185>.
- [36] 射水市役所, ”共通課題-射水市”, 閲覧日 2022-02-08, <https://www.city.imizu.toyama.jp/appupload/EDIT/024/024383>.

