Analytic Multilayer Perceptron based Experiment Design for Nonlinear Systems *

Markus Stadlbauer * Christoph Hametner **
Stefan Jakubek *** Thomas Winsel ****

* Institute of Mechanics and Mechatronics
Vienna University of Technology
Wiedner Hauptstrasse 8-10/325/A5, 1040 Vienna, Austria
(e-mail: markus.stadlbauer@tuwien.ac.at)
** (e-mail: christoph.hametner@tuwien.ac.at)
*** (e-mail: stefan.jakubek@tuwien.ac.at)
**** AVL List GmbH
Hans List Platz 1,8020 Graz
(e-mail: Thomas.Winsel@avl.com)

Abstract: Data based modelling requires informative input and output process data gained from experiments in order to parametrize a model. Model based design of experiments is targeted to make experiments as informative as possible so as to reduce experimental effort by simultaneously gathering all relevant information of the process. In this paper a novel analytic approach for optimal model based design of experiments is presented, which can be applied to either nonlinear static or nonlinear dynamic systems. The analytic optimisation of the proposed method is based on multilayer perceptron networks. The presented DoE approach also considers the incorporation of input signal constraints and its effectiveness is demonstrated by means of a nonlinear static and a nonlinear dynamic simulation example. A comparison with state of the art DoE methods is given.

Keywords: DoE, A/D - optimality, Fisher information matrix, nonlinear systems, perceptron network, optimal experiment design

1. INTRODUCTION

The target of system identification is the mathematical description and modelling of a process based on measured input and output data. The achievable quality of the model essentially depends on the information content of the used data from a preceding experiment, e.g. Ljung [1999]. Optimal experiment design (OED) is targeted to retrieve the information which is required to properly parametrize a model with as little effort as possible, e.g. Pronzato [2008]. In this context it is fundamental to distinguish between design of experiment (DoE) for static and dynamic processes. In the static case only the spatial distribution of the system inputs is subject to optimisation whereas in the dynamic case also the temporal behaviour of the input signals is decisive.

If the process is unknown space-filling designs for the distribution of the amplitudes are desired, in order to cover the entire input space. Here, space optimal designs, which are based on distance criteria between the design points are state of the art methods, e.g. Santner et al. [2003]. For linear dynamic systems pseudo random binary signals (PRBS) have proven to be a very powerful tool for identification tasks, Goodwin and Payne [1977]. Amplitude modulated pseudo random binary signals (APRBS) have been introduced in order to track the behaviour of nonlinear systems, see e.g. Nelles [2001].

As opposed to these general DoE approaches, model based design of experiments is specifically adjusted to the system which has to be identified, because an existing model is used for the maximization of the information obtained from the experiment. The Fisher information matrix is a statistical measure of the information content of data respectively about the covariance of estimated parameters, e.g. Kay [1993]. Popular design criteria like the trace of (A-optimality) and the determinant of (D-optimality) are based on the Fisher information matrix, see e.g. Franceschini and Macchietto [2008]. In general model based DoE is applicable if:

- A model of a system is available and some parts of the system have been modified without changing the system characteristics significantly.
- A model of a related system is at disposal.
- The system acts in changed operating conditions.

For dynamic systems model based DoE can also be used for online experiment design. Here the model is continuously update with actual measurement data so that the DoE of sequentially generated future system inputs depends on the results of previous experiments. In this context the compliance of input and output constraints is necessary in order to prevent violation of system limits.

The present paper proposes and discusses an analytic approach for an optimal DoE by the use of static and
dynamic multilayer perceptron (MLP) networks. The proposed optimisation of the Fisher information matrix of an MLP is appropriate for designing experiments for a wide class of deterministic static and dynamic systems with measurement noise.

In literature different methods for OED based on neural networks are proposed: Cohn [1996] applies OED to learning problems by allowing the learner at each time step to select a new input from a candidate set. In the work of Deflorian and Klöpper [2009] a locally sequential optimal DoE approach is formulated as a model predictive control problem and the optimisation of excitation signals is based on a given APRBS. In the papers of Ayeb et al. [2006] and Ayeb et al. [2005] the D-optimality criterion is applied for the reduction of training data for global dynamic MLPs. The approach of Doherty et al. [1996] seeks to improve the performance of a dynamic neural network by the use of modified random amplitude signals. As opposed to a random amplitude signal which is characterised by a fixed time constant and a random amplitude, the input signal is modified so that the distribution of the system output is forced to a more uniform distribution.

