UNIVERSITY OF THESSALY
SCHOOL OF ENGINEERING
DEPARTMENT OF MECHANICAL ENGINEERING

Identification of Structures using Vibration Measurements

Nikolaos A Dimitriou

Vasiliki T Tsilomeleti

System Dynamics Laboratory

Submitted in partial fulfillment of the requirements for the degree of Diploma in Mechanical Engineering at the University of Thessaly

Volos, October 2023

Approved by the Committee on Final Examination:

1. **Prof. Costas Papadimitriou**
   Professor of Structural Dynamics
   Department of Mechanical Engineering, University of Thessaly, Greece
   email: costasp@uth.gr

2. **Assoc. Prof. Bouzakis Emmanouil**
   Associate Professor of Manufacturing Processes
   Department of Mechanical Engineering, University of Thessaly, Greece
   email: ebouzakis@uth.gr

3. **Assoc. Prof. Konstantinos Ampountolas**
   Professor of Automatic Control Systems in Mechanical Engineering
   Department of Mechanical Engineering, University of Thessaly, Greece
   email: k.ampountolas@uth.gr

IDENTIFICATION OF STRUCTURES USING VIBRATION MEASUREMENTS

Abstract

The primary objective of this thesis is to identify the dynamic properties of structures using vibration measurements, with a specific focus on the Chalkida cable bridge. This was achieved by utilizing vibration data obtained from accelerometers strategically placed on the bridge, specifically gathering ambient data to derive essential parameters such as natural frequencies, damping ratios, and mode shapes. To facilitate this analysis, we developed a MATLAB code capable of importing data from the sensors, plotting each sensor's data separately in the time domain for visual validation.

To determine the natural frequencies, Welch's method was employed, converting the data into the frequency domain. Power spectral density diagrams were then generated, revealing the inherent natural frequencies of the structure. Further refinement of modal properties was conducted using the SSI_COV (Subspace System Identification of Covariance) method, a MATLAB function obtained from the web. This method operates on time domain sensor data, providing matrices containing modal frequencies, mode shapes, and damping ratios.

To ensure the reliability of our modal properties, the initial 30-minute dataset was divided into two 15-minute segments. The SSI_COV function was applied separately to each segment, allowing us to identify modal properties for each segment. Additionally, the modal assurance criterion was utilized to assess the similarity between the mode shapes extracted from each segment.

It is important to note that the MATLAB code developed for this analysis is versatile and can be applied to a wide range of structures, irrespective of size or the number of sensors used. This work not only advances our understanding of the dynamic behavior of the Chalkida bridge but also presents a universally applicable methodology for analyzing similar structures.

Περίληψη

Ο κύριος στόχος αυτής της διπλωματικής είναι να καθοριστούν οι δυναμικές ιδιότητες των κατασκευών, με ειδική εστίαση στη καλωδιωτή γέφυρα της Χαλκίδας. Αυτό επιτεύχθηκε χρησιμοποιώντας δεδομένα που συλλέχθηκαν από αισθητήρες που τοποθετήθηκαν στη γέφυρα, συγκεκριμένα συλλέγοντας δεδομένα περιβαλλοντικής κατάστασης για την εξαγωγή βασικών παραμέτρων, όπως ιδιοσυχνότητες, πυκνότητες απόσβεσης και τρόπους ταλάντωσης. Για να διευκολύνουμε αυτήν την ανάλυση, αναπτύξαμε έναν κώδικα MATLAB ικανό να εισάγει δεδομένα από τους αισθητήρες, να σχεδιάζει τα δεδομένα κάθε αισθητήρα ξεχωριστά στο πεδίο του χρόνου για οπτική επαλήθευση. Στη συνέχεια, χρησιμοποιήθηκαν μόνο επαληθευμένα δεδομένα στην ανάλυση.

Για τον προσδιορισμό των φυσικών συχνοτήτων, χρησιμοποιήθηκε η μέθοδος Welch's, μετατρέποντας τα δεδομένα στο πεδίο της συχνότητας. Δημιουργήθηκαν στοιχειομετρικά διαγράμματα πυκνότητας ισχύος, αποκαλύπτοντας τις φυσικές συχνότητες της δομής. Η περαιτέρω βελτίωση των ταλαντώσεων πραγματοποιήθηκε χρησιμοποιώντας τη μέθοδο SSI_COV (Ταυτοποίηση Συστήματος Υποχώρου της Κατανομής), μια συνάρτηση MATLAB που αποκτήθηκε από το διαδίκτυο. Αυτή η μέθοδος εφαρμόζεται στα δεδομένα των αισθητήρων στο πεδίο της συχνότητας, παρέχοντας πίνακες που περιέχουν τις ιδιοσυχνότητες, τις μορφές κύματος και τις πυκνότητες απόσβεσης.

Για να διασφαλίσουμε την αξιοπιστία των ταλαντώσεων μας, τα αρχικά δεδομένα 30 λεπτών διαμερίστηκαν σε δύο τμήματα των 15 λεπτών. Η συνάρτηση SSI_COV εφαρμόστηκε ξεχωριστά σε κάθε τμήμα, επιτρέποντάς μας να εντοπίσουμε τις ταλαντώσεις για κάθε ένα από αυτά. Επιπλέον, χρησιμοποιήθηκε το κριτήριο ταλαντωτικής εγγύησης για να αξιολογηθεί η ομοιότητα μεταξύ των μορφών κύματος που εξήχθησαν από κάθε τμήμα.

Σημειώνεται ότι ο κώδικας MATLAB που αναπτύχθηκε για αυτήν την ανάλυση είναι ευέλικτος και μπορεί να εφαρμοστεί σε μια ευρεία γκάμα κατασκευών, ανεξαρτήτως μεγέθους ή αριθμού αισθητήρων που χρησιμοποιούνται. Αυτή η εργασία όχι μόνο προάγει την κατανόηση της δυναμικής συμπεριφοράς της γέφυρας της Χαλκίδας, αλλά παρουσιάζει επίσης μια εφαρμόσιμη μεθοδολογία για την ανάλυση παρόμοιων κατασκευών.

Acknowledgments

First of all, we would like to thank our thesis supervisor, Prof. Costas Papadimitriou, for his valuable assistance and guidance during our studies at the Department of Mechanical Engineering. His high level of knowledge and teaching attracted us in the Structural Dynamics Field.

In addition upon the completion of our studies, we would also like to thank the members of the Academic Staff of the Department for providing us knowledge and foundations in the field of Mechanical Engineering.

We would like to extend our sincere appreciation to Dr. Konstantinos Morfidis and Dr. Thomas Salonikios, both Researchers at the Earthquake Planning & Protection Organization (E.P.P.O.), Institute of Engineering Seismology and Earthquake Engineering. Their invaluable support and collaboration in providing critical measurements under ambient vibrations for the Chalkida bridge have been instrumental in advancing this research.

Above all, we are grateful to our parents for their love and constant support over the years.

Contents

List of Figures

List of Tables

1 Modal Analysis [1]

Modal analysis is a technique employed to gain insight into the dynamic behavior of a system by determining its natural frequencies, damping factors, and mode shapes. It involves breaking down the system's vibrations into a set of simple harmonic motions known as natural modes of vibration. These modes are determined by the physical properties and spatial distributions of the system. Modal analysis can be carried out either theoretically, using mathematical models and differential equations, or experimentally through modal testing, which involves data acquisition and processing techniques. Both approaches yield valuable modal data that aids in comprehending and predicting the dynamic response of structures.

The primary objective of modal analysis, whether theoretical or experimental, is to establish the modal model of a dynamic system. This modal model offers a detailed representation of the system's dynamic characteristics, encompassing natural frequencies, damping factors, and mode shapes. Subsequently, this modal model finds applications in various areas such as design, problem-solving, and analysis.

Theoretical modal analysis commences with the physical properties of the system, such as mass, stiffness, and damping, and utilizes them to derive the modal model. This process entails describing the system using matrices that represent these properties. On the other hand, experimental modal analysis derives the modal model from measured data, such as frequency response functions (FRFs) or free vibration response data. This approach involves extracting the modal parameters from the measured data.

Once the modal model is obtained, it can be employed in several ways. Some applications involve the direct utilization of the modal data obtained from measurements, while others utilize the modal data for further analysis and calculations. These applications include:

1. Structural dynamics and vibration control: Modal analysis aids in comprehending and predicting the dynamic behavior of structures, enabling the design and implementation of effective vibration control strategies.

2. Finite element model updating: Modal analysis can be used to refine and update finite element models of structures based on measured modal data. This improves the accuracy of the models and enhances their predictive capabilities.

3. Structural health monitoring: Modal analysis can be utilized in monitoring the health of structures by tracking changes in their modal parameters. Deviations from the baseline modal data can indicate damage or degradation in the structure.

4. Modal synthesis: Modal analysis allows for the synthesis of complex vibration patterns by combining individual mode shapes and their corresponding modal parameters. This enables the creation of realistic vibration simulations and virtual testing scenarios.

The main natural frequencies of the Euripus cable-stayed bridge in Chalkida are presented. The recordings from the bridge's permanent monitoring system, conducted in real-time, are utilized for this purpose. Welch's method is employed for analyzing these recordings, and the frequencies corresponding to the peaks observed in the graphs are determined. The conducted modal analysis is considered accurate in three aspects: i) determining the number of significant eigenvalues, ii) determining the relative modal mass for each mode, and iii) determining the direction of modal oscillation (whether along or around one of the three axes). Additionally, a comparison is attempted between the natural frequencies obtained from the current sensor arrangement and the eigenvalues (along with their corresponding mode shapes) in different periods of the bridge's operation. This allows for the identification of significant differences, which, if present, require interpretation to assess the structural integrity of the bridge. The excitation level of the dominant modes, which represent the highest percentages of the active mass of the structure, naturally depends on the magnitude of the bridge excitation.

2 Cable-stayed bridge of Euripus in Chalkida [2]

Figure 1: Euripus Bridge in Chalkida

2.1 The bridge

The high cable-stayed bridge of Euripus is part of the Chalkida bypass road, connecting the Voiotian coast with the Evoian coast. It is a significant bridge in Greece and one of the largest of its kind in Europe. The bridge has a total length of 694.50m and a usable width of 12.60m. The bridge comprises approach spans, with lengths of 4x35.875m (Voiotia) and 4x39.00m (Evoia). These spans are constructed using precast prestressed double-T beams and cast-in-place prestressed traffic deck slabs. The central section of the bridge has a length of 395.00m, featuring a main span of 215.00m and two side spans of 90.00m each. The main span allows for the passage of large vessels to and from Chalkida, and it has a clearance height of 35.51m. The bridge deck is made of reinforced concrete and is suspended by steel cables from two towers located approximately 90 m above sea level. The towers are founded on different types of rock: basaltic conglomerate for one tower (M5, Voiotian) and Cretan limestone for the other (M6, Evoian). The towers consist of twin-box steel frames and have heights of 88.35m (M5) and 84.26m (M6). The foundation of the towers is constructed using a pile cap system with Φ120 piles. The central section of the bridge is supported by piers M4 and M7, and it is suspended from the tower heads M5 and M6 using suspension cables. The supports on the piers M4 and M7 allow sliding in the longitudinal direction, while they are hinged in the transverse direction. To suspend the bridge deck, two levels of cables are used, converging at the heads of the towers. The bridge deck is designed to accommodate the flow of traffic and provide a vital transportation link between the Voiotian and Evoian coasts.

2.2 Deck

The deck is formed as a 0.45m thick slab, which remains constant throughout the length of the bridge, except around the M5 and M6 towers, where it increases to 0.75m. The slab is prestressed in the transverse direction along the entire length of the bridge. Longitudinal prestressing is applied only locally in the middle section of the bridge and in the transitional pier regions. In the longitudinal direction, the slab functions as an infinitely long cable strip, while in the transverse direction, it acts as a two-way beam. The concrete used for the deck is prestressed with St 1670/1860 strands. Translation and rephrasing: The bridge deck consists of a uniformly thick slab measuring 0.45m, except for the area around towers M5 and M6, where it increases to 0.75m. The slab is prestressed in the transverse direction across the entire length of the bridge. Localized longitudinal prestressing is applied in the middle section of the bridge and at the transitional piers. In the longitudinal direction, the slab acts as an infinitely long cable strip, while in the transverse direction, it functions as a two-way beam. The deck concrete is prestressed using St 1670/1860 strands.

2.3 Cables of the bridge

The bridge employs V.S.L type cables composed of parallel wire strands with a nominal diameter of 0.6 inches. These cables belong to the St 1670/1860 category and have a lead paint coating. The cables terminating at the M4 and M7 piers (return cables) are securely anchored. The number of wire strands in each cable is determined based on its tension. The cable cross-sections are variable and increase as the cables extend further from the towers, ranging between 9.8cm2 and 28.0cm2 (return cables). All cables are designed to be reusable and replaceable. The cable sag is controlled using a hollow head at the pier, allowing for a maximum displacement of 80mm. To ensure cable protection, they are encased in polyethylene tubes filled with cement grouting, which are additionally coated with plastic for UV radiation resistance.

2.4 Pillar of the bridge

The bridge piers essentially consist of reinforced concrete twin-column frames of B45 category. Their dimensions vary from (4.00x4.00x0.50)m at the base to (2.50x2.50x0.40)m at the top. The reinforcement ratio of the piers is below 2%. The two columns of each pier are connected to each other by two paired diaphragms, one below the deck and one below the heads. The head of the pier features an internal steel structure that receives the loads from the cables and handles their horizontal components, simultaneously transferring the vertical loads to the surrounding external concrete jacket.

2.5 Basement of the bridge

The foundation of the piers was constructed using 20 bored piles with a diameter of Φ120, interconnected at their heads with a 5.00m high pile cap. The piles are founded on the basaltic conglomerate towards the mainland side (Pier M5) and the Cretan limestone on the other side (Pier M6), with lengths of 30m and 15m respectively. The lateral action of the side openings of the central bridge on the piers M4 and M7 is achieved using articulated restrainers. Due to the negative reaction of the bridge on the piers, the piers are vertically prestressed and anchored with small dowels.

2.6 What is the purpose of the analysis

The primary objective of the analysis is to develop a MATLAB code that retrieves data from the bridge's accelerometers, validates the data, and subsequently filters and plots time histories. Following this, the code transforms the data into the frequency domain and generates power spectral density (PSD) diagrams versus frequency. Finally, the modal parameters of the bridge, including natural frequencies, mode shapes, and damping ratios, are determined using Stochastic Subspace Identification (SSI) method. These parameters are crucial for gaining insights into the dynamic characteristics of the bridge.

We utilized ambient data collected in the time domain from a range of sensors installed on the bridge. This raw data was processed using advanced computational techniques to identify these crucial modal parameters.

Specifically, Welch's method was employed first. This method, a form of spectral estimation, helped us identify the dominant frequencies present in the recorded vibrations, thereby indicating the bridge's natural frequencies.

Subsequently, we applied the Subspace System Identification method. The findings, including natural frequencies, mode shapes, and damping ratios, derived from this comprehensive analysis are presented in the subsequent sections.

2.7 Sensors of the Bridge

The bridge monitoring system consists of 36 accelerometers, with nine of them located in three triaxial instruments and the remaining ones in uniaxial instruments. Currently, 16 uniaxial sensors situated on the superstructure of the bridge and one triaxial sensor positioned at the base of the pier (on the ground level) are operational. The triaxial sensor was specifically installed on the side facing Voiotia. The non-functionality of 17 instruments can be attributed to vandalism and cable theft incidents that have occurred in recent years on the bridge, emphasizing the need for high-resolution monitoring and recording devices both on the lower part of the bridge and the superstructure, capturing the movement of pedestrians and vehicles passing through. Monitoring is deemed necessary due to the vital importance of understanding the type and size of passing vehicles for correlation with the recorded data.

Out of the six uniaxial sensors operating along the longitudinal direction ("L") of the deck, two are located on the top of the Voiotia pier, and the other two are positioned on the top of the Evoia pier. Two of the sensors along the transverse direction ("T") are situated at the junction between the Evoia pier and the bridge deck, with one on each side. Additionally, there is a seventh sensor in the transverse direction, located in the triaxial sensor on the ground plate on the Voiotia pier side. Two out of the six sensors functioning in the lateral direction ("T") are placed at the tops of the two bridge piers. The other two lateral sensors are located at the connection point between the two piers and the bridge deck. There is one more lateral sensor in the middle of the deck, and the sixth lateral sensor is positioned on the deck plate, near the extreme joint on the Evoia side. Furthermore, there is an eighth sensor in the lateral direction, situated in the triaxial sensor on the ground plate on the Voiotia pier side.

The three vertical sensors consist of one located at the extreme opening of the bridge towards the Evoia side and two in the central opening of the bridge. Additionally, there is a fourth sensor in the vertical direction, present in the triaxial sensor on the ground plate on the Voiotia pier side.

In table1 we can see the number of sensors working and having reliable data and the direction they are. Also, in the diagram of the bridge the exact location of them can be found in figure 2.

Channel	Working	Direction	Position	X	Y
1	YES	Vertical	Middle Span, deck level	197.5	40
2	NO	Transverse	Middle Span, deck level	N/A	N/A
3		N/A	N/A	N/A	N/A
4	YES	Longitudinal	Pier 1, Top level (East)	305	90
5	YES	Longitudinal	Pier 2, Top level (West)	90	90
6	YES	Longitudinal	Pier 2, Top level (East)	90	90
7	YES	Transverse	Left Span, deck level (West)	0	40
8	YES	Transverse	Pier 1, deck level (West)	305	40
9	YES	Vertical	Right Span, deck level (West)	365	40
10	YES	Transverse	Pier 2, Top level (West)	90	90
11	YES	Longitudinal	Pier 1, deck level (West)	305	40
12	YES	Transverse	Pier 2, deck level (West)	90	40
13	NO	Transverse	Right Span, deck level (East)	N/A	N/A
14		N/A	N/A	N/A	N/A
15		N/A	N/A	N/A	N/A
16		N/A	N/A	N/A	N/A
17		N/A	N/A	N/A	N/A
18	YES	Transverse	Pier 1, Top level (West)	305	90
19	YES	Longitudinal	Pier 1, deck level (East)	305	40
20	YES	Longitudinal	Pier 1, Top level (West)	305	90
21		N/A	N/A	N/A	N/A
22	YES	Vertical	Middle Span, deck level (East)	237.5	40
23		N/A	N/A	N/A	N/A
24		N/A	N/A	N/A	N/A
25	YES	Longitudinal	Pier 2, bottom level (East)	90	0
26	YES	Vertical	Pier 2, bottom level (East)	90	0
27	YES	Transverse	Pier 2, bottom level (East)	90	0
28		N/A	N/A	N/A	N/A
29		N/A	N/A	N/A	N/A
30		N/A	N/A	N/A	N/A
31		N/A	N/A	N/A	N/A
32		N/A	N/A	N/A	N/A
33		N/A	N/A	N/A	N/A
34		N/A	N/A	N/A	N/A
35		N/A	N/A	N/A	N/A
36		N/A	N/A	N/A	N/A

Table 1: Table with reliable sensors (pier 1 is on Evoia side)

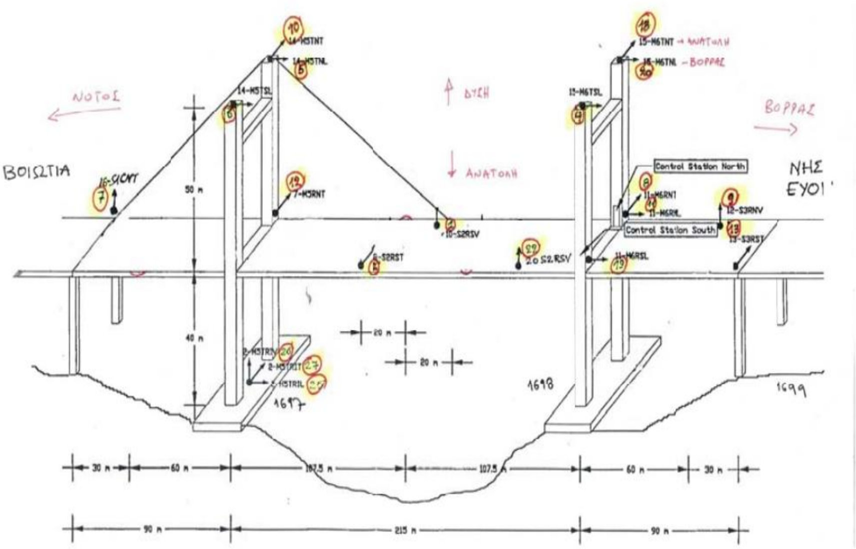

Figure 2: Location of the sensors in the bridge

3 Presentation of recordings and their processing

3.1 Diagrams in time domain

As mentioned earlier, the monitoring system collects recordings of the bridge's response to environmental, seismic, and traffic actions continuously. Modal properties of the bridge are derived from these excitations. In figures 3, 4 and 5, the oscillation waveforms from the sensors in the longitudinal, transverse, and vertical directions are presented.

Longitudinal Direction: The waveforms depict the oscillations that occur along the length of the bridge. These waveforms represent the motion of the bridge along its deck.

Transverse Direction: The waveforms illustrate the oscillations that occur perpendicular to the deck of the bridge along the horizontal direction. These waveforms represent the lateral motion of the bridge.

Vertical Direction: The waveforms show the oscillations that occur perpendicular to the deck of the bridge along the horizontal direction.

These waveforms provide information about the dynamic behavior of the bridge during oscillations.

The measurements from the sensors are saved in a file named with the date of the recordings. This file contains a txt format file for each sensor which includes time acceleration velocity and distance. A MATLAB code was developed to open the file and create a matrix with accelerations of all sensors and another one with time and then plot the time-domain graphs for each axis. In figures 3, 4 and 5 we can see the x-axis representing time in seconds and the y-axis representing acceleration in g, providing an initial visualization of the data, and allowing us to identify which recordings are damaged and which are not.

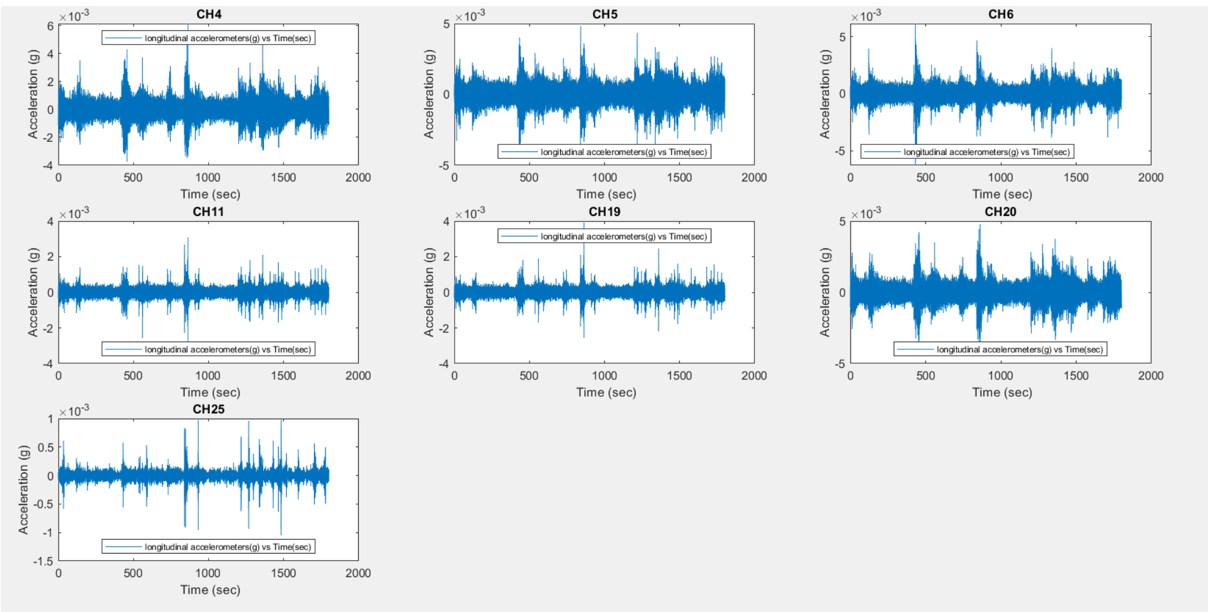

Figure 3: Longitudinal acceleration vs time

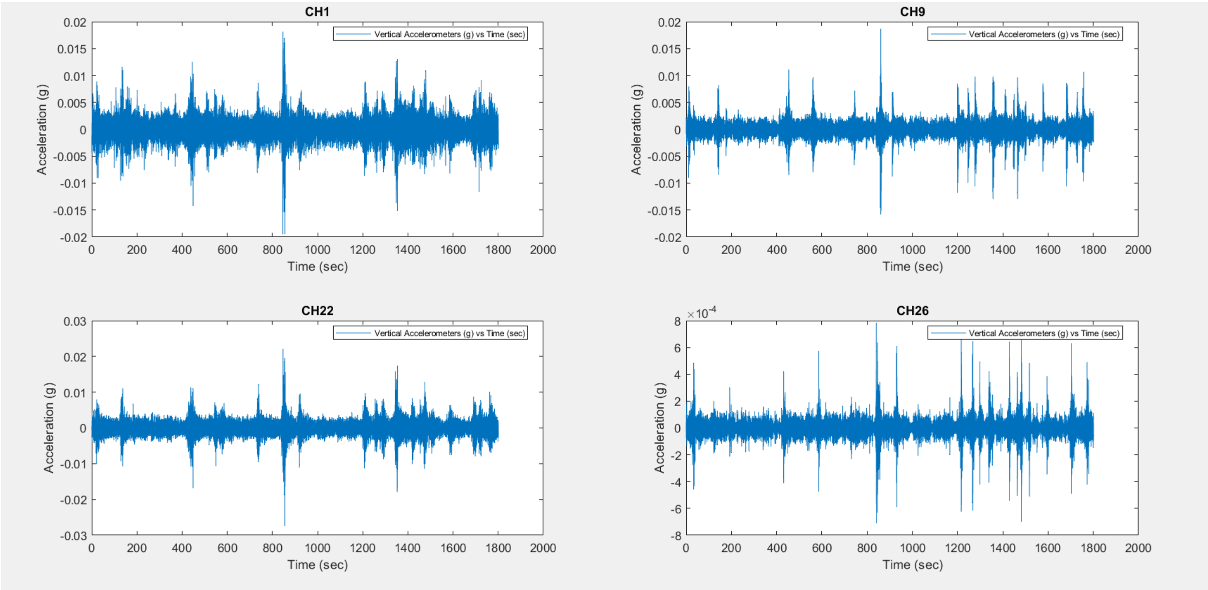

Figure 4: Vertical acceleration vs time

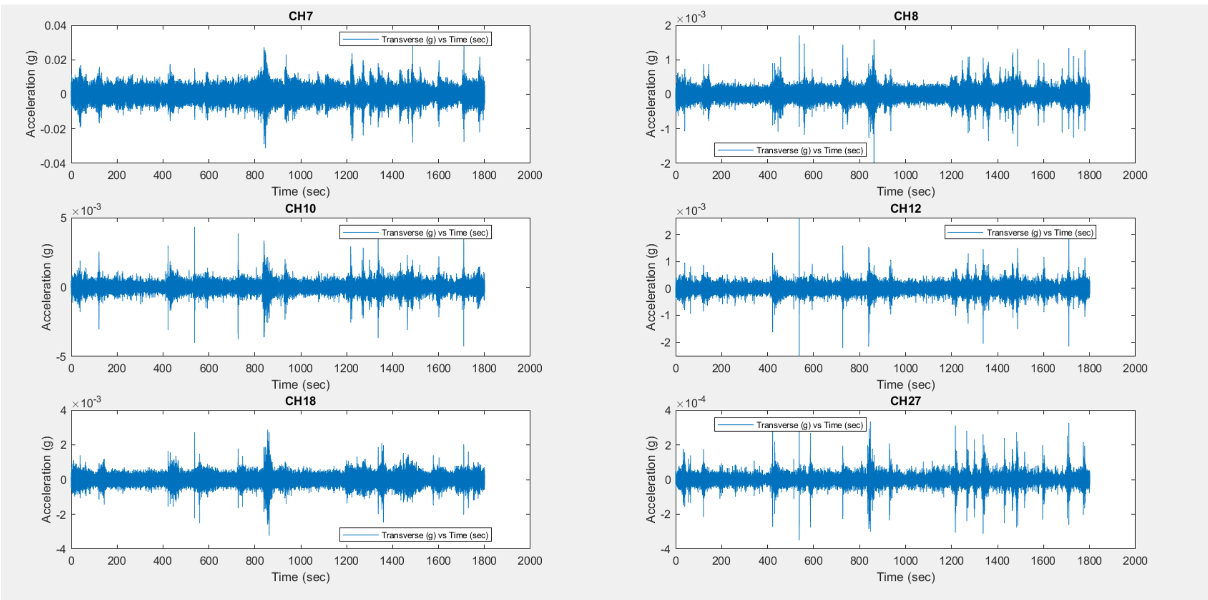

Figure 5: Transverse acceleration vs time

As expected, points on the deck of the bridge exhibit higher accelerations, while points on the bridge piers show lower accelerations. The points at the base of the bridge pier clearly have smaller values, which is logical since they are located at the foundation of the bridge.

In figure 6, time histories of accelerometers with damaged data are shown. These data are not included in the analysis.

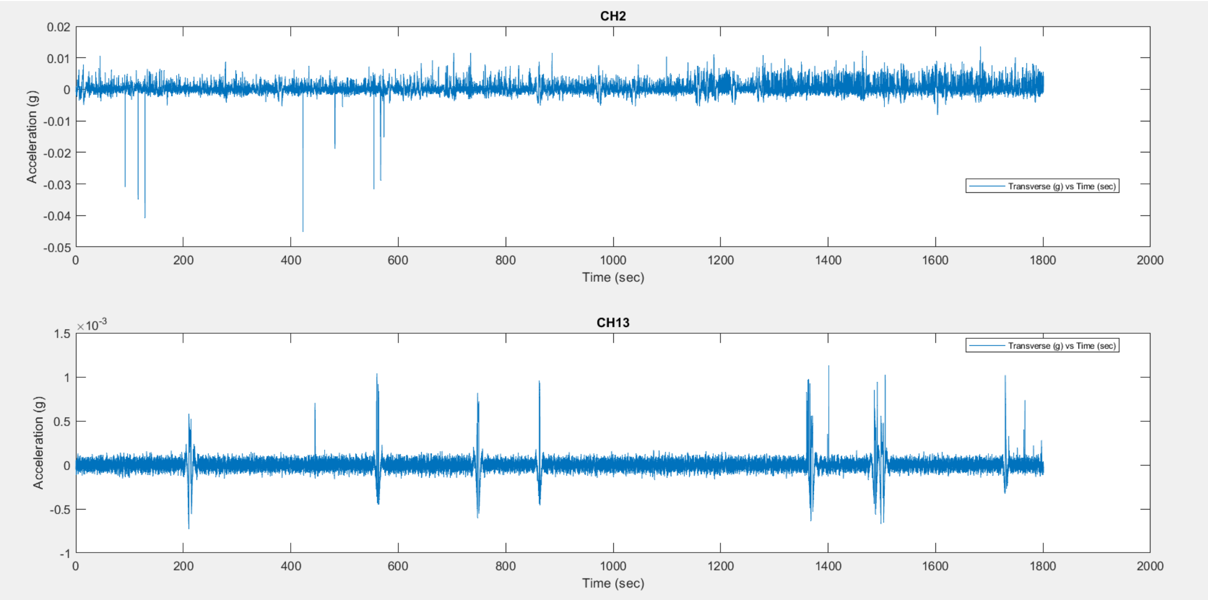

Figure 6: Time histories of damaged data

3.2 Diagrams in frequency domain

3.2.1 Welch's method

Next, the objective is to determine the natural frequencies at which the bridge undergoes oscillation. Given that the available data corresponds to ambient conditions, a rigorous approach is required to analyze the frequency content of the signals. To achieve this, the Welch's method, developed by Peter Welch, was employed as a spectral estimation technique, which is widely recognized and utilized in the fields of signal processing and data analysis.

