REDUCED-ORDER MODELING OF UNSTEADY AERODYNAMIC LOADS USING RADIAL BASIS FUNCTION NEURAL NETWORKS

M. Winter*, C. Breitsamter†
Institute of Aerodynamics and Fluid Mechanics, Technische Universität München
Boltzmannstr. 15, 85748 Garching, Germany

Abstract

An essential task in design and certification of modern aircraft is the accurate prediction of unsteady flow-induced loads. The established techniques related to aeroelastic simulations are usually based on potential flow theory. However, in the transonic flight regime this methodology is limited in its fidelity due to distinct aerodynamic nonlinearities. Recent advancements are achieved by using computational fluid dynamics (CFD) approaches to address fluid-structure-interaction problems. Besides the accuracy improvements, the computational cost increases dramatically. Hence, the development of reduced-order models (ROMs) for aeroelastic analyses becomes a research area of increasing interest. The aim of ROM methods is to efficiently describe the dominant static and dynamic characteristics of the underlying system. Therefore, a limited set of CFD-based data is exploited to calibrate the ROM. Subsequently, the obtained model can be supplied with new inputs and ideally responds equivalent to the considered system.

In this paper, a ROM approach is presented that employs radial basis function neural networks (RBF-NN) to train the dynamic relationship between the structural motion and the resulting flow-induced loads. For selecting an optimal set of basis functions, the orthogonal least squares (OLS) training technique is utilized. Since the recurrent RBF-NN is based on nonlinear system identification principles, it is suited to describe nonlinear aerodynamic effects with sufficient accuracy. Preliminary numerical investigations on the NLR 7301 supercritical airfoil show good correlation between the results obtained by the ROM methodology in comparison to the full-order CFD solution.

NOMENCLATURE

$\mathbf{C}$	=	generalized damping matrix ($N \times N$)
$\mathbf{c}_i$	=	center vector, RBF-NN
c_{ij}	=	center vector i, element j
c_p	=	pressure coefficient
c_{ref}	=	chord length, m
e_i	=	mean squared error of output element i
$\mathbf{f}$	=	time-domain discrete force vector ($N \times 1$), N
$\mathbf{f}_{gen}$	=	time-domain generalized aerodynamic force vector ($N \times 1$)
$f_{gen,i}$	=	generalized aerodynamic force vector element i, time domain
$f_{gen,ij}$	=	generalized aerodynamic force vector element i caused by an excitation of mode j, time domain
f	=	oscillation frequency, Hz
$\mathbf{G}$	=	auxiliary matrix for weight determination
$\mathbf{GAF}$	=	generalized aerodynamic force matrix, frequency domain ($N \times N$)
$\mathbf{K}$	=	generalized stiffness matrix ($N \times N$)
k	=	time increment
k_{red}	=	reduced frequency ($2\pi f \cdot c_{ref}/U_\infty$)
$\mathbf{M}$	=	generalized mass matrix ($N \times N$)
M	=	number of neurons in the hidden layer
Ma_∞	=	freestream Mach number
m	=	maximum input delay-order, $m \in \mathbb{N}$
N	=	number of considered eigenmodes
N_S	=	number of training samples
N_u	=	number of inputs, RBF-NN
N_y	=	number of outputs, RBF-NN
n	=	maximum output delay-order, $n \in \mathbb{N}$
p	=	p-nearest-neighbor model constant
p_∞	=	freestream static pressure, N/m²
$\mathbf{q}$	=	vector of generalized coordinates ($N \times 1$)
r	=	scaling factor used for network optimization
q_∞	=	freestream dynamic pressure, N/m²
t	=	time, s
U_∞	=	freestream velocity, m/s
$\mathbf{u}$	=	input vector ($N_u \times 1$), RBF-NN
$\mathbf{W}$	=	weight matrix ($N_y \times M$), RBF-NN
$\mathbf{y}$	=	system output vector ($N_y \times 1$)
$\tilde{\mathbf{y}}$	=	model output vector ($N_y \times 1$), RBF-NN
α	=	angle of attack, deg
γ	=	ratio of specific heats, 1.4 (air)
Δt	=	dimensional time step, s
$\Delta\tau$	=	nondimensional time step
$\mathbf{\Phi}$	=	modal matrix, $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \dots, \boldsymbol{\phi}_N]$
$\boldsymbol{\phi}_i$	=	vector of eigenmode i
Θ	=	normalized excitation amplitude
ρ_∞	=	freestream density, kg/m³
σ_i	=	spread/radius of neuron i, RBF-NN
τ	=	nondimensional time
$\mathbf{\Psi}_i$	=	basis function operator, RBF-NN
ω	=	$k_{red} \cdot Ma_\infty \cdot \sqrt{\gamma}$, nondimensional angular frequency

1. INTRODUCTION

In classical aeroelastic analysis, the interaction of structural, aerodynamic, and inertial forces is studied to determine the static and dynamic characteristics of an aircraft. Fluid-structure interaction (FSI) effects are of paramount importance regarding the limits of the flight envelope and are therefore strongly influencing safety and efficiency requirements [1].

In the context of aeroelasticity, the unsteady flow-induced

* Research Engineer, Institute of Aerodynamics and Fluid Mechanics, maximilian.winter@aer.mw.tum.de
† Professor, Institute of Aerodynamics and Fluid Mechanics

loads are mainly modeled using the well-developed linear potential flow theory [2]. For industrial applications, these methods are used because of low computational effort in combination with reliable results, especially, in the subsonic and supersonic flight regime. However, for transonic flow conditions this methodology has drawbacks due to distinct aerodynamic nonlinearities such as compression shocks, which are not covered by the underlying theory [3]. Advancements can be achieved by using recently developed computational fluid dynamics (CFD) approaches to address aeroelastic problems. Besides the improvements in quality and accuracy, the computational cost increases substantially due to the large number of influence parameters that have to be studied (e.g., number of eigenmodes, Mach number, angle of attack, flap deflection angle). Today the long turn around periods are still the limiting factor for the industrial use of CFD codes in terms of predicting unsteady aerodynamic forces.
Hence, the development of reduced-order models (ROMs) for multidisciplinary analyses becomes a research area of increasing interest. The aim of ROM methodologies is to reduce the full-order flow problem (Euler or Reynolds-averaged Navier-Stokes) to a simpler mathematical model, while maintaining the dominant physical and dynamical characteristics of the complex system. Calculating the response of a reduced model is far more efficient than solving the full-order CFD equations. In order to calibrate the ROM, a dataset of CFD-based input-output data is exploited to extract the high-quality information.
In the last years, different ROM concepts have been developed to efficiently describe unsteady aerodynamic quantities. The approaches can be divided into time domain and frequency domain ROMs, though in this work only the time domain representation is considered. A brief description of the recently developed ROMs is given below.

