

Universidade Federal do Rio Grande do Sul
Escola de Engenharia
Programa de Pós-Graduação em Engenharia Civil

**Identificação das Propriedades Dinâmicas de Estruturas
submetidas a Ações Ambientais**

Matheus Roman Carini

Porto Alegre
2021

MATHEUS ROMAN CARINI

**IDENTIFICAÇÃO DAS PROPRIEDADES DINÂMICAS DE
ESTRUTURAS SUBMETIDAS A AÇÕES AMBIENTAIS**

Tese apresentada ao Programa de Pós-Graduação em Engenharia Civil da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para obtenção do título de Doutor em Engenharia

Porto Alegre
2021

MATHEUS ROMAN CARINI

**IDENTIFICAÇÃO DAS PROPRIEDADES DINÂMICAS DE
ESTRUTURAS SUBMETIDAS A AÇÕES AMBIENTAIS**

Esta tese de doutorado foi julgada adequada para a obtenção do título de DOUTOR EM ENGENHARIA, Estruturas, e aprovada em sua forma final pelo professor orientador e pelo Programa de Pós-Graduação em Engenharia Civil da Universidade Federal do Rio Grande do Sul.

Porto Alegre, 8 de setembro de 2021

Prof. Marcelo Maia Rocha
Dr. techn., Universität Innsbruck, Áustria

Orientador

Prof. Acir Mércio Loredou-Souza
Ph. D., University of Western Ontario,
Canadá

Coorientador

Prof. Nilo Cesar Consoli
Coordenador do PPGEC/UFRGS

BANCA EXAMINADORA

Profa. Michèle Schubert Pfeil (UFRJ)
D.Sc., Universidade Federal do Rio de Janeiro, Brasil

Profa. Leticia Fleck Fadel Miguel (UFRGS)
Dra., Universidade Federal do Rio Grande do Sul, Brasil

Prof. Gustavo Javier Zani Núñez (UFRGS)
Dr., Universidade Federal do Rio Grande do Sul, Brasil

Prof. Zacarias Martin Chamberlain Pravia (UPF)
D.Sc., Universidade Federal do Rio de Janeiro, Brasil

Dedico este trabalho a meus pais, Nadil e Vânia, que sempre me apoiaram e estiveram ao meu lado.

AGRADECIMENTOS

Agradeço a Deus pela minha vida e saúde, por ter me agraciado com o dom da inteligência, por me proteger diariamente e pela família maravilhosa que tenho.

Agradeço especialmente aos meus pais, Nadil e Vânia, pelo apoio incondicional, por serem os responsáveis por toda a minha educação, me incentivarem e sempre proporcionarem todas as condições que culminaram nesta grande conquista.

Aos meus familiares pelas palavras de apoio e por me proporcionarem a base de uma vida feliz.

Agradeço ao Prof. Marcelo Rocha, professor e orientador exemplar, pela dedicação, apoio e incentivo constante durante a realização deste trabalho. Pela contribuição em minha formação pessoal e profissional.

Agradeço aos demais professores do PPGEC por todo o conhecimento transmitido.

Agradeço ao meu amigo Lucas pelas palavras sinceras, risadas e momentos inesquecíveis.

Agradeço aos meus queridos alunos por me mostrarem que a docência é a profissão que faz meus olhos brilharem.

Agradeço ao Prof. Zacarias Chamberlain e ao Jorge Braidó por tornarem possível a instrumentação da Passarela P01.

Agradeço ao José Lucas Silva Borges pelo desenvolvimento do equipamento e pela instrumentação da Passarela P02 possibilitando a sua inclusão neste trabalho.

Agradeço à divisão de Mecânica Estrutural da Universidade Católica de Leuven por disponibilizarem os dados da Ponte Z24.

Ao CNPq pela bolsa de estudos concedida.

A todos que me ajudaram, muito obrigado!

Tudo o que fizerdes, fazei-o de coração [...].

Colossenses 3:23

RESUMO

CARINI, M. R. **Identificação das propriedades dinâmicas de estruturas submetidas a ações ambientais.** Tese (Doutorado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2021.

A identificação de sistemas visa mensurar a descrição matemática de um sistema analisando as suas entradas e saídas. No contexto da Engenharia Civil, as estruturas como edifícios, torres e pontes são os sistemas e os parâmetros a serem identificados são as frequências naturais, razões de amortecimentos e formas modais. As entradas são os carregamentos aplicados e as saídas são as respostas estruturais, as quais podem ser medidas em termos de deslocamentos, velocidades ou acelerações. Para que o processo de identificação seja conduzido, é necessário que exista algum carregamento atuante na estrutura. Em estruturas de grande porte, a análise de sua vibração decorrente de ações ambientais torna-se mais adequada, como, por exemplo, aquelas causadas pelo vento em edifícios e torres, pelo tráfego em pontes e por pessoas em passarelas. A identificação de estruturas a partir de excitações ambientais é denominada identificação estocástica de sistemas. Nesta apenas a resposta (saída) é medida, sendo a excitação considerada como um processo estocástico multidimensional, representado tipicamente por um ruído branco. O principal objetivo deste trabalho é o desenvolvimento de rotinas que permitam a simulação do processo de instrumentação e identificação de uma estrutura, visando auxiliar o planejamento das medições e aproveitar ao máximo as características dos sensores disponíveis. Apresenta-se a teoria de identificação de sistemas no âmbito da Engenharia Civil, enfatizando métodos que utilizam apenas os dados da resposta estrutural. A partir da teoria apresentada, desenvolveu-se um módulo em linguagem Python denominado CESSIPy, o qual é público, aberto e encontra-se disponível na plataforma de hospedagem GitHub. Procedeu-se a simulação do processo de identificação de um edifício alto sob ação do vento, utilizando as forças medidas em túnel de vento e a técnica *High Frequency Pressure Integration*. Validaram-se as rotinas na identificação de estruturas reais: a Passarela P01 em Passo Fundo, a Passarela Domingos Martins em Canoas e a Ponte Z24 na Suíça.

Palavras-chave: identificação de sistemas; análise modal operacional; vibrações em estruturas

ABSTRACT

CARINI, M. R. **Identification of modal parameters of structures from ambient vibration measurements.** Tese (Doutorado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2021.

The process of finding a mathematical model of a dynamic system from inputs and/or outputs is called system identification. In a civil engineering context, structures, such as buildings, towers, and bridges are the systems and the identified model is a modal model consisting of eigenfrequencies, damping ratios and modal shapes. For large structures, in-operation vibration tests are cheaper and simpler. Vehicle traffic in bridges and wind in buildings are examples of ambient excitations. When the structure is excited by an unmeasurable input force and only output measurements are available, the process is called stochastic system identification. The loads are treated as multidimensional stochastic processes, typically under the assumption of having white noise characteristics. Therefore, the main aim of this thesis is to develop a computational tool for identification of civil engineering structures from output-only data. An open-source Python code was created, which is available at GitHub website. The tool implements the frequency domain Peak-Picking (PP) method and Enhanced Frequency Domain Decomposition method (EFDD), as well as the time domain Stochastic Subspace Identification techniques (SSI COV and SSI DATA) and the Instrumental Variable method (IV). Then, the identification of a tall building excited by wind load was simulated. Boundary layer wind tunnel measurements were combined with the High Frequency Pressure Integration technique to estimate building response and modal parameters were estimated from acceleration measurements. Finally, vibration data from ambient tests of the footbridge P01 in Passo Fundo, the footbridge Domingos Martins in Canoas and the bridge Z24 in Switzerland are analyzed to extract modal parameters.

Keywords: system identification in civil engineering; operational modal analysis; structural dynamics

LISTA DE FIGURAS

Figura 1.1 – Relação excitação-resposta de um sistema estrutural	23
Figura 2.1 – Principais métodos de identificação de sistemas	34
Figura 4.1 – Valores singulares	87
Figura 4.2 – Gráfico dos números de MAC gerado pela função MAC	89
Figura 4.3 – Diagrama de estabilização.....	90
Figura 4.4 – Funções do módulo CESSIPy	92
Figura 5.1 – Geometria da estrutura	95
Figura 5.2 – Formas modais da estrutura estudada	96
Figura 5.3 – Localização de tomadas de pressão no modelo do edifício	97
Figura 5.4 – Direções do vento.....	99
Figura 5.5 – Registros de acelerações no topo do protótipo.....	102
Figura 5.6 – Gráficos das densidades espectrais da resposta em acelerações	103
Figura 5.7 – Gráficos dos autoespectros normalizados e a sua média	104
Figura 5.8 – Estimativas das razões de amortecimento.....	105
Figura 5.9 – Gráficos das funções de coerência	106
Figura 5.10 – Gráfico dos espectros dos valores singulares.....	108
Figura 5.11 – Gráficos dos espectros dos valores singulares e do MAC	109
Figura 5.12 – Estimativas das frequências naturais e razões de amortecimento pelo método EFDD.....	109
Figura 5.13 – Gráficos dos valores singulares normalizados	110
Figura 5.14 – Gráficos dos valores singulares.....	111
Figura 5.15 – Diagramas de estabilização	112
Figura 5.16 – Formas modais identificadas.....	113
Figura 5.17 – Propriedades dinâmicas identificadas	118
Figura 5.18 – Frequências detectadas no Cenário 3	121
Figura 5.19 – Gráfico da aceleração RMS vs velocidade do vento e relação ruído/sinal	123
Figura 5.20 – Resultados do primeiro modo para ruído/sinal entre 0 e 100%	125
Figura 5.21 – Resultados do segundo modo para ruído/sinal entre 0 e 100%.....	126
Figura 5.22 – Resultados do terceiro modo para ruído/sinal entre 0 e 100%	127
Figura 5.23 – Resultados do primeiro modo para ruído/sinal entre 0 e 500%	129
Figura 5.24 – Resultados do segundo modo para ruído/sinal entre 0 e 500%.....	130
Figura 5.25 – Diagramas de estabilização para ruído/sinal igual a 400%	132
Figura 5.26 – Foto da Passarela P01	133

Figura 5.27 – Geometria da Passarela P01	134
Figura 5.28 – Estimativa da densidade espectral do ruído do sensor	135
Figura 5.29 – Posições do sensor na Passarela P01	135
Figura 5.30 – Espectros normalizados médios das acelerações	137
Figura 5.31 – Espectro normalizado das acelerações verticais	138
Figura 5.32 – Formas modais e frequências naturais de uma viga biapoiada	141
Figura 5.33 – Perspectiva do modelo de elementos finitos da estrutura da Passarela P01	142
Figura 5.34 – Propriedades dinâmicas do modelo em elementos finitos da Passarela P01 ...	142
Figura 5.35 – Fotos da Passarela P02	145
Figura 5.36 – Posições dos sensores na Passarela P02 em todos os ensaios	145
Figura 5.37 – Espectros normalizados médios das acelerações e das velocidades angulares	148
Figura 5.38 – Diagrama de estabilização do Ensaio 1-2	150
Figura 5.39 – Propriedades dinâmicas identificadas	151
Figura 5.40 – Visão geral da Ponte Z24	153
Figura 5.41 – Registro de aceleração vertical no sensor R3.....	154
Figura 5.42 – Numeração e localização dos sensores nos PDTs.....	156
Figura 5.43 – Diagrama de estabilização na Configuração 5	157
Figura 5.44 – Formas modais identificadas.....	160

LISTA DE QUADROS

Quadro 3.1 – Funções de peso.....	82
Quadro 5.1 – Propriedades dinâmicas identificadas pelo método BFD.....	107
Quadro 5.2 – Comparativo das frequências.....	115
Quadro 5.3 – Comparativo das razões de amortecimento.....	115
Quadro 5.4 – Comparativo dos MAC.....	116
Quadro 5.5 – Ensaio na Passarela P01.....	136
Quadro 5.6 – Frequências naturais e razões de amortecimento identificadas para o primeiro modo vertical.....	139
Quadro 5.7 – Frequências naturais e razões de amortecimento identificadas para o segundo modo vertical.....	140
Quadro 5.8 – Parâmetros da estrutura.....	141
Quadro 5.9 – Comparativo das frequências naturais.....	143
Quadro 5.10 – Ensaio na Passarela P02.....	146
Quadro 5.11 – Relações ruído/sinal das séries temporais adquiridas na Passarela P02.....	147
Quadro 5.12 – Frequências e modos identificados pelo método SSI COV.....	149
Quadro 5.13 – Frequências e razões de amortecimento identificadas pelo método SSI COV.....	150
Quadro 5.14 – Estimativas das frequências naturais da Ponte Z24 pelo método SSI COV...	158
Quadro 5.15 – Estimativas das razões de amortecimento da Ponte Z24 pelo método SSI COV.....	158
Quadro 5.16 – Comparativo das frequências naturais e razões de amortecimento estimadas para a Ponte Z24.....	159

LISTA DE SIGLAS

ANPSD – Densidade Espectral Normalizada Média

ARMA – Modelos Autorregressivos de Média Móvel

BFD – Método Básico no Domínio da Frequência

CAARC – *Commonwealth Advisory Aeronautical Research Council*

CE – Método da Exponencial Complexa

CESSIPy – *Civil Engineer Stochastic System Identification for Python*

CMIF – Método da Decomposição no Domínio da Frequência

EFDD – Método Aperfeiçoado Decomposição no Domínio da Frequência

ERA – Algoritmo de Realização de Sistemas

EVD – Decomposição em Autovalores e Autovetores

FDD – Método de Decomposição no Domínio da Frequência

FFT – Transformada Rápida de Fourier

HFPI – *High Frequency Pressure Integration*

GPS – Sistema de Posicionamento Global

ITD – Método de Ibrahim no Domínio do Tempo

IV – Método das Variáveis Instrumentais

LAC – Laboratório de Aerodinâmica das Construções

LS – Mínimos Quadrados

LSCE – Método Exponencial Complexo por Mínimos Quadrados

MAC – Critério de Concordância Modal

MIMO – Múltiplas Entradas e Múltiplas Saídas

NExT – Técnica de Excitação Natural

NPSD – Densidade Espectral Normalizada

p-LSCF – Método da Polireferência no Domínio da Frequência por Mínimos Quadrados

PDT – Ensaio de Dano Progressivo

PEM – Método de Predição de Erro

PP – Método de Detecção de Picos

PPGEC – Programa de Pós-Graduação em Engenharia Civil

PTD – Método da Polireferência no Domínio do Tempo

RD – Decremento Aleatório

SIMCES – *System Identification to Monitor Civil Engineering Structures*

SIMO – Uma Entrada e Múltiplas Saídas

SISO – Uma Entrada e Uma Saída

SSI – Identificação Estocástica de Subespaços

SSI COV – Identificação Estocástica de Subespaços baseada nas Covariâncias

SSI DATA – Identificação Estocástica de Subespaços baseada nas Séries Temporais

SVD – Decomposição em Valores Singulares

UFRGS – Universidade Federal do Rio Grande do Sul

LISTA DE SÍMBOLOS

Letras romanas maiúsculas

A – matriz dinâmica do sistema

$\mathbf{A}_p^{\text{comp}}$ – matriz de estado na forma companheira

A – área da seção transversal

B – matriz de entradas

B_i – largura da meia potência

C – matriz de influência na saída

C – constante

\mathbf{C}_f – matriz de amortecimento

D – matriz de transmissão direta

E – módulo de elasticidade

$\vec{\mathbf{F}}(t)$ – vetor de cargas

$\vec{\mathbf{F}}_m(t)$ – vetor de forças modais

$F_i(t)$ – força modal

G – matriz de covariâncias do estado no tempo $k + 1$ e da saída no tempo k

\mathbf{G}^{ref} – matriz de covariâncias reduzida do estado no tempo $k + 1$ e da saída no tempo k

\mathbf{G}_m^{ref} – matriz de covariâncias reduzida modal do estado no tempo $k + 1$ e da saída no tempo k

$\hat{\mathbf{G}}_y(\omega_i)$ – estimativa da matriz das funções de densidade espectrais (*one-sided*) das saídas

\mathbf{H}^{ref} – matriz bloco Hankel das saídas

\mathbf{I}_n – matriz identidade dimensões $n \times n$

I – momento de inércia

\mathbf{K} – matriz de rigidez

\mathbf{L} – matriz de seleção das referências

L – vão efetivo da viga

L_m – comprimento do modelo

L_p – comprimento do protótipo

\mathbf{M} – matriz de massa

N – número de amostras no tempo

\mathbf{O}_t – matriz de observabilidade

$\mathbf{O}_{p,m}$ – matriz de observabilidade modal

\mathbf{P} – matriz auxiliar empregada no modelo de espaço de estado

\mathcal{P}_t^{ref} – projeção do espaço das linhas das saídas futuras no espaço das linhas das referências passadas

\mathbf{Q} – matriz auxiliar empregada no modelo de espaço de estado

$\mathbf{Q}_r, \mathbf{N}_r, \mathbf{S}_r$ – matrizes de covariância do ruído

\mathbf{R}_i – matriz de covariâncias das saídas para um atraso i de tempo

$\hat{\mathbf{R}}_i$ – estimativa da matriz de covariâncias das saídas para um atraso i de tempo

\mathbf{R}_i^{ref} – matriz reduzida de covariâncias das saídas para um atraso i de tempo

\mathbf{S}_i – matriz com os valores singulares em ordem decrescente

$\mathbf{S}_y(\omega)$ – matriz das funções de densidade espectrais (*two-sided*) das saídas

$\hat{\mathbf{S}}_y(\omega_i)$ – estimativa da matriz das funções de densidade espectrais (*two-sided*) das saídas

$\mathbf{T}_{1|t}^{ref}$ – matriz Toeplitz das covariâncias das saídas

\mathbf{U}_i – matriz unitária complexa obtida na decomposição em valores singulares

\mathbf{V} – matriz com as formas modais observáveis

V_m – velocidade no modelo

V_p – velocidade no protótipo

$\mathbf{W}_1, \mathbf{W}_2$ – matrizes pesos

$\hat{\mathbf{X}}_t$ – matriz de sequências de estado do filtro Kalman

$\vec{Y}(\omega_i)$ – vetor da transformada discreta de Fourier das saídas

$\mathbf{Y}_p^{ref}, \mathbf{Y}_{0|t-1}^{ref}$ – matriz das saídas passadas

$\mathbf{Y}_f, \mathbf{Y}_{t|2t-1}$ – matriz das saídas futuras

Letras romanas minúsculas

c_i – amortecimento modal

\vec{e}_k – vetor ruído branco

f_m – frequência no modelo

f_p – frequência no protótipo

j – unidade imaginária

k_i – rigidez modal

l – número de saídas

m – número de entradas

m_i – massa modal

n – número de graus de liberdade do sistema

n_2 – número de ordem no espaço de estado, número de estados

n_α – ordem da parte AR

n_γ – ordem da parte MA

p – ordem do modelo ARMA

r – número de saídas de referência

t – tempo

t_m – tempo no modelo

t_p – tempo no protótipo

$\vec{u}(t)$ – vetor de deslocamentos

$\dot{\vec{u}}(t)$ – vetor de velocidades

$\ddot{\vec{u}}(t)$ – vetor de acelerações

$u_i(t)$ – resposta modal

$\vec{u}_m(t)$ – vetor das respostas modais

$\vec{x}(t)$ – vetor de estado

\vec{x}_k – vetor de estado no instante k

\vec{v}_k – vetor de entradas (excitações) no instante k

\vec{y}_k – vetor de saídas (respostas) no instante k

\vec{y}_k^{ref} – vetor de saídas de referência

\vec{y}_k^{\sim} – vetor das demais saídas

\vec{w}_k – vetor de perturbações e imprecisões do modelo no instante k

w_k – janela de dados

\vec{z}_k – vetor de ruídos na medição no instante k

Letras gregas

α – escalar

α_i – matriz com os coeficientes da parte autorregressiva

β – escalar

Γ_i^{ref} – matriz de controlabilidade

γ_i – matriz com os coeficientes da parte com média móvel

$\gamma_{i,j}^2(\omega_m)$ – função de coerência entre as saídas i e j

Δf – resolução em frequência

Δt – período de amostragem

δ_{pq} – delta de Kronecker

Θ – matriz dos autovalores do modelo com amortecimento não proporcional

ι – número de atrasos de tempo

Λ – matriz dos autovalores do modelo de elementos finitos

Λ_C – matriz dos autovalores em tempo contínuo

Λ_D – matriz dos autovalores em tempo discreto

λ_f – escala de frequência

λ_i – autovalores em tempo contínuo

λ_L – escala de comprimento

λ_t – escala de tempo

λ_V – escala de velocidade

μ – massa linear

μ_i – autovalores em tempo discreto

ξ_{ft} – razão de amortecimento estimada pelo método de ajuste ao espectro analítico

ξ_i – razão de amortecimento crítico

ξ_{mp} – razão de amortecimento estimada pelo método de meia-potência

Π_u – constante

ρ – massa específica

Σ – matriz de covariâncias dos estados

Φ – matriz das formas modais

$\vec{\varphi}_i$ – forma modal

Ψ – matriz dos autovetores do modelo no espaço de estado

Ψ_{comp} – matriz dos autovetores de $\mathbf{A}_{\text{comp}}^{b,p}$

$\vec{\psi}_i$ – vetor de formas modais complexas

Ω – matriz das frequências naturais

ω_i – frequência natural angular

ω_d – frequência natural angular amortecida

Operadores

$E[.]$ – valor esperado

$(.)^*$ – complexo conjugado

$(.)^\dagger$ – Moore-Penrose pseudo-inversa

$(.)^H$ – conjugado transposto

$(.)^T$ – transposta

SUMÁRIO

1 INTRODUÇÃO	22
1.1 OBJETIVOS	25
1.2 ORGANIZAÇÃO DA TESE.....	25
2 ESTADO DA ARTE	28
2.1 EVOLUÇÃO DOS MÉTODOS DE IDENTIFICAÇÃO	28
2.2 VIBRAÇÕES AMBIENTAIS	35
2.2.1 Barragens	35
2.2.2 Pontes	37
2.2.3 Vigas, pórticos e edifícios	41
2.2.4 Torres	43
3 MODELOS DINÂMICOS	46
3.1 TIPOS DE MODELOS	46
3.2 ELEMENTOS FINITOS	47
3.2.1 Vibrações livres não amortecidas	48
3.2.2 Amortecimento viscoso proporcional	49
3.2.3 Amortecimento viscoso não proporcional	51
3.3 ESPAÇO DE ESTADO	53
3.3.1 Propriedades dos sistemas estocásticos	55
3.3.2 Análise modal	57
3.4 MODELOS ARMA	57
4 IDENTIFICAÇÃO ESTOCÁSTICA DE SISTEMAS	60
4.1 DADOS EXPERIMENTAIS	60
4.1.1 Séries temporais	61
4.1.2 Estimativas das funções de covariâncias	62
4.1.3 Estimativas das densidades espectrais	64
4.2 MÉTODOS NO DOMÍNIO DA FREQUÊNCIA	66
4.2.1 Método Básico no Domínio da Frequência (BFD)	66
4.2.2 Método Aperfeiçoado de Decomposição no Domínio da Frequência	70
4.3 MÉTODOS BASEADOS NAS COVARIÂNCIAS	71
4.3.1 Método das Variáveis Instrumentais (IV)	72
4.3.1.1 Introdução dos sensores de referência	73
4.3.1.2 Algoritmo e implementação.....	75
4.3.2 Método de identificação estocástica de subespaços baseados nas covariâncias (SSI COV)	76
4.4 MÉTODOS NO DOMÍNIO DO TEMPO	79

4.4.1 Método de identificação estocástica de subespaços baseado nas séries temporais (SSI DATA).....	79
4.4.1.1 Algoritmo e implementação.....	82
4.5 TÉCNICAS PARA DISTINÇÃO DOS MODOS REAIS E ESPÚRIOS	85
4.5.1 Decomposição em Valores Singulares.....	86
4.5.2 Diagrama de estabilização	87
4.6 IMPLEMENTAÇÃO EM PYTHON	90
5 APLICAÇÕES.....	94
5.1 ANÁLISE NUMÉRICA DO CAARC SOB AÇÃO DO VENTO	94
5.1.1 O CAARC <i>Standard Tall Building</i>.....	94
5.1.2 <i>High-Frequency Pressure Integration (HFPI)</i>	97
5.1.3 Leis de semelhança	99
5.1.4 Um sensor no topo do edifício.....	100
5.1.4.1 Cenário 1: evento extremo com ausência de ruído	101
5.1.4.2 Cenário 2: influência do tempo de aquisição.....	117
5.1.4.3 Cenário 3: ruído absoluto igual a 2,5 mG.....	120
5.1.4.4 Cenário 4: várias relações ruído/sinal.....	124
5.2 PASSARELA P01	133
5.2.1 Descrição da Passarela P01.....	133
5.2.2 Instrumentação e ensaios	134
5.2.3 Resultados.....	136
5.2.3.1 Identificação das propriedades dinâmicas a partir das acelerações	136
5.2.3.2 Vibrações em sistemas estruturais contínuos.....	140
5.2.3.3 Modelo de elementos finitos.....	141
5.2.3.4 Comparativo.....	143
5.3 PASSARELA P02	144
5.3.1 Descrição da Passarela P02.....	144
5.3.2 Instrumentação e ensaios	144
5.3.3 Resultados.....	147
5.4 PONTE Z24.....	152
5.4.1 Breve descrição da Ponte Z24.....	152
5.4.2 Ensaios	153
5.4.3 Resultados.....	157
6 CONSIDERAÇÕES FINAIS.....	162
REFERÊNCIAS	165
APÊNDICE A – CESSIPy: código fonte	172
APÊNDICE B – CESSIPy: guia ao usuário	202
APÊNDICE C – Frequências naturais e razões de amortecimento da Ponte Z24	240

1 INTRODUÇÃO

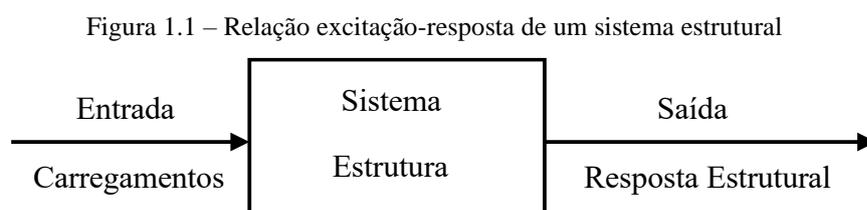
Um modelo matemático de um sistema pode ser utilizado para sua simulação, previsão da resposta para excitações conhecidas e estudo de diferentes alternativas de projeto. Entretanto, esses objetivos só podem ser alcançados se o modelo do sistema for conhecido. A validade de todas as simulações, análises e projeto do sistema depende da precisão do modelo em representar o sistema real. A elaboração de um modelo dinâmico preciso torna-se fundamental (BILLINGS, 2013).

No projeto estrutural, o método dos elementos finitos é empregado para a construção de modelos numéricos. Trata-se de uma abordagem bem estabelecida e que normalmente fornece um modelo com precisão adequada para fins de projeto. Uma vez construída a estrutura, a realização de ensaios estáticos e dinâmicos pode fornecer informações valiosas para o refinamento do modelo em elementos finitos e principalmente para o projeto de sistemas de controle de vibração em estruturas flexíveis (JUANG; PAPPA, 1985).

Na modelagem numérica, costuma-se ignorar a influência dos elementos não estruturais (como a alvenaria no caso de edifícios) na rigidez da estrutura. Conseqüentemente, pode haver uma diferença entre as frequências naturais da estrutura e do modelo em elementos finitos. Usualmente elas são subestimadas, ou seja, o prédio é mais rígido que o esperado e, portanto, o projeto será conservador. Um entendimento melhor do comportamento devido às cargas laterais pode ser feito a partir da avaliação experimental dos modos de vibração (CRISAFULLI et al., 2000; BROWNJOHN, 2002).

Atualmente, verifica-se no Brasil um aumento progressivo da altura dos edifícios e dos vãos das pontes. Essas estruturas esbeltas são suscetíveis a efeitos dinâmicos. Nesse sentido, ensaios em túnel de vento são feitos para estimar as forças devidas ao vento. As séries temporais de pressões medidas são combinadas com o modelo numérico da estrutura para a obtenção das forças estáticas equivalentes, as quais são utilizadas no projeto da estrutura. Nesse processo, admite-se que as características dinâmicas da estrutura obtidas em programas computacionais representam adequadamente o sistema real. Assim, torna-se de grande interesse a validação dos modelos empregados nas análises computacionais.

Identificação de sistemas é um método de mensurar a descrição matemática de um sistema analisando as suas entradas e saídas (BILLINGS, 2013). No contexto da Engenharia Civil, as estruturas como edifícios, torres e pontes são os sistemas e os parâmetros a serem identificados são as frequências naturais, razões de amortecimentos e formas modais. As entradas são os carregamentos aplicados, os quais podem ser determinísticos ou estocásticos, e as saídas são as respostas estruturais, as quais podem ser medidas em termos de deslocamentos, velocidades ou acelerações (Figura 1.1).



(fonte: elaborado pelo autor)

O principal objetivo da identificação de sistemas é a elaboração de um modelo matemático confiável de um sistema dinâmico com base nas séries temporais observadas e no conhecimento *a priori* do sistema. Procura-se encontrar, preferencialmente, o modelo mais simples possível e relacioná-lo com os componentes do sistema em estudo. Na Engenharia deseja-se caracterizar um comportamento complexo por meio de comportamentos simples que possam ser entendidos, manipulados e explorados (KEESMAN, 2011; BILLINGS, 2013).

A modelagem pode ser classificada em três categorias: modelagem caixa branca, modelagem caixa preta e modelagem caixa cinza. Na modelagem caixa branca admite-se que o sistema a ser modelado seja bem conhecido, assim como as leis físicas que o descrevem. Por essa razão esse tipo de modelagem é também conhecido como modelagem pela física, natureza do processo, modelagem fenomenológica ou conceitual. Se o modelo é construído sem levar em consideração o conhecimento físico diz-se que é uma modelagem caixa preta. Entre os dois tipos há a modelagem caixa cinza, na qual partes do modelo são baseadas no conhecimento físico (ANDERSEN, 1997; AGUIRRE, 2015).

O modelo matemático sempre é uma aproximação do sistema real. Na realidade, a complexidade do sistema, o conhecimento limitado do sistema e a disponibilidade limitada de dados observados prejudicam a exata descrição do sistema. Modelos matemáticos são limitados sem a confirmação da sua precisão, e testes em escala real são a única forma de avaliar a

validade das hipóteses feitas e parâmetros empregados. Tais testes fornecem estimativas do amortecimento estrutural para uso na determinação da resposta do modelo. Nota-se que modelos numéricos e ensaios experimentais se complementam, possibilitando uma melhor compreensão do comportamento dinâmico do sistema. Assim, a identificação de sistemas é considerada como uma modelagem aproximada para uma aplicação específica (BROWNJOHN et al., 1987; KEESMAN, 2011; MIGUEL et al., 2009).

Para que o processo de identificação seja conduzido, é necessário que exista algum carregamento atuante na estrutura. Essa excitação pode ser produzida por cargas de impacto, testes de vibração forçada ou excitações ambientais. Se a estrutura em estudo for de grande porte, como edifícios, pontes, torres e barragens, torna-se necessário o emprego de vibradores de grande massa para excitá-la. Nestas estruturas, a análise de sua vibração submetidas a condições ambientais é mais adequada. As excitações ambientais podem ser causadas pelo vento em edifícios e torres, pelo tráfego em pontes e por pessoas em passarelas.

As ações ambientais possuem natureza estocástica, conseqüentemente não podem ser descritas explicitamente como uma função dependente do tempo, mas caracterizadas por propriedades estatísticas como média e função de covariância. Como o sistema estrutural pode ser interpretado como uma transformação linear da excitação, isso significa que a resposta estrutural também é estocástica e pode ser representada por seus parâmetros estatísticos.

A identificação de estruturas a partir de excitações ambientais é denominada identificação estocástica de sistemas. Nestas apenas a resposta (saída) é medida, sendo a excitação considerada como um processo estocástico multidimensional, representado tipicamente por um ruído branco.

Além das aplicações citadas, a verificação da integridade estrutural faz uso de técnicas de identificação, tornando-se uma aplicação importante à medida que edifícios, pontes e barragens envelhecem e estão sujeitos a cargas crescentes. A acumulação de danos nas estruturas civis causa mudanças nas suas características dinâmicas. A sua detecção pode ser feita medindo as características dinâmicas ao longo da vida útil da estrutura. Alterações nas frequências naturais constituem o indicador mais largamente empregado. Outra aplicação é a calibração de normas, por exemplo para desenvolver relações empíricas entre as frequências naturais e alturas dos edifícios, e determinar razões de amortecimento a serem empregadas em normas de vento e sismo (ANDERSEN, 1997; BROWNJOHN, 2002).

A determinação das características dinâmicas passa por três processos: o modelo matemático da estrutura, as medições das vibrações e análise dos dados medidos. O presente trabalho apresenta procedimentos de análise dos dados, visando identificar os parâmetros do modelo matemático. Apesar de serem utilizados dados experimentais e estes serem fundamentais para a correta identificação do sistema, o processo de realização das medições não é apresentado detalhadamente, estando fora do escopo do trabalho.

1.1 OBJETIVOS

O objetivo principal deste trabalho é o desenvolvimento de rotinas que permitam a simulação do processo de instrumentação e identificação de uma estrutura, visando auxiliar o planejamento das medições e aproveitar ao máximo as características dos sensores disponíveis.

Para atingir o objetivo principal, foram definidos os seguintes objetivos específicos:

- a) a apresentação de métodos de identificação estocástica de sistemas no âmbito da Engenharia Civil;
- b) a implementação computacional dos métodos e a disponibilização dos algoritmos na internet;
- c) a aplicação dos métodos na identificação das propriedades dinâmicas de estruturas por meio de simulações e medições experimentais.

1.2 ORGANIZAÇÃO DA TESE

Inicia-se com a apresentação da teoria de identificação de sistemas no âmbito da Engenharia Civil, enfatizando métodos que utilizam apenas os dados da resposta estrutural. Nestes, o comportamento dinâmico da estrutura é modelado como linear, invariante no tempo e as excitações consideradas são ações ambientais não mensuráveis e tratadas como ruído branco gaussiano.

Existem softwares comerciais voltados para a identificação de sistemas, como ARTeMIS ou LMS Test.Lab. Entretanto estes softwares proprietários possuem desvantagens num ambiente de pesquisa, como custo de aquisição e impossibilidade de acesso à implementação dos algoritmos. Nesse sentido, a necessidade de desenvolvimento de rotinas específicas para a

identificação estocástica de sistemas se torna clara. A partir da teoria apresentada, programaram-se rotinas em linguagem Python. O módulo desenvolvido denomina-se CESSIPy (*Civil Engineer Stochastic System Identification for Python*). Trata-se de um módulo de código aberto, gratuito e que se encontra disponível para download na plataforma GitHub¹. O código foi validado por meio de simulações e pela análise de dados de estruturas reais.

Procede-se a simulação do processo de identificação de um edifício alto sob ação do vento, utilizando as forças medidas em túnel de vento e a técnica *High Frequency Pressure Integration*. São as forças do vento mais confiáveis possíveis de serem empregados em análises numéricas. Assim, assume-se que as respostas estruturais obtidas nas simulações sejam muito próximas das respostas da estrutura real.

Com as rotinas criadas, torna-se possível simular diversos cenários de instrumentação da estrutura real, podendo otimizar a localização e quantidade de sensores, visando atingir um objetivo específico. É possível ter noções do nível da resposta a ser medido para configuração do equipamento e se a resposta estrutural medida tem magnitude suficiente para a identificação confiável das suas características dinâmicas. Também conhecidas as características do sensor e do sistema de aquisição empregado, é possível determinar a partir de qual velocidade do vento as características dinâmicas da estrutura podem ser obtidas.

O presente trabalho foi dividido em seis capítulos. Este primeiro capítulo introduz o tema e apresenta os objetivos desse estudo. O segundo capítulo apresenta a evolução dos métodos de identificação de sistemas. São apresentados exemplos de identificação de estruturas civis encontrados na literatura técnica, como pontes, edifícios, barragens e torres. O terceiro capítulo apresenta modelos matemáticos para a descrição do comportamento dinâmico das estruturas. Modelos em elementos finitos, no espaço de estado e modelos autorregressivos com média móvel são discutidos. A determinação das características dinâmicas da estrutura a partir de cada uma dessas representações também é apresentada. O quarto capítulo descreve métodos de identificação estocástica de sistemas e apresenta a implementação desenvolvida em Python. Tratam-se de métodos que empregam apenas dados das saídas e supõem que as cargas sejam do tipo ruído branco gaussiano. São apresentados o Método de Detecção de Picos, o Método Aperfeiçoado de Decomposição no Domínio da Frequência, o Método das Variáveis Instrumentais e os Métodos de Identificação Estocástica de Subespaços baseados nas

¹ Disponível para download em < <https://github.com/MatheusCarini/CESSIPy> >.

Covariâncias e nas Séries Temporais. No quinto capítulo, as rotinas são empregadas para identificação das propriedades dinâmicas. Inicialmente simulou-se a identificação de um prédio alto submetido a ação do vento para diversos cenários. Em seguida, apresentam-se as identificações das Passarelas P01 e P02 submetidas às ações de pedestres, vento e tráfego rodoviário abaixo das estruturas. Na última aplicação, realiza-se a identificação da Ponte Z24 na Suíça submetida a ações ambientais e comparam-se os resultados obtidos no presente trabalho com os fornecidos por outros autores. Por fim, no sexto capítulo, apresentam-se as considerações finais.

2 ESTADO DA ARTE

Este capítulo apresenta uma revisão bibliográfica sobre métodos de identificação de sistemas. Inicialmente, um resumo da evolução das técnicas é apresentado e, em seguida, mostram-se aplicações na identificação de barragens, edifícios, pontes e torres submetidas a ações ambientais.

2.1 EVOLUÇÃO DOS MÉTODOS DE IDENTIFICAÇÃO

Os métodos de identificação podem ser divididos de acordo com o tipo de dados que empregam: séries temporais, covariâncias ou densidades espectrais. Inicia-se a apresentação pelos métodos no domínio da frequência para corresponder à ordem histórica da identificação estocástica de sistemas. Apesar das medições estarem no domínio do tempo, é muito útil analisar as suas representações no domínio da frequência. A ferramenta matemática para a conversão da série temporal para o domínio da frequência é a transformada de Fourier. Esse processo foi facilitado com o algoritmo da Transformada Rápida de Fourier (*Fast Fourier Transform* – FFT) proposto por Cooley e Tukey (1965).

A abordagem mais simples para estimar as características dinâmicas da estrutura submetida a ações ambientais é o Método de Detecção de Picos (*Peak Picking Method* – PP) também chamado de Método Básico no Domínio da Frequência (*Basic Frequency Domain Method* – BFD). Conforme o próprio nome sugere, a identificação das frequências naturais é feita selecionando-se os picos das densidades espectrais das respostas. As razões de amortecimento são determinadas a partir da largura desses picos. O método assume que o amortecimento é baixo e os modos são bem espaçados. Devido a sua simplicidade e rapidez, basicamente requer o cálculo da transformada rápida de Fourier da resposta, é, provavelmente, o método mais utilizado na Engenharia Civil (BENDAT; PIERSOL, 1993; PEETERS, 2000).

O Método da Decomposição no Domínio da Frequência (*Complex Mode Indication Function* – CMIF, também chamado de *Frequency Domain Decomposition* – FDD) é um aprimoramento do PP e surgiu como uma ferramenta para contagem do número de modos de vibração presentes nas medições. Nele aplica-se a decomposição em valores singulares da matriz espectral,

desacoplando-a em sistemas de um grau de liberdade, os quais podem ser analisados separadamente para a obtenção das características dinâmicas. A decomposição em valores singulares é utilizada para a determinação do posto da matriz espectral em função da frequência. A identificação de modos muito próximos pode ser feita analisando-se os valores singulares não nulos da matriz espectral na respectiva frequência. A existência de dois valores não nulos, por exemplo, indica a presença de dois modos (SHIH et al., 1988; BRINCKER et al., 2000; PEETERS, 2000).

Brincker et al. (2001b) melhoraram o FDD, transformando para o domínio do tempo a função de densidade espectral de cada modo de vibração, obtendo a função de autocorrelação e, a partir desta, determinando a frequência natural e a razão de amortecimento. Este aperfeiçoamento é designado por Método Aperfeiçoado de Decomposição no Domínio da Frequência (*Enhanced Frequency Domain Decomposition* – EFDD).

Existem outros métodos no domínio da frequência na literatura técnica, como o método da Máxima Verossimilhança (*Maximum Likelihood*), o qual é apresentado Schoukens e Pintelon (1991) e estendido por Guillaume et al. (1998) para fazer uso apenas dos dados das saídas medidas.

Segundo Ibrahim e Mikulcik (1973), os métodos no domínio da frequência possuíam limitações que impediam a aplicação em certos tipos de estruturas. Era necessário assumir que a estrutura testada fosse fracamente amortecida e que os modos fossem suficientemente espaçados de modo que a aproximação em sistemas com um grau de liberdade seja adequada. Essas limitações levaram ao surgimento de métodos no domínio do tempo.

Spitznogle e Quazi (1970) apontam que o emprego da transformada discreta de Fourier tem como hipótese básica que o sinal seja repetitivo ou infinito, mas ele sempre é truncado na análise prática de sinais. Desenvolveram um algoritmo para identificação de sinais não amortecidos ou com amortecimento exponencial, o qual foi chamado de Método da Exponencial Complexa (*Complex Exponential* – CE). Trata-se de um método SISO (uma entrada e uma saída – *single input single output*) pois a cada vez uma única função de resposta impulsiva é analisada. O CE foi estendido à situação SIMO (uma entrada e múltiplas saídas – *single input multiple output*) por Brown et al. (1979) recebendo o nome de Método Exponencial Complexo por Mínimos Quadrados (*Least Squares Complex Exponential* – LSCE).

Ibrahim e Mikulcik (1973) propuseram um método para identificação de sistemas a partir da resposta em vibração livre amortecida, o qual passou a ser chamado de Método de Ibrahim no Domínio do Tempo (*Ibrahim Time Domain – ITD*). As características dinâmicas são obtidas diretamente do domínio do tempo, podendo ser empregado em estruturas com amortecimento alto e frequências naturais próximas. O método requer que os deslocamentos, as velocidades e as acelerações sejam medidas, a partir das quais a matriz do sistema no espaço de estado é determinada. Os autovalores dessa matriz são empregados para a determinação das frequências naturais e razões de amortecimentos e os autovetores possibilitam a determinação das formas modais. Autovalores reais correspondem a modos superamortecidos e autovalores complexos correspondem a modos subamortecidos e ocorrem em pares conjugados. Os autores modificaram o método em 1977 para que apenas um dos sinais (ou deslocamentos ou velocidades ou acelerações) pudesse ser empregado e obtiveram êxito na determinação das frequências naturais de uma estrutura com relação ruído/sinal de 22%.

Vold et al. (1982) desenvolveram o Método da Polireferência no Domínio do Tempo (*Polyreference Time Domain – PTD*). Este foi o primeiro método a empregar dados de múltiplas entradas e múltiplas saídas (*multiple input multiple output – MIMO*) e utiliza as séries temporais da resposta em vibração livre após a aplicação de impulsos unitários nas múltiplas entradas. Ele é uma extensão à situação MIMO do método LSCE.

Em 1985, Juang e Pappa introduziram o Algoritmo de Realização de Sistemas (*Eigensystem Realization Algorithm – ERA*) como uma extensão do algoritmo de Ho-Kalman e com o objetivo de criar um modelo matemático para descrição do movimento de grandes estruturas espaciais. O algoritmo é apresentado no espaço de estado e o processo de construção da sua representação a partir de dados experimentais é chamado de realização de sistemas. Os autores buscam encontrar o modelo que possui a menor dimensão no espaço de estado que represente uma relação entrada-saída com uma precisão previamente especificada. Este problema é chamado de realização mínima e é equivalente ao problema de representação envolvendo uma sequência de matrizes reais chamadas de parâmetros de Markov (funções de resposta ao impulso). Os parâmetros de Markov descrevem no domínio do tempo a resposta em vibração livre ao impulso a partir do modelo no espaço de estado. O algoritmo inicia agrupando as respostas livres ao impulso em uma matriz bloco Hankel e, com os conceitos de controlabilidade e observabilidade, faz a decomposição em valores singulares dessa matriz. Analisando seus valores singulares não nulos, determina-se seu posto e com ele constrói-se um

modelo de ordem reduzida. A partir das matrizes reduzidas chega-se num problema de autovalores e autovetores, obtendo-se as frequências naturais, razões de amortecimento e formas modais. Os autores realizaram medições na sonda espacial Galileo com 164 acelerômetros, diversos excitadores e conseguiram identificar 34 modos com frequências entre 13 Hz e 50 Hz.

As técnicas de identificação descritas anteriormente utilizam a resposta em vibração livre da estrutura. Nestes casos, apesar das entradas não precisarem ser medidas, alguma excitação inicial é aplicada na estrutura e então a resposta livre amortecida é medida assim que a excitação é removida. Existem situações, entretanto, em que as excitações não podem ser controladas ou aplicadas. Por exemplo, para o teste de uma estrutura durante sua operação a aplicação de uma força externa pode causar uma interrupção no seu funcionamento. Outro exemplo é o caso da resposta medida durante a fase de voo de aeronaves, na qual não há o conhecimento completo da excitação (IBRAHIM, 1977).

Cole (1968) desenvolveu o Método de Decremento Aleatório (*Random Decrement – RD*) na NASA com o intuito de identificar as características dinâmicas de estruturas espaciais submetidos a vibrações ambientais. As medições obtidas eram tão complicadas que o único parâmetro que um observador poderia obter visualizando-as era o nível do sinal. Essa situação era particularmente crítica nas regiões de ocorrência de drapejamento (*flutter*), nas quais a falta de conhecimento sobre o amortecimento e a frequência poderiam levar à destruição do modelo. Ele criou um método simples e direto de transformar as respostas temporais em algo mais significativo para o observador. Basicamente transforma-se o sinal aleatório numa resposta em vibração livre amortecida e desta obtém-se os parâmetros da estrutura.

O decremento aleatório passou a ser bastante utilizado nos testes de drapejamento, nos quais a determinação das frequências naturais e razões de amortecimentos é o objetivo primário e para estes uma série temporal é o suficiente. Ibrahim (1977) expandiu o método de decremento aleatório para múltiplas respostas medidas sem alterar a correlação entre os sinais. Das medições feitas, seleciona-se uma como referência, para a qual as condições de gatilho são aplicáveis, convertendo a resposta aleatória em funções de resposta ao impulso. Como consequência, todos os métodos de identificação que empregam a resposta livre amortecida como entrada também podem ser empregados para a identificação de sistemas.

James et al. (1995) elaboraram a Técnica de Excitação Natural (*Natural Excitation Technique* – NExT) com o objetivo de identificar as características dinâmicas de aerogeradores. Essas estruturas se comportam de maneira muito diferente na condição desligada e em funcionamento. Especificamente o amortecimento total (aeroelástico e estrutural) depende da rotação da turbina e o seu conhecimento é importante para prever a vida em fadiga e reduzir a resposta ressonante. Admite-se que as forças são do tipo ruído branco. Calculam-se as funções de autocorrelação e correlação cruzada a partir das séries temporais medidas. Essas funções são expressas como a soma de decaimentos senoidais. Cada decaimento possui uma frequência e amortecimento que são idênticos aos estruturais. Em seguida, o método trata as funções de correlação como funções de resposta ao impulso e a partir delas realiza a identificação empregando os métodos descritos anteriormente, por exemplo, quando o ERA é utilizado, recebe o nome de NExT/ERA.

Os modelos autorregressivos com média móvel (*Auto-Regressive Moving Average* – ARMA) podem ser empregados para a identificação de estruturas submetidas a ações ambientais. No caso multivariado, o modelo também é chamado de modelo vetorial autorregressivo com média móvel (*Auto-Regressive Moving Average Vector* – ARMAV). A representação geral utilizada para modelar sistemas dinâmicos lineares invariantes no tempo excitados por uma entrada determinística é denominado de modelo autorregressivo com média móvel com entrada externa (*Auto-Regressive Moving Average with eXternal input* – ARMAX). Estas representações não contém o estado do sistema. Andersen (1997) emprega o Método de Predição de Erro (*Prediction Error Method* – PEM) para estimar os coeficientes do modelo ARMA. O processo de determinação dos preditores ótimos para um dado modelo resulta em um problema de otimização iterativo altamente não linear.

O método das Variáveis Instrumentais (*Instrumental Variable* – IV) corresponde ao método da Polireferência no Domínio do Tempo (PTD) após substituir a resposta ao impulso pelas covariâncias das saídas. O IV emprega a representação ARMA e identifica os coeficientes da parte AR apenas, evitando a não-linearidade causada pela parte MA. Pode-se mostrar que para a determinação das características dinâmicas da estrutura, apenas os coeficientes da parte AR são necessários. A ideia é ajustar os coeficientes para os dados das medições, resolvendo um sistema de equações sobredeterminado por mínimos quadrados. É importante salientar que o IV é uma técnica robusta para excitações não-estacionárias (PEETERS, 2000).

Os métodos de Identificação Estocástica de Subespaços (*Stochastic Subspace Identification – SSI*) visam identificar um modelo estocástico em espaço de estado para apenas saídas medidas. O nome subespaço se refere ao fato do modelo ser obtido a partir do espaço das linhas e colunas de certas matrizes. No método SSI baseado nas Covariâncias (*Covariance-driven Stochastic Subspace Identification – SSI COV*), as funções de covariâncias das saídas são calculadas e agrupadas em uma matriz Toeplitz. Empregando a decomposição em valores singulares dessa matriz, obtém-se as matrizes de observabilidade e controlabilidade, a partir das quais as matrizes do sistema no espaço de estado são calculadas.

Contrariamente ao SSI COV, o método SSI baseado nas séries Temporais (*Data-driven Stochastic Subspace Identification – SSI DATA*) evita o cálculo das funções de covariância das saídas. Ele é substituído pela projeção das linhas das saídas futuras nas linhas das saídas passadas, a qual é calculada a partir da fatoração QR da matriz bloco Hankel com séries temporais das saídas. De fato, o conceito de covariâncias e projeções estão intimamente ligados. Ambos visam cancelar o ruído. A ideia da projeção é reter toda a informação passada útil para prever o futuro. O método emprega a decomposição em valores singulares da matriz com as projeções para obter a matriz de observabilidade e a sequência de estado no filtro Kalman. As matrizes do sistema são obtidas a partir da sequência de estado do filtro Kalman (OVERSCHEE; MOOR, 1996).

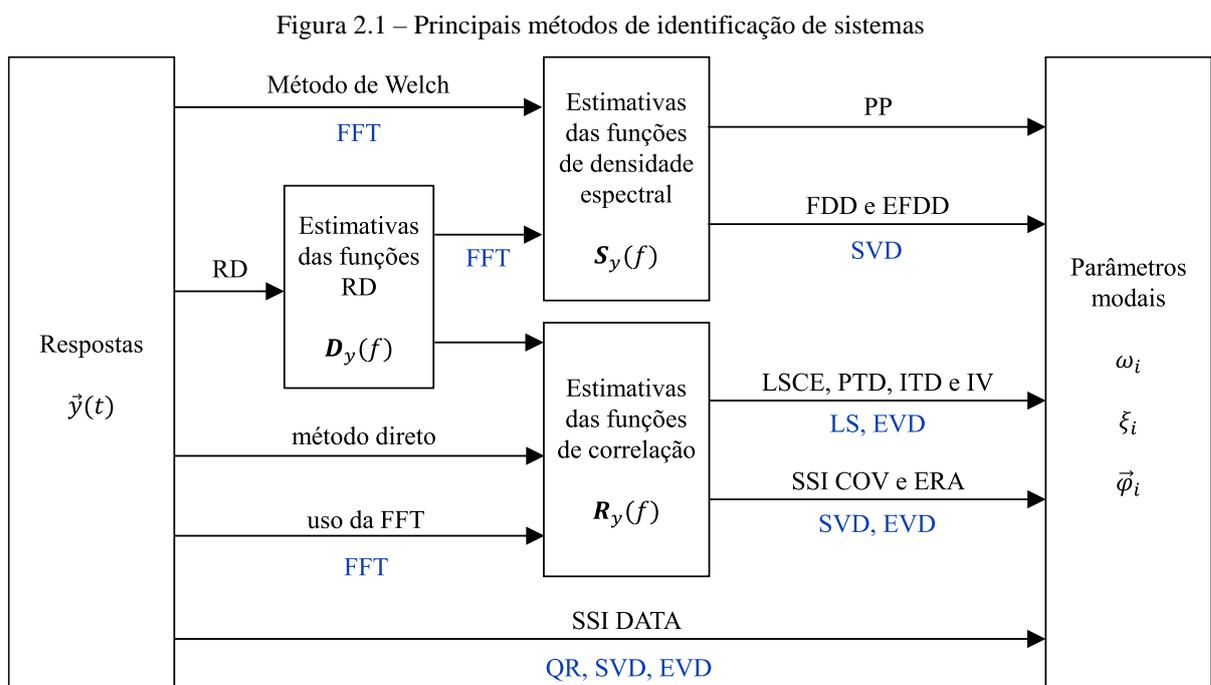
Peeters e Van der Auweraer (2005) propuseram o Método da Polireferência no Domínio da Frequência por Mínimos Quadrados (*polyreference Least-Squares Complex Frequency domain method – p-LSCF*), também conhecido pelo seu nome comercial PolyMAX. O método leva em consideração a similaridade entre as densidades espectrais da resposta de um sistema excitado por ruído branco e a função de resposta em frequência. Os parâmetros do modelo são determinados minimizando uma função de erro, a qual é a diferença entre as densidades espectrais estimadas e as medidas. A principal vantagem da técnica é a sua estabilidade, levando a diagramas de estabilização fáceis de serem interpretados.

Peeters (2000) aponta que, devido a razões históricas, essencialmente os mesmos métodos receberam nomes diferentes na literatura técnica dependendo da informação que empregam. Por exemplo, o método da Polireferência no Domínio do Tempo (PTD) aplicado às covariâncias (ao invés da resposta impulsiva) pode ser considerado um método de Variáveis Instrumentais

(IV) e o Algoritmo de Realização de Sistemas (ERA) aplicado às covariâncias é equivalente ao Método de Identificação Estocástica de Subespaços baseados nas Covariâncias (SSI COV).

Dada as semelhanças entre os métodos, alguns autores procuram descrevê-los numa forma unificada. Allemang et al. (1994) agrupam diversos algoritmos em uma abordagem chamada UMPA (*Unified Matrix Polynomial Approach*). Citam-se também os trabalhos de Zhang et al. (2002) e Zhang (2003).

A Figura 2.1 sintetiza os principais métodos descritos neste capítulo. As técnicas numéricas utilizadas são: o algoritmo da transformada rápida de Fourier (FFT), a decomposição em valores singulares (SVD), ajuste por mínimos quadrados (LS), decomposição em autovalores e autovetores (EVD) e a decomposição QR.



(fonte: adaptada de RODRIGUES, 2004)

2.2 VIBRAÇÕES AMBIENTAIS

Durante os anos sessenta e setenta, a busca de conhecimento sobre as propriedades dinâmicas de estruturas civis como prédios altos e pontes aumentou. Medições em diversos prédios altos, pontes suspensas e estruturas *offshore* foram feitas e utilizadas na identificação do sistema. A medição de grandes estruturas submetidas a ações ambientais mostrou-se uma forma rápida, barata e confiável para sua identificação.

Devido à natureza do teste dinâmico sob ações ambientais, este método possui vantagens frente a outros, como excitações impulsivas e periódicas. Excitações ambientais possuem uma grande faixa de frequências, e assim, teoricamente, podem excitar todos os modos importantes da estrutura. A sua realização também não atrapalha o funcionamento da estrutura e não necessita de equipamentos para aplicação de excitações. Uma das desvantagens é que as ações ambientais não podem ser controladas e medidas diretamente (ANDERSEN, 1997).

A identificação efetuada com registros de respostas em vibrações ambientais é chamada identificação estocástica de sistemas. Muitos estudos foram realizados no âmbito de identificação de estruturas civis, sendo praticamente impossível efetuar uma apresentação exaustiva de todos os trabalhos. Os itens a seguir apresentam alguns estudos aplicados a barragens, pontes, edifícios e torres.

2.2.1 Barragens

Barragens normalmente estão localizadas em áreas remotas onde a acessibilidade é um problema e o transporte de vibradores ao local torna-se muito complicado e caro. Nestes locais, medições sob ações ambientes são os únicos meios práticos para conduzir testes dinâmicos. Eles visam caracterizar dinamicamente a estrutura, identificar a influência do nível da água do reservatório nas frequências naturais, calibrar modelos numéricos e avaliar o envelhecimento da estrutura. Os resultados caracterizam o comportamento dinâmico em baixas amplitudes (elástico-linear). A caracterização da resposta dinâmica de barragens é fundamental para obtenção de resultados experimentais destinados a avaliar a confiabilidade de modelos numéricos a serem utilizados em simulações da resposta devida a ações dinâmicas, das quais sismos são a mais importante (MENDES et al., 2004; BUKENYA et al., 2014; ABDULAMIT et al., 2017).

Na verificação da integridade da estrutura, normalmente emprega-se a inspeção visual, na qual o engenheiro observa a superfície da barragem regularmente. Entretanto pode ser difícil encontrar danos em estruturas complexas ou grandes. Ele pode não ser visível, situando-se abaixo do nível da água ou em contato com o solo. Nesse sentido, é desejável desenvolver um sistema de monitoramento permanente de toda a estrutura com um único sensor. O monitoramento da integridade estrutural baseado em vibrações ambientais é uma solução eficiente, na qual o dano pode ser detectado a partir de mudanças nas frequências naturais (OKUMA et al., 2008).

O primeiro teste documentado sob condições ambientais data de 1986 e foi feito na barragem Contra na Suíça. Os acelerômetros foram posicionados na crista da barragem, as frequências naturais foram determinadas pelo método de Detecção de Picos e as formas modais a partir das funções de resposta em frequência. Oito modos foram detectados entre 1,8 e 4,2 Hz. Os resultados mostraram que as frequências naturais aumentam com a diminuição do nível da água no reservatório e a principal fonte de excitação eram os geradores hidrelétricos (BROWNJOHN et al., 1986).

Diversos estudos foram feitos na barragem Cabril, a mais alta barragem em arco de Portugal, com 132 metros de altura e construída em 1952. Mendes et al. (2004) empregaram 12 acelerômetros uniaxiais e realizaram gravações com 30 minutos de duração. Em outro teste, empregaram quatro acelerômetros, destes, um fixo no centro da barragem (chamado de sensor de referência). A identificação foi feita com o programa comercial ARTeMIS utilizando a Decomposição no Domínio da Frequência (FDD) e Identificação Estocástica de Subespaço (SSI DATA). Com auxílio do diagrama de estabilização foi possível identificar dois modos com frequências muito próximas. Os autores obtiveram boa concordância entre dados experimentais e o modelo numérico em elementos finitos para os quatro primeiros modos de vibração.

Okuma et al. (2008) realizaram um estudo bastante detalhado da barragem Hitotsuse no Japão, visando avaliar a segurança frente a sismos, calibrar modelo em elementos finitos e verificar a sua integridade estrutural. Foram conduzidos dois testes: um de curta duração para identificação das formas modais e outro de longa duração para capturar variações sazonais nas frequências naturais. No primeiro empregaram-se dez acelerômetros (destes, dois de referência) cobrindo 97 locais em 20 medições de 10 minutos cada. No segundo, dois acelerômetros triaxiais foram colocados na crista da barragem e coletaram dados continuamente durante 16 meses e a

identificação foi feita com base em um modelo ARMA. Os autores conseguiram determinar as formas modais dos três primeiros modos. Verificou-se que as propriedades dinâmicas da estrutura se mantiveram muito próximas às obtidas em testes realizados 42 anos antes. Um teste de longo prazo permitiu a identificação da correlação entre temperatura da superfície da barragem, nível da água e frequências naturais. Os autores confirmaram que testes sob ações ambientais podem avaliar de forma precisa as características dinâmicas, dispensando a análise da resposta sísmica ou em vibrações forçadas.

Abdulmit et al. (2017) realizaram medições ambientais em três barragens de contraforte na Romênia, as quais fazem parte de um programa de monitoramento de longo prazo no país, visto que nele existem cerca de 240 barragens construídas majoritariamente entre 1950 e 1990. Os autores realizaram medições de três minutos com frequência de aquisição de 100 Hz em duas condições: com e sem o funcionamento de equipamentos hidromecânicos. Os testes foram repetidos em estações do ano diferentes visando identificar a influência da temperatura e a identificação do sistema foi feita no domínio da frequência.

2.2.2 Pontes

Um dos primeiros estudos documentados de vibrações ambientais em pontes foi realizado por Carden (1937) nas pontes São Francisco-Oakland Bay e Golden Gate. O procedimento experimental foi feito durante e após a construção das pontes utilizando sismógrafos. As vibrações foram causadas pelo vento, água e tráfego de veículos e pessoas.

Felber (1993) aponta a necessidade de validação de modelos numéricos empregados para a determinação da resposta sísmica de pontes. O autor desenvolveu rotinas capazes de obter as características dinâmicas de forma automatizada, rápida e barata no domínio da frequência. Os ensaios realizados em três pontes demonstraram a capacidade do sistema de determinar as frequências naturais e formas modais para vibrações ambientais com amplitude de 0,02 mg até 0,2 g.

Asmussen (1997) descreve e expande a fundamentação teórica, implementa e aplica o método de Decremento Aleatório Vetorial (VRD) para identificação da ponte Queensborough no Canadá, ponte Vestvej na Dinamarca e de um modelo reduzido de ponte. Salienta que a implementação computacional do método é simples e a análise dos dados requer baixo custo computacional. Quando comparada com a técnica RD original, o VRD mostrou-se mais rápido

e confiável para identificação de estruturas com muitos pontos de medição. O autor verificou que a determinação das densidades espectrais a partir das funções de decremento aleatório (ao invés dos sinais das respostas) resulta em espectros mais fáceis de serem interpretados.

