INDUSTRIAL APPLICATION PAPER

Spline-based shape optimization of large-scale composite leaf spring models using Bayesian strategies with multiple constraints

Jens Winter[1] · Sierk Fiebig[2] · Thilo Franke[2] · Ronald Bartz[2] · Thomas Vietor[1]

Received: 26 November 2021 / Revised: 30 March 2022 / Accepted: 11 July 2022 / Published online: 6 September 2022

Abstract
The presented paper describes a shape optimization workflow using Bayesian strategies. It is applied to a novel automotive axle system consisting of leaf springs made from glass fiber reinforced plastics (GFRP). Besides the primary objectives of cost and mass reduction, the assembly has to meet multiple technical constraints with respect to various loading conditions. The related large-scale finite element model is fully parameterized by splines, hence the general shape of the guide curve as well as the spring's height, width and material properties can be altered by the corresponding workflow. For this purpose, a novel method is developed to automatically generate high-quality meshes depending on the geometry of the respective springs. The size and complexity of the model demands the implementation of efficient optimization techniques with a preferably small number of required response function evaluations. Therefore, an existing optimization framework is extended by state-of-the-art Bayesian methods, including different kernel combinations and multiple acquisition function approaches, which are then tested, evaluated and compared. To properly address the use of GFRP as spring material in the objective function, an appropriate cost model is derived. Emerging challenges, such as conflicting targets regarding direct material costs and potential lightweight measures, are considered and investigated. The intermediate steps of the developed optimization procedure are tested on various sample functions and simplified models. The entire workflow is finally applied to the complete model and evaluated. Concluding, ideas and possibilities in improving the optimization process, such as the use of models with varying complexity, are discussed.

Keywords Bayesian optimization · Surrogate models · Shape optimization · Gaussian processes

1 Introduction

The following sections briefly introduce the presented industrial application example and relevant references describing similar optimization problems. Furthermore, optimization methods that are potentially suitable for solving the given problem are introduced. Relevant literature regarding these methods are reviewed and evaluated.

1.1 Leaf spring optimization

The starting point of the problem on which this optimization workflow is based on is the development of a novel axle system. It essentially consists of four glass fiber reinforced leaf springs, manufactured in a corresponding resin transfer molding process (see Fig. 1). The system is applied in an automotive context in the area of chassis assemblies and offers the chance of significant cost and weight reduction, speaking of approximately 7 kg per vehicle (Kersten and Fiebig 2019). All involved components have to cope with various loading conditions and the system as a whole has to meet multiple specific technical constraints, concerning strength and driving dynamics. To achieve these characteristic values, the guide curve of the springs, their rectangular cross-sections, as well as their material parameters have to be altered in the demanded optimization process. The corresponding finite element model to simulate those loading conditions will be automatically generated by the workflow

Responsible Editor: Felipe A. C. Viana

✉ Jens Winter
jens.winter@tu-braunschweig.de

[1] Institute for Engineering Design, Technische Universität Braunschweig, 38108 Brunswick, Germany

[2] Volkswagen AG, 38112 Brunswick, Germany

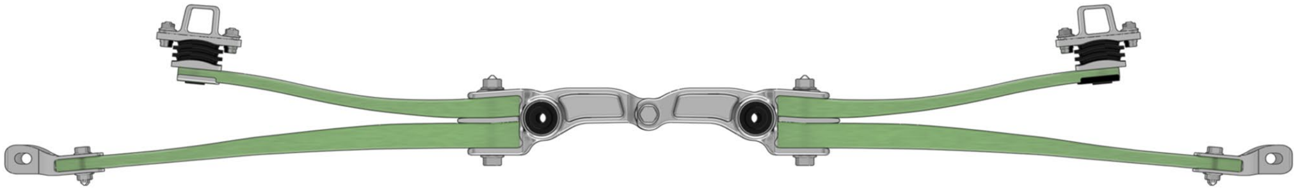

Fig. 1 Application example of glass fiber reinforced leaf springs in an automotive chassis assembly in order to achieve weight reduction (Kersten and Fiebig 2019)

itself, guaranteeing an appropriate hexahedral dominant mesh.

Existing works in the field of optimizing leaf springs are using a broad range of different optimization types and algorithms. The main objective here is the minimization of spring weights and the associated reduction in component costs, in most cases by applying new materials. Comparably basic studies are evaluating pure material substitutions while not changing the geometry of the spring at all (Ghag et al. 2018). The consideration of using composite fiber materials usually involves a redesign of the conventional spring design, but can further reduce the components weight by up to 90% (Kumar et al. 2012). Attempts in optimizing the actual geometry of springs are mostly sizing optimizations by changing the global dimensions of the spring's cross-section (Kumar et al. 2011; Rajendran and Vijayarangan 2001). By changing cross-section dimensions locally, optimization processes yield more sophisticated designs, such as a hyperbolically decreasing spring width (Shokrieh and Rezaei 2003). Finally, studies on combining different optimization types to create high-end composite leaf springs are using topology optimization as well as sizing and parameter optimization methods (Gaylo et al. 2019).

Despite the fact that there is a wide range of work on the optimization of leaf springs, none of them seem suitable for the problem at hand. Material substitution and size optimization will not be sufficient to meet the complex driving dynamic constraints of the system. The inclusion of topology optimization methods is not considered reasonable since the topology of the springs is in general already fixed. Therefore, a novel method has to be developed to modify the guide curve, the cross-sectional dimensions, and the material parameters of the spring while maintaining an adequate mesh quality.

1.2 Gaussian processes

Since the mechanical behavior of the spring system is highly nonlinear regarding geometrical and material parameters, the respective simulations are quite time-consuming. Therefore, and primarily to save license and calculation costs, the number of function evaluations have to be kept preferably low. Moreover, the non-linearity of the given model leads to a comparatively complex and inconvenient calculation of gradient information, which is therefore considered as too expensive. Optimizing such large-scale, nonlinear, non-differentiable, black-box functions is still subject of current research (Elhara et al. 2019; Al-Dujaili and Suresh 2017; Regis 2020). One way to address these kind of problems is the application of surrogate models (Barton and Meckesheimer 2006). By trying to approximate actual functions as simply and accurately as possible and by constantly readjusting those models, an appropriate optimization algorithm strives to achieve a maximum gain of knowledge using existing information toward further iterations and potential optima. In this way, optimization problems can be treated in a global manner and are less localized as in other methods (Egea et al. 2007; McDonald et al. 2007). Enabling, developing and enhancing optimization processes using surrogate models has been the object of extensive studies in the past years. The efficient use of existing information to approximate an objective function, as well as the low computational cost of evaluating such surrogate models, have increased their popularity.

Fostered by novel methods of emerging research areas, such as machine learning or artificial intelligence, especially optimization strategies using approaches from Bayesian statistics have been revived and extended (Shahriari et al. 2016). These strategies aim to localize optimal points with the aid of Gaussian processes, which do not only approximate functions but can also quantify the uncertainty of the model (Rasmussen and Williams 2006). Besides the Gaussian processes implemented in this work, other surrogate model approaches are often examined and enhanced. Popular models are radial-basis functions (Song et al. 2019) and support vector regression (Shi et al. 2020), which are relatively performant and precise. There are also studies evaluating neural networks (Springenberg et al. 2016; Alam et al. 2004) or even splines (Turner and Crawford 2008). For the presented problem, Gaussian processes were chosen for several reasons. The possibility of choosing different kernels and optimizing hyperparameters of a GP provides a great variability and expandability (Kim and Teh 2018; Duvenaud et al. 2013). This makes the method itself versatile and applicable to many different kinds of problems. Also, the estimation of uncertainty in the prediction helps picking new evaluation points and balancing exploration against exploitation. This can lead to accelerated convergence of the optimization process and helps to approach optimization problems in a more global manner. The main disadvantages of Gaussian processes, on the other hand, their high computational complexity of $\mathcal{O}(n^3)$ and their difficulties in higher dimensions, do not arise in the problem at hand (Yetilmezsoy et al. 2021; Akbari et al. 2019; Singh et al. 2021).

1.3 Bayesian optimization

The so-called Bayesian optimization is intended to identify new and promising evaluation points on a Gaussian process and thus to guide the building of the surrogate model on the one hand, but also to find optimal solutions on the other hand. This is done by choosing successive points, balancing between potentially optimal points (exploitation) and areas of the model where uncertainty is comparably high (exploration) (Frazier 2018). Therefore, Bayesian methods basically put a lot of effort into choosing the next point to evaluate in order to keep the number of required iterations low. This can have a particularly favorable effect on time-consuming function evaluations and thus on the presented problem of optimizing the leaf spring system. One of the first works toward modern Bayesian optimization is the so-called EGO algorithm (efficient global optimization, Jones et al. 1998). Due to these and following studies, the application of Bayesian strategies in conjunction with Kriging models or Gaussian processes to describe complex objective functions of black-box models is a preferred choice (Viana et al. 2012; Basudhar et al. 2012). Especially, when it comes to expensive function evaluations with incomputable derivatives and unknown convexity properties, Bayesian optimization proves its advantages against other methods, such as L-BFGS or best09 (Riche and Picheny 2021; Diouane et al. 2021). Additionally, its ability in balancing exploration and exploitation against each other during the optimization process, makes it an appropriate approach for structural shape optimization problems (Zacchei and Molina 2018; Ghosh et al. 2019; Dominguez et al. 2017).

Recent works addressing methods of Bayesian optimization are primarily aiming at improving and extending existing approaches with new ideas and algorithms. The enormous number of parameters and settings in Bayesian methods provides a high variability. This provides the opportunity to adapt and to specialize onto different optimization problems, which can be an advantage regarding the *no free lunch* theorem (Wolpert and Macready 1996). However, this also requires in-depth knowledge and profound user experience. Hence, there is a wide range of studies providing this knowledge by testing and comparing different methods against each other, for example, a comparison of different universal kriging methods within the EGO framework (Palar and Shimoyama 2018). On the other hand, recent approaches aim to reduce required user knowledge by determining appropriate model types and hyperparameters based on prior data and adjusting those parameters adaptively. In this way, the methods work more efficiently while the required user knowledge decreases (Mehmani et al. 2018).

1.4 Acquisition functions

In order to complete the workflow of Bayesian optimization, an acquisition function has to be derived from the previously described Gaussian process. In maximizing this function, the evaluation point for the following iteration can be determined. Because of its significant influence on the optimization process, a large variety of methods exist. Some of the firstly developed acquisition functions are still very prevalent. This is due to their straightforward principle and due to the fact, that they are quite effective in a large number of cases. The *probability of improvement* (PI, Kushner 1964) for example suggests the point in the given design space, which is most likely to achieve an improvement of the function value. Unfortunately, it does not evaluate how large this improvement will be. The approach of *expected improvement* (EI, Mockus et al. 1978), however, can lead to better results, since it also evaluates the size of a possible improvement. Furthermore, the so-called *scaled expected improvement* extends the common EI by proposing evaluation points with high EI at locations with a simultaneously high confidence (Noè and Husmeier 2018). Although these function do have parameters that can be used to attach more weight in exploring the design space, their focus is mostly exploitation. More exploration-based approaches would be the *upper confidence bound* (UCB, Srinivas et al. 2010) or the *entropy search* (ES, Hennig and Schuler 2012). Furthermore, the method of *predictive entropy search* (PES, Hernandéz-Lobato et al. 2014) as well as the method of *max-value entropy search* (MES, Wang and Jegelka 2018) showed promising results in extending entropy-based approaches.