Lucia et al. [4] as well as Dowell and Hall [5] gave a comprehensive overview of several reduced-order techniques; e.g., harmonic balance, Volterra theory, and proper orthogonal decomposition (POD), while showing their application to aeroelastic test cases.
In the literature, other approaches comprise the use of linear system identification concepts to obtain a ROM. An example is the eigensystem realization algorithm (ERA) [6] that is employed to construct a linear time invariant (LTI) state-space model [7]. For further computational efficiency, this method had been extended in order to excite several structural modes simultaneously, see Kim and Bussoletti [8], Silva [9], and Fleischer and Breitsamter [10].
Concurrently, various methods based on nonlinear system identification principles had been developed. Because of their mathematical foundation, these kind of ROMs should be able to capture large amplitude vibrations as well as (highly) nonlinear aerodynamic phenomena such as shock-induced limit-cycle-oscillations (LCO). In 1997, Faller and Schreck [11] proposed a recurrent multi-layer-perceptron neural network (MLP-NN) for the identification of experimentally gathered airfoil data. Subsequently, Marques and Anderson [12] used a multi-layer-based temporal neural network to predict unsteady aerodynamic forces in transonic flow. Voitcu and Wong [13] demonstrated that a neural network is suitable to describe the dynamics of an aeroelastic system. Mannarino and Mantegazza [14] extended those ideas by using a recurrent MLP-NN to approximate the lift and pitching-moment coefficients of the NACA 64A010 airfoil [15]. A similar approach was followed by Rampurawala et al. [16].
There, a non-recurrent MLP-NN (without feedback) was utilized to model the aeroelastic behavior of the 3D Goland wing [17]. In order to include dynamic memory effects, temporal derivatives of the excitation signal were added to the input vector of the neural network. Within the rotorcraft community, a surrogate model that accounts for time-dependent Mach number variations had been formulated by Glaz et al. [18]. For the nonlinear mapping, they used a kriging algorithm. Furthermore, Zhang et al. [19] employed a radial basis function neural network (RBF-NN) to investigate LCOs of the NACA 64A010 airfoil [15]. Recently, Lindhorst et al. [20] combined a RBF-NN with the POD approach to study the aeroelastic behavior of the HIRENASD-wing-fuselage configuration [21].

In the present paper, a similar approach as in Zhang et al. [19] is followed. A recurrent RBF-NN is utilized to train the dynamic relationship between the simultaneously-excited pitch-plunge motion of the two-dimensional NLR 7301 airfoil [22] and the respective generalized aerodynamic forces (GAFs). For this purpose, the modal deflections are used as the model input, and the unsteady aerodynamic forces are the RBF-NN's resulting output. The full-order CFD system is considered with the Euler-based AER-Eu solver [23]. Nevertheless, the ROM concept is also valid for aerodynamic data that has been generated using the Reynolds-averaged Navier-Stokes (RANS) equations. Once the theoretical background and several network training techniques have been described, the reduced-order results for the NLR 7301 airfoil in transonic flow are compared to the corresponding time-accurate AER-Eu results.

2. THEORY AND NUMERICAL METHODS

Firstly, the relevant basics regarding structural dynamics as well as the concept of generalized aerodynamic forces are introduced. Subsequently, the CFD solvers AER-Eu and AER-SDEu will be addressed briefly. The main focus is on a detailed discussion of the RBF-ROM method. Hence, the fundamentals of recurrent RBF-NNs as well as application aspects are explained.

2.1 Structural Dynamics and Generalized Aerodynamic Forces

In modern aeroelastic analysis, the structure of a flexible aircraft is usually discretized using the finite-element method (FEM) [1]. For a reduction of the numerous degrees of freedom, the initial finite-element model in physical space is often transferred into modal coordinates. Hence, the well-known equations of motion for a multi-modal structural system can be written in generalized coordinates as

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{C}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{f}_{\mathrm{gen}}(t) \tag{1a}$$

$$\mathbf{f}_{\mathrm{gen}}(t) = \mathbf{\Phi}^{\mathrm{T}}\mathbf{f}(t) \tag{1b}$$

A FEM-driven modal analysis leads to the modal matrix $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \dots, \boldsymbol{\phi}_{\mathrm{N}}]$ with N considered eigenmodes as well as the generalized mass, damping, and stiffness matrices, respectively, $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$. In Eq. (1a), vector $\mathbf{q}(t)$ represents the modal/generalized coordinates that can be interpreted as scaling factors for the corresponding mode shapes.

Taking only aerodynamic loads into account, the concept of generalized forces results in the relation for the i-th modal force vector element in the time domain:

$$f_{gen,i}(t) = q_\infty \cdot \int_S c_p(t) \cdot \boldsymbol{\phi}_i \circ d\mathbf{S} \quad (2)$$

In Eq. (2), the local pressure coefficient $c_p(t)$ is integrated over the aerodynamically effective surface and weighted with the corresponding eigenmode.
Because the linearized aerodynamic forces in the frequency domain are subsequently exploited for validation purposes, they will be briefly addressed. When performing classical flutter analyses, it is assumed that a linear relation between the structural deflection of an eigenmode and the generalized aerodynamic forces exists. Hence, the aeroelastic equations can be transferred into the frequency domain to treat harmonic motions. Thus, the well-known linear flutter equation in the frequency domain can be derived:

$$[-\omega^2\mathbf{M} + i\omega\mathbf{C} + \mathbf{K} - q_\infty \mathbf{GAF}(k_{red}, Ma_\infty)] \cdot \mathbf{q}_0 = 0 \quad (3)$$

Eq. (3) contains the introduced modal matrices, the physical angular frequency ω and the $\mathbf{GAF}$ matrix. Latter quantity represents the linear aerodynamic transfer function in the frequency domain, while $\mathbf{q}_0$ is the modal amplitude. When comparing harmonic excitations, the time domain solutions are Fourier analyzed to allow for a validation based on the real and imaginary parts of the $\mathbf{GAF}$ matrix. Further details on this approach can be found in Fleischer and Breitsamter [10].

2.2 Computational Fluid Dynamics – AER-Eu and AER-SDEu

In this research, the CFD solver AER-Eu is used to provide the subsequently described RBF-ROM with high-quality training data for the steady and unsteady aerodynamic loads. Additionally, AER-Eu and AER-SDEu are used for intermethod comparisons in order to assess the accuracy of the ROM approach. In this work, the purpose of the ROM is to reproduce the underlying CFD system, which is considered as the reference.
AER-Eu solves the Euler equations in conservation form by utilizing a shock-capturing finite-volume method for structured multi-block grids [23]. The spatial discretization is realized by Roe's flux-difference splitting [24], while the MUSCL extrapolation (from monotonic-upstream-scheme-for-conservation-laws) is employed in order to retain the total variation diminishing (TVD) property. The temporal integration is performed with the dual-time stepping scheme by Jameson, whereas the embedded pseudo-time solution is carried out using the lower-upper symmetric successive overrelaxation [25]. Furthermore, a deforming mesh approach had been implemented. For this purpose, a user-defined time law can be prescribed to interpolate between a reference grid and an amplitude grid [10].
Based on the previously described Euler code, the time-linearized AER-SDEu solver had been developed. The fundamental assumption is that a harmonic motion of the investigated body (aircraft) results also in a harmonic response of the unsteady aerodynamic forces according to the classical flutter theory. Hence, nonlinear effects are neglected and only the magnitude and phase of the first harmonic unsteady load have to be evaluated. Because of the linear relationship, it is possible to compute the aerodynamic quantities directly in the frequency domain. With this high-fidelity CFD method, a significantly lower computational effort compared to the dynamically fully nonlinear approach is required to calculate the GAFs for the linear flutter analysis. For further information, refer to [23, 26-28].

2.3 Aerodynamic Reduced-Order Model – Radial Basis Function Neural Network

In the following, the fundamentals of RBF-NNs as well as the utilized training strategies are discussed. A Markov chain-based approach for training the dynamic system behavior using a finite set of input-output data samples will be presented. Finally, some application guidelines regarding the simulation of unsteady aerodynamic loads are given.