Peeters (2000) analisou os dados de um extensivo estudo da Ponte Z-24 entre Bern e Zürich na Suíça. As expedições incluíram a instrumentação da estrutura em nove configurações, cada uma com 33 sensores em locais diferentes e cinco fixos (referências). Empregaram-se excitações ambientais (tráfego, vento e pedestres), induzidas por dois vibradores e por impacto. O autor empregou os métodos de Detecção de Picos (PP), Variáveis Instrumentais (IV) e Identificação Estocástica de Subespaço com Séries Temporais (SSI DATA) e concluiu que o método SSI é o mais preciso e preferível. Todos os tipos de vibrações levaram a características dinâmicas semelhantes, mas as vibrações ambientais forneceram valores com maior dispersão. Além das expedições, houve o monitoramento contínuo da estrutura durante um ano. A cada hora mediram-se temperatura do ar, velocidade do vento, umidade, dilatação da ponte e temperatura do solo, concreto e asfalto. Destes, apenas a temperatura influenciou significativamente as características dinâmicas da ponte. O autor constatou uma relação bilinear entre a temperatura e frequência natural. Com o aumento da temperatura, a rigidez da ponte diminuiu. Para temperaturas abaixo de 0°C verificou-se que o asfalto contribuiu de maneira significativa para a rigidez, o que explica o comportamento bilinear. Concluiu que a estimativa das características dinâmicas por excitações ambientais é adequada para estruturas grandes, especialmente com o aprimoramento dos sistemas de aquisição e algoritmos de identificação.

Rodrigues (2004) sintetiza diferentes métodos de identificação estocástica de sistemas, aplica-os em inúmeras estruturas civis e compara os resultados experimentais com modelos analíticos, além de expor aspectos relativos ao planejamento e à realização dos ensaios. O autor relata a experiência adquirida durante quatro anos de trabalho na identificação de estruturas submetidas a ações ambientais no Laboratório Nacional de Engenharia Civil em Porto, Portugal. Os ensaios foram solicitados com o intuito de validar as características dinâmicas das estruturas antes da sua entrada em operação e na inspeção de estruturas existentes. Resultados de ensaios em 13 pontes, um edifício e uma barragem são apresentados, apontando os erros e acertos de cada um deles. O autor constatou que os equipamentos existentes têm a capacidade de medir, com uma resolução muito boa, respostas de estruturas com amplitudes extremamente pequenas, permitindo, assim, identificá-las.

Cunha et al. (2017) relatam a experiência de mais de 20 anos de trabalho junto ao Laboratório de Vibrações e Monitoração da Faculdade de Engenharia da Universidade do Porto. Inúmeras estruturas de grande porte foram identificadas, como a Ponte Vasco da Gama, a Ponte Luiz I, a Ponte Internacional do Guadiana e o Viaduto de Millau. Este último é o viaduto mais alto do mundo, com torres de até 343 m de altura e oito vãos estaiados totalizando 2460 m de comprimento. Nele, conforme Caetano et al. (2007), realizaram-se ensaios de vibrações ambientais com quatro sismógrafos sincronizados via GPS, sendo dois fixos no vão central e dois sucessivamente posicionados em 26 locais diferentes. Com o emprego dos métodos EFDD e SSI COV, os autores identificaram 20 modos com frequência entre 0,1 e 0,8 Hz. Observaram boa concordância entre as frequências naturais e formas modais identificadas e as fornecidas pelo modelo numérico utilizado no projeto. Concluíram que as técnicas de identificação empregadas tiveram desempenho excelente, mesmo para amplitudes de vibração extremamente baixas.

Limas (2007) realizou ensaios no túnel de vento Prof. Joaquim Blessmann da Universidade Federal Rio Grande do Sul (UFRGS) de oito modelos seccionais de tabuleiros de pontes visando investigar as condições de estabilidade com relação ao drapejamento. Empregou o método de Decremento Aleatório (RD) e o ajuste das curvas de decaimento por mínimos quadrados para a determinação das frequências e razões de amortecimento de cada sistema. Estas características dinâmicas foram utilizadas para a identificação dos coeficientes de drapejamento, também conhecidos como derivativos aerodinâmicos e, a partir destes, a velocidade crítica do vento para cada sistema vento-ponte.

Também no âmbito da UFRGS, Miguel (2007) realizou simulações numéricas em MatLab de uma ponte treliçada tipo Warren contínua utilizando dois algoritmos e três variantes do método SSI DATA para a determinação dos parâmetros modais. Foram empregados dois tipos de excitações, ambiental e impulsiva, e obtiveram-se resultados bastante similares em todos os algoritmos. Em seguida, o autor empregou o Método dos Vetores de Localização de Dano (*Damage Location Vectors Method*) para a detecção de danos em barras com rigidez reduzida propositalmente obtendo êxito na localização das barras danificadas. Utilizou dados experimentais para estudar a correlação entre temperatura e frequência natural com regressão linear e redes neurais e concluiu que ambos forneceram resultados semelhantes.

Destacam-se os trabalhos de Braido (2014) e Muliterno (2020) realizados junto à Universidade de Passo Fundo em pontes rodoviárias no Estado do Rio Grande do Sul. Braido (2014) realizou ensaios com um celular munido de acelerômetro para a obtenção das propriedades dinâmicas de dois viadutos. O autor mediu a resposta após a passagem de veículos e empregou os métodos de detecção de picos e decaimento logarítmico. Comparou os resultados obtidos experimentalmente com modelos numéricos em elementos finitos e concluiu que celulares mostram-se como uma forma alternativa para a identificação das propriedades dinâmicas de obras de arte especiais.

Muliterno (2020) realizou ensaios quase-estáticos e dinâmicos em pontes rodoviárias visando a obtenção de fatores de amplificação dinâmica e a comparação com os coeficientes de impacto normativos. Empregaram-se dois celulares com acelerômetros e frequência de aquisição de 500 Hz, além de um sistema composto por acelerômetros de alta precisão com frequência de aquisição de 600 Hz para fins de comparação. A autora identificou as frequências naturais das estruturas por meio do método de detecção de picos. Observou-se boa convergência entre os resultados, demonstrando que os celulares podem ser empregados de forma alternativa, simples e econômica, para a avaliação da amplificação dinâmica em pontes e viadutos.

Diversas expedições foram feitas na Ponte Humber, localizada no Reino Unido. Trata-se de uma ponte pênsil, aberta em julho de 1981, com vão principal de 1410 metros e vãos secundários de 280 m e 530 m. A primeira instrumentação da ponte data de 1985 (BROWNJOHN et al., 1987) e utilizou três acelerômetros, dois quilômetros de cabos e equipamentos para gravação e análise. Usando o método de detecção de picos, os autores detectaram dezenas de modos de vibração abaixo de 2 Hz.

Outra investigação foi feita em julho de 2008 (BROWNJOHN et al., 2010) com uma instrumentação muito robusta e acessível: gravadores autônomos com relógios sincronizados via GPS e aquisição de dados sem fio representando o estado da arte da análise modal. Dez acelerômetros triaxiais foram empregados, dois dos quais foram mantidos fixos (referências) e os demais cobriram 76 posições ao longo dos vãos e torres em 33 configurações. Os autores concentraram-se nos modos verticais, laterais e torcionais resultando em um modelo com $76 \times 3 = 228$ graus de liberdade. Numa primeira análise, com o método de Detecção de Picos foram identificados mais de 30 modos abaixo de 1 Hz. Por causa da grande quantidade de modos nesse intervalo, os autores resolveram restringir o estudo para frequências entre 0 e 1 Hz. Usaram os

métodos NExT/ERA, SSI COV e p-LSCF e não encontraram diferenças sistemáticas nas frequências de todas as três técnicas. A identificação dos modos laterais mostrou-se mais difícil do que os verticais, mas, mesmo assim, foi possível detectar o primeiro modo lateral com frequência de 0,056 Hz (período de 18 s). Dos três métodos, o SSI COV mostrou-se melhor pois foi capaz de identificar praticamente todos os modos. O método NExT/ERA ficou em segundo lugar, embora tenha fornecido maiores valores de amortecimento.

2.2.3 Vigas, pórticos e edifícios

Trifunac (1970) realizou testes sob ações ambientais em um prédio de 39 pavimentos na cidade de Los Angeles. Foram empregados quatro sismógrafos, dois destes mantidos fixos no topo do prédio e dois posicionados em diversos pavimentos. Cada configuração foi medida por cinco minutos. Empregando o método de detecção de picos, o autor conseguiu identificar 13 modos de translação e sete modos de torção. As razões de amortecimento situaram-se entre 1,5% e 2,0% e mostraram-se ser independentes da frequência natural.

Gilles (2010) participou de um projeto para determinação das características dinâmicas de edifícios em Montreal. As frequências naturais, formas modais e amortecimentos de 39 prédios foram determinadas a partir de excitações devidas ao vento e microtremores empregando o método de Decomposição no Domínio da Frequência. Cada edifício foi instrumentado em dois ou três dias e na maioria identificaram-se os primeiros seis modos de vibração. Um sensor de referência foi mantido fixo no topo, posicionado longe do centro elástico da estrutura visando capturar a maioria dos modos de interesse. O outro sensor foi movido para localizações diferentes, onde as séries temporais com 5 minutos de duração foram gravadas. Os relógios dos sensores foram sincronizados via GPS e, quando possível, buscou-se iniciar a gravação no mesmo instante de tempo manualmente. Os autores observaram que as medições resultaram aproximadamente simultâneas e necessitaram tratamento posterior visando obter sincronização mais precisa.

Brownjohn (2002) identificou as características dinâmicas de dois edifícios em Singapura. No primeiro, um sistema de monitoramento no topo e na base foram instalados num programa de longa duração. Neste foi observado que as frequências naturais diminuiram um pouco em relação as medições realizadas cinco anos antes. Tal diferença foi atribuída ao aumento da massa do edifício devido a sua ocupação. No segundo edifício estudado empregou-se um

sistema de monitoramento sem fio com dois sensores fixos no topo e 14 acelerômetros em duas escadas de cada andar ímpar, totalizando 138 graus de liberdade. Por causa da fraca excitação do vento durante a maioria dos testes, interferências mecânicas e elétricas não identificadas, muitas medições foram repetidas e mesmo assim em certos locais apenas ruído foi detectado. O autor identificou 12 modos de vibração e concluiu que o NExT/ERA forneceu bons resultados, mostrando-se uma ferramenta poderosa.

Brownjohn (2002), em Singapura, relata que poucos proprietários permitem a publicação dos resultados e a maioria solicita que estes sejam mantidos em sigilo. Gilles (2010) conseguiu autorização de 39 proprietários de edifícios de Montreal para divulgar os resultados desde que o nome do prédio não fosse divulgado. Al-Nimry et al. (2014), na Jordânia, relata que a falta de vontade da comunidade local impôs inúmeras limitações na seleção de prédios.

Giraldo et al. (2009) realizaram estudos numéricos e experimentais em um modelo reduzido de um edifício com quatro pavimentos na escala 1:3. Mil simulações foram feitas com base num modelo em elementos finitos e com diversas relações ruído/sinal. As características modais foram obtidas com os métodos NExT/ERA, PEM e SSI DATA. Todos os algoritmos identificaram de maneira adequada os seis modos de vibração. As frequências obtidas apresentaram pequena dispersão (0,5%) enquanto as razões de amortecimento apresentaram grande dispersão (20% até 50%). Os autores observaram que o aumento do ruído não afetou as frequências naturais e amortecimentos, apenas influenciou a precisão das formas modais. Em seguida, realizaram estudo experimental com teste de impacto e identificação pelo algoritmo ERA e ações ambientais com o método SSI DATA. Constataram concordância entre os resultados dos dois métodos. Os resultados indicaram que o algoritmo SSI DATA é o melhor na identificação modal, apresentando menor dispersão de resultados e demonstrando sua robustez em medições com alta relação ruído/sinal.

Amani et al. (2004) apresentam um método para identificar as características dinâmicas da estrutura submetidas a ações ambientais baseado na teoria de modos normais e compara-o com o SSI COV. Os autores estudaram numericamente um pórtico plano com seis vãos e seis pavimentos cujas forças aplicadas são do tipo ruído branco. Também elaboraram estudo experimental de um edifício de seis pisos em escala 1:50 de alumínio, aplicaram uma aceleração na base e mediram a resposta em cada pavimento com acelerômetros. Concluem que os métodos

forneceram bons resultados, com pequena dispersão (erro inferior a 1%) e que este erro era maior nos modos de vibração com frequência mais alta.

Miguel et al. (2009) identificaram as primeiras cinco frequências naturais de uma viga metálica em balanço com seção retangular e um acelerômetro piezoelétrico na ponta. Foram realizados 15 testes de vibrações livre causadas por impactos em diferentes locais e a identificação foi feita com o método SSI DATA. Os autores compararam as frequências identificadas com as fornecidas por um modelo em elementos finitos e pela teoria de vibrações de sistemas contínuos. Os resultados teóricos e numéricos foram praticamente iguais e muito próximos aos obtidos pelo método de identificação de sistemas.

Zeni (2018) desenvolveu um estudo numérico e experimental para identificação modal e localização de dano em modelos reduzidos de vigas e pórticos planos de um vão e três pavimentos. As características dinâmicas foram determinadas pelo método SSI DATA e a detecção do dano com *Backtracking Search Algorithm*. Obteve resultados satisfatórios na localização das barras com rigidez reduzida para os casos numéricos e experimentais. O autor concluiu que o algoritmo SSI DATA é uma ferramenta extremamente robusta, capaz de identificar parâmetros com baixa variabilidade mesmo em sinais com ruído e a partir de vibrações ambientais.

2.2.4 Torres

Em um estudo icônico, Castellaro et al. (2016) determinaram as características dinâmicas da Torre Eiffel com apenas um sismógrafo situado dentro de uma sacola de compras para não chamar a atenção dos visitantes. A Torre é alta (324 m) e leve, possuindo uma forma estrutural única, com grande rigidez e baixa massa. Realizaram duas medições de 10 minutos em cada uma das três plataformas e caracterizaram o comportamento dinâmico do solo em três locais nos arredores com um tromômetro. A partir das medições identificaram seis modos de vibração, sendo quatro de flexão (0,32 Hz, 1 Hz, 1,4 Hz e 1,7 Hz) e dois de torção (1,2 Hz e 2,1 Hz). Constataram que a interação solo-estrutura é fundamental para a correta representação do comportamento dinâmico principalmente do primeiro modo. Em seguida elaboraram um modelo em elementos finitos com base nos desenhos originais de Gustave Eiffel e utilizaram os dados experimentais para sua calibração. A partir dele obtiveram a resposta dinâmica da

estrutura frente a ação de terremotos e ventos extremos. Os autores mostraram que uma grande estrutura pode ser caracterizada de maneira rápida, passiva e barata em condições operacionais.

Miguel et al. (2000) realizaram estudos numéricos para a identificação de duas torres de linhas de transmissão. A primeira é uma torre tronco-piramidal, autoportante, com 33,4 m de altura e abertura na base de 5 m. Simulou-se a sua identificação a partir da resposta impulsiva com três variantes do método SSI DATA e concluiu-se que a variante *principal component* (PC) é a melhor pois foi capaz de identificar os modos de forma mais precisa e com menor variabilidade. A outra torre estudada é uma torre japonesa com 82 m de altura, na qual simulou-se a sua identificação a partir da resposta impulsiva e devida a excitações aleatórias. Novamente empregou-se o método SSI DATA mas com dois algoritmos e três variantes, sendo que todos os resultados obtidos foram muito próximos entre si.

Liu et al. (2012) elaboraram uma versão modificada do SSI COV para identificação de sistemas em tempo real, chamada SSA-SSI-COV, e aplicaram ambos os métodos na Torre de Cantão, China, com altura total de 610 metros. Nela foi implementado um sistema de monitoramento de longo prazo com mais de 600 sensores de diversos tipos. Os autores analisaram dados de 20 acelerômetros uniaxiais em oito níveis, os quais foram gravados em dois dias. Detectaram 18 modos com frequência inferior a 2,3 Hz sendo o primeiro modo com 0,0958 Hz e o segundo com 0,1391 Hz. Um sismo ocorreu durante a última gravação e notou-se que um modo com frequência natural de 1 Hz caiu para 0,9 Hz devido a resposta não linear da torre durante o evento. Após 100 segundos do seu fim, a frequência voltou para 1 Hz. Comparando as características dinâmicas antes e após o sismo, os autores não observaram mudanças significativas.

Diversos estudos foram feitos em torres históricas na Europa. Na maioria deles buscou-se obter as propriedades dinâmicas da estrutura e, em seguida, calibrar modelos em elementos finitos. Gentile e Saisi (2007) avaliaram o campanário da Catedral de Monza, concluído em 1605, com 74 m de altura e paredes com 1,40 m de espessura feitas de tijolos maciços. Conduziram testes sob ações ambientais com 11 sensores. Velocidades e acelerações foram medidas. Devido ao baixo nível da excitação no momento dos testes (a maior velocidade detectada foi 0,15 mm/s), o uso de velocidades forneceu melhores resultados. Empregaram os métodos PP e FDD, os quais forneceram resultados semelhantes. Concluíram que a avaliação dinâmica de torres é uma

abordagem promissora para avaliação de dano desde que exista um levantamento geométrico preciso da estrutura.

3 MODELOS DINÂMICOS

Neste capítulo são apresentados modelos matemáticos para a descrição do comportamento dinâmico das estruturas. Há diversas maneiras de representar o mesmo modelo matemático, ou seja, há várias formas em que as equações que descrevem o comportamento do sistema podem ser escritas e cada uma delas é chamada de representação.

Inicialmente descrevem-se as características dos modelos empregados neste trabalho. Procedese para apresentação dos modelos em elementos finitos, bastante difundidos na Engenharia Civil. Em seguida, os modelos de espaço de estado são mostrados, os quais foram inicialmente empregados na Engenharia Elétrica. Por fim, apresentam-se os modelos autorregressivos com média móvel.

3.1 TIPOS DE MODELOS

Um modelo matemático é uma representação aproximada da estrutura real. A fim de desenvolver modelos aproximados são feitas simplificações. No presente trabalho empregam-se modelos:

- a) lineares;
- b) dinâmicos;
- c) invariantes no tempo;
- d) de parâmetros concentrados;
- e) em tempo discreto;
- f) multivariáveis.

Admite-se que o sistema se comporta de maneira linear, o que normalmente é verificado observando-se o seu comportamento em uma faixa relativamente estreita de operação. Um sistema é linear se ele satisfaz o princípio da superposição: a resposta a uma combinação linear de excitações é igual a combinação linear das respostas às excitações individuais e a resposta a

uma excitação multiplicada por uma constante é igual à constante multiplicada pela resposta da excitação.

Outra consideração importante é a invariância no tempo: o comportamento do sistema modelado não varia com o tempo. Isso significa que a dinâmica que está regulando a evolução temporal é a mesma sempre. Matematicamente, um sistema é invariante se uma excitação causa sempre a mesma resposta, independentemente do instante em que ela for aplicada. A maioria dos sistemas reais apresenta variações ao longo do tempo, as quais podem ocorrer devidas à variação na temperatura ou ao envelhecimento por exemplo.

Um modelo é chamado de modelo a parâmetros concentrados quando descreve o comportamento do sistema em pontos específicos do espaço. Esses modelos são descritos por equações diferenciais ordinárias, ao invés de equações diferenciais parciais para o caso de sistemas a parâmetros distribuídos. Assim, pressupõe-se que as variáveis de interesse variam apenas com o tempo e não no espaço.

Modelos estáticos relacionam variáveis sem quantificar sua dependência temporal, enquanto os modelos dinâmicos permitem a determinação da evolução temporal do sistema. Os estáticos normalmente são descritos por equações algébricas ao passo que os dinâmicos são descritos por equações diferenciais.

Modelos dinâmicos em tempo contínuo são descritos por equações diferenciais e representam a evolução do sistema continuamente no tempo. Os modelos dinâmicos em tempo discreto representam a evolução do sistema em instantes discretos e são descritos por equações a diferenças. A maioria dos métodos estocásticos de identificação resultam em modelos discretos no tempo ainda que praticamente todos os sistemas reais sejam contínuos no tempo.

3.2 ELEMENTOS FINITOS

A resposta dinâmica de um sistema com n graus de liberdade é determinada através da solução da equação matricial de equilíbrio dinâmico:

$$\mathbf{M} \ddot{\vec{u}}(t) + \mathbf{C}_f \dot{\vec{u}}(t) + \mathbf{K} \vec{u}(t) = \vec{F}(t) \quad (3.1)$$

Sendo:

\mathbf{M} – matriz de massa, $\mathbf{M} \in \mathbb{R}^{n \times n}$;

\mathbf{C}_f – matriz de amortecimento, $\mathbf{C}_f \in \mathbb{R}^{n \times n}$;

\mathbf{K} – matriz de rigidez, $\mathbf{K} \in \mathbb{R}^{n \times n}$;

$\vec{u}(t), \dot{\vec{u}}(t), \ddot{\vec{u}}(t)$ – vetores deslocamentos, velocidades e acelerações generalizadas, respectivamente, $\vec{u}(t), \dot{\vec{u}}(t), \ddot{\vec{u}}(t) \in \mathbb{R}^n$;

$\vec{F}(t)$ – vetor de cargas, $\vec{F}(t) \in \mathbb{R}^n$.

As estruturas são sistemas contínuos com massa distribuída e a princípio possuem infinitos graus de liberdade. Nelas, a equação 3.1 pode ser obtida a partir da aproximação em elementos finitos do sistema com n graus de liberdade. A estrutura é discretizada em um número finito de elementos e, conhecida a sua geometria e as propriedades dos materiais, as matrizes de massa e rigidez são construídas. Como o movimento do sistema é observado nas massas, a matriz de massa é uma matriz diagonal e devido ao teorema de Maxwell a matriz de rigidez é simétrica e positivo-definida. A determinação da matriz de amortecimento, entretanto, é praticamente inviável devido ao desconhecimento dos parâmetros que governam o seu comportamento. Na maioria dos casos ela é estimada a partir das razões de amortecimento obtidas experimentalmente em estruturas similares.

3.2.1 Vibrações livres não amortecidas

Deseja-se estudar a determinação das frequências naturais (autovalores) e das formas modais (autovetores) da equação 3.1. Parte-se do caso mais simples, correspondente a um sistema não amortecido em vibração livre:

$$\mathbf{M} \ddot{\vec{u}}(t) + \mathbf{K} \vec{u}(t) = 0 \quad (3.2)$$

Utilizando o método da superposição modal, escreve-se o vetor deslocamento como a soma dos deslocamentos modais:

$$\vec{u}(t) = \sum_{i=1}^n u_i(t) \vec{\varphi}_i = \Phi \vec{u}_m(t) \quad (3.3)$$

Sendo:

$u_i(t)$ – resposta modal, $u_i(t) \in \mathbb{R}$;

$\vec{\varphi}_i$ – forma modal, $\vec{\varphi}_i \in \mathbb{R}^n$.

Φ – matriz cujas colunas são as formas modais, $\Phi \in \mathbb{R}^{n \times n}$;

$\vec{u}_m(t)$ – vetor das respostas modais, $\vec{u}_m(t) \in \mathbb{R}^n$.

As formas modais $\vec{\varphi}_i$ e as respectivas frequências naturais ω_i são determinadas a partir da solução do problema de autovalores e autovetores:

$$\mathbf{K} \vec{\varphi}_i = \omega_i^2 \mathbf{M} \vec{\varphi}_i \quad (3.4)$$

Os autovetores são agrupados, um em cada coluna, formando a matriz Φ . Os autovalores ω_i , por sua vez, formam a diagonal da matriz $\mathbf{\Omega} = [\omega_i]$. Os autovetores podem ser normalizados de modo que a massa modal seja unitária, resultando:

$$\Phi^T \mathbf{M} \Phi = \mathbf{I}_n \quad (3.5)$$

$$\Phi^T \mathbf{K} \Phi = \mathbf{\Omega}^2 \quad (3.6)$$

Sendo \mathbf{I}_n a matriz identidade de dimensões $n \times n$ ($\mathbf{I}_n \in \mathbb{R}^{n \times n}$).

3.2.2 Amortecimento viscoso proporcional

Pré-multiplicando a equação 3.1 por Φ^T e utilizando a definição 3.3 chega-se a:

$$\Phi^T \mathbf{M} \Phi \ddot{\vec{u}}_m(t) + \Phi^T \mathbf{C}_f \Phi \dot{\vec{u}}_m(t) + \Phi^T \mathbf{K} \Phi \vec{u}_m(t) = \Phi^T \vec{F}(t) \quad (3.7)$$

Sendo:

$\Phi^T \mathbf{M} \Phi = [m_i]$ – matriz diagonal das massas modais, $\Phi^T \mathbf{M} \Phi \in \mathbb{R}^{n \times n}$;

$\Phi^T \mathbf{C}_f \Phi$ – matriz dos amortecimentos modais, $\Phi^T \mathbf{C}_f \Phi \in \mathbb{R}^{n \times n}$;

$\Phi^T \mathbf{K} \Phi = [k_i]$ – matriz das rigidezes modais, $\Phi^T \mathbf{K} \Phi \in \mathbb{R}^{n \times n}$;

$\Phi^T \vec{F}(t) = \vec{F}_m(t)$ – vetor de forças modais, $\vec{F}_m(t) \in \mathbb{R}^n$.

Admitindo-se amortecimento proporcional, assume-se que os autovetores diagonalizem a matriz \mathbf{C}_f :

$$\Phi^T \mathbf{C}_f \Phi = [\backslash c_{i\backslash}] = [\backslash 2\xi_i \omega_i m_{i\backslash}] \quad (3.8)$$

Sendo $\xi_i = c_i/2m_i\omega_i$ a razão de amortecimento crítico.

Assim, a equação matricial resulta em equações diferenciais desacopladas, uma para cada modo:

$$m_i \ddot{u}_i(t) + c_i \dot{u}_i(t) + k_i u_i(t) = F_i(t) \quad (3.9)$$

As quais podem ser resolvidas no domínio do tempo ou da frequência. Após a determinação das respostas modais, utiliza-se a equação 3.3 para o cálculo da resposta da estrutura, sendo possível truncá-la para um número muito menor de termos do que o número de graus de liberdade da estrutura.

Verifica-se que os autovetores são os mesmos do caso não amortecido apresentado anteriormente. A solução para os autovalores λ_i é:

$$\lambda_i, \lambda_i^* = -\xi_i \omega_i \pm j \omega_i \sqrt{1 - \xi_i^2} \quad (3.10)$$

Sendo que o superescrito * denota complexo conjugado e j é a unidade imaginária ($j^2 = -1$).

Escrevendo os autovalores na forma $\lambda_i, \lambda_i^* = a \pm bj$, as frequências naturais amortecida $\omega_{d,i}$ e não amortecida ω_i assim como a razão de amortecimento crítico ξ_i podem ser obtidas:

$$\omega_{d,i} = b \quad (3.11)$$

$$\omega_i = \sqrt{a^2 + b^2} = |\lambda_i| \quad (3.12)$$

$$\xi_i = -\frac{a}{\sqrt{a^2 + b^2}} = -\frac{\text{Re}(\lambda_i)}{|\lambda_i|} \quad (3.13)$$

O amortecimento estrutural pode ser descrito a partir das razões de amortecimento modais ξ_i , as quais podem ser determinadas experimentalmente. Com base na equação 3.8, determina-se a matriz de amortecimento completa:

$$\mathbf{C}_f = \Phi^{-T} \left[\sqrt{2\xi_i\omega_i m_i} \right] \Phi^{-1} = \mathbf{M} \Phi \left[\sqrt{1/m_i} \right] \left[\sqrt{2\xi_i\omega_i} \right] \Phi^T \mathbf{M} \quad (3.14)$$

No amortecimento de Rayleigh, um caso especial de amortecimento proporcional, a matriz de amortecimento é uma combinação linear das matrizes de massas e de rigidez:

$$\mathbf{C}_f = \alpha \mathbf{M} + \beta \mathbf{K} \quad (3.15)$$

Sendo α e β dois escalares.

3.2.3 Amortecimento viscoso não proporcional

Geralmente não há necessidade de expressar o amortecimento do sistema por meio da matriz de amortecimento porque a representação pelas razões de amortecimento modais ξ_i é suficiente. A hipótese de amortecimento proporcional não é válida para sistemas não lineares, nos quais as formas modais variam de acordo com a variação da rigidez, e na análise de sistemas lineares com amortecimento não proporcional (como em estruturas com amortecedores). No caso de amortecimento não proporcional, os autovetores do sistema não amortecido são diferentes do sistema amortecido (CLOUGH; PENZIEN, 2003).

Para a determinação dos autovetores deve-se alterar a equação de movimento 3.1. Adiciona-se o termo $\mathbf{M}\dot{\vec{u}}(t)$ e definem-se:

$$\vec{x}(t) = \begin{bmatrix} \vec{u}(t) \\ \dot{\vec{u}}(t) \end{bmatrix} \quad (3.16)$$

$$\mathbf{P} = \begin{bmatrix} \mathbf{C}_f & \mathbf{M} \\ \mathbf{M} & \mathbf{0} \end{bmatrix} \quad (3.17)$$

$$\mathbf{Q} = \begin{bmatrix} \mathbf{K} & \mathbf{0} \\ \mathbf{0} & -\mathbf{M} \end{bmatrix} \quad (3.18)$$

Sendo $\vec{x}(t)$ o vetor de estado, o qual possui $n_2 = 2n$ linhas.

A partir das definições anteriores, pode-se reescrever o sistema de n equações diferenciais de segunda ordem (3.1), resultando num sistema de n_2 equações diferenciais de primeira ordem:

$$\mathbf{P}\dot{\vec{x}}(t) + \mathbf{Q}\vec{x}(t) = \begin{bmatrix} \vec{F}(t) \\ 0 \end{bmatrix} \quad (3.19)$$

O problema de autovalores e autovetores correspondente é:

$$\mathbf{P}\Psi\Lambda_C + \mathbf{Q}\Psi = 0 \quad (3.20)$$

Sendo:

Ψ – matriz que contém os autovetores como colunas, $\Psi \in \mathbb{R}^{n_2 \times n_2}$;

Λ_C – matriz cuja diagonal contém os autovalores, $\Lambda_C \in \mathbb{R}^{n_2 \times n_2}$.

Pode-se mostrar que as matrizes Λ_C e Ψ têm a seguinte forma:

$$\Lambda_C = \begin{bmatrix} \lambda_i \\ \lambda_i \end{bmatrix} = \begin{bmatrix} \Lambda & 0 \\ 0 & \Lambda^* \end{bmatrix} \quad (3.21)$$

$$\Psi = \begin{bmatrix} \Theta & \Theta^* \\ \Theta\Lambda & \Theta^*\Lambda^* \end{bmatrix} \quad (3.22)$$

Sendo Λ e Θ as matrizes que contém, respectivamente, os autovalores e autovetores do problema de segunda ordem original.

Se todos os autovalores forem pares complexos conjugados, o sistema é subamortecido. Assumindo este comportamento, os pares de autovalores complexos podem ser escritos na mesma forma do caso com amortecimento proporcional:

$$\lambda_i, \lambda_i^* = -\xi_i \omega_i \pm j \sqrt{1 - \xi_i^2} \omega_i \quad (3.23)$$

3.3 ESPAÇO DE ESTADO

No item anterior definiu-se o vetor de estado $\vec{x}(t)$, que contém os deslocamentos e velocidades dos n graus de liberdade. A equação 3.19 é, portanto, uma equação de estado, mas não está na forma usualmente empregada nos métodos de identificação estocástica.

A representação do sistema em espaço de estado pode ser feita em tempo contínuo ou tempo discreto. Até aqui todas equações foram expressas em tempo contínuo. Restringe-se ao estudo de modelos em espaço de estado em tempo discreto devida à natureza discreta dos dados experimentais e até mesmo simulações numéricas. Uma descrição detalhada de modelos em tempo contínuo e sua equivalência com modelos em tempo discreto pode ser encontrada em Peeters (2000) e Rodrigues (2004).

Os modelos estocásticos em espaço de estado em tempo discreto, lineares e invariantes no tempo são descritos matematicamente pelas seguintes equações:

$$\vec{x}_{k+1} = \mathbf{A}\vec{x}_k + \mathbf{B}\vec{v}_k + \vec{w}_k \quad (3.24)$$

$$\vec{y}_k = \mathbf{C}\vec{x}_k + \mathbf{D}\vec{v}_k + \vec{z}_k \quad (3.25)$$

Sendo:

$\vec{x}_k = \vec{x}(k\Delta t)$ – vetor de estado no instante k , $\vec{x}_k \in \mathbb{R}^{n_2}$;

\vec{v}_k – vetor de entradas (excitações) no instante k , $\vec{v}_k \in \mathbb{R}^m$;

\vec{y}_k – vetor de saídas (respostas) no instante k , $\vec{y}_k \in \mathbb{R}^l$;

\vec{w}_k – vetor de perturbações e imprecisões do modelo no instante k , $\vec{w}_k \in \mathbb{R}^{n_2}$;

- \vec{z}_k – vetor de ruídos na medição no instante k , $\vec{z}_k \in \mathbb{R}^l$;
 $k\Delta t$ – instante de tempo, $k \in \mathbb{N}$;
 Δt – período de amostragem, $\Delta t \in \mathbb{R}$;
 \mathbf{A} – matriz dinâmica do sistema, $\mathbf{A} \in \mathbb{R}^{n_2 \times n_2}$;
 \mathbf{B} – matriz de entradas, $\mathbf{B} \in \mathbb{R}^{n_2 \times m}$;
 \mathbf{C} – matriz de influência na saída, $\mathbf{C} \in \mathbb{R}^{l \times n_2}$;
 \mathbf{D} – matriz de transmissão direta, $\mathbf{D} \in \mathbb{R}^{l \times m}$;
 l – número de saídas, $l \in \mathbb{N}$;
 m – número de entradas, $m \in \mathbb{N}$;
 n_2 – número de ordem no espaço de estado, número de estados, $n_2 \in \mathbb{N}$;

A primeira equação é chamada de equação de estado e modela o comportamento dinâmico do sistema em tempo discreto. A segunda equação é chamada de equação de observação, pois controla qual parte das saídas é observada.

O vetor \vec{w}_k considera os ruídos e imprecisões que fazem com que o modelo linear invariante no tempo seja inadequado, como pequenas não-linearidades no sistema real. O vetor \vec{z}_k considera os ruídos presentes nas medições. Ambos não são mensuráveis, podem ser modelados como ruído branco com média zero e matriz de covariância:

$$E \left[\begin{pmatrix} \vec{w}_p \\ \vec{z}_p \end{pmatrix} \begin{pmatrix} \vec{w}_p^T & \vec{z}_p^T \end{pmatrix} \right] = \begin{pmatrix} \mathbf{Q}_r & \mathbf{S}_r \\ \mathbf{S}_r^T & \mathbf{N}_r \end{pmatrix} \delta_{pq} \quad (3.26)$$

Sendo:

- $E[.]$ – valor esperado;
 δ_{pq} – delta de Kronecker;
 p, q – dois instantes arbitrários de tempo;
 $\mathbf{Q}_r, \mathbf{R}_r, \mathbf{N}_r$ – matrizes de covariância do ruído.

É uma hipótese básica tratar a excitação como um processo gaussiano estacionário do tipo ruído branco com média nula (KOZIN; NATKE, 1986; IBRAHIM et al., 1996). Segundo Rodrigues (2004), caso esta hipótese não for cumprida, ou seja, se as forças de excitação contiverem componentes com frequências dominantes, nestas frequências surgirão polos da matriz de estado do sistema identificado ou picos nas funções de densidade espectral.

A representação em espaço de estado não é única. De fato, existem infinitas representações possíveis, podendo-se representar um mesmo sistema por inúmeros vetores de estado, perdendo assim seu significado físico.

A identificação de subespaços de sistemas puramente estocásticos consiste em modelar as matrizes do sistema em função apenas das saídas. Neste caso, não é possível distinguir o vetor de entradas \vec{v}_k dos vetores de ruídos \vec{w}_k e \vec{z}_k . Logo, o vetor de entradas é modelado implicitamente pelos vetores de ruído. Esse problema pode ser escrito matematicamente fazendo-se $\vec{v}_k = 0$ nas equações 3.24 e 3.25, obtendo-se:

$$\vec{x}_{k+1} = \mathbf{A}\vec{x}_k + \vec{w}_k \quad (3.27)$$

$$\vec{y}_k = \mathbf{C}\vec{x}_k + \vec{z}_k \quad (3.28)$$

3.3.1 Propriedades dos sistemas estocásticos

Algumas propriedades dos sistemas estocásticos são apresentadas, as quais são detalhadas em Overschee e Moor (1996). Admite-se que o processo estocástico seja estacionário com média zero:

$$E[\vec{x}_k] = 0 \quad (3.29)$$

$$E[\vec{x}_k \vec{x}_k^T] = \mathbf{\Sigma} \quad (3.30)$$

Sendo $\mathbf{\Sigma}$ a matriz de covariâncias dos estados, a qual independe do tempo k . Como \vec{w}_k e \vec{z}_k possuem média zero e são independentes do estado atual, sabe-se que:

$$E[\vec{x}_k \vec{z}_k^T] = 0 \quad (3.31)$$

$$E[\vec{x}_k \vec{w}_k^T] = 0 \quad (3.32)$$

A matriz de covariâncias das saídas para um determinado atraso de tempo i é definida:

$$\mathbf{R}_i = E[\vec{y}_{k+i} \vec{y}_k^T] \quad (3.33)$$

Define-se a matriz de covariâncias do estado no instante $k + 1$ e da saída no instante k (*next state-output*):

$$\mathbf{G} = E[\vec{x}_{k+1} \vec{y}_k^T] \quad (3.34)$$

Considerando estacionariedade, as propriedades de \vec{w}_k e \vec{z}_k e as definições anteriores, torna-se possível deduzir as seguintes propriedades (OVERSCHEE; MOOR, 1996):

$$\mathbf{\Sigma} = \mathbf{A}\mathbf{\Sigma}\mathbf{A}^T + \mathbf{Q}_r \quad (3.35)$$

$$\mathbf{R}_0 = \mathbf{C}\mathbf{\Sigma}\mathbf{C}^T + \mathbf{N}_r \quad (3.36)$$

$$\mathbf{G} = \mathbf{A}\mathbf{\Sigma}\mathbf{C}^T + \mathbf{S}_r \quad (3.37)$$

E para $i = 1, 2, \dots$:

$$\mathbf{R}_i = \mathbf{C}\mathbf{A}^{i-1}\mathbf{G} \quad (3.38)$$

$$\mathbf{R}_{-i} = \mathbf{G}^T (\mathbf{A}^{i-1})^T \mathbf{C}^T \quad (3.39)$$

As últimas duas equações são importantes e possuem papel fundamental na identificação estocástica de sistemas. A matriz de covariâncias das saídas pode ser estimada a partir dos dados experimentais. Uma vez conhecida, ela pode ser decomposta conforme as equações 3.38 e 3.39 e com isso as matrizes do sistema são encontradas.

3.3.2 Análise modal

Os parâmetros modais da representação em espaço de estado em tempo discreto são obtidos a partir das matrizes \mathbf{A} e \mathbf{C} . Inicia-se com a análise dos autovalores e autovetores da matriz \mathbf{A} , decompondo-a na forma:

$$\mathbf{A} = \mathbf{\Psi} \mathbf{\Lambda}_D \mathbf{\Psi}^{-1} = \mathbf{\Psi} \left[\mu_i \right] \mathbf{\Psi}^{-1} \quad (3.40)$$

Onde $\mathbf{\Psi}$ e $\mathbf{\Lambda}_D$ contém os autovetores e autovalores de \mathbf{A} , respectivamente. Os autovetores são iguais ao do caso contínuo (3.22), por isso o emprego do mesmo símbolo, e os autovalores em tempo discreto μ_i estão relacionados com os autovalores em tempo contínuo λ_i (PEETERS, 2000):

$$\mu_i = e^{\lambda_i \Delta t} \quad \Leftrightarrow \quad \lambda_i = \frac{\ln \mu_i}{\Delta t} \quad (3.41)$$

Sendo Δt o período de amostragem.

As formas modais observáveis \mathbf{V} são obtidas pela equação 3.42 (PEETERS, 2000):

$$\mathbf{V} = \mathbf{C} \mathbf{\Psi} \quad (3.42)$$

3.4 MODELOS ARMA

Um modelo autorregressivo com média móvel pode ser escrito na forma:

$$\vec{y}_k + \alpha_1 \vec{y}_{k-1} + \dots + \alpha_{n_\alpha} \vec{y}_{k-n_\alpha} = \vec{e}_k + \gamma_1 \vec{e}_{k-1} + \dots + \gamma_{n_\gamma} \vec{e}_{k-n_\gamma} \quad (3.43)$$

Sendo \vec{y}_k o vetor das saídas e \vec{e}_k um vetor ruído branco. Os termos ao lado esquerdo da igualdade são a parte autorregressiva (AR) e os termos ao lado direito são a parte com média móvel (MA). As matrizes $\alpha_i \in \mathbb{R}^{l \times l}$ contém os parâmetros da parte AR e as matrizes $\gamma_i \in \mathbb{R}^{l \times l}$

contém os parâmetros da parte MA. No caso de múltiplas saídas, esse modelo também é chamado de ARMAV. Os escalares n_α e n_γ são as ordens da parte AR e MA, respectivamente.

Os modelos ARMA e os modelos no espaço de estado são representações equivalentes de sistemas estocásticos em tempo discreto. A principal diferença é que no espaço de estado emprega-se uma representação interna do sistema, “condição” ou “estado” do sistema, enquanto os modelos ARMA utilizam representações externas de entrada-resposta e o vetor de estado não surge explicitamente. As relações entre essas representações são apresentadas em Peeters (2000) e Rodrigues (2004), por exemplo.

Andersen (1997) e Peeters (2000) mostram que um modelo ARMA obtido a partir do modelo no espaço de estado possui coeficientes AR e MA de mesma ordem: $n_\alpha = n_\gamma = p$. Dado um vetor de estado com dimensão n_2 e sabendo que l saídas são medidas, a ordem do modelo ARMA é calculada:

$$p = \frac{n_2}{l} \quad (3.44)$$

A qual indica que n_2 deve ser um múltiplo de l . Qualquer sistema no espaço de estado pode ser representado pelo modelo ARMA desde que a ordem em espaço de estado n_2 dividida pelo número de saídas l seja igual a um número inteiro p (ANDERSEN, 1997).

Deseja-se determinar as características dinâmicas do sistema representado pelo modelo ARMA. Para isso, apenas os coeficientes da parte AR são necessários. O problema pode ser escrito na forma (PEETERS, 2000):

$$\mathbf{V}\Lambda_D^k + \alpha_1 \mathbf{V}\Lambda_D^{k-1} + \dots + \alpha_p \mathbf{V}\Lambda_D^{k-p} = 0 \quad (3.45)$$

Este problema de autovalores e autovetores de ordem p pode ser reduzido para o problema tradicional de primeira ordem com a definição da matriz de estado na forma companheira:

$$\begin{bmatrix} 0 & \mathbf{I} & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{I} \\ -\boldsymbol{\alpha}_p & -\boldsymbol{\alpha}_{p-1} & \cdots & -\boldsymbol{\alpha}_1 \end{bmatrix} \begin{bmatrix} \mathbf{V} \\ \mathbf{V}\boldsymbol{\Lambda}_D \\ \vdots \\ \mathbf{V}\boldsymbol{\Lambda}_D^{p-2} \\ \mathbf{V}\boldsymbol{\Lambda}_D^{p-1} \end{bmatrix} = \begin{bmatrix} \mathbf{V} \\ \mathbf{V}\boldsymbol{\Lambda}_D \\ \vdots \\ \mathbf{V}\boldsymbol{\Lambda}_D^{p-2} \\ \mathbf{V}\boldsymbol{\Lambda}_D^{p-1} \end{bmatrix} \boldsymbol{\Lambda}_D \quad (3.46)$$

De forma compacta:

$$\mathbf{A}_p^{\text{comp}} \mathbf{O}_{p,m} = \mathbf{O}_{p,m} \boldsymbol{\Lambda}_D \quad (3.47)$$

Sendo $\mathbf{A}_p^{\text{comp}} \in \mathbb{R}^{pl \times pl}$ a matriz de estado na forma companheira contendo os p AR coeficientes na parte inferior e $\mathbf{O}_{p,m} \in \mathbb{C}^{pl \times pl}$ a matriz de observabilidade modal.

Os autovalores discretos μ_i são os autovalores da matriz $\mathbf{A}_p^{\text{comp}}$. As formas modais observáveis \mathbf{V} são obtidas a partir das primeiras l linhas dos autovetores da matriz companheira, em notação Python:

$$\mathbf{V} = \mathbf{O}_{p,m}[:, :l, :] \quad (3.48)$$

Nota-se que apenas a parte autorregressiva do modelo é utilizada para determinar as características dinâmicas do sistema. Salienta-se que um modelo AR de ordem p não é equivalente a um modelo em espaço de estado com ordem $n_2 = pl$. A parte com média móvel é necessária para garantir que as funções de covariâncias das respostas previstas pelo modelo ARMA seja equivalente às funções de covariâncias das respostas do sistema. Teoricamente um modelo AR de ordem infinita é equivalente a um modelo ARMA de ordem finita (ANDERSEN, 1997; PEETERS, 2000).

4 IDENTIFICAÇÃO ESTOCÁSTICA DE SISTEMAS

Neste capítulo apresentam-se alguns métodos de identificação estocástica de sistemas. Como apontado anteriormente, nas medições de vibrações ambientais não há controle das forças (entradas) nem é possível medi-las de uma forma determinística. Assim assume-se a hipótese de que as excitações são realizações de um processo estocástico gaussiano do tipo ruído branco com média zero. O termo identificação **estocástica** de sistemas refere-se a essa hipótese.

Os métodos descritos detalhadamente são:

- a) o Método Básico no Domínio da Frequência (BFD);
- b) o Método Aperfeiçoado de Decomposição no Domínio da Frequência (EFDD);
- c) o Método das Variáveis Instrumentais (IV);
- d) o Método de Identificação Estocástica de Subespaços baseada nas Covariâncias (SSI COV);
- e) o Método de Identificação Estocástica de Subespaços baseada nas Séries Temporais (SSI DATA).

A descrição da formulação dos métodos não segue exatamente a forma original com que foram propostos, sendo adotada uma formulação unificada bastante próxima da apresentada por Peeters (2000) e Rodrigues (2004).

O processo começa adotando-se um certo modelo que se acredita representar a realidade. A partir de dados experimentais, parâmetros são alterados no modelo de modo a calibrá-lo com as medições. Os métodos de identificação podem ser divididos conforme o tipo de dados empregados: séries temporais, covariâncias ou densidades espectrais.

4.1 DADOS EXPERIMENTAIS

Os dados das saídas \vec{y}_k estão disponíveis em intervalos de tempo discreto e constituem a informação de base comum a todos os métodos de identificação. Neste item apresentam-se os cálculos das covariâncias, das densidades espectrais e a notação empregada no capítulo.

4.1.1 Séries temporais

As l saídas medidas são divididas em um conjunto de r sinais ou sensores de referência escolhidos adequadamente. O outro conjunto de resposta é composto de $(l - r)$ sensores. Rearranjando os dados de modo a posicionar as saídas de referência nas primeiras linhas:

$$\vec{y}_k = \begin{bmatrix} \vec{y}_k^{ref} \\ \vec{y}_k^{\sim} \end{bmatrix} \quad (4.1)$$

$$\vec{y}_k^{ref} = \mathbf{L}\vec{y}_k \quad (4.2)$$

$$\mathbf{L} = [\mathbf{I}_r \quad \mathbf{0}] \quad (4.3)$$

Sendo:

\vec{y}_k^{ref} – saídas de referências, $\vec{y}_k^{ref} \in \mathbb{R}^r$;

\vec{y}_k^{\sim} – demais saídas, $\vec{y}_k^{\sim} \in \mathbb{R}^{l-r}$;

\mathbf{L} – matriz de seleção das referências, $\vec{y}_k^{ref} \in \mathbb{R}^{r \times l}$.

Algumas saídas são classificadas como saídas de referência: aquelas que contêm informação modal relevante ou capturam uma resposta estrutural específica. A escolha dos sensores de referência corresponde a escolha das localizações das excitações em uma análise modal tradicional. Uma escolha apropriada pode ajudar na separação de modos próximos e na extração de modos com baixa amplitude (JAMES et al., 1995). O emprego de sensores de referência foi introduzido por Peeters (2000) e permite reduzir as matrizes do sistema, levando a algoritmos mais rápidos e com menor custo computacional. Caso o tempo de processamento seja de pouca importância, pode-se empregar todos os sensores como sensores de referência.

Agrupam-se as saídas em uma matriz bloco Hankel $\mathbf{H}^{ref} \in \mathbb{R}^{(r+l)\iota \times N}$ (matriz cujos elementos da antidiagonal são iguais) composta por 2ι linhas blocos e N colunas. As primeiras ι linhas blocos possuem r linhas cada e as últimas ι linhas bloco possuem l linhas. Essa matriz pode ser dividida em uma parte de saídas chamadas passadas (subscrito p) e uma parte de saídas futuras (subscrito f), conforme equação 4.4:

$$\mathbf{H}^{ref} = \frac{1}{\sqrt{N}} \begin{bmatrix} \vec{y}_0^{ref} & \vec{y}_1^{ref} & \cdots & \vec{y}_{N-1}^{ref} \\ \vec{y}_1^{ref} & \vec{y}_2^{ref} & \cdots & \vec{y}_N^{ref} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{y}_{l-1}^{ref} & \vec{y}_l^{ref} & \cdots & \vec{y}_{l+N-2}^{ref} \\ \vec{y}_l & \vec{y}_{l+1} & \cdots & \vec{y}_{l+N-1} \\ \vec{y}_{l+1} & \vec{y}_{l+2} & \cdots & \vec{y}_{l+N} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{y}_{2l-1} & \vec{y}_{2l} & \cdots & \vec{y}_{2l+N-1} \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_{0|l-1}^{ref} \\ \mathbf{Y}_{l|2l-1} \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_p^{ref} \\ \mathbf{Y}_f \end{bmatrix} \begin{matrix} \updownarrow r l & \text{"passado"} \\ \updownarrow l l & \text{"futuro"} \end{matrix} \quad (4.4)$$

Sendo l um índice a definir na análise, o qual está associado com o número de atrasos no tempo em que as funções de covariância são estimadas e N o número de colunas da matriz, considerado igual a $s - 2l + 1$ visando aproveitar todas as s amostras das respostas no tempo. Do ponto de vista estatístico, os métodos assumem que $N, s \rightarrow \infty$.

A introdução do escalar $1/\sqrt{N}$ em 4.4 permite que as funções de covariância sejam estimadas a partir da matriz Hankel, como se apresentará na próxima seção.

Nota-se que o subscrito de $\mathbf{Y}_{l|2l-1}$ são os subscritos do primeiro e do último elemento da primeira coluna da matriz. Por conveniência, definem-se as matrizes \mathbf{Y}_p^+ e \mathbf{Y}_f^- , nas quais o sobrescrito $+$ denota aumento de uma linha bloco e o sobrescrito $-$ diminuição de uma linha bloco:

$$\mathbf{H}^{ref} = \begin{bmatrix} \mathbf{Y}_{0|l}^{ref} \\ \mathbf{Y}_{l|l}^{\sim ref} \\ \mathbf{Y}_{l+1|2l-1} \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_p^{ref+} \\ \mathbf{Y}_{l|l}^{\sim ref} \\ \mathbf{Y}_f^- \end{bmatrix} \begin{matrix} \updownarrow r(l+1) \\ \updownarrow l-r \\ \updownarrow l(l-1) \end{matrix} \quad (4.5)$$

4.1.2 Estimativas das funções de covariâncias

A matriz de covariância das saídas $\mathbf{R}_i \in \mathbb{R}^{l \times l}$, assumindo ergodicidade, é:

$$\mathbf{R}_i = E[\vec{y}_{k+i}\vec{y}_k^T] = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} \vec{y}_{k+i}\vec{y}_k^T \quad (4.6)$$

A partir das saídas de referência, define-se a matriz de covariâncias reduzida $\mathbf{R}_i^{ref} \in \mathbb{R}^{l \times r}$ como as primeiras r colunas da matriz completa:

$$\mathbf{R}_i^{ref} = E[\vec{y}_{k+i}(\vec{y}_k^{ref})^T] = \mathbf{R}_i \mathbf{L}^T \quad (4.7)$$

De forma semelhante, a matriz de covariâncias reduzida do estado no tempo $k + 1$ e da saída no tempo k (*next state-output*) $\mathbf{G}^{ref} \in \mathbb{R}^{n_2 \times r}$ é definida:

$$\mathbf{G}^{ref} = E[\vec{x}_{k+1}(\vec{y}_k^{ref})^T] = \mathbf{G} \mathbf{L}^T \quad (4.8)$$

E para $i = 1, 2, \dots$:

$$\mathbf{R}_i^{ref} = \mathbf{R}_i \mathbf{L}^T = \mathbf{C} \mathbf{A}^{i-1} \mathbf{G}^{ref} \quad (4.9)$$

$$(\mathbf{R}_i^{ref})^T = \mathbf{L} \mathbf{R}_{-i} = (\mathbf{G}^{ref})^T (\mathbf{A}^{i-1})^T \mathbf{C}^T \quad (4.10)$$

as quais são equações equivalentes as propriedades 3.38 e 3.39 obtidas a partir da matriz de covariância completa. As covariâncias das saídas para até $2l - 1$ atrasos no tempo são agrupadas em uma matriz Toeplitz $\mathbf{T}_{1|l}^{ref} \in \mathbb{R}^{l \times r l}$ calculada com os dados da matriz Hankel:

$$\mathbf{T}_{1|l}^{ref} = \mathbf{Y}_f (\mathbf{Y}_p^{ref})^T = \begin{bmatrix} \mathbf{R}_l^{ref} & \mathbf{R}_{l-1}^{ref} & \cdots & \mathbf{R}_2^{ref} & \mathbf{R}_1^{ref} \\ \mathbf{R}_{l+1}^{ref} & \mathbf{R}_l^{ref} & \cdots & \mathbf{R}_3^{ref} & \mathbf{R}_2^{ref} \\ \mathbf{R}_{l+2}^{ref} & \mathbf{R}_{l+1}^{ref} & \cdots & \mathbf{R}_4^{ref} & \mathbf{R}_3^{ref} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{R}_{2l-1}^{ref} & \mathbf{R}_{2l-2}^{ref} & \cdots & \mathbf{R}_{l+1}^{ref} & \mathbf{R}_l^{ref} \end{bmatrix} \quad (4.11)$$

Como o número N de dados é finito, o estimador das covariâncias $\hat{\mathbf{R}}_i$ é obtido removendo o limite de 4.6:

$$\hat{\mathbf{R}}_i = E[\vec{y}_{k+i}\vec{y}_k^T] = \frac{1}{N} \sum_{k=0}^{N-1} \vec{y}_{k+i}\vec{y}_k^T \quad (4.12)$$

4.1.3 Estimativas das densidades espectrais

A matriz das funções de densidades espectrais das saídas $\mathbf{S}_y \in \mathbb{C}^{l \times l}$ é definida como a transformada discreta de Fourier das funções de covariâncias:

$$\mathbf{S}_y(\omega) = \sum_{k=-\infty}^{\infty} \mathbf{R}_k e^{-j\omega k \Delta t} \quad (4.13)$$

Como o número de dados é finito, as funções de covariâncias são estimadas (4.12) e não podem ser calculadas para atrasos de tempo infinitos. Uma forma de se estimar as densidades espectrais é por meio do periodograma de Welch (1967).

Inicia-se calculando a transformada discreta de Fourier das saídas \vec{Y} ponderadas por uma janela de dados w_k (BENDAT; PIERSOL, 2010, p. 368):

$$\vec{Y}(\omega_i) = \Delta t \sum_{k=0}^{N-1} w_k \vec{y}_k e^{-j\omega_i k \Delta t} \quad (4.14)$$

Sendo:

N – número de amostras no tempo;

Δt – período de amostragem;

ω – frequência.

Para o emprego da transformada rápida de Fourier, N deve ser potência de dois. Como consequência, a transformada discreta de Fourier é calculada nas frequências:

$$\omega_i = \frac{i}{N} \frac{2\pi}{\Delta t} \quad i = 0, 1, \dots, N-1 \quad (4.15)$$

Uma estimativa da matriz de funções de densidades espectrais das saídas (ou matriz espectral) é calculada:

$$\widehat{\mathbf{S}}_y(\omega_i) = \frac{1}{N \Delta t \sum_{k=0}^{N-1} |w_k|^2} \vec{Y}^*(\omega_i) \vec{Y}^T(\omega_i) \quad i = 0, 1, 2, \dots, N-1 \quad (4.16)$$

Os elementos da diagonal principal da matriz são designados por autoespectros e os demais são designados por espectros cruzados.

As densidades espectrais calculadas por 4.16 são definidas também para frequências negativas (*two-sided spectral density functions*), nas quais seus valores são simétricos aos das positivas. Deste modo, verifica-se que apenas os primeiros $N/2 + 1$ valores são únicos com a frequência de Nyquist ocorrendo para $i = N/2$. Torna-se conveniente definir as seguintes funções de densidade espectral (*one-sided spectral density functions*):

$$\widehat{\mathbf{G}}_y(\omega_i) = \begin{cases} \frac{2}{N \Delta t \sum_{k=0}^{N-1} |w_k|^2} \vec{Y}^*(\omega_i) \vec{Y}^T(\omega_i), & i = 1, 2, \dots, N/2 - 1 \\ \frac{1}{N \Delta t \sum_{k=0}^{N-1} |w_k|^2} \vec{Y}^*(\omega_i) \vec{Y}^T(\omega_i), & i = 0, N/2 \end{cases} \quad (4.17)$$

A equação anterior fornece uma estimativa com elevada variância, a qual pode ser reduzida dividindo-se o sinal em segmentos, calculando-se o periodograma de cada segmento e, em seguida, a média deles. A definição da duração de cada segmento é uma troca entre resolução em frequência e variância. Selecionando-se segmentos mais longos, aumenta-se a resolução mas também aumenta-se a variância da estimativa.

Na subdivisão dos registros, é comum considerar uma sobreposição entre eles visando aumentar o aproveitamento de toda a informação contida nos registros, visto que a janela de dados diminui consideravelmente a amplitude nas extremidades das amostras.

4.2 MÉTODOS NO DOMÍNIO DA FREQUÊNCIA

Os métodos no domínio da frequência estimam as propriedades dinâmicas a partir das densidades espectrais das respostas. Nos próximos itens são apresentados o Método Básico no Domínio da Frequência e o Método Aperfeiçoado de Decomposição no Domínio da Frequência.

4.2.1 Método Básico no Domínio da Frequência (BFD)

O Método Básico no Domínio da Frequência (BFD) ou Método de Detecção de Picos (PP) é o método mais simples de ser empregado, no qual as frequências naturais são identificadas a partir dos picos das densidades espectrais da resposta. Assume-se que toda a resposta possa ser atribuída ao modo em estudo e que qualquer efeito dos demais modos possa ser ignorado. Este método fornece estimativas adequadas para estruturas que apresentam frequências naturais significativamente afastadas, que sejam pouco amortecidas e estejam sujeitas a forças com características de ruído branco.

Rodrigues (2004) aponta que os fundamentos do método foram apresentados por diversos autores, como Bendat e Piersol (1993). No entanto se deve a Felber (1993), o desenvolvimento do método de uma forma mais sistematizada permitindo sua automatização. A apresentação que se faz em seguida é semelhante à de Felber (1993) e Rodrigues (2004).

As frequências naturais são obtidas a partir dos picos das densidades espectrais das respostas. Porém, este processo pode envolver a visualização de inúmeros gráficos. Felber (1993) propôs uma forma de sintetizar as informações das densidades espectrais em um único gráfico. A função proposta é chamada de Densidade Espectral Normalizada Média (*Averaged Normalized Power Spectral Density* – ANPSD).

Primeiramente calculam-se as Densidades Espectrais Normalizadas (*Normalized Power Spectral Density* – NPSD), tornando as áreas dos autoespectros unitárias:

$$\text{NPSD}_i(\omega_m) = \frac{\widehat{\mathbf{G}}_y[i, i](\omega_m)}{\sum_{m=1}^{N/2} \widehat{\mathbf{G}}_y[i, i](\omega_m)} \quad (4.18)$$

Em seguida calcula-se a média para obter a ANPSD:

$$\text{ANPSD}(\omega_m) = \frac{1}{l} \sum_{i=1}^l \text{NPSD}_i(\omega_m) \quad (4.19)$$

Os picos da ANPSD são utilizados para a determinação das frequências naturais. Como resulta de uma média de diferentes espectros, a ANSPD evidencia picos comuns a todos os autoespectros e suaviza aqueles presentes em apenas um. Deste modo, a ANPSD facilita a identificação das frequências correspondentes aos modos globais de vibração.

Para facilitar a análise e a distinção entre os modos de vibração, é recomendável calcular a ANPSD dos deslocamentos lineares e angulares separadamente. Também é possível efetuar algumas operações simples com os registros, como semissoma e semidiferença, visando obter séries temporais que salientem modos de vibração de determinado tipo.

Um aspecto importante é verificar se de fato o pico das densidades espectrais corresponde a um modo de vibração estrutural. A função de coerência pode ser empregada para efetuar essa confirmação. Considerando dois sinais y_i e y_j , sua estimativa é dada por:

$$\gamma_{i,j}^2(\omega_m) = \frac{|\widehat{\mathbf{G}}_y[i,j](\omega_m)|^2}{\widehat{\mathbf{G}}_y[i,i](\omega_m)\widehat{\mathbf{G}}_y[j,j](\omega_m)} \quad (4.20)$$

Ela é uma função escalar que assume valores entre 0 e 1, traduzindo, no domínio da frequência, o grau de linearidade da relação entre o sinal y_i e o sinal y_j . Valores próximos de 1 mostram que há uma forte relação de linearidade entre os dois sinais. A função de coerência tem um significado análogo ao quadrado do coeficiente de correlação utilizado em estatística (BENDAT; PIERSOL, 2010).

Nas frequências de modos globais de vibração, a função de coerência entre os sinais da resposta observada em diversos pontos deve apresentar valores próximos de 1. Por outro lado, ela pode adquirir valores baixos nas frequências de modos pouco excitados ou naqueles sinais localizados próximos a um nó da forma modal.

