Lina BEN-YOUNES

Supervised by Dr Renson

**Project Final Report**

**Automatic Detection of Mode Interactions Using Machine Learning**

**ME4 Individual Project**

Imperial College London

Department of Mechanical Engineering

Word count:

Word limit: 10,000 June 8th 2023

# Abstract

This report examines how coupled vibration modes could be automatically detected using well-established Machine Learning techniques. Combining the use of wavelet transforms and support vector machines, the presented method detects interactions between the fundamental and harmonics up to five, with an accuracy of

In order to train the support vector machines, scalogram databases were computed using the Morse wavelet transform from the MATLAB Wavelet Toolbox. The time series analysed were obtained by integrating beam and duffing equations using Python’s scipy.integrate module.

The report also presents a method which does not rely on Machine Learning techniques but rather on hard-coded logic intended to mimic the reasoning physicists would use to discern interactions. That method is less computationally demanding and allows for more flexibility in defining the threshold for detecting interaction but it fails to exploit the large amount of information given by the scalogram.

A table of contents.

Table des matières

[Abstract 2](#_Toc135992409)

[Introduction and Objectives 4](#_Toc135992410)

[Context 4](#_Toc135992411)

[Objectives 4](#_Toc135992412)

[Literature review and general concepts 6](#_Toc135992413)

[Nonlinear Normal Modes and mode interaction 6](#_Toc135992414)

[Continuous wavelet transform & Scalograms. 8](#_Toc135992415)

[Support vector machines 8](#_Toc135992416)

[Accuracy & Crossvalidation 9](#_Toc135992417)

[Implementation 10](#_Toc135992418)

[The Python Package. 10](#_Toc135992419)

[First Package: No\_ML 10](#_Toc135992420)

[Second Package: ML\_method 10](#_Toc135992421)

[Time series: Beam equation 10](#_Toc135992422)

[Producing clean data: format and pre-processing 11](#_Toc135992423)

[Attributing a class to the training dataset 13](#_Toc135992424)

[Support Vector Machine Parameters 15](#_Toc135992425)

[Results and Discussion. 16](#_Toc135992426)

[Conclusions and Future Work 17](#_Toc135992427)

[References 18](#_Toc135992428)

# Introduction and objectives

## Context

See to add references.

Confronted with complex nonlinear structures, engineers analyse various nonlinear responses, one of them being coupled vibration modes. In a nonlinear system, different vibrational modes may influence each other, affecting their individual behaviour and characteristics. These mode interactions can lead to changes in modal frequencies and energy transfer between modes.

They pose a challenging threat to the integrity of the concerned structures. Indeed, energy transfers from a local mode of low effective mass components to a global mode with high effective mass might critically jeopardize the structure.[1]

Nowadays, these interactions are detected by trained-eyed physicists and engineers. They conduct frequency sweeps and inspect scalograms of the structure’s response.

Method

The algorithms will have to be trained with those scalograms, which entails creating scalogram datasets and classifying them according to whether they present modal interactions. We will also have to write scaling and pre-processing algorithms to obtain clean data and increase the accuracy and efficiency of the machine-learning models.

We will then train and test different Machine Learning models and techniques in order to select those best suited for this classification problem. Finally, the algorithms will be tested on real-life data and examples.

## Objectives

The project goals are:

- Identifying the patterns, parameters, and favourable conditions of modal interactions.

- Creating scalogram datasets of nonlinear models with and without model interactions.

- Creating a procedure to pre-process and standardize the format of the data on which the support vector machines (SVM) are trained and used.

- Training and adjusting the parameters of the SVM.

- Testing them on real-life data.

# Literature review and general concepts

## Nonlinear Normal Modes and mode interaction

Mode interactions are a prominent feature of Nonlinear Normal Modes (NNMs). If the system’s natural frequencies are commensurate, an energy exchange between the different modes may be observed during the internal resonance.