2.3.1 Radial Basis Function Theory

Originally published by Broomhead and Lowe [29] in 1988, the RBF-NN belongs to the domain of artificial neural networks, and is able to approximate any nonlinear function to an arbitrary degree of accuracy with a finite number of neurons. Thus, the network fulfills the universal approximation theorem [30]. In Fig. 1, a schematic of a RBF-NN with one output is shown. The depicted network structure can be subdivided into three layers: the input layer for distributing the network inputs, the nonlinear hidden layer containing the RBFs and the linear output layer. Each hidden layer circle symbolizes a neuron that receives an input, processes the information in a nonlinear way, and produces an output. One single neuron allows only comparatively simple operations. However, the combination of many neurons makes this approach powerful. The mathematical formulation for a RBF-NN with an output vector consisting of N_y elements ($1 \le i \le N_y$) can be written as

$$\tilde{y}_i = \sum_{j=0}^{M} w_{ij} \cdot \boldsymbol{\Psi}_j(\|\mathbf{u} - \mathbf{c}_j\|) \quad \text{with } \boldsymbol{\Psi}_0 = 1 \quad (4)$$

where $\tilde{y}_i$ is the i-th element of the output vector, $\mathbf{W}$ a matrix containing the linear weights w_{ij}, $\mathbf{u}$ the input vector and $\mathbf{c}_j$ the center vector affiliated to neuron j. Hence, the multivariable function mapping $\mathbf{u} \rightarrow \tilde{\mathbf{y}}$ (with $\mathbf{u} \in \mathbb{R}^{N_u}$ and $\tilde{\mathbf{y}} \in \mathbb{R}^{N_y}$) is realized. For the M basis functions $\boldsymbol{\Psi}_j$ in the hidden layer, the typical approach of using the Gaussian RBF in combination with the Euclidean distance norm is inserted:

$$\boldsymbol{\Psi}_j(\|\mathbf{u} - \mathbf{c}_j\|) = \exp\left(-\frac{\sum_{k=1}^{N_u}(u_k - c_{jk})^2}{2{\sigma_j}^2}\right) \quad (5)$$

Due to the characteristics of RBFs, the output of a neuron becomes smaller when the distance between the input and the neuron's center increases. The spread parameter σ_j in Eq. (5) determines the sphere of influence of neuron j and, therefore, has a strong nonlinear effect on the overall network performance. With larger spreads, the width of the Gaussian RBF increases and therefore covers a greater regime of the input space. Thus, $\sigma_j \rightarrow 0$ implies that only an input vector equal the center vector activates neuron j.
Recalling that only input and output vectors are known for

a given training data-set, the centers, weights, and spreads have to be trained to realize a RBF-NN-based model.

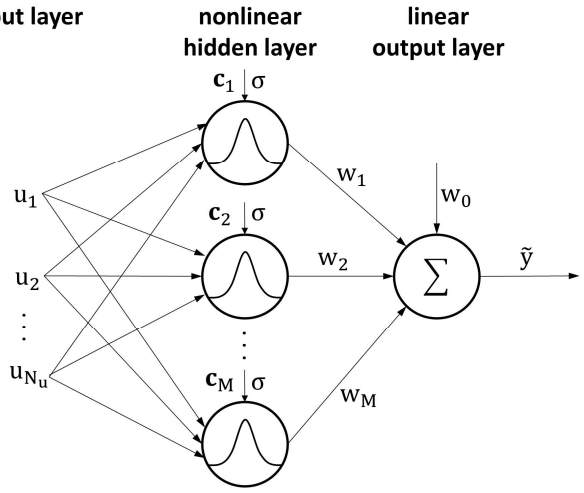

Fig 1: Schematic representation of a RBF-NN with one output $\tilde{y}$ and a constant spread σ

2.3.2 Training of the RBF-NN

In order to train the network, various methods are given in the literature. A detailed review of all approaches is beyond the scope of this paper. Hence, only the employed instances in this research are given. Further information can be found in Haykin [31] and Nelles [30].
Firstly, the approach for determining the weight matrix is examined. Since the weights can be interpreted as scaling factors for the corresponding RBFs, the overall network output is a linear superposition of all neuron outputs. For training those quantities, it is assumed that the basis functions are fixed and known a priori. Thus, a least squares formulation results for the linear weights that can be solved by using the commonly employed pseudo-inverse method [31]:

$$\mathbf{W} = (\mathbf{G}^{\mathrm{T}}\mathbf{G})^{-1}\mathbf{G}^{\mathrm{T}}\,\mathbf{y} \tag{6a}$$

$$\mathrm{G_{ij}} = \exp\left(-\frac{\left\|\mathbf{u}_{\mathrm{i}} - \mathbf{c}_{\mathrm{j}}\right\|^2}{2{\sigma_{\mathrm{j}}}^2}\right) \tag{6b}$$

Its calculation is extremely efficient compared to any nonlinear optimization technique, e.g., required for MLP-NN training. By solving Eq. (6a), the mean squared error regarding the estimated network output $\tilde{\mathbf{y}}$ and the exact system response $\mathbf{y}$ is minimized for a fixed set of $\boldsymbol{\Theta}_{\mathrm{j}}$:

$$e_i = \frac{1}{N_S}\sum_{j=1}^{N_S}\left(\tilde{y}_{ij} - y_{ij}\right)^2 \qquad \text{for } 1 \le i \le N_y \tag{7}$$

In Eq. (7), N_S represents the number of sample points that has been used for training the network.

In the preceding paragraph, it was assumed that the RBFs of the hidden layer, and hence the centers, are known. In the following, two variants for center selection are discussed briefly:

- One possible solution is to use every training sample point as a center. A consequence is that there are $M = N_S$ neurons. However, a large number of neurons should be avoided because of model overfitting effects (see Nelles [30], for instance). Hence, this approach should be used only for verification purposes.

- In the present research, the orthogonal least squares (OLS) method for linear subset selection as proposed by Chen et al. [32] has been utilized. This supervised-learning method exploits the benefits of the linear relationship between the RBFs and the weights to select the centers. Because a detailed discussion of the algorithm is beyond the scope of this paper, only the most relevant steps are presented:
  1) At the beginning, all training samples N_S are used to train a separate RBF-NN with one center, assuming a user-defined initial spread for all dimensions. For those N_S networks, the mean squared error related to the whole training data-set is evaluated, see Eq. (7). Subsequently, the center that causes the smallest prediction error is selected as the first network center vector (regressor).
  2) After the initial step, another training sample is made orthogonal to the previously chosen regressor/s by using the modified Gram-Schmidt orthogonalization. This procedure is iterated for each remaining sample. In analogy to step one, the error is evaluated, while the regressor that yields the smallest error is added as a neuron's center.
  3) Step two is repeated until a prescribed maximum number of neurons is reached or the mean squared error is smaller than a user-defined limit.

  By using the OLS approach, the available information about the exact system behavior is fully exploited. Because the initial spread has to be defined manually, further iterations due to an unfavorable choice may be necessary. As a result of the OLS method, only the most relevant sample points (in terms of the prediction error) are used as center vectors, thus the risk of undesirable over- and underfitting phenomena is minimized. Moreover, the resulting network is very compact and thereby allows an extremely fast calculation.

Lastly, methods for determining the spreads will be discussed. Two instances are employed here:

- The simplest approach is to choose a constant spread ($\sigma_{\mathrm{j}} = \sigma = \mathrm{const.}$) for all neurons. The value of the spread parameter can be optimized by a brute-force search approach in a user-defined interval. Once the procedure is terminated, the spread that yields the smallest prediction error according to Eq. (7) is selected.

- The more sophisticated approach is to allow a variable spread. In the late 1980's, Moody and Darken [33] presented the p-nearest-neighbor method. For each of the M selected centers, the p nearest centers are determined based on the Euclidean distance. Then the j-th spread value can be expressed by

$$\sigma_{\mathrm{j}} = \mathrm{r}\cdot\frac{1}{\mathrm{p}}\sqrt{\sum_{\mathrm{i}=1}^{\mathrm{p}}\left\|\mathbf{c}_{\mathrm{j}} - \mathbf{c}_{\mathrm{i}}\right\|^2} \tag{8}$$

A value of $p = 2$ or $p = 3$ is suggested to achieve adequate results. Optionally, the factor r can be handled like a constant spread. It allows a computational brute-force optimization to improve the network prediction capability.