Em estruturas com frequências naturais bem separadas e amortecimento pequeno, sujeitas a forças com características de ruído branco, a resposta na frequência de um dos seus modos de vibração é dominada por esse modo. Peeters (2000) mostra que na frequência de ressonância

as colunas (ou as linhas) da matriz de densidades espectrais podem ser utilizadas para estimar a forma modal.

Define-se uma saída de referência r , na qual a componente da forma modal adquire valor unitário. Dividindo-se as amplitudes das densidades espectrais pela densidade espectral da saída de referência, obtém-se a estimativa da forma modal:

$$\vec{\varphi}_i = \begin{bmatrix} \varphi_{i,1} \\ \vdots \\ \varphi_{i,j} \\ \vdots \\ \varphi_{i,r} \\ \vdots \\ \varphi_{i,n} \end{bmatrix} \approx \begin{bmatrix} \widehat{\mathbf{G}}_y[k, 1](\omega_i) / \widehat{\mathbf{G}}_y[r, r](\omega_i) \\ \vdots \\ \widehat{\mathbf{G}}_y[k, j](\omega_i) / \widehat{\mathbf{G}}_y[r, r](\omega_i) \\ \vdots \\ 1 \\ \vdots \\ \widehat{\mathbf{G}}_y[k, n](\omega_i) / \widehat{\mathbf{G}}_y[r, r](\omega_i) \end{bmatrix} \quad (4.21)$$

O sinal da componente modal é determinado pelo ângulo de fase do espectro cruzado. Quando o ângulo for próximo a 0° , as saídas encontram-se em fase e a componente assume valor positivo. Por outro lado, se o ângulo for próximo a $\pm 180^\circ$, as saídas encontram-se perfeitamente fora de fase e a componente é negativa (FELBER, 1993).

Caso um bom sensor de referência seja escolhido, apenas as densidades espectrais entre todos os sensores e o de referência precisam ser estimadas, não havendo a necessidade de calcular a matriz espectral completa. Teoricamente, uma coluna (ou linha) da matriz espectral é suficiente para obter a estimativa das formas modais (PEETERS, 2000).

A determinação da razão de amortecimento pode ser feita a partir do autoespectro da resposta. Apresentam-se duas formas de obtê-la: o método da meia potência e o método de ajuste de um espectro analítico.

Bendat e Piersol (1993) apresentam as condições para que seja possível obter estimativas razoáveis dos coeficientes de amortecimento:

- a) a densidade espectral da excitação deve ser razoavelmente constante no intervalo $f_i - 3B_i \leq f \leq f_i + 3B_i$, sendo B_i a largura da meia potência do modo de vibração com frequência f_i ;
- b) o coeficiente de amortecimento deve ser pequeno, $\xi_i < 5\%$;
- c) a resolução em frequência Δf deve ser muito menor do que a largura da meia potência, $\Delta f < 0,2B_i$;

- d) os modos de vibração devem ter frequências bem separadas, $f_i - f_{i-1} > 2(B_i + B_{i-1})$.

Se as condições forem cumpridas, a razão de amortecimento é estimada a partir da frequência do pico f_r e das frequências com ordenada espectral igual a metade da ordenada máxima f_a e f_b (EWINS, 2000):

$$\xi_i = \frac{f_b^2 - f_a^2}{4f_r^2} \quad (4.22)$$

Dado o caráter discreto dos autoespectros, torna-se necessário interpolar entre os pontos para a determinação das frequências f_a e f_b . Diversos autores afirmam que as razões de amortecimento obtidas por este método geralmente são superestimadas (PEETERS, 2000; RODRIGUES, 2004; BENDAT; PIERSOL, 1993).

Outra forma de estimar a razão de amortecimento é ajustando um espectro analítico ao autoespectro da resposta. A expressão 4.23 corresponde à densidade espectral da resposta em aceleração de um sistema de um grau de liberdade sujeito a uma excitação com densidade espectral constante e igual a Π_u .

$$S_y(\omega) = \frac{\Pi_u}{k^2} \left| \frac{\omega^2}{1 - (\omega/\omega_n)^2 + 2j\xi(\omega/\omega_n)} \right|^2 + C \quad (4.23)$$

Sendo:

k – rigidez do sistema de um grau liberdade;

ω_n – frequência natural;

ξ – razão de amortecimento;

C – constante que considera a influência dos demais modos, fora do pico de ressonância;

O ajuste pode ser feito pelo método de mínimos quadrados para funções não-lineares. No Python, a função `scipy.optimize.curve_fit` emprega um Método de Região de Confiança (*Trust Region Reflective*).

A função programada recebeu o nome de BFD e encontra-se no Apêndice A.

4.2.2 Método Aperfeiçoado de Decomposição no Domínio da Frequência

O Método Aperfeiçoado de Decomposição no Domínio da Frequência (*Enhanced Frequency Domain Decomposition* – EFDD) é uma extensão do BFD. Ele foi inicialmente fundamentado por Brincker et al. (2000) recebendo o nome de Método de Decomposição no Domínio da Frequência (*Frequency Domain Decomposition* – FDD) e aprofundado por Brincker et al. (2001b) sendo designado EFDD, nomenclatura adotada no presente trabalho.

A matriz espectral, em uma certa frequência, é determinada apenas por alguns modos. A quantidade de modos que contribui de forma significativa pode ser estimada pelo posto da matriz, sendo a decomposição em valores singulares utilizada para sua estimativa. Portanto, o número de valores singulares não nulos é igual ao posto da matriz e a identificação de modos muito próximos pode ser feita analisando-se os valores singulares não nulos da matriz espectral na respectiva frequência. A existência de dois valores não nulos, por exemplo, indica a presença de dois modos (SHIH et al., 1988; BRINCKER et al., 2000; PEETERS, 2000).

O resultado é exato se a excitação for um ruído branco, a estrutura for fracamente amortecida e se as formas modais de modos próximos forem geometricamente ortogonais. Se essas hipóteses não forem satisfeitas, a decomposição em sistemas com um grau de liberdade é aproximada, fornecendo, mesmo assim, resultados mais precisos do que o BFD (BRINCKER et al., 2000).

O primeiro passo é estimar as funções de densidades espectrais da resposta. Em seguida, calcula-se sua decomposição em valores singulares para cada frequência:

$$\widehat{\mathbf{G}}_y(\omega_i) = \mathbf{U}_i \mathbf{S}_i \mathbf{U}_i^H \quad (4.24)$$

Sendo $\mathbf{U}_i \in \mathbb{C}^{l \times l}$ uma matriz unitária complexa contendo os vetores singulares e $\mathbf{S}_i \in \mathbb{R}^{l \times l}$ uma matriz diagonal contendo os valores singulares em ordem decrescente. O superescrito $(.)^H$ denota complexo conjugado transposto.

Na frequência de ressonância, estima-se a forma modal como sendo igual ao primeiro vetor singular (primeira coluna de \mathbf{U}_i). Empregando-se a notação do Python:

$$\vec{\varphi}_i = \mathbf{U}_i[:,0](\omega_i) \quad (4.25)$$

Caso existam múltiplos modos com a mesma frequência natural, cada vetor singular correspondente a um valor singular não nulo é uma estimativa da forma modal.

Uma vez que um pico é aceito como representante modal, deve-se definir um trecho na sua vizinhança. A determinação desse trecho é feita com auxílio do Critério de Concordância Modal (MAC) entre o vetor singular do pico e da sua vizinhança. Conforme será mostrado, o MAC mensura a correlação entre os vetores. Se certa frequência possui um vetor singular com MAC maior do que determinado limite (por exemplo 0,90), ela pode ser considerada como parte daquele sistema. Os trechos nos quais essa condição não é cumprida recebem valores nulos.

Em seguida, transforma-se a função de densidade espectral para o domínio do tempo através da transformada inversa de Fourier, obtendo-se a função de autocorrelação de um sistema com um grau de liberdade. Os valores obtidos são ajustados à resposta de um sistema com um grau de liberdade em vibração livre amortecida para a obtenção da razão de amortecimento e da frequência natural (CLOUGH; PENZIEN, 2003):

$$y(t) = y_p e^{-\eta t} \cos(\omega_d t) \quad (4.26)$$

Sendo:

y_p – amplitude;

$\eta = \omega_n \xi$ – variável auxiliar;

Optou-se por realizar o ajuste em dois passos. No primeiro, ajusta-se o envelope exponencial $y(t) = y_p e^{-\eta t}$ obtendo-se η e y_p . No segundo passo, ajusta-se a resposta em vibração livre amortecida obtendo-se ω_n e ξ .

A implementação do método em Python denomina-se EFDD e é apresentada no Apêndice A.

4.3 MÉTODOS BASEADOS NAS COVARIÂNCIAS

Neste item são apresentados os métodos de identificação que empregam as funções de covariância. Uma propriedade importante da matriz de covariância é que ela pode ser fatorada

em função das matrizes do sistema. Primeiramente apresenta-se o método das Variáveis Instrumentais e em seguida o Método de Identificação Estocástica de Subespaço baseado nas Covariâncias.

4.3.1 Método das Variáveis Instrumentais (IV)

O método das Variáveis Instrumentais emprega as funções de covariâncias para obter a representação ARMA do sistema. Na verdade, apenas os coeficientes da parte AR são determinados e de uma maneira linear (ANDERSEN, 1997; LJUNG, 1999; PEETERS, 2000).

A representação ARMA é:

$$\vec{y}_k + \alpha_1 \vec{y}_{k-1} + \dots + \alpha_p \vec{y}_{k-p} = \vec{e}_k + \gamma_1 \vec{e}_{k-1} + \dots + \gamma_p \vec{e}_{k-p}$$

Uma boa estimativa para os coeficientes extrai o máximo possível de informação dos dados, deixando resíduos não correlacionados com os dados passados. Formalmente:

$$E[\vec{e}_k \vec{y}_{k-i}^T] = E[\vec{e}_k] E[\vec{y}_{k-i}^T] = 0 \quad (4.27)$$

Na primeira igualdade considerou-se que os ruídos e as saídas são não correlacionados. A segunda igualdade deve-se ao fato de o ruído ter média zero. O método IV começa impondo a condição anterior ao modelo ARMA visando eliminar a parte MA. Pós-multiplicando a representação por \vec{y}_{k-p-i}^T e aplicando o operador esperança, obtém-se:

$$E[\vec{y}_k \vec{y}_{k-p-i}^T] + \alpha_1 E[\vec{y}_{k-1} \vec{y}_{k-p-i}^T] + \dots + \alpha_p E[\vec{y}_{k-p} \vec{y}_{k-p-i}^T] = 0 \quad (4.28)$$

Por causa da estacionariedade $E[\vec{y}_k \vec{y}_{k-i}^T] = E[\vec{y}_k \vec{y}_{k+i}^T] = \mathbf{R}_i$ e a equação básica do método das Variáveis Instrumentais é:

$$\mathbf{R}_{p+i} + \alpha_1 \mathbf{R}_{p+i-1} + \dots + \alpha_p \mathbf{R}_i = 0 \quad (4.29)$$

Na realidade as covariâncias das saídas são estimadas ($\widehat{\mathbf{R}}_i$). Escrevendo essa equação para os atrasos de tempo i , os coeficientes AR são estimados por mínimos quadrados.

4.3.1.1 Introdução dos sensores de referência

A formulação anterior é modificada visando introduzir o conceito de sensores de referência e foi proposta por Peeters (2000). Começa-se com o modelo ARMA reverso no tempo (ou para trás) empregando apenas as saídas de referência:

$$\vec{y}_k^{ref} + \alpha_1^b \vec{y}_{k+1}^{ref} + \dots + \alpha_p^b \vec{y}_{k+p}^{ref} = \vec{e}_k^b + \gamma_1^b \vec{e}_{k+1}^b + \dots + \gamma_p^b \vec{e}_{k+p}^b \quad (4.30)$$

O modelo está rodando para trás no tempo: as saídas atuais são escritas em termos das saídas futuras. Como apenas as saídas de referência são empregadas, os coeficientes e os resíduos têm as dimensões $\alpha_1^b \in \mathbb{R}^{r \times r}$, $\gamma_1^b \in \mathbb{R}^{r \times r}$ e $\vec{e}_k^b \in \mathbb{R}^r$.

Se a ordem p do modelo ARMA vezes o número de saídas de referência r for igual ou maior do que a ordem do sistema no espaço de estado n_2 ($pr \geq n_2$), o sistema é sobredeterminado.

Conforme mencionado, um bom modelo leva a resíduos não correlacionados com as saídas futuras $E[\vec{e}_k^b \vec{y}_{k+i}^T] = 0$. Usando a definição dos sensores de referência, a equação do método IV (4.19) é escrita na forma:

$$\mathbf{L} \mathbf{R}_{-p-i} + \alpha_1^b \mathbf{L} \mathbf{R}_{-p-i+1} + \dots + \alpha_p^b \mathbf{L} \mathbf{R}_{-i} = 0 \quad (4.31)$$

Transpondo a equação anterior e escrevendo-a para q atrasos no tempo, o seguinte sistema de equações é obtido:

$$\begin{bmatrix} \mathbf{R}_p^{ref} & \mathbf{R}_{p-1}^{ref} & \dots & \mathbf{R}_1^{ref} \\ \mathbf{R}_{p+1}^{ref} & \mathbf{R}_p^{ref} & \dots & \mathbf{R}_2^{ref} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{p+q-1}^{ref} & \mathbf{R}_{p+q-2}^{ref} & \dots & \mathbf{R}_q^{ref} \end{bmatrix} \begin{bmatrix} (\alpha_1^b)^T \\ (\alpha_2^b)^T \\ \vdots \\ (\alpha_p^b)^T \end{bmatrix} = - \begin{bmatrix} \mathbf{R}_{p+1}^{ref} \\ \mathbf{R}_{p+2}^{ref} \\ \vdots \\ \mathbf{R}_{p+q}^{ref} \end{bmatrix} \quad (4.32)$$

A partir dele determinam-se os coeficientes da parte AR reversa por mínimos quadrados. O número inteiro q está relacionado com a sobredeterminação do sistema: há qlr equações para pr^2 incógnitas.

Pode-se mostrar que um modelo ARMA reverso é equivalente ao chamado modelo de inovação reverso (*backward innovation model*) em espaço de estado e que a relação entre os coeficientes da parte AR e as matrizes no espaço de estado é:

$$(\mathbf{G}^{ref})^T (\mathbf{A}^T)^p + \sum_{i=1}^p \alpha_i^b (\mathbf{G}^{ref})^T (\mathbf{A}^T)^{p-i} = 0 \quad (4.33)$$

Introduzindo a decomposição em autovalores e autovetores de \mathbf{A} , pós-multiplicando a equação 4.33 por Ψ^{-T} e definindo $\mathbf{G}_m^{ref} = \Psi^{-1} \mathbf{G}^{ref}$, essa equação pode ser reescrita na forma:

$$(\mathbf{G}_m^{ref})^T \Lambda_D^p + \alpha_1^b (\mathbf{G}_m^{ref})^T \Lambda_D^{p-1} + \dots + \alpha_p^b (\mathbf{G}_m^{ref})^T = 0 \quad (4.34)$$

A qual revela que os autovalores Λ_D e a matriz estocástica reduzida de participação modal \mathbf{G}_m^{ref} podem ser obtidos a partir da decomposição em autovalores da matriz companheira com os coeficientes AR reverso.

As formas modais \mathbf{V} são determinadas em um segundo passo. A fatoração da matriz das covariâncias das saídas é empregada:

$$\mathbf{R}_i^{ref} = \mathbf{C} \mathbf{A}^{i-1} \mathbf{G}^{ref} = \mathbf{C} \Psi \Lambda_D^{i-1} \Psi^{-1} \mathbf{G}^{ref} = \mathbf{V} \Lambda_D^{i-1} \mathbf{G}_m^{ref} \quad (4.35)$$

Escrevendo essa equação para p atrasos de tempo, o seguinte sistema de equação é obtido:

$$\mathbf{V} \Gamma_{p,m}^{ref} = [\mathbf{R}_p^{ref} \quad \mathbf{R}_{p-1}^{ref} \quad \dots \quad \mathbf{R}_1^{ref}] \quad (4.36)$$

Sendo $\Gamma_{p,m}^{ref}$ a matriz de controlabilidade modal estocástica revertida (*reversed extended modal stochastic controllability*):

$$\Gamma_{p,m}^{ref} = [\Lambda_D^{p-1} \mathbf{G}_m^{ref} \quad \Lambda_D^{p-2} \mathbf{G}_m^{ref} \quad \dots \quad \Lambda_D \mathbf{G}_m^{ref} \quad \mathbf{G}_m^{ref}] \quad (4.37)$$

4.3.1.2 Algoritmo e implementação

Neste item apresenta-se o algoritmo empregado para a identificação do sistema com base no método das Variáveis Instrumentais. Em Python, a função criada recebeu o nome de IV e encontra-se no Apêndice A.

Calculam-se os coeficientes da parte AR reversa resolvendo o sistema de equações por mínimos quadrados:

$$\begin{bmatrix} \mathbf{R}_p^{ref} & \mathbf{R}_{p-1}^{ref} & \cdots & \mathbf{R}_1^{ref} \\ \mathbf{R}_{p+1}^{ref} & \mathbf{R}_p^{ref} & \cdots & \mathbf{R}_2^{ref} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{p+q-1}^{ref} & \mathbf{R}_{p+q-2}^{ref} & \cdots & \mathbf{R}_q^{ref} \end{bmatrix} \begin{bmatrix} (\boldsymbol{\alpha}_1^b)^T \\ (\boldsymbol{\alpha}_2^b)^T \\ \vdots \\ (\boldsymbol{\alpha}_p^b)^T \end{bmatrix} = - \begin{bmatrix} \mathbf{R}_{p+1}^{ref} \\ \mathbf{R}_{p+2}^{ref} \\ \vdots \\ \mathbf{R}_{p+q}^{ref} \end{bmatrix} \quad (4.38)$$

no qual as matrizes do lado esquerdo e direito podem ser obtidas a partir da matriz Toeplitz.

Forma-se a matriz de estado na forma companheira com os coeficientes da parte AR reversa:

$$\mathbf{A}_{\text{comp}}^{b,p} = \begin{bmatrix} 0 & \mathbf{I} & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{I} \\ -\boldsymbol{\alpha}_p^b & -\boldsymbol{\alpha}_{p-1}^b & \cdots & -\boldsymbol{\alpha}_1^b \end{bmatrix} \quad (4.39)$$

As matrizes com os autovalores e autovetores, $\boldsymbol{\Lambda}_D$ e $\boldsymbol{\Psi}_{\text{comp}}$ respectivamente, são determinadas. A partir dos autovalores μ_i calculam-se as frequências naturais e razões de amortecimento (equações 3.12, 3.13 e 3.41).

As formas modais são determinadas em um segundo passo. A matriz \mathbf{G}_m^{ref} é obtida a partir das primeiras r linhas da matriz com os autovetores de $\mathbf{A}_{\text{comp}}^{b,p}$:

$$\mathbf{G}_m^{ref} = (\boldsymbol{\Psi}_{\text{comp}}[:, r, :])^T \quad (4.40)$$

A matriz $\boldsymbol{\Gamma}_{p,m}^{ref}$ é calculada:

$$\mathbf{\Gamma}_{p,m}^{ref} = [\mathbf{\Lambda}_D^{p-1} \mathbf{G}_m^{ref} \quad \mathbf{\Lambda}_D^{p-2} \mathbf{G}_m^{ref} \quad \dots \quad \mathbf{\Lambda}_D \mathbf{G}_m^{ref} \quad \mathbf{G}_m^{ref}] \quad (4.41)$$

E finalmente as formas modais:

$$\mathbf{V} = [\mathbf{R}_p^{ref} \quad \mathbf{R}_{p-1}^{ref} \quad \dots \quad \mathbf{R}_1^{ref}] (\mathbf{\Gamma}_{p,m}^{ref})^{-1} \quad (4.42)$$

Conforme será mostrado ao final deste capítulo, o processo de identificação é realizado para modelos com ordens crescentes e a identificação dos modos de vibração verdadeiros é feita com auxílio do diagrama de estabilização. A função `IV_iterator` faz esse processo.

Nela especifica-se uma ordem máxima p_{\max} do modelo ARMA, a partir da qual a ordem máxima do modelo em espaço de estado $n_{2,\max}$ é obtida fazendo-se $n_{2,\max} = p_{\max} r$ e a matriz Toeplitz é calculada uma única vez. Com base na equação 4.32, o método IV requer uma matriz Toeplitz com $p_{\max} + 1$ colunas bloco e q linhas bloco. Os atrasos arbitrários de tempo podem ser determinados comparando 4.11 e 4.32, resultando $\iota = p_{\max} + 1$ e q pode ser tomado igual a ι , $q = \iota$. Assim, a sobredeterminação do sistema está garantida. Modelos de ordens inferiores ($p < p_{\max}$) são facilmente obtidos simplesmente selecionando-se menos elementos da matriz Toeplitz.

4.3.2 Método de identificação estocástica de subespaços baseados nas covariâncias (SSI COV)

O método de Identificação Estocástica de Subespaços baseados nas Covariâncias (*Covariance-driven stochastic subspace identification*) aborda o problema de realização estocástica, ou seja, o problema de identificar um modelo estocástico em espaço de estado com apenas saídas medidas (PEETERS, 2000). Este método baseia-se na decomposição em valores singulares da matriz de covariâncias das saídas.

Conforme mostrou-se anteriormente, as covariâncias das saídas de referência são reunidas na matriz Toeplitz:

$$\mathbf{T}_{1|l}^{ref} = \mathbf{Y}_f(\mathbf{Y}_p^{ref})^T = \begin{bmatrix} \mathbf{R}_l^{ref} & \mathbf{R}_{l-1}^{ref} & \cdots & \mathbf{R}_2^{ref} & \mathbf{R}_1^{ref} \\ \mathbf{R}_{l+1}^{ref} & \mathbf{R}_l^{ref} & \cdots & \mathbf{R}_3^{ref} & \mathbf{R}_2^{ref} \\ \mathbf{R}_{l+2}^{ref} & \mathbf{R}_{l+1}^{ref} & \cdots & \mathbf{R}_4^{ref} & \mathbf{R}_3^{ref} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{R}_{2l-1}^{ref} & \mathbf{R}_{2l-2}^{ref} & \cdots & \mathbf{R}_{l+1}^{ref} & \mathbf{R}_l^{ref} \end{bmatrix} \quad (4.43)$$

Fatorando a matriz anterior conforme apresentado nas equações 4.9 e 4.10:

$$\mathbf{T}_{1|l}^{ref} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{l-1} \end{bmatrix} \begin{bmatrix} \mathbf{A}^{l-1}\mathbf{G}^{ref} & \cdots & \mathbf{AG}^{ref} & \mathbf{G}^{ref} \end{bmatrix} = \mathbf{O}_l \mathbf{\Gamma}_l^{ref} \quad (4.44)$$

\leftrightarrow
 n_2

Sendo $\mathbf{O}_l \in \mathbb{R}^{l \times n_2}$ a matriz de observabilidade estendida e $\mathbf{\Gamma}_l^{ref} \in \mathbb{R}^{n_2 \times l}$ a matriz controlabilidade estocástica revertida:

$$\mathbf{O}_l = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{l-1} \end{bmatrix} \quad (4.45)$$

$$\mathbf{\Gamma}_l^{ref} = [\mathbf{A}^{l-1}\mathbf{G}^{ref} \quad \cdots \quad \mathbf{AG}^{ref} \quad \mathbf{G}^{ref}] \quad (4.46)$$

Para $rl \geq n_2$ e se o sistema é controlável e observável, o posto da matriz Toeplitz é igual a n_2 . Aplicando-se a decomposição em valores singulares (SVD), usada para estimar o posto de uma matriz, à matriz Toeplitz obtém-se:

$$\mathbf{T}_{1|l}^{ref} = \mathbf{U} \mathbf{S} \mathbf{V}^T = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \mathbf{S}_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \end{bmatrix} = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^T \quad (4.47)$$

Sendo $\mathbf{U} \in \mathbb{R}^{l \times l}$ e $\mathbf{V} \in \mathbb{R}^{rl \times rl}$ matrizes unitárias e $\mathbf{S} \in \mathbb{R}^{l \times rl}$ uma matriz diagonal contendo os valores singulares em ordem decrescente. O posto da matriz é obtido contando os valores singulares não nulos. As matrizes $\mathbf{U}_1 \in \mathbb{R}^{l \times n_2}$, $\mathbf{S}_1 \in \mathbb{R}^{n_2 \times n_2}$ e $\mathbf{V}_1 \in \mathbb{R}^{rl \times n_2}$ são construídas omitindo os valores singulares nulos de \mathbf{S} e seus respectivos vetores de \mathbf{U} e \mathbf{V} .

As matrizes de observabilidade e controlabilidade podem ser obtidas:

$$\mathbf{O}_l = \mathbf{U}_1 \mathbf{S}_1^{1/2} \mathbf{T} \quad (4.48)$$

$$\mathbf{\Gamma}_l^{ref} = \mathbf{T}^{-1} \mathbf{S}_1^{1/2} \mathbf{V}_1^T \quad (4.49)$$

Sendo \mathbf{T} uma matriz não singular, a qual pode ser considerada igual a matriz identidade. Obtidas as matrizes de \mathbf{O}_l e $\mathbf{\Gamma}_l^{ref}$, utilizando a equação 4.45 e 4.46 obtém-se a matriz \mathbf{C} como as primeiras l linhas de \mathbf{O}_l e \mathbf{G}^{ref} como as últimas r colunas de $\mathbf{\Gamma}_l^{ref}$. A matriz \mathbf{A} pode ser obtida a partir da matriz de observabilidade, utilizando a notação do Python:

$$\mathbf{A} = \mathbf{O}_l[l(l-1),:]^\dagger \mathbf{O}_l[l:l+1,:] \quad (4.50)$$

Onde $(.)^\dagger$ indica matriz pseudo-inversa (Moore-Penrose) de \mathbf{O}_l pois esta é uma matriz retangular.

Com as matrizes \mathbf{A} e \mathbf{C} , é possível determinar as formas modais e as frequências naturais do sistema. Com efeito, a matriz de estado \mathbf{A} é decomposta na matriz diagonal de autovalores $\mathbf{\Lambda}_D$ e na matriz de autovetores $\mathbf{\Psi}$:

$$\mathbf{A} = \mathbf{\Psi} \mathbf{\Lambda}_D \mathbf{\Psi}^{-1} \quad (4.51)$$

Determinam-se as frequências naturais e razões de amortecimento a partir dos elementos de $\mathbf{\Lambda}_D$ (equações 3.12, 3.13 e 3.41). Finalmente, a matriz de formas modais do sistema original \mathbf{V} é calculada:

$$\mathbf{V} = \mathbf{C} \mathbf{\Psi} \quad (4.52)$$

Evidentemente, devido ao valor finito de amostras no tempo e hipóteses, as matrizes obtidas são estimativas das matrizes verdadeiras do sistema.

A implementação em Python é apresentada no Apêndice A. A função `SSI_COV` identifica o sistema de uma certa ordem em espaço de estado informada e a função `SSI_COV_iterator` itera a função `SSI_COV` para modelos de ordens crescentes. Assim como no método IV, desenha-se um diagrama de estabilização e os modos verdadeiros são identificados a partir dos polos estáveis. A matriz Toeplitz e a sua decomposição em valores singulares são calculadas uma única vez, devendo-se respeitar a inequação $ri \geq n_{2,\max}$ sendo $n_{2,\max}$ a máxima ordem do modelo em espaço de estado. Modelos de ordens inferiores ($n_2 < n_{2,\max}$) são facilmente obtidos simplesmente selecionando-se menos valores singulares de 4.47.

4.4 MÉTODOS NO DOMÍNIO DO TEMPO

A seguir descreve-se o método da Identificação Estocástica de Subespaço baseado nas séries temporais.

4.4.1 Método de identificação estocástica de subespaços baseado nas séries temporais (SSI DATA)

Este método emprega a representação do sistema em espaço de estado, é desenvolvido no domínio do tempo e evita o cálculo das funções de covariância. As saídas são organizadas em uma matriz bloco Hankel e divididas em respostas chamadas “passadas” e “futuras”. Com conceito de projeção geométrica, projetam-se as linhas das saídas futuras nas linhas das saídas passadas visando reter toda informação passada útil para prever o futuro. A apresentação que se faz em seguida do método SSI DATA segue a linha proposta por Peeters (2000) que modificou os algoritmos de Van Overschee e De Moor (1996) incluindo o conceito de sensores de referência.

Expõe-se inicialmente o conceito do filtro Kalman. Ele possui um papel importante nos algoritmos de identificação estocástica de subespaços, uma vez que este método envolve a avaliação da sequência de estado do filtro Kalman e sua utilização para estimativa das matrizes do sistema. Este filtro é uma técnica de estimação recursiva, ou seja, atualiza os parâmetros de um modelo no espaço de estado à medida que os dados do processo são disponibilizados. O seu principal objetivo é produzir uma predição ótima para o vetor de estado \vec{x}_k fazendo uso das

saídas anteriores no tempo e das matrizes do sistema disponíveis juntamente com as covariâncias conhecidas do ruído. Estas predições ótimas são representadas por $\hat{\mathbf{x}}_k$.

Dado o estado inicial estimado $\hat{\mathbf{x}}_0 = \mathbf{0}$, a covariância inicial do estado estimado $\mathbf{P}_0 = E[\hat{\mathbf{x}}_0 \hat{\mathbf{x}}_0^T] = \mathbf{0}$ e as saídas medidas y_0, \dots, y_{k-1} , o estado estimado $\hat{\mathbf{x}}_k$ é definido pelas seguintes equações recursivas:

$$\hat{\mathbf{x}}_k = \mathbf{A} \hat{\mathbf{x}}_{k-1} + \mathbf{K}_{k-1} (\vec{y}_{k-1} - \mathbf{C} \hat{\mathbf{x}}_{k-1}) \quad (4.53)$$

$$\mathbf{K}_{k-1} = (\mathbf{G} - \mathbf{A} \mathbf{P}_{k-1} \mathbf{C}^T) (\mathbf{R}_0 - \mathbf{C} \mathbf{P}_{k-1} \mathbf{C}^T)^{-1} \quad (4.54)$$

$$\mathbf{P}_k = \mathbf{A} \mathbf{P}_{k-1} \mathbf{A}^T + (\mathbf{G} - \mathbf{A} \mathbf{P}_{k-1} \mathbf{C}^T) (\mathbf{R}_0 - \mathbf{C} \mathbf{P}_{k-1} \mathbf{C}^T)^{-1} (\mathbf{G} - \mathbf{A} \mathbf{P}_{k-1} \mathbf{C}^T)^T \quad (4.55)$$

Define-se a sequência de estado do filtro Kalman $\hat{\mathbf{X}}_l \in \mathbb{R}^{n_2 \times N}$ como:

$$\hat{\mathbf{X}}_l = [\hat{\mathbf{x}}_l \quad \hat{\mathbf{x}}_{l+1} \quad \dots \quad \hat{\mathbf{x}}_{l+N-1}] \quad (4.56)$$

O método SSI DATA começa projetando o espaço das linhas das saídas futuras no espaço das linhas das saídas passadas. A ideia dessa projeção é reter toda a informação do passado que é útil para prever o futuro. A definição da projeção é:

$$\mathcal{P}_l^{ref} = \mathbf{Y}_f / \mathbf{Y}_p^{ref} = \mathbf{Y}_f (\mathbf{Y}_p^{ref})^T \left(\mathbf{Y}_p^{ref} (\mathbf{Y}_p^{ref})^T \right)^{\dagger} \mathbf{Y}_p^{ref} \quad (4.57)$$

Sendo as matrizes $\mathbf{Y}_f \in \mathbb{R}^{l \times N}$ e $\mathbf{Y}_p^{ref} \in \mathbb{R}^{r \times N}$ definidas em 4.4. A expressão 4.57 é a definição da projeção e não é a forma como ela é calculada no método. Conforme será apresentado, essa projeção é determinada por meio de um algoritmo de decomposição QR.

O principal teorema da identificação estocástica de subespaço afirma que a projeção \mathcal{P}_l^{ref} pode ser decomposta no produto da matriz observabilidade \mathbf{O}_l e da sequência de estado do filtro Kalman $\hat{\mathbf{X}}_l$:

$$\mathcal{P}_t^{ref} = \mathbf{O}_t \widehat{\mathbf{X}}_t \quad (4.58)$$

Aplicando a decomposição em valores singulares na matriz das projeções e omitindo os valores nulos:

$$\mathcal{P}_t^{ref} = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^T \quad (4.59)$$

com $\mathbf{U}_1 \in \mathbb{R}^{r_l \times n_2}$, $\mathbf{S}_1 \in \mathbb{R}^{n_2 \times n_2}$ e $\mathbf{V}_1 \in \mathbb{R}^{N \times n_2}$. A partir delas, a matriz de observabilidade estendida e a sequência de estado podem ser obtidas:

$$\mathbf{O}_t = \mathbf{U}_1 \mathbf{S}_1^{1/2} \quad (4.60)$$

$$\widehat{\mathbf{X}}_t = \mathbf{O}_t^\dagger \mathcal{P}_t^{ref} \quad (4.61)$$

Se a separação entre referências passadas e futuras for colocada uma linha para baixo, outra projeção é definida:

$$\mathcal{P}_{t-1}^{ref} = \mathbf{Y}_f^- / \mathbf{Y}_p^{ref+} = \mathbf{O}_{t-1} \widehat{\mathbf{X}}_{t+1} \quad (4.62)$$

A matriz de observabilidade \mathbf{O}_{t-1} é obtida removendo-se as últimas l linhas de \mathbf{O}_t . A sequência de estado é:

$$\widehat{\mathbf{X}}_{t+1} = \mathbf{O}_{t-1}^\dagger \mathcal{P}_{t-1}^{ref} \quad (4.63)$$

Calculam-se \mathbf{A} e \mathbf{C} :

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{C} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{X}}_{t+1} \\ \mathbf{Y}_{t|t} \end{bmatrix} \widehat{\mathbf{X}}_t^\dagger \quad (4.64)$$

Algumas variantes do método multiplicam a projeção \mathcal{P}_t^{ref} por duas matrizes pesos \mathbf{W}_1 e \mathbf{W}_2 :

$$\mathbf{W}_1 \mathcal{P}_t^{ref} \mathbf{W}_2 = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \mathbf{S}_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \end{bmatrix} = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^T \quad (4.65)$$

O Quadro 3.1 apresenta as matrizes pesos para os algoritmos: *principal component* (PC), *unweighted principal component* (UPC) e *canonical variate algorithm* (CVA). Escolhendo matrizes pesos diferentes, cada algoritmo fornece matrizes numericamente diferentes, embora, ao final do processo, as matrizes do sistema serão equivalentes entre si (OVERSCHEE; MOOR, 1996).

Quadro 3.1 – Funções de peso

Algoritmo	\mathbf{W}_1	\mathbf{W}_2
PC	\mathbf{I}	$(\mathbf{Y}_p^{ref})^T (\mathbf{Y}_p^{ref} (\mathbf{Y}_p^{ref})^T)^{-1/2} \mathbf{Y}_p^{ref}$
UPC	\mathbf{I}	\mathbf{I}
CVA	$(\mathbf{Y}_f \mathbf{Y}_f^T)^{-1/2}$	\mathbf{I}

(fonte: elaborado pelo autor)

4.4.1.1 Algoritmo e implementação

Overschee e Moor (1996) apresentam três algoritmos para determinação das matrizes do sistema no espaço de estado. O primeiro deles foi empregado na função SSI_DATA juntamente com a variante UPC.

Na implementação computacional dos algoritmos de subespaço utilizando as séries temporais das saídas torna-se útil empregar a fatoração QR na matriz Hankel (OVERSCHEE; MOOR, 1996; PEETERS, 2000):

$$\mathbf{H}^{ref} = \begin{bmatrix} \mathbf{Y}_p^{ref} \\ \mathbf{Y}_f \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_p^{ref+} \\ \mathbf{Y}_{|l}^{\sim ref} \\ \mathbf{Y}_f^- \end{bmatrix} = \mathbf{RQ}^T \quad (4.66)$$

Sendo \mathbf{Q} uma matriz ortogonal e \mathbf{R} uma matriz triangular inferior. A matriz Hankel pode ser escrita na forma:

$$\mathbf{H}^{ref} = \begin{array}{cccccc}
& & rl & r & l-r & l(l-1) & N \\
& & \leftrightarrow & \leftrightarrow & \leftrightarrow & \leftrightarrow & \leftrightarrow \\
rl \downarrow & & \left[\mathbf{R}_{11} & 0 & 0 & 0 \right] & \left[\mathbf{Q}_1^T \right] \\
r \downarrow & & \left[\mathbf{R}_{21} & \mathbf{R}_{22} & 0 & 0 \right] & \left[\mathbf{Q}_2^T \right] \\
l-r \downarrow & & \left[\mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} & 0 \right] & \left[\mathbf{Q}_3^T \right] \\
l(l-1) \downarrow & & \left[\mathbf{R}_{41} & \mathbf{R}_{42} & \mathbf{R}_{43} & \mathbf{R}_{44} \right] & \left[\mathbf{Q}_4^T \right]
\end{array} \begin{array}{l} \uparrow rl \\ \uparrow r \\ \uparrow l-r \\ \uparrow l(l-1) \end{array} \quad (4.67)$$

Calculam-se as projeções:

$$\mathcal{P}_l^{ref} = \mathbf{Y}_f / \mathbf{Y}_p^{ref} = \begin{bmatrix} \mathbf{R}_{21} \\ \mathbf{R}_{31} \\ \mathbf{R}_{41} \end{bmatrix} \mathbf{Q}_1^T \quad (4.68)$$

$$\mathcal{P}_{l-1}^{ref} = \mathbf{Y}_f^- / \mathbf{Y}_p^{ref+} = [\mathbf{R}_{41} \quad \mathbf{R}_{42}] \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \end{bmatrix} \quad (4.69)$$

E a matriz $\mathbf{Y}_{l|l}$:

$$\mathbf{Y}_{l|l} = \begin{bmatrix} \mathbf{R}_{21} & \mathbf{R}_{22} & 0 \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \\ \mathbf{Q}_3^T \end{bmatrix} \quad (4.70)$$

Aplica-se a decomposição em valores singulares na projeção:

$$\mathcal{P}_l^{ref} = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \mathbf{S}_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \end{bmatrix} = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^T \quad (4.71)$$

Determina-se o posto inspecionando os valores singulares de \mathbf{S} e particionando a SVD para obter-se \mathbf{U}_1 e \mathbf{S}_1 .

Determinam-se \mathbf{O}_l e \mathbf{O}_{l-1} :

$$\mathbf{O}_l = \mathbf{U}_1 \mathbf{S}_1^{1/2} \quad (4.72)$$

$$\mathbf{O}_{t-1} = \underline{\mathbf{O}}_t = \mathbf{O}_t[:, -l, :] \quad (4.73)$$

Onde $\underline{\mathbf{O}}_t$ denota a matriz \mathbf{O}_t sem as últimas l linhas.

Determinam-se $\hat{\mathbf{X}}_t$ e $\hat{\mathbf{X}}_{t+1}$:

$$\hat{\mathbf{X}}_t = \mathbf{O}_t^\dagger \mathcal{P}_t^{ref} \quad (4.74)$$

$$\hat{\mathbf{X}}_{t+1} = \mathbf{O}_{t-1}^\dagger \mathcal{P}_{t-1}^{ref} \quad (4.75)$$

Calculam-se \mathbf{A} e \mathbf{C} :

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{C} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{X}}_{t+1} \\ \mathbf{Y}_{t|t} \end{bmatrix} \hat{\mathbf{X}}_t^\dagger \quad (4.76)$$

A matriz de estado \mathbf{A} é decomposta na matriz diagonal de autovalores $\mathbf{\Lambda}_D$ e na matriz de autovetores $\mathbf{\Psi}$:

$$\mathbf{A} = \mathbf{\Psi} \mathbf{\Lambda}_D \mathbf{\Psi}^{-1} \quad (4.77)$$

Determinam-se as frequências naturais e razões de amortecimento a partir dos elementos de $\mathbf{\Lambda}_D$ (equações 3.12, 3.13 e 3.41). A matriz de formas modais do sistema original \mathbf{V} é calculada:

$$\mathbf{V} = \mathbf{C} \mathbf{\Psi} \quad (4.78)$$

Na equação 4.63 os termos da matriz \mathbf{Q} se cancelam, então no primeiro passo (4.67) a matriz \mathbf{Q} não precisa ser calculada. Conseqüentemente ocorre uma redução considerável de dados ao substituir a matriz Hankel pela matriz \mathbf{R} . No Python a função `numpy.linalg.qr` permite o cálculo apenas da matriz \mathbf{R} informando o parâmetro `mode='r'`.

As mesmas observações feitas no método SSI COV sobre a determinação da ordem do modelo em espaço de estado n_2 se aplicam ao SSI DATA. A função `SSI_DATA_iterator` itera a função

SSI_DATA para modelos de ordens crescentes. Desenha-se um diagrama de estabilização e os modos verdadeiros são identificados a partir dos polos estáveis. A decomposição QR, as projeções e a sua decomposição em valores singulares são calculadas uma única vez, devendo-se respeitar a inequação $ri \geq n_{2,\max}$ sendo $n_{2,\max}$ a máxima ordem do modelo em espaço de estado. Modelos de ordens inferiores ($n_2 < n_{2,\max}$) são facilmente obtidos simplesmente selecionando-se menos valores singulares de 4.71.

4.5 TÉCNICAS PARA DISTINÇÃO DOS MODOS REAIS E ESPÚRIOS

Neste item descrevem-se formas de separação entre os modos estruturais e os demais, ou seja, como distinguir modos verdadeiros dos espúrios. Na estimativa das características dinâmicas a partir de dados reais é geralmente uma boa ideia superestimar a ordem do modelo, ou seja, tentar ajustar modelos de alta ordem que contém muito mais modos do que os presentes nos dados. Depois os modos físicos são separados dos espúrios.

Tratando-se de modelos no espaço de estado, a sua representação não é única. Se a ordem for muito pequena, provavelmente acontecerá perda de informação do sistema. Por outro lado, se a ordem for muito elevada, o modelo conterá informações redundantes. Ambos os casos devem ser evitados. Assim, a realização deve ter uma dimensão mínima no espaço de estado, onde todos os modos do sistema são observados. Essa realização é chamada de realização mínima (ANDERSEN, 1997).

Uma estimativa do número mínimo de frequências naturais presentes nas medições pode ser feita contando-se os picos das densidades espectrais das respostas. Essa contagem fornece um limite inferior pois existe a possibilidade de os sensores estarem posicionados em um nó da forma modal (assim não podendo detectá-la) ou a existência de modos muito próximos, não podendo ser distinguíveis no espectro da resposta. Por isso, uma estimativa adequada do número da ordem do modelo é um passo importante no processo de identificação de sistemas (ALLEMANG et al., 1994).

Uma forma de identificar a ordem do modelo é pelo gráfico do erro. Nesse gráfico basicamente visualiza-se a curva do erro no modelo em função da sua ordem. O erro é uma grandeza normalizada que representa a capacidade do modelo em prever dados que não foram empregados para a sua identificação. Quando a ordem do modelo é insuficiente, o erro é grande,

mas quando a ordem do modelo atingir o valor correto, um aumento na sua ordem não irá resultar em uma diminuição significativa do erro (ALLEMANG et al., 1994).

A maioria das estruturas civis é subamortecida. As razões de amortecimento são baixas, não maiores do que uma pequena porcentagem. Isso cria duas formas de seleção de modos: aqueles com baixo amortecimento e que apresentam autovalores em par complexo conjugado. Técnicas mais sofisticadas são apresentadas a seguir.

4.5.1 Decomposição em Valores Singulares

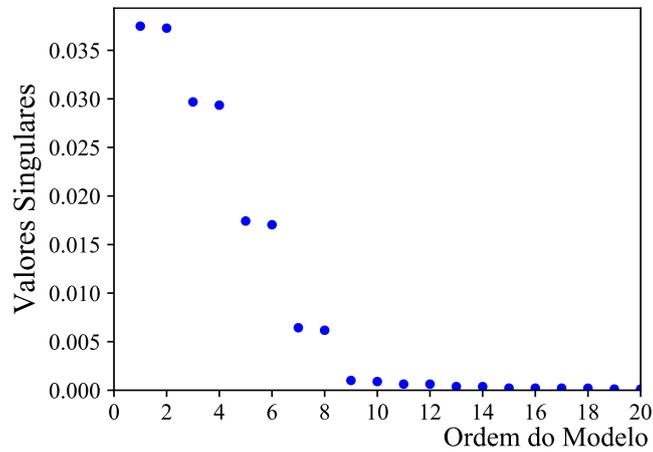
O posto da matriz dos dados experimentais fornece uma boa estimativa da ordem do modelo do sistema. Basicamente, o posto indica o número de linhas linearmente independentes. Como ele não pode ser calculado de maneira absoluta, ele é estimado pela decomposição em valores singulares.

Qualquer matriz \mathbf{A} pode ser decomposta em três matrizes \mathbf{U} , \mathbf{S} e \mathbf{V} na forma $\mathbf{A} = \mathbf{USV}^T$. As matrizes possuem características importantes as quais são úteis para descrever o sistema representado por \mathbf{A} . As matrizes \mathbf{U} e \mathbf{V} são unitárias, ou seja, $\mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{I}$ e $\mathbf{VV}^T = \mathbf{I}$. \mathbf{S} é uma matriz diagonal cujos valores representam a magnitude do produto das matrizes \mathbf{U} e \mathbf{V} . Estes valores na diagonal são chamados de valores singulares e sua magnitude diminui com o aumento do número da coluna. Isso significa que a primeira coluna de \mathbf{U} e \mathbf{V} é a aquela com maior influência na matriz \mathbf{A} (ANDERSEN, 1997).

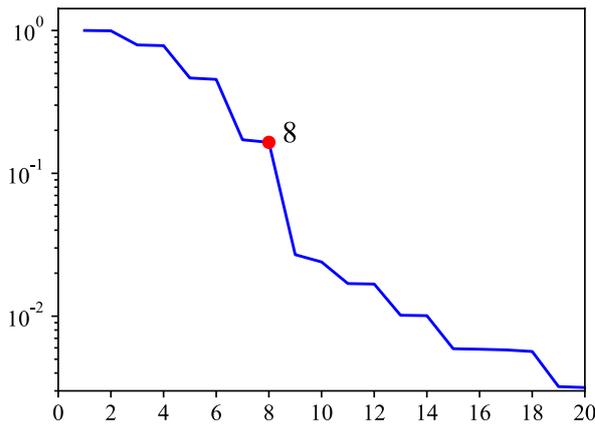
A contagem dos valores singulares não nulos permite a identificação do número de modos verdadeiros. Quando a ordem verdadeira do modelo for excedida, os valores singulares irão teoricamente para zero. Na Figura 4.1a é possível identificar oito valores singulares não nulos, indicando que se trata de um modelo com ordem no espaço de estado $n_2 = 8$. Para dados experimentais, os valores singulares não são exatamente zero, mas muito pequenos. Assim, a taxa de variação dos valores singulares em função da ordem do modelo é utilizada como indicador. Uma abordagem é normalizar os valores singulares pelo primeiro (e maior), conforme Figura 4.1b. O gráfico normalizado é tratado de forma semelhante ao gráfico do erro, no qual o posto verdadeiro é escolhido quando a curva se torna assintótica. Outra abordagem é dividir cada valor singular pelo imediatamente anterior (Figura 4.1c) formando uma curva com valores próximos da unidade se os valores não mudarem de magnitude. A ordem do modelo é

identificada quando a relação cai subitamente (ALLEMANG et al., 1994; JUANG; PAPPA, 1985). A função `plot_singular_values` plota os gráficos dos valores singulares.

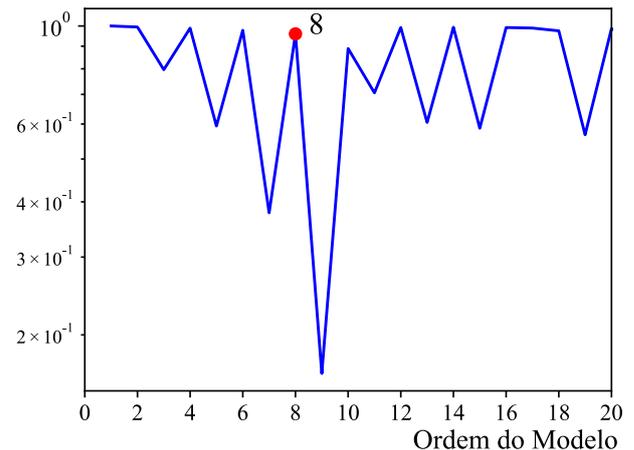
Figura 4.1 – Valores singulares



(a) valores singulares



(b) normalizados pelo primeiro



(c) normalizados pelo anterior

(fonte: elaborado pelo autor)

4.5.2 Diagrama de estabilização

No diagrama de estabilização rastreiam-se as frequências, razões de amortecimentos e formas modais estimadas em função da ordem do modelo. Conforme a ordem do modelo aumenta, mais frequências são estimadas, mas os modos verdadeiros estabilizam. Para modos bastante presentes nas medições, os parâmetros modais estabilizarão para ordens baixas. Para modos pouco excitados, os parâmetros modais podem não estabilizar até uma ordem muito alta. Não obstante, modos espúrios não estabilizarão e podem ser separados. A existência de ruídos e erros nas medições pode dificultar o emprego do diagrama. Na prática, adota-se uma tolerância,

expressa em porcentagem, para a classificação de um modo como estável (ALLEMANG et al., 1994; ANDERSEN, 1997).

O diagrama de estabilização é uma ferramenta visual poderosa para a identificação de modos verdadeiros. As frequências naturais verdadeiras estabilizarão com a dimensão do modelo, assim como as razões de amortecimento e as formas modais. Isso significa que os modos verdadeiros podem ser identificados comparando as características dinâmicas de modelos com ordens diferentes. Escolhendo, por exemplo, os seguintes critérios: 1% para frequências naturais, 5% para razões de amortecimento e 10% para formas modais, compara-se um modelo com ordem n com um modelo de ordem $n + 1$. Um modo estável requer:

$$\frac{|f^{(n)} - f^{(n+1)}|}{f^{(n)}} < 1\% \quad (4.79)$$

$$\frac{|\xi^{(n)} - \xi^{(n+1)}|}{\xi^{(n)}} < 5\% \quad (4.80)$$

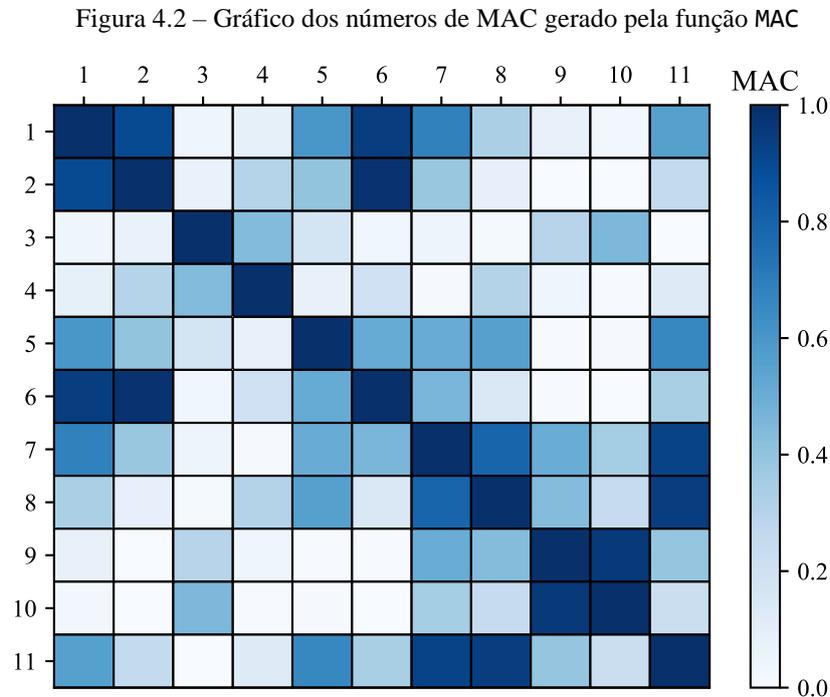
$$1 - \text{MAC}(n, n + 1) < 10\% \quad (4.81)$$

O Critério de Concordância Modal (*Modal Assurance Criterion* – MAC), proposto por Allemang e Brown (1982), é uma medida de correlação entre duas formas modais. O seu valor varia entre 0 e 1, sendo igual zero para modos não correlacionados e um para modos perfeitamente correlacionados. Sejam as formas modais complexas $\vec{\psi}_i$ e $\vec{\psi}_j$, o MAC é calculado pela equação 4.82:

$$\text{MAC}(i, j) = \frac{|\vec{\psi}_i^T \vec{\psi}_j^*|^2}{(\vec{\psi}_i^T \vec{\psi}_i^*)(\vec{\psi}_j^T \vec{\psi}_j^*)} \quad (4.82)$$

Os vetores $\vec{\psi}_i$ e $\vec{\psi}_j$ podem ser substituídos por matrizes cujas colunas são as formas modais. Neste caso, o emprego da expressão 4.82 resultará em uma matriz. Uma forma útil de visualizar

os valores do MAC é de forma gráfica, permitindo que mais dados sejam apresentados e compreendidos, conforme Figura 4.2.

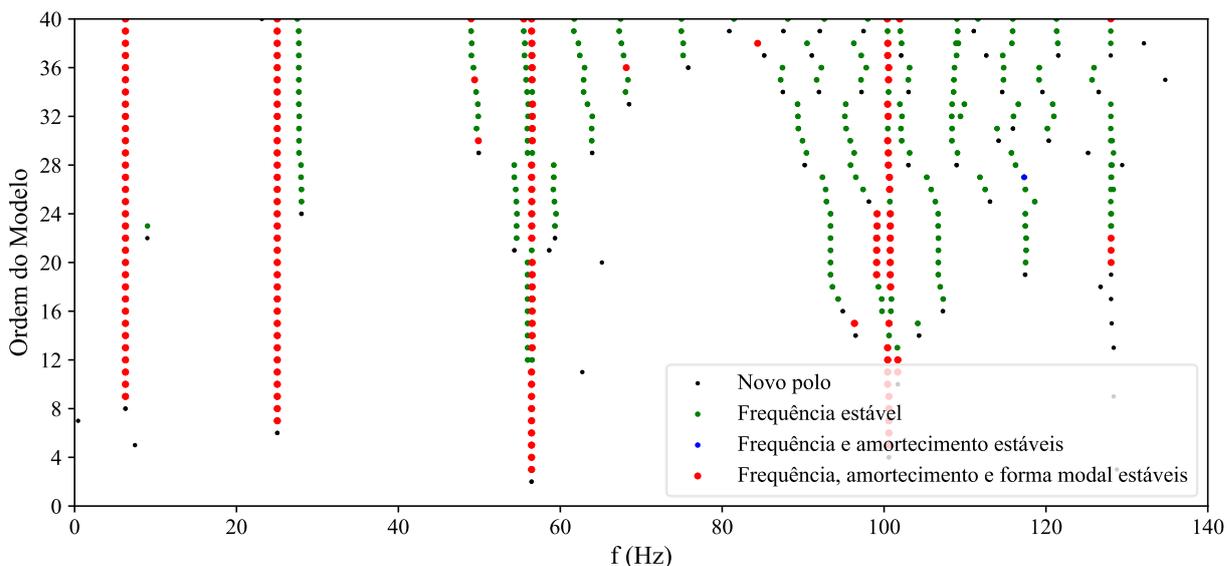


(fonte: elaborado pelo autor)

A Figura 4.3 mostra um diagrama de estabilização gerado pela função `stabilization_diagram`. Identificam-se quatro modos estáveis com frequência de 6 Hz, 25 Hz, 56 Hz e 100 Hz.

Conforme mencionado, os modos detectados são comparados com os modos de um modelo de ordem imediatamente inferior e classificados quanto a sua estabilidade pelas inequações 4.79 a 4.81. Um novo modo detectado é representado por um ponto na cor preta. Caso apenas a sua frequência estabilize (respeitando a inequação 4.69), representa-se o ponto com a cor verde. Se além da frequência, a razão de amortecimento respeitar a inequação 4.80, o ponto recebe a cor azul. Por fim, caso a frequência, razão de amortecimento e forma modal sejam estáveis, ou seja, se elas respeitam as inequações 4.79 a 4.81, desenha-se o ponto na cor vermelha.

Figura 4.3 – Diagrama de estabilização



(fonte: elaborado pelo autor)

4.6 IMPLEMENTAÇÃO EM PYTHON

Existem softwares comerciais voltados para a identificação de sistemas, como ARTeMIS ou LMS Test.Lab. Entretanto estes softwares proprietários possuem desvantagens num ambiente de pesquisa, como custo de aquisição e impossibilidade de acesso à implementação dos algoritmos. Nesse sentido, a necessidade de desenvolvimento de rotinas específicas para a identificação estocástica de sistemas se torna clara.

Diversos pesquisadores desenvolvem seus programas em MATLAB (ANDERSEN, 1997; MIGUEL, 2007; PEETERS, 2000; RODRIGUES, 2004). Também se destaca o uso de LabVIEW (HU et al., 2010; RODRIGUES, 2004) e C (ASMUSSEN, 1997; FELBER, 1993). Existem pacotes para identificação de sistemas com entradas e saídas medidas em MATLAB (por exemplo LJUNG, 2012) e Python (ARMENISE et al., 2018). Durante a elaboração deste trabalho, nenhuma implementação em Python dos métodos descritos neste capítulo foi encontrada pelo autor.

Python é uma linguagem de programação de alto nível, interpretada, com desenvolvimento comunitário e aberto criada pelo holandês Guido van Rossum e lançada em 1991. O autor priorizou a legibilidade do código, empregando espaços em branco e indentação ao invés de chaves.

Programaram-se diversas funções para a utilização dos métodos de identificação estocástica de sistemas descritos no capítulo. Esta etapa do trabalho mostrou-se muito importante para uma melhor compreensão dos métodos estudados. O módulo desenvolvido denomina-se CESSIPy (*Civil Engineer Stochastic System Identification for Python*). Trata-se de um módulo de código aberto, gratuito e que se encontra disponível para download na plataforma GitHub². Visando facilitar o seu uso e implementação, elaborou-se um guia para usuário, o qual encontra-se no Apêndice B. O código foi validado por meio de simulações e pela análise de dados de estruturas reais. Os resultados são apresentados no próximo capítulo.

Empregou-se a distribuição de Python feita pela Anaconda³ versão 3.7, a qual agrupa diversos pacotes de computação científica. Os códigos foram programados no ambiente de desenvolvimento integrado Spyder, com uso das bibliotecas NumPy, SciPy, Matplotlib e Pandas.

A Figura 4.4 mostra as principais funções elaboradas, parte das quais foram citadas ao longo deste capítulo. A seguir apresenta-se uma breve descrição de cada uma delas. Os respectivos códigos encontram-se no Apêndice A.

Funções auxiliares empregadas para organizar as séries temporais de dados:

- a) `rearrange_data`: reordena as séries temporais posicionando as saídas de referência nas primeiras linhas;
- b) `Toeplitz`: calcula a matriz Toeplitz com as estimativas das funções de covariância;
- c) `projection`: calcula a matriz Hankel, decompondo-a nas matrizes QR e obtendo as matrizes \mathcal{P}_{t-1}^{ref} , \mathcal{P}_t^{ref} e $\mathbf{Y}_{|t}$.

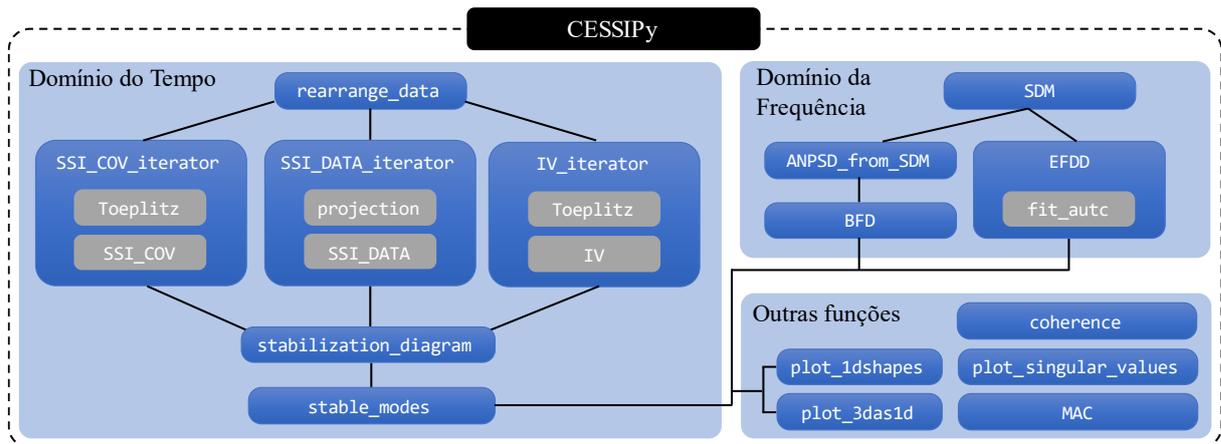
As funções dos métodos de identificação no domínio do tempo, as quais fornecem as frequências naturais, razões de amortecimento e formas modais para um modelo em espaço de estado com ordem informada pelo usuário, são:

- a) `SSI_COV`;
- b) `SSI_DATA`;
- c) `IV`.

² Disponível para download em < <https://github.com/MatheusCarini/CESSIPy> >.

³ Disponível para download em < <https://www.anaconda.com/distribution> >.

Figura 4.4 – Funções do módulo CESSIPy



(fonte: elaborado pelo autor)

Os seus respectivos iteradores SSI_COV_iterator, SSI_DATA_iterator e IV_iterator, os quais fornecem as características dinâmicas para modelos de ordem crescente.

Referente ao domínio da frequência, as funções programadas são:

- SDM: calcula e plota a matriz com as estimativas das densidades espectrais;
- ANPSD_from_SDM: calcula a ANPSD a partir das densidades espectrais;
- coherence: calcula e plota as funções de coerência;
- BFD: estima as propriedades dinâmicas pelo Método Básico no Domínio Frequência;
- EFDD: estima as propriedades dinâmicas pelo Método Aperfeiçoado de Decomposição no Domínio da Frequência.

