Evolutionary based sparse regression for the experimental identification of Duffing oscillator

Saeideh Khatiry Goharoodi[a,b,*], Kevin Dekemele[a], Mia Loccufier[a], Luc Dupre[a], Guillaume Crevecoeur[a,b]

[a]*Department of Electromechanical, Systems and Metal Engineering, Ghent University, B-9052 Zwijnaarde, Belgium*
[b]*EEDT Decision and Control, Flanders Make, Belgium.*

Abstract

In this paper an evolutionary based sparse regression algorithm is proposed and applied onto experimental data collected from a Duffing oscillator setup and numerical simulation data. Our purpose is to identify the Coulomb friction terms as part of the ordinary differential equation of the system. Correct identification of this nonlinear system using sparse identification is hugely dependent on selecting the correct form of nonlinearity included in the function library. Consequently, in this work the evolutionary based sparse identification is replacing the need for user knowledge when constructing the library in sparse identification. Constructing the library based on data driven evolutionary approach is an effective way to extend the space of nonlinear functions, allowing for the sparse regression to be applied on an extensive space of functions. The results show that the method provides an effective algorithm for the purpose of unveiling the physical nature of the Duffing oscillator. In addition, the robustness of the identification algorithm is investigated for various levels of noise in simulation. The proposed method has possible applications to other nonlinear dynamic systems in mechatronics, robotics and electronics.

Keywords: Duffing oscillators, ODE identification, Friction identification, Genetic programming, Sparse regression

1. Introduction

The Duffing oscillator is a nonlinear dynamic system with a considerable number of engineering applications and presents a key benchmark in nonlinear system analysis. The ordinary differential equation of this system consists of a cubic nonlinear term which can result in chaotic behavior and bifurcation. Suppression strategies are required to accommodate for this behavior in flexible

*Corresponding author
Email address: Saeideh.KhatiryGoharoodi@ugent.be (Saeideh Khatiry Goharoodi)

robotic manipulators and high precision mechatronic systems to increase their efficacy [1]. The control performance is however drastically affected by modeling errors in the system parameters [2, 3]. Also the design of flexible manipulators and high precision systems depend on the characteristics of the Duffing oscillator parameter variations [4]. Additionally the design of harvesting devices from vibrations rely on Duffing type dynamic equations and when characterized well can be used as a tool for further analysis [5].

This work focuses on identifying the nonlinear ordinary differential equation of the Duffing system that consists of difficult to discover friction terms. Nonlinear system identification is a vast research field. The progress of this research area can be followed via several surveys including earlier works by Billings [6] and Mehra [7] as well as more recent studies [8], [9] and [10].

In cases that the nonlinear model structure can be obtained from first principles and is a priori known, the identification problem boils down to parameter estimation. Many works have been done in this area such as [11] where the physical parameter values are directly estimated using measured data. In many works such as [12], [13] and [14], the least squares method is used in order to estimate the parameter values. Others report the usage of genetic programming for the same purpose ([15], [16]).

Parameter estimation using a fixed model structure based on captured data has been previously applied on Duffing oscillator type systems. In [17], the parameters of a numerical fractional-order Duffing system has been identified using sequential differential evolution method. Other algorithms such as nonlinear subspace identification method, particle swarm optimization, Volterra-Wiener based model and Wiener-type cascade model were used to numerically estimate the parameters of Duffing-type systems [18, 19, 20, 21]. In a more recent attempt, authors in [22] have used a tailored sequential Monte Carlo algorithm within a Markov Chain Monte Carlo (MCMC) scheme to identify the parameters of Duffing in a Bayesian manner.

Alternatively when the model structure is not a priori known, the form of the model needs to be discovered. Different black-box model structures can be considered to form the system equations. In [23] a modeling method for nonlinear systems using polynomial nonlinear state space equations was introduced. Furthermore, NARMAX models have been used in [24] and [25] to represent nonlinear systems. Genetic algorithm and genetic programming have been also introduced in this field. In [26] genetic programming is used in a multiobjective fashion to generate global nonlinear models. Authors in [27] apply genetic programming to discover nonlinear differential equations. More examples of genetic algorithm application for system identification are [28], [29] and [30]. Other modeling methods are including but not limited to neuro-fuzzy methods [31] and high-order neural network structures [32].

Black-box identification of the Duffing equation has also been a matter of investigation. In [33], explorative genetic programming is used to identify the model of a noisy Duffing-van der Pol oscillator using numerical simulation data. Artificial neural networks have been used to determine the mathematical model of the damped Duffing by [34]. A similar approach was proposed based on a set of basis functions and applying least-squares in [35].

A more exploitation based nonlinear system identification approach was recently proposed in [36]. In this approach, a fixed matrix of candidate terms is first built upon prior expert knowledge. Subsequently, a linear system of equations is formulated using this matrix. The dominant terms in the constructed matrix later form the identified equation of the system. The sequential threshold least-squares algorithm is applied to find the true model of the system, depending on choosing the accurate value of the regularization parameter. A revised version of this method using the Alternating Direction Method of Multipliers (ADMM) has been successfully implemented on captured data from an experimental Duffing setup [37].

The biggest criticism towards sparse identification method lies in selecting *ad-hoc* the appropriate library functions. This problem can be observed in [37] as the identification fails to discover the friction terms existing in the experimental data as these complex non-polynomial terms are lacking in the library of functions. When identifying an experimental dataset, the friction forces within Duffing oscillators form an important model uncertainty that also arises in many other mechatronic applications such as in hydraulic actuators [38]. Nonlinear friction model parameters are reconstructed, mostly based on a priori given friction model structures such as Coulomb, Stribeck, etc. friction models. Once these friction models are correctly identified they can be used in control algorithms [39]. Consequently, in this paper we aim at implementing an evolutionary based sparse identification algorithm on numerical and experimental Duffing system. The combination of genetic programming and sparse identification algorithm has been previously suggested in [40], however no methodology has been proposed so far.

In this paper a revised version of sparse identification using the evolutionary based sparse identification algorithm is for the first time to the authors' knowledge applied on a set of real-world experimental data. This paper is organized as follows. Section 2 describes the Duffing oscillator and the collected data from the setup and simulation. Section 3 provides details on the sparse regression algorithm. Section 4 briefly introduces the genetic programming method as the base for the evolutionary construction of the library and presents the evolutionary based sparse identification algorithm to identify the model structure and parameters of the Duffing oscillator. Results and discussions are provided in Section 5 applying the identification method on experimental and numerical Duffing oscillator data. Conclusions are drawn in the final section.

