
Surrogate Modelling with Sequential Design for Expensive Simulation Applications

Joachim van der Herten, Tom Van Steenkiste,
Ivo Couckuyt and Tom Dhaene

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/67739>

Abstract

The computational demands of virtual experiments for modern product development processes can get out of control due to fine resolution and detail incorporation in simulation packages. These demands for appropriate approximation strategies and reliable selection of evaluations to keep the amount of required evaluations were limited, without compromising on quality and requirements specified upfront. Surrogate models provide an appealing data-driven strategy to accomplish these goals for applications including design space exploration, optimization, visualization or sensitivity analysis. Extended with sequential design, satisfactory solutions can be identified quickly, greatly motivating the adoption of this technology into the design process.

Keywords: surrogate modelling, sequential design, optimization, sensitivity analysis, active learning

1. Introduction

Amongst the countless research domains and the fields of research revolutionized by the merits of computer simulation, the field of engineering is a prime example as it continues to benefit greatly from the introduction of computer simulations. During product design, engineers encounter several complex input/output systems, which need to be designed and optimized. Traditionally, several prototypes were required to assure quality criteria that were met or to obtain optimal solutions for design choices and to evaluate the behaviour of products and components under varying conditions. Typically, a single prototype is not sufficient, and lessons

learnt are used to improve the design, back at the drawing table. Therefore, the development process used involves building several prototypes in order to gain more confidence in the solutions. A direct consequence of this approach is that the development process is both slow and not cost effective.

The introduction of computer simulations caused revolution: by bundling implementations of material, mechanical and physical properties into a software package, simulating the desired aspects of a system and performing the tests and experiments virtually; the number of required prototypes can be drastically reduced to only a few at the very end of the design process. These prototypes can be regarded purely as a validation of the simulations. The simulation itself can be interpreted as a *model* and serves as an abstract layer between the engineer and the real world. Performing a virtual experiment is faster and is very inexpensive. A direct consequence was an acceleration of development and system design, contributing to a shorter time-to-market and a more effective process in general. In addition, it was also possible to perform more virtual experiments, providing a way to achieve better products and design optimality.

However, as simulation software became more precise and gained accuracy over the years, its computational cost grew tremendously. In fact, the growth of computational cost was so fast it has beaten the growth in computational power resulting in very lengthy simulations on state-of-the-art machines and high performance computing environments, mainly due to the never ending drive for finer time scales, more detail and general algorithmic complexity. For instance, a computational fluid dynamics (CFD) simulation of a cooling system can take several days to complete [1], or a simulation of a single crash test was reported to take up to 36 hours to complete [2]. This introduces a new problem: large-scale parameter sweeping and direct use of this type of *computationally expensive* simulations for evaluation intensive tasks such as optimization and sensitivity analysis are impractical and should be avoided.

To counter this enormous growth in computational cost however, an additional layer of abstraction between the complex system in the real world and the engineer was proposed, more specifically between the simulation and the engineer. Rather than interacting directly with the simulator, a cheaper approximation is constructed. Roughly three approaches to obtain this approximation exist:

- **Model driven** (known as model order reduction) takes a top-down approach by applying mathematic techniques to derive approximations directly from the original simulator. This, however, exploits information about the application domain and is therefore problem specific.
- **Data driven** assumes absolutely nothing is known about the inner workings of the simulator. It is assumed to be a *black-box* and information about the response is collected from evaluations. From these data, an approximation is derived. Because this approach is very general, it is not bound to a specific domain.
- **Hybrid** is the overlap zone between both model driven and data driven. Attempts to incorporate domain-specific knowledge into a data-driven process to obtain better traceability and reach accuracy with less evaluations.

In this chapter, the focus is on a data-driven approach: the response of a simulator as a function of its inputs is mimicked by *surrogate models* (metamodels, emulators or response surface models are often encountered as synonyms). These cheap-to-evaluate mathematical expressions can be evaluated efficiently and can replace the simulator. A more extensive overview of usage scenarios and a formal description of surrogate modelling are given in Section 2. Before the surrogate model can be used, it must first be constructed and trained during the surrogate modelling process on a number of well-chosen evaluations (*samples*) to be evaluated by the simulator in order to satisfy the requirements specified upfront. The problem of selecting an appropriate set of samples is further explored in Section 3. Section 4 briefly introduces an integrated platform for surrogate modelling with sequential design. Finally, these techniques are demonstrated on three use-cases in Section 5.

2. Surrogate modelling

The introduction of this chapter highlighted the global idea of approximation and the benefits of introducing an additional layer of abstraction when simulations are expensive. Now, the data-driven approach (surrogate modelling) is discussed more in depth, both from a usability point of view and a more formal description of the technique.

2.1. Goals and usage scenarios

The most direct implementation of surrogate modelling is training a globally accurate model over the entire design space. This approximation can then replace the expensive simulation evaluations for a variety of engineering tasks such as design space exploration, parameterization of simulations or visualization.

A different use-case of surrogate models is sensitivity analysis of the complex system. Especially when many input parameters are present, it is very difficult to achieve global accuracy due to the exponential growth of the input space (known as the *curse of dimensionality*). Fortunately, not all input parameters contribute equally to the output variability, in fact some might not have any impact at all [3]. The surrogate models can be used directly for evaluation-based sensitivity analysis methods such as Sobol indices [4], Interaction indices [5] or gradient-based methods. For some kernel-based modelling methods, analytical computation of sensitivity measures is possibly resulting in faster and more reliable estimation schemes, even before global accuracy is achieved [6].

Another task at which surrogates excel is the optimization of expensive objective functions. This discipline is often referred to as surrogate-based optimization (SBO). A globally accurate surrogate model can be built and optimized using traditional optimization methods such as gradient descent, or metaheuristics such as particle swarm optimization [7]. Although this approach is correct and works faster than simulating each call of the objective function, it is not necessarily the most efficient methodology: when seeking a minimum, less samples can be devoted to the regions that are clearly shown to be the opposite. This results in specific

methodologies, which explore the search space for optima and exploit the available knowledge to refine optima. This is a difficult trade-off and will be discussed in detail in Section 3.2.

