

Identification of pre-sliding friction dynamics

U. Parlitz,^{a)} A. Hornstein, and D. Engster

Drittes Physikalisches Institut, Universität Göttingen, Bürgerstraße 42-44, D-37073 Göttingen, Germany

F. Al-Bender,^{b)} V. Lampaert, and T. Tjahjowidodo

K.U. Leuven, Department of Mechanical Engineering, Division P.M.A., Celestijnenlaan 300B, 3001 Heverlee (Leuven), Belgium

S. D. Fassois^{c)} and D. Rizos

Department of Mechanical & Aeronautical Engineering, University of Patras, GR 265 00 Patras, Greece

C. X. Wong, K. Worden,^{d)} and G. Manson

University of Sheffield, Department of Mechanical Engineering, Mappin Street, Sheffield S1 3JD, United Kingdom

(Received 7 October 2003; accepted 16 March 2004; published online 21 May 2004)

The hysteretic nonlinear dependence of pre-sliding friction force on displacement is modeled using different physics-based and black-box approaches including various Maxwell-slip models, NARX models, neural networks, nonparametric (local) models and dynamical networks. The efficiency and accuracy of these identification methods is compared for an experimental time series where the observed friction force is predicted from the measured displacement. All models, although varying in their degree of accuracy, show good prediction capability of pre-sliding friction. Finally, we show that even better results can be achieved by using an ensemble of the best models for prediction. © 2004 American Institute of Physics. [DOI: 10.1063/1.1737818]

Friction is still one of the great unknowns in real-life mechanical systems. We need to better understand it; qualify it and quantify it, if we are to achieve effective characterization of special systems let alone to be able to control them. How does one go about this task? Friction is a very complex and nonlinear phenomenon, comprising various regimes and behavioral facets. In particular, prior to true sliding, there is a regime that is characterized (dominated) by hysteretic dependence of the friction force on the displacement. This paper is devoted to this regime. From a “system identification for control purposes” viewpoint, we may have different approaches; notably, physics-based and black-box approaches. The first relies on existing model structures that describe similar phenomena and are relatively easy to implement. On the other hand, it may be advantageous to identify friction without any *a priori* knowledge at all about (the physics of) the system. This paper applies, compares and contrasts those two approaches. Furthermore, we shall show that the best results of physics-based as well as black-box models may significantly be improved by combining both types of models in an ensemble approach.

I. INTRODUCTION

Friction is defined as the resistance (in the dynamic case, friction is actually not always opposed to motion; therefore, it may be better to regard friction simply as the tangential

force developed in a sliding interface) to motion when two objects are slid against one another. Depending on its type and on the application, friction can be desirable or a drawback in a system. Examples of the first situation are brakes, clutches and friction drives; of the second, are bearings, slides, and joints. One thing is common to both situations, however, and that is: One must be able to characterize the frictional behavior and, possibly, also to be able to control it, in order to ensure proper functioning of a system. Except for viscous friction (which is proportional to the velocity), all other types of behavior exhibit strong nonlinearity in the displacement, velocity and time. Thus, rather than the presence of friction in itself, it is this nonlinear behavior that makes friction an impediment to system control.¹ A classical phenomenon that is frequently encountered in systems with friction is the so-called stick-slip motion, e.g., of a driven slider, which can appear in different guises. Although there are rule-of-thumb remedies for this phenomenon, e.g., increasing the drive stiffness and/or damping, the problem of accurate tracking/positioning in the presence of friction remains open. One thing, however, is sure: In order to deal effectively with friction, some sort of model or model-structure is needed. But since friction is the result of extremely complex interactions between the contacting surfaces and lubricants, no accurate quantitative prediction of frictional behavior is yet possible based on the given material and surface properties: One must resort invariably to experimental determination. Understanding of the mechanisms involved in friction is, on the other hand, growing so that formulation of suitable models is becoming more advanced. The unknown parameters involved in such models should generally be estimated from identification tests of each given case. Notwithstanding this, here we try both “black-box”

^{a)}Electronic mail: parlitz@dpi.physik.uni-goettingen.de

^{b)}Electronic mail: farid.al-bender@mech.kuleuven.ac.be

^{c)}Electronic mail: fassois@mech.upatras.gr

^{d)}Electronic mail: K.Worden@sheffield.ac.uk

identification techniques that assume no *a priori* knowledge of the physics of the system, and physics-based techniques that are based on an assumed model structure with unknown parameters to be identified. The objective is to gauge the effectiveness of either of the approaches in identifying frictional behavior. This has important implications in regard to system identification and control as well as in validating the identification method itself in regard to universality and robustness. Finally we shall show that instead of asking “physics-based *or* black-box?” it is more efficient in terms of prediction accuracy to combine the best models from both classes in an ensemble whose predictions consist of a superposition of the results of its individual members.

In the next section, the general friction behavior will be described, together with an outline of the most important friction models and an introduction of the test setups used for obtaining friction data for the purpose of identification. In Sec. III different identification methods are introduced including physics-based models and black-box models for time series prediction. The results obtained with these methods are presented in Secs. IV and V and contains a discussion and conclusion.

II. FRICTION CHARACTERIZATION

Frictional behavior may be divided into two main regimes: Pre-sliding and gross sliding. If we consider two objects in (frictional) contact, then there will always be a displacement resulting from an applied (tangential) force, unless the contact is infinitely stiff. The same thing can be said of any elasto-plastic, solid object that is subject to a force. Now, below a certain force threshold, if the force is held constant, the displacement will likewise remain constant (except perhaps for creeping motion). When the force is decreased to zero, not all displacement will be recovered, i.e., that there will, in general, be a residual displacement. This is the “pre-sliding” regime, in which, although there is relative motion, there are still points of unbroken contact and points of micro-slip on the two surfaces of the objects resulting in hysteresis of the force in the displacement that marks the frictional behavior in that regime. Above that force threshold, the system will be critically stable, displacement will not remain constant for a constant applied force: The object will suddenly accelerate; all connections are broken, and we have true or gross “sliding.” The term “friction force” is usually taken to mean “the resistance to the motion during true (or gross) sliding;” it usually has its maximum value at the commencement of motion (=static friction) and usually decreases with increasing relative velocity (=dynamic or kinetic friction). It has been observed and shown that the force is predominantly a function of the displacement in the pre-sliding regime showing quasi rate-independent hysteresis with nonlocal memory,² and predominantly a function of the velocity (and its derivatives) in the true sliding regime, showing velocity weakening and lag behavior.¹ The borderline between the two regimes is obviously the “pre-sliding distance” and/or the “breakaway force” (=static friction force) threshold, which are not evident to determine (at least exactly) owing to many factors.

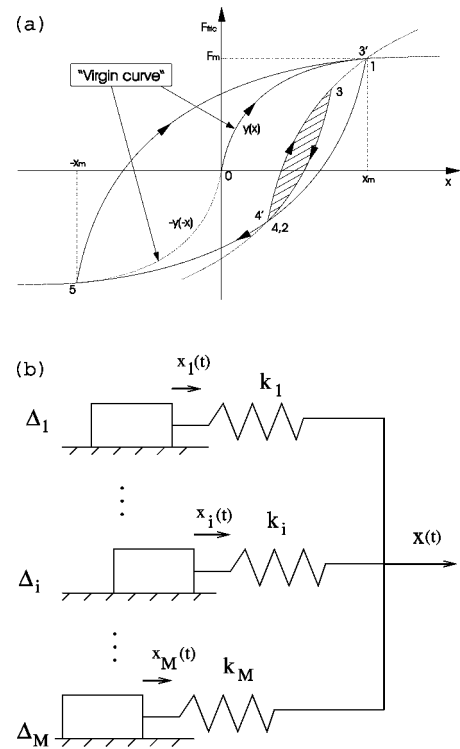


FIG. 1. (a) Hysteresis function with nonlocal memory. (b) Maxwell slip model representation.