2. Problem statement and data acquisition

In this work the proposed algorithm is applied on the Duffing oscillator both numerically and experimentally. The cubic Duffing equation as a differential equation with third-power nonlinear term is an example of a dynamic system that exhibits chaotic behavior and bifurcations. Experimental datasets are extracted from this setup to identify the underlying equation. Simulations from the same setup (using its characteristic parameters) implemented in MATLAB

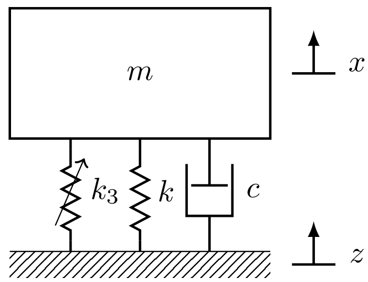


Figure 1: A mechanical Duffing oscillator subjected to imposed ground motion

environment [41], furthermore allow to examine the efficiency of the algorithm and validating the presented method. The robustness of the algorithm is investigated on the data set by increasing added noise.

2.1. Duffing oscillator: theory and experimental realization

2.1.1. Theoretical description

The mechanical Duffing oscillator with a imposed ground motion is depicted on Fig. 1 is characterized by the following dynamic equation:

$$m\ddot{x} = -c\left(\dot{x} - \dot{z}\right) - k\left(x - z\right) - k_3\left(x - z\right)^3 \tag{1}$$

with m the Duffing's mass, c its linear damping, k the linear stiffness and k_3 the cubic stiffness.

A change of coordinates to the relative ground displacement $q \triangleq x - z$ yields:

$$m\ddot{q} = -c\dot{q} - kq - k_3q^3 - m\ddot{z} \tag{2}$$

or the dynamics expressed in state space:

$$\begin{cases} \dot{q_1} = q_2 \\ \dot{q_2} = -\dfrac{c}{m}q_2 - \dfrac{k}{m}q_1 - \dfrac{k_3}{m}q_1^3 - \ddot{z} \end{cases} \tag{3}$$

2.1.2. Design principle of mechanical Duffing oscillator

To realize the mechanical Duffing oscillator, a mass-spring system, see Fig. 2, is constrained to move along a designed track $y = f(x)$. The track's shape determine the linear and nonlinear stiffness, k and k_3, (3).

If the mass is subject to a static force in the x-direction, the mass moves along the track until equilibrium is reached. It is now shown that the spring characteristic is nonlinear. The track exerts a reaction force on the followers attached to the spring, R, perpendicular to the track's curvature. The linear spring is compressed according to the track, imposing a force on the mass in

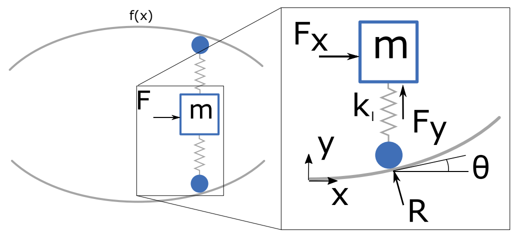


Figure 2: Design principle of Duffing oscillator

the y-direction, $F_y = k_l y(x)$. The static applied force, the reaction force on the follower, and the reaction force on the mass are related by static equilibrium:

$$F_x = 2R\sin(\theta) \quad F_y = R\cos(\theta) \Rightarrow F_x = 2F_y\tan(\theta) \tag{4}$$

with θ the tracks curvature's angle, related to the force profile by $\tan\theta = \frac{df(x)}{dx}$. If the track is $f(x) = ax^2 + b$, the spring characteristic is:

$$F_x = 2k_l f(x)\frac{df(x)}{dx} = 4k_l a(bx + ax^3) = kx + k_3x^3 \tag{5}$$

with $k = 4k_lab$ and $k_3 = 4k_la^2$. By machining a parabolic track, $f(x) = ax^2$, the linear coefficient can be simply tuned by shifting the profiles over a distance b.

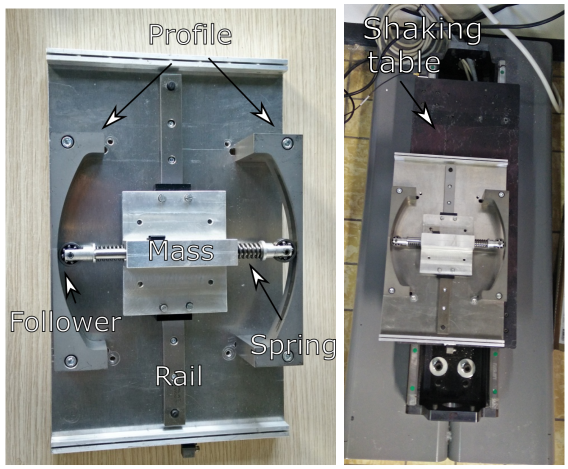
2.1.3. Experimental setup

The realized mechanical Duffing oscillator with the above mentioned design principle is shown on Fig. 3a. A mass with linear springs was fitted on a linear guide rail. Tracks with the shape with $a = 4$ m^{-1} were made from machined steel. The followers on the springs are SKF ball-bearings. The linear springs have a stiffness of $k_l = 16.7$ kN, according to the manufacturer, with the cubic stiffness then being $k_3 = 1.07$ MN/m^3. The profiles can be shifted for adjusting the b-term in Eq. (5).

To impose the ground motion, the oscillator is put on a shaking table, here a Beckhoff linear permanent magnet motor. To measure the Duffing's mass and shaking table displacement, accelerometer signals are integrated with the algorithm in [42]. For this algorithm to perform well, the signals should stay in a certain frequency band. The ground displacement imposed by the shaking table is limited in bandwidth by choosing a sine sweep and a random phase multisine.

The material contact between the followers and the track causes dry friction. The force in the y-direction F_y, will cause a perpendicular opposing friction force, μF_y, with μ the friction coefficient. The total opposing friction is:

$$F_f = 2\mu k_l(ax^2 + b)\text{sgn}(v_x) = \mu_1\text{sgn}(v_x) + \mu_2x^2\text{sgn}(v_x) \tag{6}$$



(a) (b)

Figure 3: (a) Realisation of Duffing oscillator design (b) Duffing oscillator on shaking table

with v_x the speed in the x-direction. Including the friction forces in the state space representation of the Duffing oscillator dynamics is:

$$\begin{cases} \dot{q_1} = q_2 \\ \dot{q_2} = -\dfrac{c}{m}q_2 - \dfrac{k}{m}q_1 - \dfrac{k_3}{m}q_1^3 - \dfrac{\mu_1}{m}\mathrm{sgn}(q_2) - \dfrac{\mu_2}{m}q_1^2\mathrm{sgn}(q_2) - \ddot{z} \end{cases} \tag{7}$$

The viscous damping c, linear stiffness k and dry friction coefficients μ_1 and μ_2 have to be experimentally identified.

