Automotive and Engine Technology (2021) 6:15–29
https://doi.org/10.1007/s41104-020-00073-y

ORIGINAL PAPER

A stochastic design optimization methodology to reduce emission spread in combustion engines

Kadir Mourat[1,3] · Carola Eckstein[2] · Thomas Koch[3]

Received: 11 August 2020 / Accepted: 6 November 2020 / Published online: 28 November 2020

Abstract

This paper introduces a method for efficiently solving stochastic optimization problems in the field of engine calibration. The main objective is to make more conscious decisions during the base engine calibration process by considering the system uncertainty due to component tolerances and thus enabling more robust design, low emissions, and avoiding expensive recalibration steps that generate costs and possibly postpone the start of production. The main idea behind the approach is to optimize the design parameters of the engine control unit (ECU) that are subject to uncertainty by considering the resulting output uncertainty. The premise is that a model of the system under study exists, which can be evaluated cheaply, and the system tolerance is known. Furthermore, it is essential that the stochastic optimization problem can be formulated such that the objective function and the constraint functions can be expressed using proper metrics such as the value at risk (VaR). The main idea is to derive analytically closed formulations for the VaR, which are cheap to evaluate and thus reduce the computational effort of evaluating the objective and constraints. The VaR is therefore learned as a function of the input parameters of the initial model using a supervised learning algorithm. For this work, we employ Gaussian process regression models. To illustrate the benefits of the approach, it is applied to a representative engine calibration problem. The results show a significant improvement in emissions compared to the deterministic setting, where the optimization problem is constructed using safety coefficients. We also show that the computation time is comparable to the deterministic setting and is orders of magnitude less than solving the problem using the Monte-Carlo or quasi-Monte-Carlo method.

Keywords Diesel engine · Emission modeling · Design of experiments · Experimental · Gaussian process · Stochastic modeling · Stochastic optimization · Engine calibration · Monte Carlo

Abbreviations

ARD	Automatic relevance determination
CAD a. TDCf	Crank angle degree after top dead center firing
DoE	Design of experiments
ECU	Engine control unit
EGR	Exhaust gas recirculation
GP	Gaussian process
ICE	Internal combustion engine
IQ	Injected quantity
LOOCV	Leave-one-out cross-validation
MI	Main injection
MINLP	mixed-integer nonlinear programming
MC	Monte Carlo
NOx	Nitrogen oxides
NRMSE	Normalized root-mean-square error
OP	Operation point
PI	Pilot injection
PoI	Post injection
QMC	Quasi-Monte Carlo
R^2	Coefficient of determination
SCR	Selective catalytic reduction
SoE	Start of energizing
SE	Squared exponential
VaR	Value at risk

List of symbols

α	Confidence level
μ	Mean
σ_ϵ	Noise variance
σ_s	Signal variance

✉ Kadir Mourat
kadir_mourat@hotmail.de

[1] Oberrahserstraße 53, 41748 Viersen, Germany

[2] Robert Bosch GmbH, Powertrain Solutions, Stuttgart-Feuerbach, Stuttgart, Germany

[3] Institut für Kolbenmaschinen (IFKM), Karlsruhe Institute of Technology, Karlsruhe, Germany

Σ	Sigma algebra
I	Identity matrix
k	Covariance function
$\mathcal{N}$	Normal distribution
s	Safety coefficient

1 Introduction

In light of increasing awareness concerning the detrimental effects of the transportation sector on the environment, lawmakers and consumers demand stricter legal guidelines for motor vehicles. This obliges manufacturers to comply with demanding standards such as low carbon dioxide emissions, little pollutants, and extensive onboard diagnosis functions while expecting high performance and driving comfort [1]. European legislation also stipulates that each vehicle in the field must be compliant with emission regulations, which is verified by random tests on fleet vehicles randomly taken from customers [2]. For manufacturers, this means that their product must not only comply with emission limits but needs also to be robust with regard to them.

A major contributor to the uncertainty of fleet emissions is the component tolerances of the internal combustion engine (ICE). To provide an intuition: deviations between the physical air mass flow and the measured value of the air mass flow meter may lead to a change in the volumetric efficiency of the ICE, which in turn leads to different combustion behavior and combustion products. Manufacturers usually rely on empirical evidence and conservative decisionmaking to account for this. Emission targets are artificially decreased by using safety coefficients to determine a so called enginnering target. Calibration is then performed on an ICE consisting of nominal parts and aims at meeting the engineering target. However, calibration design that is based on an engineering target lacks sound statistical evaluation. It can lead to overly conservative calibration results and leaves room for improvement in quantities of interest such as carbon dioxide, pollutants, or power output. On the contrary, it can also lead to a non-compliant calibration, which is associated with costs and even delays of the start of production. The quantification and consideration of component-related uncertainties, therefore, has the potential to better meet desired optimization targets and reduce costs.

The complex nature of engine calibration tasks [3] with usually 10 and more design parameters generally renders manual calibration suboptimal. Model-based calibration methods have proven to be a successful solution in the past and have become state of the art [1, 3, 4]. Since modeling the entire combustion process using physical models is complex and computationally intensive, statistical models turned out to be a good alternative [3–5]. In particular, Gaussian processes [6] in combination with adapted experimental designs [7] are established in the field.

Reliability-based design is of crucial importance in engineering design. Hence, numerous methods have been proposed to systematically treat uncertainties in product design and carry out stochastic [8–11] or reliability-based [12] design optimization [13–15]. Stochastic methodology has already been applied in powertrain development. [16] and [17] performed stochastic optimization aiming to quantify and minimize uncertainty resulting from different driving cycles. In [18] chance-constrained optimization was performed to improve thermal efficiency. [19] did some early work on stochastic optimization of spark ignition engines considering spark advance and injection time uncertainty. Given the resources at the time, they make use of many linear approximations, which can be problematic given the nonlinear nature of more complex calibration problems. To the authors' knowledge, there is no article addressing stochastic design optimization that considers component tolerances and is suitable for large-scale engine calibration problems. The objective of this paper is, therefore, to develop a methodology that solves this issue and incorporates it into a statistical framework.

The contribution of this article is a stochastic design optimization methodology for engine calibration problems. It is imperative to reduce the computational effort to a manageable level, taking into account the dimension of a typical calibration problem consisting of numerous operating points (OP) that need to be optimized. The proposed method aims to improve calibration results by replacing conventional safety coefficient-based calibration with uncertainty-based optimizations. The uncertainty of the design parameters is derived from the component tolerances employing a priori uncertainty analysis. The original optimization problem is reformulated to minimizing the expectation of a modified objective function taking uncertainties into account while limiting some high or low quantiles of the constraint functions. This is similar to the chance-constrained optimization setting in the literature [20, 21]. Solving this type of stochastic optimization task using Monte Carlo (MC) methods [22] can be computationally expensive for state-of-the-art calibration tasks where the engine models need to be evaluated several times. To address this issue, we approximate the necessary distribution statistics of the quantities of interest using Gaussian Process (GP) regression models. Since the GP expression has an analytical form, we can simplify the stochastic problem into a deterministic one. Thus reducing the computational burden by orders of magnitude.

This article is structured as follows: In Sect. 2 we provide some necessary fundamentals. In Sect. 3 we describe the proposed stochastic design optimization framework. We then introduce the test subject in Sect. 4. In Sect. 5, we discuss the asymptotic properties of the VaR surrogate models. In

Sect. 6, the proposed methodology is applied to an engine calibration problem. First, the performance of the method is contrasted against a state-of-the-art deterministic calibration setting followed by a comparison with the numerical alternative using the quasi-Monte Carlo (QMC) method to quantify the computational effort needed to solve the optimization problem. Finally, the results of the article are discussed in sect. 7.

2 Background

As described above, this section introduces some necessary fundamentals. First, we give some intuition on Gaussian Process regression. We then provide an overview of nonlinear constrained optimization problems.

2.1 Black-box modeling with Gaussian processes

Bayesian regression based on Gaussian processes is an approach for nonparametric modeling where an unknown system $y = f(\mathbf{x}) + \epsilon$ is modeled from some observed data [23–25]. For combustion engine models we usually assume our various system outputs to be uncorrelated. We, therefore, focus on the basic single output supervised learning setting where our goal is to approximate a function $f : \mathcal{X} \subset \mathbb{R}^{d \in \mathbb{N}} \to \mathcal{Y} \subset \mathbb{R}$ from a set of $n \in \mathbb{N}$ observations $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{n}$ [26].

