Knowledge-based Selection of Gaussian Process Surrogates

Zbyněk Pitra[1,2], Lukáš Bajer[1,3], and Martin Holeňa[1]

[1] Institute of Computer Science, Academy of Sciences of the Czech Republic
Pod Vodárenskou věží 2, 182 07 Prague 8, Czech Republic
{pitra, holena}@cs.cas.cz
[2] Faculty of Nuclear Sciences and Physical Engineering, CTU in Prague
Břehová 7, 115 19 Prague 1, Czech Republic
[3] Cisco Systems, Czech Republic
Karlovo nám. 10, 120 00 Prague, Czech Republic
bajeluk@gmail.com

Abstract Many real-world problems belong to the area of continuous black-box optimization. If the black-box function is also cost-aware, regression surrogate models are often utilized by optimization algorithms to save evaluations of the original cost-aware function. Choosing a suitable surrogate model or a suitable setting of its hyperparameters is a complex selection problem, where research into reusing knowledge represented by features of black-box function landscape is only starting. In this paper, we report the research into surrogate model selection, where knowledge from the previous experience with using the model is utilized to design a metalearing system. As a proof of concept, we provide a study investigating the influence of landscape features on the performance of various Gaussian process covariance functions as surrogate models for the state-of-the-art optimization algorithm in the cost-aware continuous black-box optimization.

Keywords: Benchmarking · Black-box optimization · Gaussian process · Landscape analysis

1 Introduction

Surrogate modeling is a technique for saving expensive evaluations of a black-box objective function during the run of an optimization algorithm. Given a set of observations, a surrogate model can be fitted to approximate the landscape of the objective function. However, which surrogate model should be chosen given a particular optimization task? Generally, no surrogate model improves the algorithm always better than all other surrogate model approaches (cf. [14,28]). The performance of each surrogate-assisted algorithm obviously depends on the properties of the data; therefore, investigation of the suitability of different models and their settings for different combinations of the data properties is very much needed.

Surrogate model selection can utilize the experience from the application of the considered models to other optimization tasks, a strategy known as *metalearing* [19]. Considering the surrogate model selection problem, it is necessary to extract information about the approximated function, which can be later utilized by a learning system to make a decision about the convenience of particular surrogate models. Therefore, features characterizing properties of the landscape of the objective function should help to better distinguish the model suitability.

In recent years, many features aiming to describe the properties of objective function landscapes have been proposed (cf. the overview in [16]). However, a majority of landscape features was utilized only for the selection of optimization algorithms and algorithm settings (a. k. a. *Algorithm Selection* or *Algorithm Configuration* problems [33]), not for the selection of surrogate models and their settings. The discussion in [14] suggests that landscape features can be used to this end, too. However, only little research in that direction is known so far.

In this paper, we report a research into designing a metalearing system for surrogate model selection according to past experience. We study relations between the performance of surrogate models and considered properties of objective function landscapes. As a proof of concept, we utilize results of the investigation in [29], where the influence of Gaussian process (GP) covariance function settings on the error of GP predictions with respect to the original fitness has been studied in connection with landscape features. We employ a classification tree showing the dependence of the most suitable covariance function on landscape features to adaptively select the most promissing covariance for the GP surrogate model in the surrogate variant of the state-of-the-art black-box optimization algorithm Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [10], the Doubly Trained Surrogate CMA-ES (DTS-CMA-ES). We evaluate the resulting algorithm performing automatical covariance function selection on the noiseless part of the COCO framework [11,12] and compare it to five DTS-CMA-ES versions without online covariance function selection.

The next section provides a brief introduction to surrogate modeling and landscape analysis. Section 3 states the proposed research problem and our approach to adress it. Section 4 presents a proof of concept of the proposed approach and its experimental results. The last section discusses the results and suggests directions for future research.

2 Background

2.1 Surrogate Modeling

Replacing an expensive function f with a trained regression model has been used to speed-up black-box optimization for many years. Such regression model, a. k. a. *surrogate model*, is trained on the already available input–output value pairs $(\mathbf{x}_i, y_i), i = 1, \ldots, N$, where $\mathbf{x}_k$ is a point in a search space and $y_k = f(\mathbf{x}_k)$ is an objective function value of x_k for $k = 1, \ldots, N$. The model is used instead

of the original expensive objective function to evaluate some of the points needed by the optimization algorithm. The *response-surface models* [26] are low-degree polynomial models and were used as the historically first models in costly continuous optimization [1,15]. Since then, other models like multi-layer perceptron- and RBF-networks [34], support vector machine regression [20], random forests [2] or Gaussian processes [2,5,27,35] were also used in black-box optimization.

Simpler models like polynomials are cheap to train; they are thus suitable for the applications where additional computational resources imposed by the model building would constitute a substantial part of the overall optimization cost. On the other hand, random forests and Gaussian processes provide estimation of the prediction uncertainty which can be used in selecting points for evaluation either with the expensive original function, or with the model fitness function [3,27].

2.2 Landscape Analysis

Landscape analysis aims at characterizing the landscape of an investigated function and deriving rules how those characteristics influence the performance of the optimization algorithm. The final goal is to formulate rules for the selection of suitable algorithms for an unknown problem according to the calculated features; this corresponds to the *Algorithm Selection* problem formulated in [33].

A large number of various landscape analysis techniques have been proposed in recent years. The following measures quantifying the characteristics of landscapes were formulated in [23]: *multi-modality*, *global structure*, *separability*, *variable scaling*, *search space homogeneity*, *basin size homogeneity*, *plateaus*, and *local to global optima contrast*. However, the majority of these *high-level properties* have the disadvantages of expert knowledge necessity, categorical character, missing important information, and requiring knowledge about the whole problem [16].

Exploratory Landscape Analysis [22] is an umbrella term for all such methods, even though originally developed for combinatorial optimization problems [25]. An important step in the development of landscape analysis was a proposal of six *low-level* easy to compute feature classes [22], each containing a number of individual features. Generally in continuous black-box optimization field, such feature is a function $\varphi : \bigcup_{N\in\mathbb{N}} \mathbb{R}^{N,D} \times \mathbb{R}^{N,1} \mapsto \mathbb{R}$ which aims to describe landscape properties utilizing a dataset of N pairs of observations $\big\{(\mathbf{x}_i, y_i) \in \mathbb{R}^D \times \mathbb{R} \,|\, i = 1, \ldots, N\big\}$. Proposed feature classes represent measures related to the distribution of the objective function values (*y-Distribution*), the relative position of each value with respect to quantiles (*Levelset*), the information extracted from linear or quadratic regression models fitted to the sampled data (*Meta-Model*), and three feature classes requiring additional objective function evaluations – the level of convexity (*Convexity*), gradient and Hessian approximation statistics (*Curvature*), and features related to local searches conducted from sampled points (*Local Search*). It was shown [22] that these low-level features relate well the above mentioned high-level properties.