In contrast to most state of the art approaches in this paper an analytic optimisation of the Fisher matrix is proposed, which does not depend on a candidate set of feasible input signals. For nonlinear dynamic systems represented by an MLP with feedback structure a method of every input signal on all future system outputs so that the necessary equations are stated. Section 3 treats the is organized in three stages. In section 4 the effectiveness of this approach. Finally, section 5 summarizes the results and gives an outlook into ongoing and future research.

2. MULTILAYER PERCEPTRON ARCHITECTURE

2.1 Multilayer Perceptron

MLP networks are widely used as a nonlinear model architecture, since they belong to the class of universal approximators, cf. Hornik [1991], Rojas [1996]. In this paper a two-layered feedforward perceptron structure for multiple input single output (MISO) systems is used, as shown in Fig.1: The nonlinear mapping of the network inputs \( \varphi \) to the model output \( \hat{y} \) is done via the input weights \( \omega \), the nonlinear sigmoid activation function \( f_i \) in the hidden layer and the output weights \( W \):

\[
\omega = \begin{bmatrix} \omega_1 & \ldots & \omega_{1n_u} \\ \vdots & \ddots & \vdots \\ \omega_{n_h0} & \ldots & \omega_{n_hn_y} \end{bmatrix}, \quad W = [W_{10} \ldots W_{1n_h}]^T
\]

Here, \( n_x \) is the number of network inputs and \( n_h \) is the number of hidden units or neurons, which determines the approximation capability of the MLP. In vector matrix notation the input to the hidden layer \( h \) and the output of the hidden layer \( H \) are given by:

\[
h = \omega \varphi^T, \quad H(h) = \begin{bmatrix} f_1(h_1) \\ \vdots \\ f_{n_h}(h_{n_h}) \end{bmatrix}
\]

For a more convenient notation the input and output weight matrices are combined to the parameter vector \( \theta \):

\[
\theta = [W_{10} \ldots W_{1n_h}, \omega_{10} \ldots \omega_{1n_u}, \ldots \omega_{n_h0} \ldots \omega_{n_hn_y}]^T
\]

The model output is a function \( g \) depending on the regression vector \( \varphi \).

\[
y = g(\varphi, \theta) = W^T \begin{bmatrix} 1 \\ H \end{bmatrix} = \sum_{j=1}^{n_h} W_{1j} \begin{bmatrix} n_x \end{bmatrix}^T (\varphi_j + \omega_{j0}) + W_{10}
\]

In this context it is important to mention that the \( j \)-th input of the hidden layer contains its own bias term \( \omega_{j0} \) with \( j = 1 \ldots n_h \) and at the output of the hidden layer the bias term \( W_{10} \) is added. It is supposed that the \( k \)-th measured system output \( y(k) \) can be approximated by the model output \( \hat{y}(k, \theta) \) plus some error \( e(k) \), which is assumed to be Gaussian having zero mean and variance \( \sigma^2 \):

\[
y(k, \theta) = g(\varphi(k), \theta) + e(k) = \hat{y}(k, \theta) + e(k)
\]

2.2 Configuration and Training of the MLP

Up till now it is not specified whether the MLP is used for static or dynamic systems. In the following the general description for dynamic systems will be presented, whereat the static case can be considered as a special case.

Dynamic MLP: Throughout this paper for dynamic systems the network is used in the output error configuration, Nørgaard et al. [2000]. The regressor vector \( \varphi(k) \) at the \( k \)-th observation consists of \( n \) past network outputs and \( m \) past inputs with dead time \( d \):

\[
\varphi(k) = [\hat{y}(k-1, \theta) \ldots \hat{y}(k-n, \theta) u_1(k-d) \ldots u_m(k-d-m)]
\]