In the Bayesian framework, we assume a GP to be a collection of random variables so that every finite set of these has a joint Gaussian distribution and is completely specified by its mean $\mu(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}_j)$ and can be written as $f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}_j))$. Given noisy observations and assuming the GP to have a zero-mean prior, its joint distribution for an arbitrary input is given as

$$p(y \mid \mathcal{D}) = \mathcal{N}(y|0, \mathsf{K} + \sigma_\epsilon^2 \mathtt{I}_n).,\tag{1}$$

where $\mathsf{K} \in \mathbb{R}^{n \times n}$ is the covariance matrix, σ_ϵ^2 the noise variance, and $\mathtt{I}_n$ the $n \times n$ identity matrix. Given this joint distribution, the predictive distribution of the GP for an arbitrary input $\mathbf{x}$ is specified by its mean and variance

$$p(f \mid \mathbf{x}, \mathcal{D}) = \mathcal{N}(f|\mathbb{E}[f(\mathbf{x})], \mathbb{V}[f(\mathbf{x})]),\tag{2}$$

with

$$\mathbb{E}[f(\mathbf{x})] = k^\top(\mathbf{x})(\mathsf{K} + \sigma_\epsilon^2 \mathtt{I}_n)^{-1}(y_1, \ldots, y_n)^\top,\tag{3a}$$

$$\mathbb{V}[f(\mathbf{x})] = 1 - k^\top(\mathbf{x})(\mathsf{K} + \sigma_\epsilon^2 \mathtt{I}_n)^{-1} k(\mathbf{x}),\tag{3b}$$

where $k(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_1), ..., k(\mathbf{x}, \mathbf{x}_n))^\top$ is a column vector containing the covariance of the input $\mathbf{x}$ and the training samples.

By definition, the covariance function is symmetric positive-definite, i.e. for any set of inputs the resulting covariance matrix is positive-definite. A popular choice for the covariance function is the squared exponential (SE) kernel given by

$$k(\mathbf{x}_p, \mathbf{x}_q) = \sigma_s^2 \exp\left(-\frac{\left\|\mathbf{x}_p - \mathbf{x}_q\right\|_l}{2}\right),\tag{4}$$

where $\sigma_s^2 \in \mathbb{R}_0^+$ is the signal variance and $\|\cdot\|_l$ designates the norm taking into account the characteristic length-scales $l = \{l_i\}$ with $i = 1, \cdots, d$ given by the sum of squares of the product $(\mathbf{x}_p - \mathbf{x}_q) l_i^{-1}$. The infinite differentiability of the SE kernel results in smooth predictive functions. Further intuition on covariance functions is given in [6].

The prior and likelihood of the GP encode our beliefs about the model we wish to learn. With $\sigma_\epsilon, \sigma_s$, and l being the hyperparameters of the model. GP hyperparameters can be learned using different methods such as integration via Markov chain Monte Carlo [25, 27]. Given the above assumptions, the hyperparameters can also be learned by maximizing the log marginal likelihood

$$\begin{aligned}&\log p(y|\sigma_\epsilon, l, \sigma_s)\\ &\quad = -\frac{\mathbf{y}(\mathsf{K} + \sigma_\epsilon^2 \mathtt{I}_n)^{-1}\mathbf{y}^\top + \log|\mathsf{K} + \sigma_\epsilon^2 \mathtt{I}_n| + n \log 2\pi}{2},\end{aligned}\tag{5}$$

with $\mathbf{y} = (y_1, \ldots, y_n)$. For covariance functions with first-order derivatives, this can be solved using gradient-based optimization.

2.2 Probabilistic nonlinear constrained optimization

Many industrial tasks such as base engine calibration are nonlinear constrained optimization problems. They aim at minimizing an objective f on a domain $\mathcal{X}$ subject to equality and inequality constraints. Since equality constraints are of no interest to this particular paper, they shall be disregarded from here on. The general mathematical formulation of a deterministic nonlinear constrained optimization problem is, therefore, given by

$$\min_{\mathbf{x}} f(\mathbf{x})\tag{6a}$$

$$\text{s.t.} c_i(\mathbf{x}) \leq 0, i = 1, \ldots, n_c\tag{6b}$$

where f is the cost function that needs to be minimized, which is subject to constraints c. This kind of problem classifies as a convex or non-convex mixed-integer nonlinear programming (MINLP) problem depending on the space $\mathcal{X}_{feasible} \subseteq \mathcal{X}$ of feasible solutions. More intuition on MINLP problems and an overview of methods for solving these are provided in [28, 29]. Most real-world applications, however, are subject to uncertainties ranging from environmental and operating conditions to manufacturing tolerances. In this probabilistic setting, the system inputs are assumed to be random quantities.

Definition 1 Let X be a multivariate random variable on a probability space (Ω, Σ, P) with sample space Ω, sigma-algebra Σ, and probability measure P. Then $Y = f(X)$ is a transformation of X under f and again a random variable on Ω. Its cumulative distribution function is given by

$$P(Y \leq y) = P(f(X) \leq y). \tag{7}$$

Given this, a possible probabilistic formulation of the nonlinear constrained optimization problem in equation 6a and 6bis given by the individual chance-constraint optimization setting

$$\min_{x} \mathbb{E}[f(X)] \tag{8a}$$

$$\text{s.t.} P(c_i(X) \leq 0) \geq \alpha_i, i = 1, \dots, n_c \tag{8b}$$

where the new objective is to minimize the first statistical moment of the distribution $f(X)$ subject to probabilistic constraints. Solving these kinds of stochastic MINLP problems involves the evaluation of multi-dimensional integrals for the objective and constraints, which can be computationally expensive even for low dimensional problems. Often a more feasible alternative, especially for high dimensional problems is, therefore, the utilization of sampling techniques such as MC sampling [22]. This involves drawing samples from the distribution X, evaluating these, and approximating the distribution $f(X)$. However, evaluating a high number of samples for optimization tasks involving a large number of function evaluations is computationally infeasible. A popular approach to tackle this issue is to simplify the stochastic problem into a deterministic task [30].

3 The stochastic design optimization framework

In this section, we propose a framework for stochastic design optimization, that aims at quantifying and considering component tolerances during the model-based calibration process. The methodology aims to find optimal design parameters for a system subject to uncertainty by solving a probabilistic constrained optimization problem. An necessity for the methodology is short computing time. This results from the requirement to keep the powertrain development cycle as short as possible. The overall objective of the methodology is to replace conventional safety factors using a priori uncertainty analysis and improve the calibration results. Figure 1 depicts the general structure of the framework.

3.1 Uncertainty analysis

In general, there are two types of uncertainties [31]: While pure random effects such as geometric tolerances due to imprecisions during production are classified as aleatoric uncertainty, systematic effects like material aging are epistemic. Both effects need to be considered in an uncertainty analysis.

To take input uncertainty into account during optimization, it has to be determined beforehand. The combustion process of an engine is mainly determined by the design parameters or the set ECU parameters. Manufacturing tolerances lead to deviations from the set values and in turn to changes in quantities such as fuel consumption and emissions. Since the design parameters of our engine models are not directly subject to uncertainty, we have to perform an uncertainty analysis that maps relevant sources of uncertainty such as manufacturing tolerances to the design parameters. This is achieved by performing an uncertainty analysis, where uncertainty propagation is used to determine the design parameter uncertainty.

Figure 2 depicts an excerpt from the uncertainty analysis we performed for the calibration problem in Sect. 6. To convey an intuition here: using the results of tolerance investigations on a set of injectors, we determine the uncertainty distribution of the injected quantity (IQ) of a single pilot injection (PI), which depends on the nominal rail pressure and nominal IQ. Since we cannot measure the emissions of each individual cylinder, we sample all cylinders and assume the sample mean to represent the final quantity that results in the measured emissions.