2.3.3 Unsteady Aerodynamic Simulation using the RBF-NN

Up to this point, the discussed RBF-NN can be used to realize the mapping of a nonlinear function. However, in order to take time dependencies and dynamic memory effects into account, the Markov chain approach must be introduced to obtain a recurrent network.
Many widely-used parametric system identification techniques (e.g., ARX and ARMA [30]) are based on the assumption that the known relationship between a finite series of former inputs and previous outputs is sufficient to predict the system response at the subsequent time step. This approach for modeling dynamic systems can be written in the following form, defining that k is the considered current discrete-time increment:

$$\mathbf{y}(k) = f\begin{bmatrix}\mathbf{u}(k), \mathbf{u}(k-1), \dots, \mathbf{u}(k-m), \\ \mathbf{y}(k-1), \dots, \mathbf{y}(k-n)\end{bmatrix} \tag{9}$$

In this manner, not only the current system input is considered for the model input vector, but also previous system inputs and outputs. The maximum dynamic delay-orders depicted in Eq. (9) are m for the inputs and n for the outputs, respectively. Furthermore, it is assumed that a constant time step $\Delta\tau$ $(k-1 \rightarrow k)$ is employed. Because the function f in Eq. (9) is typically nonlinear and multidimensional, the RBF-NN is used to model the spatial-temporal relation. For aeroelastic computations, the input vector $\mathbf{u}$ symbolizes the structural states and can be expressed by the modal deflections $\mathbf{q}$. By definition, the output vector $\mathbf{y}$ can contain either the aerodynamic lift and pitching-moment coefficients (in the 2D case) or the generalized aerodynamic forces.
Because the dynamic delays have to be specified by the user, some guidelines are given in the following: Generally, there is no mathematical restriction on the number or the sequence of the dynamic delays. However, to circumvent stability problems, the maximum output delay should be smaller than or equal to the maximum input delay. To find a good delay sequence, several simpler models (e.g., ARX, ARMA) with different delay schemes can be efficiently estimated. Then, a comparison of the prediction errors of these models according to Eq. (7) leads to a first guess for the dynamic delays of the nonlinear RBF-NN. However, finding a good delay-order pattern is an iterative task.
Assuming that a properly trained network for the considered system is given, and the dynamic delays are optimized, an unsteady dynamic simulation can be realized. It is important to highlight that only constant time steps can be handled by this approach. Hence, the same discrete time step incorporated in the training data-set must be employed for the simulation.
If the output is fed back, i.e., if output delays are used, a discrete time-marching approach must be employed, as shown in Fig. 2.

To carry out unsteady aerodynamic simulations using the RBF-NN, two options exist:

1) The complete input time-series is known: This is the case for simulating forced-motion vibrations or impulse excitations. Those structural motions are of particular interest in order to obtain the generalized aerodynamic forces in the frequency domain via Fourier analysis.

2) Past and current inputs are known, future inputs are unknown: If a coupled aeroelastic system is considered, the structural displacements and the aerodynamic loads mutually influence each other. Therefore, future inputs (structural states) depend on current outputs (aerodynamic forces). Hence, for the aeroelastic ROM, a time-marching coupling scheme is necessary; see Zhang et al. [19], for instance.

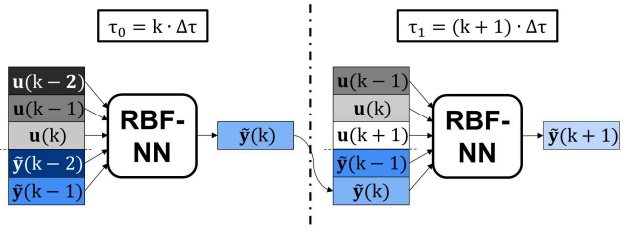

Fig. 2: Simulation of a single-input single-output (SISO) model with a maximum delay-order of two

Both described simulation types can be treated with the discussed network-based approach. However, special care must be taken at the beginning of the simulation: The RBF-NN expects as many previous inputs and outputs as defined by the maximum delay-order. Therefore, the initial condition for the structural excitation is typically set to zero or to an aeroelastic equilibrium state. The aerodynamic quantities (output) have to be defined according to the corresponding structural deflection. Thus, the initial values are usually known from the steady-state CFD solution.

2.4 Generation of Training Data - Structural Excitation

For identifying linear dynamic systems, the impulse or step excitation involving one distinct amplitude is sufficient [10, 30]. However, the nonlinear-system-identification approach requires more information about the underlying system in order to train the nonlinear ROM. Hence, the design of an adequate excitation signal is a challenging task. The signal has to stimulate the dominant and representative characteristics (amplitudes and frequencies) of the aerodynamic system. Concurrently, the generation of the training data-set must not be too computationally demanding to exploit the ROM advantages. Various excitation strategies and theories are given in the literature in order to fulfill these prerequisites. Here, the amplitude-modulated pseudo-random binary signal (APRBS) [30] is chosen for the forced-motion structural excitation. This signal can be generated from the frequently used pseudo-random binary signal (PRBS) by assigning random amplitudes to each plateau level. The main advantage of utilizing the APRBS is the large spectrum of excited frequencies and amplitudes. This property is of paramount importance for the nonlinear identification task. Furthermore, only a comparatively short excitation time-series is needed, limiting the computational cost. An example APRBS as well as the corresponding power spectral density is visualized in Fig. 3.

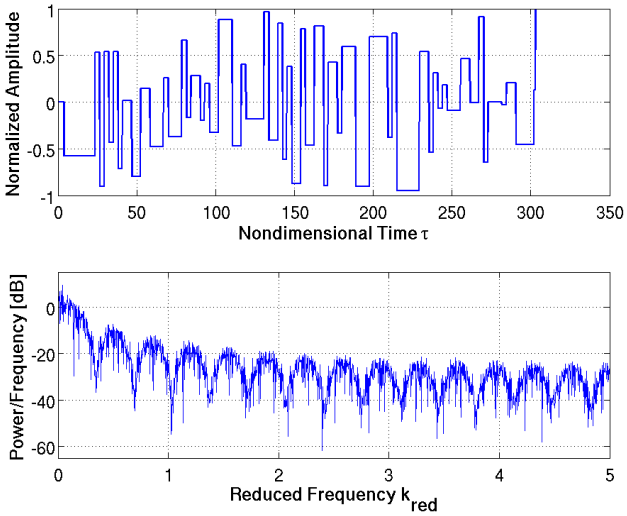

Fig. 3: APRBS signal in the time domain and the corresponding power spectral density

For each modal coordinate, a different APRBS signal is generated. Thus, coupling effects regarding the structural eigenmodes are captured.

2.5 Overview of the RBF-ROM Method

To summarize the discussed methods, a composite of the multidisciplinary approaches is given in Fig. 4.
The basis of all further computations is a steady-state AER-NS/Eu simulation for a fixed set-up characterized by the freestream Mach number and the angle of attack. Subsequently, an unsteady CFD solution is computed using one generated APRBS for each structural eigenmode. In this manner, the considered body is formed by a linear superposition of all modal shapes multiplied by the corresponding time-varying APRBS values. Hence, the AER-NS/Eu result is a time-series of the particular generalized aerodynamic force vector resulting from the prescribed motion.
Combining the APRBS inputs with the CFD-based outputs, the merged data-set must be preprocessed in order to train the RBF-NN. Therefore, the delay pattern is needed to arrange the data according to the Markov chain approach. Furthermore, the data is normalized to improve the numerical robustness. After the preprocessing step, the training data can be utilized to train the unknown quantities of the neural network.