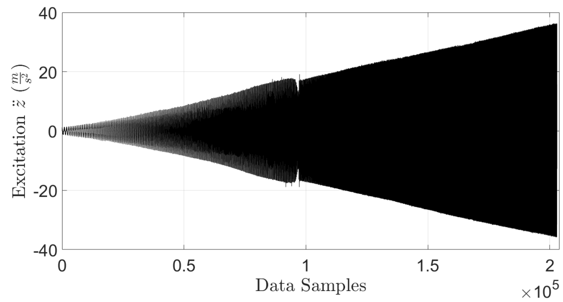
2.1.4. Experimental data

The input of the dynamical equation (7), acceleration of the shaking table $\ddot{z}$ and the relative acceleration between the mass and shaking table $\ddot{q}$, captured from the described setup are shown on Fig. 4a and Fig. 4b respectively. The excitation signal of the experiment is a sine sweep from 2 to 20 Hz. The sampling time equals 0.488 ms.

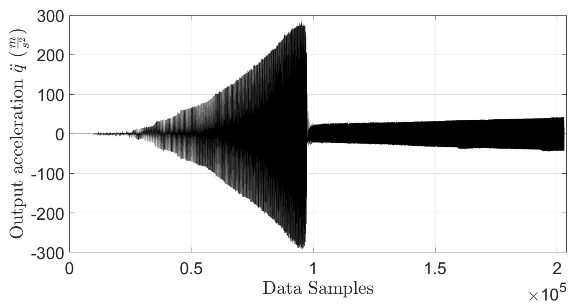
2.2. Duffing oscillator: Numerical data

In order to validate the performance of the algorithm on numerical data, the described Duffing setup has been simulated in MATLAB. The state space model used for the purpose of simulation is the same as (3).

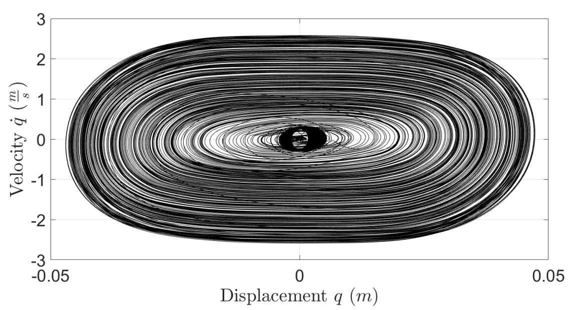
The control input of the system is a linear swept-frequency cosine presented in Fig. 5a. The noisy output acceleration obtained from numerical simulation is presented in Fig. 5b. The sampling time is 0.488 ms. The amplitude is selected such that the bifurcation is observable in the output. Considering both sets of experimental and numerical data in Fig. 4 and Fig. 5 it can be noted that the



(a)



(b)



(c)

Figure 4: Experimental data: (a) The input, the acceleration of the shaking table $\ddot{z}$ (b) The output, the relative acceleration between the mass and shaking table $\ddot{q}$ (c) Velocity versus displacement

bifurcation occurs sooner in simulation. The bifurcation of a Duffing oscillator occurs at a certain frequency of the input sweep. This frequency depends on the amplitude of the sweep, [43], which is different for the experiment and the simulation, explaining why the bifurcation happens at a different instant. The data for both (experimental, numerical) cases is divided in identification and validation parts.

3. Sparse regression

The aim of sparse regression in the field of system identification is to extract a low dimension (sparse) representation of the system from a high dimensional space of candidate representations using input and output data of the system. Considering $\mathrm{q} \in \mathbb{R}^{n\times p}$ as the data matrix with p state variables, each presented as a column of the matrix over n time instants, sparse regression determines the state space equation as a general nonlinear function g:

$$\dot{\mathrm{q}} = \mathrm{g}(\mathrm{q}, \mathrm{u}) \tag{8}$$

where $\mathrm{u} \in \mathbb{R}^{n\times 1}$ is the input of the system, $\dot{\mathrm{q}} \in \mathbb{R}^{n\times p}$ is the time derivative of the states which can be measured or numerically calculated and the q matrix (with derivatives $\dot{\mathrm{q}}$) is assumed to be fully observable.

By introducing a library of terms as functions of the states and input of the system, the identification problem can be presented as finding the sparse matrix $\xi \in \mathbb{R}^{m\times p}$ [36]:

$$\dot{\mathrm{q}} = \mathrm{A}\xi \tag{9}$$

where A is the library of (non-)linear terms.

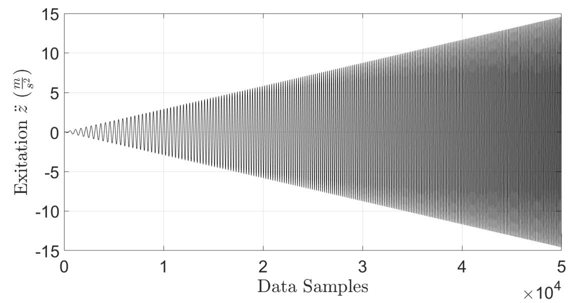
Choosing the right form of nonlinearity in the construction of the dictionary is essential in this approach which requires user knowledge. Equation (10) illustrates such a library. Each column, m, corresponds to a linear/ nonlinear term as a function of the states or the input.

$$\mathrm{A(q,u)} = \begin{pmatrix} | & | & | & & | & | & | \\ 1 & \mathrm{q} & \mathrm{q}^2 & \cdots & \mathrm{u} & \mathrm{u}^2 & \cdots \\ | & | & | & & | & | & | \end{pmatrix}_{n\times m} \tag{10}$$

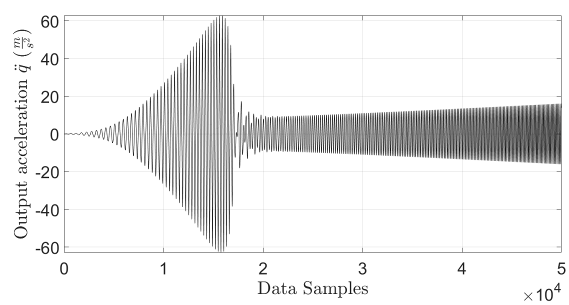
By solving Eq. (9), the dominant linear and nonlinear elements of the library A(q,u) will be chosen to combine linearly and form the equation of the system g in Eq. (8). The ξ matrix is determined by minimizing a defined optimization problem. In this paper we define the optimization problem as the elastic net regulator [44]:

$$\xi^*_{EN} = \arg\min_{\xi} \|\mathrm{A}\xi - \dot{\mathrm{q}}\|_2^2 + \lambda_1 \|\xi\|_1 + \lambda_2 \|\xi\|_2^2 \tag{11}$$

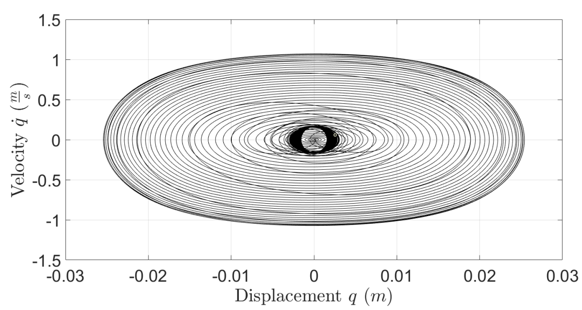
Where λ_1 and λ_2 are the hyper-parameters that are changed discretely. The order of magnitude for these hyper-parameters is defined through parameter sweep.