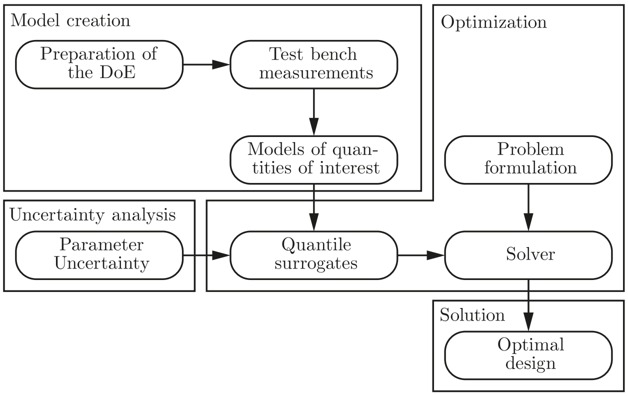

Fig. 1 Schematic representation of the framework

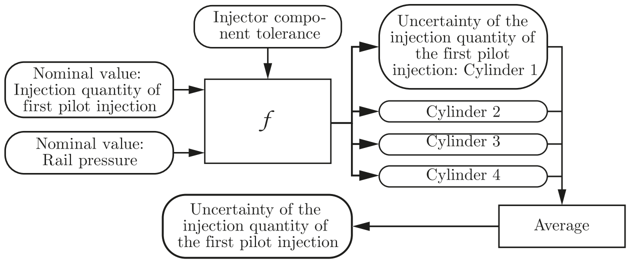
Injector component tolerance
Nominal value: Injection quantity of first pilot injection
Nominal value: Rail pressure
f
Uncertainty of the injection quantity of the first pilot injection: Cylinder 1
Cylinder 2
Cylinder 3
Cylinder 4
Average
Uncertainty of the injection quantity of the first pilot injection

Fig. 2 Excerpt from the uncertainty analysis, which illustrates the influence of the injector component tolerance on the injected quantity of the first pilot injection

3.2 Optimization problem formulation

As mentioned before, the engine calibration task is a constrained optimization problem. The aim is to minimize an objective function while making sure that some constraints are met for the entire vehicle fleet. A potential formulation for a nonlinear constrained optimization problem was given by equations 8a and 8b. This is a reasonable formulation for the combustion engine setting since it also considers the tails of the target distributions. For example, if a vehicle with a nominal ICE barely complies with emission limits, then a high number of vehicles will be surely non-compliant.

Definition 2 The value at risk of a multivariate random variable X given a confidence level $\alpha \in (0, 1)$ is given by

$$\mathrm{VaR}_\alpha(X) = \inf\{x \in \mathbb{R} : \alpha \le P(X \le x)\}. \tag{9}$$

We can reformulate equation 8b using the VaR and write the optimization problem as

$$\min_x \mathrm{VaR}_{0.50}(f(X)) \tag{10a}$$

$$\mathrm{s.t.}\mathrm{VaR}_{\alpha_i}(f(X)) \le c_{c,i}, i = 1, \dots, n_c, \tag{10b}$$

where VaR_{α_i} is a probabilistic constraint function and $c_{c,i}$ the corresponding constraint.

3.3 Surrogate model of the value at risk

As already described above, base calibration tasks are usually high dimensional MINLP problems. The associated uncertainty of the design parameters is not necessarily uniform or Gaussian and often correlated. This renders cheap approximations such as Most Probable Points [32] useless. Therefore, sampling is often the only viable way to approximate distribution statistics. However, base calibration problems usually require millions of function evaluations, making the problem extremely expensive in terms of computing time and impractical to solve. Generally, a viable way of reducing the computational burden resulting from expensive experiments or simulations is to train a surrogate model that reproduces the behavior of the experiment or simulation [33]. The idea is to create a black-box model mapping the parameters of the experiment or the simulation to the desired outputs. If such a model can be created with good model quality and using a limited number of measurements or simulation runs, then it can reduce the computational burden of optimization substantially. The resulting black-box model is relatively cheap to evaluate. In our case, we wish to train a model $g : \mathbf{x} \mapsto \mathrm{VaR}_\alpha(X)$ that maps the nominal input x to the corresponding VaR_α. Our implementation uses GPs to learn the underlying VaR. Given a set of n observations $\mathcal{E} = \{x_i, \mathrm{VaR}_\alpha(X)_i\}_{i=1}^n$, the expectation of the GP surrogate model is expressed as

$$\mathrm{VaR}_\alpha(\mathbf{x}) = k^\top(x)(\mathsf{K} + \sigma_\epsilon^2 \mathrm{I}_n)^{-1} \mathrm{VaR}_\alpha^\top, \tag{11}$$

where VaR_α is the vector containing the empirical VaR values from the observations. The empirical VaR is estimated by drawing m independent samples from a Sobol sequence [34–37]. The assumption of continuous domain needed for GPs holds true if the VaR exists. Further intuition on the approximation of distributional statistics is given by [38].

3.4 Implementation details

We have implemented the framework as a plug-in for the above-mentioned commercial software ASCMO as Matlab code. This has the advantage that ASCMO already offers several possibilities for modeling, optimization, and analysis.

For this work, in particular, we have also implemented the methodology in Python. This is necessary to ensure a high level of transparency in the calculations. We train all models using the GPRegression module from the GPy library [39]. Maximum likelihood estimates are calculated using the implemented stochastic conjugate gradient algorithm. For creating Sobol quasirandom sequences we use an implementation written by [40], which is available at https://people.sc.fsu.edu/~jburkardt/py_src/sobol/sobol.html. We use the sequential least squares programming algorithm from the scientific computing library [41] to solve the MINLP problems. Correlated random variables are generated using Cholesky decomposition, see Appendix A.

Table 1 Technical data of the research engine

Power output and torque	110kW & 340N m
Stroke volume	1.7l
Injection system	Bosch solenoid CRS2-25, 7 holes with Needle Closing Control [43]
Charging system	Turbocharger with variable geometry turbine
Exhaust-gas recirculation (EGR) system	Cooled high-pressure EGR + low-pressure EGR system
Exhaust-gas after-treatment system	Diesel oxidation catalyst + close-coupled selective catalytic reduction (SCR) catalyst on a diesel particulate filter + underfloor SCR catalyst with ammonium-slip coating

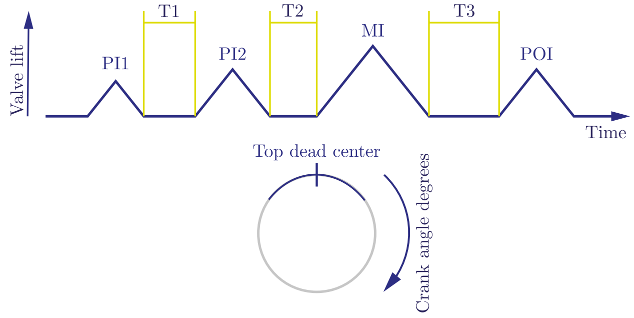

Fig. 3 Qualitative representation of the crankshaft position of a diesel engine, illustrating the injection pattern for the investigated OP

Table 2 Boundaries of the design of experiments

Design parameter	Lower bounds	Upper bounds
Intake air mass (mg)	250	375
Boost pressure (kPa)	105	125
Swirl flap actuation (–)	0.10	0.45
Common rail pressure (MPa)	45	90
IQ of the first PI (mg)	0.8	2
IQ of the second PI (mg)	0.8	3
IQ of the PoI (mg)	1	2.5
SOE of the MI (CAD a. TDCf)	1	9
T1 (μs)	500	1000
T2 (μs)	500	1000
T3 (μs)	550	900

4 Test subject

The ICE we use for our investigations serves as a platform for research purposes at Robert Bosch GmbH [42]. The test subject is an inline four-cylinder light-duty diesel engine for passenger cars that is equipped with a direct injection system. The ICEs main characteristics are given in Table 1.

For monitoring purposes, thermodynamic quantities such as temperatures, pressures, and mass flows are measured. The exhaust gas is extracted upstream of the exhaust gas after-treatment system. Nitrogen oxide (NO$_x$) emissions are measured via chemiluminescence. Soot is measured using a high-resolution smoke meter [44]. The combustion noise is calculated using an in-cylinder pressure-based method [45].

4.1 The engine model

In general, the combustion process in a diesel engine is determined by the OP and design parameters. The OP is defined by the engine speed and engine torque. The control or design parameters consist of air supply and fuel supply parameters. The air supply parameters are intake air mass, exhaust gas recirculation rate, swirl flap actuation, and boost pressure. The fuel supply parameters are common rail pressure, injected quantities of the pilot and post injections (PoI), the start of energizing (SOE) of the main injection (MI) in Crank angle degree after top dead center firing (CAD a. TDCf) and the distances between each injection in seconds, see Figure 3.

For the investigations, we create models of the relevant outputs that are sensitive with respect to the parameters named above. For the sake of simplicity, the OP is fixed to 2000min^{-1} engine speed and 400kPa brake mean effective pressure. The model is based on stationary test bench measurements from a design of experiments (DoE) that we set up using the commercial software package ASCMO [46]. The DoE samples are drawn from a Sobol sequence on $(\mathbf{x}^{lower}, \mathbf{x}^{upper}) \in \mathbb{R}^d$, with d being the number of input dimensions or design parameters and $\mathbf{x}^{lower}$ and $\mathbf{x}^{upper}$ being vectors that represent the range within which the design variables can be varied, see Table 2. For this measurement campaign, the EGR rate was not actively varied during the experiments. This leads to maximum EGR rates and avoids parameter combinations producing high nitrogen oxide emissions. The low-pressure EGR valve and throttle valve were fully open during the experiments. The injection quantity of the MI is not listed, as it results from the other parameters and the load. It is important to note that we assume the test subject to consist of nearly nominal components since we aim to create models for the nominal engine.