Funções para desenho de gráficos:

- stabilization_diagram: desenha o diagrama de estabilização a partir dos dados fornecidos pelas funções iteradoras e retorna uma matriz indicando os polos estáveis;
- plot_singular_values: desenha os valores singulares;
- plot_1dshapes: desenha as formas modais para o caso unidimensional;
- plot_3das1d: desenha as formas modais de edifícios (duas translações e uma rotação) como formas modais unidimensionais.

E algumas funções auxiliares:

- stable_modes: identifica um conjunto de modos estáveis no diagrama de interação, agrupando os modos estáveis semelhantes entre si;

- b) **MAC**: calcula o Critério de Concordância Modal e, caso especificado, fornece um gráfico dos valores.

5 APLICAÇÕES

Este capítulo é dedicado à descrição das aplicações e apresentação dos resultados. Inicialmente apresentam-se simulações da ação do vento no CAARC e sua respectiva identificação para um sensor localizado no topo. Em seguida realizam-se a identificação de duas passarelas: a Passarela P01, uma passarela em viga reta pré-moldada e protendida e a Passarela P02, construída em estrutura metálica treliçada. Finaliza-se com a identificação da Ponte Z24 na Suíça, a qual é um *benchmark* na análise modal estocástica, e comparam-se os resultados obtidos com os de outros autores.

5.1 ANÁLISE NUMÉRICA DO CAARC SOB AÇÃO DO VENTO

Neste item apresentam-se estudos numéricos da identificação de um edifício alto sob ação do vento. As forças foram obtidas a partir de séries temporais de pressões de um modelo reduzido ensaiado por Oliveira (2009) no Túnel de Vento Professor Joaquim Blessmann com o método *High-Frequency Pressure Integration* (HFPI). Do ponto de vista de simulação, trata-se de uma forma extremamente confiável de determinar a resposta estrutural de edifícios frente à ação do vento. A partir dessa resposta, aplicaram-se os métodos de identificação de sistemas.

Inicialmente são apresentadas as características da edificação estudada, em seguida descreve-se o método HFPI e, por fim, discutem-se os resultados. Salienta-se que o processo de identificação foi simulado a partir das acelerações obtidas pelo método HFPI. Não se trata de uma estrutura instrumentada e cuja aceleração foi medida.

5.1.1 O CAARC *Standard Tall Building*

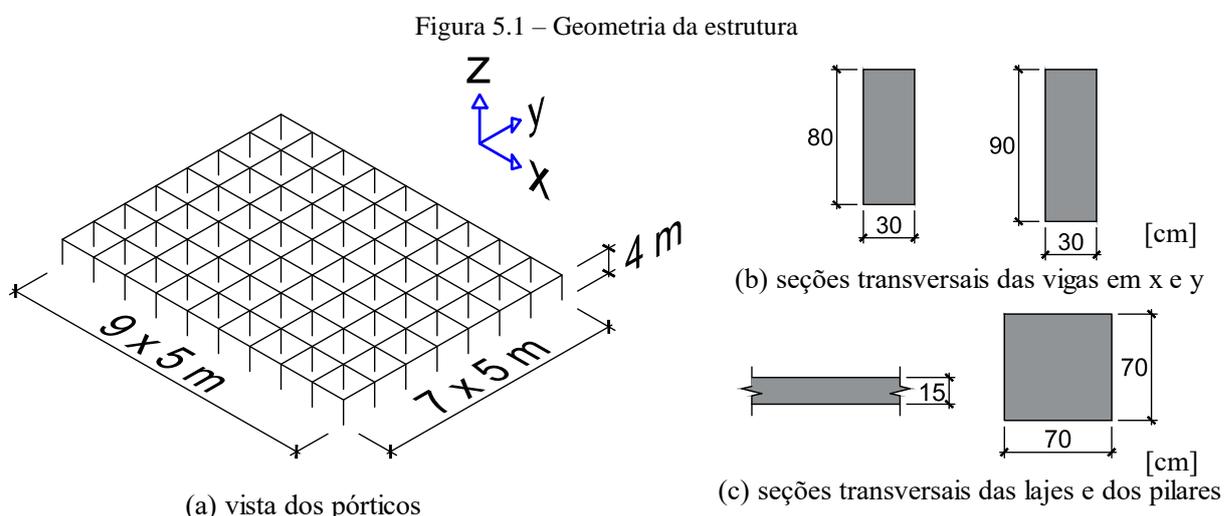
O CAARC *Standard Tall Building* é um modelo de edifício alto padronizado. Melbourne (1980) afirma que no encontro do *Commonwealth Advisory Aeronautical Research Council Coordinators in the Field of Aerodynamics* em 1969, uma especificação de um modelo foi preparada por Wardlaw e Moss visando comparar os ventos naturais simulados em diversos túneis de vento. Foi proposto um experimento simples para comparação entre as diferentes técnicas utilizadas. Era esperado que a comparação direta da resposta dinâmica e das pressões

medidas ajudassem no desenvolvimento de melhores técnicas e dessem mais credibilidade aos resultados experimentais.

O CAARC tem formato de um prisma retangular, com base de 30,48 m por 45,72 m e altura de 182,88 m (100 ft x 150 ft x 600 ft). O topo do edifício é plano, sem parapeitos, as fachadas são planas e sem detalhes geométricos. Sugere-se que o expoente da lei de potência, a qual representa o perfil vertical da velocidade do vento, seja em torno de 0,28. A camada limite simulada deve representar o vento soprando por um longo terreno de florestas ou terrenos urbanos desenvolvidos com edifícios de altura entre 6 e 15 m.

Melbourne (1980) especifica algumas características dinâmicas do CAARC, as quais foram parcialmente utilizadas neste estudo. A descrição detalhada destas características é apresentada a seguir.

Para a determinação das propriedades dinâmicas do CAARC, considerou-se uma estrutura formada por pórticos. Previram-se pórticos com nove vãos de cinco metros na direção x e com sete vãos de cinco metros na direção y. A estrutura possui 45 pavimentos com 4 metros de altura. As seções transversais dos elementos estruturais encontram-se na Figura 5.1.



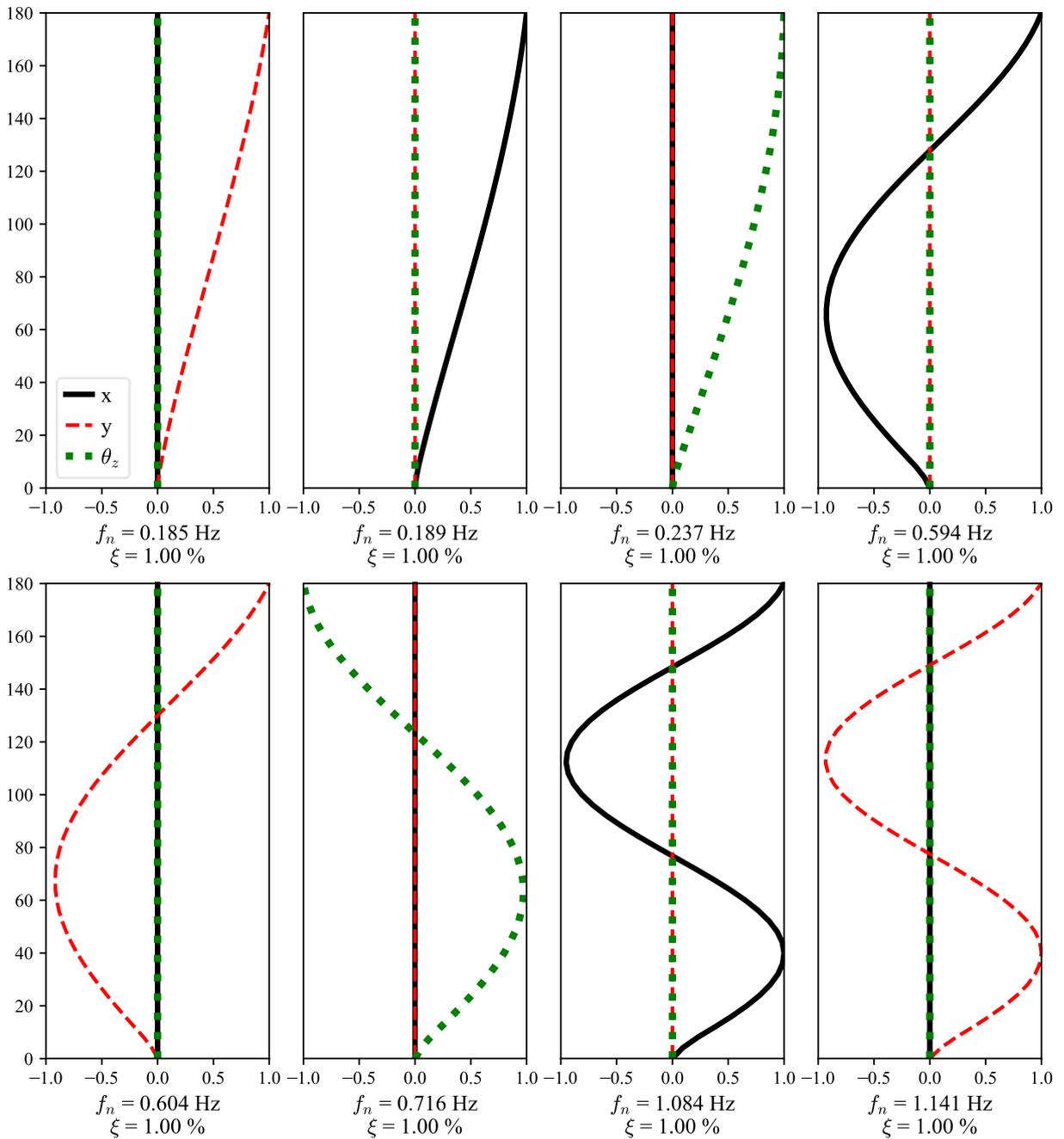
(fonte: elaborado pelo autor)

As frequências naturais e formas modais empregadas nas análises foram obtidas do modelo em elementos finitos da estrutura no software ANSYS Mechanical APDL. As lajes foram modeladas com elemento de casca SHELL93, as vigas e os pilares com elementos de viga BEAM189. Considerou-se material elástico linear, isotrópico, com módulo de elasticidade

igual a 35 GPa e coeficiente de Poisson igual a 0,2. A massa de cada pavimento foi disposta nas lajes, considerando uma massa aparente de 160 kg/m³.

A Figura 5.2 apresenta as formas modais e as frequências naturais fornecidas pelo ANSYS. As razões de amortecimento utilizadas foram iguais a 1%. Os três primeiros modos possuem forma modal aproximadamente linear, sendo o primeiro correspondente à translação da estrutura na direção y, o segundo modo corresponde à translação na direção x e o terceiro modo de torção.

Figura 5.2 – Formas modais da estrutura estudada



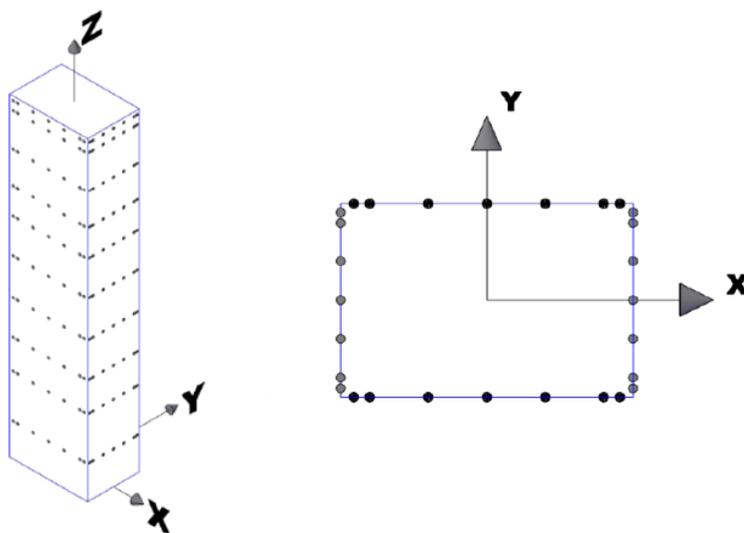
(fonte: elaborado pelo autor)

5.1.2 High-Frequency Pressure Integration (HFPI)

Modelos reduzidos são amplamente utilizados em estudos, por várias razões: rapidez na obtenção dos resultados, economia, estudo inicial de viabilidade de projeto, verificação dos efeitos de mudanças na forma exterior, entre outros (BLESSMANN, 2011). No HFPI mede-se simultaneamente as pressões em diversos locais do modelo (Figura 5.3) com transdutores elétricos. Essas séries temporais de pressões incorporam as densidades espectrais e funções de coerência corretas para o vento simulado. Com as pressões, a área de contribuição de cada transdutor e sua localização, determinam-se as séries temporais das forças ao longo da altura, e, conseqüentemente, os esforços na base.

Segundo Rocha e Loredó-Souza (2008), o HFPI é um método de análise que combina pressões dinâmicas medidas experimentalmente em túnel de vento com um modelo dinâmico teórico-numérico da estrutura, permitindo uma estimativa das amplitudes de deslocamentos, velocidades e acelerações que ocorrerão em resposta à ação do vento sobre a edificação. O método também permite a avaliação de níveis de conforto aos quais os usuários da edificação serão submetidos ao longo do tempo.

Figura 5.3 – Localização de tomadas de pressão no modelo do edifício



(fonte: adaptada de OLIVEIRA, 2009, p. 136)

O cálculo da resposta estrutural é feito pelo método de superposição modal, o qual necessita da distribuição de massa, das formas modais e frequências naturais de vibração livre. No caso de edifícios altos, as formas modais são usualmente descritas com três coordenadas por pavimento

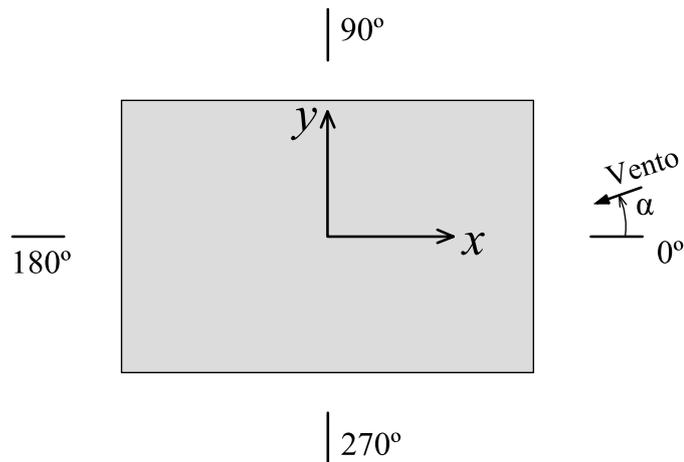
(duas de translação e uma de rotação), obtidas sobre uma linha vertical coincidente com o eixo de rigidez da estrutura. Da superposição modal resultam equações diferenciais de equilíbrio dinâmico desacopladas, cujas soluções podem ser obtidas por um método numérico como o método de Duhamel (domínio do tempo) ou método da transformada complexa de Fourier (domínio da frequência). Por fim, as soluções desacopladas são superpostas para a obtenção da resposta estrutural (ROCHA; LOREDO-SOUZA, 2008).

A validade dos resultados do HFPI depende da possibilidade de ser desprezada a interação fluido-estrutura, que pode produzir efeitos de instabilidade ou de segunda ordem na resposta estrutural. Em termos gerais, podem-se considerar como válidos os resultados do HFPI quando as deformações da estrutura não implicarem alterações significativas na forma do escoamento do ar em seu entorno. Esta hipótese é geralmente válida para edifícios altos, mas não pode ser adotada, por exemplo, para tabuleiros de pontes suspensas ou estaiadas mais propensos a drapejamento (*flutter*). Nos casos em que a interação fluido-estrutura é relevante, faz-se indispensável a utilização de modelos reduzidos aeroelásticos (ROCHA; LOREDO-SOUZA, 2008).

No presente trabalho empregaram-se dados de medidas de pressões em alta frequência de ensaios realizados por Oliveira (2009) no Laboratório de Aerodinâmica das Construções da UFRGS. Trata-se de um modelo rígido, construído com chapas de acrílico de 3 mm de espessura e escala geométrica 1:400. O modelo foi instrumentado com 280 tomadas de pressão, distribuídas em 10 linhas horizontais de 28 tomadas, conforme Figura 5.3.

As pressões nas tomadas foram medidas com a utilização de cinco módulos de transdutores eletrônicos de pressão da marca Scanivalve Corp., modelo ZOC33, cada um com 64 sensores, capazes de medir pressões de até 20 mm de água, com imprecisão de apenas 0,12%. Os ensaios foram realizados com velocidades em torno de 26 m/s para 24 incidências do vento, espaçadas de 15 em 15 graus (Figura 5.4). A cada configuração foram adquiridas 8192 medidas de pressão por tomada, a uma taxa de aquisição de aproximadamente 488 Hz (OLIVEIRA, 2009).

Figura 5.4 – Direções do vento



(fonte: elaborado pelo autor)

5.1.3 Leis de semelhança

Além da correta simulação do vento natural, a modelagem em túnel de vento requer que sejam respeitadas uma série de relações entre características geométricas e dinâmicas das estruturas do edifício real e do modelo (OLIVEIRA, 2009). Essas relações são chamadas de fatores de escala e alguns deles são apresentados a seguir.

A escala de comprimento λ_L , dada pela equação 5.1, é a razão entre um comprimento do modelo L_m e o seu correspondente no protótipo L_p . O modelo construído por Oliveira (2009) possui um fator de escala de comprimento de 1:400.

$$\lambda_L = \frac{L_m}{L_p} \quad (5.1)$$

A escala de velocidade é calculada pela equação 5.2. Os ensaios foram realizados com velocidade do vento em torno de 26 m/s (OLIVEIRA, 2009). Conforme será apresentado nos próximos itens, nas simulações adotaram-se várias velocidades do vento no protótipo, resultando em diferentes escalas de velocidades.

$$\lambda_V = \frac{V_m}{V_p} \quad (5.2)$$

Sendo:

λ_v – escala de velocidade;

$V_m = 26 \text{ m/s}$ – velocidade no modelo;

V_p – velocidade no protótipo.

Definidas as escalas de comprimento e de velocidade, determina-se a escala de tempo pela equação 5.3. A duração do ensaio realizado por Oliveira (2009) é de 16,77 s.

$$\lambda_t = \frac{t_m}{t_p} = \frac{\lambda_L}{\lambda_v} \quad (5.3)$$

Sendo:

λ_t – escala de tempo;

$t_m = 16,77 \text{ s}$ – tempo no modelo;

t_p – tempo no protótipo.

A escala de frequência é o inverso da escala de tempo (equação 5.4). Segundo Oliveira (2009), empregou-se a frequência de amostragem no modelo de aproximadamente 488 Hz.

$$\lambda_f = \frac{f_m}{f_p} = \frac{1}{\lambda_t} \quad (5.4)$$

Sendo:

λ_f – escala de frequência;

$f_m = 488 \text{ Hz}$ – frequência no modelo;

f_p – frequência no protótipo.

5.1.4 Um sensor no topo do edifício

As séries temporais das pressões obtidas no túnel de vento foram combinadas com as características dinâmicas da estrutura para a obtenção dos deslocamentos, velocidades e acelerações em cada pavimento. Neste item simula-se o caso de um único sensor posicionado no topo do edifício, exatamente no centro em planta baixa, capaz de medir acelerações lineares em x e y e aceleração angular em relação ao eixo z. No processo de identificação, essas acelerações foram as únicas informações empregadas.

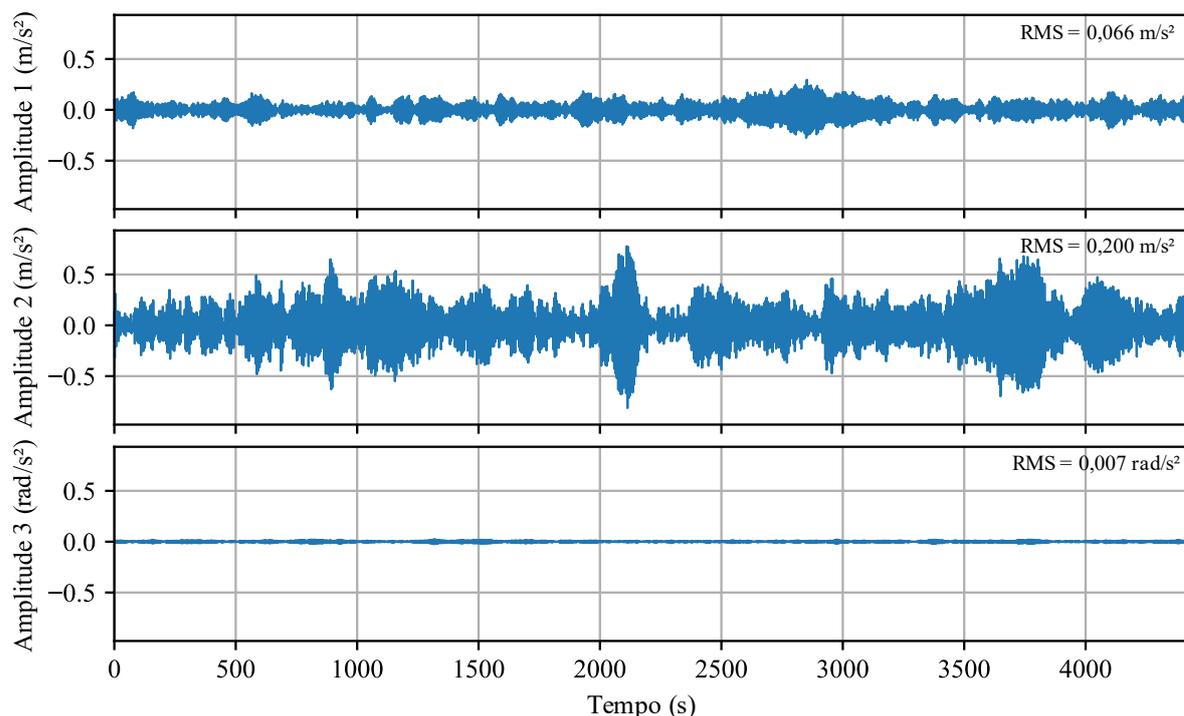
Das 24 direções do vento disponíveis, empregou-se apenas a direção de 0 graus, correspondente ao vento paralelo ao eixo x. Simulações preliminares foram realizadas para a determinação de alguns parâmetros. As tolerâncias do diagrama de estabilização foram definidas: 1% para frequências, 5% para amortecimentos e 10% para formas modais. Restringiu-se o estudo para as frequências entre 0 e 1 Hz e apenas as razões de amortecimento com valor inferior a 3% foram classificadas como estáveis. A ordem máxima dos modelos em espaço de estado foi de 48. O parâmetro ι foi considerado igual a 40, valor comumente adotado na literatura técnica, por exemplo empregado por Peeters (2000).

5.1.4.1 Cenário 1: evento extremo com ausência de ruído

A primeira análise foi feita com velocidade do vento no topo do edifício igual a 40 m/s, tratando-se, portanto, de um evento extremo com baixíssima probabilidade de ocorrência. Este estudo visa apresentar uma explicação detalhada do processo de identificação. Não foi considerada a existência de ruído nas medições.

A Figura 5.5 apresenta os registros de acelerações empregados no processo de identificação. Denotou-se a aceleração na direção x, paralela ao vento, pelo índice 1, na direção y, transversal ao vento, pelo índice 2 e angular pelo índice 3. Para a velocidade do vento adotada, o tempo total de aquisição no protótipo é de aproximadamente 4450 segundos e a frequência de aquisição igual a 1,84 Hz (equações 5.3 e 5.4, respectivamente). Isso indica que, devido ao teorema de Nyquist, apenas frequências naturais abaixo de 0,92 Hz poderão ser detectadas.

Figura 5.5 – Registros de acelerações no topo do protótipo

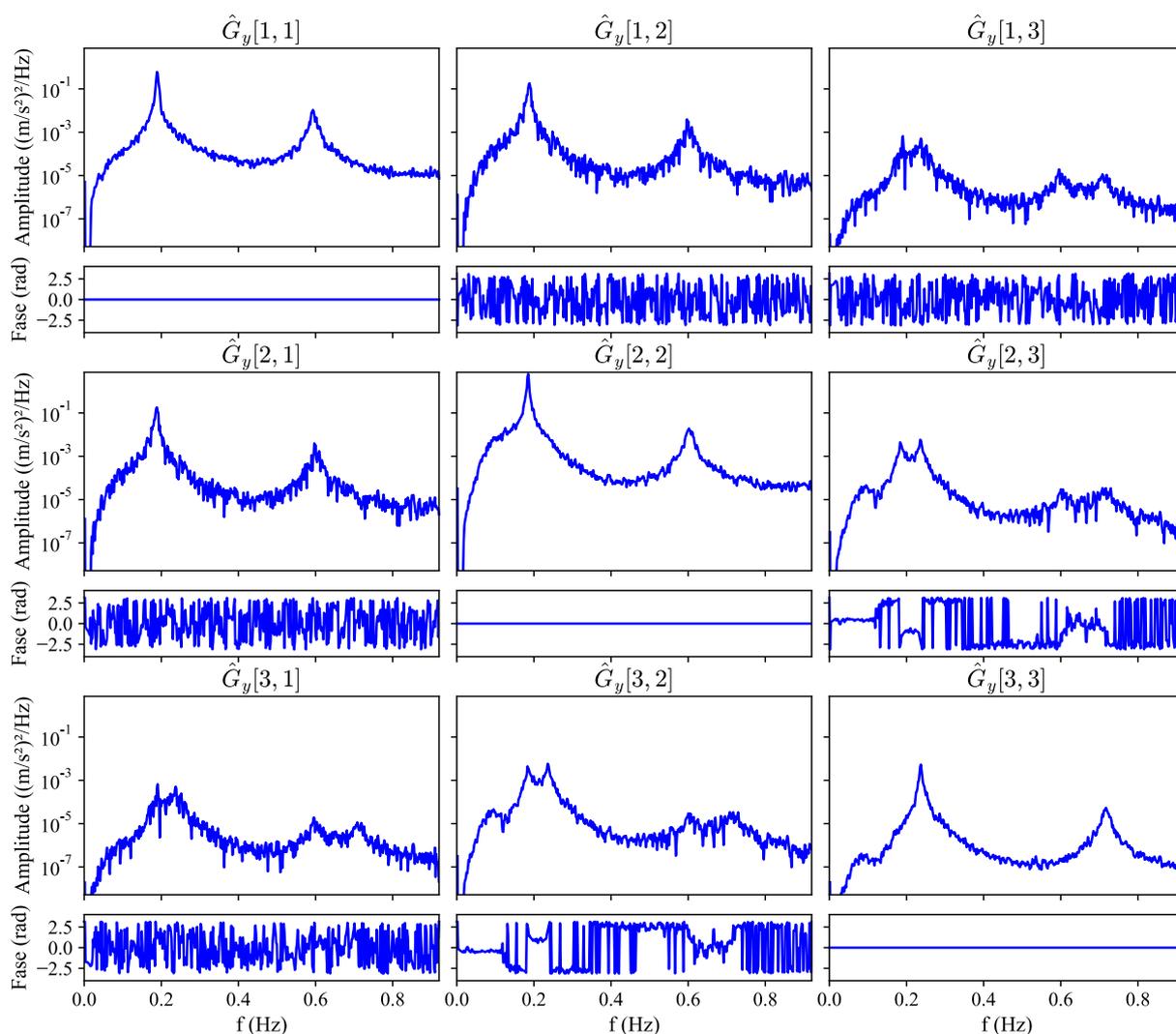


(fonte: elaborado pelo autor)

Inicia-se a apresentação dos resultados pelo Método Básico no Domínio da Frequência (BFD). A partir das séries temporais de acelerações no topo, estimaram-se as funções de densidades espectrais das respostas pelo método de Welch conforme apresentado no item 4.1.3. Os registros com 8192 valores foram divididos em 15 amostras com 1024 valores cada. Considerou-se uma sobreposição de 50% e utilizou-se uma janela de dados do tipo Hanning. Obtiveram-se estimativas com uma resolução em frequência de $\Delta f \cong 0,0018 \text{ Hz}$.

A Figura 5.6 mostra as estimativas das funções de densidades espectrais das respostas separadas em amplitude e fase. Na diagonal representam-se os autoespectros, cuja fase é nula, e os demais gráficos mostram os espectros cruzados.

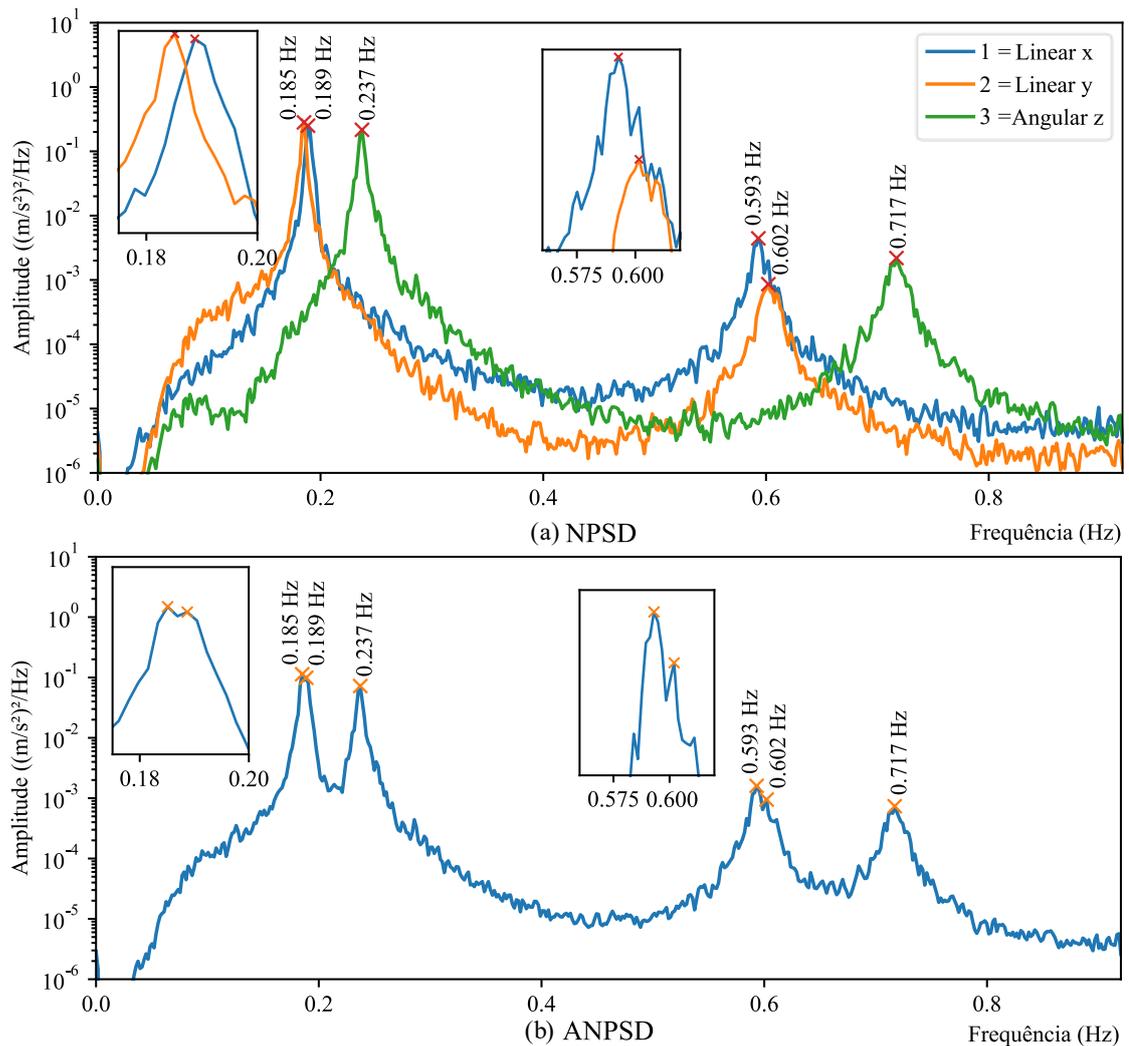
Figura 5.6 – Gráficos das densidades espectrais da resposta em acelerações



(fonte: elaborado pelo autor)

A partir das funções de densidades espectrais determinaram-se os autoespectros normalizados (NPSD) e a sua média (ANPSD), os quais são apresentados na Figura 5.7. No gráfico dos NPSD encontram-se indicadas as frequências correspondentes aos pontos de maior amplitude. É possível distinguir com clareza seis picos, dois para cada NPSD, os quais correspondem aos modos de vibração da estrutura. Referente aos modos com frequências naturais próximas (primeiro e segundo modos; quarto e quinto modos), verifica-se que os picos ocorrem em frequências distintas. No gráfico da ANPSD, por sintetizar os autoespectros em uma única curva, não é possível diferenciar as frequências naturais próximas. Neste exemplo, com apenas três séries temporais, não seria muito importante calcular a ANPSD.

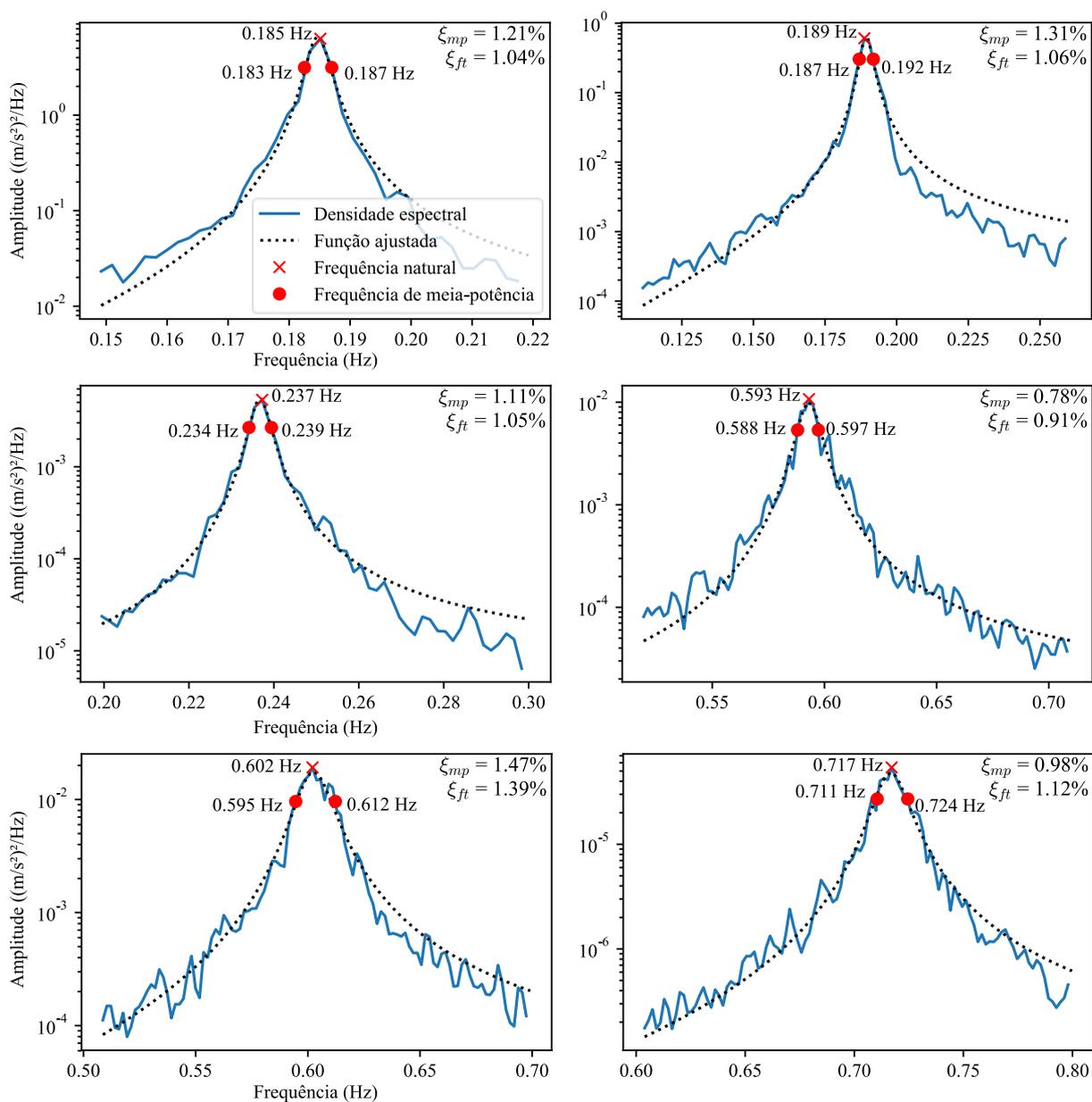
Figura 5.7 – Gráficos dos autoespectros normalizados e a sua média



(fonte: elaborado pelo autor)

Uma vez determinadas as frequências naturais pelos picos dos autoespectros, parte-se para a estimativa das razões de amortecimento por meio dos métodos de meia-potência (ξ_{mp}) e ajuste ao espectro analítico (ξ_{ft}). A Figura 5.8 apresenta os resultados obtidos. As frequências de meia-potência foram calculadas interpolando-se linearmente os valores discretos do autoespectro.

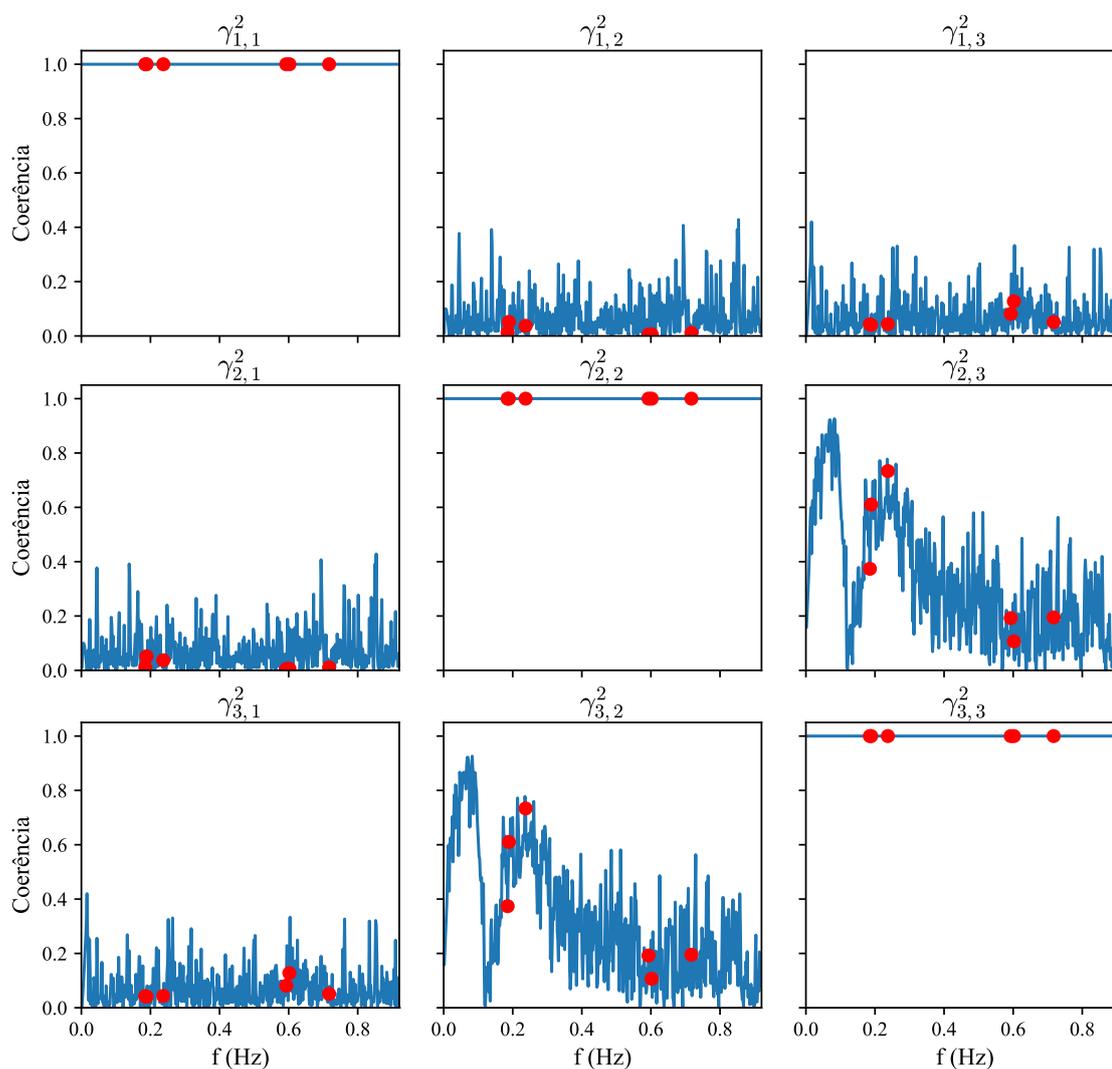
Figura 5.8 – Estimativas das razões de amortecimento



(fonte: elaborado pelo autor)

Os gráficos das funções de coerência são mostrados na Figura 5.9, nos quais indicaram-se com pontos vermelhos os valores assumidos nas frequências naturais estimadas. Verifica-se que os elementos da diagonal possuem valores unitários pois se tratam da função de coerência do sinal consigo próprio. Em relação aos restantes, observa-se que na maioria dos pontos a função assume valor próximo de zero, indicando que não há correlação entre os sinais. Teoricamente, como as formas modais são independentes em cada direção, estas funções de coerência deveriam assumir valores nulos.

Figura 5.9 – Gráficos das funções de coerência



(fonte: elaborado pelo autor)

Um resumo das informações apresentadas nas figuras anteriores e as formas modais estimadas pelo método BFD encontram-se no Quadro 5.1. Optou-se por apresentar os resultados apenas para os três primeiros modos de vibração pois a caracterização precisa dos modos superiores requer mais sensores ao longo da altura.

Verifica-se que a primeira forma modal identificada corresponde à translação da estrutura na direção y e é idêntica à simulada. A segunda forma modal foi identificada com precisão razoável. Devido à proximidade da primeira frequência natural, surge uma componente na direção y ($\varphi_{2,y} = 0,299$) que não existe na estrutura simulada. De forma semelhante, constata-se que a função de coerência entre os sinais 2 e 3 não é nula na terceira frequência de ressonância ($\gamma_{2,3}^2 = 0,733$), tampouco o ângulo de fase é próximo a 0° ou $\pm 180^\circ$, indicando que a respectiva

componente estimada ($\varphi_{3,y} = -1,021$) não é precisa. Conclui-se que não foi possível determinar com precisão a terceira forma modal pelo método BFD.

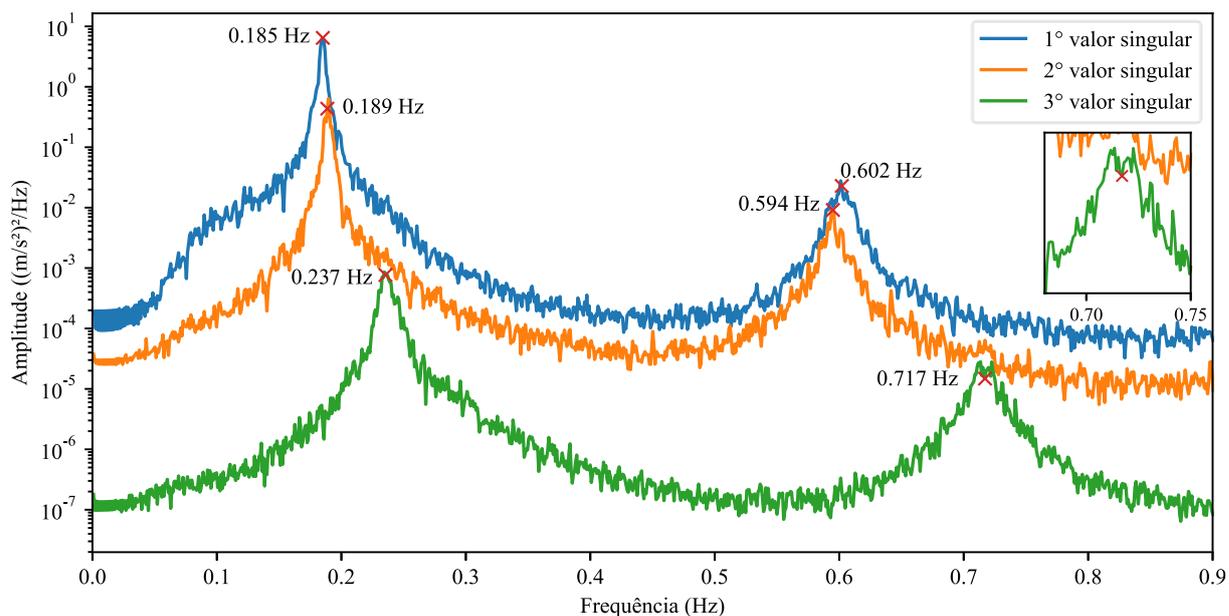
Quadro 5.1 – Propriedades dinâmicas identificadas pelo método BFD

Modo	Componente	Densidade espectral		γ^2	φ_i
		Amplitude	Fase		
1 0,185 Hz	x	0,10	-68°	0,012	0,016
	y	6,3	0°	1	1,000
	θ_z	$3,6 \times 10^{-3}$	-130°	0,374	-0,001
2 0,189 Hz	x	0,61	0°	1	1,000
	y	0,18	74°	0,052	0,299
	θ_z	$4,6 \times 10^{-4}$	-65°	0,040	0,001
3 0,237 Hz	x	$5,1 \times 10^{-4}$	147°	0,043	-0,096
	y	$5,4 \times 10^{-3}$	133°	0,733	-1,021
	θ_z	$5,3 \times 10^{-3}$	0°	1	1,000

(fonte: elaborado pelo autor)

Prossegue-se a identificação do sistema no domínio da frequência, a partir deste ponto por meio do método EFDD. Realizou-se a decomposição em valores singulares da matriz espectral, obtendo-se os espectros dos valores singulares apresentados na Figura 5.10. Representaram-se as frequências naturais escolhidas, as quais encontram-se nos picos dos espectros, com exceção do sexto modo. Neste observam-se dois picos próximos no espectro do terceiro valor singular, optando-se por definir a frequência natural como sendo aquela que está entre os dois picos.

Figura 5.10 – Gráfico dos espectros dos valores singulares



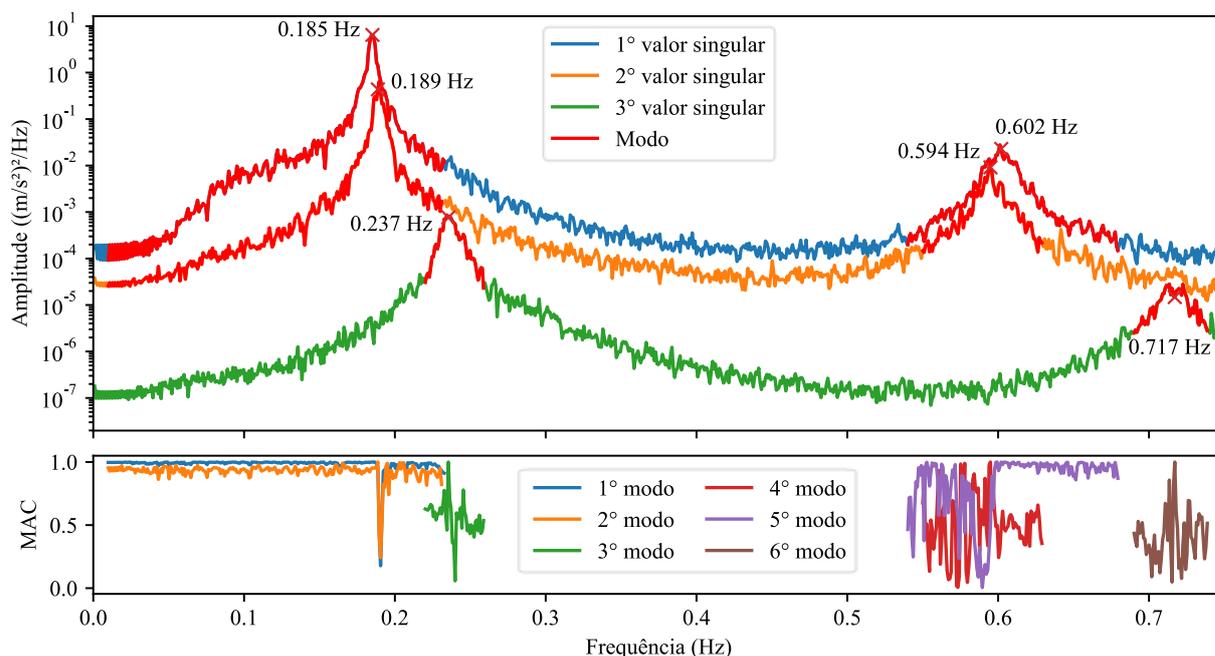
(fonte: elaborado pelo autor)

Em seguida, nas seis frequências naturais identificadas, obtiveram-se os correspondentes vetores singulares, os quais são estimativas das formas modais pelo método EFDD. De posse desses vetores, calcularam-se os MAC entre eles e os vetores singulares na vizinhança dos picos. Esses valores servem como indicador do trecho a ser empregado no cálculo da transformada inversa de Fourier. Os resultados são apresentados na Figura 5.11.

Nota-se que, com exceção dos dois primeiros modos, os valores do MAC na vizinhança dos picos são inferiores a 0,8. Para o terceiro, quarto e sexto modos não foi possível definir um trecho dos espectros para a aplicação da transformada inversa de Fourier e obtenção das funções de autocorrelação. Consequentemente, as razões de amortecimento não puderam ser estimadas.

Ressalta-se que no método EFDD, conforme proposto por Brincker et al. (2001b), obtém-se as frequências naturais e razões de amortecimento a partir da função de autocorrelação. Nos casos em que não foi possível determiná-la, estimaram-se as frequências naturais a partir dos picos dos espectros dos valores singulares.

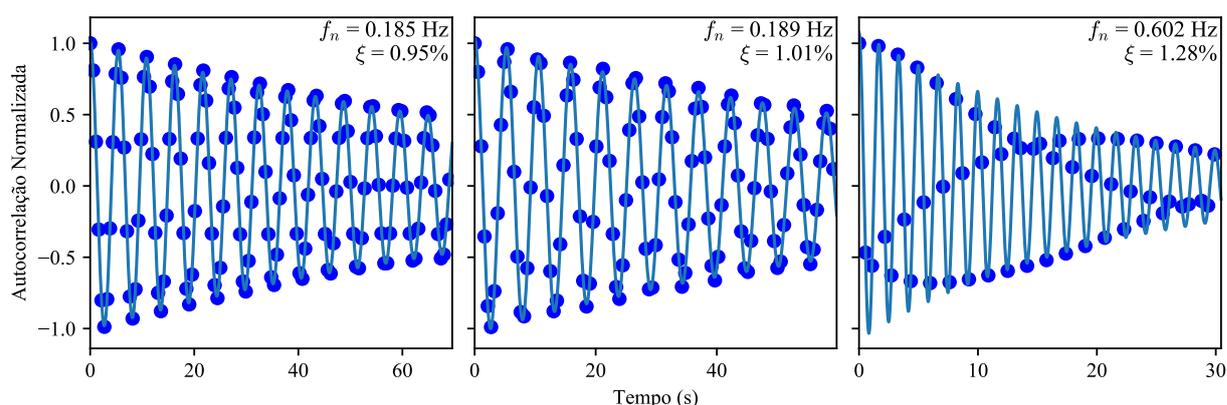
Figura 5.11 – Gráficos dos espectros dos valores singulares e do MAC



(fonte: elaborado pelo autor)

As funções indicadas em vermelho na Figura 5.11 para o primeiro, segundo e quinto modos foram transformadas para o domínio do tempo por meio da transformada inversa de Fourier, obtendo-se as funções de autocorrelação apresentadas na Figura 5.12. As frequências naturais e razões de amortecimento foram estimadas através do ajuste da função teórica aos valores discretos.

Figura 5.12 – Estimativas das frequências naturais e razões de amortecimento pelo método EFDD

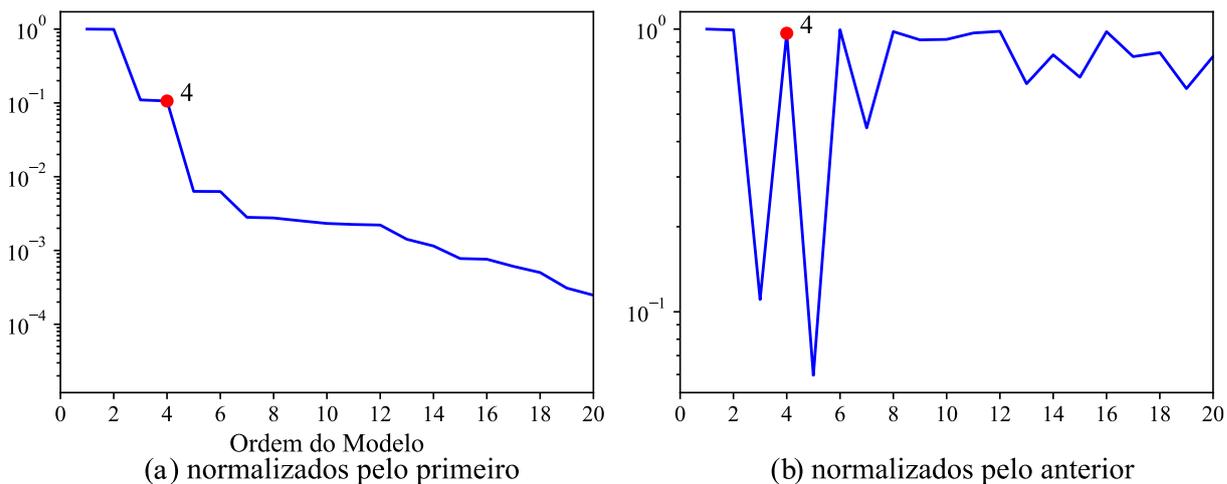


(fonte: elaborado pelo autor)

Parte-se para apresentação dos resultados referentes aos métodos no domínio do tempo, iniciando pela análise dos valores singulares. Conforme pode ser visualizado na Figura 5.2, o

modelo possui seis modos de vibração detectáveis (abaixo de 0,92 Hz), indicando que sua ordem em espaço de estado é 12. Os três métodos analisados (SSI COV, SSI DATA e IV) forneceram gráficos de valores singulares normalizados semelhantes, optando-se por apresentar apenas um deles (Figura 5.13). Graficamente nota-se a presença de quatro valores singulares não nulos.

Figura 5.13 – Gráficos dos valores singulares normalizados

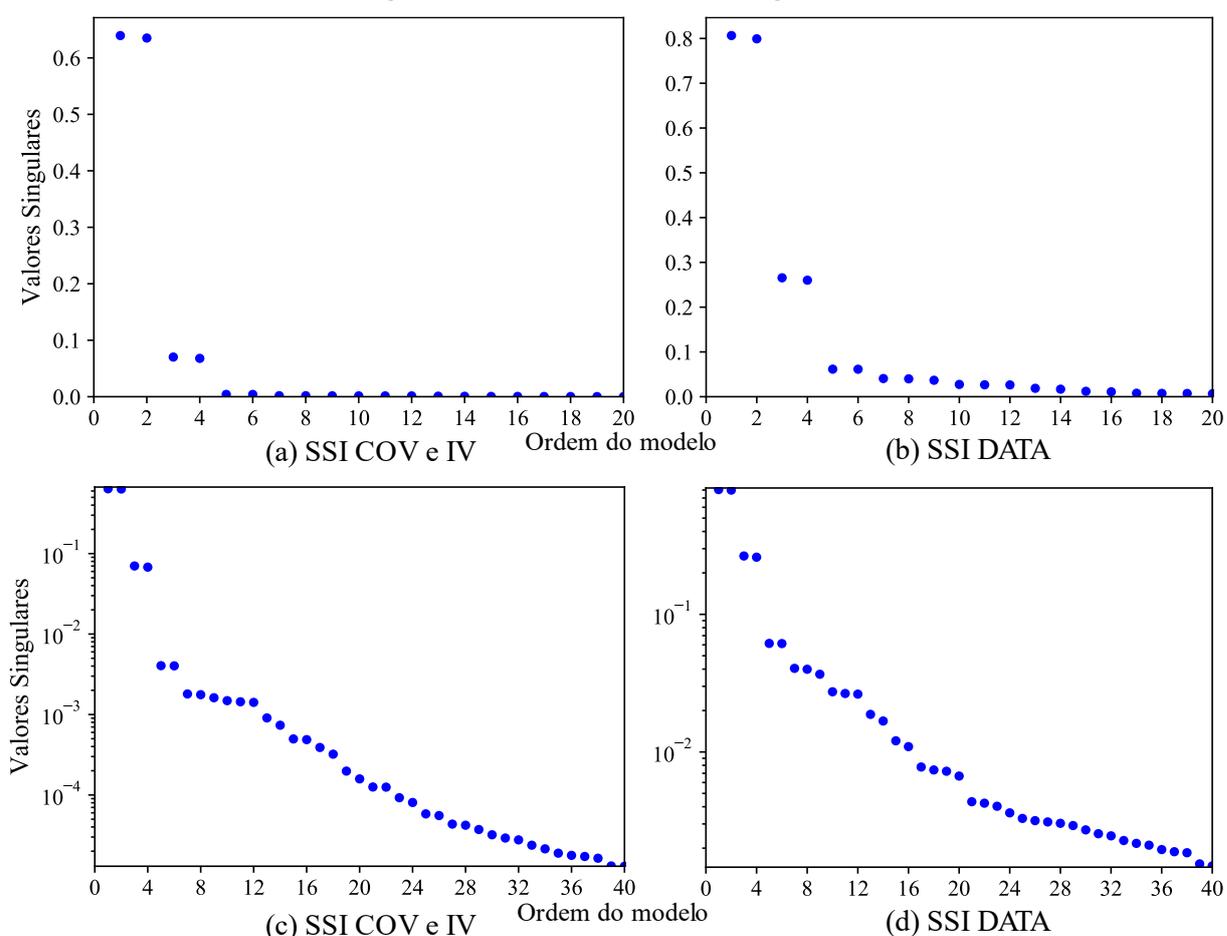


(fonte: elaborado pelo autor)

Referente ao gráfico dos valores singulares em valor absoluto, quando apresentado em escala aritmética (Figura 5.14a e 5.14b), identificam-se quatro valores não nulos. A análise do gráfico em escala logarítmica (Figura 5.14c e 5.14d), entretanto, mostrou-se mais complicada. Para o método SSI DATA, os quatro primeiros valores singulares destacam-se, mas os seguintes apresentam um comportamento semelhante até o vigésimo. Já no diagrama dos métodos SSI COV e IV destacam-se seis valores. Após o décimo segundo, os valores singulares passam a diminuir mais rapidamente, o que poderia indicar que a ordem em espaço de estado é 12. Portanto, na análise dos valores singulares, não foi possível identificar de maneira precisa o número de modos de vibração presentes nas medições.

A Figura 5.15 mostra os diagramas de estabilização. Praticamente todos os modos estáveis (pontos vermelhos) correspondem aos modos da estrutura simulada, os quais possuem frequências de 0,185 Hz, 0,189 Hz, 0,237 Hz, 0,594 Hz, 0,604 Hz e 0,716 Hz. Os métodos forneceram diagramas semelhantes, sendo capazes de detectar separadamente os modos com frequências muito próximas. Verifica-se que o sexto modo, com frequência de vibração igual a 0,716 Hz, só estabilizou com modelos de ordem alta, principalmente no método SSI COV.