A. Friction model structures

Although this paper deals exclusively with pre-sliding friction identification, it may be instructive to give an overview of a representative state-of-the-art friction model structure.

Referring to Refs. 3 and 4, friction force dynamics can be described, with the help of a state variable z representing average asperity deflection that satisfies a first order nonlinear d.e., as follows:

$$\frac{dz}{dt} = v \left(1 - \operatorname{sgn} \left(\frac{F_h(z)}{s(v)} \right) \left| \frac{F_h(z)}{s(v)} \right|^n \right), \quad (1)$$

$$F_f = F_h(z) + \sigma_1 \frac{dz}{dt} + \sigma_2 v. \quad (2)$$

Here, F_f is the friction force, which is seen to be composed of (i) a function $F_h(z)$ that, for vanishingly small relative velocity v , corresponds to hysteresis function with nonlocal memory (see further below); and for steady-state sliding, becomes equal to the Stribeck (or velocity weakening) behavior $s(v)$, and (ii) a viscous part in the state and in the velocity, characterized by the parameters σ_1 and σ_2 , respectively. For a more detailed exposure of friction modeling, the reader is referred to Ref. 5.

During pre-sliding, the behavior will be dominated by that of F_h which will then tend to a hysteresis function in the relative displacement x . Since this has a direct bearing on the type of friction dealt with in the rest of this paper, we give here a more detailed sketch of pre-sliding hysteresis. Referring to Fig. 1(a), in a “virgin” contact, the motion starts at point (0,0) along $y(x)$, termed the virgin curve, which char-

acterizes the hysteresis function. If the motion reverses at point 1, it will follow the path (1-2-5), which is $y(x_m) - 2y((x - x_m)/2)$. If the motion should reverse again at point 2, it would follow (2-3-3'); if now there be a reversal at point 3, the trajectory will follow (3-4). When point 4 (or 2) is reached, the shaded inner hysteresis loop is closed and the motion proceeds along (1-2-5) as before. Points 2 and 3 of the closed loop are wiped out of the memory of the system, but all other reversal points must be kept in the memory (hence the term *nonlocal memory*). This type of behavior may be modeled using either memory stacks,³ which is exact but difficult to implement, or by the easier Maxwell slip model,^{4,6} which is however approximate. The latter is depicted in Fig. 1(b): The hysteresis system is modeled as M massless elasto-slip elements in parallel (the system can thus have M reversal points at most). Each element i ($1 \leq i \leq M$) is characterized by a stiffness k_i , element's position $x_i(t)$, spring deflection $\delta_i(t) = x(t) - x_i(t)$ and maximum spring deflection Δ_i (before the element i starts to slip); the input displacement $x(t)$ is common to all elements. The total hysteretic force F_h is given by the summation of all elements' spring forces, $F_i(t)$:

$$F_h(t) = \underbrace{k_1 \cdot \Delta_1 \cdot \bar{\delta}_1(t)}_{F_1(t)} + \dots + \underbrace{k_M \cdot \Delta_M \cdot \bar{\delta}_M(t)}_{F_M(t)}, \quad (3)$$

if $|\delta_i(t)| < \Delta_i$ (stick) then

$$\begin{aligned} \bar{\delta}_i(t) &= \frac{\delta_i(t)}{\Delta_i}, \\ x_i(t+1) &= x_i(t), \end{aligned} \quad (4)$$

else (slip)

$$\begin{aligned} \bar{\delta}_i(t) &= \text{sgn}(\delta_i), \\ x_i(t+1) &= x(t) - \text{sgn}(\bar{\delta}_i) \cdot \Delta_i. \end{aligned} \quad (5)$$

Finally, the transition from pre-sliding to gross sliding, and back, can be quite a complex process and is thus outside the scope of this paper.

B. Acquisition of friction data

In order to obtain experimental friction data, be it for identification or model testing, several dedicated test setups have been built, notably (i) dry sliding friction test set-up, (ii) pre-rolling friction setup (short stroke and long stroke), and (iii) an ultra accurate new test setup for investigating pre-sliding and transition to gross sliding. The underlying principle of measurement, being the same for each setup, we shall outline the last one here. Figures 2 and 3 show a schematic and a picture of the developed tribometer, which is constructed based on design rules described in Ref. 7. The objective is to measure the dynamic friction force in the sliding interface between the friction block (5) and the actuator block (6). The instrument can be roughly divided into three parts: An actuator part (containing components 6,7,9-10), a friction part (components 2-5 and 12), and a loading part (components 13-16). The different parts are decoupled as much as possible: The actuation part and friction part are

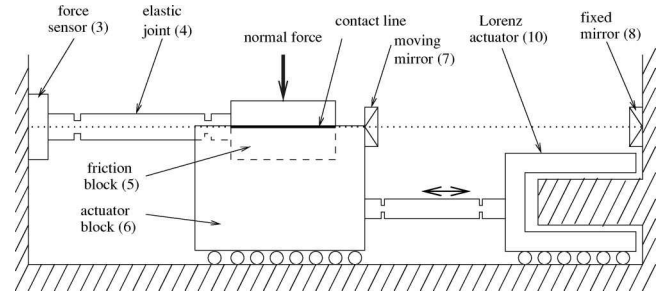


FIG. 2. Schematic of the developed tribometer. The friction sensor and displacement sensor are placed in line with the contact lines.

only coupled by the friction interface under investigation and the loading part and friction part are completely separated by the use of an air-bearing ensuring that all the tangential forces are directed to the force sensor.

The actuator part consists of three main components: A Lorenz actuator (10), a moving block (6) and a displacement sensor. The Lorenz actuator linearly actuates, by means of a stinger (9), the moving block which makes contact with the friction block. The displacement of the moving block is measured using a Renishaw laser interferometer which measures the distance between a mirror fixed to the moving block (7) and a mirror fixed to the frame (8). The Lorenz actuator is current-driven and by feeding back the position signal into a controller a desired displacement can be obtained; therefore, the setup is capable of imposing forces or desired displacement trajectories.

The friction part, shown in Fig. 3, is the critical part of the tribometer and has two important components: The friction block (5), on which the friction force acts, and a force cell (3), which measures the friction force. The elastic joint (4) between the friction block and the force cell consists of two pairs of elastic hinges. The purpose of the hinge is to set off small vertical, lateral and rotational alignment errors of the friction block (all perpendicular to the direction of displacement) caused by positioning the friction block on the moving block. The principle of minimal compliance (or maximum stiffness)⁷ of the friction part in the longitudinal direction is crucial for this setup. A small compliance in the friction part, results in a negligible displacement of the friction block. The standstill of the friction block has two crucial advantages: (i) The relative displacement between both

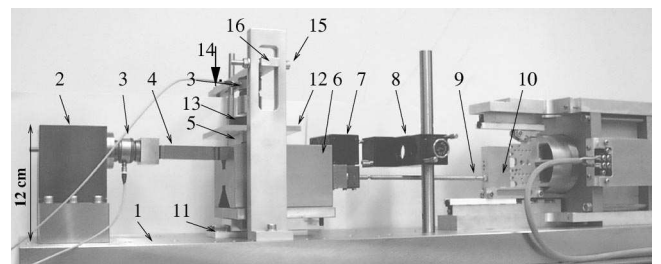


FIG. 3. General picture of the developed tribometer with the following components: (1) frame, (2) support, (3) force sensor, (4) elastic joint, (5) friction block, (6) actuator block, (7) moving mirror, (8) fixed mirror, (9) thin rod, (10) Lorenz actuator, (11) linear guideway, (12) Plexiglas, (13) air-bearing, (14) load, (15) rotation point, (16) lever.

