A new structure for nonlinear black-box system identification using the Extended Kalman Filter

Karol Bogdanski* and Matthew C Best
Department of Aeronautical and Automotive engineering, Loughborough University, UK

Abstract: A new tool for black-box nonlinear system identification of multi-input multi-output systems is presented in this paper. The new structure extends the conventional, linear state space model into a nonlinear framework, where each parameter is a nonlinear function of inputs or states. The method works iteratively in the time domain using an Extended Kalman Filter. The model retains a state space structure in modal canonical form, which ensures that a minimal number of parameters need to be identified and also produces additional information in terms of system eigenvalues and dominant modes. This structure is completely black-box, requiring no physical understanding of the process for successful identification, and it is possible to easily expand the order and complexity of nonlinearities, whilst ensuring good parameter conditioning. A simple nonlinear example illustrates the method, and identification of a highly nonlinear brake model is also presented. These examples show the method can be applied as a mechanism for model order reduction; it is equally well suited as a tool for nonlinear plant system identification. In both capacities this new method is valuable, particularly as the generation of simplified models for the whole vehicle and its subsystems is an increasingly important aspect of modern vehicle design.

Keywords: Nonlinear System Identification, Black-box, Extended Kalman Filter, State Space, Brake model.

1 INTRODUCTION

System identification is the process of selecting an appropriately accurate model structure and fitting its unknown parameters to obtain a suitable mapping of the available input-output data. Many papers have been published in the last decades on structured *grey-box* parametrisation, where one or more parameters of a perfectly known model are identified to match the data with increasing precision [1], or to produce real-time adaptive models [2]. Many of these employ iterative algorithms based on nonlinear Kalman Filters, where the parameters to be identified are concatenated with, or entirely replace the state vector. Examples can be found in [3] or in the more recent [4], where a full vehicle model is identified from experimental data, via both Extended and Unscented Kalman Filters. The major drawback of such methods is that they require good engineering knowledge and experience of the application, for a precise formulation of the process function.

**Corresponding author: Department of Aeronautical and Automotive engineering, Loughborough University, UK*
email: k.bogdanski@lboro.ac.uk

Less common in the literature and more complex, is the topic of *black-box* identification. This is the difficult task of prescribing a structure which is generic enough to replicate the response of any nonlinear dynamic system, given appropriate parametrisation, solely based on input-output data and no *a-priori* physical knowledge. A wide analysis of linear and nonlinear system identification is available in Soderstrom and Stoica [5] and Ljung [6]. These include model validation techniques and identification by means such as the recursive instrumental variables or the prediction error method. Juditsky et al [7] deeply focus on the mathematical basis of nonlinear black-box identification, while Sjoberg et al [8] give a comprehensive overview of nonlinear black-box methods, from a user's approach. These classic references tend to be mathematically obscure however, and hence difficult for the user to implement.

In recent times, the approach of trying to artificially reproduce the mechanisms of human learning through Artificial Neural Network (ANN) methods, has become increasingly popular. ANN can achieve superb performance, eg in [9], but as with most black-box methods, do not give any insight into the virtually unknown model that has been

identified. A comparison between black-box and grey-box identification in the automotive field can be found in [10], where Savaresi et al have successfully identified magneto-rheological damper models through both a nonlinear semi-physical model and a Nonlinear AutoRegressive eXogenous (NARX) structure. In this paper, the performance achieved by the black-box method outruns the state-of-the-art grey-box semi-physical model, which also proves computationally heavier, despite the low number of parameters. The identification is however control-oriented and no physical system knowledge can be obtained from the NARX model.

More recently, Van Mulders et al [11] have employed polynomial-based black-box nonlinear state-space systems, where the parameters are identified by the least-squares method. This structure consists of a linear state-space model with polynomials added to the process and measurement equations. One major drawback is the resulting large number of parameters, with corresponding concerns over conditioning. It also appears that the linear state-space system is somewhat disconnected from the nonlinear polynomial addition and the two can cancel each other out, with parameter divergence as a consequence. Other papers, such as [12] and [13] take the frequency domain based route and estimate the system transfer function based on step input tests or multi-frequency sinusoidal signals. This method can be effective for system control purposes, such as automotive traction control implementation, but is less well suited to model order reduction applications.

In this paper, a new approach to Kalman Filter – based system identification is presented. A novel nonlinear state space structure is used to identify Multi-Input Multi-Output (MIMO) data, with each parameter of the state space matrices allowed to vary as a nonlinear function over a certain range of inputs or states. It develops the work of [14], using a simpler structure for the nonlinearities and an improved process for applying necessary parameter constraints. What we present is fully black-box identification, solely based on input-output time histories using no a-priori engineering knowledge of the system. The authors have also applied the Unscented Kalman Filter (UKF), developed by Julier et al [15] to this black-box structure, but, although easier to code, the UKF proves to be computationally inefficient due to the large number of parameters identified.

In the next section, the structure of the filter and its implementation will be described in detail, with the introduction of a simple example, to demonstrate the capabilities of the filter. We will then present the black-box identification of a highly nonlinear full vehicle brake model, taken from the automotive industry. This allows further discussion of the filter's structure, while showing its full capabilities for model order reduction applications.

2 METHOD

2.1 Structured EKF for grey-box parametric identification

The well-known Extended Kalman Filter (EKF) was first developed by the Dynamic Analysis Branch of NASA for the Apollo spacecraft real-time navigation system, from the seminal paper published in 1960 by R.E. Kalman [16]. It is here considered in its identifying form for structured, *grey-box* parametrisation, sometimes also referred to as *dual* estimation. Based on prior engineering knowledge of the system, nonlinear plant and sensor models *f* and *h* are defined, both being first-order differential functions of the state vector *x*, measurements *y*, inputs *u* and system parameters θ:

$$\dot{z} = \begin{pmatrix} \dot{x} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} f(x,u,\theta) + \omega \\ 0 \end{pmatrix}$$
$$y = h(x,u,\theta) + \upsilon \tag{1}$$

Unlike in the traditional EKF, the problem now comprises estimation of the extended state vector *z*, where some or all of the parameters θ of the model are concatenated with the true states. These parameters are assumed to be constant in time; hence their derivatives are modeled as zero. As in all Kalman Filtering applications, ω represents the modelling error and υ the measurement error; these are assumed Gaussian and uncorrelated white noise sequences, with zero mean. The design *Q*, *S* and *R* matrices are obtained for the EKF algorithm as the covariance matrices of the error sequences:

$$Q = E(\omega\omega^T), \quad S = E(\omega\upsilon^T), \quad R = E(\upsilon\upsilon^T) \tag{2}$$

