Space-filling Optimization of Excitation Signals for Nonlinear System Identification

Volker Smits[1][a] and Oliver Nelles[2][b]

[1] *DEUTZ AG, Ottostr. 1, Cologne, Germany*

[2] *Institute of Mechanics and Control - Mechatronics, University of Siegen, Paul-Bonatz-Str. 9-11, Siegen, Germany*

Keywords: Design of Experiment, Genetic Algorithm, Space-filling, System Identification of Multi-variate Nonlinear Dynamic Systems, Optimal Excitation Signals, APRBS, GOATS, iGOATS.

Abstract: The focus of this paper is on space-filling optimization of excitation signals for nonlinear dynamic multi-variate systems. Therefore, the study proposes an extension of the Genetic Optimized Time Amplitude Signal (GOATS) to multi-variate nonlinear dynamic systems, an incremental version of GOATS (iGOATS), a new space-filling loss function based on Monte Carlo Uniform Distribution Sampling Approximation (MCUDSA), and a compression algorithm to significantly speed up optimizations of space-filling loss functions. The results show that the GOATS and iGOATS significantly outperform the state-of-the-art excitation signals Amplitude Pseudo Random Binary Signal (APRBS), Optimized Nonlinear Input Signal (OMNIPUS), and Multi-Sine in the achievable model performances. This is demonstrated on a two-dimensional artificially created nonlinear dynamic system. Beside the good expectable model quality, the GOATS and iGOATS are suitable for the usage for stiff systems, supplementing existing data, and easy incorporation of constraints.

1 INTRODUCTION

"A model is worth a thousand datasets" (Rackauckas et al., 2021). This adage becomes even more obvious for a special kind of model – the data-based model. As the name suggests, these models are based on data. Therefore, they can only represent the information they can extract from the data used for model training (Heinz and Nelles, 2017; Heinz et al., 2017; Tietze, 2015). The field of Design of Experiment (DoE) is well-known for creating experiments to maximize the amount of information in the data of those experiments. Following the adage "A model is worth a thousand datasets" (Rackauckas et al., 2021), the adage "A DoE is worth a thousand datasets" also seems appropriate.

On the one hand, DoEs can be distinguished by the purpose of the model whether it shall describe the transient behavior besides the stationary behavior (dynamic DoE) or whether it only shall describe the stationary behavior (static DoE). On the other hand, a distinction can be made whether the design is created offline (passive) DoE or online (active) DoE (Heinz and Nelles, 2017). Popular offline dynamic DoEs are the step-based excitation signals such as the Optimized Nonlinear Input Signal (OMNIPUS), the Amplitude Random Binary Signal (APRBS) (and its variations), and the sinusoidal-based excitation signals Chirp and Multi-Sine (Heinz and Nelles, 2017; Heinz et al., 2017; Hoagg et al., 2006; Nelles, 2013; Tietze, 2015).

One recent study has shown that the global optimization approach of a step-based excitation signal via a genetic algorithm (GA) - Genetic Optimized Amplitude Time Signal (GOATS) - has outperformed the OMNIPUS, APRBS, Chirp, and Multi-Sine for three artificially created single-input-single-output (SISO) nonlinear dynamic systems (Smits and Nelles, 2021). The objective of the global optimization of the GOATS has been a good space-filling coverage of the space $\underline{X}$ spanned by the system's inputs $\underline{u} = [u_1, u_2, \dots, u_p]^T$ and outputs $\underline{y} = [y_1, y_2, \dots, y_o]^T$. However, the findings of the study are limited to SISO systems (Smits and Nelles, 2021).

The present paper aims to add to the current literature by extending these findings to multi-variate nonlinear dynamic systems. In this study, we also aim to develop a novel step-based signal which combines the advantages of the OMNIPUS and GOATS in order to

[a] https://orcid.org/0000-0001-8004-7957
[b] https://orcid.org/0000-0002-9471-8106

Smits, V. and Nelles, O.
Space-filling Optimization of Excitation Signals for Nonlinear System Identification.
DOI: 10.5220/0011338700003271
In *Proceedings of the 19th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2022)*, pages 255-262
ISBN: 978-989-758-585-2; ISSN: 2184-2809

weaken one major disadvantage of the OMNIPUS.

The OMNIPUS's major advantage is its incremental design so that the subsequences of the signal are space-filling. The GOATS's main advantage is its high degree of freedom due to the utilization of a global approach. Unfortunately, with the advantage of the OMNIPUS also comes the disadvantage of a lack in the degree of freedom. That is, in every iteration, OMNIPUS optimizes only one amplitude of one input. To overcome this limitation and to weaken the resulting disadvantage of previous suboptimal optimized sequences, an incremental version of the GOATS (iGOATS) is developed by considering a bigger and more complex subsequence inside one iteration. The iGOATS optimizes all inputs simultaneously and the number of subsequent steps considered in one iteration can be selected by the user. In addition, five loss functions are examined for the optimization of the GOATS and iGOATS including a novel space-filling loss function based on Monte Carlo Uniform Distribution Sampling Approximation (MCUDSA). Furthermore, a compression approach is developed to speed up the space-filling optimization. For better visualization, the investigation is shown on an artificial nonlinear first-order dynamic multi-input-single- output (MISO) system with two inputs ($p = 2$, $o = 1$).

