Comparing the performance of machine learning and data-driven approaches to reduced-order modelling for non-linear dynamics

Tom Marsland

Abstract

This research aims to address the issue of computationally expensive finite element (FE) dynamic analysis associated with geometrically non-linear systems. The two approaches explored in this research were reduced-order modelling (ROM) and machine learning (ML). First, a simple 2DoF system was considered to highlight and validate the ROM process. Then the ROM and ML methods were applied to a clamped-clamped beam system to determine their effectiveness for more realistic and complex systems. ROM achieved a 93.61% reduction in computational cost whilst retaining decent accuracy. Further optimisation of this ROM improved accuracy by an additional 94.67% with only a minor increase in computational cost. The ML models delivered a 97.16% increase in accuracy compared to the unoptimised ROM and a 94.68% reduction in computational time compared to the full-order model, when a pre-existing training dataset was available and the correct hyperparameters were used. When this pre-existing training data set was not available, the accuracy remained the same, but the computational time increased by a factor of 24 compared to the full-order model. This study highlighted the effectiveness of each method by comparing the computational cost and accuracy, depending on the availability of training data. ROM was found to be less accurate and more expensive than an ML model if pre-existing training data is available. However, if this data is not available, the trade-off for this increase in accuracy comes at a substantial computational cost.

Supervised by Dr T Hill
School of Electrical, Electronic and Mechanical Engineering
University of Bristol
2025

Declaration

This project report is submitted towards an application for a degree at the University of Bristol. The report is based upon independent work by the candidate. All contributions from others have been acknowledged and the supervisor is identified on the front page. The views expressed within the report are those of the author and not of the University of Bristol.

I hereby assert my right to be identified as the author of this report. I give permission to the University of Bristol Library to add this report to its stock and to make it available for consultation in the library, and for inter-library lending for use in another library. It may be copied in full or in part for any bona fide library or research worker on the understanding that users are made aware of their obligations under copyright legislation.

I hereby declare that the above statements are true.

tmarsland

Certification of ownership of the copyright in a dissertation presented as part of and in accordance with the requirements for the relevant degree at the University of Bristol.

This report is the property of the University of Bristol Library and may only be used with due regard to the author. Bibliographical references may be noted but no part may be copied for use or quotation in any published work without prior permission of the author. In addition, due acknowledgement for any use must be made.

Contents

1 Introduction

Traditionally, dynamists use linear normal modes (LNMs) to study and evaluate dynamic systems. LNMs are helpful because they have clear, physical interpretations and provide valuable mathematical properties that aid in analysing linear systems. One of the most useful properties they hold is the ability to decouple the equations of motion of a system, resulting in efficient and simplified analysis for linearly assumed systems.

The primary assumption underlying linear analysis is that the system's response is directly proportional to its input. From this linear assumption, powerful analysis properties arise, such as invariance, modal superposition and modal reduction. These properties demonstrate how LNMs and linear assumptions can significantly ease the burden of mathematics for system analysis.

Once a system begins to exhibit non-negligible amounts of non-linearity, this assumption becomes no longer accurate. In these cases, the strong mathematical properties which arise from linear assumptions begin to break down and become inapplicable. This results in the need for new techniques to analyse these systems.

In recent years, there has been a growing demand within the engineering industry for enhanced system performance. As a result, this increasing demand leads to new, modern, and developing designs that focus on solutions and structures that are thinner, lighter, and more flexible. Consequently, this results in designs which exhibit significant amounts of geometric non-linear behaviours. Some examples of these systems that experience this geometric non-linearity include modern space launch vehicles, such as the SpaceX Falcon 9 [1], aeroelastic models of morphing wings [2], long flexible wind turbine blades [3,4], micro-electromechanical systems [5], full-scale aircrafts [6], musical instruments [7] and suspension bridges [8]

In certain structures, these individual LNMs begin to interact with one another, causing energy transfer between them, which results in a complex and unpredictable dynamic response. Applying linear analysis techniques in such cases often leads to inaccurate analysis as they fail to capture these non-linear behaviours. This can result in inefficient, ineffective and potentially unsafe designs [1,2,3,4,5,6,7,8].

Historically, finite element analysis (FEA) methods have been used to numerically analyse dynamic simulations for non-linear systems due to their ability to accurately represent complex geometries and behaviours.

In linear systems, the equations of motion can be decoupled using a linear modal transformation, which significantly eases the computational burden associated with analysis. However, non-linear systems lack this decoupling, as all degrees of freedom can interact with one another in a non-linear manner [9,10]. This results in significant computational costs, especially when modelling real-world geometrically complex systems. These real-world systems can involve millions of degrees of freedom, leading to substantial computational costs due to the coupling of the modes. Despite this, capturing the non-linear dynamic behaviours is crucial for making accurate predictions, conducting thorough analyses, and developing robust designs.

To reduce the large computational costs associated with complex non-linear dynamic FE analysis, reduced-order modelling (ROM) techniques can be employed. ROM can significantly ease the computational burden associated with these analyses. ROM works by capturing the system's dominant dynamics using a limited number of LNMs [9,10]. This significantly simplifies the analysis while preserving the primary characteristics of the system. As a result, ROM enables computationally cheap analyses, making them valuable for examining large and complex non-linear systems where full-scale dynamic FEA would be prohibitively expensive.

Machine learning (ML) has been a field of study for several decades, with its origin dating back to the 1950s, when Arthur Samuel first introduced the concept of machine learning from data. In recent years, machine learning (ML) has experienced rapid growth, evolving into a powerful and adaptable tool across a wide range of fields due to its ability to learn patterns and produce accurate predictions. U.S. private investment in this industry alone grew to $109.1 billion in 2024 [11]. For non-linear dynamic systems, ML provides a data-driven alternative to traditional simulation-based methods such as FEA and ROM. ML plans to tackle the problem of computationally expensive simulations by learning and generalising system behaviours from precomputed data, allowing for fast, precise predictions.

This research explores the application of ML and data-driven methods as an alternative or complement to ROM for simulating non-linear dynamic systems and reducing the computational costs associated with traditional FEA analysis. This research aims to assess their respective strengths and potential applications for future engineering systems by comparing the performance, accuracy and computational cost of both approaches.

2 Literature Review

2.1 Historical Non-linear analysis techniques

To create effective, analogous tools for systems that experience significant amounts of non-linearity, the first method dynamicists attempted to develop were Non-Linear Normal Modes (NNMs). First introduced by Rosenberg [12], NNMs were used to describe the pattern of motion between points in a non-linear system, similar to LNMs. The key difference is that the mode shape can deform as the amplitude of oscillation changes. In contrast to linear systems, where mode shapes and frequencies are fixed for a given undamped response, NNMs allow the frequency and mode shape to vary with amplitude [12,13,14]. However, all displacement coordinates still maintain a fixed phase relationship, meaning they move synchronously in a structured manner, either in-phase or anti-phase with a reference coordinate [12,13,14]. This explanation is commonly referred to as 'A Vibration in Unison. '[12].

