AIAA Aviation Forum and ASCEND co-located Conference Proceedings
29 July - 2 August 2024, Las Vegas, Nevada
AIAA AVIATION FORUM AND ASCEND 2024

10.2514/6.2024-4158

Evaluation of Reduced Order Aerodynamic Models for Transonic Flow over a Multiple-Swept Wing Configuration

Mehdi Ghoreyshi[1]*, Pooneh Aref[1]†, Mario Stradtner[2]‡,
Anastasios Panagiotopoulos[3]§, Michel van Rooij[4]¶, Peter Hans Leonard Blom[4]‖, and Steven Hulshoff[3]**

[1] *High Performance Computing Research Center, U.S. Air Force Academy*
USAF Academy, Colorado 80840, United States

[2] *German Aerospace Center (DLR)*
Braunschweig, 38108, Germany

[3] *Delft University of Technology*
Delft, 2629 HS, The Netherlands

[4] *NLR - Royal Netherlands Aerospace Centre*
Amsterdam, 1059 CM, The Netherlands

Efficient input data generation for reduced-order model applications to accurately predict aerodynamic performance and stability characteristics over a large part of a fighter aircraft's flight envelope is a major challenge. In this paper, aerodynamic reduced-order models are created from two pseudorandom binary sequence (PRBS) training maneuvers. During these maneuvers, the angle of attack and pitch rate change in a periodic and deterministic manner which is characterized by white-noise-like properties. Typical PRBS signals include sudden input variations between two distinct values, such as minimum and maximum angles of attack. However, the signals used in this paper were modified to have the step changes to depend on the simulation time. In the first motion, the aircraft undergoes a signal at a constant Mach number of 0.85. In the second motion, the Mach number varies in an optimized manner from 0.1 to 0.9. The test case is a generic triple-delta wing configuration. Simulations were run using the DoD HPCMP CREATE$^{\text{RM}}$-AV/Kestrel simulation tools. A prescribed-body motion was used to vary input parameters under given freestream conditions (Mach number and angle of attack). Different reduced-order methods were applied, that comprise regression, feed-forward neural network and auto-regressive surrogate modeling techniques to predict integrated force and moment coefficients and a proper-orthogonal decomposition based neural network approach for surface pressure prediction. Once models of integrated forces and moments were created, they were used to predict static and stability derivatives at different angles of attack and Mach numbers. Models were then used to predict aerodynamic responses to arbitrary motions including pitch sinusoidal, chirp, Schroeder, and step. Model predictions were compared with actual CFD data. Overall, a good agreement was found for all models. Models to predict surface pressure data were also able to accurately predict the upper surface pressure data at different spanwise and chordwise locations at different angles of attack for both static and dynamic runs.

I. Introduction

There is a growing interest of using computational aerodynamics in aircraft conceptual design. Computational aerodynamics models range from potential flow solvers, such as vortex lattice methods, to Euler equations, and extend up to Reynolds-Averaged Navier Stokes (RANS) equations. The accuracy and computational demands vary from

---

[*] Research Scientist, USAFA/DFAN, AIAA Associate Fellow.
[†] Aerospace Research Engineer, USAFA/DFAN.
[‡] Research Scientist, Institute of Aerodynamics and Flow Technology, Lilienthalplatz 7.
[§] M.Sc. Student, Faculty of Aerospace Engineering
[¶] Senior Scientist, Dept. of Flight Physics and Loads.
[‖] Junior R&D Engineer, Dept. of Flight Physics and Loads
[**] Assistant Professor, Faculty of Aerospace Engineering.

model to model, with RANS models being the most expensive and accurate. According to Gu et al. [1] and Ciampa et al. [2], there are four levels of aerodynamic modeling fidelity: Levels 0, 1, 2, and 3. Level 0 involves semi-empirical methods or historical databases. Level 1 includes basic linear aerodynamic models. Level 2 uses computational methods for designing aircraft components like wings and fuselage, incorporating more detailed physics and non-linear phenomena. Level 3, the highest level, represents cutting-edge physics simulations. Computational expenses range from under one second for Level 0, to minutes for Level 1, hours for Level 2, and days for Level 3.

At their highest practical application, Reynolds Averaged Navier Stokes(RANS) equations could accurately predict the underlying flow physics and the unsteady and nonlinear aerodynamics of air vehicles under various flight conditions and flow speeds. Many unexpected aerodynamic issues due to turbulence, flow separation, and shock waves, could be identified and fixed early in the design stage to prevent adverse aircraft behavior. However, because of the high computational cost of using RANS equations for all flow conditions of interest, they are typically used for aerodynamic estimation at the design point rather than for generating an aerodynamic database, or performing stability and control (S&C) analysis, or aircraft design optimization.

In the aircraft design cycle, wind tunnel and flight testing take place towards the end. Thus, addressing aerodynamic or S&C issues discovered during flight testing may require repeating a significant portion of the design cycle; this turn-around time can lead to costly delays in the design and production of the vehicles. Therefore, it is essential to reduce risks during prototype testing by early detection of aerodynamic-related problems. The main limitation to use “high-fidelity" or “physic-based" simulations in a multi-disciplinary design approach is the computational expense. For example, performing S&C analysis over an aircraft’s flight envelope requires aerodynamic data for tens of thousands different states to encompass all angles of attack, sideslip angles, aircraft speeds, control surface deflections, and the time rates of Euler angle changes. This is unfeasible with a brute-force approach. If it were possible to create Reduced-Order Models (ROMs) that maintain accuracy with only hundreds or even tens of simulations, this would allow for the early-stage simulation of a real aircraft, including all multidisciplinary interactions across the entire flight envelope, and deliver data with the accuracy necessary for development and certification [3]. Validated and precise ROMs can assist in performing virtual testing before actual flight tests, which are costly and might pose risks. Virtual flight testing helps identify unforeseen issues in aircraft handling, thereby minimizing the need for multiple physical prototypes that are both expensive and time-consuming. Additionally, validated ROMs can accelerate design iterations in a virtual environment, enabling quick testing and design improvements without incurring the costs of physical changes. An additional benefit of having accurate numerical predictions during aircraft design is the ability to optimize the design for improved aerodynamic performance. With the availability of aerodynamic data, control surface sizing, performance prediction (e.g., range), and structural analysis can also be performed. It should be noted that this is not intended to completely replace wind tunnel or flight testing with computational methods, but rather to provide accurate predictions early in the design stage when experiments are not yet available. In summary, aircraft designs incorporating advanced computational methods will diminish development costs and the number of necessary experiments and will speed up the design cycle. This is particularly crucial for the design of highly maneuverable aircraft, unstable aircraft, novel configurations like the Blended Wing Body (BWB), unconventional unmanned combat air vehicles (UCAV), and next-generation fighter aircraft that lack historical aerodynamic data.

To facilitate an automated physics-based or Computational Fluid Dynamics (CFD)-based aircraft design, three key procedures must be addressed: geometry definition/mesh generation, flow simulation, and utilization of engineering data from the flow solver output for specific design objectives. These aspects have been discussed by Ghoreyshi et al. [4] and Gu et al. [1]. Consequently, there is no requirement to explore automated geometry and mesh generation. However, the use of ROMs or System Identification (SID) techniques is still necessary for S&C analysis of this configuration using CFD.

ROMs offer a concise representation of unsteady flow dynamics using a limited number of spatial/temporal modes, typically fewer than one hundred, in contrast to the extensive number of grid points present in full-order models, which can range from 5 to 50 million or more [5]. This allows ROMs to swiftly predict responses to various inputs compared to the time-consuming computations required by full CFD solutions. Various ROM techniques are available, including indicial response methods [6], Proper Orthogonal Decomposition (POD) [7], Volterra theory [8], and Neural Network (NN) [9] and Machine Learning (ML) approaches [10, 11]. System Identification falls within the realm of Reduced Order Modeling, yet it is distinct in its dedicated focus on constructing precise models derived from observed data. Some methods include regression methods [12], state space representations, transfer functions, the auto-regressive with exogenous input (ARX) model [13, 14], surrogate-based recurrence framework (SBRF) [15, 16], Radial Basis Functions (RBF) [17], and NN methods [18, 19].

In this paper, generic CFD-based maneuver simulations as well as ROMs are employed for the Future Fighter

Demonstrator (FFD) use case of the NATO Science & Technology Organization research task group AVT-351 [20] to predict aircraft performance and S&C characteristics at transonic flow conditions. Note that, transonic aerodynamics is very challenging to predict due to shock waves and their movement. The reduced-order modeling or SID task includes the definition and computation of a training or input signal [21], the selection of a mathematical model (such as the order of regression parameters), and the choice of modeling techniques that yield the best fit to the observed data (e.g., number of neurons and hidden layers) [22]. Input signals include two pseudorandom binary sequence (PRBS) motions. During these maneuvers, the angle of attack and pitch rate change in a periodic and deterministic manner characterized by white-noise-like properties. Typical PRBS signals include sudden input variations between two distinct values, such as minimum and maximum angles of attack. However, the signals used in this paper were modified to have the step changes depend on time. In the first motion, the aircraft undergoes a signal at a constant Mach number of 0.85. Upper surface pressure data were recorded for six spanwise sections. In the second motion, the Mach number varies in an optimized manner from 0.1 to 0.9. All simulations were run using the DoD HPCMP CREATE$^{\text{RM}}$-AV/Kestrel simulation tools. A prescribed-body motion was used to vary input parameters under given freestream conditions (Mach number and angle of attack). Using these input signals, different ROM techniques were investigated to approximate the full-order aerodynamic model. USAFA used models based on regression and a feed-forward neural network approach, whereas DLR employed a surrogate-based recurrence framework, both to predict integrated forces and moment coefficients. NLR utilized Long Short-Term Memory (LSTM) neural network models in combination with enriched Proper Orthogonal Decomposition (ePOD) to predict the sectional surface pressure data as a function of angle of attack and pitch rate.

This paper is structured as follows: First, we describe the reduced-order modeling methods. Then, we provide details on the test case, computational grids, and the flow solver. Following that, we present results for the prediction of forces and moments from various signals, including predictions of new signals. Subsequently, we introduce ROMs for predicting surface pressure data. Finally, we draw conclusions of this joint effort within the NATO Science & Technology Organization research task group 351.

II. Numerical flow solver

Kestrel is the fixed-wing product of the CREATE$^{\text{TM}}$-AV program funded by the DoD High Performance Computing Modernization Program (HPCMP). The objective of the CREATE$^{\text{TM}}$ program is to diminish expenses, time, and risks associated with DoD acquisition programs. This was achieved through the creation and implementation of cross-disciplinary, physics-centered software applications tailored for the design and assessment of military aircraft, naval vessels, and radio frequency antenna systems. The initiative expanded its scope in 2012 to encompass ground vehicles as well, serving DoD engineering entities [23].

According to Sears and Morton [23], Kestrel is an advanced, multi-dimensional analysis tool designed to handle a diverse spectrum of linked physical processes, encompassing aerodynamics, thermochemistry, structural dynamics, thermodynamics, propulsion, and flight controls. The code has a Python-based infrastructure that integrates Python, C, C++, or Fortran-written components [24]. Kestrel version 12.8 is used in this work. The code has been extensively tested and a variety of validation documents have been reported [25, 26].

Kestrel employs a Common Scalable Infrastructure (CSI) to adopt a modular approach in linking computational tools necessary for conducting comprehensive assessments of fixed-wing aircraft across various disciplines. Kestrel CFD solvers include KCFD [27], COFFE [28], and KCFD/SAMAir [29]; the KCFD flow solver is used in this study. KCFD uses a second-order accurate cell-centered finite-volume discretization while SAMAir utilizes a fifth-order finite-volume discretization on Cartesian meshes [30]. KCFD solves the unsteady, three-dimensional, compressible RANS equations on hybrid unstructured grids [31] using the Method of Lines (MOL) to separate temporal and spatial integration schemes from each other [27]. The spatial residual is computed via a Godunov type scheme and second-order spatial accuracy is obtained through a least squares reconstruction. The numerical fluxes at each element face are computed using various exact and approximate Riemann schemes with a default method based on HLLE++ scheme [32]. In addition, the code uses a subiterative, point-implicit method (a typical Gauss-Seidel technique) to improve the temporal accuracy. Kestrel provides the capability to utilize tetrahedral grids and mixed-element unstructured meshes, allowing for flexible combinations of tetrahedra, prisms, and pyramids [33]. Some of the turbulence models available within Kestrel include Spalart–Allmaras (SA), Spalart–Allmaras with rotational/curvature correction (SARC), Menter's SST [34], and Delayed Detached Eddy Simulation (DDES) with SARC. To generate the data for ROM developemnt, the Prescribed Body Motion capability of Kestrel was used. The available motion types in Kestrel include: constant rate pitch/yaw/roll, sinusoidal, pitch/yaw/roll motion with time varying amplitude and frequency, ramp, and step motions. Arbitrary motion can be prescrived using external motion definition files that contain time instants, Euler angles, and mesh reference point coordinates at each time instant. In addition to CSI, this study uses other elements of the Kestrel software, such as the KUI (Kestrel User Interface), JobView, and Carpenter, a grid manipulation program. KUI is a GUI for setting up the input deck. JobView allows to visualize the set up case, orientation, reference points (cg, moment, mesh), wind direction vector, tap points, Cartesian extent, and even a preflight of prescibed-body motions. Carpenter functions as a grid manipulation tool, enabling the conversion of multiple grid formats into Kestrel's preferred .avm format, along with providing several other beneficial functionalities such as translation/rotation, mirroring and subsetting the grid [33].

III. Reduced Order Aerodynamic Models

For a linear system, the system output is found as:

$$y(t) = \int_0^t g_0(\tau)u(t-\tau)d\tau \tag{1}$$

where y is the output, u is the input, and g_0 is the system impulse. In multi-input systems, superposition is applied. However, estimating impulse functions is a very challenging task, whether through computational or experimental methods. This model is only applicable to linear systems or systems with small amplitude responses. Full-order models, such as statistical analysis (white-box modeling) or look-up tables (brute-force approach) based on system observations, can serve as alternatives. However, gathering or simulating all these observations can be very time-consuming. Reduced-order aerodynamic models aim to accurately represent a system with significantly less data compared to full-order models.

In this paper, a system identification method is used at USAFA for estimation of aerodynamic coefficients and surface pressure estimation. DLR has applied a SBRF architecture employing Gaussian Process Regression (GPR) surrogate models to estimate aerodynamic coefficients. NLR has developed a modal decomposition method used for the dimensional reduction of high-fidelity data combined with Neural Networks to predict aerodynamic coefficients and surface pressure data.

A. System identification

SID is dedicated to creating mathematical models that can accurately describe system behavior. According to Galrinho [35], the SID process comprises four steps: 1) system excitation for data collection, 2) selection of model structures, 3) training the models and selecting the best one from the candidates, and 4) model validation. SID models are classified into three types: grey-box models, which are built on partial system knowledge; white-box models, which are based on statistical analysis of observations; and black-box models, which are created solely from data. This paper describes a gray-box identification approach based on linear, static regression to model the FFD aerodynamics at different flight conditions. The model is called grey box because we assume a regression model or relationship between the aerodynamic coefficients and input parameters. This model is static since the output depends only on the present inputs, without considering past inputs. A least-square error will then be used to estimate the model unknowns.

In more detail, a functional relationship (e.g., a polynomial) is assumed between forces and moments and the input parameters. A forced motion (i.e., a training maneuver) is then used to estimate the model unknowns. The model accuracy depends on the type of forced motion and input parameter excitation. Forced motion can be used to vary Mach number, angle of attack, acceleration terms, and angular rates in a single computation. A forced motion can be thought of as a computational flight test without kinematic restrictions (e.g., G-force) of the aircraft or pilot.

In this study, a third order polynomial model in angle of attack was chosen for the aerodynamic coefficients at constant Mach number:

$$C_j = C_{j0} + \beta_1\alpha + \beta_2\alpha^2 + \beta_3\alpha^3 + \beta_4 q + \beta_5\alpha.q + \beta_6\alpha^2.q + \beta_7\alpha^3 \tag{2}$$

Note that for the motions of this study $q = \dot{\alpha}$. For the motion with varying Mach number, the new model is assumed as:

$$C_j = C_{j0}+\beta_1\alpha+\beta_2\alpha^2+\beta_3\alpha^3+\beta_4 q+\beta_5\alpha.q+\beta_6\alpha^2.q+\beta_7\alpha^3+\beta_8 M+\beta_9 M.\alpha+\beta_{10}M.q+\beta_{11}M^2+\beta_{12}M^2.\alpha+\beta_{13}M^2.q \tag{3}$$

where $C_j = [C_L, C_D, C_m]$ correspond to lift, drag, and pitch moment coefficient. α is angle of attack. q is normalized pitch defined as $Q.c/(2V_\infty$ where Q is pitch rate in rad/s, c is the mean aerodynamic chord, and V_∞ denotes freestream velocity. M is Mach number as well. The unknowns $\vec{\beta}$ are found by a least-square method using the input and output data of the training maneuver. In more detail, Equations 2- 3 could be written as:

$$Y = \bar{\beta}.X + e \tag{4}$$

where Y is the output vector, containing all C_j data at each time step of training signal simulation. $\bar{\beta}$ is a vector of model parameters and X is a matrix of input vector at each time step of training signal simulation. e is the error vector between actual output and assumed model. The best set of β parameters that minimize the error are found as:

$$\bar{\beta} = \left(X^T.X\right)^{-1}.X^T.Y \tag{5}$$

B. Artificial Neural Network

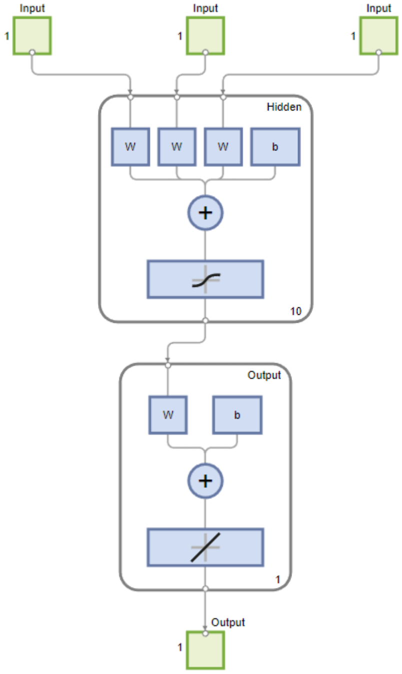
A feed-forward neural network (FFNN) was evaluated at USAFA to model the surface pressure data and aerodynamic coefficients of lift, drag, and pitch moments based on input signals. A FFNN, is one of the simplest and most-used types of artificial neural networks. FFNN's are simple and easy to implement. ANN was developed to mimic the human brain. ANN contains interconnected neurons arranged in different layers of 1) an input layer, 2) one or more hidden layers, 3) and an output layer. The information moves forward from input to hidden layers, and to the output layer. Each neuron or node, calculates a weighted sum of its inputs and processes the result through an activation function. The input parameters are fed into the input layer, which is the network's first layer. While multiple inputs can be used, it is preferable that the input data are not correlated.

Hidden layers are situated between the input and output layers. These layers contain neurons that perform a linear transformation on the input followed by a non-linear activation function. There can be one or more hidden layers, each with a varying number of neurons. Common activation functions include ReLU (Rectified Linear Unit), sigmoid, and tanh. The final layer, the output layer, generates the network's output. The architecture of the FFNN used in this study is illustrated in Figure 1 for constant-Mach PRBS signal.

The network has three inputs (α in radian, q normalized pitch rate in 1/rad, and $\alpha.q$). For the PRBS signal with varying Mach number, additional input for Mach was added.

In more detail, FFNN consists of a single hidden layer with 10 neurons. The Levenberg–Marquardt algorithm was employed for training. Inputs are transmitted from the input layer to the neurons in the hidden layer. The signals are weighted by coefficients ω, combined with a bias term b_k, and then passed through a nonlinear activation function. The transformed signals are then forwarded to the output layer.

Fig. 1 FFNN used for training PRBS signal at constant Mach.



In this study, three different FFNN were trained corresponding to three different outputs for lift, drag, and pitch moment coefficients. For surface pressure data, a single FFNN was trained for each spanwise position with output consisting of pressure coefficient data at different chordwise positions.

C. Surrogate-Based Recurrent Framework

An auto-regressive modeling approach to account for unsteady aerodynamic effects in predicting time-series of the aerodynamic force and moment coefficients is employed. In order to implement a nonlinear mapping function which approximates computational expensive simulations into the SBRF modeling approach, we rely on a GPR model (see Rasmussen and Williams [36]). A software package developed by DLR, the Surrogate Modeling for AeRo-Data Toolbox in python (SMARTy) [37], is used to integrate and apply the SBRF modeling approach. SMARTy provides various building blocks, such as regression and dimensionality reduction techniques, in a single software package. A detailed description on the development and application of the SBRF modeling approach can be found in [38]. In the following, we briefly introduce the model architecture and its application specifics for this work.