All tasks described so far were forward tasks, mapping samples from a design space to the output or objective space. It is also possible to do the opposite: this is referred to as inverse surrogate modelling. This can be interpreted as identifying the areas of the design space corresponding to a certain desired or feasible output range. Typical approaches involve training a (forward) surrogate model first, then optimizing the model using an error function between the output and the desired output as objective function. This optimization is often preferred to be a robust optimization to account for the error of the forward surrogate model [8]. Specific sampling schemes to identify these regions directly were also proposed [9]. Finally, it is also possible to translate the inverse problem into a forward problem involving discretizing the output (feasible/infeasible point) and learning the class boundaries.

Because the concept of surrogate modelling is both flexible as well as generic, allowing several modifications tailored for the task at hand, it has been applied in wide range of fields including metallurgy, economics, operations research, robotics, electronics, physics, automotive, biology, geology, etc.

2.2. Formalism

Formally, the surrogate modelling process can be described as follows. Given an expensive function f and a collection of data samples with corresponding evaluations represented by D , we seek to find an approximation function \tilde{f} :

$$\begin{aligned} \arg \max_{t \in T} \arg \min_{\theta \in \Theta} -\Lambda(\kappa, \tilde{f}_{t,\theta} D) \\ \text{subject to } \Lambda(\kappa, \tilde{f}_{t,\theta} D) \leq \tau. \end{aligned} \quad (1)$$

It is clear that the selection of the approximation function is a complex interaction of several aspects, more specifically: κ represents an error function such as the popular root-mean-square error (RMSE), τ the target value for the quality as expressed by the error function, all under the operation of the model quality estimator Λ . The quality estimator drives the optimization of both the model type t out of the set of available model types T and its hyperparameters θ . Typical examples of surrogate model types are Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), Radial Basis Functions (RBFs), rational and polynomial models, Kriging and Gaussian Process (GP) models. Examples of hyperparameter optimization include tuning kernel parameters or regularization constants or identifying the optimal order of a polynomial or the most appropriate architecture for a neural network.

The choice of Λ is therefore crucial to obtain a satisfying surrogate model at the end of the process as it is the metric driving the search for θ . This aspect is often overlooked at first, requiring several iterations of the process to obtain satisfactory results. Consulting the users of the surrogate model and defining what is expected from the model and what is not, is a good starting point. These requirements can then be formally translated into a good quality estimator. Unfortunately, defining Λ does not end with casting user requirements. Often, hyperparameters have a direct impact on the model complexity or penalization thereof, thus tuning them is tricky due to the bias-variance trade-off [10].

For instance, a straightforward approach is minimizing the error between the surrogate model response and the true responses for the samples used to train the surrogate. This is often referred to as *training error* or *sample error* and pushes the hyperparameter optimization to favour complex models interpolating the data points perfectly. Although this solution might be considered satisfactory at first sight, in reality this rarely provides a good model as the optimization problems do not consider model quality at other, unobserved samples in the design space. This approach inevitably leads to very unreliable responses when these unobserved samples are to be predicted, hence the model is said to be *overfitting* or to have poor *generalization performance*. Popular quality estimators accounting for generalization performance include crossvalidation and validation sets.

3. Experimental design

A typical requirement for the surrogate model is to be *accurate*, hence the objective of the modelling process is the ability to obtain accuracy with only a small number of (expensive) simulator evaluations. A different kind of requirement is optimizing the response of the simulator. This requires fast discovery of promising regions and fast exploitation thereof to identify the (global) optimum. Hence, a different strategy for selecting the samples is required as the accuracy of the surrogate in non-optimal is of lesser importance. A refined set of model requirements and the goal of the process are required, as they greatly affect the choice of samples to be evaluated. The choice of samples is referred to as the experimental design.

3.1. One-shot design

The traditional approaches to generate an experimental design are the *one-shot* designs. Prior to any evaluation, all samples are selected in a space-filling manner: at this point, no further information is available due to the black-box assumption on the simulator itself (as part of the data-driven approach). Therefore, the information density should be approximately equal over the entire design space and the samples are to be distributed uniformly. To this end, several approaches related to *Design of Experiments* (DoEs) have been developed. However, only the space-filling aspect has an impact within in the context of computer experiments, as other criteria such as blocking and replication lose their relevance [11]. This led to the transition and extension of these existing statistical methods to computer experiments [11, 12]. Widely applied are the factorial designs (grid-based) [13] and optimal (maximin) Latin Hypercube designs (LHDs) [14]. Both are illustrated in **Figures 1** and **2**, respectively, for a two-dimensional input space and 16 samples. An LHD avoids collapsing points should the input space be projected into a lower dimensional space. Other approaches include maximin and minimax designs, Box and Behken [15], central composite designs [16] and (quasi-)Monte Carlo methods [17–19].

Despite their widespread usage, these standard approaches to generate experimental designs come with a number of disadvantages. First and foremost: the most qualitative designs (with the best space-filling properties) can be extremely complex to generate (especially for problems with a high-dimensional input space) due to their geometric properties. For instance, gener-

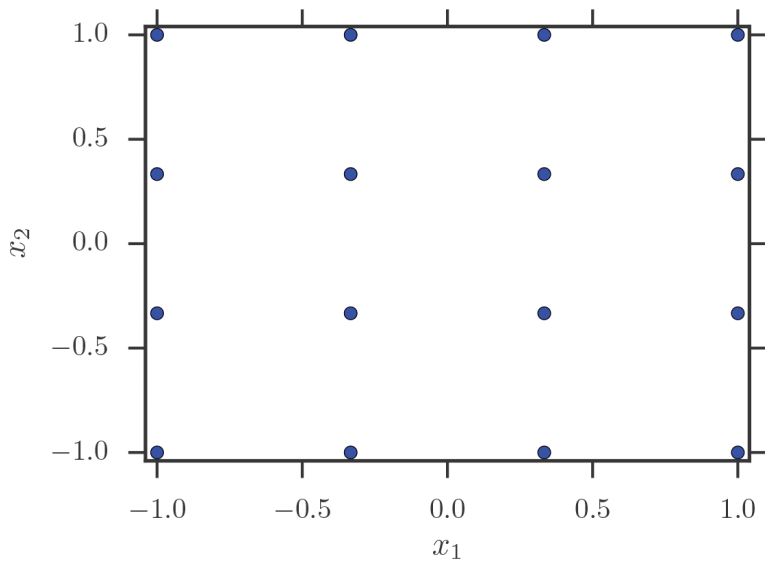


Figure 1. Two-dimensional factorial design with four levels per dimension.

