ANN BASED ROM FOR THE PREDICTION OF UNSTEADY AEROELASTIC INSTABILITIES

Abdul Rampurawala[1], Ken Badcock[1], and Simao Marques[1]

[1]Department of Engineering,
University of Liverpool
K.J.Badcock@liverpool.ac.uk

Keywords. ANN, ROM, Flutter,

Abstract. A Reduced-Order Model (ROM) for the prediction of aeroelastic instabilities is presented. The unsteady nonlinear aerodynamic system is characterised by an Artificial Neural Network (ANN) to a set of network weights. The system is trained on a time history of simultaneous forced oscillation of the normal modes as input and generalised forces as output. Network weights are then used to approximate the aerodynamic force in the structural equation of motion to obtain the structural response. Results from the 3D Goland wing are presented and compared against full order CFD. It is shown that the ROM can predict aeroelastic instabilities with reasonable accuracy at a cost of less than one typical unsteady aeroelastic computation.

NOMENCLATURE

J	Jacobian
$\mathbf{W}$	network weights
$\mathbf{e}$	network error vector
l	number of hidden layers
$\prod$	network activation function
$\mathbf{f}$	generalised forces
$\boldsymbol{\omega}$	reduced natural structural frequencies
$\mathbf{\Phi}$	mode shapes
$\mathbf{U} = [\eta_1, \dot{\eta_1}, \eta_2, \dot{\eta_2}, ..\eta_m, \dot{\eta_m}]$	system input
$\eta, \dot{\eta}$	generalised coordinate,generalised velocity
m	number of structural modes
A	amplitude of forced oscillations

Superscripts

n	time index
o	output layer
h	hidden layer

Subscripts

i	input layer index
j	hidden layer index
k	output layer index

1 INTRODUCTION

High fidelity dynamically coupled CFD/CSM methods have been demonstrated for the prediction of transonic aeroelastic instabilities. However there is a cost to be paid in terms of computational effort due to the large number of degrees of freedom involved in coupled CFD/CSM dynamic simulations. Inspite of increases in computational resources available in recent times, wider application of high fidelity coupled simulations remains elusive in a production environment. There is clearly a need to reduce the overall size of the computational problem while retaining the physical complexities of the model. Over the last few years one of the approaches being widely researched is the use of aerodynamic Reduced Order Models (ROMs) for aeroelasticity. ROMs are simplified models of usually nonlinear unsteady aerodynamics that attempt to reproduce flow nonlinearities of the original model at a much reduced computational cost. The CFD can be replaced with a ROM in a coupled CFD/CSM analysis to obtain an aeroelastic response. A review of some of the ROM approaches can be found in references [1–3].

In this paper the inputs to the system identification dataset are the modal displacements and velocity and the outputs are the generalised forces. The dataset consists of modal input-output time histories which are used to train the ANN. The training dataset consists of multiple modal inputs and multiple force outputs at each time step. The generalised force output at any instance of time due to modal displacements is dependent on the displacements from other modes at that instance. The system relates the generalised forces to the modal displacements and velocity.

The simultaneous excitation of mutiple modal degrees of freedom allows the ROM to capture the nonlinear cross coupling of the degrees of freedom. As opposed to linearised ROMs the assumption of superposition of responses from perturbation of individual degrees of freedom in not made and hence the presented ROM is able to capture coupling nonlinearities of the dynamic system. Flutter predictions using the proposed ROM has yielded good comparison with time marching predictions. An order of magnitude reduction in computational effort has been achieved.

A Time Delay Neural Network (TDNN) based ROM approach was followed in [4] to form the network weights. The TDNN uses past values of the input-mapping for each instance of network training data.

$$f(i) = \Lambda(\eta_i, \eta_{i-1}..\eta_{i-T}, f_{i-1}, f_{i-2}, ..f_{i-T})$$

where, f_i is the output at the current time level, η is the system input and T is the maximum user defined time interval, within which the input and output data are included in each training data instance. By including data from previous time steps to predict output at the latest time level a degree of time dependency is built in the training. In the current approach a TDNN is not required as we make use of the assumption that the aerodynamic response at any instance in time is a function of both the modal displacements and the modal velocities from previous time steps.

2 METHOD

A Reduced Order Model based on Artificial Neural Network is presented for identifying the flutter point at a given Mach number. The proposed ROM consists of 3 main parts. The first is the unsteady CFD computation for calculating the time history of the aerodynamic reponses f due to sysem inputs. The second is the training of the neural network on the time history and final step is the calculation of the structural response at varying dynamic pressures. The CFD computation is the most expensive step of the ROM whereas the calculation of the structural response takes a few seconds. An order of magnitude saving in computational time compared to time marching calculations can be achieved.

Unsteady CFD computations at a chosen Mach number are performed with all the structural modes oscillating simultaneously with small amplitude. The length of the unsteady calculations typically ranges from 5 to 10 cycles of the lowest frequency mode. The number of cycles is case specific but for the cases studied in this paper 10 cycles were found adequate to train the Artificial Neural Network (ANN). The usual CFD time marching best practices are to be followed with respect to time step, flow convergence etc. The generalised coordinate η, the generalised velocity $\dot{\eta}$ and the generalised force f for each mode at each time step are recorded. The generalised force is the coupling link between the CFD and he structure. A feed forward error backpropagation/Levenberg-Marquardt scheme is used for training. The output from the ANN training are network weights which contain the system characteristics and can be used to reproduce the aerodynamic response required in the solution of the structural equation. The generalised force is dimensionalised with the dynamic pressure at which the aeroelastic response is required and is obtained from the network weights from the following equation,

$$f_k(t) = \sum_{j=1}^{l} H_j W^o_{j,k} \tag{1}$$

where,

$$H_j = \prod \sum_{i=1}^{2m} U(t)_i W^h_{i,j} \tag{2}$$

In equations 1 and 2 the network weights $\mathbf{W}^o$ and $\mathbf{W}^h$ contain the system characteristics that reproduces an output response for any combination of system inputs η and $\dot{\eta}$. $\prod$ in equation 2 is the ANN output activation function used in the network training. In the following subsections the 3 parts that make up the proposed ROM are discussed.