Considering nonlinear and unsteady aerodynamics as a discrete-time dynamical system [39]

$$\begin{aligned} \mathbf{x}_{t+\Delta t} &= f(\mathbf{x}_t, \mathbf{u}_t) \\ y_t &= h(\mathbf{x}_t) \end{aligned} \tag{6}$$

with $\mathbf{x}$, $\mathbf{u}$, y, and $h(\mathbf{x})$ denoting the state variables, external inputs, output quantity, and a mapping function of the system states to the output, we are able to derive an equivalent input/output relationship. This means, the output quantity of interest at any time instance t_k depends on both, the instantaneous input at time t_k and the input time history. Let the time t_{k+1} at one-step-ahead be defined as $t_{k+1} = t_k + \Delta t$ with k denoting the current discrete time step and Δt the constant time step size. According to [14], the input/output relationship accounting for time-delay history is written as

$$y(t) = \Phi(y_{k-1}, \ldots, y_{k-n}, u_k, u_{k-1}, \ldots, u_{k-m}) + \varepsilon(\mathbf{x}) \tag{7}$$

with a nonlinear mapping function Φ as a function of external inputs at the current and previous time steps, so-called delayed input and output quantities. As nonlinear mapping function a GPR model is fitted to the training data to relate integrated coefficients for lift, drag and pitching moment ($C_L(t)$, $C_D(t)$, and, $C_m(t)$) to the model inputs. Note that, a quasi-steady GPR model is fitted to initialize delayed states of the input vector for time-series prediction. Thus, the quasi-steady GPR is inherently built and allows a comparison of a quasi-steady GPR model (QS-GPR) with the unsteady SBRF modeling approach. The SBRF model provides recursive one-step ahead predictions to obtain time-series data. For static predictions, a Cauchy convergence controlled time-series prediction at fixed input state is employed. Stability derivative estimates are calculated based on a least-squares approach from predicted time-series of sinusoidal reference motions.

D. ePOD-LSTM

When pressure distributions are highly detailed, the training of neural networks (NN) for their representation becomes computationally inefficient, even when including Auto-Encoder/Decoders (see Fresca and Manzoni [40]). Therefore an alternative is often used, in which a reduced basis is first established using a truncated POD or a similar order-reduction technique, followed by the modeling of the resulting time-variant mode amplitudes using a Neural Network (NN). For the latter, Recurrent Neural Networks (RNN) are often used. This approach is well described by Mohan and Gaitonde [41], as well as by Catalani [42] and Bourier [43], who consider the representation of subsonic pressure distributions using truncated POD basis combined with Long Short-Term Memory (LSTM) RNN. An additional benefit to this approach is that prediction errors can be distinguished into projection errors, i.e. a limitation from the selected truncated POD basis, and network errors resulting from e.g. inadequate training input or network design. Moreover, the POD basis provides a boundary condition on the predicted pressure distribution.

The representation of transonic flows however, is more challenging due to the presence of discontinuities. The accurate representation of discontinuities using a truncated POD, for example, requires including a large number of relatively low-energy modes. Consequently, POD-LSTM Reduced Order Models (ROMs) must learn to describe a large number of mode amplitudes across all training datasets. This requirement escalates the training complexity of the NN, not least because as observed by [43], an accurate prediction of temporal behaviour over a range of frequencies becomes more challenging as the number of POD modes increases.

To address this problem, an alternative approach employing an enriched Proper Orthogonal Decomposition (ePOD) is introduced here. In the ePOD, discontinuous enrichment modes are added to the reduced-order basis to represent the discontinuous parts of the pressure distribution. This allows the remainder of the data to be represented using a standard truncated POD. This dramatically lowers the number of modes needed to accurately represent pressure distributions in transonic flows. As in [42] and [43], the time variant parameters of the resulting basis are modelled using an LSTM. The two main phases of this ePOD-LSTM approach are described below.

1. Model construction and training

Enriched Proper Orthogonal Decomposition The representation of a pressure distribution using a combination of POD and enrichment modes is written as:

$$C_P = \sum_{k=1}^{r} a_k(t)\phi_k(x)^T + \phi_e(x, p(t)) + \bar{C_P} \tag{8}$$

Where $\phi_k(x)$ are the spatial modes with corresponding amplitudes $a_k(t)$, and $\phi_e(x, p(t))$ are the enrichment modes, with time-variant parameters $p(t)$. $\bar{C_P}$ is the time-averaged pressure distribution, which is defined separately to allow the model to focus on the prediction of fluctuations.

The first step in defining the enrichment functions is to obtain a map of the discontinuities. Here physics-based sensors are employed. Specifically, the gradient of the pressure fluctuations is monitored and flagged if it exceeds a specified limit (default: $\frac{\partial C_P^*}{\partial x} >= 1$). It is important to note that the raw data is first passed through a low-pass filter to smooth out high-frequency noise in the data set, to prevent it affecting the efficiency of the shock sensor [44]. Once all the locations flagged by the sensor are obtained, the shock centers are defined as the points of maximum pressure gradient, as discussed in [45].

The next step is to define the enrichment domains, i.e. the local regions in space and time within the data set where the enrichment function will be used. This is done with a user-defined constant which defines how many CFD mesh points before and after the shock location will be included in the enrichment domain. The enrichment functions are then fit to the pressure fluctuation data within the enrichment domains, as illustrated in Figure 2.

First, a target function is defined which smoothly interpolates the C_P^* values at the start and end of the domain (here a linear function is used). The enrichment function of the form described in 9 is then fitted to the test function, which represents the difference between the C_P^* data and the target function. The parameters $p(t)$ of this enrichment function are the amplitudes and locations of two interior control points: $\{a_1(t), a_2(t), x_1(t), x_2(t)\}$:

$$\phi_e(x, p(t)) = \begin{cases} \frac{a_1 x}{x_1}, & \text{for } x < x_1 \\ \frac{(a_2 - a_1)x}{x_2 - x_1} + \frac{a_1 x_2 - a_2 x_1}{x_2 - x_1}, & \text{for } x_1 < x < x_2 \\ \frac{-a_2 x}{1 - x_2} + \frac{a_2}{1 - x_2}, & \text{for } x > x_2 \end{cases} \tag{9}$$

Here, the fit is determined using a non-linear least-squares algorithm. For the case considered below, a fixed number of two enrichment domains is used to represent the two shocks present in the data.

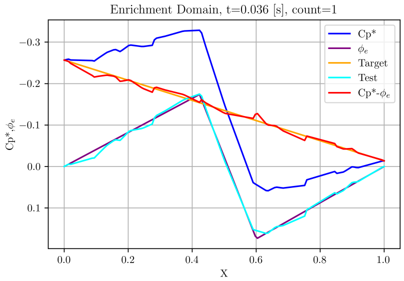


Fig. 2 Enrichment function fitting.

Once the reduced basis is obtained, values of $a_k(t)$ and $p(t)$ are determined using the $C_P^*(t)$ data. These must be matched to suitable input vector values. For the case under consideration, the input signal in the angle of attack-pitch rate sample space suggests that the derivatives of these parameters might significantly influence the instantaneous pressure distribution. Thus, a time-variant input vector is defined to include the angle of attack (AoA), the first and second time derivatives of AoA, the pitch rate q, and the first time derivative of q. This input vector and the $a_k(t)$ and $p(t)$ values form the data set used for the LSTM training.

LSTM Neural Network Recurrent Neural Networks (RNNs) [46] improve on feed-forward neural networks by incorporating the output of adjacent time steps, introducing a temporal dimension to the model [47]. The network's edges that connect neighboring time steps are named recurrent edges. RNNs operate on sequences of data (e.g., time-series, time coefficients), and their weights are determined through backpropagation through time [48]. Long Short-Term Memory (LSTM) network was introduced by Hochreiter and Schmidhuber [49] to address the issues of vanishing or exploding derivatives and short transmission of information in standard RNNs. The LSTM architecture replaces the hidden layer of a standard RNN with a memory 'cell.' Each memory cell contains a node with a self-connected recurrent edge of fixed weight, creating paths through time where gradients can flow without vanishing or exploding [47, 50]. A significant improvement is to make the weight on this self-loop depend on the context instead of being fixed, as proposed by Gers et al [51]. Previous studies conducted at NLR by Catalani [42] and Bourier [43] have demonstrated that LSTM Neural Networks outperform other ANN or regression models in terms of accuracy for the direct prediction of POD time coefficients. Furthermore, Bourier [43] conducted a sensitivity analysis to determine the optimal hyperparameters for the baseline model. The outcomes of this analysis, combined with the findings from Catalani's study [42], were used to

Table 1 LSTM Neural Networks Hyperparameters.

Hyperparameters	Values
Number of LSTM layers	2
Number of LSTM units	64
Number of dense layers	1
Number of dense units	128
Number of time-steps	10
Batch size	32
Drop-out rate	0.2
Model optimization method	ADAM
Model loss function	Custom

define the optimal hyperparameters for LSTM training in predicting the pressure distribution over different sections of the DLR-F22 model. For more details on how and why these values were derived, please refer to the respective thesis reports. The main hyperparameter values for the LSTM surrogate model are presented in Table 1.

The main difference between the previous neural network parameters and the current ones lies in the loss function. The custom loss function is a modification of the function proposed by Giovanni in [42], integrated with the enrichment function. Specifically, the loss function used in training the LSTM neural network is the Mean Square Error (MSE) between the projected and predicted pressure distributions. The predicted time coefficients $\{\hat{a}_k\}_{n=1}^{r}$ are multiplied by the POD modes, and the predicted parameters $\hat{p}$ are introduced to the enrichment function 9. Consequently, the predicted pressure distribution is constructed according to 8. It is then compared to the ePOD expansion of the true targets $\{\hat{a}_k\}_{n=1}^{r}$ and p, as shown in Equation 10:

$$\mathscr{L} = \frac{1}{N_x}\sum_{i=1}^{N_x} || \sum_{k=1}^{r} a_k(t)\phi_k(x)^T + \phi_e(x, p(t)) - \sum_{k=1}^{r} \hat{a}_k(t)\phi_k(x)^T - \phi_e(x, \hat{p}(t))||^2 \tag{10}$$

After defining the architecture of the LSTM and constructing the neural network, the final step of this stage is training the network. Considering the dataset structure and the limited overlap between the available maneuvers, we implemented the following training strategy: The Schroeder maneuver, selected for training (for details, see IV), is partitioned into three datasets: training, validation, and testing. Specifically, the first 700 points are used for training, the next 250 points for validation, and the final 50 points for testing.

2. *Testing stage*

In this stage of the ePOD-LSTM model, the trained LSTM neural network is employed to predict the time-variant parameters of the reduced basis for new, unseen datasets. Specifically, the model forecasts the normalized time coefficients for the ePOD modes identified during the construction of the reduced basis, as well as the parameters for the enrichment function. Using the reduced-order basis defined in the construction stage (refer to 8), the pressure distribution can be reconstructed with the de-normalized coefficients and parameters. A visual representation of the ePOD-LSTM model workflow represented in Figure 3.

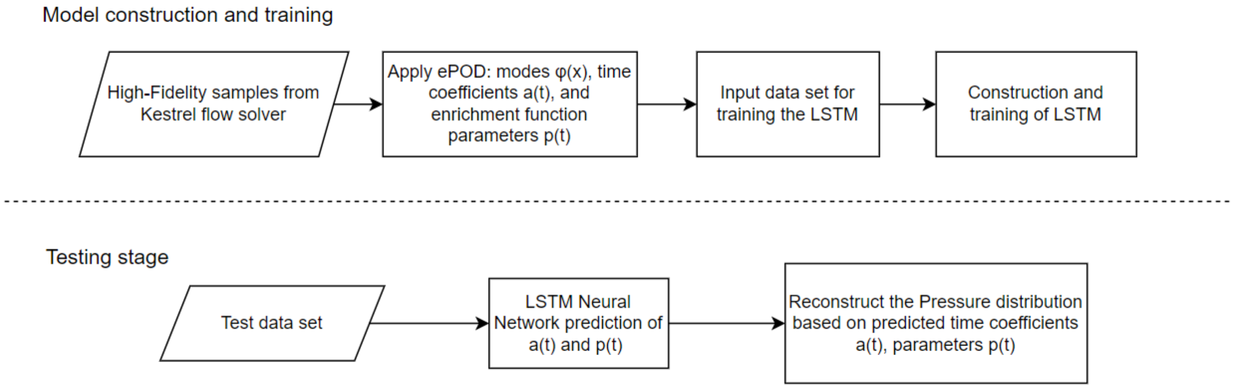


Fig. 3 The architecture of the ePOD-LSTM Reduced Order Model.

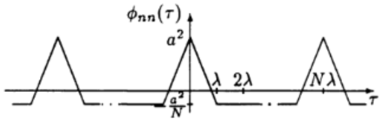
IV. Training Signal Design

As mentioned earlier, the first step toward system identification is system excitation for data collection. A MATLAB code was developed at USAFA to create different signals. These signal need mean, frequency, duration and amplitudes of inputs such as angle of attack. Pitch angles could be varied independent of angle of attack. In addition, Mach number could be constant or changing during signal time. Minimum and Maximum Mach number values are input and the Mach could change in form of linear, quadratic, probablic, or optimal sinusoidal (Schroeder). Plots of input signals, input space coverage, and power spectral density are provided. For quasi-steady signals, the reduced frequency $\kappa = \omega.c/(2V)$ where $\omega = 2\pi f$, c is reference length and V is freestream velocity. The MATLAB code then writes input motion data for use in Kestrel software.

Motion types include: Chirp (a signal with linearly increasing frequency in time), Schroeder (optimal frequency sinusoidal), Sinusoidal, Random, PRBS and Step signals. Some of these signals could have either a constant or a varying mean.

In this paper, two PRBS signals are used. PRBS is a periodic signal in which, the angle of attack and pitch rate change in a periodic and deterministic manner characterized by white-noise-like properties. Typical PRBS signals include sudden input variations between two distinct values, such as minimum and maximum angles of attack. However, the signals used in this paper were modified to have the step changes depend on time.

Fig. 4 PRBS signal characteristics [52].



A PRBS signal example is shown in Figure 4. The maximum possible period for a maximum length sequence N is:

$$N = 2^m - 1 \tag{11}$$

where m is the order of the PRBS. In Fig. 4 of Ref. [52], λ is the shifting time or the duration of shortest impulse, and a is the PRBS amplitude. λ and m should be carefully selected; one criterion here is to have reduced frequency less than 0.01 for a quasi-steady state assumption. The power spectral density of the PRBS signal shows that all frequencies up to $\omega_{max}\lambda/2 = \pi/4$ are excited. ω_{max} is related to the shorter time constant T_{min} by:

$$\omega_{max} = \frac{3}{T_{min}} \tag{12}$$

therefore $\lambda = \pi/6T_{min}$. In this study, a PRBS signal was designed for 4 seconds with mean angle of attack of 10° at constant freestream Mach number of 0.85. The amplitude a increases linearly from 0 to 10° for the first half of motion and then linearly falls to 0 for the second half of the signal. Number of shifts was set to 40. The designed PRBS signal, named PRBS1 in this paper, and its input parameters are shown in Figures 5(a),(c),(e). PRBS1 signal was designed for the FFD test case at Mach 0.85. Figure 5(a) shows the angle of attack variations with time. Note that the vehicle is set at a wind vector with 10° angle of attack. Pitch angle is zero and it varies as $\alpha(t) - 10$ where $\alpha(t)$ values are given in Figure 5(a). In this way, the pitch rate and the time-rate of changes in angle of attack are the same. Figure 5(c) shows angle of attack in degrees versus pitch rate in deg/s (Q). Note that the signal maximum frequency and hence the pitch rate was limited to have a maximum reduced frequency of 0.01 for quasi-steady aerodynamic behavior. Figure 5(c) shows that the PRBS signal has an excellent coverage of $\alpha - q$ input space. Figure 5(e) shows the $\alpha - \ddot{\alpha}$ space coverage of the PRBS1 signal. $\ddot{\alpha}$ data are given in $^\circ/s^2$ units.

Figure 5(e) shows a large range of changes in $\ddot{\alpha}$, though the most points are located at the center of plot.

Figure 6, shows the PSD analysis of the signal and its periodogram. As anticipated, the PRBS signal excites all frequencies equally. In addition, Figure 6 shows reduced frequency values for the signal. Maximum reduced frequency values does not exceed 0.01 to ensure quasi-steady aerodynamic assumption.

Following this, a new PRBS signal of similar design was generated. Unlike the previous signal, the Mach number in this one is not steady at Mach 0.85 but changes between 0.1 and 0.9. For Mach variations, a Schroeder signal was designed. This signal was generated from optimization of amplitude and frequency spectra of multisines. The frequency range was selected to have a reduced frequency range of 0.002 to 0.01.

According to Morelli [53], a Schroeder signal has excellent frequency content and a low peak factor, a measure of the ratio of maximum input amplitude to input energy. In comparison to a Chirp signal, Schroeder provides better input for frequency domain dynamic model identification. The new PRBS signal is named PRBS2 in this paper and is shown in Figs. 5(b),(d),(f). Another difference with PRBS1 is the that PRBS2 signal duration was extended to six seconds as shown in Fig. 5(b). Because of a longer duration, the pitch rate values are smaller compared with the PRBS1. This is shown in Fig. 5(d). The Mach number follows a Schroeder signal which has a series of multisines to cover the Mach range of 0.1 to 0.9. Mach number variations are shown in Fig. 5(f).

V. Test Case Description

The test case in this study is the FFD model, which is a generic triple delta wing configuration. Its planform is based on the baseline DLR-F22 wind tunnel model [54] as part of DLR studies on the technology assessment and a design approach of a Future Fighter Demonstrator [55]. A revision of the wing design of the DLR-F22 model by ONERA within AVT-351 [20] led to ONERA_DLR_M421 wind tunnel model which serves as FFD use case in this study [56]. In comparison with the baseline DLR-F22 model a moderate thickness, positive camber, and twist distribution was added. This model design was utilized to investigate flight dynamic characteristics in multiple static and dynamic wind tunnel experiments at ONERA. Aircraft with multiple swept leading edges exhibit rapid vortical flow topology evolution and intense flow unsteadiness throughout their flight regime. Sweep angles of the triple-delta wing are 75, 45 and 75 degrees. Some associated phenomena are vortex-interactions, vortex-shock interactions and vortex breakdown that influence stability and controllability [57, 58].

Table 2 Flow conditions for the FFD wind tunnel model.

case	α [deg]	M_∞ [-]	p [Pa]	T [K]	Re_∞ [-]	l_{ref} [m]	A_{ref} [m^2]	$\mathbf{x}_r$ [m]
PRBS1	[0-20°]	0.85	49881	266.5	$\approx 3 \times 10^6$	0.272592	0.0807265	0.212833/0/0
PRBS2		[0.1 - 0.9]						

Table 2 shows the flow conditions for the two PRBS signals. Flow simulations of PRBS1 signal are performed at transonic conditions, $M = 0.85$, and an ambient pressure of 49 881 Pa, a temperature of 266.5 K, resulting in a Reynolds number of about three million.

Figure 7(a) displays the hybrid computational grid on the half model, consisting of about forty million nodes and about 150 000 elements. The boundary layer is resolved with quadrilaterals, while tetrahedra extend from boundary layer edge to the farfield. To better capture vortices, a refined region was specified above the model as shown in the figure. The first off-body grid-nodes all satisfy $y^+ < 1.0$ in sublayer scale. The farfield distance is set to one hundred times the chord length.

In addition, tap points were defined for six spanwise locations over the upper surface of the FFD. These tap points are shown in Fig. 7(b) . Note that three locations were chosen to be near the wing kink locations. The y positions of these tap points are [0.05, 0.09, 0.136, 0.18, 0.22, 0.28] m. These points move with the mesh. Output data include tap coordinates and pressure coefficient data at every 200 time step. Note that not all points are exactly located on the surface. The Kestrel option of "Closest Solution Value" was used. However, Kestrel reports a "Found" parameter with values of 0 or 1; 0 means no solution found (e.g., the point is perhaps outside the mesh domain and located inside the plane). Considering the points with a Found value of 1, there are 237, 293, 189, 161, 123, and 89 points for slices 1 to 6, where slice 6 is the one near the wingtip.

Experimental data at subsonic speeds and conducted at ONERA test facilities are available for validation of numerical simulations and to guide reduced order modeling activities. Farcy et al. [56] detailed the experimental data. The wind tunnel model has a span of 0.6m. Two set of experiments were conducted: static and dynamic. Static runs had a freestream Mach number of 0.1733. Mach number of dynamic tests was set to 0.1 Total temperature was 305K with a total pressure of 100,000 Pa. Static runs cover -20/+60 angle of attack range at zero and nonzero sideslip angles. Pitch oscillation tests were conducted at mean angles of 0 to 30° with a 5-degree amplitude and frequencies of 1,2, and 3 Hz. The pitch rotation point is 0.447 from the model's nose. The wind tunnel model has a rear-mounted sting and was placed on an internal six-component load balance.