2.2 Problems with Traditional Linear Analysis

For an undamped and forced response, the typical equation of motion (EoM) for a geometrically non-linear system takes the form of a second-order differential equation [6,9,10,13,15]

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{K}\mathbf{x} + \mathbf{f}(\mathbf{x}) = \mathbf{F}(t), \qquad [1]$$

where $\mathbf{f}(\mathbf{x})$ refers to the $N \times 1$ vector of non-linear restoring forces, $\mathbf{M}$ and $\mathbf{K}$ refer to the mass and stiffness matrices with shape $N \times N,$ and $\boldsymbol{F}(t)$ refers to the $N \times 1$ vector of the applied forces [9,10,13,15]. N refers to the number of degrees of freedom of the full model, and $\mathbf{x}$ represents a generalised coordinate system that describes the motion of the system, [9,10,13,15,16,17] where $\mathbf{x}_i$ represents each coordinate

$$\mathbf{x} = [x_1, x_2, x_3 .... x_N]^T. \quad [2]$$

It is typically assumed that these non-linear restoring forces are represented by polynomial equations of quadratic and cubic order in these generalised coordinate systems [9,10,13,15,16,17]. They are typically represented in a form such as

$$\mathbf{f}(\mathbf{x}) = \mathbf{K}^{(2)}\mathbf{x}\mathbf{x} + \mathbf{K}^{(3)}\mathbf{x}\mathbf{x}\mathbf{x}, \quad [3]$$

where the tensors, $\mathbf{K}$, represent the coefficients of each term, essentially characterising the geometric non-linear behaviour of the system [9,10,13,14,15,16,17].
The FE model is now considered in its modal space, by converting from geometric coordinates to modal coordinates, $\boldsymbol{q}$. This transformation can be applied using the linear modal transform

$$\mathbf{x} = \boldsymbol{\Phi}\boldsymbol{q}\,, \quad [4]$$

where $\boldsymbol{\Phi}$ represents the $N \times N$ matrix, which consists of the system's linear mode shapes. These modeshapes are mass-normalised, meaning that each mode shape has a mass contribution of “1”, such that the following equation is satisfied [9,10,14,15,16,17]

$$\boldsymbol{\Phi}^T\mathbf{M}\boldsymbol{\Phi} = \mathbf{I}. \quad [5]$$

The reason for mass normalising is to scale each mode consistently, allowing for a more meaningful comparison of force contributions. This new mass-normalised modal transformation also alters the stiffness matrices into $\boldsymbol{\Lambda}$ [9,10,14,15,16,17],

$$\boldsymbol{\Phi}^T K\boldsymbol{\Phi} = \boldsymbol{\Lambda}. \quad [6]$$

As a result of the mode shape matrix being mass-normalised, this new $N \times N$ matrix, $\boldsymbol{\Lambda}$, becomes a diagonal matrix possessing the squares of the corresponding natural frequencies [9,10,14,15,16,17]. Such that the following eigenvalue problem can be solved by using the natural frequencies present in $\boldsymbol{\Lambda}$

$$(\mathbf{K} - \omega_n^2\mathbf{M})\boldsymbol{\Phi}_n = \mathbf{0}. \quad [7]$$

This modal coordinate transformation can then be substituted into Equation [1] to create an EoM that describes the motion modally, as opposed to a geometric coordinate system,

$$\ddot{\mathbf{q}} + \boldsymbol{\Lambda}\mathbf{q} + \mathbf{f}_q(\mathbf{q}) = \mathbf{F}_q. \quad [8]$$

Equation [8] expresses the system's EoM in modal coordinates, which separates the linear and non-linear modal contributions, making it easier to identify and analyse how non-linearity influences each mode. The presence of the coupling term, $\mathbf{f}_q(\mathbf{q})$, introduces the computational challenge that traditional FEA analysis faces. Since this prevents the decoupling of the governing equations of motion, and results in computationally expensive simulation times [10,14,15,16,17].

2.3 Reduced Order Modelling

ROM offers a balance between computational efficiency and precision for the dynamic analysis of geometrically non-linear systems. Historical techniques focus on two different types of methods, intrusive and non-intrusive [15,16,18]. In this report, the primary technique focused on is the non-intrusive method.

The primary difference between intrusive and non-intrusive methods lies in how the ROM parameters are determined and what they represent. The Intrusive methods depend on the prior knowledge of the coefficients of the full-order EoM, relating to $\mathbf{K}^{(2)}$ and $\mathbf{K}^{(3)}$ in Equation [3] [15,18,19,20]. Conversely, non-intrusive methods function independently of any information about the full-order systems EoMs [15,18,19,20]. This makes non-intrusive methods more suited towards applications where the full-order equations of motion are unknown, such as in industrial settings. Thus, making non-intrusive methods more robust and more widely applicable [15,18,19,20].

As non-intrusive methods operate without requiring direct access to the full-order equations of motion, certain data from the full-order model (FOM) must be extracted to ensure the ROM is properly calibrated and accurately represents the system. This consists of a set of static equilibrium solutions that capture the system's non-linear behaviour, allowing the ROM to predict the original model's response accurately [15,18, 19,20].

The primary objective when working with reduced-order modelling is to project the full-order model onto a reduced subset of modes [15,18,19]. A new modal matrix, $\mathbf{\Phi}_r$, consisting of the mode shapes corresponding to the selected reduced modes can be formed, where r denotes the chosen reduced modes [15,18,19].

As before, a modal transform can be applied, but now with this reduced subset of modes. To achieve a closer approximation of the full-order model's response, this transformation must also account for the geometric non-linearities present in the system, [21,22]

$$\mathbf{x} = \mathbf{\Phi}_r \boldsymbol{r} + \mathbf{\Phi}_s \boldsymbol{g}(\boldsymbol{r}) \,, \qquad [9]$$

where $\mathbf{\Phi}_r$ has its original meaning of the modeshape matrix, but of the reduced coordinate system, and $\boldsymbol{r}$ is the $r \times 1$ vector of the selected reduced coordinates [21,22]. The additional $\mathbf{\Phi}_s \boldsymbol{g}(\boldsymbol{r})$ term represents the effect of the response from the suppressed modes. These suppressed modes are approximated as functions of the retained coordinates through quasi-static coupling, a type of modal coupling for LNMs based on their dynamic responses [21,22]. Typically, the coupling function takes the form of a quadratic function of the retained modes, [21,22]

$$\boldsymbol{g}(\boldsymbol{r}) = \mathbf{B}\boldsymbol{r}\boldsymbol{r}, \qquad [10]$$

Where $\mathbf{B}$ represents a tensor consisting of the coupling coefficients, and the function $\boldsymbol{g}(\boldsymbol{r})$ represents the quasi-static modal coupling function [21,22]. Later, physical examples of quasi-static coupling are given to illustrate the concept more clearly. Applying this transformation to the original EoM, Equation [1], results in a new equation of motion for the system, using only the selected reduced modes while accounting for the geometric non-linearities present

$$\ddot{\mathbf{r}} + \mathbf{\Lambda}_r \boldsymbol{r} + \mathbf{f}_r(\boldsymbol{r}) = \mathbf{\Phi}_r^T \mathbf{F}, \qquad [11]$$

where all terms have their previous meanings, and the additional quasi-static coupling function is included in the non-linear $\mathbf{f}_r(\boldsymbol{r})$ term.

For indirect methods, a set of non-linear static solutions derived from finite element models is needed to calibrate the ROM. In this project, the force-based method was chosen over the alternative displacement-based approach [14,18,19] because it more effectively captures modal coupling effects when the reduced subspace includes only low-frequency modes [14,18,19].

The force-based method involves applying a static force to the system that follows the shape of the relevant reduced modes. Such that the deformations can reflect the system's non-linearity when these reduced modes are excited. The resulting non-linear displacements are then extracted and substituted into Equation [12] to determine the non-linear coefficients for the ROM. Equation [12] is a rearranged static form of Equation [11], [14,18,19]

$$\mathbf{f}_r(\boldsymbol{r}) = \mathbf{\Phi}_r^T \mathbf{F} - \mathbf{\Lambda}_r \boldsymbol{r}. \qquad [12]$$

2.4 Machine Learning

Machine Learning models learn patterns and relationships through the use of neural networks. These neural networks consist of layers of neurons that apply mathematical functions to a series of inputs to produce outputs. These layers are interconnected, such that the output of one neuron becomes the input of another. The input and output layers in the neural network consist of the parameters the model is trying to learn a relationship for, and the values that are being predicted. In between the input and output layers are hidden layers, which are additional layers of neurons that enable the network to learn complex patterns by allowing multiple stages of neuron processing. Figure 1 represents the architecture of a simple Neural Network.