S is considered null, as in most research papers on the matter [17] and also based on the analysis of Hodgson and Best [18]. The EKF performs a linearisation at each time step, approximating the nonlinear model and measurement functions *f* and *h* through Jacobian matrices, defined:

$$F = \frac{\partial f(x,u,\theta)}{\partial z}, \; H = \frac{\partial h(x,u,\theta)}{\partial z} \tag{3}$$

The full algorithm, here implemented in the continuous process – discrete measurement form, consists of the following recursive equations:

$$K_k = P_k H_k^T (H_k P_k H_k^T + R_k)^{-1} \tag{4}$$
$$P_k^* = (1 - K_k H_k) P_k \tag{5}$$
$$P_k = P_k^* + T(F_k P_k^* + P_k^* F_k^T + Q) \tag{6}$$
$$\hat{z}_{k+1} = \hat{z}_k + T f_k + K\nu_k \tag{7}$$

Where v_k is the *innovation* vector, defined as the difference between measurement and observation prediction:

$$v_k = y_k - h_k \tag{8}$$

and T is the Euler integration interval, which must be set small relatively to the dynamics of the system. At each time-step iteration, the algorithm updates F and H, and computes an estimation of the state error covariance P.

2.2 Unstructured EKF for black-box system identification through linear interpolation

We seek to produce a tool for *black-box* identification of nonlinear input-output data, with nonlinear process and measurement models which are as generic as possible. However, we must impose some form of structure to ensure a minimal parameter model is produced; otherwise the identifying EKF will tend to drift the parameters to non-sensible values due to poor conditioning in the identification process. We also need a structure that will allow expansion of the parameter set to virtually any nonlinear function, and permits easy switching between higher and lower system orders. We therefore start from a generic continuous process, discrete measurement state-space structure:

$$\begin{aligned} \dot{x}_k &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \end{aligned} \tag{9}$$

A canonical form for A, B, C and D is preferable, to minimise the number of elements to be identified. Of the possible choices, the *control* canonical form has the advantage of displaying the coefficients of the transfer function denominator in the first row of the A matrix; it doesn't however appear to be a good candidate, as its parameters are of different orders of magnitude. Given the identification starts with no a-priori knowledge of the system, the parameters are all nominally initialised to the same value. In the control canonical form, some of those will then need to shift to high magnitudes and some others to much lower orders; this would be a slow process via EKF.

The *modal* canonical form appears to be a more sensible choice, since it has approximately normalised parameters throughout the A, B, C, and D matrices. The structure is *modal* because the poles of the transfer function appear in the diagonal of the A matrix and this gives a further advantage in providing additional information of the identified reduced-order system, in terms of its most relevant modes. A hypothetical third order system in the modal canonical form, with two inputs and two outputs will then be described by:

$$A = \begin{pmatrix} \sigma_1 & \omega_1 & 0 \\ \omega_1 & \sigma_1 & 0 \\ 0 & 0 & \sigma_2 \end{pmatrix} \qquad B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix}$$

$$C = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{pmatrix} \qquad D = \begin{pmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{pmatrix} \tag{10}$$

Where the *2x2* sub-matrix in A represents a complex conjugate pair of the eigenvalues $\sigma_1 \pm j\,\omega_1$ and σ_2 is a real pole. We are allowed to vary the number of states and define eigenvalues in any combination of conjugate pairs and real poles. An automated process could be easily employed to select whether a conjugate pair or two real poles are more appropriate at a certain position of the A matrix. For example, identification of $\omega = 0$ would motivate the reduction of a complex pair to two real poles.

To identify nonlinear input-output data we now impose every non-zero element of the state-space matrices to be represented by a nonlinear function of the input or state it multiplies. The first eigenvalue σ_1 can therefore be expressed as a function g_{σ_1} of x_1, across a given domain of this state; element b_{11} is a function of the input u_1 and so on. The third order system of eq.10 can be now expressed as:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} g_{\sigma_1}(x_1) & g_{\omega}(x_2) & 0 \\ -g_{\omega}(x_1) & g_{\sigma_1}(x_2) & 0 \\ 0 & 0 & g_{\sigma_2}(x_3) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} g_{b_{11}}(u_1) & g_{b_{12}}(u_2) \\ g_{b_{21}}(u_1) & g_{b_{22}}(u_2) \\ g_{b_{31}}(u_1) & g_{b_{32}}(u_2) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} g_{c_{11}}(x_1) & g_{c_{12}}(x_2) & g_{c_{13}}(x_3) \\ g_{c_{21}}(x_1) & g_{c_{22}}(x_1) & g_{c_{23}}(x_1) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} g_{d_{11}}(u_1) & g_{d_{12}}(u_2) \\ g_{d_{21}}(u_1) & g_{d_{22}}(u_2) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

$$\begin{pmatrix} y_{c_1} \\ y_{c_2} \end{pmatrix} = \begin{pmatrix} \Phi_1(b_{11}, b_{12}, b_{21}, b_{22}) \\ \Phi_2(b_{31}, b_{32}) \end{pmatrix} \tag{11}$$

Here we also set additional outputs y_{c_i} in order to constrain the parameters of the B matrix, with the dual objective of normalising each modal state (x_3) or state pair (x_1 and x_2) while avoiding the parameter conditioning problems that arise if B and C elements are allowed to vary freely. A fixed number of nodes is then prescribed for each nonlinear function, as shown in Figure 1.

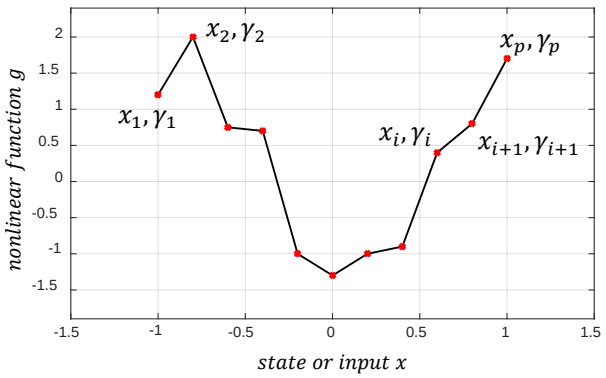

Figure 1. Every element of the *A*, *B*, *C*, and *D* matrices is a nonlinear function of a state or an input.

