2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Model-Based Design of Experiments based on Local Model Networks for Nonlinear Processes with Low Noise Levels

Benjamin Hartmann, Tobias Ebert and Oliver Nelles

Automatic Control, Mechatronics
Department of Mechanical Engineering
University of Siegen
D-57068 Siegen, Germany
benjamin.hartmann@uni-siegen.de

Abstract— **Most common methods for experiment design are classical, geometric designs and optimal designs. Both categories of methods don't incorporate specific information about the process behavior into the design of experiments. In the case of optimal design often the underlying model structure is chosen as low order polynomial which is very restricted in its flexibility and causes problems, if used for higher-dimensional problems. Furthermore, the focus of these approaches lies on the minimization of the variance error. However, in many applications the process noise is negligible in comparison to the highly nonlinear behavior which usually causes a large bias error. Therefore, this paper presents the new algorithm HilomotDoE which is an active learning algorithm that aims to minimize the bias error of the model. This is achieved by an iterative refinement of a local model network and simultaneously the addition of a certain amount of measurement points. Demonstration examples and theoretical comparisons with the common D-optimal design show the usefulness of HilomotDoE for the mentioned problem class.**

I. Introduction

Generally, process data modeling is only a single step in many necessary process chains, see e.g. Fig. 1. But the quality of the entire process chain crucially depends on well distributed measurements. The practitioner seeks to keep the effort for data acquisition as low as possible, because measurements are mostly both, time-consuming and expensive. Furthermore, some kind of measurements can not be repeated with the same environmental configurations.

Design of experiments is a well known approach to efficiently collect data such that existing resources can be used in the most effective way. The application of the major theory in this field described, e.g. in [5], [6] or [15], has the power to significantly reduce measurement costs and to improve generalization properties of data-based modeling strategies.

In many disciplines one can observe that the measurement process is totally decoupled from the subsequent modeling step. Modeling approaches are mostly treated as passive recipients of data and the benefit that could be achieved from interaction between model estimation and the ability to influence the measurement process is totally neglected. As proposed in [3], [1] or [23] *active learning* strategies lead to improved generalization at the cost of higher computational effort.

This work was supported by the German Research Foundation (Deutsche Forschungsgemeinschaft (DFG), project code NE 656/3-2).

This contribution focuses on processes where the most modeling benefit is achieved with minimization of rather the bias than the variance error, because usually low output noise levels are expected. The author has experience with a real world application that can be used as example at this point: Experiment design in the framework of engine development.

In order to optimize the actuating variables of a combustion engine with respect to its fuel consumption, exhaust emissions and other criteria the engineer must have a well generalizing high performance model. Figure 1 shows the general procedure steps in combustion engine development. For all subsequent steps the role of measurements is essential. In research for engine development this topic rather has gained inferior attention. Most of the scientific literature in this field is restricted to methods based on precise mathematical derivations, see e.g. [18], [26], [25].

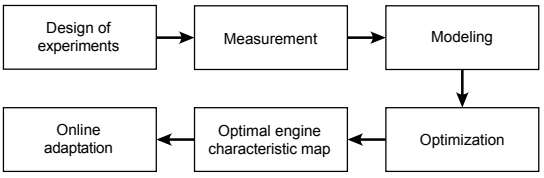

Fig. 1. Process chain of typical application: Generation of adaptive engine characteristic maps.

Mostly, the measurements on a engine test bench are gathered stationary, i.e. at each operating point of the engine the application engineer has to wait until the process is at steady state. The stationary point is then result of averaging a certain amount of data points, e.g. the mean of 50 samples. Therefore, the measured data can be expected to be almost noise-free, though it's a non-ergodic process. For this kind of application the challenging task is more concentrated on minimizing the bias error than to minimize the parameter variances of the model as it is the goal for an D-optimal experiment design. This contribution proposes a new strategy to cope with such requirements.