For the presented optimization problem, an adjusted EI algorithm called *top-two expected improvement* will be implemented (Qin et al. 2017). Instead of picking only the best point found during the acquisition function optimization, the two best points are determined. With a defined probability, the first or the second point will be chosen for simulation and evaluation. This leads to a more exploratory behavior in methods that are mainly driven by exploitation.

1.5 Consideration of constraints

In order to optimize a given problem including constraints, each of these constraints is represented and modeled individually using a corresponding GP throughout the optimization process. This way, for each GP, a respective acquisition function can be calculated. These can then be combined to form an overall acquisition function, whose maximum will be the evaluation point in the following iteration. Most existing approaches aim to modify established acquisition functions in a way that excludes all invalid regions according to the current constraint surrogate model. Constrained Bayesian optimization strategies based on EI are quite common, although most of these methods can basically be applied to various acquisition function approaches (Schonlau et al. 1998; Gelbart et al. 2014; Gardner et al. 2014). However, a relevant drawback of these basic techniques is their inability to evaluate and differentiate the invalidity of parameter sets. Attempts in overcoming these problems by using information provided by the corresponding GP in infeasible regions is a subject of current research (Picheny 2014; Gramacy and Lee 2010). Furthermore, alternative concepts have emerged in recent years, in particular, lookahead algorithms (Lam et al. 2018), information-based approaches (Hernandéz-Lobato et al. 2015, 2016) or the application of augmented Lagrangian methods (Gramacy et al. 2016; Gramacy and Lee 2010).

The multidimensional design space for the given problem is expected to be infeasible in most regions. Thus, the *probability of feasibility* (Schonlau 1997) approach is chosen for the constraint functions, whereas the above described top-two EI method will be used for the objective function. This will lead to a combined acquisition function optimization which tries to maximize all single functions simultaneously and in an equivalent way. Comparable to a multi-objective optimization, the approach aims to find the most feasible points in design space. In addition, a weighting of different output values can be applied easily.

2 Contribution

The work presented provides a framework for setting up and running spline-based shape optimizations using Bayesian approaches. The application example in this case is a rather extensive one, considering different computationally intensive load cases, nonlinear behavior and technical constraints, such as component strength and various driving dynamic parameters. Since the corresponding simulation model uses a 3D solid composite mesh to represent the GFRP leaf springs, a novel method for creating this mesh is developed. This includes an extended parameterization approach based on non-uniform rational B-splines (NURBSs) to be able to represent and modify the geometry and dimensions of the springs. In addition, an adaptive meshing approach is implemented, which ensures constant element and mesh quality regardless of the dimensions of the current spring. In order to be able to consider different strategies during the product development process of components made from composite materials, an appropriate cost model is derived. It enables the user to decide whether the focus of optimization should be on direct material costs or on lightweight design and thus the component mass. Based on the given application example, it is expected that the objective function, which depends on several output variables, will be multimodal and invalid in many areas. To model the objective function with sufficient accuracy, a wide variety of kernels were tested prior to the actual optimization. The prediction quality of the resulting models was then evaluated and compared with each other using the R^2-predict value. The challenges described here and their proposed solutions are detailed in the following sections.

Following the *Three-Columns-Concept* of structural optimization by Eschenauer (1989), the upcoming sections are divided in the elaboration of structural model, optimization model and optimization algorithm. Section 3 introduces the application example as well as the simulation model. The optimization model, including the formulation of an objective cost function and constraints are described in Sect. 4. In the following Sect. 5, the Bayesian approach is introduced with respect to the given problem. The implementation and choice of various kernels and acquisition functions are described, as well as further configuration options to control the optimization process. Results and performance of trial optimization runs on the given problem are evaluated and compared to each other in Sect. 6. Concluding, Sect. 7 summarizes the outcomes and provides an outlook on possible following research aspects.

3 Structural model

As described in Sect. 1, the application of leaf springs in a novel rear axle assembly will serve as generic example for the introduced optimization method. It represents the problem of optimizing systems which are, compared to academic optimization tasks, rather complex and therefore computationally intensive. Further areas of application would be, for example, the evaluation of crashworthiness (Shi et al. 2012) or the examination of problems regarding fluid dynamics (Park et al. 2018). The upcoming section outlines the simulation model and its characteristics, followed by a brief introduction of the parameterization of the structural model.

3.1 Simulation model

The given leaf spring axle system mainly consists of two halves of nearly symmetric assemblies in *XZ* plane. Each of these contain two leaf springs, held by an aluminum casted hinge. The whole assembly is mounted at six bearing points (see Fig. 2). Point **a** is directly connected to the body of the vehicle, while point **b** is attached to the wheel carrier.

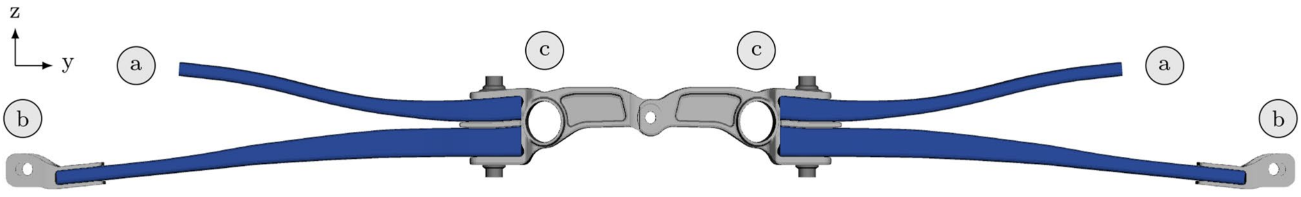

Fig. 2 Simulation model of the rear axle assembly consisting of four GFRP leaf springs (blue) and an aluminum casted hinge (gray). The system is connected to the body of the vehicle in point **a**, to the wheel carrier in point **b** and to the subframe in point **c**

Point **c** consists of a rubber bearing, which is connected to the vehicle's subframe. Relevant load cases regarding the assessment of the spring system evaluate, on the one hand, the components' strength using the Puck failure criterion for fiber reinforced composites (Puck 1996). On the other hand, the axle system has to fulfill several driving dynamic parameters, which are for example the total spring rate, the toe-in angle or the camber angle of the system. These values have to be as close as possible to the corresponding target values, in order to achieve an appropriate driveability. Overall the validation of the system's performance is composed of nine kinematic simulations and five misuse load cases. These load cases lead to nonlinear behavior of various types. Above all, plastic material behavior as a result of high deformation should be mentioned here. Furthermore, bolted and clamped connections and the formulation of the corresponding contacts cause additional non-linearities. Finally, the use of nonlinear spring and damper elements also contributes to the complexity of the model. Taking all these types of non-linearities into account, the calculation of objective and constraint function gradients is considered to be too time-consuming in terms of implementation and iteration duration.

Due to their comparably voluminous geometry, modeling the springs in a shell-based manner does not lead to the required quality in results. This applies to both the driving dynamics as well as to strength values. Furthermore information regarding the exact location of weak spots would be lost in a shell model. Therefore, a 3D model made from solid elements is necessary for the given problem. For the original model, this results in a simulation with approximately 2.8 million degrees of freedom and a normalized CPU time of about 60 h on a high-performance cluster using 96 CPUs. In an attempt to speed up the simulations, a coarser mesh was used for areas which are not part of the current evaluation and were furthermore substituted by surrogate elements, if possible. Due to these measures, the number of degrees of freedom were reduced to 800,000, resulting in a normalized CPU time of approximately 12 h on the same high-performance computer (HPC). Assuming an optimization with 50 function evaluations, this would still lead to an overall duration of nearly a month. Additional ways to further speed up the optimization process would be the parallelization of multiple designs on different nodes of an HPC cluster. However, this requires that the optimizer is able to generate several promising designs per iteration. In most cases, an optimization-based product development process is faster than looping manually between design and simulation phases. Although, the goal should be to further reduce development costs and therefore minimizing the duration of an optimization. Preferred measures to achieve this is, on the one hand, by speeding up function evaluations or, on the other hand, by improving the optimization algorithm.

3.2 Parameterization

As depicted in Fig. 3 different design stages foster different optimization types. As the fundamental topology of the leaf springs is already determined and is furthermore limited by the applied manufacturing process, topology optimization methods are not appropriate in this case. In contrast, a sizing optimization to meet the demanded targets regarding driving dynamics and strength is simply not sufficient. Concluding, shape optimization methods seem to be most convenient for the given problem. Moreover, looking into the different shape optimization methods, three types of parameterization can be taken into account according to Bletzinger (2017): node-based,

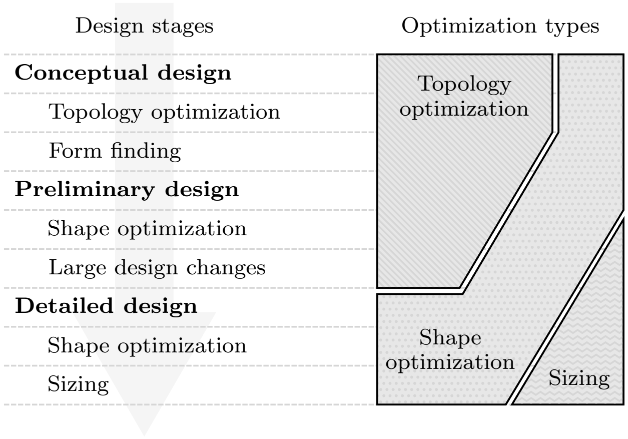

Fig. 3 Different optimization types allocated to different stages of design according to Bletzinger (2017)

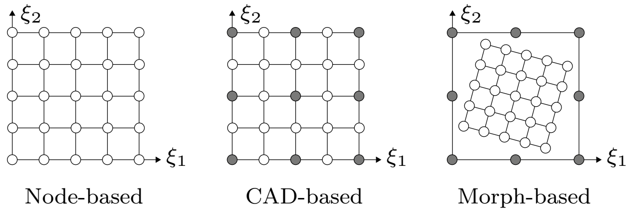

Fig. 4 Shape optimization approaches distinguishing between design (gray nodes) and analysis (white nodes) grid according to Bletzinger (2017)

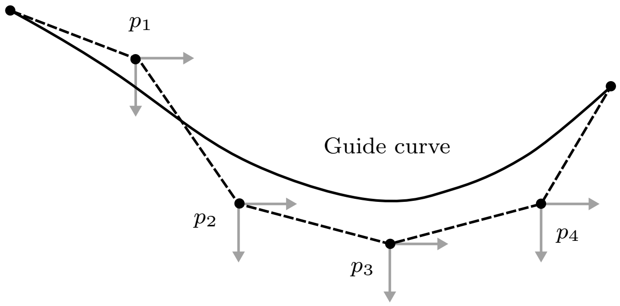

Fig. 5 Parameterization of the leaf spring model by defining a NURBS curve, which represents the guide curve of the component

CAD-based or morph-based approaches (see Fig. 4). The applied discretization method for the model uses a different number of elements and fiber layers depending on the current height of the spring. Since this requires the removal and insertion of complete layers of elements, both, a morph- or a node-based parameterization would be not applicable for the given scenario. The CAD-based shape optimization, however, seem to fit well for the present problem.