(a)



(b)



(c)

Figure 5: Numerical data: (a) The input, acceleration of the shaking table $\ddot{z}$ (b) The output, the relative acceleration between the mass and shaking table $\ddot{q}$ (c) Velocity versus displacement

4. Evolutionary based sparse regression methodology

4.1. Genetic programming

Genetic programming (GP) is a subclass of genetic algorithms that was first presented by Koza in 1992 [45]. The basic idea of genetic programming is to evolve populations of equations based on the captured data and the fitness function evaluation of the simulation of each equation, where each equation is presented as a tree.

In the first generation a population is randomly constructed by combining the numbers, variables and mathematical operations. Terminal nodes of the trees are occupied by variables and numbers. The operators consisting of basic algebraic operations ($+$, $-$, $\times$, $/$), functions (sin, cos, tan, abs, sgn) or user-defined functions fill in the non-terminal nodes called the primitives. Afterwards, the population can vary in two ways: crossover and mutation. A crossover happens when two parent trees randomly exchange branches to form new offspring (Fig. 6). Mutation involves random alteration of a parent's subtree (Fig. 7). In the next step, the algorithm evaluates the fitness of each tree. The next generation is built based on the fitness evaluation. Following, the algorithm cycles through this loop until it reaches the stopping criteria or its convergence. A typical error metric such as least squares or root mean squared error is used as the fitness measure.

4.2. ESparse algorithm

Following the description of sparse regression and genetic programming, in this section the proposed algorithm is described. As presented in ESparse algorithm 1, the identification procedure consists of two main steps:

1. Construction of the library ($\underset{n\times m}{\mathrm{A}}$) using genetic programming
2. Performing the sparse regression

In each iteration an ODE equation is realized by solving a layered optimization problem: the individual trees in the population are used as the functions to build the $\underset{n\times m}{\mathrm{A}}$ library in (10). Next, the sparse regression is performed on the constructed library by solving (11). Based on this approach alternative to a predefined library, the sparse regression is applied on a dynamic set of functions generated from the genetic programming. The advantage of this method is clearly its ability to generate an explorative library consisting of an extensive space of functions derived from the captured data. Moreover, this alternation between the explorative step 1 and the exploitative step 2, allows a reduction in the number of terms for regression.

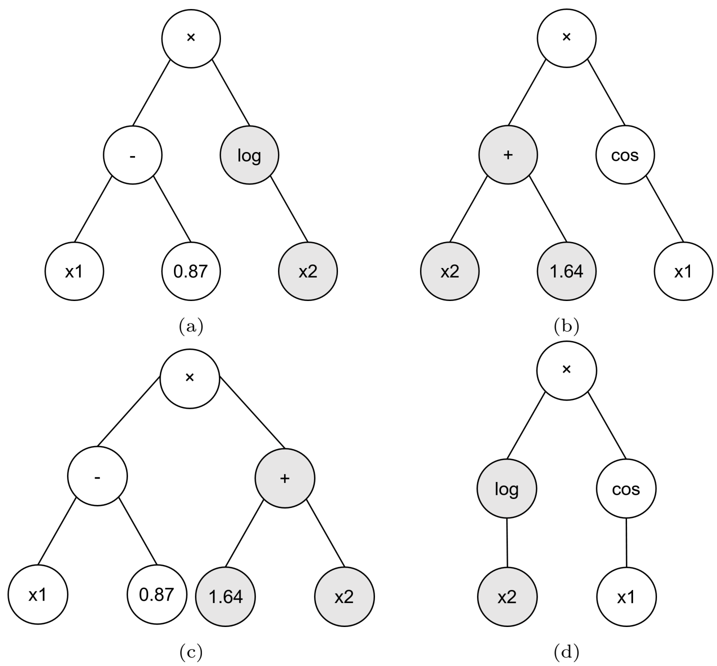


Figure 6: Illustration of the genetic programming crossover (a) First parent before crossover with a randomly selected branch (b) Second parent before crossover with a randomly selected branch (c) First offspring after crossover (d) Second offspring after crossover

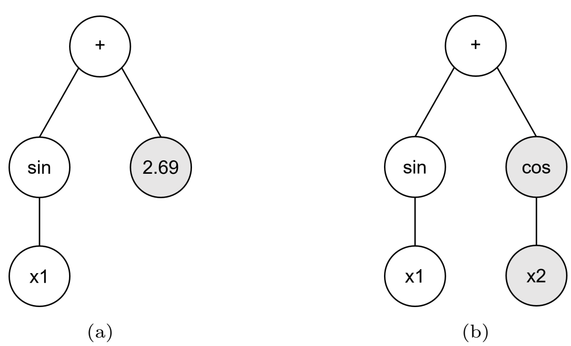


Figure 7: Illustration of the genetic programming mutation (a) Parent before mutation with a randomly selected branch (b) Offspring after mutation

Algorithm ESparse Algorithm

Require: Time-varying measurement data: $\underset{n\times p}{\mathrm{q}}$, $\underset{n\times p}{\dot{\mathrm{q}}}$ and $\underset{n\times 1}{\mathrm{u}}$. Population size r, number of generations k and probabilities of crossover and mutation.

```
procedure
    Initialize the population of size r randomly
    for i = 0 : k do
        Construct dictionary A (n×m) based on the individuals
        Solve the regression problem:
```

$$\xi^{*}_{EN} = \arg\min_{\xi} \|\mathrm{A}\xi - \dot{\mathrm{q}}\|_2^2 + \lambda_1 \|\xi\|_1 + \lambda_2 \|\xi\|_2^2$$

```
        Compute the fitness function: Mean square error
        Generate new population using crossover and mutation
    end for
end procedure
```

5. Results and discussion

In this section, the ability of identifying the correct form of the Duffing equation using the method from section 4.2 in case of both numerical and noisy experimental datasets is analyzed. Both sets of data are captured from the Duffing oscillator described in section 2, as a nonlinear dynamic system benchmark. In case of experimental data we are specifically looking for the identification of the state space including the friction term as in Eq. 7. We also investigated the robustness of the algorithm with respect to noise in the data. By changing the level of added noise in simulation and how the accuracy of the identified model is affected by that noise provides a means to assess the robustness.