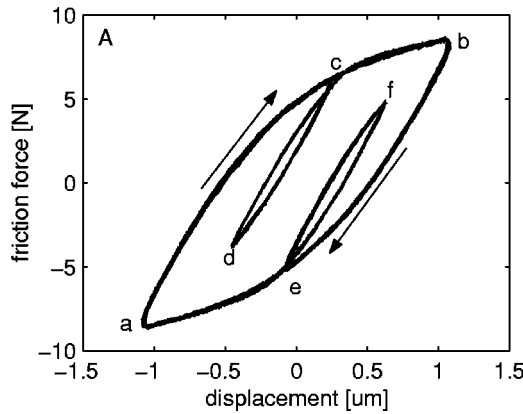


FIG. 4. Typical friction force as a function of the displacement in pre-sliding. The given desired displacement signal as a function of time is indicated by the letters a, b, c, Similar results are obtained if the position signal is applied 50 times faster.

blocks equals the absolute displacement of the moving block, and (ii) the measured force equals the friction force by the lack of inertial force. The compliance of the friction part is given as the sum of the compliances of the support, the elastic joint and the force cell. The overall compliance of the friction part is dominated by the elastic joint. For a peak-to-peak friction force equal to 10 N the displacement of the friction block equals 0.12 μm (as verified experimentally).

The compliance of the actuator part of the force chain is not as critical as for the friction part, but a low compliance for this part will facilitate the design of a good controller to impose the different displacement trajectories between the two blocks. For more detail about design and commissioning of this tribometer, the reader is referred to Ref. 8. Note that, although the setup has been very carefully designed and realized, not any arbitrary motion trajectory may be applied to the actuator block (slider), since that depends on the controllability of motion in the presence of friction. However, periodic motion can always be imposed.

Experimental results for pre-sliding friction force measured for a given displacement signal are shown in Fig. 4.

Figure 5 shows the experimentally measured time series used for modeling in the following sections. The force signal has a slow trend due to leakage of charge from the (piezo-crystal) force sensor. This is almost unavoidable when quasi-

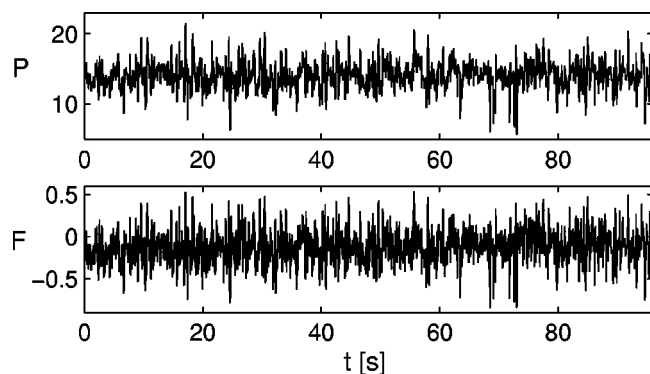


FIG. 5. Experimental time series of position $P(t)$ and friction force $F(t)$ vs time.

static force measurements are carried out. On the other hand, in most identification measurements on real machines, one will be confronted with the same problem. Thus, instead of attempting to “perfect” the measurement, we decided to consider this trend as part of the identification problem.

III. IDENTIFICATION METHODS

The problem of system identification in its most general form is the construction of a function which maps the inputs of the system to the outputs. For friction modeling the input is the (desired) displacement $P(t)$ and the output is the force $F(t)$ (to be applied). To model such an input–output relation one may either derive physics-based models (using first principles) or apply some black-box modeling technique (without referring to the underlying physical system). In the following we shall use both approaches.

The physics-based models used are all based on the Maxwell-slip model presented in Sec. II A and include linear regression, dynamic linear regression and nonlinear regression approaches that will be introduced in more detail in Secs. III A, III B, and III C, respectively.

The black-box models employed are polynomial NARX models, feedforward neural networks, local (nonparametric) models and dynamical networks. Except for the latter these methods solve a (standard) regression where for some given data set $\{(x_i, y_i)\}$, $i = 1, \dots, N$, with vector inputs $x_i \in \mathbf{R}^d$ and scalar outputs $y_i \in \mathbf{R}$ a mapping f from the inputs to the outputs is found. Usually this mapping depends on some finite number M of parameters θ_j

$$\hat{y}_i = f(x_i | \theta_1, \dots, \theta_M), \tag{6}$$

which are unknown and which have to be determined during the modeling procedure. There are two criteria for the choice of the function f and optimal parameters, accuracy and simplicity. The former guarantees that function outputs \hat{y}_i deviate little from the true values y_i . The latter is a precaution to avoid overfitting phenomena for short and noisy data samples. In practice balance between the two criteria is found by using an optimization procedure which depends on the application at hand.

The regression problem can be easily extended to modeling dynamics from a time series $\{u_t\}$, $t = 1, \dots, T$, by setting

$$y_i = u_t \text{ and } x_i = (u_{t-1}, \dots, u_{t-d}), \tag{7}$$

with $i = (d + 1), \dots, T$ and the embedding dimension d . For continuous systems an appropriate time delay τ may be taken into account: $x = (u_{t-\tau}, u_{t-2\tau}, \dots)$. If the actual time series value u_t does not only depend on past values u_{t-1}, \dots, u_{t-d} but also on values of another time series $\{w_t\}$, these values can be included into the regression procedure by expanding the input vector

$$x_i = (u_{t-1}, \dots, u_{t-d}, w_t, w_{t-1}, \dots). \tag{8}$$

Proper selection of suitable model functions and robust estimation of parameters are crucial for successful (black-box) modeling and will be discussed in more detail for each method separately.

The experimental time series used for identification is shown in Fig. 5. It consists of 24 000 samples measured with

a sampling time of $t_S=4$ ms and a resolution of 14 bits. The first 16 000 data points are used for training and validation of the derived models and the last 8000 samples provide the test set for independent evaluation (cross validation). As measure of performance we use the normalized mean-square-error (MSE) defined by

$$\text{MSE}(\hat{y}) = \frac{100}{N\sigma_y^2} \sum_{i=1}^N (\hat{y}_i - y_i)^2, \quad (9)$$

where y is the output (in our case the friction force F), σ_y^2 is its variance and the caret denotes an estimated quantity. If the mean of the output signal \bar{y} is used as the model (i.e., $\hat{y}_i = \bar{y}$ for all i), the MSE is 100. An MSE of less than 5.0 indicates good agreement while one of less than 1.0 reflects an excellent fit. In addition to the MSE the normalized maximum errors on the test set are determined for the different models:

$$\text{MAX}(\hat{y}) = \frac{1}{\sigma_y} \max_{i=1,N} (|\hat{y}_i - y_i|). \quad (10)$$

A. The linear regression (LR) approach

Three different physics-based identification approaches, designated as linear regression (LR), dynamic linear regression (DLR), and nonlinear regression (NLR) are postulated and assessed that are all driven by the (scalar) position signal. All these models use the Maxwell slip model and are, therefore, capable of accounting for the hysteresis, with non-local memory, relationship between the displacement (excitation) and pre-sliding friction force (response). Among their main advantages is simplicity and physical interpretation. All three approaches are based upon minimization of a quadratic cost function of the form

$$\mathcal{J} \triangleq \sum_{t=1}^N e^2(t), \quad (11)$$

with N designating the number of samples used and $e(t)$ the error consisting of the difference between the measured force $F(t)$ and the model provided force $F_M(t)$:

$$e(t) \triangleq F(t) - F_M(t). \quad (12)$$

A more detailed analysis and description of the proposed approaches may be found in Refs. 9 and 10.