For a fixed number of neurons \( n_h \) the weights have to be adjusted to the experimental input and output data. It is supposed that the input signal \( u \) with \( N \) observations and \( n_u \) different inputs and the associated output signal \( \hat{y} \) is available for the training of the neuronal network:
\[ \tilde{u} = \begin{bmatrix} u(1) \\ \vdots \\ u(N) \end{bmatrix} = \begin{bmatrix} u_1(1) & \ldots & u_n(1) \\ \vdots & \ddots & \vdots \\ u_1(N) & \ldots & u_n(N) \end{bmatrix} \quad \tilde{y} = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix} \] (8)

Static MLP: For nonlinear static systems there is no feedback of past model outputs and due to the time independence of the system \( m \) and \( d \) are zero:

\[ \varphi(k) = [u_1(k) \ldots u_n(k)] \] (9)

MLP Training: For the training of the network a standard Levenberg Marquardt algorithm is used. In doing so the weights which minimise a quadratic cost function \( V_N(\theta) \) based on the prediction error \( \epsilon(k) \) are determined:

\[ V_N(\theta) = \frac{1}{2N} \sum_{k=1}^{N} (y(k)-\hat{y}(k, \theta))^2 = \frac{1}{2N} \sum_{k=1}^{N} \epsilon^2(k, \theta) \] (10)

3. ANALYTICAL MLP BASED DOE

3.1 Fisher Information Matrix

Model based DoE aims at making experiments as informative as possible. In doing so a measure for the information content is needed, in order to be able to optimise input signals. Since the Fisher information matrix gives statistical evidence about the information content of the underlying data and the covariance of the estimated parameters, respectively, it is a basis for common design criteria like A/D-optimality.

The Fisher information matrix is a real valued function of the parameter sensitivity vector \( \psi(k) \), which describes the dependence of the model output on the model parameters:

\[ \psi(\varphi(k)) = \frac{\partial \hat{y}(\varphi(k), \theta)}{\partial \theta} \] (11)

The Fisher information matrix comprises the parameter sensitivity vectors of all observations \( k = 1 \ldots N \) according to

\[ \mathbf{I}(\psi) = \frac{1}{\sigma^2} \hat{\psi}^T \hat{\psi} \quad \hat{\psi} = \begin{bmatrix} \psi^T(\varphi(1)) \\ \vdots \\ \psi^T(\varphi(N)) \end{bmatrix}, \] (12)

where the parameter sensitivity matrix \( \hat{\psi} \) combines the parameter sensitivities for all observations. The calculation of the parameter sensitivity is done separately for the MLP output weights \( \psi_W \) and the input weights \( \psi_o \):

\[ \psi_W(\varphi(k), \theta) = \frac{\partial \hat{y}(\varphi(k), \theta)}{\partial W} = \begin{bmatrix} 1 \\ H(h(\varphi(k), \theta)) \end{bmatrix} \] (13)

\[ \psi_o(\varphi(k), \theta) = \frac{\partial \hat{y}(\varphi(k), \theta)}{\partial \omega} = d_i a g(H'(h(\varphi(k), \theta)))W_i [1 \ var(\phi(k))] \in \mathbb{R}^{[n_h] \times [1+n_o]} \] (14)

The output weight vector without bias term is indicated with \( \hat{W} \)

\[ \hat{W} = \begin{bmatrix} W_{11} & \ldots & W_{1n_h} \end{bmatrix}^T \] (15)

\[ H'(h(k)) = \begin{bmatrix} \partial H_i(h_i(k)) \\ \vdots \\ \partial H_{i+h}(h_i(k)) \end{bmatrix} \quad i = 1 \ldots n_h, \] (16)

\[ H'(h(\varphi(k), \theta)) = 1-H(h(\varphi(k), \theta)) \circ H(h(\varphi(k), \theta)), \] (17)

and \( d i a g(H'(h(\varphi(k), \theta))) \) is the diagonal matrix of the derivative of the output of the hidden layer with respect to \( h(\varphi(k), \theta) \) and \( \circ \) denotes the Hadamard product.

3.2 Design Criteria

Popular design criteria based on the Fisher information matrix are A- and D- optimality. In order to get a scalar value for optimality of the Fisher information matrix, A-optimality uses the trace of \( \mathbf{I}^{-1}(\psi) \), which is equivalent to the sum of the parameter variances:

\[ J_A(\psi) = Tr(\mathbf{I}^{-1}(\psi)) \] (18)

D-optimality uses the determinant of \( \mathbf{I} \), which is in contrast to A-optimality more sensitive to single parameter covariances, since the determinant equals the product of the eigenvalues:

\[ J_D(\psi) = \det(\mathbf{I}(\psi)) \] (19)

The design criterion \( J \) can only be influenced by changing the input signal. For the optimisation of the design criterion usually a candidate set of feasible inputs is generated, from which certain inputs are selected that optimise the design criterion.