VI. Results and Discussions

All simulations were run using the Kestrel flow solver. Second-order spatial and temporal accuracy was used. The grids are half geometries with far-field, symmetry, and no-slip wall boundary conditions for the airplane. All solutions begin at a 10° angle of attack, corresponding to the mean aerodynamic angle of PRBS signals. For PRBS1, the free-stream Mach number was set to 0.85 with a total temperature of 305 K and a total pressure of 305 Pa. For the PRBS2 signal, the freestream Mach number was set to 0.4. The Mach number was then varied using a prescribed-body motion in Kestrel. In this approach, if the grid moves opposite to the incoming velocity, the relative velocity between the freestream and vehicle will increase, and hence the Mach number. If the grid moves in the direction of the freestream velocity, the Mach number will decrease. The relative velocity is set so the Mach number matches the profile shown in Fig. 5 (f).

All motions were run with a time step of 0.0002 seconds and eight Newton subiterations. The motions begin after 2,000 startup iterations and continue for an additional 20,000 iterations for a signal duration of four seconds. Advective temporal damping was set to 0.01. The Spalart–Allmaras turbulence model with rotational/curvature correction (SARC) was used. The cost of running these signals is approximately 120,000 CPU hours using 2,880 processors. Simulations were run on the U.S. DoD HPCMP Carpenter machine, which is an HPE Cray EX4000 system located at the ERDC DSRC in Mississippi. It has 1,440 standard compute nodes, 4 large-memory nodes, and 8 GPU nodes (a total of 278,272 compute cores). It has 518 TB of memory and is rated at 15.75 peak PFLOPS.

A. SID Modeling at Constant Mach 0.85

First results show the SID (Regression), FFNN, and SBRF predictions using the PRBS1 signal (the signal at constant Mach 0.85). This signal was run in Kestrel with predicted C_L, C_D, Cm values are shown in Fig. 8. For referencing, Fig. 8(a) shows the input data (angle-of-attack time history) as well. Note that SID, FFNN, and SBRF modeling predictions depends on the chosen time step, or more specifically, by the non-dimensional time step, $t^* = t.V_\infty/c$, where t is time step in second. Ideally, we would like to keep t^* about 0.01, however, this makes simulations very expensive to run. A time step of 0.0002 seconds was chosen for the FFD simulations at Mach 0.85. Figure. 8 shows a nonlinear aerodynamic behavior with respect to the angle of attack and pitch rate.

The predicted coefficients shown in Fig. 8 are then rearranged according to Eq. 2 and then Eq. 5 was used to estimate the model parameters, i.e. $\beta_j, j = [0-7]$. The parameter estimation took about one or less than one second. In addition, three different FF neural networks were trained corresponding to C_L, C_D, C_m data of the PRBS1. Networks have one hidden layer containing 10 neurons. Levenberg-Marquardt algorithm was used. Network training time was less than one minute. For the SBRF model, a reduced training set was used to reduce computational cost and overfitting of the GPR model due to spatially highly correlated sample locations. The sample points were randomly split into a smaller training set consisting of about 10 % of the initial size of the PRBS signal. Time-delay quantities were still computed based on the full time-series data accounting for the original time step size. Here, no time-delayed input was considered for the two exogenous input variables α and q, but time-delayed output information is used instead to augment the input matrix with output quantities obtained at $t_k - \Delta t$ with $\Delta t = 0.002$ sec. This corresponds to a time step at t_{k-10} with k denoting the current time step. For model prediction, then $C_{j,t_{k-10}}$, for $j = L, D,$ or m, is recursively fed back as model input from a single time step at $k - 10$ with k denoting the current time step. In addition, the quasi-steady GPR (QS-GPR) model which is inherently built when using the SBRF modeling approach is compared to the other models. Computational cost to generate the SBRF model (including the QS-GPR model) on the reduced sample set was less than a minute on a desktop computer.

Regression, FFNN, SBRF models were then used to reconstruct the PRBS1 signal. These model predictions of the PRBS1 signal are shown against CFD (actual) data in Figure 9. Overall, a very good agreement was found by all three models at all simulation times. In more detail, the mean squared error (MSE) values of the models are given in Table 3. MSE values are small with best predictions provided by quasi-steady and unsteady GPR models. These models usually provide highly accurate training data reconstruction, since only a small value as regularization term is added to the models. Nevertheless, the choice of only using a subset of sample points as training set, did not lead to an increase of model training errors. Besides these two, the FFNN model provides best predictions for C_D. In the following, we use the QS-GPR model for comparison of model predictions at static and quasi-steady conditions (such as static and quasi-steady damping stability derivative estimates), and the SBRF model for generic, prescribed maneuver predictions similar to the PRBS signal.

Table 3 PRSB1 prediction mean squared error using regression, FFNN, QS-GPR, and SBRF.

MSE	Model	C_L	C_D	C_m
	Regression	0.01658	0.00419	0.005722
	FFNN	0.01458	0.003608	0.004487
	QS-GPR	0.00025	0.00001	0.00003
	SBRF	0.00029	0.00001	0.00003

Up next, we will look at how the regression and FFNN models perform at different angles of attack. These angles were picked to cover the full range of the signal, from the lowest to the highest values. (i.e. 0 to 20°) with an increment of 0.5°. The way regression model is used for these predictions is by setting up the pitch rate to zero in models, i.e.

$C_{jstatic} = reg_j\ (\alpha, q = 0)$. Likewise, in FFNN, the input columns corresponding to the pitch rate and its products (e.g. $\alpha.q$) are set to zero values. Static predictions of the FFD model at Mach 0.85 for the regression, FFNN and QS-GPR models are given in Fig. 10. In addition, static data from Widhlam et al. [59] are used for cross-plot purposes. Figure 10 shows that both, regression and FFNN, predict very similar lift and drag coefficient values and the predictions match well with those shown by Widhalm et al. for the FFD and Mach 0.85. Likewise, regression and FFNN predictions are similar for the pitch moment coefficients with small discrepancies at some angles of attack. The predictions show the same trend as Widhalm et al. presented in Ref. [59]. However, model predictions overestimate the pitch moment data at small angles of attack and did not predict the dip seen around 20 angle of attack in previous studies. In summary, the models do their best at the average angle of attack of the training signal, but their accuracy fades a bit near the max and min values, i.e. [0°-5°] and [15°-20°] range.

QS-GPR predictions shown in Figure 10 are able capture the general trend, but show minor (C_L and C_D) and large oscillations (C_m) around the mean. The SBRF model performed similarly to the QS-GPR model. Root mean squared error (RMSE) evaluated using the static reference data from Widhlam et al. [59] for the QS-GPR model gives values of 0.04140, 0.01112, and 0.01073 for C_L, C_D, and C_m respectively (RMSE using SBRF model gives values of 0.03091, 0.00898, and 0.01201 for C_L, C_D, and C_m respectively). These oscillations indicate an overfitting issue during model fit on the one hand, and that probably the type of training signal is not very suitable in combination with GPR-based models. In particular for the latter, a reason can be found in the large step changes between two distinct angles of attack which are characteristic for the PRBS signal. Though a large range of the angular rate q is covered by the signal, the PRBS signal captures only a small low-frequency or quasi-steady content. For instance, very large values are obtained during the (smoothed) step changes from one angle of attack to another, but only constant values of $5\ ^\circ/\mathrm{s}$ and $-5\ ^\circ/\mathrm{s}$, respectively, are obtained for the portions between the step changes. This seems to cause problems primarily for the GPR-based models.

For predicting the curve slope with the regression and FFNN models, we applied the same method as used for the static case; first static data of $C_{jstatic1}$ are estimated for α angles. Then $C_{jstatic2}$ are found for $\alpha + 0.5°$ angles. The slope is then the difference of these values divided by 0.5° increment. The lift, drag, and pitch moment curve slopes predicted by regression and FFNN models are shown in Fig. 11 and again compared with those reported by Widhalm et al. [59]. Static stability estimates from the QS-GPR were calculated based on a least-squares approach using a time-series prediction of a sinusoidal motion at a frequency of $f = 1$ Hz and an amplitude of 5°. Note, at mean angles of attack equal to 0° 20°, the sinusoidal time-series prediction includes an extrapolation. Those predictions of $C_{j,\alpha}$ for $j = L, D,$ or m are added to Fig. 11.

Overall, the trends are similar but FFNN shows more nonlinearity in the plots than regression models. For drag, all models show the same slope and similar data to Ref. 59. For lift coefficient, the slopes predicted in the range of [5-15]° have a better match with previous data. Note that the input space is mostly covered around the mean angle of attack of 10. For the pitch moment, the FFNN provides a better match with earlier data than the regression model. The regression model exhibits a linear slope with the angle of attack and slightly overestimates in comparison to both the FFNN and previous data. The QS-GPR model also overestimates for angles from 5° to 10° and predicts a rather constant slope.

Next predictions correspond to dynamic derivatives of $CL_q + CL\dot{\alpha}$, $CD_q + CD\dot{\alpha}$, and $Cm_q + Cm\dot{\alpha}$. Note that PRBS signals used in this study have combined effects of changes in pitch rate and angle of attack because $q = \dot{\alpha}$. For isolated terms, pitch angles should be varied independent of the angle of attack. For the convenience we drop $\dot{\alpha}$ terms, and name these dynamic derivatives CL_q, CD_q, and Cm_q.

For prediction of these dynamics terms, models were fed with input of angle of attack and pitch rate of zero $Cj_{q0} = C_j(\alpha, q = 0)$ and at $q = 11/rad$, i.e $Cj_{q0} = C_j(\alpha, q = 1)$. Dynamic derivatives are the difference of these estimated terms. Note that q is non-dimensional pitch rate defined as $q = Q.c/(2V_\infty)$ where Q is pitch rate in rad/s. Regressions and FFNN model predictions of these dynamic derivatives are shown in Figure 12 and compared with estimations using the sinusoidal time-series predictions from the QS-GPR model and those given in Ref. [59]. Overall, good agreement was found with earlier studies of the FFD at Mach 0.85. Dynamic derivatives appear to be relatively unaffected by changes in the angle of attack up to 20°. Note that estimation of these dynamic derivatives could still be a challenging task from CFD and time consuming using traditional pitch oscillation motions.

In addition to static and stability derivative predictions, the created model could be used for prediction of new time-accurate signals with the input space within the training signal used for model creation. One example, is pitch oscillations. Different motions with combinations of the mean, amplitude, and frequency could be defined and then predicted by models. Figure 13 shows regression, FFNN, and QS-GPR model predictions of a pitch oscillation with 10° mean angle, 5° amplitude, and frequency of 1Hz. Lift, drag, and pitch moment predictions are plotted against CFD data. The lift and drag show small thin loops but a nonlinear type. CFD and model predictions match very well. For pitch moments, the hysteresis loop is more visible and is characterized by large nonlinearities at $\alpha = 10°$ and an eight-shape curve around the maximum angle of attack. FFNN, regression, and QS-GPR models show discrepancies in predicting those nonlinearities. Predictions do not match with CFD data at all angles, but are able to reproduce the general trend of the time history for C_m, that was computed with CFD. The regression and FFNN model are generally more smooth compared with the QS-GPR model which is characterized by oscillations for some angles of attack. A reason of this discrepancy might be running CFD case at a different set up (time step, sub-iterations) than the training data, in addition to the previously mentioned poor input signal coverage of the low frequency content.

In addition, FFNN was used to predict different signals such as Chirp motions with constant or varying mean, Schroeder, and a step signal. For the latter two, also predictions of the SBRF model were available. Model predictions are compared with CFD data in Figures 14- 17. Figure 14 shows a chirp signal with mean 10 and amplitude 10 and four-second duration. The initial frequency is 2Hz and linearly increases with time. FFNN predictions match very well with CFD ones. Lift, drag, and pitch moment coefficients show MSE values of 0.026 0.0077, and 0.0087, respectively.

Figure 15 shows another Chirp motion with a varying mean angle of attack. The mean angle follows a sinusoidal curve with 1Hz frequency. The chirp has an initial 4Hz frequency, again linearly increasing with time. FFNN predictions again match very well with CFD data of this signal. MSE values are 0.022, 0.0056, and 0.0075 for lift, drag, and pitch moment coefficients, respectively. The largest discrepancies are seen at pitch moment at large angles and large frequencies.

A Schroeder signal was designed in which the mean angle of attack again follows a sinusoidal motion with 1Hz frequency. This motion was run in CFD with the same set up as PRBS1. Figure 16 compares FFNN and SBRF predictions of the Schroeder signal with time-accurate data. MSE values calculated from the FFNN model predictions are 0.024, 0.0058, and 0.0079 for lift, drag, and pitch moment coefficients, respectively. For the SBRF model, MSE values of 0.00082, 0.00003, and 0.00009 were obtained for lift, drag, and pitch moment coefficients, respectively. A good match between CFD reference and both models is observed for C_D and C_L, whereas the magnitude of C_m during the first two-thirds is not well predicted. This is improved when a better coverage between training data and predicted motion exists, as can be seen for the last third of the time-series at angles of attack below 15°.

Finally, Figure 17 shows a step motion in which the signal begins at 10-deg angle of attack. The aircraft is held at this angle about 0.2 seconds and then suddenly undergo a negative unit step, and angle of attack drops one degree. The aircraft is then at zero angle of attack at two seconds. Then, it will go undergo the same motion but with a positive unit step until it reach 10 after 4 seconds. All shown forces and moments have a jump in coefficients at the steps. FFNN model predicts the static and jumps in the coefficients with reasonable accuracy with 0.0089, 0.0013, 0.0019 MSE values for lift, drag and pitching moment coefficient. The main discrepancies correspond to the step locations. The SBRF model shows a similar performance in terms of error metrics as for the Schroeder signal (MSE values for lift, drag and pitching moment coefficient are 0.00013, 0.000002, and 0.00009, respectively). However, best accuracy using the SBRF model is obtained for C_D, whereas for C_L and C_m at some steps a high frequency oscillation around a constant mean with larger deviations of the predicted mean around the minimum angle of attack are visible. Note that these oscillations are not present when evaluating the QS-GPR model, which achieves an overall similar prediction accuracy of the step motion (MSE values for lift, drag and pitching moment coefficient are 0.00009, 0.000004, and 0.00002, respectively).

B. SID Modeling with Mach Variation

This section focuses on extending aerodynamic models to account for Mach number effects. To achieve this, a new PRBS signal was designed to cover angles of attack ranging from 0 to 20 degrees over six seconds. In this new motion, the Mach number varies between 0.1 and 0.9 following a Schroeder motion, with the initial Mach number set at 0.4. Two regression models were developed: 1) one model similar to the one used for PRBS1 using Eq. 2 with inputs of α and q. The second model is based on Eq. 3 which includes Mach number as an additional input. The predictions of these two models are compared with actual CFD data from running PRBS2 in Kestrel, as shown in Fig. 18. Using Eq. 2 models show root mean squared errors of 0.039, 0.013, and 0.024 for C_L, C_D, and C_m, respectively, with large discrepancies in pitch moment and coefficients at large angles of attack. However, the second model with Mach number as an additional input has MSE values of 0.02 for lift, 0.0067 for drag, and 0.017 for pitch moment coefficient. Additionally, a FFNN model was created to predict lift, drag, and force moments, using seven inputs (α, q, M, $\alpha.q$, $\alpha.M$, $q.M$, $\alpha.q.M$).

Regression models were then used to predict static and stability derivatives of the FFD at different Mach numbers and angle of attacks. Figure 19 shows static and slope data at Mach 0.1 and 0.85. Previous CFD data at Mach 0.85 were obtained from Widhalm et al. [59]. For Mach 0.1, the ONERA wind tunnel data are plotted. Figure 19 shows that static and slope data match well with wind tunnel data at Mach 0.1 for the ranges of shown angles (this range corresponds to the signal input range). Lift and its slope curve underestimate Ref. 59 data at Mach 0.85. The same applies to drag and pitch moment coefficient at Mach 0.85. More interestingly, model predictions at Mach 0.9 (not shown in this paper) match better with previous CFD data at Mach 0.85. A few suggestion to improve model prediction accuracy is to refine signal with a longer duration and including more training data at transonic speeds than subsonic regime.

From the PRBS2 signal, dynamic derivatives of C_{Lq}, C_{Dq}, and C_{mq} were estimated at Mach numbers of 0.1 and 0.85. These predictions are compared against data from Ref. [59] and wind tunnel data [56] in Fig. 20. In addition, the models created from the PRBS2 signal were used to predict a sinusoidal pitch motion at Mach 0.1, with mean angle of 10°, amplitude of 5° and the motion frequency of 1Hz. The model predictions are also compared with wind tunnel data (this corresponds to test run 149 in wind tunnel documents) in Fig. 20. The predictions shows that dynamic derivatives are insensitive to the angle of attack variations up to 20° for both shown Mach numbers. However, dynamic derivative depend on Mach number. e.g. C_pmq becomes more negative with increasing Mach number. Figure 20 shows that model predictions fall in the range of previous and measured data. The model predictions at Mach 0.85 show largest discrepancies. The sinusoidal predictions at Mach 0.1 match very well with wind tunnel data, though ROM performance should be compared with CFD rather than measured data.

Likewise models created for the PRBS1 signal, the new models could be used to predict the responses to new motions even including Mach number variations. As an example, a Chirp motion with linearly increasing frequency with time was defined. This motion runs for four seconds. It has a mean angle of 10°, amplitude of 5° and initial frequency of 2Hz. The Mach number is not fixed and varies between 0.5 to 0.9 following a Schroeder motion. This motion is shown in Figure 21. The $\alpha - Q$ space coverage is also shown in the Figure as well. Model predictions of this motion are compared with time-accurate simulations data in Figure 21 as well. Overall, the predictions match well with CFD especially for lift and drag coefficients. Discrepancies in amplitude and phase are due to slight differences in dynamic derivative predictions. Additionally, three motions were defined at Mach 0.1: Sinusoidal, Random, and Schroeder. These motions are shown in Fig. 22 and have a duration of four seconds, except for the Sinusoidal that has 3.7 seconds. Sinusoidal is a combination of five signals of constant frequency; as time progresses, the signal frequency and amplitude increases in time. Random was generated with different amplitude and frequency combinations. Finally, a Schroeder motion was defined in which the mean angle of attack follows a sinusoidal motion with a frequency of 0.5Hz. The motion data were fed into models and the predictions are compared against time-accurate data in Fig. 22. Again, the overall trends were predicted well with small discrepancies at the peak values.

C. Modeling Surface Pressure Data

1. FFNN

The PRBS1 signal was run again in Kestrel, however, this time with pressure tap points of Figure 7 (b). This includes about 1,092 data points. Note that each slices has different number of points; those near the wing tip has less points than those near root. The tap points only correspond to the upper surface sections. Tap data are defined using coordinates of x,y,z. At every 20 time step, pressure coefficient data at these locations are recorded. Note that angle of attack and pitch rate vary according to the PRBS1 signal shown in Fig. 4. Data were then rearranged to have input data of $[\alpha, q, \alpha.q]$ and corresponding C_p data for each slice. Then FFNN were used to train the models based on these data for each sections. The model predictions of two slices are shown here, slice 1 and 3. These slices are shown in Figures 23 and 24, respectively. In these figures, the PRBS1 signal, the pressure section, and the initial time C_p data are shown. FFNN models took about one hour to be trained due to amount of data. These models were then saved to be used for prediction of pressure data of new signals. Model training data for slice1 and slice3 are shown in Figure 25at different time instants. In this figure, the input signal, section data from CFD and model predictions are shown. Depending on the angle of attack, pitch rate, and slice locations, shock waves formed over the upper surface at different chordwise locations. Figure 25 shows that models were trained to capture the shock waves and pressure coefficient values with the best accuracy. Note that Slice 3 has less negative pressure data at the upper surface than slice1 for high angles of attack. The shock positions are visible with sudden change in Cp values especially for angles of 10 and higher. The shock becomes stronger and moves aftwards with increasing angle of attack.

The NN models were then used to predict the pressure data of a step motion shown in Figure 17(a). Note that tap data were predicted at the center of each segment; it was assumed that these data correspond to static data as the pitch rate is zero during each step segment. Model predictions at different angles of attack at slice sections of 1 and 3 are shown in Figure 26. C_p predictions match very well with data calculated from time-accurate simulations.

2. ePOD-LSTM

Projection and training error As previously discussed, the initial stage involves constructing a reduced-order basis using the proposed enriched Proper Orthogonal Decomposition (ePOD) technique. To demonstrate the advantage of this approach over the standard method in modeling transonic flow regimes, we assess the Mean Squared Error (MSE). Specifically, we compute the projection error of the ePOD technique and compare it to the corresponding error of the standard POD method using an equal number of POD modes, as well as to the total of POD modes and enrichment parameters. For instance, in the case of ePOD utilizing 10 modes and 8 enrichment function parameters (4 parameters per shock), it is compared to the standard mode using 10 modes, and to the standard method using 18 POD modes.