The following Section II gives an overview over common methods for experiment design. Section III compares the

attributes of passive and active learning methods. The main part of the paper is described in Section IV, namely the introduction of the HilomotDoE algorithm for active experiment design. Finally, in Section V some demonstration examples are shown and the paper ends up with the conclusions in Section VI.

II. Methods for Experimental Design

As a branch of statistics, methods for design of experiments (DoE) are tools to generate measurement samples with as low experimental effort as possible or even make the reasonable experiment realization possible at all in the case of high-dimensional processes, respectively. The goal of experiment design is to systematically investigate the relationships between influencing variables and interesting dependent variable [14].

In principle, DoE-methods can be distinguished into two categories [13], [16]:

1) *Classical, geometric experiment designs*: The samples distribution in the input space is arranged according to a predefined pattern without any consideration of the output variable. Commonly used experiment designs are known as: *Full* and *Fractional Factorial*, *Central Composite* and *Box-Behnken*, *Latin Hypercube*, et cetera [22].
2) *Optimal experiment designs*: Goal of optimal strategies is the minimization of the model output variance, the variance of model parameters or similar quantities. For this purpose a certain model structure must be assumed which describes the input-output-mapping correctly. Common approaches are known as *D*-, *A*- and *V-optimality* [21]. Typically assumed model structures are low order polynomials.

Furthermore, sometimes application-based strategies can be found in literature where prior knowledge about the specific problem is exploited, e.g. with application of a fuzzy-rule-based approach like proposed in [9].

Generally, the most applicated method is the D-optimal experiment design. Numerous publications favor the D-optimal design, e.g. [7], [17]. At first, this method generates a regression matrix $\underline{X}$ which is of polynomial structure. Then, the determinant of the covariance matrix, which is a measure for the model variance, is minimized:

$$\det\left((\underline{X}^{\mathrm{T}}\underline{X})^{-1}\right) \rightarrow \min\,, \tag{1}$$

or its inverse is maximized, respectively ($\det\left(\underline{X}^{\mathrm{T}}\underline{X}\right) \rightarrow \max$). Usually, approaches that minimize the model variance are built on the basis of polynomial functions of second or third order, although in [1] the extension to multilayer-perceptron networks is already introduced. However, the calculations for linearly parametrized models like polynomials are considerably less complex.

The mentioned experiment design methods are based on *polynomial* models. That is, all drawbacks with respect to regression with polynomials, like e.g. bad interpolation and extrapolation behavior and increased risk of overfitting [19], are reflected in the experiment design. The popular D-optimal design solely minimizes the variance error. In literature many application attempts can be found, see e.g. [8], [4].

The DoE-methods of both categories don't incorporate the process output characteristics. However, especially for complex, nonlinear processes the crucial point is to minimize the *bias error*. These nonlinear relationships often can only insufficiently be covered with polynomial structures. Hence, D-optimal designs are only optimal with respect to a certain model assumption. If these model assumptions do not match the underlying process that has to be investigated, the probability of neglecting important nonlinear relationships of the process is high.

The approaches of the first category are universally applicable. The distribution of data points in the input space is independent of the problem and obstinately follows a certain scheme. They can't be very effective, because they do not incorporate any structural information about the estimated relationship between input and output variables. These disadvantages should be avoided with methods from the second category.

DoE-methods of the second category try to find a suitable choice of measurement points in order to minimize the variance error. The bias error is not taken into account. Therefore, these approaches are only suggestive, if the variance amount of the overall model error is high in comparison to the amount of the bias error.

An experiment design that minimizes the variance error is called *V-optimal* [21]. However, much more popular and computational less demanding is the minimization of the volume of the confidence ellipsoid. This is achieved with the minimization of the determinant of the parameter covariance matrix. Therefore, this method is called *D-optimal* [21]. Sometimes an *A-optimal* design is used which aims to reduce the asphericity of the confidence ellipsoid and therefore minimizes the average variances of the parameters [21]. From a practical point of view all these approaches aim to achieve the reduction of the variance error while they simultaneously neglect or ignore the bias error, respectively.

