

サロゲート型進化計算におけるモデルの推定精度が探索性能に与える影響の分析

埴 裕貴[†], 原田 智広^{††}, 三浦 幸也[†]

東京都立大学システムデザイン学部[†], 埼玉大学大学院理工学研究科^{††}

1 はじめに

特定の制約条件下において目的関数 (評価関数) を最大化または最小化することを最適化という。最適化問題は一般的に以下の数式で表される。

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && g_i(x) \leq 0 \quad (i = 1, \dots, n) \\ & && h_j(x) = 0 \quad (j = 1, \dots, p) \end{aligned}$$

ここで, x は設計変数, f は目的関数, g は不等式制約関数, h は等式制約関数を表し, n と p は各制約関数の数を示す。

最適化問題の解法の一つとして自然界の進化のメカニズムを模倣して最適解を探索する最適化手法の一つである進化的アルゴリズム (Evolutionary Algorithm: EA)⁹⁾ がある。EA は, 特に高次元空間や複雑な制約条件を持つ問題に対して優れた性能を発揮する。しかし, 目的関数の計算コストが高い場合, 最適化の過程で多くの評価を行うEAの計算時間が膨大になる課題がある。この課題を解決するために, 高コストな解評価を機械学習を用いて低コストな近似モデル (サロゲートモデル) で置き換えることで計算時間を短縮するサロゲート型EA (Surrogate-assisted EA: SAEA)^{2, 3, 4)} が注目されている。

SAEAの探索性能はサロゲートモデルの推定精度に大きく影響を受ける¹⁵⁾。具体的には, サロゲートモデルの推定精度が高いとより正確な評価値の予測が可能であり, 実評価関数の使用回数が抑えられ, 実行時間の削減につながる。一方, サロゲートモデルの精度が高いことによ

て局所最適解に早期収束する可能性がある。この場合, 逆に精度が低いサロゲートモデルによる探索は評価の低い解の探索機会を増加し, 局所解を脱する可能性が生じる。

本論文では, SAEAにおいてサロゲートモデルの推定精度が探索性能に与える影響を分析する。具体的には, 推定精度を調整可能なサロゲートモデルを実際の目的関数を使って擬似的に構築し, 精度の異なるサロゲートモデルを用いるSAEAの探索性能を比較する。サロゲートモデルとして, 本論文はSAEAの一般的な探索構造である事前選択モデルと個体ベースモデルの2種類を扱う。

2 サロゲートモデルの分類

過去数十年にわたり, 多くの異なるサロゲートモデルを使用したSAEAが提案されてきた。これらはTongらの分類¹⁴⁾ によれば主に二つの大きなカテゴリ, 事前選択モデル (PS: Pre-selection model) と個体ベースモデル (IB: Individual-based model) に分類される。これらのカテゴリのうち, PSは絶対評価値モデルとランクモデル, そして分類ベースモデルに分けられ, IBは絶対評価値モデルとランクモデルに分けられる。

2.1 事前選択モデルSAEA

PSは生成した子個体を実評価するかをサロゲートモデルを用いて事前に決定するモデルである。絶対評価値モデルを用いるPSでは子個体候補を複数作成し, それらの評価値をサロゲートモデルを使用して予測し, その中で優れたものを子個体として事前選択する。ランクモデルでは子個体の候補を複数作成したのち, サロゲートを用いて子個体候補の順位を推定し, ランクの高いものを子個体として事前選択する。最後に, 分類ベースモデルでは親個体を基準とした優劣を学習し, 生成された子個体と親個体のどちらが優れているかを予測する。親個体よりも子個体が優れていると予測されれば子個体を実評価し, その実評価が親より優れるならば, 親と置き換え, そう

Analysis of the impact of model accuracy on search performance in surrogate-assisted evolutionary algorithms

[†] Yuki Hanawa (hanawa-yuki@ed.tmu.ac.jp)

^{††} Tomohiro Harada (tharada@mail.saitama-u.ac.jp)

[†] Yukiya Miura (miura@tmu.ac.jp)

Faculty of System Design, Tokyo Metropolitan University,

(†)

Graduate School of Science and Engineering, Saitama University (††)