As observed in Figure 27, the Mean Squared Error (MSE) for the ePOD method is consistently lower than that of the standard POD with an equivalent number of parameters across all time steps in the PRBS1 dataset. Specifically, the average MSE for the projection error of the ePOD method is $2.25 \cdot 10^{-4}$, whereas for the standard POD with the same number of parameters, it is $2.42 \cdot 10^{-4}$ and for the same number of modes, it increases to $5.74 \cdot 10^{-4}$.

Furthermore, in the case of the Schroeder maneuver, the MSE of the ePOD method generally outperforms that of the standard POD. On average, the MSE is $2.78 \cdot 10^{-4}$ for ePOD, $3.38 \cdot 10^{-4}$ for POD with the same number of parameters, and significantly higher at $8.62 \cdot 10^{-4}$ for standard POD with the same number of modes. This pattern holds true except for three specific instances in time. Further investigation into these time steps reveals that the fitting error was significant in these cases, caused by high oscillations following the shock discontinuity, which resulted in underfitting of the previously described fitting methodology, as observed in Figure 28. Moreover, the MSE is obtained by averaging the error for the pressure distribution at each time instance. Consequently, small pressure fluctuations within the dataset contribute to this error. The proposed method is specifically designed to handle shock discontinuities, whereas the standard POD method with more modes may better address these small pressure fluctuations in regions of the dataset where no pressure discontinuity occurs. In addition, the pressure distribution for selected time instances for the PRBS1 signal and Schroeder maneuver are illustrated in figures 30 and 31 respectively. It becoming apparent that the proposed method it is performing remarkably well in the discontinuity region outperforming the standard methods and facilitating the application of the model for the prediction of the pressure distribution in the transonic flow regime.

The following results concern the training stage of the neural network. Figure 29(a) presents the weighted training and validation loss for the Schroeder maneuver. In Figure 29(a) 95% of the weighted version, is the contribution of loss function 10 and 5% from the mean square distance between the predicted and true time coefficients and parameters. The validation loss reaches a plateau of approximately $4 \cdot 10^{-3}$ after the first 400 epochs, while the training loss continues to decrease, eventually reaching a minimum of about $4 \cdot 10^{-4}$. Furthermore, the normalized time coefficients and the parameters of the enrichment function are depicted in 32. The normalized first and last time coefficients, along with the parameters of the second shock (the strongest and most significant one) predicted by the neural network, are compared with the actual validation data. A particularly interesting observation relates to the frequency and periodicity of these data. By comparing the first and last time coefficients, it can be derived that as the number of POD modes increases, the frequency of the corresponding time coefficients also increases. Consequently, it becomes more challenging for the neural network to predict these highly oscillating coefficients. Thus, by utilizing the ePOD reduced-order basis, the total number of predicted parameters decreases, and the convergence of the neural network improves.

Testing error In this paragraph, the results from the testing stage of the ePOD-LSTM Reduced Order Model are presented. The predicted pressure distribution is compared with both the projected and true pressure distributions. The Mean Squared Error (MSE) between the projected-predicted, true-projected, and true-predicted pressure distributions for the last 30 time steps of the testing dataset is illustrated in Figure 29(b). The error between the true and projected values represents the maximum level of accuracy that the neural network can achieve. Essentially, an optimally designed and perfectly trained neural network would achieve a similar level of error as seen in the projection. Moreover, the MSE of the projected-predicted and true-predicted values shows comparable trends. A maximum error of $6 \cdot 10^{-3}$ occurs at time step 12, while the average error for both comparisons remains below $3 \cdot 10^{-3}$. Furthermore, Figure 33 displays the predicted, projected, and true values of the pressure distribution for section 1 at selected time steps. These plots demonstrate that, despite using a limited training dataset, the ePOD-LSTM model accurately predicts the true pressure distribution, even in challenging discontinuity locations.

VII. Conclusions

Using CFD for aerodynamic system identification of air vehicles is an active area of research. This study investigates the use of different SID techniques to estimate the integrated forces and moments and surface pressure data of a generic fighter configuration at transonic speeds. Aerodynamic reduced-order models were created from two pseudorandom binary sequence (PRBS) training maneuvers. During these maneuvers, the angle of attack and pitch rate change in a periodic and deterministic manner which is characterized by white-noise-like properties. Typical PRBS signals include sudden step variations between two distinct values, such as minimum and maximum angles of attack. However, the signals used in this paper were modified to have the step changes to depend on the simulation time. In the first motion, the aircraft undergoes a signal at a constant Mach number of 0.85. In the second motion, the Mach number varies in an optimized manner from 0.1 to 0.9. Different reduced-order methods were applied, that comprise regression, feed-forward neural network and auto-regressive surrogate modeling techniques to predict integrated force and moment coefficients and a proper-orthogonal decomposition based neural network approach for surface pressure prediction. Once models of integrated forces and moments were created, they were used to predict static and stability derivatives at different angles of attack and Mach numbers. Models were then used to predict aerodynamic responses to arbitrary motions including pitch sinusoidal, chirp, Schroeder, and step. Model predictions were compared with actual CFD data. Overall, a good agreement was found for all models. Models to predict surface pressure data were also able to accurately predict the upper surface pressure data at different spanwise and chordwise locations at different angles of attack for both static and dynamic runs.

The results indicate the need for a careful and model-specific selection of training signals and require a more detailed analysis in particular with respect to the GPR-based models, i.e. the quasi-steady GPR and unsteady SBRF models that both suffer from overfitting. Even though the design of generic training signals in combination with a SBRF model for the same test case has been successfully demonstrated at subsonic conditions in [38]; for this transonic scenario only limited applicability could be shown. However, the new PRBS signal with sudden step-like changes, as used in this work, differs to those multi-sine and frequency sweep signals of previous work. Future work will focus on an extended research on and assessment of input signal design in conjunction with an application of unsteady modeling techniques at transonic, but also varying Mach numbers.

A machine learning-based surrogate model was constructed to predict the surface pressure distribution in the transonic flow regime. This model combines enriched Proper Orthogonal Decomposition (POD) and a Long Short-Term Memory (LSTM) neural network. The enriched POD basis is proposed to better capture discontinuities in the pressure distribution which arise due to the presence of shocks and are not captured by a standard truncated POD basis. The enriched POD method outperforms the standard method using an equal number of POD modes, as well as a number of standard POD modes that equals the total of POD modes and enrichment parameters.

After deriving the reduced-order basis, the construction of the LSTM neural network follows. The network's parameters are based on previous studies conducted at NLR [42, 43], with some modifications to integrate the enriched basis. The Schroeder maneuver is divided into subsets used for training, validation, and testing datasets. The training loss of the LSTM reaches a minimum of about $4 \cdot 10^{-4}$, while the validation loss is approximately $4 \cdot 10^{-3}$. An interesting observation is that as the number of POD modes increases, the frequency of the time coefficients also increases, making it more difficult for the neural network to predict these highly oscillating fluctuations. Hence, the enriched basis not only reduces the total number of parameters that need to be predicted but also improves the convergence of the neural network.

The ePOD-LSTM model accurately predicts the time coefficients and enrichment parameters of the ePOD reduced-order basis for the test dataset. The average prediction error is less than $3 \cdot 10^{-3}$, with a maximum error of $6 \cdot 10^{-3}$. These results demonstrate that the proposed enriched Proper Orthogonal Decomposition (ePOD) method efficiently represents discontinuities in the pressure distribution, thereby facilitating the implementation of the surrogate model in the transonic flow regime. By employing the ePOD-LSTM surrogate model, the number of predicted parameters is reduced, which results in improved model convergence and a more accurate prediction of the pressure distribution.

The future work will extend these results to include new signals based on the optimization of the Fisher information matrix (FIM) which involve creating signals that maximize the amount of information that can be extracted from a system or process. New signals with different Mach variations will be tested. The effects of neural network training algorithms will be investigated. Finally, techniques based on POD will be examined to model the flowfield around the FFD configuration.

VIII. Acknowledgments

This paper has been approved for public release with unlimited distribution; Distribution Unlimited; PA Number USAFA-DF-2024-515.

Mehdi Ghoreyshi and Pooneh Aref sponsorship is through the US Air Force Academy under agreement number

FA7000-20-2-0014. Flow simulation material presented in this paper is a product of the HPCMP CREATE^{TM}-AV project sponsored by the U.S. Department of Defense HPC Modernization Program Office. The authors would like to acknowledge the support received from the CREATE-AV Quality Assurance team. All CFD simulations were run at Carpenter machines at the ERDC DoD Supercomputing Resource Center (DSRC).

Mario Stradtner would like to thank the German MoD and The Federal Office of Bundeswehr Equipment, Information Technology and In-Service Support (BAAINBw) for their support of the military research at DLR and the support to attend the NATO/STO AVT Task Group meetings.

Michel van Rooij would like to acknowledge the support from the Knowledge & Innovation branch of the Netherlands Ministry of Defence (grant number L2222).

References

[1] Gu, X., Ciampa, P., and Nagel, B., "An automated CFD analysis workflow in overall aircraft design applications. CEAS Aeronaut J 9: 3–13," , 2018.

[2] Ciampa, P. D., Nagel, B., and La Rocca, G., "Preliminary design for flexible aircraft in a collaborative environment," *The International Conference of the European Aerospace Societies (CEAS), Linköping, Sweden*, 2013.

[3] Kroll, N., Abu-Zurayk, M., Dimitrov, D., Franz, T., Führer, T., Gerhold, T., Görtz, S., Heinrich, R., Ilic, C., Jepsen, J., et al., "DLR project Digital-X: towards virtual aircraft design and flight testing based on high-fidelity methods," *CEAS Aeronautical Journal*, Vol. 7, 2016, pp. 3–27.

[4] Ghoreyshi, M., Badcock, K., Da Ronch, A., Vallespin, D., and Rizzi, A., "Automated CFD analysis for the investigation of flight handling qualities," *Mathematical Modelling of Natural Phenomena*, Vol. 6, No. 3, 2011, pp. 166–188.

[5] Ghoreyshi, M., Jirásek, A., and Cummings, R. M., "Reduced order unsteady aerodynamic modeling for stability and control analysis using computational fluid dynamics," *Progress in Aerospace Sciences*, Vol. 71, 2014, pp. 167–217.

[6] Ghoryeshi, M., Jirásek, A., and Cummings, R. M., "Computational Investigation into the Use of Response Functions for Aerodynamic Loads Modeling," *AIAA Journal*, Vol. 50, No. 6, 2012, pp. 1314–1327.

[7] Hall, K. C., Thomas, J. P., and Dowell, E. H., "Reduced-Order Modelling of Unsteady Small-Disturbance Using a Frequency-Domain Proper Orthogonal Decomposition Technique," AIAA Paper 1999–655, January 1999.

[8] Da Ronch, A., McCracken, A., Badcock, K. J., Ghoreyshi, M., and Cummings, R. M., "Modeling Unsteady Aerodynamic Loads," AIAA Paper 2011–6524, August 2011.

[9] Dias Ribeiro, M., Stradtner, M., and Bekemeyer, P., "Unsteady reduced order model with neural networks and flight-physics-based regularization for aerodynamic applications," *Computers & Fluids*, Vol. 264, 2023, p. 105949. doi:10.1016/j.compfluid.2023.105949.

[10] Stradtner, M., Liersch, C. M., and Bekemeyer, P., "An aerodynamic variable-fidelity modelling framework for a low-observable UCAV," *Aerospace Science and Technology*, 2020, p. 106232. doi:10.1016/j.ast.2020.106232.

[11] Immordino, G., Da Ronch, A., and Righi, M., "Steady-State Transonic Flowfield Prediction via Deep-Learning Framework," *AIAA Journal*, 2024, pp. 1–17.

[12] Allen, J., and Ghoreyshi, M., "Forced motions design for aerodynamic identification and modeling of a generic missile configuration," *Aerospace Science and Technology*, Vol. 77, 2018, pp. 742–754.

[13] Leontaritis, I. J., and Billings, S. A., "Input-output parametric models for non-linear systems Part I: deterministic non-linear systems," *International Journal of Control*, Vol. 41, No. 2, 1985, pp. 303–328. doi:10.1080/0020718508961129.

[14] LEONTARITIS, I. J., and Billings, S. A., "Input-output parametric models for non-linear systems Part II: stochastic non-linear systems," *International Journal of Control*, Vol. 41, No. 2, 1985, pp. 329–344. doi:10.1080/0020718508961130.

[15] Glaz, B., Liu, L., and Friedmann, P. P., "Reduced-order nonlinear unsteady aerodynamic modeling using a surrogate-based recurrence framework," *AIAA journal*, Vol. 48, No. 10, 2010, pp. 2418–2429.

[16] Ghoreyshi, M., Cummings, R. M., Ronch, A. D., and Badcock, K. J., "Transonic aerodynamic load modeling of X-31 aircraft pitching motions," *AIAA journal*, Vol. 51, No. 10, 2013, pp. 2447–2464.

[17] Ghoreyshi, M., Jirasek, A., and Cummings, R. M., "Computational approximation of nonlinear unsteady aerodynamics using an aerodynamic model hierarchy," *Aerospace Science and Technology*, Vol. 28, No. 1, 2013, pp. 133–144.

[18] Lui, H. F., and Wolf, W. R., "Construction of reduced-order models for fluid flows using deep feedforward neural networks," *Journal of Fluid Mechanics*, Vol. 872, 2019, pp. 963–994.

[19] Huang, R., Hu, H., and Zhao, Y., "Nonlinear reduced-order modeling for multiple-input/multiple-output aerodynamic systems," *AIAA journal*, Vol. 52, No. 6, 2014, pp. 1219–1231.

[20] Stradtner, M., Drazen, D., and van Rooij, M., "Introduction to AVT-351: Enhanced Computational Performance and Stability & Control Prediction for NATO Military Vehicles," *AIAA SCITECH 2023 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2023. doi:10.2514/6.2023-0820.

[21] O'Neill, C., and Arena, A., "Comparison of Time Domain Training Signals for CFD Based Aerodynamic System Identification," *42nd AIAA Aerospace Sciences Meeting and Exhibit*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2004. doi:10.2514/6.2004-209.

[22] Pfifer, H., and Danowsky, B. P., "System Identification of a small flexible aircraft-invited," *AIAA Atmospheric Flight Mechanics Conference*, 2016, p. 1750.

[23] Sears, D. R., and Morton, S. A., "HPCMP CREATETM–The Remarkable Success of a DoD Software Factory," AIAA Paper 2023–2014, January 2023.

[24] Morton, S. A., and McDaniel, D. R., "A fixed-wing aircraft simulation tool for improving DoD acquisition efficiency," *Computing in Science & Engineering*, Vol. 18, No. 1, 2015, pp. 25–31. doi:10.1109/MCSE.2015.133.

[25] Chyczewski, T. S., Lofthouse, A. J., Gea, L.-M., Cartieri, A., and Hiller, B. R., "Summary of the First AIAA Stability and Control Prediction Workshop," *AIAA 2022-1680*, 2022. doi:10.2514/6.2022-1680.

[26] Aref, P., Hummer, C., Ghoreyshi, M., Jirasek, A., and Seidel, J., "Application of the HPCMP CREATETM-AV Kestrel to the 1st AIAA Workshop for Integrated Propeller Prediction," *AIAA Paper 2020-2674*, 2020. doi:10.2514/6.2020-2674.

[27] McDaniel, D. R., Nichols, R. H., Eymann, T. A., Starr, R. E., and Morton, S. A., "Accuracy and Performance Improvements to Kestrel's Near-Body Flow Solver," *AIAA Paper 2016–1051*, January 2016. doi:10.2514/6.2016-1051.

[28] Glasby, R. S., and Erwin, J. T., "Introduction to COFFE: The Next-Generation HPCMP CREATETM-AV CFD Solver," *AIAA Paper 2016–0567*, January 2016. doi:10.2514/6.2016-0567.

[29] Eymann, T. A., Nichols, R. H., Tuckey, T., and McDaniel, D. R., "Cartesian Adaptive Mesh Refinement with the HPCMP CREATE™-AV Kestrel Solver," *AIAA Paper 2015–0040*, January 2015. doi:10.2514/6.2015-0040.

[30] Glasby, R., Erwin, J., Eymann, T., Nichols, R., McDaniel, D., Karman Jr, S., Stefanski, D., and Holst, K., "Results from DoD HPCMP CREATETM-AV Kestrel for the 3rd AIAA High Lift Prediction Workshop," *AIAA Paper 2018–1256*, January 2018. doi:10.2514/6.2018-1256.

[31] Morton, S. A., Tillman, B., McDaniel, D. R., Sears, D. R., and Tuckey, T. R., "Kestrel–A Fixed Wing Virtual Aircraft Product of the CREATE Program," *AIAA Paper 2009–0338*, January 2009. doi:10.2514/6.2009-0338.

[32] Tramel, R., Nichols, R., and Buning, P., "Addition of improved shock-capturing schemes to OVERFLOW 2.1," *AIAA Paper 2009–3988*, June 2009. doi:10.2514/6.2009-3988.

[33] Krist, S. E., Ratnayake, N. A., and Ghaffari, F., "Kestrel Results at Liftoff Conditions for a Space Launch System Configuration in Proximity to the Launch Tower," AIAA Paper 2019–3400, June 2019.

[34] Menter, F. R., "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications," *AIAA Journal*, Vol. 32, No. 8, 1994, pp. 1598–1605. doi:10.2514/3.12149.

[35] Galrinho, M., "Least squares methods for system identification of structured models," Ph.D. thesis, KTH Royal Institute of Technology, 2016.

[36] Rasmussen, C. E., and Williams, C. K. I., *Gaussian processes for machine learning*, 3rd ed., Adaptive computation and machine learning, MIT Press, Cambridge, Mass., 2008.

[37] Bekemeyer, P., Bertram, A., Hines Chaves, D. A., Dias Ribeiro, M., Garbo, A., Kiener, A., Sabater, C., Stradtner, M., Wassing, S., Widhalm, M., Goertz, S., Jaeckel, F., Hoppe, R., and Hoffmann, N., "Data-Driven Aerodynamic Modeling Using the DLR SMARTy Toolbox," *AIAA AVIATION 2022 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2022. doi:10.2514/6.2022-3899.

[38] Stradtner, M., and Bekemeyer, P., "Nonlinear Unsteady Aerodynamic Reduced-Order Modeling Using a Surrogate-Based Recurrent Framework," *AIAA AVIATION 2024 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2024 (submitted for publication).

[39] Glaz, B., Liu, L., and Friedmann, P. P., "Reduced-Order Nonlinear Unsteady Aerodynamic Modeling Using a Surrogate-Based Recurrence Framework," *AIAA Journal*, Vol. 48, No. 10, 2010, pp. 2418–2429. doi:10.2514/1.J050471.

[40] Fresca, S., and Manzoni, A., "POD-DL-ROM: Enhancing deep learning-based reduced order models for nonlinear parametrized PDEs by proper orthogonal decomposition," *Computer Methods in Applied Mechanics and Engineering*, Vol. 388, 2022, p. 114181.

[41] Mohan, A. T., and Gaitonde, D. V., "A deep learning based approach to reduced order modeling for turbulent flow control using LSTM neural networks," *arXiv preprint arXiv:1804.09269*, 2018.

[42] Catalani, G., "Machine Learning Based local Reduced Order Modeling for the prediction of Unsteady Aerodynamic Loads." 2022.

[43] Bourier, S., "Development of a CFD data-driven surrogate model using the neural network approach for prediction of aircraft performance characteristics," 2021.

[44] Gallagher, N. B., "Whittaker Smoother," *white paper Eigenvector Research, Inc., www. eigenvector. com*, 2018.

[45] Kovacs, L., Passaggia, P.-Y., Mazellier, N., and Lago, V., "Detection method for shock-waves in viscous flows," *Experiments in Fluids*, Vol. 63, No. 1, 2022, p. 11.

[46] Rumelhart, D. E., Hinton, G. E., and Williams, R. J., "Learning representations by back-propagating errors," *nature*, Vol. 323, No. 6088, 1986, pp. 533–536.

[47] Lipton, Z. C., Berkowitz, J., and Elkan, C., "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[48] Brunton, S. L., Noack, B. R., and Koumoutsakos, P., "Machine learning for fluid mechanics," *Annual review of fluid mechanics*, Vol. 52, 2020, pp. 477–508.

[49] Hochreiter, S., and Schmidhuber, J., "Long short-term memory," *Neural computation*, Vol. 9, No. 8, 1997, pp. 1735–1780.

[50] Goodfellow, I., Bengio, Y., and Courville, A., *Deep Learning*, MIT Press, 2016. `http://www.deeplearningbook.org`.