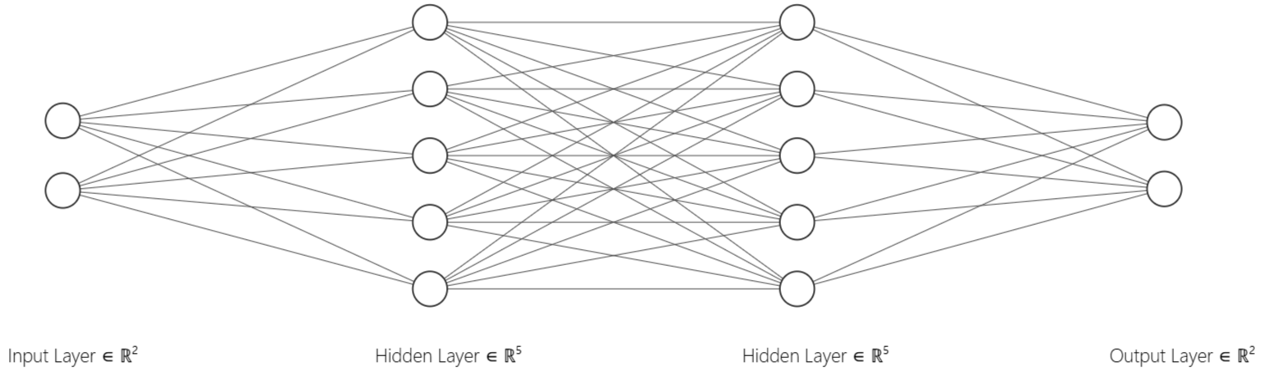

Fig 1 - A Neural Network architecture diagram representing a feed-forward neural network with input and output layers of size 2 and 2 hidden layers of size 5 each.

Each neuron’s input is a weighted sum of outputs from the previous layer, this weighted sum follows a linear summation,

$$z = \sum_{i=1}^{n} w_i \cdot x_i + b_i, \qquad [13]$$

where w_i is the weight, x_i is the input and b_i is the bias term of neuron i. This weighted sum is then passed through an activation function.

Non-linearity is introduced into the system via activation functions. These functions introduce non-linearity into the system by applying non-linear functions to the outputs of each neuron, which enables the network to learn complex relationships. With multiple hidden layers, these activation functions allow groups of neurons to respond to specific patterns, features, or ranges when predicting an output.

In this report, the machine learning algorithm utilises Rectified Linear Unit (ReLU) activation functions for the input and hidden layers due to its simplicity. This simplicity enables the model to converge faster and allows the network to handle complex patterns effectively [23]. The equation representing the ReLU activation function can be seen in Equation [14], [23]

$$f(x) = max(0, x) \, . \qquad [14]$$

The machine learning model processes system inputs, such as stiffness values and beam geometries, through these layers of neurons. Where each layer applies weights and biases to the input data, which then passes through an activation function before being passed on to the next layer.

A loss function then measures the error between the network's predictions and the actual target values. This makes it a crucial part of an ML model, as it enables the model to assess the accuracy of its predictions. One of the most commonly used loss functions is the Mean Squared Error (MSE), which calculates the average squared differences between the predicted and actual values [24],

$$MSE = \frac{1}{N}\sum_{i-1}^{N}(y_i - \widehat{y}_i)^2, \qquad [15]$$

where y_i and $\widehat{y}_i$ represent the real and predicted values of the model. ML models are designed to learn by adjusting these neuron parameters by minimising the error between their predictions and the actual data. The parameters are adjusted using an optimisation algorithm. Many optimisation algorithms are available, each with distinct strengths and weaknesses. [25]

The Adam optimiser [26] was selected in this study due to its ability to minimise errors and converge efficiently. It achieves this through the use of 2 main properties, momentum and adaptive learning rates [26]. Momentum tracks the gradients of the loss function with respect to the changing ML parameters, while adaptive learning rates adjust the step size for each parameter based on how quickly it is learning and improving. This combination allows the Adam optimiser to find solutions more rapidly and effectively than many other optimisers. [25] In this thesis, a comparison is made between simple optimisers, such as Stochastic Gradient Descent (SGD), and more complex optimisers, like Adam, to highlight the importance of selecting the correct optimiser.

In many systems, the order of magnitude of parameters and data points differs significantly in size. The problem that ML algorithms face with this difference in size is that the algorithm may perceive these larger values to be more important. Data normalisation ensures that features with larger numerical ranges do not dominate the training process and takes the form of

$$x_{norm} = \frac{x-\mu}{\sigma}. \qquad [16]$$

This normalisation centres the data around zero by subtracting the mean, μ, and scales the data such that the standard deviation, σ , is 1. This helps machine learning by making features more comparable, which leads to an improvement in convergence for gradient-based optimisations.

3 Motivating Example

A common type of non-linear modal coupling is membrane stretching, which is a type of coupling between LNMs [17,18]. Physically, this can refer to a system which is forced in one mode, but a response in a separate mode appears. This phenomenon can be used to help select the reduced modes which characterise the main dynamics of the system, as mentioned before.

To explore and validate this phenomenon and its potential application in forming a ROM, a simple 2-degree-of-freedom (2-DoF) oscillator is used to demonstrate and explain these effects. A similar example is also explored by Nicolaidou E in "Accounting for Quasi-Static Coupling in Nonlinear Dynamic Reduced-Order Models" [27]. Figure 2 illustrates the 2-DoF system, which allows for motion in both the x and y axes. When this system is excited purely in a single axis, the system may appear to respond in that direction, as expected in linear dynamics. However, once this excitation increases, it becomes apparent that this is not the case and motion in the y-direction begins to appear, despite not being directly excited [27].

This unexpected response is due to the geometric non-linearity present in the system, which causes coupling between the degrees of freedom [27]. As the mass oscillates along the x-axis, this geometric non-linearity causes the restoring force now to become a function of the x and y directions. This corresponding secondary mode can then be treated as a function of the first since it was not originally excited.

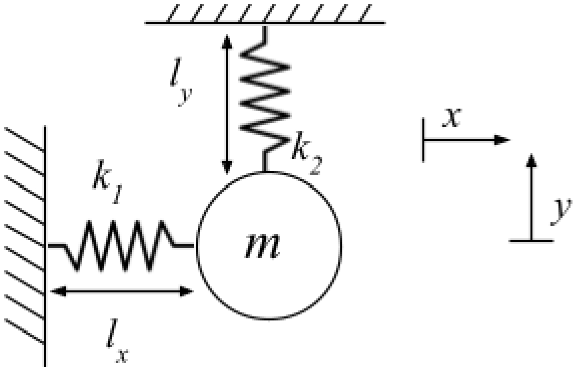

Fig 2 - A diagram of the 2DoF system used for the motivating example

The 2-DoF oscillator used consisted of a point mass, *m*, of mass 0.5 kg, constrained by two perpendicular springs. These springs have unstretched and undeformed lengths, l_x and l_y, of 0.1m and have distinctly different stiffnesses of k_1 and k_2, being 50 N/m and 1000 N/m, respectively. Maintaining a large difference between these stiffnesses is essential for clearly demonstrating these coupling characteristics.

Although this 2-DoF system is computationally inexpensive due to its low number of degrees of freedom, a ROM will still be developed to demonstrate the ROM methodology and its applicability to dynamic systems.