We start with the linear regression (LR) approach, where the pre-sliding friction force is approximated by a LR(M) model from within the model class LR(M, \mathbf{k}):

$$F(t) = \sum_{i=1}^M k_i \cdot \Delta_i \cdot \bar{\delta}_i(t) + e(t). \quad (13)$$

This structure is obtained directly from the Maxwell slip model (see Sec. II A above) with M superimposed elements. $\bar{\delta}_i(t)$ ($i=1, \dots, M$) designates the normalized (with respect to Δ_i) i th spring deflection and $e(t)$ the model error (assumed to be a zero-mean uncorrelated random sequence with variance σ_e^2). In this model structure the Δ_i 's are preassigned as equally spaced within the range of the asperity deformation;⁴ an apparently gross approximation. The model parameters to

be estimated thus are the stiffnesses k_i ($i=1, \dots, M$), collected into the vector \mathbf{k} . As the model structure is linear in the parameters, minimization of the cost function \mathcal{J} leads to a linear regression type estimator for \mathbf{k} .

Model order selection (that is selecting the number M of superimposed elements) is based upon the successive estimation (training) of models for increasing M and evaluation of the MSE criterion.

B. The dynamic linear regression (DLR) approach

In this approach the pre-sliding friction force is approximated via a DLR(M, n) model.^{9,10} This is also based upon the Maxwell slip model with M superimposed elements, but the normalized spring deflections $\bar{\delta}_i(t)$, collected into the vector $\bar{\boldsymbol{\delta}}(t) \triangleq [\bar{\delta}_1(t) \dots \bar{\delta}_M(t)]^T$, are now driven through a band of finite impulse response (FIR) filters (each one of order n) to produce the pre-sliding friction force. The DLR model class is thus of the form DLR($M, n, \bar{\boldsymbol{\theta}}$):

$$F(t) = \sum_{j=0}^n \boldsymbol{\theta}_j^T \cdot \bar{\boldsymbol{\delta}}(t-j) + e(t), \quad (14)$$

with $\boldsymbol{\theta}_j$ ($j=0, \dots, n$) designating the FIR filter band's j th coefficient vector, $\bar{\boldsymbol{\theta}}$ the composite FIR coefficient vector, and $e(t)$ the model error. Note that, like in the previous case, the approximation of preassigned (equally spaced) Δ_i 's is utilized.

The main advantage of this model structure is in the extra dynamics and complexity due to the FIR filter band, which may account for discrepancies between the previous LR model structure and the actual pre-sliding friction dynamics. Also notice that as the DLR(M, n) model structure is still linear in the parameters, minimization of the cost function \mathcal{J} still leads to a linear regression type estimator for $\bar{\boldsymbol{\theta}}$.

Here model order selection is based upon the estimation (training) of models corresponding to various values of n for any given M . The final model is selected following consideration of various values of M .

C. The nonlinear regression (NLR) approach

The nonlinear regression (NLR) approach^{9,10} shares the basic Maxwell slip representation of the pre-sliding friction force, its important difference from the LR approach, however, being that the thresholds (Δ_i 's) are no longer pre-assigned, but, instead, estimated along with the stiffnesses (k_i 's). The NLR model class is thus of the form NLR($M, \mathbf{k}, \boldsymbol{\Delta}$):

$$F(t) = \sum_{i=1}^M k_i \cdot \Delta_i \cdot \bar{\delta}_i(t) + e(t), \quad (15)$$

with M designating the number of superimposed elements, \mathbf{k} the stiffness vector, and $\boldsymbol{\Delta}$ the threshold vector.

The NLR approach thus corresponds to complete identification of the Maxwell slip model through elimination of the preassigned threshold approximation. This is naturally expected to lead to increased accuracy, yet, the price paid for it is that the linearity in the model parameters is now lost. As a

consequence, minimization of a quadratic function of the model error now leads to a nonlinear regression type estimator for the parameter vectors \mathbf{k} and $\mathbf{\Delta}$.

NLR model parameter estimation is thus based upon a postulated two-phase, hybrid, optimization scheme. The first (*pre-optimization*) phase utilizes probabilistic genetic algorithm (GA) based optimization¹¹ in order to explore large areas of the parameter space and locate regions where global or local minima may exist. The second (*fine-optimization*) phase utilizes the Nelder–Mead Downhill Simplex algorithm¹² (p. 289) for locating the exact global or local minima within the previously obtained regions. The Nelder–Mead algorithm makes use of cost function evaluations but not of derivatives, which are not defined everywhere as the cost function is nonsmooth in the parameter space.

This two-phase scheme is capable of locating (with high probability) the true global minimum of the cost function and circumventing problems associated with local minima, which are quite common in this case.⁹

Model order selection is done in the same way as with LR(M) models.

D. NARX models

We shall now turn to black-box models starting with NARX models which are a nonlinear extensions of the well known ARX (AutoRegressive with eXogenous input) models.^{13–16} In the regression formalism the model form is described by some arbitrary nonlinear function $\hat{y}_i = f(x_i | \theta)$ where the input vector x_i includes past values of the time series $\{u_i\}$ and values of the exogenous input $\{w_i\}$.

How the parameters $\theta = (\theta_1, \dots, \theta_M)$ enter into the model function is not defined. Practical aspects advise to choose f as a linear sum of so-called *basis functions* ϕ_j

$$f(x_i | \theta) = \sum_{j=1}^M \theta_j \phi_j(x_i). \tag{16}$$

In this way the parameters θ_j contribute only quadratically to the sum of squared errors

$$V(\theta) = \sum_{i=1}^N (y_i - f(x_i | \theta))^2, \tag{17}$$

reducing the least squares fit to a simple convex minimization problem with a well defined solution.

Theoretically any function type can serve as basis function. Nevertheless it should meet two requirements. First of all the basis functions have to be sufficiently flexible and complex. A combination of them should be able to approximate the potentially complex relationship between the inputs x_i and the outputs y_i . Obviously for nonlinear regression this implies that the basis functions have to be also nonlinear in some way.

The second requirement is contrary to the first and states that the basis functions should be as simple as possible. This is partly a matter of practicability as simple basis functions reduce computational efforts. However, it also refines the control on the complexity of the model. Every basis function ϕ_j with a corresponding nonzero coefficient θ_j increases the complexity of the model by a small amount. Choosing an

appropriate number M of basis functions effectively determines the trade-off between accuracy and complexity of the model (see the forward orthogonal regression¹³).

In our models we have used *monomials* for basis functions

$$\phi(x) = \prod_{i=1}^d x_i^{p_i} \text{ for the input } x \in \mathbf{R}^d, \tag{18}$$

with the maximum degree $p = \sum_i p_i$. Other popular choices are *radial basis functions*, *rational functions*, and *wavelets*.