We use $n = 675$ measurements to create a model of each target quantity. The outputs are modeled independently using zero-mean prior GP regression models with additive Gaussian noise and automatic relevance determination (ARD) SE kernel [6, 47]. The model hyperparameters are estimated by maximizing the log marginal likelihood. In order to make meaningful design decisions through optimization, the model quality must be high. This is especially true for a stochastic optimization scenario. A common issue with supervised learning algorithms such as GPs is *overfitting* [48, 49]. This is basically the phenomenon of the algorithm memorizing the training data and not generalizing properly. An overfit model makes perfect predictions for the training data. When confronted with unknown test data, though, the predictions will be arbitrary. Although GPs under the premise of large amounts of data are generally robust against overfitting [6], the model can still overfit when maximizing the log marginal likelihood. One way of ruling out overfitting phenomena is to validate the model on unseen test data using proper performance metrics such as the normalized root-mean-square error (NRMSE) and the coefficient of determination (R^2), see Appendix B. Since in our case we only have a limited amount of training samples, we need to validate the model without the use of test samples. A valid way to accomplish this is to use the leave-one-out cross-validation (LOOCV) method, see Appendix C.

Figure 4 shows two scatter plots derived from the nitrogen oxide model. The left plot shows the observation and prediction value pairs. The model quality is excellent and the error between prediction and observation is in the measurement error range. The plot to the right shows value pairs consisting of observed values and model predictions using the leave-one-out method. The correspondence is very good and indicates that the model generalizes well.

Table 3 NRMSE, R^2, NRMSE LOOCV, and R^2 LOOCV values of the engine models

Target	NRMSE	R^2	NRMSE LOOCV	R^2 LOOCV
NO_x (g/kWh)	0.0003	0.997	0.012	0.995
Soot (FSN)	0.0011	0.980	0.047	0.948
Noise (dBA)	0.0015	0.954	0.033	0.968

Table 3 shows the NRMSE LOOCV, and R^2 LOOCV results of the three engine models. We see that model quality is high for each of the three models with R^2 values close to 1 and the error is within the measurement accuracy range. The RMSE LOOCV and R^2 LOOCV are similar to the RMSE and R^2, which indicates good generalization for each of the three models.

4.2 System uncertainty

In this section, we show the results of the uncertainty analysis we have performed for the above-described test subject. Combustion engines are subject to a variety of uncertainties, ranging from environmental and operating conditions to manufacturing tolerances. In this article, we focus on manufacturing tolerances in the most relevant components since these are the main contributors to the overall uncertainty during stationary operation.

Table 4 shows a list of the design parameters with the corresponding uncertainty distributions, which result from the uncertainty analysis we have performed for the engine.

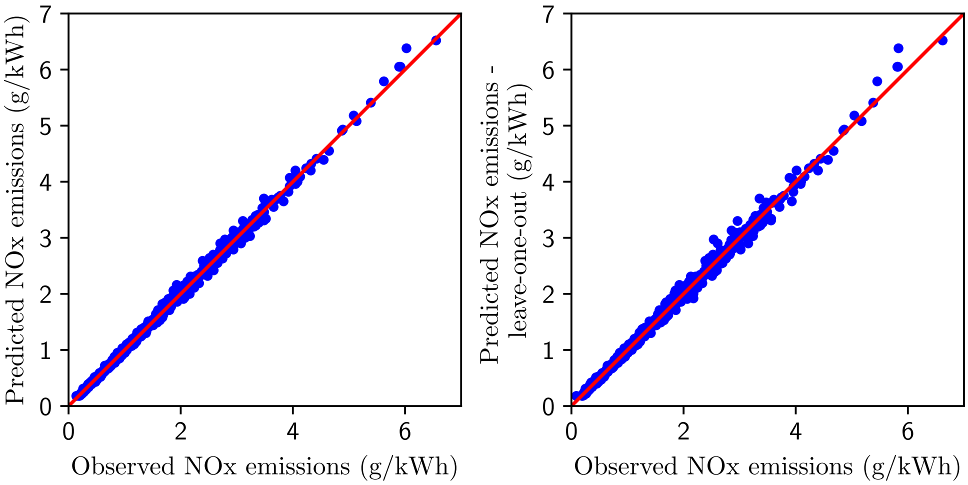

Fig. 4 Scatter plot derived from the nitrogen oxide model showing predicted vs. observed value pairs for the training data (left) and the leave-one-out method (right)

Table 4 Results of uncertainty analysis

Design parameter	Distribution	Correlation matrix										
Intake air mass (mg)	$\mathcal{N}$(0,12.25)	1	0	0	0	0	0	0	0	0	0	0
Boost pressure (kPa)	$\mathcal{N}$(0,0.49)	0	1	0	0	0	0	0	0	0	0	0
Swirl flap actuation (–)	$\mathcal{N}$(0,0.0001)	0	0	1	0	0	0	0	0	0	0	0
Common rail pressure (MPa)	$\mathcal{N}$(0,1)	0	0	0	1	0	0	0	0	0	0	0
IQ of the first PI (mg)	$\mathcal{N}$(-0.05,0.135)	0	0	0	0	1	1	1	0	0	0	0
IQ of the second PI (mg)	$\mathcal{N}$(-0.05,0.135)	0	0	0	0	1	1	1	0	0	0	0
IQ of the PoI (mg)	$\mathcal{N}$(-0.05,0.135)	0	0	0	0	1	1	1	0	0	0	0
SOE of the MI (CAD a. TDCf)	$\mathcal{N}$(0,0.09)	0	0	0	0	0	0	0	1	0	0	0
T1 (µs)	$\mathcal{N}$(0,12.5)	0	0	0	0	0	0	0	0	1	1	1
T2 (µs)	$\mathcal{N}$(0,12.5)	0	0	0	0	0	0	0	0	1	1	1
T3 (µs)	$\mathcal{N}$(0,12.5)	0	0	0	0	0	0	0	0	1	1	1

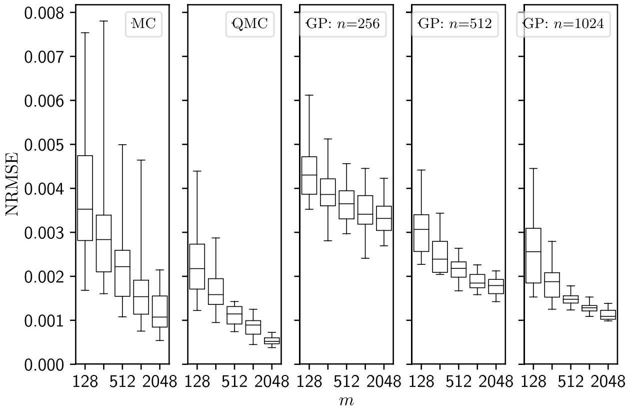

Fig. 5 Performance of the GP surrogate with $n = 256$, 512, and $n = 1024$ training data points, of MC sampling and QMC sampling as a function over m. The performance is measured using the NRMSE on a set of $n_{test} = 100$ test samples. The failure probability is $1 - \alpha = 0.05$

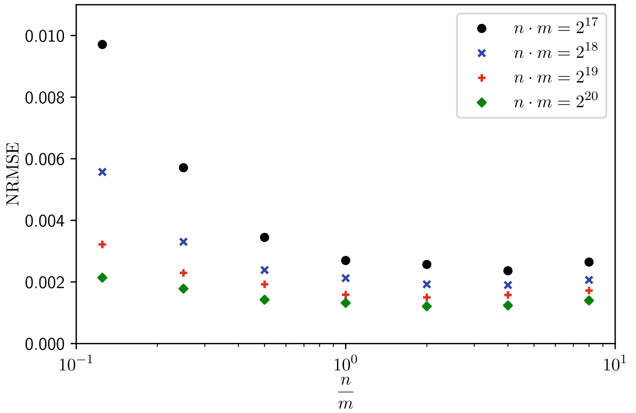

Fig. 6 Performance of the GP surrogate with $n \cdot m = 2^{17}$ (black circle), $n \cdot m = 2^{18}$ (blue cross), $n \cdot m = 2^{19}$ (red plus) and $n \cdot m = 2^{20}$ (green diamond) function calls required for training over the ratio $\frac{n}{m}$. The performance is measured using the NRMSE on a set of $n_{test} = 100$ test samples

5 Asymptotic properties of the VaR surrogates

In this section, we discuss the asymptotic properties of the VaR surrogate models. We investigate the performance of the GP surrogates with respect to the number of training samples n and the number of random samples m used to calculate the VaR. We further investigate which ratio of n and m is most efficient in terms of computation for a given engine out soot emission model. Figure 5 shows the performance of three GP surrogates with $n = 256$, 512, and 1024 training data points as a function of m. The surrogate models are trained assuming zero-mean prior ARD SE kernel and additive Gaussian noise. The input parameter uncertainty is given in Table 2 and the chosen confidence level is 0.95. The training data is randomly drawn from the multivariate uniform distribution given in Table 2. The plot also shows the performance of empirical estimates using pure MC and QMC. The predictive performance is measured on a separate test data set with $n_{test} = 100$ using the NRMSE. The box plots are the result of repeating this experiment with 50 random initializations. Since the training samples are drawn randomly, the trained model differs each time. Similarly, the MC and QMC results vary since the bootstrap is initialized randomly. The residual error e_i for each test sample is the difference of the prediction and an empirical VaR using $m_{test} = 10^6$ quasirandom samples that is close to the *true* VaR.