As the process of parameterization and automated model generation is already described in detail in a preceding conference article (Winter et al. 2019), this section will only briefly outline the applied methods. The principal shape of the leaf spring can be described by a guide curve, along which a corresponding cross-section is extruded. This curve's geometry will be expressed by a NURBS (see Fig. 5). To additionally influence the height and the width of the component locally and simultaneously keep the number of optimization parameters low, the common control points (composed of the three spatial coordinates and the control point weight) are extended in the following way:

$$p_i = \begin{pmatrix} x_i \\ y_i \\ z_i \\ \omega_i \\ h_i \\ w_i \end{pmatrix} = \begin{pmatrix} \text{Control point coordinate } X \\ \text{Control point coordinate } Y \\ \text{Control point coordinate } Z \\ \text{Control point weight} \\ \text{Section height} \\ \text{Section width} \end{pmatrix}. \tag{1}$$

The number of control points of each spring curve has a direct influence on the shape of the spring itself and, of course, on the number of input parameters for the optimization. In principle, any number of control points can be specified in the optimization workflow. Though, using too few control points would limit the design freedom given by the method. Apart from that, using too many control points would complicate the optimization process. Since this would also lead to less smooth spring designs in many cases and therefore to low strength values, six control points were chosen for the optimization problem at hand, as shown in Fig. 5. Besides the parameters per NURBS control point, the global fiber volume content (FVC) of each spring can be altered during the optimization. Theoretically, changing the fiber orientation per layer would affect the mechanical behavior of the system. Although, the parameter of fiber orientation has no direct effect on the mass of the spring and, if any, a negative influence on the manufacturing costs. However, it must be taken into account that a change in fiber orientation entails a degradation of the mechanical behavior. This in turn requires an increase in the size of the spring and thus indirectly does have an influence on the mass and cost of the spring. Beyond that, orienting the fibers along the spring's guide curve will result in the optimal mechanical performance due to the given load types and directions. Hence, fiber orientation will for now be disregarded.

After generating the guide curve based on the given design variables, the corresponding solid mesh can be created. To ensure a model of sufficient quality, equidistant points along the spline has to be calculated using a method based on the *regula falsi* (Bronstein et al. 2007). In a next step, local coordinate systems are created in each of these points. Using the tangent of the curve as local X direction, local Y and Z directions are calculated via the method of rotation-minimizing frames (Wang et al. 2008). Following, the FE nodes from the starting non-design cross-section of the spring are copied in every local coordinate system using translation and rotation operations. To fit the respective nodes to the local height of the spring given by the design variables of the optimization, they are scaled in width (Y) and height (Z). If a tolerance, which is based on the element size of the non-design areas, gets violated by this scaling, a node row will be inserted or removed. As a result of this, either a new layer of elements will be generated or an existing layer will be closed. A depiction of different stages of the above described model generation is shown in Fig. 6.

4 Optimization model

The optimization model is defined as the mathematical formulation of the optimization task itself. It links the analysis variables of the structural model and therefore

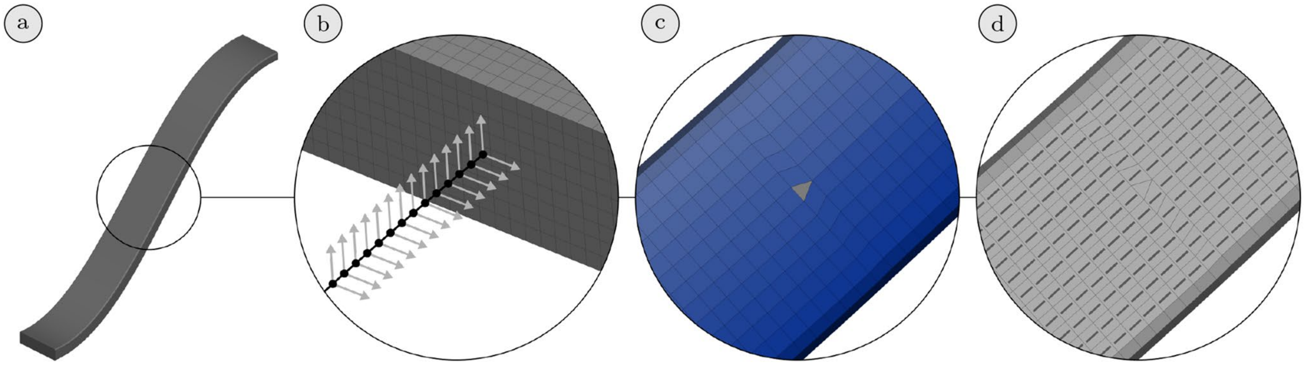

Fig. 6 Automated finite element model creation of the leaf spring (**a**), starting with the parameterized spline and corresponding local coordinate systems (**b**), generation of elements (**c**) and calculation of fiber orientation vectors (**d**)

the engineering problem with the design variables and parameters of the optimization algorithm. First of all, this includes the objective function of the optimization, as well as implicit and explicit constraints that have to be satisfied.

4.1 Objective function

In most cases, structural optimization seeks to minimize the mass of a given component in compliance with technical constraints, such as stress or strength. In contrast, for the given application example a minimization of the component's cost is aspired:

$$\textbf{minimize}(C_{\mathrm{S}}(\bar{x})). \tag{2}$$

Here, C_{S} denotes the overall costs of the current design, subject to all design variables. Usually minimizing costs is synonymous to minimizing mass. For the given application example of the leaf spring this is not necessarily the case. The mass of a spring m_{S} is calculated as the sum of the mass of the fiber portion m_{F} and the mass of the matrix portion m_{M} of the spring:

$$m_{\mathrm{S}} = m_{\mathrm{F}} + m_{\mathrm{M}} = V_{\mathrm{S}} \cdot (\varphi_{\mathrm{S}} \cdot \rho_{\mathrm{F}} + (1 - \varphi_{\mathrm{S}}) \cdot \rho_{\mathrm{M}}). \tag{3}$$

Whereas ρ_{F} and ρ_{M} denote the fiber and matrix densities, V_{S} represents the total volume of the current spring design and φ_{S} denotes the FVC. The direct material costs of the same spring leads to:

$$C_{\mathrm{DMC}} = m_{\mathrm{F}} \cdot C_{\mathrm{F}} + m_{\mathrm{M}} \cdot C_{\mathrm{M}}. \tag{4}$$

Since the density of the used glass fibers is higher than the density of the matrix material, a higher FVC would lead to heavier springs. In contrast, the costs per kilogram of glass fibers are likely to be lower than the costs of matrix material in the current scenario. Thus, the heavier the spring gets due to an increasing FVC, the cheaper it gets in terms of direct material costs (of course this highly depends on the costs for fiber and matrix material of the given optimization problem and can therefore lead to different conclusions in other scenarios). This does not take possible subsequent costs into account, which could arise from a higher component mass. Especially penalty payments due to carbon dioxide emissions could play a role in this context. Unfortunately, these additional costs are difficult to quantify because of their situational character. To still consider these kind of costs, they are usually represented by a lightweight cost factor. This factor should be defined by the product owner and indicates, what he is willing to pay as an extra amount per kilogram to foster lightweight measures.

Figure 7 depicts the total costs C_{S} for an exemplary GFRP leaf spring design with defined geometry and thus fixed volume. In order to be able to investigate the influence of the lightweight cost factor C_{LW} and the fiber volume content φ_{S}, both variables are varied within a certain range. In case of a comparably low value for C_{LW}, meaning that the product owner is not willing nor able to invest in lightweight measures, the direct material costs are dominant. So, due to the lower price per kilogram of fibers, designs with a higher FVC and thus a higher spring mass will always be preferred over more lightweight designs. However, in higher regions of C_{LW}, designs with lower FVCs are favorable, leading to spring designs with higher direct material costs and a lower mass. The threshold for this generic example is located at $C_{\mathrm{LW}} = 8.76$ €/kg (see black line). If the specified C_{LW} is below this threshold, the optimizer will always prefer designs with a higher FVC and thus lower direct material cost, but a higher spring mass. On the other hand, if the given C_{LW} is above the threshold, the optimizer will prefer designs with a lower FVC and thus a higher direct material cost, but a lower

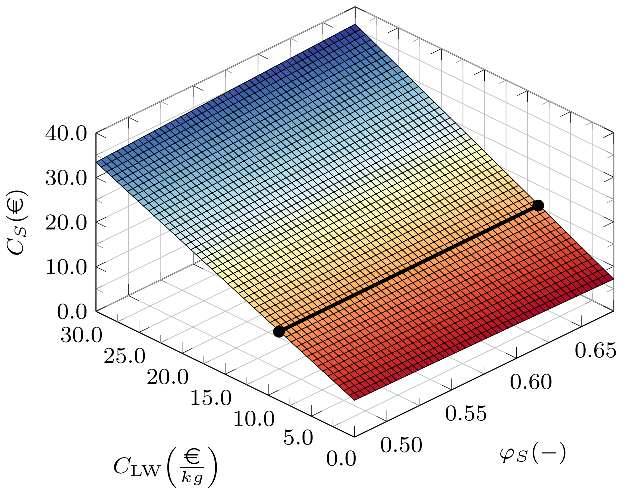

Fig. 7 Component cost depending on lightweight cost factor and fiber volume content

spring mass. It has to be mentioned that this threshold will be varying depending on the current component's volume and therefore will be different for every design proposal.

4.2 Constraints

Besides the above described objective function, the system has to fulfill four different types of constraints (see Fig. 8). First and foremost, the strength of the leaf spring has to be ensured, avoiding fiber and matrix failures. There are various types of failure criteria for fiber materials which could be applied. The Tsai–Wu criterion (1971) is relatively simple and its computational effort is comparably low. Unfortunately, there is no distinction in different modes of failure, which makes the identification of measures to improve the design difficult. The Hashin criterion in contrast distinguishes between fiber and matrix failure (Hashin 1980). Furthermore, the Cuntze and the Puck criterion have an even more detailed differentiation regarding load type and direction (Cuntze 1997; Puck 1996). The latter has been chosen for the current optimization, since it gives very detailed information about different failure modes and its computational effort is neglectable, compared to simulation time. Thus, the following strength constraints has to be met, where $f_{\mathrm{E}_{\mathrm{FF}}}$ is the strength regarding fiber fracture and $f_{\mathrm{E}_{\mathrm{I}}}$ is the strength regarding matrix or interfiber fracture. The Puck criteria are dimensionless quantities and are calculated from the quotient of an equivalent stress and a certain strength value (σ/R). Therefore, a value of 1.0 should not be exceeded. Although this applies in principle to both quantities, preliminary studies have shown that the interfiber fracture criterion is less critical, which is why the limit was set to 2.0.

$$\begin{aligned} g_1(\bar{x}) &= \max(f_{\mathrm{E}_{\mathrm{FF}}}(\bar{x})) \quad \leq 1.0, \\ g_2(\bar{x}) &= \max(f_{\mathrm{E}_{\mathrm{I}}}(\bar{x})) \quad \leq 2.0. \end{aligned} \tag{5}$$

Since the optimization aims to optimize the shape of both springs, it has to be assured that there is no contact between the upper and the lower leaf spring. This leads to the following constraint, which checks the overall maximum contact pressure $P_{\mathrm{S}_1\mathrm{S}_2}$ between both components in every simulation step:

$$g_3(\bar{x}) = \max(P_{\mathrm{S}_1\mathrm{S}_2}(\bar{x})) \leq 0.01\,\mathrm{MPa}. \tag{6}$$