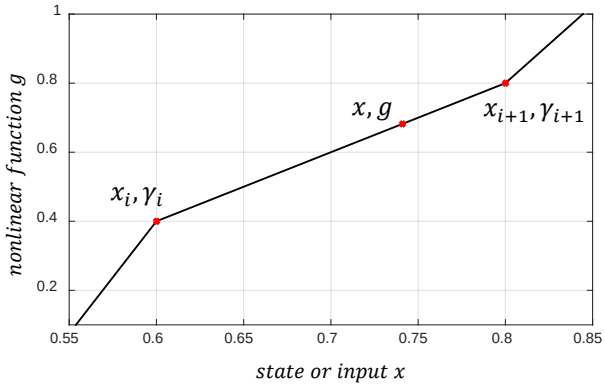

Figure 2. Generic *i-th* interval in the nonlinear function of an element in *A*, *B*, *C* or *D*.

The extended state vector becomes:

$$z=\left(x_1 \;\cdots\; x_n \;\; \gamma_1\left(g_{\sigma_1}\right) \;\cdots\; \gamma_p\left(g_{\sigma_1}\right) \;\; \gamma_1\left(g_{\omega}\right) \;\cdots\; \gamma_p\left(g_{\omega}\right) \;\cdots\; \gamma_1\left(g_{b_{11}}\right) \;\cdots\; \gamma_p\left(g_{b_{11}}\right) \;\cdots\right)^T \tag{12}$$

where x_i are the actual *n*-states, and γ_i are the *y*-ordinates corresponding to the nodes of each nonlinear-varying element in the *A*, *B*, *C*, and *D* matrices; *p* is the fixed number of nodes for each nonlinearity. By normalising the inputs and outputs over the interval [-1,+1] and constraining the states to be similarly normalised, all variables operate in the specified domain of *g*. The domain of each nonlinear function is divided into *(p* - 1*)* equally-spaced regions, so *p* parameters are identified for each nonlinear function *g*. Increasing the number of nodes might improve the accuracy of the identification, but will also increase the total number of parameters in the model. For example, a single order, single input system with two outputs will result in an extended vector of 127 total elements (*p* = 21 parameters for each nonlinear element in the *A*, *B*, *C*, and *D* matrices, plus one state). Full nonlinearities can be applied to some but not all elements of the state-space description, i.e. some elements might only be described by *p* = 2 and we will see an example of this at the end of this section. The exact value of each *A*, *B*, *C*, and *D* element at any given time step is calculated via linear interpolation:

$$g=\frac{(x-\chi_i)}{(\chi_{i+1}-\chi_i)}\cdot\gamma_{i+1}+\frac{(\chi_{i+1}-x)}{(\chi_{i+1}-\chi_i)}\cdot\gamma_i \tag{13}$$

where *x* refers here to the state or input appropriate to the nonlinearity. The coding of the Jacobians requires particular attention, as each nonlinear parameter is a function of the state or input it multiplies, but also of the γ_i ordinates it depends on, that are also part of the extended state vector. Only the nodes that define the interval we are falling into at each instant in time need to be considered; the derivatives will be zero with respect to every other node. Figure 2 represents a generic i^{th} interval in the nonlinear domain of Figure 1.

The derivatives of the nonlinear function *g* with respect to the state *x* and parameters γ_i and γ_{i+1} are computed from eq.13 as follows:

$$\frac{\partial g}{\partial x}=\frac{\gamma_{i+1}-\gamma_i}{\chi_{i+1}-\chi_i} \qquad \frac{\partial g}{\partial \gamma_i}=\left(\frac{\chi_{i+1}-x}{\chi_{i+1}-\chi_i}\right) \qquad \frac{\partial g}{\partial \gamma_{i+1}}=\left(\frac{x-\chi_i}{\chi_{i+1}-\chi_i}\right) \tag{14}$$

Therefore, given a third order, single-input, single-output example, with a conjugate pair and a real pole, we have the following process and measurement models:

$$f=\dot{z}=\begin{pmatrix} x_1\cdot g_{\sigma_1}(x_1)+x_2\cdot g_{\omega}(x_2)+u\cdot g_{b_1}(u) \\ -x_1\cdot g_{\omega}(x_1)+x_2\cdot g_{\sigma_1}(x_2)+u\cdot g_{b_2}(u) \\ x_3\cdot g_{\sigma_2}(x_3)+u\cdot g_{b_3}(u) \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$y=\left(x_1\cdot g_{c_1}(x_1)+x_2\cdot g_{c_2}(x_2)+x_3\cdot g_{c_3}(x_3)+u\cdot g_d(u)\right) \tag{15}$$

and an extended vector defined as:

$$z=\left(x_1\cdots x_3 \;\; \gamma_1\left(g_{\sigma_1}\right) \;\cdots\; \gamma_p\left(g_{\sigma_1}\right) \;\cdots\; \gamma_1\left(g_{b_1}\right) \;\cdots\; \gamma_p\left(g_{b_1}\right) \;\cdots\; \gamma_1\left(g_{c_1}\right) \;\cdots\; \gamma_1\left(g_d\right) \;\cdots\right)^T \tag{16}$$

The Jacobian *F* of the process model then becomes:

$$F=\begin{pmatrix} \frac{\partial f_1}{\partial z_1} & \frac{\partial f_1}{\partial z_2} & \frac{\partial f_1}{\partial z_3} & \cdots & \cdots & \cdots & \cdots & \frac{\partial f_1}{\partial z_n} \\ \frac{\partial f_2}{\partial z_1} & \frac{\partial f_2}{\partial z_2} & \frac{\partial f_2}{\partial z_3} & \cdots & \cdots & \cdots & \cdots & \frac{\partial f_2}{\partial z_n} \\ \frac{\partial f_3}{\partial z_1} & \frac{\partial f_3}{\partial z_2} & \frac{\partial f_3}{\partial z_3} & \cdots & \cdots & \cdots & \cdots & \frac{\partial f_3}{\partial z_n} \\ 0 & \cdots & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \cdots & \cdots & \cdots & \cdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & \cdots & \cdots & \cdots & 0 \end{pmatrix}=$$

$$= \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & 0 & \frac{\partial f_1}{\partial \gamma_1(g_{\sigma_1})} & \cdots & \frac{\partial f_1}{\partial \gamma_p(g_{\sigma_1})} & \frac{\partial f_1}{\partial \gamma_1(g_{\omega})} & \cdots & 0 & \cdots & \frac{\partial f_1}{\partial \gamma(g_{b_1})} & \cdots \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & 0 & \frac{\partial f_2}{\partial \gamma_1(g_{\sigma_1})} & \cdots & \frac{\partial f_2}{\partial \gamma_p(g_{\sigma_1})} & \frac{\partial f_2}{\partial \gamma_1(g_{\omega})} & \cdots & 0 & \cdots & \cdots & \frac{\partial f_2}{\partial \gamma(g_{b_2})} \\ 0 & 0 & \frac{\partial f_3}{\partial x_3} & 0 & \cdots & 0 & 0 & \cdots & 0 & \frac{\partial f_3}{\partial \gamma_1(g_{\sigma_2})} & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 & \cdots & 0 & \cdots & 0 & \cdots & \cdots & \cdots & 0 \end{pmatrix} \tag{17}$$