The present paper is structured as follows. First, the methods are introduced and explained. The method section starts with the signal types developed by the authors – GOATS and iGOATS. After that, the loss functions, optimization problems, and GAs which are used for optimization of the GOATS and iGOATS are introduced. Following that, the compression algorithm for the speed up of the optimization is illustrated. The last section of the method section deals with the modeling approach which is used for model training according to the different excitation signals.

The experiment section describes the investigated artificial process and the concrete design of the excitation signals and the test signal. After that, the different excitation signals are analyzed in their space-filling property and their achieved model quality. At the end, a conclusion and an outlook are given.

2 METHOD

2.1 Signal Types

GOATS. The GOATS is a global optimized excitation signal where the occurrence and the duration of predefined amplitude levels e.g., via a static DoE method like an optimal Latin Hypercube (LHC) are optimized (Bates et al., 2004; Smits and Nelles, 2021). The design of the GOATS is illustrated in Fig. 1. The optimization parameters of the GOATS are the permutation of the predefined amplitude levels $\underline{\theta}_p$ and the duration of each level $\underline{\theta}_d$ (Smits and Nelles, 2021).

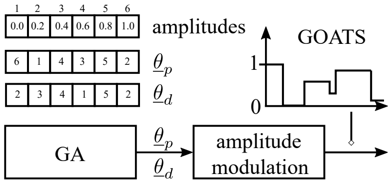


Figure 1: Example of a GOATS.

iGOATS. The iGOATS is an incremental global optimized excitation signal. It optimizes all inputs simultaneously and the number of subsequent steps can be defined by the user. The following pseudocode illustrates the procedure. It starts with an initial sequence e.g., with the upper limits of the inputs or already existing data. In every loop, a subsequence for every input dimension is optimized simultaneously and appended to the existing sequence. The number of subsequent steps h in a single iteration of the iGOATS algorithm can be chosen by the user. Furthermore, the user can specify whether all h subsequent steps should be appended or only a steps should be appended. Theoretically, the last step does not benefit from the planning feature for the next step if all subsequent steps $a = h$ are attached. Conversely, the last attached step is not negatively affected by the simultaneous planning of the next step and the planning by its predecessor. The planning feature is given when at least two subsequent steps $h = 2$ are considered. Note that, the computational demand significantly rises when only $a = h - 1$ steps are appended, because more iterations are necessary to reach the desired signal length N_{des}. The amplitude levels $\underline{\Theta}_a = [\underline{\theta}_{a,1}, \ldots, \underline{\theta}_{a,p}]$ and durations for the levels $\underline{\Theta}_d = [\underline{\theta}_{d,1}, \ldots, \underline{\theta}_{d,p}]$ of the h subsequent steps are the optimization parameters of the iGOATS, where $\underline{\theta}_{a,v} = [\theta_{a,v}(1), \ldots, \theta_{a,v}(h)]^T$ and $\underline{\theta}_{d,v} = [\theta_{d,v}(1), \ldots, \theta_{d,v}(h)]^T$.

Algorithm 1: iGOATS algorithm.

```
Step 1:  Initialize sequence
while N < N_des do
  Step 2:  Start GA with current sequence
  Step 3:  Append sequence with a subsequent
  steps of h optimized subsequent steps via
  the GA
end while
```

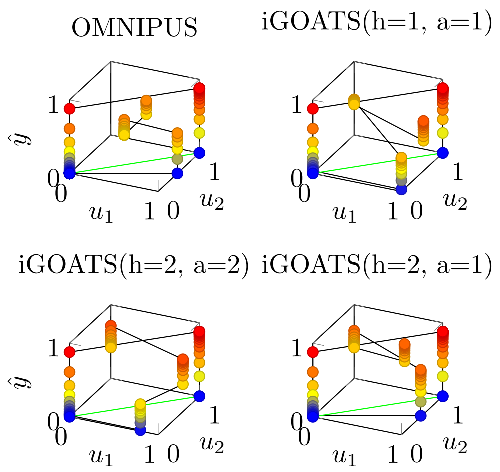


Figure 2: Comparison of OMNIPUS and iGOATS for the first five steps. The green line denotes the initial sequence.

Note that, the resulting independent sequences $\underline{u}_1, \ldots, \underline{u}_p$ have to be concatenated for model evaluation and the shortest sequence length will define the signal duration.