The Welch's method [3] offers an effective means to transition from the time domain representation of signals to the frequency domain representation. This transformation enables a detailed examination of the frequency components present in the signal. One notable advantage of Welch's method is its ability to alleviate the effects of spectral leakage that may arise when analyzing signals with finite duration.

The mitigation of spectral leakage is accomplished by dividing the input signal into overlapping segments and subsequently applying a windowing function. This process enhances the accuracy of the power spectrum estimation by reducing the spurious spectral leakage artifacts. By segmenting the signal and employing windowing, Welch's method provides a more precise estimation of the power spectrum, allowing for accurate identification of the natural frequencies associated with the bridge oscillation. Overlap refers to the overlap between consecutive segments of the data used for calculating the Fourier transform. This overlap is typically specified as a percentage or a fraction of the segment length. During the application of Welch's method, the data is divided into overlapping segments, and the Fourier transform is computed for each segment. The segments are typically windowed to minimize spectral leakage. By overlapping the segments, the statistical properties of the estimated power spectral density can be improved. The amount of overlap chosen affects the trade-off between frequency resolution and variance reduction in the estimated power spectral density. Higher overlap values, such as 50% or 0.5, indicate a significant overlap between consecutive segments, resulting in better frequency resolution at the expense of increased computational cost. Lower overlap values, such as 25% or 0.25, reduce the overlap between segments, leading to lower frequency resolution but with reduced computational cost.

The utilization of Welch's method facilitates the transformation of the time-domain signals to the frequency domain, enabling the investigation and identification of the natural frequencies at which the bridge oscillates. This technique is particularly advantageous due to its capability to mitigate spectral leakage effects, thereby improving the accuracy of the power spectrum estimation.

Given a matrix with the accelerations from the sensors and their corresponding recordings, the power spectral density (PSD) can be calculated using Welch's method in MATLAB. By estimating the sampling frequency from the frequency of each recording and setting the overlap to 50%, the PSD matrix is generated.

The application of Welch's method in the frequency domain allows for a comprehensive analysis of signal characteristics. The distinct peaks and patterns observed in the transformed signal reveal the dominant frequencies present in the original signal. These peaks serve as key indicators of the underlying behavior and dynamics of the signal.

Analyzing the magnitude and location of these peaks provides crucial information about the signal's characteristics. Higher peak magnitudes indicate the presence of stronger or more significant frequency components within the signal. This insight helps identify important frequency modes or phenomena that contribute prominently to the overall signal behavior.

The Power Spectral Density (PSD) plot serves as a visual representation of the power distribution across different frequencies in the signal. Peaks observed in the PSD plot correspond to specific frequency components exhibiting higher power or energy levels compared to neighboring frequencies. Analyzing the characteristics of these peaks, including their frequency, magnitude, and shape, allows for insights into the signal's behavior, vibrations, or disturbances affecting the system.

The analysis of peaks in the PSD plot enables a detailed exploration of the frequency content and spectral characteristics of the signal. This knowledge finds applications in various fields, including signal processing, communications, and scientific research. It enables tasks such as signal analysis, noise identification, system diagnosis, and the extraction of valuable information from the signal for further processing and interpretation.

3.2.2 Power spectral density vs Time diagrams

In figures 7, 8 and 9 are presented the recordings in the frequency domain in vertical longitudinal and transverse direction. The x-axis represents the frequencies, while the y-axis represents the power spectral density. By examining the magnitudes and locations of these peaks, important information can be gleaned about the behavior and characteristics of the signal. The height of each peak corresponds to the strength or amplitude of the corresponding frequency component. Peaks with higher magnitudes indicate more pronounced or significant frequency content. This provides an overview of the eigenfrequencies of the bridge.

The peaks observed in the following diagrams indicate that the bridge is oscillating at certain natural frequencies, primarily between 0-5 Hz.

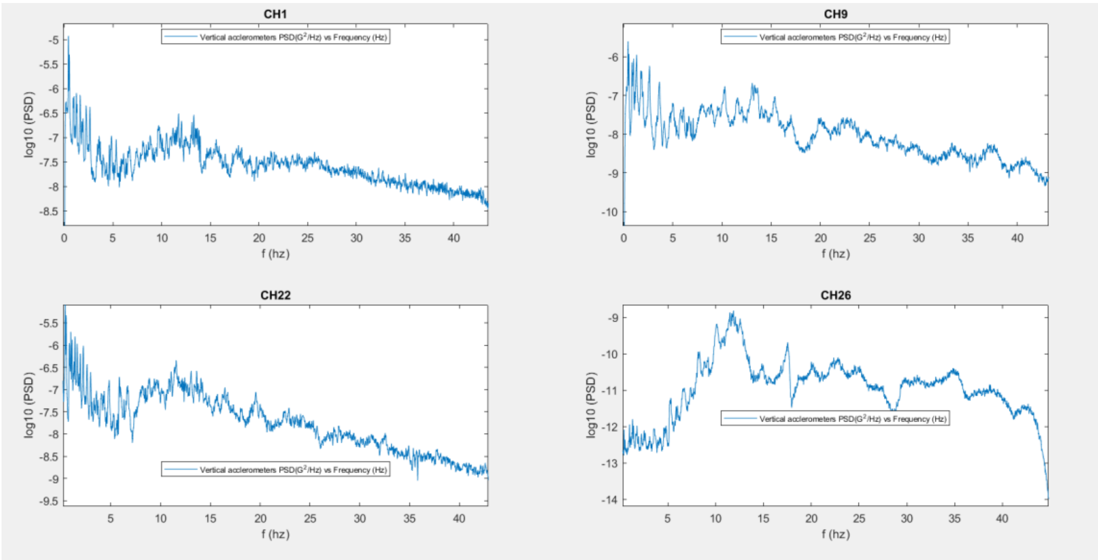

Figure 7: Vertical PSD vs Frequency

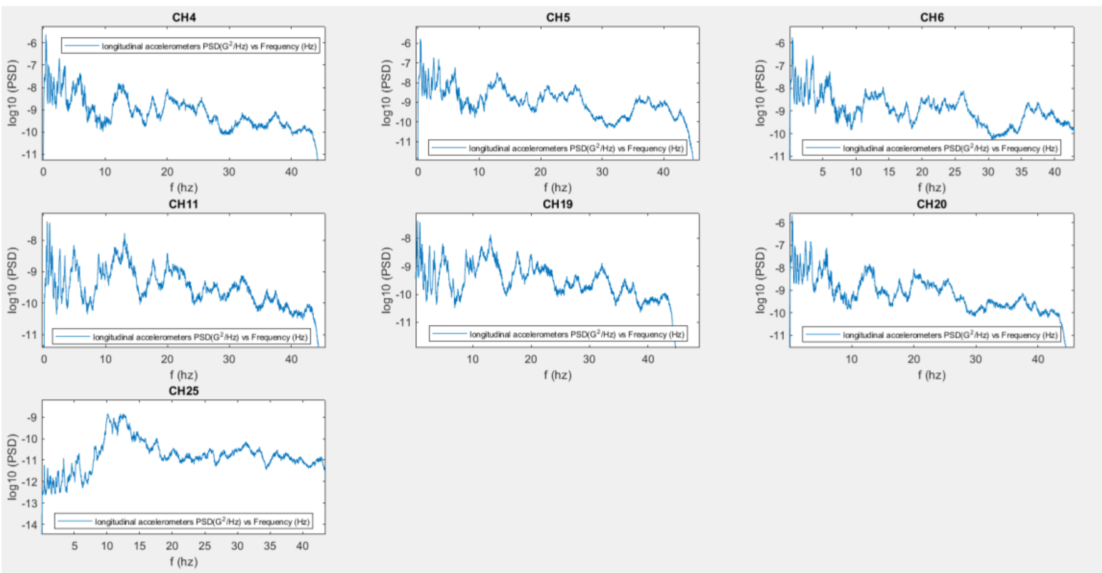

Figure 8: Longitudinal PSD vs Frequency

Figure 9: Transverse PSD vs Frequency

Additionally, the code has the capability to zoom in on the frequency domain as needed to enhance the visibility of the peaks and, consequently, the eigenfrequencies. This zooming functionality allows for a clearer depiction of the frequency peaks, enabling better identification of the eigenfrequencies. From the diagrams in figures 10, 11 and 12 it is obvious the frequency between 0 and 5 Hz are much more significant than the others as they have bigger peaks.

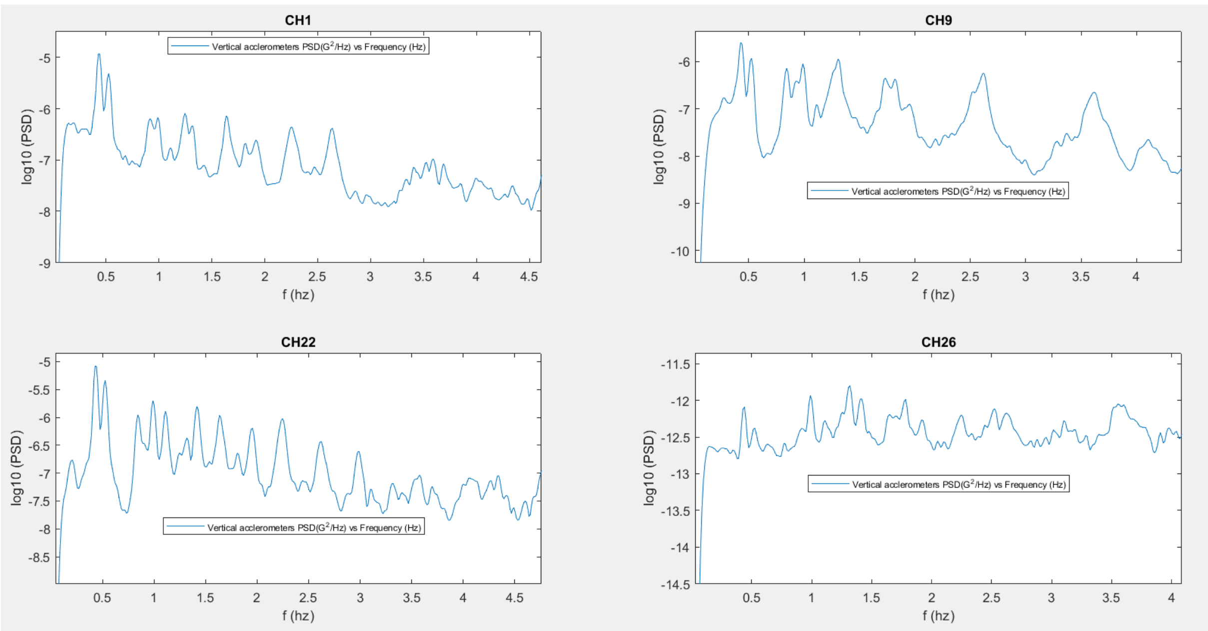

Figure 10: Vertical PSD vs Frequency 0-6 Hz

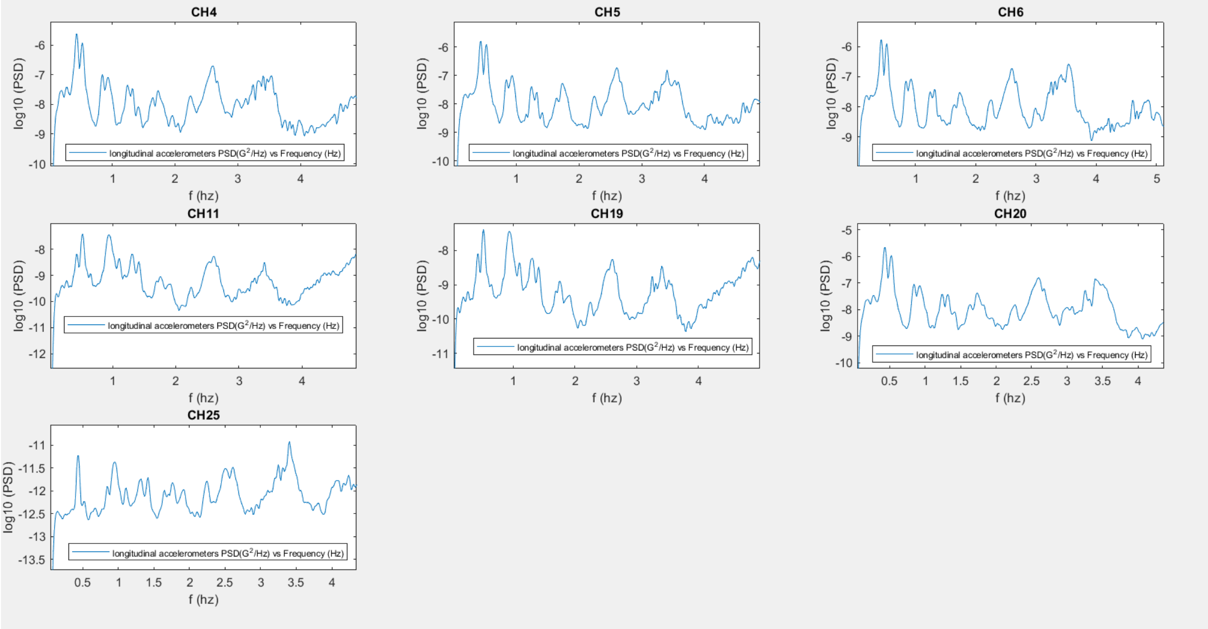

Figure 11: Longitudinal PSD vs Frequency 0-6 Hz

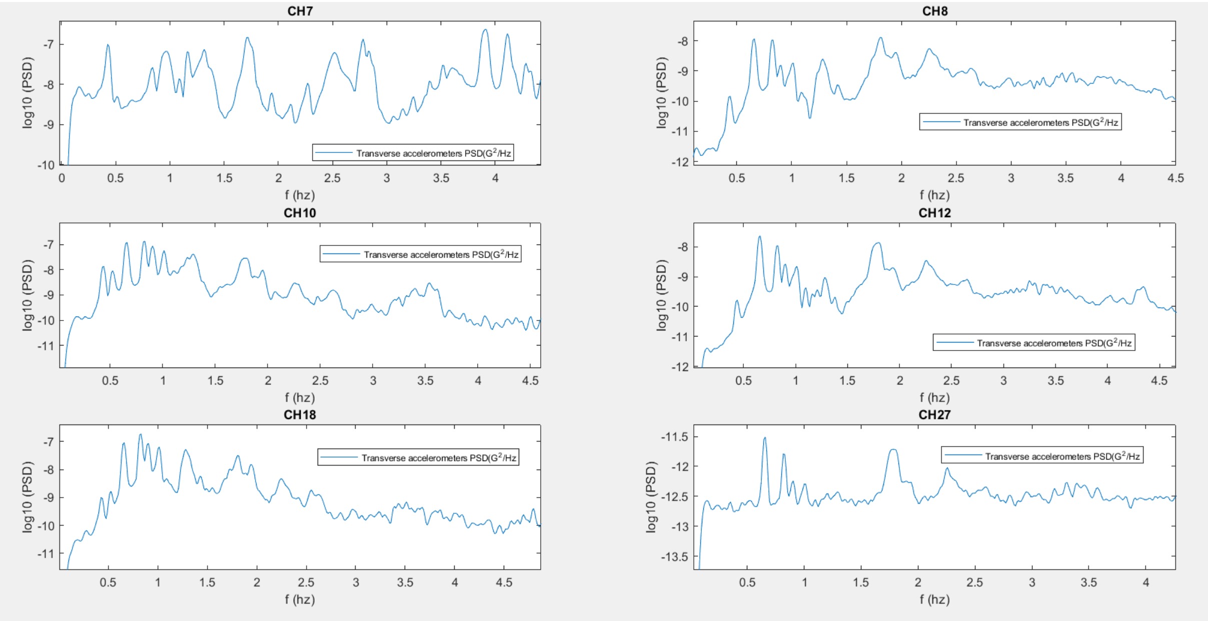

Figure12: Transverse PSD vs Frequency 0-6 Hz

In figure 13 PSD of sensors with damaged data are shown.

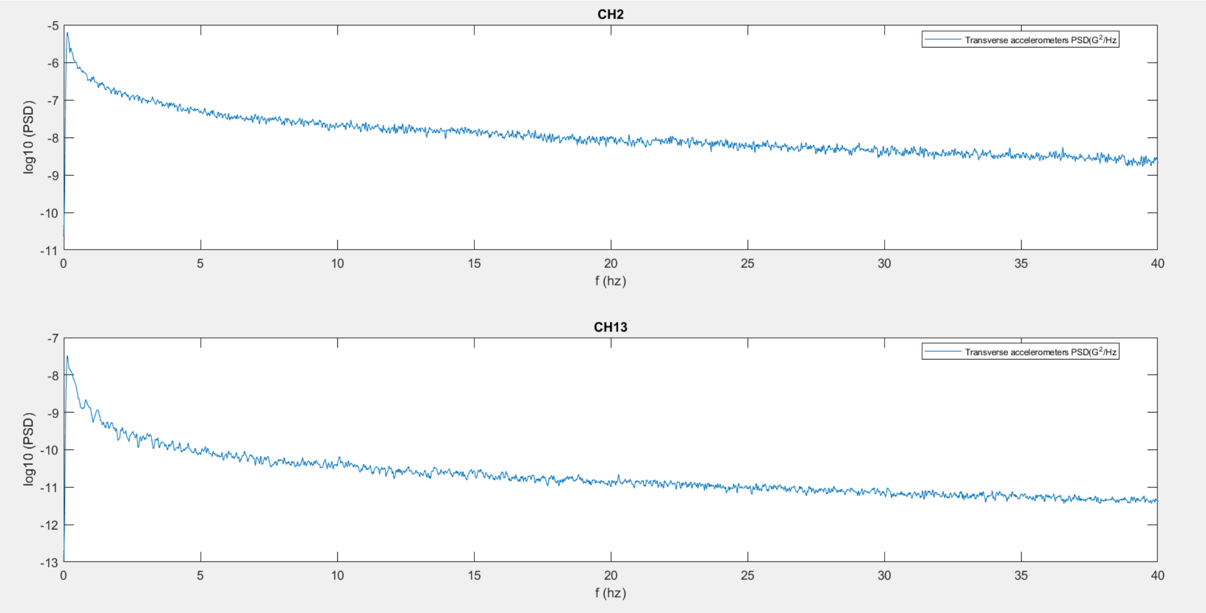

Figure 13:PSD vs Frequency of damaged data

3.3 Stabilization diagrams

In the field of vibration analysis, a stabilization diagram [4] is a graphical representation that aids in the understanding and interpretation of the dynamic behavior of structures or systems. It provides valuable insights into the stability and performance characteristics of the system under investigation.

A stabilization diagram can be represented using frequency (in Hz) on the horizontal axis and the number of poles on the vertical axis. The number of poles refers to the order or complexity of the mathematical model used to describe the system's dynamics.

By plotting the frequency and corresponding number of poles, the stabilization diagram provides a visual representation of the system's stability and response characteristics. Each point on the diagram represents a specific frequency range and the corresponding number of poles required to accurately capture the system's behavior at that frequency.

The stabilization diagram helps in understanding the relationship between frequency and the complexity of the model required to represent the system accurately. It enables the identification of critical frequencies where a higher number of poles may be necessary to capture complex dynamics or unstable modes of vibration.

Lines of poles that remain stable over a range of model orders can be identified as the resonance frequencies of the structure. These are the frequencies at which the structure naturally tends to vibrate.

By plotting all these frequencies in a stabilization diagram (with frequency on one axis and model order on the other), which frequencies are stable across multiple model orders can be visualized. This allows the differentiation of the true physical modes from the computational artifacts or noise. In the following figures there are the results of plotting these diagrams.