Due to connected non-design-areas, the first and the last control point of the curve are fixed. The coordinates of the following points (after first and before last) are only alterable in the normal direction of the adjacent joint plane. This already leads to a curve which is in most cases quite smooth. To further promote smooth guide curves with a preferably low

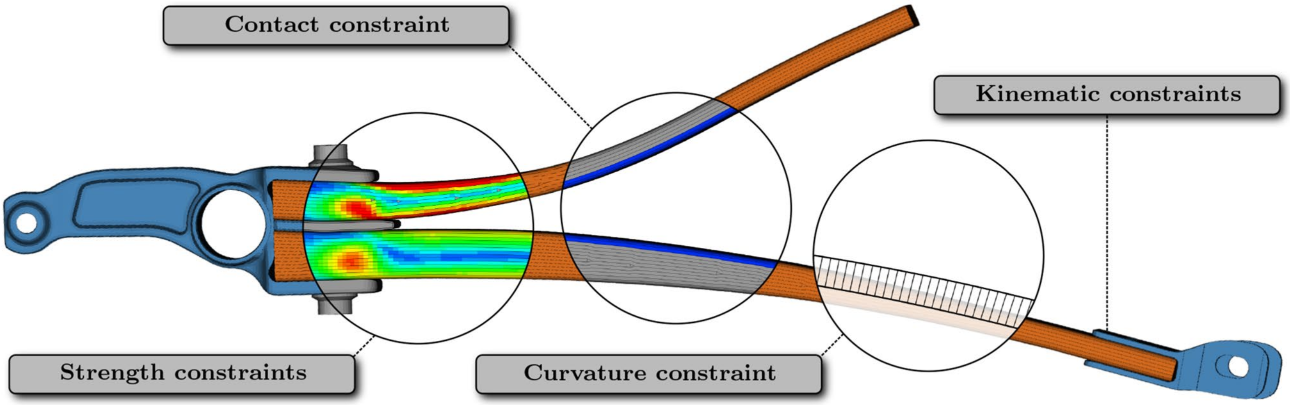

Fig. 8 Implicit constraints of the optimization model including spring-to-spring contact, spline curvature, structural strength and driving dynamic parameters

curvature, the optimization workflow offers the possibility to define a maximum curvature $C''_{\max}$ for the spline:

$$g_4(\overline{x}) = \max(C''(\overline{x})) \leq 0.02\,\mathrm{mm}^{-1}. \tag{7}$$

Finally, the system has to satisfy the following eight specific driving dynamic values to reproduce the kinematic characteristics of the benchmark system as good as possible:

1. Toe-in angle
2. Camber angle
3. Roll steer (empty)
4. Roll steer (full)
5. Spring rate
6. Stabilization rate
7. Long. suspension
8. Wheel load

For each of these characteristic values, the relative error ΔD_n between simulation result of the current design D_{Cn} and the result of the nominal design D_{Nn} has to be smaller than 5%. The index $n = \{1, \dots, 8\}$ denotes the corresponding driving dynamic values from the upper list.

$$g_n(\overline{x}) = |(D_{\mathrm{Cn}}/D_{\mathrm{Nn}}) - 1.0| = \Delta D_n(\overline{x}) \leq 0.05. \tag{8}$$

Taking all the above described constraints into account will probably result in an objective space, which is invalid in many regions. Therefore, the localization of valid designs is considered to be most challenging.

5 Optimization algorithm

As already elaborated in Sect. 1.2, Bayesian optimization methods and the use of metamodels such as Gaussian Processes fit well on the given optimization problem. This is due to the fact, that the given objective and constraint functions do not allow a closed-form formulation depending on the design variables. Furthermore, their numerical determination require an elaborate calculation of nonlinear system responses in terms of time and cost. Optimizing on surrogate models to minimize the number of needed evaluations can save a significant amount of both. Furthermore the hardly feasible calculation of gradients and the highly nonlinear response functions exclude methods based on gradient information, such as sequential quadratic programming or the method of moving asymptotes. The presented optimization workflow was implemented in C++14 and grounded on the optimization platform LEOPARD of the Volkswagen AG.

5.1 Bayesian optimization

Optimization methods based on the Bayesian approach had firstly been introduced and evaluated by Kushner (1964) and Mockus (1975). The theoretical foundation of Bayesian optimization methods is the so-called *Bayes' theorem* which can be expressed in the following way:

$$P(M \mid E) \propto P(E \mid M) \cdot P(M). \tag{9}$$

Broadly speaking, the posterior probability of a given model M given evidence E is proportional to the likelihood of E given M multiplied by the prior probability of M. The prior in this case is referred to as the existing belief of the range of possible objective functions. For example, this could be knowledge regarding shape, smoothness, convexity or complexity. These and further characteristics make certain objective functions more plausible than others (Brochu et al. 2010).

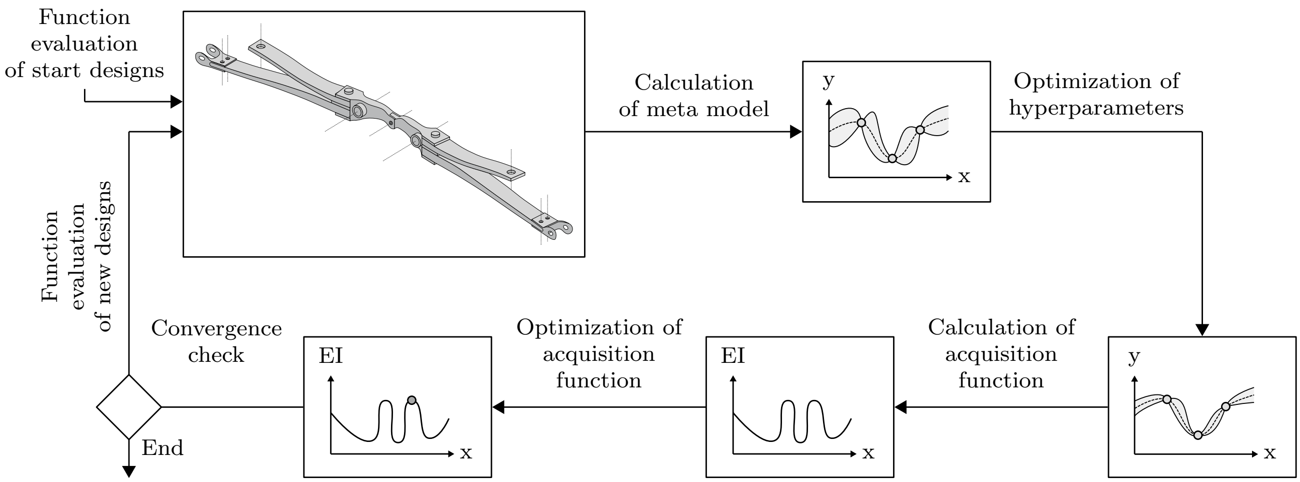
Function evaluation of start designs
Calculation of meta model
y
x
Optimization of hyperparameters
Function evaluation of new designs
Convergence check
End
EI
x
Optimization of acquisition function
EI
x
Calculation of acquisition function
y
x

Fig. 9 Bayesian optimization workflow, starting with the evaluation of start designs, followed by the calculation and optimization of the meta model, onto the generation and optimization of an acquisition function and looping back to a function evaluation of a new design proposal

The optimization framework under the use of Bayesian methods is presented in Fig. 9. In order to start the optimization process with a reasonable surrogate model, a previously defined number of designs are simulated and evaluated. These initial designs are found using an advanced Latin hypercube approach to generate space-filling DOEs (Viana et al. 2009). Based on these evaluations, the surrogate models of the objective function and possible constraint functions are build. Following, the hyperparameters of the model are optimized using a suitable gradient method, such as L-BFGS, and thus the quality of the model itself. Instead of optimizing directly on the surrogate model, an acquisition function is derived from it, which rates points based on the given information of the surrogate model. This way, the most promising points to evaluate in following iterations will be picked by maximizing the acquisition function using numerical optimization methods.

5.2 Gaussian processes

Using Gaussian processes as surrogate models to optimize black-box functions offers some fundamental advantages. First and foremost, every finite parameter combination of the given design space is a multivariate Gaussian distribution:

$$\hat{x} = (x_1, x_2, \dots, x_n) \sim \mathcal{N}(\mu, \Sigma). \tag{10}$$

Therefore, a Gaussian process not only provides a prediction of the possible function value but also a confidence interval, assessing the quality of the prediction at the requested point.

$$f(\cdot) \sim \mathcal{GP}(\mu(\cdot), \Sigma(\cdot,\cdot)). \tag{11}$$

Thus, given a set of input points X, and their corresponding function evaluations $f(X)$, the posterior distribution of a requested point $\hat{x}$ can be calculated:

$$\begin{bmatrix} f(X) \\ f(\hat{x}) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu(X) \\ \mu(\hat{x}) \end{bmatrix}, \begin{bmatrix} \Sigma(X,X) & \Sigma(X,\hat{x}) \\ \Sigma(\hat{x},X) & \Sigma(\hat{x},\hat{x}) \end{bmatrix} \right). \tag{12}$$

A further in-depth treatment of Gaussian processes, their theoretical backgrounds and their wide range of possibilities in a practical context can be found in Rasmussen and Williams (2006).

Another important benefit of Gaussian processes are their variability. Here, the underlying covariance function, also known as kernel function, have the biggest impact on the type and shape of the Gaussian process. In principal, it evaluates the influence of each point to each other point, based on different types of distance metrics. In the presented framework, various kernel types has been implemented (see Fig. 10) to cover a possibly wide range of different functions. Examples would be the *dot-product* kernel for modeling polynomial curves, the *exp-sine-squared* kernel having periodic characteristics or kernels aiming to describe more irregular functions, such as *rational-basis function*, *rational-quadratic* or *Matérn* kernels.

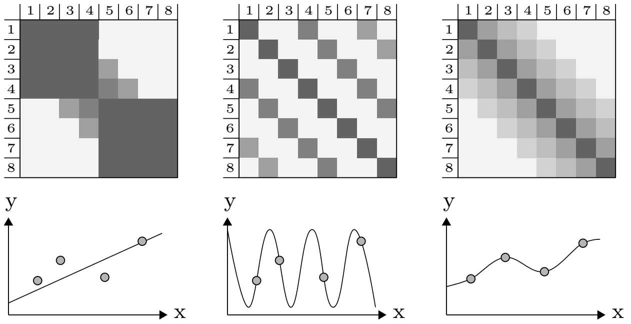

Fig. 10 Generic visualization of three different kernels, their covariance matrices and their characteristic function shape: dot-product (left), sine-squared (center) and radial-basis function (right)

5.3 Acquisition functions

In order to simultaneously optimize the objective function and find most feasible solutions, a combined acquisition function approach will be applied. For the objective function, the method of expected improvement will be used as followed:

$$\mathrm{EI}(x) = (\mu(x) - f(x^+) - \xi)\Phi(\mathrm{Z}) + \sigma(x)\phi(Z). \tag{13}$$

Here $\mu(x)$ is the mean and $\sigma(x)$ is the variance at location x. The parameter ξ controls the ratio between exploration and exploitation, with higher values putting a stronger weight on exploration. For the presented studies this parameter was set to 0, 01. Φ represents the cumulative distribution function (CDF), while ϕ is the probability density function (PDF) of the standard normal distribution. $f(x^+)$ denotes the objective function value of the best solution found so far. Finally, the term Z stands for the improvement at location x:

$$Z(x) = \begin{cases} \dfrac{\mu(x) - f(x^+) - \xi}{\sigma(x)} & \text{if } \sigma(x) > 0, \\ 0 & \text{if } \sigma(x) = 0. \end{cases} \tag{14}$$

For the constraint functions, on the other hand, the probability of feasibility approach is used. The parameter b in this case represents the boundary of the given constraint, while the $\hat{f}(x)$ is the predicted constraint value at location x:

$$\mathrm{PF}(x) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{b - \hat{f}(x)}{\sigma(x)}\right)\right]. \tag{15}$$

The above calculated acquisition function values are then summed up in the following way, where A_o is the acquisition function value of the respective objective function and $A_{c,i}$ are the acquisition function values of all constraint functions.

$$A_t = A_o + \sum_{n=1}^{n_c} A_{c,i}. \tag{16}$$

Since the PF function has a codomain of [0; 1], whereas the EI function has a codomain of $\mathbb{R}$, the sum of these function values does not result in an equivalent weighting of the functions. Therefore, the EI function has to be scaled to the same codomain in a preceding optimization step.