A good example for the disadvantages of the OMNIPUS and the planning feature of the iGOATS can be extracted by comparing the diagrams of Fig. 2. Figure 2 shows the point distribution of the two-dimensional system described by the Eq. (12) - Eq. (17) for the excitation via OMNIPUS and iGOATS. In comparison to the iGOATS, the OMNIPUS cannot reach the input combination ($u_1 = 0, u_2 = 1$) in the first five steps since it does not simultaneously optimize both inputs. The OMNIPUS needs several steps to drive the systems towards to the upper left corner.

The lower diagrams show the iGOATS with two subsequent steps h. Step four and five of the iGOATS ($h = 2$, $a = 2$) and step three and four of the iGOATS ($h = 2$, $a = 1$) steer the model output $\hat{y}$ respectively the system output y towards one just to drive it with step five respectively step four towards zero. This results in a transient response more close to the upper left corner ($u_1 = 0$, $u_2 = 1$, $\hat{y} \approx 1$) which would not be reached without the planning feature (see iGOATS ($h = 1$, $a = 1$)). As illustrated in Fig. 2 the space-filling property of the iGOATS ($h = 2$, $a = 2$) and iGOATS ($h = 2$, $a = 1$) does not differ much.

2.2 Loss Functions and Optimization Problems

Loss Functions. The optimization objectives for the genetic optimization are the cross correlation between the input sequences (see Eq. 5) and the space-filling property of the space $\underline{X} = [\underline{u}_1, \ldots, \underline{u}_p, \underline{y}_1, \ldots, \underline{y}_o]$ spanned by the system's inputs $\underline{u}_v = [u_v(1), \ldots, u_v(N)]^T$, where $v = 1, \ldots, p$ and outputs $\underline{y}_m = [y_m(1), \ldots, y_m(N)]^T$, where $m = 1, \ldots, o$. Therefore, $\underline{X} \in \mathbb{R}^{N \times n}$ where N defines the number of samples respectively the signal duration and n the number of inputs and outputs. Examples of space-filling optimized sequences are illustrated in Fig. 2 and Fig. 4. Note, that for the optimization of the space-filling criteria a proxy model (in this study a linear proxy model) $\hat{y}(\underline{u})$ is needed to roughly approximate the system's outputs ($\tilde{\underline{X}} = [\underline{u}_1, \ldots \underline{u}_p, \hat{\underline{y}}_1 \ldots, \hat{\underline{y}}_o]$, $\tilde{\underline{X}} \in \mathbb{R}^{N \times n}$).

The space-filling loss functions can be further subdivided into designs inspired by maximin design (AE, Eq. (1) and MMNS, Eq. (2)) and designs which try to approximate a uniform distribution (MCUDSA, Eq. (2) and FA, Eq. (4)). The loss functions inspired by maximin design penalize too close points. The MCUDSA and FA losses try to minimize deviation to a uniform distribution. Therefore, they make use of support points $\underline{S}$ which approximate space-filling coverage in the unit cube ($\underline{S} \in \mathbb{R}^{M \times n}$). Thereby, M defines the number of support points. In this study, $\underline{S}$ which is used during the optimization is created by $M = 1000$ points via the static DoE method optimal LHC (Bates et al., 2004). The following list summarizes the loss function equations. Note that, the i-th row of a matrix e.g., $\tilde{\underline{X}}$ is defined as $\tilde{\underline{x}}_i^T$.

- Audze Eglais (AE) (Audze and Eglais, 1977)

$$L_{\text{AE}} = N \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{k=i+1}^{N} \left(\|\tilde{\underline{x}}_i - \tilde{\underline{x}}_k\|_2 \right)^{-2} \quad (1)$$

- Maximum Nearest Neighbor Sequence (MNNS) (Heinz et al., 2017)

$$L_{\text{MNNS}} = -\frac{1}{L} \sum_{k=N+1}^{N+L} \min\left(\|(\tilde{\underline{x}}_i - \tilde{\underline{x}}_k\|_2 \right) + \#u_{v,l_v} d_{\max}, \forall i \in \{1, \ldots, N\} \quad (2)$$

- Monte Carlo Uniform Distribution Sampling Approximation (MCUDSA)

$$L_{\text{MCUDSA}} = \frac{N}{M} \sum_{i=1}^{N} \min\left(\|\tilde{\underline{x}}_i - \underline{s}_k\|_2 \right), \quad \forall k \in \{1, \ldots, M\} \quad (3)$$

- Fast and Simple Dataset Optimization (FA) (Peter and Nelles, 2019; Smits and Nelles, 2021)

$$L_{\text{FA}} = N \sum_{i=1}^{N} |1 - \hat{q}(\tilde{\underline{x}}_i)|, \quad (4)$$

$$\hat{q}(\tilde{\underline{x}}_i) = \frac{1}{M} \sum_{i=1}^{N} \frac{e^{-\frac{1}{2}[\underline{s}_k - \tilde{\underline{x}}_i]^T \Sigma^{-1} [\underline{s}_k - \tilde{\underline{x}}_i]}}{\sqrt{(2\pi)|\underline{\Sigma}|}}$$

$$\forall k \in \{1, \ldots, M\}, \underline{\Sigma} = \text{diag}(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)$$

