
CIplus
Band 8/2016

Model-based Methods for Continuous and Discrete Global Optimization

Thomas Bartz-Beielstein, Martin Zaefferer

Technology
Arts Sciences
TH Köln

Model-based Methods for Continuous and Discrete Global Optimization[☆]

Thomas Bartz-Beielstein*, Martin Zaeferrer

TH Köln, Faculty of Computer Science and Engineering Science, Steinmüllerallee 1, 51643 Gummersbach, Germany

Abstract

The use of surrogate models is a standard method to deal with complex, real-world optimization problems. The first surrogate models were applied to continuous optimization problems. In recent years, surrogate models gained importance for discrete optimization problems. This article, which consists of three parts, takes care of this development. The first part presents a survey of model-based methods, focusing on continuous optimization. It introduces a taxonomy, which is useful as a guideline for selecting adequate model-based optimization tools. The second part provides details for the case of discrete optimization problems. Here, six strategies for dealing with discrete data structures are introduced. A new approach for combining surrogate information via stacking is proposed in the third part. The implementation of this approach will be available in the open source R package SPOT2. The article concludes with a discussion of recent developments and challenges in both application domains.

Keywords: Surrogate, Discrete Optimization, Combinatorial Optimization, Metamodels, Machine learning, Expensive optimization problems, Model management, Evolutionary computation

[☆]This is an extended version of the contribution [1]

*Corresponding author. Phone: +49 2261 8196 6391

Email addresses: thomas.bartz-beielstein@th-koeln.de (Thomas Bartz-Beielstein), martin.zaefferer@th-koeln.de (Martin Zaeferrer)

1. Introduction

Model-based optimization (MBO) plays a prominent role in today's modeling, simulation, and optimization processes. It can be considered as the most efficient technique for expensive and time-demanding real-world optimization problems. Evaluating a cheaper surrogate model instead of the expensive objective function may significantly reduce time, space, and computing costs. Especially in the engineering domain, MBO is an important practice. Recent advances in computer science, statistics, and engineering in combination with progress in high-performance computing provide tools for handling problems, which were considered unsolvable only a few decades ago.

The first part of this article presents a survey of MBO for continuous and discrete global optimization problems. Our goal is to show connections and conceptual differences between these two domains and to discuss properties of state-of-the art MBO algorithms. Despite its growing relevance, contributions to the discrete domain have been largely disregarded and were listed as open challenges. For example, Simpson et al. [2] present an interesting history of development in the field. Another survey of metamodeling techniques, which focuses on the practitioners perspective, is given by Wang and Shan [3]. Both of these surveys mention problems from the discrete domain, but do not discuss them in depth.

The term *global optimization* (GO) will be used for algorithms and problems where the goal is to find and explore global optimal solutions with complex, multimodal objective functions [4]. Furthermore, we will focus on GO problems which belong to the class of difficult (expensive) *black-box functions*, i.e., functions for which the analytic form is unknown. Thus, nearly no structural information (e.g., number of local extrema, derivate information) is available. This setting arises in many real-world systems when the explicit form of the objective function f is not readily available, e.g., if the user has no access to the source code of a simulator. Surrogates are a popular choice, because the time required for building the surrogate is negligible compared to the evaluation of

the real-world function f .

The remainder of this article is structured as follows. Section 2 introduces a taxonomy of search algorithms and presents basic definitions. After introducing instance-based stochastic search algorithms, Section 3 describes modeling approaches for stochastic algorithms. We differentiate between models that use a distribution and models that use an explicit surrogate model. Section 4 introduces model-based optimization, which is the first choice for many optimization problems in industry. Problems and algorithms from the discrete, combinatorial domain are then introduced in Section 5. Using two MBO algorithms, namely EvoLS and SPO, recent trends and new developments in MBO are described in Section 6. Finally, a summary and an outlook are given in Section 7, including important challenges in continuous and discrete MBO.

2. Taxonomy and Definitions

We consider the continuous *optimization problem* given by

$$\text{Minimize: } f(\vec{x}) \quad \text{subject to } \vec{a} \leq \vec{x} \leq \vec{b},$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is referred to as the *objective function* and \vec{a} and \vec{b} denote the lower and upper bounds of the search space, respectively. The objective function f is assumed to be an expensive-to-evaluate black-box.

In the discrete case, $\vec{x} \in \mathbb{R}^n$ is not true anymore. Rather, a candidate solution is some discrete data structure, or object. Typical discrete data structures include, e.g., ordinal integers, categorical variables, binary variables, permutations, strings, trees or graphs in general. In most of these discrete cases, bounds \vec{a} and \vec{b} are also not required anymore.

From the algorithmic perspective, this survey focuses on search heuristics, which are mostly implemented using stochastic or random procedures. Deterministic, exact GO algorithms, i.e., algorithms providing theoretical guarantees that the attained solution is the global one within some pre-specified tolerance, are not further discussed [5]. The terms “random” and “stochastic” will be used synonymously in the remainder of this article.

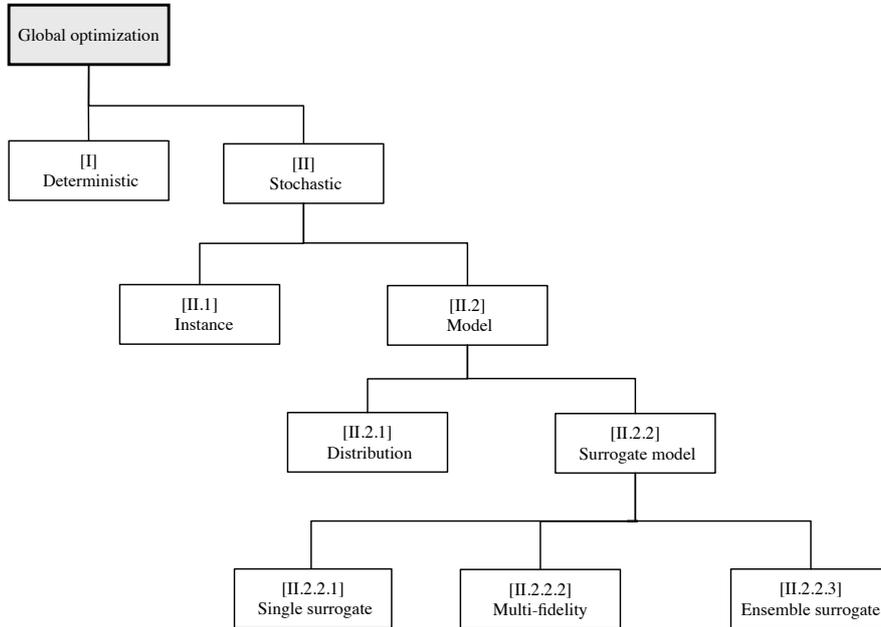


Figure 1: An essential taxonomy of model-based approaches in GO. This taxonomy is applicable for continuous and discrete GO problems.

Stochastic search algorithms can further be categorized as *instance-based* or *model-based* algorithms [6]. Furthermore, there are basically two model-based approaches: (a) *distribution-based* models and (b) *surrogate* models. We consider three important representatives of surrogate model based optimization: (i) *Single surrogate based optimization* uses one model for accelerating the search process, (ii) *multi-fidelity metamodeling* uses several models of the same real system and plays an important role in *computational fluid dynamics* (CFD) and *finite element modeling* (FEM) based simulation and optimization, and (iii) *ensemble surrogate based optimization* combines two or more different surrogate models. This categorization (or taxonomy) of GO algorithms is summarized in Figure 1.

Typical situations, which may occur in MBO, are illustrated in Figure 2.

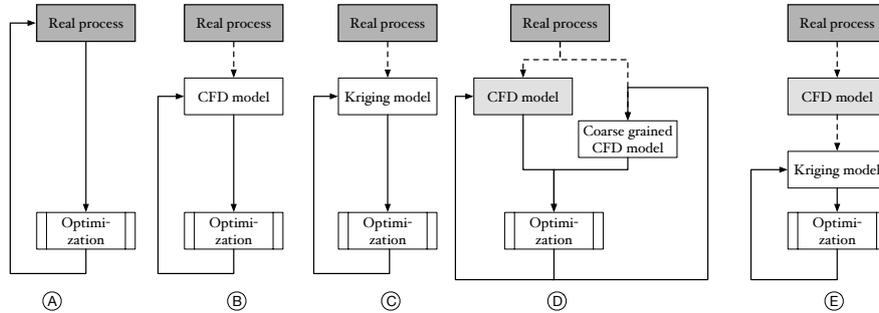


Figure 2: Surrogate model based optimization. *Dotted* lines denote the data flow, which is used for the model building. *Solid* lines represent the optimization loop. Situation A illustrates an optimization loop without any model. In situation B, a CFD model is used to accelerate the evaluation of the complex real-world function. Kriging is used as a surrogate to accelerate this evaluation in situation C. In addition to a fine grained CFD model, a coarse grained model is used in the multi-fidelity approach (situation D). Models can be stacked as shown in situation E, where a Kriging model (surrogate) is used to accelerate the CFD simulations.

70 Kriging is a frequently employed type of surrogate model [7]. Figure 2 shows (in situation C) that the Kriging model can be constructed as a surrogate model of a complex real-world process. Furthermore, a Kriging model can be used as a surrogate model of the relatively complex CFD simulation model (as illustrated in situation E). Some authors define a model as a direct abstraction of real-
75 world processes and a surrogate as a second abstraction, which highlights the properties of the model itself. Following these definitions, the Kriging model in situation C is a model, whereas the Kriging model in situation E is a surrogate. As this has the potential of confusion, we will not use this categorization.

80 Instead, we will use the term “model” for a simplified abstraction of a complex object, whereas the term “surrogate” will be used for the subset of models, which can be represented by an explicit functional relationship [8, 9]. Usually, a data-driven process, which comprehends the following two steps, is used for constructing a surrogate: (i) generation of data through sampling and (ii) mathematical function fitting. The terms “surrogate”, “surrogate model”, and

85 “metamodel” will be used synonymously in the following.

3. Stochastic Search Algorithms

An iterative search algorithm that uses a stochastic procedure to generate the next iterate is referred to as a *stochastic search algorithm*. The next iterate can be a candidate solution to the GO or a probabilistic model, where solutions
90 can be drawn from. Stochastic search algorithms are considered robust and easy to implement, because they do not depend on any structural information of the objective function, such as gradient information or convexity. This feature is one of the main reasons for the popularity of stochastic search in the domain of GO.

95 Instance-based algorithms ([II.1]) maintain a *single solution*, \vec{x} , or *population*, $P(t)$, of candidate solutions. The iteration or time step is denoted as t . The construction of new candidate solutions depends explicitly on the previously generated solutions. Simulated annealing [10], *evolutionary algorithms* (EAs) [11, 12], and tabu search [13] are prominent representatives of this cate-
100 gory. The key elements of instance-based algorithms are shown in Algorithm 1. This pseudo code focuses on the fundamental structure of the algorithm and skips some implementation details, e.g., $P(t)$ represents the population at time step t and the corresponding function values.

Algorithm 1 Instance-based algorithm

- 1: $t = 0$. $P(t) = \text{SetInitialPopulation}()$.
 - 2: Evaluate($P(t)$) on f .
 - 3: **while** not TerminationCriterion() **do**
 - 4: Generate new solutions $P'(t)$ according to random mechanism.
 - 5: Evaluate($P'(t)$) on f .
 - 6: Select the next population $P(t + 1)$ from $P(t) \cup P'(t)$.
 - 7: $t = t + 1$.
 - 8: **end while**
-

4. MBO: Model-based Algorithms

105 Model-based optimization algorithms ([II.2]) generate a population of new candidate solutions $P'(t)$ by exploiting a model (surrogate or distribution). The model reflects structural properties of the underlying true function f . Model-based optimization algorithms are based on the idea that by adapting the model, the search is directed into regions with improved solutions.