3.3.1 Stabilization diagrams without zoom

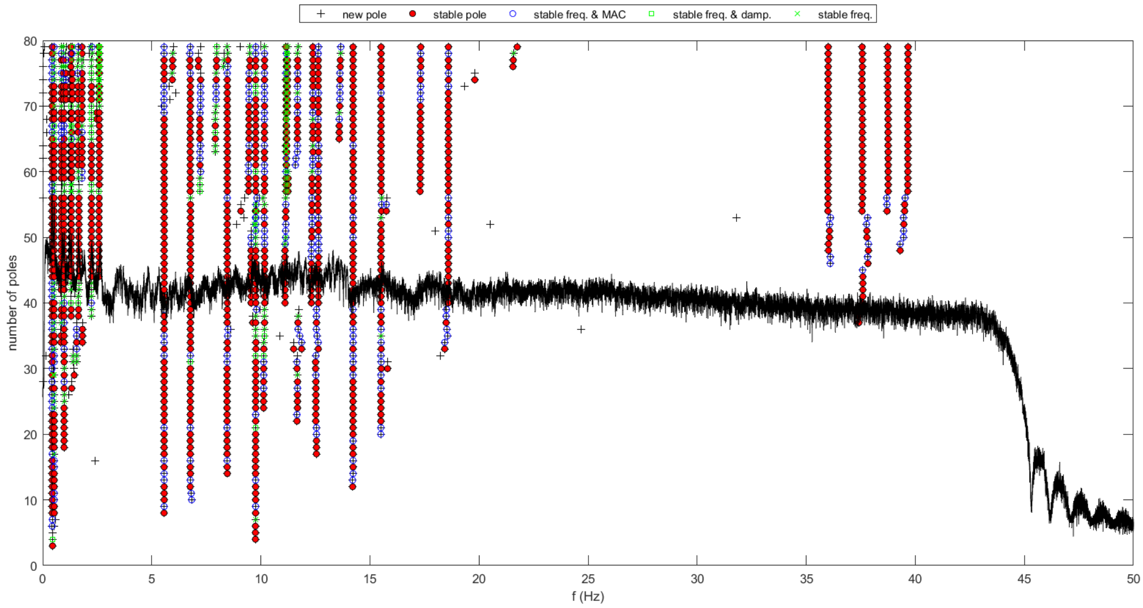

Figure 14: Stabilization diagram of 1st sensor

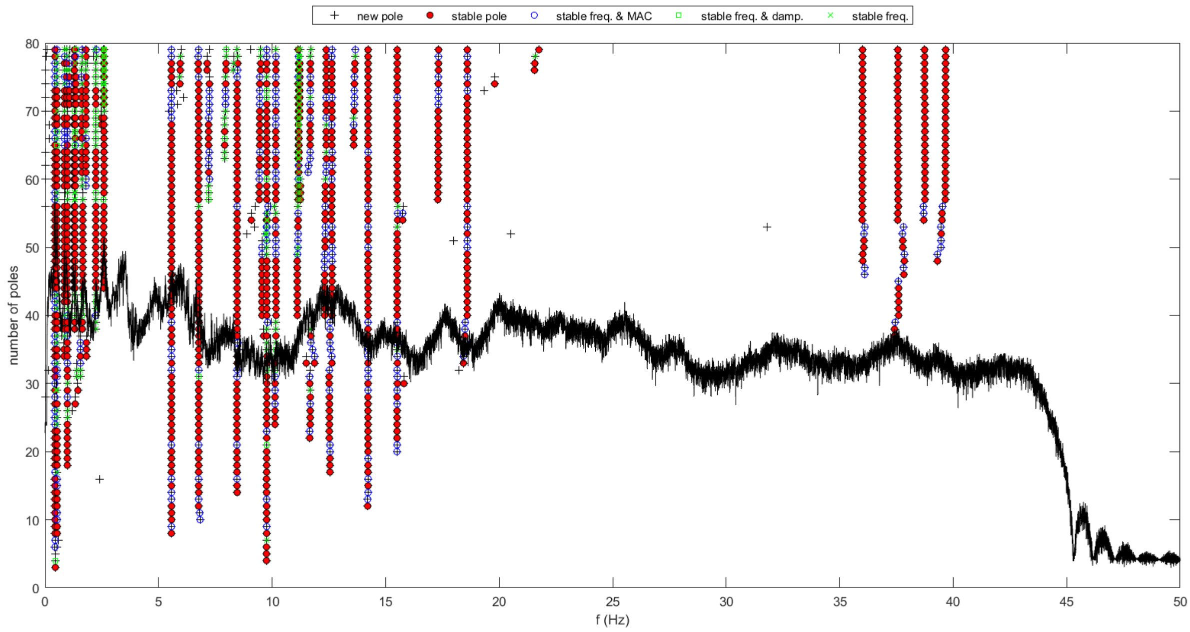

Figure 15: Stabilization diagram of 4th sensor

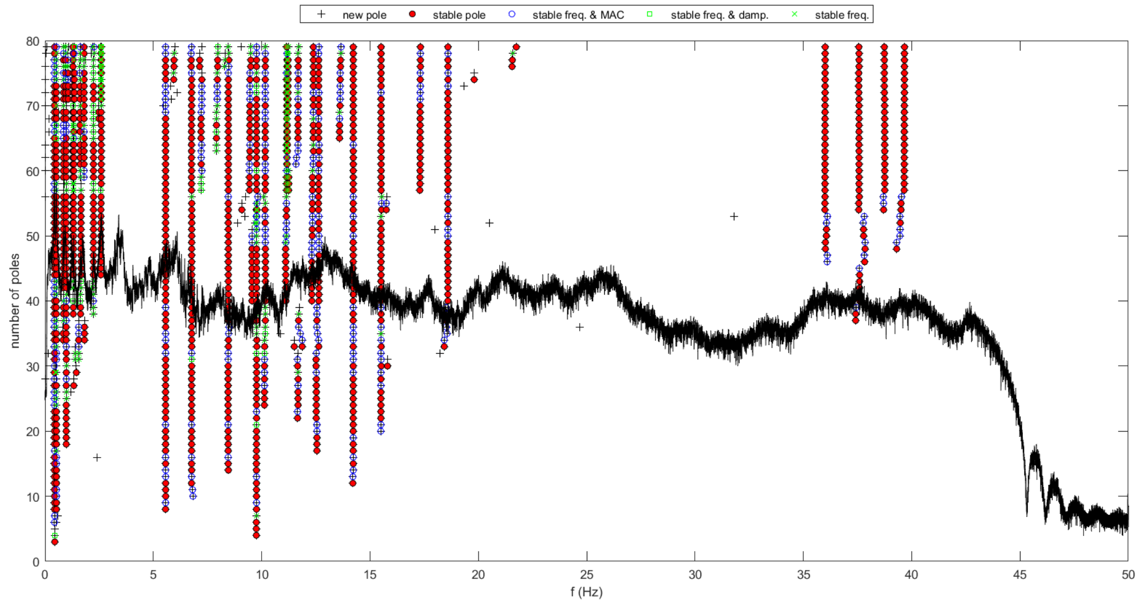

Figure 16: Stabilization diagram of 5th sensor

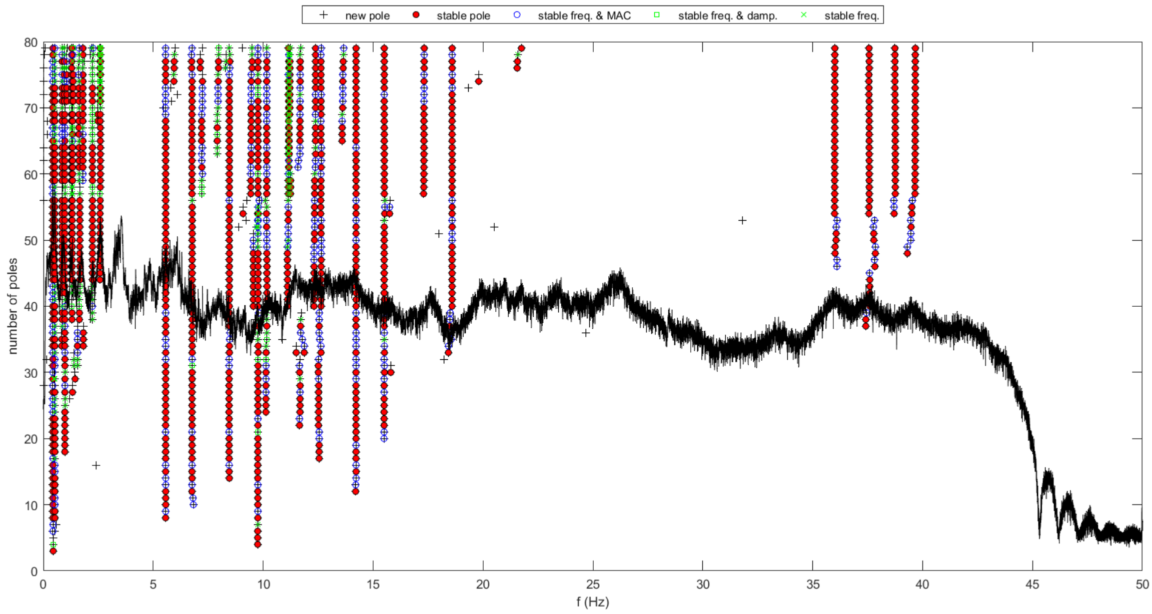

Figure 17: Stabilization diagram of 6th sensor

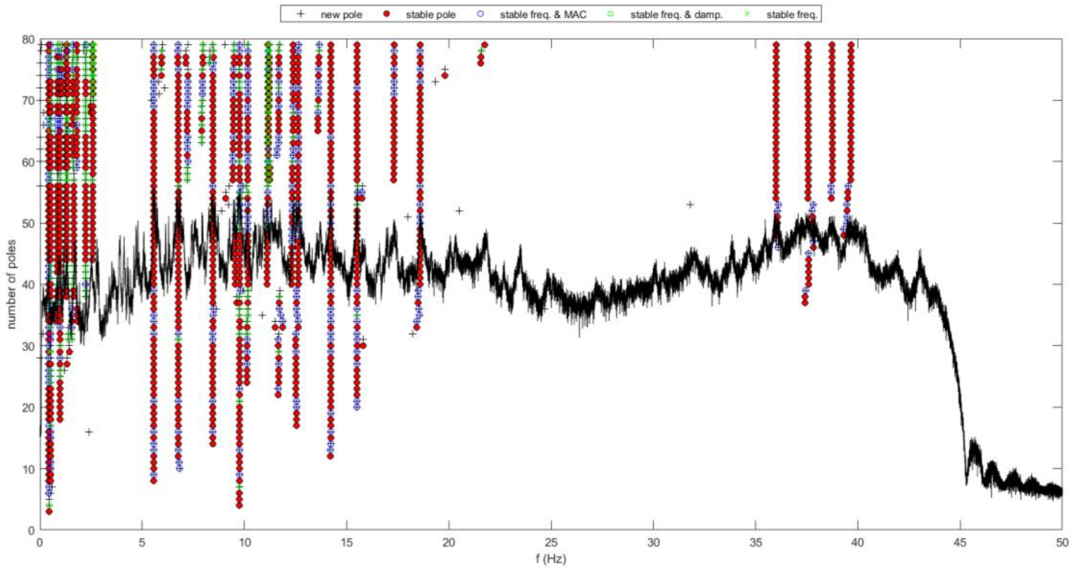

Figure 18: Stabilization diagram of 7th sensor

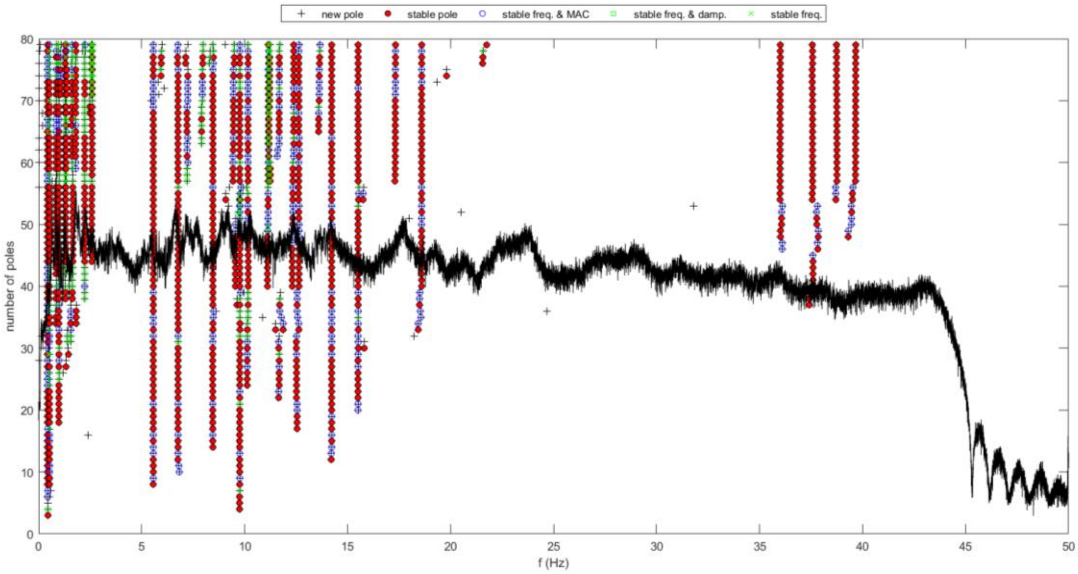

Figure 19: Stabilization diagram of 8th sensor

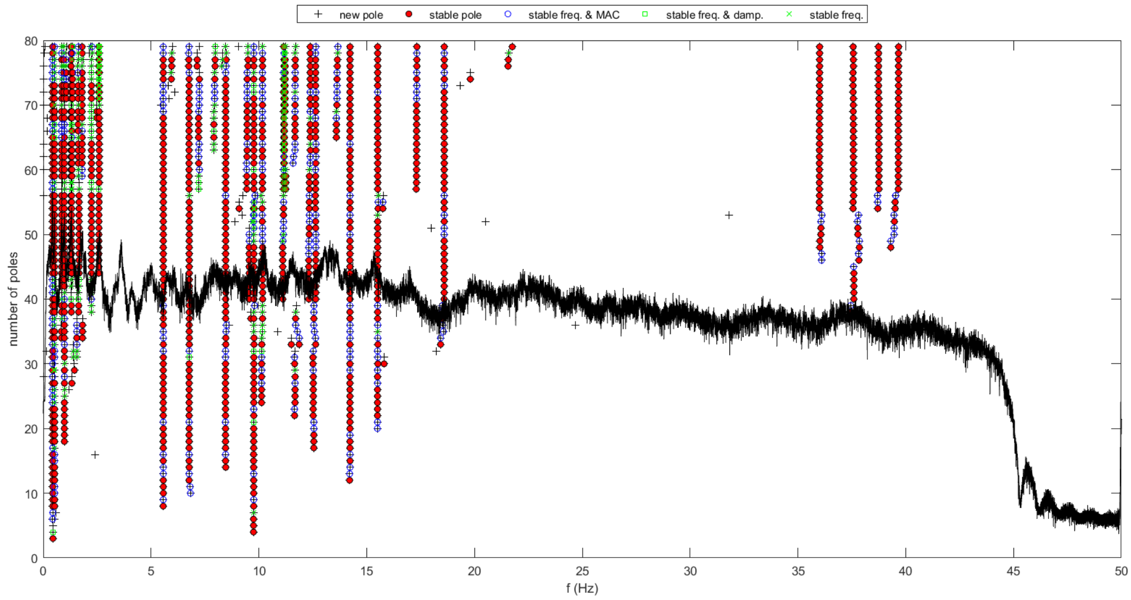

Figure 20: Stabilization diagram of 9th sensor

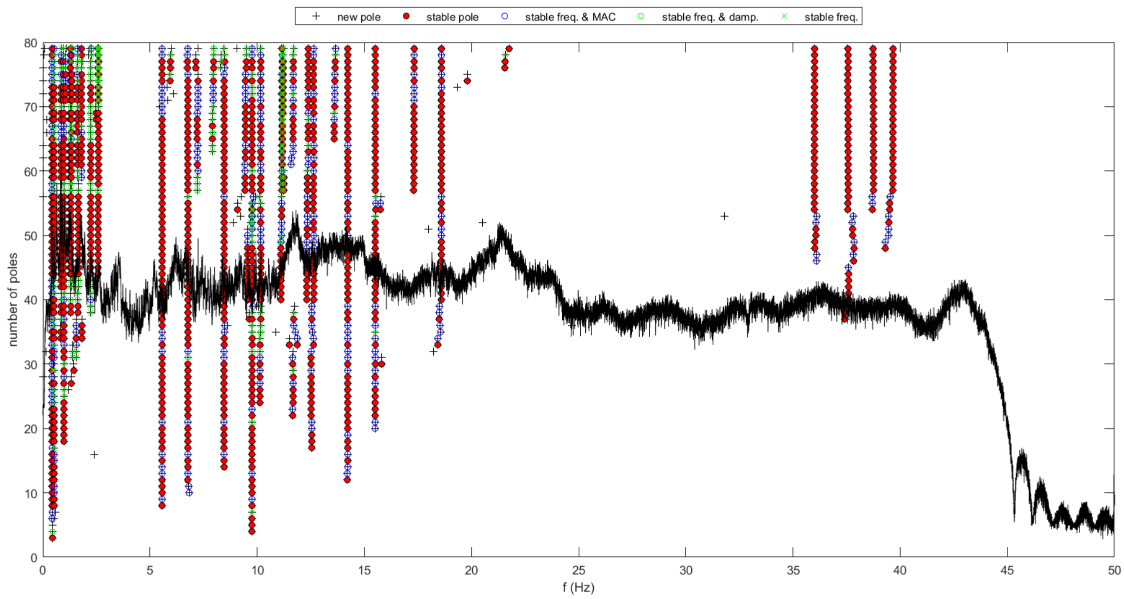

Figure 21: Stabilization diagram of 10th sensor

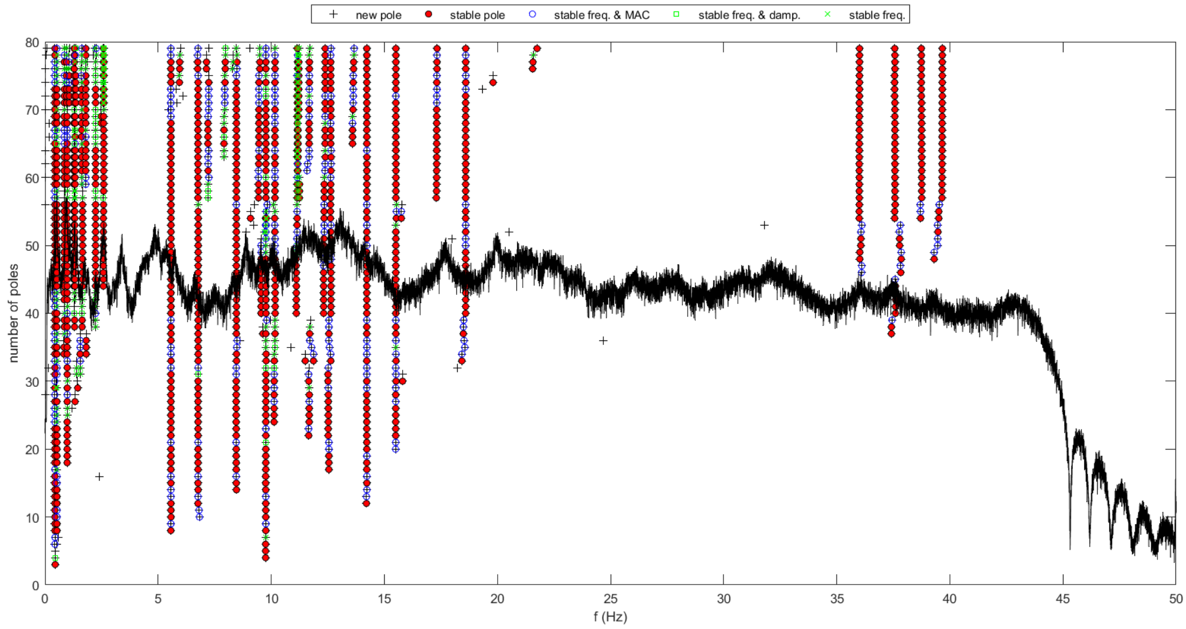

Figure 22: Stabilization diagram of 11th sensor

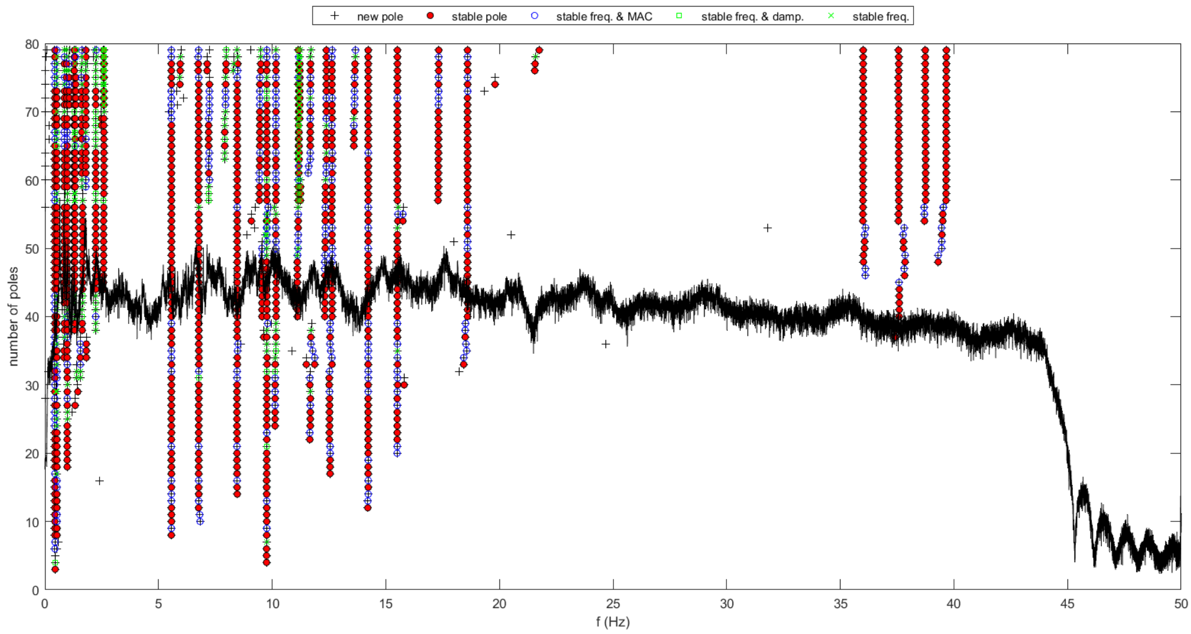

Figure 23: Stabilization diagram of 12th sensor

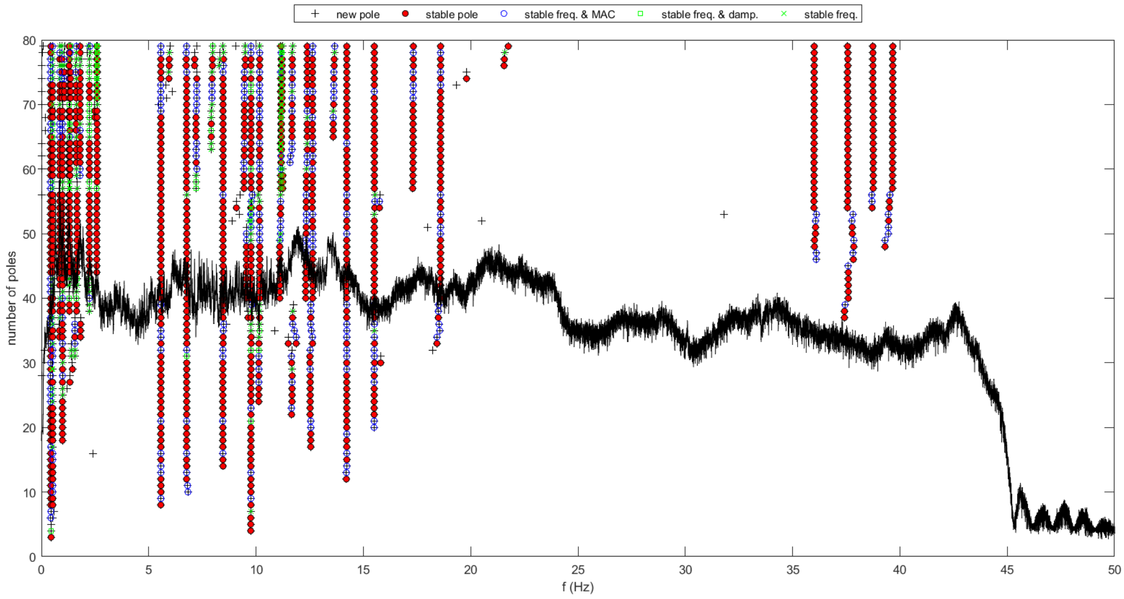

Figure 24: Stabilization diagram of 18th sensor

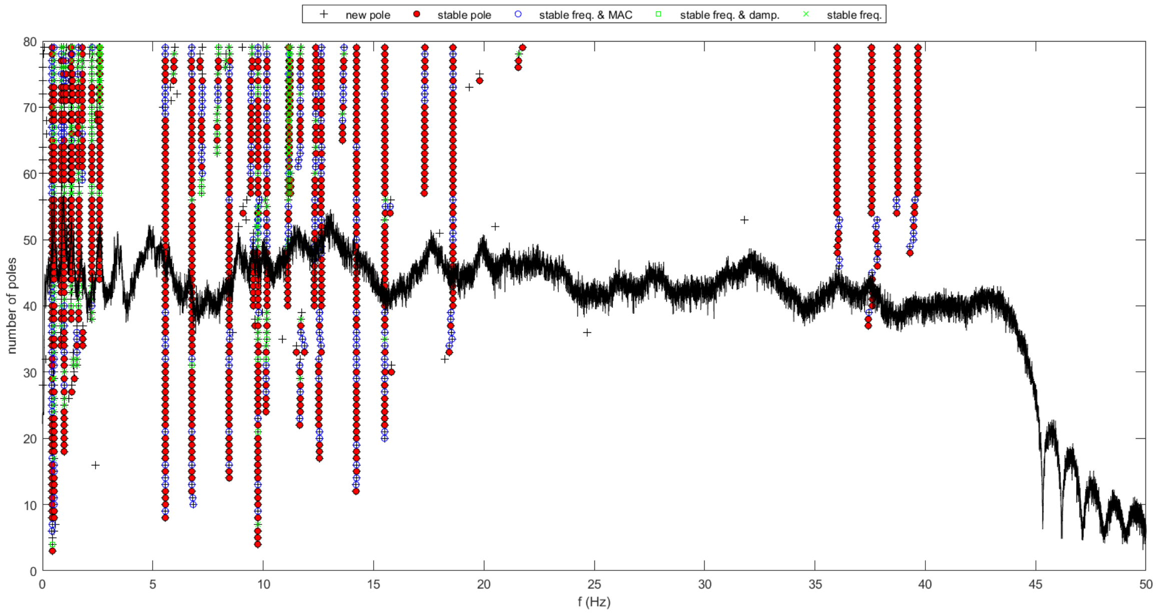

Figure 25: Stabilization diagram of 19th sensor

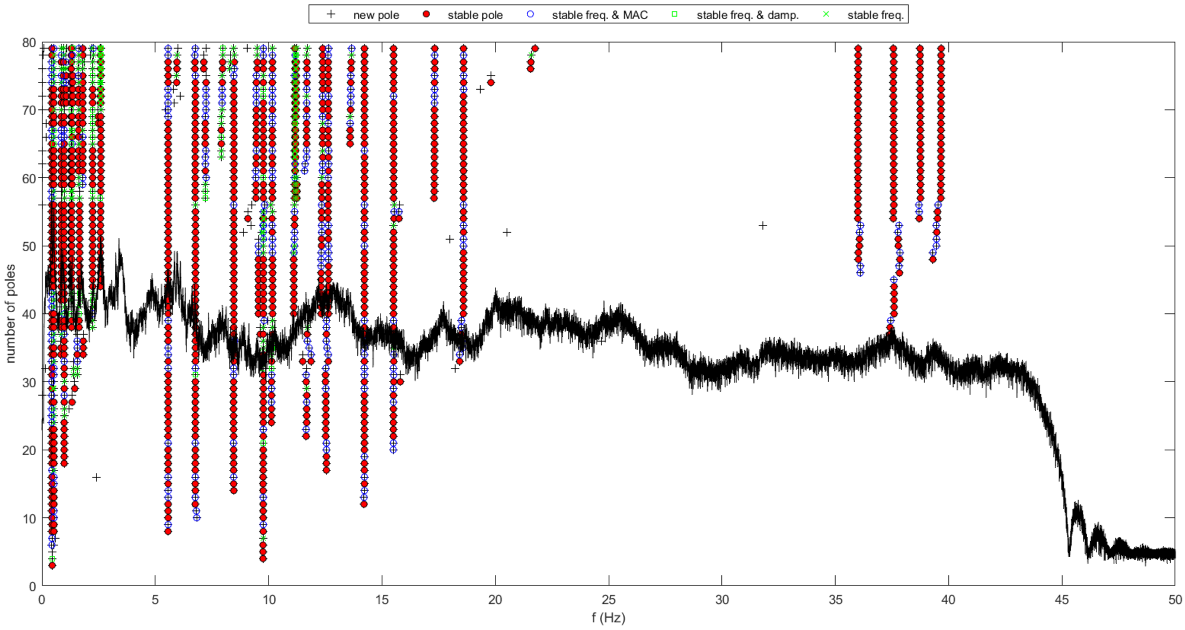

Figure 26: Stabilization diagram of 20th sensor

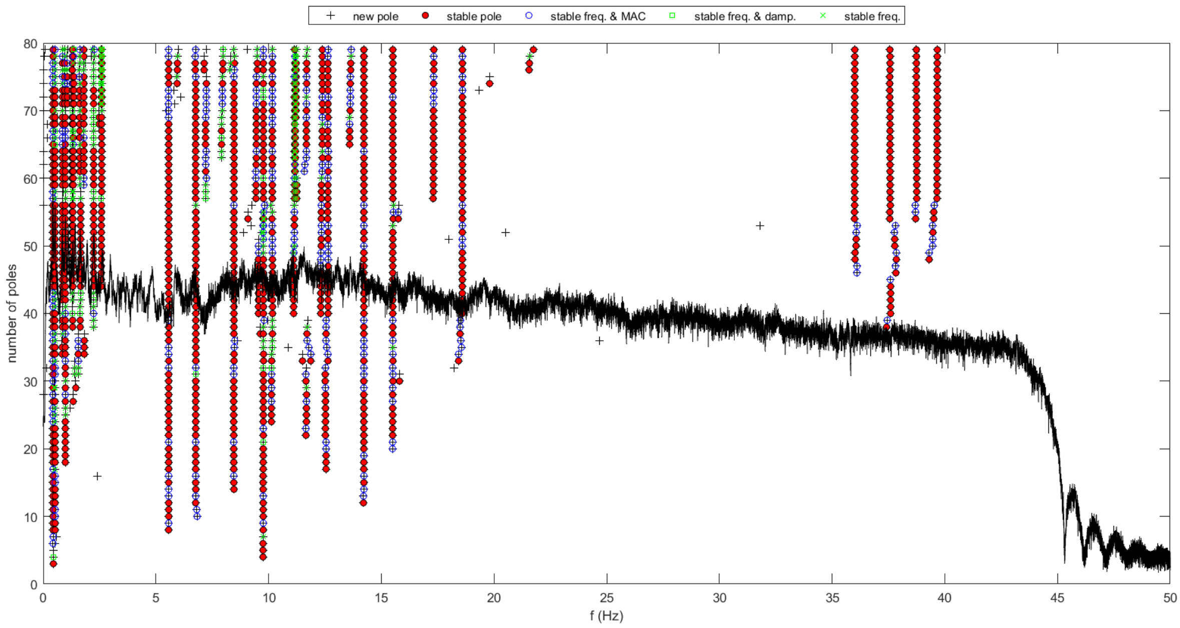

Figure 27: Stabilization diagram of 22nd sensor

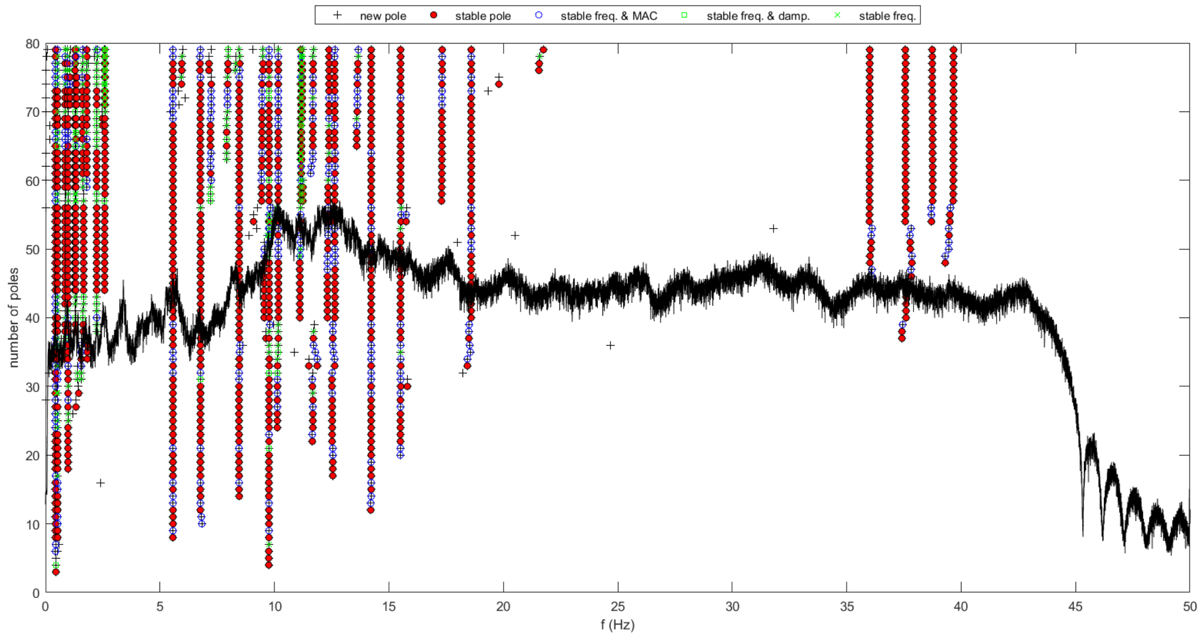

Figure 28: Stabilization diagram of 25th sensor

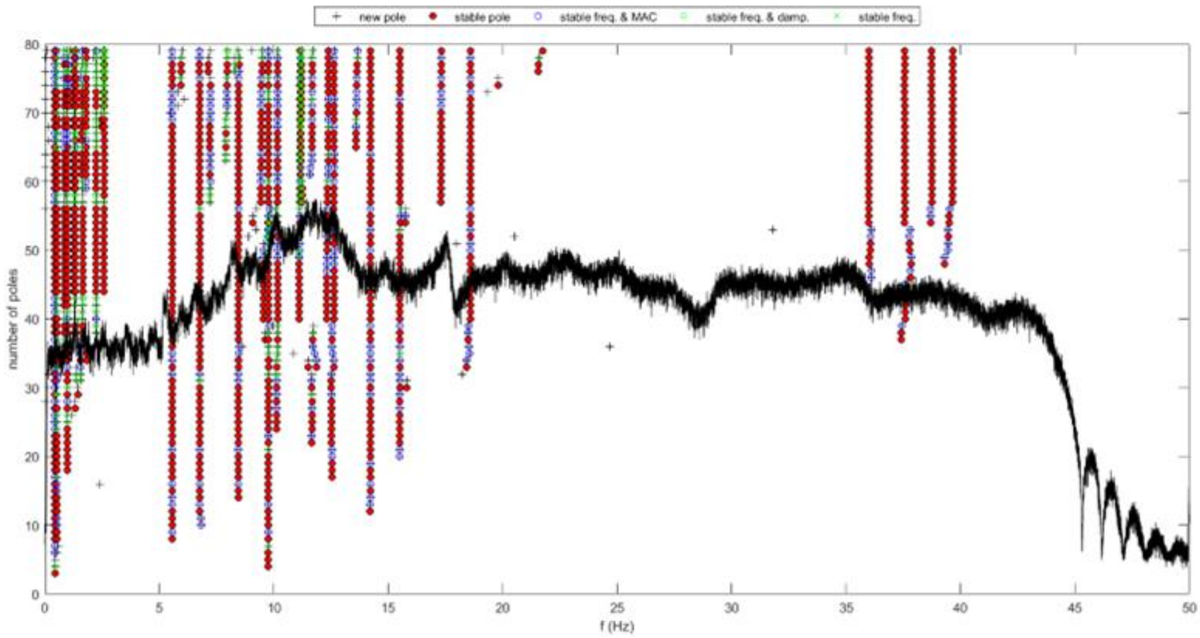

Figure 29: Stabilization diagram of 26th sensor

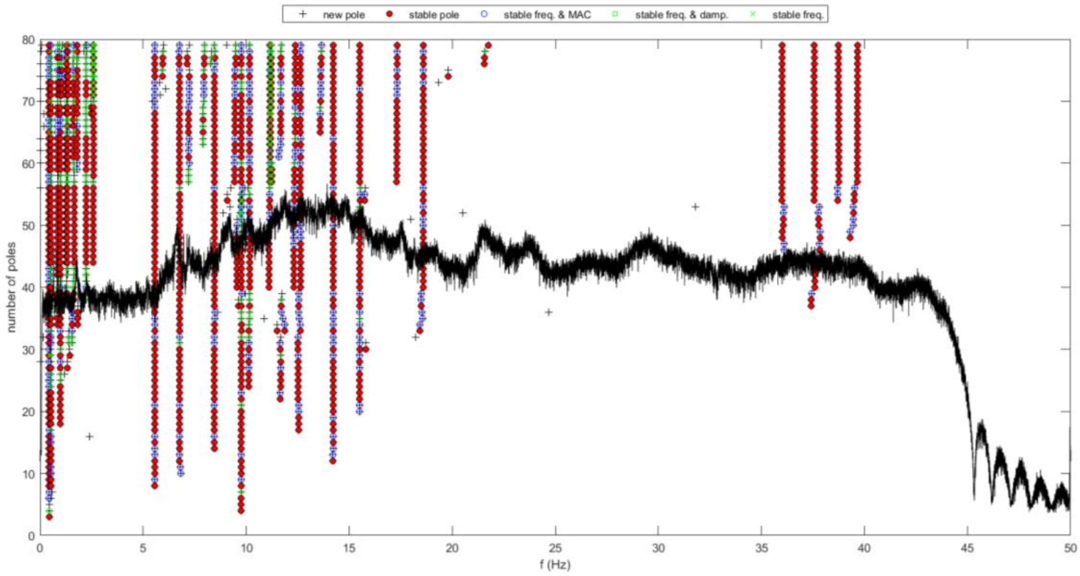

Figure 30: Stabilization diagram of 27thst sensor

3.3.2 Stabilization Diagram Between 0 and 5 Hz

In the case of stabilization diagrams, it is possible to zoom in on specific frequency ranges of interest. Since the eigenfrequencies range from 0 Hz to 5 Hz, zooming in on this frequency range allows for a closer examination of the eigenfrequencies and their corresponding characteristics. Although a zoom in a different region can be done through the MATLAB code.