[51] Gers, F. A., Schmidhuber, J., and Cummins, F., "Learning to forget: Continual prediction with LSTM," *Neural computation*, Vol. 12, No. 10, 2000, pp. 2451–2471.

[52] Bányász, C., and of Automatic Control, I. F., *Adaptive Systems in Control and Signal Processing 1995: A Postscript Volume from the Fifth IFAC Symposium, Budapest, Hungary, 14-16 June 1995*, IFAC Series, International Federation of Automatioc Control, 1995. URL `https://books.google.com/books?id=kNVSAAAAMAAJ`.

[53] Morelli, E. A., "Multiple input design for real-time parameter estimation in the frequency domain," *IFAC Proceedings Volumes*, Vol. 36, No. 16, 2003, pp. 639–644.

[54] Rein, M., "Subsonic, Transonic and Supersonic Wind Tunnel Tests of the Generic Slender Wing Configuration DLR-F22 with Leading-Edge Vortex Controllers and Strakes," Tech. rep., DLR, 2022. URL `https://elib.dlr.de/185854/`.

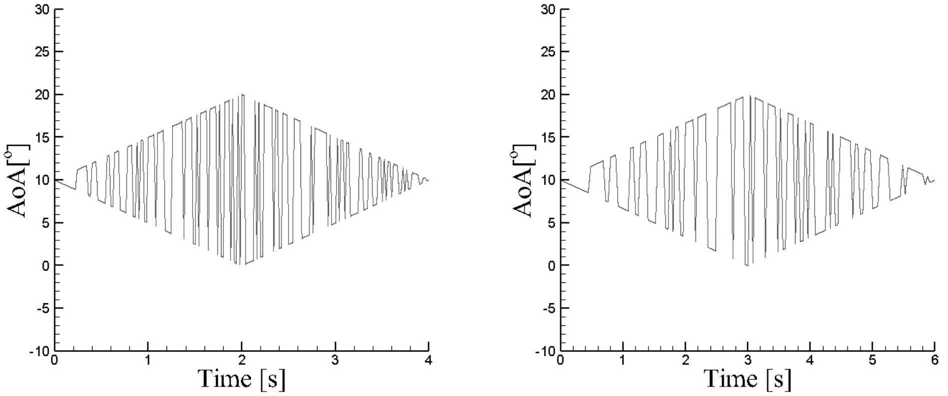
[55] Liersch, C. M., Schütte, A., Moerland, E., and Kalanja, M., "DLR Project Diabolo: An Approach for the Design and Technology Assessment for Future Fighter Configurations," *AIAA AVIATION 2023 Forum*, [2023].

[56] Farcy, D., Tanguy, G., Vauchel, N., Dubot, G., and Garnier, E., "Static and dynamic aerodynamic coefficients evaluation on a generic fighter configuration in ONERA low speed wind tunnels," *AIAA AVIATION 2024 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2024 (submitted for publication).

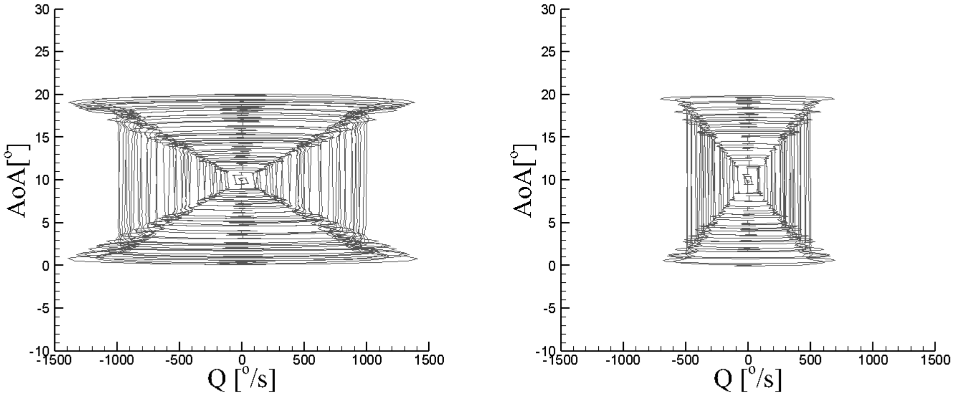
[57] Cummings, R. M., Morton, S. A., and McDaniel, D. R., "Experiences in accurately predicting time-dependent flows," *Progress in Aerospace Sciences*, Vol. 44, 2008, pp. 241–257. doi:10.1016/j.paerosci.2008.01.001.

[58] Schütte, A., and Hummel, D., “Numerical Design Studies on the Roll Stability of a Multi-Delta-Wing Configuration,” *Journal of Aircraft*, 2022, pp. 1–17. doi:10.2514/1.C037128.

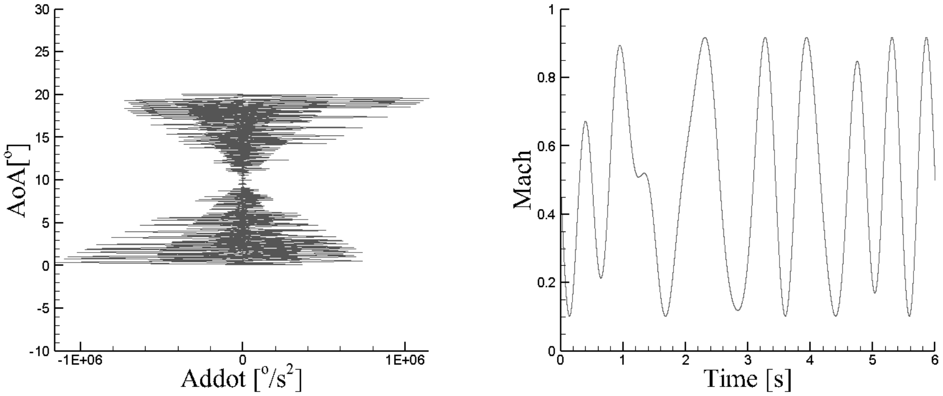
[59] Widhalm, M., Stradtner, M., Schuette, A., Ghoreyshi, M., Jirasek, A., and Seidel, J., *AIAA Paper 2023–4199*, June 2023. doi:10.2514/6.2023-4199.



(a) PRBS1 (b) PRBS2



(c) PRBS1 (d) PRBS2



(e) PRBS1 (f) PRBS2

Fig. 5 PRBS1 and PRBS2 signals

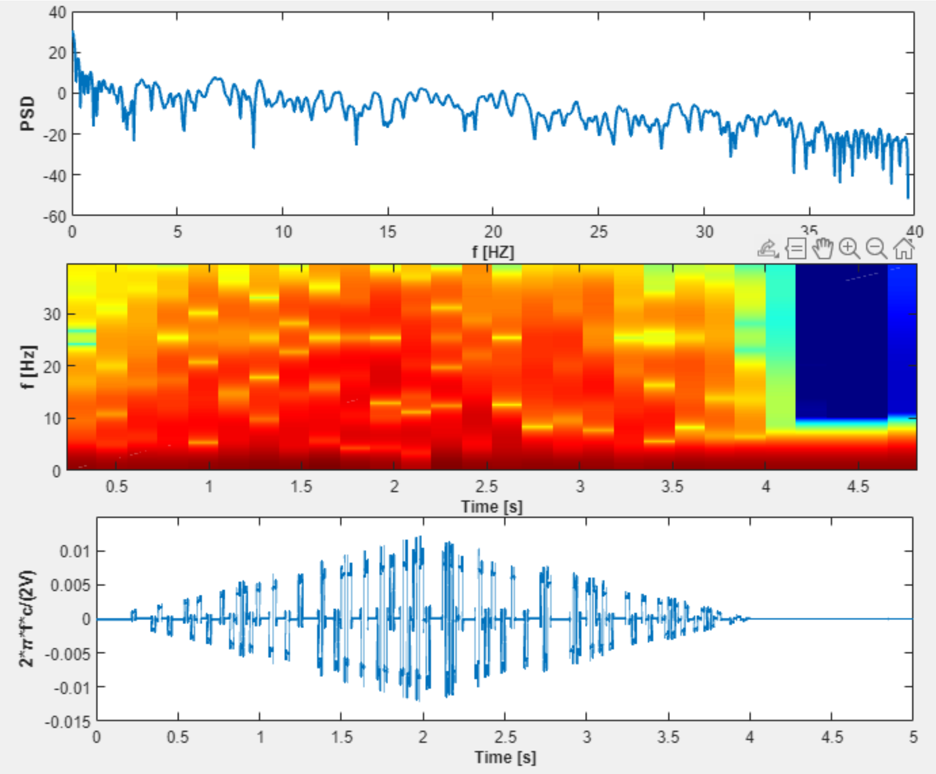
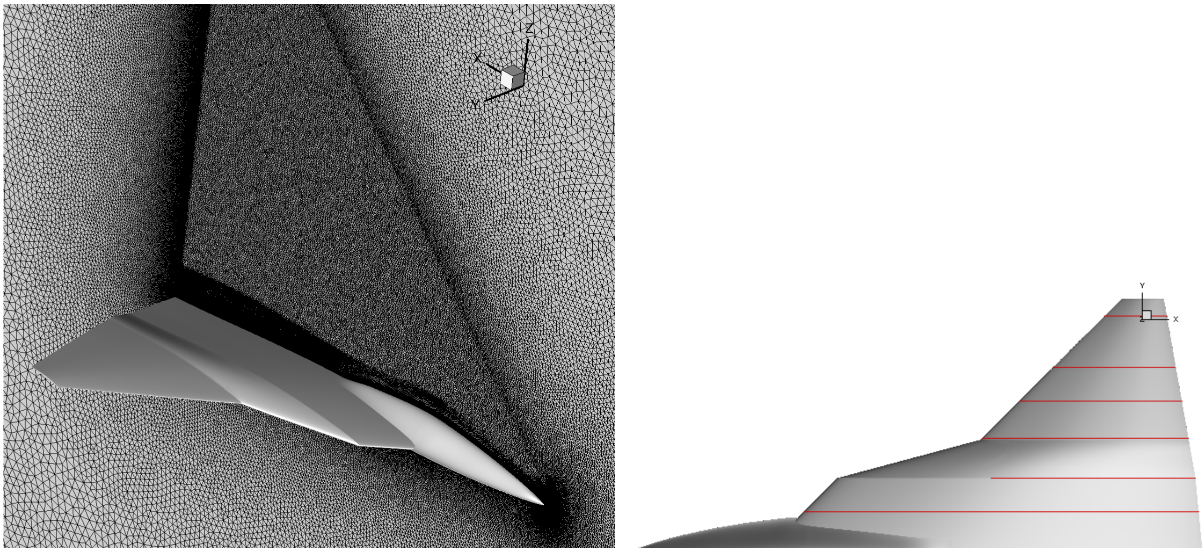


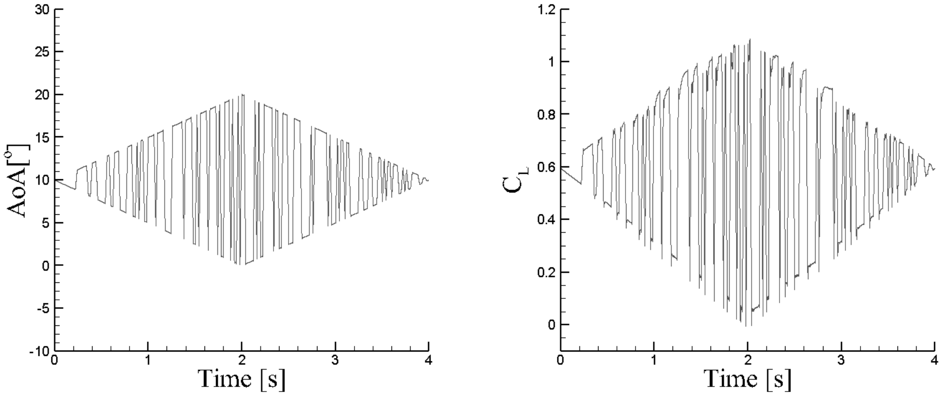
Fig. 6 Power spectral density of the PRBS signal at constant Mach 0.85.



(a) Computational grid

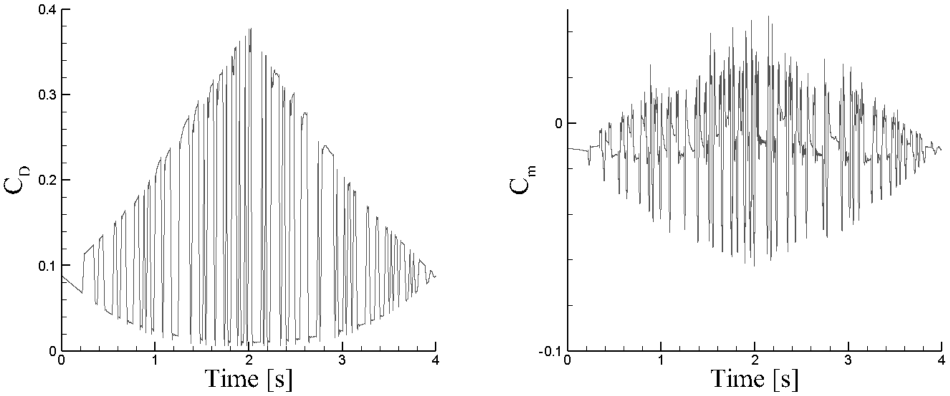
(b) Tap Points

Fig. 7 (a) Near field of the computational grid of the generic triple delta fighter aircraft, and (b) Tap points at six difference spanwise slices.



(a) Input

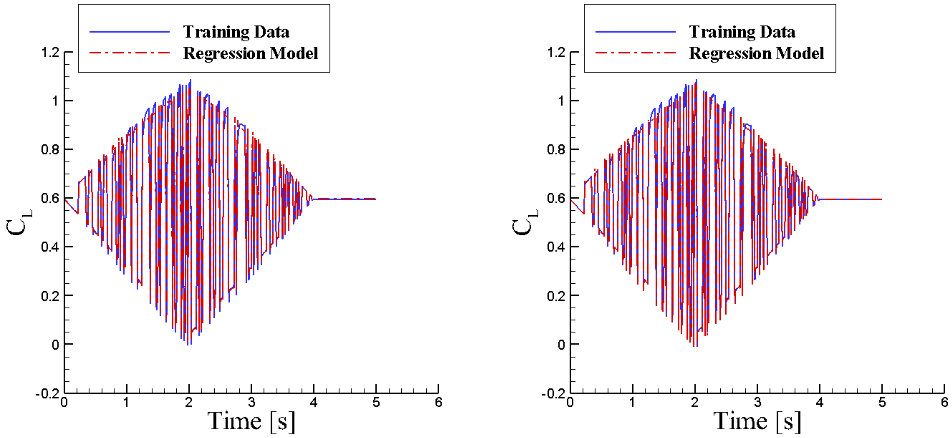
(b) Lift coeff



(c) Drag coeff

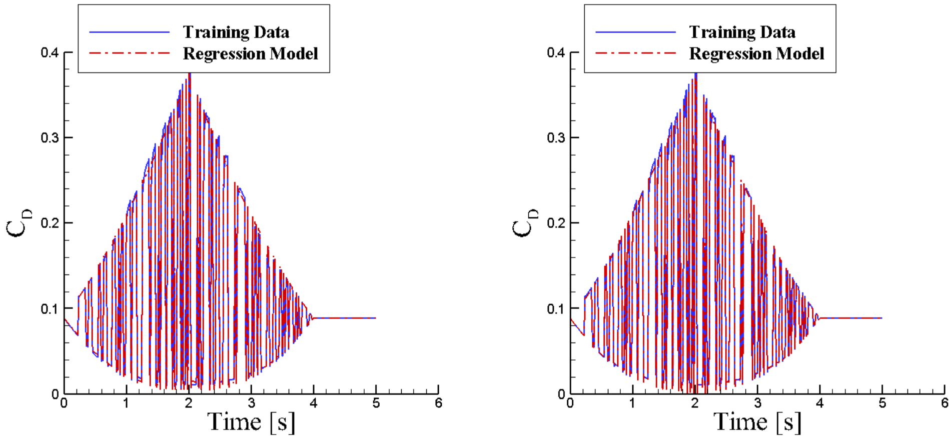
(d) Pitch-moment coeff

Fig. 8 PRBS1 (constant Mach 0.85) input and output data.



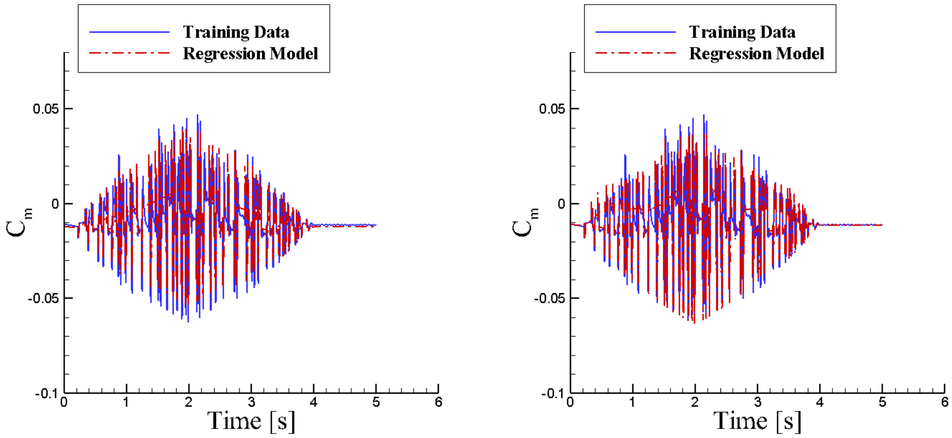
(a) Regression-CL

(b) NN-CL



(c) Regression-CD

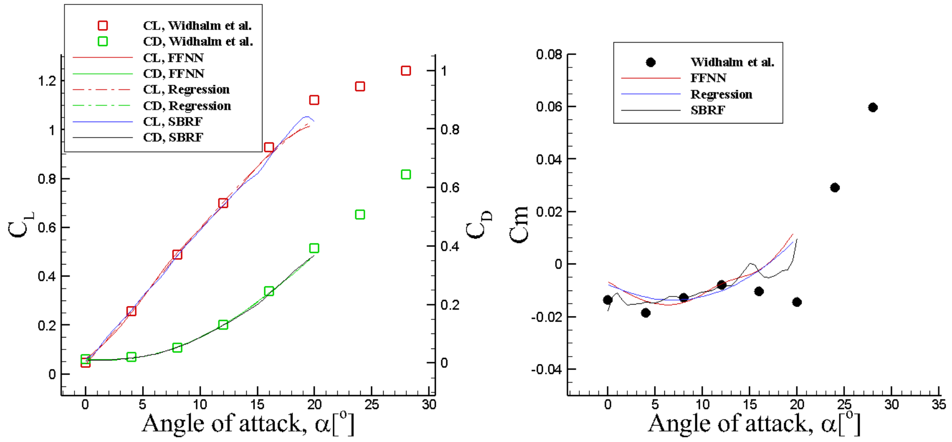
(d) NN-CD



(e) Regression-Cm

(f) NN-Cm

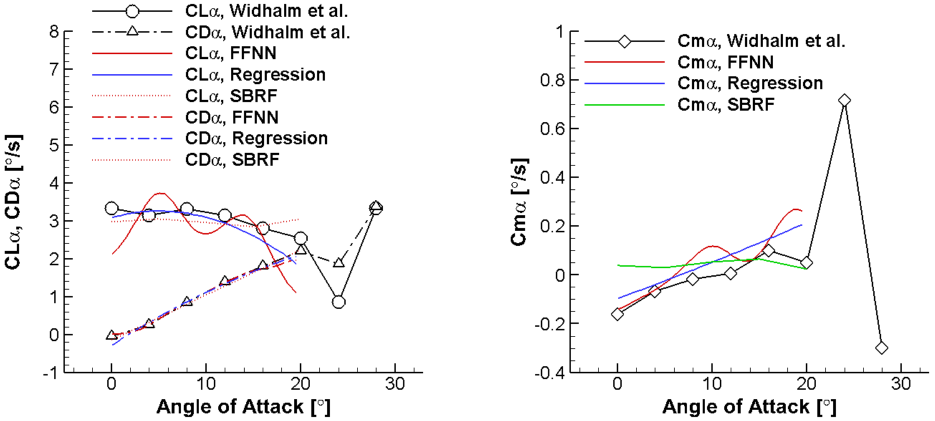
Fig. 9 Prediction of PRBS1 signal using regression and FFNN models.