6 Results

The following section presents the results of the implemented optimization workflow on different versions of the given optimization problem.

6.1 Comparison to alternative algorithms

In order to prove the capability of the Bayesian approach as well as its superiority compared to other numerical optimization methods on nonlinear, non-differentiable and computation-extensive problems, it is evaluated on different test functions. As alternative optimization methods, Simulated Annealing (Kirkpatrick et al. 1983) and an enhanced Firefly-Algorithm (Yang 2008) has been chosen for the comparison. Respective test functions are the one-dimensional *Rastrigin* function, the two-dimensional *Branin* function and the *Hartman* function in three and six dimensions. Since the optima of these functions are known, the convergence criteria chosen for them is an acceptable relative deviation of 1% and an absolute deviation of 0.01 to the optimum. Since at this stage there is insufficient knowledge regarding the shape and nature of the actual objective function, a variety of different test functions were used.

Table 1 shows the required function evaluations of each algorithm for each test function. The presented values are the mean values of respectively 10 optimization runs per combination. Simulated Annealing was not able to converge on the six-dimensional *Hartman* function and shows the weakest performance in general compared to the other two approaches. This may be due to its highly randomized character and its possibility in accepting new but worse design proposals. The enhanced Firefly-Algorithm has a better performance but still needed a significantly larger amount of function evaluations.

Table 1 Number of function evaluations needed until convergence criteria fulfilled for Simulated Annealing (SA), Enhanced Firefly (FF) and Bayesian Optimization (BO) on different test functions

	Rastrigin (1D)	Branin (2D)	Hartman (3D)	Hartman (6D)
SA	481	6.820	1.254	–
FF	75	870	582	13.943
BO	12	29	17	77

Although, it has to be mentioned, that an optimization iteration using the Bayesian approach needs more computation time, not considering the function evaluation itself. Optimizing the hyperparameters for fitting the surrogate model as well as optimizing the acquisition function are the main reasons for the comparably high computation effort. Nevertheless, the more complex and time-consuming the function evaluation itself will be, the more worthwhile will be the usage of Bayesian optimization approaches.

6.2 Simplified optimization of lower GFRP spring

Since the actual simulation model of the given problem is quite large and the corresponding optimization will be very time- and cost-intensive, some preliminary evaluations on the problem can be very helpful. On the one hand, to ensure the functionality of the workflow and the optimization, on the other hand to learn about the characteristics of the optimization problem itself and to get an idea of the shape and nature of the objective function.

In a first step, a simplified version of the GFRP leaf spring system based on the actual components will be simulated. All non-design components are therefore represented by simple shell, rigid or connector elements. Only the lower springs will be modeled by hexahedral elements and optimized in terms of cost. Design variables will be the FVC of the lower springs, as well as X and Y coordinates of the corresponding NURBS control point and the height and width of the spring's cross-section. Figure 11 shows the respective leaf spring in a randomly generated start design. Considered technical constraints will be the fiber and matrix strength g_1 and g_2, as well as the contact between lower and upper spring g_3, presented in Sect. 4.2. To further simplify the optimization, only two of the five given misuse load cases will be simulated: the *vertical impact*, in which a strong force is applied on both wheels in the global Z direction, and the *extreme curve*, in which forces in both the global Y and Z directions are applied to the left wheel only. Furthermore the objective function in this case will be the calculated costs for the spring component introduced in Sect. 4.1. To study the influence of the lightweight factor C_{LW}, three different optimization runs are compared with each other, starting with $C_{\mathrm{LW1}} = 0.00$ €/kg to evaluate the conventional cost function. The subsequent runs intend to cover different use cases: a lightweight factor of $C_{\mathrm{LW2}} = 7.00$ €/kg represents the current development of vehicles with combustion engines, while a factor of $C_{\mathrm{LW3}} = 18.00$ €/kg is common in the development of electric vehicles (Siebel 2020). At this point,

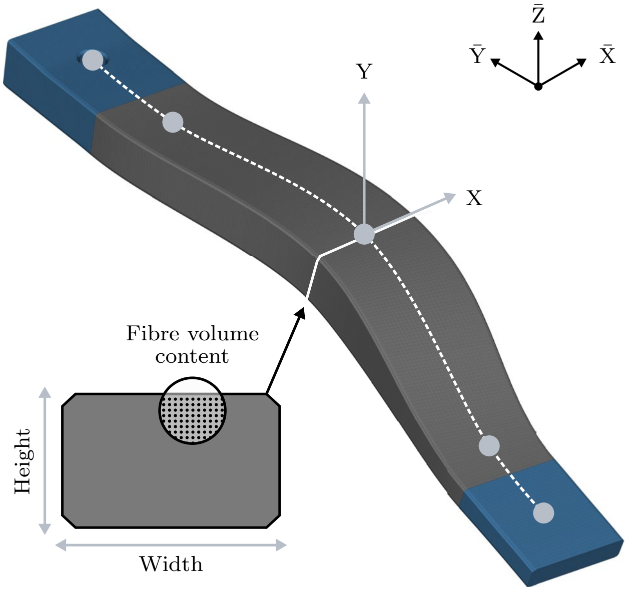

Fig. 11 Randomly generated start design of the leaf spring depicting the input variables for the optimization of mass. Blue elements represent the non-design space, gray elements represent the design space of the spring. The shape of the spline curve is defined by the X and Y coordinates of the center control point. The first, second, fourth and fifth control points are fixed to guarantee a smooth spring curve design

it must be pointed out that there are generally very few publications providing concrete values. Furthermore, these values are very much dependent on the selected scenario and its boundary conditions. In principle, it is possible for the user to compare different scenarios at this point. For reasons of data protection, the cost function is replaced by a normalized fitness function in the following.

The normalized cost objective function, as well as the components mass for all three optimization runs are shown in Fig. 12. Due to the use of the above described lightweight factors, the objective functions are located on distinct levels, as the price per kilogram is different for each run. Still, the costs can be reduced significantly in every configuration. In contrast, the mass of the leaf spring takes a different course for the first optimization run. In order to further reduce the cost objective function, which in the case of C_{LW1} is equivalent to a reduction of direct material costs, the fiber volume content of the component is increased in latter iterations. This follows the expectations of the introduced cost model by favoring higher values of FVC due to its lower price, which again leads to an increase in component mass. Thus, the product owner's decision to assess the design based solely on direct material costs would result in a heavier but cheaper product. The factors C_{LW2} and C_{LW3} again represent scenarios, in which the product owner is willing to invest in lightweight measures. For these cases, the tipping point described in Fig. 7 has been crossed, resulting in a minimization of cost accompanied by a minimization of mass. However, this is only worthwhile if the cost difference can be offset in the further course of the product life cycle. Figure 13 shows the cost-optimal design of the first optimization run using C_{LW1}.

The development of different input variables is shown in Fig. 14, where the geometric variables shown refer solely to the center control point. As already stated, the respectively chosen configurations for the presented cost model leads to

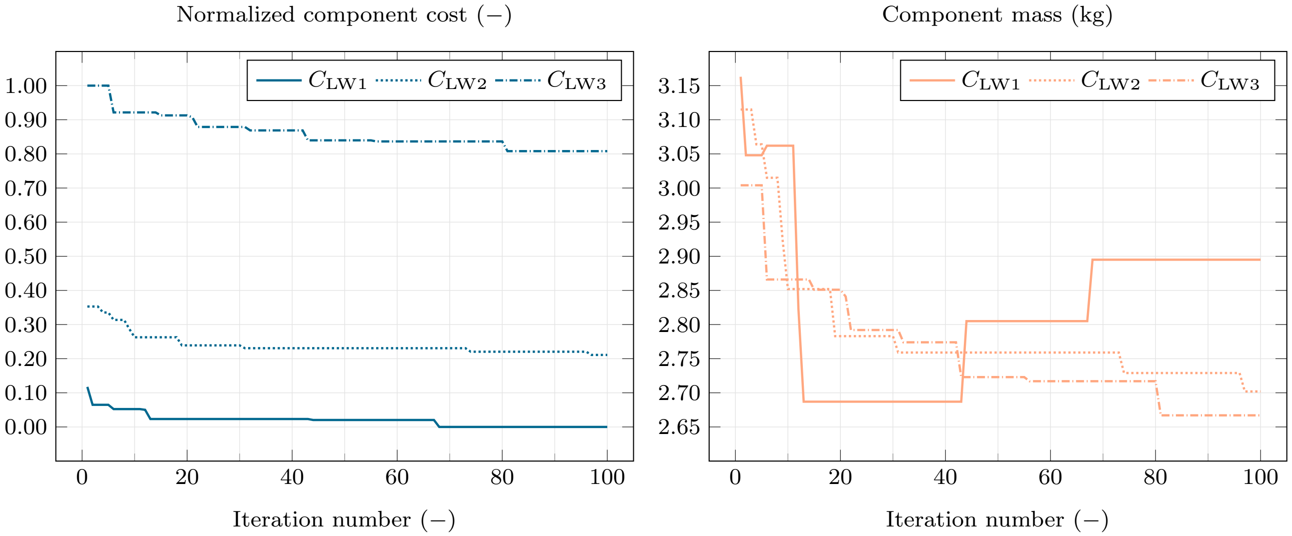

Fig. 12 Normalized cost objective (left graph) and mass (right graph) history of the simplified optimization of the lower spring presented in Sect. 6.2 within 100 iterations. The three different colors represent optimization runs using the varying lightweight factors $C_{\mathrm{LW1}} = 0.00$ €/kg, $C_{\mathrm{LW2}} = 7.00$ €/kg and $C_{\mathrm{LW2}} = 18.00$ €/kg

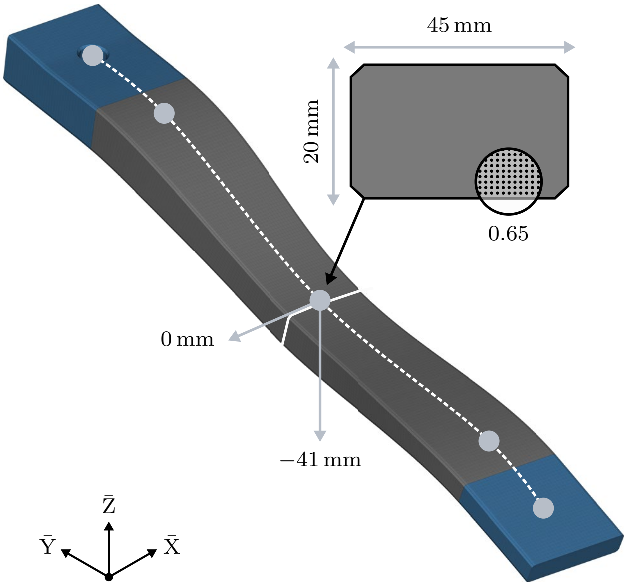

Fig. 13 Cost-optimized leaf spring design of iteration 68 using a lightweight factor of $C_{\mathrm{LW1}} = 0.00$ €/kg

a different behavior in choosing the FVC. It was furthermore found that the coordinates of the spring's guide curve have a comparably low influence on the component's mass as they only influence the length of the spring. In contrast, the dimensions of the cross-section have a more direct influence on the component's mass and therefore its cost. This is mainly caused by the limits chosen for the design variables. In the current example, the nominal leaf spring design has a total mass of 3.60kg. Nominal design in this case refers to the mean value of each design variable being used. Setting the spring height to its upper bound increases the mass of the spring by 1.53kg and by 0.40kg when setting the spring's width to its upper bound. Changing the coordinates to its upper limit only results in a mass increase of 0.06 kg.