110 One of the key ideas in MBO is the replacement of expensive, high fidelity, fine grained function evaluations, $f(\vec{x})$, with evaluations, $\hat{f}(\vec{x})$, of an adequate cheap, low fidelity, coarse grained model, M . An additional benefit can be seen in the smoothing property of the cheap model, because it uses an approximation of the possibly noisy data [2].

115 This section describes two different MBO approaches: (i) distribution based ([II.2.1]) and (ii) surrogate-model based optimization ([II.2.2]).

4.1. Distribution-based Approaches

If the metamodel is a distribution, the most basic form of an MBO can be implemented as shown in Algorithm 2:

Algorithm 2 Distribution-based algorithm

- 1: $t = 0$. Let $p(t)$ be a probability distribution.
 - 2: **while** not TerminationCriterion() **do**
 - 3: Randomly generate a population of candidate solutions $P(t)$ from $p(t)$.
 - 4: Evaluate($P(t)$) on f .
 - 5: Generate updated distribution $p(t + 1)$ with population (samples) $P(t)$.
 - 6: $t = t + 1$.
 - 7: **end while**
-

Distribution-based algorithms generate a sequence of iterates (probability distributions) $\{p(t)\}$ with the hope that

$$p(t) \rightarrow p^* \text{ as } t \rightarrow \infty,$$

120 where p^* is a limiting distribution, which assigns most of its probability mass to
the set of optimal solutions. So it is the probability distribution (as opposed to
the candidate solutions in instance-based algorithms) that is propagated from
one iteration to the next.

Estimation of distribution algorithms (EDA) are well known distribution-
125 based algorithms, which became popular in the field of EAs. Classic EAs vary
candidate solutions (also called individuals) with variation operators, such as
mutation and recombination. In EDAs, variation operators were replaced by
a distribution based procedure: the new population of candidate solutions is
generated according to the probability distribution induced or estimated from
130 the promising candidate solution from the current population. Larraaga and
Lozano [14] review different ways of using probabilistic models as EDA instan-
tiations.

Although distribution-based approaches play an important role in GO, they
will not be discussed in detail in this article. The reader is referred to Hauschild
135 and Pelikan [15], who discuss advantages and outline many of the different types
of EDAs. In addition, Hu et al. [16] present recent approaches and a unified
view on distribution-based approaches. We will concentrate on surrogate model-
based approaches, which have their origin in statistical design and analysis of
experiments, especially in response surface methodology [2].

140 4.2. Surrogate Model-based Approaches

In general, surrogates are used, when the determination of a process out-
come is very expensive or time consuming. Surrogates imitate the behavior of
the real model as closely as possible, while being computationally cheaper to
evaluate. The surrogate model is also known as a response surface, metamodel,
145 approximation, coarse grained, or simply the *cheap* model. First approaches
were developed for stochastic physical experiments and used traditional design
of experiment methods. In the late 1990s, deterministic computer experiments
gained popularity. This development shifted the focus of surrogate modeling to
the *design and analysis of computer experiments* (DACE) approach [17].

150 Three variants of *surrogate model-based optimization* (SBO) became popular
in the last years: (i) single surrogate (category [II.2.2.1]), (ii) multi-fidelity
approximation ([II.2.2.2]), and (iii) ensemble surrogate ([II.2.2.3]) approaches.
The first approach uses one model, the second one uses several instances with
different parameterizations of the *same* model, and the third approach uses
155 *several different* models.

4.2.1. Single Models

Simple surrogate models are constructed with a data-driven approach. They
can be refined by integrating additional points or domain knowledge, e.g., con-
straints, into the surrogate.

160 A wide range of surrogates was developed in the last decades. Classical
regression models such as *polynomial regression* or response surface methodol-
ogy [18], *support vector machines* (SVM) [19], artificial neural networks [20],
radial basis functions [21], or *Gaussian process* (GP) models, which are some-
times also referred to as DACE or Kriging [22, 23, 24, 25, 26] are the most
165 prominent modeling approaches.

Due to this overwhelming number of approaches, it is desirable to find a
good starting point for exploring this important field. Fortunately, Forrester
et al. [7] present a comprehensible introduction to SBO with several examples.
Furthermore, Wang et al. [3] list popular metamodeling techniques and the re-
lated components, such as experimental design, sampling methods, metamodels,
170 and model fitting techniques.

Surrogate-assisted evolutionary algorithms are a popular variant of SBO.
These are evolutionary algorithms that decouple the evolutionary search and
the direct evaluation of the objective function. A good introduction and a
comprehensive survey is presented by Jin [27, 28].
175

Model-based optimization can be improved by integrating a sequential strat-
egy, which enables a model refinement during the optimization process. Addi-
tional points, which are evaluated on the expensive function, f , can be used for
building the surrogate M . As a consequence of this refinement, the surrogate

180 M depends on the time step, i.e., $M(t)$. A minimalistic *surrogate model-based optimization* (SBO) algorithm is shown in Algorithm 3.

Algorithm 3 Surrogate model based optimization (SBO) algorithm

```

1:  $t = 0$ .  $P(t) = \text{SetInitialPopulation}()$ .
2: Evaluate( $P(t)$ ) on  $f$ .
3: while not TerminationCriterion() do
4:   Use  $P(t)$  to build a cheap model  $M(t)$ .
5:    $P'(t + 1) = \text{GlobalSearch}(M(t))$ .
6:   Evaluate( $P'(t + 1)$ ) on  $f$ .
7:    $P(t + 1) = P(t) \cup P'(t + 1)$ .
8:    $t = t + 1$ .
9: end while

```

Adaptive sampling, a well-known refinement strategy, proceeds as follows: An initial model, which uses a limited amount of sample points from the expensive objective function, is refined by identifying new points, so-called *infill*
185 *points*. In the selection of adequate infill points, two conflicting goals have to be satisfied. The infill points can be selected with respect to

- exploration, i.e., improving the model quality (related to the model M)
- or
- exploitation, i.e., improving the optimization and determining the opti-
190 mum (related to the objective function f).

Adaptive sampling tries to find a balance between exploration and exploitation. A popular adaptive sampling method is *expected improvement* (EI) [29, 30], which is discussed in Jones' seminal paper [31].

MBO methods have been found to be useful in many applications. For exam-
195 ple, simulation-based design of complex engineering problems, e.g., structural design of vehicles, ships, or airplanes use CFD/FEM methods. The solvers require a large number of computer simulations to guarantee an exact solution. Hence, simulation-based design is one of the most popular and successful application areas for MBO.

200 There are several publications that describe simulation-based design MBO
applications in aerospace design [32]. A combination of a genetic algorithm
and neural networks for aerodynamic design optimization is suggested by Ha-
jela et al. [33]. The development of effective numerical methods for managing
the use of approximation concepts in optimization for a 31-variable helicopter
205 rotor design, which was part of a collaboration between Boeing, IBM, and Rice
University, is described by Booker et al. [34, 35]. Giannakoglou [36] discusses
an aerodynamic shape design problem. A surrogate-assisted evolutionary opti-
mization framework, which is applied to an airfoil shape optimization problem
using computational fluid dynamic is presented by Zhou et al. [37].

210 Obviously, the huge variety of models results in many innovative approaches
for combining successful methods and improving existing optimization algo-
rithms. Ratle [38] creates an approximate model of the fitness landscape using
Kriging interpolation to accelerate the convergence of EAs. Jin and et al. [39]
investigate the convergence property of an *evolution strategy* (ES) with neural
215 network based fitness evaluations. Emmerich et al. [40] introduce a surrogate-
assisted ES. They present results from classical test functions (artificial land-
scapes) and from an airfoil-shape optimization problem. A significant reduction
in the number of function evaluations was observed in both domains. Jin and
Sendhoff [41] use clustering techniques and neural networks ensembles to reduce
220 the number of function evaluations. Branke and Schmidt [42] propose not to
evaluate every candidate solution (individual), but to just estimate the objec-
tive function value of some of the individuals. The reduction in the number of
function evaluations is obtained by estimating an individual’s function value on
the basis of previously observed objective function values of neighboring indi-
225 viduals. Zhou et al. [37] present a surrogate-assisted EA framework, which uses
computationally cheap hierarchical surrogate models constructed through on-
line learning to replace the exact computationally expensive objective functions
during evolutionary search.

230 Although the application of surrogate modeling techniques has progressed
remarkably in the past last decades, the question remains “How far have we

really come?” This issue is addressed by Simpson et al. [2]. They conclude that while methods in the field have improved, the issues and challenges mainly remain the same: dimensionality, computational complexity, noisy data, handling of mixed continuous / discrete data and model validation (cf. Section 7).
235 They attribute this to the fact that the complexity and difficulty of problems increased simultaneously. Better tools (algorithms as well as hardware) allow to handle more difficult problems.

4.2.2. Multi-fidelity Models

In addition to single surrogates, *multi-fidelity modeling* methods were developed. Multi-fidelity modeling (or synonymously multi-fidelity metamodeling) uses several models of the same real system, where each model has its own degree of detail representing the real process. Sun et al. [43] describe a multi-fidelity optimization approach for sheet metal forming process. Further examples of multi-fidelity modeling are presented by Ulaganathan et al. [44].
240 The authors analyze the performance of Kriging [31] when multi-fidelity gradient data is introduced along with multi-fidelity function data to approximate black-box simulations.

Koziel et al. [45] present a methodology for fast multi-objective antenna optimization with *co-Kriging*. Co-Kriging is an extension of Kriging, which
250 uses the correlations between the models of various fidelities, so that cheap- and expensive simulation data can be combined into one surrogate model [46, 47]. Co-Kriging-based sequential design strategies are presented by Le Gratiet and Cannamela [48]. The authors simulate a spherical tank under internal pressure. Further applications from the water industry are published by Razavi et al [49].
255 Tuo et al. [50] proposed a finite-element analysis with its mesh density as the tuning parameter. A problem in casting simulation is used to illustrate this approach. Kleijnen [51] presents an overview of the most recent approaches in simulation practice, including multi-fidelity modeling.

4.2.3. Ensembles

260 The model building and selection process is crucial for the effectivity and efficiency of SBO. The EI approach, which was introduced in Section 4.2.1, handles the initialization and refinement of a surrogate model, but not the selection of the model type. The popular *efficient global optimization* (EGO) algorithm uses a Kriging model, because Kriging inherently determines the prediction vari-
265 ance, which is necessary for the EI criterion [30]. But there is no proof that Kriging is the best choice. Alternative surrogate models, e.g., regression trees, support vector machines, or lasso and ridge regression may be better suited, depending on the problem. For example, Müller and Shoemaker [52] reported that Kriging models and combinations including Kriging performed in general
270 worse than when RBF models are used.

An *a priori* selection of the best suited surrogate model is often impossible in the framework treated in this article, because of the black-box setting described in Section 1.

Regarding the model choice, the user can decide whether to use (i) a single,
275 global model during the optimization, i.e. category [II.2.2.1] in Figure 1, (ii) multi-fidelity models, i.e., category [II.2.2.2], or (iii) multiple models, M_i , $i = 1, 2, \dots, p$, which is category [II.2.2.3]. Ensembles comprehend an aggregate of several models. In contrast to multi-fidelity models, ensembles do not necessarily combine models of different fidelity. Classical ensemble methods rather combine
280 data driven models of similar or identical fidelity. One example are random forests, which represent an ensemble of simple tree-based models.