MC sampling shows significant variance in performance. QMC sampling shows the best performance with a significantly lower variance than the MC method. The GP surrogate models show low variance in performance, even with small n and m. Given a sufficiently high amount of training samples, the GP surrogate shows good performance only second to the QMC method. [38] have already argued that under certain conditions for a given confidence level a GP-based quantile estimate is asymptotically consistent and efficient. These conditions being: $\mathcal{X}$ is a convex and compact subset of $\mathbb{R}^d$, the training data is independent and identically distributed in $\mathcal{X}$, the assumption of continuous domain for the VaR holds and there exists a ratio $\frac{n}{m}$ that is efficient in terms of function evaluations. They established a bound under which as $n, m \to \infty$ the mean of the GP posterior almost surely converges to the true VaR at the fastest rate possible in terms of function evaluations.

Figure 6 depicts the GP surrogate performance as a function of the ratio $\frac{n}{m}$ for an increasing amount of function calls required for training. It is evident that model performance increases as $n, m \to \infty$. For this model, $\frac{n}{m} \approx 2$ seems to be a good choice to maximize model performance.

6 Application to an engine calibration problem

To demonstrate the fitness and strength of the proposed framework, we apply it to the engine calibration problem described above. For the sake of simplicity, the problem is formulated for a single OP. In practice, it is common to optimize several interconnected OPs simultaneously [50]. For this OP, we aim to minimize nitrogen oxide emissions while ensuring that soot and noise emissions are kept below an upper bound for 95% of all realizations. We further wish to find a solution inside the initial parameter space $(x^{lower}, x^{upper}) \in \mathbb{R}^d$ described by Table 2, since the GPs we construct are not suited for extrapolation.

Using the engine models we constructed in Sect. 4.1, we train surrogate models to emulate the VaR. We sample $n = 500$ training data points from the above mentioned bounded parameter space and draw $m = 1000$ QMC samples from the distribution given by Table 4 to estimate the empirical VaR for each point. This equals $\frac{n}{m} = 0.5$ and is only slightly worse than $\frac{n}{m} = 2$ performance-wise. However, considering the time complexity of a standard GP of $\mathcal{O}(n^2)$, the computation time needed for a prediction is reduced fourfold. The corresponding mean estimate of the emulator can be expressed according to equation 11. Figure 7 shows two scatter plots derived from the VaR surrogate of the soot model. The left plot shows observation and prediction value pairs for the training data. The right plot shows observation and prediction value pairs for $n_{test} = 100$ test data points. The correspondence is nearly perfect for the training and test samples.

Table 5 NRMSE, R^2, NRMSE LOOCV, and R^2 LOOCV values of the engine models

Target	NRMSE (training)	R^2 (training)	NRMSE (test)	R^2 (test)
NO$_x$ (g/kWh)	0	1	0.0001	1
Soot (FSN)	0	1	0.0002	1
Noise (dBA)	0.0001	1	0.0011	0.9975

Table 5 shows the NRMSE and R^2 results for all three surrogate models. The surrogates emulate the VaR perfectly, even for the unseen test samples, and generalize well.

6.1 Problem formulation

To assess the benefit of our methodology, we contrast it to the deterministic setting using safety coefficients. Both tasks can be formulated as MINLP problems. The deterministic scenario is expressed as

$$\min_{x} f^{\,\mathrm{NO_x}}(\mathbf{x}) \tag{12a}$$

$$\text{s.t.}\, l^{\,\mathrm{Soot}} * s^{\,\mathrm{Soot}} - f^{\,\mathrm{Soot}}(\mathbf{x}) \leq 0, \tag{12b}$$

$$l^{\,\mathrm{Noise}} - s^{\,\mathrm{Noise}} - f^{\,\mathrm{Noise}}(\mathbf{x}) \leq 0, \tag{12c}$$

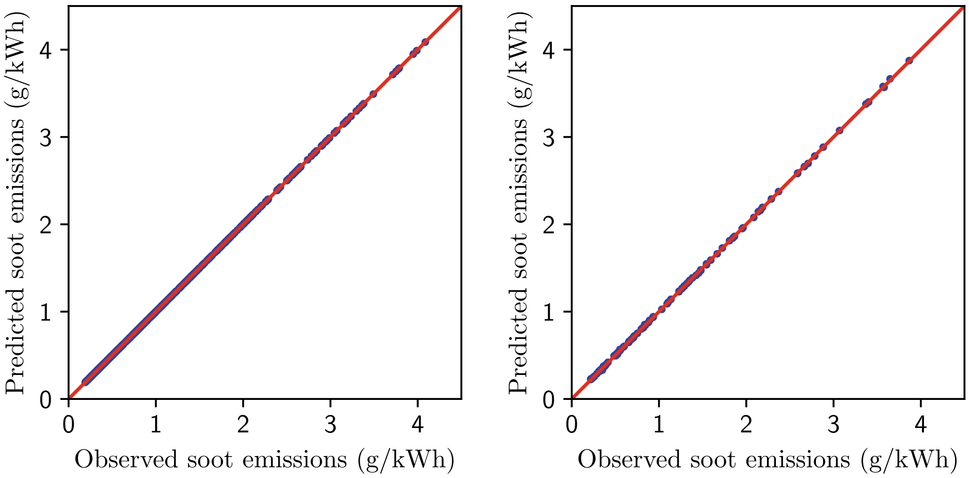

Fig. 7 Scatter plot derived from the soot model showing predicted vs. observed value pairs for the training data (left) and test data (right)

Table 6 Constraints used for optimization

Target	l	s	α
Soot	0.5FSN	0.7	0.95
Noise	86dBA	1dBA	0.95

$$x_i^{lower} \leq x_i \leq x_i^{upper} \forall i = 1, \cdots, d \tag{12d}$$

where f is the predictive mean of the engine model, l is an upper limit for admissable emissions, and s is a safety coefficient. The safety coefficients are empirical values based on expert knowledge accounting for tolerance behavior. The settings we chose for this particular problem are given in Table 6. The given safety coefficients are the state of the art choice for the given calibration task.

The corresponding problem formulation for the stochastic setting is given by

$$\min_{\mathbf{x}} \mathrm{VaR}_{0.50}^{\mathrm{NO_x}}(\mathbf{x}) \tag{13a}$$

$$\mathrm{s.t.} l^{\mathrm{Soot}} - \mathrm{VaR}_{\alpha}^{\mathrm{Soot}}(\mathbf{x}) \leq 0, \tag{13b}$$

$$l^{\mathrm{Noise}} - \mathrm{VaR}_{\alpha}^{\mathrm{Noise}}(\mathbf{x}) \leq 0, \tag{13c}$$

$$x_i^{lower} \leq x_i \leq x_i^{upper} \forall i = 1, \cdots, d. \tag{13d}$$

6.2 Optimization results

In this section, we present the optimization results. The Sequential Least-Squares Quadratic Programming algorithm [51] is used for solving the above-formulated problems. The method can be used to solve MINLP problems, which are subject to equality and inequality constraints and to lower and upper bounds on the model parameters. The optimization problem is solved for $n_{init} = 20$ quasirandom initialization in an attempt to find a global optimum. The function tolerance is set to 10^{-5}. For the remaining settings, the default values are selected. Analytical formulations for the GP derivatives are provided for the stochastic setting.

Figure 8, Figure 9 and Table 7 show the optimization results for both scenarios. Although the ultimate goal is the same, we see two distinct results. It is evident that the prior assumptions we made for the deterministic case are too conservative. The $\mathrm{VaR}_{0.95}$ for soot and noise is well below the actual upper bound we were aiming for. By selecting smaller safety coefficients, an improvement in nitrogen oxide emissions would have been possible for this particular setting. On the other hand, for a different choice of design parameters, a smaller value could also lead to non-compliance with the constraints. Without prior knowledge of the output variance, a sufficiently large margin must be chosen.