Concerning the demanded technical constraints, fiber strength seem to be most critical for the given application example. While the contact constraint g_3 can be satisfied in 96% and the matrix strength constraint g_2 in 86% of all iterations, the fiber strength constraint only achieves a validity ratio of 58%. The spring's height seems to have a crucial influence on this. While valid design proposals could be found inside the limits of all other variables, for the height only a small corridor between 13.6 and 25.6 mm resulted in valid designs. This makes particular sense when comparing these values with the spring heights of the neighboring non-design areas, which range from 14.7 to 28.2 mm. Thus, especially springs with a rather constant cross-sectional course achieve good values regarding fiber strength.

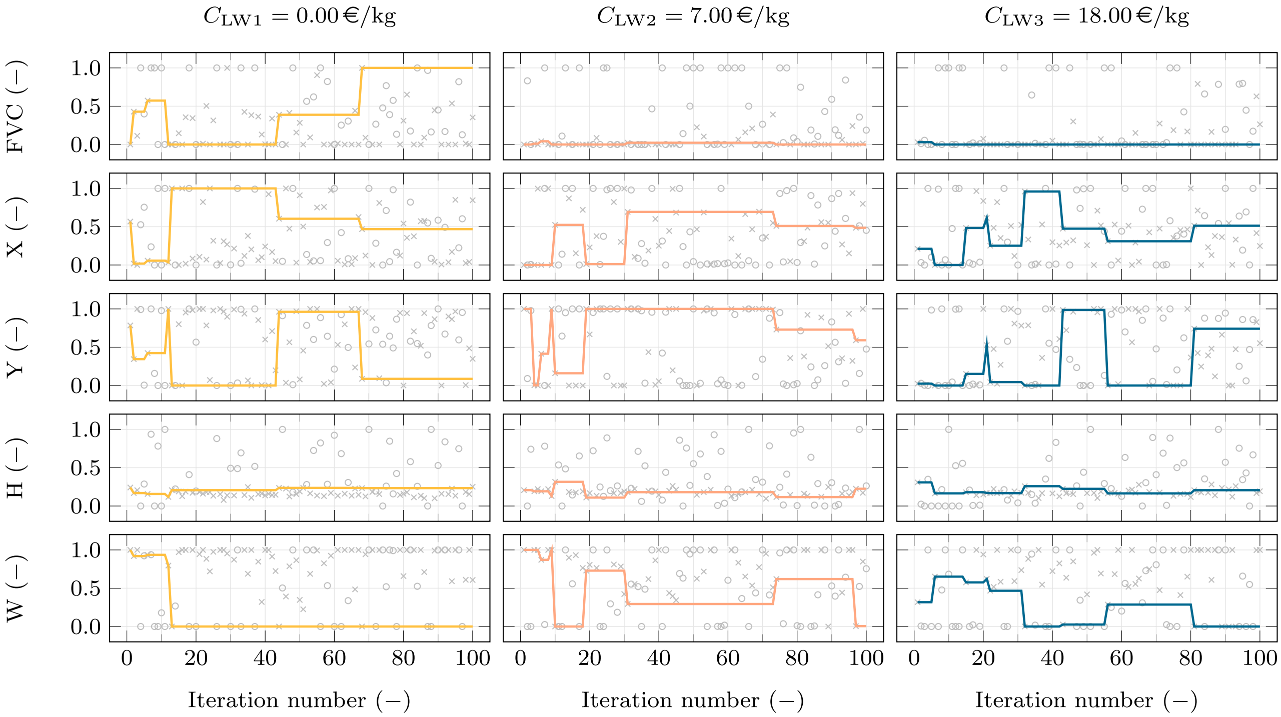

Fig. 14 Normalized input variable history of the simplified optimization of the lower spring presented in Sect. 6.2 within 100 iterations. FVC represents the fiber volume content, X and Y are the coordinates of the control point and H and W are the cross-section's height and width. The gray marks represent design proposals in each iteration, which are either valid (cross) or invalid (circle)

6.3 Optimization of the GFRP leaf spring system

After testing the algorithm and evaluating its characteristics, the optimization of the application example presented in Sect. 3 will follow. Besides the already introduced constraints of strength and contact, the remaining eight constraints regarding driving dynamic behavior will be added. Furthermore the so far used simplified model will be replaced by the full assembly model.

To generate a reasonable initial surrogate model, a DOE consisting of 25 designs is calculated. The number of initial data points should not be too small, as it can give both the user and the optimizer a global overview over the result space. Again, this number is limited upwards by the available computing capacity and time, since the DOE is only a jump-start of the actual optimization. Based on this initial DOE, different kernels are tested on every given output. Besides the optimized log-marginal likelihood values for the trained GPs, the so-called R^2-prediction-value will be used as an assessment criterion (Myers et al. 2009). Points are removed consecutively from the training data set and are then predicted by the reduced model. These predicted values $\hat{y}_{i-1}$ are then compared to the true values y_i. By dividing the squared sum of these errors with the squared sum of the difference between the true values y_i and the mean value $\overline{y}$, the R^2-prediction-value can be obtained. The closer this value is to 1.0 the better the model.

$$R^2_{\text{Pred}} = 1 - \frac{\text{PRESS}}{\text{SQT}} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_{i-1})^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}. \tag{17}$$

Table 2 shows the results of the kernel testing. As can be seen in the last row, different outputs lead to very different general predictability of the kernels. For example, the cost objective function f_1 is relatively easy to model by various kernels, since most of the input parameters are geometrical parameters, directly influencing the mass of the spring and therefore the components cost. Outputs regarding strength or driving dynamic behavior on the other hand are harder to model. However, using the right kernel can improve the predictability significantly. The entries marked bold in Table 2 achieved the best results in terms of predictability. Since these kernels seem to fit best to the respective response functions, they are used in the upcoming optimization.

The carried out DOE has also confirmed the assumption made in Sect. 4.2. Large areas of the result space will be invalid regarding the technical constraints. None of the designs proposed by the DOE can fulfill all of the imposed constraints. As a consequence, the search for a fully valid design seems to be of more importance than finding a cost-optimal solution. In other words, if the optimizer is able to find valid designs, the focus can then be shifted toward cost-optimal solutions. Therefore, the objective function will not be the components cost alone, but a fitness value composed of the sum of all normalized constraint values and the cost value.

Due to performance reasons, the maximum number of iterations of the current optimization has been limited to 200 iterations. As can be seen in Fig. 15 the optimization is able to continuously find better solutions regarding the introduced fitness value. The components cost, however, varies and is not necessarily descending as design proposals are dominant if they are able to satisfy more constraints.

Table 2 R^2-predict values for different kernels using the initial DOE-set as training

Kernel type	R^2-predict (%)											
	f_1	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}	g_{11}
RBF Isotropic	90	56	5	38	58	66	56	67	62	34	70	**77**
RBF Anisotropic	67	47	13	48	76	68	54	62	64	52	51	61
Matern 0.5 Isotropic	76	58	16	**63**	64	62	55	65	67	27	68	70
Matern 1.5 Isotropic	**99**	58	10	46	71	69	67	**75**	66	21	**74**	68
Matern 2.5 Isotropic	**99**	57	10	44	65	68	62	74	65	20	73	73
Matern 0.5 Anisotropic	*66*	47	14	52	72	68	**69**	60	**70**	50	70	*58*
Matern 1.5 Anisotropic	78	**60**	**17**	42	**84**	77	64	62	69	62	72	70
Matern 2.5 Anisotropic	73	51	15	48	71	75	62	73	*51*	62	67	73
Rational-Quadratic Isotropic	97	58	13	56	69	69	66	72	66	20	71	71
Rational-Quadratic Anisotropic	84	57	12	61	67	**91**	**69**	74	67	**63**	71	68
Exp Sine Square	91	53	11	39	65	69	60	74	65	21	71	71
Exponentiation Dot	97	*0*	*0*	*36*	*54*	*53*	*39*	*52*	53	*0*	*35*	59
Mean	85	50	11	48	68	70	60	68	64	36	66	68

Every value shown in the table represents the mean value of 16 fitting and hyperparameter optimization runs. The best prediction values for each output is marked bold, the worst prediction values are marked text italics

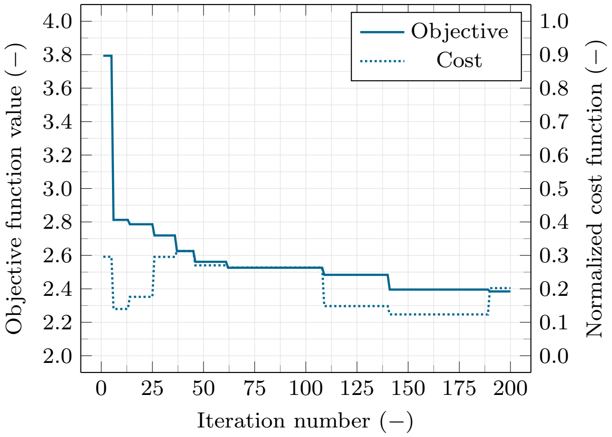

Fig. 15 Objective function history within 200 iterations. The optimum value is reached at iteration 190

Nevertheless, the costs only vary within a relatively small range, which indicates that areas outside this range are much more invalid. This, in turn, would make it possible to reduce the size of the design space of existing design variables in order to search for valid solutions in more detail. On the other hand, the results has shown that the optimization problem as formulated may be overconstrained, resulting in null-feasible regions. Especially the values for g_5 (toe-in angle), g_6 (camber angle) and g_8 (longitudinal suspension) were not once inside the feasible deviation of 5%. A possible reason for this might be an insufficient design space regarding some of the driving dynamics constraints. This may be due to the defined parameterization, wherefore it will be extended in future works (see Sect. 7).

Figure 16 shows the resulting spring geometry of iteration 190. The design is valid regarding six of the eleven imposed constraints, including Puck strength, contact, spring rate, stabilization rate and wheel load. The corresponding history of normalized design variables is shown in Fig. 17. Compared to the preceding optimization runs the range of found optimal solutions is much broader regarding the given design variables. This indicates that the optimizer identifies various areas which are similarly promising. These areas are then explored consecutively to find better solutions.