Only a few publications discuss the reduction of the bias error, although in [2] already numerous arguments can be found that analyze the importance of the problem. [2] discusses the problem that a D-optimal design can be very disadvantageous, if the requirements for this approach are not satisfied, namely the dominance of the variance error. In many applications this criterion is not met.

III. Passive versus Active Learning

For time and financial reasons in all application fields only a restricted contingent for measurements is available. The efficiency of exploitation of given measurement resources crucially influences all subsequent process steps. An iterative synchronization between experiment and modeling during the measurement allows the tuning of new measurements according to the underlying process behavior.

The basic idea of *active learning* is to create an interaction between the modeling and the measurement procedure. This strategy focuses on finding a tradeoff between improved

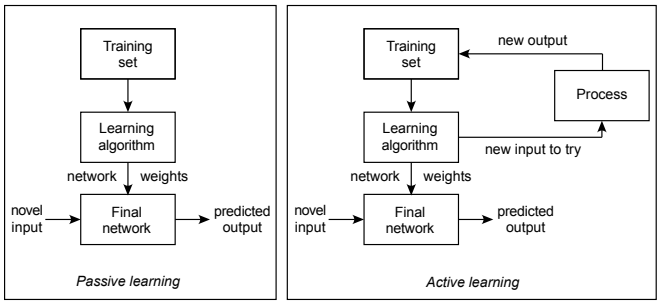

Fig. 2. Comparison of passive and active learning. In contrast to passive learning the active learning approach iteratively trains the data and sets new measurement points based on the previous training set which can improve the model generalization significantly. However, this iterative procedure is computational more expensive and therefore time-consuming [1].

generalization of the model, which is typically the case with active learning, and the amount of computational cost, which usually increases with this approach, too. The main goal is to build a good model with the fewest data [1].

As pointed out with Fig. 2 the difference between active and *passive* learning is that in case of active learning the learning algorithm interacts with the process such that the training set is iteratively updated. In case of passive learning the learning algorithm gets the complete training set with measurements that are taken beforehand in one step.

IV. Model-based Design of Experiments with the HilomotDoE algorithm

HilomotDoE stands for *HIerarchical LOcal MOdel Tree for Design of Experiments*. It is an active, model-based and incremental design of experiments method that generates a sample distribution in order to minimize the bias error of the model. The modeling is done with a local model network. The output $\hat{y}$ of a local model network (LMN) with p inputs $\underline{u} = [u_1\ u_2\ \cdots\ u_p]^T$ can be calculated as the interpolation of M local model outputs $\hat{y}_i(\cdot)$, $i = 1, \ldots, M$ [19],

$$\hat{y} = \sum_{i=1}^{M} \hat{y}_i(\underline{u})\Phi_i(\underline{u})\,, \tag{2}$$

where the $\Phi_i(\cdot)$ are called validity functions. These validity functions describe the regions where the local models are valid; they describe the contribution of each local model to the output. From the fuzzy logic point of view (2) realizes a set of M fuzzy rules where the $\Phi_i(\cdot)$ represent the rule premises and the $\hat{y}_i(\cdot)$ are the associated rule consequents. Because a smooth transition (no switching) between the local models is desired here, the validity functions are smooth functions between 0 and 1. For a reasonable interpretation of local model networks it is furthermore necessary that the validity functions form a *partition of unity*:

$$\sum_{i=1}^{M} \Phi_i(\underline{u}) = 1\,. \tag{3}$$

Thus, everywhere in the input space the contributions of all local models sum up to 100%. In principle, the local models can be chosen of arbitrary type. This paper deals only with local models of linear type:

$$\hat{y}_i(\underline{u}) = w_{i,0} + w_{i,1}u_1 + w_{i,2}u_2 + \ldots + w_{i,p}u_p\,. \tag{4}$$

However, an extension to higher degree polynomials or other linearly parameterized model classes is straightforward.