To reduce the computing time, an ensemble of different, possibly local, models can be generated and evaluated in parallel. Each model may employ the same candidate solutions (from the population P) and the corresponding obser-
285 vations from expensive function evaluations. Haftka et al. [53] present a review of surrogate methods that provide parallelization.

Since each of the p surrogates proposes one or even more different candidate solutions, a selection and/or a combination mechanism is required. Conventio-

ally, surrogate models are assessed and chosen according to their estimated true
290 error [54, 55, 2]. Generally, attaining a surrogate model that has minimal error is
the desired feature. The *mean absolute error* (L1 norm), the *mean square error*
(MSE) (or its pendant the *root mean square error* (RMSE)) are commonly used
as performance metrics. Methods from statistics, statistical learning [56], and
machine learning [57], such as the simple holdout approach, cross-validation,
295 and the bootstrap are also important in this context.

Several selection and combination mechanisms for surrogates were developed
in the last years. A simple approach determines the best model, i.e., the model
with the smallest prediction error, and determines the next candidate solution
based on that model. Alternatively, candidate solutions from several models
300 can be combined. Zerpa et al. [58] use multiple surrogate models and build an
adaptive weighted average model of the individual surrogates. Goel et al. [59]
explore the possibility of using the best surrogate model or a weighted average
surrogate model instead of one single model. Model quality, i.e., the errors in
surrogates, is used to determine the weights assigned to each model. Sanchez et
305 al. [60] present a weighted-sum approach for the selection of model ensembles.
The models for the ensemble are chosen based on their performance and the
weights are adaptive and inversely proportional to the local modeling errors.
Tenne and Armfield [61] propose a surrogate-assisted memetic algorithm which
generates accurate surrogates using multiple cross-validation tests.

310 Huang et al. [62] use several simulation models for a semiconductor manu-
facturing system. They propose an ordinal transformation to utilize the results
from several cheap models. The function values of all solution candidates is
evaluated on every cheap model and the individuals are ranked. The authors
observe that despite the big bias in the results from the cheap models, the rel-
ative order among solutions is actually quite accurate. This order can be used
315 to accelerate the selection process in EAs significantly. To reduce variability
and bias in the results from the cheap models, the authors apply an optimal
computing budget allocation scheme.

Multiple models can also be used to partition the search space. The treed

320 Gaussian process approach uses regression trees to partition the search space
into separate regions and to fit local GP surrogates in each region [63]. Nelson
et al. [64] propose an algorithm, which creates a tree-based partitioning of an
aerodynamic design space and employs independent Kriging surfaces in each
partition. Couckuyt et al. [65] propose to combine an *evolutionary model selec-*
325 *tion* (EMS) algorithm with the EI criterion in order to dynamically select the
best performing surrogate model type at each iteration of the EI algorithm.

5. Surrogate Models in Discrete Optimization

Compared to their frequent use for real-valued problem domains, surrogate
model driven approaches are relatively scarce in combinatorial or mixed opti-
330 mization [28]. Discrete problems deal, e.g., with ordinal integers, categorical
(qualitative) variables, permutations, strings, trees, or other graph based data
structures. They may be mixed among themselves, or mixed with continuous
variables. Ordinal integer variables can often be handled quite similarly to real
valued variables. Others, like trees, are too complex to be easily represented by
335 numeric vectors.

Few expensive, real-world combinatorial optimization problems have been
published, e.g., in the engineering domain [66, 67, 68, 69], bioinformatics [70], or
data science [71]. Not all of them make use of surrogate models. This scarcity is
unlikely due to a lack of problems in this field. Rather, the availability of suitable
340 methods (i.e., surrogate models, category [II.2.2]) is not well known or these
methods are not easily accessible to experts in potential application domains.
Thus, we provide a survey of surrogate modeling methods for combinatorial,
discrete problems. The general taxonomy for these methods is the same as for
the continuous case, which was illustrated in Figure 1.

345 5.1. Strategies for Dealing with Discrete Structures

Table 1 presents a tabulated overview of the literature on discrete, surrogate
model-based optimization. This overview presents important stepping stones

Table 1: Overview of surrogate models in combinatorial optimization. Column *data* lists data type: Mixed (mix), ordinal (ord), categorical (cat), binary (bin), permutation (per), signed permutation (-per), trees (tre), other (oth). Column *s* lists the strategy, see Section 5.1. Column *dim* lists the dimensionality of the problem, where applicable. Abbreviations introduced in the table: *genetic algorithm* (GA), *non-dominated sorting genetic algorithm II* (NSGA2), *simulated annealing* (SA), *artificial neural networks* (ANN), *ant colony optimization* (ACO), *multi-start local search* (MLS). This table is continued on page 17.

data	STR	model	optimizer	cost	budget	dim	topics	ref.
mix, cat, ord	1,3	Kriging, Tree	visual, statistical analysis	high	≤ 100	2, 9	parameter tuning	[72]
mix, ord, cat	6	RBFN	ES	low / \sim high	560 / 280	15 / 23	benchmark, medical image analysis	[73]
mix, ord, cat	3,6	Random Forest, Kriging	MLS	\sim high	-	4-76	algorithm tuning	[74]
mix bin, cat	6	RBFN + cluster	GA	low	2,000	12	benchmark, chemical industry	[75]
mix, ord, cat	4	SVM	NSGA2	?	2,000	10	FEM, multi criteria	[76]

and interesting applications in the field, thus showcasing the development. For a more extensive table we refer to the tabular overview in the supplemental material of this article¹. In Table 1, the employed modeling strategies, model types, optimizers, and problem-related details are specified. The table lists works on (mixed) integer problems, binary representations, permutation problems, tree structures, and other representations.

To deal with modeling in combinatorial search spaces, a set of six *strategies* (STR) can be identified in the literature, which are referenced in column “STR” of Table 1:

¹The tabular overview will be kept up-to-date on the second author’s home page https://martinzaefferer.de/?page_id=134

Table 1: continued

data	STR	model	optimizer	cost	budget	dim.	topics	ref.
bin	1/3	ANN	SA	high	?	16	real world, pump positioning	[77]
bin	6	RBFN	GA	low	dim ²	10–25	NK-Landscape	[78]
-per	2	custom	brute force	high	28	6	real world: weld sequence	[66]
per	6	RBFN	GA	low	100	30–32	benchmark	[79]
per	6	Kriging	GA	low	100	12–32	benchmark	[80]
per	6	Kriging	ACO	low	100 - 1,000	50–100	benchmark, tuning	[81]
tre	6	RBFN	GA	low	100		symbolic regression	[82]
tre	5,6	k-NN	GA	high	30,000		phenotypic similarity, genetic programming	[83]
tre	5	Random Forest	GA	low	15,000		benchmark, genetic programming	[84]

STR-1 The naive approach: As long as the data can still be represented as a vector (binary variables, integers, categorical data, permutations) the modeling technique may simply ignore the discrete structure, and work as usual. A potential drawback of this approach is, that the model's input space may have large areas of redundancy. Or else, this approach may create large areas of infeasible solutions. Depending on the optimizer, this may deteriorate performance compared to more sophisticated approaches.

STR-2 Custom modeling: A specific modeling solution is tailored to fit the needs of a certain application. On the one hand, this procedure can be very efficient, because it integrates significant prior knowledge into the model. On the other hand, it may introduce bias and may be hard to transfer to other applications or data structures. This approach is not applicable for true black-box problems.

STR-3 Inherently discrete models: Some models already are discrete in their own design. One example are tree-based models, like regression trees or random forests. On the one hand, these models are easy to use, because no or only minor adaptations are necessary. On the other hand, this strategy may fail if the discrete structures become more complex (e.g., trees or other graph structures). Also, such models may not always provide desired features, e.g., the nice properties derived from the uncertainty estimates of a Kriging model.

STR-4 Mapping: Often, discrete variables or structures may be mapped to a more easily handleable representation. Examples for this approach are the random key mapping for permutations or dummy variables for categorical variables. Similarly to strategy STR-1, this approach may introduce redundancy or infeasibility into the data structure. Establishing reasonable mappings becomes harder for data structures with increasing complexities.

STR-5 Feature extraction: Instead of directly modeling the relation between an object (or its representation) and its quality, it is possible to calculate real-valued features of the objects. For example, some features of a tree or graph can be extracted (path lengths, tree depths, etc.). These numeric features can then be modeled with standard techniques.

STR-6 Similarity-based modeling: Where available, measures of (dis)similarity may be used to replace continuous measures that are, e.g., employed in similarity-based models like *k-nearest neighbor* (*k*-NN), *support vector machines* (SVM), *radial basis function networks* (RBFN), or Kriging. While this approach is potentially very powerful, a drawback is the requirement of proper measures. This may be problematic if these measures have to fulfill further requirements, like definiteness.

The presented six strategies are not necessarily mutually exclusive. Depending on the point of view, a mapping approach can be interpreted as similarity-

400 based approach, or vice versa. Or else, features may be extracted and used for modeling, while at the same time applying some inherently discrete approach to the data. Thus, some methods may either combine several strategies, or else, can be classified as belonging to several strategies.

None of the six strategies can be broadly preferred to the others. Even the 405 naive approach may be adequate, if the problem is sufficiently simple. Problem type and application restrictions will govern the selection of a suitable approach. The subsequent sections present key characteristics of strategies considered by the authors to be most relevant.

5.2. Custom Models

410 One way of using surrogate-models for combinatorial problems is to employ customized, application specific solutions (STR-2). An example is the work by Voutchkov et al. [66]. They optimize a weld sequence, which is represented as a signed permutation, and find a near-optimal welding sequence by evaluating only 28 out of 46,080 possible sequences. The surrogate model replaces an 415 expensive FEM by estimating the influence of each individual element in the weld sequence, based on the observations made in previously tested sequences. In addition to the function values, the surrogate also exploits intermediate results that reflect the impact of individual sequence-elements, depending on their position in the weld sequence. Exploiting these intermediate results is a clear 420 advantage over more simple, function-value driven approaches. On the other hand, the applicability of this model is restricted to this specific setup and cannot be easily transferred to completely different application areas. The surrogate modeling approach in [66] has since been adapted and applied to a girth weld path optimization problem by Asadi and Goldak [67]. Due to the nature 425 of their problem (number of weld sub-passes, rotational symmetry) only 48 sequences are possible. They evaluated 15 of these 48 sequences to find an optimal solution. Nguyen et al. [85] propose to use simplifications of the target function (a simulation model for job shop scheduling) as a surrogate model in genetic programming. These applications show that custom models are beneficial, if

430 domain knowledge is available.

5.3. Mapping

As stated in Section 5.1, a frequently chosen approach is mapping from the more complex, discrete space to another more easy to handle space (STR-4). Classical regression provides an establishes mapping approach for dealing with
435 discrete, categorical parameters, which is based on dummy variables and contrasts. A categorical variable can be mapped to a set of dummy variables, where each dummy represents a single level of the original variable. This is the same as the one-hot encoding, i.e., only one of several bits can be 1. The n levels of the categorical variable are mapped to n binary variables. The binary variables are
440 still discrete, but can be handled by standard regression approaches. Else, one can map to $(n - 1)$ binaries: The one missing level is represented by all dummy variables being zero. This is essentially a contrast. Contrast variables represent a comparison between different levels. Clearly, both approaches increase the number of model parameters. Dummy variables or contrasts are probably
445 among the most frequently applied methods to deal with categorical variables, e.g., in the context of the response surface methodology [86].