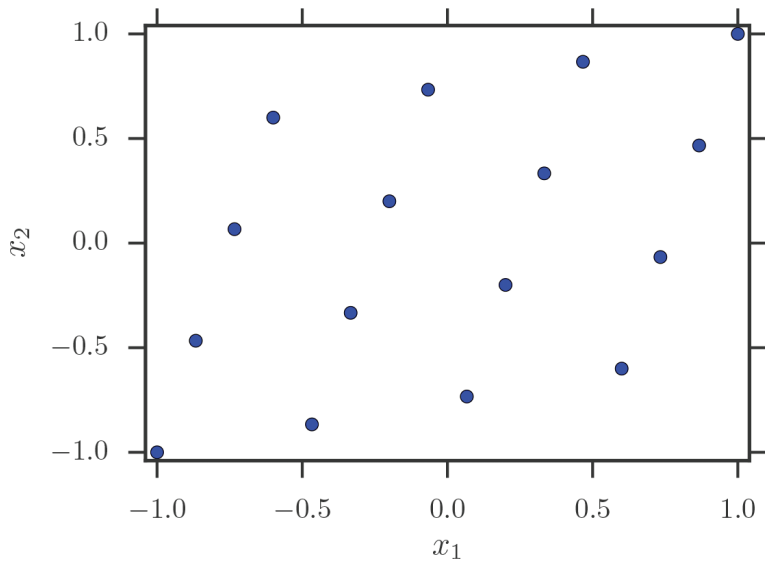


Figure 2. Two-dimensional optimal maximin Latin Hypercube design of size 16.

ating an LHD with optimal maximin distance is very time-consuming process. In fact, the generation of an optimal LHD is almost a field of its own, with several different methods for faster and reliable generation [14, 20]. Fortunately, once a design is generated, it can be reused.

For some other design methodologies, it is not possible to generate them for an arbitrary size. Given the expensive nature of each evaluation, this can result in an unacceptable growth of required simulation time. Factorial designs for instance always have size k^d with level k and dimension d , making them infeasible choices for problems with many input parameters.

Another disadvantage of one-shot methodologies is the arbitrary choice of size of the design. The choice should depend entirely on the nature of the problem (i.e. larger design spaces with more complex behaviour require more evaluations). However, this information is unavailable at the time the design is generated. Hence, one-shot approaches risk selecting too few data points resulting in an underfitted model, or selecting too much data points causing loss of time and computational resources.

3.2. Sequential experimental design

As a solution, sequential design was adopted [21]. This methodology starts from a very small one-shot design to initiate the process. After evaluation of these samples a model is built, and a loop is initiated which is only exited when either one of the specified stopping criteria is met. Within the loop, an *adaptive sampling* algorithm is run to select additional data points for evaluation which are used to update the model. **Figure 3** displays this process graphically.

This approach has a number of advantages. First of all, constraints on the surrogate modelling process can be explicitly imposed through the stopping criteria. Typical criteria include how well the model satisfied the model requirements, a maximum number of allowed evaluations, or a maximum runtime. Second, the *adaptive sampling* method can be designed to select new data points specifically in terms of the requirements. Sampling to obtain a globally accurate model will differ from sampling to discover class boundaries or sampling to obtain optima. These choices can also be guided by all information available about the input-output behaviour: when n samples have been selected, a history of intermediate models and all simulator responses is avail-

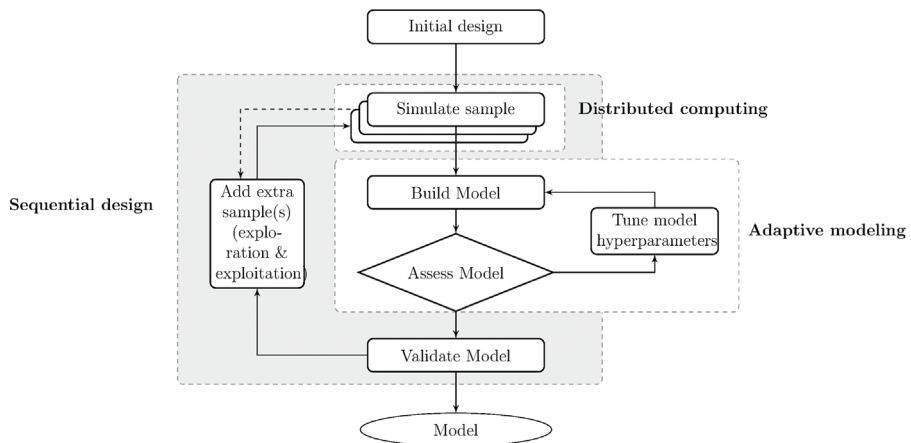


Figure 3. Surrogate modelling with sequential design.

able to guide the selection of new samples. Because of the information available, this selection no longer has to be purely based on a black-box approach, and information can be exploited.

Roughly, all methods for adaptive sampling are based on any of the following criteria (discussed more in detail below):

- Distance to neighbouring points (space-filling designs)
- Identification of optima
- Model uncertainty
- Non-linearity of the response
- Feasibility of the candidate point w.r.t. constraints

Depending on the goal and model requirements, a strategy can be designed involving a complex combination of these criteria. Fundamentally, each approach will involve two competing objectives:

1. **Exploration:** sampling regions of the design space where proportionally only little information has been acquired.
2. **Exploitation:** sampling promising (w.r.t to the goal) regions of the design space.

It is clear, however, that a good strategy strikes a balance between these goals as both are required to obtain satisfactory results.

Exploration-based algorithms are typically less involved with the goal of the process. They are crucial to assure that no relevant parts of the response surface are completely missed. Roughly the space-filling and model uncertainty criteria focus mostly on exploration. Space-filling sequential experimental design usually involves distance to neighbouring points, e.g. the maximin/minimax criteria, potentially complemented with projective properties [22]. Model uncertainty is either explicitly available, or must be derived somehow. Bayesian model types represent the former type of models, e.g. the prediction variance of Kriging and Gaussian Process models, which can be applied directly for maximum variance sampling [23, 24] or maximum entropy designs [25]. For these kinds of models, a better way is expressing the uncertainty on the model hyperparameters resulting in approaches to reduce this uncertainty and hence, enhancing the overall model confidence [26]. Model uncertainty can also be derived by training several models and comparing their responses. Areas with most disagreements are then marked for additional samples. This can be a very effective approach in combination with ensemble modelling.

