

GOAL-ORIENTED ACTIVE LEARNING WITH LOCAL MODEL NETWORKS

JULIAN BELZ¹, KONRAD BAMBERGER², OLIVER NELLES¹ & THOMAS CAROLUS²

¹Institute of Mechanics and Control Engineering, University of Siegen, Germany.

²Institute of Fluid- and Thermodynamics, University of Siegen, Germany.

ABSTRACT

A methodology for goal-oriented active learning with local model networks (LMNs) is proposed. It is applied for the generation of training data for a computational fluid dynamics (CFD) metamodel. The used metamodel is an LMN trained with data originating from CFD simulations. This metamodel describes the total-to-static efficiency for a given design point, defined by the pressure rise at a specific volume flow rate, depending on geometrical parameters of an impeller of centrifugal fans. The goal-oriented nature originates from three main targets that are addressed simultaneously during the active learning procedure. (I) The concentration on possibly optimal geometries and (II) the focus on areas in the input space where the metamodel's performance is considered to be worst. Additionally, (III) new measurements should differ from already simulated geometries as much as possible. With these goals three important issues in modeling are addressed simultaneously: (I) optimality, (II) model bias, (III) model variance/uniformly space-filling property. In order to fulfill all goals, special properties of LMNs are utilized (embedded approach). Through the structure of LMNs, it is possible to assign *local* model errors to specific areas in the input space. New measurements are preferably placed in such high-error regions, while concentrating on presumably optimal geometries that differ most from the ones already available in the training data. In the field of fluid machinery, the range of achievable design points is usually identified by the Cordier diagram. While the design points obtained in the passive learning phase fairly agree with the standard Cordier diagram, an extension of achievable design points was observed due to the proposed goal-oriented learning strategy. In addition, the total-to-static efficiency could be improved in some areas of the Cordier diagram.

Keywords: active learning, aerodynamic optimization, design of experiments, experimental modeling, impeller of centrifugal fans, metamodeling.

1 INTRODUCTION

Measurements play the key role in experimental modeling. The quality of the data used to build models restricts the performance that can be achieved. Since measurements are always time-consuming and might be very expensive, the aim is often to find the necessary minimum of measurements in order to fulfill some task sufficiently well. Therefore, the design of experiments (DoE) should contain the most informative measurements. In principle there are two ways to obtain such a DoE, which are faced in Fig. 1. If a passive learning strategy is pursued, all points contained in the DoE are known before the first measurement is carried out. This experimental design might be optimized according to some optimality criterion, see Refs. [1–3] for common criteria. In case of an active learning strategy the experimental design is not known completely a priori. Information gathered through already obtained measurements are used to select new *queries*. A query specifies the point in the input space at which a label or the output value should be measured. In the context of metamodeling tasks, a more common expression for active learning is sequential sampling, see Refs. [4, 5].

Active learning strategies have the potential to achieve the same model quality with significant less data compared to passive learning strategies, as stated e.g. in Refs. [6–9]. Active learners can be distinguished by their query strategies, i.e. how they choose the next query. A good survey on possible query strategies for active learners is given in [10]. The proposed

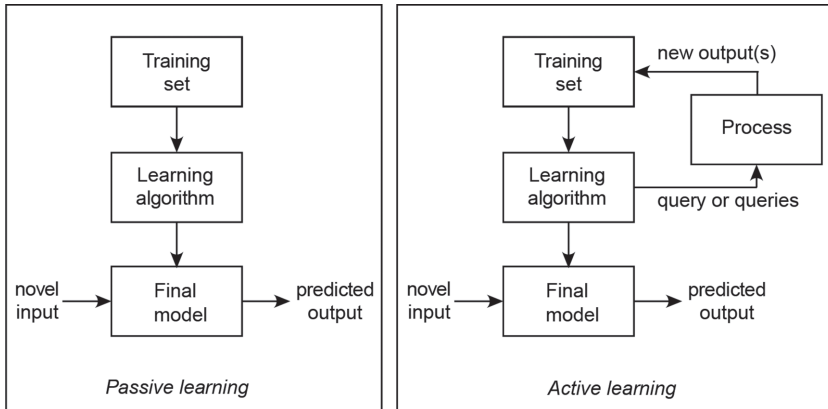


Figure 1: Comparison of passive and active learning strategies.

goal-oriented active learning strategy here is an extension of the hierarchical local model tree for design of experiments (HilomotDoE) proposed in Ref. [11]. The main focus of the query strategy of HilomotDoE is to reduce the error of a local model network (LMN). Since the error of the model can be decomposed into a bias and variance part [12], HilomotDoE addresses and reduces both error parts as described in [13].