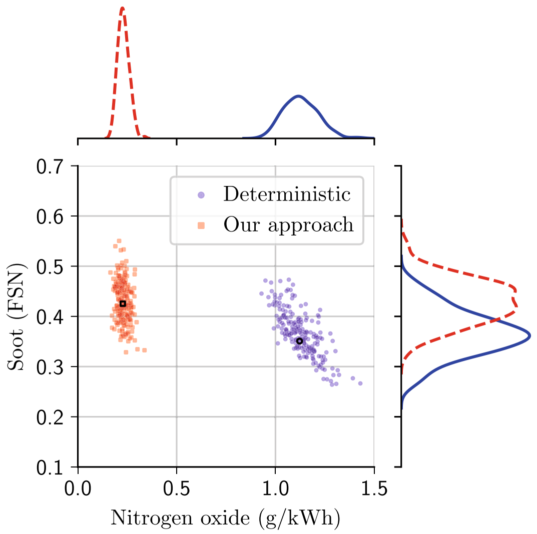

Fig. 8 Nominal optimization results for nitrogen oxide and soot emission of the deterministic (black circle) and stochastic (black rectangle) solution. And 200 samples drawn from the target distribution at these settings for both the deterministic (blue circle) and stochastic solution (red rectangle) with corresponding probability density estimates

In contrast to the deterministic setting, our probabilistic approach renders a good estimate for the VaR. This way, we obtain a solution that meets the desired boundary conditions almost exactly. The expected $\mathrm{NO_x}$ emissions are reduced from 1.129g/kWh to 0.281g/kWh. This corresponds to a decrease of 79.54%. The expected soot and noise emissions are slightly higher at 0.430FSN and 85.841dBA but do comply with the limits for 95% of the realizations.

Table 7 Optimization results for the different optimization scenarios. The various statistics were calculated using 10^6 QMC samples

	$\mathrm{NO_x}$ (g/kWh)			Soot (FSN)			Noise (dBA)		
	Nom.	Mean	$Q_{0.95}$	Nom.	Mean	$Q_{0.95}$	Nom.	Mean	$Q_{0.95}$
Deterministic	1.122	1,129	1,279	0,351	0,367	0,450	85,000	85,015	85,236
Our approach	0.227	0,231	0,281	0,425	0,430	0,504	85,350	85,383	85,841
QMC	0,218	0,220	0,268	0,432	0,436	0,505	85,483	85,520	85,977

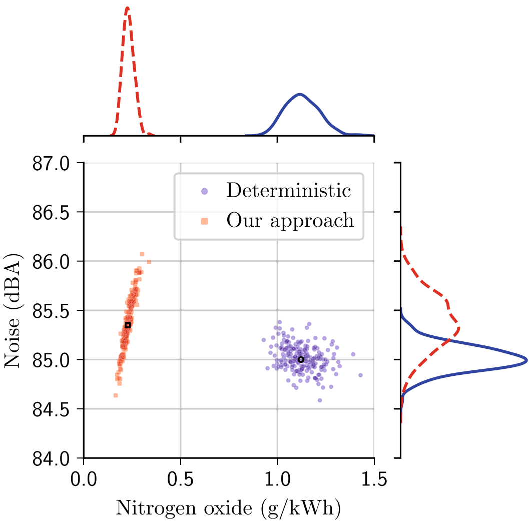

Fig. 9 Nominal optimization results for nitrogen oxide and noise emission of the deterministic (black circle) and stochastic (black rectangle) solution. And 200 samples drawn from the target distribution at these settings for both the deterministic (blue circle) and stochastic solution (red rectangle) with corresponding probability density estimates

6.3 Gain in computing time

In this section, we compare the computation time of our method with the deterministic setting and a stochastic QMC equivalent. The aim is to keep computing time as low as possible. The reason for this is that in practice multiple OPs need to be optimized simultaneously and a high number of constraints must be satisfied. At the same time, the cost function needs to be minimized for a high number of driving trajectories, which leads to high numbers of function evaluations [16]. As mentioned above, the optimization problem is solved for $n_{init} = 20$ quasirandom initialization in an attempt to find a global optimum. All computations are carried out on a machine with an Intel Xeon E3-1505M CPU, 32 GB Ram, and an Nvidia Quadro M2000M GPU. We first compare the time needed to compute the optimization task of the deterministic setting with our approach. The computation time for both methods is 15.87s and 9.26s respectively. The time difference is mainly due to the higher prediction costs of the original engine models. As mentioned above, the time complexity of a standard GP is of $\mathcal{O}(n^2)$. We further assess the amount of computation time that is saved compared to the setting where the stochastic problem is solved numerically. We, therefore, compare our method with the QMC method. To ensure similar accuracy as with the surrogate model, 500 quasirandom samples are drawn for estimating the empirical VaR. The optimization results for both solutions are given in Table 7. The QMC method finds a similar optimum as our approach. Differences in the solutions result mainly from the termination criteria of the optimizer. Model inaccuracy and numerical errors account for slight differences only. The theoretical time complexity for both methods should be of $\mathcal{O}(n^2 n_e m)$, where n is the number of training samples of the respective models, m is the number of draws needed to evaluate the VaR, and n_e is the number of total function calls needed for optimization. If we assume that both algorithms need the same amount of function evaluations to find the optimum, our approach in this scenario should be approximately 900 times faster than the QMC method. The measured computing time for our approach and the QMC method is 9.26s and 1691.76s respectively. Our surrogate-based method is $\sim$ 183 times faster than the QMC approach. The difference to the theoretically possible computational time gain is due to the higher impact of load times on the overall shorter computing times of our approach. For larger optimization tasks, the impact should be significantly lower and the computational time gain is expected to be closer to the theoretical time gain. The time gain we obtain for the simplified problem we have solved may seem noteworthy. As mentioned before though, real-world calibration problems involve significantly more function evaluations and optimization usually takes several hours. In view of this, we can achieve a considerable reduction in computing time and enable large-scale optimizations without the need for parallel computing on multiple machines.

6.4 Conclusions

Stochastic design optimization was performed for a representative engine calibration problem in a single OP using the proposed method. The performance of our method was evaluated against the corresponding deterministic optimization scenario using safety coefficients. Compared to the deterministic scenario, it was possible to reduce the expected nitrogen oxide emissions for the OP by 79.54%. The imposed soot and noise constraints and design space boundaries were met in both cases. The reduction in nitrogen oxide emissions is mainly due to the accurate estimation of emission distributions. Due to the analytical form of the GP surrogates, evaluating the VaR is extremely cheap and computing time is comparable to the deterministic method.

In a second step, we contrasted our method to the numerical equivalent solution using QMC sampling. The global optimum was found in both cases with slight numerical differences. Our method showed a decrease in computing time by a factor of 183. This reduction is likely to be even greater for large-scale problems. Due to the low computing time, the method can be applied to large-scale engine calibration problems without the need for parallel computing.

7 Summary and outlook

In this paper, we proposed and demonstrated a method for efficiently solving large-scale stochastic optimization problems subject to nonlinear constraints. Our main motivation for developing the method is to employ it in the field of base engine calibration. The premise of our method is that a model of the system under study exists, which can be evaluated cheaply. Furthermore, it is essential that the stochastic optimization problem can be formulated such that the objective function and the constraint functions can be expressed using the VaR. The main idea is to derive analytically closed formulations for the VaR, which are cheap to evaluate and thus reduce the computational effort of evaluating the objective and constraints. The VaR is therefore learned as a function of the input parameters of the initial model using a supervised learning algorithm. For this work, we employ Gaussian process regression models.

For demonstration purposes, we have employed the methodology on an engine base calibration problem given a realistic optimization scenario in a single OP. We also performed a state-of-the-art deterministic optimization given a realistic engineering target. For the given scenario the objective was to minimize nitrogen oxide emissions while complying with an upper limit for soot and noise emissions. The optimization results show that the consistent treatment of the problem as a statistical one helps to maximize performance compared to the deterministic approach. Our stochastic approach reduces nitrogen oxide emissions by almost 80% compared to the deterministic scenario. The constraints on soot and noise emissions are complied with. This significant decrease shows a high potential for the reduction of overall raw emissions for practical application of the methodology.

In a second step, we demonstrated the computational efficiency of the method by comparing the computing time of the optimization process to the deterministic case and an equivalent numerical QMC solution. Our method performed on par with the deterministic scenario timewise. It reduced computational effort by nearly 200 times compared to the QMC alternative while converging on the same global optimum.

In this article, we focused on base calibration problems. Another application of uncertainty quantification in the field of engine calibration could be to use uncertainty analysis to assess the overall system uncertainty. This would facilitate decision making in the selection of engine components.