Another example of a mapping approach is the random key mapping developed by Bean [87]. Random keys are vectors of real numbers from a fixed interval, e.g., the interval $[0, 1]$. To map from this real parameter space to per-
450 mutation representations, the random key values are simply sorted. Random keys were originally developed to enable the application of continuous, model-free genetic algorithms to permutation problems. They have since been used in probabilistic models which are part of EDAs, see also Section 4.1 and 5.6. Notable drawbacks are redundancy in the real valued parameter space as well
455 as the non-bijection character of the mapping.

5.4. Feature Extraction

Of the few approaches that deal with modeling for tree-representations (symbolic regression, genetic programming), many use feature extraction for modeling. Hildebrandt and Branke [83] extract features of the phenotypic behavior

460 of evolved dispatching rules for job shop scheduling. Here, these features are based on the rule decisions for a small set of reference situations. The Euclidean distance is computed on these features and is then employed in a k -NN surrogate model. From a different point of view, this process can as well be seen as a part of the (dis-)similarity calculation, and hence part of strategy STR-6, 465 rather than STR-5. Hildebrandt and Branke compare the phenotypic approach to a genotypic distance measure: the structural Hamming distance. The phenotypic feature extraction approach has since also been investigated by Nguyen et al. [88], who improved handling of replications and the selection scheme.

Pilat and Neruda [84] take a slightly different approach. They extract fea- 470 tures from the genotype of candidate solutions, e.g., the depth of the tree, summarizing statistics of numeric constants in the tree, or the cardinality of certain arguments in the tree.

Some of these recent approaches use a rather simple surrogate model, i.e., k -NN [83, 88]. This is tightly linked to the comparatively high evaluation bud- 475 gets (tens of thousands). More complex models might become too expensive when data-sizes grow to these dimensions. On the other hand, using such large budgets may be necessary due to the extremely large search spaces that genetic programming is typically dealing with. Compared to the more common, model-free genetic programming approaches, few tens of thousands of evaluations are 480 in fact a small budget.

5.5. (Dis)similarity based models

A promising approach that recently gained more traction is similarity-based modeling (STR-6). Fonseca et al. [89] defined similarity-based models as models that keep a memory of solutions and estimate the performance of new samples 485 by comparing them to that memory. Fonseca et al. list fitness inheritance [90], fitness imitation [91, 27] and k -NN [92] as examples. They test a genetic algorithm supported by a k -NN model on a set of numerical, continuous test functions. Bernardino et al. [93] perform similar tests with artificial immune systems. In both cases Hamming and Euclidean distances are used as mea-

490 sures of dissimilarity, showing that this approach does not depend on a specific
measure.

However, the k -NN model is not able to predict whether or not a candidate
solution will provide improvement over existing solutions. For that purpose,
another set of similarity or distance based models is of interest: RBFN, SVM,
495 and Kriging (Gaussian process regression). Hemker [94] describes in his Ph.D.
thesis an approach to deal with continuous and ordinal integer variables, ap-
plied to electrical engineering and water resource management problems. Their
approach is based on Kriging, essentially treating all parameters with the same
distance function, hence not directly applicable to categorical parameters or
500 more complex representations. Li et al. [73] proposed an adaptation of RBFNs.
These adaptations are based on a weighted distance measure, replacing the
usual distance measure employed in RBFN. Their distance-based RBFN model
was tested with mixed integer optimization problems. It has since been ap-
plied to a multi objective building design optimization problem by Brownlee
505 and Wright [95]. Mixed optimization problems also arise in algorithm tuning.
In this context, Hutter [74] also used a Kriging model with a Hamming distance
based correlation function to handle categorical variables. A combination of
Kriging and tree-based models called treed Gaussian processes has been used
by Swiler et al. [96]. They applied a treed Gaussian process model, a Kriging
510 model, and a smoothing spline technique to build surrogate models for mixed
integer problems. Coelho et al. [97] and Herrera et al. [76] applied a kernel-
based regression method to mixed-variable optimization problems. A different
approach with RBFs for the case of a mixed (discrete and continuous) optimiza-
tion problem is taken by Bajer and Holena [75]. Instead of creating one RBFN
515 based on a weighted distance between candidate solutions they use Hamming
distance to cluster the discrete variables, then fit a standard RBFN with the
continuous variables for each cluster.

Moraglio and Kattan [78] adapted an RBFN to arbitrary distance measures
to model arbitrary combinatorial optimization problems. Their approach has
520 also been applied to quadratic assignment problems [79]. A similar conceptual

extension for Kriging was investigated by Zaefferer et al. [80]. Here, Kriging-based EGO [30] showed positive results when applied to combinatorial problems. Zaefferer et al. [98] also showed that *maximum likelihood estimation* (MLE) can be used to efficiently choose one measure from a set of distance measures.

525 An investigation by Cáceres et al. [81] reports negative results for the application of EGO to permutation problems. There, an ant colony optimization algorithm was not outperformed by the Kriging-based variant of the same algorithm. Smith et al [71] describe a study on extremely high-dimensional test instances, employing RBFN models. While the employed models showed some
530 promise, a proposed ensemble of models performed poorly.

The above modeling approaches use distances and kernels in a standard, straight-forward way. However, it is often important to consider if either the employed distances are *conditionally negative semi-definite* (CNSD) or *positive semi-definite* (PSD). Definiteness is a frequent requirement for modeling meth-
535 ods based on distances or kernels. Dealing with the possible indefiniteness of a function is hence of utmost concern. In fact, lack of definiteness is one possible source of some previously observed numerical problems, e.g., in [80, 98]. A recent study by Zaefferer and Bartz-Beielstein [99] dealt with the issue of definiteness in the context of Kriging based optimization. They transfer and extend methods
540 from the field of SVMs, which were previously used for machine learning with structured data, c.f., the survey by Schleif and Tino [100]. While, to the best of our knowledge, SVMs have rarely been used as surrogates in combinatorial optimization, they have been applied to learning problems with combinatorial data (see e.g., [101, 102, 103]). Hence, they are a promising choice.

545 Most of the above references make use of distance measures in genotype space. For the case of genetic programming, Hildebrandt and Branke [83] show, that a distance in phenotype space may be an excellent choice, see also the preceding section.

5.6. Other models

550 One important set of models that is mostly ignored the above considerations are classic probabilistic models (category [II.2.1]). For a description, see Section 4.1. Examples of combinatorial optimization algorithms employing such models are variants of EDA or ACO [6, 15]. Compared to the earlier presented works, these algorithms usually operate under different assumptions and are
555 often applied to less expensive (although not necessarily cheap) problems. Still, they can be considered to use surrogate models. If necessary, a mapping between search and objective function can be established (cf. [104, 105]). Applications of ACO for (expensive) combinatorial problems, e.g., include the works by Cáceres et al. [81] and Teixeira et al. [69]. EDAs have been applied to problems in building design [106] and reactor fuel management [107].
560

From the strategy point of view, these distribution models can again be classified into the earlier described six categories. For instance, there are EDAs which employ a mapping approach (STR-4), e.g., EDAs for permutation problems using the random key mapping [108]. Other EDAs use problem-specific
565 solutions (STR-2). One example are EDAs based on the (generalized) Mallows model for permutations [109, 110, 111]. These may also be categorized as being dissimilarity or distance-based (STR-6), but are not easily extended to arbitrary other distances.

Another set of models not explicitly mentioned yet are graph-based models
570 like Bayesian networks or *Markov random fields* (MRFs). These are of interest, as they may directly encode the graph structure of the candidate solution representations [112]: Directed graphs in Bayesian networks and undirected graphs in MRFs. In this graph, discrete variables may be represented by nodes, and dependencies between variables by edges. Graph based models
575 are closely linked to and have been employed in distribution based approaches, e.g., [113, 114, 104, 105, 106]. Mostly, these approaches can be classified as inherently discrete models (STR-3).

Kriging can be seen as a special case of Gaussian MRFs, that is, the fully connected graph case. However, Kriging does not model the graph structure of

580 the representation: nodes are candidate solutions, not variables.

6. Recent Trends and Developments

The model error is not the only criterion for selecting the surrogate. Recent approaches such as the *evolvability learning of surrogates* (EvoLS) algorithm implement local models for each offspring individually [115]. This results in an adaptive semi-partition [116] of the search space. In contrast to this local procedure, Bartz-Beielstein [1, 117] presents a global approach, which combines 585 information from every surrogate, even from surrogates, which have a low accuracy. These approaches will be presented in Section 6.1 and 6.2, respectively.

6.1. Evolvability Learning of Surrogates

590 The naive approach of choosing the surrogate according to its estimated error on f does not use the available information efficiently. Poorly selected experimental designs might introduce some bias: Models, which have a low estimated error in irrelevant regions of the search space can outperform models that have a higher error, but reflect the overall structure of the search space much better. This situation is illustrated in Figure 3. Additionally, Viana and 595 Haftka [118] observe that an improved global accuracy of the surrogate is not necessary for obtaining the best result.

The EvoLS algorithm, which is introduced by Le et al. [115], extends the basic surrogate-assisted EA. The authors of EvoLS recommend selecting surrogate models that enhance search improvement in the context of optimization. 600 EvoLS processes information about the (i) different fitness landscapes, (ii) state of the search, and (iii) characteristics of the search algorithm to statistically determine the so-called *evolvability* of each surrogate model. The evolvability of a surrogate model estimates the expected improvement of the objective function value that the new candidate solution, \vec{x}' , has gained after a local search 605 has been performed on the related surrogate model. The local optimizer on the surrogate model guarantees (theoretically) convergence to the stationary point of the exact objective function [119], [120].

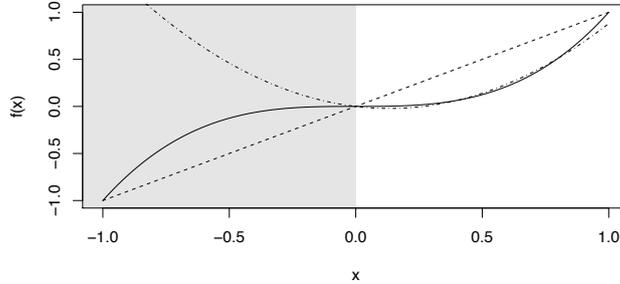


Figure 3: A simple, artificial example which illustrates that the surrogate model with the smallest prediction error is not always the best choice. The solid line represents the true cubic function f . A *dash-dotted* line is used to plot quadratic surrogate, whereas a *dotted* line depicts the linear surrogate. Consider a minimization problem. The initial design points are located in the interval $I_+ = \{x : 0 \leq x \leq 1\}$. Compared to the linear model, the quadratic surrogate has a relatively small prediction error in I_+ . However, from a global search perspective, which also includes the *grey shaded* region, the quadratic model is misleading. The linear model has a larger prediction error in I_+ , but will guide the search in the region of optimal values.

A minimalistic EvoLS algorithm is shown in Algorithm 4. The pseudo code
 610 presents a simplistic description of the fundamental states, e.g., the EvolutionaryVariation() operation in line 4 includes a parental selection, recombination, and mutation step.

The surrogate selection (line 6) uses the evolvability, which is a function of the candidate solution, \vec{x} , and the surrogate M . The model with the highest
 615 evolvability is used to perform a local search (line 7). The EvoLS algorithm presents a sophisticated way of exploiting local information during the surrogate modeling phase.