2.1 Unsteady CFD

All computations were performed using the Parallel Multi-Block (PMB) flow solver [5] of University of Liverpool, which has been continually revised and updated over a number of years. The solver has been successfully applied to a variety of problems including cavity flows, hypersonic film cooling, spiked bodies, flutter and delta wing flows amongst others. The fluid and structural equations are solved on separate grids. The aerodynamic force is calculated over the fluid surface grid and is interpolated to the structural grid using the Constant Volume Tetrahedron (CVT) transformation scheme. [6, 7] To prepare the training data for the ANN full time marching CFD calculation are performed, and the input-output history recorded. Unsteady CFD calculations are started from steady state solutions. The Goland wing considered here has symmetric cross sectional profile so static deformations, if any, are assumed to to have no effect on the dynamic instability. All the modes are simultaneously oscillated with their natural frequencies of vibration. The amplitude of oscillations of each mode is scaled with the factor ω_1^2/ω_i^2 where ω_1 is the lowest frequency mode. The reason for this is that in a full time marching calculation the amplitude of each mode usually decreases with increasing frequencies. The amplitude of each mode has an effect on the generalised force output of every other mode, hence to train the system for a realistic output the input should be as close to the values observed in time marching simulations. The lowest frequency mode has the largest amplitude A. The forced modal inputs are calculated from the following equations,

$$\eta_i^n = Asin(\omega_i n \Delta t)\frac{\omega_1^2}{\omega_i^2} \tag{3}$$
$$\dot{\eta}_i^n = \frac{\eta_i^n - \eta_i^{n-1}}{\Delta t}$$

At every time step the generalised forces f_i^n the generalised coordinate η_i^n and the generalised velocity $\dot{\eta}_i^n$ are recorded. The format of a typical training data set with m structural modes is as follows,

$$\begin{array}{cccccccccc}
\eta_1^1 & \dot{\eta}_1^1 & \eta_2^1 & \dot{\eta}_2^1 \; \ldots & \eta_m^1 & \dot{\eta}_m^1 & f_1^1 & f_2^1 & \ldots & f_m^1 \\
\eta_1^2 & \dot{\eta}_1^2 & \eta_2^2 & \dot{\eta}_2^2 \; \ldots & \eta_m^2 & \dot{\eta}_m^2 & f_1^2 & f_2^2 & \ldots & f_m^2 \\
 & & & & & & & & & . \\
 & & & & & & & & & . \\
 & & & & & & & & & . \\
\eta_1^n & \dot{\eta}_1^n & \eta_2^n & \dot{\eta}_2^n \; \ldots & \eta_m^n & \dot{\eta}_m^n & f_1^n & f_2^n & \ldots & f_m^n
\end{array}$$

2.2 ANN

A neural network is a collection of interconnected neurons or nodes with each node receiving a number of inputs either from the original input data, or from the output of other neurons in the network. The transfer of input to the node is weighted. Each node also has a single threshold value. The weighted sum of the inputs is formed, and the threshold subtracted, to compose the activation of the neuron. The activation signal is passed through an activation function to produce the output of the neuron (Fig. 1) shows a typical neural network with a layer of inputs, a hidden layer and an output layer. The input weights are summed and passed through an activation function, usually a nonlinear function like the hyperbolic tangent or exponential sigmoid. The output from the hidden layer forms the input to the output layer. The weighted sum of the outputs from the hidden layer is once again passed through a linear or nonlinear function to obtain the output. If the activation function in the hidden layers is nonlinear then the network has the capability to map the linear combination of system inputs to a nonlinear output. The most commonly used activation function is the sigmoid. The layers of nodes between the input and output are referred to as hidden layers. A network can have as many hidden layers as required. For smooth functions a few number of hidden units are needed, for wildly fluctuating functions more hidden units may be required [8]. A standard feedforward backpropogation method has been implemented [8,9]. The network weights update switches between error backpropagation and a second order (LMA) Levenberg-Marquardt algorithm. The switching is done after a prescribed number of backpropagation epochs (M) have been completed or the network error has reduced to a level where LMA is stable (E) (Fig. 2).