An interesting topic for future research is the extension of the method to larger calibration problems, where the entire operating range of the engine consisting of more than one OP is considered and optimized simultaneously. This adds additional uncertain parameters such as route, traffic, and driving style to the probabilistic problem [17]. A major challenge for solving this kind of task is to create global engine models that are sensitive with regard to the design parameters and are able to capture engine dynamics at the same time. However, recent proposals seem to be promising [52, 53].

Acknowledgements The authors would like to thank Jürgen Frick, Alexandra Fritsch, and Andre Wiedersberg, for their support while conducting the engine test bench measurements used for this work. The authors would also like to thank Robert Bosch GmbH for the resources provided to produce this work.

Funding Open Access funding enabled and organized by Projekt DEAL..

On generating correlated random variables

Let $\mathbf{Z} : \Omega \rightarrow \mathbb{R}^n$ be a $n \times 1$ normal random vector on a probability space (Ω, Σ, P) consisting of correlated random variables $(Z_1, \cdots, Z_n)$. Given the mean $\boldsymbol{\mu}_Z$ and the positive-definite covariance C_Z, we wish to generate samples for $\mathbf{Z}$. Let $\mathbf{X} : \Omega \rightarrow \mathbb{R}^n$ be a $n \times 1$ random vector with $(X_1, \cdots, X_n)$ being independent normal random variables with mean $\boldsymbol{\mu}_X = 0$ and covariance $\mathsf{C}_X = \mathsf{I}_n$. With I_n being the $n \times n$ identity matrix. Then we can write $\mathbf{Z}$ as

$$\mathbf{Z} = \mathbf{A} + \mathsf{B}\mathbf{X}, \tag{14}$$

with B yet to be determined and

$$\mathbf{A} = \boldsymbol{\mu}_Z - \mathsf{B}\boldsymbol{\mu}_X = \boldsymbol{\mu}_Z. \tag{15}$$

The covariance of $\mathbf{Z}$ is defined as

$$\begin{aligned} \mathsf{C}_Z =& \mathbb{E}[(\mathbf{Z} - \mathbb{E}(\mathbf{Z}))(\mathbf{Z} - \mathbb{E}(\mathbf{Z}))^\top] = \mathbb{E}[\mathsf{B}\mathbf{X}\mathbf{X}^\top\mathsf{B}^\top] \\ =& \mathsf{B}\mathsf{I}_n\mathsf{B}^\top = \mathsf{B}\mathsf{B}^\top. \end{aligned} \tag{16}$$

If the covariance of Z is an $n \times n$ real-valued symmetric matrix, it can be expressed as

$$\mathsf{C}_Z = \mathsf{Q}\Lambda\mathsf{Q}^\top = \mathsf{Q}\sqrt{\Lambda\Lambda^\top}\mathsf{Q}^\top, \tag{17}$$

where Q is the eigenmatrix of C_Z and Λ a diagonal matrix with the entries being the eigenvalues of C_Z. Using equation 16 and 17, we can write

$$\mathsf{B} = \mathsf{Q}\sqrt{\Lambda} \tag{18}$$

and compute it as the Cholesky decomposition of C_Z. Using $\mathbf{A}$ and B we can calculate $\mathbf{Z}$ given samples $\mathbf{X}$.

Performance metrics

A widely used metric that measures the difference between measured data points y and corresponding model predictions $\hat{y}$ is the NRMSE. It measures the normalized average quadratic error of the model predictions. Since the error is squared, it is disproportionately large for high error values, making it sensitive to outliers. An NRMSE of 0 means a perfect model fit. We denote the NRMSE as

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2}}{y_{max} - y_{min}}, \tag{19}$$

where $e = \hat{y} - y$ is the residual and y_{min} and y_{max} are the minimum and maximum values of the observations $\mathbf{y}$.

The coefficient of determination is another commonly used metric. It measures the error of the model proportional to the variation in the data. The coefficient of determination can be described as

$$\mathrm{R}^2 = 1 - \frac{\sum_{i=1}^{n} e_i^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}. \tag{20}$$

A R^2 of 1 means a perfect model fit.

Leave-one-out cross-validation

A valid way of evaluating the performance of a model without the need for test samples is to use the leave-one-out cross-validation method. The idea is to first split the training inputs X consisting of n samples into n sets $\{\mathsf{X}_{-1}, \cdots, \mathsf{X}_{-n}\}$, where X_{-i} is the set containing all training samples but the ith. In a second step the model is trained on $(\mathsf{X}_{-i}, \mathbf{Y}_{-i})$, with $\mathbf{Y}$ being the respective outputs. And a prediction is made for the ith training sample. This way, the error of the model for the ith training point can be determined for all $i \in \mathbb{N}$, with the predicted values not being used for estimation. The calculation cost of this method is usually prohibitive, but in the case of GP regression, there are suitable approximations. One way to compute a numerical approximation to this is described in [54] and is written as

$$\hat{y}_{-i} = y_i - w_{ii}^{-1} v_i, \tag{21}$$

where w_{ii} is the ith diagonal entry of the $n \times n$ Woodbury identity denoted as

$$\mathsf{w} = (\mathsf{K} + \sigma_\epsilon^2 \mathrm{I}_n)^{-1} \tag{22}$$

and v_i is the ith entry of the $1 \times n$ vector

$$\mathbf{v} = (\mathsf{K} + \sigma_\epsilon^2 \mathrm{I}_n)^{-1}\mathbf{y}. \tag{23}$$

The ith residual of the LOOCV approximation is, therefore, given by

$$e_i = \hat{y}_{-i} - y_i = -w_{ii}^{-1} v_i \tag{24}$$

References

1. Langouët, H., Métivier, L., Sinoquet, D., Tran, Q.H.: Optimization for Engine Calibration. In: EngOpt 2008 : International conference on engineering optimization, pp. 1–5. E-Papers Servicos Ed. Ltda., Rio de Janeiro, Brazil (2008). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf
2. European Commission: commission regulation (EC) 692/2008 of 18 July 2008 implementing and amending Regulation (EC) No 715/2007 of the European Parliament and of the Council on type-approval of motor vehicles with respect to emissions from light passenger and commercial vehicles (Euro 5 and Euro 6) and on access to vehicle repair and maintenance information (2008). OJ L199. https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A32008R0692
3. Berger, B.: modeling and optimization for stationary base engine calibration. Ph.D. thesis, Technical University of Munich, Munich, Germany (2012). http://mediatum.ub.tum.de/?id=1108936
4. Beatrice, C., Napolitano, P., Guido, C.: Injection parameter optimization by doe of a light-duty diesel engine fed by bio-ethanol/RME/diesel blend. Appl Energy **113**, 373–384 (2014). https://doi.org/10.1016/j.apenergy.2013.07.058
5. Friedrich, C., Auer, M., Stiesch, G.: Model based calibration techniques for medium speed engine optimization: Investigations on common modeling approaches for modeling of selected steady state engine outputs. SAE Int J Eng **9**(4), 1989–1998 (2016). https://doi.org/10.4271/2016-01-2156
6. Rasmussen, C.E., Williams, C.K.I.: Gaussian processes for machine learning Adaptive computation and machine learning. MIT Press, Cambridge (2006). http://www.gaussianprocess.org/gpml/chapters/RW.pdf
7. Röpke, K., von Essen, C.: DoE in engine development. Qual Reliab Eng Int **24**(6), 643–651 (2008). https://doi.org/10.1007/978-1-4614-0237-4
8. Birge, J.R., Louveaux, F.: Introduction to stochastic programming. Springer, Berlin. **2**, (2011). https://doi.org/10.1007/978-1-4614-0237-4
9. Chen, Y., Yuan, Z., Chen, B.: Process optimization with consideration of uncertainties—an overview. Chin J Chem Eng **26**(8), 1700–1706 (2018). https://doi.org/10.1016/j.cjche.2017.09.010
10. Kleijnen, J.P.C.: Kriging metamodeling in simulation: a review. Eur J Oper Res **192**(3), 707–716 (2009). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf0