- Cross correlation of the input sequences (XCor)

$$L_{\text{XCor}} = \frac{2}{p(p-1)} \sum_{v=1}^{p} \sum_{j=i+1}^{p} (\underline{u}_v * \underline{u}_j)[l] \tag{5}$$

$$(\underline{u}_v * \underline{u}_j)[l] = \sum_{i=1}^{N} u_v(i) u_j(i+l)$$

where time lag $l = 0$

The term $+\#u_{v,l_v} d_{\max}$ of Eq. (2) denotes a LHC-based penalization term which can be used to ensure a non-collapsing design. Thereby, $\#u_{v,l_v}$ represents the counter of already chosen amplitude levels for each input dimension and $d_{\max} = \sqrt{n}$ a factor to weigh the counter (Heinz et al., 2017)[1]. L in (2) defines the overall length of subsequent steps.

The original calculation of the FA is slightly adapted by the use of supporting points instead of the data set itself to decouple the pdf estimation of the data set. The $\hat{q}$ of the FA loss function L_{FA} can be interpreted as a n-dimensional pdf estimation where the kernels are placed on the supporting points $\underline{S}$. The standard deviations σ of the covariance matrix $\underline{\Sigma}$ of Eq. (4) are calculated by the Silverman‘s rule-of-thumb (Silverman, 1986).

Note, that all loss functions are constructed as minimization problems and the multiplication with N in the Eq. (1), Eq. (3), and Eq. (4) is performed to produce a trade-off between the signal duration and the space-filling coverage.

Optimization Problems. While the iGOATS only uses the MNNS loss function for optimization, the GOATS is examined for the three space-filling loss functions (AE, MCUDSA, FA) in a single objective optimization and in a multi objective optimization according to the AE and XCor loss functions. Note that, changes in $\underline{\theta}_p, \underline{\theta}_d$, $\underline{\Theta}_a$, and $\underline{\Theta}_d$ result in different system inputs $\underline{u}_1, \ldots, \underline{u}_p$. Consequently, it also results in different proxy model outputs $\underline{\hat{y}}_1, \ldots, \underline{\hat{y}}_o$ and in changes in the matrix $\underline{\tilde{X}}$.

$$\text{single-GOATS} : \min_{\underline{\theta}_p, \underline{\theta}_d} (f(\underline{\tilde{X}}(\underline{\theta}_p, \underline{\theta}_d))) \tag{6}$$

$$\text{multi-GOATS} : \min_{\underline{\theta}_p, \underline{\theta}_d} (f_{\text{AE}}(\underline{\tilde{X}}(\underline{\theta}_p, \underline{\theta}_d)), \quad f_{\text{XCor}}(\underline{u}_1(\underline{\theta}_p, \underline{\theta}_d), \ldots, \underline{u}_p(\underline{\theta}_p, \underline{\theta}_d))) \tag{7}$$

$$\text{single-iGOATS} : \min_{\underline{\Theta}_a, \underline{\Theta}_d} (f_{\text{MNNS}}(\underline{\tilde{X}}(\underline{\Theta}_a, \underline{\Theta}_d))) \tag{8}$$

[1](Heinz et al., 2017) addresses the topic more comprehensive, l_v denotes the level index

2.3 Genetic Algorithm

GAs are meta-heuristic global optimizers which are capable of simultaneously optimizing several parameter types respectively encodings like permutations, real-valued arrays, integer arrays, and binary representations of real and integer values (which include the parameter types of the signals GOATS and iGOATS) without needing a gradient which makes them suitable to optimize all of the proposed loss functions. For the single objective optimization via the GA the diversity-guided genetic algorithm is used (Ursem, 2002). The diversity div of the population is calculated by the mean of the standard deviation of each parameter type[2] of all individuals of the population.

$$div = \frac{1}{K} \sum_{k=1}^{K} \sigma_k \tag{9}$$

The diversity guided GA differs from the procedure of a simple GA through its separation of the genetic operators crossover and mutation in one generation controlled by a diversity mechanism (high diversity → crossover, low diversity → mutation) (Ursem, 2002).

As the selection scheme for single objective optimization the popular Tournament Selection is applied (Goldberg and Deb, 1991; Razali and Geraghty, 2011). In contrast to the diversity-guided GA of single objective optimization, the multi objective optimization problems are optimized via the Non Dominated Sorting Algorithm II (NSGA II) without diversity guiding (Deb et al., 2000). Instead, the probability for the crossover and mutation is calculated adaptively inspired by the work of Lin (Lin et al., 2003).

The crossover and mutation operators for the GOATS are chosen as in (Smits and Nelles, 2021). The real and natural number parameter types of the iGOATS are represented in a binary encoding. For this binary representation, the well-known single point crossover and flip mutation is used (Sivanandam and Deepa, 2008). The selection of genetic operators in this study is done experimental on the present problem and omitted in this contribution to conserve space.