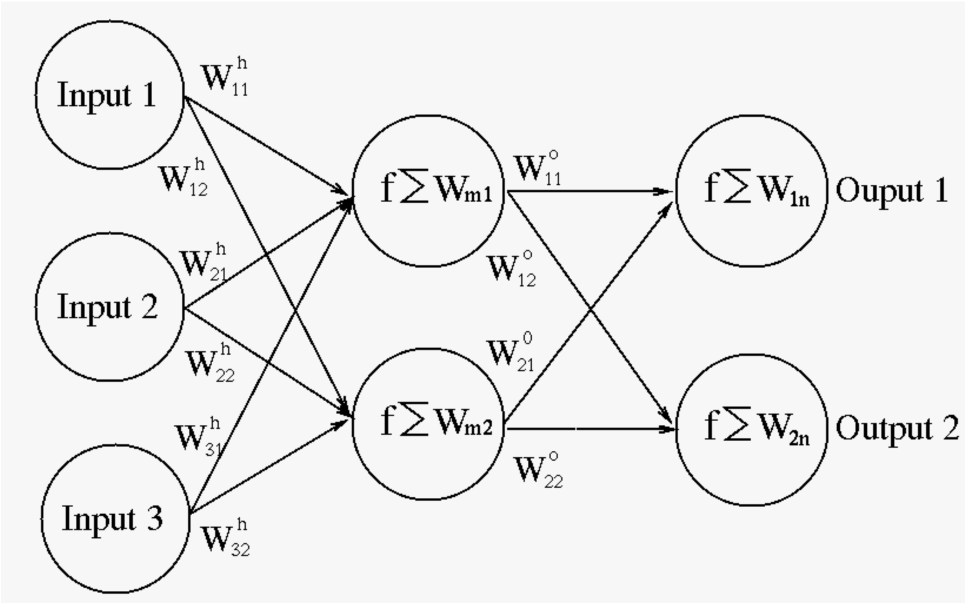


Figure 1: A typical ANN topology

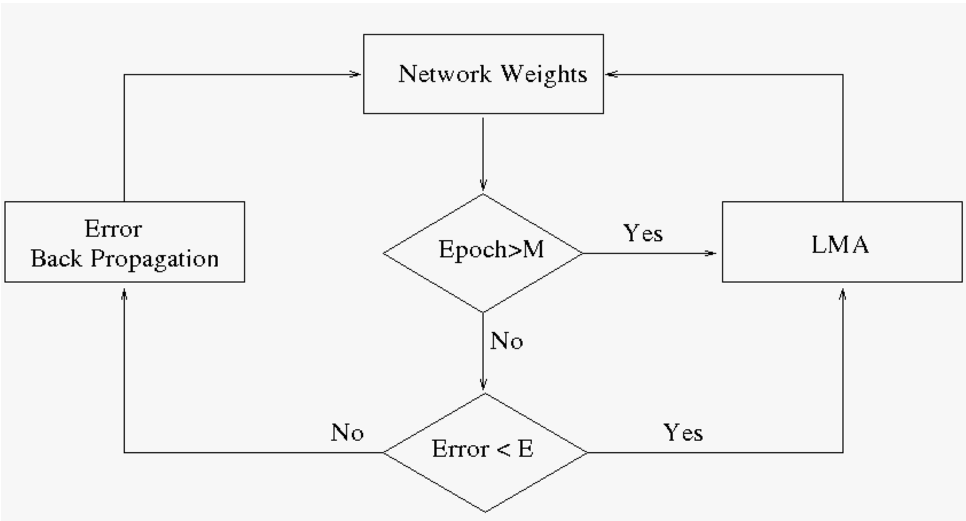


Figure 2: Switching from backpropagation to LMA error minimisation

2.2.1 Error Backpropagation

The standard feedforward error backpropagation is one of the popular network weight update schemes. The central idea is that the error of hidden layer units is determined by the backpropagation of error from the output layer. Reproducing Equations 1 and 2 the output from the hidden layer is given by,

$$H_j = \prod \sum_{i=1}^{2m} U_i W^h_{i,j} \tag{4}$$

$\prod$ is the activation function of the hidden layer. In the current work this function is a sigmoid.

$$\prod(x) = \frac{1}{1+e^{-x}}$$

The output from the output layer is obtained from,

$$f_k = \sum_{j=1}^{l} H_j W^o_{j,k} \tag{5}$$

The mean square error of the predicted and target output is calculated,

$$e = \sum_{k=1}^{m} (f_k^{target} - f_k)^2 \tag{6}$$

The error gradient with respect to change in network weights is calculated. The network weights are updated with the error gradient contribution and a momentum term. The change in weight should be proportional to the error gradient and the proportionality constant is known as the learning rate (β). To avoid oscillations the weight update is made dependant on the previous value by a factor α. Suitable values of α and β are case dependant.

$$\Delta W = \beta(\frac{\partial e}{\partial W}) + \alpha \Delta W \tag{7}$$

$$W = W + \Delta W \tag{8}$$

2.2.2 Levenberg-Marquardt Algorithm (LMA)

The LMA is an efficient method for minimising the sum square error of nonlinear functions. In an ANN the error e is to be minimised given the network weights as function parameters. The first derivative of the network errors is used to update the network weights. The performance index to be minimised is,

$$F(\mathbf{W}) = \mathbf{e^T e} \tag{9}$$

where, $\mathbf{W} = [W^h_{1,1}, W^h_{2,1}, ...W^h_{2m,l}, W^o_{1,1}, W^o_{2,1}, ...W^o_{l,m}]$ is a vector of all the weights of the system and $\mathbf{e}$ is the error vector comprising of all the errors in a training example. The LMA is written as,

$$\mathbf{W_{i+1}} = \mathbf{W_i} - [J^T(\mathbf{W_i})J(\mathbf{W_i}) + \mu I]^{-1} J^T(\mathbf{W_i})\mathbf{e}(\mathbf{W_i}) \tag{10}$$

where μ is a weighting factor < 1 that decreases as e approaches minimum. Implementation of LMA in form of C libraries from [10] has been used in the current work.

2.3 Structural Response

Finite element methods allow for the static and dynamic response of a structure to be determined. Stiffness ($\mathbf{K}$) and mass ($\mathbf{M}$) matrices are used to determine the equation of motion of an elastic structure subjected to an external force $\mathbf{f}$ as

$$\mathbf{M}\delta\ddot{\mathbf{x}}_s + \mathbf{K}\delta\mathbf{x}_s = \mathbf{f} \tag{11}$$

where $\delta\mathbf{x}_s$ is a vector of displacements on a grid of points $\mathbf{x}_s$. Because the structural system under consideration is assumed to be linear, its characteristics are determined once and for all prior to making the flutter calculations, so that $\mathbf{M}$ and $\mathbf{K}$ are constant matrices generated, in this case, by the commercial package `NASTRAN`.

The structural deflections $\delta\mathbf{x}_s$ are defined at a set of grid points $\mathbf{x}_s$ by

$$\delta\mathbf{x}_s = \sum \eta_i \phi_i \tag{12}$$

where ϕ_i are the mode shapes and η_i the generalised displacements. Here the η_i depend on time but the mode shapes do not. The values of ϕ_i and ω_i are calculated by solving the eigenvalue problem

$$[\mathbf{M} - \omega_i^2 \mathbf{K}]\phi_i = \mathbf{0}. \tag{13}$$

The eigenvectors are scaled so that

$$[\phi_i]^T \mathbf{M}[\phi_i] = 1. \tag{14}$$

Projecting the finite element equations onto the mode shapes results in the equations

$$\frac{d\dot{\eta}_i}{dt} + \omega_i^2 \eta_i = \phi_i^T \mathbf{f} \tag{15}$$

This equation is solved by a two stage Runge-Kutta method, with the latest estimate of the $\mathbf{f}$ obtained from equation 1 given the inputs $\mathbf{U}^n$ at the latest time level. The structural equations are decoupled from the ROM and the structural timestep is independent of the timestep used in the ANN training.