5.1. Numerical Duffing

When applying the ESparse algorithm on the captured input/output dataset from Duffing oscillator simulations (Fig. 5), the state space equation is identified. The first 16000 samples (the head of the arrow) are selected for validation while the remainder are used for identification. For the numerical analysis to follow, the parameters in Eq. 3 are assumed to have the values: $m = 0.49(kg)$, $k = 487(N.m^{-1})$, $k_3 = 1.07e6(N.m^{-3})$ and $c = 1.8(N.s.m^{-1})$. The evolutionary parameters and values are presented in Table 1. Moreover q, $\dot{q}$ and $\ddot{z}$ are the inputs of the GP denoted as X0, X1 and X2. The theoretical ODE equation together with the identified model for different levels of signal to noise ratio (SNR) are shown in Table 2. The associated error percentage is calculated using validation data.

5.1.1. Robustness analysis

To demonstrate the robustness of the algorithm, various levels of Gaussian white noise with zero mean were added to the data set. Fig. 8 presents the tree

Table 1: Evolutionary parameters for the numerical Duffing

Evolutionary parameter	Value
Population size	80
Crossover rate	0.9
Mutation rate	0.1
Number of generations	30
Basis functions	*plus*, *minus*, *times*, *abs*, *sgn*

Table 2: Identified models by the ESparse algorithm, numerical Duffing

	Identified ODE equation	% error
Theoretical reference	$\ddot{q} = -3.67\dot{q} - 993.88q - 2.18e6q^3 - \ddot{z}$	-
SNR = 20 dB	$\ddot{q} = -3.67\dot{q} - 994.19q - 2.18e6q^3 - \ddot{z}$	0.7
SNR = 19.5 dB	$\ddot{q} = -3.67\dot{q} - 994.81q - 2.19e6q^3 - 0.99\ddot{z}$	1.9
SNR = 19 dB	$\ddot{q} = -3.67\dot{q} - 984.64q - 2.15e6q^3 - \ddot{z}$	3.7
SNR = 18.5 dB	$\ddot{q} = -3.73\dot{q} - 867.83q - 1.72e6q^3 - 1.02\ddot{z}$	11.2

of the identified equation in case of SNR = 19.5 dB. Moreover, the comparison between actual and identified validation data for SNR = 19.5 dB and SNR = 18.5 dB is presented in Fig. 9a and Fig. 9b respectively.

A more general assessment for large ranges of signal to noise ratio (SNR) were performed and the results are presented in Fig. 10. Each data point in this figure corresponds to the mean of the accuracy percentage of 20 identification runs. Error bars are as well depicted that relate to the standard deviation. The results suggests that the proposed algorithm possesses the capability to reveal both the structure of the governing equation as well as the parameter values of the Duffing system. For SNR values from 20 to approximately 17 dB that correspond to increasing noise level, the accuracy of the identified parameter values decreases and ultimately the accuracy of the identification procedure itself. However no additional terms appear in the discovered model indicating the robustness of the presented algorithm. As can be observed in the Fig. 10, for low SNR (lower than approximately 17 dB) more terms are added to the equations. This clearly indicates that the data becomes overfitted by the identified model ultimately resulting in deteriorated accuracies.

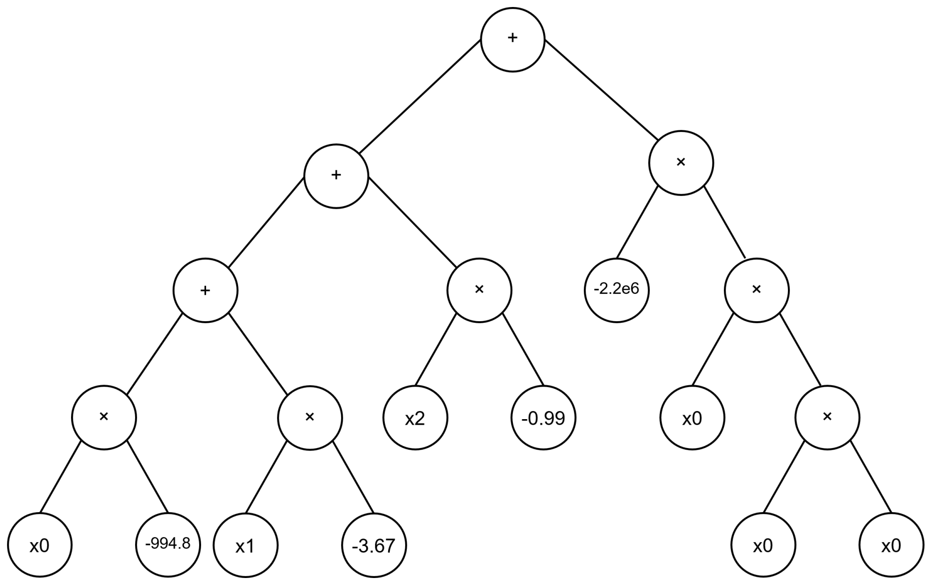


Figure 8: Tree presentation of the identified numerical Duffing with SNR = 19.5 dB

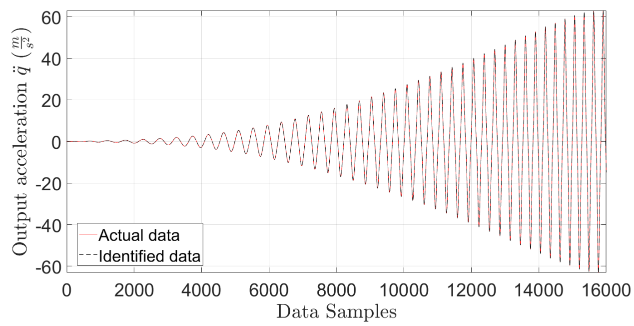
5.2. *Experimental Duffing*

Similar to the numerical Duffing data, the ESparse is applied onto experimental Duffing data with the purpose to identify the Duffing equation. We conducted three experiments with the same input acceleration profile under the same conditions, resulting in data presented in Fig. 11. In all cases, the first 90000 data samples of the control input and the output are selected for validation and the rest are used for training.