2.4 Compression

The following compression algorithm is designed to speed up the evaluation of the space-filling loss functions. The speed up is achieved through considering not all N data points for the loss functions but only the most relevant in a space-filling sense. The general

[2]e.g., for GOATS: $\underline{\theta}_p$ and $\underline{\theta}_d$, K :=number of parameter types

idea of the compression algorithm is first to divide the time series of the system's output in the time domain by critical points and then to select the data points between the critical points uniformly in y-direction. The following pseudo code gives an overview of the general procedure. It is to note, that the critical points can

Algorithm 2: Compression algorithm.

```
1. Step: Normalize system response
2. Step: Identify critical points c
3. Step: Calculate distances Δ in
y-direction between critical points c
4. Step: Select n_i = Δ_i · α space-filling
points between c_i and c_{i+1}
for i=1:number of critical points-1 do
  4.1 Step: Create linear slope between c_i
  and c_{i+1} with n_i points
  4.2 Step: Select nearest neighbor
  (smallest Δy) for each point of the slope
  in y between c_i and c_{i+1}
end for
5. Step: Concatenate critical and
space-filling points
```

be identified in several ways. For aperiodic system responses – excited via a step-based excitation signal – only the points where the steps occur are sufficient. If the system response is not aperiodic or non-step-based excitation signals are used, the critical points can be identified by curve analysis (e.g., finding sign changes in the gradient $\Delta y(t)/\Delta t$, finding maximum curvatures $\Delta^2 y(t)/\Delta^2 t$) or by an consideration of deviations of the integrals between the true $\int y(t)\,dt$ and an approximated version $\int \tilde{y}(t)\,dt$ (e.g., by a linear extrapolation of the last two critical points). The number of points n_i which are selected between two critical points is calculated by the product of Δ_i and a user-defined factor α.[3]

Fig. 3 shows an illustration of the compression algorithm for a subsection of the system response of the linear proxy model excited by OMNIPUS. The parameters for the compression algorithm in this study has been chosen to $\alpha = 10$ and only the step position as critical points has been used. Beside the selected points in Fig. 3, the critical points are selected as well as mentioned in Algorithm 2.

Table 1 summarizes the comparison of the evaluation speed for different loss functions and for several number of points. The experiments are performed on the linear proxy model of the present process. Table 1 shows that a reduction of the evaluation speed can be achieved by compression. The reduction is significantly and lies in the range from 3.8 to 15 times.

[3] $\underline{\Delta}$ denotes the vector of differences in y-direction between each critical point

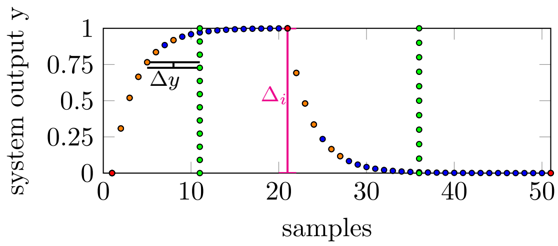


Figure 3: Compression example, red: critical points, orange: selected points, green: slope points.

Table 1: Comparison of the compression effect for the space-filling loss function. Evaluation time of compressed data in ms - Compared on an *Intel Core i7-8750 @* 2.20 GHz with the Julia Programming Language (Bezanson et al., 2017).

loss functions	number of points 500	1000	2000	speed up factor
AE	0.02	0.07	0.28	15
MCUDSA	9.13	21.7	52.6	5.5
MNNS	0.49	0.91	1.77	3.8
FA	28.2	87.1	148	4.5
compression algorithm	0.10	0.20	0.38	

For an optimization in a global manner, like for the GOATS, the evaluation speed of algorithm itself is important. It reduces the accelerations to the range from 3 to 6 times.

Another interesting question is: How does the compression affects the space-filling optimization? Exemplary, the deterministic optimization of the OMNIPUS is consulted for this analysis. Fig. 4 shows the effect of the compression on the space-filling property of the OMNIPUS. The point distribution is nearly identical because no important data points are omitted by the compression. Hence, the compression has no negative effect on the optimization. Therefore, it is carried out for all optimization of all signal types where a space-filling loss function is used.

2.5 Modeling Approach

Beside the discussed loss functions, one could wonder how to quantify the quality of an excitation sig-

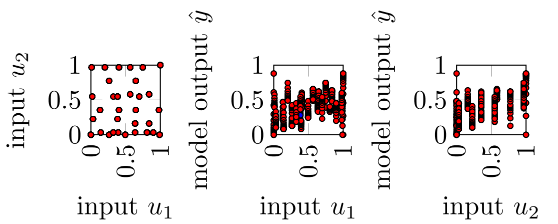


Figure 4: Effect of compression during optimization of OMNIPUS.

nal. One straightforward and reasonable approach is to compare the model qualities which can be achieved by the excitation signals. This approach of rating the excitation signals is too computational expensive to use it directly in an optimization, but for rating the results it is appropriate. Therefore, a deterministic model training is preferable, because a stochasticity during training impedes the analysis (Smits and Nelles, 2021).