A. Functioning of the HilomotDoE algorithm

The HilomotDoE algorithm generates the partitioning of a local model network and does the DoE sampling simultaneously. The partitioning is based on the axes-oblique partitioning algorithm that is introduced in [20] and which was further investigated in e.g. [12], [11]. The DoE sampling is done with a pseudo-random Monte Carlo approach which is explained in this section, too.

The basic rule of HilomotDoE is that a fixed amount of data points is assigned to each local model. In each iteration of the partitioning algorithm the global model is refined by adding a new local model. Therefore, in each iteration of HilomotDoE the same amount of new sample points is added to the training set.

In order to make an active, complexity dependent DoE sampling possible, the algorithm works as follows: First of all, a first model with a low amount of data has to be estimated. These samples should be distributed homogeneously such that they cover most of the design space, if no prior knowledge about the process is available. Then, this first model is split into two submodels. Now, new measurement points are set such that each local model region possesses the same amount of data points. In the next step, the updated training set allows the estimation of a second, further refined model. The procedure is iterated as long until a certain stop criterion is met. For example, this could be the maximum number of measurement points.

The main attributes of the HilomotDoE algorithm can be summarized as follows:

- To each local model the same amount of data samples is assigned. The bound of a local model i is defined with the validity function $\underline{\Phi}_i(\underline{u})$.
- Iteratively, the global model is refined by adding a new local model. This ensures that in each iteration the same number of new data points is generated.
- For the placing of new measurement points the data points that already exist are considered such that a homogenous sampling is achieved.
- The global model is updated during each measurement cycle. The model structure adapts to the real process behavior.
- To those regions where the process shows strong nonlinearities measurements are assigned with high density. If the process is almost linear in a certain region, it is sufficient to cover this region with only few measurements.
- The model is refined as long as a stop criterion is met, such as the maximum number of measurement points.
- Design of experiments, measurement and modeling are concurrent operations.

- It is easily possible to incorporate design space information that is known a priori.

A crucial part of the HilomotDoE algorithm is the pseudo-Monte Carlo sampling algorithm which is described next.

B. Pseudo-Monte Carlo Sampling Algorithm

The DoE sampling algorithm used in HilomotDoE is inspired by the generation of pseudo-random numbers. In literature many approaches can be found that aim to generate homogeneously distributed data samples. For example, Halton sequences [10] or Sobol sequences [24] can be mentioned that are commonly used for high-dimensional integrations. The sampling in HilomotDoE has a similar goal, namely the generation of new points which are far away from existing points. For this purpose, an algorithm is used in HilomotDoE which selects from a set of random candidate points the sample that has maximum distance to all existing points. The sampling functions as follows:

1) Generate a candidate set $\mathcal{Z} \in \mathbb{R}^p$ of equally distributed random samples $\{\underline{z}_j\}_{j=1}^{N_c}$, where p is the number of input or design variables.
2) Calculate distance matrix $\underline{D} = \{d_{ij}\} \in \mathbb{R}^{N \times N_C}$ from all candidate points $\{\underline{z}_j\}_{j=1}^{N_c}$ to all esxisting points $\{\underline{u}_i\}_{i=1}^{N}$, see Fig. 3. The distance d_{ij}^2 is defined as the Mahalonobis norm:

$$d_{ij}^2 = ||\underline{u}_i - \underline{z}_j||_{\underline{\Sigma}}^2 = (\underline{u}_i - \underline{z}_j)^{\mathrm{T}} \underline{\Sigma} (\underline{u}_i - \underline{z}_j)\,, \qquad (5)$$

   where $\underline{\Sigma}$ is the covariance matrix. If the design space has equally scaled axes, the covariance matrix $\underline{\Sigma}$ can be chosen as identity matrix ($\underline{\Sigma} = \underline{I}$). Then the Mahalonobis norm equals the Euclidean norm.
3) The i-th row of the distance matrix $\underline{D}$ contains the distances from all candidate points $\{\underline{z}_j\}_{j=1}^{N_c}$ to the i-th existing point $\underline{u}_i$. To each existing point $\underline{u}_i$ the nearest neighbor candidate point $\underline{z}_{i,\mathrm{NN}}$ is selected, that is, the candidate point with minimum distance d_{ij} to the existing point $\underline{u}_i$. This results in a distance vector $\underline{d}_{\mathrm{NN}} \in \mathbb{R}^{N \times 1}$ containing all nearest neighbor distances.
4) Finally, the candidate point is selected which causes the maximum entry value in $\underline{d}_{\mathrm{NN}}$, i.e. the candidate point with maximum nearest neighbor distance.