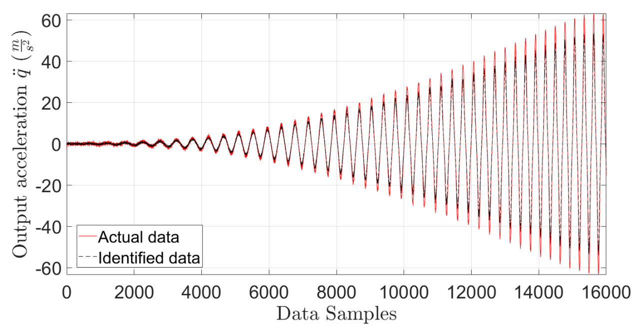
Table 3: Evolutionary parameters for the experimental Duffing

Evolutionary parameter	Value
Population size	150
Crossover rate	0.8
Mutation rate	0.2
Number of generations	40
Basis functions	*plus*, *minus*, *divide*, *times*, *abs*, *sgn*

The evolutionary parameters of the genetic programming are given in Table 3. Table 4 summarizes the identified model obtained from the three experiments using the ESparse algorithm. Additionally, Fig. 12 presents the tree of the identified equation with 5.6 % error. The terms appearing in these equations are well supported by the theoretical model from Eq. (7) that includes Coulomb



(a) SNR = 19.5 dB



(b) SNR = 18.5 dB

Figure 9: Comparison of the actual and identified numerical Duffing for (a) SNR = 19.5 dB (b) SNR = 18.5 dB

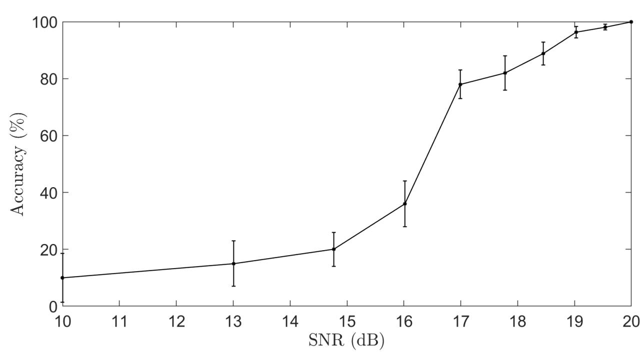
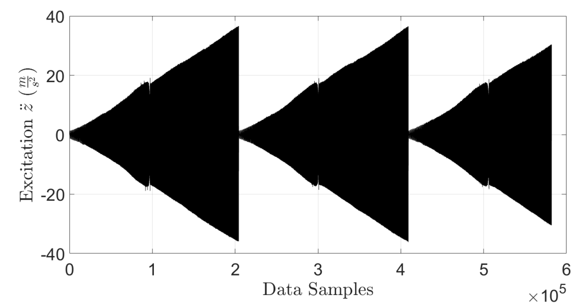
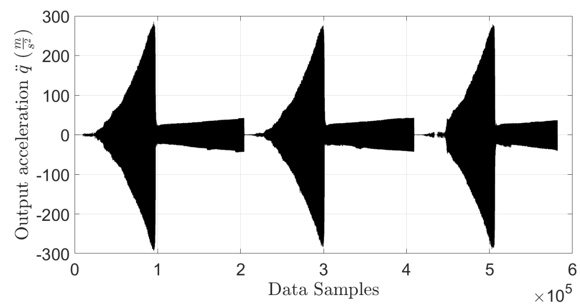


Figure 10: Mean and standard deviation of the identification accuracy for various levels of SNR.



(a)



(b)

Figure 11: Experimental data: (a) The input, the acceleration of the shaking table $\ddot{z}$ (b) The output, the relative acceleration between the mass and shaking table $\ddot{q}$

friction. The results demonstrate the ability of the algorithm to identify non-polynomial nonlinearities. Comparison between the actual and identified output acceleration data is illustrated in Fig. 13. A clear correlation between the two sets of data can be observed from Fig. 13b.

Table 4: Identified models by the ESparse algorithm, Noisy experimental Duffing

Exp.	Identified Duffing	% error
1:	$\ddot{q} = -1.10\dot{q} - 691.92q - 2.37e6q^3 - 2.93\text{sgn}\dot{q} - 7.63e3q^2\text{sgn}(\dot{q}) - 1.02\ddot{z}$	3.9
2:	$\ddot{q} = -1.23\dot{q} - 714.60q - 2.24e6q^3 - 4.11\text{sgn}\dot{q} - 8.23e3q^2\text{sgn}(\dot{q}) - 1.04\ddot{z}$	5.6
3:	$\ddot{q} = -1.22\dot{q} - 716.11q - 2.35e6q^3 - 3.81\text{sgn}\dot{q} - 8.26e3q^2\text{sgn}(\dot{q}) - 1.04\ddot{z}$	4.7

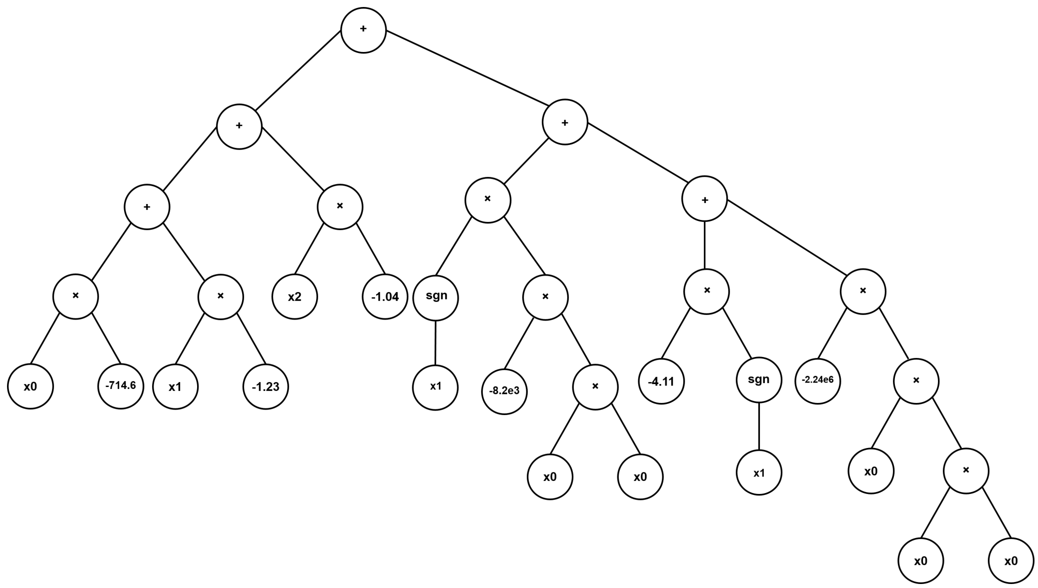
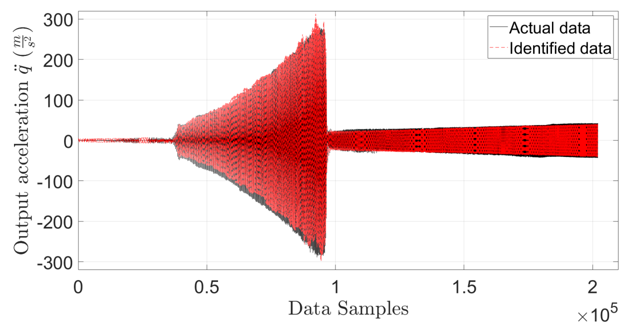