To simulate the full-order system, the assumed equation of motion must be calculated. This was done by first calculating the kinetic energy of the mass and the potential energy stored in the springs. Then, using the Lagrangian, 2 equations of motion were derived. Representing the motion in the x and y directions, respectively

$$m\ddot{x} + (k_1 + k_2)x + k_1 l_0\left[1 - \frac{(x+l_0)}{\sqrt{y^2+(x+l_0)^2}}\right] - k_2 l_0\left[\frac{x}{\sqrt{x^2+(y+l_0)^2}}\right] = 0, \quad [17]$$

$$m\ddot{y} + (k_1 + k_2)y - k_1 l_0\left[\frac{y}{\sqrt{y^2+(x+l_0)^2}}\right] + k_2 l_0\left[1 - \frac{(y+l_0)}{\sqrt{x^2+(y+l_0)^2}}\right] = 0. \quad [18]$$

To simplify the analysis, these equations can be simply referred to as

$$m\ddot{x} + F_1(x, y) = 0, \quad [19]$$

$$m\ddot{y} + F_2(x, y) = 0. \quad [20]$$

Since the ROM is formulated and depended on dominant modal responses, it is essential to express the system in terms of its LNMs. A linear modal transform was used to express these equations of motion in terms of the linear modal coordinates [27]

$$q_1 = mx, \quad [21]$$

$$q_2 = my. \quad [22]$$

These transformations can be substituted into Equations [19] and [20], to create

$$\ddot{q}_1 + F_1(q_1, q_2) = 0, \quad [23]$$

$$\ddot{q}_2 + F_2(q_1, q_2) = 0. \quad [24]$$

In the above equations q_i correspond to the primary modes of oscillation and F_i correspond to the coupled stiffness functions. The two equations of motion proved the modal coupling within this system and were numerically simulated to analyse the system's full-order dynamic behaviour. To test the coupling theory, the system was excited purely in the x-direction, ensuring that no external force was applied in the y-direction. Such that this excitation correlated to an excitation of the first mode, q_1. The system was then allowed to freely oscillate from this equilibrium position.

During this simulation, the displacements in the x and y directions were both tracked over time. This enabled the creation of a phase portrait, as shown in Figure 3.

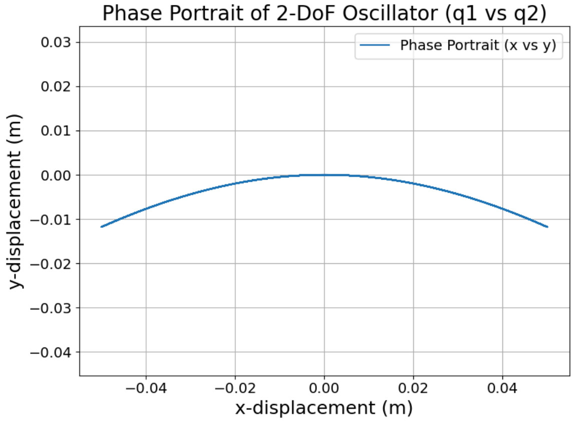

Fig 3 - The phase portrait of the 2DoF system when released from an equilibrium position

The phase portrait of the full-order system illustrates the system's coupling behaviour. Even though the 2DoF oscillator was forced exclusively in the first mode (x-direction), a response in the second mode (y-direction) is observed, as evidenced by an arched path. This means that for this loading case, the non-excited DoF can be treated as a function of the primary, horizontally excited DoF.

To develop a reduced-order model of the 2-degree-of-freedom system, the dominant modes which form the reduced basis must first be identified. Given that the stiffness k_2 is significantly larger than that of k_1, it can be assumed that the 2nd linear natural frequency of the system is much higher than the first. This means q_1 can be chosen to create the reduced basis.

q_1 can be chosen as the basis for the reduced-order model because the second mode is assumed to be quasi-statically coupled to the first [21,22,27]. Quasi-static coupling implies that the second mode behaves as a static, non-linear function of the dominant first mode. Since the natural frequency of q_2 was deemed much higher, the inertial term can be disregarded in the equations of motion. By applying this assumption to Equation [24], a function relating q_1 and q_2 can be created,

$$F_2(q_1, q_2) = 0. \qquad [25]$$

This equation can then be rearranged to express the q_2 response as a function of q_1. This function of q_2 in terms of q_1 can be substituted into Equation [23] to create

$$\ddot{q}_1 + F_1(q_1) = 0, \qquad [26]$$

where $F_1(q_1)$ consists of a complex equation involving q_1 such that it can be represented as a high-order polynomial function [27]. In this report, it is assumed that the polynomial will be of the 9th order. So the main dynamics of the system can be captured in the following reduced basis equation of motion

$$\ddot{q}_1 + \omega^2 q_1 + \sum_{i=2}^{9} q_1^i \alpha_i = 0, \qquad [27]$$

where α_i represents the coefficients for each monomial. These coefficients are determined using the static solutions from the FOM.

To obtain the required static displacements, the original EoMs, Equations [17] and [18], were utilised. However, because the focus is on static behaviour, the inertial terms were removed from both equations, and the right-hand side of Equation [17] was set equal to the applied static force in the first mode. These EoMs were solved simultaneously to compute the static deflection of q_1 for each loading case. [27]

The static form of the ROM was then fit to the FOM static data to determine the non-linear coefficients for the ROM. The static form of the ROM refers to Equation [27] after removing the inertial term and equating the right-hand side to the static force. This was then fit to the FOM static dataset using a least squares approach, which determined the coefficients for each monomial in the ROM.

Often, in dynamics, dynamicists focus on frequency responses rather than time-domain responses. This is due to the ability of frequency responses to capture system behaviours and trends more effectively than time domain responses, where one would need to run many simulations simultaneously to draw meaningful conclusions. In linear systems, this can typically involve tools such as Nyquist and Bode plots. In non-linear dynamics, an analogous analysis tool is a backbone curve.

A backbone curve illustrates how the system's response frequency changes with vibrational amplitude, providing insight into the system's non-linear behaviours. It effectively captures the periodic NNM present in the system. Where each NNM describes the intrinsic, coordinated motions that the system undergoes, and the backbone curve shows how these characteristics evolve with amplitude and frequency [27].

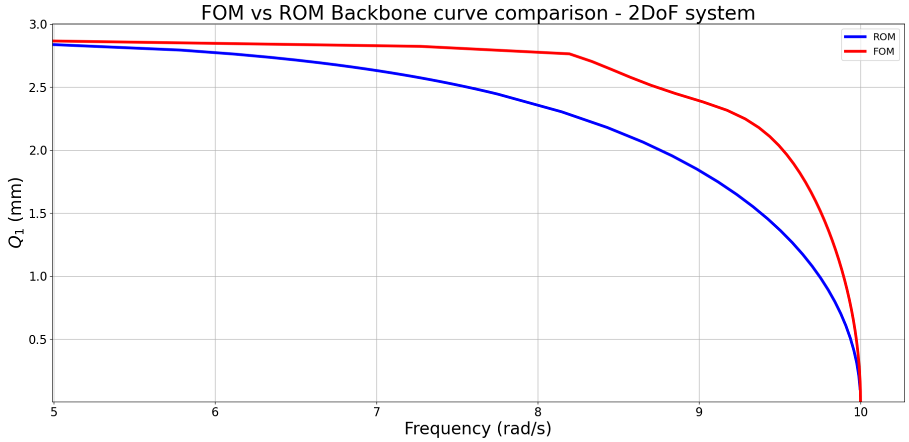

Fig 5 - A comparison between the ninth order ROM and the FOM, showing how the amplitudes of oscillation vary with frequency against the response amplitude.

Figure 5 illustrates the difference between the ROM and the FOM models for the 2Dof system. As seen, the ROM accurately captures the system's main dynamics. The significant reduction in computational cost offered by the ROM often justifies the slight trade-off in accuracy. Later in the study, these discrepancies will be addressed and minimised through the use of data-driven optimisation methods. In the context of this report, computational cost will be evaluated in terms of simulation runtime.

It is worth noting that if a linear assumption were used, the response would be represented by a vertical line at the first natural frequency, 10rad/s. This indicates that linear assumptions are only valid for small deflections, as the low-amplitude solution approaches the linear natural frequency of the system's first mode.