The *cell-mapping* approach [18] discretizes the input space to a user-defined number of blocks (i. e., cells) per dimension. Afterwards, the corresponding

features are based on the relations between the cells and points within. Three cell-mapping feature classes were defined: features extracting information based on the location of the best and worst observation within a cell w.r.t. the corresponding cell center, aggregated cell-wise information on the gradients between each point of a cell and its corresponding nearest neighbor, and estimated convexity of representative observations from three successive cells in each dimension. Additionally, the *Generalized Cell Mapping* features are based on estimated transition probabilities of moving from one cell to one of its neighboring cells. Using those probabilities, the *barrier tree* [7] can be constructed to represent the local optima by tree leaves and landscape ridges by the branching nodes. It should be noted that cell-mapping approach is less useful in higher dimensions where majority of cells is empty and feature computation can require a lot of time and memory.

Nearest better clustering (*NBC*) features [17] are based on the detection of funnel structures. The calculation of such features is based on the comparison of distances from observations to their nearest neighbors and their *nearest better neighbors*, which are the nearest neighbors among the set of all observations with a better objective value. In [21], the set of *dispersion features* comparing the dispersion among the data points and among subsets of these points from the dataset is proposed. The *information content* features of a continuous landscape are derived in *Information Content of Fitness Sequences* approach [24] as the adaptation of methods for calculating of the information content of discrete landscapes. In [16], three feature sets were proposed: the features providing basic information about the data such as the number of points, boundaries or dimension (*Basic*), aggregated information about coefficients of linear models fitted in each cell, and information obtained from *principle component analysis* measuring the proportion of principle components needed to explain a user-defined percentage of variance. A comprehensive survey of landscape analysis methods can be found, e. g., in [25].

Research into using landscape features for surrogate modeling selection has started only recently. In [36], the *fitness distance correlation* was utilized for automatic selection between polynomial and RBF models and their settings as surrogates for a particle swarm optimization algorithm. In [30], we have investigated relationships between two surrogate models (GP and RF) and a set of landscape features. In [29], we have proposed the set of landscape features based on the state variables of the CMA-ES algorithm (*CMA features*) and investigated the relationships of GP covariance functions to landscape features.

3 Landscape Analysis for Surrogate Model Selection

3.1 Surrogate Model Selection Problem

The surrogate model selection problem can be formalized as follows: In an iteration i of a surrogate-assisted algorithm A, a set of surrogate models $\mathcal{M}$ with hyperparameters $\boldsymbol{\theta}$ are trained utilizing particular choices of the training set $\mathcal{T}$. The training set $\mathcal{T}$ is selected out of an *archive* $\mathcal{A}$ ($\mathcal{T} \subset \mathcal{A}$) using some

training set selection method (*TSS*). The archive contains all points in which the fitness f has been evaluated so far $\mathcal{A} = \{(\mathbf{x}_i, f(\mathbf{x}_i))| \, i = 1, \ldots, N\}$. Afterwards, the surrogate model $M \in \mathcal{M}$ is utilized to evaluate new set of points (*population*) $\mathcal{P} = \{\mathbf{x}_k | k = 1, \ldots, \alpha\}$, where $f(\mathbf{x}_k)$ can be obtained using the expensive black-box fitness function and $\alpha \in \mathbb{N}$ depends on the strategy for the selection of new points for evaluation by the models from $\mathcal{M}$. The main question related to this problem is: How can we select the most convenient models from the set $\mathcal{M}$ (and possibly $\boldsymbol{\theta}$) according to $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{P}$?

3.2 Proposed Methodology

We suggest to use the metalearing approach based on landscape features to tackle the surrogate model selection problem.

Learning phase: First, a set of datasets $\mathcal{D} = \{\mathcal{A}^{(l)}, \mathcal{T}^{(l)}, \mathcal{P}^{(l)}\}_{l=1}^{L}$, $L \in \mathbb{N}$, is created (ideally via recording the datasets from independent runs of the algorithm A). Second, for each l, each model $M \in \mathcal{M}$ with hyperparameters $\boldsymbol{\theta}_M$ is trained on $\mathcal{T}^{(l)}$ and its performance is assessed with some error measure ε on $\mathcal{P}^{(l)}$. Third, each dataset from $\mathcal{D}$ is characterized using a set of landscape features Φ. In this way, a mapping $S_M : \Phi \to \mathcal{M}$ or $S_\theta : \Phi \to \bigcup_{M \in \mathcal{M}} \boldsymbol{\Theta}_M$ from feature space to $\mathcal{M}$ or $\bigcup_{M \in \mathcal{M}} \boldsymbol{\Theta}_M$ is learned, where $\boldsymbol{\Theta}_M$ stands for the set of possible hyperparameters of the model M.

Application phase: In each iteration i of an algorithm A, the landscape features Φ are calculated on datasets $\mathcal{A}^{(i)}, \mathcal{T}^{(i)}, \mathcal{P}^{(i)}$. After that, the mapping S is used to select the surrogate model $M \in \mathcal{M}$ and its hyperparameters $\boldsymbol{\theta}_M \in \boldsymbol{\Theta}_M$. The selected $M \in \mathcal{M}$ is trained on $\mathcal{T}^{(i)}$ and then utilized for predicting fitness values of the elements of $\mathcal{P}^{(i)}$.

4 Proof of Concept

4.1 Learning phase

Optimization Algorithm Considering cost-aware black-box single-objective optimization of continuous functions, the CMA-ES [10] has been many times successfully improved using surrogate models to save fitness function evaluations [13,15,20,27]. The DTS-CMA-ES [3,27] has been shown a valuable representative of such surrogate-assisted versions of the CMA-ES. Therefore, we have utilized DTS-CMA-ES to play the role of the algorithm A in our concept.