On the other hand, exploitation methods clearly pursue the goal of the process. In case a global accurate model is required, a very effective approach is raising the information density in regions with non-linear response behaviour (e.g. LOLA-Voronoi [27], FLOLA-Voronoi [28]). The latter approach does not even require intermediate models as it operates on local linear interpolations. For optimization purposes, specific adaptive sampling

methodologies can be applied, depending on the specific tasks. For single-objective optimization examples of such sampling methods include CORS [29] and Bayesian optimization acquisition functions such as Expected Improvement [30] (combined with Kriging models this corresponds to the well-known Efficient Global Optimization approach [31]), the Knowledge Gradient [32] and Predictive Entropy Search [33]. Many of these methods for optimization can also be used in combination with a method, which learns about the feasibility of input regions of the design space: during the iterative process an additional model learns the feasibility from the samples (as reflected by the simulation thereof). This information is then used during the selection of new samples with specific criteria such as the Probability of Feasibility (PoF) [34].

Surrogate-based optimization has also been extended to problems with two or more (potentially conflicting) objectives. The goal of this type of multi-objective optimization is the identification of a Pareto front of solutions, which presents the trade-off between these objectives. Existing approaches include hypervolume-based methods such as the Hypervolume Probability of Improvement (HvPoI) [35, 36], the Hypervolume Expected Improvement [37] or multi-objective Predictive Entropy Search [38].

4. SUMO Toolbox

Designed as a research platform for sequential sampling and adaptive modelling using MATLAB, the SUMO Toolbox [39, 40] has grown into a mature design tool for surrogate modelling, offering a large variety of algorithms for approximation of simulators with continuous and discrete output. The software design is fully object oriented allowing high-extensibility of its capabilities. By default, the platform follows the integrated modelling flow with sequential design but can also be configured to approximate data sets, use a one-shot design, etc.

The design goals of the SUMO Toolbox support approximation of expensive computer simulations of complex black-box systems with several design parameters by cheap-to-evaluate models, both in a regression and a classification context. To obtain these goals, the SUMO Toolbox offers sequential sampling and adaptive modelling in a highly configurable environment, which is easy to extend due to the microkernel design. Distributed computing support for evaluations of data points is also available, as well as multi-threading to support the usage of multi-core architectures for regression modelling and classification. Many different plugins are available for each of the different sub-problems, including many of the algorithms and methods mentioned in this chapter.

The behaviour of each software component is configurable through a central XML file, and components can easily be added, removed or replaced by custom implementations. The SUMO Toolbox is free for academic use and is available for download at <http://sumo.intec.ugent.be>. It can be installed on any platform supported by MATLAB. In addition, a link can be found to the available documentation and tutorials to install and configure the toolbox including some of its more advanced features.

5. Illustrations

To illustrate the flexibility of the surrogate modelling framework with sequential design, some example cases are considered. The SUMO Toolbox was used for each case.

5.1. Low-noise amplifier

This test case consists of a real world problem from electronics. A low-noise amplifier (LNA), a simple radio frequency circuit, is the typical first stage of a receiver, providing the gain to suppress noise of subsequent stages. The performance of an LNA can be determined by means of computer simulations where the underlying physical behaviour is taken into account. For this experiment, we chose to model the input noise-current, in function of two (normalized) parameters: the inductance and the MOSFET width. The response to the inputs for this test case is smooth with a steep ridge in the middle. This type of strong non-linear behaviour is difficult to approximate.

The model type for this problem is an ANN, trained with Levenberg-Marquard backpropagation with Bayesian regularization (300 epochs). The network topology and initial weights are optimized by a genetic algorithm, with a maximum of two layers. The process initiates by combining a 10-point LHD with a two-level factorial design (the corner points). As adaptive sampling methodology, the FLOLA-Voronoi algorithm was chosen to select a single-point iteration. Once the steep ridge has been discovered, the information density in this area will be increased. As model quality estimator, crossvalidation was used, with the root relative square error (RRSE) function:

$$\text{RRSE}(x, \tilde{x}) = \sqrt{\frac{\sum_{i=1}^n (x_i - \tilde{x}_i)^2}{\sum_{i=1}^n (x_i - \bar{x})^2}}. \quad (2)$$

The stopping criterion was set to an RRSE score below 0.05. This was achieved after evaluating a total of 51 samples. A plot of the model is shown in **Figure 4**. Additionally, the distribution of the samples in the two-dimensional input space is shown in **Figure 5**. The focus on the ridge can be clearly observed. In comparison, repeating the experiment in a one-shot setting with an LHD of 51 points and keeping all other settings results in an RRSE score of only 0.11. This is mostly caused by an inadequate detection of the non-linearity (only a few samples are near the non-linearity), whereas a lot of samples are on the smoothly varying surfaces.

5.2. Optimization: gas cyclone

The next illustration is more involved and is a joint modelling process aiming both at design optimality as well as feasibility. The goal is to optimize the seven-dimensional geometry of a gas cyclone. These components are widely used in air pollution control, gas-solid separation for aerosol sampling and industrial applications aiming to catch large particles such as vacuum cleaners. An illustration is given in **Figure 6**. In cyclone separators, a strongly swirling turbulent flow is used to separate phases with different densities. A tangential inlet generates a complex swirling motion of the gas stream, which forces particles toward the outer

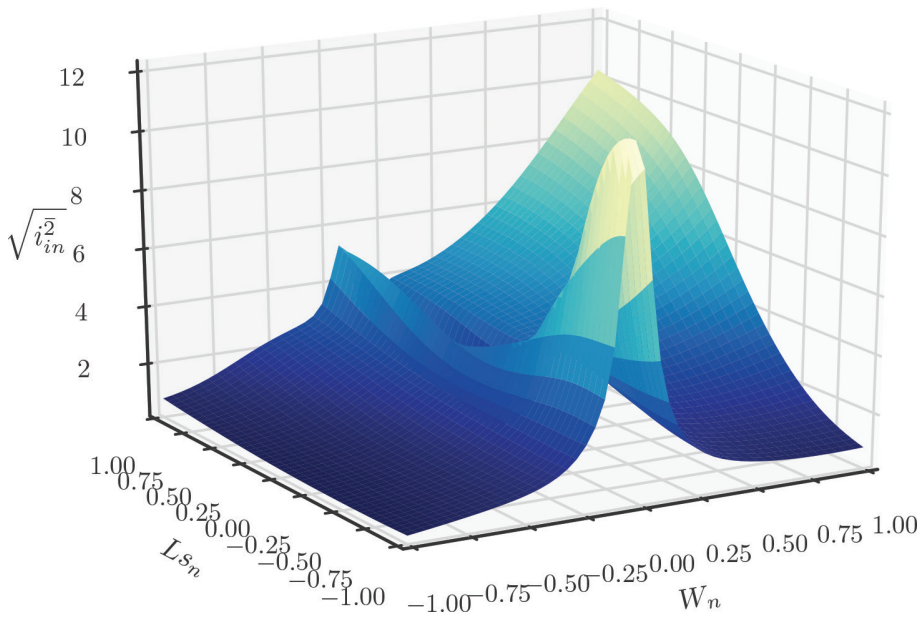


Figure 4. LNA: final surrogate model for the LNA illustration. The sharp peak is clearly present.