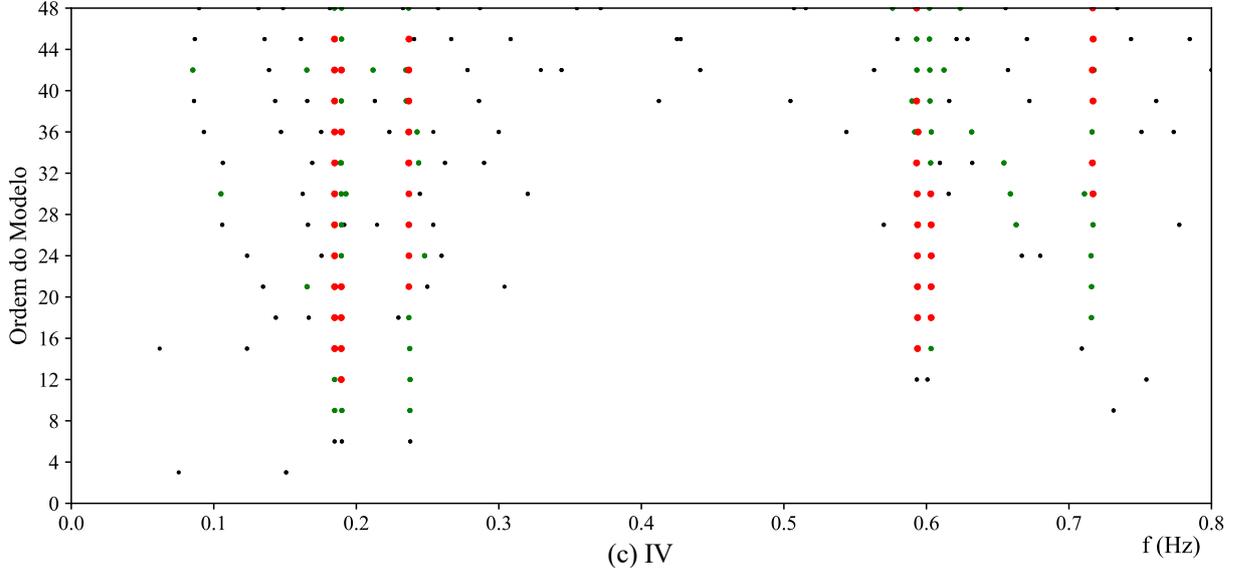
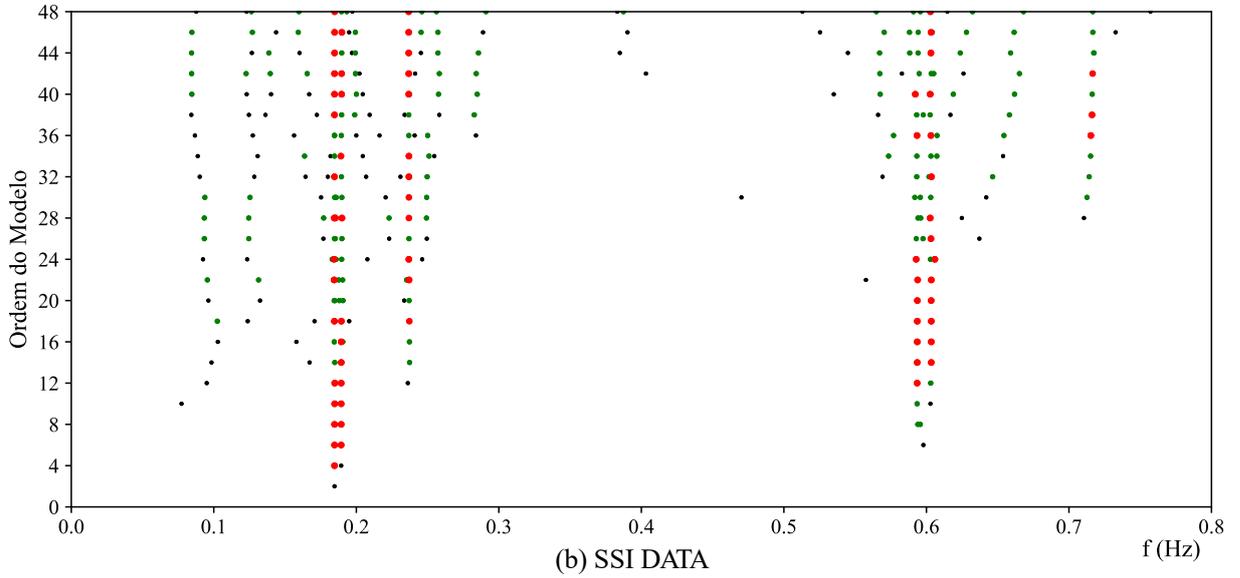
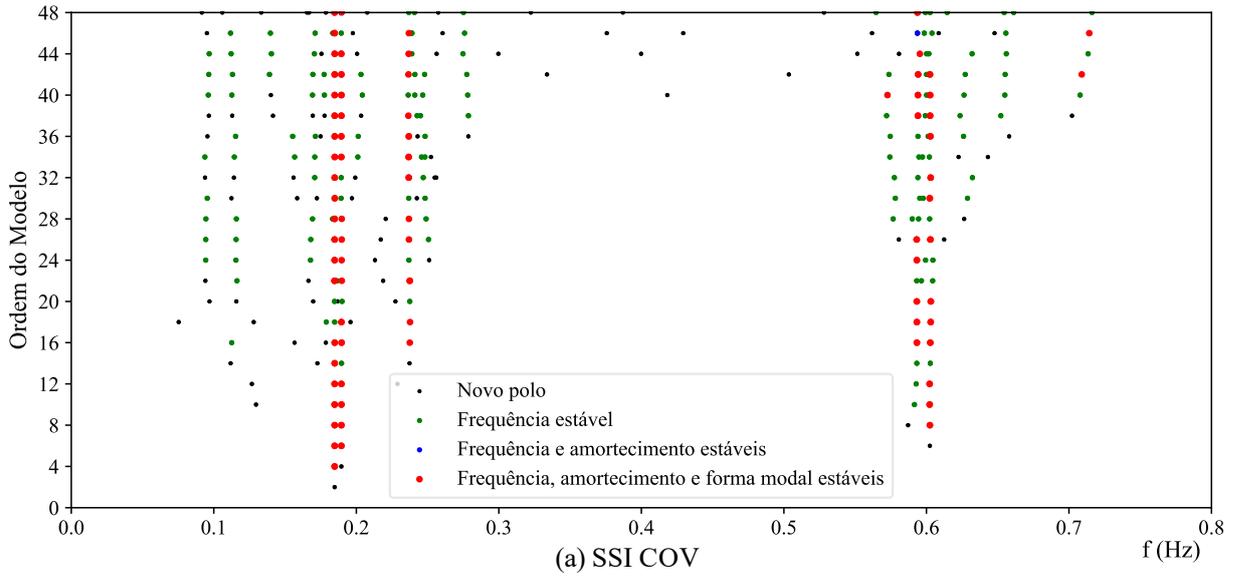
Figura 5.14 – Gráficos dos valores singulares



(fonte: elaborado pelo autor)

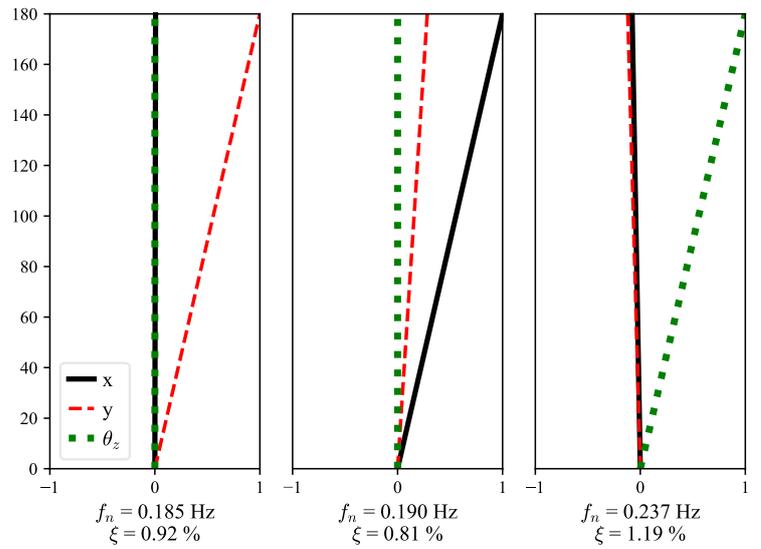
A Figura 5.16 mostra as três primeiras formas modais identificadas em todos os métodos. As formas modais associadas a frequências mais elevadas não são mostradas pois para a sua correta identificação é necessário empregar mais sensores ao longo da altura do edifício. Todos os métodos no domínio do tempo obtiveram formas modais semelhantes para o primeiro e segundo modo de vibração. A primeira forma modal identificada é idêntica à simulada. Na segunda nota-se uma pequena componente de translação em relação ao eixo y que não existe na estrutura simulada. Já na terceira forma modal, correspondente ao modo torcional, é possível visualizar uma diferença entre os métodos no domínio do tempo e no domínio da frequência. Os primeiros foram capazes de identificar com mais precisão o modo simulado.

Figura 5.15 – Diagramas de estabilização

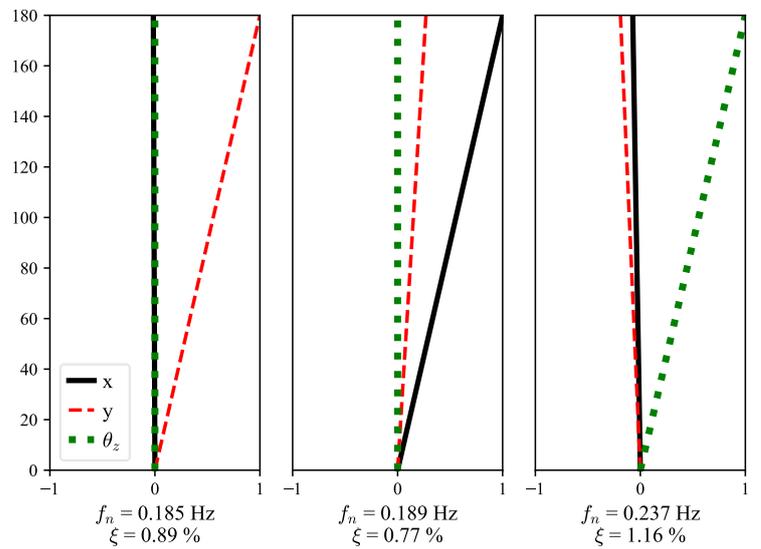


(fonte: elaborado pelo autor)

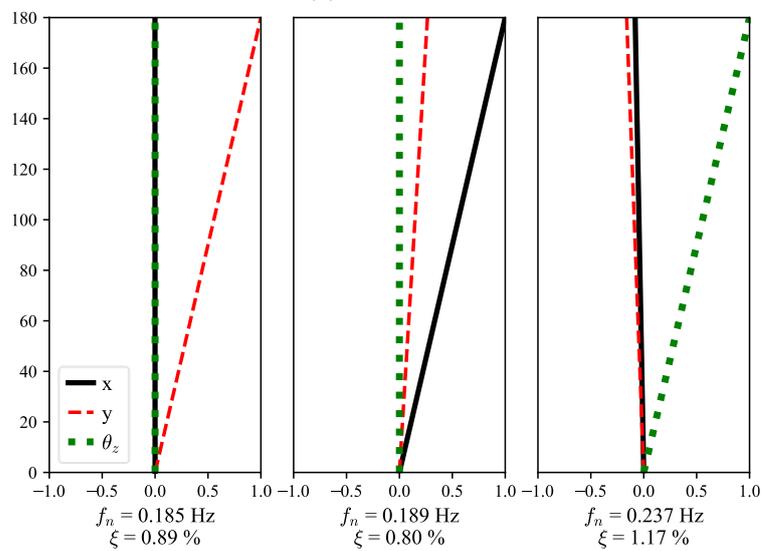
Figura 5.16 – Formas modais identificadas



(a) SSI COV



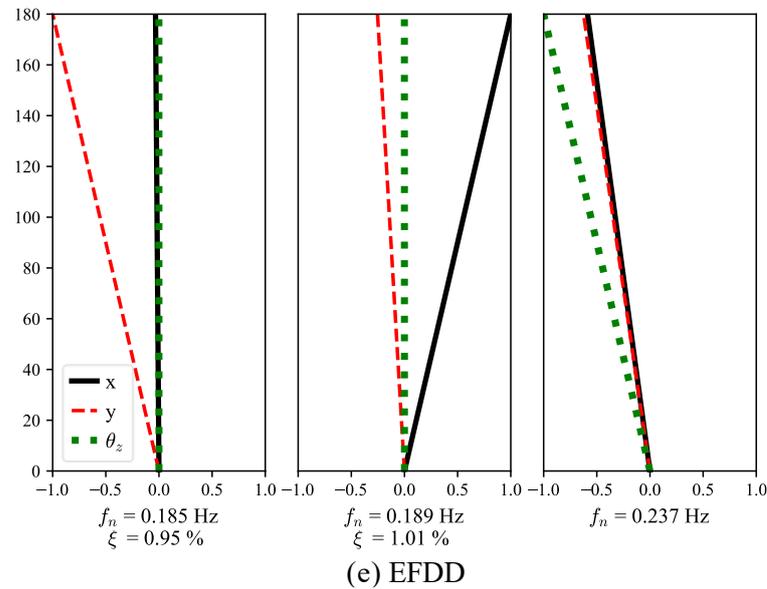
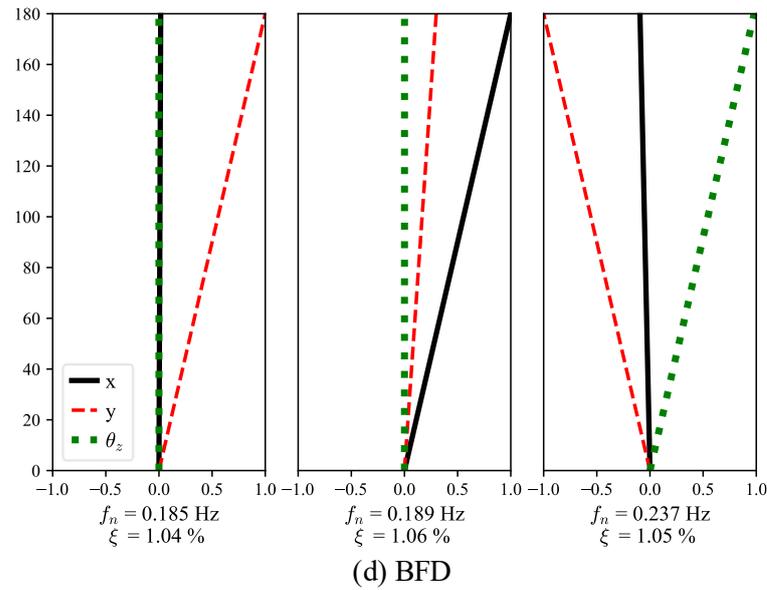
(b) SSI DATA



(c) IV

continua

continuação



(fonte: elaborado pelo autor)

Os Quadros 5.2, 5.3 e 5.4 resumem os resultados obtidos. Para os métodos SSI COV, SSI DATA e IV, as frequências e razões de amortecimento do quadro são os valores médios dos modos estáveis visualizados no diagrama de estabilização. Nota-se que todos os métodos forneceram frequências naturais praticamente iguais às simuladas. Referente às razões de amortecimento, verifica-se que o ajuste ao espectro analítico forneceu, na maioria dos casos, razões de amortecimento inferiores ao método da meia-potência. Os métodos SSI e IV forneceram razões de amortecimento semelhantes.

Quadro 5.2 – Comparativo das frequências

Modo	Frequências naturais (Hz)					
	Simu- lada	BFD	EFDD	IV	SSI COV	SSI DATA
1	0,185	0,185	0,185	0,185	0,185	0,185
2	0,189	0,189	0,189	0,189	0,190	0,189
3	0,237	0,237	0,237	0,237	0,237	0,237
4	0,594	0,593	0,594	0,594	0,594	0,593
5	0,604	0,602	0,602	0,603	0,603	0,603
6	0,716	0,717	0,717	0,717	0,716	0,716

(fonte: elaborado pelo autor)

Quadro 5.3 – Comparativo das razões de amortecimento

Modo	Razões de amortecimento						
	Estru- tural	Meia- potência	Ajuste espectro	EFDD	IV	SSI COV	SSI DATA
1	1,00%	1,21%	1,04%	0,95%	0,89%	0,92%	0,89%
2	1,00%	1,31%	1,06%	1,01%	0,80%	0,81%	0,77%
3	1,00%	1,11%	1,05%	-	1,17%	1,19%	1,16%
4	1,00%	0,78%	0,91%	-	1,00%	0,98%	1,00%
5	1,00%	1,47%	1,39%	1,28%	1,14%	1,19%	1,13%
6	1,00%	0,98%	1,12%	-	1,09%	1,24%	1,02%

(fonte: elaborado pelo autor)

O Quadro 5.4 mostra os MAC calculados a partir das formas modais simuladas e identificadas. O valor unitário indica que a forma modal identificada é igual à simulada. Como empregou-se apenas um sensor, consideraram-se apenas as componentes modais correspondentes ao topo da edificação ($z = 180$ m), sendo as demais alturas ignoradas.

Verifica-se que os métodos no domínio do tempo identificaram as formas modais com excelente precisão (MAC acima de 0,9) e obtiveram resultados semelhantes entre si. Isso significa que eles foram capazes de separar os modos de flexão nas direções x e y e os modos de torção, sem

identificar componentes acopladas. Os métodos no domínio da frequência, por sua vez, identificaram as formas modais com menor precisão, principalmente para os modos de torção (terceiro e sexto modos).

Quadro 5.4 – Comparativo dos MAC

Modo	MAC				
	BFD	EFDD	IV	SSI COV	SSI DATA
1	0,9997	0,9990	0,9999	0,9999	0,9998
2	0,9180	0,9397	0,9342	0,9260	0,9321
3	0,4876	0,5763	0,9683	0,9800	0,9605
4	0,9964	0,4733	0,9999	0,9986	0,9997
5	0,9985	0,9996	0,9995	0,9999	0,9986
6	0,7076	0,4042	0,9974	0,8149	0,9969

(fonte: elaborado pelo autor)

Considerando todas as propriedades dinâmicas identificadas, constata-se que os métodos no domínio do tempo forneceram resultados excelentes e semelhantes entre si. O método BFD forneceu bons resultados, com exceção das formas modais torcionais. Por outro lado, não foi possível aplicar o método EFDD na identificação de todos os modos de vibração. Os valores do MAC na vizinhança dos picos mostraram-se baixos, não permitindo a determinação da função de autocorrelação.

Neste cenário inicial foi possível verificar a robustez dos métodos frente a um evento extremo. Espera-se que a presença de ruído, menor tempo de aquisição e a ação de uma menor velocidade do vento dificultem o processo de identificação. Esta situação, portanto, serve como referência para as demais.

5.1.4.2 Cenário 2: influência do tempo de aquisição

O segundo cenário visa analisar a capacidade dos métodos na identificação da estrutura a partir de registros com menores durações. Manteve-se a velocidade do vento no topo do edifício igual a 40 m/s empregada no exemplo anterior e consideraram-se tempos de aquisição de 278 s (4,6 min), 556 s (9,3 min), 1113 s (18,5 min), 2225 s (37,1 min) e 4450 s (74,2 min).

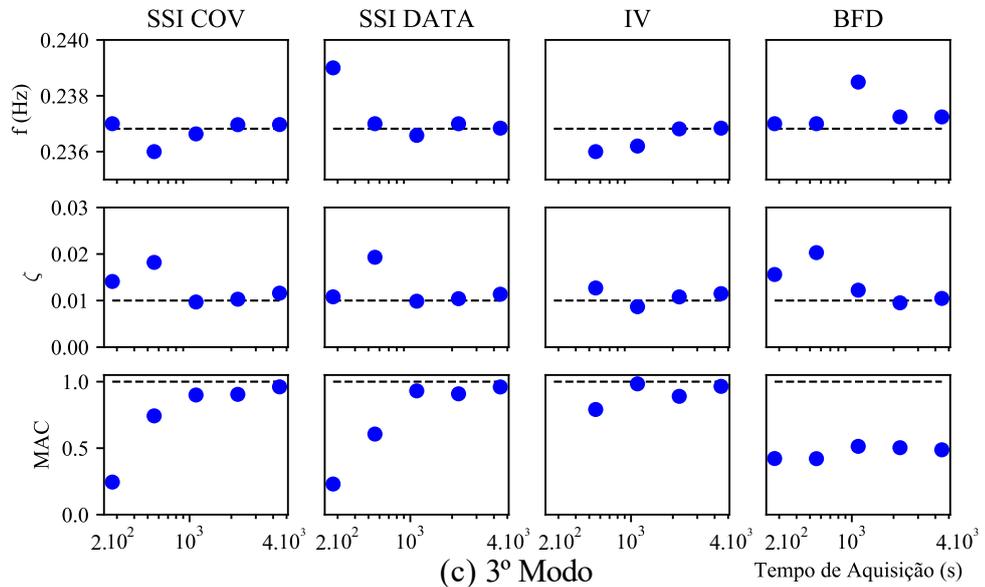
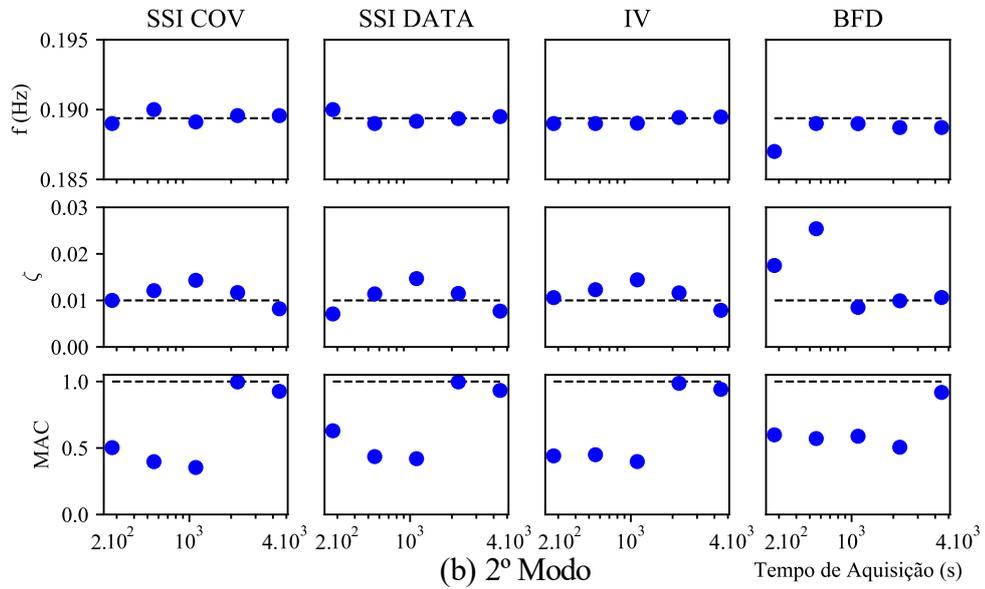
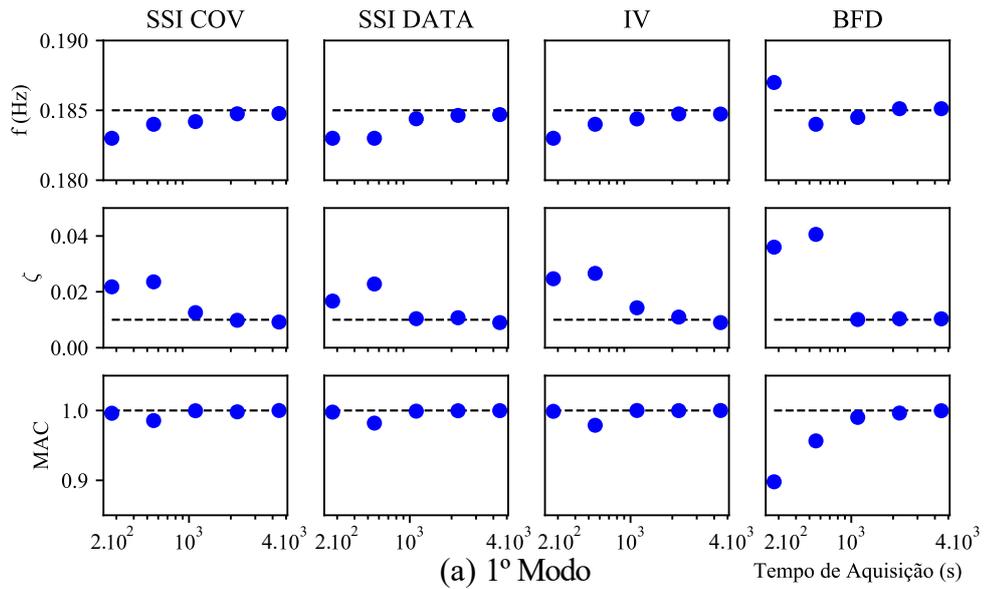
Os resultados são apresentados na Figura 5.17 para os seis modos de vibração. No eixo horizontal representa-se o tempo de aquisição e no eixo vertical a propriedade dinâmica. As linhas horizontais tracejadas correspondem às propriedades da estrutura simulada e os pontos azuis às propriedades identificadas. Nos casos em que o modo não é identificado, o ponto é omitido.

Os métodos no domínio do tempo forneceram resultados muito próximos para os três primeiros modos. Para os demais modos, o método SSI DATA mostrou-se ligeiramente melhor, sendo capaz de identificar o quinto e sexto modos para registros mais curtos do que os métodos IV e SSI COV respectivamente.

O método BFD foi capaz de detectar todos os modos de vibração mesmo para os registros com duração de apenas 4,6 minutos. Isso ocorreu porque os picos dos autoespectros normalizados mostraram-se claros e em frequências distintas. No entanto, as razões de amortecimento e formas modais identificadas mostraram-se menos precisas do que as estimadas pelos demais métodos.

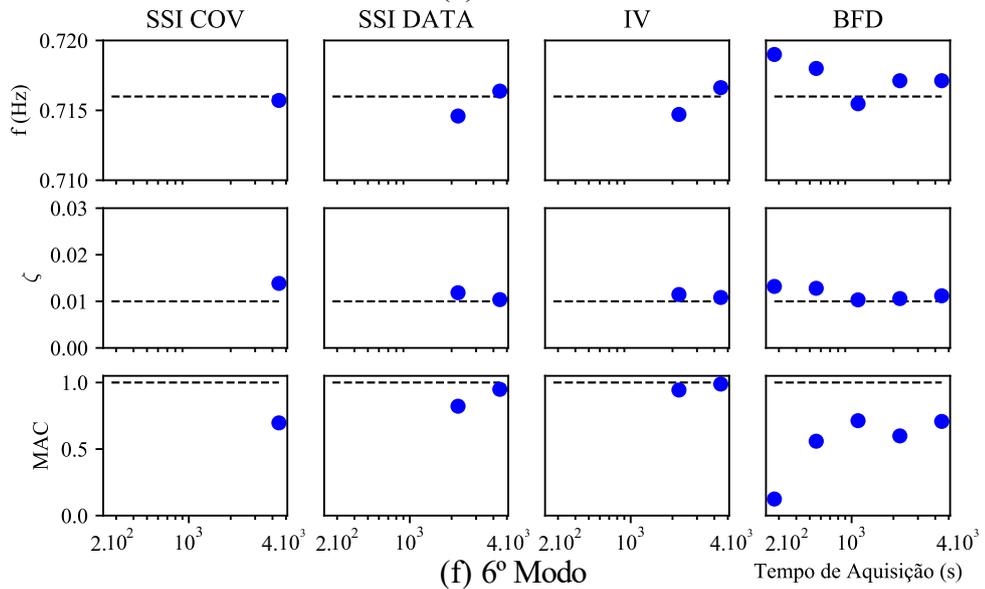
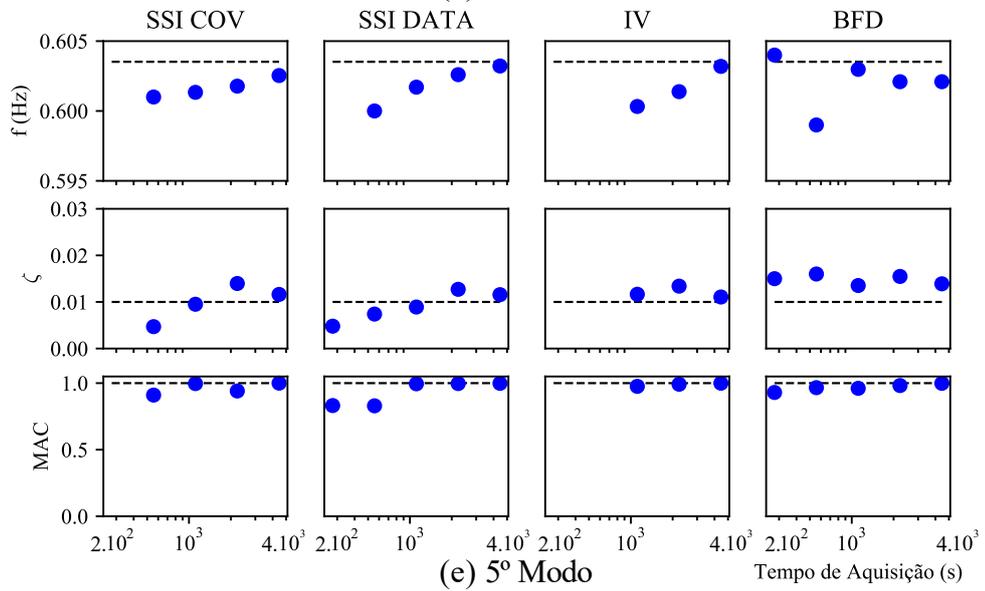
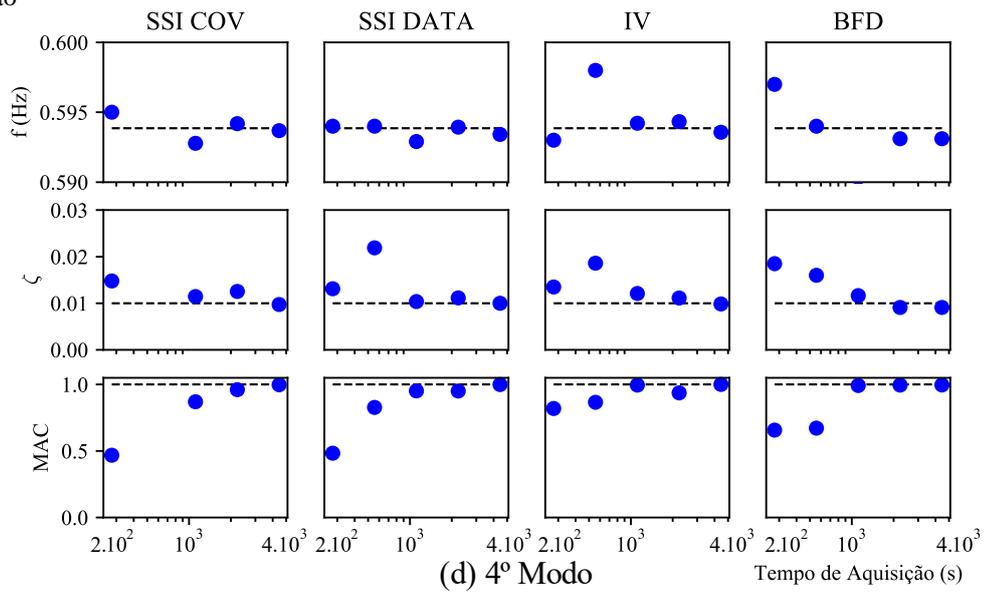
Com base na presente simulação, é possível obter uma estimativa do tempo de aquisição necessário para a identificação da estrutura. De uma forma geral, registros com duração de 18,5 minutos mostraram-se adequados para a identificação com boa precisão, o que corresponde a cerca de 200 ciclos para a frequência fundamental. As exceções são a segunda forma modal e o sexto modo de vibração, os quais foram identificados com precisão apenas para registros com pelo menos 37 minutos.

Figura 5.17 – Propriedades dinâmicas identificadas



continua

continuação



(fonte: elaborado pelo autor)

5.1.4.3 Cenário 3: ruído absoluto igual a 2,5 mG

No terceiro cenário estuda-se a capacidade dos métodos de identificar a estrutura na presença de ruídos nas medições. Foi considerado um ruído branco gaussiano com média zero e desvio-padrão igual 0,025, simulado por meio da função `numpy.random.randn` do Python. Para as acelerações lineares em x e y a sua unidade é m/s^2 , sendo igual a 2,5 mG, e para a aceleração angular sua unidade é rad/s^2 .

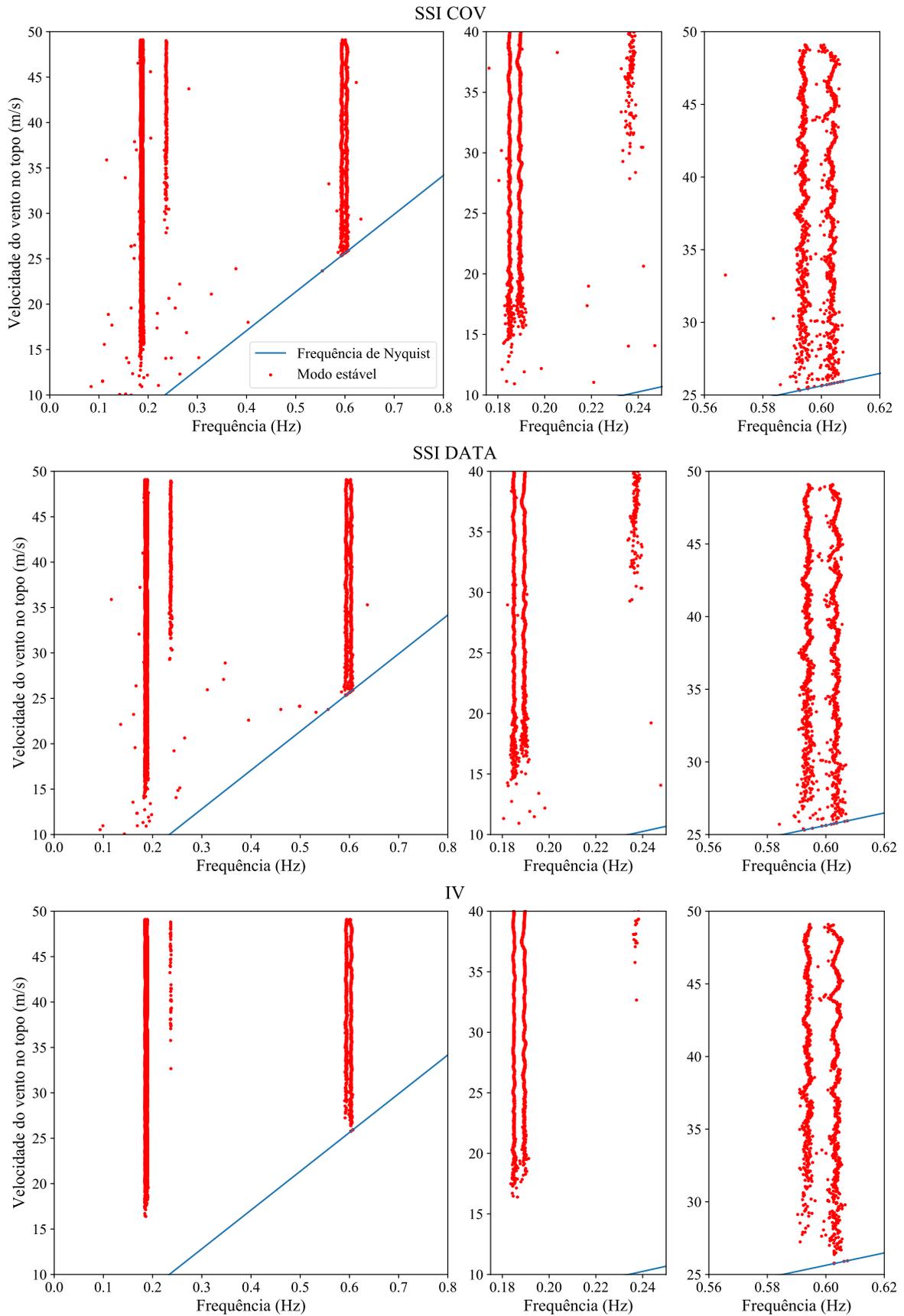
Um parâmetro importante na identificação de edifícios é a amplitude da resposta a ser medida. Com exceção de eventos extremos, ela é baixíssima. Isso indica a necessidade de sensores com alta sensibilidade e o emprego de métodos capazes de identificar as propriedades dinâmicas a partir de registros com relação ruído/sinal elevadas. Esta última consideração é estudada neste terceiro cenário e será aprofundada no quarto.

Realizaram-se análises para várias velocidades do vento no topo, iniciando com 10 m/s até cerca de 50 m/s. Totalizaram-se mil simulações em velocidades diferentes, com uma variação de velocidade entre elas igual a $(50-10)/1000 = 0,04$ m/s. Optou-se por empregar os métodos SSI COV, SSI DATA e IV.

Dada a enorme quantidade de diagramas de estabilização gerados, desenvolveu-se uma rotina para identificação de modos estáveis de maneira automatizada chamada de `stable_modes`. Assim, os diagramas de estabilização e as formas modais não foram analisados manualmente. O critério para representação de uma frequência nos gráficos é aquela que apresenta pelo menos quatro modos estáveis para modelos de ordens diferentes, ou seja, quatro polos estáveis (pontos vermelhos) no diagrama de estabilização para uma mesma frequência. Isso indica que os pontos representados nos resultados se referem a, no mínimo, quatro modelos de ordens diferentes que possuem frequência, amortecimento e forma modal estáveis.

A Figura 5.18 apresenta o gráfico das frequências identificadas em função da velocidade do vento no topo do edifício. A frequência de Nyquist está representada em azul. Ela mostrou-se uma limitação para a detecção do quarto e quinto modo. A partir das análises realizadas, não é possível concluir se estas frequências naturais poderiam ser identificadas com velocidades menores do vento.

Figura 5.18 – Frequências detectadas no Cenário 3



(fonte: elaborado pelo autor)

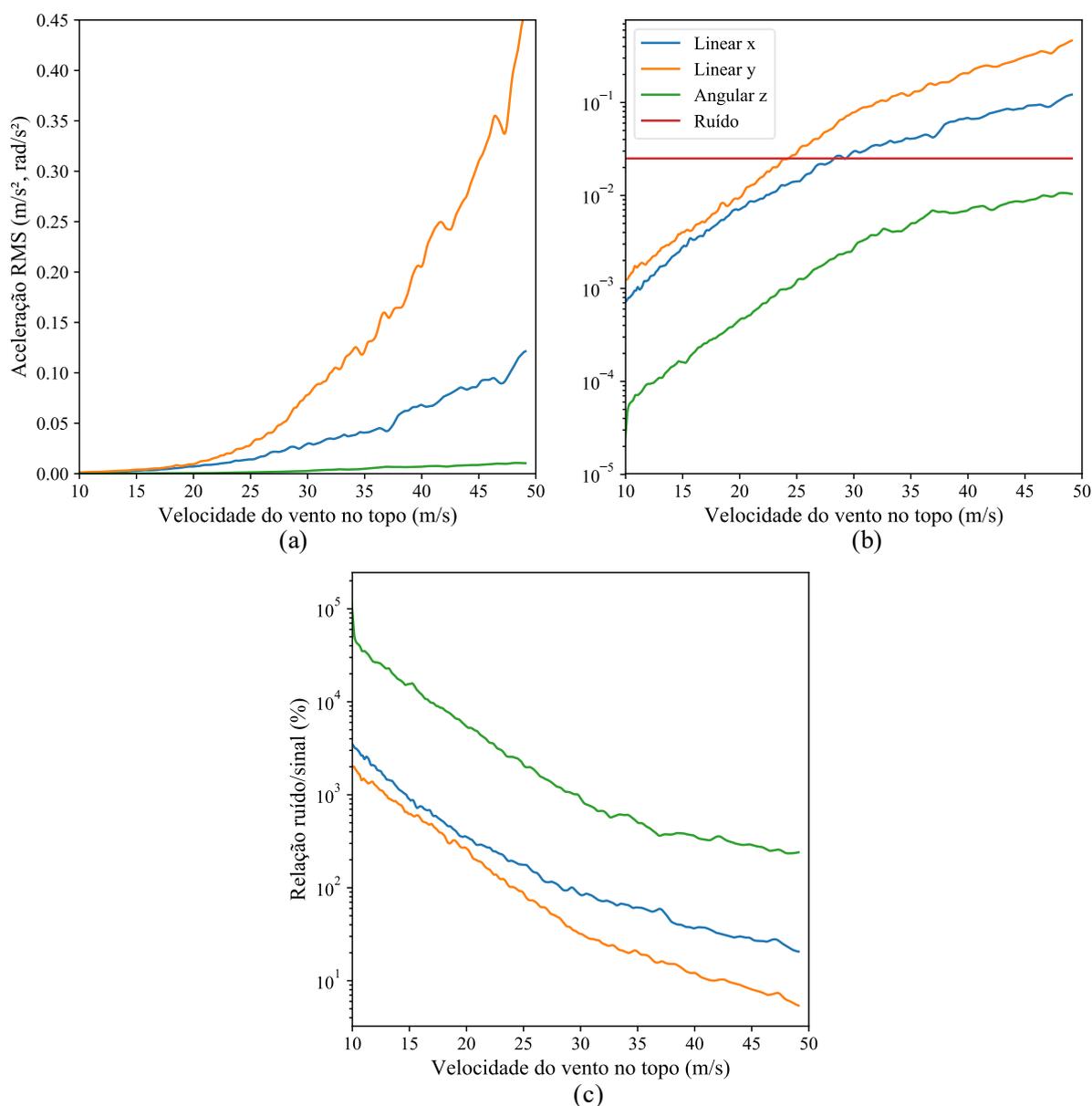
Os métodos SSI DATA e SSI COV forneceram resultados semelhantes. Os dois primeiros modos surgem com velocidade do vento igual a 15 m/s e são claramente distinguíveis a partir de 20 m/s. O terceiro modo aparece para velocidades do vento a partir de 30 m/s. São detectadas possíveis frequências associadas ao quarto e quinto modos para velocidades acima de 30 m/s.

Nota-se uma clara diferença entre os métodos SSI e o IV. Neste último não há frequências espúrias detectadas, ou seja, todos os modos estáveis detectados representam os verdadeiros. Os modos são detectados para velocidades do vento um pouco mais elevadas que os métodos SSI, principalmente para o terceiro modo (torção), no qual mesmo para velocidades altas do vento observam-se lacunas, indicando que ele não foi detectado em todas as simulações. Por fim, verifica-se que o sexto modo (torção), com frequência de 0,716 Hz, não foi detectado.

As Figuras 5.19a e 5.19b mostram curvas dos valores rms das acelerações em função das velocidades do vento e o valor rms do ruído (constante e igual a 0,025). Nota-se que a aceleração angular é uma ordem de magnitude inferior do que as acelerações lineares. A aceleração em y (resposta transversal) é a maior, estando associada ao fenômeno de desprendimento de vórtices.

Na análise de sinais, é comum mensurar o ruído pela relação ruído/sinal, a qual encontra-se na Figura 5.19c. Para a velocidade de 20 m/s, na qual os dois primeiros modos foram detectados, a razão é de aproximadamente 300%. Já para o terceiro modo, que aparece para velocidades de 30 m/s, a relação é em torno de 1000%. Essas razões elevadas indicam que o processo de identificação pode ser feito mesmo para amplitudes da resposta baixas, o que normalmente ocorre no caso de edifícios.

Figura 5.19 – Gráfico da aceleração RMS vs velocidade do vento e relação ruído/sinal



(fonte: elaborado pelo autor)

No terceiro cenário foi possível verificar a capacidade dos métodos de lidar com ruídos nas medições de valor rms igual a 0,025. Notou-se que o método IV fornece uma menor quantidade de modos estáveis, mas aqueles identificados como estáveis correspondem exatamente aos verdadeiros. Os métodos SSI, por sua vez, foram capazes de detectar modos simulados para velocidades do vento mais baixas e, portanto, para uma maior relação ruído/sinal. Entretanto, estes últimos métodos classificam uma maior quantidade de modos espúrios como estáveis. Por fim, salienta-se a dificuldade de identificar os modos de torção a partir das acelerações angulares, visto que resultaram uma ordem de grandeza inferior às acelerações lineares.

5.1.4.4 Cenário 4: várias relações ruído/sinal

O quarto cenário estudado visa avaliar o comportamento dos métodos para diversas razões ruído/sinal. Definiu-se uma velocidade do vento no topo fixa para todas análises e igual a 25 m/s, a qual corresponde a uma probabilidade de ser excedida de 63% e a um período de retorno de aproximadamente um ano no município de Porto Alegre.

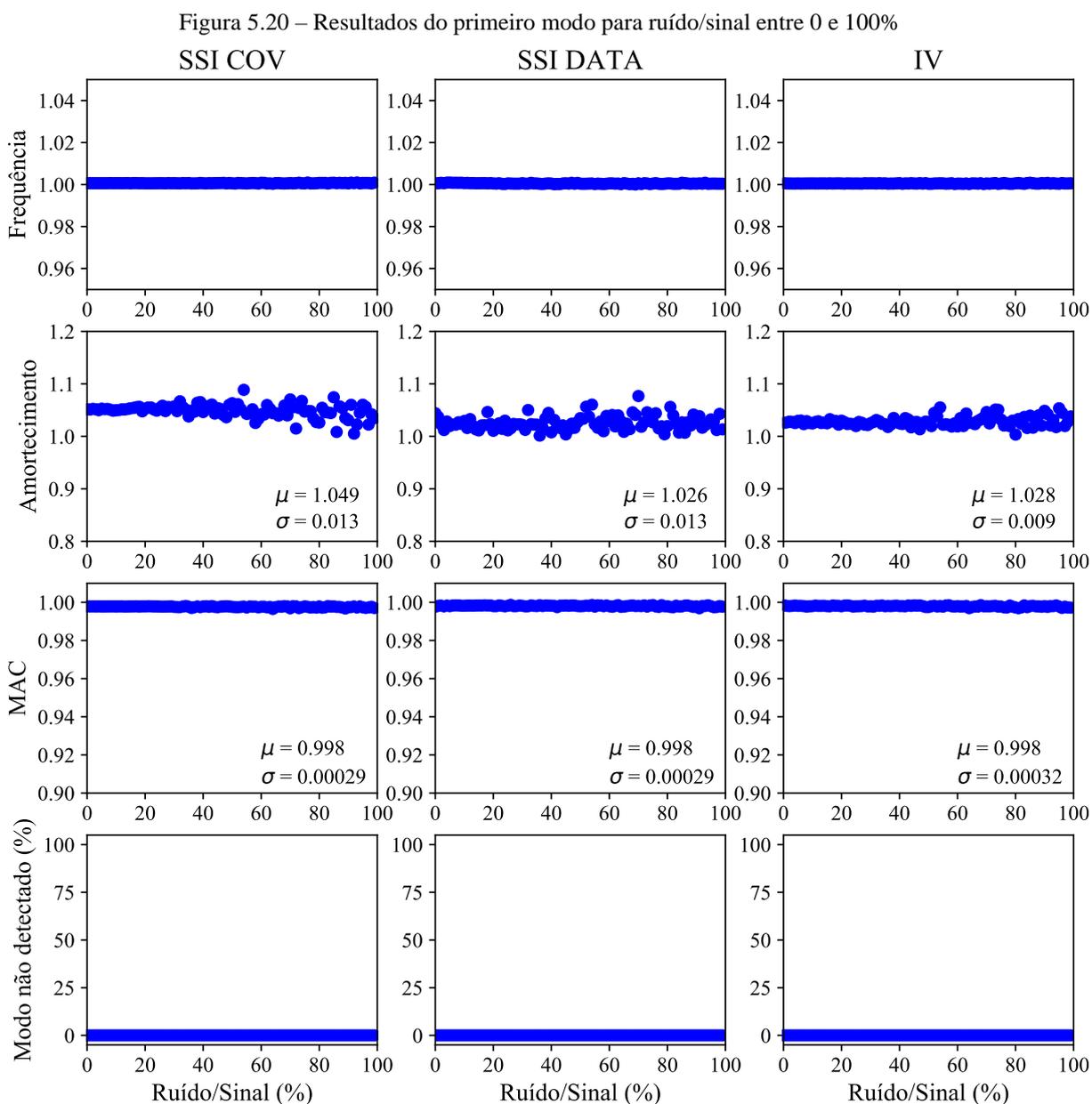
De forma análoga ao exemplo anterior, considerou-se ruído branco gaussiano com média zero, simulado por meio da função `numpy.random.randn`. O seu valor rms foi calculado a partir de uma fração do valor rms de cada aceleração individualmente. Desse modo, sinais com menor valor rms terão ruídos com menor valor rms.

Inicialmente estudaram-se razões ruído/sinal de 0% até 99%, variando de 1 em 1%. Realizaram-se dez simulações para cada uma dessas porcentagens, totalizando mil simulações. Destas dez, calcularam-se as médias das frequências, das razões de amortecimento e do MAC. Representaram-se estes valores médios nos gráficos mostrados a seguir. Caso a frequência detectada possuísse uma diferença superior a 1% da verdadeira, ela é omitida do cálculo da média e representada como modo não detectado.

Elaborou-se uma rotina para automatizar a identificação dos modos estáveis próximos às frequências verdadeiras. Empregou-se o mesmo critério do Cenário 3: um modo é detectado caso apresente no mínimo quatro polos estáveis, de modelos de ordens diferentes, para uma mesma frequência, ou seja, no mínimo quatro pontos vermelhos no diagrama de estabilização. Devido à grande quantidade de informação gerada, os diagramas e as formas modais não foram visualizados. Possíveis modos estáveis com frequências afastadas das verdadeiras não foram analisados.

Os resultados são apresentados na forma gráfica. As frequências detectadas foram divididas pelas verdadeiras. Portanto, o valor 1 para frequência indica que ela é igual a frequência simulada. O mesmo raciocínio foi empregado para as razões de amortecimento. Nas formas modais calculou-se o MAC entre a forma modal identificada e a forma modal verdadeira. Neste caso, MAC igual a 1 significa que as formas modais identificada e verdadeira são idênticas. Por fim, desenhou-se o gráfico da porcentagem de simulações nas quais os modos não foram detectados.

A Figura 5.20 apresenta os resultados para o primeiro modo. Todos os métodos foram capazes de identificá-lo para relações ruído/sinal de até 100% em todas as 1000 simulações, conforme pode ser verificado nos gráficos inferiores. Referente às frequências e ao MAC, não é possível visualizar alterações na faixa de ruído estudada. Já para as razões de amortecimento, nota-se um aumento da dispersão com aumento do ruído.

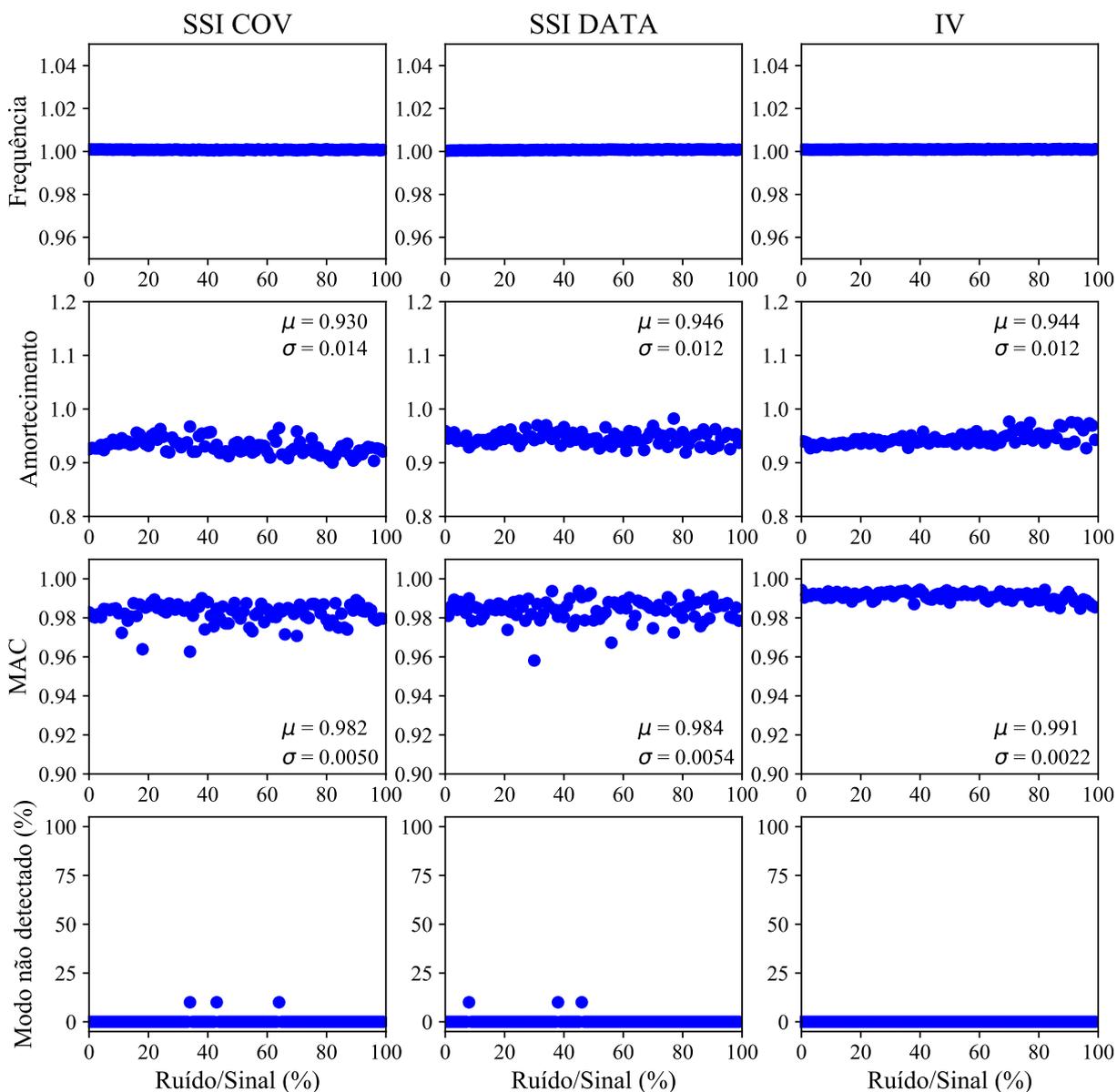


(fonte: elaborado pelo autor)

Procede-se à apresentação dos resultados para o segundo modo (Figura 5.21). Os métodos foram capazes de identificá-lo para relações ruído/sinal de até 100% em praticamente todas as 1000 simulações, com exceção de três simulações nos métodos SSI. As frequências obtidas

mostraram-se precisas na faixa de ruído estudada. Para as razões de amortecimento, o aumento da dispersão com o aumento do ruído mostra-se menos visível. O MAC apresenta maior dispersão em relação ao primeiro modo, verificando que o método IV fornece formas modais ligeiramente mais precisas e com menor dispersão. Para os dois primeiros modos, não é possível identificar nenhuma tendência de deterioração dos resultados na faixa de ruído estudada.

Figura 5.21 – Resultados do segundo modo para ruído/sinal entre 0 e 100%

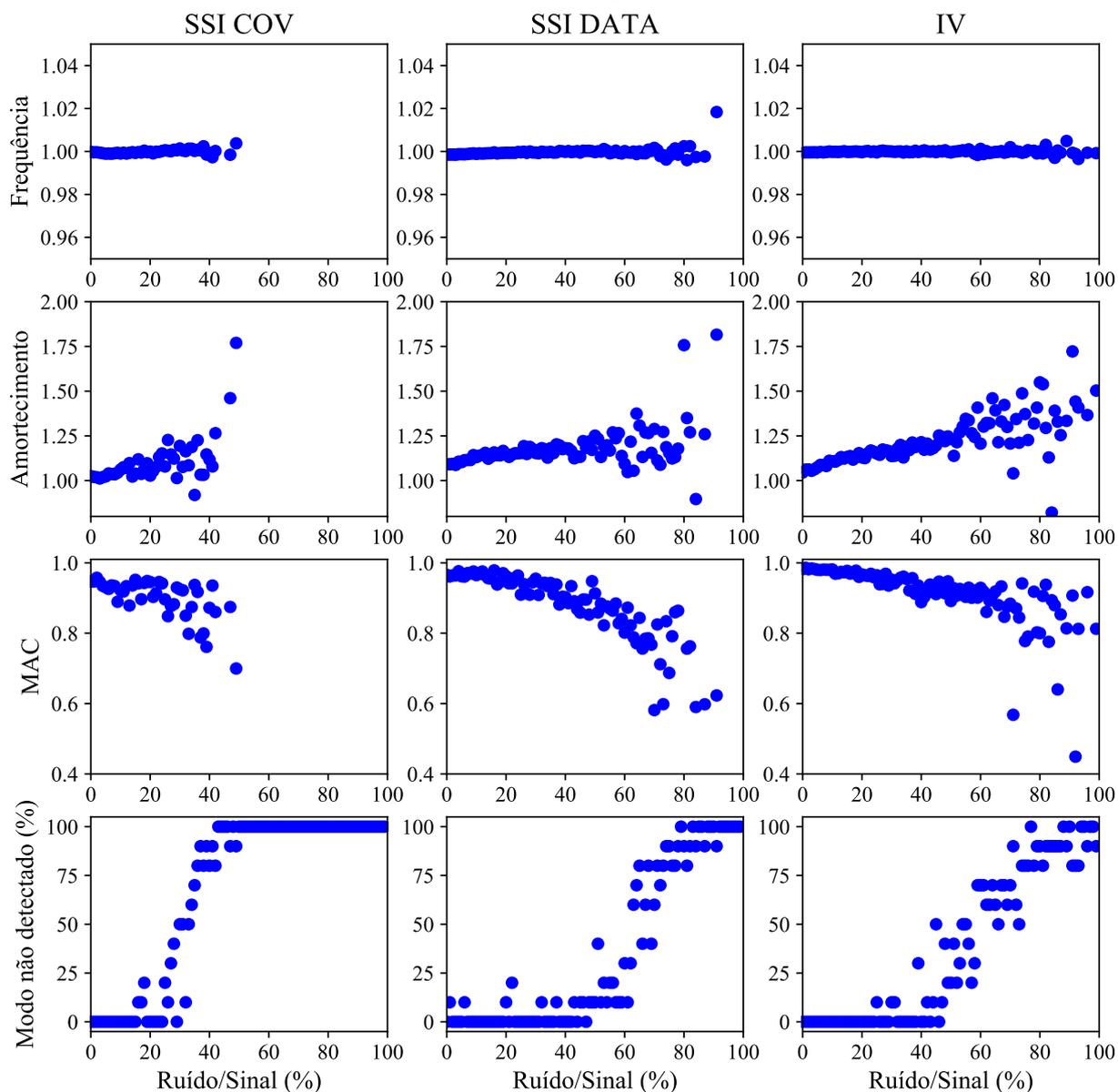


(fonte: elaborado pelo autor)

A Figura 5.22 apresenta os resultados para o terceiro modo. Não foi possível detectar com precisão este modo em todas as simulações. O método SSI COV foi capaz de identificá-lo para relação ruído/sinal de até 30%, o SSI DATA para 50% e o IV para 50%. Verifica-se uma

tendência de aumento do amortecimento com o aumento do ruído. Nas porcentagens citadas, observa-se uma queda abrupta do MAC, a qual mostrou-se menos brusca no método IV. Este mostrou-se capaz de identificar o terceiro modo em uma maior quantidade de simulações. Por outro lado, o SSI COV não foi capaz de detectar praticamente nenhuma vez o modo com relação ruído/sinal superior a 50%.

Figura 5.22 – Resultados do terceiro modo para ruído/sinal entre 0 e 100%



(fonte: elaborado pelo autor)

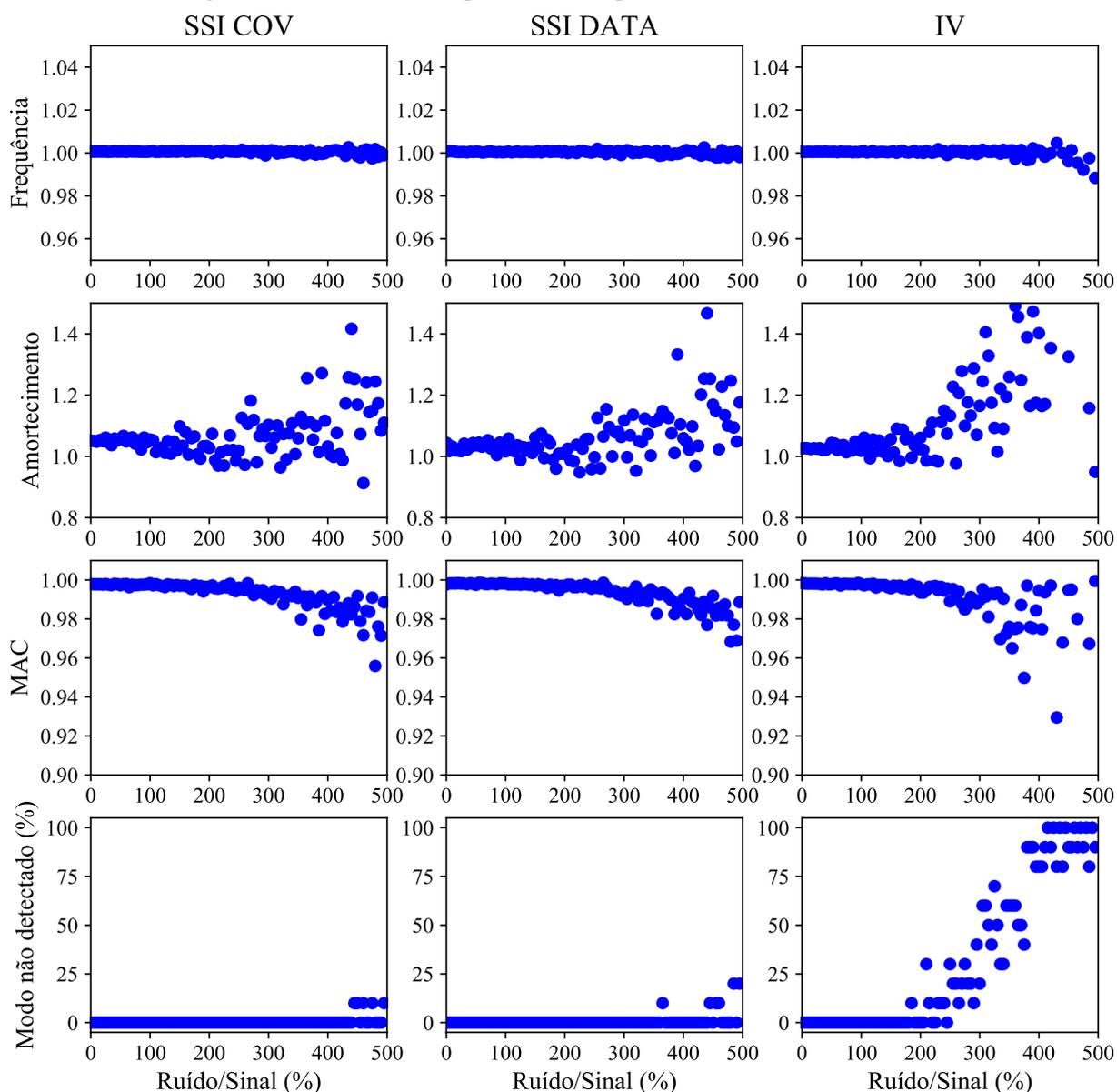
O fato do terceiro modo não ser identificado no intervalo estudado pode estar associado à amplitude da aceleração angular: ela é uma ordem de magnitude inferior em relação às acelerações lineares. Para a relação ruído/sinal no intervalo de 0% e 100%, todos os métodos

forneceram resultados semelhantes na identificação dos dois primeiros modos. Referente ao terceiro modo, constatou-se que o método IV forneceu resultados mais próximos aos verdadeiros, sendo capaz de identificá-lo em quase todo o intervalo estudado.

Dado os bons resultados na identificação dos dois primeiros modos para ruídos entre 0 e 100%, expandiu-se o intervalo em estudo para relações de até 495%. Neste caso, realizaram-se dez simulações para cada 5% de incremento do ruído, totalizando mil simulações. Essas relações ruído/sinal são elevadíssimas e não são usuais na análise de sinais. Entretanto, como a amplitude da resposta em edifícios é baixa, o estudo da capacidade dos métodos em realizar a identificação a partir de sinais com ruído elevado torna-se importante.

A Figura 5.23 mostra os gráficos do primeiro modo. Os métodos SSI mostraram-se capazes de identificá-lo em todo o intervalo estudado, sendo que no SSI COV o modo não foi identificado em apenas cinco simulações e no SSI DATA oito. A razão ruído/sinal igual a 250% foi limitante para o IV. Referente ao amortecimento, nota-se uma tendência de aumento da dispersão com aumento do ruído. No intervalo analisado, os métodos SSI forneceram ótimas estimativas para formas modais.