Surrogate Model and Hyperparameters As a surrogate model, the DTS-CMA-ES uses Gaussian processes [31] due to their ability to estimate the whole distribution of the fitness function. In the DTS-CMA-ES, the Gaussian process model setting is fixed during the whole optimization process, so is the GP covariance function. An essential GP hyperparameter is the type of covariance function. In [32], we have proposed to select the covariance function for a GP-based surrogate model for the CMA-ES using a Bayesian approach.

Mapping The results in [29] suggested that mapping from the space of features calculated on $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{P}$ to the value set of a categorical hyperparameter can be represented by a classification tree.

Error Measure The CMA-ES state variables are adjusted according to the ordering of μ best points from the current population. Therefore, the Ranking Difference Error [3] is a convenient measure of model error for the DTS-CMA-ES

$$RDE_\mu(\hat{\mathbf{y}}, \mathbf{y}) = \frac{\sum_{i:(\rho(\mathbf{y}))_i \leq \mu} |(\rho(\mathbf{y}))_i - (\rho(\hat{\mathbf{y}}))_i|}{\max_{\pi \in \text{Permutations of } (1,\ldots,\lambda)} \sum_{i:\pi(i)\leq\mu} |\, i - \pi(i)|}\,, \tag{1}$$

where $(\rho(\mathbf{y}))_i$ is the rank of y_i among the components of $\mathbf{y}$.

Dataset To generate a set of datasets $\mathcal{D}$, we have used independent runs of the DTS-CMA-ES on the 24 noiseless single-objective benchmarks from the COCO framework [11,12] in dimensions 2, 3, 5, 10, and 20 on instances 11–15. Using each of the 8 different covariance functions from [29] in each of those independent runs, data from 25 uniformly selected generations were recorded. The runs of the algorithm were terminated in cases when the limit of 250 function evaluations per dimensions was exceeded or when the target fitness value 10^{-8} was reached. The details[1] of generating the datasets can be found in [29].

Landscape Features The following 6 feature classes were employed to characterize all the sets $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{P}$ from the datasets in $\mathcal{D}$: *y-Distribution*, *Levelset*, *Meta-Model*, *NBC*, *Dispersion*, *Information Content*, and *CMA features*. In addition, the *dimension* D and the *number of observations* N from the *Basic* feature class were also utilized. The rest of features from classes described in Subsection 2.2 were excluded, mainly due to requiring additional evaluations of the objective function f.

Classification Tree for Covariance Functions The classification tree T depicted in Figure 1 has been obtained in [29] and represents the influence of landscape features on the most suitable covariance function. To train the tree T, all the sets described by features in the previous paragraph were divided into 8 classes according to which of the 8 considered GP model settings achieved the lowest RDE_μ. The tree was trained using the MATLAB implementation of the CART algorithm [4], where all features were considered as continuous variables. The fully-grown tree was pruned to depth 8 resulting in the shown tree T. The set of training points and the respective population is denoted $\mathcal{T}_\mathcal{P} = \mathcal{T} \cup \{(\mathbf{x}, \circ) | \, \forall\, \mathbf{x} \in \mathcal{P}\}$, where $\circ$ indicates the unknown fitness value of a point from the current population $\mathcal{P}$.

[1] Source code covering all mentioned experiments is available on `http://uivty.cs.cas.cz/~cma/ecml2019/source.zip`

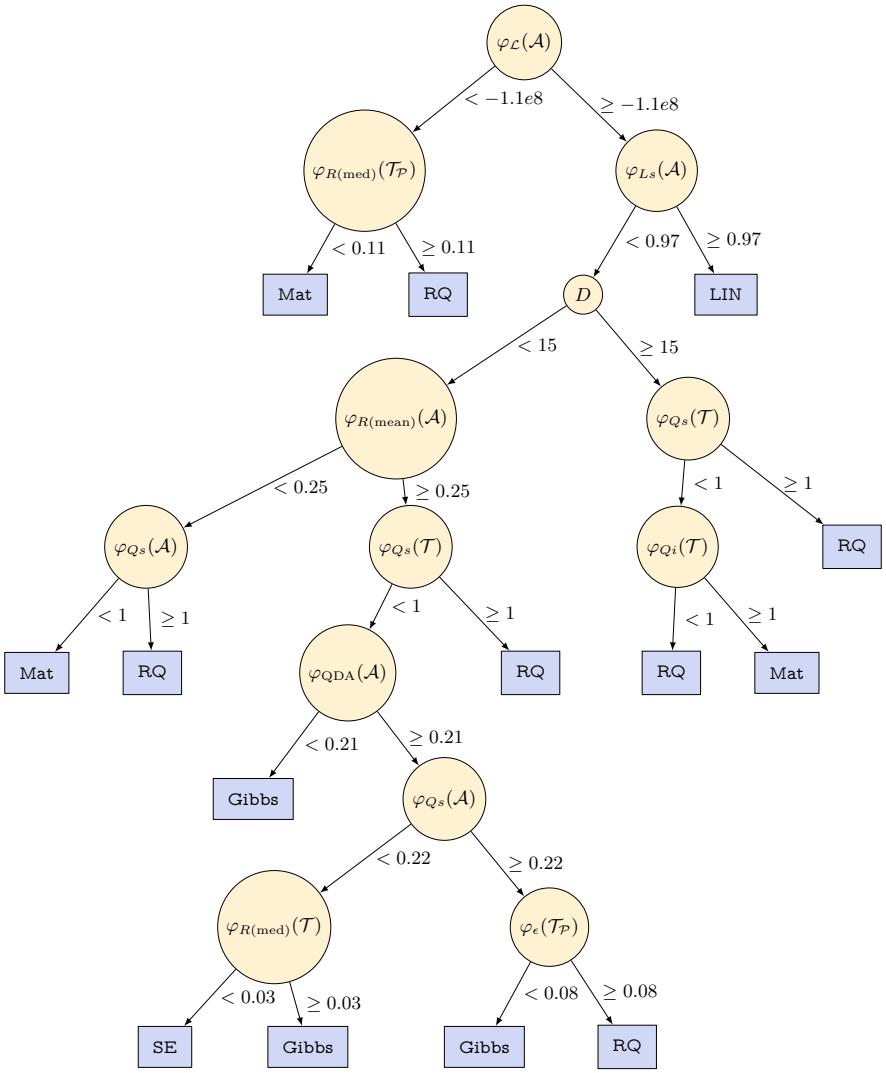

Figure 1: Classification tree T selecting the most suitable covariance function based on landscape features [29]. In each iteration of the DTS-CMA-ES, the landscape features in the splitting nodes are calculated on sets in brackets, i. e., archive of points evaluated so far $\mathcal{A}$, GP model training set $\mathcal{T}$, and the set of training points and current population $\mathcal{T}_\mathcal{P} = \mathcal{T} \cup \{(\mathbf{x}, \circ) | \forall\, \mathbf{x} \in \mathcal{P}\}$, where $\circ$ indicates an unknown fitness value of a point from the current population $\mathcal{P}$. The covariance function is determined by the leaf reached by the sequence of splitting nodes decisions. The features used for node splits are explained in the text of Subsection 4.1. Covariances in leaves are listed in Table 1.