Algorithm 1 PS-CM¹⁴⁾

```
1: Create the initial sample of  $5 \times d$  individuals
   using Latin Hypercube Sampling (LHS) 8)
2: Evaluate all individuals in the sample
3: Set  $FE = 5 \times d$ 
4: Select the best  $N$  individuals from the sample
   for the initial population
5: while  $FE < maxFE$  do
6:   Perform crossover and mutation operators to
   generate offspring
7:   for each individual do
8:     Find its parent as the reference individual
9:     Build classification-based model
10:    Predict the label of offspring
11:    if predictive label is positive then
12:      Evaluate the offspring with real func-
      tion
13:       $FE = FE + 1$ 
14:      if  $offspring\_fitness < parent\_fitness$  then
15:        Replace parent with offspring
16:      end if
17:    end if
18:  end for
19: end while
```

でないなら親個体をそのまま次世代に引き継ぐ。PSを用いるSAEAとして、制約付き最適化におけるSACCDE¹⁶⁾やCHDE+ELDR⁵⁾、多目的最適化におけるKTA2¹¹⁾やMCEA/D¹²⁾などがある。

Algorithm 1に本研究で扱う分類モデルを用いるPS (PS with classification model: PS-CM)の疑似コードを示す。ここで、 d は設計変数の次元数、 N は母集団サイズ、 $maxFE$ は最大評価回数を表す。PS-CMでは、交叉と突然変異によって生成された子個体が親個体よりも優れているかをサロゲートモデルで推測し、優れていると推測されれば実評価関数で評価する。一方、劣っていると推測されれば、子個体を実評価せずに棄却する。

2.2 個体ベースモデルSAEA

IBは、親個体から子個体を生成し、各世代において生成された全ての子個体から評価が優れると推定される一部の個体を実評価し、親個体と全ての子個体からそれらの評価に基づいて次世代の集団を選ぶモデルである。絶対評価値モ

Algorithm 2 IB-AFM¹⁴⁾

```
1: Create the initial sample of  $5 \times d$  individuals
   using Latin Hypercube Sampling (LHS)
2: Evaluate all individuals in the sample
3: Set  $FE = 5 \times d$ 
4: Select the best  $N$  individuals from the sample
   for the initial population
5: while  $FE < maxFE$  do
6:   Perform crossover and mutation operators to
   generate offspring
7:   Predict the fitness of the offspring
8:   Evaluate  $p_{sm} \times N$  best predicted individuals
   with real function
9:    $FE = FE + p_{sm} \times N$ 
10:  Select  $N$  best individuals from parents and
   offspring for the next generation
11: end while
```

デルを用いるIBでは新たに作成された子個体の評価値を予測し、予測評価値の優れた個体を実評価したあと、次世代に残す個体を決定する。一方、ランクモデルを用いるIBは子個体集団の順位を予測し、予測ランクと学習データの相関が一定以上になるまで高ランクの個体から実評価を行う。一定以上の相関が得られれば、ランク上位の個体を次世代の母集団として選択する。IBを用いるSAEAとして、GPEME⁶⁾やVESAEA¹³⁾、RFMOISR¹⁰⁾などがある

Algorithm 2に本研究で扱う絶対評価値モデルを用いるIB (IB with absolute fitness model: IB-AFM)の疑似コードを示す。IB-AFMでは、交叉と突然変異によって生成された子個体の評価値をサロゲートモデルを用いて推測する。そして、上位 p_{sm} に当たる個体を実評価関数で再評価し、親個体と子個体の中から上位のものを次世代集団として選択する。このとき、実評価していない個体は予測評価値に基づいて比較する。

3 実験

サロゲートモデルの精度による影響を調査するため、実評価値を用いて疑似的に表現したサロゲートモデルを使用する実験を行う。CEC2015のベンチマーク関数¹⁾を対象に、サロゲートモデルを使用しない場合と異なるサロゲートモデルを用いるSAEAの探索性能を比較する。

3.1 実験設定

本実験で調査するサロゲートモデルはPS-CMとIB-AFMを想定し、これらのサロゲートモデルの推定精度を調整可能な形で疑似的に再現する。これらのアルゴリズムは、参考論文¹⁴⁾で紹介されたMATLABのソースコードを基に実装する。また、サロゲートモデルを使用しないモデル (No surrogate model: NoS) を比較する。母集団サイズは40、各問題で独立した20試行を行い、それらの平均をとる。

交叉には、Extended intermediate crossover⁷⁾を使用する。まず母集団から交叉率に従って選択された2つの親個体に対し、 γ から $1+\gamma$ の間の任意の値 α_i を選び、子個体の i 番目の変数 t_i を下記の式で決定する。

$$t_i = \alpha_i v_i + (1 - \alpha_i) w_i \quad (1)$$

ここで v_i と w_i は親個体の i 番目の設計変数を表す。本研究で、 γ は参考文献¹⁴⁾に則り、0.4とする。突然変異は、設計変数の上限と下限の間の一様突然変異を用いる。交叉率 pc 、突然変異率 pm は参考文献に則り、 $\{pc, pm\} = \{0.7, 0.3\}$ とする。本実験では、最大評価回数を2000回に限定し、その中で探索過程で発見された最良解の推移を分析する。