Kerschen, G., Peeters, M., Golinval, J., & Vakakis, A. (2009). Nonlinear normal modes, Part I: A useful framework for the structural dynamicist. *Mechanical Systems and Signal Processing, 23*(1), 170-194. <https://doi.org/10.1016/j.ymssp.2008.04.002>

When exciting one mode, one may produce a large-amplitude response on another. Such interactions are identified as n:m interactions, meaning the n-th harmonic of a mode matches the m-th of another.

Such interaction may therefore be observed when the system:

* Is non-linear: if the system is linear, the resonances are decoupled.
* Is subject to high forcing: for low forcing, the equations governing the system may be approximated by their linearization.
* Its frequencies fi and fj are approximately such that: k.fi = p.fj, for low integer values of k and p. We may observe k:p interactions.

In such cases, mode interaction would be identified by the presence of high amplitude coefficients for frequency fj when the system is subject to excitation of frequency fi. So in the case of 1:k interactions, the scalogram would present high amplitude coefficients for the k-th harmonic.

An example can be found in [?].

|  |
| --- |
| Une image contenant texte, Tracé, capture d’écran, ligne  Description générée automatiquement |
| Figure [?] from [?] |

The scalogram in Figure [?] displays the response of a small satellite to a frequency sweep. One identifies the 2:1 interaction, hereby circled, between two internally resonant modes of the structure. The second harmonic of mode 3 matches the frequency of mode 7. Mode 3 involves an out-of-phase motion of the inertia wheel, and mode 7 consists of an axial motion of the telescope supporting panel.

Noël, J., Renson, L., & Kerschen, G. (2014). Complex dynamics of a nonlinear aerospace structure: Experimental identification and modal interactions. *Journal of Sound and Vibration, 333*(12), 2588-2607. <https://doi.org/10.1016/j.jsv.2014.01.024>

## Continuous wavelet transform & Scalograms.

Wavelet transforms are mathematical tools used to analyse changing features in data. As the response to a frequency sweep is studied, the feature analysed is frequency.

Wavelet transforms address the limitations of the Fourier transform. While Fourier analysis quantifies the contributions of sine waves of specific frequencies over the full length of the signal, wavelet analysis decomposes signals into wavelets shifted in time and scaled in frequency. A wavelet is a rapidly decaying, wave-like oscillation. This enables wavelets to represent data across multiple scales.

Inc. The MathWorks. (nd) *Wavelet Transforms in MATLAB.* <https://fr.mathworks.com/discovery/wavelet-transforms.html4> [Accessed 20th June 2023]

Wavelet transforms can be classified into continuous wavelet transforms (CWT) and discrete wavelet transforms (DWT). Unlike DWT, discrete approximations of CWT are redundant in information. However, DTW only allows for a restrictive choice of wavelets and is not shift-invariant, the wavelet analysis method used throughout the project is CWT and therefore, this section will cover continuous wavelet transforms.

Inc. The MathWorks. (nd) *Continuous and Discrete Wavelet Transforms* <https://uk.mathworks.com/help/wavelet/gs/continuous-and-discrete-wavelet-transforms.html> [Accessed 22nd June 2023]

We should think of the graph of as a single ‘cycle’, the wavelet is assumed such that:

* It is compactly supported.

But it may be chosen such that it has additional properties such as smoothness. The mother wavelet is dilated by the scale and shifted to the position . We define:

The scale parameter a is inversely proportional to the instantaneous frequency at time b. We shall therefore define the continuous wavelet transform such that we get the relation:

Where is a constant.

The continuous wavelet transform of f corresponding to a choice of wavelet is:

Meyer, Y. (1992), *Wavelets and Operators*, Cambridge, UK: Cambridge University Press

As mentioned in the abstract, physicists use scalograms to characterise mode interaction. Scalograms are graphs displaying the absolute value of the CWT of a signal, plotted as a function of time and frequency, as seen in Fig [?].

## Support vector machines

The scope of the project was limited to 1:i interactions for values of i < 6. The detection of mode interactions corresponds to a classification problem. The different classes are:

* No interaction
* 1:2 interaction
* 1:3 interaction
* 1:4 interaction
* 1:5 interaction

The common Machine Learning classification techniques are Naïve Bayes, Random forests, Support Vector Machines (SVMs), and K-Nearest-Neighbours (KNN). The choice of SVMs for this project is justified by the format of the data used in this project.