To ensure consistent measurement of computation times, all simulations were run on the same hardware. This research utilised a 6-core AMD Ryzen 5 3600 processor, operating under identical conditions with minimal background processes running. This approach guarantees a fair comparison between the FOM and the ROM computational times. Each simulation was run 10 Times, and the average simulation time was found to ensure that any random discrepancies could be accounted for.

The FOM was found to take 151.94 seconds to simulate. In contrast, the process to create the ROM, including static data generation and polynomial fitting, took 28.55 seconds, resulting in an 81.18% reduction in total computational time, while maintaining high accuracy.

Although the absolute value of the time saved seems modest, the result demonstrates the computational efficiency of reduced-order modelling. In more complex systems, the benefits of reduced-order modelling become significantly more evident by offering a substantial reduction in simulation times without compromising accuracy. This example illustrates the process and validation of ROM for dynamic systems that exhibit geometric non-linearities and demonstrates the computational efficiency and accuracy of the method.

4 Application to a more complex system

4.1 ROM vs FOM for an FE model

The feasibility of applying reduced-order modelling and machine learning for a more complex FE system is now considered. For complex models, traditional analytical techniques often become obsolete due to the system's complexity. In such cases, finite element software is used to analyse these systems numerically. Abaqus [28] was selected as the FEA platform due to its capability to handle geometric non-linearities and dynamic simulations effectively.

The complex system chosen for analysis was a clamped-clamped beam. While other systems, such as cantilever beams, were considered, they often exhibited additional nonlinear effects that fall outside the scope of this study. This application demonstrates that if ROM or ML techniques perform well on simple non-linear FE structures, they can be expected to carry over this performance for even more complex systems.

The first step in identifying the backbone curve for the system involved extracting the system's linear mass and stiffness matrices.

Before using these matrices to solve the eigenvalue problem, the boundary conditions of the system were first defined, since in Abaqus, fixed boundary conditions alter the stiffness matrices. Both ends are fully restrained for a clamped beam, with no translation or rotational movement at the boundaries. In Abaqus, these constraints are implemented by assigning very large stiffness values, on the order of 10^{36}, to the corresponding DoFs. The DoFs that correspond to these values were removed from the mass and stiffness matrices to ensure these artificially large values did not skew the eigenvalues. These new reduced matrices were then used to solve the eigenvalue problem, which calculated the system’s linear natural frequencies and mode shapes. The lowest natural frequencies were then identified, and the modes which would create the reduced basis were selected.

For the system to exhibit periodic behaviour, it must be released from appropriate initial conditions, as described in the invariant manifold approach for NNMs. To determine these positions, a set of static modal forces, in the form of the selected dominant modes, were applied to the system, and the resulting equilibrium displacements were extracted. These displacements corresponded to the initial conditions from which the system was released.

These static solutions were also used to construct the system’s ROM, obtained by recording the static displacement of the reference coordinate, as described by the force-based approach. The midpoint of the beam was selected as an appropriate reference location as it corresponded to the point of maximum displacement under the first linear mode shape.
Using this force-displacement data, the static 9th-order ROM was fitted to determine the non-linear coefficients. Table 1 presents the characteristics of the chosen clamped-clamped beam, which was made from stainless steel. The material properties were obtained from Granta EduPack [29].

Table 1 - The characteristics of the clamped-clamped beam and their associated values

Characteristic	Value
Length	750mm
Width	25mm
Height	5mm
Young's modulus (Steel)	257GPa
Density (Steel)	7850kgm^{-3}
Poisson Ratio (Steel)	0.3

The system was initially displaced with the modal equilibrium position obtained from the static loading. Once displaced, the system was allowed to oscillate freely from these initial conditions. The physical displacement and velocity histories of all the unconstrained degrees of freedom were recorded and then converted into modal response values.

The periodic motions for each force value were determined through the use of a zero-crossing function. The zero-crossing function identifies the periodic motions of a system by calculating the time it takes the system to return to its initial conditions. From this, the relevant frequencies and the corresponding amplitudes were extracted. Repeating this process for different force values created the backbone curve.

The system was excited through an appropriate force range, such that the non-linearities were excited to a non-negligible amount. It should be noted that the force values required to trigger this non-linear behaviour are specific to each system, varying with geometry, material properties, and boundary conditions. Similarly, the resulting deflection magnitudes that characterise non-linear behaviour also differ across systems, requiring calibration on a case-by-case basis. The resulting backbone curve for the FOM of the FE clamped-clamped beam is shown in Figure 6**.**

The lowest natural frequency was found to be 117.2 rad/s, corresponding to the first mode, with the subsequent lowest natural frequency of 636.1 rad/s, corresponding to the second mode. The natural frequency of the second mode was sufficiently larger than the first mode, such that a single-mode reduced-order model of the first mode is appropriate to model this system.

From Figure 6, it is evident that the system exhibits strong non-linear behaviour. The relationship between response amplitude and frequency is observed to change significantly, indicating that a linear analysis would be insufficient for capturing the system's dynamics. This present non-linearity resulted in a significant FEA computational cost. The system was found to contain 1795 DoFs, resulting in a total computational cost for running this simulation was 2107.20 seconds.

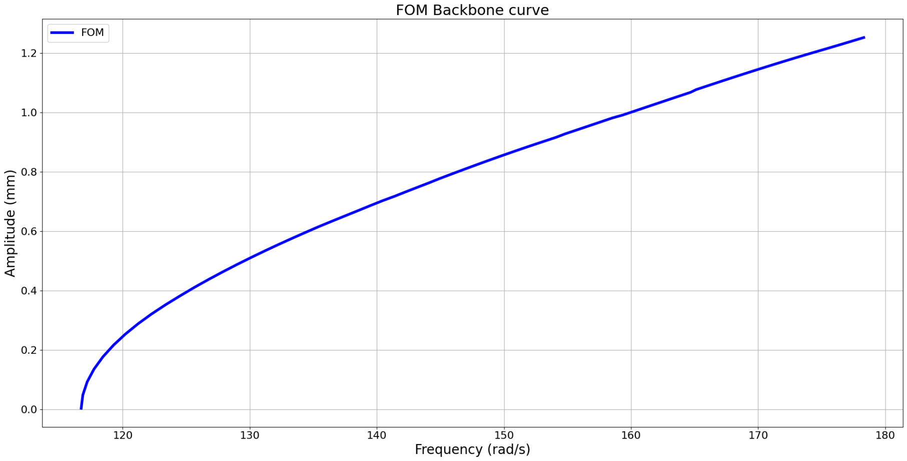

Fig 6 - The resulting backbone curve produced by the full-order clamped-clamped beam system, simulated in Abaqus

Figure 7a shows that the static form of the ROM accurately captures the static behaviour of the full-order model, indicating that the ROM can account for the effects of the suppressed modes under the quasi-static assumption. Since the ROM only requires nine coefficients to be determined, only nine static displacement points need to be obtained. Any additional points will lead to an over-constrained problem, resulting in a waste of computational resources.

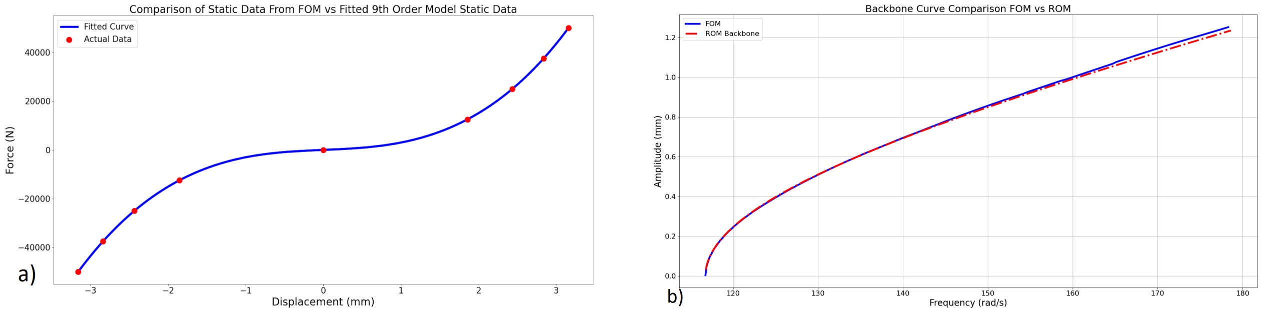

Fig 7 - a) The static solution dataset extracted from Abaqus by applying a static force and recording the resulting displacements and the fitting of the ninth-order static ROM to this data b) The comparison between the FOM backbone curved compared to the ROM for the clamped-clamped beam.