A model architecture which can be optimized via a deterministic training algorithm – Hierarchical Local Model Tree (HILOMOT) – is the architecture of Local Model Networks. It achieves good model qualities and does not need much hyperparameter tuning (Nelles, 2013, 2006). HILOMOT divides the space incrementally in an axis-oblique manner and constructs local models in the subspaces. A sum of the outputs of the local models $\hat{y}_i(\underline{x})$ weighted by the validation functions $\Phi_i(\underline{z})$ results in the overall model output $\hat{y}$. Thereby, $\underline{x}$ and $\underline{z}$ are defined by the user as subsets of all inputs $\underline{u}$ (Nelles, 2006):

$$\hat{y}(\underline{x},\underline{z}) = \sum_{i=1}^{M} \hat{y}_i(\underline{x}) \cdot \Phi_i(\underline{z}) \text{ , where} \sum_{i=1}^{M} \Phi_i(\underline{z}) = 1. \quad (10)$$

3 EXPERIMENT

3.1 Process

The experiment examines a two-dimensional nonlinear artificially created dynamic system. The investigated process is a superposition of a first-order Hammerstein process with an arctangent function as stationary nonlinearity and a first-order Wiener process with a quadratic function as stationary nonlinearity. Therefore, it yields strong enough nonlinearities and dynamic aspects for a proper investigation of the excitation signals with multiple inputs. The dominant time constants of the process for each input are identified by step responses ($T_1 = 0.5\,\text{s}$ and $T_2 = 1.6\,\text{s}$). Derived from the system response of linear first-order time-invariant system, the length of one subsequence is limited to the following interval.

$$T_i/T_0 < L_i < 3pT_i/T_0 \quad (11)$$

The following equations define the process.

$$y(k) = 0.5y_1(k) + 0.5y_2(k) \quad (12)$$
$$y_1(k) = 0.2f_1(u_1(k-1)) + 0.8y(k-1) \quad (13)$$
$$y_2(k) = f_2(v(k)) \quad (14)$$
$$v(k) = 0.1u_2(k-1) + 0.9v(k-1) \quad (15)$$
$$f_1(x) = \frac{\text{atan}(8x-4)+\text{atan}(4)}{2\text{atan}(4)} \quad (16)$$
$$f_2(x) = x^2 \quad (17)$$

The process will be excited in the amplitude range of $(0.0, 1.0)$ for both inputs and a sample period of $T_0 = 0.1\,\text{s}$ is considered.

3.2 Training Signals and Test Signals

Training Signals. All training signals (APRBS, Multi-Sine, OMNIPUS, GOATS, and iGOATS) are investigated for different durations and compared on their space-filling property and their achievable model quality on the test data which is described later.

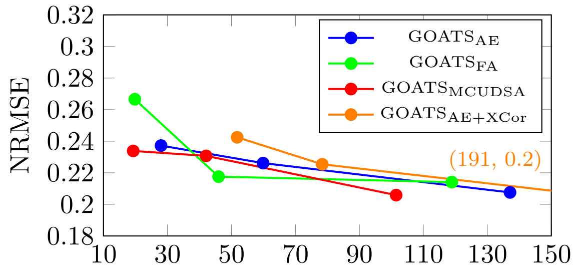
The OMNIPUS and iGOATS are incremental so that every subset can be used. For both signals, the LHC-based penalization term is used for a better comparison to the APRBS and GOATS. The hyperparameter settings for the OMNIPUS are derived from the suggestions of Heinz and Nelles (Heinz et al., 2017). The number of subsequent steps h of the iGOATS considered for optimization and appending in a single iteration has been chosen to two in order to enable the planning capability. The appending of only $a = h - 1$ subsequent steps has been analyzed but it has not shown an advantage for this investigation.

The durations of the GOATS and APRBS are mainly influenced by number of amplitude levels which are used. Therefore, three numbers of amplitude levels 25, 50, and 100 are investigated to achieve different durations.

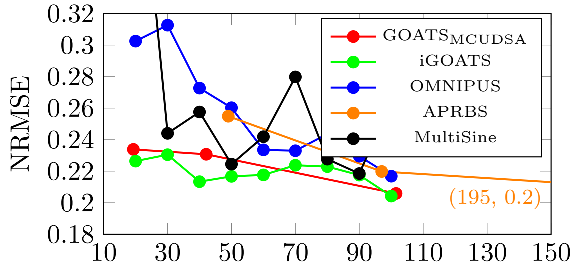
The APRBSs share the same amplitude levels as the GOATSs while the permutations of the levels are random and the dwell time is chosen to 1 s. Furthermore, one APRBS with an average model quality on the test data for each number of amplitude levels has been selected of a set of 100 APRBSs for comparison.