(a) CL/CD

(b) Cm

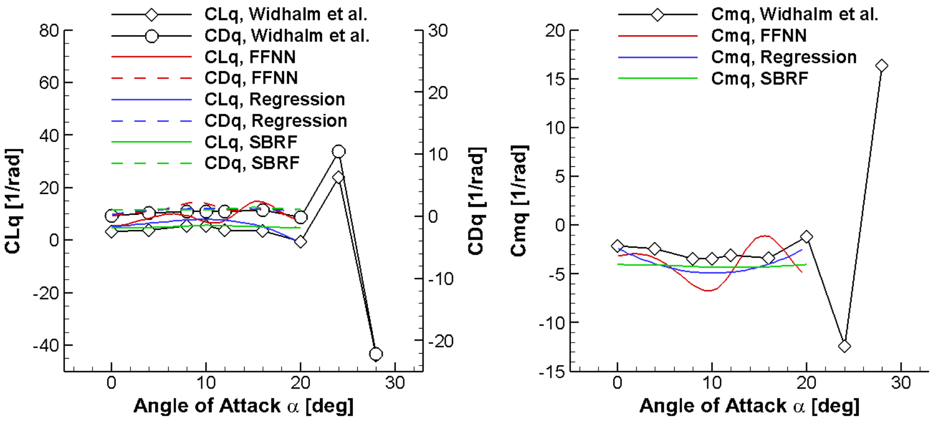
Fig. 10 Prediction of static data using PRBS1 signal and regression/FFNN models.



(a) CL/CD

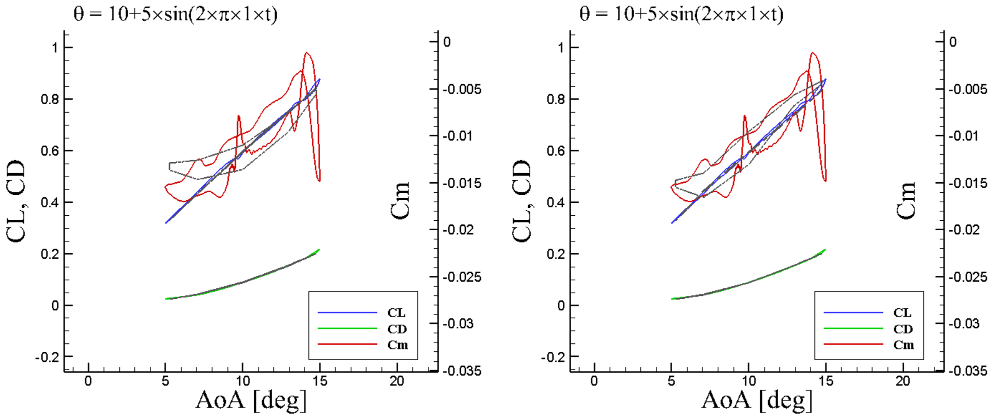
(b) Cm

Fig. 11 Prediction of slope data using PRBS1 signal and regression/FFNN models.

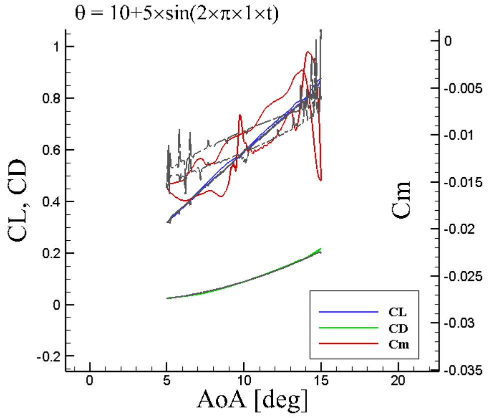


(a) CLq/CDq (b) Cm

Fig. 12 Prediction of dynamic derivative data using PRBS1 signal and regression/FFNN models.

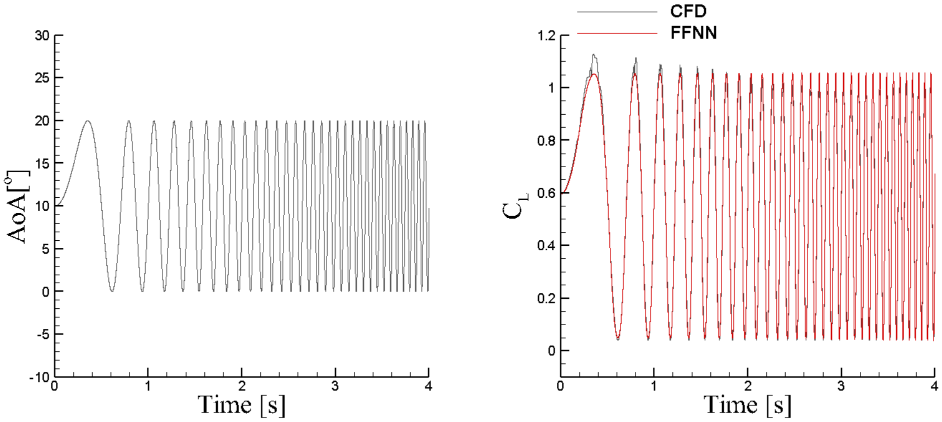


(a) Regression (b) FFNN

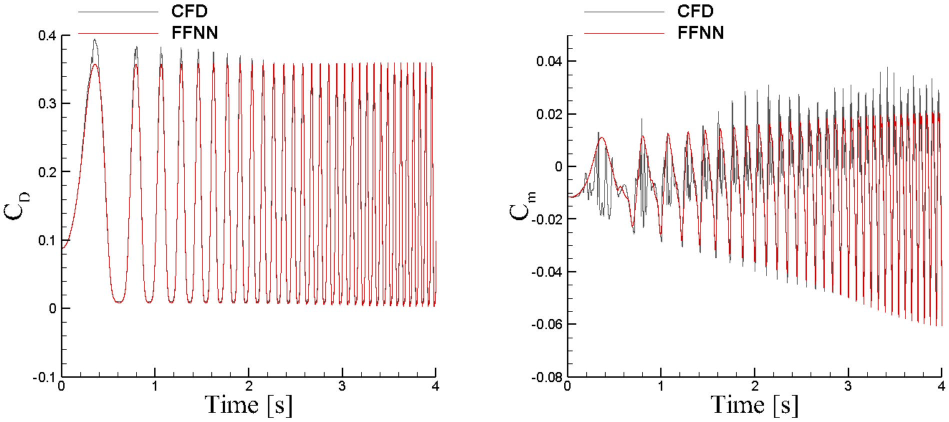


(c) QS-GPR

Fig. 13 Prediction of pitch oscillation using PRBS1 signal and regression/FFNN/SBRF models. Black lines show model predictions.

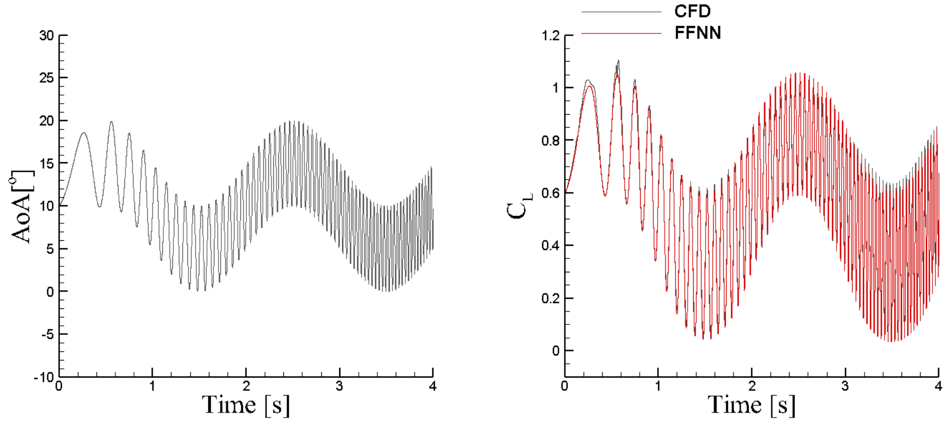


(a) AoA (b) CL

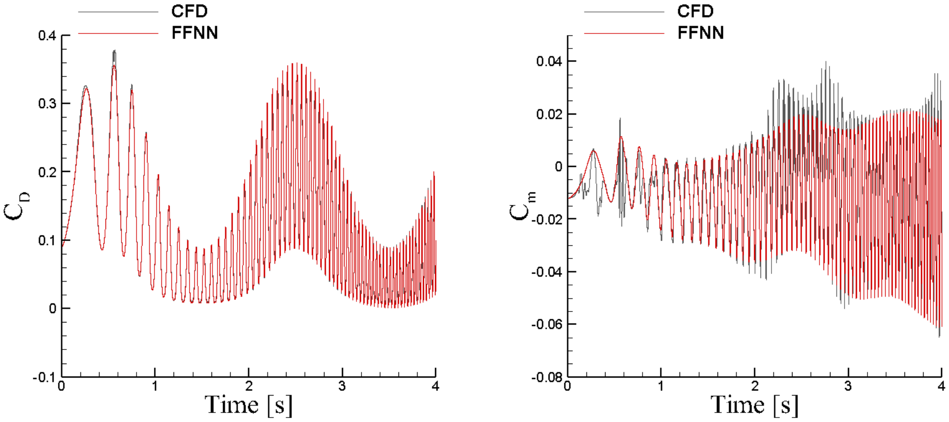


(c) CD (d) Cm

Fig. 14 Prediction of Chirp with constant mean using PRBS1 signal and FFNN models.

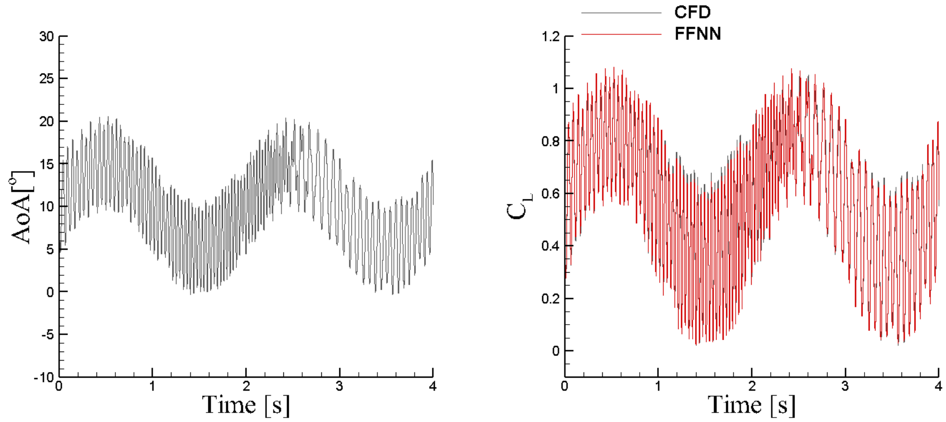


(a) AoA (b) CL



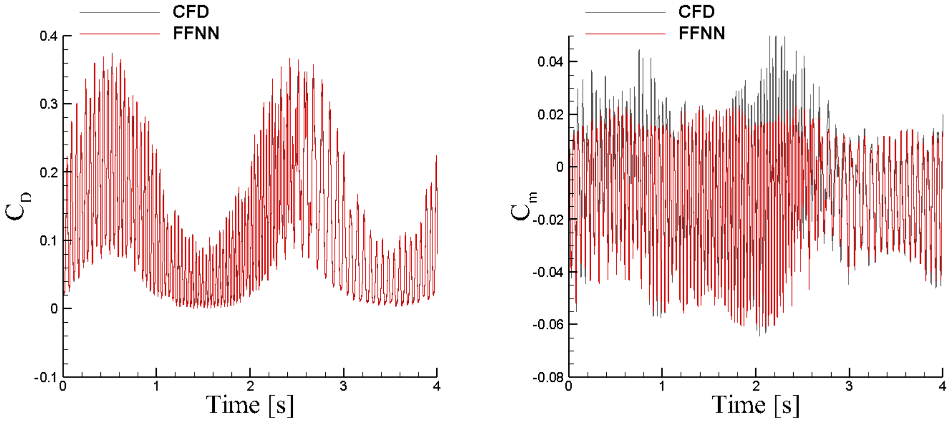
(c) CD (d) Cm

Fig. 15 Prediction of Chirp with varying mean using PRBS1 signal and FFNN models.



(a) AoA

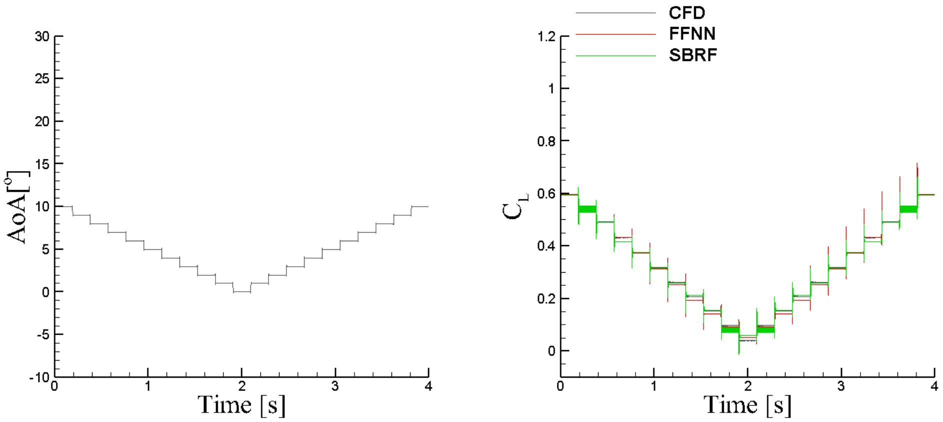
(b) CL



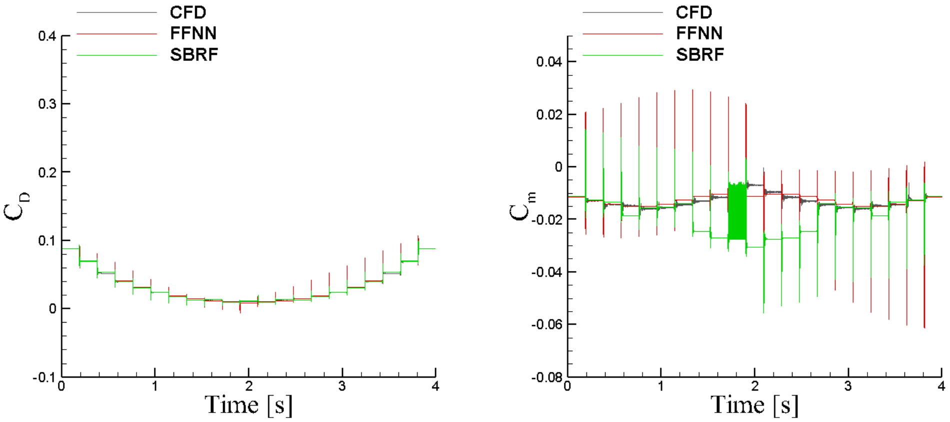
(c) CD

(d) Cm

Fig. 16 Prediction of Schroeder with varying mean using PRBS1 signal and FFNN models.

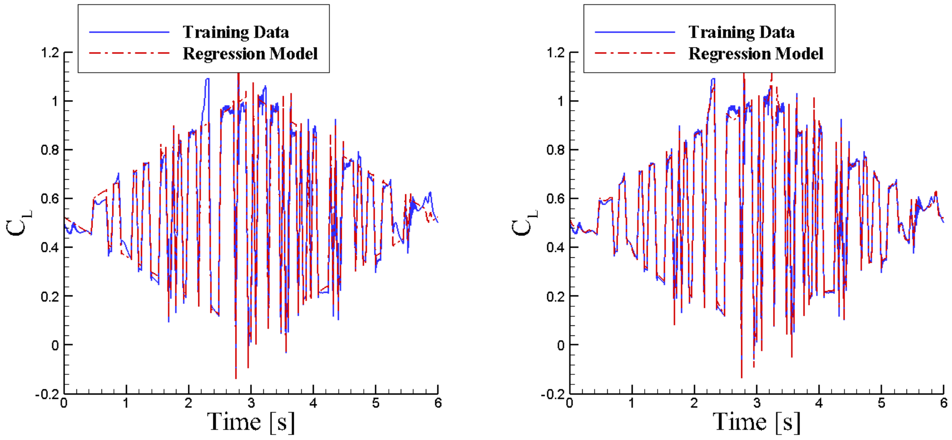


(a) AoA (b) CL

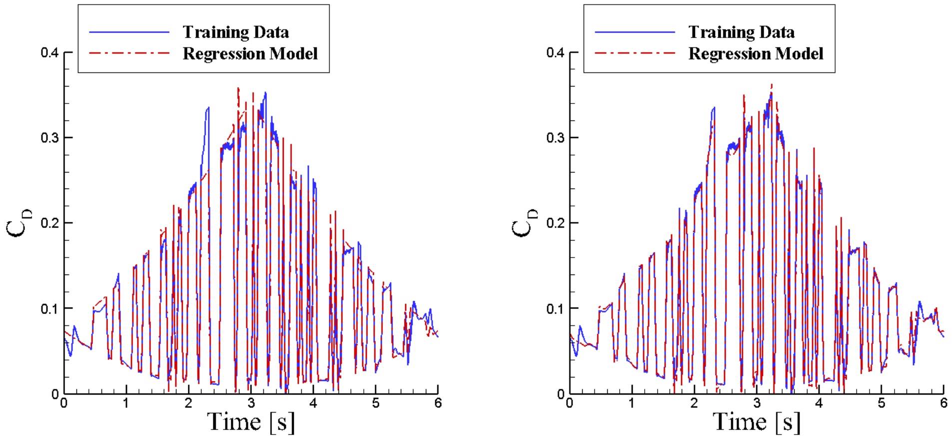


(c) CD (d) Cm

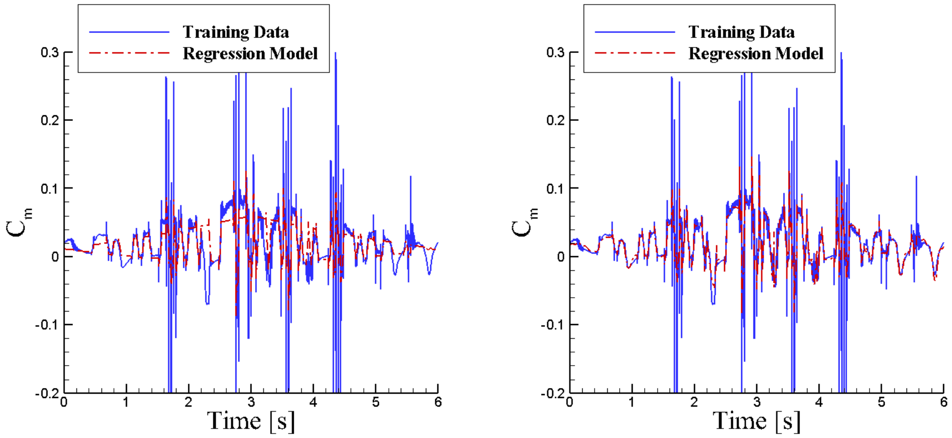
Fig. 17 Prediction of a step signal; using PRBS1 signal and FFNN/SBRF models.