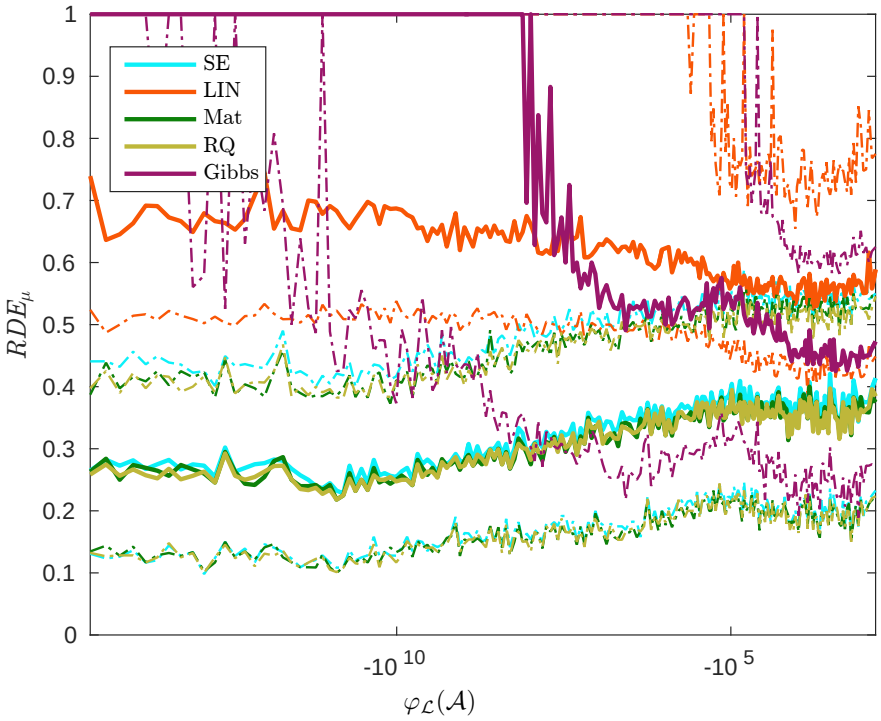

Figure 2: Median (solid lines) and $1^{\text{st}}/3^{\text{rd}}$ quartiles (dash-dot lines) of RDE_μ values dependency on $\varphi_{\mathcal{L}}(\mathcal{A})$ for all tested covariances calculated on all available datasets.

The features employed in the tree T represent various landscape properties: D is the dimension of the investigated function; $\varphi_{\mathcal{L}}$ is the log-likelihood of the set of points $\mathbf{X}$ with respect to the CMA-ES sampling distribution [29] (see Figure 2 for the average RDE_μ dependency on $\varphi_{\mathcal{L}}(\mathcal{A})$[2]); $\varphi_{R(\text{mean})}$ and $\varphi_{R(\text{med})}$ denote two ratios of the mean and median distances of the 'best' objectives vs. 'all' objectives [21]; φ_{Ls}, φ_{Qs}, and φ_{Qi} represent the adjusted R^2 (i. e., the model fit) of linear, quadratic simple, and quadratic with interactions fitted regression models [22]; φ_{QDA} is the mean missclassification error of Quadratic Discriminant Analysis on points divided into two classes according to the fitness values with median as a threshold [22]; φ_ϵ denotes the argument of the maximum information content of the fitness sequence [24].

The covariance functions located in leaves of the tree T are listed in Table 1.

[2] Figures of the RDE_μ dependencies on the remaining features can be found on an authors' webpage: `http://uivty.cs.cas.cz/~cma/ecml2019/`.

Table 1: Considered GP covariance functions. Notation: d – metric measuring the distance $d(\mathbf{x}_p, \mathbf{x}_q)$, hyperparameters σ_0 – scalar multiplication factor, σ_f^2 – signal variance, ℓ – characteristic lenght-scale (spatially varying in the Gibbs [9] covariance, where $\ell(\mathbf{x})$ is an arbitrary positive function of $\mathbf{x}$), and $\alpha > 0$.

name	kernel
linear (LIN)	$\mathbf{K}_{\mathrm{LIN}}(\mathbf{x}_p, \mathbf{x}_q) = \sigma_0^2 + \mathbf{x}_p^\top \mathbf{x}_q$
squared-exponential (SE)	$\mathbf{K}_{\mathrm{SE}}(d; \sigma_f, \ell) = \sigma_f^2 \exp\left(-\frac{d^2}{2\ell^2}\right)$
rational quadratic (RQ)	$\mathbf{K}_{\mathrm{RQ}}(d; \sigma_f, \ell) = \sigma_f^2 \left(1 + \frac{d^2}{2\ell^2 \alpha}\right)^{-\alpha}$
Matérn $\frac{5}{2}$ [31] (Mat)	$\mathbf{K}^{\frac{5}{2}}_{\mathrm{Mat}}(d; \sigma_f, \ell) = \sigma_f^2 \left(1 + \frac{\sqrt{5}d}{\ell} + \frac{5d^2}{3\ell^2}\right) \exp\left(-\frac{\sqrt{5}d}{\ell}\right)$
Gibbs [9]	$\mathbf{K}_{\mathrm{Gibbs}}(\mathbf{x}_p, \mathbf{x}_q) = \sigma_f^2 \left(\frac{2\ell(\mathbf{x}_p)\ell(\mathbf{x}_q)}{\ell(\mathbf{x}_p)^2 + \ell(\mathbf{x}_q)^2}\right)^{D/2} \exp\left(-\frac{(\mathbf{x}_p - \mathbf{x}_q)^\top (\mathbf{x}_p - \mathbf{x}_q)}{\ell(\mathbf{x}_p)^2 + \ell(\mathbf{x}_q)^2}\right)$