Scalograms were transformed into a set of vectors each corresponding to a time sample of the scalogram, each of these vectors must be attributed a class. The dimension of the vectorial space is 120.

* Naïve Bayes relies on strong independence assumptions between the features of the data, which are not valid on the constructed dataset.
* K-Nearest-Neighbours: KNN is very computationally expensive, not only when trained but also when used to predict a class. Therefore, it would not be suited to classify data in a vectorial space of dimension 120.
* Random forests are a collection of decision trees. This method is best suited for data that are a mixture of numerical and categorical features, which is not the case in this project.

This section will therefore cover SVM algorithms only. The aim of Support Vector classification is to find optimal separating hyperplanes in a high-dimensional feature space. There are several ways to define the optimality of the hyperplane, in this project, the maximal margin classifier was used.

Cristianini, N., Shawe-Taylor, J. (2005). *Support Vector Machines*. In Cambridge University Press eBooks (pp. 93–124). <https://doi.org/10.1017/cbo9780511801389.008>

This means that SVM considers the training points at the extreme of their classes, i.e., the support vectors, and places the decision boundary, or separating hyperplanes, as far from these as possible. The margin is the shortest distance between the support vectors and the boundary. The SVM algorithm searches for a decision boundary that would maximise the margin.

|  |
| --- |
| Figure 1 : Définir la « marge » entre les classes : le critère que les SVM cherchent à optimiser. |
| Figure [?] 2D example of Support Vector Machine  Inc. The MathWorks. (nd) *Support Vector Machines* <https://fr.mathworks.com/discovery/support-vector-machine.html>  [Accessed 28th June 2023] |

### Linear SVM

Equations

Linear discriminant functions:

### Kernels

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

It uses kernels to project the input data from the input space to a higher (infinite) dimension called the feature space where the data is linearly separable, which corresponds to a hyper-plane in the input space as shown in the example in Figure 13. Numerous kernels are used in SVM applications, including quadratic (or radial), Gaussian, linear, and many more.

The other common parameter is γ, which defines how much influence a single training sample has on the hyper-plane position. The higher the value of γ, the greater the "bending" of the boundary decision in the input space.

This project used the radial basis function kernel or rbf kernel. If the hyper-plane fits the data very "tightly" in which the plane touches all three support vectors, it often translates to overfitting in the input space and a "loose" fit in the feature space often translates to lower accuracy but better generalisation capabilities.

Therefore, a budget function must be defined to control the amount of fitting in the higher dimensions, also known as the margin error, C. The higher the value of C, the higher the error allowed in the fitting and more misclassified points but the smaller the overfitting to the training data. The rbf kernel has a second tuning parameter,

In order to learn non-linear relations with a linear machine, we need to select a set of non-linear features and to rewrite the data in the new representation. This is equivalent to applying a fixed non-linear mapping of the data to a feature space, in which the linear machine can be used. Hence, the set of hypotheses we consider will be functions of the type

where F is a non-linear map from the input space to some feature space. This means that we will build non-linear machines in two steps: first a fixed non-linear mapping transforms the data into a feature space F, and then a linear machine is used to classify them in the feature space. As shown in Chapter 2, one important property of linear learning machines is that they can be expressed in a dual representation. This means that the hypothesis can be expressed as a linear combination of the training points, so that the decision rule can be evaluated using just inner products between the test point and the training points:

If we have a way of computing the inner product ()) in feature space directly as a function of the original input points, it becomes possible to merge the two steps needed to build a non-linear learning machine. We call such a direct computation method a kernel function

### Soft/Hard Margin

In the case of a hard margin, no support vector can be on the wrong side of the boundary, this is at the expense of a more complex decision boundary. A soft-margin SVM gives a smoother decision boundary by misclassifying some training points.