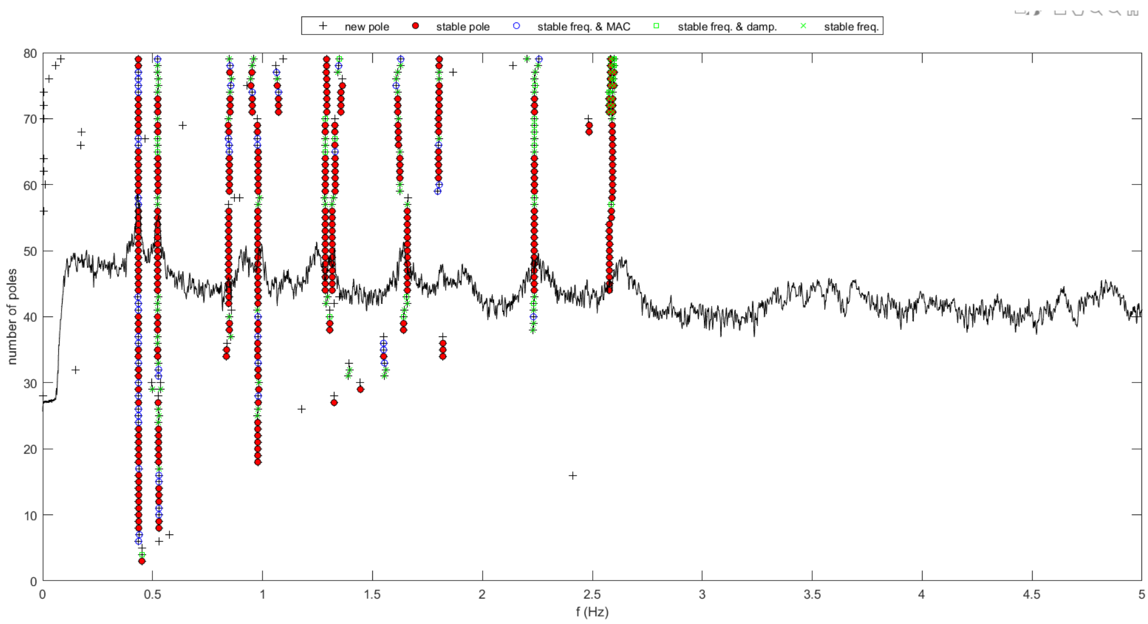

Figure 31: Stabilization diagram of 1st sensor

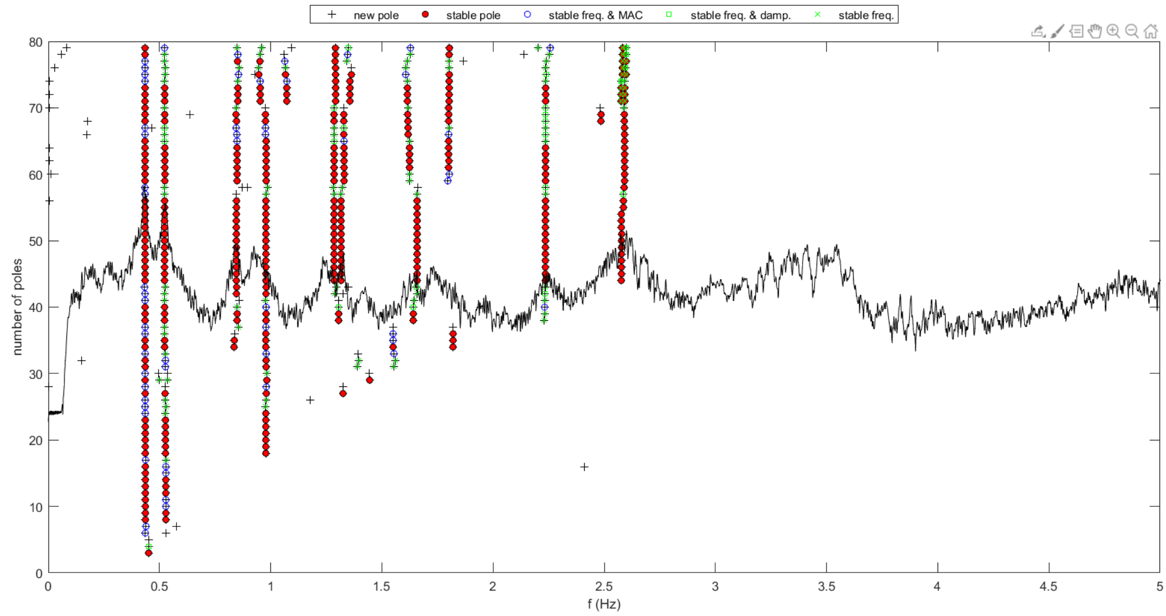

Figure 32: Stabilization diagram of 4th sensor

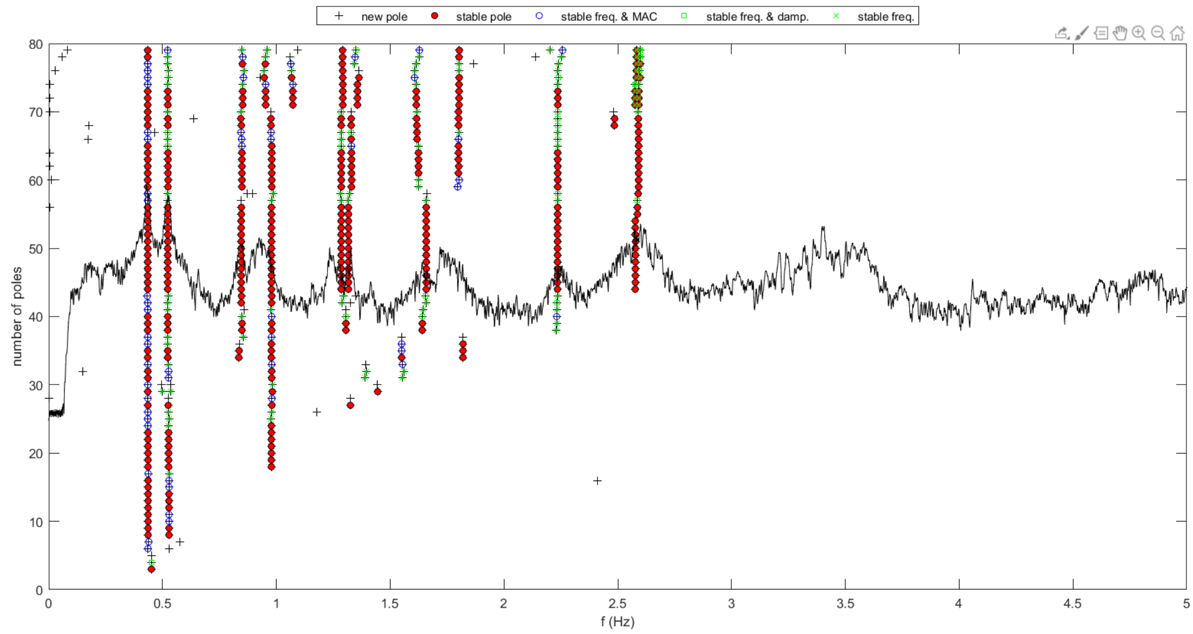

Figure 33: Stabilization diagram of 5th sensor

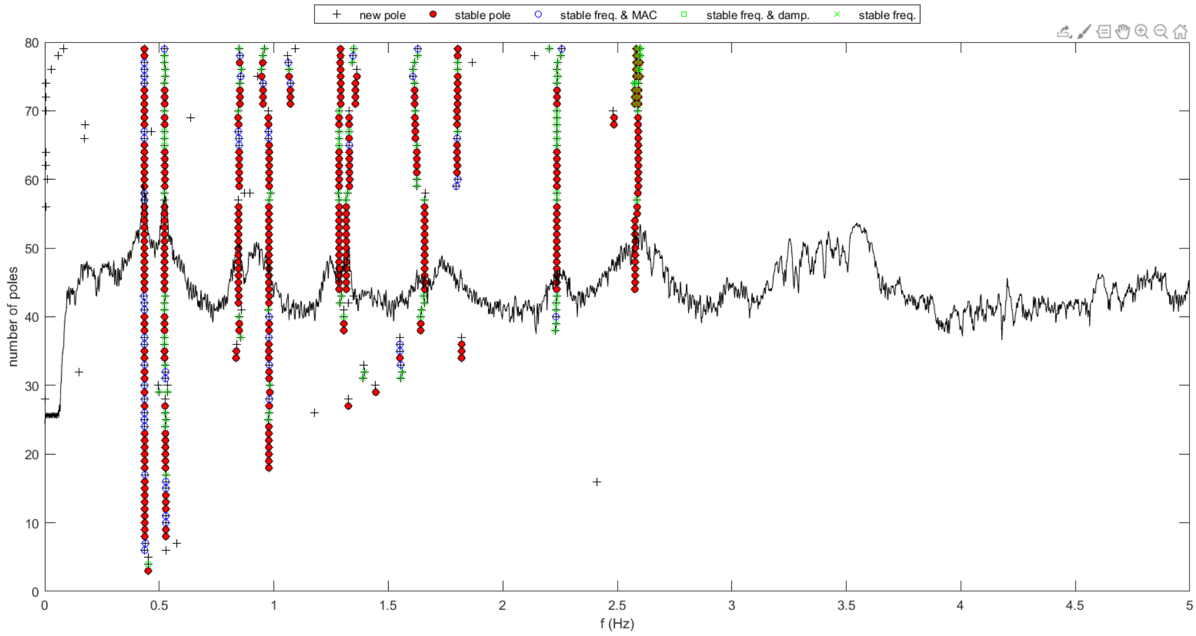

Figure 34: Stabilization diagram of 6th sensor

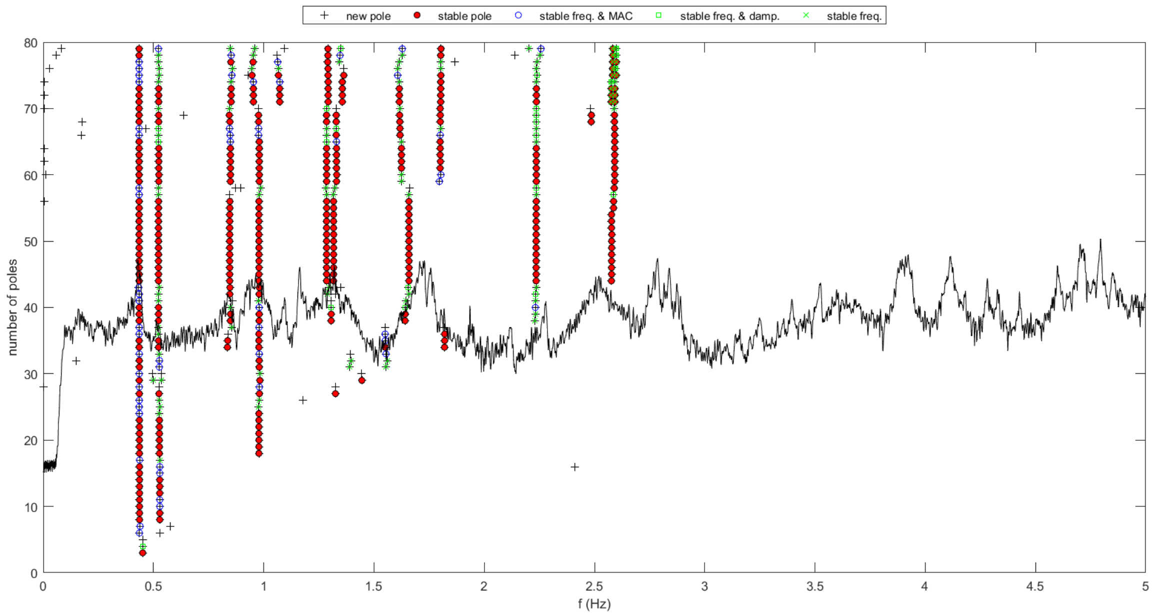

Figure 35: Stabilization diagram of 7th sensor

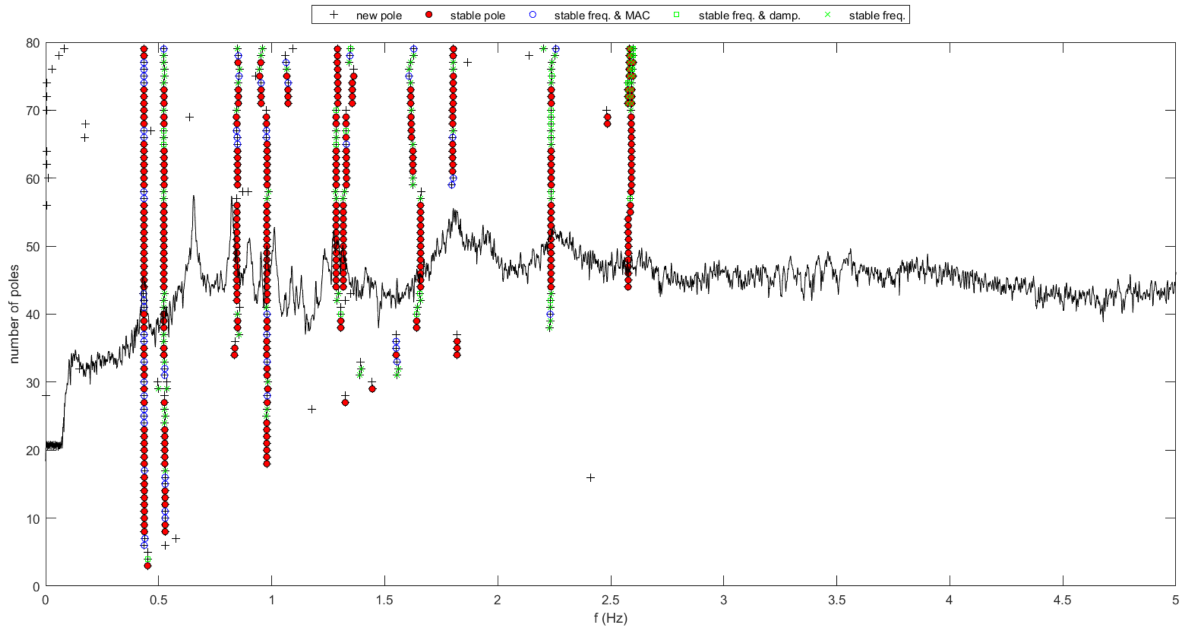

Figure 36: Stabilization diagram of 8th sensor

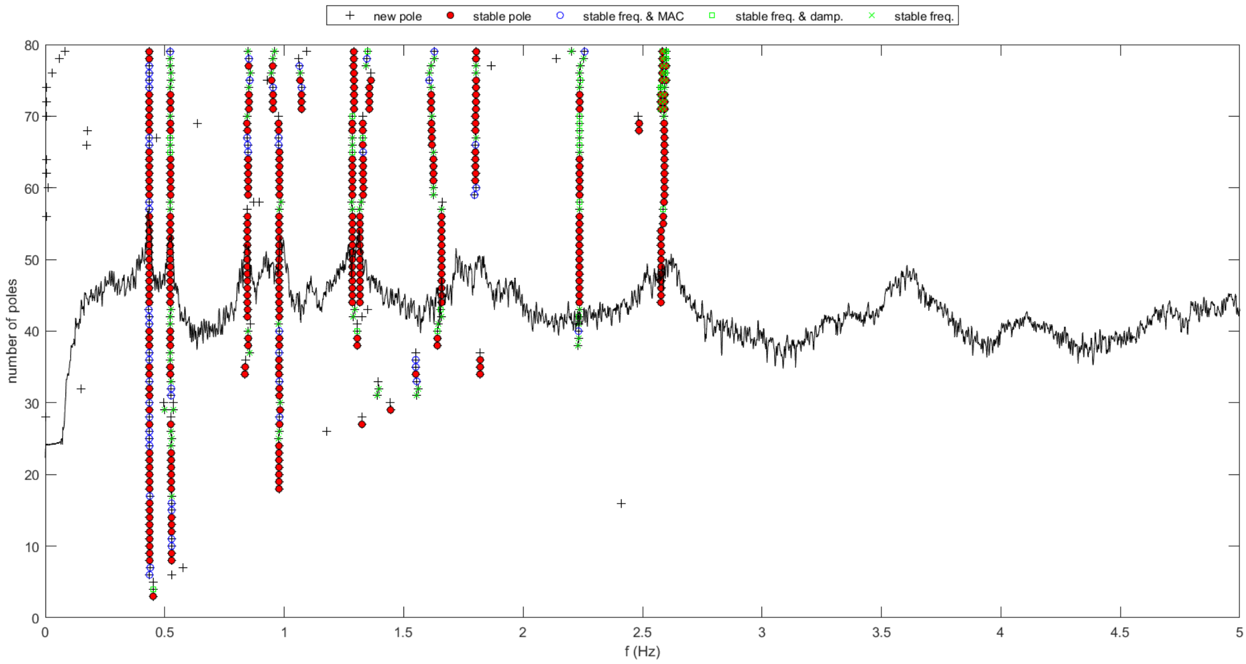

Figure 37: Stabilization diagram of 9th sensor

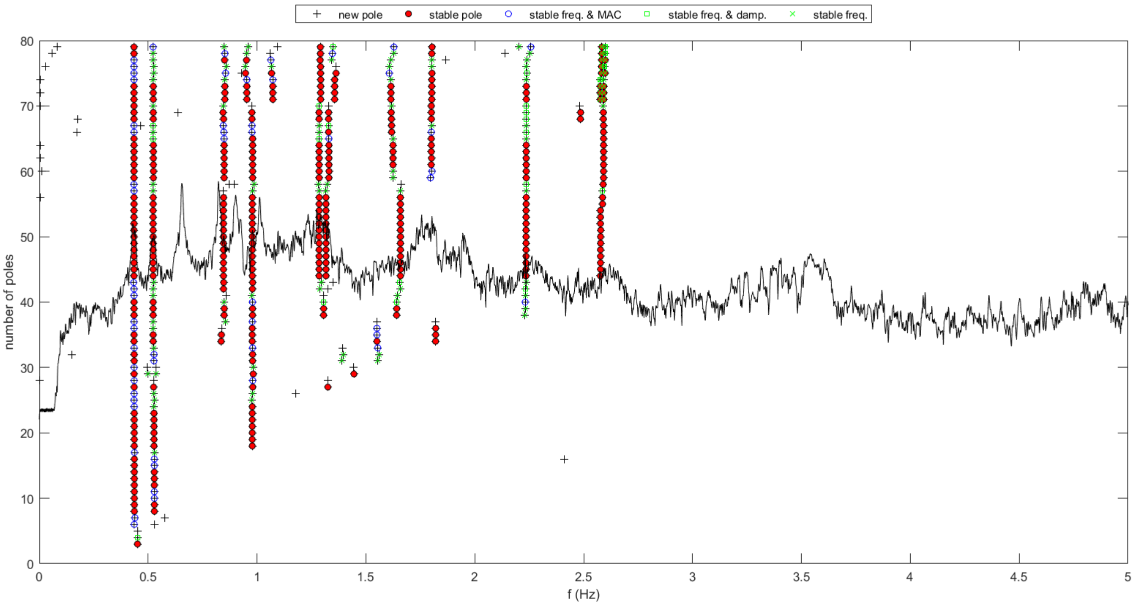

Figure 38: Stabilization diagram of 10th sensor

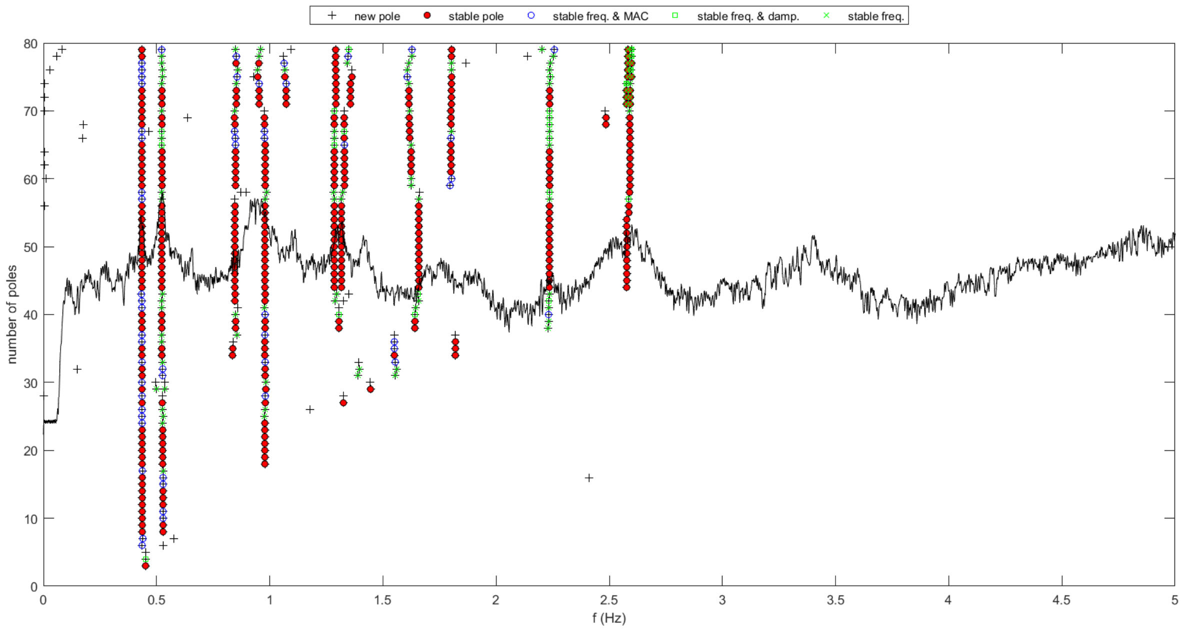

Figure 39: Stabilization diagram of 11th sensor

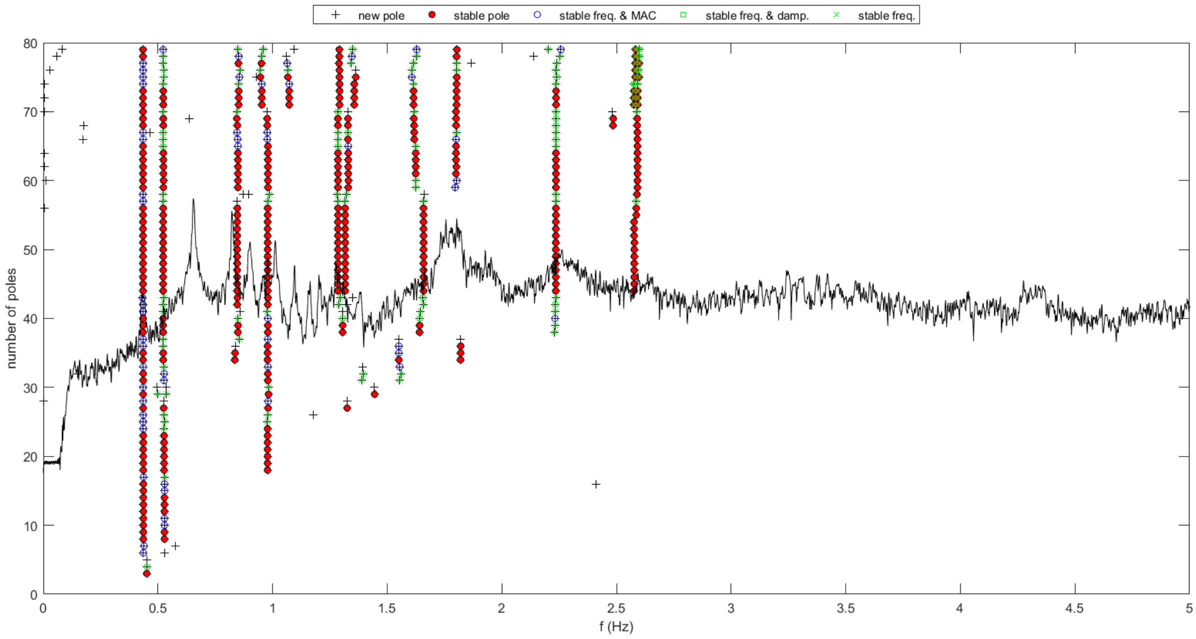

Figure 40: Stabilization diagram of 12th sensor

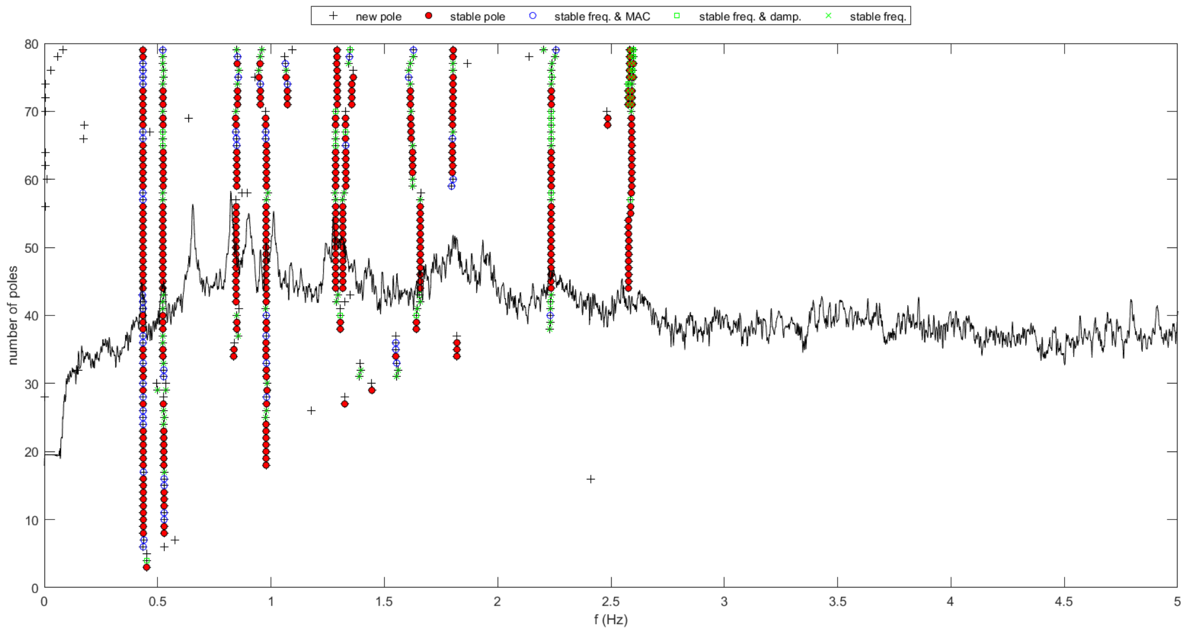

Figure 41: Stabilization diagram of 18th sensor

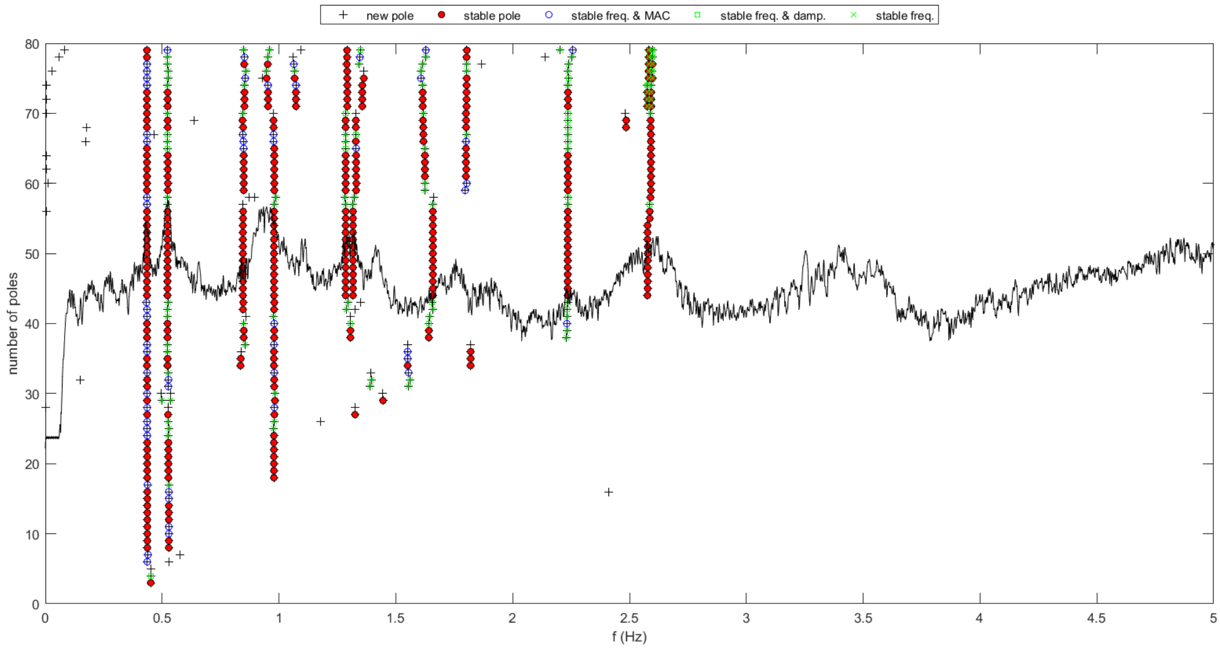

Figure 42: Stabilization diagram of 19th sensor

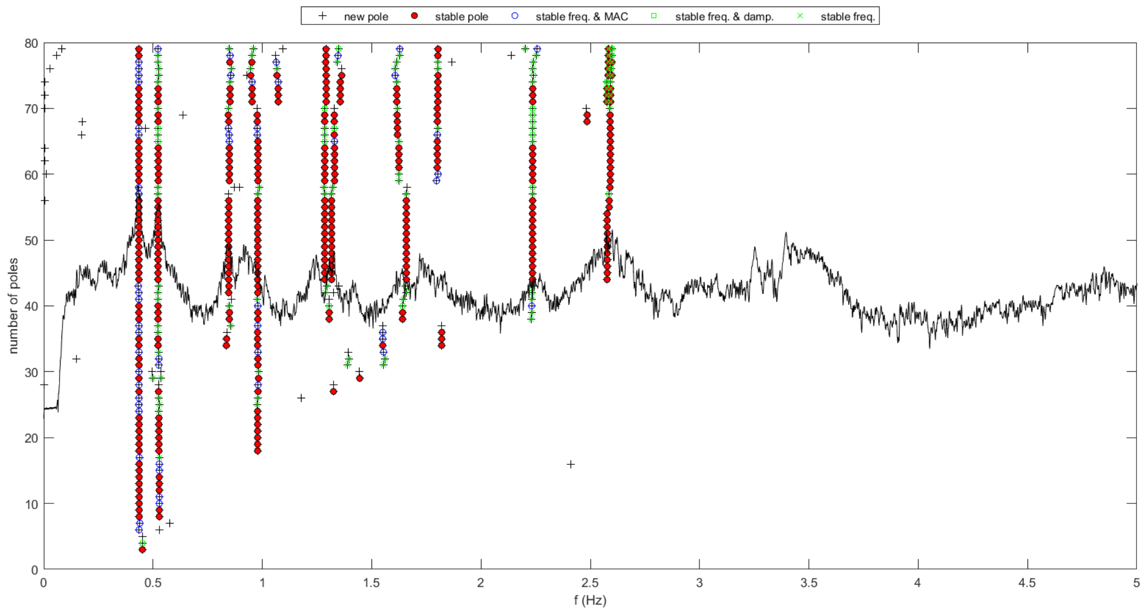

Figure 43: Stabilization diagram of 20th sensor

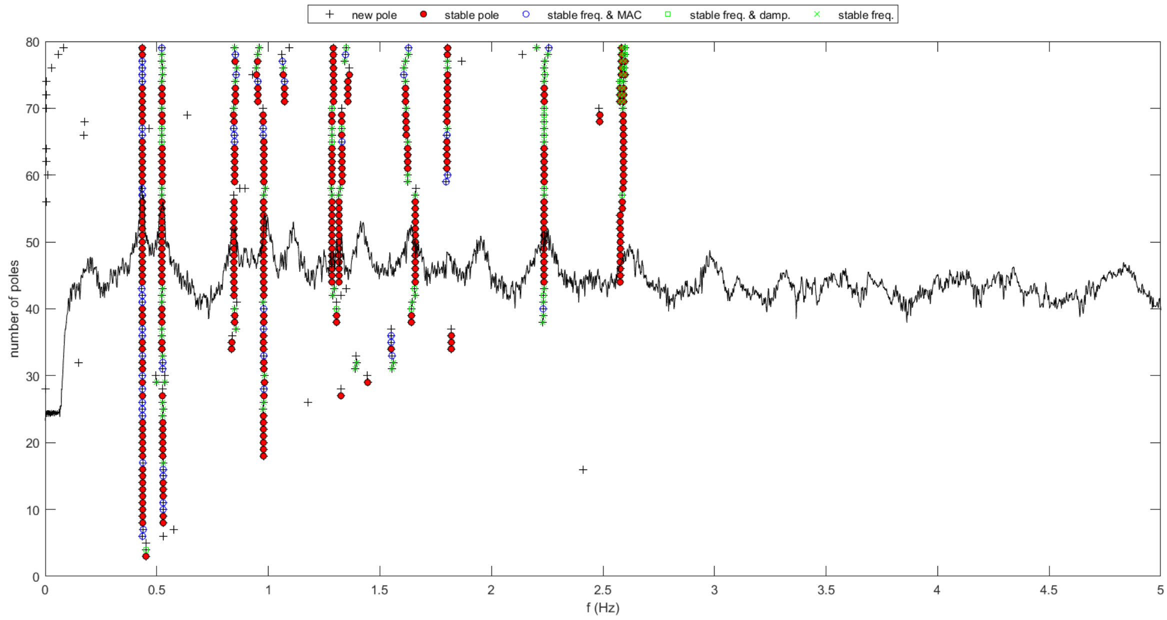

Figure 44: Stabilization diagram of 22nd sensor

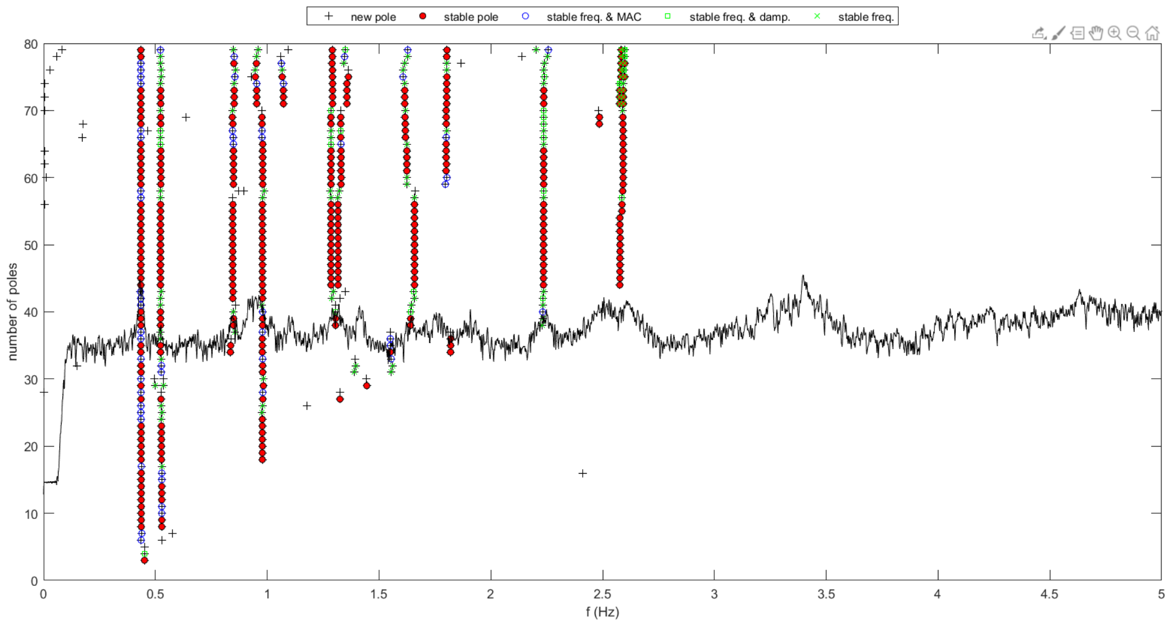

Figure 45: Stabilization diagram of 25th sensor

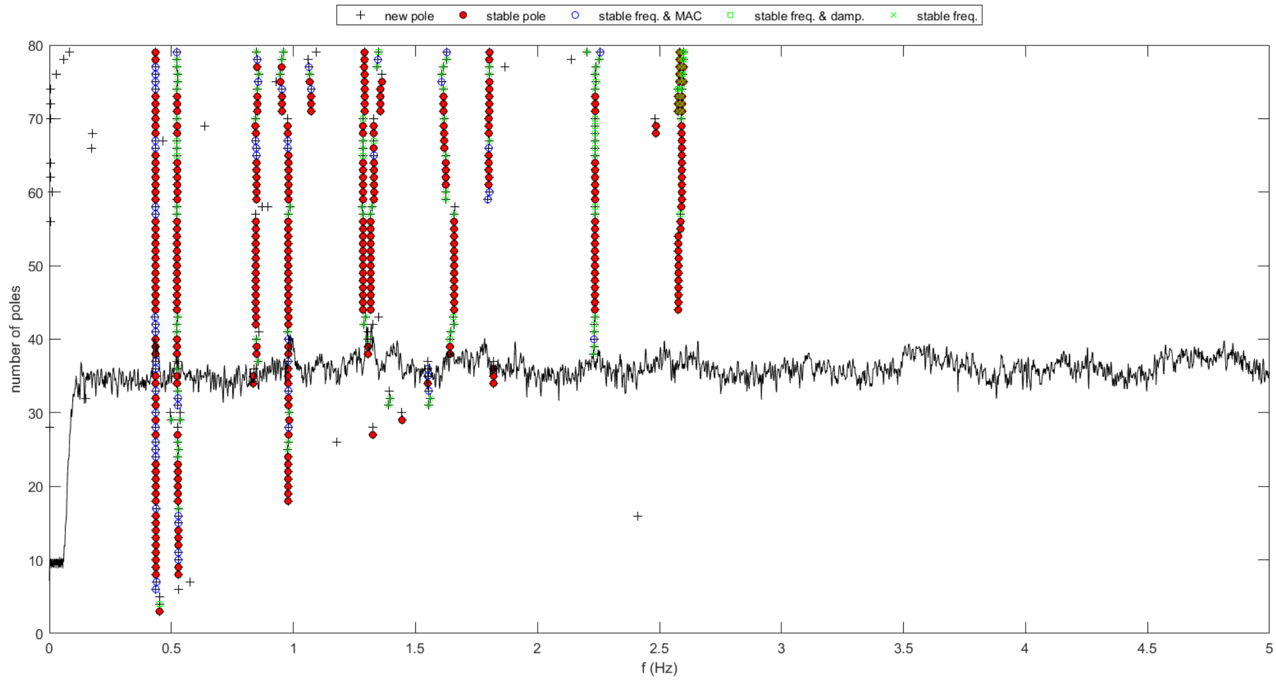

Figure 46: Stabilization diagram of 26th sensor

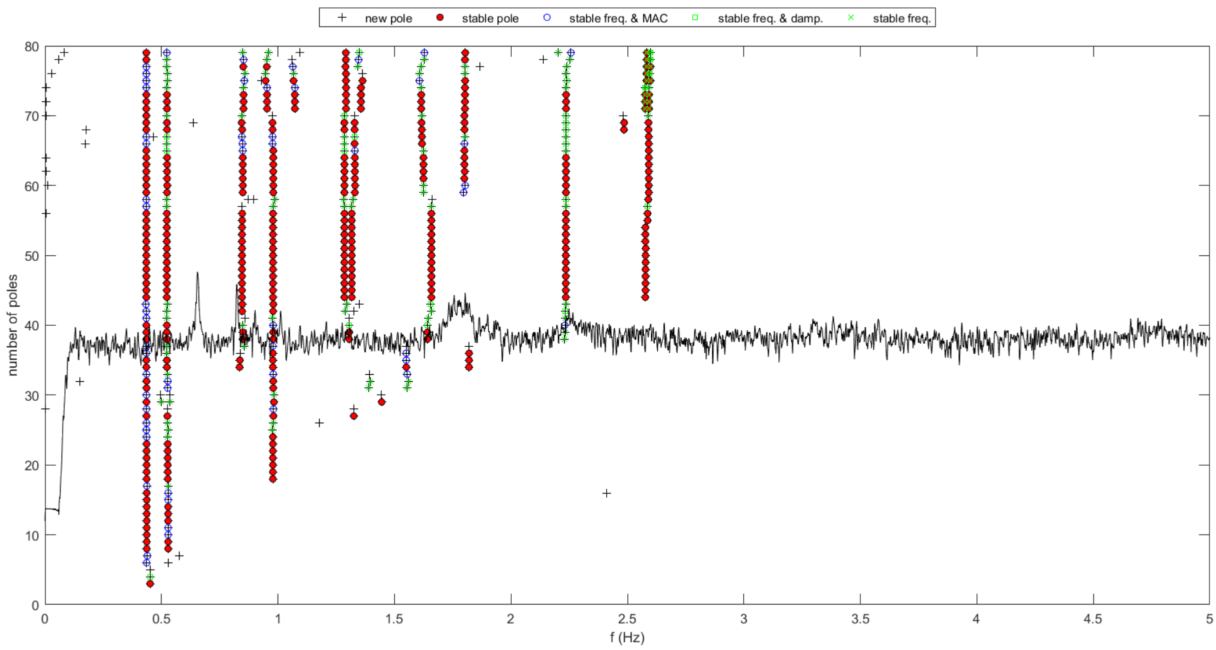

Figure 47: Stabilization diagram of 27th sensor

Stabilization diagrams typically depict eigenfrequencies as peaks. In Figures 36, 38, 40, 41, and 47, distinct and prominent modes at 0.7 Hz and 0.95 Hz are evident, although the software used did not label or identify these modes. It is important to note that these figures specifically utilize transverse sensor data. To further identify these modes, we conducted a dedicated analysis using MATLAB code, focusing exclusively on transverse data in the SSI_COV process. In figures 48, 49, 50, 51 and 52 there are the stabilization diagrams from Transverse data. In all peaks there are also the missing frequencies.

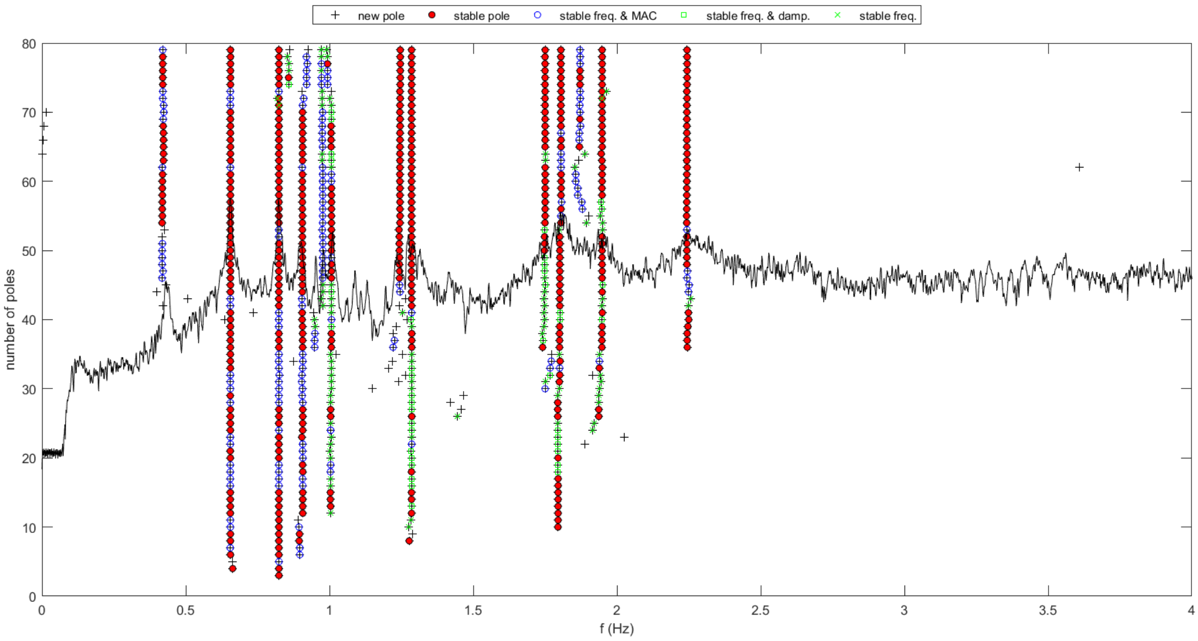

Figure 48: Stabilization diagram of 8th sensor

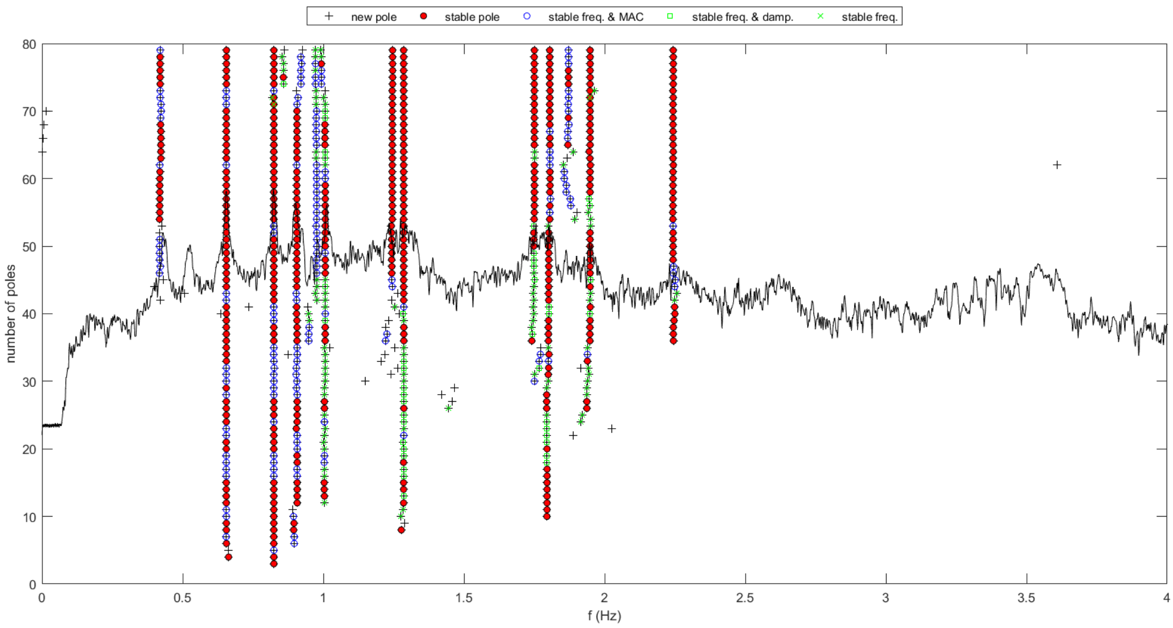

Figure 49: Stabilization diagram of 10th sensor

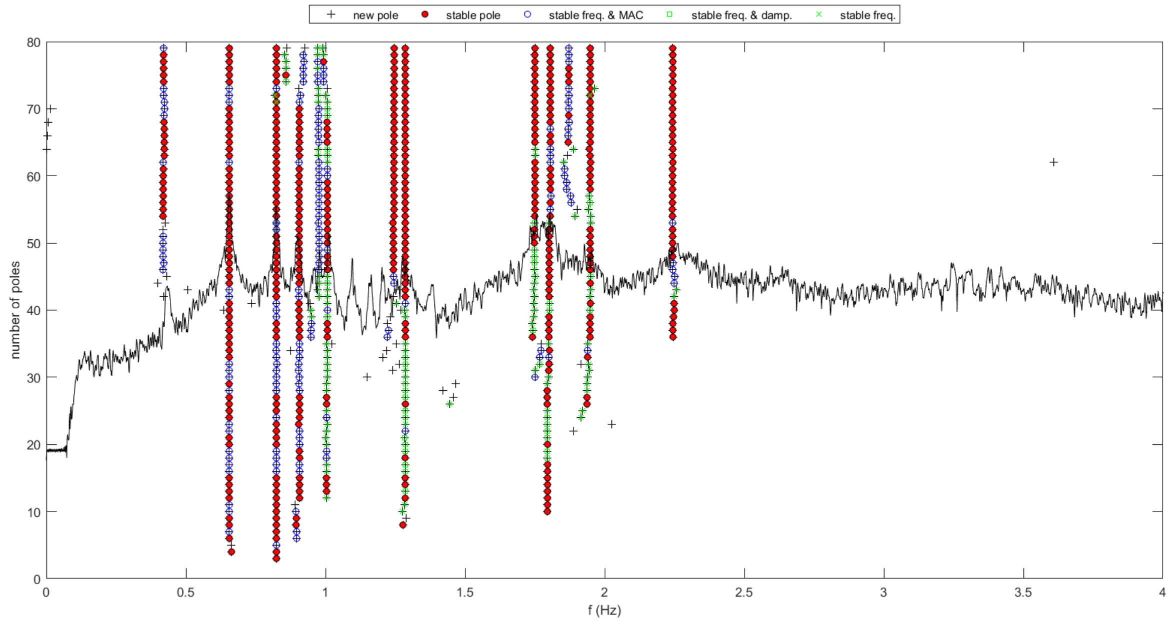

Figure 50: Stabilization diagram of 12th sensor

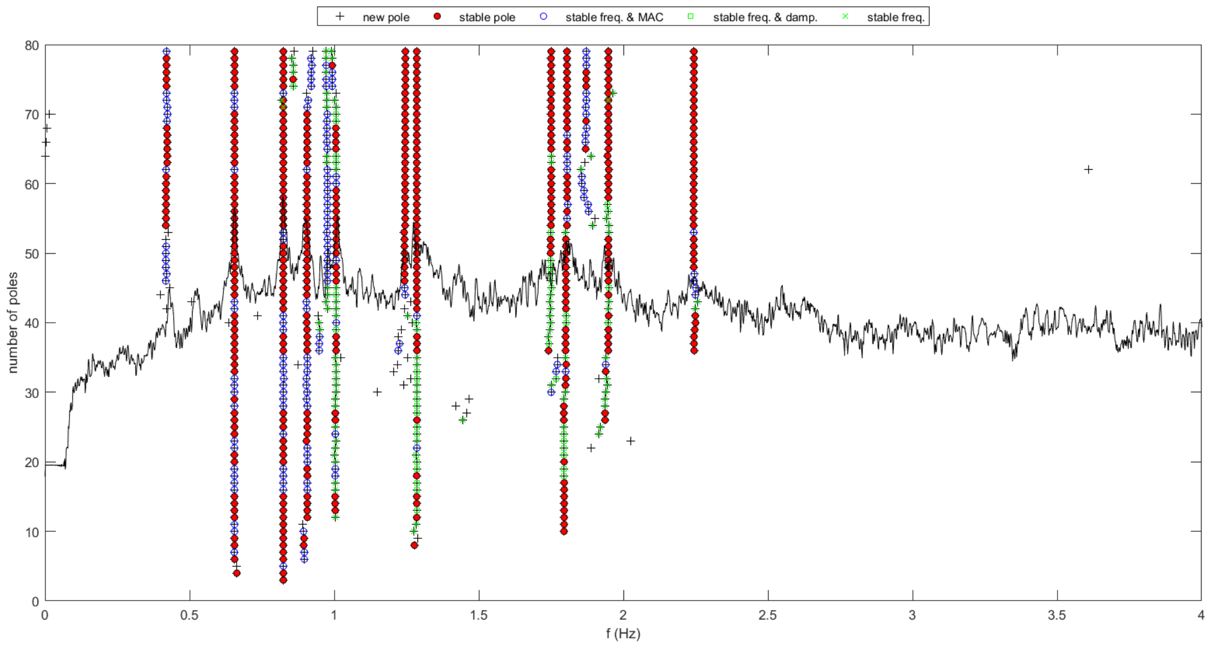

Figure 51: Stabilization diagram of 18th sensor

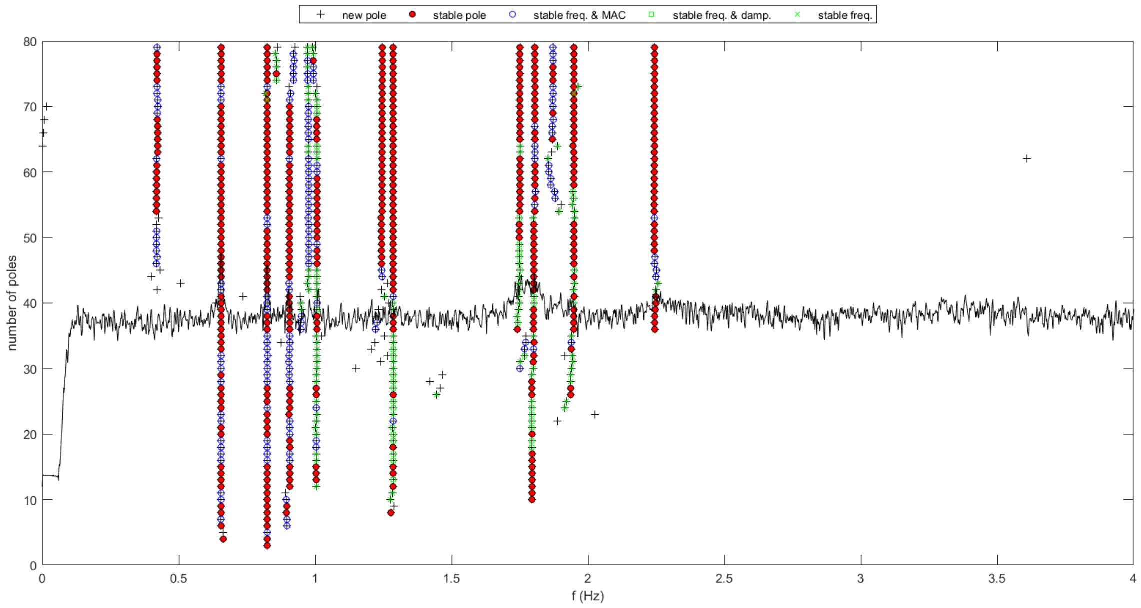

Figure 52: Stabilization diagram of 27th sensor

3.4 Modal Parameter Identification: A Comprehensive Analysis using SSI_COV Method

The eigenfrequencies at which the bridge oscillates are evident from the peaks observed in the graphs. Subsequently, a custom MATLAB function called SSI_COV by (E. Cheynet) found on the MathWorks website [5] was utilized to find mode natural frequencies mode shapes and damping ratios.

The function SSI_COV is employed within the realm of Subspace System Identification, a critical technique for elucidating the underlying dynamics of a system from measured input and output data. This method is instrumental in a variety of scientific and engineering applications, including structural health monitoring, where it is particularly effective for identifying modal parameters of structures such as bridges.

The algorithm employed by SSI_COV hinges on the covariance method. Essentially, it computes the covariance matrix of the data and projects this data onto a specific subspace. By analyzing this subspace, SSI_COV can reveal the structure's hidden dynamics, including its order (the number of independent state variables in the system), which is a critical aspect for model identification.

By utilizing SSI_COV, can gain insights into the frequency content of the bridge response and identify the specific frequencies at which the bridge exhibits pronounced vibrations. The SSI_COV function offers several advantages in bridge vibration analysis. Firstly, it allows for a more accurate determination of the bridge's resonant frequencies, which are critical in understanding the structural behavior and potential vulnerabilities. Additionally, SSI_COV aids in assessing the damping properties of the bridge system, providing insights into the dissipation of energy during vibrations. This information is crucial for evaluating the bridge's dynamic response to various loading conditions and environmental factors. In summary, the SSI_COV function in MATLAB is a valuable tool for the analysis and characterization of bridge vibrations. Its ability to estimate cross-covariance between input and output signals in the frequency domain enables researchers to identify resonant frequencies, assess damping properties, and gain a deeper understanding of the dynamic behavior of the bridge. This information is essential for structural health monitoring, design optimization, and overall assessment of the bridge's performance and integrity.

The function takes as input the matrix "y," which contains the time series of ambient vibration data, sampled with a time step "dt."

In addition to the mandatory inputs, SSI_COV function also accepts optional parameters specified using the ‘’varargin’’ argument. These optional parameters include 'Ts' for the time lag used in covariance calculation, 'methodCOV' for selecting the method of covariance estimation (either 1 or 2), 'Nmin' and 'Nmax' to set the minimum and maximum model order, 'eps_freq' for frequency accuracy, 'eps_zeta' for damping accuracy, 'eps_MAC' for accuracy in Modal Assurance Criterion (MAC), and 'eps_cluster' for the maximal distance within each cluster.

The output of the SSI_COV function includes the identified eigenfrequencies stored in the variable "fn," the modal damping ratios in "zeta," and the mode shapes in "phi." Additionally, the function may also provide further output variables stored in the "varargout" structure, which can be useful for generating stabilization diagrams.

3.4.1 Parameters for SSI_COV

When selecting parameters for the SSI_COV function, it is essential to consider the characteristics of your specific data and the objectives of your analysis. Here are some guidelines to help you choose the parameters:

1. Time Lag (Ts): The time lag parameter determines the lag used in covariance calculations. It should be chosen carefully to capture the correlation between different time steps. A good starting point is to set Ts to a value that corresponds to the autocorrelation decay of your data.

2. Method for Covariance Estimation (methodCOV): The methodCOV parameter allows you to choose between two methods for covariance estimation: 1 or 2. You may try both methods and assess which one provides more accurate results for your data.