The calibrated ROM could now be simulated dynamically to produce its backbone curve. The backbone curve created by the ROM is visible in Figure 7b. It exhibits a strong resemblance to the full-order model, capturing the main frequency-amplitude relationship. To compare the accuracy of this method with other methods, the MSE of the ROM prediction was calculated. Where the error corresponded to the frequency difference for each response value. The ROM showed an MSE of 1.01428 (rad/s)^2. This indicates that, on average, each predicted frequency deviated from the actual value by 1.0071 rad/s for a given response amplitude.

The computational cost of just the ROM simulation was 18.35 seconds. This represents a significant reduction in computational time for the ROM versus FOM simulations.
However, this only accounts for the simulation part of the ROM. To capture the full computational cost, the cost of calibration and static deflection data generation must also be included. These computational costs for the static displacements and the polynomial fitting were 116.25 seconds and 0.01 seconds, respectively. Resulting in a total computational cost of 134.61 seconds. This resulted in a 93.61% decrease in computational cost when compared to the FOM simulation. These results again highlight the effectiveness of reduced-order modelling for appropriate systems, significantly reducing computational costs whilst retaining high accuracy.

4.2 Machine Learning

4.2.1 Initialisation

The machine learning model maps input parameters to output predictions based on patterns learned during the training process. This process consisted of feeding the model a training dataset of backbone curves generated from similar structures with varying geometries. The training data was manually generated and consisted of beams with coarser meshes to ensure that training data generation times remained computationally feasible, since no widely available dataset for this system existed. The inputs to the model consist of the varying parameters for each beam, in this case, the length. The outputs corresponded to the dynamic response, consisting of amplitude and response frequency.

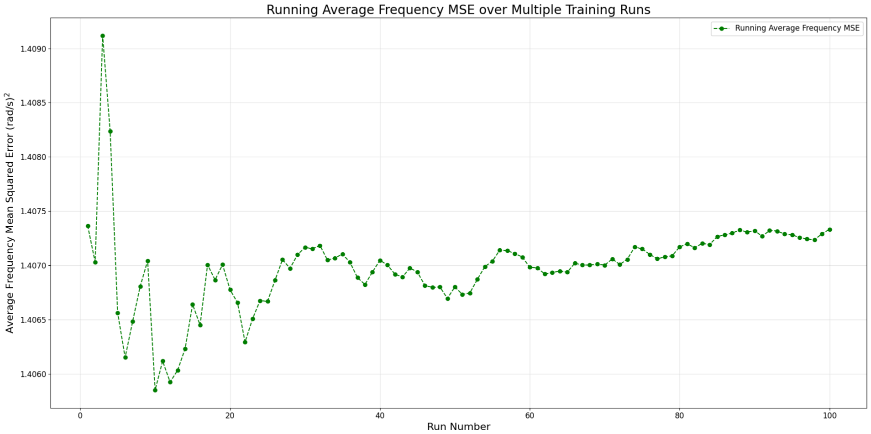

Fig 8 - The evolution of the average MSE over time for a Machine Learning model for the clamped-clamped beam system.

To initialise each ML training operation, the initial weights and biases were randomised to speed up convergence. A result of this is that the finalised MSE of the prediction varied every time it was run. The evolution of the average MSE was investigated to determine the true effectiveness of each model. Figure 8 illustrates the evolution of the average MSE across repeated training runs for an arbitrary system.

It is important to note that this MSE corresponds to the model's performance, how well it fits the backbone curves it was trained on, rather than the MSE of the predicted clamped-clamped beam being modelled. The figure shows that if each ML model were run only once, the resulting MSE would not accurately represent that specific model. Some convergence can be observed around 60 iterations, although true convergence is not achieved due to inherent randomness.

In this study, whenever an MSE for a model was found, the model was trained 60 times separately, and an average MSE was determined, ensuring that each model's average MSE was representative of that specific model.

4.2.2 Training Data

The more training data there is, the more data and situations the machine learning model can learn from, hence increasing the model's robustness. This comes with a downside of increased computational cost associated with training this data. To determine the optimal set of training data, an accuracy vs training data range study was performed to investigate this relationship. This study involved training an ML model on a dataset with equal starting and finishing values of lengths, but varying step sizes. The resulting model MSE was then calculated, and each combination was run 60 times to ensure convergence to an average MSE.

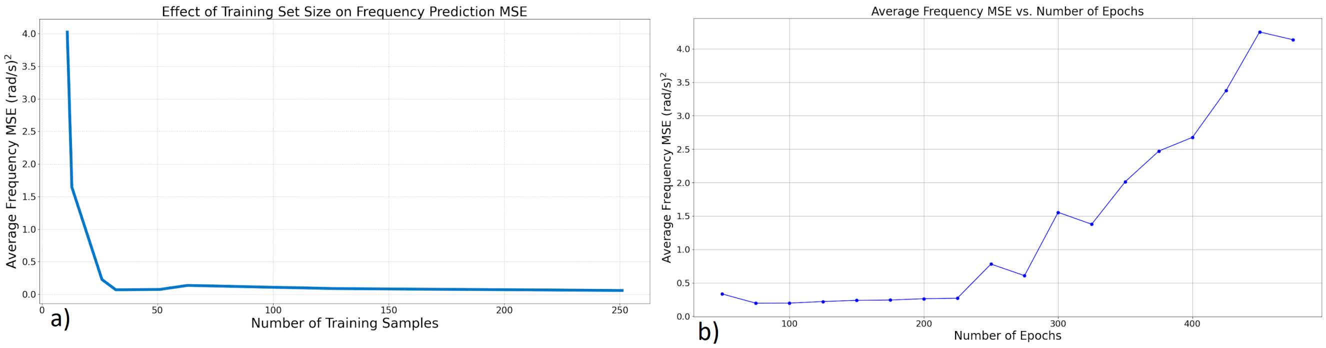

Fig 9 - a) The relationship between the Number of training samples used to train an ML model and the MSE of that model, b) The relationship between the MSE and the number of Epochs for a given ML model

Figure 9a shows that beyond 50 training samples, the MSE exhibits diminishing returns. While using more samples in the training set may lead to minor improvements in the MSE, it also increases the computational cost of training this data, unless a pre-computed dataset is available. To optimise the ML model, the lower value was taken to ensure minimal computational costs whilst retaining high accuracy.

The range and distribution of training values also significantly impact the model's accuracy and robustness. Training on 50 values within a narrow range (e.g., 725–775 mm) resulted in higher accuracy, but reduced robustness. Whilst training on a broader range (e.g., 550–1050 mm) had the opposite effect. The wider range was used to ensure that this ML model accurately represented a realistic model, as this would allow for predictions of multiple systems for which typical ML models are used.

An important point to note is that when selecting training data ranges, the value of the model that is ultimately being predicted must fall within this range of training values. Otherwise, the model's precision will lose significant amounts of accuracy and become outright useless, as the model will not have learnt patterns for data of this range.

4.2.3 Epochs

An epoch refers to one complete pass through the entire training data. Increasing the number of epochs allows the model to learn underlying patterns in the data more accurately. It would seem intuitive that the more epochs used, the more accurate the prediction will be. An epoch vs accuracy study was carried out to determine the actual relationship. This involved training a model using various numbers of epochs and comparing the MSE values of these models.