Algorithm 1 Covariance function selection in DTS-CMA-ES model training

Input: $\mathcal{A}$ (archive), $\mathcal{P}$ (population), $N_{\max}$ (maximum training set size), *TSS* (training set selection method), r (maximal radius of selected points), μ (GP mean function), σ (CMA-ES step-size), $\mathbf{C}$ (CMA-ES covariance function)

1: $\{(\mathbf{x}_k, y_k)\}_{k=1}^{N_{\max}} \leftarrow$ select max. $N_{\max}$ points from $\mathcal{A}$ using *TSS* and r
2: $\mathbf{x}_k \leftarrow$ transform $\mathbf{x}_k$ into the $(\sigma)^2\mathbf{C}$ basis $\quad k = 1, \ldots, N_{\max}$
3: $y_k \leftarrow$ normalize y_k to zero mean and unit variance $\quad k = 1, \ldots, N_{\max}$
4: $\mathbf{K} \leftarrow T(\mathcal{A}, \mathcal{T} = \{(\mathbf{x}_k, y_k)\}_{k=1}^{N_{\max}}, \mathcal{P})$
5: $\boldsymbol{\theta} \leftarrow$ fit the hyperparameters of $(\mu, \mathbf{K})$ by likelihood maximization

Output: M – trained GP model with hyperparameters $\boldsymbol{\theta}$

4.2 Application phase

Covariance Function Selection The implementation of the selection of the covariance function for the DTS-CMA-ES based on the classification tree T is quite straightforward. We have modified the original algorithm only in the GP model training method (see Algorithm 1). We have incorporated an additional step applying covariance function selection using the classification tree T between the training set transformation and fitting the GP hyperparameters $\boldsymbol{\theta}$.

Covariance selection validation setup We have compared the described adaptive DTS-CMA-ES that online chooses the covariance function using the tree T (denoted as T-DTS) with five DTS-CMA-ES versions that use solely one covariance from Table 1. The comparison was performed on the noiseless part of the COCO framework using instances 1–5 and 81–90 of all 24 benchmark functions in dimensions 2, 3, 5, 10, and 20. Each of the six DTS-CMA-ES versions had a budget of $250D$ fitness function evaluations to reach the target value 10^{-8} from the function optimum. Except the choice of the covariance function, the DTS-CMA-ES was tested in its non-adaptive version using the overall best settings from [3].

4.3 Results

Results from the comparison of six DTS-CMA-ES versions are depicted in Table 2 and Figures 3 and 4. The graphs in Figures 3 and 4 show the dependence of the scaled best-achieved logarithms $\Delta_f^{\log}$ of median distances Δ_f^{med} to the optimal fitness value on the number of cost-aware fitness evaluations divided by the dimension. Medians Δ_f^{med}, 1st, and 3rd quartiles are calculated from 15 independent instances for each respective algorithm, function, and dimension. The scaled logarithms of Δ_f^{med} are calculated as

$$\Delta_f^{\log} = \frac{\log \Delta_f^{\text{med}} - \Delta_f^{\text{MIN}}}{\Delta_f^{\text{MAX}} - \Delta_f^{\text{MIN}}} \log_{10}\left(1/10^{-8}\right) + \log_{10} 10^{-8}\,, \tag{2}$$

where Δ_f^{MIN} (Δ_f^{MAX}) is the minimal (maximal) distance $\log \Delta_f^{\text{med}}$ found among all the compared algorithms for the particular function f and dimension D between 0 and 250 function evaluations per D. The resulting values are scaled to interval $[-8, 0]$, where -8 corresponds to Δ_f^{MIN} and 0 to Δ_f^{MAX}. More detailed results can be found on an authors' webpage[3].

We have tested the statistical significance of performance differences on 24 COCO functions in $5D$ using the Iman and Davenport's improvement of the Friedman test [6]. The test was conducted separately for two function evaluation budgets. Let $\#\text{FE}_\text{T}$ be the smallest number of function evaluations at which at least one DTS-CMA-ES version reached the precision $\Delta_f^{\text{med}} \leq 10^{-8}$, or $\#\text{FE}_\text{T} = 250D$ if no version reached the precision within $250D$ evaluations. The DTS-CMA-ES versions are ranked on each COCO function with respect to Δ_f^{med} at a given budget of function evaluations. The null hypothesis of equal performance of all versions is rejected for the higher function evaluation budget $\#\text{FEs} = \#\text{FE}_\text{T}$, as well as for the lower budget $\#\text{FEs} = \frac{\#\text{FE}_\text{T}}{4}$ (in both cases, $p < 10^{-3}$).

We test pairwise differences in the performance using the post-hoc Friedman test [8] with the Bergmann-Hommel correction controlling the family-wise error. The numbers of functions at which one DTS-CMA-ES version achieved a higher rank than the other are enlisted in Table 2. The table also contains the pairwise statistical significances.

From the results in Table 2 and in Figures 3 and 4, we can consider the results of the T-DTS, and the DTS-CMA-ES with SE, Mat, and RQ covariances being statistically equivalent meaning that neither of them is significantly better than the other one. Looking on the detailed results on the authors' webpage[3], those covariances provided the best performance on the functions f_5, f_{8-11}, and f_{14}. On the other hand, slightly worse results can be observed on functions f_7, f_{13}, f_{16}, and f_{20}. On functions f_6 and $f_{17,18}$ the T-DTS results more or less follow SE, Mat, and RQ performance although the best performance was provided by the Gibbs covariance. The results on multimodal functions f_{22-24} show increasing T-DTS performance with growing dimension. The versions using LIN and Gibbs

[3] `http://uivty.cs.cas.cz/~cma/ecml2019/`

Table 2: A pairwise comparison of the algorithms in $5D$ over the COCO for different evaluation budgets. The number of wins of the i-th algorithm against the j-th algorithm over all benchmark functions is given in i-th row and j-th column. The asterisk marks the row algorithm being significantly better than the column algorithm according to the Friedman post-hoc test with the Bergmann-Hommel correction at the family-wise significance level $\alpha = 0.05$.