E. Neural networks

Artificial neural networks have come into recent prominence because of their ability to learn input–output relationships by training on measured data and they appear to show some promise for the system identification problem. The most often used forms are the multi-layer perceptron (MLP) and radial basis function (RBF), the model used here will be the MLP. In order to form a model with a neural network it is necessary to specify the form of the inputs and outputs; in this case, the NARX functional form will be used,

$$y_i = f(y_{i-1}, \dots, y_{i-n_y}; x_{i-1}, \dots, x_{i-n_x}), \tag{19}$$

where y_i denotes the time series to be predicted and x_i is some additional input time series. In the case of the MLP with a linear output neuron, the appropriate structure for a single-input–single-output (SISO) system is

$$y_i = s + \sum_{j=1}^{n_h} w_j \tanh \left(\sum_{k=1}^{n_y} v_{jk} y_{i-k} + \sum_{m=0}^{n_x-1} u_{jm} x_{i-m} + b_j \right), \tag{20}$$

where n_x and n_y denote the number of input and output lags, respectively, and the w 's, u 's, and v 's are the weights, or if a nonlinear output neuron is used,

$$y_i = \tanh \left[s + \sum_{j=1}^{n_h} w_j \tanh \left(\sum_{k=1}^{n_y} v_{jk} y_{i-k} + \sum_{m=0}^{n_x-1} u_{jm} x_{i-m} + b_j \right) \right]. \tag{21}$$

As a linear output neuron is often adopted for regression problems, the structure (20) is used here.

Some of the earliest examples of the use of neural networks for system identification and modeling are the work of Chu *et al.*¹⁷ and Narendra and Parthasarathy.¹⁸ Masri *et al.* are amongst the first structural dynamicists to exploit the techniques.¹⁹ The latter work is interesting because it demonstrates “dynamic neurons” which are said to increase the utility of the MLP structure for modeling dynamical systems. One of the most comprehensive programs of work to date is that of Billings and co-workers starting with Ref. 20 for the MLP structure and Ref. 21 for the RBF.

The neural networks described further were trained using the scaled conjugate gradient algorithm²² with a 3 layer structure. The strategy for optimizing the neural network structure involves using a training set to establish weights and a validation set to fix the optimum hidden layer number,

initial conditions, stopping time and number of lags. The program is summarized by the following pseudo-code:

```

for different input lags=0 to 6
  (where 0 indicates the current input) {
for different output lags=0 to 6
  (where 0 indicates that no output lags was used in the
network) {
for number of hidden layer neurons=1 to 10 {
for different random initial conditions=1 to 20 {
  train network on training data
  evaluate on validation data
  terminate training at minimum on validation set}}}}}.
    
```

For simplicity the maximum number of input-lags n_x was assumed equal to the maximum number of input-lags n_y . During training, time-varying learning coefficients and momenta were used. These were initially set high to allow potential movement between local minima and then annealed to allow fine-tuning. The networks updated the weights at each presentation of an input–output pair, i.e., an epoch of unity was used.

F. Local models

In contrast to the *global* models discussed so far, *local* models do not use any training data until queried with some point \mathbf{x} . A small neighborhood of \mathbf{x} is located in the training set and a simple model using only the training points lying in this neighborhood is constructed.²³

The most common choice for the neighborhood is to locate the k nearest neighbors $\mathbf{x}_{nn_1}, \dots, \mathbf{x}_{nn_k}$ of \mathbf{x} (*fixed mass*), i.e., the k points in the training set which have the smallest distance to the query point according to some arbitrary metric $\|\cdot\|$ (usually Euclidean). To find the nearest neighbors we use a fast algorithm called *ATRIA*, which relies on a binary search tree built in a preprocessing stage.²⁴

The model used in the neighborhood of the query point is usually fairly simple. A *locally constant* model computes a weighted average of the images of the nearest neighbors

$$\hat{f}(\mathbf{x}) = \frac{\sum_{i=1}^k w_i y_{nn_i}}{\sum_{i=1}^k w_i} \tag{22}$$

Besides their speed of computation, locally constant models are very robust, as their predictions always remain in the data range given by the nearest neighbors. The weights w_i are usually drawn from a monotonically decreasing weight function, so that the influence of the furthest nearest neighbors is decreased. Otherwise, the model output becomes discontinuous, as shifting the query point \mathbf{x} results in points suddenly entering or leaving the neighborhood.

A *locally linear* model fits a linear function

$$\hat{f}(\mathbf{x}) = \mathbf{a}^T \cdot \mathbf{x} + a_0 = \tilde{\mathbf{a}}^T \cdot \tilde{\mathbf{x}}, \tag{23}$$

(with $\tilde{\mathbf{a}} = [\mathbf{a}; a_0]$ and $\tilde{\mathbf{x}} = [\mathbf{x}; 1]$) in the neighborhood of the query point by minimizing the weighted sum of squared errors

$$V(\mathbf{a}, a_0) = \sum_{i=1}^k w_i^2 (y_{nn_i} - \tilde{\mathbf{a}}^T \cdot \tilde{\mathbf{x}})^2. \tag{24}$$

The solution for $\tilde{\mathbf{a}}$ is given by

$$\tilde{\mathbf{a}} = (\mathbf{X}_W^T \mathbf{X}_W)^{-1} \mathbf{X}_W^T \cdot \mathbf{y}_W = \mathbf{X}_W^\dagger \cdot \mathbf{y}_W, \tag{25}$$

where $\mathbf{X}_W = \mathbf{W} \cdot \mathbf{X}$, $\mathbf{y}_W = \mathbf{W} \cdot \mathbf{y}$, $\mathbf{X} = [\tilde{\mathbf{x}}_{nn_1}^T, \dots, \tilde{\mathbf{x}}_{nn_k}^T]^T$, $\mathbf{y} = [y_{nn_1}, \dots, y_{nn_k}]^T$ and $\mathbf{W} = \text{diag}([w_1, \dots, w_k])$.²⁵ The term \mathbf{X}_W^\dagger denotes the pseudoinverse of \mathbf{X}_W , which can be calculated using the singular value decomposition $\mathbf{X}_W = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T$, where $\mathbf{S} = \text{diag}([\sigma_1, \dots, \sigma_k])$ with the singular values σ_i . The pseudoinverse is then given by $\mathbf{X}_W^\dagger = \mathbf{V} \cdot \mathbf{S}^{-1} \cdot \mathbf{U}^T$.

Locally linear models give usually more accurate estimations than locally constant ones, but they need an additional regularization method to secure stability. One popular approach for regularization is the *truncated principal component regression* (TPCR). During the calculation of the pseudoinverse \mathbf{X}_W^\dagger small singular values in the diagonal matrix \mathbf{S} are set to zero. This can be further improved by *soft thresholding*, where singular values lying in a specific interval $[s_1, s_2]$ are smoothly weighted down to zero.²⁵

For locally linear models, four types of parameters must be chosen: The number of nearest neighbors k and the metric used to locate these, the weighting function for the weights w_i , and the regularization parameters s_1, s_2 . Additionally, one must also find good values for the embedding parameters. Instead of the usually chosen delay embedding with the two parameters dimension and delay, a nonuniform embedding is used, which allows varying delays between the components of the input vector.

For finding good parameter values, one can use a training procedure like *cross validation*. Here, the data set is split into a training and a test set which are used for training and validating the model, respectively. Local models allow an “extreme” form of this procedure, the *leave-one-out cross validation* (LOOCV), where the test set is reduced to one single point. Such an implementation is possible because local models are lazy learners, which delay any model calculations until they are queried. Of course, one has to repeat this validation procedure with enough different test points to get a good estimation of the actual model error.

A genetic algorithm which minimizes the LOOCV error is used for optimizing the delays of the embedding and the number of nearest neighbors.