6.2. Sequential Parameter Optimization

Early versions of the *sequential parameter optimization toolbox* (SPOT) combined methods from *design of experiments*, *response surface methodology*, *design and analysis of computer experiments*, and regression trees for the analysis of
 620 algorithms [121, 122, 123]. The SPOT was developed as a toolbox for the analysis and for an understanding of the working principles of EAs. It implements a

Algorithm 4 EvoLS algorithm. See [115] for a more detailed description.

```
1:  $t = 0$ .  $P(t) = \text{SetInitialPopulation}()$ .
2: Evaluate( $P(t)$ ) on  $f$ .
3: while not TerminationCriterion() do
4:    $P'(t) = \text{EvolutionaryVariation}(P(t))$ .
5:   for each  $\vec{x}$  in  $P'(t)$  do
6:      $M = \text{SurrogateSelection}(\vec{x})$ .
7:      $\vec{x}' = \text{LocalSearch}(M, \vec{x})$ .
8:     Evaluate( $\vec{x}'$ ) on  $f$ .
9:      $P(t + 1) = P(t) \cup \{\vec{x}'\}$ .
10:  end for
11:   $t = t + 1$ .
12: end while
```

sequential, surrogate model based approach to optimization. Information from
625 the surrogate model is used for analyzing the working principles of EAs. Effects
of the the algorithm parameters, e.g., the population size or the type of recomb-
ination operator, and their interactions can be statistically analyzed. Nowadays,
the SPOT is an established parameter tuner and an optimization algorithm,
which has been extended in several ways. For example, Hutter et al. [124]
630 benchmark an SPOT derivative, the so-called *sequential model-based algorithm
configuration* (SMAC) procedure, with the *black-box optimization benchmark*
(BBOB) set of test functions [125]. They demonstrate that with a small budget
of $10 \times d$ evaluations of d -dimensional functions, SMAC in most cases out-
performs the state-of- the-art blackbox optimizer *covariance matrix adaptation*
635 *evolution strategy* (CMA-ES).

The most recent version, SPOT2, integrates a state-of-the-art ensemble
learner. The SPOT2 *ensemble engine* can be briefly outlined as follows (see
Algorithm 5): The portfolio or set of surrogate models, \mathfrak{M} , includes a large
amount of surrogates, such as regression trees and random forest, *least angle*
640 *regression* (LARS), and Kriging, see line 2 in Algorithm 5. As such, it includes

several models that can also handle continuous, ordinal, categorical, and binary variables. Thus, SPOT2 can also deal with combinatorial optimization problems. The SPOT2 ensemble engine uses cross validation to select an improved model from the portfolio of candidate models [126]. It implements methods for
645 creating a weighted combination of several surrogate models to build the improved model and methods, which use stacked generalization to combine several level-0 models of different types with one level-1 model into an ensemble [127]. The level-1 training algorithm is typically a relatively simple linear model, see line 5 in Algorithm 5. Figure 1 in [117] illustrates the stacking procedure.

Algorithm 5 Sequential parameter optimization with stacking

```

1:  $t = 0$ .  $P(t) = \text{SetInitialPopulation}()$ .
2: Select the set of cheap models  $\mathfrak{M}$ .
3: Evaluate( $P(t)$ ) on  $f$ .
4: while not TerminationCriterion() do
5:   Use  $P(t)$  to build a stacked model  $M(t)$  using  $\mathfrak{M}$ .
6:    $P'(t + 1) = \text{GlobalSearch}(M(t))$ .
7:   Evaluate( $P'(t + 1)$ ) on  $f$ .
8:    $P(t + 1) = P(t) \cup P'(t + 1)$ .
9:    $t = t + 1$ .
10: end while

```

650 The implementation of the proposed stacking approach is available in the open source R package SPOT2, made available in the supplemental material of this article².

Results indicate that the SPOT2 ensemble engine can lead to significant performance improvements, which is illustrated by the following example: Re-
655 bolledo et al. [128] present a comparison of different data driven modeling meth-

²The SPOT2 software package and the supplemental material, which is necessary to run the following experiments, can be downloaded from the first author's home page <http://www.gm.fh-koeln.de/~bartz/SPOT/>

Table 2: Comparison of the MSE values of individual models and a stacked ensemble. Smaller values are better. Best results are shown in *bold* face. Values are based on the gas sensor data example [128]. Numbers in brackets indicate standard deviations (based on ten repeats). Since Kriging and the linear model are deterministic, their standard deviation is zero. Numbers for the individual models differ from previous results [128], because different model implementations are used.

	Kriging	Linear Model	Random Forest	SPOT2 Ensemble
sensor 1	0.495 (0)	0.593 (0)	0.594 (0.014)	0.328 (0.085)
sensor 2	0.419 (0)	0.577 (0)	0.463 (0.009)	0.287 (0.066)

ods. The models are built on industrial data for the development of a robust gas sensor. The data contains a limited amount of samples with high variance. The mean square error of the models implemented in a test dataset is used as the comparison strategy. Two sensors were tested in this comparison.

660 Table 2 compares the results received by a stacked ensemble with the results of individual models which are contained in the ensemble. Training and testing were repeated ten times, to account for randomness in the random forest model as well as the ensemble model. While Kriging is the best individual model, the ensemble consistently outperforms it. The larger variance in the ensemble 665 results can be attributed to the dependence on the random sampling in the cross-validation procedure. In the level-1 model of the ensemble, Kriging received the highest coefficients. This real-world application example demonstrates the potential merit of stacked ensembles. The code of this example is made available in the supplemental material of this article.

670 7. Summary and Outlook

Model-based optimization (and in particular SBO) approaches are probably the most efficient methods for expensive and time-demanding real-world optimization problems. Although this observation is clearly true in the engineering domain, similar tendencies can be observed in many other domains, e.g., drug- 675 design [129, 28].

Summarizing, the contribution of this article can be described as follows: A comprehensible taxonomy of MBO algorithms for global optimization problems (Figure 1) is proposed. A survey of SBO algorithms, which includes the most recent publications from continuous and the combinatorial problem domains, is given. Six strategies for dealing with modeling in combinatorial search spaces are developed. Working principles of two state-of-the-art SBO algorithms were shown: (i) EvoLS, which constructs a local metamodel for every new candidate solution, and (ii) SPOT2, which uses a global ensemble engine to combine a broad variety of surrogate models. The survey presented in the first sections of this article as well as the examples in Section 6 emphasize the trend to ensemble based metamodels. Due to the emerging-field nature of SBO, and especially combinatorial SBO, several challenges remain to be solved. This article concludes with a (subjective) selection of the most challenging problems.

7.1. Model Selection

The selection of an adequate surrogate plays a crucial role in the SBO process. The surrogate should reflect the underlying complex process as exact as possible and should be as simple as possible. The estimation of the model quality is an open research question. For example, the definition of the corresponding training sets for the holdout or cross-validation approaches represents a critical issue for the accuracy and efficiency of the surrogates. Viana and Haftka [118] report that it is beneficial to run EGO with multiple surrogates. Interestingly, they also observe that an improved global accuracy of the surrogate is not necessary for obtaining the best result.

The number of potential surrogate model types and selection strategies is huge. The earlier mentioned stacking approach (cf. Section 6.2) can prove to be a helpful starting point. Building stacked ensembles of surrogate models may help to understand how different approaches perform, and how they can interact. Besides providing more accurate predictions, stacking may enable researchers to design more promising modeling approaches for complex data structures.

Model selection is still a challenging research topic with many open questions,

which is especially true in the combinatorial domain. There, little guidelines are available, due to the relative scarcity of publications.

7.2. *Definiteness*

In case of similarity-based models (strategy STR-6, see Section 5.5), definiteness is a crucial issue. While first results are available for Kriging models [99],
710 some important details require further attention. First, more extensive experiments are of interest. Secondly, more efficient handling of prediction for new data samples would be beneficial for performance in practice. And finally, a theoretical analysis could provide a more sound foundation for the existing ap-
715 proaches.

7.3. *Dimensionality*

The question of dimensionality, i.e., the number of variables, is an important issue. For continuous problems, it is often stated that distance-based models like Kriging perform poorly for larger dimensional problems. A rough threshold
720 of approximately 20 dimensions is frequently specified for Kriging, e.g., see [7]. This is closely linked to the specific distance function [130]. For instance, Manhattan distance will be less affected by such issues, compared to Euclidean distance [130]. For the general, discrete case with an arbitrary distance function such knowledge is often not available. At the same time, feature selection
725 or other dimensionality reduction methods may not be instantly available for problems with complex, discrete data representations. Thus, further research on dimensionality issues is recommendable. This is especially of interest for high dimensional real-world problems as, e.g., investigated by Smith et al. [71]. Simpson et al. [2] list some recent approaches to tackle this “curse of dimension-
730 ality” and problems related to generate adequate surrogates in high-dimensional and complex non-linear search spaces.

Dimensionality can also be interpreted as the number of samples, which leads to additional challenges. Nowadays, an increasing number of applications generate large data sets. Instead of very expensive and small data sets, huge data sets

735 have to be processed in this situation. Several applications from bioinformatics,
social media, or climate science rely on the modeling of massive data sets with
up to billions of data samples. Recently developed methods rely on special ma-
trix structures (Kronecker or Toeplitz). For example, Wilson et al. [131] present
early results for massively scalable Gaussian processes, which enables the use of
740 Gaussian processes on billions of data points. The development of multi-fidelity
surrogates, which use coarse grained models to obtain similar results as the ex-
act model might be very interesting. Furthermore, methods for data sets that
do not satisfy special matrix structures are of great interest.

7.4. Benchmarking

745 An important issue is the set of benchmark or test functions, used to evaluate
algorithm and modeling performance. Previous approaches try to compose a set
of test functions with many different features, e.g., by using step, linear, convex,
and sinusoidal functions. A more recent approach [132] uses an infinite number
of test problem instances to prevent an over fitting (or better: over learning) of
750 the competing algorithms. The test problems are based on real-world problem
instances, which are systematically modified.

While benchmarking is still not resolved for continuous model-based opti-
mization, the situation is even less settled in the discrete domain. Of the few
published, real-world, expensive, combinatorial problems, most are not openly
755 accessible. Even in case of availability, the benchmark set would be rather small
and the expense of computation would hinder broader experimental studies.
Thus, most of the benchmark problems are well known cheap-to-compute prob-
lems, e.g., the traveling salesperson problem (see the overview in Table 1). It
is questionable whether performances estimated for these problems are actually
760 representative for real-world expensive problems. In fact, the simple problem
structure may be unable to give proper credit to complex models like Kriging.
This may be one reason for results as, e.g., observed by Caceres et al. [81].

7.5. Multiple Objectives

Last, but not least, the problems discussed so far for single objective SBO exist also for model-based multi-objective optimization algorithms. The question of using global or local models is discussed in several publications, e.g., Isaacs et al. [133] present a local approach, which maintains an external archive, whereas Pilat and Neruda [134] present an approach by aggregating metamodels for evolutionary multiobjective and many-objective optimization. Horn et al. [135] present a taxonomy for model-based multi-objective optimization algorithms, which can be recommended as a starting point.

Acknowledgements

This work has been supported by the *Bundesministerium für Wirtschaft und Energie* under the grants KF3145101WM3 and KF3145103WM4. This work is part of a project that has received funding from the *European Unions Horizon 2020 research and innovation program* under grant agreement No 692286.

References

- [1] T. Bartz-Beielstein, A Survey of Model-Based Methods for Global Optimization, in: G. Papa, M. Mernik (Eds.), *Bioinspired Optimization Methods and their Applications*, 2016, pp. 1–18.
- [2] T. Simpson, V. Toropov, V. Balabanov, F. Viana, Design and Analysis of Computer Experiments in Multidisciplinary Design Optimization: A Review of How Far We Have Come - Or Not, in: *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2012, pp. 1–22. doi:10.2514/6.2008-5802.
- [3] G. G. Wang, S. Shan, Review of Metamodeling Techniques in Support of Engineering Design Optimization, *Journal of Mechanical Design* 129 (4) (2007) 370–380. doi:10.1115/1.2429697.