5*D*	T-DTS		LIN		SE		Matérn		RQ		Gibbs	
$\#\text{FEs}/\#\text{FE}_\text{T}^1$	$^1/_4$	1	$^1/_4$	1	$^1/_4$	1	$^1/_4$	1	$^1/_4$	1	$^1/_4$	1
T-DTS	—	—	22.5*	24*	10.5	12	11.5	12	12.5	11	17.5	22.5*
LIN	1.5	0	—	—	0.5	0	0.5	0	0.5	0	0.5	0
SE	13.5	12	23.5*	24*	—	—	10.5	11.5	13.5	9.5	15.5	20*
Matérn	12.5	12	23.5*	24*	13.5	12.5	—	—	10.5	10.5	16.5	23.5*
RQ	11.5	13	23.5*	24*	10.5	14.5	13.5	13.5	—	—	14.5	22*
Gibbs	6.5	1.5	23.5*	24	8.5	4	7.5	0.5	9.5	2	—	—

covariance functions provide considerably lower performance in comparison with the remainder. Variability of length-scale utilized by Gibbs covariance function helps the DTS-CMA-ES to converge on hard-to-regress f_6 and on multimodal *Schaffer*'s functions $f_{17,18}$ especially in higher dimensions, where the performance of DTS-CMA-ES using Gibbs covariance in GP model is the best of all compared versions.

A possible reason of the T-DTS results may lie in an imbalance of the input dataset for decision tree. Covariances SE, Mat, and RQ performed almost similar and, in average, provided the overall best prediction performance among tested covariances on the set of datasets $\mathcal{D}$. Therefore, these three covariances were marked as best on most of datasets and the remaining two (LIN and Gibbs) were best on minority of datasets. The trained classification tree was probably not able to capture such imbalance of the input data and predicted LIN or Gibbs as the most convenient covariances more often than it was necessary.

5 Conclusion and Future work

This article investigates the surrogate model selection problem for continuous single-objective black-box optimizers in the context of reusing knowledge through landscape analysis. The proposed concept was applied to select a hyperparameter of Gaussian process models, namely the covariance function, and was utilized during the DTS-CMA-ES run to save costly fitness evaluations. The DTS-CMA-ES upgraded with hyperparameter selection was compared to five DTS-CMA-ES versions using different covariances on the set of noiseless benchmarks.

The presented proof of concept has shown that the methodology can be utilized for hyperparameter selection. The tree-assisted DTS-CMA-ES had a performance equivalent to DTS-CMA-ES versions with successful fixed covariance functions. On the other hand, the classification tree as a mapping of values of

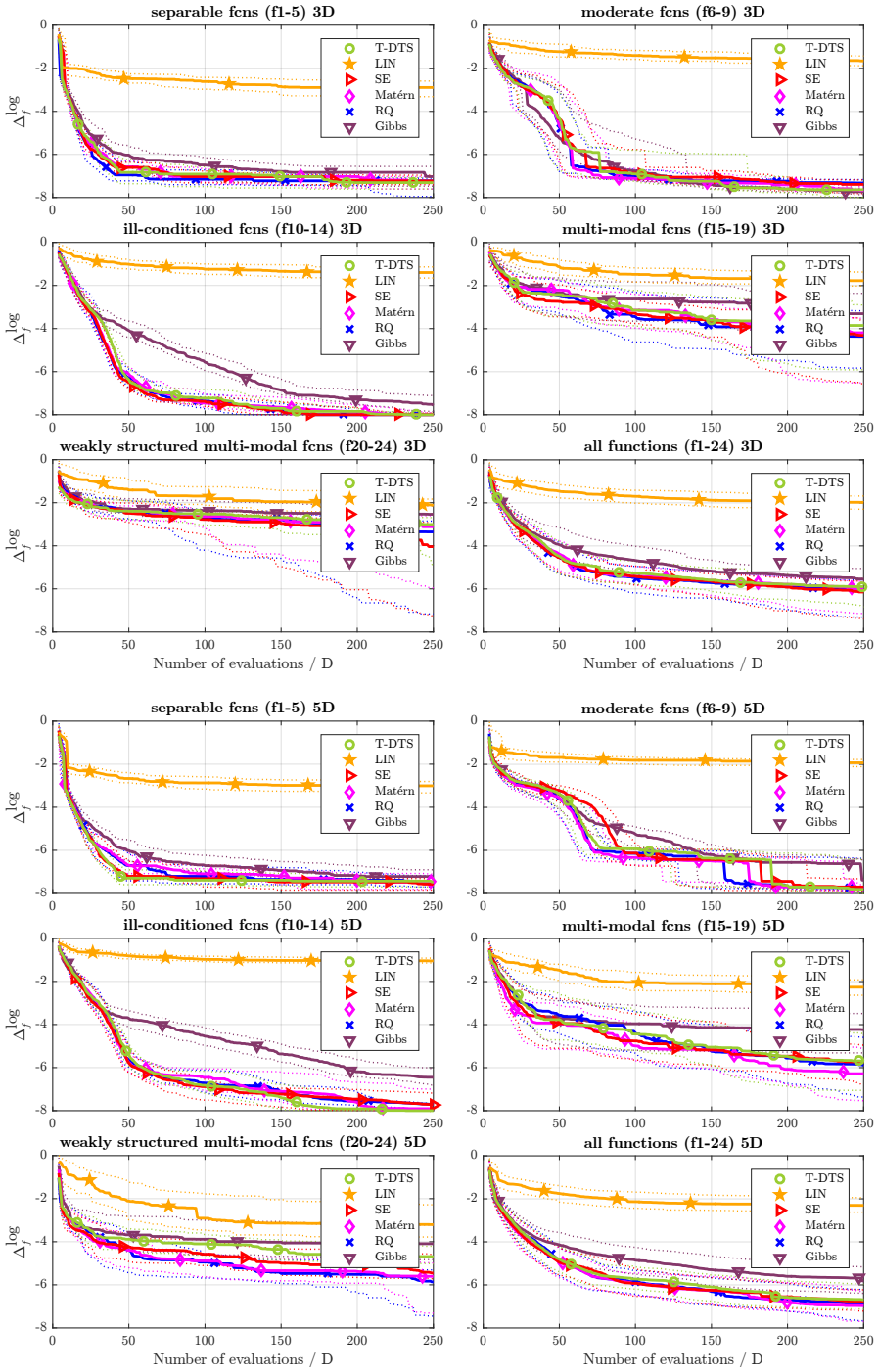

Figure 3: Scaled medians (solid) and $1^{\text{st}}/3^{\text{rd}}$ quartiles (dotted) distances $\Delta_f^{\log}$ averaged over the groups of noiseless COCO functions in $3D$ and $5D$ for different settings of DTS-CMA-ES GP covariance function.