3 RESULTS

Aeroelastic instabilities on the Goland wing were predicted using the proposed ROM. Results for all cases are compared with full time marching simulations.

3.1 Goland Wing

The Goland wing has a chord of 6 feet and a span of 20 feet. It is a rectangular cantilever with a 4% thick parabolic cross section. The structural model follows the description given in [11]. The CFD grid is block structured and uses an O-O topology. This allows the grid points to be clustered on the tip region which is critical for accurate aerodynamic contribution to the aerodynamic response. The fine grid has 236 thousand points and the extracted coarse grid has 35 thousand points. The first four structural modes were retained for the aeroelastic computations (Table 1).

	Mode 1	Mode 2	Mode 3	Mode 4
Frequency (Hz)	1.981449	4.048638	9.687665	13.49333

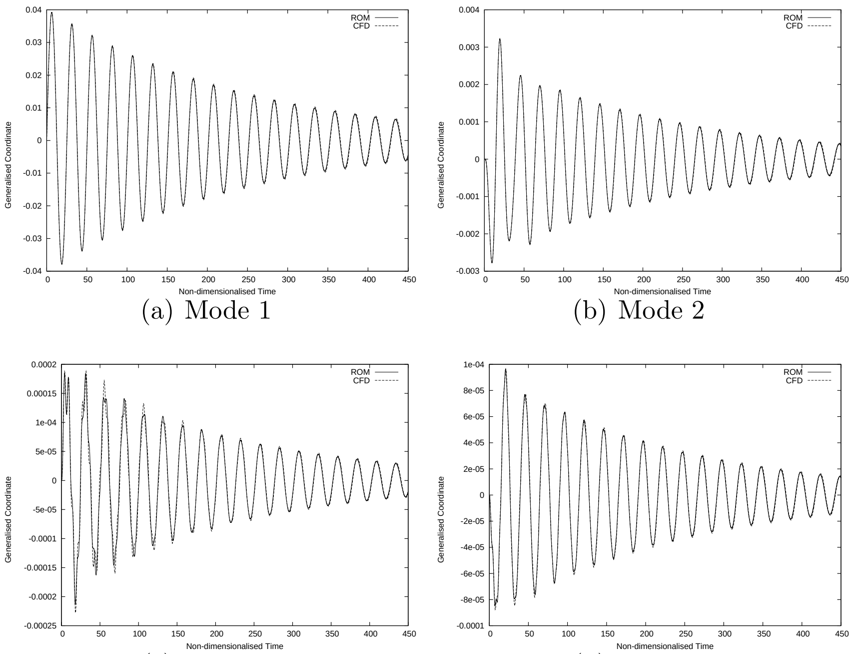
Table 1: Natural frequencies of Goland clean wing

	Mode 1	Mode 2	Mode 3	Mode 4
Frequency (Hz)	1.72	3.05	9.18	11.10

Table 2: Natural frequencies of Goland wing with store

3.1.1 Reproducing a Free Response

The ROM is trained on an output from a typical time marching aeroelastic computation. The aeroelastic response is then reproduced by the ROM and compared with the original computation. This is a basic check on the ROM to verify that it can exactly reproduce the response it was trained on. Two unsteady time marching aeroelastic computations are performed at Mach 0.85, velocity 323.5 ft/s and densities 2.3771×10^{-3} and 4×10^{-3} slugs/ft^3. A timestep of 0.5 was used. The $\eta, \dot{\eta}$ and f at each timestep are recorded for the first 400 steps and used for training. The ANN is trained till the global error dropped to 1×10^{-4}. The initial perturbation of 0.01 to the generalised velocity is applied for all ROM and CFD simulations. Figures 3 and 4 show the comparison of full CFD and ROM responses.



(a) Mode 1 (b) Mode 2

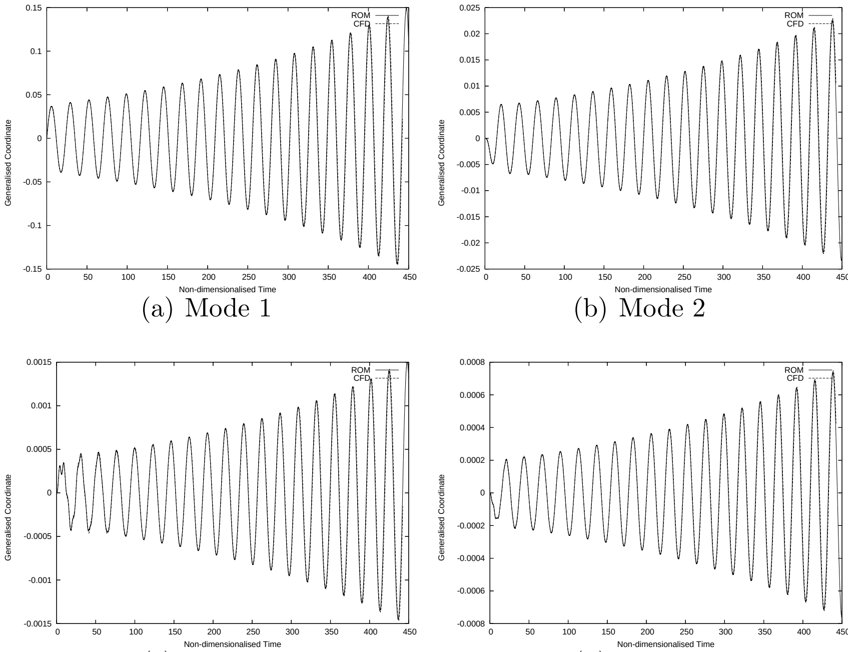
(c) Mode 3 (d) Mode 4

Figure 3: Aeroelastic response at Mach 0.85 and density 2.3771×10^{-3} slugs/ft^3

3.1.2 Flutter Prediction

Having confirmed the ability of the ROM trained on a particular free response to reproduce exactly the same response, the next step is to generate a general ROM capable of predicting the flutter point at a given Mach number. The ROM process to identify the dynamic pressure at which flutter occurs is as follows,

- Starting from a steady solution begin unsteady CFD computation with simultaneous forced oscillation of all the modes with their natural frequency (Section 2.1). The simulation is allowed to run for about 10 cycles of the lowest frequency mode. Figure



(a) Mode 1 (b) Mode 2

(c) Mode 3 (d) Mode 4

Figure 4: Aeroelastic response at Mach 0.85 and density 4×10^{-3} slugs/ft^3

5 shows the oscillations of a typical forced unsteady calculation.