Here:

$$\frac{\partial f}{\partial x_i} = \frac{\partial f}{\partial g}\frac{\partial g}{\partial x_i} + \frac{\partial f}{\partial x_i} \tag{18}$$

and:

$$\frac{\partial f}{\partial \gamma_i(g)} = \frac{\partial f}{\partial g}\frac{\partial g}{\partial \gamma_i(g)} \tag{19}$$

The Jacobian *H* of the nonlinear measurement model is calculated with the same procedure, on the nonlinear functions of the *C* and *D* matrices.

2.3 Implementation

Eqns.4-7 can now be applied to a given time history of input-output test data, specifically acquired in *N* samples for the purpose of identification. A difficult and very often experience-based decision in every Kalman Filter application is the initial setting of the covariance matrices *Q*, *R* and *P*. From eq.2:

$$cov(\omega) = Q \quad cov(\nu) = R \tag{20}$$

The process error is not known or predictable here, so the model covariance is nominally set to $Q = \lambda I$, where *I* is the identity matrix and λ is the only tuning parameter; this defines the speed of variation of the identified parameters. The elements of *Q* relative to the actual states of the system are all set zero, since we make the assumption that the errors are in the parameter settings, not in the model structure. *Q* then becomes:

$$Q = \begin{pmatrix} 0 & \cdots & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & \cdots & \vdots \\ \vdots & \cdots & 0 & \lambda & \cdots & \cdots & 0 \\ \vdots & \cdots & \cdots & \cdots & \ddots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & \cdots & \cdots & \lambda \end{pmatrix} \tag{21}$$

P is initially and sensibly set equal to *Q*. An initialisation of all the parameters is needed; these can be nominally set to zero, or alternatively a simple linear identification can be performed on the data using the structure represented in eq.10 with constant elements in *A*, *B*, *C* and *D*. For further information on the linear identification see [14]. In this case, each of the parameters in the linear state space model initialises the relative nonlinear function: all the $\gamma_i(g_{\sigma_1})$ ordinates (*i =1…p)* of the nonlinear first eigenvalue are nominally assigned to the identified linear value of σ_1, and so on. Using the initial parameter set, the filter completes one full iteration by operating on all the *N* samples of the data and then iteratively repeats the process through the input-output time history, effectively *rinsing* the model through the data. Each iteration starts with the parameter set that has been identified at the end of the previous iteration.

The error covariance matrix *R* is numerically re-evaluated at the end of each iteration and this is computed from eq.20 where the error υ between the simulated output of the identified system and the original data is computed on the whole time-history using the most recent available parameter set. Constraints are applied to the input matrix using the additional outputs y_c. One constraint is applied to each state associated with a single pole, and one is required for each pair of states associated with an eigenvalue conjugate pair. So in this case:

$$y = \begin{pmatrix} x_1 \cdot g_{c_1}(x_1) + x_2 \cdot g_{c_2}(x_2) + x_3 \cdot g_{c_3}(x_3) + u \cdot g_d(u) \\ \Phi_1(g_{b_1}, g_{b_2}) \\ \Phi_2(g_{b_3}) \end{pmatrix} \tag{22}$$

The model for each constraint equation is the sum of squared γ_i parameters of the nonlinear functions of the *B* matrix elements that refer to the given real pole or conjugate pair:

$$\begin{aligned} \Phi_1(g_{b_1}, g_{b_2}) &= \sum_{j=1}^{p_{b_1}} \gamma_j^{\,2}(g_{b_1}) + \sum_{j=1}^{p_{b_2}} \gamma_j^{\,2}(g_{b_2}) \\ \Phi_2(g_{b_3}) &= \sum_{j=1}^{p_{b_3}} \gamma_j^{\,2}(g_{b_3}) \end{aligned} \tag{23}$$

The outputs y_c are then initialised according to the starting values of the γ_i (to give zero error) and are then adapted after each iteration *i* of the algorithm according to the maximum absolute value of the normalised state or state pair, across the identifying data. Here:

$$y_{c_1}(i) = \frac{y_{c_1}(i-1)}{\max(|x_1|,|x_2|)}$$
$$y_{c_2}(i) = \frac{y_{c_2}(i-1)}{\max(|x_3|)} \quad (24)$$

The EKF therefore has additional innovations which continuously constrain the magnitude of the elements of *B* to ensure all states fully occupy but do not exceed the prescribed domain, [-1,+1].

3 RESULTS

3.1 Identification of a simple model

A simple example will now be used to illustrate the method. Consider the single order, single output and single input system described by the following process and measurement equations:

$$\dot{x} = -x^3 + 0.2u$$
$$y = x \quad (25)$$

With no a priori knowledge of the system dynamics assumed in the idetnfiication process, it is sensible and standard practice to use a broad-bandwidth random input to excite the plant. Here the input is white noise at a high sampling frequency (500Hz) filtered to remove all content above 25Hz. This produces smooth data, allowing better and faster identification through the Kalman Filter (which depends on Euler integration). It also has the advantage of exciting the system through a wide range of potentially important frequencies, with the one simple assumption that the relevant system dynamics occur within a known bandwidth to (nominally) 25Hz. A different set of data is of course used for the validation process, on which all the performance results are based. Identification performance is measured using percentage *explanation:*

$$R_x = 1 - \frac{\sum_{k=1}^{N}(y_k - h_k)^2}{\sum_{k=1}^{N} y_k^2} \quad (26)$$

The tuning parameter λ provides stable running of the filter on this data over a wide range of values, between $10^{-2} < \lambda < 10^{-10}$; with 10^{-2} at the limit of filter stability, and 10^{-10} still effective, but with an inconveniently slow optimisation. At the selected $\lambda = 10^{-5}$ the filter is comfortably stable and achieves an optimised result in under 40 seconds.