The Multi-Sines are created for different durations in the interval $t_{stop} = 10\,\text{s} - 100\,\text{s}$ within a frequency interval of $0.01\,\text{Hz} - 1\,\text{Hz}$ and an optimized Schroeder Phase (Schroeder, 1970). The number of sine waves considered for each Multi-Sine is calculated by $0.75 t_{stop}/s$. In addition, the Multi-Sines for the input u_1 and u_2 are optimized in their cross correlation.

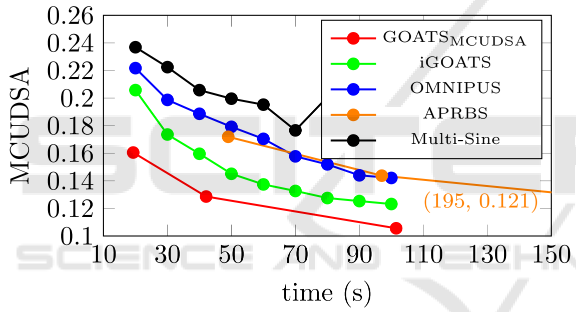
Test Signal. The test signal is a combination of APRBS, Ramp (Tietze, 2015), Chirp, and Multi-Sine with equal durations of 100 s each to cover different aspects of the process. The chosen dwell time for the APRBS and Ramp is 1 s. The Chirp and Multi-Sine are designed in the frequency range of $0.01\,\text{Hz} - 1\,\text{Hz}$.



(a) Different GOATS compared by NRMSE



(b) Best signals compared to state-of-the-art signals by NRMSE



(c) Best signals compared to state-of-the-art signals by MCUDSA

Figure 5: Comparison of excitation signals for different durations.

4 ANALYSIS OF THE TRAINING SIGNALS

The model performance is measured by the Normalized Root Mean Squared Error (NRMSE)

$$NRMSE = \sqrt{\frac{\sum_{i=1}^{N}(y(i)-\hat{y}(i))^2}{\sum_{j=1}^{N}(y(j)-\bar{y})^2}} \tag{18}$$

of the test data. Fig. 5 shows the model performances and the space-filling property of the investigated training signals.

First, in Fig. 5a the different optimizations of the GOATS are compared according to the NRMSE on the test data. The $\text{GOATS}_{\text{MCUDSA}}$ outperforms the GOATS_{AE}, GOATS_{FA}, and $\text{GOATS}_{\text{AE+XCor}}$ for short signal durations. Additionally, the optimizations according to MCUDSA result in shorter excitation signals with a comparable or superior performance compared on the number of amplitude levels (25, 50, 100). The multi objective optimization according to AE and XCor does not have a better influence on the model performance compared to the single objective according to AE. One explanation might be, that the separate influences of u_1 and u_2 on y can be extracted sufficiently from the data, even when the cross correlation of u_1 and u_2 is not optimized. Note that, the "best" solution for the multi objective optimization has been taken by normalizing the resulting pareto front of optimization and selecting the individual with the fitness closest to the origin $\underline{0}$.

Consequently, the overall winner $\text{GOATS}_{\text{MCUDSA}}$ is compared according to the NRMSE on the test data in Fig. 5b to the iGOATS, OMNIPUS, APRBS, and the Multi-Sine. Fig. 5b shows, that the GOATS and iGOATS significantly surpass the Multi-Sine in the model performance on the test data for short signal durations.

Higher NRMSE values than 0.32 are omitted in Fig. 5 for better visibility of the performance of excitation signals. The iGOATS and $\text{GOATS}_{\text{MCUDSA}}$ have comparable model performances and they outperform the performance of the OMNIPUS e.g., by approximately 34 % for the signal duration of 20 s and 20 % for the signal duration of 50 s which is even more significant compared to the findings in (Smits and Nelles, 2021). This can be explained by the higher degree of freedom of the approaches and the higher input dimension as in (Smits and Nelles, 2021). Due to the higher degree of freedom, the iGOATS and GOATS have a better space-filling property which is depicted in Fig. 5c. Fig. 5c shows the MCUDSA loss function values without the factor N for different durations evaluated on $2^{15} = 32768$ supporting points $\underline{S}$ generated via a Sobol sequence (Bratley and Fox, 1988; Joe and Kuo, 2003). Furthermore, the iGOATS and GOATS exceed the average NRMSE of an APRBS illustrated in Fig 5b, e.g., by 15 % for the signal duration of 50 s which is similar to the findings in (Smits and Nelles, 2021). At this point, it should be noted that APRBSs also often provide worse model performances than the average performance, especially for short signal durations. In contrast, the GOATS, iGOATS, and OMNIPUS overcome this disadvantage due to an ensured good coverage of the space. Therefore, they deliver a reliable expectable model performance.