11. Lucor, D., Enaux, C., Jourdren, H., Sagaut, P.: Stochastic design optimization: application to reacting flows. Comput Methods Appl Mech Eng **196**, 5047–5062 (2007). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf1
12. Bertsimas, D., Brown, D., Caramanis, C.: Theory and applications of robust optimization. SIAM Rev **53**(3), 464–501 (2011). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf2
13. Calfa, B., Grossmann, I., Agarwal, A., Bury, S., Wassick, J.: Data-driven individual and joint chance-constrained optimization via kernel smoothing. Comput Chem Eng **78**, 51–69 (2015). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf3
14. Ning, C., You, F.: Data-driven adaptive nested robust optimization: general modeling framework and efficient computational algorithm for decision making under uncertainty. AIChE J **63**(9), 3790–3817 (2017). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf4
15. Yao, W., Chen, X., Luo, W., van Tooren, M., Guo, J.: Review of uncertainty-based multidisciplinary design ooptimization methods for aerospace vehicles. Prog Aerosp Sci **47**(6), 450–479 (2011). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf5
16. Kolmanovsky, I., Siverguina, I., Lygoe, B.: Optimization of powertrain operating policy for feasibility assessment and calibration: stochastic dynamic programming approach. pp. 1425–1430. IEEE, New York City, NY, USA (2002). https://doi.org/10.1109/ACC.2002.1023221
17. Wasserburger, A., Hametner, C., Didcock, N.: Risk-averse real driving emissions optimization considering stochastic influences. Eng Optim (2019). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf6
18. Shen, X., Shen, T.: Chance-constrained optimization for torque tracking control with improving fuel economy in spark-ignition engines. SICE J Control Measure Syst Integ **11**(4), 365–371 (2018). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf7
19. Rizzo, G., Pianese, C.: A stochastic approach for the optimization of open-loop engine control systems. Annal Oper Res **31**, 545–568 (1991). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf8
20. Charnes, A., Cooper, W.W.: Chance-constrained programming. Manage Sci **6**(1), 73–79 (1959). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.618.7080&rep=rep1&type=pdf9
21. Miller, B.L., Wagner, H.M.: Chance constrained programming with joint constraints. Oper Res **13**(6), 879–1060 (1965). https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A32008R06920
22. Metropolis, N., Ulam, S.: The Monte Carlo Method. J Am Stat Assoc **44**(247), 335–341 (1949). https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A32008R06921
23. Mackay, D.J.C.: Gaussian processes—a replacement for supervised neural networks? (1997). https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A32008R06922
24. Williams, C.K.I.: Prediction with Gaussian processes: from linear regression to linear prediction and beyond. Springer, Berlin (1998). pp. 599–621. https://doi.org/10.1007/978-94-011-5014-9_23
25. Williams, C.K.I., Rasmussen, C.E.: Gaussian processes for regression. In: D.S. Touretzky, M.C. Mozer, M.E. Hasselmo (eds.) Proceedings of the 8th international conference on neural information processing systems, NIPS'95, pp. 514–520. Max-Planck-Gesellschaft, MIT Press, Cambridge (1995). https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A32008R06923
26. Shalev-Shwartz, S., Ben-David, S.: Understanding machine learning: from theory to algorithms. Cambridge University Press, Cambridge (2014). https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A32008R06924
27. Neal, R.M.: MCMC using Hamiltonian dynamics. Chapman & Hall / CRC Press, Boca Raton. (2011). https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A32008R06925
28. Belotti, P., Kirches, C., Leyffer, S., Linderoth, J., Luedtke, J., Mahajan, A.: Mixed-integer nonlinear optimization. Acta Numerica **22**, 1–131 (2013). https://doi.org/10.1017/S0962492913000326
29. Kronqvist, J., Bernal, D.E., Lundell, A., Grossmann, I.E.: A review and comparison of solvers for convex MINLP. Opt Eng **20**(2), 397–455 (2019). https://doi.org/10.1007/s11081-018-9411-87
30. Paiva, R., Crawford, C., Suleman, A.: A robust and reliability-based design optimization framework for wing design. AIAA J (2014). https://doi.org/10.2514/1.J0521618
31. Bjarnadottir, S., Li, Y., Stewart, M.G.: Chapter nine-climate adaptation for housing in hurricane regions, pp. 271–299. Butterworth-Heinemann, Oxford (2019). https://doi.org/10.1016/B978-0-12-816782-3.00009-79
32. Du, X., Chen, W., Wang, Y.: Most probable point-based methods. Springer, Berlin (2010). https://doi.org/10.1007/978-1-4419-6606-3_70
33. Hwang, J.T., Martins, J.R.: A fast-prediction surrogate model for large datasets. Aerosp Sci Technol **75**, 74–87 (2018). https://doi.org/10.1016/j.ast.2017.12.0301
34. Sobol, I.M.: On the distribution of points in a cube and the approximate evaluation of integrals. USSR Comput Math Mathe Phys **7**(4), 86–112 (1967). https://doi.org/10.1016/0041-5553(67)90144-92
35. Sobol, I.M.: Sensitivity estimates for nonlinear mathematical models. Math Model Comput Exp **1**(4), 407–414 (1993)
36. Sobol, I.M.: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math Comput Simul **55**(1), 271–280 (2001)
37. Sobol, I.M., Asotsky, D., Kreinin, A., Kucherenko, S.: Construction and comparison of high-dimensional sobol' generators. Wilmott **2011**(56), 64–79 (2011). https://doi.org/10.1002/wilm.100563
38. Plumlee, M., Tuo, R.: Building accurate emulators for stochastic simulations via quantile kriging. Technometrics **56**(4), 466–473 (2014). https://doi.org/10.1080/00401706.2013.8609194
39. The GPy authors: GPy: A Gaussian process framework in python (2012). http://mediatum.ub.tum.de/?id=11089365
40. Chisari, C.: SOBOL—The sobol quasirandom sequence (2011). http://mediatum.ub.tum.de/?id=11089366
41. Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Jarrod Millman, K., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C., Polat, İ., Feng, Y., Moore, E.W., Vand erPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1. 0 Contributors: SciPy 1.0–fundamental algorithms for scientific computing in python (2019)
42. Krüger, M., Bareiss, S., Kufferath, A., Naber, D., Ruff, D., Schumacher, H.: Further optimization of nox emissions under the eu 6d regulation. In: M. Bargende, H.C. Reuss, A. Wagner, J. Wiedemann (eds.) 19. Internationales stuttgarter symposium, pp. 876–895. Springer, Berlin (2019)
43. Wintrich, T., Rothe, S., Bucher, K., Hitz, H.J.: Diesel injection system with closed-loop control. MTZ Worldwide **79**(9), 54–59 (2018). https://doi.org/10.1007/s38313-018-0062-07
44. AVL List GmbH: emission measurement instruments - AVL smoke meter (2018). http://mediatum.ub.tum.de/?id=11089368

45. Maurya, R.K.: Knocking and combustion noise analysis. Springer, Berlin. pp. 461–542. (2019). https://doi.org/10.1007/978-3-030-11954-6_9
46. Hoffmann, S., Schrott, M., Huber, T., Kruse, T.: Model-based methods for the calibration of modern internal combustion engines. MTZ Worldwide **76**(4), 24–29 (2015). https://doi.org/10.1007/s38313-014-1024-99
47. Neal, R.M.: Bayesian learning for neural networks, lecture notes in statistics. vol. 118, 1 edn. Springer, Berlin (1996). https://doi.org/10.1007/978-1-4612-0745-0
48. Andrews, J.L.: Addressing overfitting and underfitting in gaussian model-based clustering. Comput Stat Data Anal **127**, 160–171 (2018). https://doi.org/10.1016/j.apenergy.2013.07.0580
49. Ying, X.: An overview of overfitting and its solutions. J Phys Conf Ser **1168**, 022022 (2019). https://doi.org/10.1088/1742-6596/1168/2/0220221
50. Schillinger, M., Mourat, K., Hartmann, B., Eckstein, C., Jacob, M., Kloppenburg, E., Nelles, O.: Modern Online DoE Methods for Calibration Constraint Modeling, Continuous Boundary Estimation, and Active Learning. Expert Verlag, Tübingen, Germany (2017). https://doi.org/10.1016/j.apenergy.2013.07.0582
51. Kraft, D.: Software package for sequential quadratic programming. Tech Rep DFVLR-FB 88-28, DLR German Aerospace Center—Institute for Flight Mechanics, Cologne, Germany (1988)
52. Zimmer, C., Meister, M., Nguyen-Tuong, D.: Safe active learning for time-series modeling with gaussian processes. In: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (eds.) Advances in neural information processing systems 31, pp. 2730–2739. Curran Associates, Inc. (2018). https://doi.org/10.1016/j.apenergy.2013.07.0583
53. Schillinger, M., Hartmann, B., Jacob, M.: Dynamic safe active learning for calibration. In: Röpke, C., Gühmann, C. (eds.) International conference on calibration methods and automotive data analytics. pp. 258–277. Expert Verlag, Tübingen, Germany (2019)
54. Vehtari, A., Tolvanen, V., Mononen, T., Winther, O.: Bayesian leave-one-out cross-validation approximations for Gaussian latent variable models. J Mach Learn Res 17:1–38 (2016). https://doi.org/10.1016/j.apenergy.2013.07.0584