As shown in Figure 9b, it is clear that the relationship between the number of epochs and model accuracy is not strictly proportional. After a certain point, increasing the number of epochs causes the models MSE to rise. This is due to overtraining and overfitting, which occur when the model is trained for too long and starts to learn false patterns in the training data rather than the actual underlying trends. As a result, the model's performance drastically reduces. The increase in epochs also results in a higher computational cost, as more pass-throughs are required. Therefore, it is crucial to select an appropriate number of epochs to maintain efficiency. For this system, 50 epochs were chosen to be optimal.

4.2.4 Results and ML Optimisation selection

With these hyperparameters set to their identified optimal values, which limit computational cost while retaining accuracy, the ML model can now be used to predict the response for the full-order clamped-clamped beam. It also allows the study of different model optimisers, SGD vs Adam, for this study.

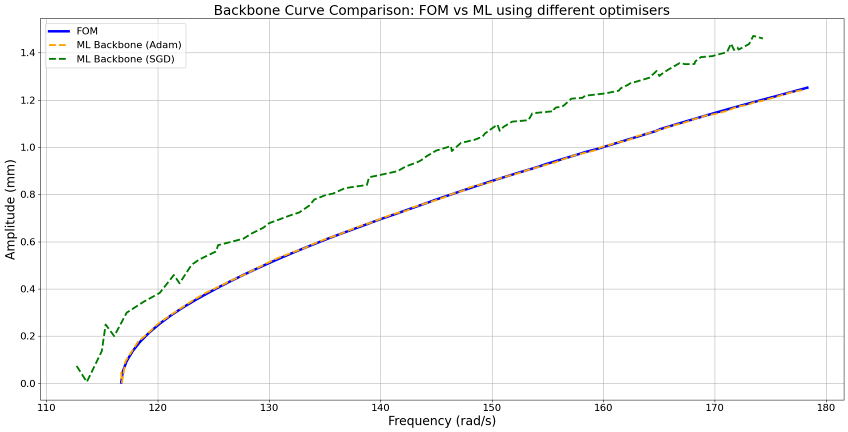

Fig 10 - The comparison between the FOM backbone curve and the machine learning prediction for the clamped-clamped beam.

It can be seen from Figure 10 that the ML model is highly accurate in predicting the FOM response when an appropriate optimiser is used. Both ML models were trained using the same training data ranges, epochs, and network architectures. The model trained using the Adam optimiser shows a significantly more accurate prediction than the model trained using SGD. The MSEs were 0.028791 (rad/s)^2 for Adam and 207.700029 (rad/s)^2 for SGD. Utilising the accuracy value for the Adam case, the ML prediction is 97.16% more accurate in predicting the FOM response than the ROM.

This improved accuracy can be credited to Adams' adaptive learning rates and momentum-based updates, which help prevent the model from getting stuck in local minima. The SGD optimiser, which lacked these properties, demonstrated this by reaching a minimum MSE in just 18 epochs and then fluctuating about that error for the remaining epochs, resulting in a higher MSE compared to the Adam optimiser. These results highlight the importance of choosing the right optimiser for the task.

Once the ML model is trained, the computational cost of predicting the backbone curve is extremely low, at just 0.01 seconds. Just like the ROM, the computational constraints come from training and calibrating the model. Specifically, the main cost is attributed to creating the training data and training the model. These values were found to be 50795.35 seconds and 7.14 seconds, respectively. Resulting in a combined computational cost of 50802.49 seconds.

The effectiveness of the ML model depends on whether the pre-computed training data is readily available. If not, the cost of achieving this accuracy is 377 times the computational cost of the ROM and a 24 times increase from the original FOM.

However, if this pre-computed data set is readily available, the MSE will remain the same. However, the cost of achieving this accuracy is significantly reduced, showing a 94.68% decrease in computational time compared to the ROM and a 99.66% decrease compared to the FOM. Table 2 summarises the computational cost and accuracy of the ROM, ML, and FOM for these scenarios.

Table 2 - The comparison of the FOM, ROM and ML models. Comparing the computational cost along with the MSE of each method

Model Type	Computational Time (s)	MSE $(rad/s)^2$
FOM	2107.20	-
ROM	134.61	1.014284
ML (No pre-computed dataset)	50802.49	0.028791
ML (Pre-computed dataset)	7.15	0.028791

5 Optimisation of ROM

Table 2 compares the accuracies of each method, with the ROM experiencing the highest MSE. This is expected, as the ROM is based on assumptions of dominant modes, meaning it will, by nature, be unable to fully capture system behaviours. In contrast, ML uses predictions based on data and trends to more accurately predict system behaviours.

To limit the MSE as much as possible, the ROM was optimised based on the FOM dynamic response. This optimisation process consisted of iteratively adjusting the non-linear coefficients of the ROM to minimise the prediction error. The same MSE loss function was used where the error corresponded to the frequency difference for each response value. The optimisation process worked by randomly adjusting each coefficient, simulating the ROM with these parameters and then comparing the MSE. If the MSE decreased, the algorithm saved these parameters and continued the process with the new parameters until convergence or a specified number of iterations had been reached.

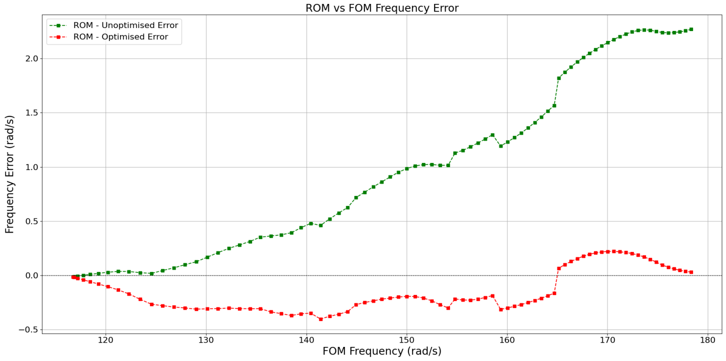

Fig 11 - The comparison of the optimised ROM and the unoptimised ROM for the frequency error for each response value

Figure 11 shows that the average error in the optimised ROM has significantly reduced compared to the original ROM. This can be further evidenced by the MSE of the optimised ROM, 0.054547 $(rad/s)^2$**,** resulting in a 94.67% error reduction from the original ROM, which incurs an additional computational cost of 37.34 seconds. However, this significant increase in accuracy comes with a major downside, as it limits the range of applications for which this ROM can be used. Since it is calibrated based on the FOM response for a given forcing range, this ROM can not be simulated with these coefficients for different force ranges. To optimise for a different force range, one would have to create a new ROM within this new force range and then apply the optimisation. Another disadvantage is that this optimisation process requires knowledge of the FOM response, making it only suitable for scenarios where repeated analysis and evaluation of a system are necessary, as the FOM response would need to be simulated.

6 Conclusion

This report has demonstrated the challenges of analysing systems which exhibit strong geometrical non-linear behaviours. Systems which are often associated with high-performance requirements and complex design considerations. For these systems, traditional linear methods, which are commonly used by dynamists, become ineffective and computationally expensive for FE systems. In this research, the computational cost associated with a simple FE model of a clamped-clamped beam with 1795 DoFs was 2107.20 seconds.

The research aimed to reduce this computational cost through the use of theoretical modal reduction techniques and emerging data-driven approaches, while retaining accuracy and optimising these processes to maximise their performance.

Each method presents its own set of advantages and disadvantages and are better suited toward specific scenarios. ROM is advantageous in scenarios where computational resources and times are severely limited and approximate yet still accurate analyses are acceptable. In this study, ROM achieved a 93.61% reduction in computational cost compared to the FOM, making it suitable for cases such as early-stage design analysis, where cheap and approximate analysis are appropriate. The ROMs accuracy can then be significantly improved through the use of optimisation processes. In this case, optimisation improved accuracy by 94.67% relative to the unoptimised ROM by iteratively adjusting the ROM coefficients, while increasing computational cost by only 37.24 seconds.