With an unknown system it is sensible to first consider the simplest possible case: a non-dynamic, zero-order model comprising *D* elements alone. Such a model could be identified using simple ordinary least-squares techniques, but our filter can also be employed. As we would expect, this returns a very poor fit, with R_x < 1%. Linear models and nonlinear extensions of various orders and with various choices of *p* can then be explored.

In order to determine the lowest order, simplest model that achieves acceptably high performance (a subjective decision) the method should be applied repeatedly in a systematic way, starting with the zero order case and identifying models with progressively higher order and number of nonlinear nodes *p*. As model order and complexity increases, the resulting (validation) performance increases to a plateau; further, when unnecessarily high order models are identified, the parameters may diverge due to poor conditioning and/or repeated eigenvalues appear in the result. We see an example of this in Section 3.2. Thus by systematically identifying several models, the best performing, well conditioned and lowest order option becomes apparent.

Here a first order linear model identification is now run and the best fit is achieved at an explanation of 95.21%. This uses constant *A, B, C,* and *D* matrices, as per eq.10 with only one eigenvalue, identified at -0.66. The next step adds nonlinearity, with low resolution (for simplicity, only $p = 2$ per nonlinear function). This slightly improves the performance to 95.58%, but the obtained results provide useful information: Figure 3 shows the evolution of the identified parameters over the iterations (plot a), along with the evolution of the trace of the *P* matrix (plot b). We see good convergence, which shows good conditioning. Figure 4 illustrates the final model and it is clear that greater resolution is needed in the *A* matrix, due to the considerable variation in value that the system is trying to achieve, while *B* and *C* resemble constant values, and the *D* matrix is almost zero.

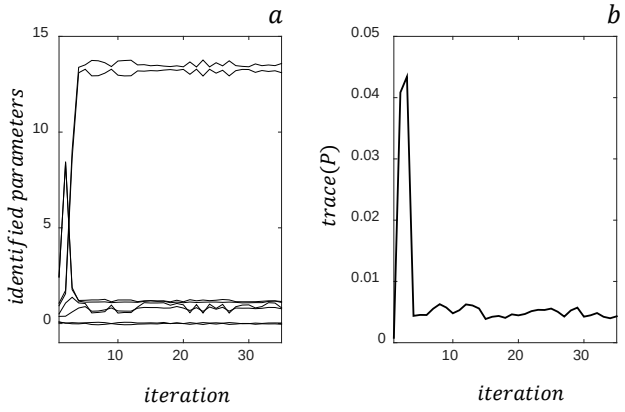

Figure 3. Parameter convergence and trace of *P* for nonlinear identification results with low accuracy in the *A, B, C,* and *D* functions.

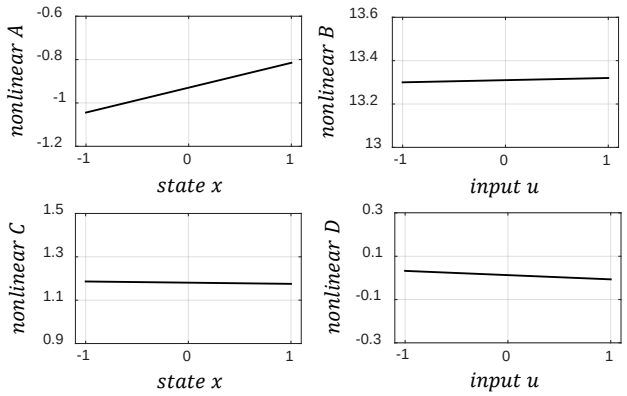

Figure 4. Nonlinear identification results with low accuracy in the *A, B, C,* and *D* functions.

A further identification run with greater resolution achieves 99.92%, with a nonlinear *A* function now free to display an approximately quadratic form (with $p = 3$, Figure 5a). And finally a high-resolution (21 node) nonlinear function gives the freedom to return 100% accuracy in the validation output and an almost perfectly shaped quadratic curve in the *A* matrix nonlinearity (Figure 5b).

Despite the higher complexity, the model shows no variation for both the nonlinear elements in the *B* and *C* matrices, and *D* is still correctly identified as zero (not shown). Note that, other than slower operation of the identifying filter, there is no disadvantage in increasing *p* for all elements of *B, C* and *D*; because the parameters are distributed over the normalised range of the input and states, they generally retain good conditioning. Care must be taken with the selection of *p* for the *A* matrix terms however, as we see in the next section.

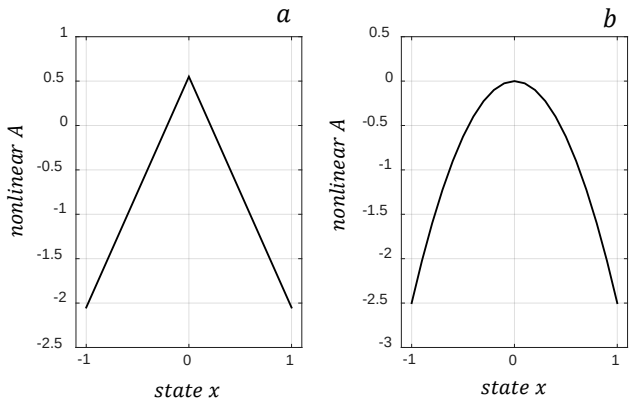

Figure 5. Nonlinear identification results with increasing resolution in the *A* function.

3.2 Identification of a brake model

The new Filter is tested for the identification of a highly nonlinear automotive brake model. Two sets of input-output data have been provided by Jaguar Land Rover (an identification set and a validation set). The recorded output represents the hydraulic pressure at the calipers while the input is the pressure at the brake pedal. The identification set again consists of white noise, filtered above 25Hz and the validation set comprises a nominal sequence of brake applications of varying intensity. No other physical description of the system has been provided.

In this case the data is single sided, so domain intervals of [0,1] are applied. As in the previous example, $\lambda = 10^{-5}$ and a zero-order nonlinear input-output relationship is first sought (*D* matrix look-up table - Figure 6). This achieves an identification performance of R_x = 98.01%, which confirms the value of a direct nonlinear relationship and sets a benchmark R_x performance.