The proposed method for model based DoE realizes the minimization of (18) and the maximization of (19), respectively by an analytical calculation of optimised input signals.

3.3 Analytical Optimisation of the Design Criterion

The proposed method for the analytical calculation of optimised input signals is done in two steps:

- Compute the gradient of the design criterion with respect to the network input or the dynamic inputs, respectively.
- Update the inputs recursively, optionally observe constraints.

The main task of the proposed method is based on the calculation of the gradient of the design criterion with respect to the input signal \( u(k) \). The gradient serves as a basis for input signal improvement. Goodwin and Payne [1977] proposed this method for linear systems in combination with a gradient algorithm.

**Determination of the gradient:** The derivative of the design criterion with respect to the input signal is calculated by the use of the chain rule in three steps:

\[ \frac{dJ}{du_i(k)} = \sum_{l=1}^{N} \frac{dJ}{d\psi^T(\varphi(l), \theta)} \frac{d\psi(\varphi(l), \theta)}{d\varphi(l)} \frac{d\varphi(l)}{du_i(k)} \] (20)

(i) The derivative of the design criterion with respect to the \( l \)-th parameter sensitivity vector is required. This results in (21) for A-optimality and in (22) for D-optimality, cf. Petersen and Pedersen [2008]:

\[ \frac{dJ_A}{d\psi^T(\varphi(l), \theta)} = -2S(l)\tilde{\psi}^T \tilde{\psi}^{-2} \] (21)

\[ \frac{dJ_D}{d\psi^T(\varphi(l), \theta)} = 2S(l)J_D(\tilde{\psi})\tilde{\psi}^T \tilde{\psi}^{-1} \] (22)

\[ S(l) = [0 \ldots 1 \ldots 0] \in \mathbb{R}^{1 \times N} \] (23)

Here, \( S(l) \) represents the single entry vector, which equals 1 at the \( l \)-th position and 0 elsewhere.

In (21) and (22) it is stated, that the inverse of the Fisher information matrix must exist, in order to calculate its
derivative. Fukumizu [1996] showed that a singular Fisher information matrix based on a MLP network can be made invertible by the elimination of redundant neurons.

(ii) The derivative of the parameter sensitivity vector for the output weights \( \psi_W(\phi(l), \Theta) \) with respect to the regressor vector \( \phi(l) \) is given by:

\[
\frac{d\psi_W(\phi(l), \Theta)}{d\phi(l)} = \text{diag}(H'(h(\phi(l), \Theta)))\hat{\omega} + \text{diag}(H''(h(\phi(l), \Theta)))\hat{\omega}\phi(l, \Theta) \in \mathbb{R}^{[1+n_u] \times n_u}
\]

(24)

\[
\hat{\omega} = \begin{bmatrix}
\omega_{11} & \ldots & \omega_{1n_u} \\
\vdots & \ddots & \vdots \\
\omega_{n_u1} & \ldots & \omega_{n_un_u}
\end{bmatrix}
\]

(25)

The input weights without bias terms are marked with \( \hat{\omega} \) and the derivative of the parameter sensitivity of the input weights \( \psi_{\hat{\omega}}(\phi(l), \Theta) \) with respect to the \( i \)-th component of the regressor vector \( \phi(l) \) is given by:

\[
\frac{d\psi_{\hat{\omega}}(\phi(l), \Theta)}{d\phi(l)} = \text{diag}(H'(h(\phi(l), \Theta)))\hat{\omega} + \text{diag}(H''(h(\phi(l), \Theta)))\hat{\omega}\phi(l, \Theta)
\]

(26)

\[
H''(h(\phi(l), \Theta)) = -2H'h(\phi(l), \Theta) + H'h(\phi(l), \Theta)
\]

(27)

\[
e_i = [0 \ldots 0 \ldots 1 \ldots 0] \in \mathbb{R}^{1 \times n_u}
\]

(28)

The direction vector of the \( i \)-th component of \( \phi(l) \) is indicated by \( e_i \).