The extension of HilomotDoE proposed in this paper is meant for models that should be utilized for optimization tasks and is described further in Section 2. It is successfully used for the generation of training data for a computational fluid dynamics (CFD) metamodel. This metamodel describes the total-to-static efficiency for a given design point depending on geometrical parameters of an impeller and is valid for all typical design points of centrifugal fans according to the Cordier diagram. The optimization task is to find geometrical parameters of the impeller for a given design point such that the efficiency of the centrifugal fan is maximized, see Section 3 for more details. A customization and application of the proposed active learning strategy for the CFD metamodel task together with the corresponding results is described in Section 4 before Section 5 concludes this paper.

2 ACTIVE LEARNING STRATEGY

The pursued active learning strategy deals with models utilized for optimization tasks and it exploits special properties of LMNs. Through the structure of LMNs it is possible to assign *local* model errors to specific areas in the input space. New measurements are preferably placed in such high-error regions, while concentrating on presumably optimal geometries that differ most from the ones already available in the training data. Therefore a lot of *potential* queries are generated, from which only one or a small subset is chosen to be measured. These potential queries will be named *candidate points* from now on. In order to focus on presumably optimal geometries, all existing candidate points are obtained through optimizations. For these optimizations the currently available model is used. In order to concentrate on areas with high local errors, only candidate points lying in these regions are considered during the selection of new queries. The similarity between the leftover candidate points and the already existing training data is evaluated in terms of the Euclidean distance. As a result, the candidate point closest to the largest hole in the training data within the area of the highest model error is chosen as query. More details about LMNs, the active learning strategy, and the new generation method of the candidate points are given in Sections 2.1, 2.2 and 2.3.

2.1 Local model networks

LMNs follow a divide-and-conquer strategy. The whole input space, here spanned by the physical inputs \underline{u} , is divided into subregions. In each subregion a local model (LM) \hat{y}_i is estimated. So called validity functions Φ_i define regions in which the LMs are valid and how interpolation between neighboring LMs should be conducted. According to [14] the model output \hat{y} of a LMN with M LMs is calculated by

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

For a reasonable interpretation the so-called partition of unity has to be fulfilled [12], i.e. the validity functions Φ_i have to sum up to one at any point in the input space.

For the active learning strategy the training algorithm Hilomot (hierarchical local model tree) [15] is used with local affine models. The schematic LMN training procedure of Hilomot can be explained with the help of Fig. 2. The algorithm incrementally grows an LMN, i.e. in each iteration, the number of LMs is increased by one. At the beginning there is only one LM valid in the whole \underline{u} -input space. Then all axes-orthogonal splits through the center of the so-called *parent* LM are tested (dashed lines in Fig. 2). Hilomot uses the best orthogonal split as an initialization for a nonlinear split optimization in which the position and orientation of that split are further adjusted. Only the current split is optimized, all already existing splits are kept unchanged. The parameters of the two LMs affected by the split adjustment are obtained through a weighted least squares technique nested in the nonlinear split optimization. After that the LM with the worst *local* error measure (gray shaded regions in Fig. 2) are subdivided further. The resulting validities are utilized to weight the errors of the whole LMN yielding the local error measures. After the first split is performed one additional split is tested before the nonlinear optimization starts, which is the direction of the parent split going through the center of the parent LM. The algorithm stops as soon as a measure for the LMN’s generalization performance gets worse. Usually Akaike’s information criterion (AIC) [16] or a distinct validation data set is used to evaluate the generalization performance of the LMN. Through the nonlinear split optimization Hilomot allows for an axes-oblique input space partitioning yielding advantages especially for high-dimensional input spaces. For more details on how the validities are constructed and additional advantages of Hilomot compared to other LMN training algorithms please refer to Ref. [12].

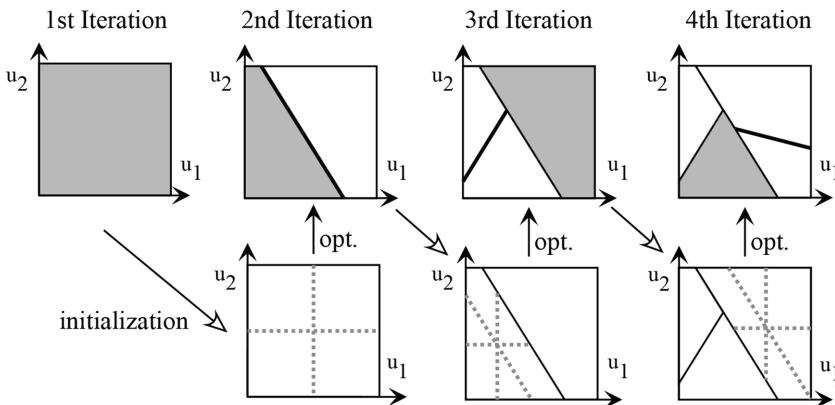


Figure 2: Schematic Hilomot training procedure for a 2-dimensional input space.