Compared to a equally distributed sampling the pseudo-Monte Carlo sampling covers the design space much more homogenously as can be seen in Fig. 4. Furthermore, the Monte Carlo approach takes already existing measurements into account for setting the new data points which is not the case for equally distributed sampling.

C. Influence of amount of samples and process noise

One challenging point of the proposed active learning approach is to set the number of data points per local model. If there is a restricted amount of data that can be measured, a compromise between two extreme cases has to be found:

On the one hand, to each local model can be assigned exactly as many points as there are local model parameters

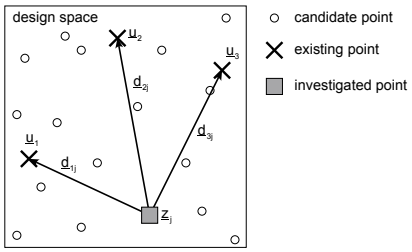

Fig. 3. Distance calculation in design space. The nearest (already existing) neighbor point of each candidate point is evaluated and the candidate with maximum nearest neighbor distance is chosen for next measurement. This ensures a wide spread data distribution.

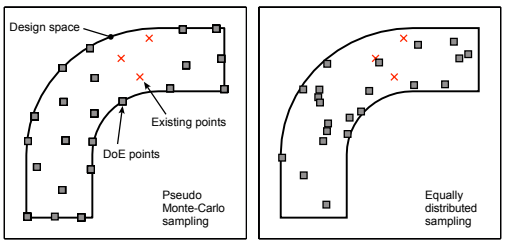

Fig. 4. Comparison of DoE samples that are generated with the proposed pseudo Monte-Carlo sampling (left) with equally distributed sampling in a given design space (right). Furthermore, the Monte Carlo approach takes already existing measurements into account.

that have to be estimated. In the case of local *linear* models this would mean $p+1$ parameters, where p is the number of inputs. This procedural method leads to a very flexible partitioning because of the high number of local models that has to be generated by HilomotDoE in order to set all measurement points. Unfortunately, with this procedure the nonlinear split optimization with HilomotDoE will become not very efficient because the optimization is done with a sparse amount of data. Therefore, many iterations are necessary in order to cover the nonlinearities of the process.

The other extreme would be the generation of one global model with all data points. This procedure leads to a space filling experiment design due to the pseudo-Monte Carlo sampling, but there is no interaction between measurement and modeling. Hence, there is no chance to reproduce the process nonlinearity or to minimize the bias error, respectively.

In order to show the dependency of the number of local model points on the resulting global model error an academic example is presented in Figs. 5 and 6. The process follows the equation: $y = \frac{0.1}{0.1+0.5(1-u_1)+0.5(1-u_2)}$. It is modeled with three data point strategies, i.e. 3, 6 and 9 points per local model (LM). Stop criterion for HilomotDoE was a total of 90 samples. The simple example shows, that the strategy with 6 points per local model leads to the best validation error, see Fig. 5. Furthermore, it becomes clear that in the presence of noise the choice of the strategy with the most data points per local model is the best.

Figure 6 exemplarily compares the resulting model and the process for the 6-point strategy. The partitioning and the generated data points show clearly that with respect to the complexity adaptive DoE algorithm HilomotDoE the density

of measurements is high in regions where the process shows severe nonlinear behavior and low in regions of almost linear characteristic.