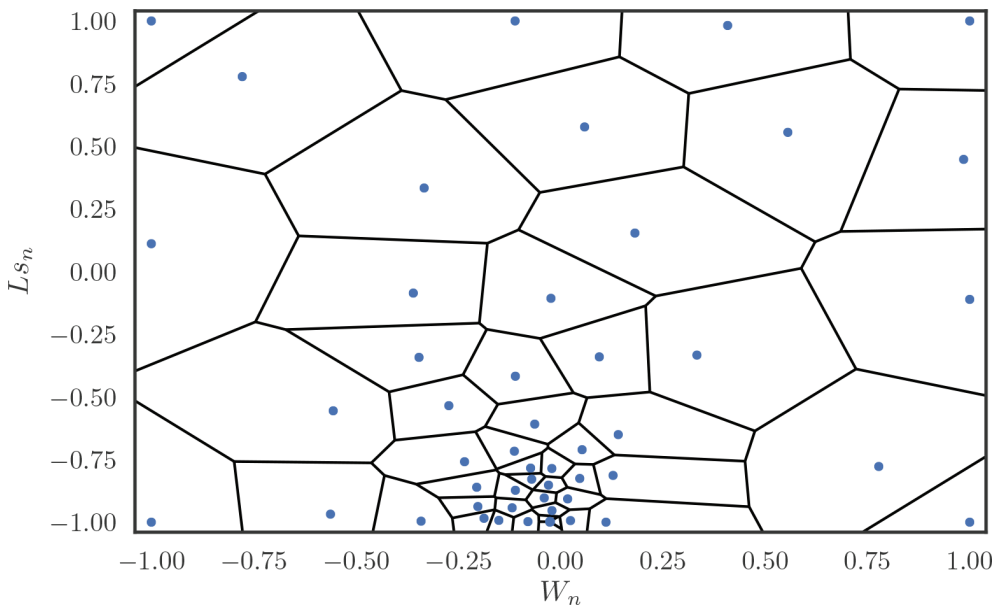


Figure 5. LNA: sample distribution as constructed sequentially by the FLOLA-Voronoi adaptive sampling algorithm.



Figure 6. Cyclone: illustration of a gas cyclone.

wall where they spiral in the downward direction. Eventually, the particles are collected in the dustbin (or flow out through a dipleg) located at the bottom of the conical section of the cyclone body. The cleaned gas leaves through the exit pipe at the top. The cyclone geometry [41] is described by seven geometrical parameters: the inlet height a , width b , the vortex finder diameter D_v and length S , cylinder height h , cyclone total height H_t and cone-tip diameter B_c . Modifying these parameters has an impact on the performance and behaviour of the gas cyclone itself.

Design optimality for the gas cyclone, however, is not represented by a unique and optimal number. In fact, it is represented by two different aspects: the pressure loss (expressed by the Euler number) and the cut-off diameter, which is expressed by the Stokes number. Both aspects represent a trade-off, and the proper scaling to sum both into a single objective is unknown. Hence, the correct way to proceed is to identify a set of Pareto optimal solutions representing the trade-off inherent to this problem, rather than a single solution. Presented with this trade-off, the designer has to make the final decision on what the optimal design should be. The shape of the Pareto front is informative and of great value for the designer (w.r.t. robustness of the solution for example). For the optimization, the two outputs of a simulation corresponding to these objectives are approximated with the built-in Kriging models [22].

In addition, geometry optimization usually involves constraints as some configurations are not feasible, or result in gas cyclones, which do not work according to specifications. In addition to the Euler and Stokes objectives, the simulation of a sample also emits four binary values indicating if their corresponding constraint was satisfied or not (denoted as c_1 , c_2 , c_3 and c_4). As each evaluation is computationally demanding, this additional knowledge should be included in order to maximize the probability of selecting feasible solutions. Each of the constraint outputs will therefore be approximated by a probabilistic SVM. For selecting new samples, both the Pareto optimality and the feasibility need to be considered. To this end, the HvPoI and PoF criteria are used, respectively. The PoF score is not computed explicitly (as it would be for a Gaussian Process) but interpreted as the SVM probability for the class representing feasible samples. This results in the following joint criterion:

$$\alpha(x) = \text{HvPoI}(x) \prod_{\{c_1, c_2, c_3, c_4\}} \text{PoF}_c(x) \quad (3)$$

For each iteration of the sequential design, this criterion is optimized resulting in a new sample maximizing the probability of a more feasible and more optimal solution. To start, an LHD is constructed in seven dimensions. The kernel bandwidth and regularization constant hyperparameters for the SVMs are optimized with the DIRECT optimization algorithm [42], using crossvalidation with the popular F_1 -score of the positive class as error function. The hyperparameters for the Kriging models are optimized with maximum-likelihood estimation. The sampling criterion is first optimized randomly with a dense set of random points, then the best solution serves as a starting point for applying gradient descent locally to refine the solution. The stopping criterion was set to a maximum of 120 evaluated data points.