5 CONCLUSION

The present study has aimed to extend the GOATS to multi-variate nonlinear dynamic systems, to develop a new signal type iGOATS, to create a new space-filling loss function MCUDSA, and to produce a compression algorithm to significantly speed up optimizations of space-filling loss functions.

The GOATS has been successfully extended to multi-variate nonlinear dynamic systems with a superior expectable model quality and space-filling property.

Furthermore, a new signal type – iGOATS – has been developed. The iGOATS combines the good expectable model qualities of the GOATS with the incremental feature of the OMNIPUS. Consequently, the GOATS and iGOATS surpass the OMNIPUS, APRBS, and Multi-Sine significantly especially for short signal durations on the artificially two-dimensional nonlinear dynamic process.

The new space-filling loss function MCUDSA for the optimization of the GOATS slightly outperforms the AE and FA loss functions in this investigation. However, for greater and more complex systems the AE might be interesting as well due to the faster evaluation speed and optimization.

The approach to accelerate the optimization speed of space-filling loss functions for dynamic DoEs via compressing the data shows that the evaluation can be sped up between 3 – 6 times according to the used loss function including the computational effort of the compression algorithm itself.

In future research, the GOATS and iGOATS have to be examined for higher dimensional, higher order and real world dynamic nonlinear systems.

REFERENCES

Audze, P. and Eglais, V. (1977). New approach for planning out of experiments. *Problems of Dynamics and Strengths*, 35:104–107.

Bates, S. J., Sienz, J., and Toropov, V. V. (2004). Formulation of the optimal latin hypercube design of experiments using a permutation genetic algorithm. In *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, volume 45.

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM review*, 59(1):65–98.

Bratley, P. and Fox, B. L. (1988). Algorithm 659: Implementing Sobol's Quasirandom Sequence Generator. *ACM Transactions on Mathematical Software (TOMS)*, 14(1):88–100.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *International conference on parallel problem solving from nature*, pages 849–858.

Goldberg, D. E. and Deb, K. (1991). A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. *Foundations of genetic algorithms*, 1:69–93.

Heinz, T. O. and Nelles, O. (2017). Iterative Excitation Signal Design for Nonlinear Dynamic Black-Box Models. *Procedia Computer Science*, pages 1054–1061.

Heinz, T. O., Schillinger, M., Hartmann, B., and Nelles, O. (2017). Excitation signal design for nonlinear dynamic systems with multiple inputs – A data distribution approach. In Röpke, K. and Gühmann, C., editors, *International Calibration Conference - Automotive Data Analytics, Methods, DoE*, pages 191–208. expertVerlag.

Hoagg, J. B., Lacy, S. L., Babuška, V., and Bernstein, D. S. (2006). Sequential multisine excitation signals for system identification of large space structures. In *Proceedings of the American Control Conference*, pages 418–423.

Joe, S. and Kuo, F. Y. (2003). Remark on Algorithm 659: Implementing Sobol's quasirandom sequence generator. *ACM Transactions on Mathematical Software*, 29(1):49–57.

Lin, W. Y., Lee, W. Y., and Hong, T. P. (2003). Adapting crossover and mutation rates in genetic algorithms. *Journal of Information Science and Engineering*, 19:889–903.

Nelles, O. (2006). Axes-Oblique Partitioning Strategies for Local Model Networks. In *IEEE International Symposium on Intelligent Control*, pages 2378–2383. IEEE.

Nelles, O. (2013). *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. Springer Science & Business Media.

Peter, T. J. and Nelles, O. (2019). Fast and simple dataset selection for machine learning. *at-Automatisierungstechnik*, 67(10):833–842.

Rackauckas, C., Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., Skinner, D., Ramadhan, A., and Edelman, A. (2021). Universal differential equations for scientific machine learning.

Razali, N. M. and Geraghty, J. (2011). Genetic algorithm performance with different selection strategiesin solving TSP. In *Proceedings of the World Congress on Engineering*, volume 2, pages 1–6.

Schroeder, M. (1970). Synthesis of low-peak-factor signals and binary sequences with low autocorrelation (corresp.). *IEEE Transactions on Information Theory*, 16(1):85–89.

Silverman, B. W. (1986). Density estimation for statistics and data analysis. *CRC press*, 26.

Sivanandam, S. N. and Deepa, S. N. (2008). *Introduction to genetic algorithms.* Berlin: Springer.

Smits, V. and Nelles, O. (2021). Genetic optimization of excitation signals for nonlinear dynamic system identification. In *Proceedings of the 18th International Conference on Informatics in Control, Automation and Robotics - ICINCO*, pages 138–145. INSTICC, SciTePress.

Tietze, N. (2015). *Model-based calibration of engine control units using gaussian process regression*. PhD thesis, Technische Universität Darmstadt.

Ursem, R. (2002). Diversity-Guided Evolutionary Algorithms. In *Parallel Problem Solving from Nature — PPSN VII*, pages 462–471.