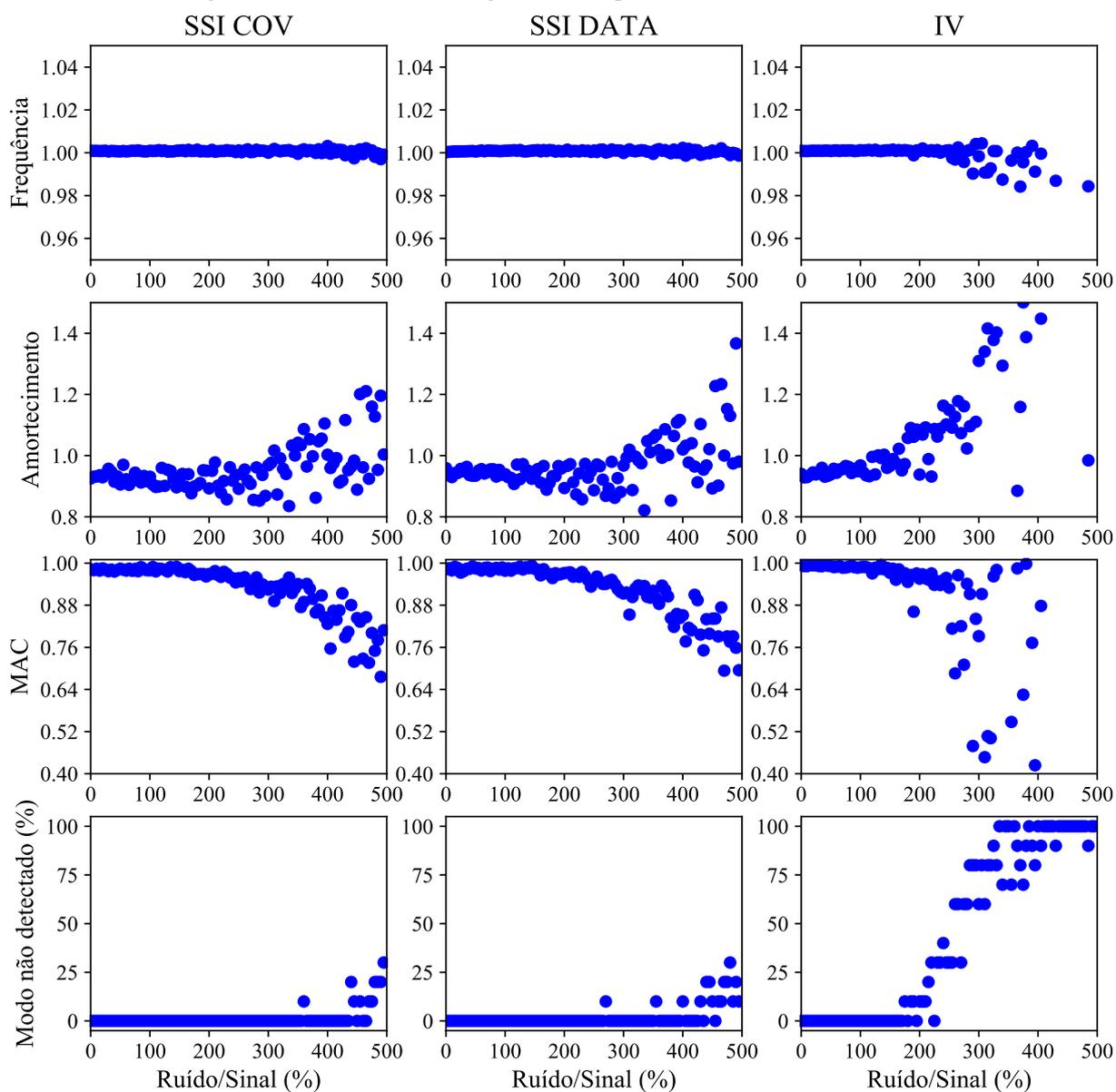
Figura 5.23 – Resultados do primeiro modo para ruído/sinal entre 0 e 500%



(fonte: elaborado pelo autor)

Prossegue-se para a análise dos resultados do segundo modo, os quais são apresentados na Figura 5.24. Novamente os métodos SSI foram capazes de detectá-lo em praticamente todo o intervalo analisado. As exceções foram 16 simulações para o SSI COV e 22 para o SSI DATA. O IV detectou o modo para relação ruído/sinal de até 200%. Nesta análise, nota-se a tendência de aumento do amortecimento com o aumento do ruído. Os métodos conseguiram identificar a forma modal com precisão inferior quando comparado ao primeiro modo.

Figura 5.24 – Resultados do segundo modo para ruído/sinal entre 0 e 500%



(fonte: elaborado pelo autor)

A Figura 5.25 mostra os diagramas de estabilização para a razão ruído/sinal igual a 400%. Os métodos SSI DATA e SSI COV forneceram gráficos semelhantes, sendo capazes de identificar os dois primeiros modos de maneira estável (pontos vermelhos). Notam-se muitas frequências estáveis (pontos verdes), as quais não correspondem a modos verdadeiros.

O método IV não identificou modos de maneira estável. Uma distinção importante entre este método e os métodos SSI, é que nele apenas as frequências verdadeiras estabilizaram para modelos de ordens diferentes, ou seja, as colunas de pontos verdes no diagrama do IV

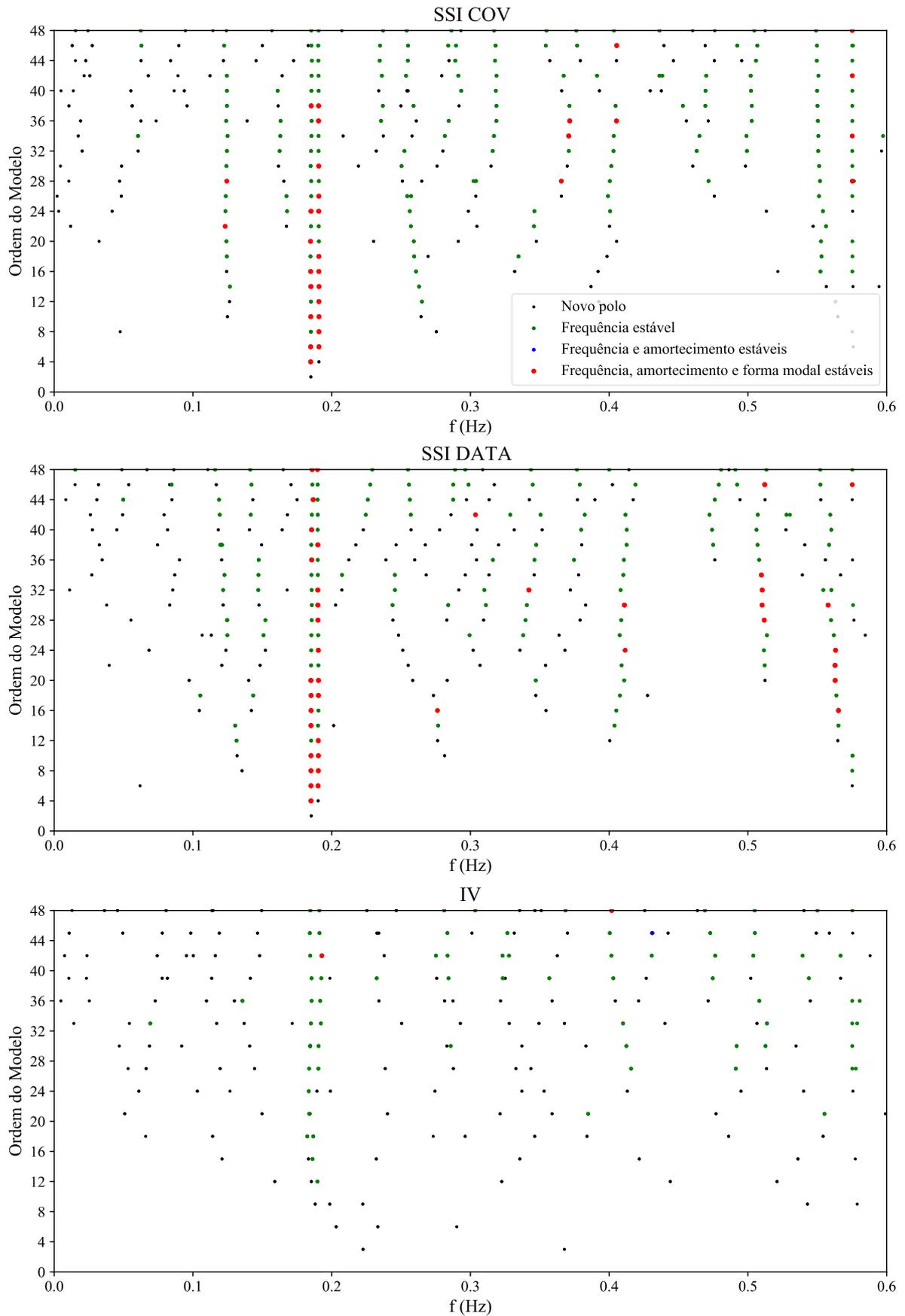
correspondem apenas a modos verdadeiros (primeiro, segundo e quarto modo da estrutura simulada).

Neste item realizaram-se simulações com relações ruído/sinal entre 0 e 495%. A partir da análise dos resultados pode-se afirmar que:

- a) a frequência é a característica dinâmica que pôde ser determinada com maior precisão;
- b) a razão de amortecimento é a propriedade com maior sensibilidade ao ruído, há um aumento da sua dispersão com o aumento do ruído;
- c) a variabilidade da forma modal identificada, medida a partir do MAC, aumenta com o aumento do ruído; foi possível detectar a forma modal com precisão considerável (MAC acima de 0,7) no intervalo de ruído estudado.

Referente à capacidade dos métodos na identificação de sistemas a partir de sinais com ruído elevado verificou-se uma divergência nos resultados. O método IV mostrou-se melhor para identificar o terceiro modo e pior para identificar o primeiro e segundo modo. Os métodos SSI foram capazes de identificar os dois primeiros modos em toda a faixa de ruído estudada. Esse comportamento observado pode ser um indicativo que (i) os métodos SSI COV e SSI DATA são capazes de identificar modos bastante presentes nas medições para relações ruído/sinal elevadas e (ii) o método IV consegue detectar modos pouco presentes nas medições para relações ruído/sinal maiores do que os métodos SSI DATA e COV. Os métodos SSI COV e SSI DATA forneceram resultados muito parecidos, com exceção do terceiro modo, no qual o método SSI DATA mostrou-se melhor.

Figura 5.25 – Diagramas de estabilização para ruído/sinal igual a 400%



(fonte: elaborado pelo autor)

5.2 PASSARELA P01

No estudo da Passarela P01, realizaram-se ensaios de medição da resposta às ações de pedestres, vento e passagem de tráfego rodoviário sob a estrutura visando identificar as propriedades dinâmicas com o emprego de um sensor. Este estudo foi feito em parceria com o pesquisador Jorge Braido, cuja Tese de Doutorado está em fase de elaboração⁴ junto ao Programa de Pós-Graduação em Engenharia Civil e Ambiental da Universidade de Passo Fundo (PPGEng/UPF) sob orientação do professor Zacarias Chamberlain.

5.2.1 Descrição da Passarela P01

A Passarela P01 é uma passarela em viga reta de concreto pré-moldado e protendido, com um vão, comprimento igual a 23,40 metros e largura de 2,20 metros. Está localizada na cidade Passo Fundo, Rio Grande do Sul, e atravessa a BR 285 no km 297. Uma foto da passarela é apresentada na Figura 5.26.

Figura 5.26 – Foto da Passarela P01

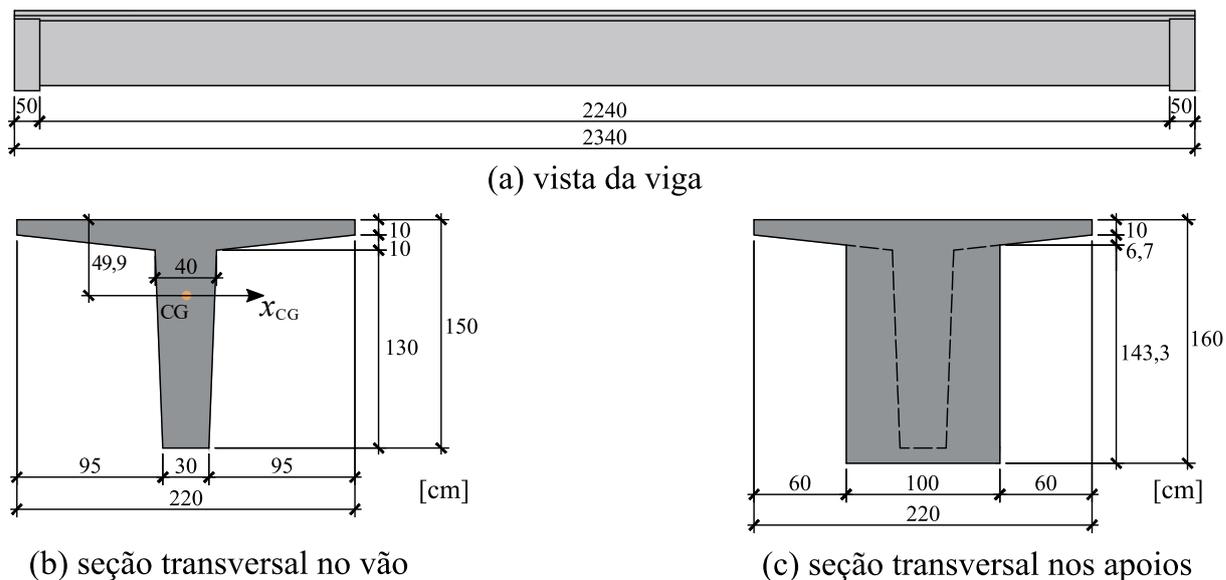


(fonte: foto do autor)

⁴ Tese de Doutorado intitulada Avaliação da Condição Estrutural de Pontes Através da Análise da Taxa de Amortecimento Modal da Superestrutura, que está sendo elaborada no Programa de Pós-Graduação em Engenharia Civil e Ambiental da Universidade de Passo Fundo (PPGEng/UPF), por Jorge Dalmas Braido sob a orientação de Zacarias Martin Chamberlain Pravia.

A geometria da viga é mostrada na Figura 5.27. As dimensões foram fornecidas pelo fabricante e conferidas no local. Nota-se que ela possui enrijecedores nas extremidades e é apoiada nos pilares por dois aparelhos de apoio em cada pilar. Juntas separam a viga das rampas de acesso.

Figura 5.27 – Geometria da Passarela P01

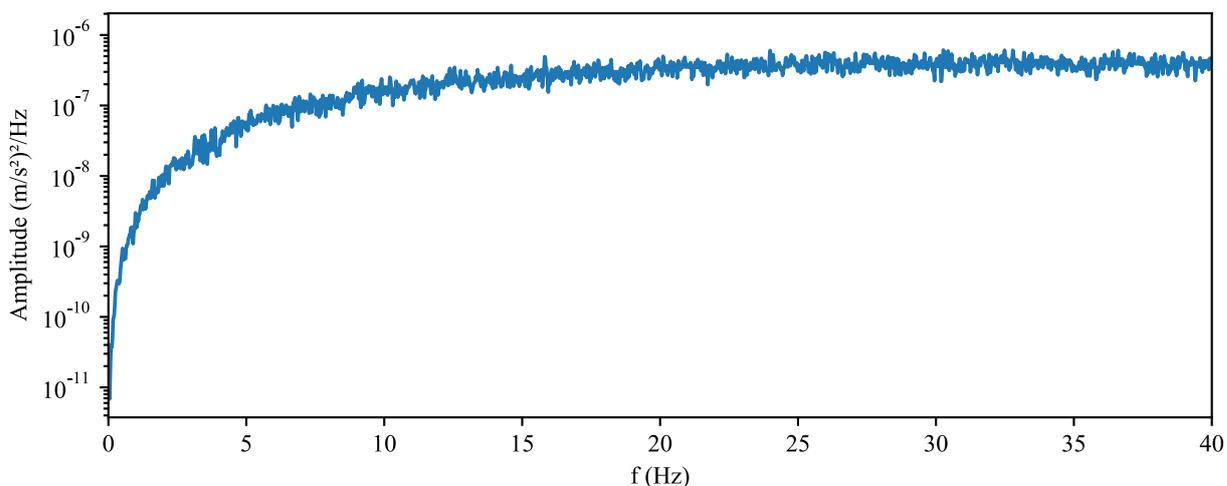


(fonte: elaborado pelo autor)

5.2.2 Instrumentação e ensaios

Empregou-se um sensor STMicro LSM6DSM nos ensaios realizados na Passarela P01. Trata-se de um acelerômetro triaxial presente em celulares topo de linha. A frequência de amostragem foi de aproximadamente 424 Hz. Como a amplitude da aceleração devida às ações ambientais é baixa, torna-se importante caracterizar o ruído intrínseco do sensor. Assim, realizaram-se medições do sensor em repouso visando estimar o valor rms do ruído e a sua densidade espectral. A estimativa obtida para o valor rms do ruído foi de $0,006 \text{ m/s}^2$ e a densidade espectral é mostrada na Figura 5.28. Nota-se que a densidade espectral é crescente até 15 Hz, aproximadamente, e possui amplitude praticamente constante a partir desta frequência.

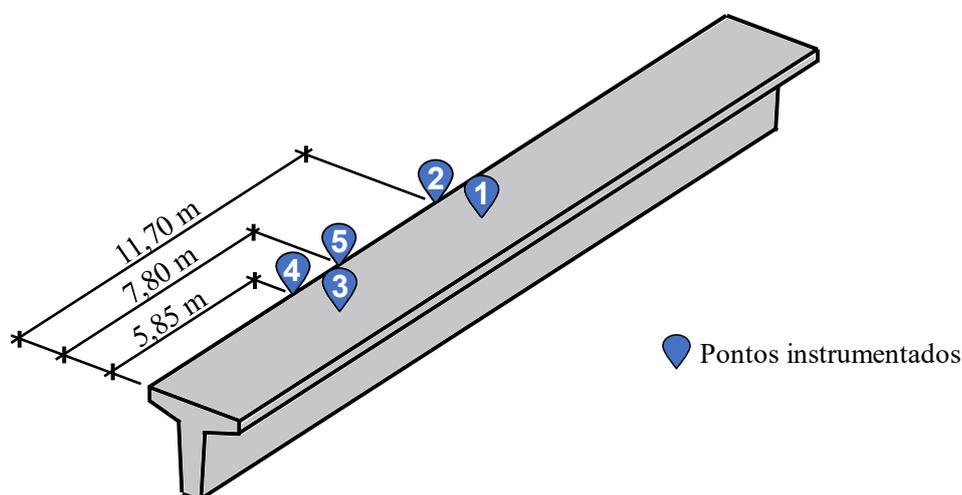
Figura 5.28 – Estimativa da densidade espectral do ruído do sensor



(fonte: elaborado pelo autor)

A resposta em aceleração vertical e transversal da estrutura foi medida em cinco posições, conforme indicado na Figura 5.29. Nas localizações 1 e 3, posicionou-se o acelerômetro exatamente na metade da largura e nas posições 2, 4 e 5, posicionou-se o acelerômetro na lateral da passarela.

Figura 5.29 – Posições do sensor na Passarela P01



(fonte: elaborado pelo autor)

Realizaram-se dois ensaios em cada uma das quatro primeiras localizações. Um ensaio denominado impulsivo (I), no qual mediu-se a resposta da estrutura devida a pulos simultâneos de duas pessoas e outro nomeado ambiental (A), no qual mediu-se a resposta devida ao caminhar dos pesquisadores, vento e tráfego rodoviário abaixo da estrutura. Nenhum pedestre utilizou a passarela durante a realização dos ensaios. Na posição 5, apenas a resposta ambiental

foi medida, por um período mais longo do que as demais. O Quadro 5.5 apresenta a nomenclatura adotada para os ensaios, a localização do acelerômetro, quantidade de amostras no tempo, duração, valor rms da aceleração e a estimativa da relação ruído/sinal dada pela divisão do valor rms do ruído do sensor pelo valor rms do sinal.

Quadro 5.5 – Ensaios na Passarela P01

Ensaio	Posição	Resposta	Valores	Duração (s)	Aceleração rms (m/s ²)		Ruído / Sinal	
					Trans-versal	Vertical	Trans-versal	Vertical
1I	1	Impulsiva	32768	77	0,008	0,011	76%	56%
1A	1	Ambiental	131072	309	0,007	0,009	86%	68%
2I	2	Impulsiva	32768	77	0,011	0,023	57%	26%
2A	2	Ambiental	131072	309	0,007	0,009	90%	66%
3I	3	Impulsiva	32768	77	0,009	0,019	68%	32%
3A	3	Ambiental	131072	309	0,006	0,008	96%	73%
4I	4	Impulsiva	32768	77	0,010	0,028	57%	21%
4A	4	Ambiental	131072	309	0,006	0,008	98%	75%
5A	5	Ambiental	262144	619	0,006	0,010	99%	62%

(fonte: elaborado pelo autor)

5.2.3 Resultados

A seguir apresentam-se os resultados da identificação da Passarela P01 a partir dos nove ensaios. Utilizaram-se os métodos BFD, SSI COV, SSI DATA e IV. Também se estimaram as frequências naturais por meio da teoria de sistemas contínuos e por meio de um modelo em elementos finitos. Por fim, comparam-se os resultados obtidos.

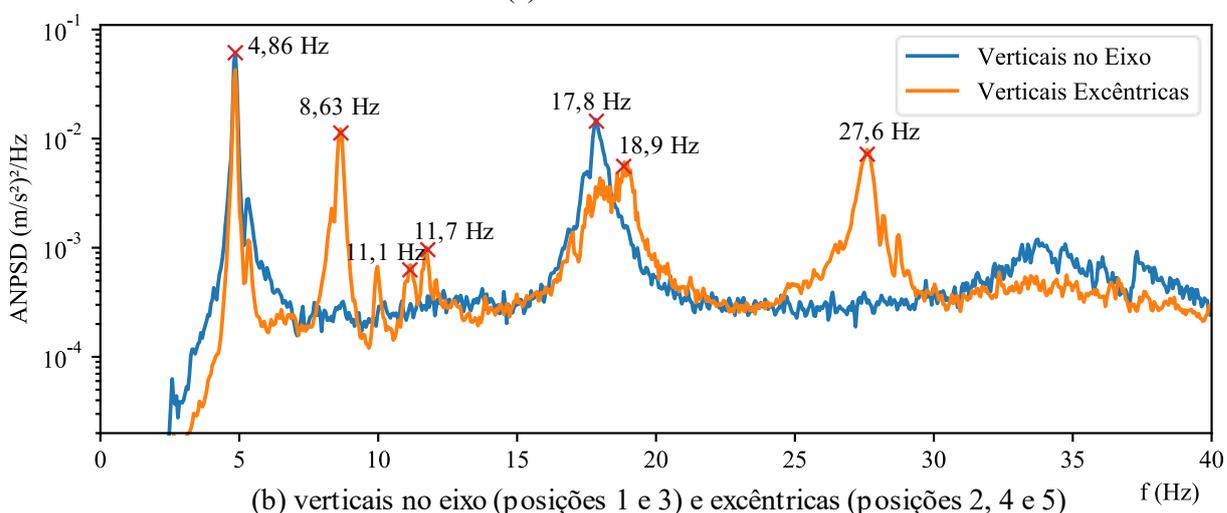
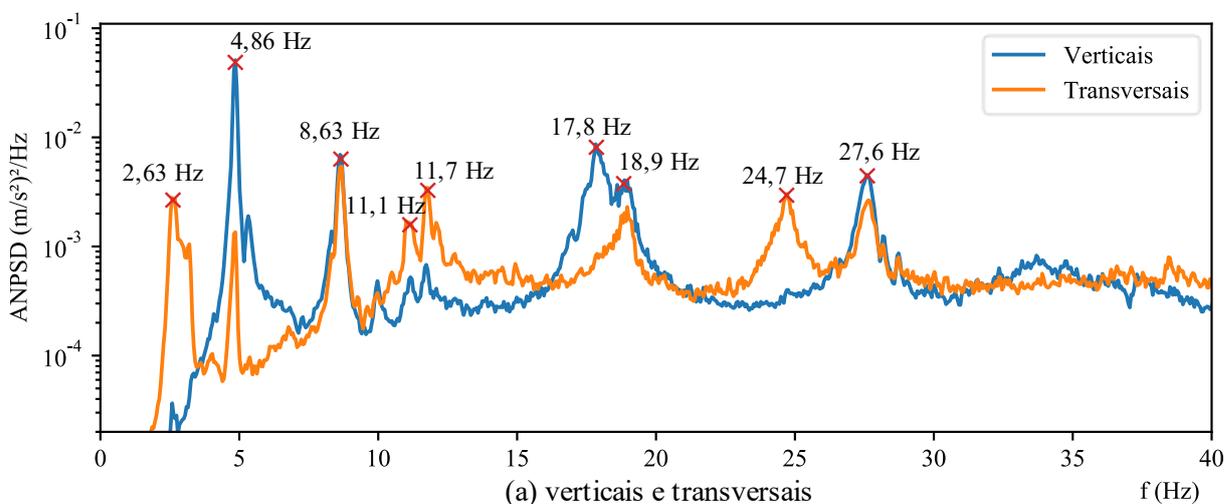
5.2.3.1 Identificação das propriedades dinâmicas a partir das acelerações

Inicia-se a apresentação dos resultados pelo Método Básico no Domínio da Frequência (BFD). A partir das séries temporais de acelerações, estimaram-se as funções de densidades espectrais das respostas pelo método de Welch. Os registros foram divididos em segmentos com 8192

valores cada. Considerou-se uma sobreposição de 50% e utilizou-se uma janela de dados do tipo Hanning. Obtiveram-se estimativas com uma resolução em frequência de $\Delta f \cong 0,052 \text{ Hz}$.

A Figura 5.30a apresenta os espectros normalizados médios das acelerações verticais e transversais e a Figura 5.30b mostra os espectros normalizados médios das acelerações verticais medidas no eixo da passarela e excêntricas. Nelas indicam-se as frequências correspondentes aos picos mais evidentes. Identificaram-se nove picos na faixa de 0 até 40 Hz.

Figura 5.30 – Espectros normalizados médios das acelerações

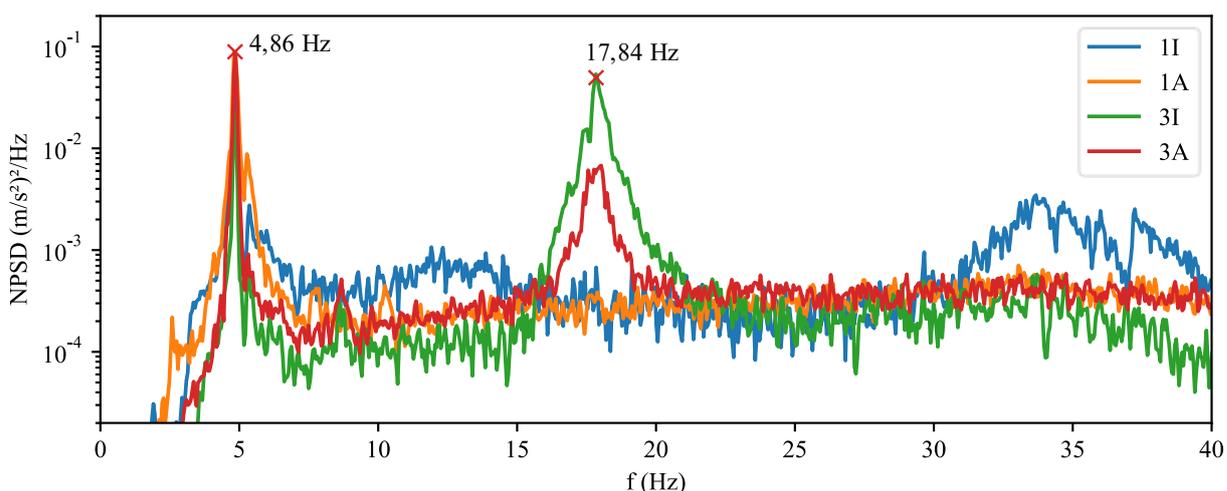


(fonte: elaborado pelo autor)

Realizou-se um estudo visando separar os modos verticais, laterais, torcionais e acoplados. Identificaram-se dois modos laterais, com frequências de 2,63 Hz e 24,7 Hz, e dois modos verticais com frequências de 4,86 Hz e 17,8 Hz. Devido ao emprego de um único sensor, não foi possível classificar os modos restantes entre torcionais e acoplados. Deste modo optou-se

por restringir o estudo aos modos verticais. A distinção desses para os demais foi feita com auxílio do ANPSD das acelerações verticais no eixo (Figura 5.30b) e das densidades espectrais das posições 1 e 3 (Figura 5.31). Verifica-se que dos nove picos existentes nos ANPSDs, apenas dois estão presentes nestes autoespectros. Nota-se que no autoespectro da localização 3 existem dois picos, um na frequência de 4,86 Hz e outro na frequência de 17,84 Hz. Tais frequências correspondem ao primeiro e segundo modo de vibração vertical, respectivamente. No autoespectro da aceleração vertical da localização 1, por sua vez, há apenas um pico na frequência de 4,86 Hz. Tal fato indica que o segundo modo apresenta forma modal com componente nula no centro do vão. Conforme será mostrado no próximo item, pela teoria de vibração livre em uma viga biapoiada, o centro do vão corresponde a um nó da segunda forma modal, o que pode explicar a ausência do pico nos espectros dos ensaios 1I e 1A.

Figura 5.31 – Espectro normalizado das acelerações verticais



(fonte: elaborado pelo autor)

Os Quadros 5.6 e 5.7 apresentam as frequências naturais e razões de amortecimento estimadas para o primeiro e segundo modo vertical, respectivamente. Empregaram-se os métodos de identificação para cada registro separadamente. Não foi possível estimar as propriedades dinâmicas para todos os ensaios pelos métodos BFD e IV. As razões de amortecimento indicadas para o BFD foram estimadas pelo método de meia potência. Para os métodos no domínio do tempo, empregou-se o filtro passa-baixa Chebyshev Tipo I de oitava ordem com uma frequência de corte de 42,4 Hz e depois reamostraram-se as séries temporais para a frequência de 106 Hz. Esse procedimento de filtragem e reamostragem corresponde à aplicação da função `scipy.signal.decimate` do Python. As tolerâncias do diagrama de estabilização

foram de 1% para frequências e de 10% para amortecimentos. A ordem máxima dos modelos em espaço de estado foi de 40. O parâmetro ι foi considerado igual a 40.

Referente às propriedades do primeiro modo apresentadas no Quadro 5.6, nota-se que todos os métodos forneceram estimativas para a frequência natural próximas entre si, com o valor de 4,8 Hz aproximadamente. No que tange ao amortecimento, o método BFD forneceu estimativas mais elevadas para a maioria dos ensaios e os métodos no domínio do tempo forneceram estimativas próximas. Nota-se uma considerável dispersão nos valores estimados para o amortecimento, constando-se que os três primeiros ensaios forneceram estimativas maiores (entre 1,61% e 3,55%) do que os demais (entre 0,54% e 1,50%). Verifica-se que ensaios com amplitudes de respostas semelhantes forneceram amortecimentos diferentes, indicando que a dispersão pode estar associada ao nível de ruído e a variabilidade do próprio amortecimento estrutural. Dada a grande variabilidade das estimativas obtidas, não foi possível definir um valor para a razão de amortecimento.

Quadro 5.6 – Frequências naturais e razões de amortecimento identificadas para o primeiro modo vertical

Ensaio	BFD		SSI COV		SSI DATA		IV	
	f (Hz)	ξ_{mp}	f (Hz)	ξ	f (Hz)	ξ	f (Hz)	ξ
1I	4,86	1,49%	4,78	2,42%	4,79	3,55%	4,80	2,87%
1A	4,81	1,74%	4,85	1,61%	4,85	1,75%	4,87	1,71%
2I	4,81	1,65%	4,78	1,95%	4,80	2,20%	4,79	2,32%
2A	4,86	1,50%	4,84	0,66%	4,84	0,68%	4,84	0,68%
3I	4,86	1,22%	4,83	0,82%	4,84	0,88%	4,83	1,09%
3A	4,86	1,26%	4,84	0,54%	4,84	0,57%	4,84	0,56%
4I	4,86	1,50%	4,84	0,78%	4,84	0,87%	-	-
4A	4,86	1,07%	4,84	0,59%	4,84	0,63%	4,84	0,59%
5A	4,86	1,36%	4,83	0,91%	4,83	0,95%	4,83	1,09%

(fonte: elaborado pelo autor)

Em relação às propriedades dinâmicas estimadas para o segundo modo, as quais constam no Quadro 5.7, também constata-se que os métodos forneceram estimativas relativamente

próximas entre si, sendo a menor frequência estimada igual a 17,82 Hz e a maior igual a 18,15 Hz, indicando uma diferença de 1,74% entre os valores. Da mesma forma que no primeiro modo, verifica-se uma grande variabilidade nas razões de amortecimento estimadas (entre 1,43% e 4,20%).

Quadro 5.7 – Frequências naturais e razões de amortecimento identificadas para o segundo modo vertical

Ensaio	BFD		SSI COV		SSI DATA		IV	
	f (Hz)	ξ_{mp}	f (Hz)	ξ	f (Hz)	ξ	f (Hz)	ξ
3I	17,84	1,43%	17,92	1,97%	17,91	1,93%	17,88	1,96%
3A	17,82	2,33%	17,84	1,86%	17,83	1,87%	17,83	1,86%
4I	-	-	18,07	2,90%	18,10	2,81%	-	-
4A	-	-	18,09	3,74%	18,15	4,17%	-	-
5A	-	-	18,03	3,56%	18,15	4,20%	-	-

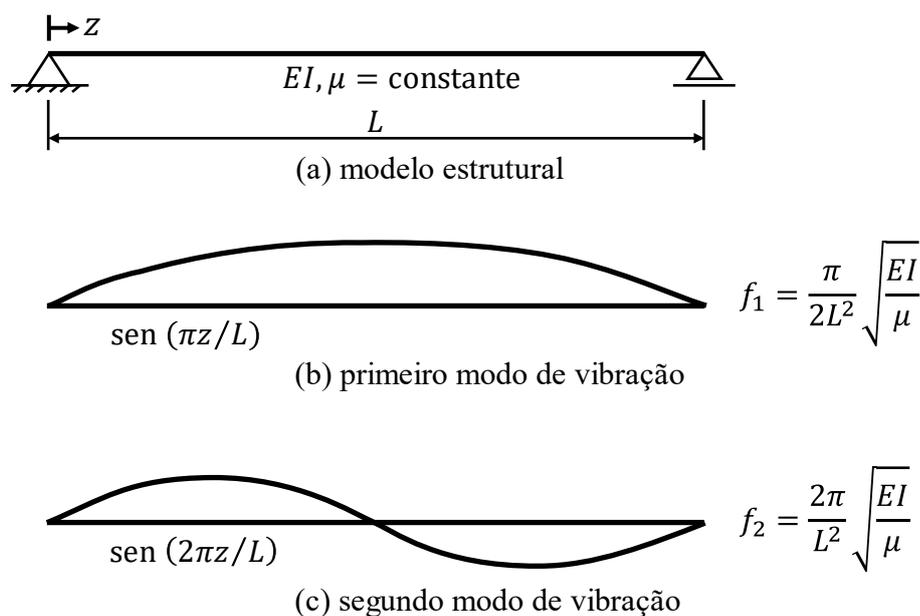
(fonte: elaborado pelo autor)

5.2.3.2 Vibrações em sistemas estruturais contínuos

A determinação das frequências naturais de vibração livre de uma viga pode ser feita de forma analítica empregando-se a teoria de vibração de sistemas contínuos. Esta pode ser encontrada na literatura técnica, por exemplo em Clough e Penzien (2003). A Figura 5.32 mostra as formas modais e frequências naturais para os dois primeiros modos de vibração de uma viga biapoiada em função do módulo de elasticidade do material E , do momento de inércia da seção I , da massa linear μ e do vão L .

No Quadro 5.8 encontram-se os valores dos parâmetros utilizados para estimativa das frequências naturais. O módulo de elasticidade do concreto foi estimado a partir da sua resistência característica à compressão, cujo valor de 30 MPa foi fornecido pelo fabricante. As propriedades geométricas foram calculadas a partir da geometria da seção bruta de concreto do vão mostrada na Figura 5.27b, desprezando-se a influência das armaduras. A massa linear foi determinada multiplicando-se a área da seção transversal pela massa específica do concreto armado, sendo desconsiderada a massa das guardas metálicas. O vão da viga foi tomado como a distância entre centros de apoios.

Figura 5.32 – Formas modais e frequências naturais de uma viga biapoada



(fonte: adaptado de CLOUGH; PENZIEN, 2003)

Quadro 5.8 – Parâmetros da estrutura

Parâmetro	Símbolo	Valor
Vão efetivo da viga	L	22,90 m
Módulo de elasticidade do concreto	E	30 GPa
Massa específica do concreto armado	ρ	2500 kg/m ³
Área da seção transversal	A	0,805 m ²
Momento de inércia em relação ao eixo central x	$I_{x,cg}$	0,171774 m ⁴
Massa linear	μ	2012,5 kg/m

(fonte: elaborado pelo autor)

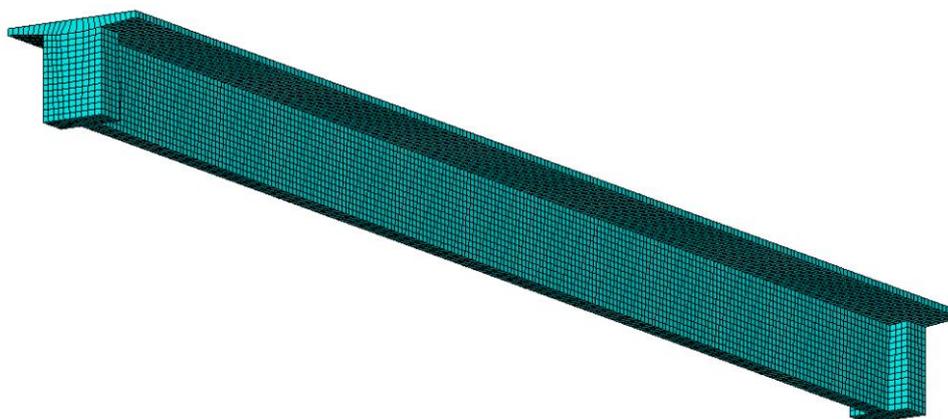
Substituindo os valores apresentados no Quadro 5.8 nas equações da Figura 5.32, obtiveram-se as frequências de 4,79 Hz para o primeiro modo e 19,17 Hz para o segundo modo.

5.2.3.3 Modelo de elementos finitos

Adicionalmente à identificação das propriedades dinâmicas a partir das acelerações e ao modelo analítico, desenvolveu-se um modelo de elementos finitos da estrutura no software ANSYS Mechanical APDL. A viga foi modelada com o elemento tridimensional SOLID186. Considerou-se material elástico linear, isotrópico, com módulo de elasticidade igual a 30 GPa,

coeficiente de Poisson igual a 0,2 e massa específica de 2500 kg/m³. A Figura 5.33 mostra o modelo.

Figura 5.33 – Perspectiva do modelo de elementos finitos da estrutura da Passarela P01



(fonte: elaborado pelo autor)

Na Figura 5.34 representam-se as formas modais e as frequências naturais dos dois primeiros modos verticais.

Figura 5.34 – Propriedades dinâmicas do modelo em elementos finitos da Passarela P01



(fonte: elaborado pelo autor)

5.2.3.4 Comparativo

O Quadro 5.9 apresenta um comparativo das frequências naturais estimadas. Nota-se excelente concordância entre os resultados para o primeiro modo. Para o segundo modo, as frequências identificadas a partir da resposta estrutural são bastante próximas. Entretanto as frequências fornecidas pela formulação analítica e pelo modelo em elementos mostraram-se mais afastadas. Tomando a frequência identificada pelo método SSI COV como referência, a solução analítica forneceu uma estimativa 6,6% superior e o modelo em elementos finitos 9,1% inferior.

Quadro 5.9 – Comparativo das frequências naturais

Método	Primeiro modo vertical	Segundo modo vertical
	f (Hz)	f (Hz)
BFD	4,85	17,83
SSI COV	4,83	17,99
SSI DATA	4,83	18,03
IV	4,83	17,86
Analítico	4,79	19,17
Elementos Finitos	4,80	16,35

(fonte: elaborado pelo autor)

Neste item identificaram-se as propriedades dinâmicas da Passarela P01 com o emprego de um sensor. Optou-se por restringir o estudo apenas aos modos puramente verticais, sendo identificados dois modos de vibração com frequências de 4,8 Hz e 18 Hz aproximadamente. Essas frequências estimadas foram comparadas com as frequências obtidas pela teoria de vibração de sistemas contínuos e por um modelo em elementos finitos, verificando-se excelente concordância entre os resultados para o primeiro modo de vibração e aceitável para o segundo modo. Graças à excelente capacidade dos métodos em analisar registros com relação ruído/sinal elevadas, conclui-se que é possível estimar as propriedades dinâmicas de uma forma simplificada até mesmo com o emprego de um sensor presente em telefones celulares.

5.3 PASSARELA P02

Neste item apresenta-se a identificação das propriedades dinâmicas da Passarela P02 submetida às ações induzidas pelos pesquisadores, vento e passagem de tráfego sob a estrutura. Empregaram-se sensores desenvolvidos por Borges (2021b), cuja Dissertação de Mestrado foi elaborada junto ao Programa de Pós-Graduação em Engenharia Civil da Universidade Federal do Rio Grande do Sul (PPGEC/UFRGS) sob orientação do prof. Marcelo Maia Rocha.

5.3.1 Descrição da Passarela P02

A P02 é a Passarela Domingos Martins, localizada na cidade de Canoas, Rio Grande do Sul, que atravessa a BR 116 no km 264,57. Trata-se de uma passarela metálica, treliçada, com um vão de 46 metros, altura de 4,00 metros e largura de 3,25 metros.

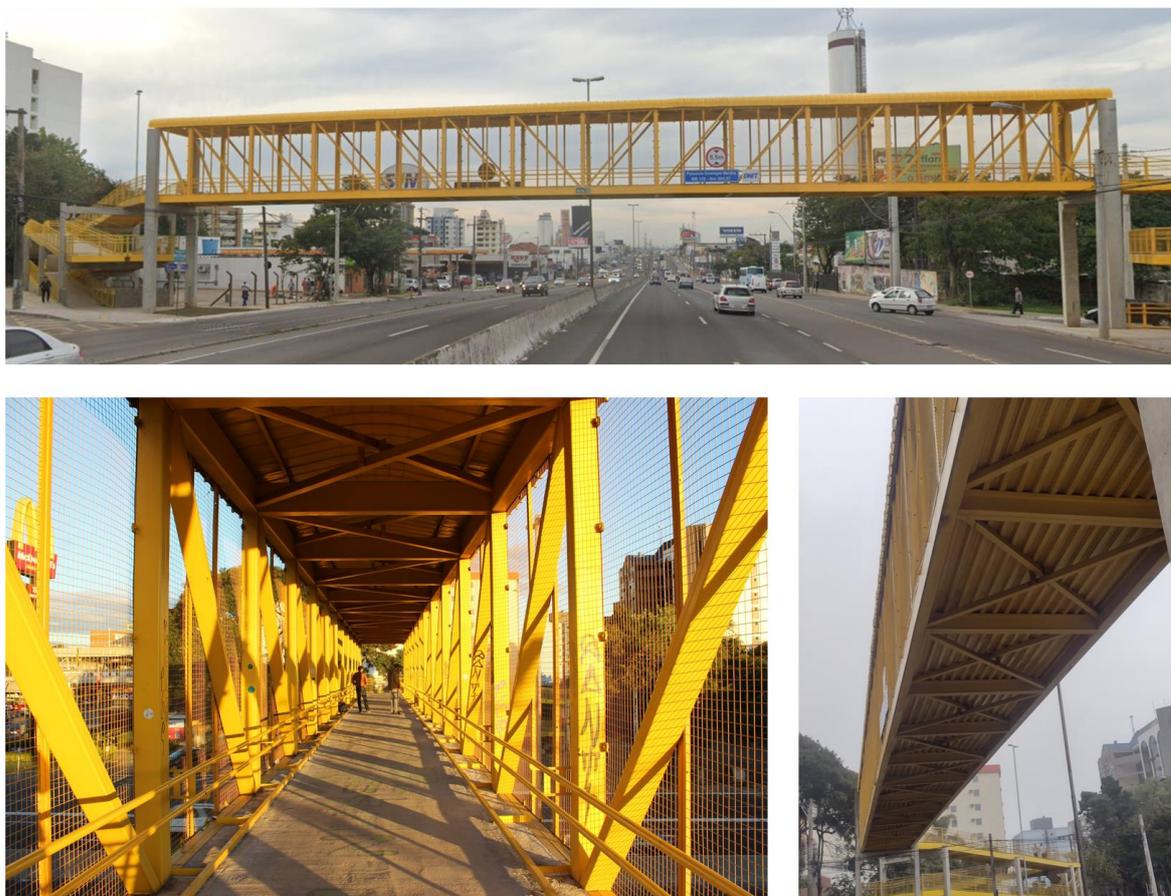
A estrutura é composta por duas treliças verticais do tipo Pratt e duas treliças horizontais, uma no plano do piso e outra no plano da cobertura. A laje é mista do tipo *steel deck* e é suportada por vigas transversais. Todos os perfis são do tipo tubular retangular, com exceção das vigas de suporte da laje, as quais são em perfil I. Três fotos da passarela são apresentadas na Figura 5.35.

5.3.2 Instrumentação e ensaios

A instrumentação e a realização dos ensaios da Passarela P02 ficaram a cargo do prof. Marcelo Maia Rocha e do Mestre José Lucas Silva Borges. Empregaram-se três sensores MPU-6500, cada um configurado para medir duas acelerações lineares e uma velocidade angular. Adquiriram-se registros com duração de 29,998 segundos e frequência de amostragem igual a 1000 Hz. A descrição completa do equipamento pode ser encontrada em Borges (2021b).

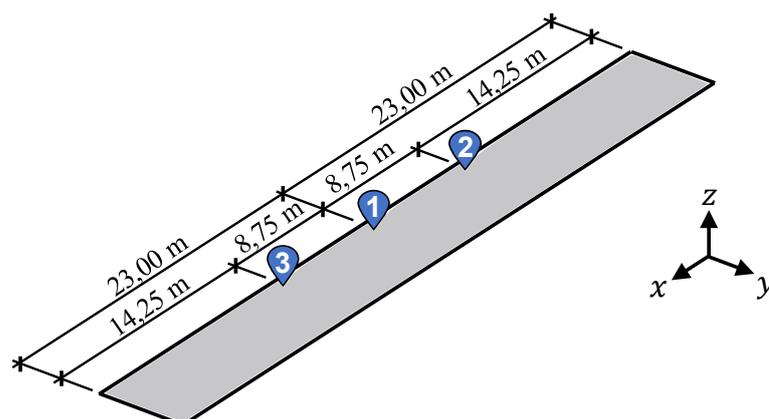
Os sensores foram posicionados no piso da estrutura e mantidos na mesma localização em todos os ensaios, conforme indicado na Figura 5.36. O sensor 1 foi posicionado no centro do vão e os sensores 2 e 3 a aproximadamente 1/3 e 2/3 do vão. Transversalmente, os sensores foram posicionados de forma excêntrica para possibilitar a passagem dos pedestres durante os ensaios. Mediram-se as acelerações lineares em relação aos eixos y e z (transversal e vertical, respectivamente) e a velocidade angular em torno do eixo x .

Figura 5.35 – Fotos da Passarela P02



(fonte: BORGES, 2021a)

Figura 5.36 – Posições dos sensores na Passarela P02 em todos os ensaios



(fonte: elaborado pelo autor)

Realizaram-se nove ensaios, os quais foram classificados em quatro casos conforme o tipo de excitação induzida pelos pesquisadores. No primeiro caso, os pesquisadores correram a partir de uma extremidade e pularam no centro do vão. No segundo caso, três pessoas pularam de forma síncrona no centro da passarela buscando excitar o primeiro modo vertical de vibração.

Excitação semelhante foi empregada no terceiro caso, mas com as pessoas pulando de forma mais acelerada e a 1/3 do vão. Por fim, no quarto caso, os pesquisadores correram de uma extremidade a outra da passarela. O Quadro 5.10 mostra a nomenclatura adotada para os ensaios, o caso de excitação, o número de valores no tempo e a duração dos registros.

Quadro 5.10 – Ensaios na Passarela P02

Ensaio	Excitação		Valores	Duração (s)
	Caso	Descrição		
1-1	1	Corrida de uma extremidade e pulo no centro do vão	29998	29,998
1-2	1	Corrida de uma extremidade e pulo no centro do vão	29998	29,998
2-1	2	Pulos no centro do vão	29998	29,998
2-2	2	Pulos no centro do vão	29998	29,998
2-3	2	Pulos no centro do vão	29998	29,998
2-4	2	Pulos no centro do vão	29998	29,998
3-1	3	Pulos a 1/3 do vão	29998	29,998
3-2	3	Pulos a 1/3 do vão	29998	29,998
4-1	4	Corrida de uma extremidade a outra	29998	29,998

(fonte: elaborado pelo autor)

O Quadro 5.11 apresenta as relações ruído/sinal dos registros adquiridos. Segundo Borges (2021b), os valores rms dos ruídos dos sensores são de 0,0198 m/s² para aceleração no eixo *y*, 0,0318 m/s² para a aceleração no eixo *z* e 0,067°/s para o giroscópio. A partir destes e dos valores rms das respostas medidas, determinaram-se as relações mostradas no Quadro 5.11. Nota-se que os registros das acelerações verticais possuem as menores relações ruído/sinal.

Quadro 5.11 – Relações ruído/sinal das séries temporais adquiridas na Passarela P02

Ensaio	Ruído / Sinal								
	Sensor 1			Sensor 2			Sensor 3		
	Trans.	Vert.	Ang.	Trans.	Vert.	Ang.	Trans.	Vert.	Ang.
1-1	60%	13%	27%	22%	16%	32%	75%	23%	75%
1-2	58%	14%	37%	30%	17%	37%	51%	17%	52%
2-1	72%	13%	81%	82%	15%	79%	85%	15%	85%
2-2	60%	10%	72%	75%	12%	75%	81%	11%	79%
2-3	56%	8%	74%	78%	9%	87%	77%	9%	88%
2-4	59%	15%	37%	31%	21%	41%	53%	19%	56%
3-1	53%	13%	25%	22%	17%	29%	30%	11%	32%
3-2	57%	13%	20%	20%	13%	26%	69%	32%	76%
4-1	57%	14%	23%	20%	15%	26%	57%	21%	49%
Média	59%	12%	44%	42%	15%	48%	64%	18%	66%

(fonte: elaborado pelo autor)

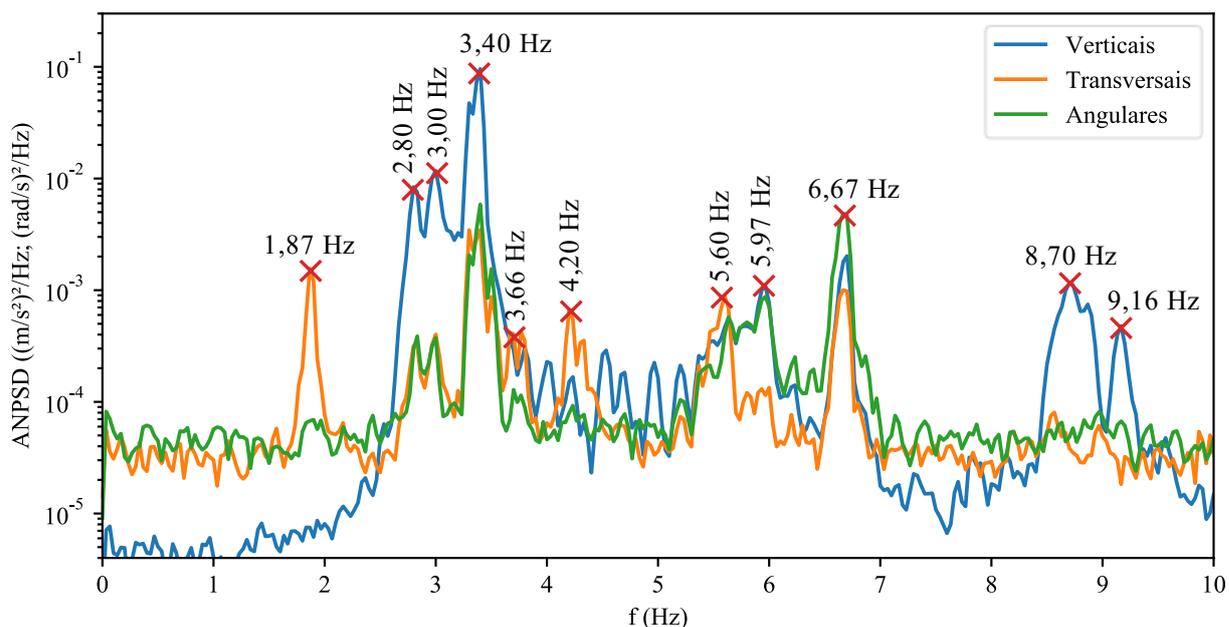
5.3.3 Resultados

Neste item apresentam-se as propriedades identificadas da Passarela P02 a partir dos nove ensaios. Realizou-se uma análise semelhante à da Passarela P01 empregando os métodos BFD e SSI COV e restringiu-se o estudo às frequências entre 0 Hz e 10 Hz.

Começa-se a apresentação dos resultados pelo domínio da frequência. Estimaram-se as funções de densidades espectrais das respostas a partir dos registros de aceleração vertical e transversal e velocidade angular. Dada a curta duração dos registros, empregou-se apenas um segmento com 29998 valores obtendo-se estimativas com uma resolução em frequência de $\Delta f \cong 0,033 \text{ Hz}$.

A Figura 5.37 apresenta os espectros normalizados médios das acelerações verticais e transversais e das velocidades angulares. Nela indicam-se as frequências correspondentes aos picos mais evidentes. Identificaram-se 11 picos na faixa de 0 até 10 Hz.

Figura 5.37 – Espectros normalizados médios das acelerações e das velocidades angulares



(fonte: elaborado pelo autor)

Em seguida, realizou-se a identificação no domínio do tempo pelo método SSI COV. As séries temporais foram classificadas de acordo com a sua direção (acelerações transversais, acelerações verticais e velocidades angulares) e analisadas separadamente com o objetivo de classificar o modo de acordo com seu tipo: vertical, transversal, torcional e acoplado. Considerando nove ensaios e três análises por ensaio (uma para cada direção), obtiveram-se 27 diagramas de estabilização, os quais foram interpretados manualmente. Identificaram-se 11 modos entre as frequências de 0 e 10 Hz. Os resultados obtidos encontram-se no Quadro 5.12. Por exemplo, o primeiro modo, com frequência de 1,88 Hz, foi identificado em 67% dos registros de aceleração transversal, em nenhum registro de aceleração vertical e também em nenhum registro de velocidade angular. Tal fato permitiu a sua classificação como modo lateral. Como os sensores não foram posicionados no eixo da passarela e embora o braço de alavanca associado com o movimento angular seja pequeno, não é possível realizar uma distinção precisa entre modos torcionais e acoplados com componente vertical e torcional. Na análise que se segue, emprega-se a numeração dos modos conforme Quadro 5.12.

Dos 11 modos identificados, constata-se que alguns aparecem em poucos ensaios. Podem tratar-se de modos pouco excitados ou espúrios. Optou-se por aprofundar a análise apenas nos modos presentes em pelo menos 50% dos ensaios, eliminando estes modos pouco excitados e identificados em apenas parte dos ensaios. Dos 11 modos identificados, sete enquadraram-se neste critério e são indicados em verde no Quadro 5.12.

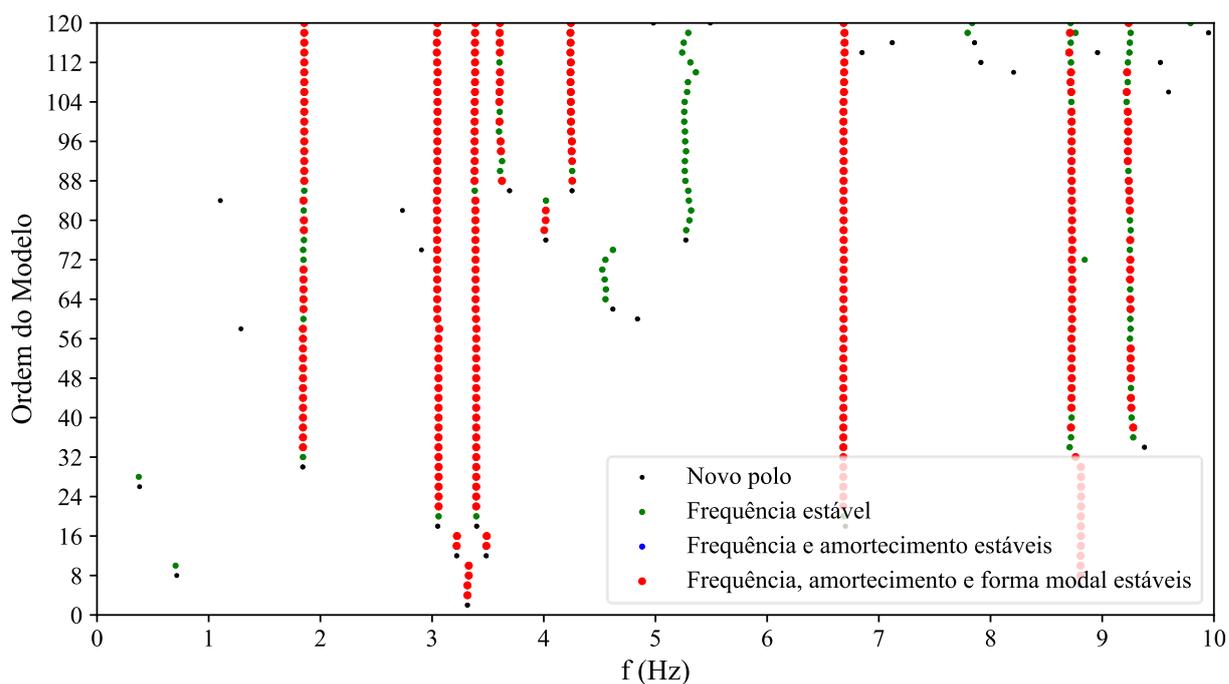
Quadro 5.12 – Frequências e modos identificados pelo método SSI COV

Modo	Frequência (Hz)	Fração de ensaios em que o modo foi identificado			Tipo de modo
		Transversal	Vertical	Torcional	
1	1,88	67%	0%	0%	Lateral
2	2,81	11%	22%	11%	Inconclusivo
3	3,03	0%	56%	11%	Vertical
4	3,39	78%	100%	67%	Acoplado
5	3,72	89%	0%	11%	Lateral
6	4,22	78%	0%	22%	Acoplado
7	5,66	33%	22%	11%	Inconclusivo
8	5,96	0%	11%	22%	Inconclusivo
9	6,68	44%	78%	89%	Acoplado
10	8,75	0%	89%	22%	Vertical
11	9,21	0%	33%	0%	Inconclusivo

(fonte: elaborado pelo autor)

Prossegue-se a análise realizando a identificação para cada ensaio empregando-se as nove séries temporais simultaneamente gerando nove diagramas de estabilização. Os registros originais com 29998 valores foram reamostrados para a frequência de 62,5 Hz com a função `scipy.signal.decimate` do Python resultando em 1874 amostras no tempo ($N = 1874$). Para cada ensaio, empregaram-se os nove registros ($l = 9$), sendo todos considerados como referência ($r = 9$). O número de atrasos no tempo foi escolhido igual a 40 ($t = 40$) e a ordem máxima do modelo em espaço de estado igual a 120 ($n_{2,max} = 120$). O diagrama de estabilização do Ensaio 1-2 é apresentado na Figura 5.38, sendo possível identificar oito prováveis modos de vibração.

Figura 5.38 – Diagrama de estabilização do Ensaio 1-2



(fonte: elaborado pelo autor)

Estimaram-se as propriedades dinâmicas para cada ensaio separadamente, obtendo-se até nove conjuntos de propriedades para cada modo de vibração. As médias das frequências naturais e as razões de amortecimento são apresentadas no Quadro 5.13. Constata-se uma grande variabilidade entre as razões de amortecimento estimadas.