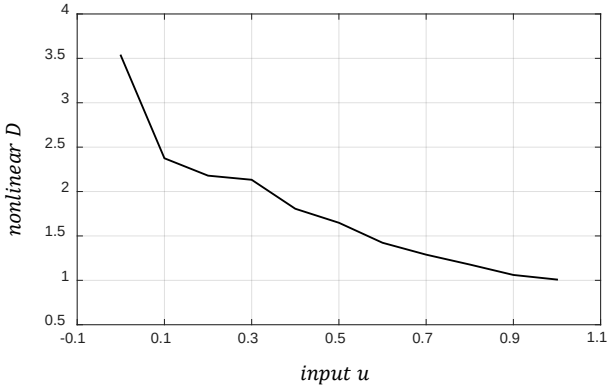

Figure 6. Nonlinear direct input-output relationship, zero-order model identification.

Further, close observation of the input-output time-histories shows a dynamic model is clearly needed: there is a time delay in the response at low magnitude (Figure 7a), though interestingly, this does not appear to be the case at high amplitudes (Figure 7b), where no phase error is evident. Note also the relationship between input and output (Figure 7c) which shows a further challenge to the identification; the model must cope with hard, step nonlinearities here as there is clearly a deadband in the response at low input magnitude.

A first-order linear state-space identification achieves performance R_x = 94.86%, so we can see nonlinearity modelling is more critical than transient modelling alone.

Next, a nonlinear single-order system with high resolution $p = 21$ causes instability, whereas basic resolution in *A* ($p = 3$), with full complexity elsewhere has the effect of preventing over-parametrisation and gives an accuracy of 97.50%. Figure 8 shows how the parameters vary with increasing iteration of the algorithm; here we see the $p = 21$ case has divergent parameters, but the $p = 3$ case shows clear convergence. Divergence will only occur where the model structure has become poorly conditioned; in this case the deadband in the output at low magnitudes is causing the eigenvalue to drift to unstable positive values for the unused

lower range of the input (and hence states), so associated B and C parameters diverge. By restricting the nonlinearity resolution in *A* (only) to limit eigenvalue variability in this region, the parameters converge and the final model shown in Figure 9 gives 99.60% performance, which is illustrated in Figure 10.

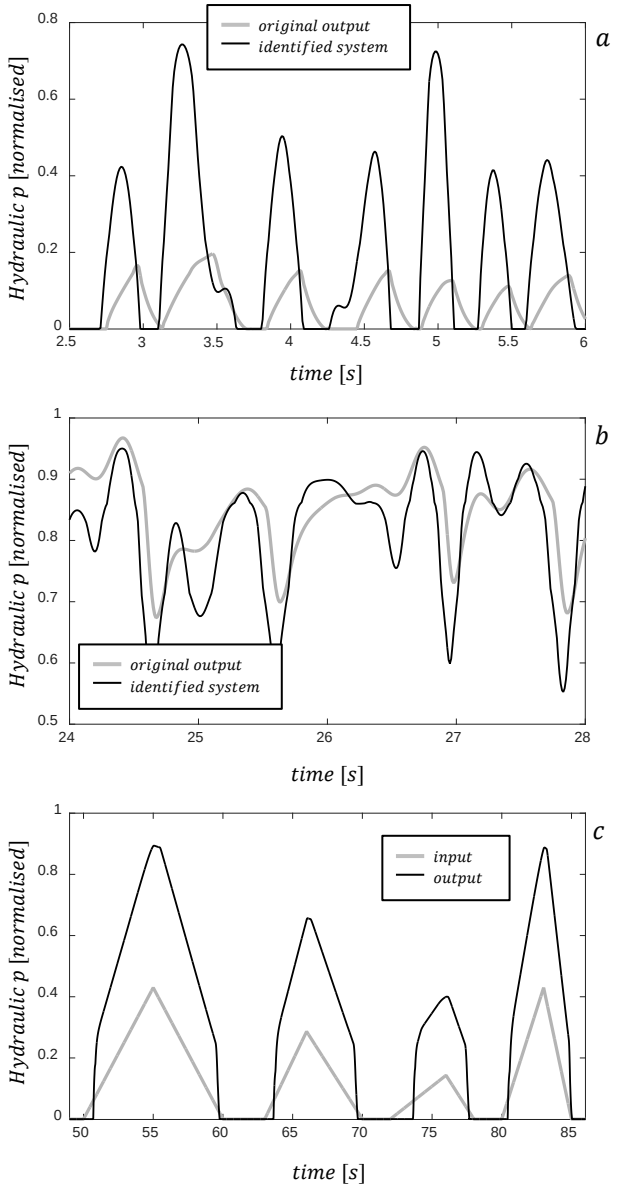

Figure 7. Evidence of dynamic behaviour in input-output time history at low (*a*) and high (*b*) magnitudes, and evidence of deadband (c – validation data shown).

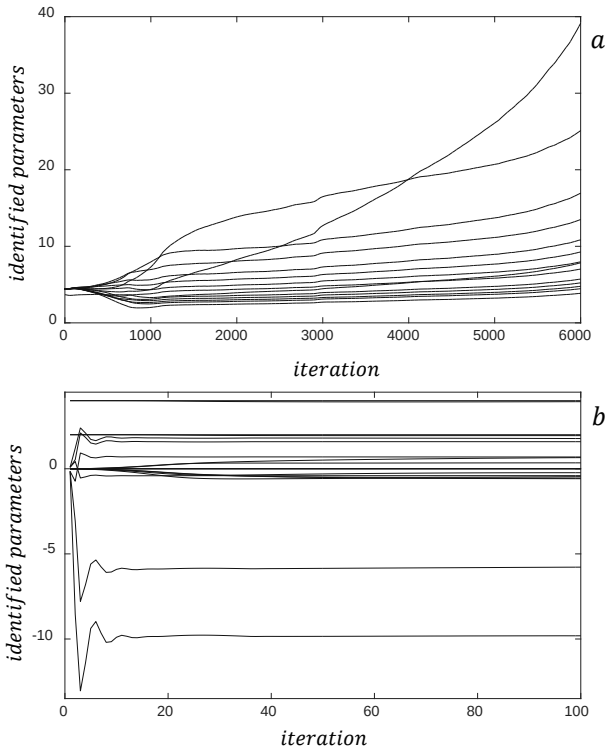

Figure 8. Divergence of the identified parameters in the p = 21 case (a - only diverging parameters shown) and convergence for the case of p = 3 (b).