3. Model Order Range (Nmin and Nmax): Selecting the appropriate range for the model order is crucial. It determines the number of modes to be identified. Consider the complexity of your system and the expected number of dominant modes. Start with a small range and gradually increase it until you observe convergence in the identified modes.

4. Accuracy Parameters (eps_freq, eps_zeta, eps_MAC, eps_cluster): These parameters control the convergence criteria for frequency, damping ratio, Modal Assurance Criterion (MAC), and cluster distance, respectively. Adjust these parameters based on the desired accuracy of the identified modal parameters. Smaller values will result in higher precision but may require more computational time.

In the context of parameter selection for the modal analysis algorithm, it is crucial to acknowledge that some iterative refinement may be necessary. It is recommended to initiate the process with conservative or default parameter values and progressively fine-tune them based on the obtained results and the specific characteristics of the data. Seeking guidance from relevant literature and consulting with experts in the field of modal analysis can provide valuable insights and recommendations for optimizing parameter selection. By adopting this approach, a more accurate and reliable modal analysis can be achieved for the specific objectives outlined in the thesis.

In this case, the time lag was set to 5 seconds, and the minimum and maximum order of the system (Nmin and Nmax) were set to 3 and 80 respectively. These settings were determined after observation revealed that further increasing the time lag and the maximum order, or reducing the minimum order, does not significantly influence the precision of the results. However, it was found that these modifications drastically increase the computational time. This understanding allows for an optimized balance between computational efficiency and the accuracy of the system identification results.

Upon observing the stabilization diagrams, it is evident that the eigenfrequencies derived from the SSI_COV analysis are prominently represented. These frequencies, signifying the structure's natural modes of vibration, appear as stable poles persisting across multiple model orders in the diagram. This clear depiction confirms the efficacy of the SSI_COV method in extracting accurate and meaningful dynamic characteristics from the ambient vibration data. The stability of these modes over a range of model orders not only attests to their authenticity but also provides reassurance about their reliability amidst potential computational artifacts or noise. Hence, the stabilization diagrams serve as a valuable visualization tool, effectively confirming the veracity of the identified eigenfrequencies from the SSI_COV analysis.

3.5 Modal Parameters presentation

The SSI_COV method was applied to the vibration analysis in order to identify the modal parameters of the M-DOF system. Through this analysis, the modal frequencies, damping ratios, and mode shapes were successfully detected and calculated. The obtained modal parameters provide crucial insights into the dynamic behavior of the system under ambient vibrations. These parameters were organized and presented in a comprehensive table, facilitating a clear overview of the identified modes and their corresponding characteristics. By presenting the modal parameters in a structured table, it becomes easier to compare and analyze the vibrational properties of different modes within the system. This information serves as a valuable resource for further analysis and understanding of the system's response to ambient vibrations.

3.5.1 Eigenfrequencies and damping ratios

The eigenfrequencies obtained using the SSI_COV method provide valuable insights into the vibrational behavior of the bridge. These eigenfrequencies represent the natural frequencies at which the bridge oscillates when subjected to ambient vibrations. Each eigenfrequency corresponds to a specific mode of vibration, indicating the different patterns and shapes of motion exhibited by the bridge structure. The determination of these eigenfrequencies is crucial in understanding the dynamic characteristics of the bridge and assessing its structural integrity. The results confirm the patterns observed in the frequency domain, as the peaks indicating eigenfrequencies are represented as values in the table 2. From the power spectrum, it appears that the predominant frequencies, characterized by higher peaks, are between 0 Hz and 5 Hz. Therefore, we can focus our attention on the following eigenfrequencies for further analysis. Damping ratio emerges as a fundamental parameter for comprehending the vibrational characteristics of these structures. Its determination often involves sophisticated methods, among which the Subspace State-Space System Identification Covariance (SSI_COV) function stands out. It is essential to note that the SSI_COV function can derive damping ratios for each eigenfrequency, where each eigenfrequency corresponds to a distinct mode of bridge oscillation. Consequently, presenting damping ratios alongside eigenfrequencies enhances the comprehensibility of the results, providing a holistic view of a bridge's dynamic behavior. In Tables 2a, b presented eigenfrequencies and damping ratios derived from SSI_COV function.

Frequencies	Damping Ratios	Frequencies	Damping Ratios
0,4354	0,0098	6,7714	0,0147
0,5227	0,0134	8,4602	0,0132
0,5242	0,0139	9,7623	0,0061
0,5270	0,0041	9,7678	0,0071
0,8458	0,0227	10,1674	0,003
0,8488	0,0234	11,0850	0,0336
0,9782	0,0316	11,1575	0,0254
0,9788	0,0379	12,3163	0,0027
1,2851	0,0279	12,3370	0,0014
1,2919	0,0213	12,3849	0,0031
1,3167	0,0331	12,5234	0,0124
1,3301	0,0311	12,6344	0,0063
1,6587	0,0118	14,2238	0,002
1,8014	0,0378	15,5091	0,0016
2,2354	0,0115	17,3119	0,0061
2,2361	0,0117	18,5944	0,0029
2,5775	0,0173	36,0000	0,0032
2,5912	0,0168	37,5669	0,0117
5,5679	0,0092	38,7484	0,0056
6,6720	0,0153	39,6601	0,009

Table 2a: Eigenfrequencies and Damping ratios

FN(hz)	Damping ratios
0,6555	0,0041
0,9061	0,0077

Table 2b: Additional Eigenfrequencies and Damping ratios from transverse data

3.5.2 Mode Shapes

Mode shapes in structural dynamics describe the spatial distribution of displacement or deformation patterns exhibited by a vibrating structure at specific eigenfrequencies. They play a fundamental role in understanding the dynamic behavior and response of the system. Analytically, mode shapes can be represented as a series of shape functions that depict the relative amplitudes and phases of displacement at various points along the structure. In the specific case of bridge analysis, with 17 sensors placed along its length, the mode shapes can be visualized as patterns of displacements observed at these sensor locations. Each mode shape corresponds to a unique vibration pattern associated with a specific eigenfrequency. In Tables 3a, b we can see the matrix of mode shape. It has as many rows as the natural frequencies and as many columns as the number of sensors.