Figure 7 shows the scores for all evaluated samples on both objectives. The bullets and squares represent the samples forming the Pareto front. The black-box constraints were learned as the optimization was proceeding, hence many evaluated samples do not satisfy the constraints (as this was unknown at that time): 8% of the evaluated samples however satisfy the constraints. Fortunately, four of them are Pareto optimal and represent valid optimal configurations. The exact optimal Pareto front was unknown upfront: in order to provide a comparison and verify the integrity of the identified solutions the traditional NSGA-II [43] multi-objective optimization algorithm was applied directly on the CFD simulations for a total of 10,000 evaluations. Clearly, the Pareto optimal solutions found by the surrogate-based approach form a similar

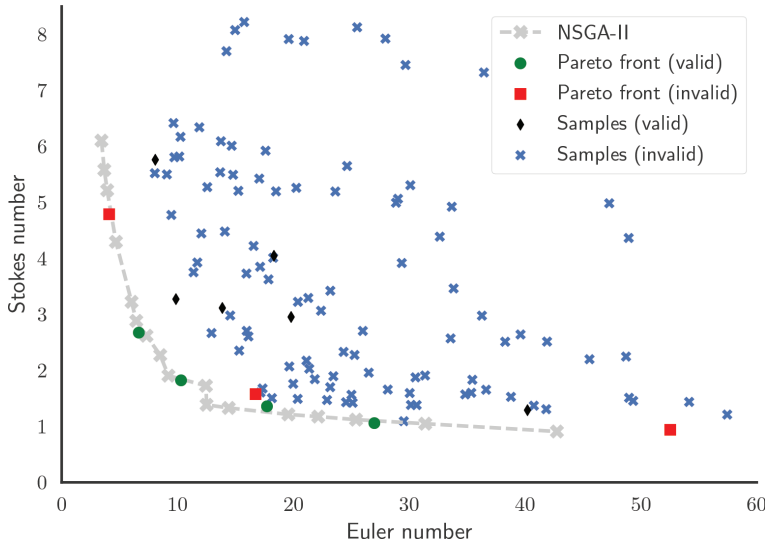


Figure 7. Cyclone: Pareto front obtained after 120 evaluations for the gas cyclone optimization. The plot also distinguishes between feasible and infeasible points and shows the Pareto front obtained by NSGA-II after an extensive 10,000 evaluations.

front. Our approach was able to identify these solutions with significantly fewer evaluations and hence significantly faster. Therefore, the identified Pareto front is a very good approximation given the budget constraint of 120 evaluations.

5.3. Satellite braking system

Finally, we demonstrate the use of surrogate models for performing analysis into the relevance of input parameters. A simulation of a braking system of the Aalto-1 student satellite [44, 45] was modelled with sequential design. The brake consists of a small mass m attached to a tether, which is extended at a constant speed v_{feed} . The satellite is spinning around with an angular velocity ω_{sat} , which is also the angular velocity of the mass at the beginning of the deployment. As the distance of the tip of the tether to the satellite increases, the angular velocity of the tip, ω_{tip} , decreases. This results in a displacement angle γ of the tether from its initial balance position. This causes a tangential force negative to the rotational direction of the satellite, causing it to spin around slower. The same tangential force accelerates the tip, which results in a decrease of the angle, until the tether has extended sufficiently again to further decrease the rotation of the satellite. **Figure 8** illustrates the setup graphically.

Although the displacement angle effectively causes the braking effect, it must remain within an acceptable range to prevent a range of undesired effects and issues. To this end, a simulation for γ was developed with five input parameters (the time after deployment time, initial angular velocity of the satellite, the mass of the tip, the deployment speed of the tether and

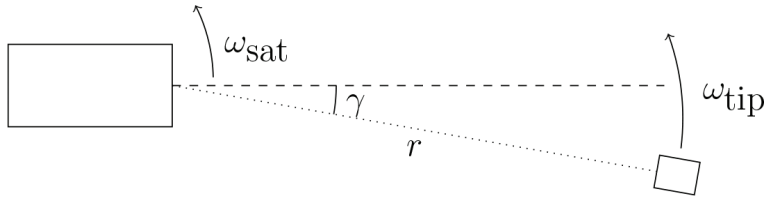


Figure 8. Satellite: illustration of the breaking system.

the width of the satellite). Using surrogate models, the aim is to find the parameters, which influence the displacement angle the most.

To approach this problem, the process starts from a LHD of 20 points. Iteratively, a space-filling sequential design selects 10 additional samples for evaluation on the simulator. The space-filling approach incorporates both the maximin distance and projective properties as described in Ref. [46]. The model type selected is a GP with the Matérn $3/2$ covariance function with Automated Relevance Determination (ARD). When 300 samples were evaluated, the process was terminated and the analytical approach presented in Ref. [6] was used to compute the first-order Sobol indices, as well as the total Sobol indices (first-order indices augmented with all indices of higher order interactions containing this parameter) [4]. Both indices are plotted in **Figure 9**.

From the results, it can be clearly observed that the mass of the tip has no impact at all on the displacement angle. It can therefore be disregarded from any further design decisions. All other parameters do have some impact, as expected. The deployment speed clearly has the

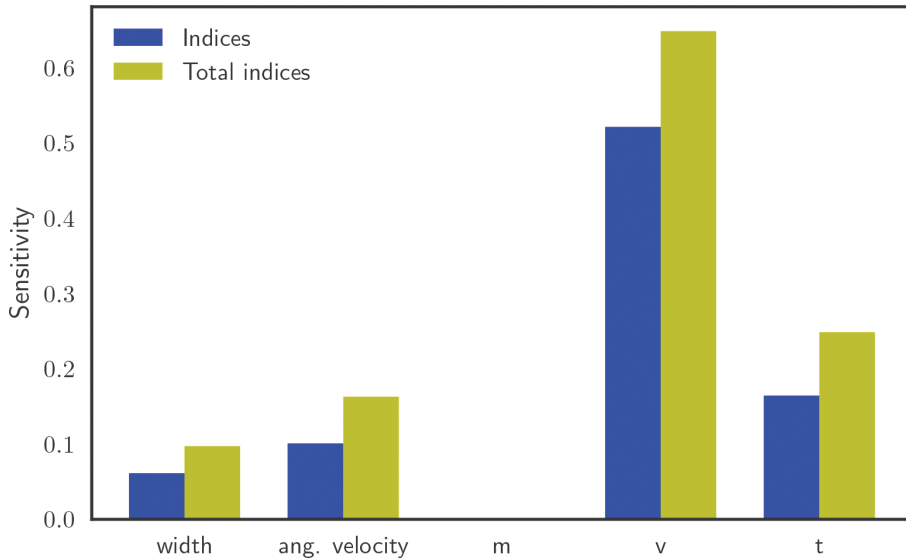


Figure 9. Satellite: Sobol indices for the final GP model.

highest impact. This is intuitive, as faster deployment results in more significant differences between the angular velocities of the satellite and the tip.