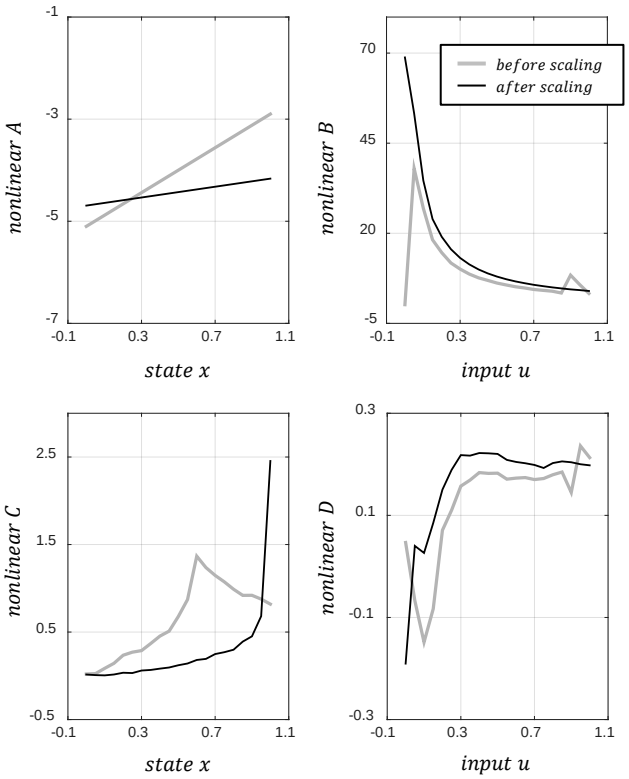

Figure 9. *A, B, C* and *D* nonlinear functions. Single-order system, before and after scaling.

Dynamic behaviour replication is now good at both low and high magnitudes, as shown in Figure 10a and Figure 10b; validation data can be seen in Figure 10c. However, the corresponding normalised state behaviour has an interesting characteristic (Figure 11) with a plateau at 0.5 and only few excursions to the normalised peak.

This effect is due to there being very little data at the highest input magnitudes. It does not restrict the validity or accuracy of the model but does allow us to illustrate parameter robustness. By now applying a simple scaling of 1/0.5 to the g_b elements and restarting the filter, the model is rapidly re-optimised (within 5 iterations). This produces smoother nonlinear variations (Figure 9 – after scaling). It is important to note that the overall performance index does not change following this intervention.

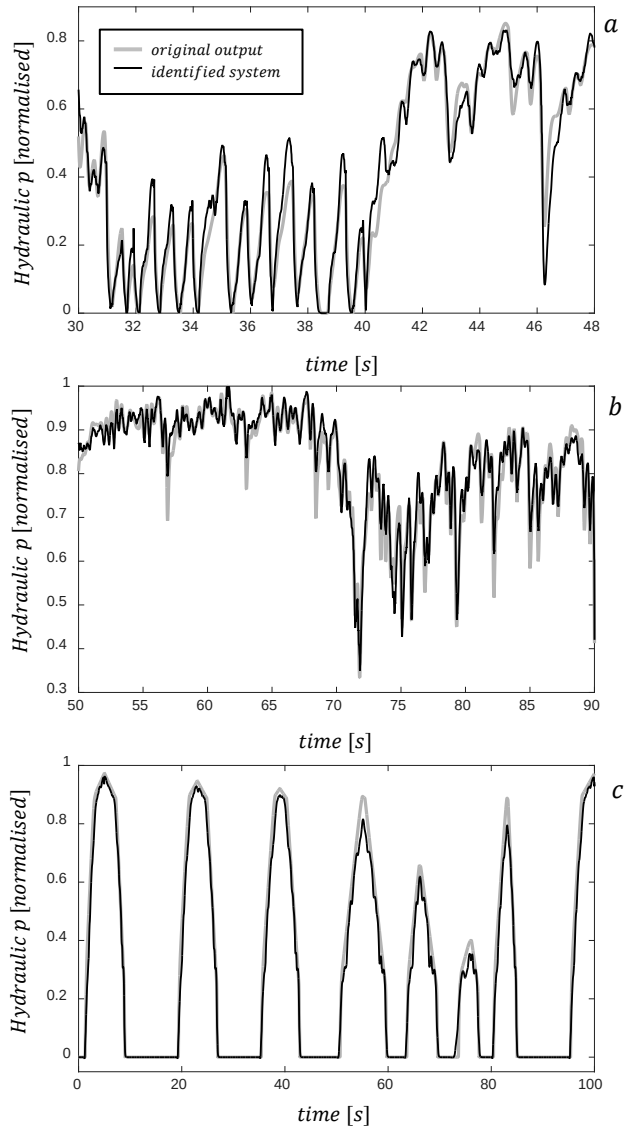

Figure 10. Nonlinear single-order low (*a*) and high (*b*) magnitude dynamic behaviour. Full validation data-set also shown (*c*).

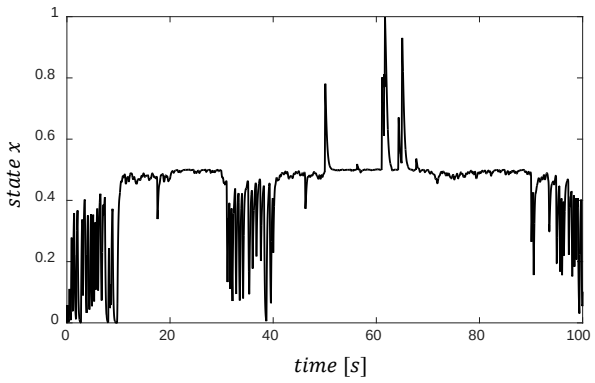

Figure 11. Normalised state behaviour for nonlinear single-order system.

Progressively higher orders can now be explored, until a satisfactory model has been identified or a good compromise between complexity and accuracy has been found. This process could easily be automated; for example an overnight run of optimisations would yield accuracy and parameter convergence statistics on several model order and nonlinearity complexity combinations. For this example, a summary of the process is given in Table 1 in the Appendix.

4 CONCLUSIONS

A novel method for MIMO black-box system identification has been presented, which can be employed for model order reduction applications as well as conventional nonlinear plant system identification problems. The new structure is based on an extension of the linear state space system where each parameter of the *A, B, C* and *D* matrices becomes a nonlinear function of an input or state. The method is easy to implement, operating in the time domain using the well-known Extended Kalman Filter.

The prescribed structure provides effective models for smoothly nonlinear systems and can also approximate hard nonlinearities such as the dead-band in the brake model example considered here. It would be naïve to assume the structure is capable of replicating all behaviour in systems with multiple combinations of harsh nonlinearity, but it does provide enough flexibility to map the most significant nonlinear effects and variations in system dynamics; it is able to identify the best minimal order approximation to system reponse using a simple nonlinear structure.