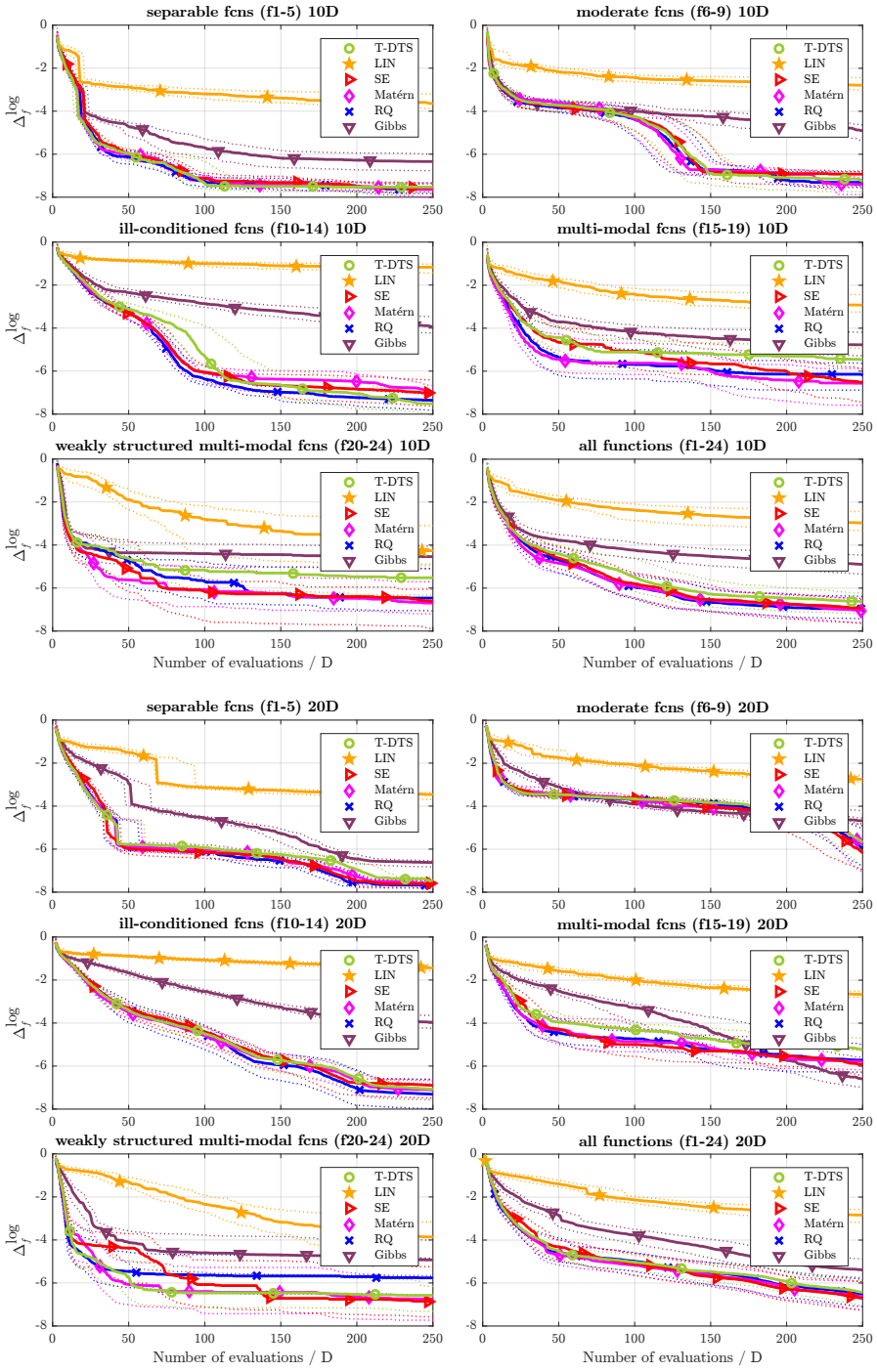

Figure 4: Scaled medians (solid) and $1^{\text{st}}/3^{\text{rd}}$ quartiles (dotted) distances $\Delta_f^{\log}$ averaged over the groups of noiseless COCO functions in $10D$ and $20D$ for different settings of DTS-CMA-ES GP covariance function.

landscape features to the covariance functions for the DTS-CMA-ES seems not to have learned very accurately.

Future research should be focused mostly on deeper understanding of the surrogate model selection problem and the possibilities of landscape analysis in this context. The investigation of various mappings to models and their hyperparameters capable to capture relationships between landscape features and surrogate model performance is definitely needed. Another direction is to extend the presented research also to other kinds of surrogate models.

Acknowledgements The reported research was supported by the Czech Science Foundation grants Nos. 17-01251S and 18-18080S and by the Grant Agency of the Czech Technical University in Prague with its grant No. SGS17/193/OHK4/3T/14. Further, access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CESNET LM2015042), is greatly appreciated.

References

1. Auger, A., Schoenauer, M., Vanhaecke, N.: LS-CMA-ES: A second-order algorithm for covariance matrix adaptation. In: Parallel Problem Solving from Nature - PPSN VIII. pp. 182–191 (2004)
2. Bajer, L., Pitra, Z., Holeňa, M.: Benchmarking Gaussian processes and random forests surrogate models on the BBOB noiseless testbed. In: Proceedings of the 17th GECCO Conference Companion. ACM, New York, Madrid (July 2015)
3. Bajer, L., Pitra, Z., Repický, J., Holeňa, M.: Gaussian process surrogate models for the CMA Evolution Strategy. Evolutionary Computation **0**(0), 1–33 (0). https://doi.org/10.1162/evco_a_00244, pMID: 30540493
4. Breiman, L.: Classification and regression trees. Chapman & Hall/CRC (1984)
5. Büche, D., Schraudolph, N.N., Koumoutsakos, P.: Accelerating evolutionary algorithms with Gaussian process fitness function models. IEEE Transactions on Systems, Man, and Cybernetics, Part C **35**(2), 183–194 (2005)
6. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research **7**, 1–30 (2006)
7. Flamm, C., Hofacker, I.L., Stadler, P.F., Wolfinger, M.T.: Barrier Trees of Degenerate Landscapes. Zeitschrift für Physikalische Chemie International Journal of Research in Physical Chemistry and Chemical Physics **216**(2), 155–173 (2002)
8. García, S., Herrera, F.: An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research **9**, 2677–2694 (2008)
9. Gibbs, M.N.: Bayesian Gaussian Processes for Regression and Classification. Ph.D. thesis, Department of Physics, University of Cambridge (1997)
10. Hansen, N.: The CMA evolution strategy: A comparing review. In: Towards a New Evolutionary Computation, pp. 75–102. No. 192 in Studies in Fuzziness and Soft Computing, Springer Berlin Heidelberg (Jan 2006)
11. Hansen, N., Auger, A., Finck, S., Ros, R.: Real-parameter black-box optimization benchmarking 2012: Experimental setup. Tech. rep., INRIA (2012)