2.2 Active learning with local model networks

The active learning algorithm HilomotDoE proposed in Ref. [11] is described in the following. The novelty of this paper regarding the candidate point generation is outlined in Section 2.3. Another innovation is concerned with the generation of more than one query and is explained at the end of this section. As already mentioned the active learning strategy with LMNs aims at reducing the error of the final model. It is assumed that the highest error reduction can be obtained if the queries are placed in the areas of the input space with the highest local error measure $e_{LM,i}$ $i \in \{1, \dots, M\}$. For the calculation of the local error measures $e_{LM,i}$ the measured outputs \underline{y} , the local model outputs $\hat{\underline{y}}$, the effective number of local parameters $n_{eff,i}$ and the local validity matrix $\underline{Q}_i = \text{diag}(\Phi_i(\underline{u}))$ are required:

$$e_{LM,i} = \sqrt{\frac{(\underline{y} - \hat{\underline{y}})^T \underline{Q}_i (\underline{y} - \hat{\underline{y}})}{\text{trace}(\underline{Q}_i) - n_{eff,i}}}, i \in \{1, \dots, M\}. \quad (2)$$

Each squared error between an LM output and the measured output is weighted with the corresponding validity value and is divided by the leftover degrees of freedom. For more details on how to compute the effective number of parameters $n_{eff,i}$ for each LM please refer to Ref. [13].

Figure 3 visualizes the situation during the active learning phase for a two-dimensional input space if only one query is sought. An LMN is trained with Hilomot based on all currently available data. As one result the partitioning of the input space is obtained and for each LM a local error measure according to (2) can be calculated, see Fig. 3a. In the original, not yet extended version of HilomotDoE, candidate points are generated through random sampling from a uniform distribution (dots in Fig. 3b). From all randomly generated candidate points only the ones lying in the LM with the worst local error measure are considered in the following. The candidate point inside the worst LM that fills the greatest hole of the already measured training data is chosen as query. In order to find the greatest hole, the nearest neighboring training data point to each considered candidate is determined. The candidate point

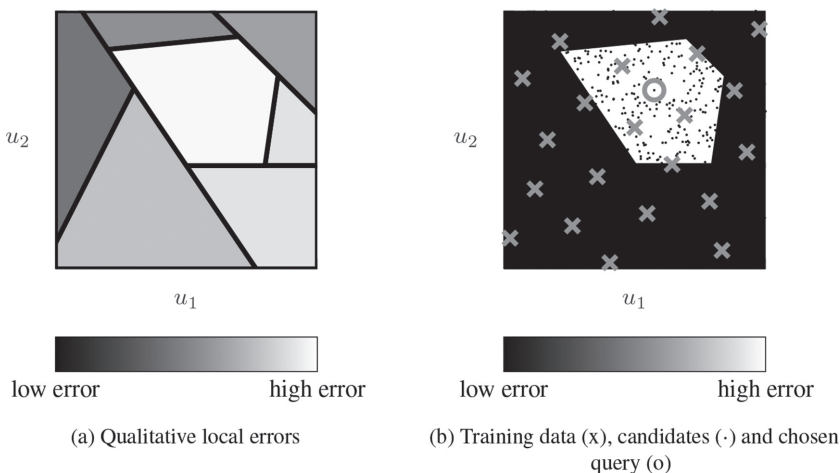


Figure 3: Partitioning of a LMN with local errors (a) and with focus on the worst LM together with training data (x), candidates (-) and the chosen query (o).

with the greatest distance to its nearest neighboring training data point is chosen as query. During the distance calculations, all training data samples are considered, but only candidates within the worst performing LM are taken into account. After the measurement for the current query has finished the training data set is updated, a new LMN is trained with Hilomot and the next query can be determined.