- The training data for the ANN is the time history of $\eta, \dot{\eta} andf$ from the forced unsteady simulations. The ANN is trained until the global error drops to an acceptable level, usually 1×10^{-4}.
- Successful completion of ANN training outputs a set of network weights contain the system characteristics and will be able to generate a meaningful aerodynamic response to system inputs. The modal equation 15 is solved using the a 2 step Runge-Kutta method with the generalised force calculated from equation 1 and dimensionalised with the dynamic pressure at which the response is sought.
- Aeroelastic responses at a range of dynamic pressures is calculated to identify the flutter point.

The above ROM process was used to trace the flutter boundary of the clean goland wing. The ROM aeroelastic response at Mach 0.7, velocity 323.5 ft/s and densities 4.6×10^{-3} and 4.8×10^{-3} slugs/ft^3 is compared with full CFD results in Figures 6 and 7. Figure 8 shows the flutter boundary traced with full CFD and ROM. It can be seen that the ROM is able to predict a transonic dip although the match with the CFD is not exact. Further refinement to the ROM method is needed, specifically in the generation of the unsteady CFD training data. In a CFD based aeroelastic simulation the frequencies of the modes change due to aerodynamic damping. This has an effect on the unsteady force distribution and this not currently captured as the modes are oscillated with their natural frequencies. A possible enhancement would be to incorporate a bidirectional chirp function to modify the natural frequencies in the forced oscillations.

3.1.3 LCO Prediction

The ROM methodology is tested for its ability to predict Limit Cycle Oscillation (LCO) type of instability. The heavy version of the Goland wing was chosen for this. It has a wing tip store installed and undergoes a LCO at Mach 0.92. This was shown using Euler

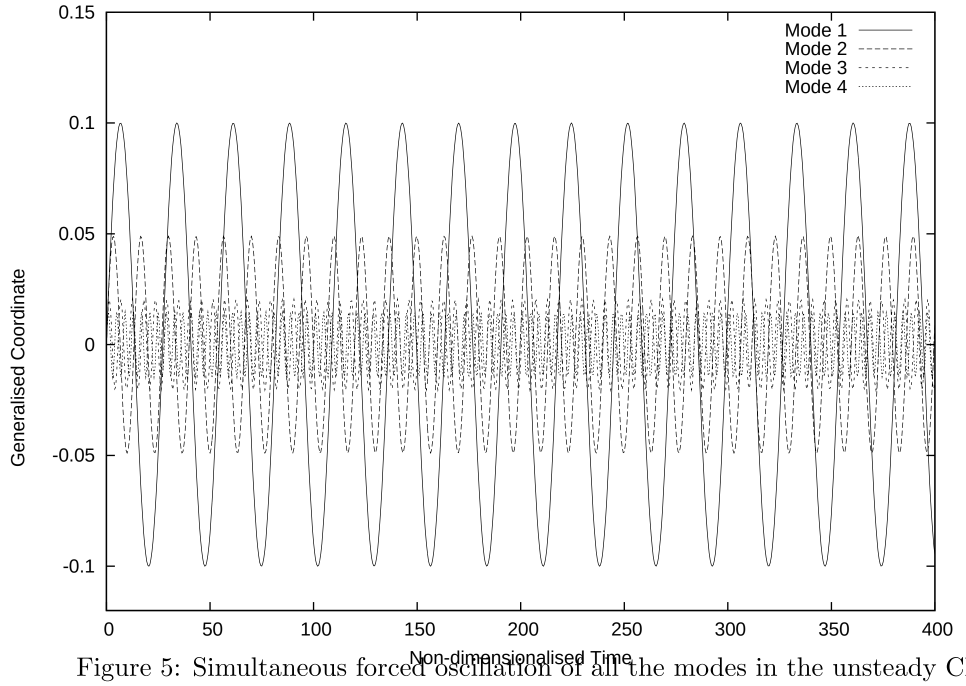
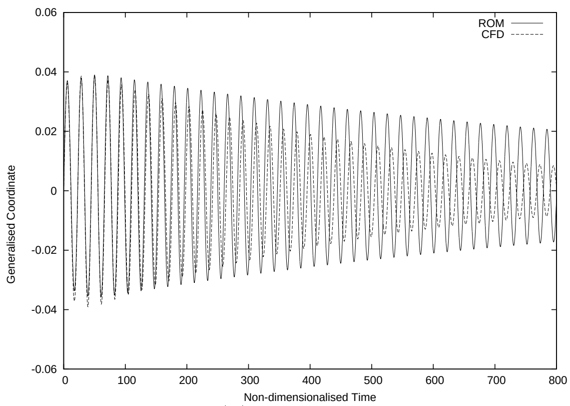
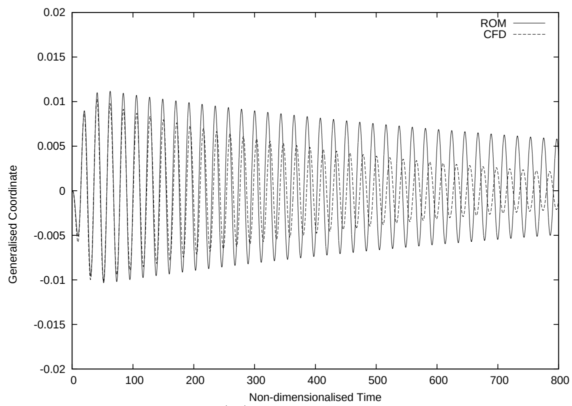