12. Hansen, N., Finck, S., Ros, R., Auger, A.: Real-parameter black-box optimization benchmarking 2009: Noiseless functions definitions. Tech. Rep. RR-6829, INRIA (2009), updated February 2010
13. Hansen, N.: A Global Surrogate Assisted CMA-ES. In: GECCO. Prague, Czech Republic (Jul 2019). https://doi.org/10.1145/3321707.3321842
14. Jin, R., Chen, W., Simpson, T.: Comparative studies of metamodelling techniques under multiple modelling criteria. Structural and Multidisciplinary Optimization **23**(1), 1–13 (2001)
15. Kern, S., Hansen, N., Koumoutsakos, P.: Local Meta-models for Optimization Using Evolution Strategies. In: Parallel Problem Solving from Nature - PPSN IX. Lecture Notes in Computer Science, vol. 4193, pp. 939–948. Springer Berlin Heidelberg (2006)
16. Kerschke, P.: Comprehensive feature-based landscape analysis of continuous and constrained optimization problems using the R-package flacco. ArXiv e-prints (2017)
17. Kerschke, P., Preuss, M., Wessing, S., Trautmann, H.: Detecting funnel structures by means of exploratory landscape analysis. pp. 265–272. GECCO '15, ACM (2015)
18. Kerschke, P., Preuss, M., Hernández, C., Schütze, O., Sun, J.Q., Grimme, C., Rudolph, G., Bischl, B., Trautmann, H.: Cell mapping techniques for exploratory landscape analysis. In: EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation V. pp. 115–131. Springer International Publishing (2014)
19. Lemke, C., Budka, M., Gabrys, B.: Metalearning: a survey of trends and technologies. Artificial Intelligence Review **44**(1), 117–130 (Jun 2015)
20. Loshchilov, I., Schoenauer, M., Sebag, M.: Self-adaptive surrogate-assisted covariance matrix adaptation evolution strategy. In: Proceedings of the 14th GECCO. pp. 321–328. GECCO '12, ACM, New York, NY, USA (2012)
21. Lunacek, M., Whitley, D.: The dispersion metric and the cma evolution strategy. pp. 477–484. GECCO '06, ACM (2006)
22. Mersmann, O., Bischl, B., Trautmann, H., Preuss, M., Weihs, C., Rudolph, G.: Exploratory landscape analysis. pp. 829–836. GECCO '11, ACM (2011)
23. Mersmann, O., Preuss, M., Trautmann, H.: Benchmarking evolutionary algorithms: Towards exploratory landscape analysis. pp. 73–82. PPSN XI, Springer Berlin Heidelberg (2010)
24. Muñoz, M.A., Kirley, M., Halgamuge, S.K.: Exploratory landscape analysis of continuous space optimization problems using information content. IEEE Transactions on Evolutionary Computation **19**(1), 74–87 (2015)
25. Muñoz, M.A., Sun, Y., Kirley, M., Halgamuge, S.K.: Algorithm selection for black-box continuous optimization problems. Inf. Sci. **317**(C), 224–245 (2015)
26. Myers, R., Montgomery, D.: Response Surface Methodology: Process and Product in Optimization Using Designed Experiments. John Wiley & Sons, Inc., New York, NY, USA, 1st edn. (1995)
27. Pitra, Z., Bajer, L., Holeňa, M.: Doubly trained evolution control for the Surrogate CMA-ES. In: Proceedings of the PPSN XIV: 14th International Conference, Edinburgh, UK, September 17-21. pp. 59–68. Springer International Publishing, Cham (2016)
28. Pitra, Z., Bajer, L., Repický, J., Holeňa, M.: Overview of surrogate-model versions of Covariance Matrix Adaptation Evolution Strategy. GECCO '17, ACM (2017)
29. Pitra, Z., Repický, J., Holeňa, M.: Landscape analysis of Gaussian process surrogates for the covariance matrix adaptation evolution strategy. pp. 691–699. GECCO '19, ACM (2019)

30. Pitra, Z., Bajer, L., Repický, J., Holeňa, M.: Transfer of knowledge for surrogate model selection in cost-aware optimization. In: Krempl, G., Lemaire, V., Kottke, D., Calma, A., Holzinger, A., Polikar, R., Sick, B. (eds.) ECML PKDD 2018: Workshop on Interactive Adaptive Learning. Proceedings. pp. 89–94. ECML PKDD 2018, Dublin, Ireland (Sep 2018)
31. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. Adaptative computation and machine learning series, MIT Press (2006)
32. Repický, J., Holeňa, M.: Automated selection of covariance function for Gaussian process surrogate models. In: ITAT 2018 Proceedings. CEUR Workshop Proceedings, vol. 2203, pp. 64–71. CEUR-WS.org (2018)
33. Rice, J.R.: The algorithm selection problem. Advances in Computers, vol. 15, pp. 65 – 118. Elsevier (1976)
34. Sun, C., Jin, Y., Cheng, R., Ding, J., Zeng, J.: Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems. IEEE Transactions on Evolutionary Computation (2017)
35. Ulmer, H., Streichert, F., Zell, A.: Evolution strategies assisted by Gaussian processes with improved preselection criterion. In: The 2003 Congress on Evolutionary Computation, 2003. CEC '03. vol. 1, pp. 692–699 Vol.1 (Dec 2003)
36. Yu, H., Tan, Y., Sun, C., Zeng, J., Jin, Y.: An adaptive model selection strategy for surrogate-assisted particle swarm optimization algorithm. pp. 1–8. SSCI '16 (2016)