ML methods excel in scenarios where high accuracy and computationally cheap analysis are required. The effectiveness depends on whether pre-existing training datasets are available. For systems where this data is available, ML models can deliver very precise and cost-effective predictions, achieving a 97.16% increase in accuracy compared to the unoptimised ROM and a 94.68% decrease in computational time, making it suitable in all scenarios where the FOM model requires high computational challenges. However, if this database is not available, it significantly reduces the effectiveness of ML models, since the computational cost of creating this training data is immense. Resulting in a 24 and 377 times increase in computational cost compared to the FOM and the ROM, respectively.

It is also important to note that the effectiveness of ML methods depends heavily on hyperparameter choices, such as the optimiser, training data range, number of epochs and model architecture. Improper configuration was found to significantly degrade both accuracy and efficiency.

Ultimately, the choice between using ROM and ML depends on the availability of training data for the ML model. ROMs can be calibrated on a case-by-case basis, allowing them to be applied to a wide range of systems as long as the correct loading conditions are present. In contrast, the effectiveness of ML models depends heavily on the availability of pre-existing training data, which is typically more abundant for standard systems, making their use more tailored. Both methods are capable of producing accurate and low-cost predictions, with the ML model generally achieving the best performance in both metrics.

7 FUTURE WORK

While this project successfully developed and compared ROM and ML approaches for geometrically non-linear systems, several directions could be explored in future work to further improve accuracy and performance.

Hybrid ROM-ML Integration
An area for development and research is the use of machine learning to augment the ROM. In this thesis, machine learning was not technically used to optimise the ROM, as there was no training data due to time constraints. A hybrid framework could be developed where an ML model is trained on residuals and ROM predictions to further enhance the accuracy and adaptability of the ROM.

Application of ROM to non-dynamic systems
While this project focused on dynamic structural systems, ROM techniques could be applied to other types of systems, such as fluid dynamics, thermodynamics, control systems and other areas. Future work could investigate whether ROM is as effective in these domains or develop analogous tools and identify its strengths and limitations.

Multi-Modal ROMs
This work focused primarily on single-mode ROMs. Future studies could explore the performance of multi-mode ROMs, where multiple dominant modes are selected. The accuracy and computational cost associated with these ROMs could be compared to that of single-mode ROMs for those systems to determine how critical it is to use Multi-Modal ROMs in these scenarios.

Alternative ROM error metrics
In this project, the ROM and ML accuracy were assessed based on frequency differences. Future work could consider other error metrics, such as the differences between the final point of the oscillation and the actual initial conditions. This error could then be identified and corrected by an ML model or optimised.

8 REFERENCES

[1] Abdullah AM, Ain QU, Kaleem U, Khan, Riaz J (2023). Modelling, Simulation, and Control of a Flexible Space Launch Vehicle. pp. 1-2.
[2] Fonzi N, Brunton SL, Fasel U (2020). Data-driven nonlinear aeroelastic models of morphing wings for control. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences. 476(2239): 1-2.
[3] Wang Z, Lu Z, Yi W, Hao J, Chen Y (2023). A study of nonlinear aeroelastic response of a long flexible blade for the horizontal axis wind turbine. Ocean Engineering. 279: 1-2.
[4] Ozan Gözcü, Dou S (2020). Reduced order models for wind turbine blades with large deflections. Journal of Physics: Conference Series. 1618(5): 052046, pp. 1-3.

[5] Lazarus A, Thomas O, Deü JF (2012). Finite element reduced order models for nonlinear vibrations of piezoelectric layered beams with applications to NEMS. Finite Elements in Analysis and Design. 49(1): 35–37.

[6] Gaëtan Kerschen, Maxime Peeters, Jean Claude Golinval, Cyrille Stéphan (2013). Nonlinear Modal Analysis of a Full-Scale Aircraft. Journal of Aircraft. 50(5): 1409-1413.

[7] Antoine Chaigne, Cyril Touzé, Olivier Thomas (2005). Nonlinear vibrations and chaos in gongs and cymbals. Acoustical Science and Technology. 26(5): 403-410.

[8] Han Q, Wu C, Liu M, Wu H (2022). Numerical Methods for Geometric Nonlinear Dynamic Analysis of Suspension Bridges Under Earthquake. Journal of Earthquake Engineering. 27(6): 1355–1357.

[9] Nicolaidou E (2022). Reduced-order modelling of nonlinear dynamic structures. University of Bristol. pp. 4-7.

[10] Touzé C, Vizzaccaro A, Thomas O (2021). Model order reduction methods for geometrically nonlinear structures: a review of nonlinear techniques. Nonlinear Dynamics. 105(2), pp 1–13.

[11] Zhang C, et al. (2025). The AI Index 2025 Annual Report. Stanford Institute for Human-Centered Artificial Intelligence. pp. 3-4.

[12] Rosenberg RM (1960). Normal Modes of Nonlinear Dual-Mode Systems. Journal of Applied Mechanics. 27(2): 263–268.

[13] Kerschen G, Peeters M, Golinval JC, Vakakis AF (2008). Nonlinear normal modes, Part I: A useful framework for the structural dynamicist. Mechanical Systems and Signal Processing. 23(1): 171–180.

[14] Nicolaidou E (2022). Reduced-order modelling of nonlinear dynamic structures. University of Bristol. pp. 8-12.

[15] Lazarus A, Thomas O, Deü JF (2012). Finite element reduced order models for nonlinear vibrations of piezoelectric layered beams with applications to NEMS. Finite Elements in Analysis and Design. 49(1): 38–40.

[16] Tartaruga I, Elliott A, Hill TL, Neild SA, Cammarano A (2019). The effect of nonlinear cross-coupling on reduced-order modelling. International Journal of Non-Linear Mechanics. 116: 7–16.

[17] Nicolaidou E (2022). Reduced-order modelling of nonlinear dynamic structures. University of Bristol. pp. 13-18.

[18] Mignolet MP, Przekop A, Rizzi SA, Spottswood SM (2013). A review of indirect/non-intrusive reduced order modeling of nonlinear geometric structures. Journal of Sound and Vibration. 332(10): 2441–2443.

[19] Gordon R, Hollkamp J (2011). Reduced-order models for acoustic response prediction. pp. 7-26.

[20] Hollkamp JJ, Gordon RW (2008). Reduced-order models for nonlinear response prediction: Implicit condensation and expansion. Journal of Sound and Vibration. 318(4-5): 1140–1143.

[21] Nicolaidou E (2022). Reduced-order modelling of nonlinear dynamic structures. University of Bristol. pp. 16-18.

[22] Nicolaidou E, Melanthuru R, Hill T, Neild A (2022). Accounting for Quasi-Static Coupling in Nonlinear Dynamic Reduced-Order Models. University of Bristol. pp. 1-12.

[23] Dubey SR, Singh SK, Chaudhuri BB (2021). Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark. pp. 92-108.

[24] IBM (2024). Loss Function [Internet]. Available from: https://www.ibm.com/think/topics/loss-function

[25] Schmidt RM, Schneider F, Hennig P (2020). Descending through a Crowded Valley - Benchmarking Deep Learning Optimisers. pp. 1-8.

[26] Kingma DP, Ba J (2017). Adam: A Method for Stochastic Optimization. pp. 1-4.

[27] Nicolaidou E, Melanthuru R, Hill T, Neild A (2022). Accounting for Quasi-Static Coupling in Nonlinear Dynamic Reduced-Order Models. University of Bristol. pp. 4-12.

[28] Dassault Systèmes (2017). Abaqus. Dassault Systèmes, Providence, RI.

[29] ANSYS Inc. (2022). Granta EduPack. ANSYS Inc., Cambridge, UK.