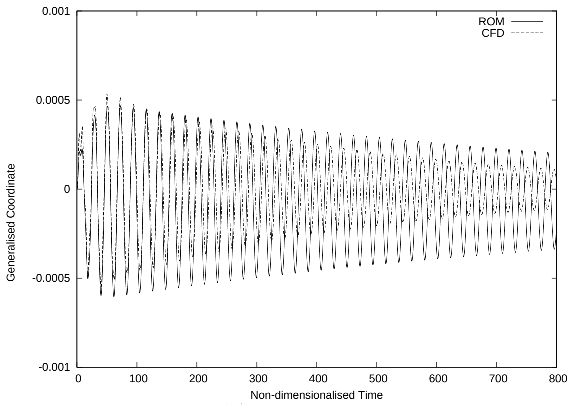
Figure 5: Simultaneous forced oscillation of all the modes in the unsteady CFD simulation



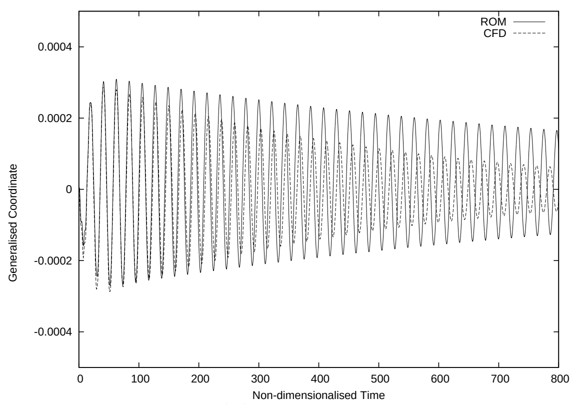
(a) Mode 1



(b) Mode 2



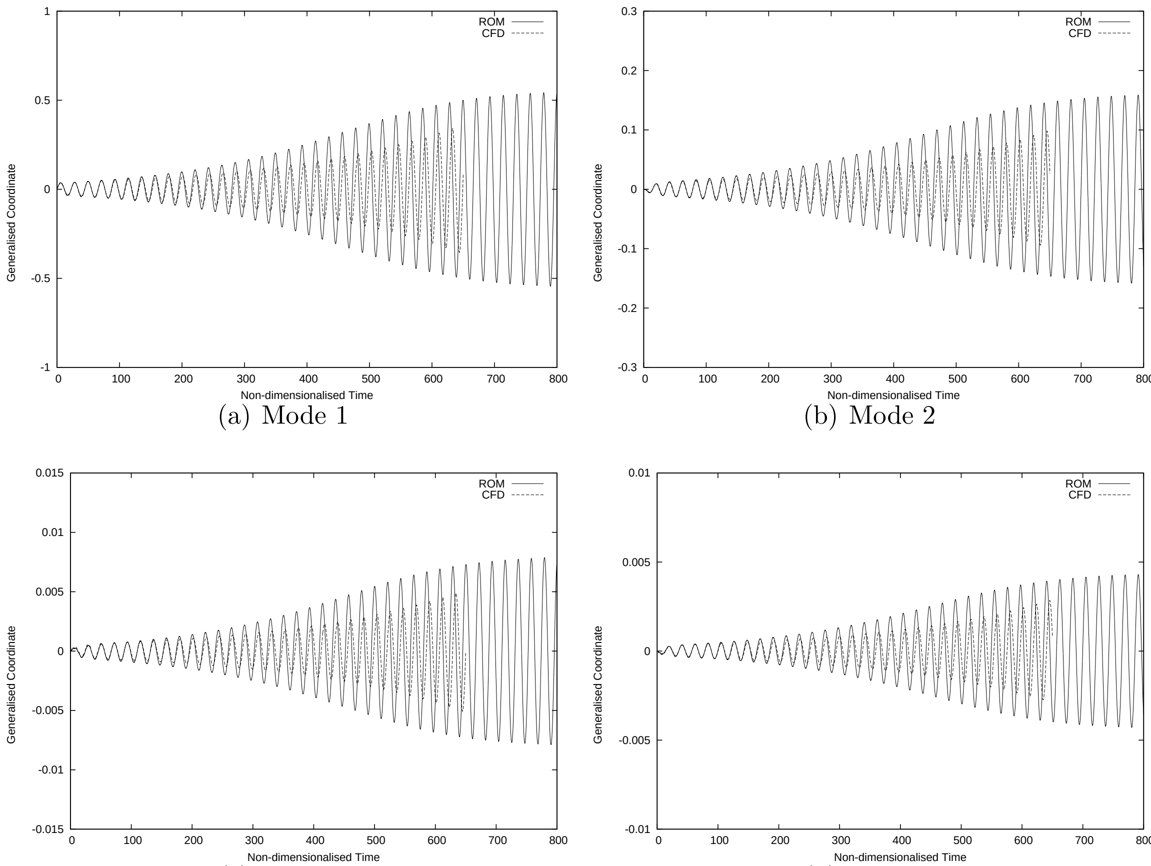
(c) Mode 3



(d) Mode 4

Figure 6: Aeroelastic response at Mach 0.7 and density 4.6×10^{-3} slugs/ft^3

equations in [12] and transonic small disturbance in [11]. Recently LCO prediction on this test case was shown using a reduce order model based on the Hopf bifurcation theory in [13]. The steps as described in section are performed. The first 4 natural modes of