- 790 [4] M. Preuss, *Multimodal Optimization by Means of Evolutionary Algorithms*, Natural Computing Series, Springer International Publishing, Cham, 2015. doi:10.1007/978-3-319-07407-8.
- [5] A. Neumaier, Complete search in continuous global optimization and constraint satisfaction, *Acta Numerica* 13 (2004) 271–369.
- 795 [6] M. Zlochin, M. Birattari, N. Meuleau, M. Dorigo, Model-Based Search for Combinatorial Optimization: A Critical Survey, *Annals of Operations Research* 131 (1-4) (2004) 373–395. doi:10.1023/B:ANOR.0000039526.52305.af.
- [7] A. Forrester, A. Sóbester, A. Keane, *Engineering Design via Surrogate Modelling*, Wiley, 2008.
- 800 [8] J. P. C. Kleijnen, A comment on blanning’s metamodel for sensitivity analysis: The regression metamodel in simulation, *Interfaces* 5 (3) (1975) 21–23.
- [9] O. Garitselov, S. P. Mohanty, E. Kougianos, A comparative study of metamodels for fast and accurate simulation of nano-cmos circuits, *IEEE Transactions on Semiconductor Manufacturing* 25 (1) (2012) 26–36. doi:10.1109/TSM.2011.2173957.
- 805 [10] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, Optimization by Simulated Annealing, *science* 220 (4598) (1983) 671–680. doi:10.1126/science.220.4598.671.
- 810 [11] A. E. Eiben, J. E. Smith, *Introduction to Evolutionary Computing*, Springer, Berlin, Heidelberg, New York, 2003.
- [12] T. Bartz-Beielstein, J. Branke, J. Mehnen, O. Mersmann, Evolutionary Algorithms, *WIREs Data Mining and Knowledge Discovery* 4 (2014) 178–195.
- 815

- [13] F. Glover, M. Laguna, Tabu search, in: C. R. Reeves (Ed.), *Modern Heuristic Techniques for Combinatorial Problems*, John Wiley & Sons, Inc., New York, NY, USA, 1993, Ch. Tabu Search, pp. 70–150.
- [14] P. Larraaga, J. A. Lozano, *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*, Kluwer, Boston MA, 2002.
- [15] M. Hauschild, M. Pelikan, An introduction and survey of estimation of distribution algorithms, *Swarm and Evolutionary Computation* 1 (3) (2011) 111–128. doi:10.1016/j.swevo.2011.08.003.
- [16] J. Hu, Y. Wang, E. Zhou, M. C. Fu, S. I. Marcus, A Survey of Some Model-Based Methods for Global Optimization, in: D. Hernández-Hernández, J. A. Minjárez-Sosa (Eds.), *Optimization, Control, and Applications of Stochastic Systems*, Birkhäuser Boston, Boston, 2012, pp. 157–179. doi:10.1007/978-0-8176-8337-5_10.
- [17] J. Sacks, W. J. Welch, T. J. Mitchell, H. P. Wynn, Design and analysis of computer experiments, *Statistical Science* 4 (4) (1989) 409–435.
- [18] K. B. W. G E P Box, On the Experimental Attainment of Optimum Conditions, *Journal of the Royal Statistical Society. Series B (Methodological)* 13 (1) (1951) 1–45.
- [19] V. N. Vapnik, *Statistical learning theory*, Wiley (1998).
- [20] J. M. Zurada, Analog implementation of neural networks, *IEEE Circuits and Devices Magazine* 8 (5) (1992) 36–41. doi:10.1109/101.158511.
- [21] M. J. D. Powell, Radial basis functions for multivariable interpolation: A review, in: J. C. Mason, M. G. Cox (Eds.), *Algorithms for Approximation*, Clarendon Press, New York, NY, USA, 1987, pp. 143–167.
- [22] M. Schonlau, *Computer Experiments and Global Optimization*, Ph.D. thesis, University of Waterloo, Ontario, Canada (1997).

- [23] D. Büche, N. N. Schraudolph, P. Koumoutsakos, Accelerating Evolutionary Algorithms With Gaussian Process Fitness Function Models, *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* 35 (2) (2005) 183–194. doi:10.1109/TSMCC.2004.841917.
- [24] A. B. Antognini, M. Zagoraiou, Exact optimal designs for computer experiments via Kriging metamodeling, *Journal of Statistical Planning and Inference* 140 (9) (2010) 2607–2617. doi:10.1016/j.jspi.2010.03.027.
- [25] J. P. C. Kleijnen, Kriging metamodeling in simulation: A review, *European Journal of Operational Research* 192 (3) (2009) 707–716. doi:10.1016/j.ejor.2007.10.013.
- [26] T. J. Santner, B. J. Williams, W. I. Notz, *The Design and Analysis of Computer Experiments*, Springer, Berlin, Heidelberg, New York, 2003.
- [27] Y. Jin, A comprehensive survey of fitness approximation in evolutionary computation, *Soft Computing* 9 (1) (2005) 3–12.
- [28] Y. Jin, Surrogate-assisted evolutionary computation: Recent advances and future challenges, *Swarm and Evolutionary Computation* 1 (2) (2011) 61–70. doi:http://dx.doi.org/10.1016/j.swevo.2011.05.001.
- [29] J. Močkus, On Bayesian Methods for Seeking the Extremum, in: *Optimization Techniques IFIP Technical Conference*, 1974, pp. 400–404.
- [30] D. R. Jones, M. Schonlau, W. J. Welch, Efficient Global Optimization of Expensive Black-Box Functions, *Journal of Global Optimization* 13 (1998) 455–492.
- [31] D. R. Jones, A Taxonomy of Global Optimization Methods Based on Response Surfaces, *Journal of Global Optimization* 21 (2001) 345–383.
- [32] A. I. J. Forrester, A. J. Keane, Recent advances in surrogate-based optimization, *Progress in Aerospace Sciences* 45 (1-3) (2009) 50–79. doi:10.1016/j.paerosci.2008.11.001.

- [33] P. Hajela, E. Lee, Topological optimization of rotorcraft subfloor structures for crashworthiness considerations, *Computers & Structures* 64 (1-4) (1997) 65–76. doi:10.1016/S0045-7949(96)00143-5.
- [34] A. J. Booker, J. E. Dennis Jr, P. D. Frank, D. B. Serafini, V. Torczon, Optimization Using Surrogate Objectives on a Helicopter Test Example, in: *Computational Methods for Optimal Design and Control*, Birkhäuser Boston, Boston, MA, 1998, pp. 49–58. doi:10.1007/978-1-4612-1780-0_3.
- [35] A. J. Booker, J. E. Dennis Jr, P. D. Frank, D. B. Serafini, V. Torczon, M. W. Trosset, A rigorous framework for optimization of expensive functions by surrogates, *Structural Optimization* 17 (1) (1999) 1–13. doi:10.1007/BF01197708.
- [36] K. C. Giannakoglou, Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence, *Progress in Aerospace Sciences* 38 (1) (2002) 43–76. doi:10.1016/S0376-0421(01)00019-7.
- [37] Z. Zhou, Y. S. Ong, P. B. Nair, A. J. Keane, K. Y. Lum, Combining Global and Local Surrogate Models to Accelerate Evolutionary Optimization, *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* 37 (1) (2007) 66–76. doi:10.1109/TSMCC.2005.855506.
- [38] A. Ratle, Accelerating the convergence of evolutionary algorithms by fitness landscape approximation, in: A. E. Eiben, T. Bäck, M. Schoenauer, H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature — PPSN V: 5th International Conference Amsterdam, The Netherlands, Proceedings*, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 87–96. doi:10.1007/BFb0056852.
- [39] Y. Jin, M. Olhofer, B. Sendhoff, On Evolutionary Optimization with Approximate Fitness Functions., in: D. W. et al. (Ed.), *Proceedings of the*

Genetic and Evolutionary Computation Conference GECCO, San Francisco, CA, USA, 2000, pp. 786–793.

- 900 [40] M. Emmerich, A. Giotis, M. Özdemir, T. Bäck, K. Giannakoglou, Metamodel-assisted evolution strategies, in: J. J. M. Guervós, P. Adamidis, H. G. Beyer, J. L. Fernández-Villacañás, H. P. Schwefel (Eds.), *Parallel Problem Solving from Nature—PPSN VII, Proceedings Seventh International Conference, Granada*, Springer, Berlin, Heidelberg, New York, 2002, pp. 361–370.
- 905 [41] Y. Jin, B. Sendhoff, Reducing Fitness Evaluations Using Clustering Techniques and Neural Network Ensembles, in: *Parallel Problem Solving from Nature - PPSN XIII - 13th International Conference, Ljubljana, Slovenia, September 13-17, 2014. Proceedings*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, pp. 688–699. doi:10.1007/978-3-540-24854-5_71.
- 910 [42] J. Branke, C. Schmidt, Faster convergence by means of fitness estimation, *Soft 9 (1) (2005)* 13–20. doi:10.1007/s00500-003-0329-4.
- [43] G. Sun, G. Li, S. Zhou, W. Xu, X. Yang, Q. Li, Multi-fidelity optimization for sheet metal forming process, *Structural and Multidisciplinary Optimization* 44 (1) (2011) 111–124. doi:10.1007/s00158-010-0596-5.
- 915 [44] S. Ulaganathan, I. Couckuyt, F. Ferranti, E. Laermans, T. Dhaene, Performance study of multi-fidelity gradient enhanced kriging, *Structural and Multidisciplinary Optimization* 51 (5) (2014) 1017–1033. doi:10.1007/s00158-014-1192-x.
- 920 [45] S. Koziel, A. Bekasiewicz, I. Couckuyt, T. Dhaene, Efficient Multi-Objective Simulation-Driven Antenna Design Using Co-Kriging, *IEEE Transactions on Antennas and Propagation* 62 (11) (2014) 5900–5905. doi:10.1109/TAP.2014.2354673.
- [46] M. Kennedy, Predicting the output from a complex computer code when

- 925 fast approximations are available, *Biometrika* 87 (1) (2000) 1–13. doi:
10.1093/biomet/87.1.1.
- [47] A. Forrester, A. Sóbester, A. Keane, Multi-fidelity optimization via surrogate modelling, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science* 463 (2088) (2007) 3251–3269. doi:
930 10.1098/rspa.2007.1900.
- [48] L. Le Gratiet, C. Cannamela, Kriging-based sequential design strategies using fast cross-validation techniques with extensions to multi-fidelity computer codes, *arXiv.org arXiv:1210.6187v2*.
- [49] S. Razavi, B. A. Tolson, D. H. Burn, Review of surrogate modeling in water resources, *Water Resources Research* 48 (7) (2012) 1–32. doi:10.
935 1029/2011WR011527.
- [50] R. Tuo, C. F. J. Wu, D. Yu, Surrogate Modeling of Computer Experiments With Different Mesh Densities, *Technometrics* 56 (3) (2014) 372–380. doi:
10.1080/00401706.2013.842935.
- 940 [51] J. P. C. Kleijnen, *Design and Analysis of Simulation Experiments*, International Series in Operations Research and Management Science, Springer International Publishing, 2015.
- [52] J. Müller, C. A. Shoemaker, Influence of ensemble surrogate models and sampling strategy on the solution quality of algorithms for computationally expensive black-box global optimization problems, *Journal of Global Optimization* 60 (2) (2014) 123–144. doi:10.1007/s10898-014-0184-0.
945
- [53] R. T. Haftka, D. Villanueva, A. Chaudhuri, Parallel surrogate-assisted global optimization with expensive functions – a survey, *Structural and Multidisciplinary Optimization* 54 (1) (2016) 3–13. doi:10.1007/
950 s00158-016-1432-3.
- [54] R. Jin, W. Chen, T. W. Simpson, Comparative studies of meta-modelling techniques under multiple modelling criteria, *Structural and*

Multidisciplinary Optimization 23 (1) (2001) 1–13. doi:10.1007/s00158-001-0160-4.