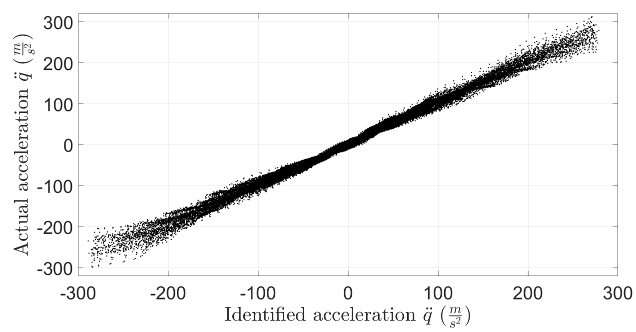
Figure 12: Tree presentation of the identified experimental Duffing 5.6 % error

5.3. Comparison to other available methods

To substantiate the advantages of the proposed ESparse algorithm, a comparison to other available methods with respect to performance measures run time and % error are drawn in Table 5. Sparse regression (see Section 3) and genetic programming (see Section 4.1) are applied on the same dataset. For the purpose of having a fair comparison, crossover and mutation probabilities as well as the employed basis functions applied for genetic programming are same as those employed in the ESparse algorithm (Table 3). However to achieve the correct model of the system using genetic programming, the population size and number of generations have to increase to 250 and 80 respectively. As suggested by the results, ESparse algorithm is capable of converging to the model with the same level of accuracy with much less computational effort. As for sparse regression method, we had to manually include the sign function as being part of the library of (non-)linear terms (coming from knowledge gained with the ESparse algorithm) since otherwise the model structure cannot be discovered, whereas the ESparse algorithm automatically builds the proper library using genetic programming.



(a)



(b)

Figure 13: Comparison of the actual and identified experimental Duffing: (a) comparison over time (b) actual acceleration versus identified

Table 5: Comparison of performance measures, experimental Duffing

Method	Run time (s)	% error
Genetic programming (exp. 1)	449.411	5.7
Genetic programming (exp. 2)	425.884	5.3
Genetic programming (exp. 3)	539.042	4.1
Sparse regression (exp. 1)	2.092	3.9
Sparse regression (exp. 2)	2.563	4.6
Sparse regression (exp. 3)	2.677	5.2
ESparse algorithm (exp. 1)	12.626	3.9
ESparse algorithm (exp. 2)	12.171	5.6
ESparse algorithm (exp. 3)	12.125	4.7

5.4. Advantages and limitations of the method

5.4.1. Advantages

The proposed methodology has major benefits in comparison to sparse regression and genetic programming based methods for non-parametric identification. The evolutionary based sparse regression requires lower computational effort relative to genetic programming based algorithms. For GP based algorithms to converge to the true solution, large populations with high number of generations are typically required. Nonetheless, the presented ESparse algorithm has the ability to converge to the correct model with less computational effort and having a balanced model complexity since ESparse alternates between exploration (genetic programming) and exploitation (sparse regression). Therefor, the algorithm can discover the system equation with fewer generations and smaller populations.

As for the sparse regression method, the strict model assumptions prior to identification can limit the model complexity while the dynamic library of functions in evolutionary based sparse regression allows for discovery of more complex models by extending the search space and replacing the need for user knowledge for the construction of the library with data driven GP step.

5.4.2. Limitations

Although the proposed algorithm allows to identify more complex non-polynomial terms in the equation such as friction terms, the basic building blocks are required to be included in the pool of the basic functions of the GP algorithm. Otherwise the identified system will only be composed of available blocks which may not represent the nature of the system accurately.

6. Conclusion

In this paper, an evolutionary based sparse regression algorithm for discovering both the structure and the parameter values of the system has been proposed. The methodology is used for the purpose of identifying the Duffing oscillator system using both numerical and noisy experimental data. In case of numerical Duffing, the data is polluted with different levels of noise to study the robustness of the algorithm. Furthermore the approach is challenged to discover governing dynamics that include non-polynomial nonlinear Coulomb friction terms, from noisy experimental Duffing data. As shown by the percentage of the identification error, the algorithm is effective in unveiling the physical nature of the Duffing oscillator. The proposed method has possible applications to other nonlinear systems such as in mechatronics, robotics and electronics.

7. Acknowledgments

This work was supported by the ICON project Multi-Sensor and MODA of Flanders Make, the Strategic Research Centre for the Manufacturing Industry; and the FWO research project G.0D93.16N. This research received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme.

8. Data Availability

All the data are included in the article. If there is a further demand for data, the author can provide depending on their availability.

9. Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] B. Chen, J. Huang, J. Ji, Control of flexible single-link manipulators having duffing oscillator dynamics, Mechanical Systems and Signal Processing 121 (2019) 44–57.

[2] K. Chen, T. Yang, K. Ou, J. Yin, Design of command shapers for residual vibration suppression in duffing nonlinear systems, Mechatronics 19 (2) (2009) 184–198.

[3] R. Naik, S. Mhalsekar, Vibration attenuation of a nonlinear oscillator by using the bound on a system, Mechanics Research Communications 85 (2017) 12–15.

[4] I. Diaz, E. Pereira, V. Feliu, J. Cela, Concurrent design of multimode input shapers and link dynamics for flexible manipulators, IEEE-ASME Transactions on Mechatronics 15 (4) (2010) 646–651.

[5] D. Barton, S. Burrow, L. Clare, Energy harvesting from vibrations with a nonlinear oscillator, Journal of Vibration and acoustics - Transactions of the ASME 132 (2) (2010) 021009.

[6] S. A. Billings, Identification of nonlinear systems–a survey, in: IEE Proceedings D (Control Theory and Applications), Vol. 127, IET, 1980, pp. 272–285.

[7] R. Mehra, Nonlinear system identification: selected survey and recent trends, IFAC Proceedings Volumes 12 (8) (1979) 77–83.

[8] G. Kerschen, K. Worden, A. F. Vakakis, J.-C. Golinval, Past, present and future of nonlinear system identification in structural dynamics, Mechanical systems and signal processing 20 (3) (2006) 505–592.

[9] O. Nelles, Nonlinear system identification: from classical approaches to neural networks and fuzzy models, Springer Science & Business Media, 2013.