Cristianini, N., Shawe-Taylor, J. (2005). *Support Vector Machines*. In Cambridge University Press eBooks (pp. 93–124). https://doi.org/10.1017/CBO9780511801389.008

|  |  |
| --- | --- |
| Une image contenant art, peinture, Caractère coloré, capture d’écran  Description générée automatiquement | Une image contenant art, peinture, Caractère coloré, capture d’écran  Description générée automatiquement |
| 1. Hard Margin SVM | 1. Soft margin SVM |
| Figure [?] Comparison between Hard and Soft Margin SVMs [same ref] | |

Soft margin consists in introducing a parameter C called the budget, which quantifies to which extent one allows the margin constraint to be violated.

## Accuracy & Cross-validation

Accuracy & Specificity

# Implementation

To automatise the detection of mode interactions, a Python package was created. Like most ML projects, the package was implemented in 3 phases: dataset building, pre-processing, and testing.

## The Python Package.

The goal was to create a package containing 6 functions in two Python modules, a demo file and a dataset.

### First Package: No\_ML

The No\_ML package was meant to address the issue of automation with no Machine Learning techniques. It would mimic the reasoning physicists would use to discern interactions. Its functions were:

*scale(scalogram, frequencies, f0, fend)* which returns the scaled scalogram and takes the following arguments:

* *scalogram* is the unscaled scalogram (matrix of size [time array size] x [frequency array size])
* *frequencies* is the list of corresponding frequencies.
* *f0* and *fend* are the initial and final frequencies in the conducted frequency sweep.

*interaction2(scalogram, scaled=True, threshold=2, frequencies=None, f0=None, fend=None)* returns a dictionary, the keys correspond to the interaction and the values are the interaction score for each interaction. Its parameters are:

* *scalogram* is the scalogram to analyse, it is assumed to be scaled, unless one specifies *scaled=False* and therefore must give values to the scaling parameters frequencies, *f0* and *fend*.

*norm(scalogram, i, scaled=True, frequencies=None, f0=None, fend=None)* returns the relative amplitude for the i-th harmonic, with the same arguments as previously.

### Second Package: ML\_method

*interaction(scalogram, model=default, scaled=True, frequencies=None, f0=None, fend=None)* has the same purpose as *interaction2* but relies on SVM.

*model(threshold=2, chosen\_kernel='linear')* creates and trains the SVM for the wished threshold. One can specify different kernels but the default is linear.

*Cross\_validate(model, threshold=2, k=5)* prints the accuracy on training and testing datasets.

## Time series: Beam equation

The datasets were created using equations adapted from that of a cantilever beam with a nonlinear spring at the tip in [paper].

With the following notations:

* is the parameter of interest, it is the static force–displacement of the beam resulting from the excitation computed at three different axial coordinates, and therefore a 3x1 matrix.
* is a 3x1 matrix where is the i-th modal response of the system.
* is the mode shape matrix. It is a 3x3 matrix such that is the mode shape of the i-th mode at axial coordinate
* is the mode shape vector at the tip of the cantilever.
* , where is the resonant response frequency of the n-th mode.
* is the damping coefficient.
* was a frequency sweep.
* is the non-linear function that introduces the mode interaction.

Shaw, A. D., Hill, T. R., Neild, S. A., & Friswell, M. I. (2016). Periodic responses of a structure with 3:1 internal resonance. *Mechanical Systems and Signal Processing*, 81, 19–34. <https://doi.org/10.1016/j.ymssp.2016.03.008>

Various values of and were used to create the database. The resonant response frequencies were chosen such that , for .

The time series were computed using the fourth-order Runge-Kutta function of Python’s SciPy library. The wavelet transforms were computed using MATLAB’s cwt function in the Wavelet Toolbox.

## Producing clean data: format and pre-processing

A scalogram corresponds to a list of three-dimensional data, a time value, a frequency, and an amplitude. The format of the scalograms computed by MATLAB is an array containing the values of frequency, and a matrix containing (ai,j), where ai,j is the amplitude of frequency indexed i in the frequency array, at time tj.