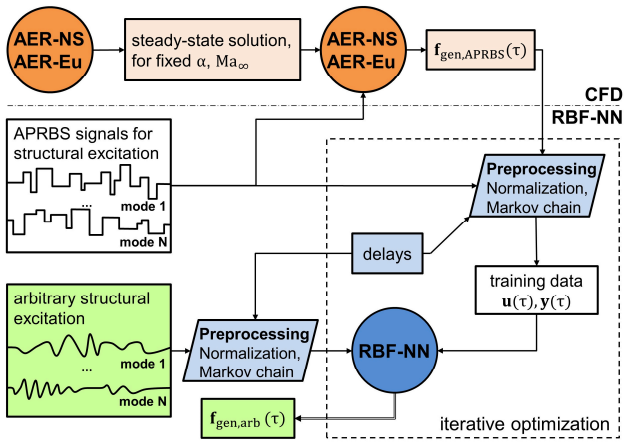

Fig. 4: Scheme of the RBF-ROM method

Because of the iterative nature of finding an optimal delay pattern and RBF-NN configuration, a further optimization may be necessary. Therefore, the available CFD-based data-set is typically segmented, while around 70% of the data is used for training. The remaining data is exploited for verification purposes in order to realize a stable and accurate model.
If the RBF-NN-based model optimization yields satisfactory results, the ROM can be used for aerodynamic simulations. Therefore, an arbitrary structural excitation can be preprocessed and fed into the RBF-ROM. However, the excitation amplitudes should be covered by the training data to achieve adequate results. After renormalization, the unsteady aerodynamic forces of the modeled system are obtained.

3. RESULTS

In order to demonstrate the RBF-ROM approach, the method is applied to the two-dimensional NLR 7301 transonic airfoil [22, 34], while the RBF-NN algorithms are implemented in MATLAB [35]. For this 2D case, the accuracy as well as the computational efficiency is analyzed and compared to the respective AER-Eu simulations. For a further comparison of the RBF-ROM with AER-SDEu, the ROM-based generalized aerodynamic forces resulting from harmonic excitations are transferred to the frequency domain via Fourier analysis.

3.1 NLR 7301 Airfoil - Setup

The frequently investigated NLR 7301 airfoil offers a distinct nonlinear aerodynamic behavior in the transonic flight regime due to the presence of a strong shock and is thereby suited to assess the simulation accuracy of the nonlinear RBF-ROM approach. According to Weber et al. [36] and Tang et al. [34], the supercritical wing is studied at $\mathrm{Ma}_\infty = 0.753$ and $\alpha = 0$. Furthermore, the investigated 2D airfoil is characterized by a chord length of $c_{\mathrm{ref}} = 0.3\ \mathrm{m}$, which is also interpreted as the reference length for computing the reduced frequency. Since the focus is on intermethod comparison, arbitrary excitations are considered in this work, while the ROM outputs are compared to the CFD results. Because this paper presents preliminary solutions, only inviscid flow was taken into account, although viscous effects have a huge impact on the aerodynamic and aeroelastic characteristics [36]. The consideration of viscous flow remains a topic for future research.
The linear two-degree-of-freedom structural model for the NLR 7301 airfoil is adopted from Tang et al. [34]. Once the modal analysis is performed, the resulting eigenmodes are adequately scaled for the unsteady aerodynamic simulations. In this context, two conflicting requirements occurred: Because the nonlinear effects are of major concern, the deflections should be large in order to induce distinct shock motions. However, for comparison with the AER-SDEu solutions, the maximum deflections are limited, due to the linearization assumption. As a trade-off, the maximum deflection of the plunge-dominated mode was scaled to 3.3 percent of the chord length while the pitching-dominated eigenmode was scaled to a maximum angle of attack of 3 degree. The deformed CFD grids are generated from the scaled modal shapes and the reference CFD grid by using the thin-plate spline (TPS) interpolation approach followed by the transfinite interpolation (TFI) for mesh reconstruction.

The employed reference CFD grid is discretized by 14,396 cells in a four-block C-H topology using the commercial ICEM-CFD software package [37]. A grid-sensitivity study was carried out to ensure the independence of the solution from the spatial discretization. In Fig. 5, the CFD grid is visualized along with the steady-state Mach number distribution. The figure shows a typical transonic flow field with a crisp shock on the upper surface.

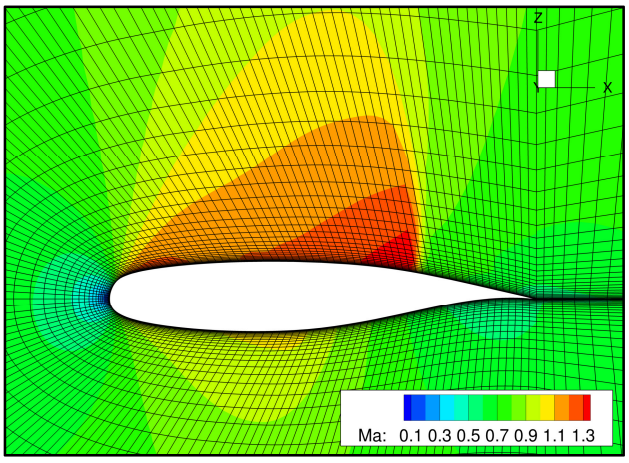

Fig. 5: Structured reference CFD grid and computed steady-state Mach number contours of the NLR 7301 airfoil at $\mathrm{Ma}_\infty = 0.753$ and $\alpha = 0$ (AER-Eu)

3.2 Unsteady Aerodynamic Results

Originating from the steady-state solution, the unsteady flow computation was conducted to evaluate the response of the aerodynamic system to the simultaneously superimposed modal deflection. The APRBS-based time law for each generalized coordinate is shown in the top plot of Fig. 6. As a result of the particular AER-Eu simulation, the time-series of the generalized force vector $\mathbf{f}_{\mathrm{gen}}(\tau)$ is obtained. Fig. 6 shows the first component of this vector as a function of the nondimensional simulation time. In this work, a nondimensional time step of $\Delta\tau = 0.1$ was employed. The relation to the physical time step is defined as

$$\Delta t = \sqrt{\frac{p_\infty}{\rho_\infty}} \cdot \Delta\tau \tag{10}$$

Based on the excitation signal, the aerodynamic response is computed for 3042 discrete time steps. Thus, a 2 x 3042 matrix results for the GAFs as well as the structural input. In the following, 70% of this data is considered as the training set for conditioning the nonlinear RBF-ROM. The remaining 30% of the set is used for model verification.

According to the workflow proposed in Fig. 4, each input and output quantity is normalized by removing the mean value and dividing by the corresponding standard deviation. Subsequently, the data has to be arranged according to the Markov chain approach. Therefore, the maximum delay for the inputs, as well as for the outputs, was specified to be 12, while all delays between 1, 2, …, 12 are included as well. Because the current/undelayed input is also embedded, the network input vector consists of 2 x 13 + 2 x 12 = 50 elements. The output vector is formed by the undelayed GAFs.

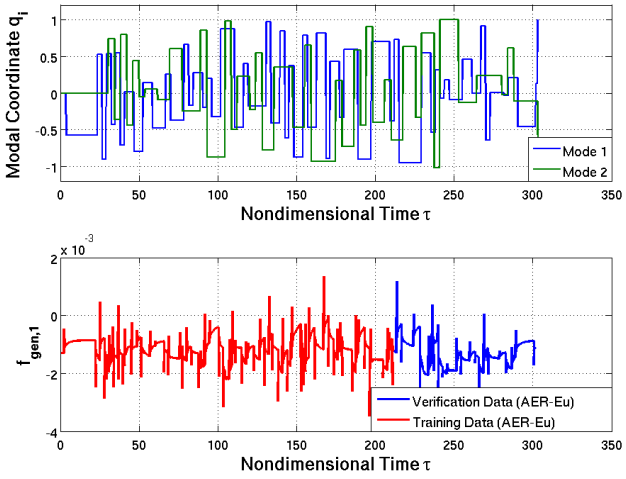

Fig. 6: Training and verification data consisting of the two simultaneously excited modal coordinates and the generalized aerodynamic force responses (only first component is shown); NLR 7301 airfoil at $\mathrm{Ma}_\infty = 0.753$ and $\alpha = 0$