For algorithmic efficiency reasons it might be reasonable to request more than one query at a time. If $n_q > 1$ queries are demanded, a new strategy for HilomotDoE is proposed. Still Hilomot is used to train an LMN and yield a partitioning of the input space. Then, all local error measures are calculated and normalized, such that their sum equals one:

$$\hat{e}_{LM,i} = \frac{e_{LM,i}}{\sum_j^M e_{LM,j}} \tag{3}$$

The number of demanded queries from each LM $n_{qLM,i}$ is obtained by rounding the overall number of demanded queries n_q multiplied with the corresponding normalized local error measure $\hat{e}_{LM,i}$:

$$n_{qLM,i} = [n_q \cdot \hat{e}_{LM,i}]. \tag{4}$$

With this approach more queries are demanded from an LM the bigger its local error measure is in relation to the local error measures of all other LMs.

2.3 Generation of candidate points

Another novelty of this paper lies in the generation of the candidate points, which are subsequently used by HilomotDoE. A block diagram of the whole goal-oriented active learning procedure is shown in Fig. 4. The candidate points are generated through an evolutionary optimization algorithm based on the currently available model, trained with all yet accessible measurements. More details of the used evolutionary optimization algorithm can be found in Ref. [17]. With the help of the model the objective function $J(\underline{u})$ can be determined for each design vector $\underline{u}^T = [u_1 \ u_2 \ \dots \ u_{nd}]$. In order to distribute the initial values for the optimization runs in a space-filling manner throughout the whole input space, the extended deterministic local search (EDLS) algorithm proposed in Ref. [18] is used to generate a maximin Latin Hypercube (LH) design. To prevent the optimization from generating too similar candidate points, n_c constraints for each initialization point coming from the LH design are applied. If no constraints would be demanded almost all optimizations might lead to the same optimum. More details about the used constraints follow in the next paragraph. All resulting candidate points are provided to HilomotDoE, that determines queries as described in Section 2.2. After the queries are measured, they are added to the available measurements. Through the altered training data the model and therefore the outcome of optimizations with the new model might change, even if neither the constraints nor the LH design is changed.

If no constraints would be applied to the optimization runs, a lot of very similar or identical candidate points could possibly be the result, depending on the properties of the optimization problem. Additionally, these constraints might reflect later usage scenarios, if some design parameters are bounded, e.g. due to limited available construction space. If the number of optimization variables is n_d there are $n_c = 2^{n_d}$ possible constraint combinations specifying which variables are held fixed. For each point contained in the LH design all possible constraint combinations are applied. Constrained optimization variables are held fixed at the

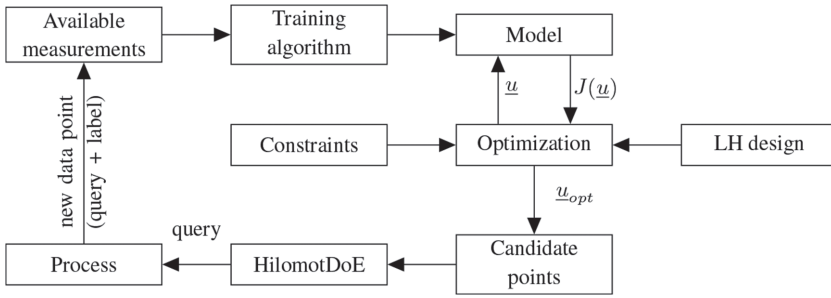


Figure 4: Block diagram of the goal-oriented active learning procedure.

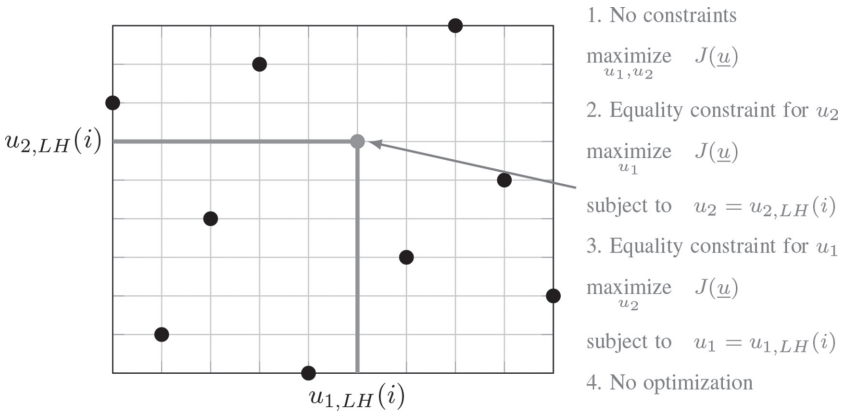


Figure 5: Illustration of all optimization scenarios for point *i* of an LH design with two inputs.