(a) CL-Eq. 2 (b) CL-Eq. 3

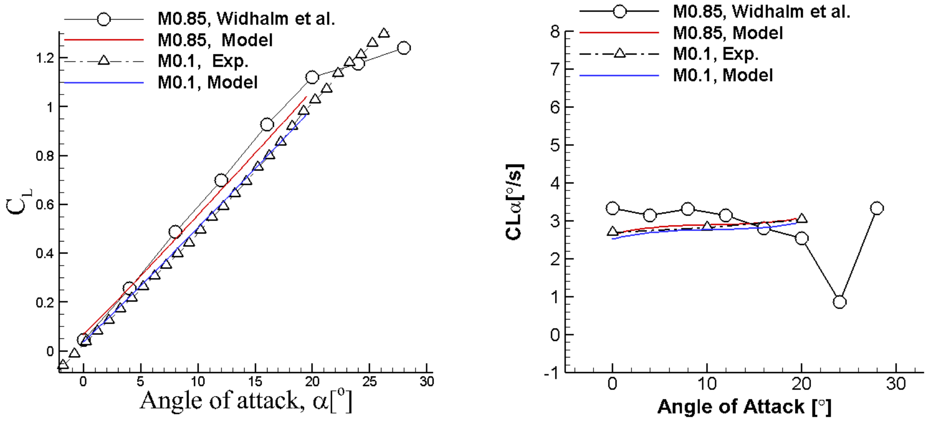


(c) CD-Eq. 2 (d) CD-Eq. 3

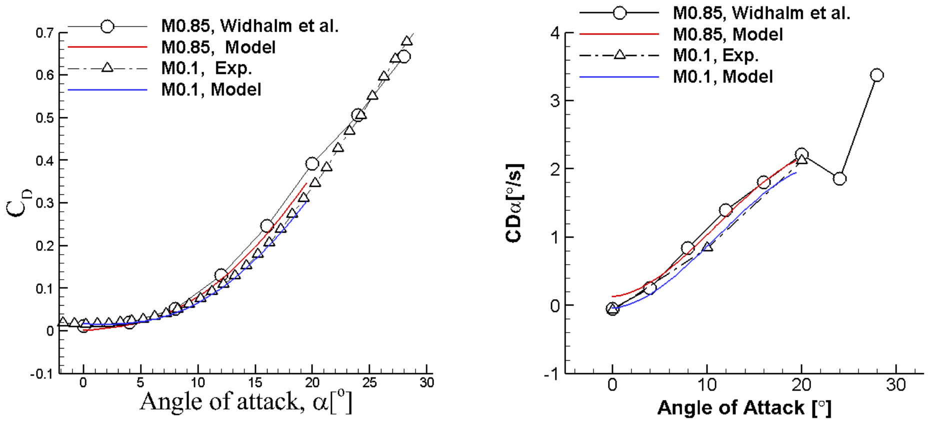


(e) Cm-Eq. 2 (f) Cm-Eq. 3

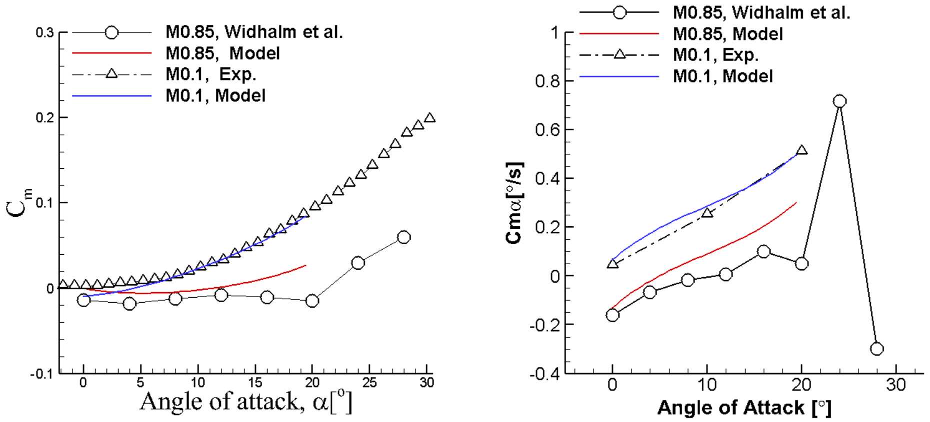
Fig. 18 Regression prediction of the PRBS2 signal using equations 2 and 3.



(a) CL (b) CLa

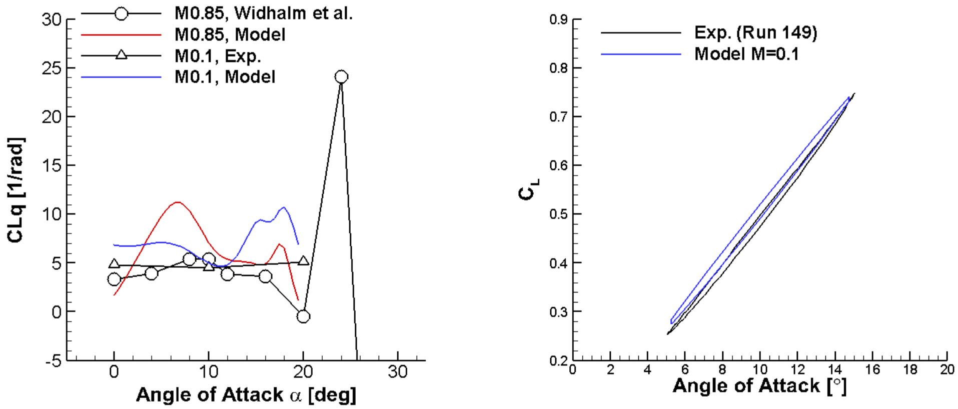


(c) CD (d) CDa

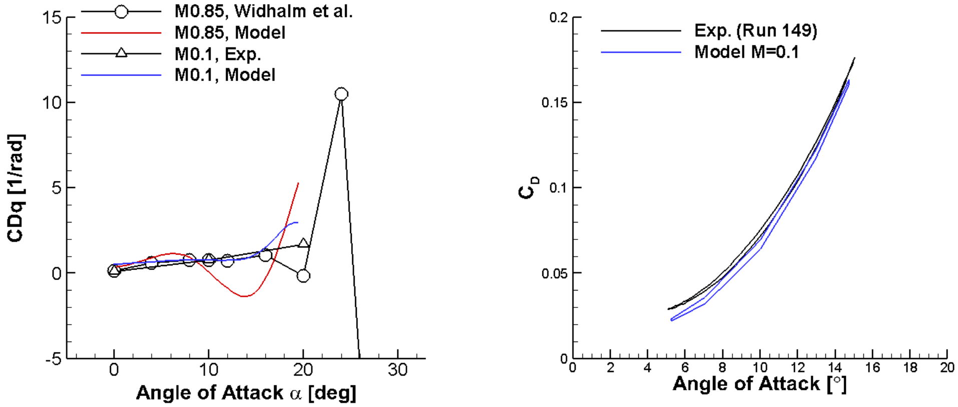


(e) Cm (f) Cma

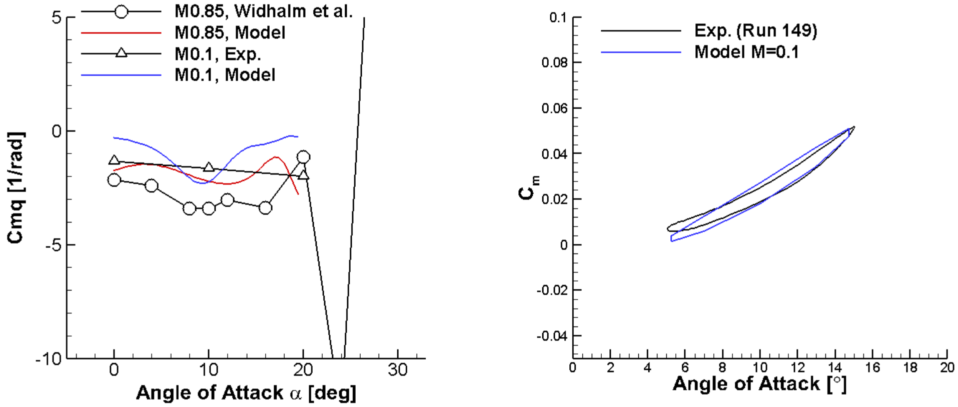
Fig. 19 Regression prediction of static and slope data at Mach 0.1 and Mach 0.85 using the PRBS2 signal.



(a) CLq+CLadot (b) CL Sinusoidal, M0.1, f1Hz

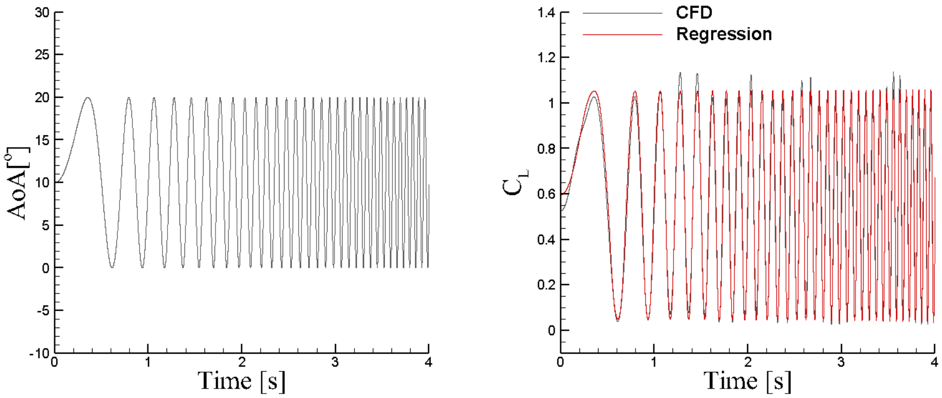


(c) CDq+CDadot (d) CD Sinusoidal, M0.1, f1Hz

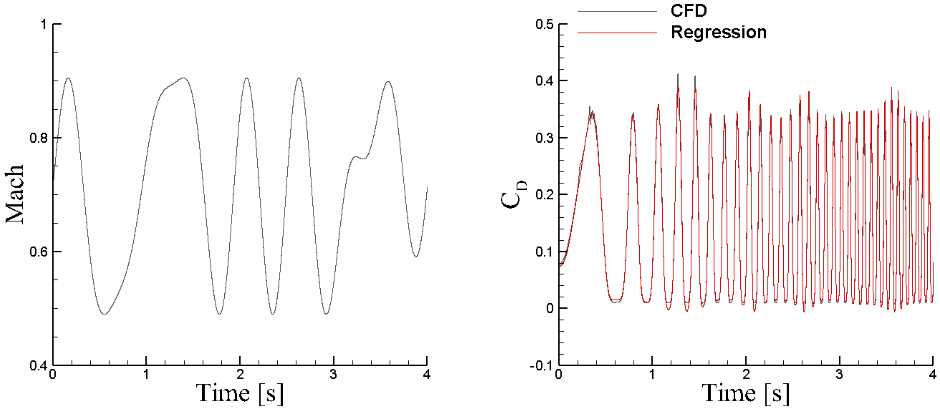


(e) Cmq+Cmadot (f) Cm Sinusoidal, M0.1, f1Hz

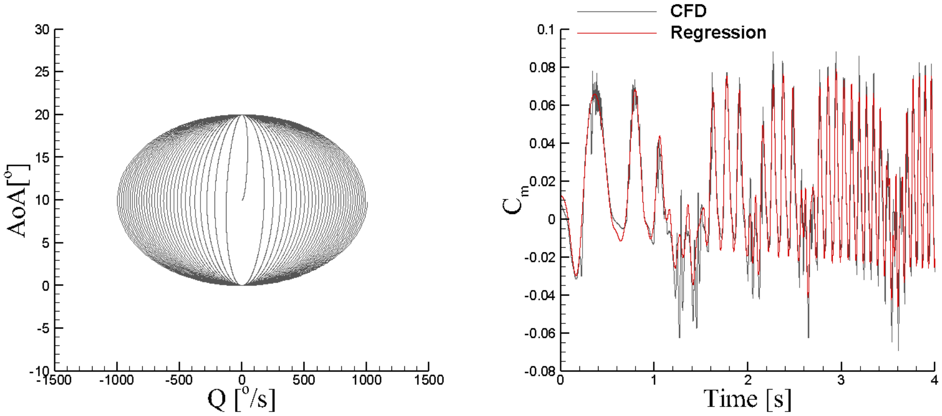
Fig. 20 Regression prediction of dynamic derivatives at Mach 0.1 and Mach 0.85 and a pitch sinousoial at Mach 0.1 with f1Hz using the PRBS2 signal.



(a) AoA (b) CL

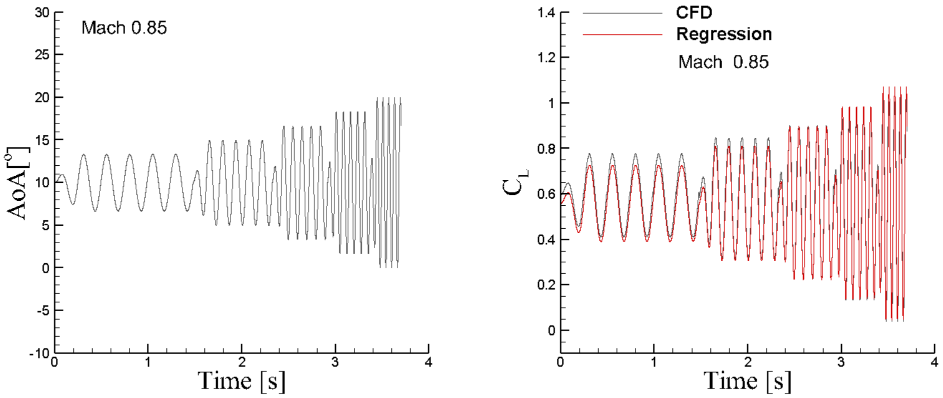


(c) Mach (d) CD



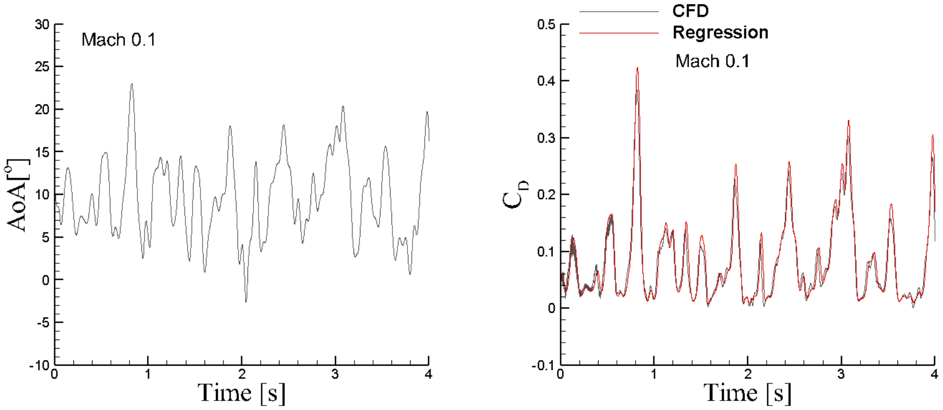
(e) AoA-Q (f) Cm

Fig. 21 Regression prediction of a Chirp motion with varying Mach number using the PRBS2 signal.



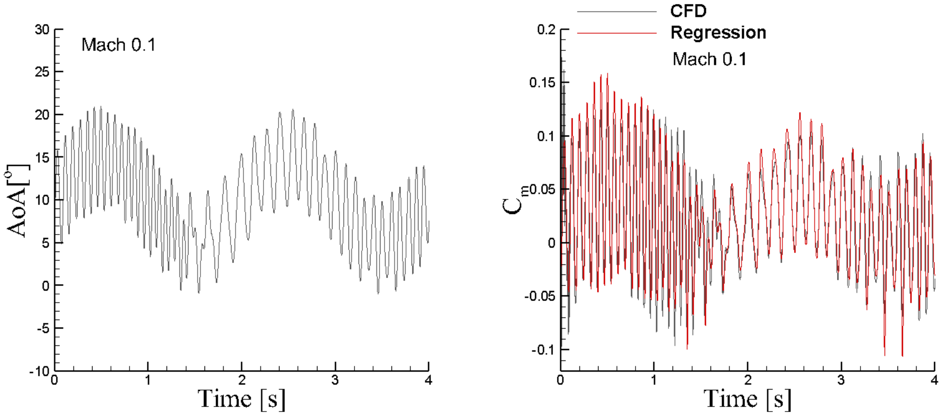
(a) Sinusoidal AoA, M0.85

(b) CL



(c) Random AoA, M0.1

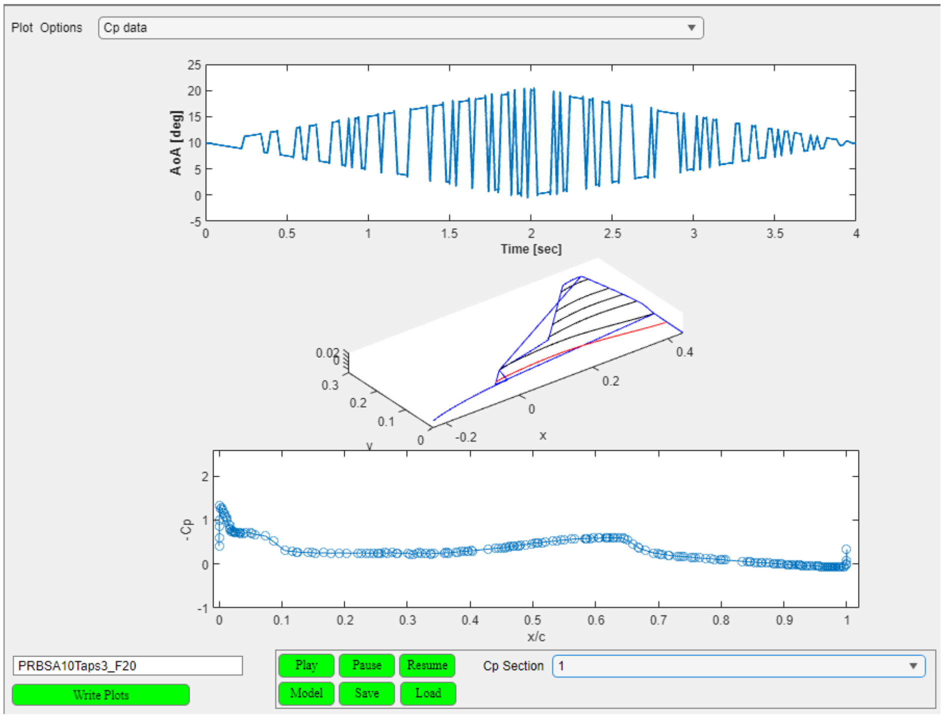
(d) CD



(e) Schroeder AoA, M0.1

(f) Cm

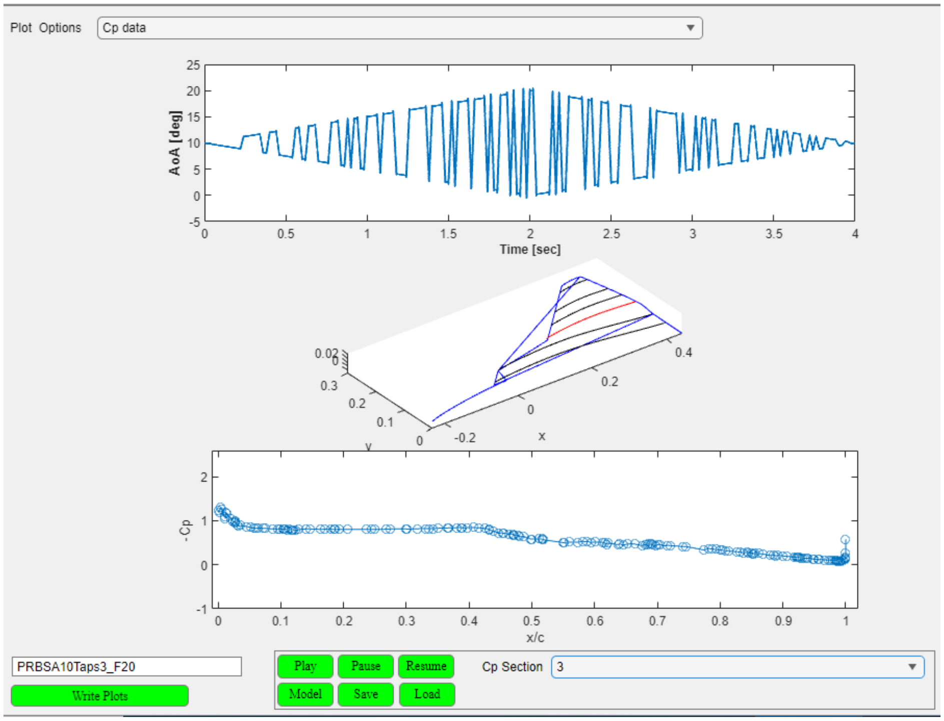
Fig. 22 Evaluating the regression model for prediction of different signals using the PRBS2 signal.