To prove the assumption made in Sect. 6.1, claiming that Bayesian optimization can be superior to other methods, the comparison of algorithms has been repeated on the calculated meta model resulting from the present optimization. The firefly optimization needed 5931 function evaluations with the recommended swarm size of 50 to achieve a same or better objective value averaged. The simulated annealing approach on the other hand reached the same results after an average number of 1758 iterations. However, it must be mentioned here that recommended default parameters were chosen for both algorithms.

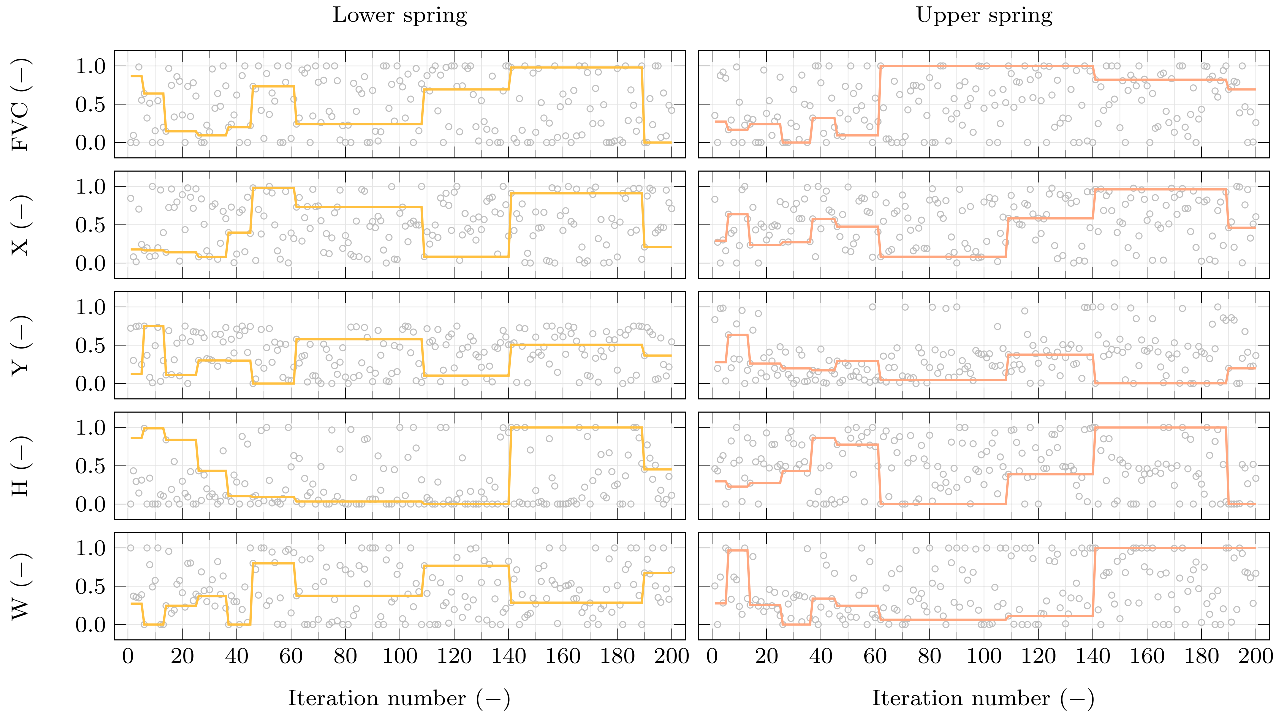

Fig. 16 Normalized input variable history within 200 iterations. FVC represents the fiber volume content, *X* and *Y* are the coordinates of the control point and *H* and *W* are the cross-section's height and width. The gray marks represent design proposals in each iteration, which are either valid (cross) or invalid (circle)

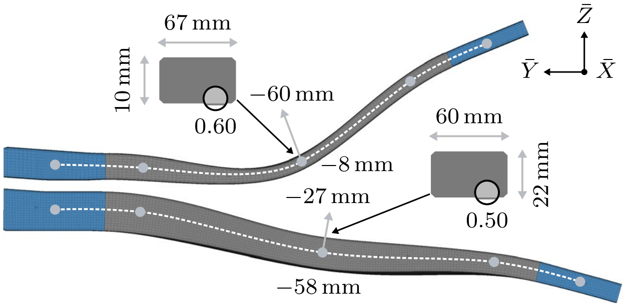

Fig. 17 Cost-optimized design of the GFRP leaf spring system at iteration 190. The technical constraints regarding strength are fulfilled and improved compared to the reference design. Furthermore, the optimization was able to satisfy 3 of 8 driving dynamic constraints

7 Conclusion and outlook

The presented paper has shown that Bayesian optimization can be a powerful tool for solving free-form shape optimization problems of structural components. Preliminary studies proved its ability in finding optimal solutions in a significantly smaller number of needed function evaluations compared to other algorithms. However, a various number of problems arose during implementing and optimizing, of which the most relevant will be described in the following sections.

7.1 Extended parameterization

For the optimization problem of the GFRP leaf spring system it has become evident, that the chosen parameterization can indeed provide good solutions in terms of cost and strength. The imposed driving dynamic constraints on the other hand cannot be satisfied completely. Therefore, the parameterization will be extended in further steps. Enabling the translation, rotation and scaling of the systems' connection to the vehicle on the end of the lower spring will likely enlarge the response space in the desired way.

7.1.1 Boundary issue

Based on the findings during the optimization process the so-called boundary issue has to be addressed in higher dimensional problems (Swersky 2017). It arises directly from the use of Gaussian processes as surrogate model. The larger the distance to any sampling point is to a queried point, the variance of the GP generally increases. Since the largest distance to any point will be located at the boundaries of the given parameter space, these points are likely to have the highest variance. Points with a high variance again are highly probable to get picked as next evaluation point, since the probability to achieve a new optimal solution at this point increases. Therefore, Bayesian optimization algorithms often explore the boundaries of a given search space. This is on the one hand reasonable, on the other hand it consumes a significant amount of function evaluations and optimization iterations. The given problem consists of 10 input dimensions, totally resulting in 1024 border elements. For the further development it is planned to implement approaches, which can help to overcome the described boundary issue. An example would be the application of special functions, such as the horseshoe prior (Carvalho et al. 2009).

7.1.2 Reduced model

In an attempt to further reduce simulation time, the volume elements of the spring components have been converted into shell elements. By calculating nodal thickness values, the spring's geometry can be modeled in a sufficient manner. The reduced model is able to produce similar results at least for the driving dynamic values, compared to the volume-based model. However, the strength values cannot be calculated precisely enough and therefore would have to be validated by the former simulation model. An investigation on how to use the reduced model to meet the demanded driving dynamic constraints can be helpful to shorten the optimization process as a whole.

7.1.3 Kernel selection

As shown in Sect. 6.3 a profound kernel selection can improve optimization processes significantly. The goal should therefore be to implement an automated kernel selection algorithm into the presented framework. This could be the selection of previously defined kernels based on an initial DOE. Alternatively, more in-depth methods like the compositional kernel search could be realized (Duvenaud et al. 2013).

Author contributions Not applicable.

Funding Open Access funding enabled and organized by Projekt DEAL. This work was financed by the Volkswagen AG. Results, opinions and conclusions are not necessarily those of the Volkswagen AG.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Replication of results The method presented in this paper is part of the commercial optimization environment LEOPARD of Volkswagen AG. Nevertheless, using the proposed numerical schemes, integration into other optimization environments and thus the reproduction of the presented results is possible without any restrictions.

References

Akbari M, Salmasi F, Arvanaghi H, Karbasi M, Farsadizadeh D (2019) Application of Gaussian process regression model to predict discharge coefficient of gated piano key weir. Water Resour Manag 33:3929–3947. https://doi.org/10.1007/s11269-019-02343-3

Alam FM, McNaught KR, Ringrose TJ (2004) A comparison of experimental designs in the development of a neural network simulation metamodel. Simul Model Pract Theory 1(12):559–578. https://doi.org/10.1016/j.simpat.2003.10.006

Al-Dujaili A, Suresh S (2017) Embedded bandits for large-scale blackbox optimization. In: Proceedings of the 31st AAAI conference on artificial intelligence. pp 758–764. https://doi.org/10.5555/3298239.3298351

Barton RR, Meckesheimer M (2006) Chap 18: metamodel-based simulation optimization. In: Handbooks in operations research and management science, vol 13. Elsevier B.V., pp 535–574. https://doi.org/10.1016/S0927-0507(06)13018-2

Basudhar A, Dribusch C, Lacaze S, Missoum S (2012) Constrained efficient global optimization with support vector machines. Struct Multidisc Optim 46:201–221. https://doi.org/10.1007/s00158-011-0745-5

Bletzinger KU (2017) Shape optimization. In: Encyclopedia of computational mechanics, vol 2. Wiley, Hoboken. https://doi.org/10.1002/9781119176817

Brochu E, Cora VM, de Freitas N (2010) A tutorial on Bayesian optimization of expensive cost functions with application to active user modelling and hierarchical reinforcement learning. arXiv:1012.2599

Bronstein IN, Semendyayev KA, Musiol G, Mühlig H (2007) Handbook of mathematics. Springer, Berlin. https://doi.org/10.1007/978-3-540-72122-2

Carvalho CM, Polson NG, Scott JG (2009) Handling sparsity via the horseshoe. In: Proceedings of the 12th international conference on artificial intelligence and statistics, pp 73–80. http://proceedings.mlr.press/v5/carvalho09a (accessed date: 1 August 2022)

Cuntze RG (1997) Neue Bruchkriterien und Festigkeitsnachweise für unidirektionalen Faserkunststoffverbund unter mehrachsiger Beanspruchung - Modellbildung und Experimente. In: Fortschritts-Berichte VDI: Reihe 5, Grund und Werkstoffe, VDI-Verlag, Düsseldorf

Diouane Y, Picheny V, Riche RL, Perrotolo ASD (2021) TREGO: a trust-region framework for efficient global optimization. HAL Open Science. https://hal.archives-ouvertes.fr/hal-03450072 (accessed date: 1 August 2022)

Dominguez S, Nigam N, Shahriari B (2017) A combined finite element and Bayesian optimization framework for shape optimization in spectral geometry. Comput Math Appl 74:2874–2896. https://doi.org/10.1016/j.camwa.2017.08.044

Duvenaud D, Lloyd JR, Grosse R, Tenenbaum JB, Ghahramani Z (2013) Structure discovery in nonparametric regression through compositional kernel search. In: Proceedings of the 30th international conference on machine learning, vol 28, pp. 1166–1174. arXiv:1302.4922

Egea JA, Vries D, Alonso AA, Banga JR (2007) Global optimization for integrated design and control of computationally expensive process models. Ind Eng Chem Res 46:9148–9157. https://doi.org/10.1021/ie0705094

Elhara O, Varelas K, Nguyen D, Tusar T, Brockhoff D, Hansen N, Auger A (2019) COCO: the large scale black-box optimization benchmarking test suite. arXiv https://doi.org/10.48550/arXiv.1903.06396

Eschenauer HA (1989) Chap 1: the "three columns" for treating problems in optimum structural design. In: Optimization: methods and applications, possibilities and limitations, vol 47. Springer, Berlin, pp 1–21. https://doi.org/10.1007/978-3-642-83814-9

Frazier PI (2018) A tutorial on Bayesian optimization. Computer Science. https://arxiv.org/pdf/1807.02811.pdf

Gardner JR, Kusner MJ, Xu ZE, Weinberger KQ, Cunningham JP (2014) Bayesian optimization with inequality constraints. In: Proceedings of the 31st international conference on machine learning, vol 32, pp 1–9. https://doi.org/10.5555/3044805