- 955 [55] L. Shi, K. Rasheed, A Survey of Fitness Approximation Methods Applied in Evolutionary Algorithms, in: Computational Intelligence in Expensive Optimization Problems, Springer, Berlin, Heidelberg, 2010, pp. 3–28. doi:10.1007/978-3-642-10701-6_1.
- [56] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer, Berlin, Heidelberg, New York, 2001.
- 960 [57] K. P. Murphy, Machine learning: a probabilistic perspective, MIT press, 2012.
- [58] L. E. Zerpa, N. V. Queipo, S. Pintos, J.-L. Salager, An optimization methodology of alkaline-surfactant-polymer flooding processes using field scale numerical simulation and multiple surrogates, Journal of Petroleum Science and Engineering 47 (3–4) (2005) 197–208. doi:10.1016/j.petrol.2005.03.002.
- 965 [59] T. Goel, R. T. Haftka, W. Shyy, N. V. Queipo, Ensemble of surrogates, Structural and Multidisciplinary Optimization 33 (3) (2006) 199–216. doi:10.1007/s00158-006-0051-9.
- 970 [60] E. Sanchez, S. Pintos, N. V. Queipo, Toward an Optimal Ensemble of Kernel-based Approximations with Engineering Applications, in: The 2006 IEEE International Joint Conference on Neural Network Proceedings, IEEE, 2006, pp. 2152–2158. doi:10.1109/IJCNN.2006.246987.
- 975 [61] Y. Tenne, S. W. Armfield, A Versatile Surrogate-Assisted Memetic Algorithm for Optimization of Computationally Expensive Functions and its Engineering Applications, in: Success in Evolutionary Computation, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 43–72. doi:10.1007/978-3-540-76286-7_3.

- 980 [62] E. Huang, J. Xu, S. Zhang, C. H. Chen, Multi-fidelity Model Integration for Engineering Design, *Procedia Computer Science* 44 (2015) 336–344. doi:<http://dx.doi.org/10.1016/j.procs.2015.03.002>.
- [63] R. B. Gramacy, tgp: An R Package for Bayesian Nonstationary, Semiparametric Nonlinear Regression and Design by Treed Gaussian Process Models, *Journal of Statistical Software* 19 (9) (2007) 1–46.
985
- [64] A. Nelson, J. Alonso, T. Pulliam, Multi-Fidelity Aerodynamic Optimization Using Treed Meta-Models, in: 25th AIAA Applied Aerodynamics Conference, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2012, pp. 1–19. doi:[10.2514/6.2007-4057](https://doi.org/10.2514/6.2007-4057).
- 990 [65] I. Couckuyt, F. De Turck, T. Dhaene, D. Gorissen, Automatic surrogate model type selection during the optimization of expensive black-box problems, in: 2011 Winter Simulation Conference - (WSC 2011), IEEE, 2011, pp. 4269–4279. doi:[10.1109/WSC.2011.6148114](https://doi.org/10.1109/WSC.2011.6148114).
- [66] I. Voutchkov, A. Keane, A. Bhaskar, T. M. Olsen, Weld sequence optimization: The use of surrogate models for solving sequential combinatorial problems, *Computer Methods in Applied Mechanics and Engineering* 194 (30-33) (2005) 3535–3551.
995
- [67] M. Asadi, J. Goldak, Combinatorial optimization of weld sequence by using a surrogate model to mitigate a weld distortion, *International Journal of Mechanics and Materials in Design* 7 (2) (2011) 123–139.
1000
- [68] C. Teixeira, J. Covas, T. Stützle, A. Gaspar-Cunha, Optimization of co-rotating twin-screw extruders using pareto local search, in: *Advances in Intelligent and Soft Computing*, Springer Science + Business Media, 2010, pp. 3–10. doi:[10.1007/978-3-642-11282-9_1](https://doi.org/10.1007/978-3-642-11282-9_1).
- 1005 [69] C. Teixeira, J. A. Covas, T. Stützle, A. Gaspar-Cunha, Multi-objective ant colony optimization for the twin-screw configuration problem, *Engineering Optimization* 44 (3) (2012) 351–371.

- [70] P. A. Romero, A. Krause, F. H. Arnold, Navigating the protein fitness landscape with Gaussian processes, *Proceedings of the National Academy of Sciences* 110 (3) (2013) E193–E201.
1010
- [71] J. Smith, C. Stone, M. Serpell, Exploiting diverse distance metrics for surrogate-based optimisation of ordering problems: A case study, in: *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference, GECCO '16*, ACM, New York, NY, USA, 2016, pp. 701–708.
1015 doi:10.1145/2908812.2908854.
- [72] T. Bartz-Beielstein, S. Markon, Tuning search algorithms for real-world applications: a regression tree based approach, in: *Proceedings of the 2004 Congress on Evolutionary Computation*, Institute of Electrical and Electronics Engineers (IEEE), Portland, OR, US, 2004, pp. 1111–1118.
1020 doi:10.1109/cec.2004.1330986.
- [73] R. Li, M. T. M. Emmerich, J. Eggermont, E. G. P. Bovenkamp, T. Bäck, J. Dijkstra, J. H. C. Reiber, Metamodel-assisted mixed integer evolution strategies and their application to intravascular ultrasound image analysis, in: *2008 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2008, pp. 2764–2771. doi:10.1109/CEC.2008.4631169.
1025
- [74] F. Hutter, H. H. Hoos, K. Leyton-Brown, Sequential model-based optimization for general algorithm configuration (extended version), *Tech. Rep. TR-2010-10*, University of British Columbia, Department of Computer Science, available online: <http://www.cs.ubc.ca/~hutter/papers/10-TR-SMAC.pdf> (2010).
1030
- [75] L. Bajer, M. Holeňa, Surrogate model for continuous and discrete genetic optimization based on rbf networks, in: *Intelligent Data Engineering and Automated Learning – IDEAL 2010*, Vol. 6283 LNCS, 2010, pp. 251–258.
- [76] M. Herrera, A. Guglielmetti, M. Xiao, R. Filomeno Coelho, Metamodel-assisted optimization based on multiple kernel regression for mixed vari-
1035

- ables, *Structural and Multidisciplinary Optimization* 49 (6) (2014) 979–991.
- [77] S. V. N. Rao, S. Manju, Optimal pumping locations of skimming wells, *Hydrological Sciences Journal* 52 (2) (2007) 352–361.
- 1040 [78] A. Moraglio, A. Kattan, Geometric generalisation of surrogate model based optimisation to combinatorial spaces, in: *Proceedings of the 11th European Conference on Evolutionary Computation in Combinatorial Optimization, EvoCOP’11*, Springer, Berlin, Heidelberg, Germany, 2011, pp. 142–154.
- 1045 [79] A. Moraglio, Y.-H. Kim, Y. Yoon, Geometric surrogate-based optimisation for permutation-based problems, in: *Proceedings of the 13th Annual Conference Companion on Genetic and Evolutionary Computation, GECCO ’11*, ACM, New York, NY, USA, 2011, pp. 133–134.
- [80] M. Zaefferer, J. Stork, M. Friese, A. Fischbach, B. Naujoks, T. Bartz-Beielstein, Efficient global optimization for combinatorial problems, in: *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation, GECCO ’14*, ACM, New York, NY, USA, 2014, pp. 871–878.
- 1050 [81] L. Pérez Cáceres, M. López-Ibáñez, T. Stützle, Ant colony optimization on a limited budget of evaluations, *Swarm Intelligence* (2015) 1–22.
- 1055 [82] A. Moraglio, A. Kattan, Geometric surrogate model based optimisation for genetic programming: Initial experiments, Tech. rep., University of Birmingham (2011).
- [83] T. Hildebrandt, J. Branke, On using surrogates with genetic programming, *Evolutionary Computation* 23 (3) (2015) 343–367.
- 1060 [84] M. Pilát, R. Neruda, Feature extraction for surrogate models in genetic programming, in: *Parallel Problem Solving from Nature – PPSN XIV*, Springer Nature, 2016, pp. 335–344. doi:10.1007/978-3-319-45823-6_31.

- 1065 [85] S. Nguyen, M. Zhang, K. C. Tan, Surrogate-assisted genetic programming with simplified models for automated design of dispatching rules, *IEEE Transactions on Cybernetics* (2016) 1–15doi:10.1109/tcyb.2016.2562674.
- 1070 [86] R. H. Myers, D. C. Montgomery, C. M. Anderson-Cook, *Response surface methodology: process and product optimization using designed experiments*, John Wiley & Sons, 2016.
- [87] J. C. Bean, Genetic algorithms and random keys for sequencing and optimization, *ORSA Journal on Computing* 6 (2) (1994) 154–160. doi:10.1287/ijoc.6.2.154.
- 1075 [88] S. Nguyen, M. Zhang, M. Johnston, K. C. Tan, Selection schemes in surrogate-assisted genetic programming for job shop scheduling, in: *Simulated Evolution and Learning, 10th International Conference, SEAL*, Springer Science + Business Media, 2014, pp. 656–667. doi:10.1007/978-3-319-13563-2_55.
- 1080 [89] L. Fonseca, H. Barbosa, A. Lemonge, A similarity-based surrogate model for expensive evolutionary optimization with fixed budget of simulations, in: *Proceedings of the Congress on Evolutionary Computation (CEC'09)*, IEEE, New York, NY, USA, 2009, pp. 867–874.
- 1085 [90] R. E. Smith, B. A. Dike, S. A. Stegmann, Fitness inheritance in genetic algorithms, in: *Proceedings of the 1995 ACM Symposium on Applied Computing, SAC '95*, ACM, New York, NY, USA, 1995, pp. 345–350.
- [91] H.-S. Kim, S.-B. Cho, An efficient genetic algorithm with less fitness evaluation by clustering, in: *Proceedings of the Congress on Evolutionary Computation (CEC'01)*, Vol. 2, IEEE, New York, NY, USA, 2001, pp. 887–894.
- 1090 [92] N. S. Altman, An introduction to kernel and nearest-neighbor nonparametric regression, *The American Statistician* 46 (3) (1992) 175–185.