Being in the modal canonical form it also provides further information in terms of the most relevant modes of the system. The tool is fully black-box and requires no a-priori engineering knowledge of the system to be identified. It has been successfully applied to a brake feel model from the automotive industry. The identification process consists of progressively expanding the order of the state space system, starting from a zero-order nonlinear input-output relationship,

up to an order where a good compromise between complexity and performance is found.

Accuracy of fit and parameter convergence behaviour are the main indicators in the process. Together with the nonlinear mapping of the eigenvalues, these allow the user to develop insight into the model behaviour, most suitable model order and degree of nonlinearity in an intuitive way. Hence, although the filter operates on the data as a fully black-box process, consecutive optimisations allow the user to develop multiple solutions, through which they gain insight into the most appropriate and robust model to use. In this case, the model identified to fit the brake data is accurate, as demonstrated at both low and high magnitudes in the identification data and in the validation results. Further work will focus on automating the process, with the aim of delivering a completely black-box tool that can be operated with minmal user interpretation.

ACKNOWLEDGEMENT

This work is supported by Jaguar Land Rover and the UK-EPSRC under Grant EP/K014102/1 as part of the jointly funded Programme for Simulation Innovation (PSi).

REFERENCES

[1] Xie, Zongbo, and Jiuchao Feng. "Real-time nonlinear structural system identification via iterated unscented Kalman filter." Mechanical Systems and Signal Processing 28 (2012): 309-322.

[2] Best, Matt C., T. J. Gordon, and P. J. Dixon. "An extended adaptive Kalman filter for real-time state estimation of vehicle handling dynamics." *Vehicle System Dynamics* 34.1 (2000): 57-75.

[3] Sun, Fengchun, et al. "Adaptive unscented Kalman filtering for state of charge estimation of a lithium-ion battery for electric vehicles." *Energy* 36.5 (2011): 3531-3540.

[4] Best, Matt C., and Karol Bogdanski. “Full Vehicle and Tyre Identification using Unscented and Extended Identifying Kalman Filters” (2016).

[5] Söderström, Torsten, and Petre Stoica. *System identification*. Prentice-Hall, Inc., 1988.

[6] Ljung, Lennart. "System identification." *Signal Analysis and Prediction*. Birkhäuser Boston, 1998. 163-173.

[7] Juditsky, Anatoli, et al. "Nonlinear black-box models in system identification: Mathematical foundations." *Automatica* 31.12 (1995): 1725-1750.

[8] Sjöberg, Jonas, et al. "Nonlinear black-box modeling in system identification: a unified overview." *Automatica* 31.12 (1995): 1691-1724.

[9] Li, Zongyan, and Matt C. Best. "Optimisation of the input layer structure for feed-forward NARX neural network." (2015).

[10] Savaresi, Sergio M., Sergio Bittanti, and Mauro Montiglio. "Identification of semi-physical and black-box non-linear models: the case of MR-dampers for vehicles control." *Automatica* 41.1 (2005): 113-127.

[11] Van Mulders, Anne, et al. "Two nonlinear optimization methods for black box identification compared." *Automatica* 46.10 (2010): 1675-1681.

[12] Tørdal, Sondre Sanden, Andreas Klausen, and Morten K. Bak. "Experimental System Identification and Black Box Modeling of Hydraulic Directional Control Valve." (2015): 225-235.

[13] Corno, Matteo, and Sergio M. Savaresi. "Experimental identification of engine-to-slip dynamics for traction control applications in a sport motorbike." *European Journal of Control* 16.1 (2010): 88-108.

[14] Best, Matt C., and Karol Bogdanski. "Extending the Kalman filter for structured identification of linear and nonlinear systems." (2016).

[15] Julier, Simon J., and Jeffrey K. Uhlmann. "New extension of the Kalman filter to nonlinear systems." *AeroSense'97*. International Society for Optics and Photonics, 1997.

[16] Kalman, Rudolph Emil. "A new approach to linear filtering and prediction problems." *Journal of basic Engineering* 82.1 (1960): 35-45.

[17] Carlson, Christopher R., J. Christian Gerdes, and J. David Powell. "Practical position and yaw rate estimation with GPS and differential wheelspeeds."*Proceedings of AVEC 6th International Symposium*. 2002.

[18] Hodgson, G., and Matt C. Best. "A parameter identifying a Kalman filter observer for vehicle handling dynamics." *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 220.8 (2006): 1063-1072.

APPENDIX

Table 1. Summary of identification results for different order models.

System Order	Description	Comments	Linear Model *Rx*	Linear Model Eigenvalues	Nonlinear Model *Rx*	Time taken (per iteration)
0	*D* matrix only (look-up table)	Evidence of nonlinearity in the input; further evidence of dynamic behaviour clear from input-output relationship.	-	-	98.01%	0.66s
1	Basic resolution in *A;* Higher resolution in *B*, *C* and *D*	Low resolution in A prevents over-parametrisation; evidence found of minimum input threshold from negative eigenvalues at low magnitude;	94.86%	-1.47	99.60%	1.22s
2	Eigenvalues in a conjugate pair	Good performance but eigenvalue optimised with zero frequency, suggesting two real poles;	60.49%	-2.11 ± j 0.02	99.00%	2.13s
2	Two real poles	Good performance but nonlinear model identifies poles very close together;	94.70%	-1.09; -4.74	99.20%	1.92s
3+	Progressively higher orders	Over-parametrised and increasingly computationally heavy, no better performance achieved;	94+%	-	99+%	2.70s+

Karol Bogdanski received a BSc in Mechanical Engineering from the University of Padua, Italy and a MSc in Automotive Systems Engineering from Loughborough University, UK. He is currently pursuing his PhD in the Department of Aeronautical and Automotive Engineering at Loughborough, as part of the Programme for Simulation Innovation (PSi), jointly funded by Jaguar Land Rover and the UK-EPSRC. His interests are in system identification, Kalman filters and vehicle dynamics.

Matt Best received his BEng and PhD from Loughborough University, UK, and is a Senior Lecturer in Vehicle Dynamics and Control in the Department of Aeronautical and Automotive Engineering at Loughborough. He has published over 50 papers on system identification, estimation and control of vehicles and drivers, and has a particular interest in the use of Kalman filters for both state and parameter estimation.