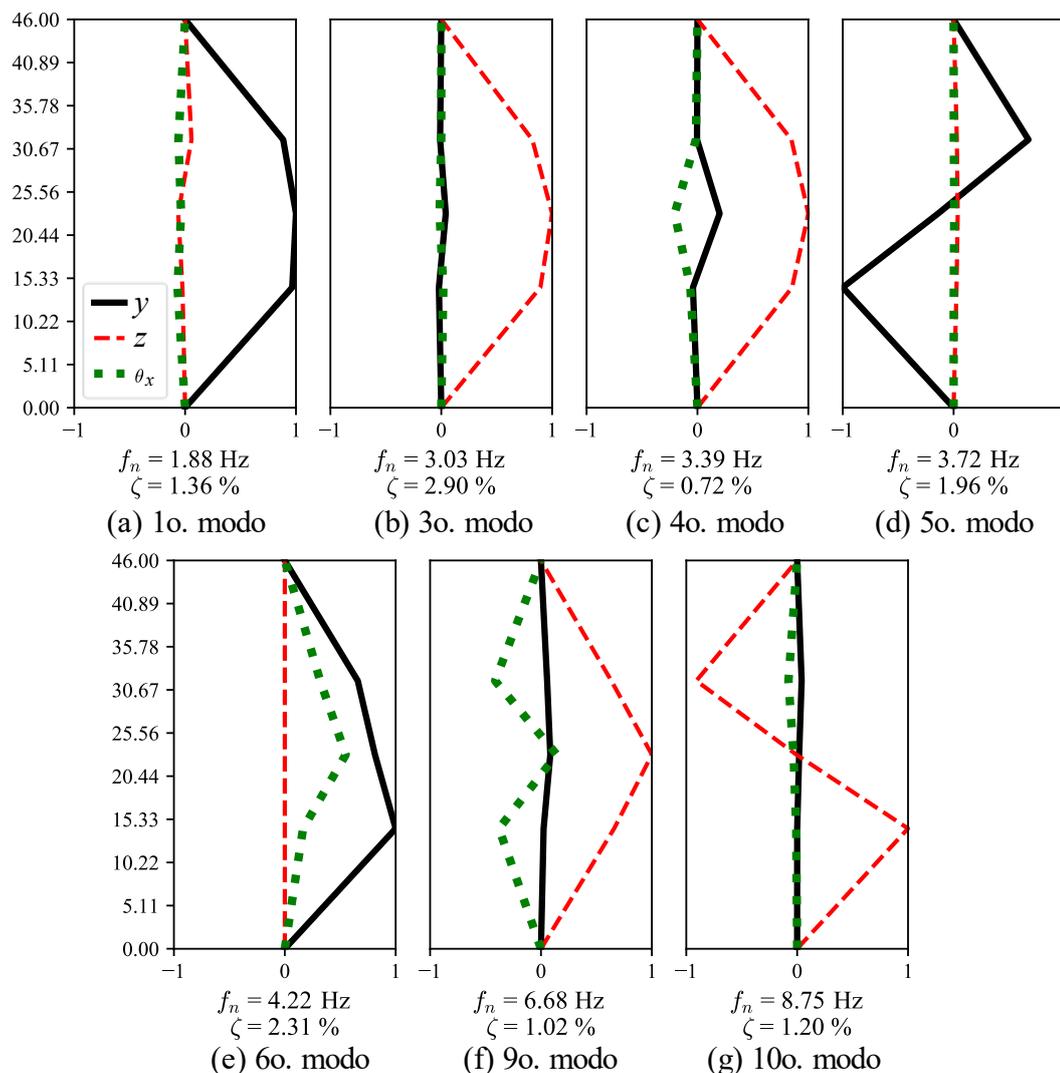
Quadro 5.13 – Frequências e razões de amortecimento identificadas pelo método SSI COV

Modo	Frequência (Hz)	Razão de amortecimento			
		Mínima	Máxima	Média	Desvio-padrão
1	1,88	0,56%	1,97%	1,09%	0,53%
3	3,03	0,88%	5,18%	2,90%	1,78%
4	3,39	0,16%	1,11%	0,72%	0,27%
5	3,72	1,16%	1,45%	1,34%	0,16%
6	4,22	0,81%	4,21%	2,31%	1,70%
9	6,68	0,31%	1,58%	0,80%	0,53%
10	8,75	0,58%	1,86%	1,20%	0,43%

(fonte: elaborado pelo autor)

As formas modais estimadas são apresentadas na Figura 5.39. Na sua representação, denotou-se a componente transversal como a componente na direção y , a vertical na direção z e a torcional na direção θ_x . O primeiro modo é um modo simétrico lateral. O terceiro é um modo simétrico vertical. O quarto modo apresenta combinação de modo transversal, vertical e torcional. O quinto modo é um modo antissimétrico lateral. A sexta forma modal apresenta componente transversal e torcional e a nona apresenta componente vertical e torcional. Finalmente, o décimo modo é um modo vertical antissimétrico. As formas modais do quarto, sexto e nono modos não foram identificadas com precisão, o que pode estar associado com a dificuldade de as ações os excitarem ou devido a diferença significativa das amplitudes dos registros.

Figura 5.39 – Propriedades dinâmicas identificadas



(fonte: elaborado pelo autor)

Neste item apresentou-se a identificação da Passarela P02, excitada principalmente pelo caminhar, corrida e pulos de três pessoas e instrumentada com três sensores. Entre as frequências de 0 a 10 Hz, identificaram-se onze possíveis modos de vibração, parte dos quais teve a sua forma modal estimada pelo método SSI COV. Identificaram-se dois modos puramente laterais, com frequências de 1,88 Hz e 3,72 Hz e dois modos verticais com frequência de 3,03 Hz e 8,75 Hz. Para a identificação mais precisa dos demais modos, sugere-se o aumento das localizações instrumentadas, empregando-se um sensor de referência e os demais sensores móveis. A combinação dos dispositivos desenvolvidos por Borges (2021b) com os algoritmos elaborados no presente trabalho possibilita a realização de diversos trabalhos futuros no âmbito da identificação modal estocástica.

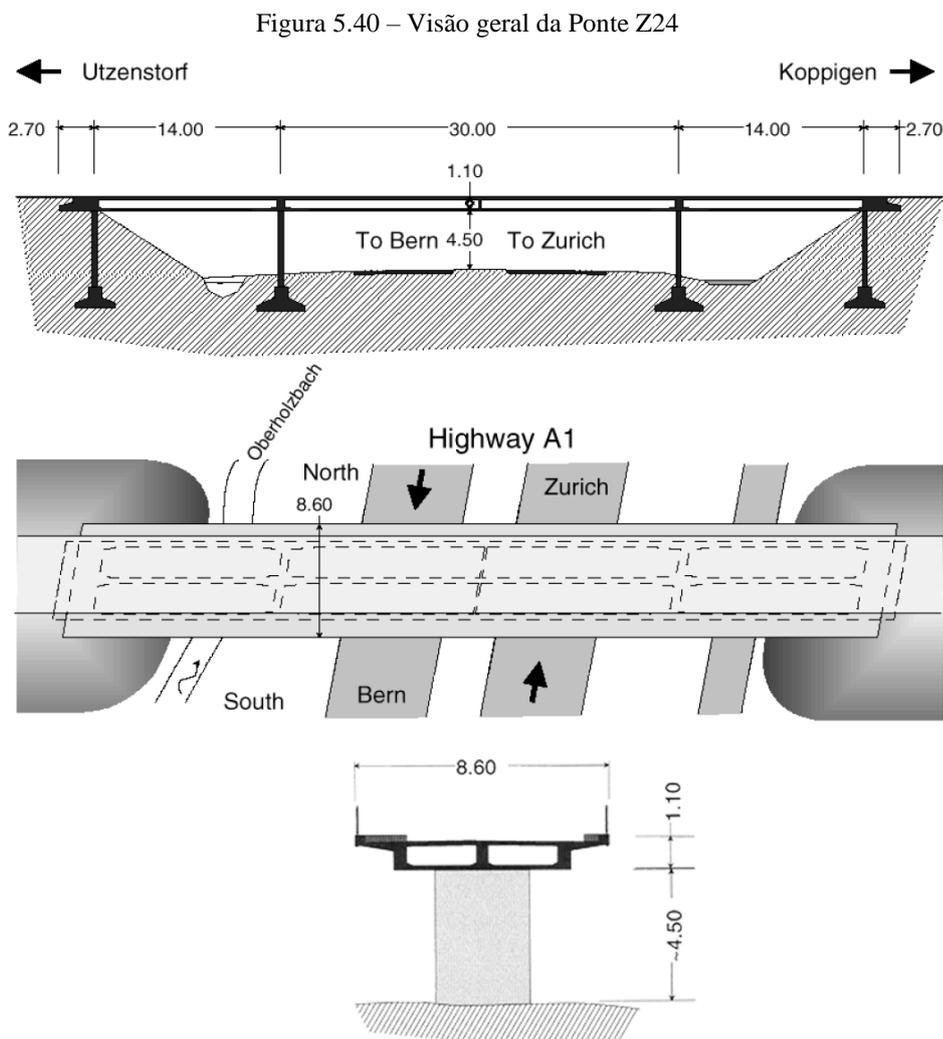
5.4 PONTE Z24

No ano de 1998, a Ponte Z24 foi escolhida para a realização de um extensivo estudo de identificação modal, monitoramento contínuo de longo prazo e localização e detecção de dano. O trabalho foi realizado no âmbito do projeto de pesquisa BE-3175, *System Identification to Monitor Civil Engineering Structures* (SIMCES), com apoio do programa europeu Brite-Euram. Os dados estão disponíveis para a comunidade científica e foram empregados em diversos estudos. Um dos principais objetivos do projeto SIMCES era fornecer uma prova da viabilidade do monitoramento da integridade estrutural por meio de medições de vibrações em escala real, ensaios de longa duração e testes com realização intencional de danos em uma estrutura representativa. Sete grupos de seis países europeus foram envolvidos. A divisão de Mecânica Estrutural da Universidade Católica de Leuven coordenou o projeto (DE ROECK, 2003).

5.4.1 Breve descrição da Ponte Z24

A Ponte Z24 era uma ponte reta e levemente esconsa. Estava localizada no Cantão de Berna próxima a cidade de Soleura, na Suíça. De acordo com a base de dados de pontes suíças, a Z24 era uma amostra representativa das pontes existentes no país. Possuía três vãos, duas faixas e comprimento total de 60 metros. O tabuleiro era em viga protendida de seção caixão com duas células. A viga era rotulada nos encontros extremos e rigidamente ligada aos pilares centrais. A Figura 5.40 apresenta uma visão geral da ponte.

Embora não houvesse nenhum problema estrutural conhecido, a ponte construída em 1963 foi demolida no final de 1998. Uma nova ferrovia adjacente à rodovia demandava uma ponte mais larga. Esse fato possibilitou que os pesquisadores causassem danos intencionalmente com o objetivo de identificá-los por meio de vibrações.



(fonte: KRÄMER, 1999)

5.4.2 Ensaios

Antes da ponte ser demolida, diversos ensaios foram realizados. Eles podem ser divididos em três grupos (PEETERS, 2000):

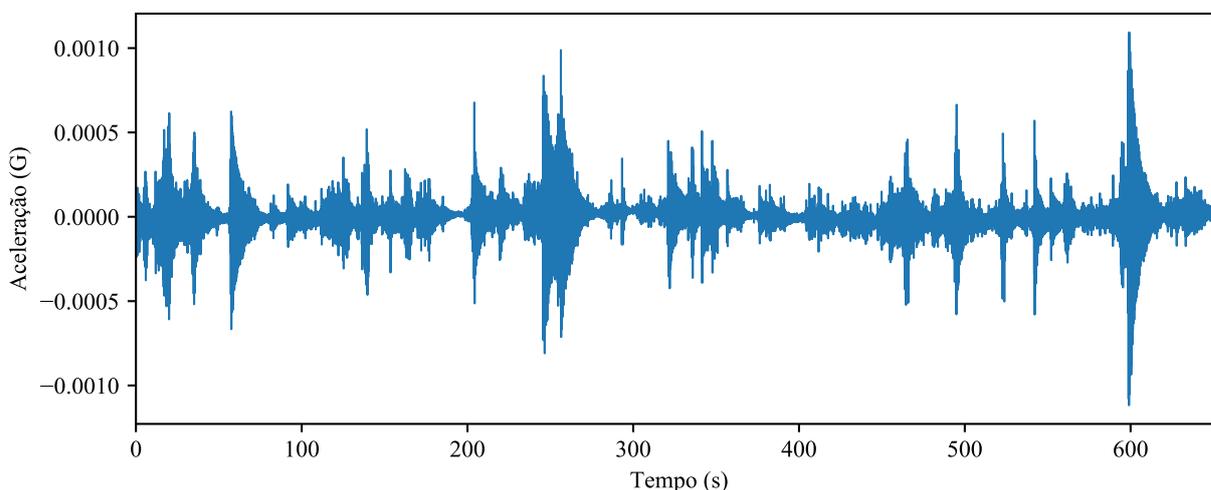
- a) monitoramento contínuo de longo prazo: este ensaio começou um ano antes da demolição e visava quantificar as influências das variáveis ambientais, como a temperatura, nas propriedades dinâmicas da ponte;

- b) ensaios intermitentes de curto prazo: o objetivo era comparar os resultados de diversos métodos de identificação de sistemas para diferentes tipos de excitações;
- c) Ensaios de Danos Progressivos (*Progressive Damage Tests* – PDT): foram realizados durante um mês imediatamente antes da demolição e visavam provar que danos na estrutura alteram as propriedades dinâmicas.

Os dados dos ensaios estão disponíveis para a comunidade científica⁵ e se tornaram um *benchmark* na identificação modal estocástica. Além disso, a Ponte Z24 foi adotada como estudo de caso pelo Grupo de Engenharia Civil na XIX Conferência Internacional de Análise Modal (*International Modal Analysis Conference* – IMAC). Diversos participantes receberam os registros de acelerações e utilizaram o método de sua preferência para realizar a identificação da estrutura.

No presente trabalho, empregaram-se os registros do Ensaio de Danos Progressivos 08 (PDT 08). Este ensaio foi escolhido por ser empregado em vários estudos, possibilitando a comparação dos resultados obtidos com os demais pesquisadores. Nele mediu-se a resposta da estrutura submetida a ações ambientais, as quais eram o tráfego da rodovia sob a ponte, o vento e o caminhar dos pesquisadores. Um registro típico de aceleração é mostrado na Figura 5.41.

Figura 5.41 – Registro de aceleração vertical no sensor R3



(fonte: elaborado pelo autor)

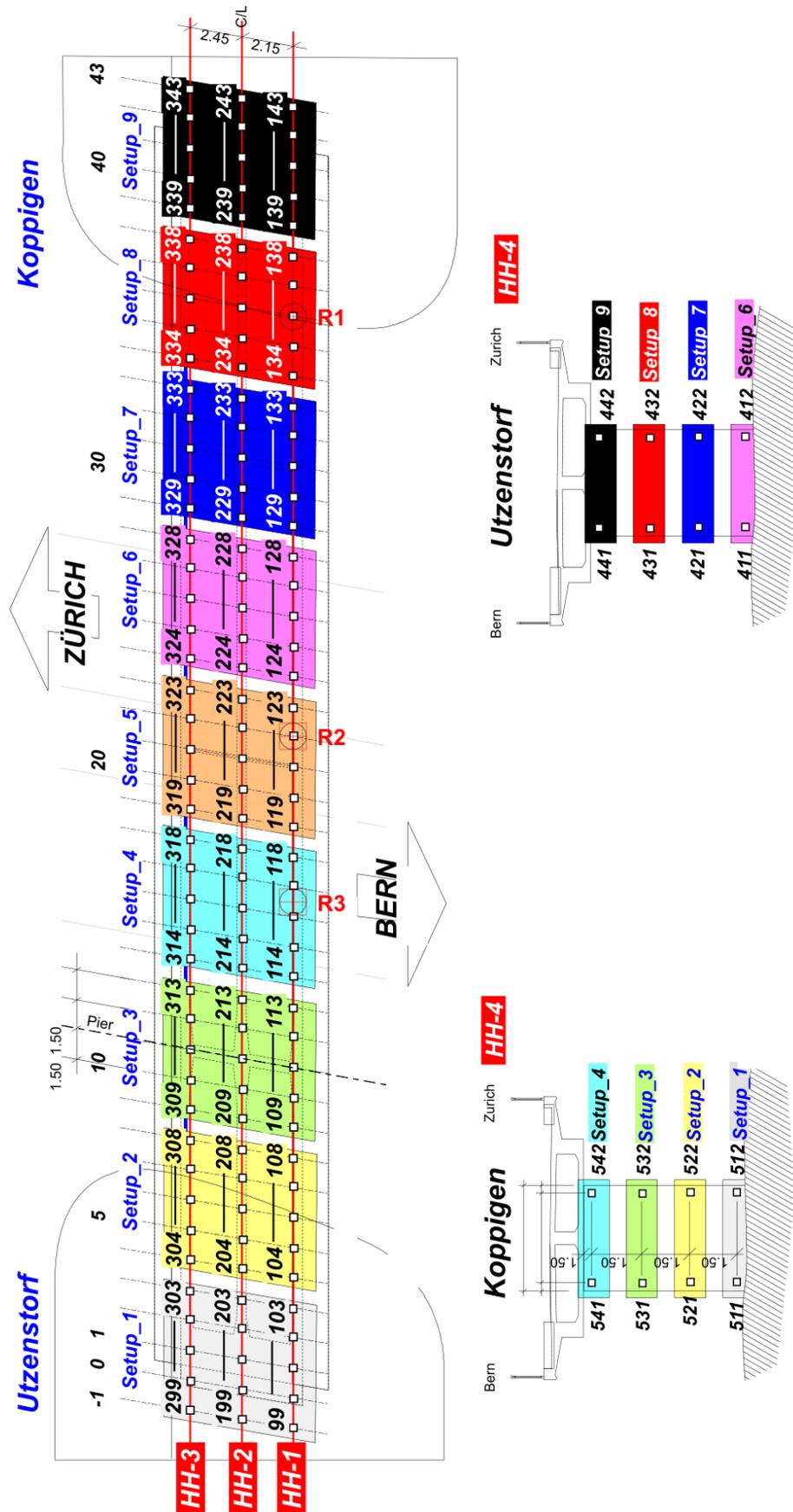
Visando identificar as formas modais detalhadamente, as acelerações da ponte foram medidas em nove configurações (*setups*). Empregaram-se acelerômetros do tipo *force balance* uni, bi e

⁵ Disponível em: < <https://bwk.kuleuven.be/bwm/z24> >. Acesso em: 26 de nov. 2020.

triaxiais. Vinte e sete registros foram adquiridos na configuração 5 e 33 nas demais. Destes registros, cinco foram definidos como referências, aqueles cuja posição é idêntica para todas as configurações. Os demais tiveram sua posição alterada em cada configuração, conforme é apresentado na Figura 5.42. Considerando as nove configurações, foram instrumentadas 135 posições no tabuleiro, 45 em cada uma das três linhas longitudinais, e 16 nos pilares.

Em cada configuração, uma malha com 15 acelerômetros foi posicionada no tabuleiro, sendo dois triaxiais, três biaxiais e dez uniaxiais, totalizando 22 registros. Posicionaram-se dois acelerômetros triaxiais em um dos pilares e três acelerômetros de referência (R). A frequência de aquisição foi de 100 Hz e o tempo de aquisição foi de 10 min e 55,36 s para cada configuração, correspondendo a 65536 amostras.

Figura 5.42 – Numeração e localização dos sensores nos PDTs



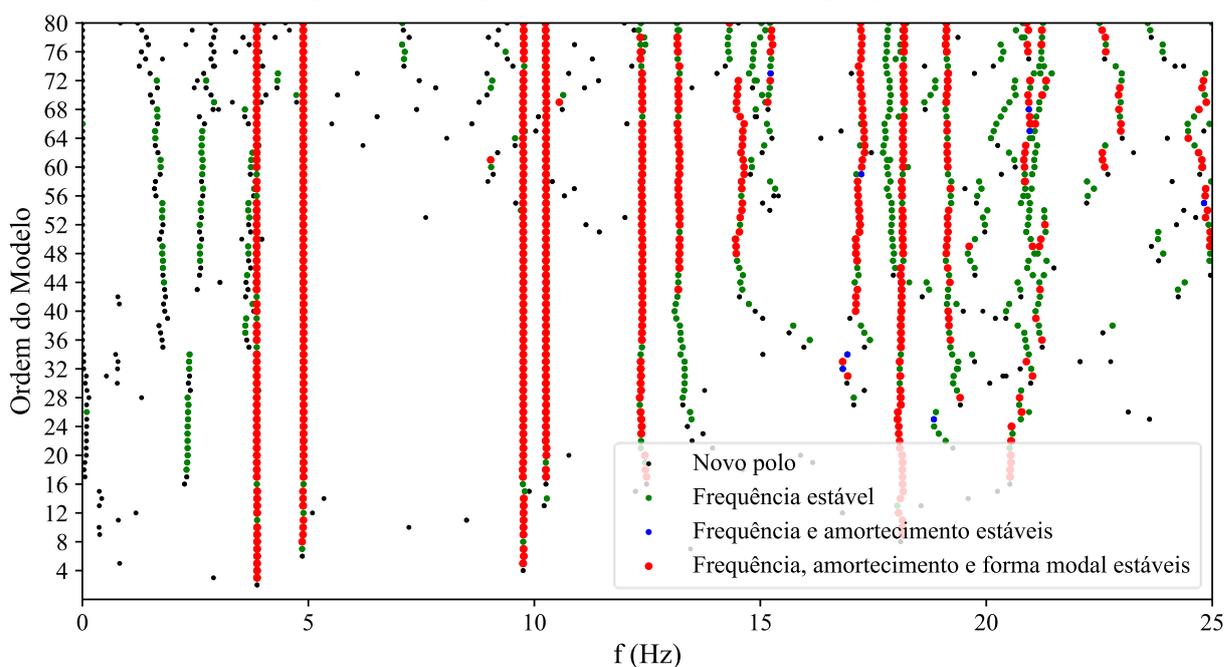
(fonte: adaptado de KRÄMER, 1999)

5.4.3 Resultados

Neste item apresentam-se os resultados da identificação da Ponte Z24 no PDT 08 submetida a ações ambientais. Empregaram-se os métodos SSI COV, SSI DATA e IV. A seguir são apresentados e discutidos os resultados do método SSI COV. Os demais resultados encontram-se no Apêndice C.

Na identificação empregaram-se as 65536 amostras no tempo ($N = 65536$), 33 registros de acelerações ($l = 33$), sendo destes cinco de referências ($r = 5$). O número de atrasos no tempo foi escolhido igual a 40 ($\iota = 40$) e a ordem máxima do modelo em espaço de estado igual a 80 ($n_{2,max} = 80$). Um diagrama de estabilização típico é apresentado na Figura 5.43.

Figura 5.43 – Diagrama de estabilização na Configuração 5



(fonte: elaborado pelo autor)

Realizou-se a identificação da estrutura para cada configuração separadamente, obtendo-se nove conjuntos de propriedades dinâmicas para cada modo de vibração. As frequências naturais e as razões de amortecimento estimadas para cada configuração são apresentadas nos Quadros 5.14 e 5.15, respectivamente, assim como as médias e desvios padrões. Um total de sete modos puderam ser identificados.

Nota-se, no Quadro 5.14, que as frequências identificadas para as várias configurações são muito próximas entre si na maioria dos casos. Referente às razões de amortecimento, constata-se no Quadro 5.15 que as estimativas para os três primeiros modos apresentaram pequena variabilidade.

Quadro 5.14 – Estimativas das frequências naturais da Ponte Z24 pelo método SSI COV

Modo	Frequência natural (Hz)										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	3,87	3,87	3,85	3,85	3,86	3,85	3,86	3,86	3,85	3,86	0,009
2	4,91	4,91	4,90	4,88	4,89	4,89	4,91	4,92	4,88	4,90	0,014
3	9,76	9,77	9,76	9,74	9,76	9,77	9,71	9,74	9,75	9,75	0,018
4	10,25	10,24	10,28	10,28	10,26	10,24	10,32	10,31	10,40	10,29	0,050
5	12,50	12,45	12,48	12,01	12,39	12,48	12,47	12,43	12,43	12,40	0,151
6	13,63	13,43	13,14	13,23	13,21	13,43	13,21	13,19	13,16	13,29	0,168
8	19,08	19,04	19,14	19,08	19,13	19,11	19,15	19,59	19,18	19,16	0,165

(fonte: elaborado pelo autor)

Quadro 5.15 – Estimativas das razões de amortecimento da Ponte Z24 pelo método SSI COV

Modo	Razão de amortecimento										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	0,94%	0,95%	0,98%	0,43%	0,80%	0,83%	0,83%	0,92%	0,86%	0,84%	0,17%
2	1,22%	1,20%	1,23%	1,27%	1,23%	0,91%	1,49%	1,41%	1,40%	1,26%	0,17%
3	1,52%	1,31%	1,28%	1,31%	1,70%	1,26%	1,27%	1,27%	1,33%	1,36%	0,15%
4	1,45%	1,08%	1,51%	1,40%	1,69%	1,19%	1,44%	0,99%	2,38%	1,46%	0,41%
5	3,40%	3,08%	3,34%	2,31%	2,27%	3,47%	2,67%	2,64%	2,90%	2,90%	0,46%
6	5,81%	4,02%	4,65%	2,68%	3,75%	5,10%	4,06%	2,08%	5,12%	4,14%	1,20%
8	2,02%	2,01%	1,94%	2,02%	1,66%	2,35%	1,86%	3,89%	1,95%	2,19%	0,66%

(fonte: elaborado pelo autor)

As médias das frequências naturais e razões de amortecimento são apresentadas no Quadro 5.16 juntamente com resultados obtidos por outros autores. O sétimo modo de vibração, encontrado por Marchesiello et al. (2001), não pode ser identificado.

Quadro 5.16 – Comparativo das frequências naturais e razões de amortecimento estimadas para a Ponte Z24

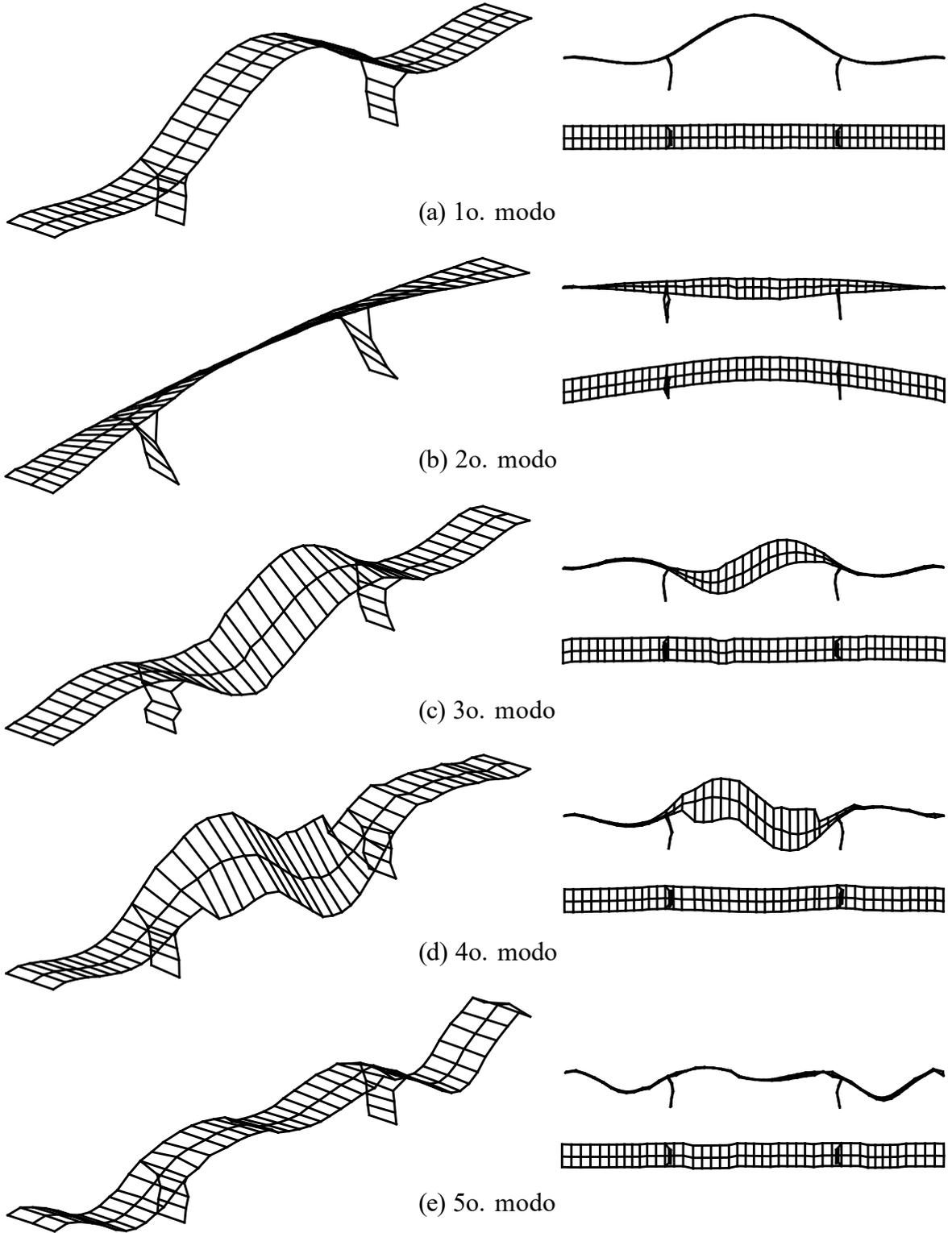
Modo	Este trabalho		Brincker et al. (2001a)		Luscher et al. (2001)		Marchesiello et al. (2001)		Peeters et al. (2000)		Yan e Miyamoto (2006)	
	f (Hz)	ξ (%)	f (Hz)	ξ (%)	f (Hz)	ξ (%)	f (Hz)	ξ (%)	f (Hz)	ξ (%)	f (Hz)	ξ (%)
1	3,86	0,84	3,86	0,79	3,85	0,80	3,85	1,20	3,86	0,9	3,86	0,62
2	4,90	1,26	4,89	1,61	4,89	1,52	4,91	1,47	4,90	1,4	4,91	1,14
3	9,75	1,36	9,78	1,26	9,82	1,83	9,76	1,41	9,77	1,3	9,73	1,42
4	10,29	1,46	10,31	1,36	10,30	1,74	10,29	1,55	10,31	1,4	10,25	1,77
5	12,40	2,90	12,50	1,36	12,70	6,05	12,39	3,03	12,50	2,5	12,31	4,11
6	13,29	4,14	13,10	1,65	-	-	13,22	3,96	13,19	3,0	13,58	2,93
7	-	-	-	-	-	-	17,27	4,64	-	-	-	-
8	19,16	2,19	-	-	-	-	19,02	2,34	19,00	2,0	-	-

(fonte: elaborado pelo autor)

As formas modais estimadas para cada configuração foram combinadas com auxílio dos sensores de referência e são apresentadas na Figura 5.44. O primeiro modo é um modo simétrico de flexão vertical. O segundo é um modo de flexão transversal combinado com torção do tabuleiro. O terceiro e o quarto são combinações de flexão vertical com a torção do vão central. O quinto modo é um modo simétrico de flexão vertical com maior amplitude nos vãos laterais. O sexto modo é um modo antissimétrico de flexão vertical com contribuição significativa da rigidez axial dos pilares. Finalmente, o oitavo modo é um modo de torção. As primeiras cinco formas modais foram estimadas com boa qualidade. Por exemplo, é claramente visível o comportamento rígido da ligação do tabuleiro com os pilares. As formas modais do sexto e oitavo modo não foram bem identificadas, o que pode estar associado com a dificuldade de as ações ambientais excitarem modos com frequências elevadas. Comparando-se visualmente as

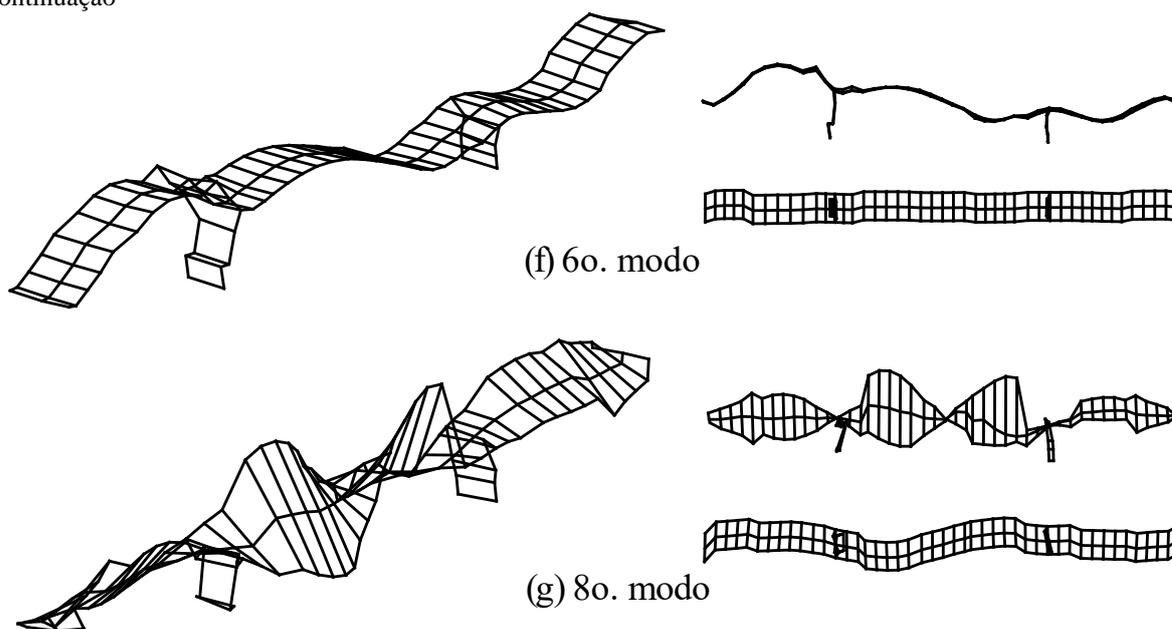
formas modais deste trabalho com as estimadas por outros autores, observou-se boa concordância nos resultados.

Figura 5.44 – Formas modais identificadas



continua

continuação



(fonte: elaborado pelo autor)

No Apêndice C encontram-se as frequências naturais e razões de amortecimento estimadas pelos métodos SSI COV, SSI DATA e IV. De uma forma geral, os métodos forneceram resultados semelhantes entre si. Dada a enorme quantidade de informação (33 registros x 65536 amostras x 9 configurações), os métodos SSI COV e IV mostraram-se melhores porque realizaram a identificação de forma mais rápida. A interpretação do diagrama de estabilização do IV mostrou-se mais difícil pois a cada nova linha r novos estados são identificados e como consequência muitos polos são representados no gráfico.

Neste item realizou-se a identificação de uma estrutura real submetida a ações ambientais. A excelente qualidade dos registros realizados por Krämer (1999) permitiu a identificação de sete modos de vibração. As rotinas programadas foram capazes de processar rapidamente a grande quantidade de informação disponível. Do ponto de vista operacional, a maior parte do tempo foi dedicada à preparação dos dados e à elaboração dos gráficos das formas modais. Também validaram-se os algoritmos com dados e resultados externos. As propriedades dinâmicas estimadas mostraram-se excelentes e próximas às obtidas por outros autores.

6 CONSIDERAÇÕES FINAIS

O presente trabalho abordou a identificação estocástica de sistemas no âmbito da Engenharia Civil, cuja finalidade é a estimativa das propriedades dinâmicas de estruturas a partir da sua resposta medida experimentalmente devida às ações ambientais. Apresentaram-se os métodos de identificação estocástica de sistemas: BFD, EFDD, SSI COV, SSI DATA e IV. Elaborou-se uma implementação em Python dos algoritmos, os quais foram aplicados na simulação e identificação do CAARC sob a ação do vento, na identificação de duas passarelas, uma em viga em viga reta pré-moldada e protendida e outra em estrutura metálica treliçada e na identificação da Ponte Z24 na Suíça.

Os métodos de identificação foram divididos de acordo com as informações que eles empregam: densidades espectrais, séries temporais ou funções de covariâncias. Descreveram-se dois métodos no domínio da frequência (BFD e EFDD), os quais utilizam as estimativas das funções de densidade espectral obtidas através do procedimento proposto por Welch (1967). O método BFD estima as propriedades dinâmicas diretamente das densidades espectrais da resposta. O método EFDD, por sua vez, emprega a decomposição em valores singulares da matriz espectral decompondo-a em sistemas com um grau de liberdade.

Os métodos SSI utilizam a representação do modelo em espaço de estado. No método SSI COV realiza-se a decomposição em valores singulares da matriz Toeplitz com as funções de covariâncias para a obtenção das matrizes do sistema e como consequência as características dinâmicas. O método IV também emprega as funções de covariâncias mas identifica os parâmetros de um modelo ARMA. Por fim, o SSI DATA evita o cálculo das covariâncias utilizando o conceito de projeção proposto por Van Overschee e De Moor (1996), a qual é calculada a partir da fatoração QR das séries temporais.

Desenvolveu-se o CESSIPy, um módulo em Python de código aberto para a estimativa das propriedades dinâmicas a partir das saídas medidas. No CESSIPy implementaram-se os cinco métodos de identificação de sistemas descritos. Elaborou-se documentação para facilitar o seu entendimento e uso. Através das rotinas desenvolvidas, pesquisadores podem adaptar e implementar o código em seu próprio software e engenheiros podem rapidamente identificar as

características dinâmicas a partir de registros de acelerações, sem a necessidade de ter um profundo conhecimento da implementação dos algoritmos.

Realizaram-se simulações numéricas da identificação do *CAARC Standard Tall Building* submetido à ação do vento. Empregaram-se as séries temporais das pressões medidas em túnel de vento e determinou-se a resposta pelo HFPI. Neste estudo foi possível comparar os métodos entre si e verificar a sua robustez na presença de ruído nas medições. A partir das rotinas programadas, obtiveram-se informações úteis para o planejamento da instrumentação da estrutura, como uma estimativa da velocidade do vento a partir da qual ela pode ser identificada e o tempo mínimo de aquisição. Visto que os estudos experimentais são caros e trabalhosos, o CESSIPy mostrou-se uma ferramenta útil para o planejamento das medições.

No estudo do CAARC, não foi possível empregar o EFDD na identificação de todos os modos por causa dos baixos valores do MAC na vizinhança de alguns picos dos espectros dos valores singulares. Os demais métodos forneceram bons resultados para a maioria dos modos. Observou-se que os três métodos no domínio do tempo identificaram as características dinâmicas com boa concordância entre si. Estes métodos mostraram-se robustos, sendo propícios para a realização de estudos paramétricos e foram capazes de realizar a identificação para relações ruído/sinal elevadas.

Realizou-se a instrumentação e identificação da Passarela P01, uma passarela em viga reta de concreto pré-moldado e protendido com um sensor. Mediu-se a resposta da estrutura submetida a ações ambientais, as quais eram o caminhar de pedestres, o vento e o tráfego rodoviário abaixo da estrutura, e impulsiva, proveniente dos pulos de pesquisadores. Foi possível detectar nove picos nos espectros normalizados médios das acelerações verticais e transversais entre as frequências de 0 Hz e 40 Hz. Em seguida, identificaram-se os dois primeiros modos de vibração verticais. As estimativas obtidas pelos métodos BFD, SSI COV, SSI DATA e IV foram comparadas com as fornecidas pela teoria de vibração em sistemas contínuos e um modelo em elementos finitos. Constatou-se excelente concordância entre os resultados para o primeiro modo de vibração vertical com frequência natural de cerca de 4,85 Hz. Neste estudo conclui-se que é possível realizar a identificação das propriedades dinâmicas de uma forma simplificada com sensores presentes em telefones celulares, mesmo em registros com relações ruído/sinal entre 21% e 75%.

A partir das séries temporais adquiridas pelo equipamento desenvolvido por Borges (2021b), identificaram-se as propriedades dinâmicas da Passarela P02, uma passarela em estrutura metálica treliçada com laje mista de aço e concreto e 46 metros de vão. Entre as frequências de 0 e 10 Hz, identificaram-se onze possíveis modos de vibração. Destes, sete tiveram sua forma modal estimada sendo identificados dois modos transversais, dois verticais e três acoplados. Constatou-se que o primeiro modo de vibração identificado, com frequência de 1,88 Hz, é um modo transversal. A combinação dos algoritmos elaborados no presente trabalho com os dispositivos desenvolvidos por Borges mostra-se promissora, possibilitando a realização de diversos estudos na área de identificação de sistemas a partir de vibrações.

Analisaram-se os dados de vibrações ambientais de Krämer (1999) provenientes de um extensivo estudo da Ponte Z24 na Suíça, a qual tornou-se um *benchmark* na identificação estocástica de sistemas. As rotinas programadas foram capazes de processar os 33 registros de acelerações de cada uma das nove configurações e identificaram sete modos de vibração com frequências entre 0 Hz e 20 Hz. Os resultados obtidos foram comparados com os de outros pesquisadores, proporcionando a validação das rotinas com dados e resultados externos.

Os algoritmos desenvolvidos neste trabalho poderão ser utilizados em futuros estudos. Como continuação do presente trabalho, sugerem-se os seguintes temas:

- a) a instrumentação de estruturas, a identificação das propriedades dinâmicas e o comparativo com as obtidas por modelos numéricos;
- b) o estudo de métodos de atualização de modelos numéricos a partir das características dinâmicas identificadas;
- c) a identificação das propriedades dinâmicas de diversos edifícios para a obtenção de relações empíricas entre a sua altura e frequência natural;
- d) o estudo e a implementação de métodos de identificação em tempo real no CESSIPy;
- e) a instrumentação de um edifício que faça parte da base de dados do LAC para comparação da resposta medida com a obtida pelo HFPI;
- f) o estudo da relação entre a amplitude da resposta e as propriedades dinâmicas identificadas.

REFERÊNCIAS

AGUIRRE, L. A. **Introdução à Identificação de Sistemas**: técnicas lineares e não lineares: teoria e aplicação. Belo Horizonte: Editora UFMG, 2015.

AL-NIMRY, H.; RESHEIDAT, M.; AL-JAMAL, M. Ambient Vibration Testing of Low and Medium Rise Infilled RC Frame Buildings in Jordan. **Soil Dynamics and Earthquake Engineering**, Irbid, v. 59, p. 21-29, Apr. 2014.

ALLEMANG, R. J.; BROWN, D. L. A Correlation Coefficient for Modal Vector Analysis. In: International Modal Analysis Conference, 1st, Orlando. **Proceedings...** Orlando: IMAC, p. 110-116, Nov. 1982.

ALLEMANG, R. J.; BROWN, D. L.; FLADUNG, W. A. Modal Parameter Estimation: a unified matrix polynomial approach. In: International Modal Analysis Conference, 1994, Honolulu. **Proceedings...** Honolulu: IMAC, p. 501-514, 1994.

AMANI, M. G.; RIERA, J. D.; CURADELLI, R. O. Identificación de estructuras sometidas a excitaciones ambientales. In: JORNADAS SUD-AMERICANAS DE INGENIERÍA ESTRUCTURAL, 31a., 2004, Mendoza. **Memorias...** Mendoza: Universidad Nacional de Cuyo, 2004. p. 1-12.

ANDERSEN, P. **Identification of Civil Engineering Structures using Vector ARMA Models**. 1997. 244 f. Thesis (Ph. D.) – Department of Building Technology and Structural Engineering, Aalborg University, Aalborg, 1997.

ARMENISE, G.; VACCARI, M.; CAPACI, R. B. di; PANNOCCHIA, G. **System Identification Package for Python: SIPPY**. Pisa, 2018. Disponível em <<https://github.com/CPCLAB-UNIFI/SIPPY>>. Acesso em: 2 abr. 2020.

ASMUSSEN, J. C. **Modal Analysis Based on the Random Decrement Technique**: application to civil engineering structure. 1997. 227 f. Thesis (Ph. D.) – Department of Building Technology and Structural Engineering, Aalborg University, Aalborg, 1997.

BENDAT, J. S.; PIERSOL, A. G. **Engineering Applications of Correlation and Spectral Analysis**. 2nd ed. New York: Wiley, 1993.

BENDAT, J. S.; PIERSOL, A. G. **Random Data**: analysis and measurement procedures. 4th ed. New York: Wiley, 2010.

BILLINGS, S. A. **Nonlinear System Identification**: NARMAX methods in the time, frequency and spatio-temporal domains. Chichester: Wiley. 2013.

BLESSMANN, J. **Aerodinâmica das construções**. 3. ed. Porto Alegre: Editora da UFRGS, 2011.

BORGES, J. L. S. Publicação eletrônica [mensagem pessoal]. Mensagem recebida por <matheuscarini@gmail.com> em 22 jun. 2021.

- BORGES, J. L. S. **Desenvolvimento de um dispositivo para monitoramento dinâmico de estruturas**. 2021. 100 f. Dissertação (Mestrado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2021b.
- BRAIDO, J. D. **Avaliação da superestrutura de pontes através de parâmetros dinâmicos**. 2014. 107 f. Dissertação (Mestrado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil e Ambiental, Universidade de Passo Fundo, Passo Fundo, 2014.
- BRINCKER, R.; JENSEN, J. L.; KRENK, S. Spectral Estimation by the Random DEC Technique. **Fracture and Dynamics**, Aalborg, v. 9008, n. 20, p. 1-10, Apr. 1990.
- BRINCKER, R.; ZHANG, L.; ANDERSEN, P. Modal Identification from Ambient Responses using Frequency Domain Decomposition. In: International Modal Analysis Conference, 18th, 2000, San Antonio. **Proceedings...** San Antonio: IMAC, p. 625-630, Feb. 2000.
- BRINCKER, R.; ANDERSEN, P.; CANTIENI, R. Identification and Level I Damage Detection of the Z24 Highway Bridge. **Experimental Techniques**, Aalborg, v. 25, p. 51-57, Dec. 2001a.
- BRINCKER, R.; VENTURA, C. E.; ANDERSEN, P. Damping Estimation by Frequency Domain Decomposition. In: International Modal Analysis Conference, 19th, 2001, Orlando. **Proceedings...** Orlando: IMAC, p. 698-703, Feb. 2001b.
- BROWN, D. L.; ALLEMANG, R. J.; MERGEAY, M. Parameter Estimation Techniques for Modal Analysis. **SAE Transactions**, Cincinnati, v. 88, p. 828-846, 1979.
- BROWNJOHN, J. M. W.; SEVEM, R. T.; TAYLOR, C. A. Ambient Vibration Survey of Contra Dam. **Research Report University of Bristol**, Bristol, 1986.
- BROWNJOHN, J. M. W.; DUMANOGLU, A. A.; SEVERN, R. T.; TAYLOR, C. A. Ambient Vibration Measurements of the Humber Suspension Bridge and Comparison with Calculated Characteristics. **Proceedings of the Institution of Civil Engineers**, Bristol, v. 83, n. 3, p. 561-600, 1987.
- BROWNJOHN, J. M. W. Ambient Vibration Studies for System Identification of Tall Buildings. **Earthquake Engineering and Structural Dynamics**, Singapore, v. 32, n. 1, p. 71-95, Apr. 2002.
- BROWNJOHN, J. M. W.; MAGALHES, F.; CAETANO, E.; CUNHA, A. Ambient Vibration Re-Testing and Operational Modal Analysis of the Humber Bridge. **Engineering Structures**, Sheffield, v. 32, n. 8, p. 2003-2018, Aug. 2010.
- BUKENYA, P.; MOYO, P.; BEUSHAUSEN, H.; OOSTHUIZEN, C. Health Monitoring of Concrete Dams: a literature review. **Journal of Civil Structural Health Monitoring**, Cape Town, v. 4, n. 4, p. 235-244, Nov. 2014.
- CAETANO, E.; MAGALHÃES, F.; CUNHA, A.; FLAMAND, A.; GRILLAUD, G. Comparison of stochastic identification methods applied to the natural response of Millau Viaduct. In: INTERNATIONAL CONFERENCE ON EXPERIMENTAL VIBRATION

ANALYSIS FOR CIVIL ENGINEERING, 2nd, 2007, Porto. **Proceedings...** Porto: EVACES, 2007. Não paginado.

CASTELLARO, S.; PERRICONE, L.; BARTOLOMEI, M.; STEFANO, I. Dynamic Characterization of the Eiffel Tower. **Engineering Structures**, Bologna, v. 126, n. 1, p. 628-640, Nov. 2016.

CLOUGH, R. W.; PENZIEN, J. **Dynamics of Structures**. 3rd ed. Berkeley: Computers & Structures, 2003.

COLE, H. A. On-The-Line Analysis of Random Vibrations. In: AIAA/ASME STRUCTURES, STRUCTURAL DYNAMICS AND MATERIALS CONFERENCE, 9th, 1968, Palm Springs. **Proceedings...** Palm Springs: American Institute of Aeronautics and Astronautics, 1968. Não paginado.

COOLEY, J. W.; TUKEY, J. W. An Algorithm for the Machine Calculation of Complex Fourier Series. **Mathematics of Computation**, Princeton, v. 19, n. 90, Apr. 1965.

CRISAFULLI, F. J.; CARR, A. J.; PARK, R. Analytical Modelling of Infilled Frame Structures: a general review. **Bulletin of the New Zealand Society for Earthquake Engineering**, v. 33, n. 1, p. 30-47, Mar. 2000.

CUNHA, A.; CAETANO, E.; MAGALHÃES, F.; MOUTINHO, C. Dynamic identification and continuous dynamic monitoring of bridges: different applications along bridges life cycle. **Structures and Infrastructure Engineering**, Porto, v. 14, n. 4, p. 445-467, Dec. 2017.

DE ROECK, G. The state-of-the-art of damage detection by vibration monitoring: the SIMCES experience. **Journal of Structural Control**, Leuven, v. 10, n. 2, p. 127-134, Apr. 2003.

DRAGOIESCU, C.; GARBER, J.; KUMAR, K. S. A comparison of force balance and pressure integration techniques for predicting wind-induced responses of tall buildings. In: STRUCTURES CONGRESS, 2006, Ontario. **Proceedings...** Ontario: ASCE, 2006.

EWINS, D. J. **Modal Testing: theory, practice and application**. 2nd ed. Baldock: Research Studies, 2000.

FELBER, A. J. **Development of a Hybrid Bridge Evaluation System**. 1993. 277 f. Thesis (Ph. D.) – Department of Civil Engineering, University of British Columbia, British Columbia, 1993.

GENTILE, C.; SAISI, A. Ambient Vibration Testing of Historic Masonry Towers for Structural Identification and Damage Assessment. **Construction and Building Materials**, v. 21, n. 6, p. 1311-1321, Jun. 2007.

GILLES, D. In Situ Dynamic Properties of Buildings in Montréal Determined from Ambient Vibrations Records. **Structural Engineering Series Report**, Montreal, n. 2010-03, 2010.

GIRALDO, D. F.; SONG, W.; DYKE, S. J.; CAICEDO, J. M. Modal Identification through Ambient Vibration: Comparative Study. **Journal of Engineering Mechanics**, v. 135, n. 8, p. 759-770, Aug. 2009.

GUILLAUME, P.; HERMANS, L.; VAN DER AUSWERAER, H. Maximum Likelihood Identification of Modal Parameters from Operational Data. In: Noise and Vibration Engineering, 1998, Leuven. **Proceedings...** Leuven: ISMA, p. 1887-1893, 1998.

HOLMES, J. D.; TSE, T. K. T. International High-Frequency Base Balance Benchmark Study. **Wind and Structures**, Victoria, v. 18, n. 4, p. 457-471, Apr. 2014.

HU, W.; CUNHA, A.; CAETANO, E.; MAGALHÃES, F.; MOUTINHO, C. LabVIEW toolkits for output-only modal identification and long-term dynamic structural monitoring. **Structure and Infrastructure Engineering**, v. 6, n. 5, p. 557-574, Oct. 2010.

IBRAHIM, S. R. Modal Confidence Factor in Vibration Testing. **Journal of Spacecraft**, v. 15, n. 5, p. 313-316, Sept. 1978.

IBRAHIM, S. R.; MIKULCIK, E. C. A Time Domain Modal Vibration Test Technique. **The Shock and Vibration Bulletin**, v. 43, n. 4, p. 21-37, 1973.

IBRAHIM, S. R.; MIKULCIK, E. C. A Method for the Direct Identification of Vibration Parameters from the Free Response. **The Shock and Vibration Bulletin**, v. 47, n. 4, p. 183-198, 1977.

IBRAHIM, S. R. Random Decrement Technique for Modal Identification of Structures. **Journal of Spacecraft and Rockets**, Norfolk, v. 14, n. 11, p. 696-700, May 1977.

IBRAHIM, S. R.; ASMUSSEN, J. C.; RUNE, B. Modal Parameter Identification from Responses of General Unknown Random Inputs. In: INTERNATIONAL MODAL ANALYSIS CONFERENCE, 14th, 1996, Dearborn. **Proceedings...** Dearborn: IMAC, 1996. p. 446-452.

JAMES, G. H.; CARNE, T. G.; LAUFFER, J. P. The Natural Excitation Technique (NExT) for Modal Parameter Extraction from Operating Structures. *Journal of Analytical and Experimental Modal Analysis*, Albuquerque, v. 10, n. 4, p. 260-277, Oct. 1995.

JUANG, J. N.; PAPPA, R. S. An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction. **Journal of Guidance Control and Dynamics**, Hampton, v. 8, n. 5, p. 620-627, Sept. 1985.

LIU, Y.; LOH, C.; NI, Y. Stochastic Subspace Identification for Output-Only Modal Analysis: application to super high-rise tower under abnormal loading condition. **Earthquake Engineering and Structural Dynamics**, Taiwan, v. 42, n. 4, p. 477-498, Jun. 2012.

LJUNG, L. **System Identification: theory for the user**. 2nd ed. Upper Saddle River: Prentice Hall, 1999.

_____. **System Identification Toolbox for use with MATLAB**. 8th ed. MathWorks, 2012.

LIMAS, L. FRAGA. **Determinação da Influência de Seções Transversais na Resposta Dinâmica de Pontes através de Ensaios em Túnel de Vento e Identificação de Sistemas**. 2007. 223 f. Tese (Doutorado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2007.

LUSCHER, D. J.; BROWNJOHN, J. M. W.; SOHN, H.; FARRAR, C. R. Modal Parameter Extraction of Z24 Bridge Data. In: International Modal Analysis Conference, 19th, 2001, Kissimmee, Florida. **Proceedings...** Kissimmee: IMAC, não paginado, Feb. 2001.

KEESMAN, K. J. **System Identification: an introduction**. London: Springer, 2011.

KOZIN, F.; NATKE, H. G. System Identification Techniques. **Structural Safety**, Amsterdam, v. 3, p. 269-316, 1986.

KRÄMER, C. **Brite-Euram Project SIMCES, Task A1 and A2: long term monitoring and bridge tests**. Technical report 168'349/20e, EMPA, Dübendorf, Switzerland, Jan. 1999.

MARCHESIELLO, S.; PIMBO, B. A. D.; SORRENTINO, S. Application of the CVA-BR Method to the Z24 Bridge Vibration Data. In: International Modal Analysis Conference, 19th, 2001, Kissimmee, Florida. **Proceedings...** Kissimmee: IMAC, p. 864-869, Feb. 2001.

MELBOURNE, W. H. Comparison of Measurements on the CAARC Standard Tall Building Model in Simulated Model Wind Flows. **Journal of Wind Engineering and Industrial Aerodynamics**, Clayton, v. 6, n. 1-2, p. 73-88, July 1980.

MENDES, P.; OLIVEIRA, S.; GUERREIRO, L.; BAPTISTA, M. A.; CAMPOS COSTA, A. Dynamic Behaviour of Concrete Dams Monitoring and Modeling. In: World Conference on Earthquake Engineering, 13th, 2004, Vancouver. **Proceedings...** Vancouver: WCEE, não paginado, 2004.

MIGUEL, L. F. F. **Identificação de Sistemas e Avaliação da Integridade de Estruturas Treliçadas**. 2007. 160 f. Tese (Doutorado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2007.

MIGUEL, L. F. F.; MENEZES, R. C. R. ; MIGUEL, L. F. F. Identificação Estocástica de Subespaços de Torres de Linhas de Transmissão de Alta Voltagem. **Revista Sul-Americana de Engenharia Estrutural**, Passo Fundo, v. 4, p. 75-92, Ago. 2007.

MIGUEL, L. F. F.; MIGUEL, L. F. F.; THOMAS, C. A. K. Theoretical and experimental modal analysis of a cantilever steel beam with a tip mass. **Journal of Mechanical Engineering**, Porto Alegre, v. 223, p. 1535-1541, Jan. 2009.

MULITERNO, B. K. **Avaliação do fator de amplificação dinâmica em pontes rodoviárias utilizando smartphones**. 2020. 170 f. Dissertação (Mestrado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil e Ambiental, Universidade de Passo Fundo, Passo Fundo, 2020.

OLIVEIRA, M. G. K. **Desenvolvimento de uma balança dinâmica de três graus de liberdade para estudo dos efeitos de flexo-torção em edifícios altos submetidos à ação do vento**. 2009. 207 f. Tese (Doutorado em Engenharia) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2009.

OVERSCHEE, P. V.; MOOR, B. D. **Subspace identification for linear systems: theory – implementation – applications**. Leuven: Kluwer, 1996.

PEETERS, B. **System Identification and Damage Detection in Civil Engineering**. 2000. 238 f. Proefschrift (Doctoraat in de toegepaste wetenschappen) – Departement Burgerlijke Bouwkunde, Katholieke Universiteit Leuven, Leuven, 2000.

PEETERS, B.; MAECK, J.; DE ROECK, G. Excitation Sources and Dynamic System Identification in Civil Engineering. In: European COST F3 Conference on System Identification and Structural Health Monitoring, 2000, Madrid. **Proceedings...** Madrid: COST, p. 341-350, June 2000.

PEETERS, B.; VAN DER AUWERAER, H. PolyMAX: a revolution in operational modal analysis. In: International Operational Modal Analysis Conference, 1st, 2005, Copenhagen. **Proceedings...** Copenhagen: IOMAC, não paginado, 2005.

ROCHA, M. M.; LOREDO-SOUZA, A. M. Integração de pressões em alta frequência para o cálculo da resposta dinâmica de estruturas sujeitas à ação do vento. Laboratório de Aerodinâmica das Construções. Universidade Federal do Rio Grande do Sul. **Relatório Técnico**. Porto Alegre, 2008.

RODRIGUES, J. **Identificação Modal Estocástica: métodos de análise e aplicações em estruturas de engenharia civil**. 2004. 484 f. Dissertação (Doutorado em Engenharia Civil) – Faculdade de Engenharia, Universidade do Porto, Porto, 2004.

SHIH, C. Y.; TSUEI, Y. G.; ALLEMANG, R. J.; BROWN, D. L. Complex Mode Indication Function and its Applications to Spatial Domain Parameter Estimation. **Mechanical Systems and Signal Processing**, Cincinnati, v. 2, n. 4, p. 367-377, May 1988.

SCHOUKENS, J.; PINTELON, R. **Identification of Linear Systems: a practical guideline to accurate modeling**. Brussels: Pergamon, 1991.

SPITZNOGLE, F. R.; QUAZI, A. H. Representation and Analysis of Time-Limited Signals Using a Complex Exponential Algorithm. **The Journal of Acoustical Society of America**, Dallas, v. 47, n. 5, p. 1150-1155, 1970.

TRIFUNAC, M. D. Ambient Vibration Test of a Thirty-Nine Story Steel Frame Building. **Earthquake Engineering Research Report**, Pasadena, n. 70-02, July 1970.

VOLD, H.; KUNDRAT, J.; ROCKLIN, G. T.; RUSSEL, R. A Multi-Input Modal Estimation Algorithm for Mini-Computers. **SAE Technical Paper Series**, Detroit, v. 91, n. 1, p. 815-821, Jan 1982.

WELCH, P. D. The Use of Fast Fourier Transform for the Estimation of Power Spectra: a method based on time averaging over short, modified periodograms. **IEEE Transactions on Audio and Electroacoustics**, New York, v. 15, n. 2, p. 70-73, June 1967.

YAN, B.; MIYAMOTO, A. A Comparative Study of Modal Parameter Identification Based on Wavelet and Hilbert-Huang Transforms. **Computer-Aided Civil and Infrastructure Engineering**, Yamaguchi, Japan, v. 21, n. 1, p. 9-23, Jan. 2006.

ZENI, G. **Detecção de Dano em Estruturas utilizando Identificação Modal Estocástica e um Algoritmo de Otimização**. 2018. 73 f. Dissertação (Mestrado em Engenharia) –

Programa de Pós-Graduação em Engenharia Mecânica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2018.

ZHANG, L.; BRINCKER, R.; ANDERSEN, P. An Unified Approach for Two-Stage Time Domain Modal Identification. In: International Conference on Structural Dynamics Modelling, 2002, Madeira, Portugal. **Proceedings...** Madeira: ICUSD, não paginado, 2002.

ZHANG, L. On the Two-Stage Time Domain Modal Identification. In: International Modal Analysis Conference, 19th, 2003, Kissimmee. **Proceedings...** Kissimmee: IMAC, não paginado, 2003.