It makes no difference that the scalogram displays an interaction at time t1 rather than t2, we are interested in whether the scalogram presents interaction at all. The time parameter is therefore irrelevant to the identification problem. If the time array was of length n, the scalogram is split into n data points. Each data point is a frequency array and an amplitude array. Figure [?] (a) presents three of these data points.

The scalogram is then scaled along its frequency axis. The amplitude of the excitation frequency should be the same coefficient in the amplitude array. This is generalised to all the amplitude coefficients. The n-th coefficient in the amplitude array is the amplitude of frequency , for . The standard frequency array is now

We chose to stop at because we limited the scope of the project to detecting interactions up to 1:5.

Using the frequency array (fi) and the amplitude array (ai,j), the data points are created by dividing (fi) by the excitation frequency at time tj. We then get and the corresponding amplitudes. Using interpolation, we compute the amplitudes corresponding to , a default value can be used to avoid extrapolation, as seen in Figure [?]. This is essentially what the function *scale(scalogram, frequencies, f0, fend)* does in the Python package.

|  |  |
| --- | --- |
| Une image contenant texte, diagramme, capture d’écran, Tracé  Description générée automatiquement | 1. Unscaled data points |
| Une image contenant texte, diagramme, capture d’écran, Tracé  Description générée automatiquement | 1. Normalised frequency |
| Une image contenant texte, diagramme, capture d’écran, Tracé  Description générée automatiquement | 1. Interpolated |
| Figure [?] 3 data points through the steps of pre-processing. | |

The last part of the pre-processing was to normalise the amplitudes, such that the amplitude of the excitation frequency would always be 1. The integral of amplitude over the excitation frequency was computed and the amplitude array was then divided by it.

An example of a scaled scalograms is presented in Figure [?]

|  |  |
| --- | --- |
| Une image contenant capture d’écran, texte, ligne, Caractère coloré  Description générée automatiquement | Une image contenant texte, capture d’écran, Caractère coloré, ligne  Description générée automatiquement |
| 1. Unscaled scalogram | 1. Scaled scalogram |
| Fig [?] Dataset presenting 1:2 interactions.  The top right part of (b) correspond to information that is not present in the original scalogram and is therefore replaced with a default value. That value is chosen to be the 25th percentile of the coefficient in the data point. | |

## Attributing a class to the training dataset

We define five norms:

* , for

Each scalogram was scaled and thenPlot

* is the amplitude of the i-th harmonic

|  |  |  |
| --- | --- | --- |
| Une image contenant capture d’écran, texte, ligne, Caractère coloré  Description générée automatiquement | Une image contenant texte, capture d’écran, Caractère coloré, ligne  Description générée automatiquement | Une image contenant texte, Tracé, capture d’écran, diagramme  Description générée automatiquement |
| 1. Unscaled scalogram | 1. Scaled scalogram | 1. Amplitudes |
| Fig [?] Dataset presenting 1:2 interactions | | |

|  |  |  |
| --- | --- | --- |
| Une image contenant capture d’écran, texte, Rectangle, ligne  Description générée automatiquement | Une image contenant texte, capture d’écran, Caractère coloré  Description générée automatiquement | Une image contenant texte, capture d’écran, Tracé, diagramme  Description générée automatiquement |
| 1. Unscaled scalogram | 1. Scaled scalogram | 1. Amplitudes |
| Fig [?] Dataset presenting 1:3 interactions | | |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 1. Unscaled scalogram | 1. Scaled scalogram | 1. Amplitudes |
| Fig [?] Dataset presenting 1:4 interactions | | |

|  |  |  |
| --- | --- | --- |
|  |  | Une image contenant texte, capture d’écran, Tracé, diagramme  Description générée automatiquement |
| 1. Unscaled scalogram | 1. Scaled scalogram | 1. Amplitudes |
| Fig [?] Dataset presenting 1:5 interactions | | |

## Support Vector Machine Parameters

Data imbalance for each before / after class imbalance

# Results and Discussion.

Added benefit of machine learning

Demo file

# Conclusions and Future Work

Blah

Different sweeps

Different modes

Data imbalance

# References

(see Appendix D).