Table 1: Each column contains one constraint for the optimization. An optimization variable u_j might either be constrained to a specific value ($u_{j, LH}(i)$) or can be free, i.e. the value is determined through the optimization (opt.).

Optimization scenario:	1	2	3	4
Opt. variable u_1	opt.	opt.	$u_{1, LH}(i)$	$u_{1, LH}(i)$
Opt. variable u_2	opt.	$u_{2, LH}(i)$	opt.	$u_{2, LH}(i)$

value of the corresponding LH design point. As an example Fig. 5 together with Table 1 illustrate this for $n_d = 2$ optimization variables for the *i*-th point of the LH design. Four optimization scenarios arise with different constraints; all listed in Fig. 5 and Table 1. The variable $u_{j, LH}$ denotes the value coming from a point in the LH design that is fixed during the corresponding optimization run. In case of opt. in a field of Table 1, the value of the corresponding variable is yielded by the optimization. With this approach $N_c = N_{LH} \cdot n_c$ optimization runs are necessary to obtain all candidate points.

3 APPLICATION: AERODYNAMIC OPTIMIZATION OF CENTRIFUGAL FANS

The proposed goal-oriented active learning strategy is applied in the field of centrifugal fans. The general purpose of fans is to generate a gaseous fluid flow under build-up of pressure.

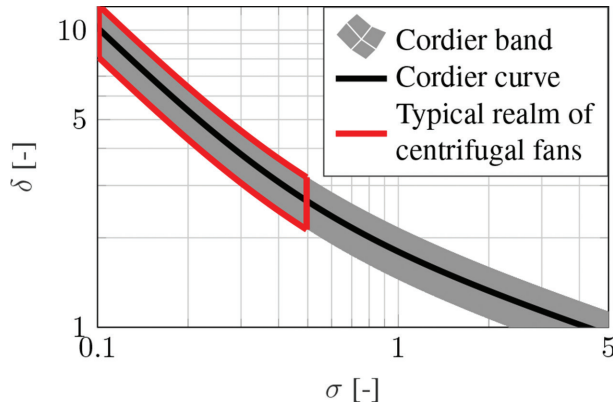


Figure 6: Cordier diagram with indication of the typical realm of centrifugal fans.

Typically, the design of a new fan comprises two main targets. Firstly, the design point (i.e. the desired flow rate Q and pressure rise Δp) must be fulfilled. Secondly, the shaft power P_{shaft} shall be as low as possible. The achievability of the first design target mainly depends on the choice of the outer fan diameter D and the rotational speed n . Cordier [19] found that the specific fan diameter

$$\delta = \frac{D}{\left(\frac{8}{\pi^2}\right)^{1/4} \left(\frac{\Delta p}{\rho}\right)^{-1/4} Q^{1/2}} \tag{5}$$

and the specific fan speed

$$\sigma = \frac{n}{(2\pi^2)^{-1/4} \left(\frac{\Delta p}{\rho}\right)^{3/4} Q^{-1/2}} \tag{6}$$

of all fans and pumps lie in a narrow band around the curve depicted in Fig. 6, which were later known as the Cordier curve and the Cordier diagram, respectively. The original Cordier diagram is based on fan performance data stemming from the 1950s, but its validity was confirmed in numerous more recent studies, see e.g. the work by Willinger *et al.* [20–22]. Figure 6 furthermore indicates the typical realm of centrifugal fans, which is in the area of low specific fan speeds but high specific fan diameters. The rest of the Cordier band is associated with other fan types such as axial or mixed flow. The second design target (the minimization of P_{shaft}) is equivalent to the maximization of the aerodynamic efficiency defined as

$$\eta = \frac{Q \cdot \Delta p}{P_{shaft}} \tag{7}$$

In the present application, only the impeller as the key component with respect to aerodynamic efficiency is investigated. The impeller geometry is described by nine geometrical parameters including the inner diameter, the inlet width, the outlet width, the inlet blade angle and the outlet blade angle. Those parameters are supposed to be most relevant in order to