3.2 実装手法

本論文ではPS-CMとIB-AFMにおいてサロゲートモデルの精度を調整可能にするために、目的関数を直接使用してサロゲートモデルを疑似的に再現する。なお、いずれのモデルでも疑似サロゲートに用いる実評価はアルゴリズム中の実評価回数 (FE) には含めない。

PS-CMで使用する疑似サロゲートモデルの疑似コードをAlgorithm 3に示す。Algorithm 3では、サロゲートモデルを学習する代わりに生成された子個体を実評価し、親個体と比較する。このとき、サロゲートの推定精度 sp を設定し、一定の確率 $(1 - sp)$ で優劣の判定 ($label$) を反転させることでサロゲートモデルの精度を疑似的に再現する。この疑似サロゲートをAlgorithm 1の9-10行目で使用する。

一方、IB-AFMで使用する疑似サロゲートモデルをAlgorithm 4に示す。Algorithm 4では、生成されたすべての子個体を実評価し、親個体と子個体を合わせた集団 R を評価値順に並び替える。そのうち $round((1 - sp) \times |R|)$ 個の個体をラ

Algorithm 3 Pseudo surrogate model used in PS-CM

- 1: Evaluate the offspring
 - 2: $label \leftarrow (\text{offspring.fitness} < \text{parent.fitness})$
 - 3: **if** $\text{rand}(0,1) > sp$ **then**
 - 4: Flip $label$
 - 5: **end if**
 - 6: **return** $label$
-

Algorithm 4 Pseudo surrogate model used in IB-AFM

- 1: Evaluate the offspring
 - 2: Merge parent and offspring as R
 - 3: Sort R by their fitness
 - 4: $N_r = \text{round}((1 - sp) \times |R|)$
 - 5: Select N_r random individuals from R as P_{rand}
 - 6: Assign the remaining individuals as P_{rem}
 - 7: Randomly insert individuals in P_{rand} into random positions of P_{rem} as P_{new}
 - 8: **return** P_{new}
-

ンダムに取り除き、それらをランダムな位置に戻す。これにより、 $sp \times |R|$ 個の個体は正しい順序でそれ以外の個体の順位がランダムに入れ替わった母集団を生成する。この疑似サロゲートをAlgorithm 2の7-10行目で使用する。

サロゲートを使用しないEA (NoS) の疑似コードをAlgorithm 5に示す。PS-CMの場合、生成されたそれぞれの子個体が親個体と比較して優れているなら親個体と入れ替え、次世代の母集団として使用する。IB-AFMの場合、親個体の交叉と突然変異で生成された子個体を全て実評価し、親個体と子個体の中から上位のものを次世代の親個体として選択する。サロゲートを用いないことを除いて、交叉、突然変異、生存選択はPS-CM、およびIB-AFMと同一である。

3.3 テスト問題

本実験で使用するCEC2015のテスト問題の種類と関数形状最適値、次元数を表1に示す。目的関数 $f1$, $f2$ が単峰性、 $f4$, $f8$ が多峰性、 $f13$, $f15$ が単峰性と多峰性の関数を組み合わせた形状を持つ。本実験では、設計変数の次元数を10と30に設定する。

Algorithm 5 No Surrogate model¹⁴⁾

```
1: Create the initial sample of  $5 \times d$  individuals
   using Latin Hypercube Sampling (LHS)
2: Evaluate all individuals in the sample
3: Set  $FE = 5 \times d$ 
4: Select the best  $N$  individuals from the sample
   for the initial population
5: while  $FE < \text{maxFE}$  do
6:   Perform crossover and mutation operators to
   generate offspring
7:   /***PS-CM***/
8:   for each individual do
9:     Find its parent as the reference individual
10:    Evaluate the offspring
11:    if  $\text{offspring\_fitness} < \text{parent\_fitness}$ 
   then
12:      Replace parent with offspring
13:    end if
14:  end for
15:
16:  /***IB-AFM***/
17:  Evaluate all individuals
18:  Sort based on fitness values
19:  Select the top  $N$  individuals for the next gen-
   eration
20: end while
```

4 結果・考察

図1～図4に各実験における目的関数値の推移を示す。ここで、横軸は評価回数、縦軸は最適値との差を示しており、0に近づくほど最適解を探索できていることを表す。色の違いはNoSを含むサロゲートモデルの精度の違いを表す。