Plot Options: Cp data

PRBSA10Taps3_F20

Cp Section: 1

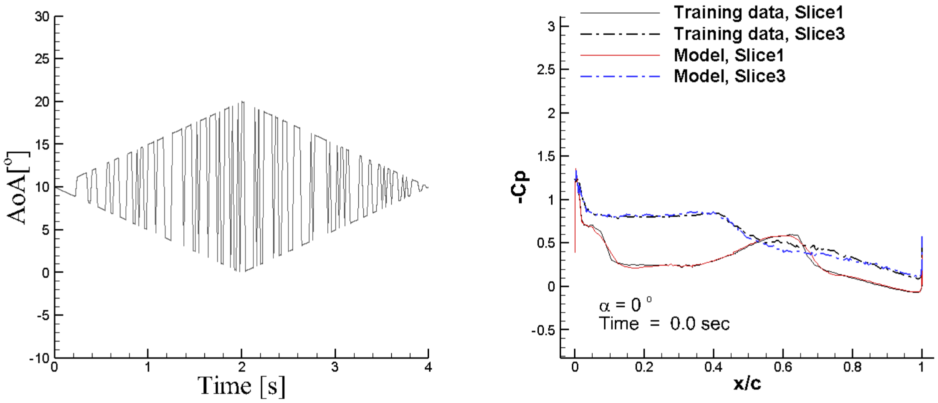
Fig. 23 **Section 1 highlighted by the red color.**


Plot Options: Cp data

PRBSA10Taps3_F20

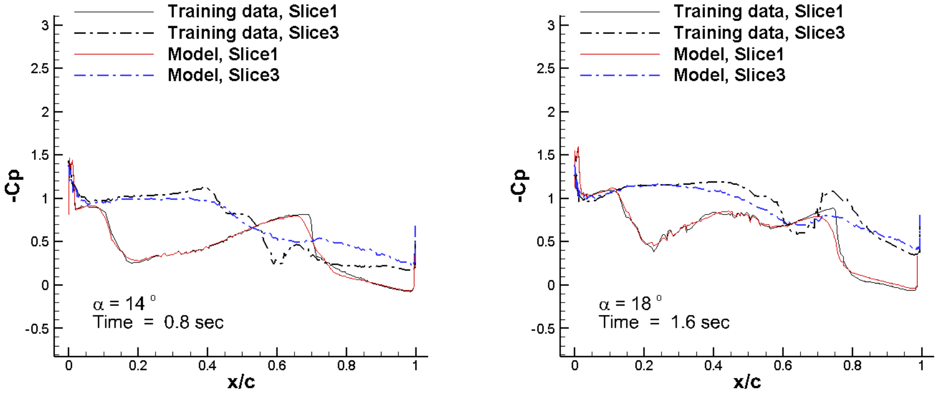
Cp Section: 3

Fig. 24 **Section 3 highlighted by the red color.**



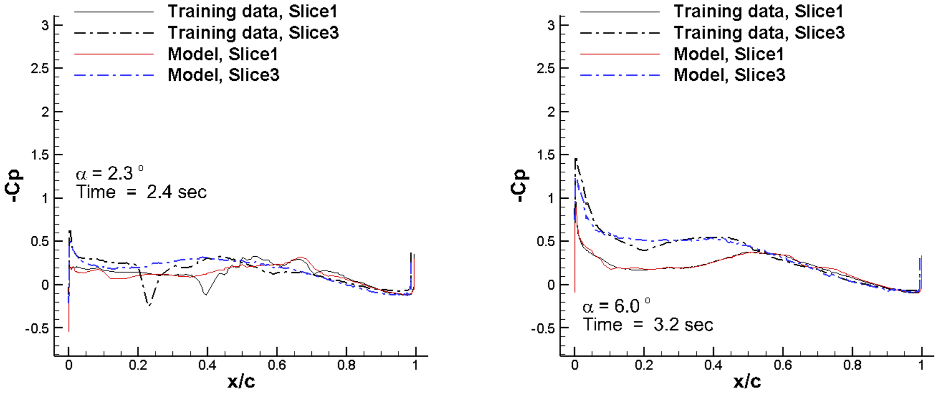
(a) PRBS1 Signal

(b) T=0s AoA=10



(c) T=0.8s AoA=14

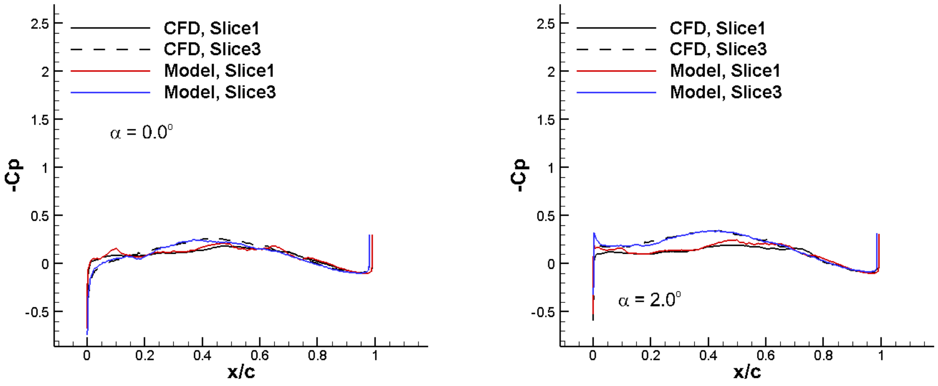
(d) T=1.6s AoA=18



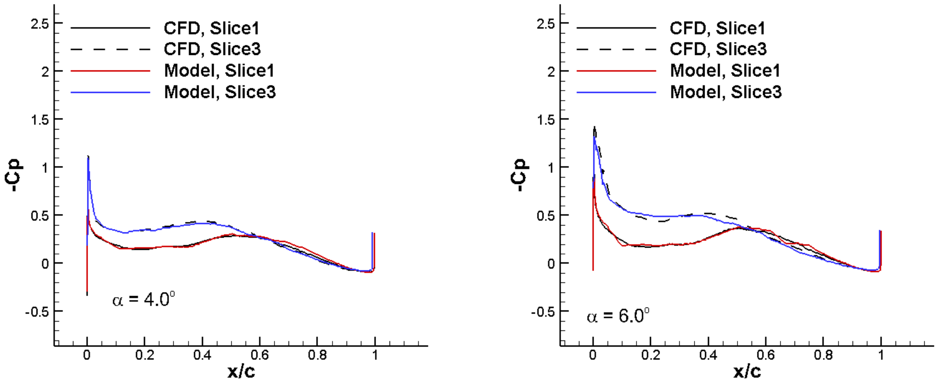
(e) T=2.4s AoA=2.3

(f) T=3.2s AoA=6

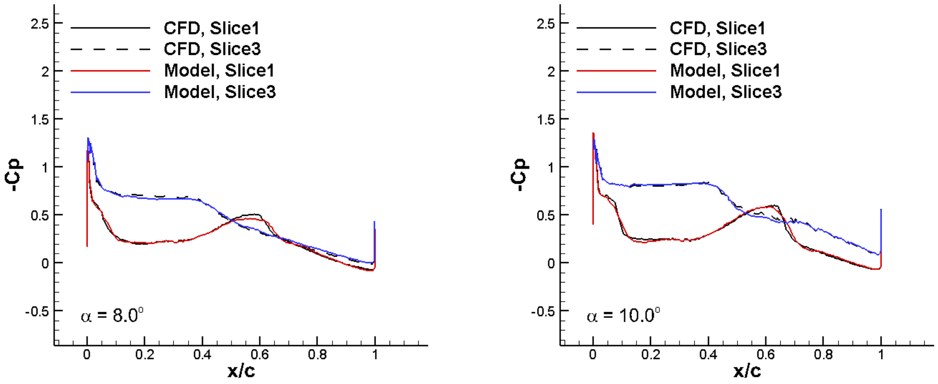
Fig. 25 Modeling pressure data at sections 1 and 3 using FFNN. Siganal is PRBS1 at Mach0.85.



(a) AoA=0 (b) AoA=2

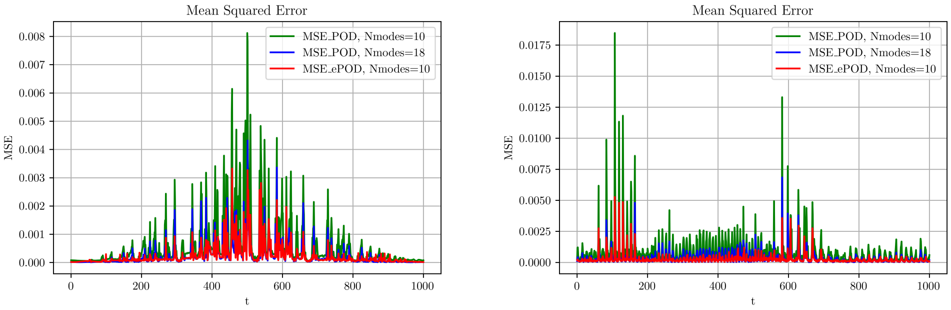


(c) AoA=4 (d) AoA=6



(e) AoA=8 (f) AoA=10

Fig. 26 Modeling pressure data at sections 1 and 3 of FFD at static angles and Mach 0.85. The model was created from the PRBS1 signal.



(a) PRBS1 Signal

(b) Schroeder maneuver

Fig. 27 Mean Squared Error for the maneuvers under consideration.

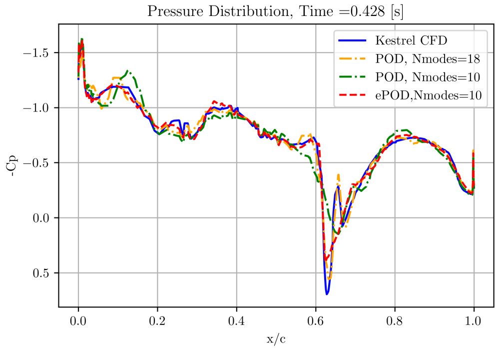
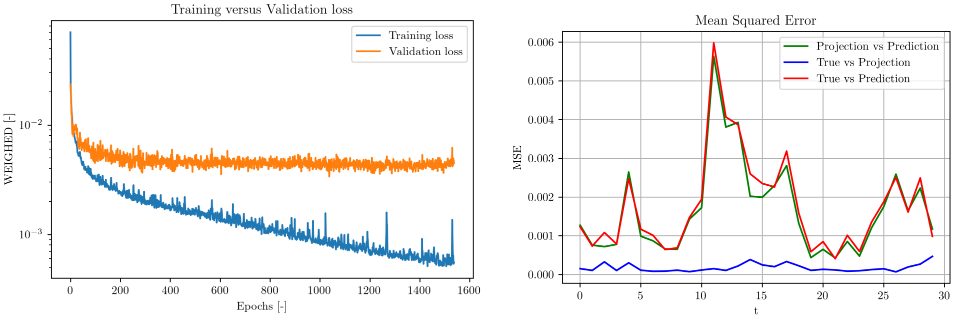


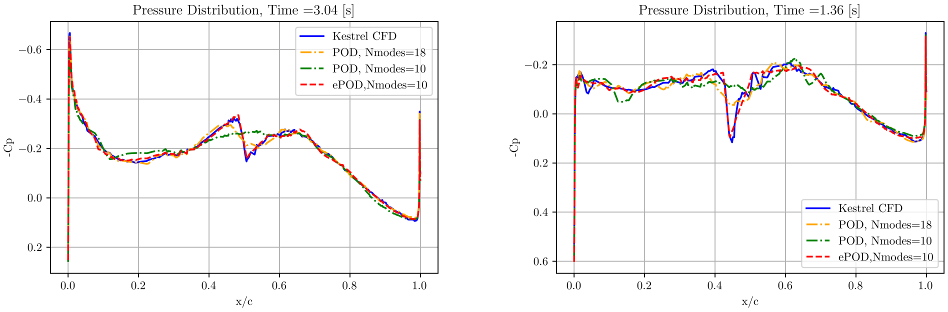
Fig. 28 Comparison between enriched and standard POD, for t=0.428 [s].



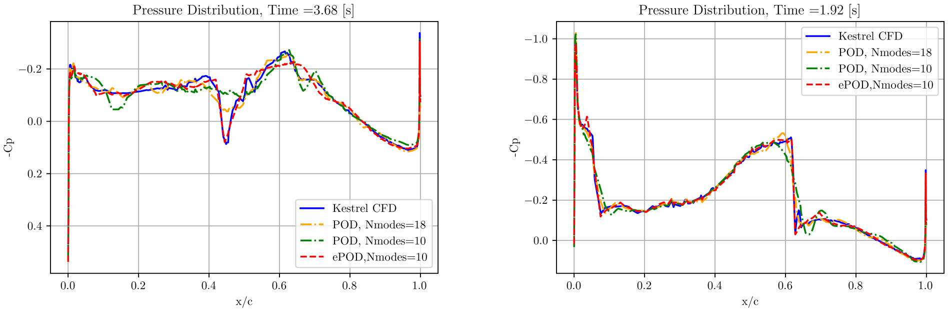
(a) Training and validation loss

(b) Performance

Fig. 29 (a) Training and validation loss. (b) Mean Squared Error, between predicted, projected and true pressure distribution.

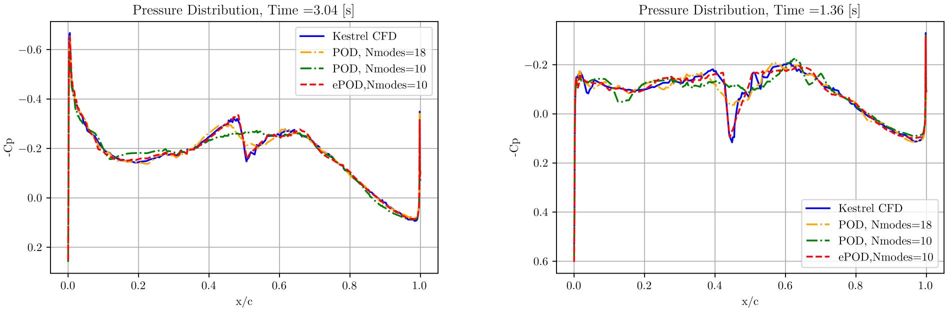


(a) t=3.04 s, AoA=5.21°   (b) t=1.36 s, AoA=17.03°

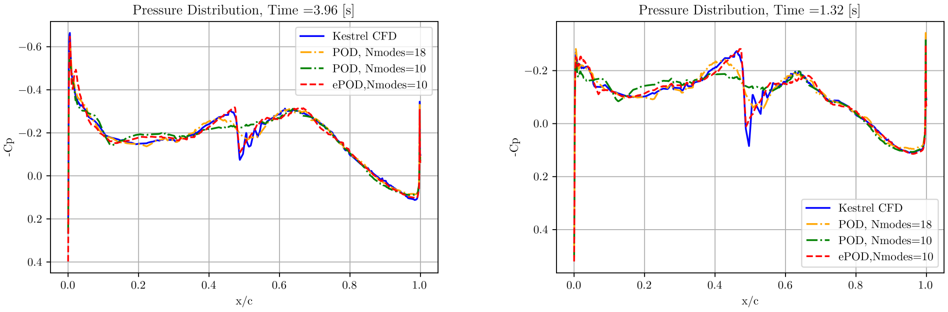


(c) t=3.68 s, AoA=8.46°   (d) t=1.92 s, AoA=0.79°

Fig. 30 Pressure coefficient reconstruction of section 1 with POD and ePOD methods, for PRBS1 signal.

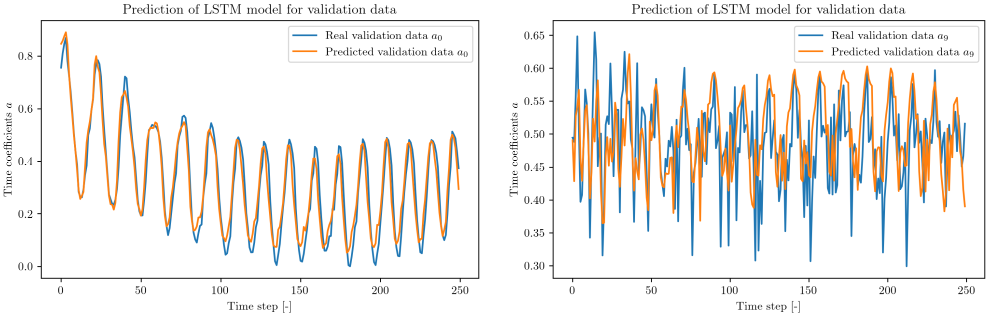


(a) t=3.04 s, AoA=4.41°   (b) t=1.36 s, AoA=3.64°

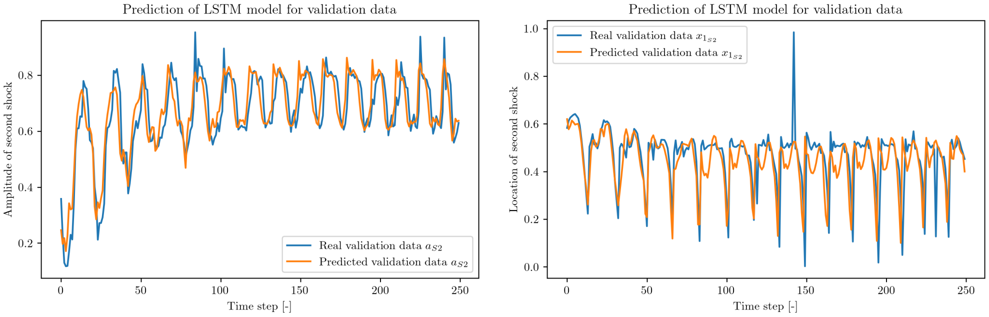


(c) t=3.96 s, AoA=4.39°   (d) t=1.32 s, AoA=4.77°

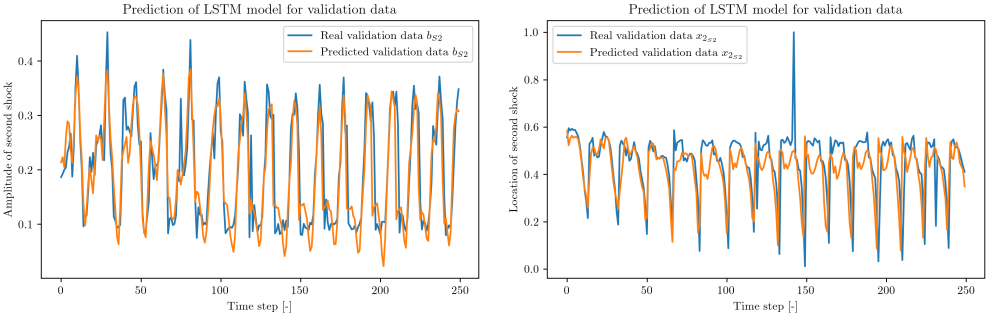
Fig. 31 Pressure coefficient reconstruction of section 1 with POD and ePOD methods, for Schroeder maneuver.



(a) First time coefficient (b) Last time coefficient

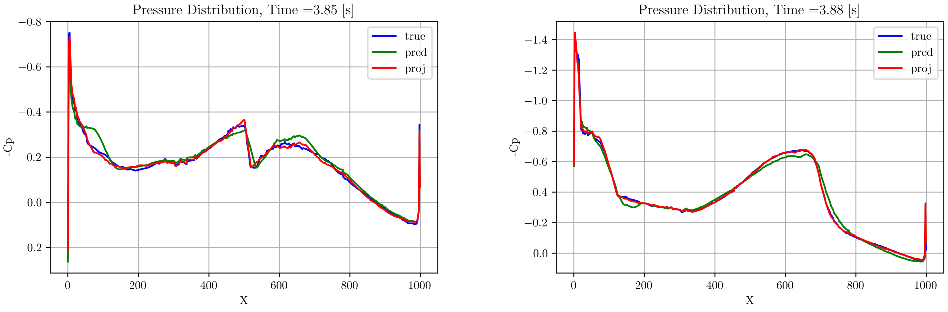


(c) Amplitude of first control point of second shock (d) Location of first control point of second shock

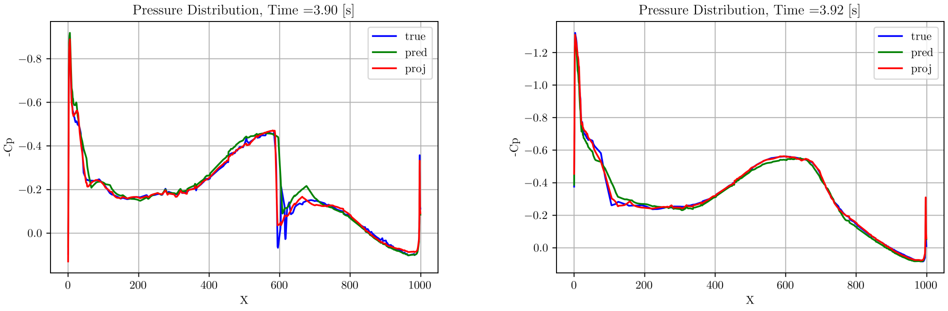


(e) Amplitude of second control point of second shock (f) Location of second control point of second shock

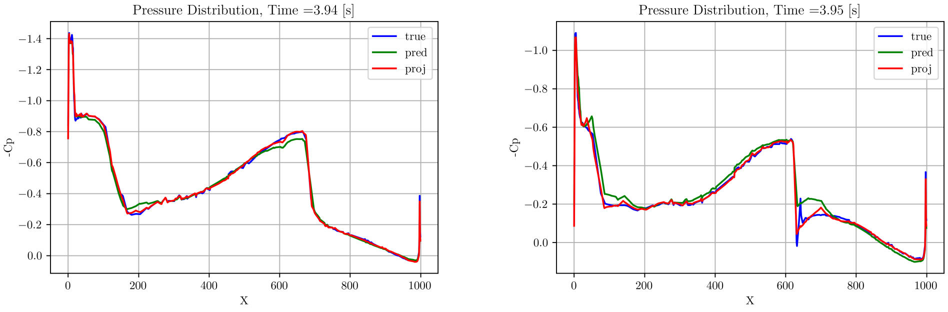
Fig. 32 Normalized validation data.



(a) t=3.04 s, AoA=5.21° (b) t=3.04 s, AoA=5.21°



(c) t=3.04 s, AoA=5.21° (d) t=3.04 s, AoA=5.21°



(e) t=3.04 s, AoA=5.21° (f) t=3.04 s, AoA=5.21°

Fig. 33 Predicted pressure distribution of section 1 with the ePOD-LSTM ROM.