After defining the input-output vectors, the RBF-NN is trained. Firstly, the OLS algorithm was used to select an optimal set of centers based on the available training data. If the prediction error was no longer reduced by adding further neurons, the selection process was terminated. Eventually, a total number of 1477 centers were selected out of 2122 available regressors. Furthermore, the p-nearest-neighbor model specified in Eq. (8) was used for the spreads in combination with the brute-force scaling factor optimization. Lastly, the linear weights are determined in order to minimize the mean squared error. It is important to highlight that this trained RBF-NN is used for all further ROM computations presented below.
After concluding the training process, a first test was conducted to check whether the ROM is able to yield adequate results. Therefore, the RBF-NN is excited by the combined training and verification APRBS (top plot of Fig. 6). Because the network was trained with 70% of this data-set, the ROM results should match the CFD reference solution quite well. However, the network training procedure minimizes only the prediction error. When a simulation with a ROM containing output delays is performed, the former calculated outputs are fed back as future network inputs. This can cause instabilities even for extremely small one-step prediction errors. Hence, the distinction between the simulation error and the prediction error must be taken into account when comparing the results. As shown in Fig. 7 for $f_{\mathrm{gen},2}$, the result nevertheless agrees well with the underlying AER-Eu computation. This also holds true for the GAF element that is not displayed here.

Next, the simulation of an arbitrary structural excitation for the same underlying system was conducted. In the context of unsteady aerodynamic RBF-ROM computations, the system is viewed as identical if Ma_∞, α, and $\Delta\tau$ remain unchanged; i.e., in accordance with the initial training data. For the following test, an amplified frequency-modulated vibration signal is defined for each modal coordinate, as shown in the top plot of Fig. 8.

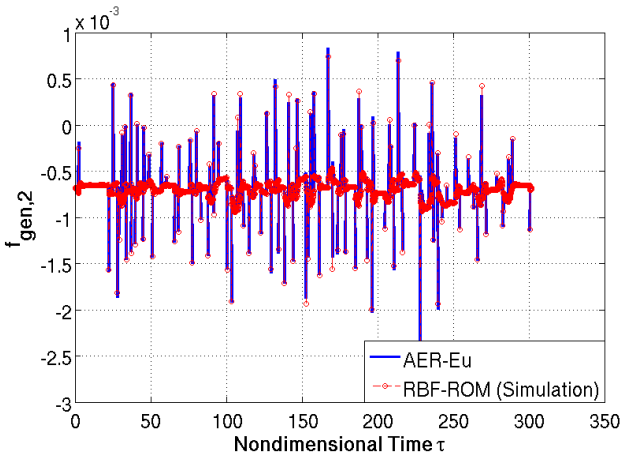

Fig. 7: APRBS simulation result of the RBF-ROM for $f_{gen,2}$; NLR 7301 airfoil at $Ma_\infty = 0.753$ and $\alpha = 0$

Subsequently, the pre-conditioned ROM is excited with the generic structural deflection. In order to evaluate the model quality, the corresponding AER-Eu solution at $Ma_\infty = 0.753$ and $\alpha = 0$ was rendered, and is plotted for $f_{gen,1}$ in Fig. 8. It is apparent that only slight deviations from the exact solution are present. Furthermore, the computation of the RBF-NN simulation for this case took approximately 3 seconds, whereas the full-order CFD solution needed more than 28 hours on an Intel Xeon 2.3 GHz processor.

Finally, three harmonic oscillations in each of the two eigenmodes were simulated in the time domain for various k_{red} using the RBF-ROM method, as well as the AER-Eu solver. For this test, the modes are not excited simultaneously to investigate the separate influence of each mode on the generalized aerodynamic forces. The modal deflection for this case can be expressed by the normalized amplitude

$$\Theta = \sin[\omega \cdot (k \cdot \Delta\tau)] \quad \text{with } k \in \mathbb{N} \tag{11}$$

The resulting aerodynamic responses are visualized in Fig. 9 by way of the respective Lissajous figures. The force $f_{gen,21}$ represents the second GAF component resulting from an excitation of the first eigenmode. One can observe good agreement between the ROM- and the CFD-generated Lissajous figures, even if nonlinear characteristics are dominant. Thus, the RBF-ROM is suited to model dynamic nonlinearities.
Additionally, the time-domain generalized aerodynamic forces resulting from the single-frequency vibrations are transformed into frequency domain using the Fourier analysis. This process is carried out for both the RBF-ROM and the AER-Eu time-series. Hence, the GAF matrix introduced in Eq. 3 results for each k_{red}. Further comparative solutions are computed with the AER-SDEu method. Real and imaginary parts for the frequency domain GAFs are depicted in Fig. 10. The diagrams indicate good conformity of the three methods, while the ROM solution offers a compromise between the nonlinear and the small-disturbance results. Latter holds true for the fidelity, especially, if distinct nonlinearities are present. However, higher frequencies (not shown here) are not covered well by this employed RBF-ROM due to the fixed discrete time step. In order to resolve vibrations of higher k_{red}, a smaller $\Delta\tau$ must be used throughout the workflow.

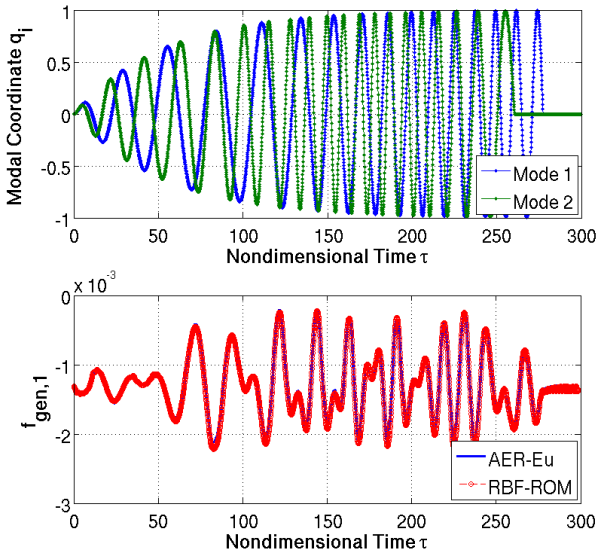

Fig. 8: Test of the RBF-ROM for a generic vibration signal; NLR 7301 airfoil at $Ma_\infty = 0.753$ and $\alpha = 0$

The computational effort of a ROM simulation (~3000 discrete time steps) is merely a few seconds on an Intel Xeon 2.3 GHz processor. However, for a total cost analysis, the generation of the training data (APRBS-CFD simulation; here: 20.8 CPU hours) as well as the training cost (0.1 CPU hours) must be considered as well. Therefore, about 21 CPU hours were needed for generating the shown RBF-ROM. In comparison, the AER-Eu computation based on the generic test signal (see Fig. 7) took about 28.3 CPU hours. Moreover, the evaluation of the harmonic excitations for six k_{red} with the AER-Eu solver required 21.3 CPU hours, while the trained ROM generated the outputs for 21 k_{red} in less than a minute. Thus, with an increased number of intended simulations, the computational advantage of the ROM increases substantially.

4. CONCLUSIONS

In this paper, an aerodynamic reduced-order modeling approach based on radial basis function neural networks was presented. The related theoretical fundamentals of aeroelasticity, artificial neural networks and nonlinear system identification were discussed. Additionally, an overview of the workflow with respect to network training strategies was given. Subsequently, the application of the RBF-ROM approach to a 2D aeroelastic problem, namely the NLR 7301 supercritical airfoil, has been shown. It was demonstrated that the neural network-based approach can adequately predict linear as well as nonlinear aerodynamic effects. This is of major concern in aerodynamically-driven limit-cycle-oscillation analyses or nonlinear response calculations, especially, in the transonic flight regime. Furthermore, the analysis of the total computational demand shows that a great potential for saving simulation time exists.
Future research will focus on a refinement of the presented method. This will include the implementation of improved parameter optimization algorithms as well as other network approaches to yield a further improvement in simulation fidelity while maintaining the model stability. Another aim will be to integrate the Mach number as a network input parameter in order to account for flow variations with a single model. Moreover, the application of the RBF-ROM method to RANS solutions will be investigated.