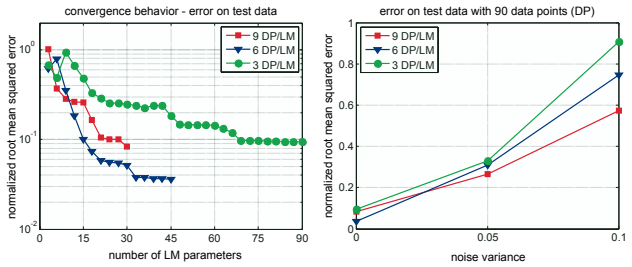

Fig. 5. Left: Convergence behavior of HilomotDoE algorithm for different number of data points per local model (LM). Right: Same experiment for three different output noise levels.

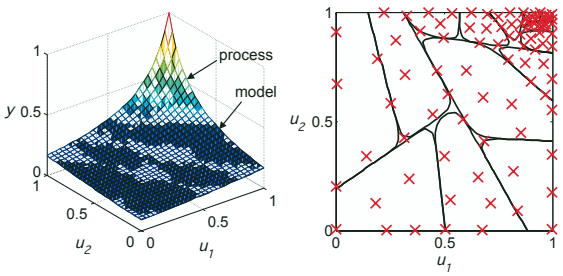

Fig. 6. Demonstration example with 2 inputs and 1 output. Left: Comparison between process and model. Right: Partitioning with 6 data samples per local model, overall 90 samples.

D. Comparison of D-optimal designs with HilomotDoE designs

Most of the publications in the field of automotive engine development deal with the application of D-optimal designs. From a scientific point of view a comparison with HilomotDoE seems to be somehow unfair, but from a practical point of view this represents the current state-of-the-art approach. Therefore, on the basis of some selective criterions D-optimal design is compared to the design via HilomotDoE as follows:

- *Goal:* In case of D-optimal design parameter variances have to be minimized, whereas the goal of the HilomotDoE algorithm is the minimization of the global model error.
- *Model structure:* User defines the underlying model structure for D-optimal designs, usually polynomials of second order. With HilomotDoE the structure is refined iteratively and follows a divide & conquer principle.
- *Process supervision:* D-optimal designs only incorporate the input variables, independently of the environment. HilomotDoE aims to minimize the bias error. Therefore, it sets the data points with respect to the underlying process.
- *Reproducibility:* D-optimal designs are reproducible, if the same regressors are used, although it is barely deterministic. The reproducibility of HilomotDoE designs is restricted. This can be circumvented, if the samples inside of the local models are distributed with a fixed geometrical design, e.g. with a latin hypercube design.
- *Underlying model:* Typically, D-optimal designs are built up with a polynomial model of low degree, e.g. second order. In the case of HilomotDoE the experiment design is taken on the basis of a local model network with an axes-oblique partitioning strategy.
- *Parsimony of sample placings:* Only a very small amount of different data samples is placed with D-optimal design. But this method tends to generate data clusters where a certain quantity of data points is placed at the same design coordinate. With HilomotDoE the amount or density of required samples, respectively, depends on the underlying process nonlinearity.
- *Suitability for high-dimensional problems:* For high-dimensional input spaces D-optimal designs are rather disadvantageous, because a huge number of polynomial regressor terms has to be considered. Due to the extremely flexible model structure HilomotDoE is especially well suited for high-dimensional problems.
- *Suitability for model-based optimization:* Could be problematic with D-optimal design, because it doesn't consider the bias error or the true process behavior, respectively. HilomotDoE minimizes the error between model and process. Nonlinearities are modeled with high accuracy. This leads to good preconditions for optimization.
- *Computation effort:* D-optimal design can be done pretty fast, but it has high effort with the candidate selection procedure. HilomotDoE designs are computationally more expensive because of the required nonlinear optimization of the local model network structure and the pseudo-Monte Carlo sampling. But in the case of measurements on an engine test bench it can be assured that HilomotDoE is fast enough to calculate a new input to try while the measurement of the old point is still running.