	Nymber of sensors																
FN(hz)	1	4	5	6	7	8	9	10	11	12	18	19	20	22	25	26	27
0.4354	-1	-0.441446	0.353424	0.3637786	0.084919	-0.00369	0.45245747	-0.0337094	-0.022307	0.003212	0.0074603	-0.024756834	-0.421695	-0.817964	-0.000608343	-0.000199305	-4.152E+09
0.5227	-0.8480333	-0.340939	-0.337609	-0.322291	-0.04208	-0.00156	0.53990434	0.04334837	-0.002856	-0.00376	0.0221077	-0.005805081	-0.35649	-1	0.000884953	0.000375601	9.29E+09
0.5242	-0.7655916	-0.326216	-0.401905	-0.370537	-0.05628	0.00151	0.48496107	0.05467214	0.003622	-0.00061	0.0170583	0.001090784	-0.324805	-1	0.001128696	0.000416647	4.30E+09
0.5270	-0.8442548	-0.375796	-0.288416	-0.3001961	-0.02103	0.00014	0.53522733	0.01806964	-0.016411	0.00136	0.0245192	-0.018984587	-0.366267	-1	0.000619929	0.000442807	4.25E+09
0.8458	-0.1417368	0.3118088	-0.148811	-0.1798721	-0.12425	0.016466	-0.7495575	0.00877617	0.045169	9.82E+09	0.0477199	0.046213669	0.3066285	1	0.000842758	6.07E+09	-8.90E+08
0.8488	-0.1876028	0.3157419	-0.140166	-0.1835766	-0.11885	0.016009	-0.7627007	0.00643156	0.043387	-0.0011	0.0506385	0.044052592	0.3072195	1	0.000828551	9.53E+08	2.54E+09
0.9782	-0.4787319	0.0674735	0.034357	0.0655902	-0.06447	-0.00249	-0.6237533	-0.001541	0.025248	0.000329	-0.004196	0.027647246	0.0581811	-1	0.000278929	-0.00017576	-4.42E+09
0.9788	-0.5591111	0.0480794	0.033959	0.0558842	-0.04951	-0.00249	-0.651961	-0.0040278	0.014975	0.003192	-0.002912	0.017728233	0.0495411	-1	0.000171644	-0.000187775	-6.10E+09
1.2851	-1	0.1774842	-0.048097	-0.0673408	0.118389	-0.01345	0.55464902	-0.0168271	-0.071068	0.016075	-0.020212	-0.064024379	0.1945574	-0.813084	-0.001016982	-0.000144955	-0.000137972
1.2919	-0.356556	-0.135563	0.096139	-0.0508746	-0.04184	-0.00389	-1	-0.0344185	0.025634	0.003975	-0.022964	0.026462295	0.0408179	-0.009068	0.00023682	8.11E+09	3.16E+09
1.3167	-0.684977	0.0700967	0.129675	-0.0385176	-0.02526	0.011729	1	-0.110337	-0.012132	0.002863	-0.038541	-0.016816495	0.3225284	-0.620468	-0.000474544	6.81E+09	-0.000143233
1.3301	-0.697115	0.0613899	0.083091	-0.0343866	0.004212	0.004843	1	-0.0899397	-0.026074	0.005666	-0.025636	-0.028440082	0.2684487	-0.821243	-0.000599463	4.12E+09	-0.000171754
1.6587	-1	-0.077362	0.225504	0.070611	0.173698	-0.02081	-0.0554268	0.15632943	-0.056112	0.053169	0.0958251	-0.043095053	-0.295915	0.1194363	-0.001857727	-0.001392931	-0.000547703
1.8014	-0.8277051	0.0652151	0.04272	-0.0132682	-0.03502	-0.08297	1	-0.0069183	0.011826	-0.00239	0.1883852	0.027451363	-0.154962	-0.464731	0.000260824	0.000160426	-7.50E+09
2.2354	-0.0046327	0.1391932	-0.224843	-0.0831728	-0.13387	0.177987	0.4708811	-0.3072998	0.084027	-0.19812	-0.267015	0.050963007	0.2569681	1	0.002611627	0.002002122	0.001532702
2.2361	-0.3674384	0.2347216	-0.343807	-0.1610129	-0.16216	0.208391	0.98036449	-0.4042306	0.123416	-0.25784	-0.290047	0.086655616	0.3322459	1	0.003827227	0.002660847	0.00201763
2.5775	0.5710161	0.9793002	-0.873111	-0.8963015	0.233732	0.005145	-1	0.07022387	0.16246	0.021426	-0.003113	0.162530902	0.7821112	-0.765122	0.003279257	0.00048896	-0.000197623
2.5912	0.56839993	0.9879234	-0.879104	-0.9006957	0.23607	-0.00038	-0.9749459	0.06860021	0.166184	0.021041	0.0088593	0.166513525	0.7759592	-0.839045	0.003193404	0.000536666	-0.000175491
5.5679	-0.0006211	-3.23E+09	-0.00032	-0.0003298	0.061224	5.67E+09	0.00031283	5.98E+09	-6.91E+09	6.67E+09	7.38E+09	-0.000167158	-0.000295	-0.00026	2.38E+09	-2.31E+09	-1.93E+08
6.6720	-0.0158896	0.005452	-0.026495	0.0131688	1	0.027659	0.02938871	-0.0322292	-0.007752	0.022016	0.0197348	0.002877052	-0.003834	0.0797872	-0.000661494	8.05E+09	-0.001698793
6.7714	-0.0016668	0.0002854	-0.000331	0.0006619	-1	-0.00246	-0.0052924	0.00336204	0.000348	0.000805	0.0012661	0.000441573	-0.000503	0.0014166	-6.41E+09	-7.21E+09	0.000181134
8.4602	-0.0053254	-0.00068	0.004313	-0.0012351	1	-0.00134	0.00219488	0.00191027	-0.002137	-0.00073	-0.001586	-0.001240942	0.0002902	-0.009672	0.000359572	0.000319895	-8.16E+09
9.7623	-0.0796683	0.0062922	0.041041	0.0211759	-1	-0.00274	0.02976049	-0.017519	-0.002625	-0.003	-0.002611	0.024381048	0.0077182	-0.101536	0.014914551	-0.008193103	-0.000901511
9.7678	-0.0036189	-2.24E+09	0.002804	0.0018615	-1	0.000637	0.00269952	-0.001136	0.001208	0.001106	0.0002336	0.000134979	0.0002737	-0.001093	0.000509397	-0.000134175	0.00116228
10.1674	-0.0079209	0.0013816	0.003415	-0.0075636	0.73913	-0.00042	-0.0192263	0.00686776	0.004806	0.007302	-0.001351	-0.007434342	-0.001169	-0.006428	-0.005850231	0.000912051	0.000305564
11.0850	-0.0374231	-0.000488	0.006151	-0.0085129	1	0.002101	0.00700112	0.00898248	-0.004107	0.001478	-0.001313	-0.005244	0.0001346	-0.016013	0.004484087	-0.005004007	-0.000705676
11.1575	-0.0016566	-0.000153	0.006469	-0.0039759	0.882353	-0.00023	-0.00712	0.00269683	-0.004443	0.000859	0.0004437	0.000650106	-0.000383	-0.0276	0.002972252	-0.003314448	-0.000667291
12.3163	-0.0344305	-0.021818	0.005788	-0.0024517	-1	0.002821	0.02705367	-0.0063878	0.005529	0.003995	0.0133171	0.004975382	-0.012145	0.0654227	0.009643459	-0.003236728	0.000234535
12.3370	-0.044086	-0.024969	0.010644	-0.0012324	-1	0.002227	0.03678399	-0.0055906	0.008076	0.003419	0.0148895	0.005252903	-0.013391	0.0803419	0.011339918	-0.002822105	0.000167126
12.3849	-0.0250576	-0.011523	0.011661	0.0202591	-1	-0.00524	0.00322669	-0.0291096	0.000518	-0.00467	0.006662	0.010481104	-0.006806	0.0545244	0.011589488	0.004736586	-0.000194667
12.5234	-0.0184537	-0.0182	-0.017595	-0.007231	-1	-0.00197	0.00243029	0.01152109	-0.004836	0.007709	0.0064018	0.0036104	-0.016565	-0.033905	-0.000573449	-0.008083137	0.001929894
12.6344	0.00897698	0.0137116	0.032068	0.0115542	1	0.003712	0.00067987	-0.0213375	0.009503	-0.01181	-0.001351	-0.000464122	0.0153066	0.0611903	0.001564516	0.012174134	-0.003144217
14.2238	-0.0319915	0.0026844	0.020776	0.0084655	1	0.015389	-0.0370024	-0.0819406	0.002198	0.003213	0.0059786	-0.000246937	0.0156071	-0.030955	-0.000544143	0.001684626	0.003388297
15.5091	-0.0009977	0.0093657	-0.002201	-0.0002154	0.952381	-0.00137	-0.0132017	-0.0054688	0.004328	0.003716	-0.002175	0.005308853	0.0016395	-0.053932	-0.00239714	0.001021172	0.000693295
17.3119	-0.0162593	-0.005382	-0.023245	-0.0205958	-1	0.012947	0.01703446	0.02477506	-0.014763	-0.02079	0.0297722	-0.002788971	-0.010487	-0.015201	0.003699361	-0.001006498	-0.00030757
18.5944	-0.0030142	0.0076286	-0.001704	-0.0024489	-1	-0.00243	0.00292298	-0.0010606	0.002087	-0.00244	-0.002734	0.005099567	0.0017419	-0.014377	8.83E+09	0.001971019	-0.000405855
36.0000	-0.0097361	-0.00236	-0.006291	-0.0026067	-1	-0.00122	0.00254406	0.00105264	0.002677	-0.00052	-0.000706	0.001976873	-0.001647	-0.004388	0.00127485	-0.00045291	5.13E+09
37.5669	0.00122002	0.0017753	0.003276	-0.0005528	0.84	0.000439	-0.0007226	0.00028463	-0.000871	-0.00023	0.0002861	-0.000970957	0.0016846	0.0021208	-4.58E+09	0.000121113	0.000208739
38.7484	-0.0111663	-0.000429	-0.000649	-0.0003254	1	0.000286	-0.002545	0.00350863	-0.000568	-0.00282	1.11E+09	0.000215217	0.0010357	-0.002263	-0.000728459	-0.000246198	0.00039988
39.6601	-0.0042416	0.0020808	-0.008363	0.0039855	-1	0.001207	-0.0031784	0.00692392	-1.72E+09	-0.00062	-0.000281	-0.000106311	-0.001373	-0.002559	0.0003043	-0.000217523	0.00223883

Table 3a: Mode shapes

	Nymber of sensors				
FN(hz)	8	10	12	18	27
0,6555	0,3147	1	-0,4434	0,8766	0,0049
0,9061	0,1151	0,9974	-0,1761	0,9935	0,0023

Table 3b: Additional Mode shapes of transverse data

3.5.3 Modal Assurance Criterion

In tables 2 and 3 similar eigenfrequencies are observed and as a result mode shapes and damping ratio. To check the similarity between them Modal Assurance criterion is used. The Modal Assurance Criterion (MAC) [6] is a widely employed technique in vibration analysis to quantitatively assess the similarity or correlation between mode shapes. It serves as a valuable tool for validating and verifying the accuracy of mode shapes obtained through experimental or numerical modal analysis. By comparing the spatial distributions of mode shapes, the MAC provides a quantitative measure ranging from 0 to 1, where a value of 1 indicates perfect similarity and a value of 0 represents no correlation.

$$MAC(\{\varphi_r\},\{\varphi_s\}) = \frac{|\{\varphi_r\}^{*t}\{\varphi_s\}|^2}{(\{\varphi_r\}^{*t}\{\varphi_r\})(\{\varphi_s\}^{*t}\{\varphi_s\})}$$

Figure 53: Modal Assurance Criterion

The calculation of the Modal Assurance Criterion (MAC) between mode shapes, a fundamental aspect of modal analysis, is carried out by the code using the formula in figure 53. Significance lies in the code's ability to assess the similarity of mode shapes based on their MAC values. A predefined MAC threshold, typically set at 0.8, is employed to categorize mode shapes and their corresponding eigenfrequencies as 'similar' when their MAC value is surpassed by this threshold. This categorization process results in the creation of distinctive groups of mode shapes, each characterized by their eigenfrequencies and damping ratios. Importantly, it is ensured by the code that only mode shapes with a MAC value below 0.8 are retained in these groups, thus enabling the filtering and isolation of modes that display dissimilar vibrational patterns. Also, a mat file is saved with these mode shapes eigenfrequencies and damping ratios and the modal parameters before calculating MAC criterion. So, all mode shapes which have mac number over 0.8 are considered the same and its natural frequencies and damping ratios. Due to the limited number of the sensors some eigenfrequencies over 6 Hz seem to be similar and mac number close to 1. In table 4 are presented eigenfrequencies and damping ratios with no similarity between them which means mac number below 0.8. Also, there are the additional two eigenfrequencies with mode shapes which are recognized from transverse sensors and are dominant. Below 6 Hz were found 14 eigenfrequencies and corresponding damping ratios are from 1% to 3%.

Fn(Hz) 0-30 min	Damping Ratios
0.4354	0.0098
0.5227	0.0134
0.8458	0.0227
0.9782	0.0316
1.2851	0.0279
1.2919	0.0213
1.3167	0.0331
1.6587	0.0118
1.8014	0.0378
2.2354	0.0115
2.5775	0.0173
5.5679	0.0092
MODE SHAPES FROM TRANSVERSE DATA	
0.6555	0.0041
0.9061	0.0077

Table 4: Eigenfrequencies and its Damping ratio

By analyzing and interpreting the identified eigenfrequencies, valuable information can be obtained regarding the fundamental modes of vibration and their corresponding frequencies, enabling further analysis and evaluation of the bridge's dynamic response.

The damping ratio obtained through the SSI_COV function holds particular significance due to its capacity to unveil critical insights into the energy dissipation behavior of bridge structures when subjected to dynamic loads. By meticulously scrutinizing these ratios, one can gain a profound understanding of how a bridge responds to external forces, thereby facilitating a comprehensive assessment of its structural integrity and operational viability.

It is worth noting that damping ratios are influenced by a multitude of factors encompassing material properties, structural design, and environmental conditions. Nevertheless, the SSI_COV function emerges as a robust and dependable tool for precisely estimating these ratios. As such, its utilization in the realm of structural health monitoring plays a pivotal role in ensuring the safety and long-term durability of bridge structures.

In essence, the damping ratio, a dimensionless parameter, characterizes the rate at which oscillations within a system decay following a disturbance. This concept carries significant importance in the field of engineering, particularly when striving to comprehend how structures like bridges react to external forces such as wind, traffic, or seismic events.

In the context of bridge engineering, the damping ratio offers valuable insights into the rate at which a bridge would return to a state of equilibrium after experiencing an external event, such as an earthquake or strong gusts of wind. It essentially quantifies the dissipation of energy in each oscillation cycle.

Expressed as a fraction of critical damping, the damping ratio provides a measure of how effectively a system dissipates energy. Critical damping, the minimum level of damping required to bring a system to rest in the shortest time without oscillations, serves as a reference point. Specifically, a damping ratio of 1 indicates critical damping, where the system quickly returns to equilibrium without oscillation. A damping ratio less than 1 denotes underdamping, resulting in oscillations, while a damping ratio greater than 1 signifies overdamping, leading to a slower return to equilibrium.

In table 5 are presented mode shapes that display dissimilar vibrational patterns. There are also the additional patterns from transverse data. There are zeros in longitudinal and vertical sensors because these patterns were derived from only transverse sensors.

	Nymber of sensors																
FN(hz)	1	4	5	6	7	8	9	10	11	12	18	19	20	22	25	26	27
0.4354	-1	-0.441446	0.353424	0.3637786	0.084919	-0.00369	0.45245747	-0.0337094	-0.022307	0.003212	0.0074603	-0.024756834	-0.421695	-0.817964	-0.000608343	-0.000199305	-4.15E+09
0.5227	-0.8480333	-0.340939	-0.337609	-0.322291	-0.04208	-0.00156	0.53990434	0.04334837	-0.002856	-3.76E-03	0.0221077	-0.005805081	-0.35649	-1	0.000884953	3.76E-04	9.29E+09
0.8458	-0.1417368	0.3118088	-0.148811	-0.1798721	-0.12425	0.016466	-0.7495575	0.00877617	0.045169	9.82E+09	0.0477199	0.046213669	0.3066285	1	0.000842758	6.07E+09	-8.90E+08
0.9782	-0.4787319	0.0674735	0.034357	0.0655902	-0.06447	-0.00249	-0.6237533	-0.001541	0.025248	0.000329	-0.004196	0.027647246	0.0581811	-1	0.000278929	-0.00017576	-4.42E+09
1.2851	-1	0.1774842	-0.048097	-0.0673408	0.118389	-0.01345	0.55464902	-0.0168271	-0.071068	0.016075	-0.020212	-0.064024379	0.1945574	-0.813084	-0.001016982	-1.45E-04	-1.38E-04
1.2919	-0.356556	-0.135563	0.096139	-0.0508746	-0.04184	-0.00389	-1	-0.0344185	0.025634	0.003975	-0.022964	0.026462295	0.0408179	-0.009068	0.00023682	8.11E+09	3.16E+09
1.3167	-0.684977	0.0700967	0.129675	-0.0385176	-0.02526	0.011729	1	-0.110337	-0.012132	0.002863	-0.038541	-0.016816495	0.3225284	-0.620468	-0.000474544	6.81E+09	-0.000143233
1.6587	-1	-0.077362	0.225504	0.070611	0.173698	-0.02081	-0.0554268	0.15632943	-0.056112	0.053169	0.0958251	-0.043095053	-0.295915	0.1194363	-0.001857727	-0.001392931	-5.48E-04
1.8014	-0.8277051	0.0652151	0.04272	-0.0132682	-0.03502	-0.08297	1	-0.0069183	0.011826	-0.00239	0.1883852	0.027451363	-0.154962	-0.464731	0.000260824	0.000160426	-7.50E+09
2.2354	-0.0046327	0.1391932	-0.224843	-0.0831728	-0.13387	0.177987	0.4708811	-0.3072998	0.084027	-0.19812	-0.267015	0.050963007	0.2569681	1	0.002611627	0.002002122	0.001532702
2.5775	0.5710161	9.79E-01	-0.873111	-0.8963015	0.233732	5.14E-03	-1	7.02E-02	1.62E-01	2.14E-02	-3.11E-03	0.162530902	0.7821112	-0.765122	3.28E-03	4.89E-04	-1.98E-04
5.5679	-0.0006211	-3.23E+09	-0.00032	-0.0003298	0.061224	5.67E+09	0.00031283	5.98E+09	-6.91E+09	6.67E+09	7.38E+09	-0.000167158	-0.000295	-0.00026	2.38E+09	-2.31E+09	-1.93E+08
MODE SHAPES FROM TRANSVERSE DATA																	
0.6555	0	0	0	0	0	0.3147	0	1	-0.4434	0.8766	0	0	0	0	0	0	0.0049
0.9061	0	0	0	0	0	0.1151	0	0.9974	-0.1761	0.9935	0	0	0	0	0	0	0.0023

Table 5: Mode Shapes with its eigenfrequencies

By examining the mode shapes, it is possible to identify the regions of the bridge that experience higher or lower amplitudes of displacement during vibration. The mode shapes obtained from your analysis provide valuable insights into the dynamic characteristics of the bridge. They allow for the identification of critical areas of deformation and facilitate an assessment of the structural response under different loading conditions. By combining the mode shapes with the associated eigenfrequencies and damping ratios, you gain essential information for structural integrity evaluation, modal analysis, and further refinement of the bridge's design.

This information allows us to visualize and comprehend the patterns and characteristics of the bridge's vibrational modes. Analyzing the mode shape matrix helps identify areas of the bridge that experience pronounced displacements for each eigenfrequency. High-amplitude values in specific columns indicate that the corresponding sensors are positioned in regions where the bridge exhibits significant motion for the given mode. Conversely, low-amplitude values or values close to zero in certain columns suggest minimal displacement in those areas. The mode shape matrix serves as a valuable tool for understanding the vibrational behavior of the bridge and provides essential information for structural assessment and design considerations. By studying the patterns and variations within the matrix, engineers can gain valuable insights into the distribution of energy and displacement across the bridge, which is crucial for assessing structural integrity, identifying potential areas of concern, and optimizing design strategies to enhance the bridge's performance and resilience.

Mode shapes can vary from -1 to 1. This is because mode shapes are usually normalized for practical reasons. Normalization is a mathematical process that adjusts the values measured on different scales to a common scale. In the context of mode shapes, the values are typically normalized so that the maximum absolute value is 1 (or 100%). This does not mean that the physical displacement of the structure is limited to this range, but that the relative displacements of different parts of the structure are expressed in this range.

The purpose of normalization is to simplify the comparison and combination of mode shapes. Since the absolute values of the mode shapes are not usually of interest, it is easier to work with them if they are normalized to a standard range.

It's important to note that the negative values in mode shapes don't necessarily mean the structure is moving in a negative direction. Instead, they indicate a phase difference in the oscillation. For example, if one part of the structure is oscillating in the positive direction, a part that is oscillating in the opposite direction at the same time would be represented by a negative value in the mode shape.

3.5.3 Plot Mode Shapes

The mode shape matrix, composed of rows and columns, provides significant insights into the behavior of the bridge under different eigenfrequencies. Each row in the matrix corresponds to a specific eigenfrequency, representing a distinct mode of vibration for the bridge. On the other hand, each column represents an individual sensor location along the bridge. By examining the values within each row, we can observe the spatial distribution of displacement or deformation across the bridge for a particular eigenfrequency. The magnitude and sign of the values indicate the amplitude and direction of the displacement at each sensor location.

Understanding how a bridge moves at different eigenfrequencies solely from the mode shape matrix can be challenging. To address this issue, mode shapes [7] are visualized by plotting them on the bridge for each eigenfrequency. This visualization was achieved by creating the bridge geometry and sensor positions in MATLAB. Subsequently, all mode shapes were superimposed onto the bridge for clarity, as demonstrated in Figures 54-66.

In these diagrams, the blue lines represent the bridge structure itself. Each sensor is depicted with a red vector, symbolizing the transformation associated with that specific sensor location. The length of these vectors serves as a visual indicator of the magnitude of the transformation in relation to the other transformation vectors.

The decision to focus on mode shapes for eigenfrequencies below 5 Hz is deliberate and has its basis in the importance of these oscillations. Below 5 Hz, the bridge experiences vibrations and oscillations that are more critical. Therefore, examining mode shapes within this frequency range is pivotal for a comprehensive understanding of the bridge's dynamic behavior.

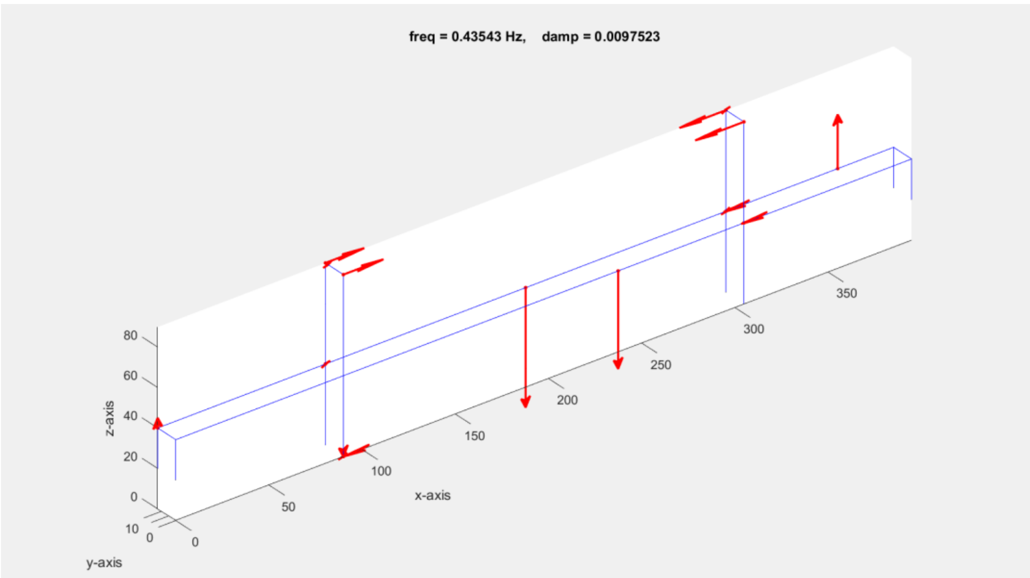

Figure 54: Mode shape in the bridge for 0.43543 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the longitudinal direction to negative direction. In deck level the deformation is smaller than in the top. Pier 2 (Voiotia side) deforms along its longitudinal axis to positive direction. Deck deforms for each span along the vertical axis. Right span deforms upward, middle span deforms downward and left span is supposed to move upward but there is no vertical sensor working in this part. In the bottom of piers there are neglective deformations as it is contracted.

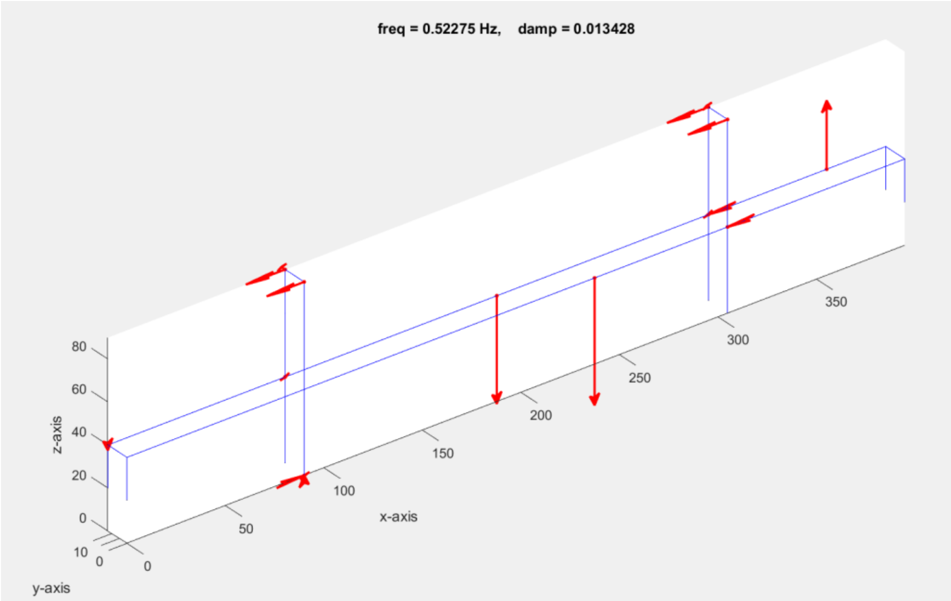

Figure 55: Mode shape in the bridge for 0.52275 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the longitudinal direction to negative direction. In deck level the deformation is smaller than in the top. Pier 2 (Voiotia side) deforms along its longitudinal axis to negative direction. Deck deforms for each span along the vertical axis. Right span deforms upward, middle span deforms downwards, and left span is supposed to move upward but there is no vertical sensor working in this part. In the bottom of piers there are neglective deformations as it is contracted.

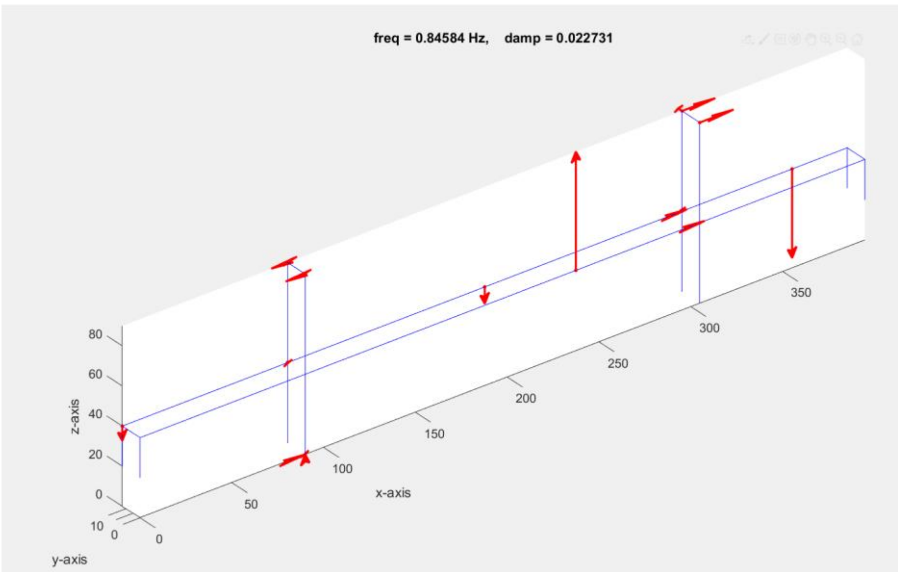

Figure 56: Mode shape in the bridge for 0.84584 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the longitudinal direction to positive direction. In deck level the deformation is smaller than in the top. Pier 2 (Voiotia side) deforms along its longitudinal axis to negative direction. Deck deforms for each span along the vertical axis. There is also torsion as in the middle span the one side deforms upward and the other downward. In the bottom of piers there are neglective deformations as it is contracted.

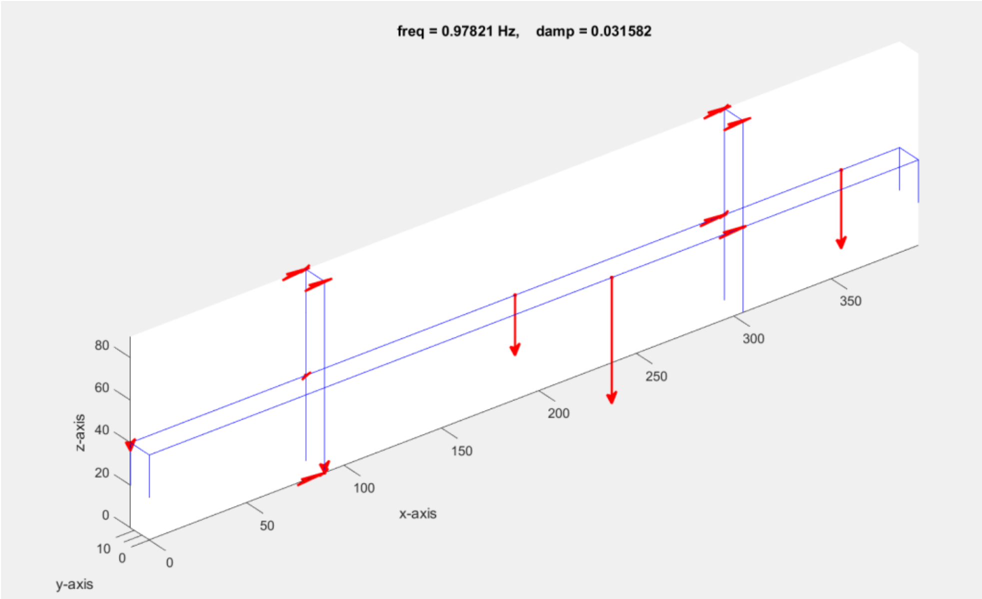

Figure 57: Mode shape in the bridge for 0.97821 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the longitudinal direction to positive direction. Pier 2 (Voiotia side) deforms along its longitudinal axis to positive direction. Deck deforms for each span along the vertical axis. Right span deforms downward and middle span deforms downward. In the bottom of piers there are neglective deformations as it is contracted.

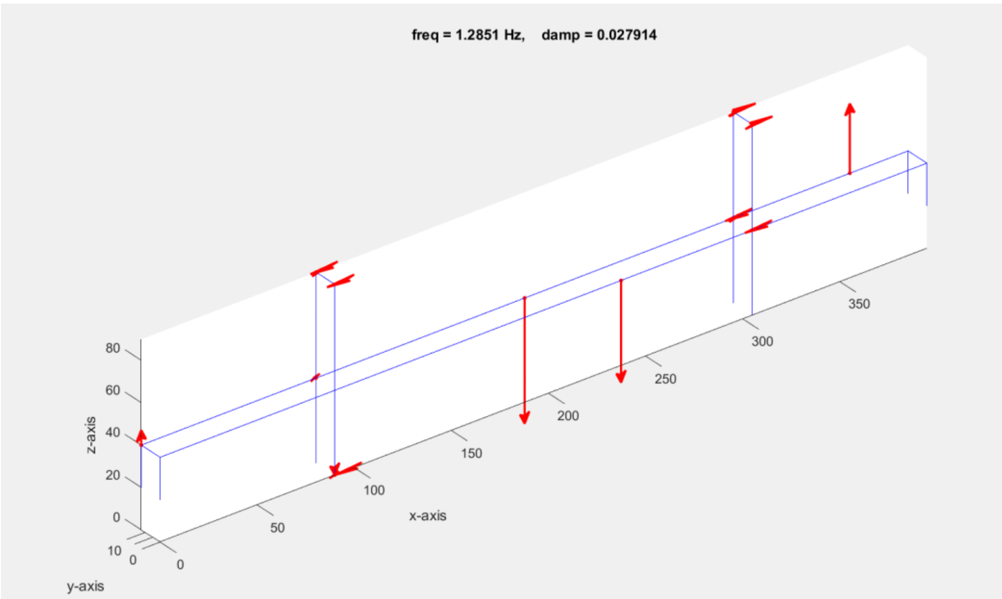

Figure 58: Mode shape in the bridge for 1.2851 Hz. Pier 1 (Evoia side) deforms along the longitudinal direction to the positive direction from top till the deck level and to the negative form deck level to bottom. Pier 2 (Voiotia side) deforms along its longitudinal axis to negative direction. Deck deforms for each span along the vertical axis. Right span deforms upward, middle span deforms downward and left span is supposed to move upward but there is no vertical sensor working in this part. In the bottom of piers there are neglective deformations as it is contracted.

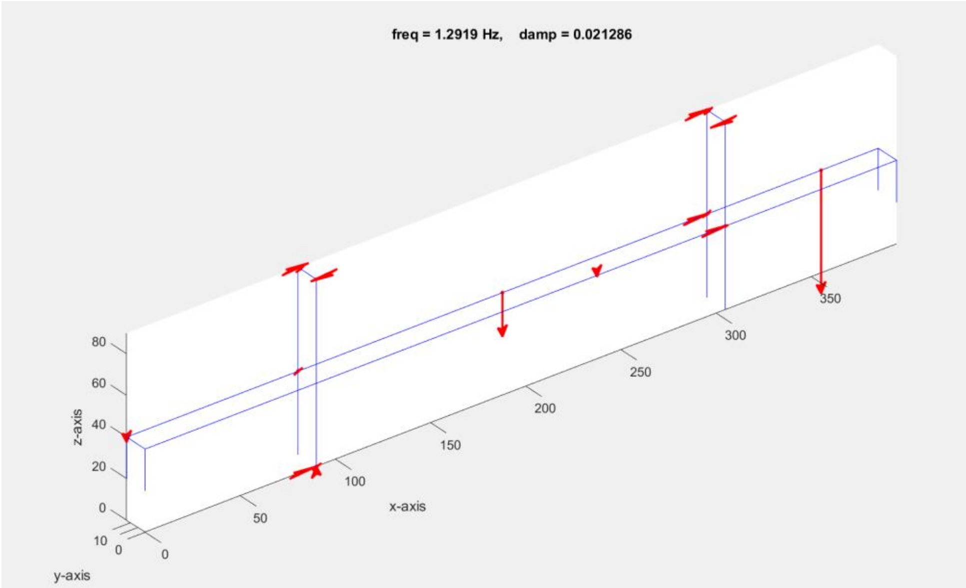

Figure 59: Mode shape in the bridge for 1.2919 Hz. Pier 1 (Evoia side) deforms along the longitudinal direction to the positive direction from bottom till the deck level. In the top there is a torsion as the one side moves to the positive and the other one to the negative side Pier 2 (Voiotia side): In the top there is a torsion as the one side the other one to the negative side. Deck deforms to the vertical direction. Right span deforms downward and middle span also. The right has larger deflection in comparison to the middle one.