For the other parameters a simple type of cyclic optimization is used, where all parameters are successively optimized with a semiglobal line search.²⁵ Although these optimization procedures do not necessarily lead to the global minimum in parameter space, they are usually able to improve prediction accuracy compared to manually chosen parameters.

G. Dynamical networks

All methods introduced in the previous subsections require a delay vector as an input. An alternative are dynamical networks (DN)^{26–28} that are trained to generate a specific output signal when driven by the corresponding input signal. The nonlinear functional relation between input and output is

here an emergent feature of generalized synchronization²⁶ between the driving process (input signal) and the dynamical network. As a special case of recurrent neural networks (RNNs) this class of systems has some internal memory. Thus they differ from memoryless models (like NARX, local models or feed forward neural networks) that operate on vector valued input and provide (static) functions between input–output pairs (x_i, y_i) . In order to use the latter for prediction purposes one has to introduce an external memory by the embedding technique described above. In contrast dynamical networks do not depend on this technique. They preserve information from the past in the states of their internal elements and can use it for dynamic tasks like prediction or classification.

Dynamical networks are a special type of RNN. However, the internal elements of the network are not restricted to neuron-like systems but may consist of any type of (stable) dynamical system.²⁶

The local dynamics of the network used here for friction modeling consists of N dynamical elements

$$x_i(t) = \alpha_i x_i(t-1) + (1 - |\alpha_i|) \sigma(I_i(t)), \quad (26)$$

with input

$$I_i(t) = w_{i0} + \sum_{j \neq i} w_{ij} x_j(t-1) + v_i u(t), \quad (27)$$

where $x_i(t)$ is the state of the i th element at time step t , w_{ij} is the connection strength of the output from the j th element to the i th element, w_{i0} the bias of the i th element, v_i the connection strength of the external input signal $u(t)$ to the i th element and $\sigma(\cdot)$ a piecewise linear sigmoid activation function

$$\sigma(z) = \begin{cases} 1 & x > 1 \\ x & -1 \leq x \leq 1 \\ -1 & x < -1 \end{cases} \quad (28)$$

The parameter $\alpha_i \in (-1.0, 1.0)$ determines the degree by which the past state of the i th element influences the present one. Higher absolute values result in a longer memory whereas for values near zero the information about the past state degrades more rapidly.

There are many ways to define the topology of dynamical networks. For our purpose we used a constructive approach consisting of two phases. In the growing phase we started with an empty network. Each iteration step we added a new element with at most two connections to the already existing ones. The procedure was stopped when ten elements were present in the network. This growing phase was followed by a restructuring phase. Each iteration step a few connections between the elements were randomly added, cut or changed and randomly chosen parameters $\alpha_i, w_{ij}, w_{i0}, v_i$ were optimized by a nonlinear minimization algorithm. If the new network structure rendered a better performance on the validation data set it was chosen to replace the old one. The restructuring procedure was stopped when no more significant improvements could be achieved.

The output of the network was represented by a linear superposition of the elemental states

$$\hat{y}(t) = \sum_{i=1}^N \theta_i x_i(t) + \theta_0,$$

where the weights $\theta_i, i=0, \dots, N$ were computed on the training data set with the usual least squares method as already described for NARX models.

IV. SIMULATION RESULTS

The goal of our modeling is to estimate the (current) force value $F(t)$ by using values of the displacement signal $P(t), P(t-1), P(t-2), \dots$. Typical results obtained by the aforementioned methods are shown in Fig. 6.

Figure 6(a) shows the last 8000 samples of the given data set which constitute the *test set*. The deviations $e(t) = \hat{F}(t) - F(t)$ of the predicted force \hat{F} and from the true measured force F are plotted for the different models in Figs. 6(b)–(6m) where also the mean-square-error or MSE (9) and the maximum error MAX (10) on the test set are given.

A. Linear regression

In contrast to black-box models, the physics-based models are not prepared to cope directly with the offset present in the displacement and friction signals due to sensor calibration. The offset compensation strategy followed within the LR approach (also within the DLR and NLR approaches), includes:

- (i) Sample mean subtraction of the displacement signal, assuming that the displacement sample mean is equal to the experimental displacement offset;
- (ii) incorporation of an extra parameter, denoted as b , into the physics-based models in order to account for the experimental friction offset. Thus the LR model is a member of the class $LR(M, k, b)$ and becomes

$$F(t) = \sum_{k=1}^M k_i \cdot \Delta_i \cdot \bar{\delta}_i(t) + b + e(t). \quad (29)$$

Model estimation and order selection are based upon two disjoint data sets, designated as *training set* and *validation set*. These data sets are derived from the first 16 000 data points. The first set is used for parameter estimation. The final model is selected by considering the mean square error (MSE) of the candidate models within both sets.

Model order selection results are presented in Fig. 7, which depicts the MSE as a function of the model order. Figure 7(a) refers to the training set and Fig. 7(b) to the validation set.

As Fig. 7(a) indicates the MSE criterion is, for the LR sequence of models, decreasing as the order increases. The behavior of the LR model in the second set [Fig. 7(b)] is not as smooth. From the LR sequence of models the LR(6) model is finally selected.

The performance assessment of the selected LR(6) model within the test set is presented in Fig. 6(b) in which the MSE criterion along with the corresponding maximum error are provided. These results indicate that the LR(6)-based friction simulation is in good overall agreement with its measured counterpart.

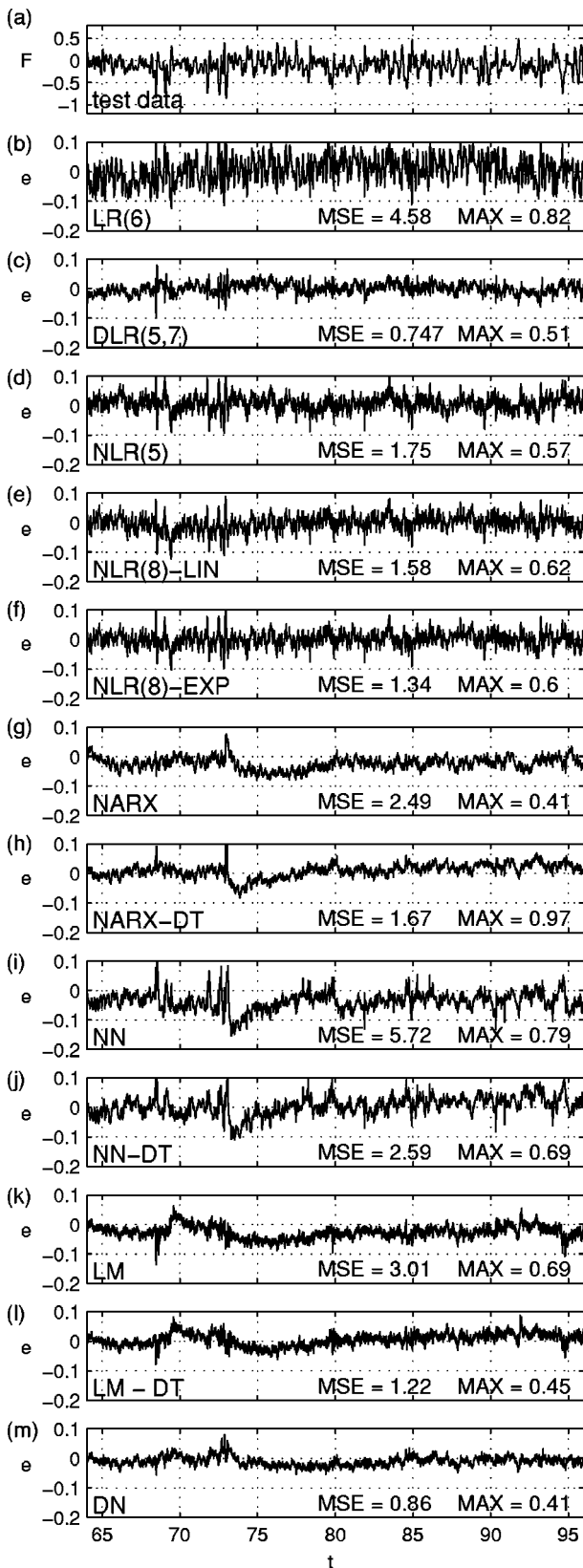
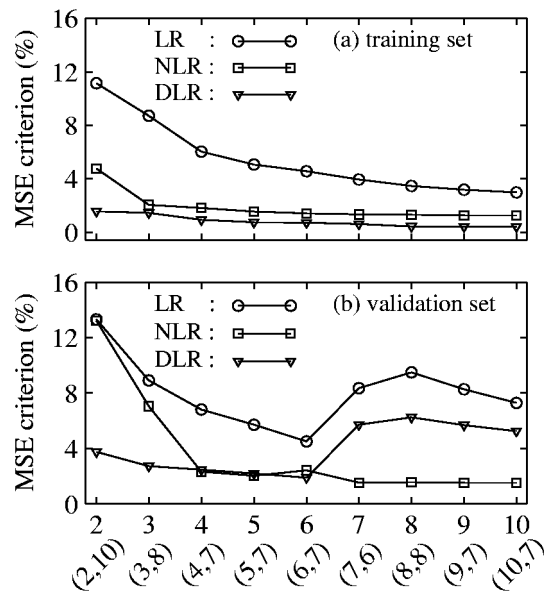