4.1 PS-CM

図1の10次元の問題をPS-CMで探索した際の目的関数値の推移を比較する。

単峰性関数の結果を示す図1aと図1bでは、評価回数の少ない段階 ($f1$ は104回, $f2$ は131回まで)では精度の違いによる探索性能の傾向は確認できないが、評価回数が増えると推定精度が高いほど探索性能が高いことがわかる。NoSと比較をすると、 $f1$ ではサロゲートモデルの精度が0.5の時と同程度の推移を示している。 $f2$ では精度が1.0の時には明確に優れた性能を示す。一方、精度0.9以下ではNoSとの差はみられず、NoSは

Table 1: ベンチマークの詳細と最適値¹⁴⁾

| 問題 | 関数形状 | 最適値 | 次元数 |
|-------|------|------|--------|
| $f1$ | 単峰性 | 100 | 10, 30 |
| $f2$ | | 200 | |
| $f4$ | 多峰性 | 400 | |
| $f8$ | | 800 | |
| $f13$ | 混合 | 1300 | |
| $f15$ | | 1500 | |

精度0.5と同等の性能を示す。ただし、評価回数が増えるとより目的関数値の小さな解を探索できており、精度が0.5であっても評価回数が1200回を超えるとNoSよりも優れた探索ができている。また、評価回数が100回程度の探索初期には全ての精度のサロゲートモデル使用時でNoSよりも優れた探索性能を示す。

次に図1cと図1dの多峰性問題では、どちらも推定精度が高くなるほど探索性能が向上している。NoSは精度1.0の結果より劣るが、評価回数が2000回のとき $f4$ と、 $f8$ で精度0.5よりも優れた探索性能を有する。一方、多峰性の場合にも評価回数100回程度までは精度が0.5のサロゲートモデルの使用であってもNoSを上回る探索を実現している。しかし、評価回数が増加するごとに精度0.5のサロゲートモデルはNoSより探索性能が同等または劣る結果となる。

最後に複合関数の結果である図1eと図1fではどちらも精度1.0のサロゲートモデルを使用したときに最も優れた解を探索できているが、 $f13$ ではサロゲートモデルの精度に関わらず、NoSとの探索能力の差は小さい。一方、 $f15$ では、精度0.6以上であればNoSよりも優れた探索ができおり、精度が上がるほど、その探索速度は向上している。また、評価回数が少ない場合はサロゲートモデルを使用する場合の方が良い探索性能を示している。

以上の結果からいずれの目的関数であっても評価回数が100回程度の少ない場合にはサロゲートモデルを使用した探索が常に優れるが、より多く2000回の評価を行う場合は精度が0.5のサロゲートモデルの探索性能はサロゲートモデルを使用しないEAと同等または、劣る結果となった。精度が0.5の場合、親より優れるはずの子個体を棄却してしまうことと、劣るはずの個体を実評