- [93] H. S. Bernardino, L. G. Fonseca, H. J. C. Barbosa, Surrogate-assisted artificial immune systems for expensive optimization problems, in: W. P. dos Santos (Ed.), *Evolutionary Computation*, InTech, 2009, Ch. 10, pp. 179–198.
- 1095
- [94] T. Hemker, Derivative free surrogate optimization for mixed-integer non-linear black box problems in engineering, Ph.D. thesis, Technische Universität Darmstadt (December 2008).
- [95] A. E. Brownlee, J. A. Wright, Constrained, mixed-integer and multi-objective optimisation of building designs by nsga-ii with fitness approximation, *Applied Soft Computing* 33 (2015) 114–126.
- 1100
- [96] L. P. Swiler, P. D. Hough, P. Qian, X. Xu, C. Storlie, H. Lee, Surrogate models for mixed discrete-continuous variables, in: M. Ceberio, V. Kreinovich (Eds.), *Constraint Programming and Decision Making*, Springer International Publishing, Cham, 2014, Ch. Surrogate Models for Mixed Discrete-Continuous Variables, pp. 181–202. doi:10.1007/978-3-319-04280-0_21.
- 1105
- [97] R. Filomeno Coelho, M. Herrera, M. Xiao, W. Zhang, On-line metamodel-assisted optimization with mixed variables, in: J. Magalhães-Mendes, D. Greiner (Eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*, Vol. 39 of *Computational Methods in Applied Sciences*, Springer International Publishing, 2015, pp. 1–15.
- 1110
- [98] M. Zaefferer, J. Stork, T. Bartz-Beielstein, Distance measures for permutations in combinatorial efficient global optimization, in: T. Bartz-Beielstein, J. Branke, B. Filipič, J. Smith (Eds.), *Parallel Problem Solving from Nature–PPSN XIII*, Springer, Cham, Switzerland, 2014, pp. 373–383.
- 1115
- [99] M. Zaefferer, T. Bartz-Beielstein, Efficient global optimization with indefinite kernels, in: *Parallel Problem Solving from Nature–PPSN XIV*, Springer, 2016, pp. 69–79.
- 1120

- [100] F.-M. Schleif, P. Tino, Indefinite proximity learning: A review, *Neural Computation* 27 (10) (2015) 2039–2096.
- [101] T. Gärtner, A survey of kernels for structured data, *SIGKDD Explor. Newsl.* 5 (1) (2003) 49–58.
- 1125 [102] T. Gärtner, J. Lloyd, P. Flach, Kernels and distances for structured data, *Machine Learning* 57 (3) (2004) 205–232.
- [103] D. Haussler, Convolution kernels on discrete structures, Tech. Rep. UCSC-CRL-99-10, Department of Computer Science, University of California at Santa Cruz (1999).
- 1130 [104] S. Shakya, Deum: A framework for an estimation of distribution algorithm based on markov random fields, Ph.D. thesis, School of Computing, Faculty of Design and Technology, The Robert Gordon University, Aberdeen, UK (2006).
- [105] A. Brownlee, J. McCall, Q. Zhang, Fitness modeling with markov networks, *IEEE Transactions on Evolutionary Computation* 17 (6) (2013) 862–879.
- 1135 [106] A. E. Brownlee, Mining Markov network surrogates for value-added optimisation, in: *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion - GECCO 16 Companion*, Association for Computing Machinery (ACM), 2016, pp. 1267–1274. doi:10.1145/2908961.2931711.
- 1140 [107] S. Jiang, A. Ziver, J. Carter, C. Pain, A. Goddard, S. Franklin, H. Phillips, Estimation of distribution algorithms for nuclear reactor fuel management optimisation, *Annals of Nuclear Energy* 33 (11–12) (2006) 1039–1057. doi:<http://dx.doi.org/10.1016/j.anucene.2006.03.012>.
- 1145 [108] J. Ceberio, E. Irurozki, A. Mendiburu, J. A. Lozano, A review on estimation of distribution algorithms in permutation-based combinatorial opti-

- mization problems, *Progress in Artificial Intelligence* 1 (1) (2012) 103–117. doi:10.1007/s13748-011-0005-3.
- 1150 [109] C. L. Mallows, Non-null ranking models, *Biometrika* 44 (1–2) (1957) 114–130.
- [110] J. Ceberio, A. Mendiburu, J. A. Lozano, Introducing the mallows model on estimation of distribution algorithms, in: *Neural Information Processing - 18th International Conference, ICONIP 2011, Springer Berlin Heidelberg*, 2011, pp. 461–470. doi:10.1007/978-3-642-24958-7_54.
- 1155 [111] J. Ceberio, E. Irurozki, A. Mendiburu, J. A. Lozano, A distance-based ranking model estimation of distribution algorithm for the flowshop scheduling problem, *IEEE Transactions on Evolutionary Computation* 18 (2) (2014) 286–300. doi:10.1109/TEVC.2013.2260548.
- 1160 [112] R. Allmendinger, C. A. C. Coello, M. T. M. Emmerich, J. Hakanen, Y. Jin, E. Rigoni, Surrogate-assisted multicriteria optimization (wg6), Tech. Rep. 1, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2015).
- [113] M. Pelikan, D. E. Goldberg, E. Cantu-Paz, BOA: The bayesian optimization algorithm, in: W. Banzhaf, et al. (Eds.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO’99)*, Morgan Kaufmann, 1999, pp. 525–532.
- 1165 [114] P. Larrañaga, R. Etxeberria, J. A. Lozano, J. M. Peña, Combinatorial optimization by learning and simulation of bayesian networks, in: *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence, UAI’00*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000, pp. 343–352.
- 1170 [115] M. N. Le, Y. S. Ong, S. Menzel, Y. Jin, B. Sendhoff, Evolution by adapting surrogates, *Evolutionary Computation* 21 (2) (2013) 313–340. doi:10.1162/evco_a_00079.
- 1175

- [116] M. Kryszkiewicz, J. F. Peters, H. Rybinski, A. Skowron (Eds.), *Rough Sets and Intelligent Systems Paradigms*, Vol. 4585 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. doi:10.1007/978-3-540-73451-2.
- 1180 [117] T. Bartz-Beielstein, *Stacked Generalization of Surrogate Models - A Practical Approach*, Tech. Rep. 5/2016, TH Köln, Köln (2016).
- [118] F. A. Viana, R. T. Haftka, L. T. Watson, *Why not run the efficient global optimization algorithm with multiple surrogates?*, in: *51st AIAAAS-MEASCEAHSASC Structures Structural Dynamics and Materials Conference*, no. April, AIAA, 2010, pp. AIAA-2010.
- 1185 [119] N. M. Alexandrov, J. E. Dennis Jr, R. M. Lewis, V. Torczon, *A trust-region framework for managing the use of approximation models in optimization*, *Structural Optimization* 15 (1) (1998) 16–23. doi:10.1007/BF01197433.
- 1190 [120] Y. S. Ong, P. B. Nair, A. J. Keane, *Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling*, *AIAA Journal* 41 (4) (2003) 687–696. doi:10.2514/2.1999.
- [121] T. Bartz-Beielstein, *Experimental Analysis of Evolution Strategies—Overview and Comprehensive Introduction*, Reihe CI. SFB 531 157/03, University Dortmund (Nov. 2003).
- 1195 [122] T. Bartz-Beielstein, K. E. Parsopoulos, M. N. Vrahatis, *Design and analysis of optimization algorithms using computational statistics*, *Applied Numerical Analysis and Computational Mathematics (ANACM)* 1 (2) (2004) 413–433.
- 1200 [123] T. Bartz-Beielstein, C. Lasarczyk, M. Preuss, *Sequential Parameter Optimization*, in: B. McKay, et al. (Eds.), *Proceedings 2005 Congress on Evolutionary Computation (CEC'05)*, Edinburgh, Scotland, IEEE Press, Piscataway NJ, 2005, pp. 773–780. doi:10.1109/CEC.2005.1554761.

- [124] F. Hutter, H. H. Hoos, K. Leyton-Brown, An Evaluation of Sequential
1205 Model-based Optimization for Expensive Blackbox Functions, in: Proceedings of the 15th Annual Conference Companion on Genetic and Evolutionary Computation, ACM, New York, NY, USA, 2013, pp. 1209–1216. doi:10.1145/2464576.2501592.
- [125] N. Hansen, S. Finck, R. Ros, A. Auger, Real-Parameter Black-Box Optimization Benchmarking 2009: Noiseless Functions Definitions, Research
1210 Report RR-6829, INRIA (2009).
- [126] M. J. van der Laan, S. Dudoit, Unified cross-validation methodology for selection among estimators and a general cross-validated adaptive epsilon-net estimator: Finite sample oracle inequalities and examples, Working
1215 paper 130, University of California, Berkeley, Division of Biostatistics, School of Public Health, University of California, Berkeley (2003).
- [127] D. H. Wolpert, Stacked generalization, *Neural Networks* 5 (2) (1992) 241–259. doi:10.1016/S0893-6080(05)80023-1.
- [128] M. A. Rebolledo Coy, S. Krey, T. Bartz-Beielstein, O. Flasch, A. Fischbach, J. Stork, Modeling and optimization of a robust gas sensor,
1220 *Cologne Open Science* 03/2016, TH Köln, Cologne (2016).
- [129] D. Douguet, e-lea3d: a computational-aided drug design web server, *Nucleic Acids Research* arXiv:<http://nar.oxfordjournals.org/content/early/2010/05/05/nar.gkq322.full.pdf+html>,
1225 doi:10.1093/nar/gkq322.
- [130] C. C. Aggarwal, A. Hinneburg, D. A. Keim, On the surprising behavior of distance metrics in high dimensional space, in: *Database Theory — ICDT 2001*, Vol. 3, Springer Science + Business Media, Munich, 2001, pp. 420–434.
- 1230 [131] A. G. Wilson, C. Dann, H. Nickisch, Thoughts on Massively Scalable Gaussian Processes, arXiv.org arXiv:1511.01870v1.

- 1235 [132] T. Bartz-Beielstein, How to Create Generalizable Results, in: J. Kacprzyk, W. Pedrycz (Eds.), Springer Handbook of Computational Intelligence, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015, pp. 1127–1142. doi: 10.1007/978-3-662-43505-2_56.
- 1240 [133] A. Isaacs, T. Ray, W. Smith, An evolutionary algorithm with spatially distributed surrogates for multiobjective optimization, in: M. Randall, H. A. Abbass, J. Wiles (Eds.), Progress in Artificial Life: Third Australian Conference; ACAL 2007 Gold Coast, Australia, December 4-6, 2007 Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 257–268. doi:10.1007/978-3-540-76931-6_23.
- 1245 [134] M. Pilát, R. Neruda, Aggregate meta-models for evolutionary multiobjective and many-objective optimization, Neurocomputing 116 (2013) 392–402, advanced Theory and Methodology in Intelligent Computing – Selected Papers from the Seventh International Conference on Intelligent Computing (ICIC 2011). doi:http://dx.doi.org/10.1016/j.neucom.2012.06.043.
- 1250 [135] D. Horn, T. Wagner, D. Biermann, C. Weihs, B. Bischl, Model-based multi-objective optimization: Taxonomy, multi-point proposal, toolbox and benchmark, in: A. Gaspar-Cunha, C. Henggeler Antunes, C. C. Coello (Eds.), Evolutionary Multi-Criterion Optimization: 8th International Conference, EMO 2015, Guimarães, Portugal, March 29 –April 1, 2015. Proceedings, Part I, Springer International Publishing, Cham, 2015, pp. 64–78. doi:10.1007/978-3-319-15934-8_5.

Kontakt/Impressum

Diese Veröffentlichungen erscheinen im Rahmen der Schriftenreihe "Ciplus". Alle Veröffentlichungen dieser Reihe können unter <https://cos.bibl.th-koeln.de/home> abgerufen werden.

Köln, Januar 2012

Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.
Datum der Veröffentlichung: 14.11.2016

Herausgeber / Editorship

Prof. Dr. Thomas Bartz-Beielstein,
Prof. Dr. Wolfgang Konen,
Prof. Dr. Boris Naujoks,
Prof. Dr. Horst Stenzel
Institute of Computer Science,
Faculty of Computer Science and Engineering Science,
TH Köln,
Steinmüllerallee 1,
51643 Gummersbach
url: www.ciplus-research.de

Schriftleitung und Ansprechpartner/ Contact editor's office

Prof. Dr. Thomas Bartz-Beielstein,
Institute of Computer Science,
Faculty of Computer Science and Engineering Science,
TH Köln,
Steinmüllerallee 1, 51643 Gummersbach
phone: +49 2261 8196 6391
url: <http://www.spotseven.de>
eMail: thomas.bartz-beielstein@th-koeln.de

ISSN (online) 2194-2870

**Technology
Arts Sciences
TH Köln**

Supported by:



on the basis of a decision
by the German Bundestag

Grant No. KF3145101WM3 and KF3145103WM4



This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 692286.

Technology
Arts Sciences
TH Köln