adapt the fan geometry to a large variety of potential design points [23]. Optimal geometrical parameters are found by the evolutionary algorithm described in Ref. [17]. The objective function is maximization of total-to-static efficiency η_{ts} with penalty terms for the violation of the desired design point. The focus on total-to-static efficiency originates from the assumption that the fan exhausts into an open environment. Under these circumstances, the kinetic energy of the fluid at the fan exit must be considered as loss and Δp must be diminished accordingly when computing the efficiency with eqn. (7). The objective function is evaluated using metamodels of CFD. CFD is a numerical simulation technique, which solves the flow field in a discretized computational domain for given boundary conditions (e.g. flow rate, ambient pressure, velocity of moving walls, etc.). A detailed description of the CFD model can be found in [24]. This reference furthermore contains detailed information about the parameterization of the impeller. Since CFD simulations are time-consuming and require considerable computational resources, there is a strong interest to keep the number of query points to a minimum. At the same time, the metamodels must be extremely precise for optimal geometries to avoid that the optimizer exploits weaknesses of the metamodel instead of finding the real aerodynamic optimum. Given these two requirements, the active learning strategy suggested in this paper is well-suited to be applied in the fan optimization problem.

4 ACTIVE LEARNING FOR THE CFD METAMODEL

In case of the active learning for the CFD metamodel the procedure illustrated in Fig. 4 is applied with minor adjustments to meet problem specific needs. A maximin LH design for an 11-dimensional input space is generated with $N_{LH} = 586$ samples. 2 of the 11 inputs specify a design point defined by the pressure rise and a specific volume flow rate. These two inputs always belong to the set of constrained optimization variables, since the efficiency is maximized for these design points. The remaining $n_d = 9$ inputs correspond to the geometric parameters that should be optimized. It follows that there are $n_c = 2^{n_d} = 512$ possible optimization problems with all constraint combinations. As explained in Section 2.3, all possible combinations of constrained optimization variables are applied to each point contained in the LH design, resulting in $N_c = 512 \cdot 586 = 300032$ candidate points after all optimization runs are finished. Here, 500 queries are demanded in each loop of the active learning strategy. As soon as one loop is finished, i.e. the optimization of all candidate points is accomplished and queries are calculated, the list of queries to be CFD-simulated is updated. This procedure avoids any time lags between CFD simulations due to not in time finished query optimizations.

4.1 Results

The influence of the proposed goal-oriented active learning strategy is evaluated based on three aspects. First, the model quality over the number of samples available for the training is assessed. Second, the extension of achievable design points in the Cordier diagram is shown. And third, the improvements of the achievable total-to-static efficiency are visualized for all achievable points in the Cordier diagram.

For the model quality assessment CFD-based optimizations are completed for 545 different design points. The optimized designs together with the achieved total-to-static efficiencies serve as test data for the models generated with different amounts of training data. The target value of the model is the total-to-static efficiency η_{ts} . Figure 7 shows the curve for the model quality versus the training data amount. Lower root mean squared errors (RMSE) correspond

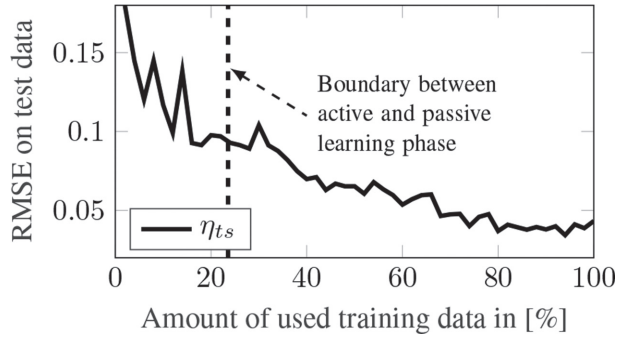


Figure 7: RMSE on test data originating from CFD-based optimization runs vs. amount of training data.

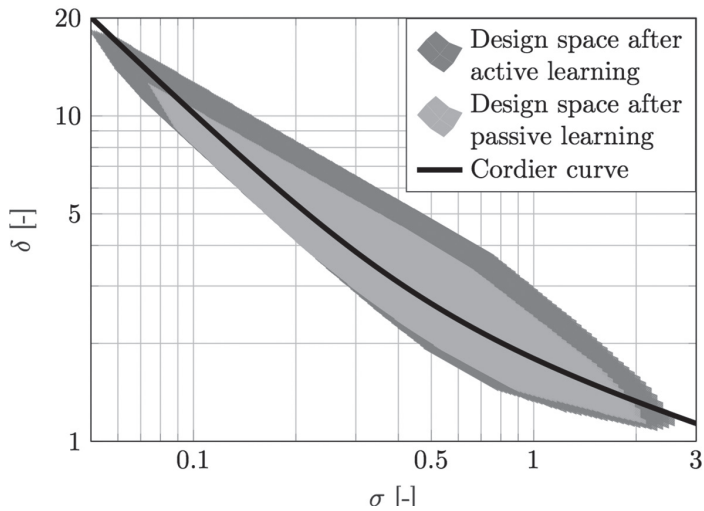


Figure 8: Extension of achievable design point area in the Cordier diagram.

to better generalization performances. The dashed line marks the point, where the passive learning phase has ended and the active learning strategy has started.