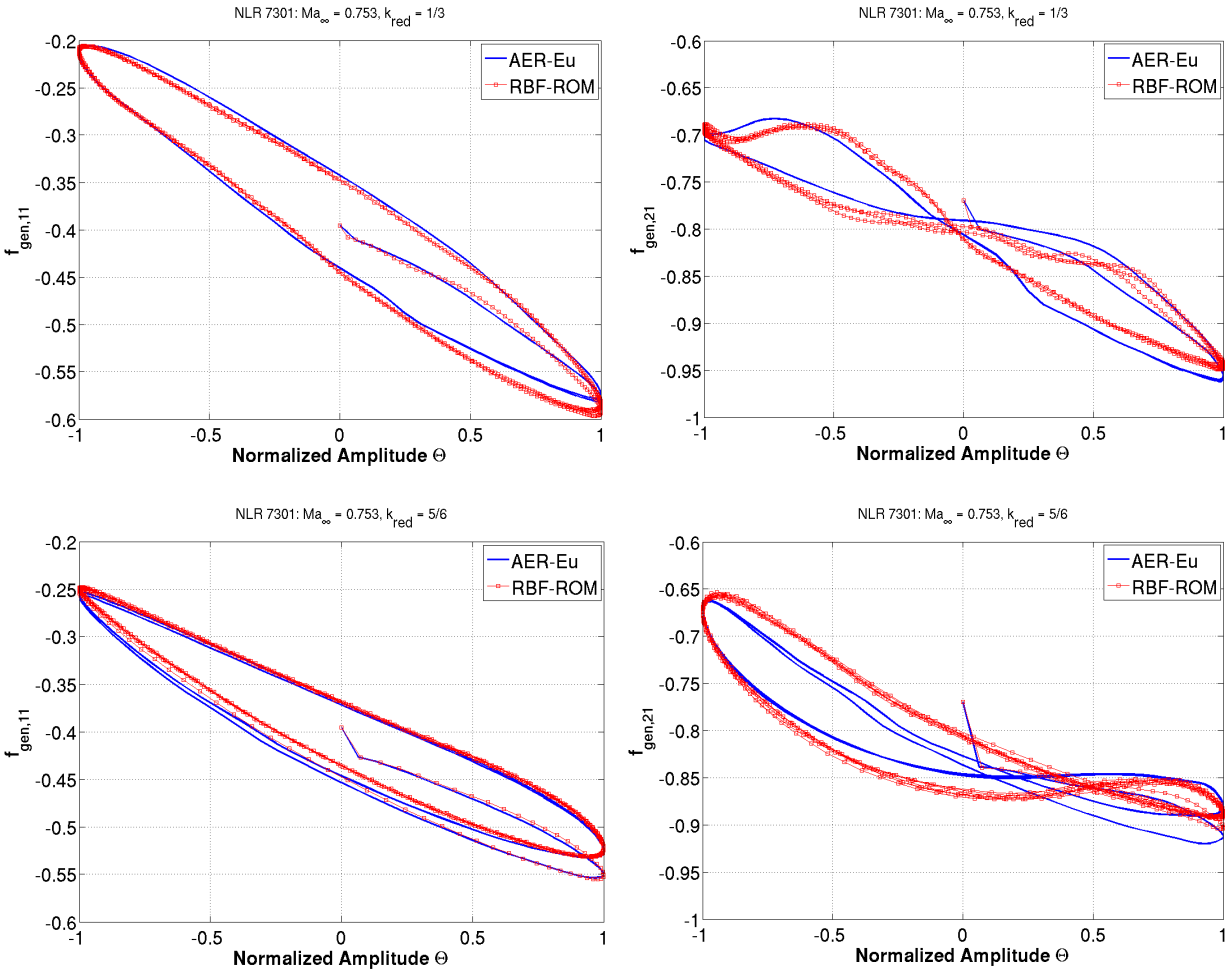

Fig. 9: Lissajous figures of the generalized aerodynamic forces $f_{gen,i1}$ resulting from three harmonic excitation cycles of the first eigenmode. Comparison of RBF-ROM results with AER-Eu results; NLR 7301 airfoil at $Ma_\infty = 0.753$ and $\alpha = 0$. Top: $k_{red} = 1/3$, Bottom: $k_{red} = 5/6$

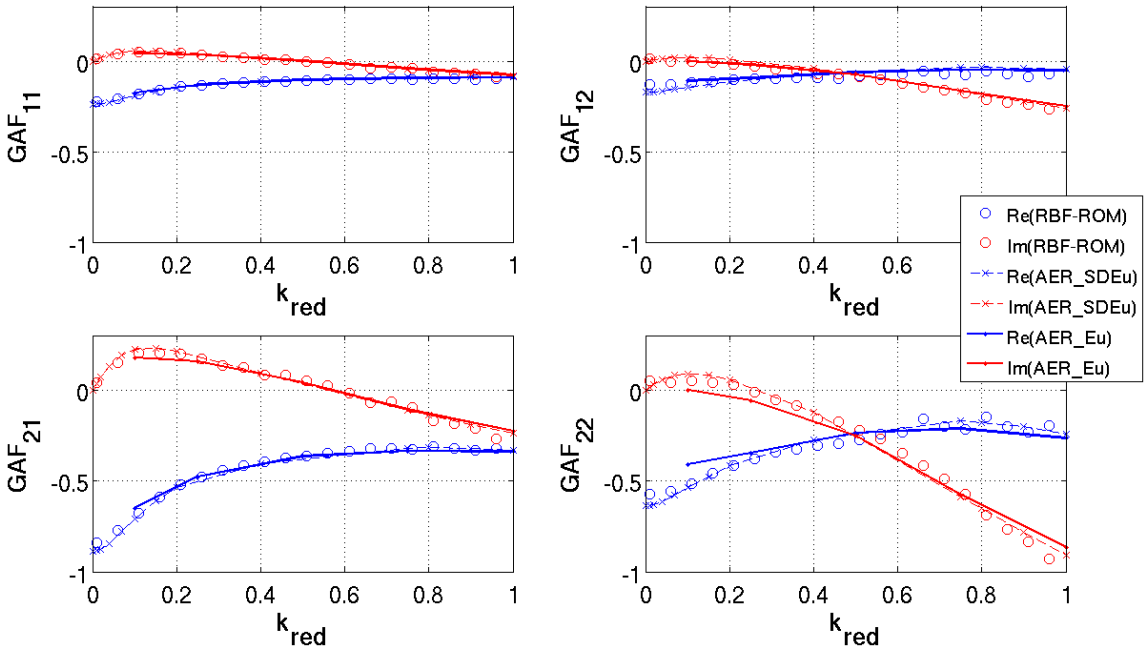

Fig. 10: Real and imaginary parts of the frequency-domain GAF matrix elements over k_{red} at $Ma_\infty = 0.753$ and $\alpha = 0$ for the NLR 7301 supercritical airfoil. Comparison between AER-Eu, AER-SDEu, and RBF-ROM.

REFERENCES

[1] J. R. Wright and J. E. Cooper, Introduction to Aircraft Aeroelasticity and Loads, West Sussex, England, U.K.: John Wiley & Sons, 2007.

[2] O. O. Bendiksen, "Review of Unsteady Transonic Aerodynamics: Theory and Applications," *Progress in Aerospace Sciences, Vol. 47,* pp. 135-167, 2011.