(iii) For static systems the regressor vector \( \phi(l) \) only contains inputs \( u_i(l) \) (9). Therefore the derivative of the regressor vector with respect to the inputs is given by:

\[
\frac{d\phi(l)}{d\phi(l)} = \delta_{ij}\delta_{lk} 1 \leq i, j \leq n_u \delta_{ij} = \begin{cases} 1, & \text{for } i = j \\ 0, & \text{for } i \neq j \end{cases}
\]

(29)

Here, \( \delta_{ij} \) is the Kronecker delta, which is 1 for \( i = j \) and 0 for \( i \neq j \) and \( i, j \) denote the input index from 1 to \( n_u \).

Dynamic inputs do not only have a direct effect on the model output but also an indirect influence via the past \( n \) model outputs according to (7). Taking this fact into account the derivative of \( \hat{y}(l, \Theta) \) with respect to \( u_i(l-j) \) is given by (30) and it can be used for the calculation of the derivative \( \hat{y}(l+1, \Theta) \) with respect to \( u_i(l-j) \) in the next time step. In such a way the derivatives of all model outputs, which are needed for the regressor vector at observation \( l \) can be determined recursively:

\[
\frac{d\hat{y}(l, \Theta)}{d\phi(l)} = \frac{d\phi(l)}{d\phi(l)} \frac{d\hat{y}(l-1, \Theta)}{d\phi(l)} + \ldots + \frac{d\phi(l)}{d\phi(l)} \frac{d\hat{y}(l-n, \Theta)}{d\phi(l)} \frac{d\hat{y}(l-j, \Theta)}{d\phi(l)} + \ldots +
\]

(30)

The derivative of \( \hat{y}(\phi(l), \Theta) \) with respect to the elements of the regressor vector is given in vector matrix notation by:

\[
\frac{d\phi(l)}{d\phi(l)} = WH'\text{diag}(H'(h(\phi(l), \Theta)))\hat{\omega}
\]

(31)

Recursive input signal optimisation: The calculation of the gradient (20) for all different inputs \( u_i(k) \) with \( i = 1 \ldots n_u \) for all observations \( k = 1 \ldots N \) is the basis for the recursive input signal optimisation. Since the compliance to input constraints has to be assured during the optimisation procedure a constrained gradient method is chosen, Seyr and Jakubek [2007]. The principle of the method is explained in Fig. 2 for a two dimensional example. At every iteration (indexed by \( \nu \)) the quadratic difference between the gradient \( \delta^{(\nu)} \triangledown J^{(\nu)} \) and the input signal increment \( \Delta u \) is minimised while simultaneously the feasible area defined by the constraint vector \( G \leq 0 \) is approached. Here, \( \delta^{(\nu)} \) denotes the variable step length of the gradient method. Mathematically the problem is stated as:

\[
(\Delta \bar{u} - \delta^{(\nu)} \triangledown J^{(\nu)})^T(\Delta \bar{u} - \delta^{(\nu)} \triangledown J^{(\nu)}) \rightarrow \min \Delta \bar{u}
\]

(32)

s.t. \( G^{(\nu)} + \triangledown G^{(\nu)^T} \Delta \bar{u} \leq 0 \)

(33)

For example, in order to constrain the input signal to the interval \([\bar{u}_{\min}, \bar{u}_{\max}]\) the following constraints are formulated for \( \Delta \bar{u} \):

\[
\bar{u} \leq \bar{u}_{\max}, \quad \bar{u}^{(\nu)} + \Delta \bar{u} \leq \bar{u}_{\max}, \quad I \Delta \bar{u} - \bar{u}_{\max} + \bar{u}^{(\nu)} \leq 0
\]

(34)

\[
\bar{u} \geq \bar{u}_{\min}, \quad \bar{u}^{(\nu)} + \Delta \bar{u} \geq \bar{u}_{\min}, \quad -I \Delta \bar{u} + \bar{u}_{\min} - \bar{u}^{(\nu)} \leq 0
\]

(35)

For the solution of this inequality constrained optimisation problem see Seyr and Jakubek [2007].