(a) Mode 1 (b) Mode 2

(c) Mode 3 (d) Mode 4

Figure 7: Aeroelastic response at Mach 0.7 and density 4.8×10^{-3} slugs/ft^3

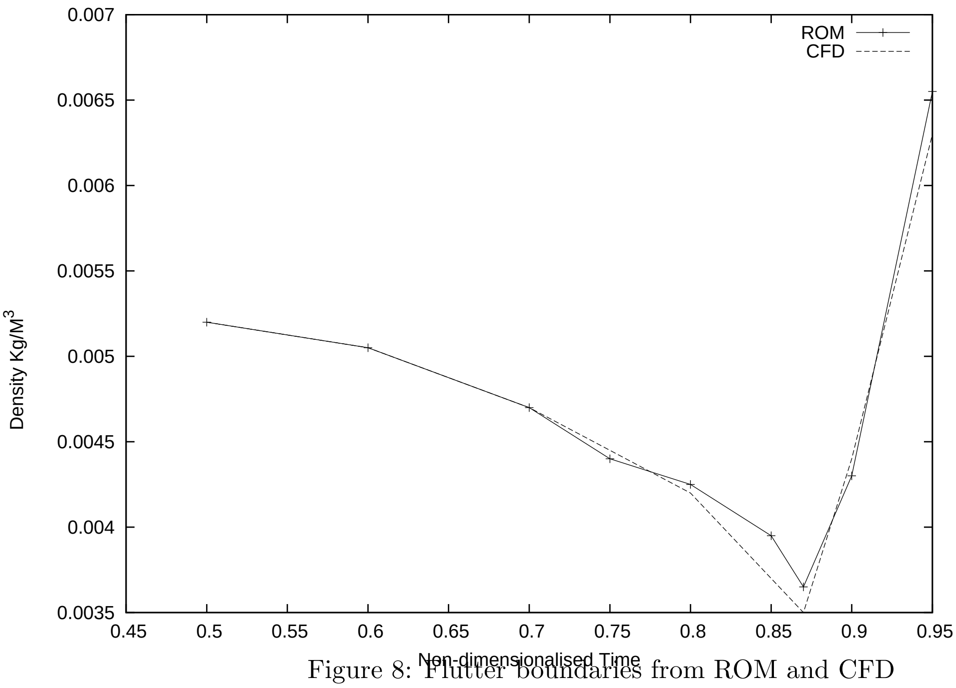
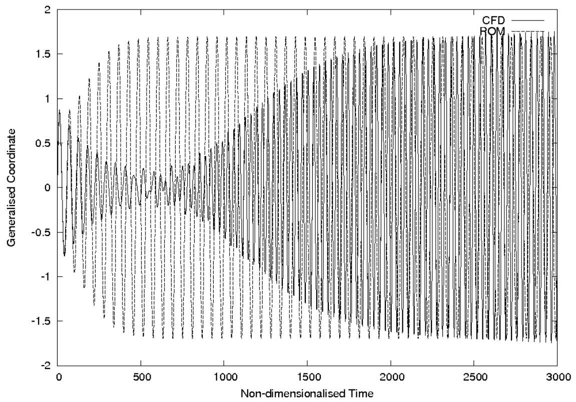
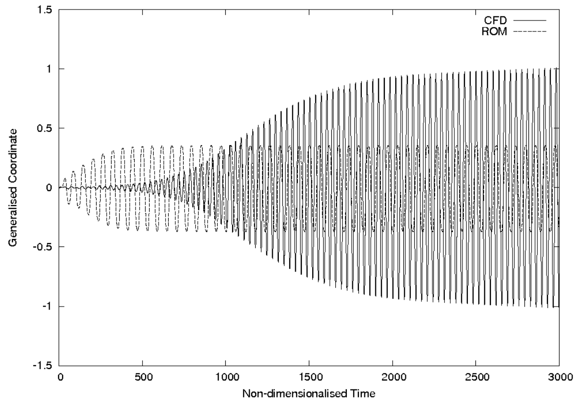


Figure 8: Flutter boundaries from ROM and CFD

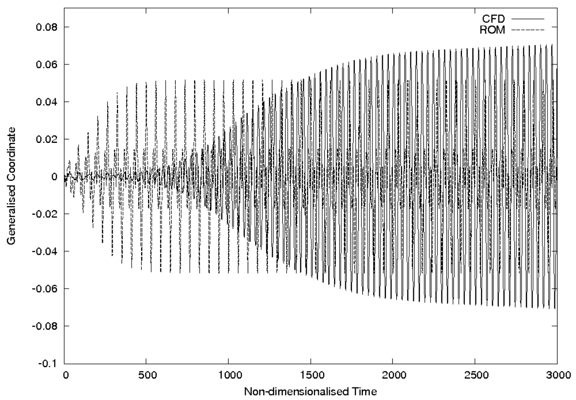
vibration are included in the computation (Table). Unsteady time marching calculations are performed at sea level density and a range of velocities to measure the growth of the LCO amplitude. The ROM is constructed by training the ANN with oscillation of the modes at natural frequencies. Figure show the comparison between full CFD and ROM for the first and second modes at velocity of 600 ft/sec and density of 0.00273 slugs/ft^3