Gaylo R, Farahani S, Schmueser D, Pilla S (2019) Optimization of a mono-composite leaf spring using a hybrid fiber-layup approach. Int J Interact Des Manuf 14:407–421. https://doi.org/10.1007/s12008-019-00636-w

Gelbart MA, Snoek J, Adams RP (2014) Bayesian optimization with unknown constraints. In: Proceedings of the 30th conference on uncertainty in artificial intelligence, vol 14, pp 250–259. https://doi.org/10.5555/3020751.3020778

Ghag MD, Ghadigaonkar PP, Nikam M, Teli SN (2018) Design and material optimization of jeep leaf spring. In: Global meet on advances in design, materials and thermal engineering

Ghosh S, Mondal S, Kapat JS, Ray A (2019) Parametric shape optimization of pin fin arrays using surrogate model based Bayesian methods. In: AIAA Propulsion and Energy 2019 Forum. https://doi.org/10.2514/6.2019-3830

Gramacy RB, Lee HKH (2010) Optimization under unknown constraints. arXiv:1004.4027

Gramacy RB, Gray GA, Digabel SL, Lee HKH, Ranjan P, Wells G, Wild SM (2016) Modeling an augmented Lagrangian for blackbox constrained optimization. Technometrics 58:1–11. https://doi.org/10.1080/00401706.2015.1014065

Hashin Z (1980) Failure criteria for unidirectional fiber composites. J Appl Mech 47:329–334. https://doi.org/10.1115/1.3153664

Hennig P, Schuler CJ (2012) Entropy search for information-efficient global optimization. J Mach Learn Res 13:1809–1837 arXiv:1112.1217

Hernandéz-Lobato JM, Hoffmann MW, Ghahramani Z (2014) Predictive entropy search for efficient global optimization of black-box functions. In: Advances in neural information processing systems, vol 1. arXiv:1406.2541

Hernandéz-Lobato JM, Gelbart MA, Hoffmann MW, Adams RP, Ghahramani Z (2015) Predictive entropy search for Bayesian optimization with unknown constraints. In: Proceedings of the 32nd international conference on machine learning, vol 37, pp 1699–1707. arXiv:1502.05312

Hernandéz-Lobato JM, Gelbart MA, Adams RP, Hoffmann MW, Ghahramani Z (2016) A general framework for constrained Bayesian optimization using information-based search. J Mach Learn Res 17:1–53. arXiv:1511.09422

Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. J Glob Optim 1(13):455–492. https://doi.org/10.1023/A:1008306431147

Kersten T, Fiebig S (2019) Challenges and benefits for lightweight chassis components focusing on rear axle development. In: 10th International Munich chassis symposium 2019. https://doi.org/10.1007/978-3-658-26435-2_18

Kim H, Teh YW (2018) Scaling up the automatic statistician: scalable structure discovery using Gaussian processes. In: Proceedings of the 21st international conference on artificial intelligence and statistics, vol 21. https://doi.org/10.48550/arXiv.1706.02524

Kirkpatrick S, Gelatt CD Jr, Vecchi MP (1983) Optimization by simulated annealing. Science 220:671–680. https://doi.org/10.1126/science.220.4598.671

Kumar DS, Radhika S, Ravindra K, Suman KNS (2011) Improved design optimization of a composite leaf spring using swarm intelligence. Int J Appl Eng Res 6:193–200

Kumar A, Charyulu TN, Ramesh C (2012) Design optimization of leaf springs. Int J Eng Res Appl 2:759–765

Kushner HJ (1964) A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. J Basic Eng 86:97–106. https://doi.org/10.1115/1.3653121

Lam RR, Poloczek M, Frazier PI, Willcox KE (2018) Advances in Bayesian optimization with applications in aerospace engineering. In: AIAA non-deterministic approaches conference. https://doi.org/10.2514/6.2018-1656

McDonald DB, Grantham WJ, Tabor WL, Murphy MJ (2007) Global and local optimization using radial basis function response surface models. Appl Math Model. https://doi.org/10.1016/j.apm.2006.08.008

Mehmani A, Chowdhury S, Meinrenken C, Messac A (2018) Concurrent surrogate model selection (COSMOS): optimizing model type, kernel function, and hyper-parameters. Struct Multidisc Optim 1(57):1093–1114. https://doi.org/10.1007/s00158-017-1797-y

Mockus J (1975) On Bayesian methods for seeking the extremum. In: Optimization techniques IFIP technical conference. https://doi.org/10.1007/3-540-07165-2_55

Mockus J, Tiesis V, Zilinskas A (1978) Bayesian approach to global optimization. In: Dixon LCW, Szego GP (eds) Towards global optimization, vol 2. North-Holland Publishing Company, Amsterdam, pp 117–129

Myers RH, Montgomery DC, Anderson-Cook CM (2009) Response surface methodology, vol 3. Wiley, Hoboken. https://doi.org/10.1080/00224065.2004.11980252

Noè U, Husmeier D (2018) On a new improvement-based acquisition function for Bayesian optimization. arXiv:1808.06918

Palar PS, Shimoyama K (2018) On efficient global optimization via universal kriging surrogate models. Struct Multidisc Optim 1(57):2377–2397. https://doi.org/10.1007/s00158-017-1867-1

Park S, Na J, Kim M, Lee JM (2018) Multi-objective Bayesian optimization of chemical reactor design using computational fluid dynamics. Comput Chem Eng 1(119):25–37. https://doi.org/10.1016/j.compchemeng.2018.08.005

Picheny V (2014) A stepwise uncertainty reduction approach to constrained global optimization. In: Proceedings of the 17th international conference on artificial intelligence and statistics, vol 33, pp 787–795

Puck A (1996) Festigkeitsanalyse von Faser-Matrix-Laminaten. Hanser Fachbuch, Munich

Qin C, Klabjan D, Russo D (2017) Improving the expected improvement algorithm. In: 31st Conference on neural information processing systems. arXiv:1705.10033

Rajendran I, Vijayarangan S (2001) Optimal design of a composite leaf spring using genetic algorithms. Comput Struct 79:1121–1129. https://doi.org/10.1016/S0045-7949(00)00174-7

Rasmussen CE, Williams CKI (2006) Gaussian processes for machine learning, vol 1. The MIT Press, Cambridge

Regis RG (2020) Large-scale discrete constrained black-box optimization using radial basis functions. In: IEEE symposium series on computational intelligence, pp 2924–2931. https://doi.org/10.1109/SSCI47803.2020.9308581

Riche RL, Picheny V (2021) Revisiting Bayesian optimization in the light of the COCO benchmark. Struct Multidisc Optim 64:3063–3087. https://doi.org/10.1007/s00158-021-02977-1

Schonlau M (1997) Computer experiments and global optimization. PhD Thesis, University of Waterloo. https://doi.org/10.5555/926099

Schonlau M, Welch WJ, Jones DR (1998) Global versus local search in constrained optimization of computer models. N Dev Appl Exp Des 34:11–25. https://doi.org/10.1214/lnms/1215456182

Shahriari B, Swersky K, Wang Z, Adams RP, de Freitas N (2016) Taking the human out of the loop: a review on Bayesian optimization. Proc IEEE 104:148–175. https://doi.org/10.1109/JPROC.2015.2494218

Shi L, Yang RJ, Zhu P (2012) A method for selecting surrogate models in crashworthiness optimization. Struct Multidisc Optim 1(46):159–170. https://doi.org/10.1007/s00158-012-0760-1

Shi M, Lv L, Sun W, Song X (2020) A multi-fidelity surrogate model based on support vector regression. Struct Multidisc Optim. https://doi.org/10.1007/s00158-020-02522-6

Shokrieh MM, Rezaei D (2003) Analysis and optimization of a composite leaf spring. Compos Struct 60:317–325. https://doi.org/10.1016/S0263-8223(02)00349-5

Siebel T (2020) Wieviel kostet Leichtbau? Springer Professional. https://www.springerprofessional.de/en/leichtbau/konstruktion---entwicklung/wieviel-kostet-leichtbau-/17788540 (accessed date: 1 August 2022)

Singh A, Nagar J, Sharma S, Kotiyal V (2021) A Gaussian process regression approach to predict the k-barrier coverage probability for intrusion detection in wireless sensor networks. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2021.114603

Song X, Lv L, Sun W, Zhang J (2019) A radial basis function-based multi-fidelity surrogate model: exploring correlation between high-fidelity and low-fidelity models. Struct Multidisc Optim 1(60):965–981. https://doi.org/10.1007/s00158-019-02248-0

Springenberg JT, Klein A, Falkner S, Hutter F (2016) Bayesian optimization with robust Bayesian neural networks. In: 30th Conference on neural information processing systems, vol 1(30). https://doi.org/10.5555/3157382.3157560

Srinivas N, Krause A, Kakade SM, Seeger M (2010) Gaussian process optimization in the bandit setting: no regret and experimental design. In: Proceedings of the 27th international conference on machine learning, pp 1015–1022. https://doi.org/10.1109/TIT.2011.2182033

Swersky K (2017) Improving Bayesian optimization for machine learning using expert priors. PhD Thesis, University of Toronto. http://hdl.handle.net/1807/79505 (accessed date: 1 August 2022)

Tsai SW, Wu EM (1971) A general theory of strength for anisotropic materials. J Compos Mater 5:58–80. https://doi.org/10.1177/002199837100500106

Turner CJ, Crawford RH (2008) Selecting an appropriate metamodel: the case for NURBS metamodels. In: 31st Design automation conference. https://doi.org/10.1115/DETC2005-85043

Viana FAC, Venter G, Balabanov V (2009) An algorithm for fast optimal Latin hypercube design of experiments. Int J Numer Methods Eng 82:135–156. https://doi.org/10.1002/nme.2750

Viana FAC, Haftka RT, Watson LT (2012) Efficient global optimization algorithm assisted by multiple surrogate techniques. J Glob Optim 56:669–689. https://doi.org/10.1007/s10898-012-9892-5

Wang Z, Jegelka S (2018) Max-value entropy search for efficient Bayesian optimization. In: Proceeding of the 34th international conference on machine learning. arXiv:1703.01968

Wang W, Jüttler B, Zheng D, Liu Y (2008) Computation of rotation minimizing frames. ACM Trans Graph. https://doi.org/10.1145/1330511.1330513

Winter J, Fiebig S, Franke T, Vietor T (2019) NURBS-based shape and parameter optimization of structural components with an adaptive amount of control points. In: Proceedings of the 13th world congress of structural and multidisciplinary optimization (WCSMO-13)

Wolpert DH, Macready WG (1996) No free lunch theorems for search. The Santa Fe Institute, p 10

Yang XS (2008) Nature-inspired metaheuristic algorithms, vol 1. Luniver Press, Bristol

Yetilmezsoy K, Sihag P, Kiyan E, Doran B (2021) A benchmark comparison and optimization of Gaussian process regression, support vector machines and M5P tree model in approximation of the lateral confinement coefficient for CFRP-wrapped rectangular/square RC columns. Eng Struct. https://doi.org/10.1016/j.engstruct.2021.113106

Zacchei E, Molina JL (2018) Shape optimization of double-arch dams by using parameters obtained through Bayesian estimators. Iran J Sci Technol Trans Civ Eng 43:649–662. https://doi.org/10.1007/s40996-018-0223-z

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.