[10] J.-P. Noël, G. Kerschen, Nonlinear system identification in structural dynamics: 10 more years of progress, Mechanical Systems and Signal Processing 83 (2017) 2–35.

[11] K. Mohammad, K. Worden, G. Tomlinson, Direct parameter estimation for linear and non-linear structures, Journal of Sound and Vibration 152 (3) (1992) 471–499.

[12] S. Billings, W. Voon, Least squares parameter estimation algorithms for non-linear systems (1984).

[13] M. L. Johnson, L. M. Faunt, [1] parameter estimation by least-squares methods, in: Methods in enzymology, Vol. 210, Elsevier, 1992, pp. 1–37.

[14] F. Ding, Combined state and least squares parameter estimation algorithms for dynamic systems, Applied Mathematical Modelling 38 (1) (2014) 403–412.

[15] B. Jiang, B. Wang, Parameter estimation of nonlinear system based on genetic algorithm, Control Theory and Applications 17 (1) (2000) 150–152.

[16] L. Yao, W. A. Sethares, Nonlinear parameter estimation via the genetic algorithm, IEEE Transactions on signal processing 42 (4) (1994) 927–935.

[17] L. Lai, Y.-D. Ji, S.-C. Zhong, L. Zhang, Sequential parameter identification of fractional-order duffing system based on differential evolution algorithm, Mathematical Problems in Engineering 2017 (2017).

[18] E. Gandino, S. Marchesiello, Identification of a duffing oscillator under different types of excitation, Mathematical Problems in Engineering 2010 (2010).

[19] G. Quaranta, G. Monti, G. C. Marano, Parameters identification of van der pol–duffing oscillators via particle swarm optimization and differential evolution, Mechanical Systems and Signal Processing 24 (7) (2010) 2076–2095.

[20] M. Barahona, C.-S. Poon, Detection of nonlinear dynamics in short, noisy time series, Nature 381 (6579) (1996) 215–217.

[21] G. Chen, Y. Chen, H. Ogmen, Identifying chaotic systems via a wiener-type cascade model, IEEE Control Systems 17 (5) (1997) 29–36.

[22] T. Rogers, T. Schön, A. Lindholm, K. Worden, E. Cross, Identification of a duffing oscillator using particle gibbs with ancestor sampling, in: Journal of Physics: Conference Series, Vol. 1264, IOP Publishing, 2019, p. 012051.

[23] J. Paduart, L. Lauwers, J. Swevers, K. Smolders, J. Schoukens, R. Pintelon, Identification of nonlinear systems using polynomial nonlinear state space models, Automatica 46 (4) (2010) 647–656.

[24] S. Chen, S. A. Billings, Representations of non-linear systems: the narmax model, International journal of control 49 (3) (1989) 1013–1032.

[25] A. Rahrooh, S. Shepard, Identification of nonlinear systems using narmax model, Nonlinear Analysis: Theory, Methods & Applications 71 (12) (2009) e1198–e1202.

[26] K. Rodriguez-Vazquez, C. M. Fonseca, P. J. Fleming, 'identifying the structure of nonlinear dynamic systems using multiobjective genetic programming, IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 34 (4) (2004) 531–545.

[27] G. J. Gray, D. J. Murray-Smith, Y. Li, K. C. Sharman, T. Weinbrenner, Nonlinear model structure identification using genetic programming, Control Engineering Practice 6 (11) (1998) 1341–1352.

[28] T. Kumon, M. Iwasaki, T. Suzuki, T. Hashiyama, N. Matsui, S. Okuma, Nonlinear system identification using genetic algorithm, in: 2000 26th Annual Conference of the IEEE Industrial Electronics Society. IECON 2000. 2000 IEEE International Conference on Industrial Electronics, Control and Instrumentation. 21st Century Technologies, Vol. 4, IEEE, 2000, pp. 2485–2491.

[29] J. Madár, J. Abonyi, F. Szeifert, Genetic programming for the identification of nonlinear input- output models, Industrial & engineering chemistry research 44 (9) (2005) 3178–3186.

[30] A. Akramizadeh, A. A. Farjami, H. Khaloozadeh, Nonlinear hammerstein model identification using genetic algorithm, in: Proceedings 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002), IEEE, 2002, pp. 351–356.

[31] R. Babuška, H. Verbruggen, Neuro-fuzzy methods for nonlinear system identification, Annual reviews in control 27 (1) (2003) 73–85.

[32] E. B. Kosmatopoulos, M. M. Polycarpou, M. A. Christodoulou, P. A. Ioannou, High-order neural network structures for identification of dynamical systems, IEEE transactions on Neural Networks 6 (2) (1995) 422–431.

[33] A. Bolourchi, S. F. Masri, O. J. Aldraihem, Development and application of computational intelligence approaches for the identification of complex nonlinear systems, Nonlinear Dynamics 79 (2) (2015) 765–786.

[34] S. Masri, A. Chassiakos, T. Caughey, Identification of nonlinear dynamic systems using neural networks, Journal of Applied Mechanics 60 (1) (1993) 123–133.

[35] S. Masri, J. Caffrey, T. Caughey, A. Smyth, A. Chassiakos, Identification of the state equation in complex non-linear systems, International Journal of Non-Linear Mechanics 39 (7) (2004) 1111–1127.

[36] S. L. Brunton, J. L. Proctor, J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proceedings of the National Academy of Sciences 113 (15) (2016) 3932–3937.

[37] S. K. Goharoodi, K. Dekemele, L. Dupre, M. Loccufier, G. Crevecoeur, Sparse identification of nonlinear duffing oscillator from measurement data, IFAC-PapersOnLine 51 (33) (2018) 162–167.

[38] L. Marton, S. Fodor, N. Sepehri, A practical method for friction identification in hydraulic actuators, Mechatronics 21 (1) (2011) 350–356.

[39] Q. Pan, Y. Li, M. Huang, Control-oriented friction modeling of hydraulic actuators based on hysteretic nonlinearity of lubricant film, Mechatronics 53 (2018) 72–84.

[40] M. Quade, Symbolic regression for identification, prediction, and control of dynamical systems (2018).

[41] MATLAB, version 9.2.0.538062 (R2017a), The MathWorks Inc., Natick, Massachusetts, 2017.

[42] R. Pintelon, J. Schoukens, Real-time integration and differentiation of analog-signals by means of digital filtering, IEEE Transactions on Instrumentation and Measurements 39 (6) (1990) 923–927.

[43] A. H. Nayfeh, D. T. Mook, Nonlinear oscillations, John Wiley & Sons, 2008.

[44] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67 (2) (2005) 301–320.

[45] J. R. Koza, J. R. Koza, Genetic programming: on the programming of computers by means of natural selection, Vol. 1, MIT press, 1992.