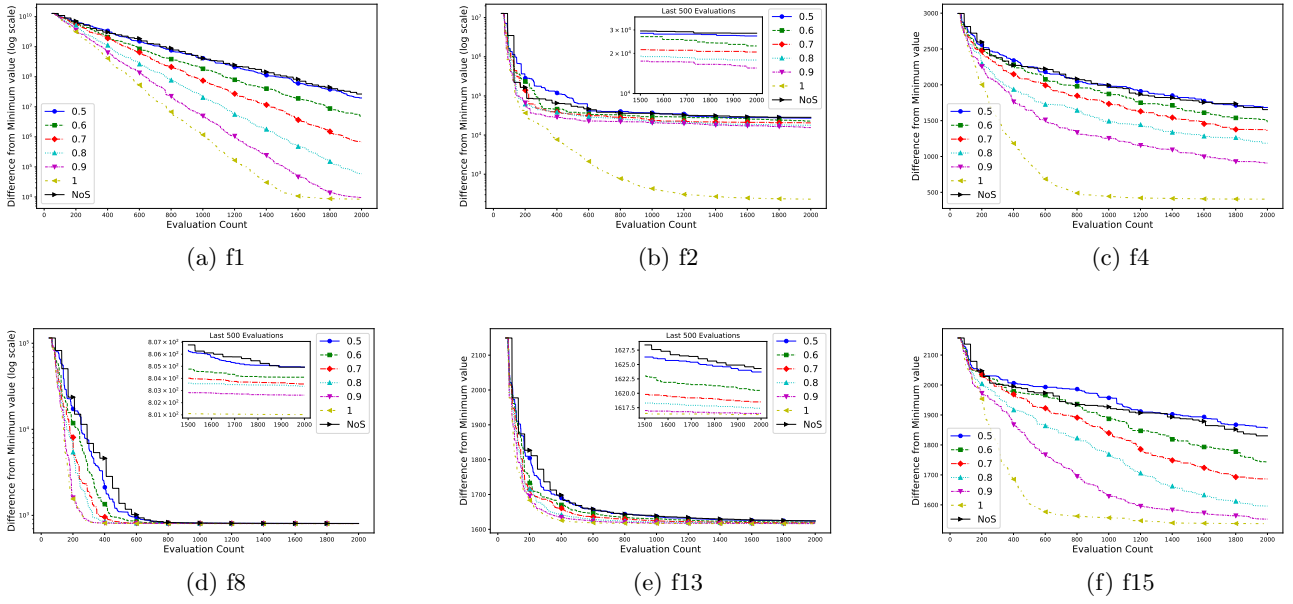


Fig. 1: PS-CMを用いて10次元問題と探索した際の目的関数値の最小値との差の推移

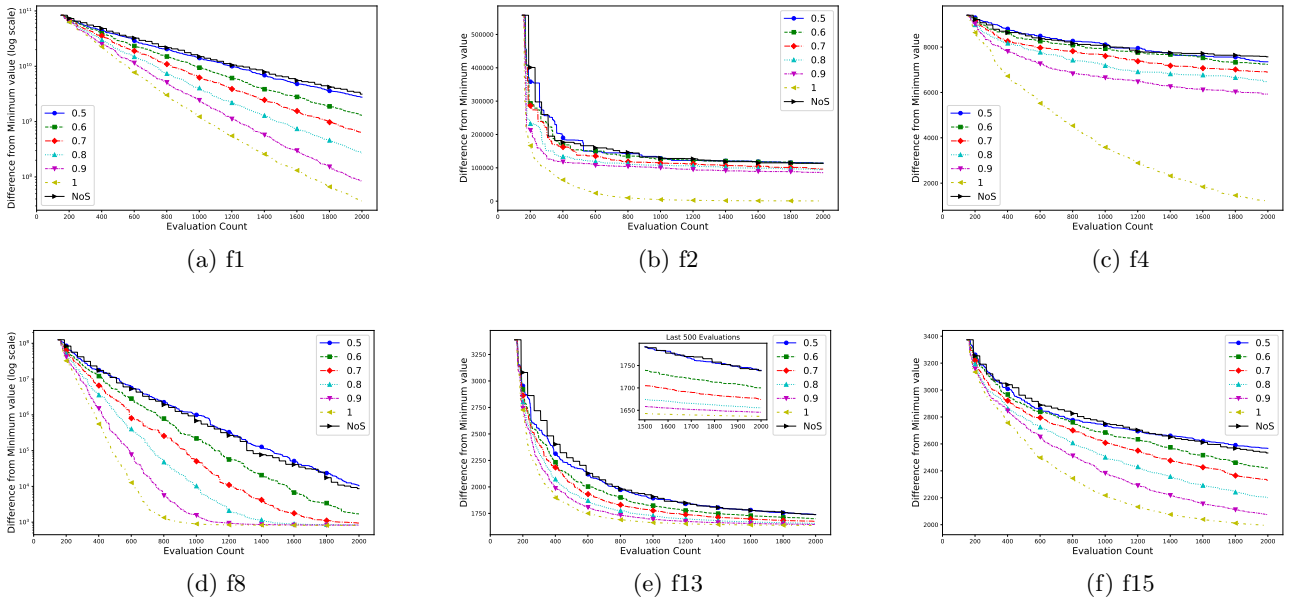


Fig. 2: PS-CMを用いて30次元問題を探索した際の目的関数値の最小値との差の推移

価することが同等に起こることになる。これは、NoSの場合の挙動と同等であるため性能が近い結果になっていると考える。また、評価回数が増えた場合サロゲートモデルの精度が低いモデルは誤判定の回数が増加し、分類の際に親個体よりも優れた子個体を見逃す要因になる。一方、劣っている個体を優れていると判断する場合はサロゲートなしのモデルが優劣に関係なく子個体を評価する方法と変わらない。よって、優れるはずの個体を棄却してしまうことが探索性能の