V. Demonstration Examples

The potential of experiment designs with HilomotDoE compared with standard D-optimal designs is shown with some examples presented in the Figs. 7 and 8. The pictures illustrate both, D-optimal experiment designs and HilomotDoE designs. The test processes are compared with the resulting model. Again, it should be mentioned that the HilomotDoE method is developed for processes where the variance error can be neglected in comparison to the bias error.

For example, on an engine test bench each measurement is taken in steady-state. To ensure stationarity the measurements have to be taken for a time period of few minutes and the resulting data point for modeling is generated by taking the mean value of steady-state measurements. Therefore, the engine process has a very low noise level. This is the reason why the investigations in this paper are taken on processes without noise.

For both, the D-optimal designs and the HilomotDoE designs the same model structure was used for experiment design as for modeling. In the case of D-optimal design the regressors were chosen as full polynomial of third degree

and MATLAB's row exchange algorithm [22] was used for candidate selection. The same polynomial regressors were used for modeling as for generation of the experiment design. In the case of the HilomotDoE design the underlying local model network is compared to the process. In all examples the same amount of data points was produced.

It becomes obvious that the model with D-optimal design is not able to achieve the main process nonlinearities. Thus, if the model assumption of a polynomial of third degree doesn't match to the real process the experiment design can be very inefficient. This shows the big advantage of the structure adaptive, active learning procedure with HilomotDoE.

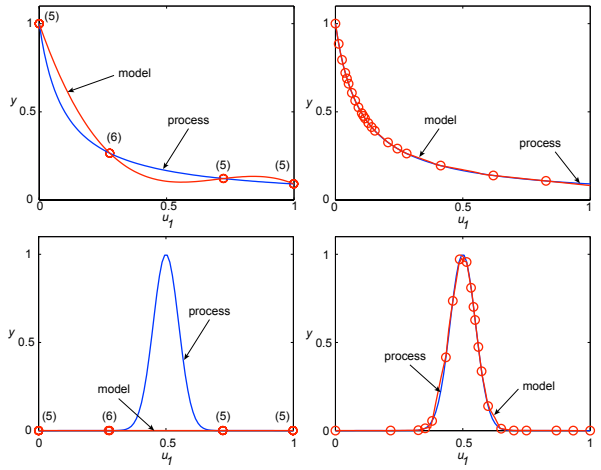

Fig. 7. Comparison of D-optimal design (left) and HilomotDoE design (right) (21 samples with each design).

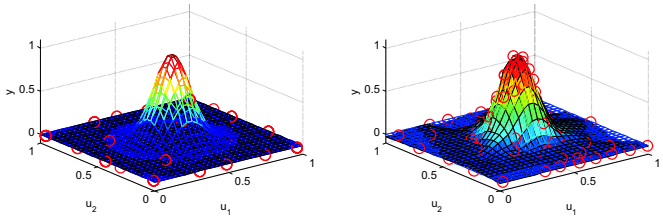

Fig. 8. Comparison of D-optimal design (left) and HilomotDoE design (right) (90 samples with each design). The process is drawn light and the model is drawn solid.

VI. Conclusions

This paper presents the new active learning algorithm HilomotDoE. It is especially developed for the problem class of processes with highly nonlinear behavior and low noise levels such as known, e.g., in the field of combustion engine calibration. HilomotDoE is based on a local model network structure. In each iteration of the algorithm the model is refined by adding a new local model. To each local model the same amount of data points is assigned. The size of the local model validity regions depends on the nonlinearity of the underlying process. Therefore, in regions where the process shows highly nonlinear behavior the density of measurements is much higher than in regions where the process is almost linear. This ensures a very effective exploitation of given measurement resources and a nonlinear process model that is of high flexibility. Experiment design, measurement and modeling are concurrent operations.

References

[1] D.A. Cohn. Neural network exploration using optimal experiment design. *Advances in neural information processing systems*, 6:679–686, 1994.

[2] D.A. Cohn. Minimizing statistical bias with queries. *Advances in neural information processing systems*, 9:417–423, 1997.