6. Conclusion

The benefits of surrogate modelling techniques have proven to be successful to work with expensive simulations and expensive objectives in general. Within this flexible methodology, and complemented with intelligent sequential sampling (sequential design) several tasks ranging from design space exploration, sensitivity analysis to (multi-objective) optimization with constraints can be accomplished efficiently with only a small number of evaluations. This greatly enhances the capabilities to virtually design complex systems, reducing the time and costs of product development cycles resulting in a shorter time-to-market. The strengths and possibilities were demonstrated on a few real world examples from different domains.

Acknowledgements

Ivo Couckuyt is a post-doctoral research fellow of FWO-Vlaanderen. The authors would like to thank NXP Semiconductors and Jeroen Croon in particular for providing the LNA simulator code.

Author details

Joachim van der Herten*, Tom Van Steenkiste, Ivo Couckuyt and Tom Dhaene

*Address all correspondence to: joachimvanderherten@intec.ugent.be

Department of Information Technology, IDLab, Ghent University - imec, Ghent, Belgium

References

- [1] Goethals K, Couckuyt I, Dhaene T, Janssens A. Sensitivity of night cooling performance to room/system design: surrogate models based on CFD. *Building and Environment*. 2012;**58**:23-36. DOI: 10.1016/j.buildenv.2012.06.015
- [2] Simpson TW, Booker AJ, Ghosh D, Giunta AA, Koch PN, Yang R-J. Approximation methods in multidisciplinary analysis and optimization: a panel discussion. *Structural and Multidisciplinary Optimization*. 2004;**27**(5):302-313. DOI: 10.1007/s00158-004-0389-9
- [3] Saltelli A. Sensitivity analysis for importance assessment. *Risk Analysis*. 2002;**22**(3):579-590. DOI: 10.1111/0272-4332.00040

- [4] Sobol IM. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and computers in simulation*. 2001;**55**(1-3):271-280. DOI: 10.1016/S0378-4754(00)00270-6
- [5] Keiichi I, Couckuyt I, Poles S, Dhaene T. Variance-based interaction index measuring heteroscedasticity. *Computer Physics Communications*. 2016;**203**:152-161. DOI: 10.1016/j.cpc.2016.02.032
- [6] Van Steenkiste T, van der Herten J, Couckuyt I, Dhaene T. Sensitivity Analysis of Expensive Black-box Systems Using Metamodeling. In: Roeder TMK, Frazier PI, Szechtman R, Zhou E, Huschka T, Chick SE, editors. *Proceedings of the 2016 Winter Simulation Conference*; 11-14 December 2016; Washington, D.C., USA; 2016. Institute of Electrical and Electronics Engineers, Inc. DOI: 10.1109/WSC.2016.7822123
- [7] Kennedy J, Eberhart R. Particle swarm optimization. In: *Proceedings of the IEEE International Conference on Neural Networks*; 27 November–1 December 1995; IEEE; Piscataway, New Jersey, USA. 1995. DOI: 10.1109/ICNN.1995.488968
- [8] Dellino G, Kleijnen JPC, Meloni C. Robust simulation-optimization using metamodels. In: Rossetti MD, Hill RR, Johansson B, Dunkin A, Ingalls RG, editors. *Proceedings of the Winter Simulation Conference*; 13-16 December 2016; Austin, TX, USA: IEEE; 2009. p. 540-550. DOI: 10.1109/WSC.2009.5429720
- [9] Couckuyt I, Aernouts J, Deschrijver D, De Turck F, Dhaene T. Identification of quasi-optimal regions in the design space using surrogate modeling. *Engineering with Computers*. 2013;**29**(2):127-138. DOI: 10.1007/s00366-011-0249-3
- [10] Vapnik V. *The Nature of Statistical Learning Theory*. 2nd ed. New York: Springer-Verlag; 2000. 334 p. DOI: 10.1007/978-1-4757-3264-1
- [11] Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. *Statistical Science*. 1989;**4**(4):409-423.
- [12] Kleijnen, JPC. *Design and Analysis of Simulation Experiments*. 2nd ed. Springer International Publishing; New York City, USA. 2015. 337 p. DOI: 10.1007/978-3-319-18087-8
- [13] Montgomery, DC. *Design and Analysis of Experiments*. 8th ed. John Wiley & Sons; Hoboken, New Jersey, USA. 2008. 752 p. DOI: 10.1002/qre.458
- [14] Van Dam ER, Husslage B, Den Hertog D, Melissen H. Maximin Latin hypercube designs in two dimensions. *Operations Research*. 2007;**55**(1):158-169. DOI: 10.1287/opre.1060.0317
- [15] Box GEP, Behnken D. Some new three level designs for the study of quantitative variables. *Technometrics*. 1960;**2**:455-475. DOI: 10.1080/00401706.1960.10489912
- [16] Raymond HM, Montgomery DC, Anderson-Cook CM. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. 4th ed. John Wiley & Sons; Hoboken, New Jersey, USA. 2016. 856 p.