Figure 8 shows the extension of achievable regions in the Cordier diagram through the proposed goal-oriented active learning strategy. For almost each viable specific fan speed σ higher specific fan diameters δ are possible. Additionally, lower and higher specific fan speeds could be achieved through the active learning phase compared to the passive one.

The absolute total-to-static efficiencies in the area of possible impeller designs after the active learning strategy and the corresponding improvements are shown in Fig. 9a and b, respectively. The absolute efficiency could be improved up to a value of $\Delta\eta_{ts} = 0.3$ ($\approx 100\%$) by the active learning strategy. The improvements are mostly achieved above the Cordier curve, where the specific fan diameter δ is relatively high. The efficiency improvements are only shown in areas, where data from the passive learning phase is available, compare to Fig. 8. That is the reason, why the area covered by possible designs is less in Fig. 9b compared to Fig. 9a.

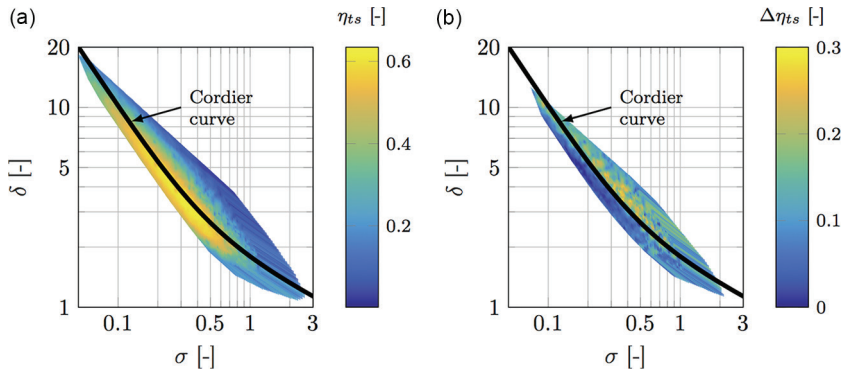


Figure 9: Absolute total-to-static efficiencies (a) and improvements through the active learning strategy (b).

5 CONCLUSION

The proposed goal-oriented active learning strategy with local model networks (LMN) is meant for models that should be utilized for optimization tasks. Active learning strategies allow for an interactive adaptation of the design of experiments (DoE) to the process under investigation. Information gathered through already obtained measurements are used to select new *queries*. This adaptation has the potential to achieve the same model quality with significant less data compared to a-priori fixed DoEs (passive learning). Through an extension of the already proposed hierarchical local model tree for design of experiments (HilomotDoE) [25] algorithm, a goal-orientation is introduced. These three goals are (I) the concentration on possibly optimal geometries, (II) the focus on areas in the input space with minor generalization performance and (III) a high diversity of training data samples. Therefore, the way *potential* queries or candidate points are generated is adjusted, such that all three goals are met. The proposed methodology is applied to the generation of training data for a computational fluid dynamics (CFD) metamodel. It is shown, that the area of achievable design points is extended, see Fig. 8, and the total-to-static efficiencies could be improved up to 100% in some areas of the Cordier diagram, see Fig. 9. Even though the application incorporates a metamodel, there is no principal restriction to metamodeling tasks of the proposed active learning strategy.

ACKNOWLEDGMENT

This work was funded by the German Ministry for Economic Affairs and Energy (BMWi), the German Federation of Industrial Research Associations (AiF) and the Research Association for Air and Drying Technology (FLT).

REFERENCES

- [1] Fedorov, V.V., *Theory of Optimal Experiments*, Elsevier, 1972.
<https://doi.org/10.2307/2334826>
- [2] Atkinson, A., The usefulness of optimum experimental designs. *Journal of the Royal Statistical Society Series B (Methodological)*, pp. 59–76, 1996.
<https://doi.org/10.2307/2344398>