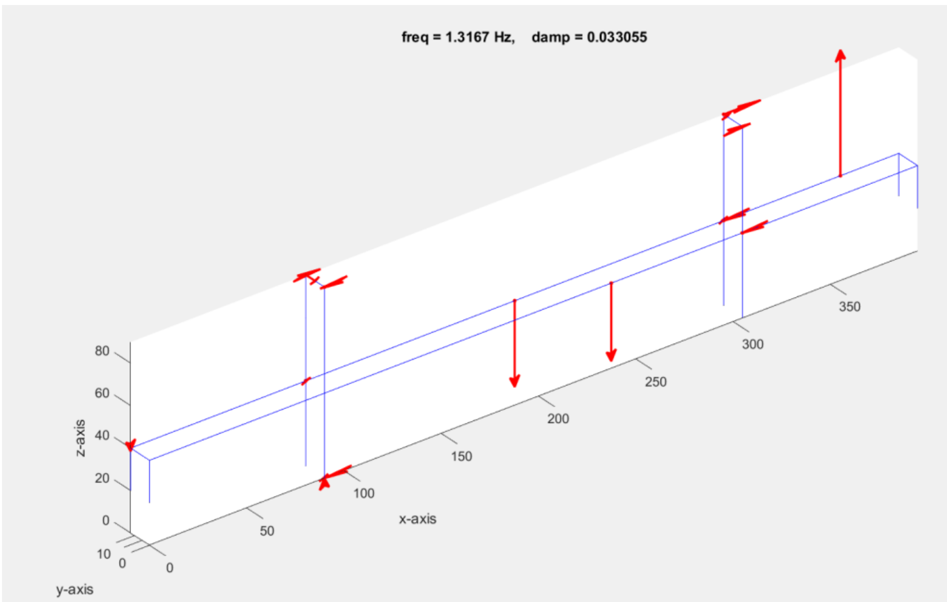

Figure 60: Mode shape in the bridge for 1.3167 Hz. Pier 1 (Evoia side) deforms along the longitudinal direction to the negative direction from bottom till the deck level and to the positive in the top. Pier 2(Voiotia side): In the top there is a torsion as the one side moves to the positive and the other one to the negative side. Deck deforms for each span along the vertical axis. Right span deforms upward, middle span deforms downward and left span is supposed to move upward but there is no vertical sensor working in this part.

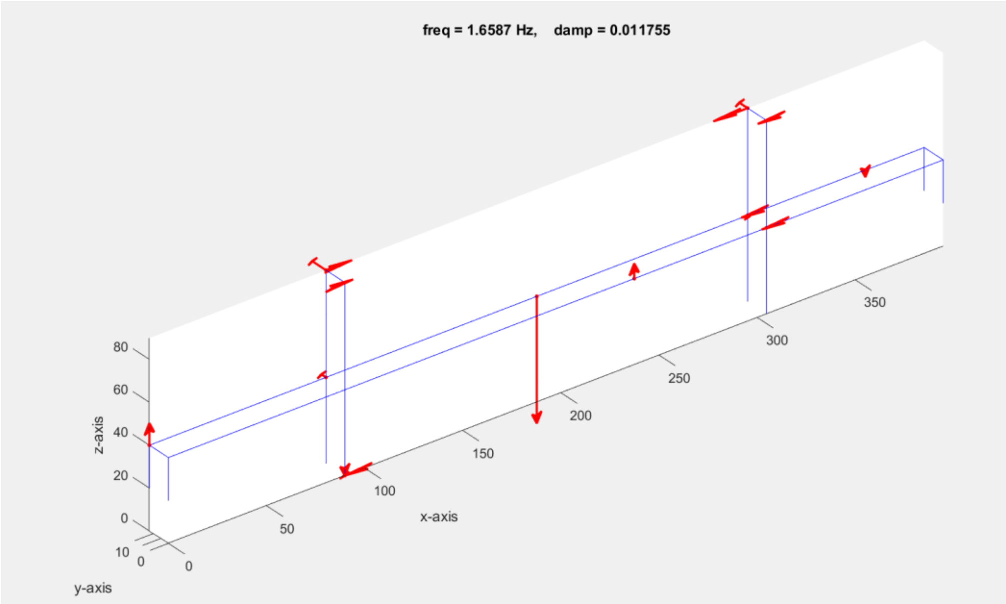

Figure 61: Mode shape in the bridge for 1.6587 Hz. Pier 1 (Evoia side) deforms in its first bending mode along longitudinal the direction to negative direction. Pier 2 (Voiotia side) deforms along its longitudinal axis to positive direction. Deck deforms for each span along the vertical axis. There is also torsion as in the middle span the one side deforms upward and the other downward.

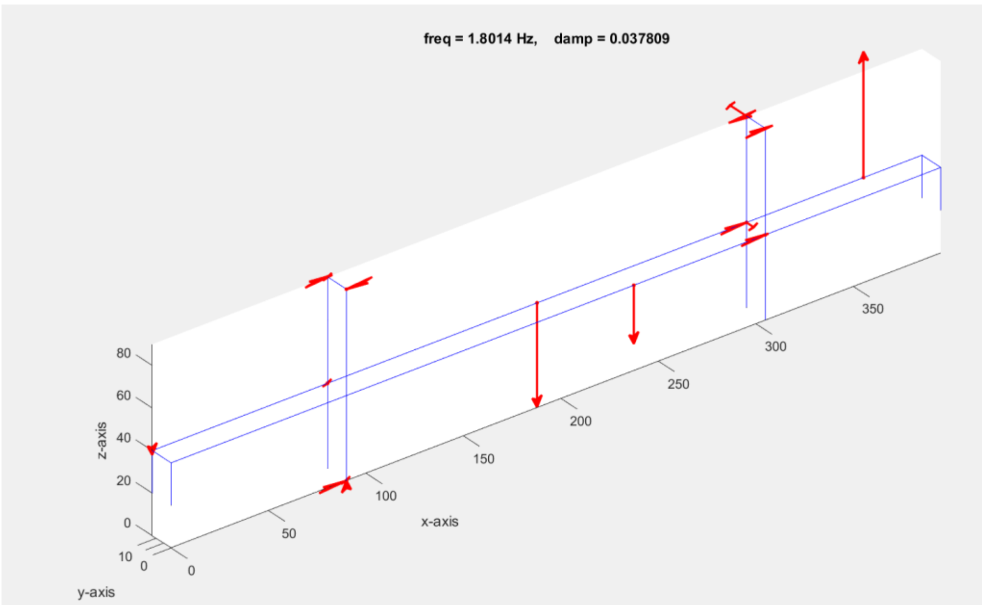

Figure 62: Mode shape in the bridge for 1.8014 Hz. Pier 1 (Evoia side) deforms to the positive direction from bottom till the deck level. In the top there is a torsion as the one side moves to the positive and the other one to the negative side. Pier 2 (Voiotia side): In the top there is a torsion as the one side moves to the positive and the other one to the negative side. Deck deforms for each span along the vertical axis. Right span deforms upward, middle span deforms downward and left span is supposed to move upward but there is no vertical sensor working in this part.

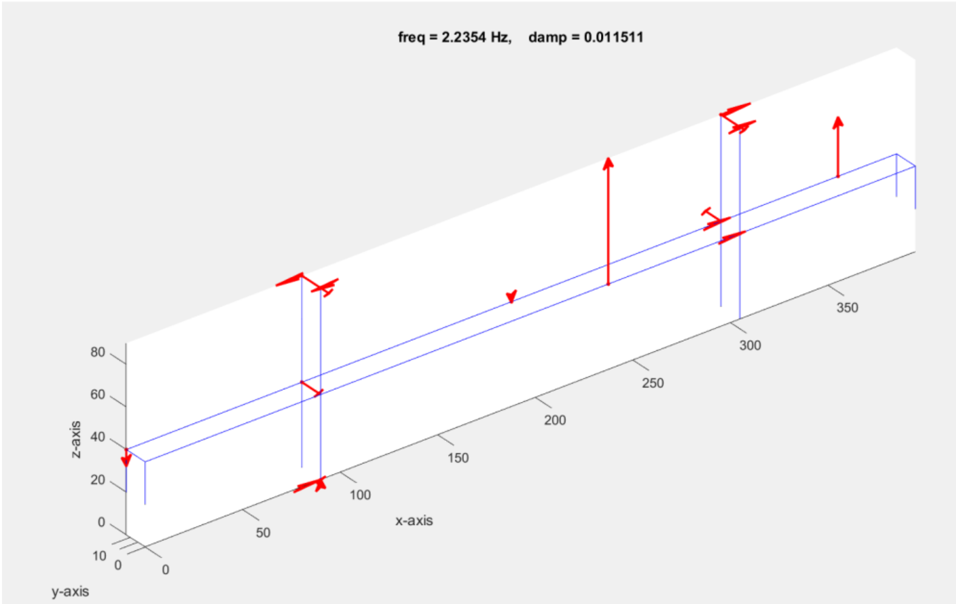

Figure 63: Mode shape in the bridge for 2.2354 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the longitudinal direction to positive direction. Pier 2 (Voiotia side) deforms along its longitudinal axis to negative direction. Deck deforms for each span along the vertical axis. There is also torsion as in the middle span the one side deforms upward and the other downward.

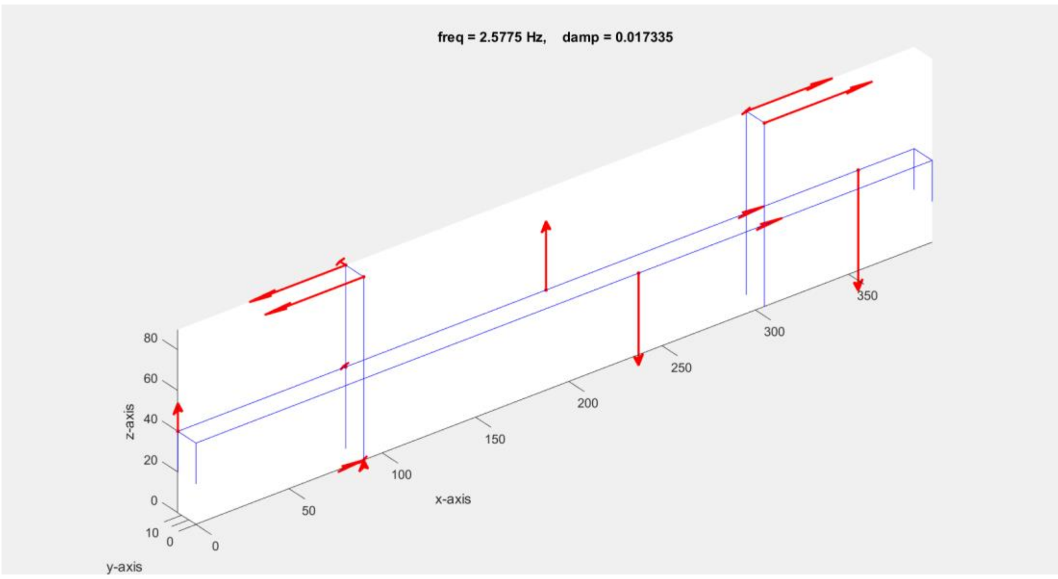

Figure 64: Mode shape in the bridge for 2.5775 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the longitudinal direction to positive direction. In deck level the deformation is smaller than in the top. Pier 2 (Voiotia side) deforms along its longitudinal axis to negative direction. Deck deforms for each span along the vertical axis. Right span deforms upward, middle span deforms downward and left span is supposed to move upward but there is no vertical sensor working in this part.

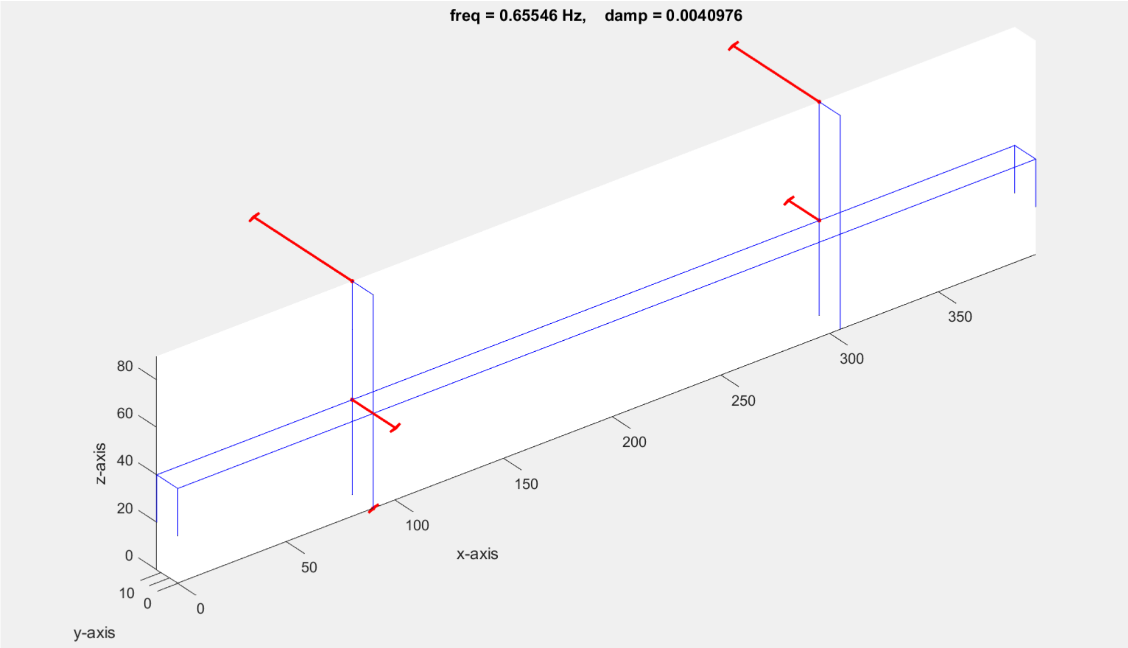

Figure 65: Additional Mode shape in the bridge for 0.65546 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the transverse direction to positive direction. In deck level the deformation is smaller than in the top. Pier 2 (Voiotia side) deforms along its transverse axis to negative direction in deck level and to the positive in the top.

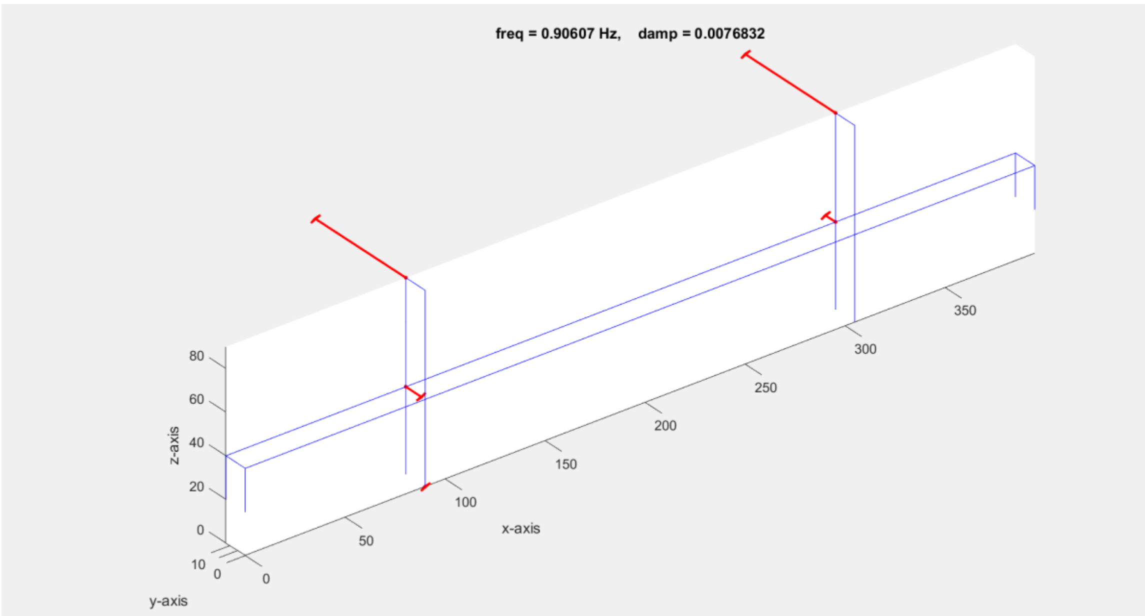

Figure 66: Additional Mode shape in the bridge for 0.90607 Hz. Pier 1 (Evoia side) deforms in its first bending mode along the transverse direction to positive direction. In deck level the deformation is smaller than in the top. Pier 2 (Voiotia side) deforms along its transverse axis to negative direction in deck level and to the positive in the top.

In figures 54-64, there is no component in the transverse direction to be activated. Probably this is because the sensors that provide measurements in the transverse direction are located on the piers of the bridge (at the deck level and at the top of the pier).

In figures 65 and 66 are presented two additional mode shapes spotted in stabilization diagrams and not recognized by SSI_COV function. There is only transverse motion because in order to identify them only transverse data was used.

In figures 54-66, there is an indicative pattern of motion of the bridge. In order to have detailed patterns more sensors are needed in all axes along Chalkida Bridge.

3.5.4 Geometry built of the bridge

A MATLAB code was generated to plot mode shapes on a bridge. The process commenced with the creation of the bridge's geometry, involving the definition of nodes distributed along the structure. Each sensor was associated with a specific node, ensuring that a one-to-one correspondence was maintained. The coordinates of these nodes were then specified in all directions in a matrix with 3 columns each one for vertical, longitudinal, and transverse direction. Following this, nodes were interconnected using vectors, and this information was subsequently input into MATLAB as a matrix detailing which nodes were connected by each vector. Additionally, sensors were defined in a matrix with four columns: the first column assigned sensor numbers, the second column indicated the node to which each sensor was affixed, the third column specified the direction of each sensor (1 for longitudinal, 2 for transverse, and 3 for vertical), and the fourth column contained values of 1 for sensors calibrated in the positive direction and -1 for sensors calibrated in the negative direction. So, an input file was generated with these three matrices which defines full the geometry of the bridge. Then another code was generated which loads eigenfrequencies mode shapes matrix and damping ratios and then plots mode shapes in the bridge for all eigenfrequencies. A complete code for this process can be found in Appendix (Chapter 6).

4 Verification of the obtained results

In order to verify the validity of the results, the time period was divided into two segments, and the aforementioned procedures were repeated. The data spanned a duration of 30 minutes and was split into two equal time intervals. The analysis was applied to both segments independently.

By conducting the analysis on separate segments of the data, it ensured that the obtained results were consistent and not influenced by any temporal variations or transient effects that might be present in the bridge response. This approach provided a robust validation of the results, as any significant discrepancies or inconsistencies between the two segments would indicate potential issues or uncertainties in the findings.

The repetition of the analysis on three distinct time segments allowed for a comprehensive assessment of the stability and reliability of the identified eigenfrequencies, modal damping ratios, and mode shapes. The consistency of the results across the two segments provided confidence in the accuracy and reproducibility of the findings, further strengthening the overall validity of the research outcomes.

By adopting this rigorous validation approach, the study ensured that the results were not biased or influenced by specific temporal characteristics of the bridge vibrations. It enhanced the credibility and confidence in the findings, supporting the conclusion that the identified modal parameters accurately represent the dynamic behavior of the bridge under ambient vibration conditions.

In table 6, all eigenfrequencies from different segments are shown in the table. Same row doesn't mean any similarity between eigenfrequencies. Similarity is provided in the following pages.

Fn(Hz) 0-30 min	Fn(Hz) 0-15 min	Fn(Hz) 15-30 min
0,4354	0,4371	0,4341
0,5227	0,5246	0,5216
0,8458	0,8452	0,9855
0,9782	0,9717	1,2867
1,2851	1,2916	1,3078
1,2919	1,3328	1,3454
1,3167	1,6577	1,6103
1,6587	2,2307	1,7787
1,8014	2,2325	2,2397
2,2354	2,5941	2,5819
2,5775	5,58	2,6039
5,5679		5,58

Table 6: eigenfrequencies for different time segment

We can ascertain that the frequencies observed from the frequency diagrams, along with those discerned through the subspace identification method, are corroborated by each time segment, thereby reflecting the reality accurately.

The Modal Assurance Criterion (MAC) has been used to verify the similarity among the mode shapes. We implemented a custom MATLAB code for the MAC calculation, and the results are presented in the figures 67-69. This table displays the MAC values for each pair of mode shapes.

More precisely, the mode shapes are compared, and a matrix is created using the earlier mentioned formula. In practice, two mode shapes are compared. The first row and first column compare the mode shape derived from the first eigenfrequency of the first segment, with the mode shape from the first eigenfrequency derived from the second segment. Hence, on the diagonal, mode shapes from two segments with similar eigenfrequencies are compared. This implies that values on the diagonal should be close to 1.

In Figures 67, 68 and 69 are present the results derived from the application of the Modal Assurance Criterion (MAC) [8] to our system of interest between (0-15 and 15-30), (15-30 and 0-30) and (0-15 and 0-30) minutes correspondingly. They are 3D diagrams, in one axis are frequencies from the first set and in the second are frequencies from the second set. In the Vertical axis is the value of MAC number. To check the similarity in all sets and verify the results mac number between similar frequencies should be close to 1 which means mac along diagonal or close to it have to be close to 1. It can be clearly seen, that the values close to 1, particularly prominent along the diagonal, attest to the robustness and precision of the system under scrutiny.

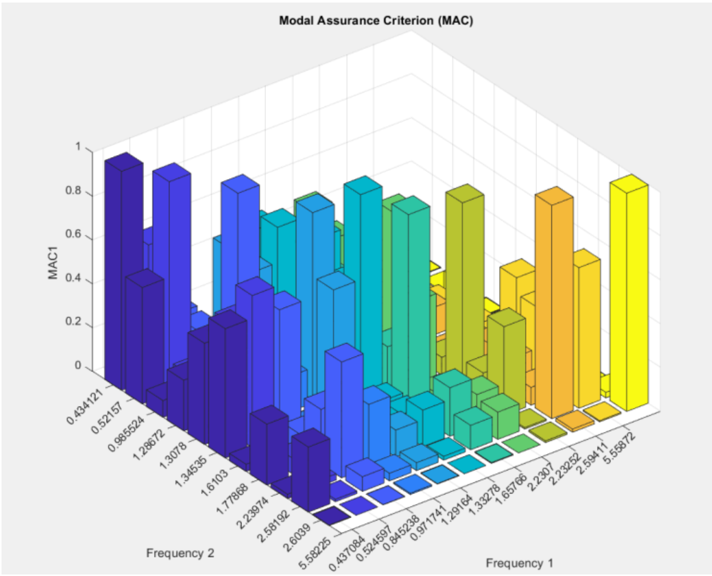

Figure 67: MAC values between 0-15 and 15-30 min

Fn(Hz)	0.4341	0.5216	0.9855	1.2867	1.3078	1.3454	1.6103	1.7787	2.2397	2.5819	2.6039	5.58
0.4371	0.999619	0.616088	0.252982	0.003317	0.457164	0.499024	0.007863	0.489198	0.009974	0.276318	0.103017	0.003028
0.5246	0.530452	0.966274	0.243925	0.010725	0.407588	0.568483	0.055111	0.417894	0.00593	0.006576	0.020251	2.77E+08
0.8452	0.07637	0.146507	0.003756	0.366311	0.09379	0.053744	0.362877	0.121333	0.416341	0.070322	0.045784	0.09844
0.9717	0.232431	0.244939	0.990148	0.327044	0.018904	0.644844	0.129859	9.29E+09	0.099573	0.125127	0.011466	0.002562
1.2916	0.462269	0.506256	0.103738	0.160804	0.868019	0.489163	0.032527	0.725101	0.009673	0.006518	0.070119	0.017341
1.3328	0.589705	0.694068	0.580477	0.001679	0.578632	0.964368	0.223827	0.381775	0.080725	0.006228	0.091137	0.003219
1.6577	0.033148	0.001429	0.075121	0.058577	0.002191	0.083043	0.888073	0.028164	0.847034	0.156524	0.410723	0.000198
2.2307	0.277667	0.071454	0.251269	0.18668	0.028237	0.085746	0.004867	0.00583	0.133824	0.148951	0.335575	0.011145
2.2325	0.019831	0.028095	0.527967	0.286267	0.12306	0.166797	0.222823	0.142863	0.407064	0.080881	8.64E+08	0.001325
2.5941	0.296802	0.011027	0.06687	0.035086	0.036038	0.017638	0.112109	0.124633	0.0069	0.973773	0.642329	0.025225
5,588	0.002746	9.54E+09	0.001353	0.005928	0.002203	2.84E+08	0.00376	7.92E+09	0.007933	0.014343	0.00528	0.996032

Table 7: MAC values between 0-15 and 15-30 min

Fn(Hz) 0-15 min	Fn(Hz) 15-30 min
0.4371	0.4341
0.5246	0.5216
0.8452	
0.9717	0.9855
	1.2867
1.2916	1.3078
1.3328	1.3454
1.6577	1.6103
	1.7787
	2.2397
2.2307	
2.2325	
2.5941	2.5819
	2.6039
5,588	5.58

Table 8: similarity in frequencies between 0-15 and 15-30 min

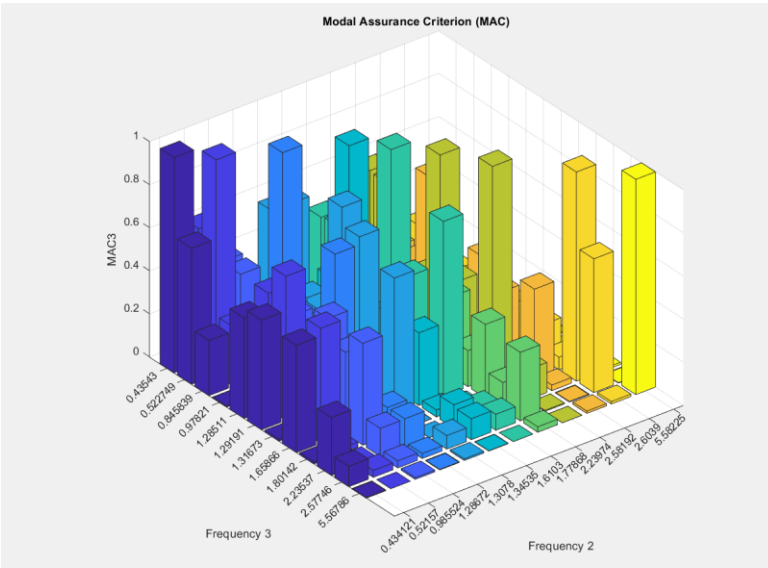

Figure 68: MAC values between 0-30 and 15-30 min

Fn(Hz)	0.4341	0.5216	0.9855	1.2867	1.3078	1.3454	1.6103	1.7787	2.2397	2.5819	2.6039	5.58
0.4354	0.999978	0.625381	0.431005	0.236143	0.576937	0.000565	0.459699	0.426862	0.589803	0.190308	0.245685	0.003812
0.5227	0.631471	0.998283	0.423051	0.26961	0.662306	0.009516	0.468812	0.174606	0.582995	0.179066	0.005639	5.86E+09
0.8458	0.255564	0.274011	0.072636	0.997348	0.279668	0.328326	0.041373	0.074051	0.021631	0.621903	0.145088	0.000955
0.9782	0.004049	0.033302	0.307659	0.282898	0.03988	0.994643	0.273209	0.10066	0.262499	0.094861	0.061833	0.004023
1.2851	0.467276	0.537215	0.390074	0.026599	0.805743	0.268926	0.986044	0.095478	0.873518	0.00026	0.054438	0.000525
1.2919	0.509519	0.66932	0.363301	0.684498	0.72116	0.012554	0.450107	0.07335	0.338546	0.411399	0.043838	4.11E+09
1.3167	0.009304	0.1031	0.48277	0.171243	0.048585	0.087938	0.069984	0.362642	0.008349	0.296185	0.202885	0.002669
1.6587	0.498109	0.533615	0.378058	0.000975	0.635694	0.337546	0.811287	0.167388	0.979508	2.69E+08	0.154587	0.000148
1.8014	0.010476	0.023242	0.473064	0.130883	0.001195	0.032328	0.000557	0.341924	0.009729	0.418354	0.058616	0.003881
2.2354	0.268308	0.012949	0.130582	0.131624	0.000299	0.108781	0.055425	0.125033	0.139848	0.025737	0.972008	0.009832
2.5775	0.094891	0.03211	0.032916	0.023269	0.062448	0.09755	0.074051	0.318199	0.004726	0.002624	0.621395	0.002267
5.5679	0.002865	0.000396	0.008033	0.002484	0.007471	0.001529	0.000276	0.025386	0.000426	0.01073	0.011476	0.999883

Table 9: MAC values between 0-30 and 15-30 min

Fn(Hz) 0-30 min	Fn(Hz) 15-30 min
0.4354	0.4341
0.5227	0.5216
0.8458	
0.9782	
	0.9855
1.2851	1.3078
1.2919	1.2867
1.3167	
1.2919	1.3078
	1.3454
1.6587	1.6103
1.8014	
	1.7787
	2.2397
	2.5819
2.2354 2.5775	2.6039
5.5679	5.58

Table 10: similarity in frequencies between 0-30 and 15-30 min

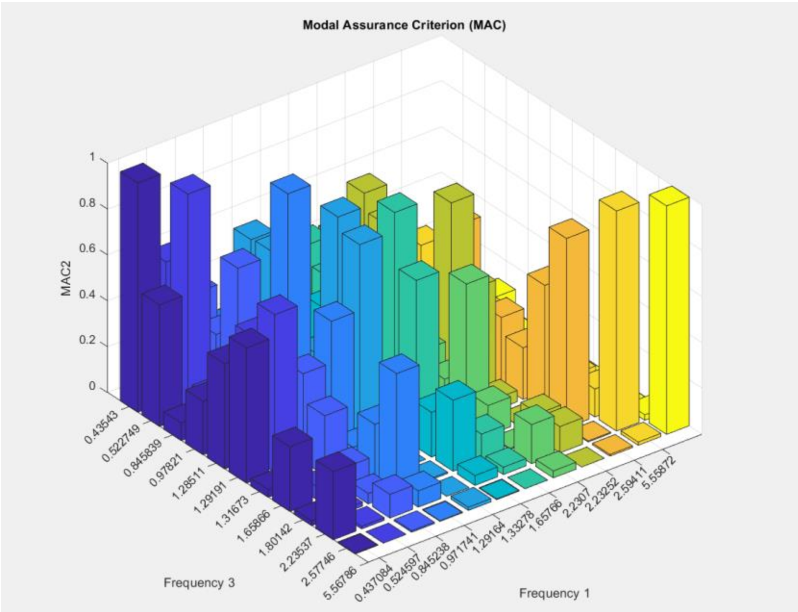

Figure 69: MAC values between 0-15 and 0-30 min

Fn(Hz)	0.4354	0.5227	0.8458	0.9782	1.2851	1.2919	1.3167	1.6587	1.8014	2.2354	2.5775	5.5679
0.4371	0.999728	0.607948	0.42323	0.235783	0.562921	0.000212	0.446169	0.432696	0.575938	0.191951	0.253703	0.00412
0.5246	0.529315	0.964822	0.304619	0.211566	0.570452	0.002435	0.362945	0.19887	0.500001	0.132866	0.003838	5.40E+09
0.8452	0.075999	0.144351	0.655839	0.008763	0.016304	0.343404	0.069613	0.085375	0.078607	0.110249	0.01978	0.100054
0.9717	0.234338	0.220408	0.040826	0.99555	0.252806	0.382681	0.028502	0.086236	0.013066	0.573178	0.138199	0.002248
1.2916	0.458639	0.529799	0.246396	0.099552	0.908883	0.121491	0.833166	0.18567	0.780337	0.004603	0.007531	0.019089
1.3328	0.588531	0.687032	0.382341	0.562086	0.850451	0.000537	0.599987	0.120758	0.468821	0.295203	0.014036	0.004077
1.6577	0.033188	0.000582	0.260622	0.075513	0.007471	0.052485	0.001118	0.597642	0.048903	0.225275	0.287423	0.000156
2.2307	0.281034	0.05644	0.100517	0.238324	5.23E+09	0.194599	0.028348	0.12876	0.000641	0.560825	0.122295	0.010226
2.2325	0.020747	0.017369	0.045902	0.519741	0.006457	0.308162	0.11442	0.005439	0.108864	0.825711	0.121979	0.001304
2.5941	0.297165	0.011435	0.102795	0.071432	3.03E+09	0.042953	0.031828	0.170729	0.11449	0.004599	0.959658	0.023383
5,588	0.002827	0.000188	0.006338	0.002693	0.01405	0.002864	0.000233	0.030167	4.75E+09	0.007198	0.013871	0.995934

Table 11: MAC values between 0-30 and 0-15 min

Fn(Hz) 0-15 min	Fn(Hz) 0-30 min
0.4371	0.4354
0.5246	0.5227
0.8452	0.8458
0.9717	0.9782
1.2916 1.3328	1.2851
	1,2919 1,3167
1.6577	1,6587
	1,8014
2.2307 2.2325	2.2354
2.5941	2.5775
5,588	5.5679

Table 12: similarity in frequencies between 0-30 and 0-15 min

To be more precise with MAC criterion it is crucial in a structure to have more sensors because a smaller number of sensors could lead to a similarity between mode shapes which may not correspond to reality.