劣化に繋がっているといえる。

図2に30次元でのPS-CMの結果を示す。30次元の場合も、10次元の時と同様に精度1.0が最も探索性能が高いことがわかる。また、10次元と比べて、全体の傾向は同様であり、評価回数が少ない時には精度の低いものであってもサロゲートモデルを使用したものが優れた探索性能を示した。このことから、次元によるサロゲートモデルの精度やNoSの探索性能の傾向には影響がないといえる。

4.2 IB-AFM

図3にIB-AFMを用いて10次元の問題を探索した際の結果を示す。

図3aと図3bに単峰性の問題に対する探索結果を示す。 f_1 では精度1.0で最も小さい目的関数値が得られている。一方、精度が0.9以下の結果を比較すると精度が高いほど、目的関数値が小さい傾向は見られず、2000回の評価を行った際には精度が0.7の結果が0.9の結果より優れていた。また、評価回数が1570まではすべての精度でNoSよりも優れた探索性能が得られた。一方、 f_2 ではNoSが最も探索性能が低く、続いて精度1.0が探索性能が低い結果となった。精度が0.5の場合に最も良い値を探索できており、評価回数270以降は常に精度0.5のサロゲートモデルが最も小さい目的関数値を得た。

次に、多峰性関数の結果を図3cと図3dに示す。 f_4 ではNoSが最終的に最も良い目的関数値を獲得した。一方、評価回数が970までは精度1.0のサロゲートを使用した結果が最も優れていた。 f_8 では、探索初期ではサロゲートモデルを使用したほうが早く最小値に近づくが、評価回数が650を超えると全てのモデルで最小値との差は10未満となる。それ以降は、精度による探索性能の傾向はなくなり、精度0.6に続いて0.8が最も小さい目的関数値となっている。精度1.0の場合は精度0.5よりも劣る結果となった。

最後に複合関数の結果を図3eと図3fに示す。 f_{13} ではサロゲートモデル使用時の方が探索速度は早いですが、精度による差はほとんどない。 f_{15} では評価回数が218回まではサロゲートモデルが探索性能に優れるが、それ以降はNoSとの探索性能は逆転し、評価回数が530を超えるとNoSが優れた探索性能を示す。

IB-AFMを使用するとき、探索過程で発見した最小値が同程度の場合、精度が高いほど良い値を探索している傾向は見られなかった。発見した最小値の差が小さいことから、推定精度による差は小さいといえる。よって、精度が高いサロゲートモデル使用時により良い探索ができているという傾向は小さいといえる。

30次元の問題を用いた結果を図4に示す。30次元の場合、図4cの f_4 を除いてサロゲートモデルの使用がNoSより良い探索性能を示し、精度が高いほど良い探索ができている。また、 f_4 を除いてNoSは精度0.6から0.7と同程度の探索性能を示す。

10次元では精度1.0のサロゲートモデルが他の精度やNoSよりもより良い探索性能を示す傾向にあったが、30次元になるとその差は縮まる傾向にある。また、多峰性の問題を示す図4cではNoSがサロゲートモデルより良い解を探索できており、その差は10次元の時よりも大きくなっている。

全体として、評価回数が少ない場合には精度の低いサロゲートモデルの使用であっても使用することでより優れた探索ができる。評価回数が2000回まで増えると、次元数が10次元の場合には、精度の高いサロゲートモデルとNoSの探索性能は同程度であり、30次元では精度の高いサロゲートモデルが探索性能で優れる傾向があることがわかった。

5 おわりに

本論文では、サロゲートモデルの推定精度がSAEAの探索性能に与える影響を分析した。実験では、PS-CMとIB-AFMの2種類を使用した。CEC2015の単峰性、多峰性、それらの複合の全6つのベンチマーク問題を使用し、10次元と30次元の設計変数に対して最大評価回数を2000回の場合に探索過程で発見した最小値の推移を分析した。

PS-CMは10次元と30次元どちらも精度1.0のサロゲートモデルが最も良い探索性能を示した。精度0.5ではサロゲートなしのモデルよりも劣る、または近い探索性能となる。ただし、精度が0.5よりも高い場合にはサロゲートなしのモデルよりも優れ、精度が高いほど探索性能が向上する。一方、評価回数が限られる場合には、精度が低くてもサロゲートモデルを使用することが有効で、少ない評価回数でより最適値に近い値を探索することが可能である。

IB-AFMは、評価回数が限られる場合には、サロゲートモデルの使用が優れることが分かった。一方、評価回数が増えるとNoSの探索性能が向上し、サロゲートモデル使用時の探索性能と同程度または、それを上回る性能を示した。評価回数が少ない場合には、推定精度の高いモデルの探索性能が良い傾向があるが、評価回数が増えるとその傾向は見られなくなる。