[3] D.A. Cohn, Z. Ghahramani, and M.I. Jordan. Active learning with statistical models. *Arxiv preprint cs/9603104*, 1996.

[4] K. Röpke (Ed.). *Design of Experiments (DoE) in Engine Development II*. Expert-Verlag, Essen, 2005.

[5] V.V. Fedorov. Theory of optimal experiments. *New York*, 1972.

[6] V.V. Fedorov and P. Hackl. *Model-oriented design of experiments*. Springer Verlag, 1997.

[7] M. Fischer and K. Röpke. Effiziente Applikation von Motorsteuerungsfunktionen für Ottomotoren. *MTZ*, 61(9):562–570, 2000.

[8] T. Fortuna, M. Mayer, H. Pflügl, and K. Gschweitl. Optimierung von Motorsteuerungen am Motorenprüfstand. In R. Isermann, editor, *Modellgestützte Steuerung, Regelung und Diagnose von Verbrennungsmotoren*, chapter D.16, pages 251–274. Springer, 2003.

[9] T. Fukuama, Y. Nozaki, and K. Tanaka. A rule-based calibraton method for diesel engines. In K. Röpke, editor, *Design of Experiments (DoE) in Engine Development II*, chapter 3.1, pages 124–132. Expert-Verlag, Essen, 2005.

[10] J.H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2(1):84–90, 1960.

[11] B. Hartmann and O. Nelles. Automatic adjustment of the transition between local models in a hierarchical structure identification algorithm. In *European Control Conference (ECC)*, Budapest, Hungary, August 2009.

[12] B. Hartmann and O. Nelles. On the smoothness in local model networks. In *American Control Conference (ACC)*, St. Louis, USA, June 2009.

[13] R. Isermann (Hrsg.). *Modellgestützte Steuerung, Regelung und Diagnose von Verbrennungsmotoren*. Springer Verlag, Berlin, Heidelberg, New York, 2003.

[14] W. Kleppmann. *Taschenbuch Versuchsplanung – Produkte und Prozesse optimieren*. Carl Hanser Verlag, Munich, Vienna, 2003.

[15] D.J.C. MacKay. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992.

[16] A. Mitterer. *Optimierung vielparametriger Systeme in der KFZ-Antriebsstrangentwicklung*. VDI Verlag, Dissertation, Germany, 2000.

[17] A. Mitterer and F. Zuber-Goos. Modellgestützte kennfeldoptimierung – ein ansatz zur steigerung der effizienz in der steuergeräteapplikation. *Automobiltechnische Zeitschrift ATZ*, pages 188–196, 2000.

[18] D.C. Montgomery. *Design and Analysis of Experiments*. Wiley, Hoboken, 2005.

[19] O. Nelles. *Nonlinear System Identification*. Springer, Berlin, Germany, 2001.

[20] O. Nelles. Axes-oblique partitioning strategies for local model networks. In *International Symposium on Intelligent Control (ISIC)*, Munich, Germany, October 2006.

[21] N.N. *MATLAB Model-Based Calibration Toolbox: User's Guide, Version 3.4*. The MATHWORKS Inc., Natick, MA, 2008.

[22] N.N. *MATLAB Statistics Toolbox: User's Guide, Version 6*. The MATHWORKS Inc., Natick, MA, 2008.

[23] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 839–846. Citeseer, 2000.

[24] IM Sobol. On quasi-Monte Carlo integrations. *Mathematics and Computers in Simulation*, 47(2-5):103–112, 1998.

[25] S. Töpfer. Statistische Versuchsplanung. In R. Isermann, editor, *Modellgesttzte Steuerung, Regelung und Diagnose von Verbrennungsmotoren*, chapter B.8, pages 121–130. Springer, 2003.

[26] M. Weber, H. Kötter, A. Schreiber, and R. Isermann. Model-based design of experiments for static and dynamic measurements of combustion engines. In K. Röpke, editor, *Design of Experiments (DoE) in Engine Development II*, chapter 3.4, pages 167–183. Expert-Verlag, Essen, 2005.