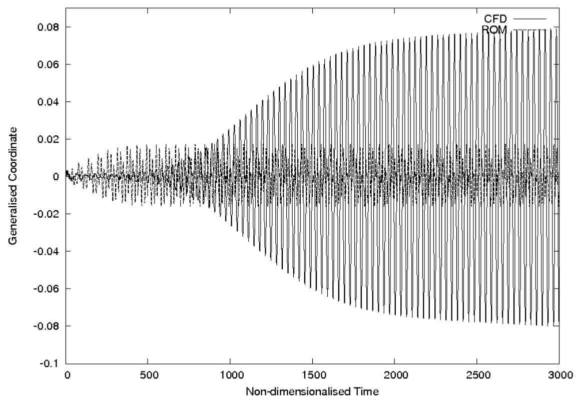
(a) Mode 1



(b) Mode 2



(c) Mode 3



(d) Mode 4

Figure 9: LCO prediction at Mach 0.92 and density 2.73×10^{-3} slugs/ft^3

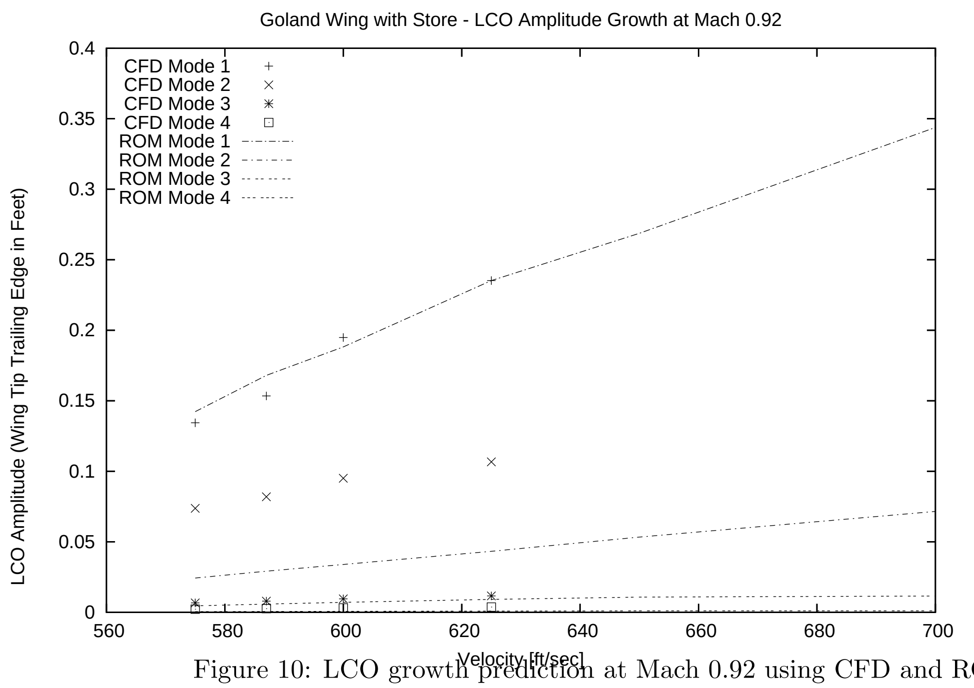


Figure 10: LCO growth prediction at Mach 0.92 using CFD and ROM

4 LIMITATIONS

The accuracy of the ROM prediction is dependant on the quality of the training data. Provided the training data represents the aerodynamics of the coupled full time marching calculations, the ROM results will follow the full CFD closely as shown in Figures 6 and 7. In a coupled system the aerodynamics is dependant on the structural contributions namely the mode shapes, the frequencies and the amplitudes. The frequencies and the modal amplitudes of the individual modes in turn depend on the aerodynamic forces and will vary with dynamic pressure. Modal frequencies and amplitudes will increase for some modes and decrease in others. Currently the training data is obtained with the forced oscillations of the modes at fixed natural frequencies and nominal amplitudes hence is not an exact representation of the aerodynamics of a full system. Ideally the training data should contain the same modal amplitudes and frequencies of the full system, however this is not possible without performing the actual time marching calculations or losing the generality of the ROM. A solution to this is to obtain the training data at a range of frequencies and amplitudes. This will offcourse add extra cost in the training effort. NEED TO ADD OBTAIN RESULTS FOR THIS BEFORE 17th

5 CONCLUSIONS

A ANN based ROM approach for prediction of nonlinear aeroelastic instabilities has been presented. Simultaneous excitation of the modal degrees of freedom in the training data accounts for the nonlinear cross coupling of degrees of freedom within the system. The proposed ROM is able to predict the aeroelastic instabilities with a reasonable level of accuracy compared to full CFD predictions. Considerable savings in computational effort is achieved as the unsteady simulation for a few cycles of the structureal modes, is performed only once for prediction of a flutter point. A further enhancement of the method is required to account for the change in structural frequencies in the flow, possibly by modifying the input frequencies in the training data with a bidirectional chirp.

6 REFERENCES

[1] Lucia, D. J., Beran, P. S., and Silva, W. A. (2004). Reduced-order modeling: New approaches for computational physics. *Progress in Aerospace Sciences*, 40, 51–117.

[2] Raveh, D. E. (2003). Identification of cfd-based unsteady aerodyncamic models for aeroelastic analysis. AIAA Paper 2003-1407, AIAA. Presented at the 44th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Norfolk, VA.

[3] Marzocca, P., Silva, W. A., and Librescu, L. (2002). Open/closed-loop nonlinear aeroelasticity for airfoils via volterra series approach. AIAA Paper 2002-1484, AIAA. Presented at the 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Denver, CO.

[4] Marques, F. D. (2002). Neural network based unsteady transonic loads reduced-order model. AIAA Paper 2002-1599, AIAA. Presented at the 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Denver, CO.

[5] Badcock, K. J., Woodgate, M. A., and Richards, B. E. (2000). Elements of computational fluid dynamics on block structured grids using implicit solvers. *Progress in Aerospace Sciences*, 36, 351–392.

[6] Goura, G. L. (2001). *Time marching analysis of flutter using Computational Fluid Dynamics*. Ph.D. thesis, University of Glasgow.

[7] Badcock, K. J., Rampurawala, A. M., and Richards, B. E. (2003). Intergrid Transformation for Aircraft Aeroelastic Simulations. *AIAA Journal*, Vol.42(9), pp. 1936–1939.

[8] Krose, B. and Smagt, P. (1996). *An Introduction to Neural Networks*. The University of Amsterdam.

[9] Haykin, S. (1994). *Neural Networks, A Comprehensive Foundation*. MacMillan, New York.

[10] Wuttke, J. (2004-8). *lmfit: a C/C++ routine for Levenberg-Marquardt minimization with wrapper for least-squares curve fitting*. based on work by B. S. Garbow, K. E. Hillstrom, J. J. More, and S. Moshier. http://www.messen-und-deuten.de/lmfit/.

[11] Beran, P. S., Knot, N. S., Eastep, F. E., et al. (2004). Numerical analysis of store induced limit cycle oscillation. *Journal of Aircraft*, Vol.41(6), pp. 1315–1326.

[12] Snyder, R. D., Scott, J. N., Knot, N. S., et al. (2003). Predictions of store-induced limit cycle oscillations using euler and navier-stikes fluid dynamics. AIAA Paper 2003-1727, AIAA.

[13] Woodgate, M. A. and Badcock, K. J. (2007). Fast prediction of transonic aeroelastic predictions and limit cycles. *AIAA Journal*, Vol.45(6), pp. 1370–1381.