- [17] Hendrickx W, Dhaene T. Sequential design and rational metamodelling. In: Kuh ME, Steiger NM, Armstrong FB, Joines JA, editors. Proceedings of the Winter Simulation Conference; 4 December 2005; Orlando, FL, USA: IEEE; 2005. pp. 290-298. DOI: 10.1109/WSC.2005.1574263
- [18] Jin R, Chen W, Sujiato A. An efficient algorithm for constructing optimal design of computer experiments. *Journal of Statistical Planning and Inference*. 2005;**134**(1):268-287. DOI: 10.1016/j.jspi.2004.02.014
- [19] Niederreiter, H. Random Number Generation and Quasi-Monte Carlo Methods. 1st ed. Society for Industrial and Applied Mathematics; Philadelphia, Pennsylvania, United States. 1992. 241 p. DOI: 10.1137/1.9781611970081
- [20] Viana FAC, Gerhard V, Vladimir B. An algorithm for fast optimal Latin hypercube design of experiments. *International Journal for Numerical Methods in Engineering*. 2010;**82**(2):135-156. DOI: 10.1002/nme.2750
- [21] Gorissen D. Grid-enabled adaptive surrogate modeling for computer aided engineering [dissertation]. Ghent, Belgium: Ghent University. Faculty of Engineering; 2010. 384 p. Available from: <https://biblio.ugent.be/publication/1163941/file/4335278.pdf>
- [22] Couckuyt I, Dhaene T, Demeester P. ooDACE Toolbox: a flexible object-oriented Kriging implementation. *Journal of Machine Learning Research*. 2014;**15**:3183-3186.
- [23] Kleijnen JPC, van Beers WCM. Application-driven sequential designs for simulation experiments: kriging metamodelling. *Journal of the Operational Research Society*. 2004;**55**(8):876-883. DOI: 10.1057/palgrave.jors.2601747
- [24] Sasena, MJ. Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations [dissertation]. 2002.
- [25] Farhang-Mehr A, Azarm S. Bayesian meta-modelling of engineering design simulations: a sequential approach with adaptation to irregularities in the response behaviour. *International Journal for Numerical Methods in Engineering*. 2005;**62**(15):2104-2126. DOI: 10.1002/nme.1261
- [26] Garnett R, Osborne M, Hennig P. Active learning of linear embeddings for Gaussian processes. In: Zhang ML, Tian J, editors. Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence; 23-27 July 2014; Quebec, Canada: AUAI Press; 2014. p. 230-239.
- [27] Crombecq K, Gorissen D, Deschrijver D, Dhaene T. A novel hybrid sequential design strategy for global surrogate modeling of computer experiments. *SIAM Journal on Scientific Computing*. 2011;**33**(4):1948-1974. DOI: 10.1137/090761811
- [28] van der Herten J, Couckuyt I, Deschrijver D, Dhaene T. A fuzzy hybrid sequential design strategy for global surrogate modeling of high-dimensional computer experiments. *SIAM Journal on Scientific Computing*. 2015;**37**(2):A1020-A1039. DOI: 10.1137/140962437
- [29] Regis, RG, Shoemaker CA. Constrained global optimization of expensive black box functions using radial basis functions. *Journal of Global Optimization*. 2005;**31**(1):153-171. DOI: 10.1007/s10898-004-0570-0

- [30] Moćkus, J. Marchuk, G.I. On Bayesian methods for seeking the extremum. In: Optimization Techniques IFIP Technical Conference; Springer; Novosibirsk, Berlin, Heidelberg. 1975. p. 400-404.
- [31] Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*. 1998;**13**(4):455-492. DOI: 10.1023/A:1008306431147
- [32] Scott W, Frazier PI, Powell W. The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression. *SIAM Journal on Optimization*. 2011;**21**(3):996-1026. DOI: 10.1137/100801275
- [33] Hernández-Lobato JM, Hoffman MW, Ghahramani Z. Predictive entropy search for efficient global optimization of black-box functions. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ, editors. *Advances in Neural Information Processing Systems 27*; 8-13 December 2014; Montreal, Canada: Curran Associates, Inc.; 2014. p. 918-926.
- [34] Gardner J, Kusner M, Weinberger KQ, Cunningham J, Xu, Z. Bayesian optimization with inequality constraints. In: Jebara T, Xing EP, editors. *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*; 21-26 June 2014; Beijing, China: JMLR.org; 2014. pp. 937-945.
- [35] Emmerich M, Beume N, Naujoks, B. Coello Coello C.A., Hernández Aguirre A., Zitzler E. An EMO algorithm using the hypervolume measure as selection criterion. In: *International Conference on Evolutionary Multi-Criterion Optimization*; 9-11 March 2005; Guanajuato, Mexico: Springer Berlin Heidelberg; 2005. p. 62-76.
- [36] Couckuyt I, Deschrijver D, Dhaene T. Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization. *Journal of Global Optimization*. 2013;**60**(3):575-594. DOI: 10.1007/s10898-013-0118-2
- [37] Emmerich M, Hingston P, Deutz AH, Klinkenberg JW. Hypervolume-based expected improvement: monotonicity properties and exact computation. In: *IEEE Congress on Evolutionary Computation (CEC)*; 5-8 June 2011; New Orleans, LA, USA: IEEE; 2011. p. 2147-2154. DOI: 10.1109/CEC.2011.5949880
- [38] Hernández-Lobato D, Hernández-Lobato JM, Shah A, Adams RP. Predictive Entropy Search for Multi-objective Bayesian Optimization. In: Balcan MF, Weinberger, KQ, editors. *Proceedings of the 33rd International Conference on Machine Learning (ICML-16)*; 19-24 June 2016; Manhattan, New York: JMLR.org; 2016. p. 1492-1501.
- [39] Gorissen D, Crombecq K, Couckuyt I, Demeester P, Dhaene T. A surrogate modeling and adaptive sampling toolbox for computer based design. *Journal of Machine Learning Research*. 2010;**11**:2051-2055.
- [40] van der Herten J, Couckuyt I, Deschrijver D, Dhaene T. Adaptive classification under computational budget constraints using sequential data gathering. *Advances in Engineering Software*. 2016;**99**:137-146. DOI: 10.1016/j.advangsoft.2016.05.016

- [41] Elsayed K. Optimization of the cyclone separator geometry for minimum pressure drop using Co-Kriging. *Powder Technology*. 2015;**269**:409-424. DOI: 10.1016/j.powtec.2015.09.003
- [42] Jones DR, Perttunen CD, Stuckman BE. Lipschitzian optimization without the Lipschitz constant. *Journal of Optimization Theory and Applications*. 1993;**79**(1):157-181.
- [43] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*. 2002;**6**(2):182-197. DOI: 10.1109/4235.996017
- [44] Kestilä A, Tikka T, Peitso P, Rantanen J, Näsälä A, Nordling K, Saari H, Vainio R, Janhunen P, Praks J, Hallikainen M. Aalto-1 nanosatellite—technical description and mission objectives. *Geoscientific Instrumentation, Methods and Data Systems*. 2013;**2**(1):121-130. DOI: 10.5194/gi-2-121-2013
- [45] Khurshid O, Tikka T, Praks J, Hallikainen M. Accommodating the plasma brake experiment on-board the Aalto-1 satellite. *Proceedings of the Estonian Academy of Sciences*. 2014;**63**(2S):258-266. DOI: 10.3176/proc.2014.2S.07
- [46] Crombecq K, Laermans E, Dhaene T. Efficient space-filling and non-collapsing sequential design strategies for simulation-based modeling. *European Journal of Operational Research*. 2011;**214**(3):683-696. DOI: 10.1016/j.ejor.2011.05.032