FIG. 6. Comparison of identification methods. Shown are the deviations $e(t) = \hat{F}(t) - F(t)$ of the predicted force \hat{F} from the true measured force F for the test time series shown in (a). MSE and MAX are the normalized mean-square-error (9) and the normalized maximum error (10), respectively, obtained with the corresponding method.



LR, NLR model order M and DLR model order (M,n)

FIG. 7. MSE criterion versus model order: (a) Training set; (b) validation set.

B. Dynamic linear regression

The DLR model order selection results are also presented in Fig. 7. The behavior of the DLR sequence of models within both sets is analogous to that of the LR model sequence. Nevertheless, the MSE criterion reaches an approximate plateau for $M \geq 5$ [Fig. 7(a)]. The DLR(5,7) model is thus selected, as it is characterized by parametric complexity (number of estimated parameters) that is significantly smaller than its DLR(6,7) counterpart. It is worth noting that the MSE criterion achieved via each DLR model within the first set is quite higher than that achieved within the second set.

The simulation performance assessment of the DLR(5,7) model is presented in Fig. 6(c). The DLR(5,7) model achieves an MSE of 0.747%, indicating excellent fit.

C. Nonlinear regression

The order selection results (Fig. 7) for the NLR sequence of models indicate that the MSE attains an approximate plateau for $M \geq 5$ [Fig. 7(a)]. For $M = 5$ a first minimum is also attained within the validation set [Fig. 7(b)]. The NLR(5) model is thus selected.

The NLR(5) simulation performance assessment results [Fig. 6(d)] show that the NLR(5) model achieves significantly better simulation performance than its LR(6) counterpart, while being characterized by 5 (versus 6) elements.

On the other hand the DLR(5,7) model achieves the lowest MSE criterion [less than half of that of the NLR(5)]. This is, however, achieved at a significantly higher parametric complexity (almost four times that of the NLR(5) model). This indicates that the FIR part of the DLR(5,7) model accounts (apart from the problem of false threshold preassignment) for extra pre-sliding friction dynamics which are not accounted for by the NLR(5) model. Nevertheless, the

NLR(5) model is judged as striking a good balance between accuracy (simulation performance) and parametric complexity (number of estimated parameters).

To better cope with the trend in the friction signal we have also tested the NLR approach with an additional linear at and exponential $a[1 - \exp(-t/b)^c]$ term in the force ansatz (15). The parameters a , b , and c are determined with the other parameters of this model by nonlinear minimization (Nelder–Mead and Trust-Region algorithm). The results are shown as NLR(8)-LIN and NLR(8)-EXP in Figs. 6(e) and 6(f), respectively.

D. NARX models

The first NARX model (Sec. III D) is a polynomial of degree three (including 32 monomials) that has been trained on the raw data with input vector $\mathbf{x}(t) = (P(t), P(t-2), \dots, P(t-18), F(t-2))$ and output $y(t) = F(t)$. The prediction accuracy obtained with this method is shown in Fig. 6(g). To take into account the trend in the force signal we have also applied the NARX-approach to a linearly detrended force signal where a linear function of time was subtracted from the full data set shown in Fig. 5. Identification results obtained with this preprocessing (NARX-DT) are shown in Fig. 6(h) indicating a slight improvement of the MSE.

E. Neural networks

Figures 6(i) and 6(j) illustrate predictions obtained with the neural network approach (Sec. III E), applied to the raw data (NN) and to a linearly detrended force time series (NN-DT). Again the MSE for the preprocessed data is smaller.

F. Local models

Figure 6(k) shows the prediction results for the original time series with a locally linear model (LM) based on 338 nearest neighbors with a constant weight function. The genetic optimization of the nonuniform embedding leads to a 15-dimensional input vector, consisting of nine components from the position and six components from past force values. The MSE is quite large which can be attributed to the fact that the local model is unable to extrapolate the trend in the force time series.

Figure 6(l) shows the prediction results using the linearly detrended force time series (LM-DT), leading to a considerably reduced MSE. The input vector consisted of nine components from the position time series and six components from the past force values. The local model used 300 nearest neighbors with a constant weight function.

G. Dynamical networks

Prediction results using a dynamical network (DN) (Sec. III G) are shown in Fig. 6(m). This approach provided the best results of all black-box methods, even without detrending the force time series. The magnitudes of errors are comparable to those obtained with the best physics-based method [DLR(5,7)]. A possible reason for this common efficiency is

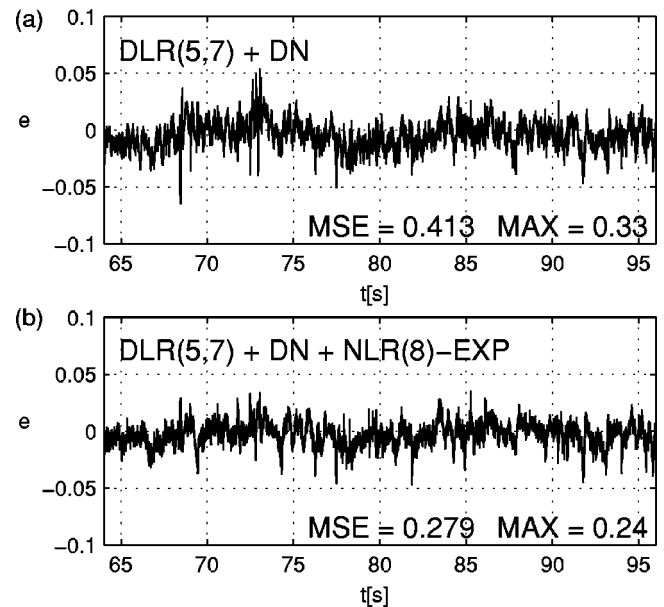


FIG. 8. Prediction errors of ensemble models including (a) DLR(5,7) and DN model; (b) DLR(5,7), DN and NLR(8)-EXP model.

the fact that both methods possess some internal memory structure that may cover the slow trend in the data.