今後の課題としては本論文で採用しなかったサロゲートモデルについても精度による探索性能への影響の分析を検討する。また、制約付き最適化問題や多目的最適化問題のように複数のサロゲートモデルを必要とするSAEAにおいて、

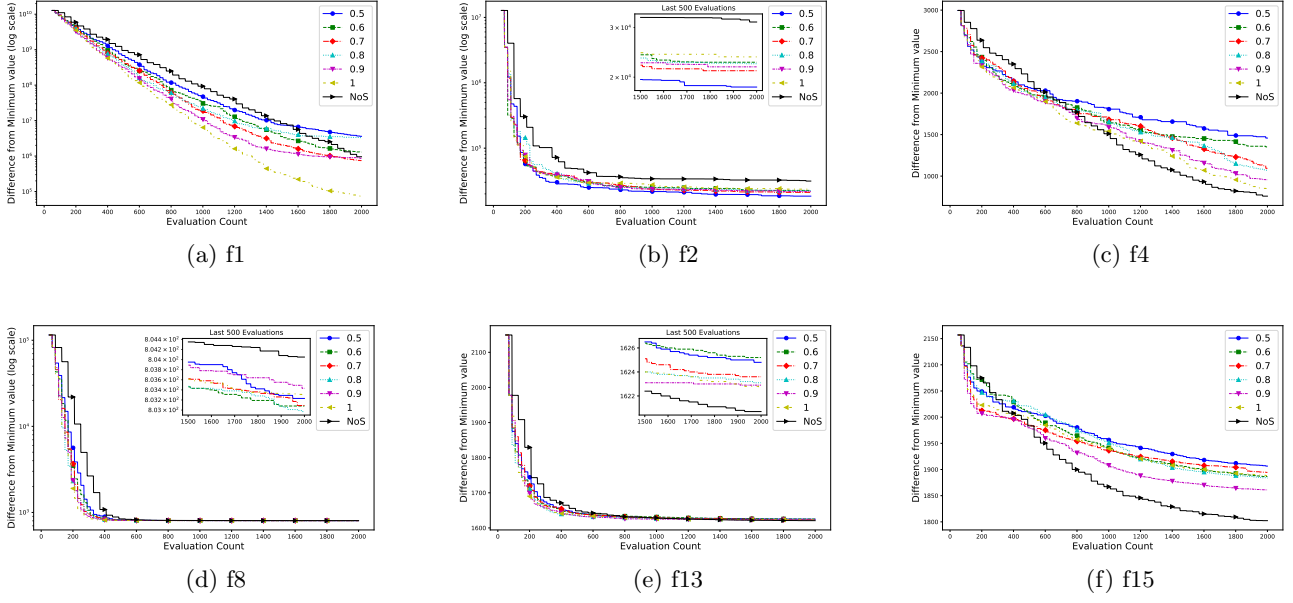


Fig. 3: IB-AFMを用いて10次元問題を探索した際の目的関数値の最小値との差の推移

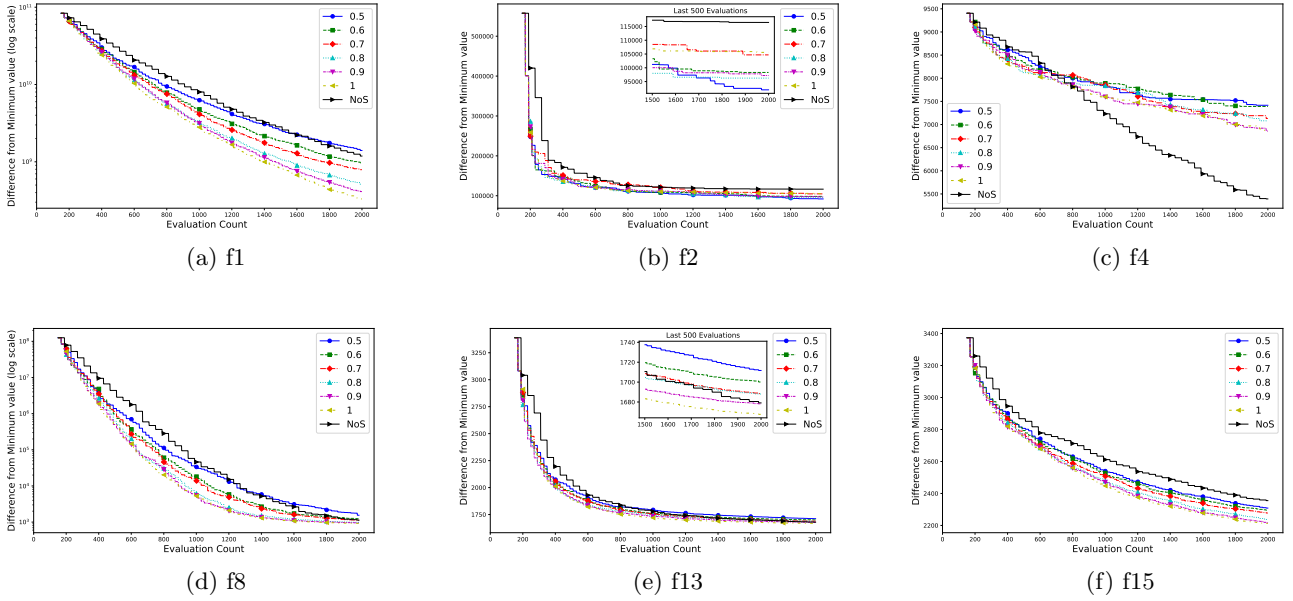


Fig. 4: IB-ABFを用いて30次元問題を探索した際の目的関数値の最小値との差の推移

推定精度と探索性能の関係を分析する。

参考文献

- 1) Q. Chen, B. Liu, Q. Zhang, J. J. Liang, P. N. Suganthan, and B. Qu. Problem definitions and evaluation criteria for cec 2015 special session on bound constrained single-objective computationally expensive numerical optimization. 2015.
- 2) C. He, Y. Zhang, D. Gong, and X. Ji. A review of surrogate-assisted evolutionary algorithms for expensive optimization problems. *Expert Systems with Applications*, 217:119495, 2023.
- 3) Y. Jin. Surrogate-assisted evolutionary computation: Recent advances and future challenges. *Swarm and Evolutionary Computation*, 1(2):61–70, 2011.
- 4) Y. Jin, H. Wang, and C. Sun. *Data-Driven Evolutionary Optimization*. Springer Interna-

tional Publishing, 2021.

- 5) H. Kano, T. Harada, and Y. Miura. Differential evolution using surrogate model based on pairwise ranking estimation for constrained optimization problems. In *2022 Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems (SCIS&ISIS)*, pp. 1–6, 2022.
- 6) B. Liu, Q. Zhang, and G. G. E. Gielen. A gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems. *IEEE Transactions on Evolutionary Computation*, 18(2):180–192, 2014.
- 7) H. Mühlenbein and D. Schlierkamp-Voosen. Predictive models for the breeder genetic algorithm i. continuous parameter optimization. *Evolutionary Computation*, 1(1):25–49, 1993.
- 8) M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.