4. DEMONSTRATION EXAMPLES

In this section the effectiveness of the proposed method for model based DoE with MLP networks is demonstrated by means of a static and a dynamic process, respectively. It is shown how the determinant of the Fisher matrix is iteratively improved under the compliance to input constraints and a comparison to traditional approaches is given. Based on this initial DoE, which is generated for both examples, the new method is applied to enhance the input signals. The achieved improvements are demonstrated by a statistical evaluation of the quality of the model that has been obtained from the optimised training data.

4.1 DoE for a static process

Process description: In Fig. 3 the used static process based on the nonlinear function \( y(u_1, u_2) \) with two inputs \( u_1 \) and \( u_2 \) is depicted:

\[
y(u_1, u_2) = u_1 e^{-u_1^2 - u_2^2}, \quad u_1, u_2 \in [-2, 2]
\]

(36)

MLP for model based DoE: In order to provide a proper approximation capability of the process a reference MLP network with 15 neurons is generated by means of the NNSYSID Toolbox for Matlab, Nørgaard et al. [2001].
In this context the reference model is considered as real process in form of a MLP network and it is used for the presented input signal optimisation. For this reason validation data is also generated with the reference MLP model in order make the evaluation of the presented method independent of the quality of the MLP reference model.

Initial DoE: As initial DoE 65 points are positioned space optimally in the $u_1-u_2$ plane. Space optimality means that the minimal distance between the design points is maximised.

Results: In order to compare the initial DoE to the presented method for model based DoE two MLPs are trained, one with data from the initial DoE and the other one with the optimised input signal and the sum of squared error is determined. The generation of the initial DoE and the training of the MLPs as well as the validation is repeated 100 times, in order to gain representative results. In Fig. 4 exemplary the iterative optimisation of the determinant of the Fisher matrix (see section 3.3) is depicted. The histogram plot in Fig. 5 indicates the relative distribution of the SSE of the initial and the model based DoE, respectively. The model based DoE shows a considerable reduction of the variance of the chi-square like SSE distribution, although the initial design is already space optimal. In Fig. 6 the improvement of the SSE is illustrated by means of a box plot. The line in the box indicates the median and the edges of the box mark the 25th and 75th percentiles. The whiskers mark the lowest and highest data point not considered as outliers, which are plotted individually by a cross.

4.2 DoE for a dynamic process

DoE for a dynamic process: In this section a single input single output (SISO) Wiener model is used to demonstrate the model based optimisation of dynamic input signals. The Wiener model under consideration is described by a linear transfer function $G(z^{-1})$ of a second order oscillatory system and a nonlinearity $NL$ at the system output.

\[ G(z^{-1}) = \frac{0.01867z^{-1} + 0.01746z^{-2}}{1 - 1.7826z^{-1} + 0.8187z^{-2}} = V(z^{-1}) \]

\[ NL : y = \arctan v \] (37)

MLP for model based DoE: As reference model a MLP network with 10 neurons is chosen, which is trained with an APRB-signal, Nelles [2001].

Initial DoE: As initial input signal again an APRB-signal with 1000 samples is used.

Results: The statistical comparison of the SSE of the initial and the model based DoE is illustrated in Fig. 8. The variance of the error of the model based DoE is considerably reduced as shown in Fig. 9. In Fig. 10 the first 300 samples of the input signal and the output of both, the initial and the model based DoE are depicted. The input signal of the model based DoE can excite the system dynamics better than the initial DoE, which means that
the dynamic optimisation makes the input signal more informative. In Fig. 11 the considerable augmentation of the determinant of the Fisher information matrix is illustrated exemplary during the iterative optimisation of the dynamic input signal.

Fig. 11. Iterative augmentation of $\det(I)$.

5. CONCLUSION & OUTLOOK

In this article a novel method for multilayer perceptron based DoE for nonlinear systems with measurement noise is proposed and the mathematical derivation is presented. The motivation for this work is the creation of an analytical method for model based experiment design, which can be used for the optimisation of both, static and dynamic input signals. In this context the implementation of input constraints is also considered. The effectiveness of the proposed concept is demonstrated by means of a nonlinear static and a nonlinear dynamic example. Future work will focus on the enhancement of the proposed method to input rate and output constraints, which will also be needed for the application of the presented approach to an online DoE procedure.

REFERENCES