H. Ensemble models

Superposition of output from different models (trained individually) may provide better results than any individual model, because such *ensemble models* typically exhibit better generalization properties and robustness features.²⁹ To investigate whether the best normalized mean squared error $\text{MSE} = 0.747$ [DLR(5,7)] and the minimal normalized maximum error $\text{MAX} = 0.41$ (DN and NARX) could be improved using an ensemble model we combined the best models without detrending. Figure 8(a) shows the remaining prediction error for a superposition of the DLR(5,7) and the DN model and in Fig. 8(b) the outputs of the DLR(5,7), DN, and NLR(8)-EXP model are averaged. The errors ($\text{MSE} = 0.279$ and $\text{MAX} = 0.24$) of the latter three models ensemble are (much) smaller than the minimal errors of the individual errors ($\text{MSE} = 0.747$ and $\text{MAX} = 0.41$). If more of the remaining (worse) models are included the prediction error of the ensemble increases again.

V. DISCUSSION AND CONCLUSIONS

The identification results achieved by the various methods considered in this paper are in general very good and may suffice for control application purposes. There are some notable differences, however, in the performance of each method as reflected in Fig. 6. First, using the performance criteria of Sec. III, we conclude that the black-box as well as the physics-based methods result in good to excellent fits. Within the black-box methods, the dynamical networks score the best; and within the physics ones, the DLR approach. This is achieved, however, at the expense of both programming and calculation intensiveness. Noteworthy is the fact that physics-based models, when we take into account their

relative simplicity, would score the overall best. This is due, in our case, to the fact that the Maxwell slip model used as a basis for those methods accounts very closely to pre-sliding friction force dynamics. Knowledge of a model (structure) representing a physical phenomenon, however approximately is, therefore, a considerable asset in identification. In the absence thereof, black-box techniques prove very powerful tools for nonlinear system identification, not requiring any *a priori* knowledge of the physics, yet producing very good fits. If very exact predictions are required combinations of different types of models within an ensemble turn out to be superior to any of the investigated (individual) models.

In conclusion, this paper has presented an overview of friction force dynamics; its characterization, measurement, and identification. Owing to its presence in most machine elements coupled with its highly nonlinear character, friction force dynamics have to be identified and modeled if effective control of machines is desired. This task is eased significantly by devising accurate measurement setups, constructing suitable models and developing efficient methods of identification, as this paper has clearly shown. An obvious next step is to extend this methodology from pre-sliding friction to gross sliding and transition.

ACKNOWLEDGMENT

The authors acknowledge the Volkswagenstiftung for financial support under Grant No. 1/76938.

- ¹B. Armstrong-Hélouvy, *Control of Machines with Friction* (Kluwer Academic, Dordrecht, 1991).
- ²T. Prajogo, "Experimental study of pre-rolling friction for motion-reversal error compensation on machine tool drive systems," Ph.D. thesis, Dept. Werktuigkunde Katholieke Universiteit Leuven, Leuven, 1999.
- ³J. Swevers, F. Al-Bender, C. Gansseman, and T. Prajogo, "An integrated friction model structure with improved presliding behavior for accurate friction compensation," *IEEE Trans. Autom. Control* **45**, 675 (2000).
- ⁴V. Lampaert, J. Swevers, and F. Al-Bender, "Modification of the Leuven integrated friction model structure," *IEEE Trans. Autom. Control* **47**, 683 (2002).
- ⁵F. Al-Bender, V. Lampaert, and J. Swevers, "Modeling of dry sliding friction dynamics: From heuristic models to physically motivated models and back," *Chaos* **14**, 446–460 (2004).
- ⁶W. Symens, F. Al-Bender, and J. Swevers, and H. Van Brussel, "Dynamic characterization of hysteresis elements in mechanical systems," *Proc. of the American Control Conference* (2002), pp. 4129–4134.
- ⁷H. Nakazawa, *Principles of Precision Engineering* (Oxford University Press, Oxford, 1991).
- ⁸V. Lampaert, F. Al-Bender, and J. Swevers, "Experimental characterization of dry friction at low velocities on a developed tribometer setup for macroscopic measurements," *Tribol. Lett.* **16** (1), 95–105 (2004).
- ⁹D. D. Rizos and S. D. Fassois, in *4th International Symposium, Investigations of Non-Linear Dynamics Effects in Production Systems* (Chemnitz, Germany, 2003).
- ¹⁰D. D. Rizos and S. D. Fassois, "Presliding friction identification based upon the Maxwell Slip model structure," *Chaos* **14**, 431–445 (2004).
- ¹¹S. K. Pal and P. P. Wang, *Genetic Algorithms for Pattern Recognition*, 1st ed. (CRC Press, Florida, 1996).
- ¹²W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Computing* (Cambridge University Press, Cambridge, 1990).
- ¹³S. A. Billings *et al.*, "Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator," *Int. J. Control* **6**, 2157–2189 (1989).
- ¹⁴L. A. Aguirre and S. A. Billings, "Retrieving dynamical invariants from chaotic data using NAR-MAX models," *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **2**, 449–474 (1995).
- ¹⁵S. A. Billings and D. Coca, "Discrete wavelet models for identification and qualitative analysis of chaotic systems," *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **7**, 1263–1284 (1999).
- ¹⁶L. A. Aguirre *et al.*, "Nonlinear multivariable modeling and analysis of sleep apnea time series," *Comput. Biol. Med.* **29**, 207–228 (1999).
- ¹⁷S. R. Chu, R. Shoureshi, and M. Tenorio, "Neural networks for system identification," *IEEE Control Syst. Mag.* **10**, 36–43 (1990).
- ¹⁸K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Netw.* **1**, 4–27 (1990).
- ¹⁹S. F. Masr, A. G. Chassiakos, and T. K. Caughey, "Structure-unknown nonlinear dynamic systems: Identification through neural networks," *Smart Mater. Struct.* **1**, 45–56 (1992).
- ²⁰S. A. Billings, H. B. Jamaluddin, and S. Chen, "Properties of neural networks with applications to modelling nonlinear dynamical systems," *Int. J. Control* **55**, 193–224 (1991).
- ²¹S. Chen, S. A. Billings, C. F. N. Cowan, and P. M. Grant, "Nonlinear systems identification using radial basis functions," *Int. J. Syst. Sci.* **21**, 2513–2539 (1990).
- ²²M. F. Moller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks* **6**, 525–533 (1993).
- ²³C. G. Atkeson *et al.*, "Locally weighted learning," *Artificial Intelligence Review* **11**, 11–73 (1997).
- ²⁴C. Merkwirth *et al.*, "Fast nearest-neighbor searching for nonlinear signal processing," *Phys. Rev. E* **62**, 2089–2097 (2000).
- ²⁵J. McNames, "Innovations in local modeling for time series prediction," Ph.D. thesis, Stanford University, 1999.
- ²⁶U. Parlitz and A. Hornstein, "Dynamical prediction of chaotic time series," *Chaos and Complexity Letters* (to be published).
- ²⁷H. Jaeger, "The echo state approach to analysing and training recurrent neural networks," GMD Report 148, German National Research Center for Information Technology (2001).
- ²⁸W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural Comput.* **14**, 2531–2560 (2002).
- ²⁹T. G. Dietterich, "Ensemble learning," in *The Handbook of Brain Theory and Neural Networks*, 2nd ed., edited by M. A. Arbib (MIT Press, Cambridge, MA, 2002).