- [3] Fedorov, V.V. & Hackl, R., *Model-Oriented Design of Experiments, Volume 125 of Lecture Notes in Statistics*, Springer Science & Business Media, 2012.
<https://doi.org/10.1007/978-1-4612-0703-0>
- [4] Jin, R., Chen, W. & Sudjianto, A., On sequential sampling for global metamodeling in engineering design. *ASME 2002 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, pp. 539–548, 2002.
<https://doi.org/10.1115/DETC2002/DAC-34092>
- [5] Wang, G.G. & Shan, S., Review of metamodeling techniques in support of engineering design optimization. *Journal of Mechanical Design*, **129**(4), pp. 370–380, 2007.
<https://doi.org/10.1115/1.2429697>
- [6] Cohn, D.A., *Minimizing statistical bias with queries*. Technical report, DTIC Document, 1995.
- [7] Cohn, D.A., Neural network exploration using optimal experiment design. *Neural Networks*, **9**(6), pp. 1071–1083, 1996.
[https://doi.org/10.1016/0893-6080\(95\)00137-9](https://doi.org/10.1016/0893-6080(95)00137-9)
- [8] Cohn, D.A., Ghahramani, Z. & Jordan, M.I., Active learning with statistical models. *Journal of Artificial Intelligence Research*, **4**, pp. 129–145, 1996.
- [9] MacKay, D.J., Information-based objective functions for active data selection. *Neural Computation*, **4**(4), pp. 590–604, 1992.
<https://doi.org/10.1162/neco.1992.4.4.590>
- [10] Settles, B., Active learning literature survey. *University of Wisconsin, Madison*, **52**, pp. 55–66, 2010.
- [11] Hartmann, B. & Nelles, O., Adaptive test planning for the calibration of combustion engines - methodology. *Design of Experiments (DoE) in Engine Development*, pp. 1–16, 2013.
- [12] Nelles, O., *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*, Springer, 2001.
- [13] Hartmann, B., *Lokale Modellnetze zur Identifikation und Versuchsplanung nichtlinearer Systeme*. Ph.D. thesis, Universität Siegen, 2014.
- [14] Murray-Smith, R. & Johansen, T., Local learning in local model networks. *Artificial Neural Networks, 1995, Fourth International Conference on, IET*, pp. 40–46, 1995.
<https://doi.org/10.1049/cp:19950526>
- [15] Nelles, O., Axes-oblique partitioning strategies for local model networks. *IEEE International Symposium on Intelligent Control*, Munich, Germany, pp. 2378–2383, 2006.
- [16] Akaike, H., A new look at the statistical model identification. *Automatic Control, IEEE Transactions on*, **19**(6), pp. 716–723, 1974.
<https://doi.org/10.1109/TAC.1974.1100705>
- [17] Bamberger, K., *Aerodynamic Optimization of Low-Pressure Axial Fans*. Ph.D. thesis, University of Siegen, 2015.
- [18] Ebert, T., Fischer, T., Belz, J., Heinz, T., Kampmann, G. & Nelles, O., Extended deterministic local search algorithm for maximin latin hypercube designs. *IEEE Symposium on Computational Intelligence in Control and Automation (CICA)*, Cape Town, South Africa, pp. 375–382, 2015.
<https://doi.org/10.1109/SSCI.2015.63>
- [19] Cordier, O., Ähnlichkeitsbedingungen für Strömungsmaschinen. *Brennstoff-Wärme-Kraft (BWK)*, **5**(10), pp. 337–340, 1953.

- [20] Willinger, R., Das CORDIER-Diagramm für Stromungsarbeitsmaschinen: Eine theoretische Begründung mittels Stufenkennlinien. *VDI-Berichte*, **2112**, pp. 17–28, 2010.
- [21] Willinger, R., Theoretical interpretation of the CORDIER-lines for squirrel-cage and cross-flow fans. *Proceeding ASME TurboExpo*, Copenhagen, Denmark, pp. 675–684, 2012.
<https://doi.org/10.1115/GT2012-68186>
- [22] Willinger, R. & Kohler, M., Influence of Blade Loading Criteria and Design Limits on the Cordier-Line for Axial Flow Fans. *Proceeding ASME TurboExpo*, Dusseldorf, Germany, 2014.
<https://doi.org/10.1115/GT2014-25140>
- [23] Bommers, L., Fricke, J. & Grundmann, R., *Ventilatoren*. Vulkan Verlag, Essen, 2003.
- [24] Bamberger, K., Belz, J., Carolus, T. & Nelles, O., *Aerodynamic optimization of centrifugal fans using CFD-trained meta-models*. 16th International Symposium on Transport Phenomena and Dynamics of Rotating Machinery (ISROMAC), Hawaii, USA, 2016.
- [25] Hartmann, B., Ebert, T. & Nelles, O., Model-based design of experiments based on local model networks for nonlinear processes with low noise levels. *American Control Conference (ACC)*, *IEEE*, pp. 5306–5311, 2011.