[3] K. J. Badcock, S. Timme, S. Marques and H. Khodaparast, "Transonic Aeroelastic Simulation for Instability Searches and Uncertainty Analysis," *Progress in Aerospace Sciences, Vol. 47,* pp. 392-423, 2011.

[4] D. J. Lucia, P. S. Beran and W. A. Silva, "Reduced-Order Modeling: New Approaches for Computational Physics," *Progress in Aerospace Sciences, Vol. 40,* pp. 51-117, 2004.

[5] E. H. Dowell and K. C. Hall, "Modeling of Fluid-Structure Interaction," *Annual Review of Fluid Mechanics, Vol. 33,* pp. 445-490, 2001.

[6] J. N. Juang and R. S. Pappa, "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," *Journal of Guidance, Vol. 8, No. 5,* pp. 620-627, 1984.

[7] W. A. Silva and R. E. Bartels, "Development of Reduced-Order Models for Aeroelastic Analysis and Flutter Prediction using the CFL3Dv6.0 Code," *Journal of Fluids and Structures, Vol. 19,* pp. 729-745, 2004.

[8] T. Kim and J. E. Bussoletti, "An Optimal Reduced-Order Aeroelastic Modeling based on a Response-based Modal Analysis of Unsteady CFD Models," 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, Seattle, WA, April 2011; also: AIAA Paper 2001-1525.

[9] W. A. Silva, "Simultaneous Excitation of Multiple-Input Multiple-Output CFD-Based Unsteady Aerodynamic Systems," 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, HI, April 2007; also: AIAA Paper 2007-1988.

[10] D. Fleischer and C. Breitsamter, "Efficient Computation of Unsteady Aerodynamic Loads Using Computational-Fluid-Dynamics Linearized Methods," *Journal of Aircraft, Vol. 50, No. 2,* pp. 425-440, 2013.

[11] W. E. Faller and S. J. Schreck, "Unsteady Fluid Mechanics Applications of Neural Networks," *AIAA 95-0529,* 9-12 January 1995.

[12] F. D. Marques and J. Anderson, "Identification and Prediction of Unsteady Transonic Aerodynamic Loads by Multi-Layer Functionals," *Journal of Fluids and Structures, Vol. 15,* pp. 83-106, 2001.

[13] O. Voitcu and Y. S. Wong, "A Neural Network Approach for Nonlinear Aeroelastic Analysis," AIAA, Denver, Colorado, 2002; also: AIAA Paper 2002-1286.

[14] A. Mannarino and P. Mantegazza, "Nonlinear Aeroelastic Reduced Order Modeling by Recurrent Neural Networks," *Journal of Fluids and Structures, Vol. 48,* pp. 103-121, 2014.

[15] J. P. Thomas, E. H. Dowell and K. C. Hall, "Nonlinear Inviscid Aerodynamic Effects on Transonic Divergence, Flutter, and Limit-Cycle Oscillations," *AIAA Journal, Vol. 40, No. 4,* pp. 638-646, 2002.

[16] A. Rampurawala, K. Badcock and S. Marques, "ANN based ROM for the Prediction of Unsteady Aeroelastic Instabilities," IFASD, Seattle, WA, 2009, also: Paper IFASD-2009-32.

[17] P. S. Beran, N. S. Knot, F. E. Eastep and et al., "Numerical Analysis of Store-induced Limit Cycle Oscillations," *Journal of Aircraft, Vol. 41, No. 6,* pp. 1315-1326, 2004.

[18] B. Glaz, L. Liu and P. P. Friedmann, "Reduced-Order Nonlinear Unsteady Aerodynamic Modeling using a Surrogate-based Recurrence Framework," *AIAA Journal, Vol. 48, No. 10,* pp. 2418-2429, 2010.

[19] W. Zhang, B. Wang, Z. Ye and J. Quan, "Efficient Method for Limit Cycle Flutter Analysis by Nonlinear Aerodynamic Reduced-Order Models," *AIAA Journal, Vol. 50, No. 5,* pp. 1019-1028, 2012.

[20] K. Lindhorst, M. C. Haupt and P. Horst, "Reduced-Order Modeling of Nonlinear, Transient Aerodynamics of the HIRENASD Wing," IFASD, Bristol, U.K., 2013; also: Paper IFASD-2013-6A.

[21] L. Reimer, A. Boucke, J. Ballmann and M. Behr, "Computational Analysis of High Reynolds Number Aero-Structural Dynamics (HIRENASD) Experiments," IFASD, Stockholm, Sweden, 2009; also: Paper IFASD-2009-130.

[22] G. Schewe and H. Deyhle, "Experiments on Transonic Flutter of a Two-Dimensional Supercritical Wing with Emphasis on the Non-Linear Effects," in *Proceedings of the Royal Aeronautical Society Conference on Unsteady Aerodynamics*, London, U.K., 1996.

[23] E. Kreiselmaier and B. Laschka, "Small Disturbance Euler Equations: Efficient and Accurate Tool for Unsteady Load Predictions," *Journal of Aircraft, Vol. 37, No. 5,* pp. 770-778, 2000.

[24] P. L. Roe, "Approximate Riemann Solvers, Parameter Vectors and Difference Schemes," *Journal of Computational Physics, Vol. 43, No. 2,* pp. 357-372, 1981.

[25] A. Jameson and E. Turkel, "Implicit Schemes and LU-Decompositions," *Mathematics of Computation, Vol. 37, No. 156,* pp. 385-397, 1981.

[26] A. Pechloff and B. Laschka, "Small Disturbance Navier–Stokes Method: Efficient Tool for Predicting Unsteady Air Loads," *Journal of Aircraft, Vol. 43, No. 1,* pp. 17-29, 2006.

[27] A. Pechloff and B. Laschka, "Small Disturbance Navier-Stokes Computations for Low Aspect-Ratio Wing Pitching Oscillations," *Journal of Aircraft, Vol. 47, No. 3,* pp. 737-753, 2010.

[28] C. Weishäupl and B. Laschka, "Small Disturbance Euler Simulations for Delta Wing Unsteady Flows due to Harmonic Oscillations," *Journal of Aircraft, Vol. 41, No. 4,* pp. 782-789, 2004.

[29] D. S. Broomhead and D. Lowe, "Multivariable Functional Interpolation and Adaptive Networks," *Complex Systems 2,* pp. 321-355, 1988.

[30] O. Nelles, Nonlinear System Identification - From Classical Approaches to Neural Networks and Fuzzy Models, Berlin Heidelberg: Springer, 2001.

[31] S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd edition, Upper Saddle River, NJ: Prentice-Hall, 1999.

[32] S. Chen, C. F. N. Cowan and P. M. Grant, "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks," *IEEE Transactions on Neural Networks, Vol. 2, No. 2,* pp. 302-309, March 1991.

[33] J. Moody and C. J. Darken, "Fast Learning in Networks of Locally-Tuned Processing Units," *Neural Computation 1,* pp. 281-294, 1989.

[34] L. Tang, R. E. Bartels, P.-C. Chen and D. D. Liu, "Numerical Investigation of Transonic Limit Cycle Oscillations of a Two-Dimensional Supercritical Wing," *Journal of Fluids and Structures, Vol. 17, No. 1,* pp. 29-41, 2003.

[35] The MathWorks, "MATLAB 2013b: User Manual," MathWorks, Natick, MA, 2013.

[36] S. Weber, K. D. Jones, J. A. Ekaterinaris and M. F. Platzer, "Transonic Flutter Computations for the NLR 7301 Supercritical Airfoil," *Aerospace Science and Technology, Vol. 5, No. 4,* pp. 293-304, 2001.

[37] ANSYS ICEM CFD 14.5, "User Manual," ANSYS, Canonsburg, PA, 2014.