The MAC analysis plays a crucial role in ensuring the reliability and quality of mode shapes in practical applications. It allows comparison of calculated mode shapes with reference mode shapes or between different analysis methods. The obtained MAC values help evaluate the consistency and accuracy of the results. Higher MAC values indicate a stronger resemblance and confidence in the identified mode shapes, while lower values suggest discrepancies or variations in the captured structural behavior.

In the context of vibration analysis, the MAC is often visualized through MAC matrices or contour plots. These visual representations provide a comprehensive understanding of the spatial agreement or differences between mode shapes. They assist in identifying localized structural behavior and potential uncertainties associated with mode shapes.

The application of the MAC analysis offers a reliable means to verify the accuracy and consistency of mode shapes, facilitating a deeper understanding of the dynamic characteristics of the analyzed structure. It serves as an essential tool in vibration analysis, enabling effective comparison, interpretation, and decision-making in various engineering applications, including structural health monitoring, modal parameter identification, and structural optimization for enhanced performance.

5. Summary

A MATLAB code was developed as a critical component of this thesis to determine the modal properties of the cable bridge of Euripus in Chalkida. This endeavor involved the collection of ambient data in the time domain, followed by the application of Welch's method to transition into the frequency domain. Subsequently, frequency domain diagrams were generated, allowing for the identification of natural frequencies based on the peaks observed within these diagrams.

Following this initial analysis, the SSI_COV function was utilized to automatically extract the modal properties of the bridge. This encompassed the determination of mode shapes, eigenfrequencies, and damping ratios. Also, MAC was used to check the similarity between mode shapes. After employing the SSI_COV method, we successfully unveiled the modal parameters of the bridge. Within the frequency range of 0 to the 5 Hz, we identified 11 eigenfrequencies and corresponding mode shapes. The damping ratios associated with these modes fall within the range of 1 to 3 percent. However, as the frequency increases beyond 5 Hz, we encountered 12 additional eigenfrequencies. Unfortunately, it remains uncertain whether these are genuine modes or spurious responses due to the limited sensor density. Furthermore, the mode types for these higher frequency modes could not be conclusively identified from the available mode shapes.

Upon visualizing these mode shapes on the bridge structure, it became evident that Pier 1 (Evoia) exhibits torsional behavior at 1.29Hz and 1.8014Hz. Similarly, Pier 2 also exhibits torsion in its top section at frequencies of 1.229Hz, 1.3167Hz, and 1.8014Hz. In a general sense, both piers exhibit movement along their longitudinal axes. The deck, on the other hand, displays vertical deformation for frequencies of 0.8458Hz, 1.2919Hz, 1.6587Hz, 2.23Hz, and 2.5775Hz. Notably, transverse deformation was not activated as sensors were exclusively placed within the piers.

In our effort to enhance the reliability of the modal analysis, we divided the 30-minute data into two segments, each spanning 15 minutes. Modal parameters were identified for both segments, and a comparison using the Modal Assurance Criterion (MAC) indicated a high degree of similarity between them. Nevertheless, it is imperative to acknowledge that a more extensive sensor network is essential to ensure the precision of the MAC criterion. This is particularly crucial because some mode shapes may appear similar but may in fact be distinct.

Furthermore, it is important to emphasize that this code's versatility extends beyond the specific application to the bridge under investigation. Users can customize the input file, specifying details such as the number and positioning of sensors, among other parameters. Consequently, this code can be adapted for the analysis of a diverse array of structures.

The MATLAB code developed and described herein constitutes a tool for modal analysis. Its ability to reliably ascertain modal properties and its adaptability to various structural scenarios make it a pertinent contribution to this area of research. In conclusion, to bolster the robustness of the SSI_COV method applied to the Chalkida bridge, the installation of additional sensors is imperative. This will not only facilitate a more comprehensive analysis but also enhance the overall reliability of the results.

6 Appendix

6.1 Input File

Initially, an input file is generated, allowing the user to specify various parameters essential for the main code. These parameters encompass the number of sensors employed in the analysis, the precise positioning of each sensor along the bridge structure, and even the specific date associated with the data under examination. This user-defined input file serves to customize and configure the analysis according to the specific requirements of the study. By providing such flexibility, the code can accommodate different scenarios and datasets, ensuring a versatile and adaptable framework for the analysis of bridge vibrations.

```
%-------------------------------------------------------
%HERE IS THE INPUT FILE WITH ALL VARIABLES FOR THE MAIN CODE
%IN THE MAIN CODE THIS FILE NA,ED INPUTS IS CALLED
%--------------------------------------------------------



% We give the file name according to the date
 Sdate = '11.20.2022';

parameterRange = '15-30t'; % specify the name where the results (mode shapes and
eigenfrequencies) should be saved in the input files
startmin = 15; % Specify the starting time you want to read from text files
endmin = 30.0165; % Specify the ending time you want to read

s = 1; % if we have ambient data, set it to 1 to apply the Welch's method, otherwise set it to
0 for FFT for earthquake data

ActiveSensors = [8 10 12 18 27]; % These are the sensor numbers to be analyzed

% Categorize the sensors based on their address.
vert = [];
long = [];
trans = [8 10 12 18 27];

%Set the frequencies I want to estimate the peaks for
freqMin = 0; % set minimum frequency
freqMax = 4; % set maximum frequency

%Set the frequency domain for the stabilization diagrams
xmin = 0; %set minimum frequency
xmax = 4; % set maximum frequency


Vertical = find(ismember(ActiveSensors, vert));
Longitudinal = find(ismember(ActiveSensors, long));
Transverse = find(ismember(ActiveSensors, trans));

% Here I'll categorize the sensors by color based on their address to plot the peaks for each
sensor
green_cols = Vertical;
red_cols = Longitudinal;
blue_cols = Transverse;

%the number of segments for welch's method
NofS=40;

%macthreshlod used for checking the similarity between mode shapes
macThreshold = 0.8;

%parameters for SSI_COV method
Ts=10;
Nmin=3;
Nmax=80;
```

6.2 Main code

This code implements a comprehensive analysis of accelerometer data in the context of my thesis. It processes data from multiple files, performs data manipulation, and visualizes the accelerometer readings over time. It utilizes Welch's method, a technique for estimating power spectral density, to analyze the frequency content of the data. By applying Welch's method with specific parameters such as window length and overlap, the code calculates the power spectral density estimate for each accelerometer channel. The resulting power spectral density values are then plotted against frequency to visualize the frequency distribution of the data. The code also extracts peak values and frequencies, providing insights into the significant features present in the data. Additionally, it applies the SSI_COV method to identify modal properties such as modal frequencies, damping ratios, and mode shapes. These modal properties are crucial for understanding the dynamic behavior of the system under investigation. Overall, this code plays a vital role in extracting meaningful information from accelerometer data, contributing to the analysis and interpretation of the system's modal characteristics, which are essential for this thesis research.

```
clear all
close all
clc;
run('inputs.m'); % the input file wit all parameters is called
data=[];

%  ---------------------------------------------------------------------------------
%     in this for loop opens the file with the name of the date which has
%     each txt file for every channel-sensors.Each txt file has four
%     columns first for time second for accelration third for velocity and
%     fourth for distance . in first row is time date and in second the
%     names(acceleration time distance velocity) so in this loop it takes
%     all txt files delete two first rows and creates two matricles one
%     with acceleration data from sensors  and one with time
% ----------------------------------------------------------------------------------
for i = 1:length(ActiveSensors)
    filename = strcat('C:\Users\Nikos\Desktop\DIPLOMATIKI\CHALKIDA\uniofthessaly\',...
    Sdate,'\CH',num2str(ActiveSensors(i)),'.',Sdate,'.TXT');
    fid = fopen(filename,'r');
    dataCell = textscan(fid,'%f %f %f %f','headerlines',2);
    fclose(fid);

     % Extract the desired range of rows
    a = cellfun(@(x) x((startmin*12000+1):(endmin*12000+1)), dataCell, 'UniformOutput',
false);

    data = [data, a{2}]; % the matrix with accleration data
    time = [a{1}];      %the matrix with time
end


data= data / 980; % CONVERTION OF ACCELERATION FROM CM/SEC^2 TO g
save([Sdate, '.mat'], 'data', 'time'); % matricles with acceleration data and and time are
saved


% plot the acceleration time histories of the vertical channels in direction from evoia to
voiotia
if ~isempty(Vertical)
  figure;
  sgtitle('Vertical accelerometers (g) vs Time (sec)')
  numPlots1 = numel(Vertical); % Determine the number of plots
  numRows1 = ceil(sqrt(numPlots1)); % Calculate the number of rows needed
  numCols1 = ceil(numPlots1/numRows1); % Calculate the number of columns needed
  for i = 1:numPlots1
      subplot(numRows1, numCols1, i)
      plot(time(:,1), data(:,Vertical(i)))
      title(['CH', num2str(vert(i))])
      xlabel('(sec)')
      ylabel('(g)')
```

```
      legend('Vertical Accelerometers (g) vs Time (sec)', 'FontSize', 7, 'Location', 'Best')
  end
end


% plot the acceleration time histories of the longitudinal channels in direction from evoia to
voiotia
if ~isempty(Longitudinal)
   figure;
   sgtitle('longitudinal accelerometers(g) vs Time(sec)')
   numPlots2 = numel(Longitudinal); % Determine the number of plots
   numRows2 = ceil(sqrt(numPlots2)); % Calculate the number of rows needed
   numCols2 = ceil(numPlots2/numRows2); % Calculate the number of columns needed
  for i = 1:numPlots2
       subplot(numRows2, numCols2, i)
       plot(time(:,1),data(:,Longitudinal(i)))
       title(['CH', num2str(long(i))])
       xlabel('(sec)')
       ylabel('(g)')
       legend('longitudinal accelerometers(g) vs Time(sec)','FontSize', 7, 'Location', 'Best')
  end
end


% plot the acceleration time histories of the Transverse channels in direction from evoia to
voiotia
if ~isempty(Transverse)
    figure;
    sgtitle('Transverse accelerometers(g) vs Time(sec)')
    numPlots3 = numel(Transverse); % Determine the number of plots
    numRows3 = ceil(sqrt(numPlots3)); % Calculate the number of rows needed
    numCols3 = ceil(numPlots3/numRows3); % Calculate the number of columns needed
  for i = 1:numPlots3
    subplot(numRows3, numCols3, i)
    plot(time(:,1),data(:,Transverse(i)))
    title(['CH', num2str(trans(i))])
    xlabel('(sec)')
    ylabel('(g)')
    legend('Transverse (g) vs Time (sec)', 'Units', 'normalized','FontSize', 7, 'Location',
'Best')
  end
end

% -----------------------------------------------------------------------------------------------
-------
% computes fft or psd analysis
% -----------------------------------------------------------------------------------------------
-------

dt = median(diff(time)); % get time step
fs = 1/(dt);
N = length(data); % length of signal
f = fs*(0:N-1)/N; % frequency vector
R=[];

  if s==0

     % Apply FFT to each column of acceleration data
     for i = 1:size(data, 2)

     % Select ith column of acceleration data
     x = data(:,i);

     % Compute FFT
     P = abs(fft(x));

     R=[R,P];
     end


   %plot fft data the vertical channels in direction from evoia to voiotia
 if ~isempty(Vertical)
     figure;
     sgtitle('Vertical accelerometers Magnitude vs Frequency')
    for i = 1:numPlots1
      subplot(numRows1, numCols1, i)
```

```
      plot(f,R(:,Vertical(i)))
      title(['CH', num2str(vert(i))])
      xlabel('Hz')
      ylabel('aplitude')
      xlim([freqMin freqMax]);
      legend('Vertical accelerometers Magnitude vs Frequency','FontSize', 7, 'Location',
'Best')
    end
 end


   % plot fft data longitudinal channels in direction from evoia to voiotia
if ~isempty(Longitudinal)
  figure;
  sgtitle('longitudinal accelerometers Magnitude vs Frequency')
  for i = 1:numPlots2
      subplot(numRows2, numCols2, i)
      plot(f,R(:,Longitudinal(i)))
      title(['CH', num2str(long(i))])
      xlabel('Hz')
      ylabel('aplitude')
      xlim([freqMin freqMax]);
      legend('longitudinal accelerometers Magnitude vs Frequency','FontSize', 7, 'Location',
'Best')
    end
end

%plot fft data Transverse channels in direction from evoia to voiotia
if ~isempty(Transverse)
  figure;
  sgtitle('Transverse accelerometers Magnitude vs Frequency')
  for i = 1:numPlots3
      subplot(numRows3, numCols3, i)
      plot(f,R(:,Transverse(i)))
      title(['CH', num2str(trans(i))])
      xlabel('Hz')
      ylabel('aplitude')
      xlim([freqMin freqMax]);
      legend('Transverse accelerometers Magnitude vs Frequency','FontSize', 7, 'Location',
'Best')
  end
end


  elseif s==1

% Choose Welch's method parameters


window_length = round(N/NofS);
window = hann(window_length); % Window function
overlap = window_length/2; % Amount of overlap between adjacent segments
B_all=[];
f_peak=[];
for i = 1:size(data, 2)
    % Calculate the power spectral density estimate using pwelch
    [B,f] = pwelch(data(:,i),window, overlap, [], fs);
    Q(:,i)= B ;
    R = log10(Q);
    [Q_peak, index_peak] = max(Q);
    f_peak = f(index_peak);
    f_peak=transpose(f_peak);
    a_peak_g(i) = sqrt((Q_peak(i)* f_peak(i)));

end

%plot  vertical channels in direction from evoia to voiotia
if ~isempty(Vertical)
  figure;
   sgtitle('Vertical acclerometers PSD(G^2/Hz) vs Frequency (Hz)')
   for i = 1:numPlots1
     subplot(numRows1, numCols1, i)
      plot(f,R(:,Vertical(i)))
      title(['CH', num2str(vert(i))])
      xlabel('Hz')
      ylabel('log10(Acc)')
      xlim([freqMin freqMax]);
```

```
     legend('Vertical acclerometers PSD(G^2/Hz) vs Frequency (Hz)','FontSize', 7, 'Location',
'Best')

  end
end

% plot longitudinal channels in direction from evoia to voiotia
if ~isempty(Longitudinal)
   figure;
   sgtitle('longitudinal accelerometers PSD(G^2/Hz) vs Frequency (Hz)')
   for i = 1:numPlots2
     subplot(numRows2, numCols2, i)
     plot(f,R(:,Longitudinal(i)))
     title(['CH', num2str(long(i))])
     xlabel('Hz')
     ylabel('log10(Acc)')
     xlim([freqMin freqMax]);
     legend('longitudinal accelerometers PSD(G^2/Hz) vs Frequency (Hz)','FontSize', 7,
'Location', 'Best')

   end
end

% plot Transverse channels in direction from evoia to voiotia
if ~isempty(Transverse)
    figure;
    sgtitle('Transverse accelerometers PSD(G^2/Hz) vs Frequency (Hz)')
    for i = 1:numPlots3
     subplot(numRows3, numCols3, i)
     plot(f,R(:,Transverse(i)))
     title(['CH', num2str(trans(i))])
     xlabel('Hz')
     ylabel('log10(Acc)')
     xlim([freqMin freqMax]);
     legend('Transverse accelerometers PSD(G^2/Hz','FontSize', 7, 'Location', 'Best')

    end
end



% Extract the data for the selected columns
green_data = a_peak_g(:,green_cols);
red_data = a_peak_g(:,red_cols);
blue_data = a_peak_g(:,blue_cols);


% Create a new figure and plot the data with the specified colors
figure;
hold on;
if ~isempty(green_data)
    stem(vert, green_data, 'g', 'Marker', 'none', 'LineWidth', 5);
end
if ~isempty(red_data)
    stem(long, red_data, 'r', 'Marker', 'none', 'LineWidth', 5);
end
if ~isempty(blue_data)
    stem(trans, blue_data, 'b', 'Marker', 'none', 'LineWidth', 5);
end
hold off;

% Add labels and a legend
xlabel('sensor Number');
ylabel('g');
legend('vertical', 'longitudinal', 'transverse');
end


%-----------------------------------------------------------------------------------------------
----------
%LOAD IMAGE WITH ALL SENSORS IN THE BRIDGE
%-----------------------------------------------------------------------------------------------
----------
% Specify the path and filename of the PNG image
filename = 'C:\Users\Nikos\Desktop\DIPLOMATIKI\sensors.png';
% Use the imread function to read the image data
img_data = imread(filename);
% Display the image in a figure window
```

```
figure;
imshow(img_data);

%----------------------------------------------------------------------------------------------
-----------
%ssi_cov FUNCTION to identify modal properties
%----------------------------------------------------------------------------------------------
-----------
SSI_COVDATA=transpose(data);
tic
[fn01,zeta0,phi0,paraPlot] = SSI_COV(SSI_COVDATA,dt,'Ts',Ts,'Nmin',Nmin,'Nmax',Nmax);
toc
%results are: fn01=frequencies, phi0 mode shapes zeta0= damping ratios ,
%paraplot=structure data useful for stabilization diagram

%----------------------------------------------------------------------------------------------
---------------
%Apply mac criterion to mode shapes to check the similarity between them
%Then  group all mode shapes with similarity and create a matrix with
%frequencies , mode shapes and damping ratios with no similarity
%----------------------------------------------------------------------------------------------
--------------


% Step 1: Normalize the mode shapes
modes_norm = sqrt(sum(phi0.^2, 2));  % Compute the Euclidean norm for each row (mode shape) of
modes3
modes_normalized = phi0 ./ modes_norm;  % Normalize each row (mode shape) of modes3


numModes = size(modes_normalized, 1);  % Number of mode shapes in modes3
groupedModeShapes = cell(0);  % Initialize an empty cell array for grouped mode shapes
groupedFrequencies = zeros(0); % Initialize an empty array for grouped frequencies
groupedDampingRatios=zeros(0); %initialize empty array for grouped Damping ratios

%in this while loop each mode shape is compared with the next one for
%similarity based on mac criterion . if mac is over a choosen number then
%mode shapes are considered the same. This continues till mac number below
%choosen number is find which indicates no similarity
i = 1;  % Initialize the index for the first mode shape
while i <= numModes
    mode1 = modes_normalized(i, :);
    group = phi0(i,:);  % Initialize a group with the current mode shape
    group_ratios = zeta0(i); % Initialize a group with the current damping ratios
    freqGroup = fn01(i); % Initialize a group with the current frequency

    j = i + 1;  % Initialize the index for the next unchecked mode shape
    while j <= numModes
        mode2 = modes_normalized(j, :);

        %calculating MAC number between mode shapes
       mac = abs(mode1 * mode2')^2 / ((mode1 * mode2')) * (mode2 * mode1');

        if mac >= macThreshold
            group = [group; phi0(j,:)];  % Add mode2 to the group
            group_ratios = [group_ratios; zeta0(j)];% Initialize a group with the current mode
shape
            freqGroup = [freqGroup; fn01(j)]; % Add the corresponding frequency
            i = j;  % Update the current index
            j = i + 1;  % Move to the next unchecked mode shape
        else
           break;  % Exit the inner loop if MAC is less than 0.8
        end
    end

    % Save the group in the cell array
    groupedModeShapes{end + 1} = group;
    groupedFrequencies{end + 1} = freqGroup;
    groupedDampingRatios{end + 1}=group_ratios;
    % Move to the next unchecked mode shape
    i = i + 1;
end

% Combine the first mode shape from each group into a new matrix
numGroups = length(groupedModeShapes);
finalModeShapes = zeros(numGroups, size(phi0, 2));  % Initialize the final matrix
finalFrequencies = zeros(1,numGroups); % Initialize the final frequencies array
```

```
finalDampingRatios=zeros(1,numGroups);%initialize the final damping ratios array
for k = 1:numGroups
    finalModeShapes(k,:)= groupedModeShapes{k}(1, :);  % Take the first mode shape from each
group
    finalFrequencies(k) = groupedFrequencies{k}(1); % Take the frequency corresponding to the
first mode shape
    finalDampingRatios(k)=groupedDampingRatios{k}(1);% Take the ratio corresponding to the
first mode shape
end

% Saving phi0 and fn01 in a MAT file
filename = sprintf('%s.mat', parameterRange);
save(filename, 'phi0',
'fn01','zeta0','finalModeShapes','finalFrequencies','finalDampingRatios');


%----------------------------------------------------------------------------------------------
-------------
%PLOT STABILIZATION DIAGRAMS
%----------------------------------------------------------------------------------------------
-------------

for j = 1:length(ActiveSensors)
    name = ActiveSensors(j);  % Get the name from the 'l' row

    [h] = plotStabDiag(paraPlot.fn, SSI_COVDATA(j,:), fs, paraPlot.status, paraPlot.Nmin,
paraPlot.Nmax);

    % Set the figure window title to the number 'j'
    set(h, 'NumberTitle', 'off', 'Name', num2str(name));
    % Set x-axis limits
    xlim([xmin,xmax]);
    % Add a common title to each plot

end
```

6.3 Modal Assurance criterion

The code performs modal analysis using the Modal Assurance Criterion (MAC) to compare the mode shapes and eigenfrequencies of two sets of data obtained from different sources. It loads eigenfrequencies and mode shapes from two MAT files: '0-15.mat' and '15-30.mat'. The code sets a tolerance value to identify similar eigenfrequencies within a certain range of difference. It then identifies similar eigenfrequencies between the datasets and removes them along with the corresponding mode shapes. After filtering eigenfrequencies below 5 Hz, the code proceeds with the analysis.

Normalization of the mode shapes is performed to ensure consistent scaling. The Euclidean norm is calculated for each row (mode shape) and used to normalize the mode shapes. The MAC values are then computed using the normalized mode shapes to assess the similarity between corresponding modes in different datasets.

The code generates three bar3 plots to visualize the MAC values. The plot represents the MAC values between the modes of the '0-15' dataset and the '15-30' dataset.. The MAC values are represented on the z-axis, while the x-axis and y-axis represent the respective eigenfrequencies of the modes being compared.

Overall, this code provides a comprehensive analysis of the modal characteristics of the datasets by evaluating the similarity between their mode shapes using the MAC criterion. It helps in assessing the consistency and correlation of the modal properties across different datasets, contributing valuable insights to my research on modal analysis.

```
clear all
close all
clc;


% Load the eigenfrequencies and mode shapes from the first MAT file
matFile1 = '0-30.mat';
data1 = load(matFile1);
freq1 = data1.finalFrequencies;  % Assuming the eigenfrequencies are stored in a variable
named 'fn01'
modes1 = data1.finalModeShapes;  % Assuming the mode shapes are stored in a variable named
'phi0'

% Load the eigenfrequencies and mode shapes from the second MAT file
matFile2 = '0-30.mat';
data2 = load(matFile2);
freq2 = data2.finalFrequencies;  % Assuming the eigenfrequencies are stored in a variable
named 'fn01'
modes2 = data2.finalModeShapes;  % Assuming the mode shapes are stored in a variable named
'phi0'

% Step 1: Normalize the mode shapes
modes1_norm = sqrt(sum(modes1.^2, 2));  % Compute the Euclidean norm for each row (mode shape)
of modes1
modes2_norm = sqrt(sum(modes2.^2, 2));  % Compute the Euclidean norm for each row (mode shape)
of modes2
modes1_normalized = modes1 ./ modes1_norm;  % Normalize each row (mode shape) of modes1
modes2_normalized = modes2 ./ modes2_norm;  % Normalize each row (mode shape) of modes2


% Step 2: Calculate the MAC values
MAC1 = zeros(size(modes1, 1), size(modes2, 1));  % Initialize the MAC matrix
for i = 1:size(modes1, 1)
    for j = 1:size(modes2, 1)
        MAC1(i, j) = abs(modes1_normalized(i, :) * modes2_normalized(j, :)')^2 / ...
            (modes1_normalized(i, :) * modes1_normalized(i, :)') * (modes2_normalized(j, :) *
modes2_normalized(j, :)');
    end
end



% Create a bar3 plot with MAC values
figure('Name', 'MAC Values 0-15,15-30');  % Set the name of the figure windowbar3(MAC2);
bar3(MAC1);
xlabel('Frequency 1');
ylabel('Frequency 2');
zlabel('MAC1');
title('Modal Assurance Criterion (MAC)');

% Customize the axis labels
xticks(1:numel(freq1));
yticks(1:numel(freq2));
xticklabels(freq1);
yticklabels(freq2);
xtickangle(45);
ytickangle(45);
```

6.4 Geometry of the bridge

In this MATLAB code the geometry of the bridge is built.

```
% script geometry_built.m
% generates the geometry of the structure to be used for modal analysis
%
%-----------------------------------------------------------------------
% Geometry of Chalkida bridge: span 3
%-----------------------------------------------------------------------
% nodal coordinates in meters (m)
% Deck nodes
% deck width=12.60m
% deck length 305
% column height = 90
```

```
% Specify the following variables required to plot the structure
% Use a coordinate system Oxyz and write coordinates in meters with respect
% to the coordinate systems.
% FEnode = [node_number  x_coord  y_coord  z_coord;
%           node_number  x_coord  y_coord  z_coord;
%           ...];
% FEelt = [element_number  node_1  node_2;
%          element_number  node_1  node_2;
%          ...];
% node_coords = [x_coord  y_coord  z_coord;
%                x_coord  y_coord  z_coord;
%                ...];
% el_nodes =
% node_dofs = [node_1  node_2;
%              node_1  node_2;
%              ...];
% sensors = [sensor_number  node_number  direction  1;
%            sensor_number  node_number  direction  1;
%            ...];
%           where direction=1 for x-dir, =2 for y-dir, =3 for z-dir
%           with respect to the global coordinate system Oxyz
node_coords=[0         0      40;
             0       12.60    40;
             30        0      40;
             30      12.60    40;
             90        0       0;
             90      12.60     0;
             90        0      40;
             90      12.60    40;
             90        0      90;
             90      12.60    90;
             177.5     0      40;
             177.5   12.60    40;
             197.5     0      40;
             197.5   12.60    40;
             217.5     0      40;
             217.5   12.60    40;
             237.5     0      40;
             237.5   12.60    40;
             305       0       0;
             305     12.60     0;
             305       0      40;
             305     12.60    40;
             305       0      90;
             305     12.60    90;
             365       0      40;
             365     12.60    40;
             395       0      40;
             395     12.60    40;
             0         0      20;
             0       12.60    20;
             395       0      20;
             395     12.60    20];

% connectivity array
% deck
el_nodes=[1 2;
          2 4;
          4 8;
          1 3;
          3 7;
          6 8;
          5 7;
          7 9;
          10 8;
          10 9;
          7 11;
          8 12;
          11 13;
          12 14;
          13 15;
          14 16;
          15 17;
          16 18;
          17 21;
          18 22;
```

```
          21 19;
          22 20;
          23 21;
          22 24;
          23 24;
          21 25;
          22 26;
          25 27;
          26 28;
          27 28;
          1  29;
          2  30;
          28 32;
          27 31];

% sensors
sensors = ...
     [1      14      3      -1;
      2      23      1      -1;
      3      10      1      -1;
      4       9      1      -1;
      5       2      3      -1;
      6      22      2      -1;
      7      26      3      -1;
      8      10      2      -1;
      9      22      1      -1;
      10      8      2      -1;
      11     24      2      -1;
      12     21      1      -1;
      13     24      1      -1;
      14     17      3      -1;
      15      5      1      -1;
      16      5      3      -1;
      17      5      2      -1];
%save('geometry', 'node_coords', 'el_nodes');
```

6.5 Plot mode shapes in the bridge

In this code mode shapes are loaded and also the geometry of the bridge. Finally mode shapes are plotted in the bridge

```
% Main program
clear all;
close all;
clc
%---------------------------------------------------------------------------
% phi - modeshape matrix
% freqHz - row vector of modal frequencies in Hz
% zeta - row vector of modal damping ratios
%---------------------------------------------------------------------------
% phi = modeshapes
% phi = [comp1_mode1    comp1_mode2   comp1_mode3  ...;
%         comp2_mode1    comp2_mode2   comp2_mode3  ...;
%         comp3_mode1    comp3_mode2   comp3_mode3  ...;
%         comp4_mode1    comp4_mode2   comp4_mode3  ...;
%        ...];
%  freqHz = modal frequencies in Hz
%  freqHz = [freq1  freq2 freq3 ...];
%  zeta = modal damping ratios
%  zeta = [zeta1  zeta2  zeta3 ...];
%
%  modes = contain the mode numbers to be plotted
%  modes = [mode1  mode3  mode4 ...]; plots phi(:,1), phi(:,3), phi(:,4)
%---------------------------------------------------------------------------
% Load modal identification results: phi, freqHz and zeta
%
%
%
%---------------------------------------------------------------------------
load('0-30.mat')
phi=transpose(finalModeShapes);
```

```
freqHz=finalFrequencies;
zeta=finalDampingRatios;
%--------------------------------------------------------------------------
%
%
%
% assign modes to be plotted from the phi modeshape matrix
modes = [2 4 ];  % Plot modes 1 and 3 included in modeshape matrix phi
% call plotting software
plotmodes(phi,freqHz,zeta,modes);
%
```

7 References

[1] Jimin He and Zhi-Fang Fu, Modal Analysis, published in USA (2001)

[2] Salonikios Th., Morfidis K., Lekidis b., Απλουστευμένη Αναγνώριση Ιδιόμορφών της Γέφυρας του Ευρίπου στην Χαλκίδα, 5ο Πανελλήνιο Συνέδριο Αντισεισμικής Μηχανικής και Τεχνικής Σεισμολογίας, 20-22/10/2022

[3] Welch's method https://www.mathworks.com/help/signal/ref/pwelch.html

[4] Jeff Zhao, Understanding the Stability Diagram, published in August 25, 2018, https://www.crystalinstruments.com/blog/2018/8/25/understanding-the-stability-diagram

[5] SSI_COV. https://www.mathworks.com/matlabcentral/fileexchange/69030-operational-modal-analysis-with-automated-ssi-cov-algorithm

[6] Miroslav Pastora, Michal Bindaa, Tomáš Harþarika (2012), Modal Assurance Criterion, published in Procedia Engineering 48 (2012) ,pages 543 - 548

[7] Modal analysis and mode shapes http://vlabs.iitkgp.ernet.in/vlabs/vlab17/web/note3.html#:~:text=A%20mode%20shape%20is%20a,all%20others%20in%20the%20structure.

[8] Randall J. Allemang, The Modal Assurance Criterion –Twenty Years of Use and Abuse, University of Cincinnati, Cincinnati, Ohio, published in Sound and Vibration (August 2003)