- 9) E. Mezura-Montes and C. A. Coello Coello. Constraint-handling in nature-inspired numerical optimization: Past, present and future. *Swarm and Evolutionary Computation*, 1(4):173–194, 2011.
- 10) E. Mezura-Montes, C. A. Coello Coello, and E. I. Tun-Morales. Simple feasibility rules and differential evolution for constrained optimization. In R. Monroy, G. Arroyo-Figueroa, L. E. Sucar, and H. Sossa eds., *MICAI 2004: Advances in Artificial Intelligence*, pp. 707–716, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.
- 11) Z. Song, H. Wang, C. He, and Y. Jin. A kriging-assisted two-archive evolutionary algorithm for expensive many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 25(6):1013–1027, 2021.
- 12) T. Sonoda and M. Nakata. Multiple classifiers-assisted evolutionary algorithm based on decomposition for high-dimensional multiobjective problems. *IEEE Transactions on Evolutionary Computation*, 26(6):1581–1595, 2022.
- 13) H. Tong, C. Huang, J. Liu, and X. Yao. Voronoi-based efficient surrogate-assisted evolutionary algorithm for very expensive problems. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1996–2003, 2019.
- 14) H. Tong, C. Huang, L. L. Minku, and X. Yao. Surrogate models in evolutionary single-objective optimization: A new taxonomy and experimental study. *Information Sciences*, 562:414–437, 2021.
- 15) K. Tsujino, T. Harada, and R. Thawonmas. Analysis of relation between prediction accuracy of surrogate model and search performance on extreme learning machine assisted moea/d. In *2020 59th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 820–825, 2020.
- 16) Z. Yang, H. Qiu, L. Gao, X. Cai, C. Jiang, and L. Chen. Surrogate-assisted classification-collaboration differential evolution for expensive constrained optimization problems. *Information Sciences*, 508:50–63, 2020.