APÊNDICE A – CESSIPy: código fonte

```

1.  # -*- coding: utf-8 -*-
2.  """
3.  CESSIPy: um módulo em Python de código aberto para Identificação Modal
4.  Estocástica de Estruturas
5.
6.  Autor: Matheus Roman Carini
7.  Email para suporte: matheuscarini@gmail.com
8.  Site: https://github.com/MatheusCarini/CESSIPy
9.
10. Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil
11.
12. Versão: 1.0
13. Data: 20210413
14. """
15.
16. #=====
17. import sys
18. import numpy          as np
19. import matplotlib.pyplot as plt
20. import matplotlib     as mpl
21.
22. from MRPy              import MRPy
23. from scipy             import signal
24. from scipy.optimize   import curve_fit
25. from matplotlib.gridspec import GridSpec
26.
27. plt.rcParams["font.family"] = "Times New Roman"
28. mpl.rcParams['mathtext.fontset'] = 'cm'
29.
30. #=====
31. # Classe auxiliar
32. #=====
33.
34. class auxclass(np.ndarray):
35.     """
36.     Uma classe criada para permitir o uso de atributos
37.     Visa facilitar a programação e leitura do código
38.     """
39.
40.     def __new__(cls, np_array):
41.
42.         return np.asarray(np_array).view(cls)
43.
44. #=====
45. # Domínio do Tempo
46. #=====
47.
48. def rearrange_data(self,ref):
49.     """
50.     Rearranja as l saídas medidas posicionando as r saídas de referência
51.     nas primeiras linhas.
52.
53.     Entradas
54.     -----
55.     self : MRPy_like
56.           Um MRPy que contém as séries temporais das saídas.
57.     ref: tuple, list
58.           Lista dos sensores de referência.
59.
60.     Saídas
61.     -----
62.     yk : MRPy_like
63.           Um MRPy com as saídas de referência nas primeiras linhas e com os
64.           atributos r e l.
65.     .. l : atributo do MRPy

```

```

66.         Número de saídas
67.     .. r : atributo do MRPy
68.         Número de saídas de referência.
69.     """
70.
71.     r = len(ref)
72.     l = self.shape[0]
73.
74.     yk = MRPy(np.empty((l,self.N)),fs=self.fs)
75.     yk.r = r
76.     yk.l = l
77.
78.     yk[:,r,:] = self[ref,:]
79.     yk[r:,:] = np.delete(self, ref, 0)
80.
81.     return yk
82.
83. #-----
84.
85. def Toeplitz(self, i):
86.     """
87.     Cria a matriz bloco Toeplitz, a qual reúne as covariâncias das saídas
88.     estimadas até 2*i-1 time lags.
89.
90.     Entradas
91.     -----
92.     self : MRPy_like
93.         Um MRPy que contém as séries temporais das saídas.
94.     i : int
95.         Número de atrasos no tempo associado com a duração das funções de
96.         covariâncias. Note que elas são estimadas até 2*i-1 time lags.
97.
98.     Saídas
99.     -----
100.    T : auxclass_like
101.        Um auxclass que contém os valores da matriz bloco Toeplitz e os
102.        atributos r, l e i.
103.    """
104.
105.    N = self.N - 2*i + 1
106.    r = self.r
107.    l = self.l
108.
109.    Ypref = np.zeros((r*i,N))
110.    Yf     = np.zeros((l*i,N))
111.
112.    for k in range(i):
113.        Ypref[k*r:k*r+r,:] = self[:,k:k+N]
114.        Yf [k*l:k*l+l,:] = self[:,k+i:k+i+N]
115.
116.    Ypref = Ypref/N**0.5
117.    Yf     = Yf     /N**0.5
118.
119.    T = auxclass(Yf @ Ypref.T)
120.
121.    T.fs, T.r, T.l, T.i = self.fs, r, l, i
122.
123.    return T
124.
125. #-----
126.
127. def SSI_COV(T, no):
128.     """
129.     Método de Identificação Estocástica de Subespaços baseado nas
130.     Covariâncias

```

```

131.
132.     Estima as frequências naturais, razões de amortecimentos e formas modais
133.     a partir da matriz bloco Toeplitz informada.
134.
135.     Entradas
136.     -----
137.     T : auxclass_like
138.         Um auxclass que contém os valores da matriz bloco Toeplitz e os
139.         atributos SVD, r, l e i.
140.     no : int
141.         Número de ordem do modelo no espaço de estado.
142.
143.     Saídas
144.     -----
145.     fn : ndarray
146.         Array com as frequências naturais.
147.     zt : ndarray
148.         Array com as razões de amortecimento.
149.     V : ndarray
150.         Array com as formas modais nas colunas.
151.
152.     Ver também
153.     -----
154.     Toeplitz, SSI_COV_iterator
155.     """
156.
157.     l = T.l
158.     i = T.i
159.     U, S, VT = T.SVD
160.
161.     U1 = U[:,no]
162.     S1 = np.eye(no)*S[:,no]
163.     Oi = U1 @ S1**0.5
164.     C = Oi[:,:]
165.
166.     A = np.linalg.pinv(Oi[:,1*(i-1),:]) @ Oi[:,1:i+1,:])
167.     Ad, Psi = np.linalg.eig(A)
168.
169.     lambda_ = np.log(Ad)*T.fs
170.     fn = np.abs(lambda_)/(2*np.pi)
171.     zt = -np.real(lambda_)/np.abs(lambda_)
172.     V = C @ Psi
173.
174.     return fn, zt, V
175.
176. #-----
177.
178. def SSI_COV_iterator(yk,i,nmin,nmax,incr=2, plot=False):
179.     """
180.     Itera a função SSI_COV para modelos de ordem nmin a nmax com
181.     incrementos incr.
182.
183.     Estima as frequências naturais, razões de amortecimentos e formas modais
184.     pelo método SSI COV para os números de ordem informados.
185.
186.     Entradas
187.     -----
188.     yk : MRPy_like
189.         Um MRPy com as séries temporais das saídas arranjadas pela função
190.         rearrange_data.
191.     i : int
192.         Número atrasos no tempo associado com a duração das funções de
193.         covariâncias. Note que elas são estimadas até 2*i-1 time lags.
194.     nmin : int
195.         Número de ordem inicial do modelo no espaço de estado.

```

```

196.     nmax : int
197.         Número de ordem do modelo no espaço de estado.
198.     incr : int, optional
199.         Variação da ordem em relação à iteração anterior. Padrão = 2.
200.     plot : bool, optional
201.         Se verdadeiro, gera o gráfico dos valores singulares da matriz
202.         Toeplitz. O padrão é falso.
203.
204.     Saídas
205.     -----
206.     FN : ndarray
207.         Array 2D com as frequências naturais. Cada linha origina-se do
208.         mesmo modelo em espaço de estado.
209.     ZT : ndarray
210.         Array 2D com as frequências naturais. Cada linha origina-se do
211.         mesmo modelo em espaço de estado.
212.     VV : ndarray
213.         Array 3D com as formas modais. A primeira dimensão refere-se à
214.         ordem do modelo em espaço de estado.
215.
216.     Notas
217.     -----
218.     As propriedades dinâmicas do modelo de ordem nmin estão em FN[0,:],
219.     ZT[0,:] e VV[0,:,:].
220.     """
221.
222.     T = Toeplitz(yk, i)
223.     T.method = 'SSI COV'
224.     if plot: plot_singular_values(T)
225.     T.SVD = np.linalg.svd(T)
226.
227.     n = np.arange(nmin, nmax+incr, incr)
228.     FN = np.zeros((n.shape[0], nmax))
229.     ZT = np.zeros((n.shape[0], nmax))
230.     VV = np.zeros((n.shape[0], T.l, nmax), dtype=np.complex_)
231.
232.     for ii, no in np.ndenumerate(n):
233.         FN[ii, :no], ZT[ii, :no], VV[ii, :, :no] = SSI_COV(T, no)
234.
235.     return FN, ZT, VV
236.
237. #-----
238.
239. def projection(yk, i):
240.     """
241.     Realiza a fatoração QR da matriz Hankel e calcula as matrizes Piref,
242.     Pilref e Yii.
243.
244.     Entradas
245.     -----
246.     yk : MRPy_like
247.         Um MRPy com as séries temporais das saídas arranjadas pela função
248.         rearrange_data.
249.     i : int
250.         Número atrasos no tempo associado com a duração das funções de
251.         covariâncias. Note que elas são estimadas até 2*i-1 time lags.
252.
253.     Saídas
254.     -----
255.     Pi : auxclass_like
256.         Um auxclass com as projeções das saídas futuras no espaço das linhas
257.         das saídas passadas e os atributos r, l e i.
258.     Pil : array_like
259.         Array com as projeções considerando a separação entre as saídas
260.         passadas e futuras uma linha para baixo.

```

```

261.     Yii : array_like
262.         Array com parte da matriz bloco Hankel.
263.     """
264.
265.     N = yk.N - 2*i + 1
266.     r = yk.r
267.     l = yk.l
268.
269.     Ypref = np.zeros((r*i,N))
270.     Yf     = np.zeros((l*i,N))
271.
272.     for k in range(i):
273.         Ypref[k*r:k*r+r,:] = yk[:,k:k+N]
274.         Yf     [k*l:k*l+l,:] = yk[:,k+i:k+i+N]
275.
276.     Ypref = Ypref/N**0.5
277.     Yf     = Yf     /N**0.5
278.     Href  = np.vstack([Ypref,Yf])
279.
280.     R = np.linalg.qr(Href.T, mode='r').T
281.
282.     Pi = auxclass(R[r*i:,r*i:] @ np.eye(r*i,N))
283.     Pi1 = R[r*i+l:,r*i+r] @ np.eye(r*i+r,N)
284.     Yii = R[r*i:r*i+l,r*i+l] @ np.eye(r*i+l,N)
285.
286.     Pi.fs, Pi.r, Pi.l, Pi.i = yk.fs, r, l, i
287.
288.     return Pi, Pi1, Yii
289.
290. #-----
291.
292. def SSI_DATA(Pi, Pi1, Yii, no):
293.     """
294.     Método de Identificação Estocástica de Subespaços baseado nas Séries
295.     Temporais
296.
297.     Estima as frequências naturais, amortecimentos e formas modais a partir
298.     das matrizes Piref, Pi1ref e Yii.
299.
300.     Entradas
301.     -----
302.     Pi, Pi1, Yii
303.         Ver projection.
304.     no : int
305.         Número de ordem do modelo no espaço de estado.
306.
307.     Saídas
308.     -----
309.     fn : ndarray
310.         Array com as frequências naturais.
311.     zt : ndarray
312.         Array com as razões de amortecimento.
313.     V : ndarray
314.         Array com as formas modais nas colunas.
315.     """
316.
317.     U, S, VT = Pi.SVD
318.
319.     U1 = U[:,no]
320.     S1 = np.eye(no)*S[:no]
321.
322.     Oi = U1 @ S1**0.5
323.     Oi1 = Oi[:-Pi.l,:]
324.
325.     Xi = np.linalg.pinv(Oi) @ Pi

```

```

326.     Xi1 = np.linalg.pinv(Oi1) @ Pi1
327.
328.     AC = np.vstack([Xi1, Yii]) @ np.linalg.pinv(Xi)
329.     A = AC[:no, :]
330.     C = AC[no:, :]
331.
332.      $\Lambda_d, \Psi = \text{np.linalg.eig}(A)$ 
333.
334.      $\lambda = \text{np.log}(\Lambda_d) * \text{Pi.fs}$ 
335.      $\text{fn} = \text{np.abs}(\lambda) / (2 * \text{np.pi})$ 
336.      $\text{zt} = -\text{np.real}(\lambda) / \text{np.abs}(\lambda)$ 
337.      $V = C @ \Psi$ 
338.
339.     return fn, zt, V
340.
341. #-----
342.
343. def SSI_DATA_iterator(yk, i, nmin, nmax, incr=2, plot=False):
344.     """
345.     Itera a função SSI_DATA para modelos de ordem nmin a nmax com
346.     incrementos incr.
347.
348.     Estima as frequências naturais, razões de amortecimentos e formas modais
349.     pelo método SSI DATA para os números de ordem informados.
350.
351.     Entradas
352.     -----
353.     yk : MRPy_like
354.         Um MRPy com as séries temporais das saídas arranjadas pela função
355.         rearrange_data.
356.     i : int
357.         Número atrasos no tempo associado com a duração das funções de
358.         covariâncias. Note que elas são estimadas até 2*i-1 time lags.
359.     nmin : int
360.         Número de ordem inicial do modelo no espaço de estado.
361.     nmax : int
362.         Número de ordem do modelo no espaço de estado.
363.     incr : int, optional
364.         Variação da ordem em relação à iteração anterior. Padrão = 2.
365.     plot : bool, optional
366.         Se verdadeiro, gera o gráfico dos valores singulares da matriz
367.         das projeções. O padrão é falso.
368.
369.     Saídas
370.     -----
371.     FN : ndarray
372.         Array 2D com as frequências naturais. Cada linha origina-se do
373.         mesmo modelo em espaço de estado.
374.     ZT : ndarray
375.         Array 2D com as frequências naturais. Cada linha origina-se do
376.         mesmo modelo em espaço de estado.
377.     WV : ndarray
378.         Array 3D com as formas modais. A primeira dimensão refere-se à
379.         ordem do modelo em espaço de estado.
380.
381.     Notas
382.     -----
383.     As propriedades dinâmicas do modelo de ordem nmin estão em FN[0, :],
384.     ZT[0, :] e WV[0, :, :].
385.     """
386.
387.     Pi, Pi1, Yii = projection(yk, i)
388.     Pi.method = 'SSI DATA'
389.     if plot: plot_singular_values(Pi)
390.     Pi.SVD = np.linalg.svd(Pi)

```

```

391.
392.     n = np.arange(nmin,nmax+incr,incr)
393.     FN = np.zeros((n.shape[0],nmax))
394.     ZT = np.zeros((n.shape[0],nmax))
395.     WV = np.zeros((n.shape[0],Pi.l,nmax),dtype=np.complex_)
396.
397.     for ii, no in np.ndenumerate(n):
398.         FN[ii,:no],ZT[ii,:no],WV[ii,:,:no] = SSI_DATA(Pi,Pi1,Yii,no)
399.
400.     return FN, ZT, WV
401.
402. #-----
403.
404. def IV(T,no):
405.     """
406.     Método das Variáveis Instrumentais
407.
408.     Estima as frequências naturais, razões de amortecimentos e formas modais
409.     a partir da matriz bloco Toeplitz informada.
410.
411.     Entradas
412.     -----
413.     T : auxclass_like
414.         Um auxclass que contém os valores da matriz bloco Toeplitz e os
415.         atributos r, l e i.
416.     no : int
417.         Número de ordem do modelo no espaço de estado.
418.
419.     Saídas
420.     -----
421.     fn : ndarray
422.         Array com as frequências naturais
423.     zt : ndarray
424.         Array com as razões de amortecimento
425.     V : ndarray
426.         Array com as formas modais nas colunas
427.
428.     Ver também
429.     -----
430.     Toeplitz
431.     """
432.
433.     r = T.r
434.     l = T.l
435.
436.     ab = np.linalg.lstsq(T[:,-no*r:],
437.                          -T[:,-(no+1)*r:-no*r], rcond=None)[0]
438.
439.     Apcomp = np.zeros((no*r,no*r))
440.     Apcomp[:-r,r:] += np.eye((no-1)*r)
441.     for kk in range(no):
442.         Apcomp[-r:,r*kk:r*(kk+1)] -= ab.T[:,-r*(no-kk)-r:r*(no-kk)]
443.
444.     Ad, Psi = np.linalg.eig(Apcomp)
445.
446.     lambda_ = np.log(Ad)*T.fs
447.     fn = np.abs(lambda_)/(2*np.pi)
448.     zt = -np.real(lambda_)/np.abs(lambda_)
449.
450.     Gmref = (Psi[:r,:]).T
451.     Gamma_ref = np.zeros((no*r,no*r),dtype=np.complex_)
452.
453.     for ii in range(no):
454.         Gamma_ref[:,ii*r:(ii+1)*r] = np.diag(Ad**(no-ii-1)) @ Gmref
455.

```

```

456.     V = T[:l,-no*r:] @ np.linalg.inv(Γmref)
457.
458.     return fn, zt, V
459.
460. #-----
461.
462. def IV_iterator(yk, i, nmin, nmax, incr=2, plot=False):
463.     """
464.     Itera a função IV para modelos de ordem nmin a nmax com incrementos
465.     incr.
466.
467.     Estima as frequências naturais, razões de amortecimentos e formas modais
468.     pelo método IV para os números de ordem informados.
469.
470.     Entradas
471.     -----
472.     yk : MRPy_like
473.         Um MRPy com as séries temporais das saídas arranjadas pela função
474.         rearrange_data.
475.     i : int
476.         Número de atrasos no tempo associado com a duração das funções de
477.         covariâncias. Note que elas são estimadas até 2*i-1 time lags.
478.     nmin : int
479.         Número de ordem inicial do modelo no espaço de estado.
480.     nmax : int
481.         Número de ordem do modelo no espaço de estado.
482.     incr : int, optional
483.         Variação da ordem em relação à iteração anterior. Padrão = 2.
484.     plot : bool, optional
485.         Se verdadeiro, gera o gráfico dos valores singulares da matriz
486.         Toeplitz. O padrão é falso.
487.
488.     Saídas
489.     -----
490.     FN : ndarray
491.         Array 2D com as frequências naturais. Cada linha origina-se do
492.         mesmo modelo em espaço de estado.
493.     ZT : ndarray
494.         Array 2D com as frequências naturais. Cada linha origina-se do
495.         mesmo modelo em espaço de estado.
496.     VV : ndarray
497.         Array 3D com as formas modais. A primeira dimensão refere-se à
498.         ordem do modelo em espaço de estado.
499.
500.     Notas
501.     -----
502.     Observe que um modelo ARMA de ordem p tem p * r polos
503.     As propriedades dinâmicas do modelo de ordem nmin estão em FN[0,:],
504.     ZT[0,:] e VV[0,:,:].
505.     """
506.
507.     T = Toeplitz(yk,i)
508.     T.method = 'IV'
509.     if plot: plot_singular_values(T)
510.
511.     n = np.arange(nmin, nmax+incr, incr)
512.     FN = np.zeros((n.shape[0], nmax*T.r))
513.     ZT = np.zeros((n.shape[0], nmax*T.r))
514.     VV = np.zeros((n.shape[0], T.l, nmax*T.r), dtype=np.complex_)
515.
516.     for ii, no in np.ndenumerate(n):
517.         FN[ii,:no*T.r], ZT[ii,:no*T.r], VV[ii,:,:no*T.r] = IV(T, no)
518.
519.     return FN, ZT, VV
520.

```

```

521. #-----
522.
523. def stabilization_diagram(FN, ZT, VV, title,
524.                          tol = np.array([[0.01,0, 100],
525.                                           [0.05,0,0.05],
526.                                           [0.10,0, 1]]), plot=True):
527.     """
528.     Plota o diagrama de estabilização a partir das matrizes com as
529.     frequências naturais, razões de amortecimentos e formas modais.
530.
531.     Entradas
532.     -----
533.     FN, ZT, VV
534.     Propriedades dinâmicas fornecidas pelas funções SSI_COV_Iterator,
535.     SSI_DATA_Iterator ou IV_Iterator.
536.     title : str
537.     Título do gráfico.
538.     tol : ndarray, optional
539.     Tolerâncias, valores mínimos e máximos utilizados para indicar a
540.     estabilidade do pivô. A primeira linha refere-se à frequência, a
541.     segundo às razões de amortecimento e a terceira aos coeficientes de
542.     MAC. O padrão é
543.     [0.01,0,100 ] Δf = 1%; fmin = 0 Hz; fmax = 100 Hz
544.     [0.05,0,0.05] Δζ = 5%; ζmin = 0%; ζmax = 5%
545.     [0.10,0,1   ] MAC >= (1 - 0.10) = 0.90
546.     plot : bool, optional
547.     Se verdadeiro, gera o gráfico do diagrama de estabilização.
548.     O padrão é falso.
549.
550.     Saídas
551.     -----
552.     stb : array_like
553.     Array booleano indicando os polos estáveis como verdadeiro. Cada
554.     linha origina-se do mesmo modelo em espaço de estado.
555.
556.     Notas
557.     -----
558.     Os polos estáveis do modelo de ordem nmax estão em stb[-1,:].
559.     """
560.
561.     nmin = np.count_nonzero(FN, axis=1)[0]
562.     nmax = np.count_nonzero(FN, axis=1)[-1]
563.     incr = (nmax-nmin)/(FN.shape[0]-1)
564.     n     = np.arange(nmin,nmax+incr,incr)
565.     stb  = np.full(FN.shape, False)
566.     stbf = np.full(FN.shape, False)
567.     stbz = np.full(FN.shape, False)
568.
569.     for ii in range(1,n.shape[0]):
570.
571.         no = n[ii]; ia = ii - 1
572.         na = n[ia]
573.
574.         # Frequências
575.
576.         b1 = (FN[ii,:no] >= tol[0,1]) & (FN[ii,:no] <= tol[0,2])
577.         dif = FN[ia,:na] - FN[ii,:no].reshape(-1,1)
578.         ind = np.abs(dif).argmin(axis=1)
579.         res = np.diagonal(dif[:,ind])
580.         b1 = (np.abs(res/FN[ii,:no]) < tol[0,0]) & b1
581.
582.         # Amortecimento
583.
584.         b2 = (ZT[ii,:no] >= tol[1,1]) & (ZT[ii,:no] <= tol[1,2])
585.         dif = ZT[ia,:na] - ZT[ii,:no].reshape(-1,1)

```

```

586.     res = np.diagonal(dif[:,ind])
587.     b2 = (np.abs(res/ZT[ii,:no]) < tol[1,0]) & b2 & b1
588.
589.     # MAC
590.
591.     MCv = MAC(VV[ia,:na],VV[ii,:no])
592.     res = np.abs(np.diag(MCv[ind,:]))
593.     b3 = (res > 1 - tol[2,0]) & b2
594.
595.     stbf[ii,:no] = b1
596.     stbz[ii,:no] = b2
597.     stb [ii,:no] = b3
598.
599.     if plot:
600.
601.         a_for = {'fontname':'Times New Roman','size':16}
602.         l_for = {'fontname':'Times New Roman','size':14}
603.         t_for = {'fontname':'Times New Roman','size':12}
604.         g_for = {'family' : 'Times New Roman','size':12}
605.
606.         plt.figure(figsize=(10,5))
607.
608.         for ii in range(n.shape[0]):
609.
610.             yi = n[ii]*np.ones(n[ii])
611.             ko = plt.scatter(FN[ii,:n[ii]],yi,s=2,c='k')
612.             go = plt.scatter(FN[ii,:n[ii]][stbf[ii,:n[ii]]],
613.                             yi[stbf[ii,:n[ii]]],s=4,c='g')
614.             bo = plt.scatter(FN[ii,:n[ii]][stbz[ii,:n[ii]]],
615.                             yi[stbz[ii,:n[ii]]],s=4,c='b')
616.             ro = plt.scatter(FN[ii,:n[ii]][stb [ii,:n[ii]]],
617.                             yi[stb [ii,:n[ii]]],s=8,c='r')
618.
619.             plt.xlim((0,tol[0,2]))
620.             plt.ylim((0,n[-1]))
621.             plt.xticks(**t_for)
622.             plt.yticks(n,**t_for)
623.             plt.xlabel('f (Hz)',**l_for)
624.             plt.ylabel('Ordem do Modelo',**l_for)
625.             plt.suptitle(title + ' Diagrama de Estabilização',**a_for)
626.             plt.legend([ko, go, bo, ro],
627.                       ["Novo polo",
628.                        "Frequência estável",
629.                        "Frequência e amortecimento estáveis",
630.                        "Frequência, amortecimento e forma modal estáveis"],
631.                       prop=g_for)
632.             plt.tight_layout(rect=[0, 0, 1, 0.97])
633.
634.         return stb
635.
636. #-----
637.
638. def stable_modes(FN, ZT, V, stb, tol=0.01, spo=6):
639.     """
640.     Automatiza o processo de escolha dos modos estruturais.
641.
642.     Analisa os pontos estáveis do diagrama de estabilização e retorna as
643.     características dinâmicas daqueles que apresentarem pelo menos spo/2
644.     polos para uma mesma frequência.
645.
646.     Entradas
647.     -----
648.     FN, ZT, V
649.     Propriedades dinâmicas fornecidas pelas funções SSI_COV_Itetator,
650.     SSI_DATA_Itetator ou IV_Itetator.

```

```

651.     stb : array_like
652.         Array booleano indicando os polos estáveis como verdadeiro,
653.         fornecida pela função stabilization_diagram.
654.     tol : float
655.         Tolerância para frequência ser considerada do mesmo modo.
656.     spo : int
657.         Número mínimo de modelos com polos estáveis para que o modo seja
658.         considerado estrutural.
659.
660.     Saídas
661.     -----
662.     fn : ndarray
663.         Array com as frequências naturais.
664.     zt : ndarray
665.         Array com as razões de amortecimento.
666.     vv : ndarray
667.         Array 2D com as formas modais como colunas
668.
669.     Notas
670.     -----
671.     Como as propriedades dinâmicas estão em pares, spo seleciona spo/2
672.     polos estáveis
673.     """
674.
675.     FN = FN[stb]
676.     ZT = ZT[stb]
677.     VV = V[0,:,stb[0]].T
678.
679.     for j in range(stb.shape[0]):
680.         VV = np.hstack((VV,V[j,:,stb[j]].T))
681.
682.     fsi = np.argsort(FN)
683.
684.     FNs, ZTs, VVs = FN[fsi], ZT[fsi], VV[:,fsi]
685.
686.     fn, zt, v = [], [], V[0,:,stb[0]].T
687.
688.     k = 0
689.
690.     for i in range(len(FN)):
691.
692.         b0 = (FNs > (1-tol)*FNs[k]) & (FNs < (1+tol)*FNs[k])
693.
694.         if b0.sum() >= spo:
695.
696.             fn = np.append(fn, (FNs[b0]).mean())
697.             zt = np.append(zt, (ZTs[b0]).mean())
698.
699.             mv = np.argmax(np.abs(VVs[:,b0]),axis=0)[0]
700.             nv = np.mean(VVs[:,b0]/VVs[mv,b0],axis=1).reshape(-1,1)
701.             v = np.hstack((v,nv))
702.
703.             k += b0.sum()
704.
705.             if k > len(FN)-1: break
706.
707.     return fn, zt, v
708.
709. #-----
710.
711. def plot_singular_values(T, figsize=(14, 4), nmx=40):
712.     """
713.     Calcula e plota os valores singulares da matriz informada.
714.
715.     Entradas

```

```

716.     -----
717.     T : auxclass_like
718.         Um auxclass com a matriz e atributo method.
719.     figsize : tuple, optional
720.         Tamanho do gráfico. Padrão é (14,4).
721.     nmx : int, optional
722.         Número de valores singulares a serem plotados.
723.     """
724.
725.     a_for = {'size':16}
726.     l_for = {'size':16}
727.
728.     S = np.linalg.svd(T, compute_uv=False)[:nmx]
729.     idx = np.argmax(S[1:]/S[:-1])
730.
731.     fig, ax = plt.subplots(1, 3,figsize=figsize)
732.     fig.suptitle('%s Valores Singulares' %(T.method), **a_for)
733.
734.     label = ['\n(a) valores singulares',
735.             'Ordem do Modelo\n(b) normalizados pelo primeiro',
736.             '\n(c) normalizados pelo anterior']
737.
738.     ax[0].plot(np.arange(1,nmx+1),S,'bo',ms=4)
739.     ax[0].set_ylabel('Valores Singulares', **l_for)
740.     ax[0].set_ylim(bottom=0)
741.
742.     ax[1].semilogy(np.arange(1,nmx+1),S/S[0],'b',idx+1,(S/S[0])[idx],'ro')
743.     ax[1].annotate('%0f' %(idx+1),(idx+1.5,(S/S[0])[idx]),**l_for)
744.
745.     ax[2].semilogy(np.arange(1,nmx+1), np.hstack((1,S[1:]/S[:-1])), 'b',
746.                   idx+1,(S[1:]/S[:-1])[idx-1],'ro')
747.     ax[2].annotate('%0f' %(idx+1),(idx+1.5,(S[1:]/S[:-1])[idx-1]),**l_for)
748.
749.     for i in range(3):
750.         ax[i].set_xticks(np.linspace(0,nmx,nmx//2+1))
751.         ax[i].tick_params(labelsize=12)
752.         ax[i].set_xlim((0,nmx))
753.         ax[i].set_xlabel(label[i], **l_for)
754.
755.     fig.tight_layout(rect=[0, 0, 1, 0.97])
756.
757.     return
758.
759.     #=====
760.     # Domínio da Frequência
761.     #=====
762.
763. def SDM(self, nperseg=None, plot=False, window='hann', nfft=None,
764.         figsize=(10,10)):
765.     """
766.     Calcula a matriz das densidades espectrais através da função signal.csd
767.
768.     Os sinais podem ser divididos em segmentos com nperseg valores visando
769.     obter estimativas suavizadas.
770.
771.     Entradas
772.     -----
773.     self : MRPy_like
774.         Um MRPy com as séries temporais.
775.     nperseg : int, optional
776.         Comprimento de cada segmento utilizado para das densidades
777.         espectrais suavizadas. O padrão é o comprimento do sinal.
778.     plot : bool, optional
779.         Se verdadeira, gera o gráfico das densidades espectrais separadas
780.         em valor absoluto e fase. O padrão é falso.

```

```

781.     window : string, optional
782.         Janela de dados. O padrão é 'hann'.
783.     nfft : int, optional
784.         Comprimento da FFT. Utilizada caso deseja-se adicionar zeros na FFT. Se
785.         None, o comprimento da FFT é nperseg. O padrão é None.
786.     figsize : tuple, optional
787.         Tamanho do gráfico. Padrão é (12,12)
788.
789.     Saídas
790.     -----
791.     PSD : auxclass
792.         Um auxclass com as densidades espectrais, as frequências no atributo
793.         f e o comprimento do trecho no atributo nperseg.
794.
795.     Ver também
796.     -----
797.     scipy.signal.csd
798.     """
799.
800.     if nperseg is None: nperseg = self.N
801.     if nfft is None: nfft = nperseg
802.     if nfft < nperseg:
803.         raise ValueError('nfft must be greater than or equal to nperseg.')
804.
805.     G = np.empty((self.NX, self.NX, nfft//2+1), dtype=np.complex_)
806.
807.     for i in range(self.NX):
808.         for j in range(self.NX):
809.             f, G[i,j] = signal.csd(self[i,:], self[j,:], self.fs,
810.                 window=window, nperseg=nperseg, nfft=nfft)
811.
812.     if plot:
813.
814.         l_for = {'fontname':'Times New Roman','size':13}
815.
816.         axis_f = [0, f[-1]]
817.         #axis_G = [10**7*np.min(np.abs(G[:, :, 1:])), 1.2*np.max(np.abs(G))]
818.
819.         n = G.shape[0] # number of double-rows
820.         m = G.shape[1] # number of columns
821.         H = 3 # relação gráfica amplitude pela fase
822.         t = 0.9 # 1-t == top space
823.         b = 0.1 # bottom space (both in figure coordinates)
824.         w = 0.05 # side spacing
825.         μ = 0.1 # minor spacing
826.         M = 0.2 # major spacing
827.
828.         spa = (t-b)/(n*(1+μ+1/H)+(n-1)*M)
829.         offb = spa*(μ+1/H)
830.         offt = spa*(1+μ)
831.         hsp1 = M+μ+1/H
832.         hsp2 = (M+μ+1)*H
833.
834.         gso = GridSpec(n,m, bottom=b+offb, top=t, hspace=hsp1, wspace=w)
835.         gse = GridSpec(n,m, bottom=b, top=t-offt, hspace=hsp2, wspace=w)
836.
837.         fig = plt.figure(figsize=figsize)
838.
839.         for i in range(n*m):
840.
841.             ax1 = fig.add_subplot(gso[i])
842.             ax2 = fig.add_subplot(gse[i])
843.
844.             ax1.semilogy(f[1:], np.abs(G[i//m, i%m, 1:]), color='blue')
845.             ax1.set_xlim(axis_f)

```

```

846.         #ax1.set_ylim(axis_G)
847.         ax1.set_xticklabels([])
848.         ax1.set_title(r'$\hat{G}_y[{:d},{:d}]$'.format(i//m+1,i%m+1),
849.                       fontsize=15)
850.
851.         if i%m is not 0:
852.             ax1.set_yticklabels([])
853.         else:
854.             ax1.set_ylabel('Amplitude ((m/s2)2/Hz)',**l_for)
855.
856.         ax2.plot(f,np.angle(G[i//m, i%m]),color='blue')
857.         ax2.set_xlim(axis_f)
858.         ax2.set_ylim([-4,4])
859.
860.         if i%m is not 0:
861.             ax2.set_yticklabels([])
862.         else:
863.             ax2.set_ylabel('Fase (rad)',**l_for)
864.
865.         if i//m == n-1:
866.             ax2.set_xlabel('f (Hz)',**l_for)
867.         else:
868.             ax2.set_xticklabels([])
869.
870.         ax1.tick_params(labelsize=12)
871.         ax2.tick_params(labelsize=12)
872.
873.         plt.show()
874.
875.     PSD = auxclass(G)
876.     PSD.f, PSD.nperseg, PSD.nfft = f, nperseg, nfft
877.
878.     return PSD
879.
880. #-----
881.
882. def ANPSD_from_SDM(PSD, plot=False, mode='interactive'):
883.     """
884.     Calcula e plota o Espectro Normalizado Médio a partir da matriz de
885.     densidades espectrais.
886.
887.     Entradas
888.     -----
889.     PSD : auxclass_like
890.         Um auxclass com as densidades espectrais, as frequências no atributo
891.         f e o comprimento do trecho no atributo nperseg.
892.     plot : bool, optional
893.         Se verdadeira, gera o gráfico das densidades espectrais separadas
894.         em valor absoluto e fase. O padrão é falso.
895.     mode : string, optional
896.         Modo de detecção de picos. Se 'interactive', o usuário deve definir
897.         os picos com o mouse. Se 'batch', os picos são definidos pelo atributo
898.         pki. O padrão é 'interactive'.
899.
900.     Saídas
901.     -----
902.     PSD : auxclass_like
903.         Um auxclass com o Espectro Normalizado Médio no atributo ANPSD e
904.         os índices dos picos pki.
905.
906.     Modo batch
907.     -----
908.     Neste modo, o auxclass deve possuir o atributo pki.
909.     pki : list
910.         Lista dos índices das frequências dos picos.

```

```

911.     """
912.
913.     try:
914.         G = PSD.diagonal()
915.         f = PSD.f
916.
917.     except AttributeError:
918.         sys.exit('PSD deve ser obtido da função SDM')
919.
920.     NPSD = np.real((G / G.sum(axis=0)).T)
921.     ANPSD = np.real(NPSD.sum(axis=0)) / G.shape[1]
922.
923.     if mode.lower() == 'interactive':
924.
925.         plt.figure(figsize=(12,5))
926.         plt.semilogy(f,np.abs(ANPSD))
927.         plt.xlim([0, f[-1]])
928.         plt.xlabel('Frequência (Hz)')
929.         plt.ylabel('ANPSD ((m/s2)2/Hz)')
930.         #plt.ylim([1E6*ANPSD.min(),1.6*ANPSD.max()])
931.
932.         plt.title('Clique nos picos com o botão esquerdo\n'
933.                 'Finalize clicando com o botão central')
934.
935.         plt.text(.7,0.01,
936.                 'Selecionar: botão esquerdo\n'
937.                 'Desfazer: botão direito\n'
938.                 'Concluir: botão do meio',transform=plt.gca().transAxes)
939.
940.         plt.tight_layout(rect=[0, 0.03, 1, 0.97])
941.         pnt = plt.ginput(n=-1,timeout=0)
942.         fn = np.array(pnt)[: ,0]
943.         pki = np.argmin(np.abs(f-fn.reshape(-1,1)),axis=1)
944.
945.         plt.close()
946.
947.     elif mode.lower() == 'batch':
948.
949.         try:
950.             pki = PSD.pki
951.
952.         except AttributeError:
953.             sys.exit('PSD deve ter atributo pki no modo batch')
954.
955.     else:
956.         sys.exit('mode deve ser interactive ou batch')
957.
958.     if plot:
959.         plt.figure(figsize=(7,7))
960.         plt.subplot(211)
961.         plt.title('NPSD')
962.         for ii, row in enumerate(NPSD):
963.             plt.semilogy(f,row,label=ii+1)
964.         plt.legend()
965.         plt.xlim([0, f[-1]])
966.         plt.xlabel('Frequência (Hz)')
967.         plt.ylabel('Amplitude ((m/s2)2/Hz)')
968.         #plt.ylim([1E5*NPSD.min(),1.6*NPSD.max()])
969.
970.         plt.subplot(212)
971.         plt.title('ANPSD')
972.         plt.semilogy(f,np.abs(ANPSD))
973.         plt.xlim([0, f[-1]])
974.         plt.xlabel('Frequência (Hz)')
975.         plt.ylabel('Amplitude ((m/s2)2/Hz)')

```

```

976.         #plt.ylim([1E5*ANPSD.min(),1.6*ANPSD.max()])
977.         plt.tight_layout()
978.         plt.plot(f[pki], ANPSD[pki], "x")
979.
980.         for i in pki:
981.             plt.annotate('{:.3f} Hz'.format(f[i]),
982.                           (f[i],ANPSD[i]*1.08), ha='center')
983.         plt.show()
984.
985.     PSD.ANPSD = ANPSD
986.     PSD.pki   = pki
987.
988.     return PSD
989.
990. #-----
991. def coherence(self, PSD=None, nperseg=None, plot=False):
992.     """
993.     Calcula a matriz das funções de coerência
994.
995.     Entradas
996.     -----
997.     self : MRPy_like
998.         Um MRPy com as séries temporais.
999.     PSD : auxclass_like, optional
1000.         Um auxclass com o atributo pki.
1001.     nperseg : int, optional
1002.         Comprimento de cada segmento utilizado para as funções de coerência
1003.         suavizadas. O padrão é igual ao comprimento do sinal.
1004.     plot : bool, optional
1005.         Se verdadeira, gera o gráfico das funções de coerência.
1006.         O padrão é falso.
1007.
1008.     Saídas
1009.     -----
1010.      $\gamma$  : auxclass_like
1011.         Um auxclass das funções de coerência e as frequências no atributo f.
1012.
1013.     Ver também
1014.     -----
1015.     scipy.signal.coherence
1016.     """
1017.     if nperseg is None:
1018.         try:
1019.             nperseg = PSD.nperseg
1020.         except AttributeError:
1021.             sys.exit('nperseg deve ser informado ou atributo')
1022.
1023.      $\gamma$  = np.empty((self.NX,self.NX,nperseg//2+1))
1024.
1025.     for i in range(self.NX):
1026.         for j in range(self.NX):
1027.             f,  $\gamma$ [i,j] = signal.coherence(self[i,:], self[j:],
1028.                                             self.fs, nperseg=nperseg)
1029.
1030.      $\gamma$  = auxclass(np.real( $\gamma$ ))
1031.      $\gamma$ .f = f
1032.
1033.     if plot:
1034.
1035.         a_for = {'fontname':'Times New Roman','size':14}
1036.         l_for = {'fontname':'Times New Roman','size':12}
1037.         t_for = {'family':'Times New Roman','size':10}
1038.
1039.         NX = self.NX
1040.

```

```

1041.     plt.figure(figsize=(8,8))
1042.
1043.     for i in range(NX):
1044.         for j in range(NX):
1045.             ax = plt.subplot(NX,NX,i*NX+j+1)
1046.             ax.plot(f,y[i,j])
1047.             if PSD is not None: ax.plot(f[PSD.pki],y[i,j][PSD.pki], 'ro')
1048.             ax.set_title(r'$\gamma^2_{\{\{d\},\{d\}\}}$'
1049.                          .format(i+1,j+1),**a_for)
1050.             ax.set_xlim([0,f[-1]])
1051.             ax.set_ylim([0, 1.05])
1052.
1053.             if j is 0:
1054.                 #ax.set_yticklabels(np.linspace(0,1,6),**t_for)
1055.                 plt.yticks(**t_for)
1056.                 ax.set_ylabel('Coerência',**l_for)
1057.             else:
1058.                 ax.set_yticklabels([])
1059.
1060.             if i == NX - 1:
1061.                 plt.xticks(**t_for)
1062.                 ax.set_xlabel('f (Hz)',**l_for)
1063.             else:
1064.                 ax.set_xticklabels([])
1065.
1066.     plt.tight_layout()
1067.
1068.     return  $\gamma$ 
1069.
1070. #-----
1071.
1072. def BFD(self, PSD, plot=False, mode='interactive'):
1073.     """
1074.     Método Básico no Domínio da Frequência
1075.
1076.     Estima as frequências naturais, razões de amortecimentos e formas modais
1077.     a partir das densidades espectrais.
1078.
1079.     Estima as razões de amortecimento pelos métodos de meia-potência e
1080.     ajuste a um espectro analítico
1081.
1082.     Entradas
1083.     -----
1084.     self : MRPy_like
1085.         Um MRPy com as séries temporais.
1086.     PSD : auxclass_like
1087.         Um auxclass com as densidades espectrais e
1088.         as frequências no atributo f.
1089.     mode : string, optional.
1090.         Modo de operação. Se 'interactive', o usuário deve definir
1091.         parâmetros com o mouse. Se 'batch', estes parâmetros devem ser
1092.         atributos de PSD. O padrão é 'interactive'.
1093.
1094.     Saídas
1095.     -----
1096.     fn : ndarray
1097.         Array com as frequências naturais.
1098.     zt : list
1099.         Lista com as razões de amortecimento. zt[0] para coeficientes do
1100.         método de meia-potência e zt[1] para ajuste ao espectro analítico.
1101.     V : ndarray
1102.         Array com as formas modais nas colunas.
1103.     PSD : auxclass_like
1104.         Um auxclass com os atributos ANPSD, pki, MGi e fint.
1105.

```

```

1106.     Modo batch
1107.     -----
1108.     Neste modo, o PSD deve ter os atributos fint, MGi e pki.
1109.     fint : array
1110.         Array com as frequências inicial e final a serem utilizadas para
1111.         ajuste do espectro.
1112.     MGi : integer array
1113.         Array com os índices dos autoespectros a serem utilizados para
1114.         determinação das formas modais e amortecimentos.
1115.     pki : integer array
1116.         Array com os índices das frequências dos picos.
1117.     fint : array
1118.         Array com as frequências inicial e final a serem utilizadas para
1119.         ajuste do espectro.
1120.
1121.     Notas
1122.     -----
1123.     Antes de usar essa função, empregar as funções SDM e ANPSD_from_SDM.
1124.     """
1125.     try:
1126.         f, pki = PSD.f, PSD.pki
1127.         G      = np.abs(PSD.diagonal().T) # autoespectros
1128.     except AttributeError:
1129.         sys.exit('PSD deve ter os atributos f, nperseg e pki')
1130.
1131.     #------
1132.
1133.     if mode.lower() == 'interactive':
1134.
1135.         print('Selecione o autoespectro de referência')
1136.
1137.         global MGi, NX
1138.         MGi, NX = [], self.NX
1139.
1140.         def onclick_select(event): # obtém número do subplot selecionado
1141.             global MGi, NX
1142.             for i in range(NX):
1143.                 if event.inaxes == ax[0,i]:
1144.                     MGi = np.array(np.append(MGi,i),dtype='int')
1145.
1146.         for i, j in enumerate(pki):
1147.             fig, ax = plt.subplots(1, self.NX,figsize=(10, 4),sharey=True,squeeze=False)
1148.             plt.suptitle('Clique no autoespectro de referência\n'
1149.                         'a ser utilizado para amortecimento e forma modal')
1150.
1151.             for k in range(self.NX):
1152.                 ax[0,k].semilogy(f,G[k])
1153.                 ax[0,k].semilogy(f[j],G[k,j], 'ro')
1154.                 ax[0,k].set_xlim((0,f[-1]))
1155.                 ax[0,k].annotate('{:.2E}'.format(G[k,j]),(f[j],G[k,j]*1.05))
1156.
1157.             ax[0,0].set_ylabel('Amplitude ((m/s2)2/Hz)')
1158.             ax[0,k//2].set_xlabel('Frequência (Hz)')
1159.             plt.tight_layout(rect=[0, 0.03, 1, 0.92])
1160.             fig.canvas.mpl_connect('button_press_event', onclick_select)
1161.             plt.ginput(n=1,timeout=30)
1162.             plt.close()
1163.
1164.             fint = np.zeros(2*len(pki))
1165.
1166.             for i,(j,k) in enumerate(zip(pki, MGi)):
1167.                 plt.figure(figsize=(10,6))
1168.                 plt.title('Clique nos extremos do intervalo '
1169.                         'para ajuste ao espectro analítico')

```

```

1170.         plt.semilogy(f,G[k])
1171.         plt.semilogy(f[j],G[k][j], 'ro')
1172.         plt.annotate('{:.3f} Hz'.format(f[j]),(f[j],G[k][j]*1.15), ha='center')

1173.         #plt.xlabel(r'$f_n$ = {:.3f} Hz'.format(f[j]))
1174.         plt.xlabel('Frequência (Hz)')
1175.         plt.ylabel('Amplitude ((m/s2)2/Hz)')
1176.         plt.xlim([0,f[-1]])
1177.
1178.         pnt = np.array(plt.ginput(n=2,timeout=0))[:,0]
1179.
1180.         id1 = np.argmin(np.abs(f-pnt.reshape(-1,1)),axis=1)
1181.         fint[2*i:2*i+2] = f[id1]
1182.
1183.         plt.close()
1184.
1185.         PSD.fint = fint
1186.         PSD.MGi = MG_i
1187.
1188.         elif mode.lower() == 'batch':
1189.             try:
1190.                 MG_i = PSD.MGi
1191.                 fint = PSD.fint
1192.             except AttributeError:
1193.                 sys.exit('PSD deve ter atributo MG_i e fint no modo batch')
1194.
1195.         else:
1196.             sys.exit('mode deve ser interactive ou batch')
1197.
1198.         #-----
1199.         def Sy(f,c1,c2,fn,ζ):
1200.
1201.             return c1*np.abs(2*np.pi*f**2/(1-(f/fn)**2+2j*ζ*(f/fn)))**2 + c2
1202.         #-----
1203.
1204.         ζmp = np.zeros((len(pki)))
1205.         ζft = np.zeros((len(pki)))
1206.         P = np.zeros((len(pki),4))
1207.
1208.         idx = np.argmin(np.abs(f-fint.reshape(-1,1)),axis=1)
1209.
1210.         if plot: fig, ax = plt.subplots(1,len(MG_i),figsize=(len(MG_i)*4,3),squeeze=False)
1211.
1212.         for i, (j, k, ii, si) in enumerate(zip(MG_i,pki,idx[:,2],idx[1:2])):
1213.
1214.             mG = G[j,k]
1215.             fa = np.interp( mG/2, G[j,ii:k+1],f[ii:k+1])
1216.             fb = np.interp(-mG/2,-G[j, k:si ],f[ k:si ])
1217.             f0 = f[k]
1218.
1219.             ζmp[i] = (fb**2-fa**2)/(4*f[k]**2) # meia potência
1220.
1221.             Pmin = (0, 0, fa, 0.000) # lower bounds
1222.             P0 = (0, 0, f0, 0.010) # initial guesses
1223.             Pmax = (mG/1E2,mG/1E3, fb, 0.05 ) # upper bounds
1224.
1225.             P[i,:], _ = curve_fit(Sy,f[ii:si],G[j,ii:si],
1226.                                  p0=P0,bounds=(Pmin, Pmax))
1227.             ζft[i] = P[i,3]
1228.
1229.             if plot:
1230.                 ax[0,i].semilogy(f[ii:si],G[j,ii:si])
1231.                 ax[0,i].semilogy(np.linspace(f[ii],f[si],100),
1232.                                   Sy(np.linspace(f[ii],f[si],100),*P[i,:]),'k:')
1233.                 ax[0,i].plot(f0,mG,'rx')

```

```

1234.         ax[0,i].plot([fa,fb],[mG/2,mG/2], 'ro')
1235.         ax[0,i].annotate('{:.3f} Hz'.format(f0),(f0,G[j,k]*1.05),
1236.                             ha='center')
1237.         ax[0,i].annotate('{:.3f} Hz'.format(fa),(fa, mG/2*1.05),
1238.                             ha='right')
1239.         ax[0,i].annotate('{:.3f} Hz'.format(fb),(fb, mG/2*1.05),
1240.                             ha='left')
1241.         ax[0,i].text(.99, .99, r'$\xi_{\{mp\}}$ = {:.2f}%'.format( $\zeta_{mp}[i]*100$ ))
1242.         +'\n'+ r'$\xi_{\{ft\}}$ = {:.2f}%'.format( $\zeta_{ft}[i]*100$ ),
1243.         horizontalalignment='right',verticalalignment='top',
1244.         transform=ax[0,i].transAxes,fontsize=11)
1245.
1246.     if plot:
1247.         ax[0,0].set_ylabel('Amplitude ((m/s2)2/Hz)')
1248.         ax[0,i//2].set_xlabel('Frequência (Hz)')
1249.         ax[0,i].legend(['Densidade espectral', 'Função ajustada',
1250.             'Frequência natural', 'Frequência de meia-potência'])
1251.         fig.tight_layout()
1252.
1253.     fn = f[pki]
1254.     V = PSD[MGi,:,pki]/PSD[MGi,MGi,pki].reshape(-1,1)
1255.     V = np.abs(V)*(1-2*((np.angle(V)>np.pi/2)+(np.angle(V)<-np.pi/2)))
1256.
1257.     return fn, [ $\zeta_{mp}$ , $\zeta_{ft}$ ], V.T, PSD
1258.
1259. #-----
1260.
1261. def EFDD(self, PSD, plot=False, mode='interactive'):
1262.     """
1263.     Método de decomposição no domínio da frequência
1264.
1265.     Obtém as frequências naturais e formas modais a partir das densidades
1266.     espectrais.
1267.
1268.     Entradas
1269.     -----
1270.     self : MRPy_like
1271.         Um MRPy com as séries temporais.
1272.     PSD : auxclass_like
1273.         Um auxclass com as densidades espectrais
1274.         e as frequências no atributo f.
1275.     plot : bool, optional
1276.         Se verdadeiro, gera o gráfico das densidades espectrais dos valores
1277.         singulares. O padrão é falso.
1278.     mode : string, optional.
1279.         Modo de operação. Se 'interactive', o usuário deve definir
1280.         parâmetros com o mouse. Se 'batch', estes parâmetros devem ser
1281.         atributos de PSD. O padrão é 'interactive'.
1282.
1283.     Saídas
1284.     -----
1285.     fn : ndarray
1286.         Array com as frequências naturais.
1287.     zt : list
1288.         Lista com as razões de amortecimento. zt[0] para coeficientes do
1289.         método de meia-potência e zt[1] par ajuste ao espectro analítico.
1290.     V : ndarray
1291.         Array com as formas modais nas colunas.
1292.     PSD : auxclass_like
1293.         Um auxclass com os atributos pki, svi, fint e tint.
1294.
1295.     Modo batch
1296.     -----
1297.     Neste modo, o PSD deve ter os atributos pki e MG_i.
1298.     pki : list

```

```

1299.     Índices dos picos do array das frequências.
1300.     svi : list
1301.     Índices dos valores singulares a serem utilizados para
1302.     determinação das formas modais.
1303.     fint : array
1304.     Array com as frequências iniciais e finais a serem utilizadas para
1305.     ajuste da obtenção das funções de autocorrelação.
1306.     tint : array
1307.     Array com os tempos iniciais e finais a serem utilizados para ajuste da
1308.     função de autocorrelação.
1309.     """
1310.     G, f, nperseg, nfft = PSD, PSD.f, PSD.nperseg, PSD.nfft
1311.
1312.     U, S, VH = np.zeros_like(G), np.zeros_like(G), np.zeros_like(G)
1313.
1314.     USV = np.zeros((self.NX, len(f)))
1315.
1316.     for i in range(len(f)):
1317.         U[:, :, i], S[:, :, i], VH[:, :, i] = np.linalg.svd(G[:, :, i])
1318.
1319.     for i in range(self.NX):
1320.         for j in range(len(f)):
1321.             USV[i, j] = np.abs(U[i, :, j] * S[i, i, j] @ VH[:, i, j])
1322.
1323.     #-----
1324.
1325.     if mode.lower() == 'interactive':
1326.
1327.         plt.figure(figsize=(10,5))
1328.
1329.         for i in range(self.NX):
1330.             plt.semilogy(f[1:], USV[i, 1:])
1331.
1332.         print('Selecione os picos no gráfico')
1333.
1334.         plt.text(.7, 0.01,
1335.                 'Selecionar: botão esquerdo\n'
1336.                 'Desfazer: botão direito\n'
1337.                 'Concluir: botão do meio', transform=plt.gca().transAxes)
1338.
1339.         plt.title('Selecione os picos')
1340.         plt.legend(["1o", "2o", "3o", "4o"])
1341.         plt.xlabel('Frequência (Hz)')
1342.         plt.ylabel('Densidades Espectrais dos Valores Singulares')
1343.         plt.xlim([0, f[-1]])
1344.         plt.tight_layout()
1345.
1346.         pnt = plt.ginput(n=-1, timeout=0)
1347.         x = np.array(pnt)[: , 0]
1348.         pki = np.abs(f-x.reshape(-1,1)).argmin(axis=1)
1349.         y = np.array(pnt)[ : , 1]
1350.         svi = np.abs(USV[:, pki]-y).argmin(axis=0)
1351.
1352.         plt.close()
1353.
1354.     #-----
1355.
1356.     fint = np.zeros(2*len(pki))
1357.
1358.     for i, (j, k) in enumerate(zip(svi, pki)):
1359.
1360.         MACv = MAC(U[:, j, [k]], U[:, j, :])[0]
1361.
1362.         fig = plt.figure(figsize=(10, 6))
1363.

```

```

1364.         try:
1365.             imin = np.abs(np.max(f[:k][MACv[:k] < 0.8])-f).argmin() + 1
1366.         except ValueError:
1367.             imin = 0
1368.         try:
1369.             imax = np.abs(np.min(f[k:][MACv[k:] < 0.8])-f).argmin()
1370.         except ValueError:
1371.             imax = -1
1372.
1373.         gs = GridSpec(2, 1, height_ratios = [4, 1])
1374.
1375.         ax0 = plt.subplot(gs[0])
1376.         ax0.semilogy(f[1:],USV[j,1:])
1377.         ax0.semilogy(f[imin:imax], USV[j][imin:imax])
1378.         ax0.semilogy(f[k],USV[j,k], 'ro')
1379.         ax0.legend(["Espectro do Valor Singular",
1380.                    "Trecho com MAC > 0.8", "Frequência Natural"])
1381.         ax0.set_xlim([0, f[-1]])
1382.         ax0.set_ylabel('Densidades Espectrais dos Valores Singulares')
1383.         ax0.set_xticklabels([])
1384.         ax0.set_title('Selecione os extremos do intervalo')
1385.
1386.         ax1 = plt.subplot(gs[1])
1387.         ax1.plot(f,MACv)
1388.         ax1.set_xlim([0, f[-1]])
1389.         ax1.set_xlabel('Frequência (Hz)')
1390.         ax1.set_ylabel('MAC')
1391.
1392.         gs.tight_layout(fig)
1393.
1394.         pnt = np.array(plt.ginput(n=2,timeout=0))[:,0]
1395.
1396.         id1 = np.argmin(np.abs(f-pnt.reshape(-1,1)),axis=1)
1397.         fint[2*i:2*i+2] = f[id1]
1398.
1399.         plt.close()
1400.
1401.         PSD.pki, PSD.svi, PSD.fint = pki, svi, fint
1402.
1403.         #-----
1404.
1405.         elif mode.lower() == 'batch':
1406.             try:
1407.                 pki = PSD.pki
1408.                 svi = PSD.svi
1409.                 fint = PSD.fint
1410.             except AttributeError:
1411.                 sys.exit('PSD deve ter os atributos pki, svi e fint no modo batch')
1412.
1413.         #-----
1414.
1415.         else:
1416.             sys.exit('mode deve ser interactive ou batch')
1417.
1418.         #-----
1419.
1420.         idx = np.argmin(np.abs(f-fint.reshape(-1,1)),axis=1)
1421.
1422.         FSD = np.zeros((len(pki),S.shape[2]))
1423.         MACv = np.zeros((len(pki),S.shape[2]))
1424.
1425.         for i, (j, k, ii, si) in enumerate(zip(svi,pki,idx[:,2],idx[1:,2])):
1426.             FSD[i,ii:si] = USV[j,ii:si]
1427.             MACv[i] = MAC(U[:,j,[k]],U[:,j,:])[0]
1428.

```

```

1429. R = np.fft.irfft(FSD) # autocorrelation
1430. env = np.abs(np.fft.ifft(FSD)) # positive envelope of autocorrelation
1431.
1432. R = R /np.max(np.abs(R),axis=1).reshape(-1,1) # normalizing by max
1433. env = env/np.max( env,axis=1).reshape(-1,1) # abs value
1434.
1435. t = np.linspace(0,self.Td*nfft/self.N, R.shape[1]) # time series
1436. te = np.linspace(0,self.Td*nfft/self.N,env.shape[1])
1437.
1438. win = (nperseg-np.arange(0,nperseg))/nperseg # triangular window
1439.
1440. R = R[:,:nperseg]/win # divide by windows to remove bias
1441. env = env[:,:nperseg]/win
1442.
1443. fn, zt, PSD = fit_autc(PSD, t, te, R, env, mode, plot)
1444.
1445. #-----
1446.
1447. if plot:
1448.
1449.     fig = plt.figure(figsize=(8,6))
1450.     gs = GridSpec(2, 1, height_ratios = [3, 1])
1451.
1452.     ax0 = plt.subplot(gs[0])
1453.
1454.     leg = ['1° valor singular', '2° valor singular', '3° valor singular']
1455.
1456.     for ii in range(G.shape[0]):
1457.         ax0.semilogy(f[1:],USV[ii,1:],label=leg[ii])
1458.
1459.     for i, (ii, si) in enumerate(zip(idx[:,2],idx[1:2])):
1460.         ax0.semilogy(f[ii:si],np.abs(FSD[i,ii:si]), 'r',label=(i//1)*"_"+'Modo')
1461.
1462.     ax0.legend()
1463.     ax0.plot(f[pki], USV[svi,pki], "x")
1464.
1465.     for jj,kk in zip(pki,svi):
1466.         ax0.annotate('{:.3f} Hz'.format(f[jj]),
1467.                     (f[jj],USV[kk,jj]*1.25), ha='center')
1468.
1469.     ax0.set_xlim([0,f[-1]])
1470.     ax0.set_xticklabels([])
1471.     ax0.set_ylabel('Amplitude ((m/s2)/Hz)')
1472.     ax0.set_title('Densidades Espectrais dos Valores Singulares')
1473.
1474.     leg = ['1° modo', '2° modo', '3° modo', '4° modo', '5° modo', '6° modo']
1475.
1476.     ax1 = plt.subplot(gs[1])
1477.     for i, (ii, si) in enumerate(zip(idx[:,2],idx[1:2])):
1478.         ax1.plot(f[ii:si],MACv[i,ii:si],label=leg[i])
1479.
1480.     ax1.legend()
1481.     ax1.set_xlim([0,f[-1]])
1482.     ax1.set_xlabel('Frequência (Hz)')
1483.     ax1.set_ylabel('MAC')
1484.
1485.     gs.tight_layout(fig)
1486.
1487. #-----
1488.
1489. V = U[:,svi,pki]
1490.
1491. return fn, zt, V, PSD
1492.
1493. #-----

```

```

1494.
1495. def fit_autc(PSD, t, te, R, env, mode='interactive', plot=False):
1496.     """
1497.     Ajusta a função de autocorrelação teórica aos pontos, estimando a
1498.     frequência natural e a razão de amortecimento.
1499.
1500.     Entradas
1501.     -----
1502.     PSD : auxclass_like
1503.         Um auxclass com os atributos f e pki.
1504.     t : ndarray
1505.         Array unidimensional com a série temporal da função de autocorrelação.
1506.     te : ndarray
1507.         Array unidimensional com a série temporal do envelope.
1508.     R : ndarray
1509.         Array com as funções de autocorrelação nas linhas.
1510.     env : ndarray
1511.         Array com os envelopes da função de autocorrelação.
1512.     mode : string, optional.
1513.         Modo de escolha do intervalo de tempo. Se 'interactive', o usuário
1514.         deve definir os limites com o mouse. Se 'batch', os limites devem
1515.         estar no atributo PSD.tint.
1516.
1517.     Saídas
1518.     -----
1519.     fn : ndarray
1520.         Array com as frequências naturais
1521.     zt : ndarray
1522.         Array com as razões de amortecimento
1523.     PSD : auxclass_like
1524.         Um auxclass com o atributo tint.
1525.
1526.     Modo batch
1527.     -----
1528.     Neste modo, o PSD deve ter os atributos f, pki e tint.
1529.     f : Array
1530.         Array com as frequências das densidades espectrais.
1531.     pki : integer array
1532.         Array com os índices das frequências dos picos.
1533.     tint : array
1534.         Array com os tempos inicial e final a serem utilizadas para ajuste da
1535.         função de autocorrelação.
1536.     """
1537.
1538.     #-----
1539.     def envelope(t, Xp, η):
1540.
1541.         return Xp*np.exp(-η*t)
1542.
1543.     def decay(t, Xp, η, fn):
1544.
1545.         ωn = 2*np.pi*fn
1546.         ζ = η/ωn
1547.         wd = ωn * (1-ζ**2)**.5
1548.
1549.         return Xp*np.exp(-η*t)*np.cos(wd*t)
1550.
1551.     #-----
1552.
1553.     if mode.lower() == 'interactive':
1554.
1555.         idx = np.zeros(2*len(PSD.pki), dtype=int)
1556.
1557.         for ii in range(len(PSD.pki)):
1558.             plt.figure(figsize=(6,4))

```

```

1559.         plt.plot(te[:len(te)//4],env[ii][:len(te)//4], 'bo')
1560.         plt.xlim([0,te[len(te)//4]])
1561.         plt.xlabel('Tempo (s)')
1562.         plt.ylabel('Autocorrelação Normalizada')
1563.         plt.title('Selecione os limites do trecho a ser ajustado')
1564.         plt.tight_layout()
1565.
1566.         pnt = np.array(plt.ginput(n=2,timeout=0))[:,0]
1567.         idx[2*ii:2*ii+2] = np.argmin(np.abs(te-pnt.reshape(-1,1)),axis=1)
1568.
1569.         plt.close()
1570.
1571.         PSD.tint = te[idx]
1572.
1573.         #-----
1574.
1575.         elif mode.lower() == 'batch':
1576.             try:
1577.                 tint = PSD.tint
1578.             except AttributeError:
1579.                 sys.exit('PSD deve ter o atributo tint no modo batch')
1580.
1581.                 idx = np.argmin(np.abs(te-tint.reshape(-1,1)),axis=1)
1582.
1583.                 #-----
1584.
1585.             else:
1586.                 sys.exit('mode deve ser interactive ou batch')
1587.
1588.                 #-----
1589.
1590.         P = np.zeros((len(PSD.pki), 2))
1591.         Q = np.zeros((len(PSD.pki), 1))
1592.
1593.         for i, (j, k) in enumerate(zip(idx[:,2],idx[1::2])):
1594.
1595.             X0 = 1.00                # initial amplitude value
1596.             ζ0 = 0.01                # initial damping value
1597.             fn = PSD.f[PSD.pki[i]]  # initial natural frequency
1598.             η0 = 2*np.pi*fn*ζ0
1599.
1600.             Pmin = (1.00*X0, 0*η0)   # lower bounds
1601.             P0 = ( X0, η0)           # initial guesses
1602.             Pmax = (1.25*X0, 5*η0)   # upper bounds
1603.
1604.             P[i,:], cv = curve_fit(envelope, te[j:k], env[i,j:k],          # fit for
1605.                                   p0=P0, bounds=(Pmin, Pmax)) # X and η
1606.
1607.             Qmin = (0.97*fn)         # lower bounds
1608.             Q0 = ( fn)               # initial guesses
1609.             Qmax = (1.03*fn)        # upper bounds
1610.
1611.             Q[i,:], cv = curve_fit(lambda x, fn: decay(x,*P[i:], fn),
1612.                                   t[2*j:2*k], R[i,2*j:2*k], p0=Q0, bounds=(Qmin, Qmax)) # fit for fn
1613.
1614.             #-----
1615.
1616.             fn = Q[:,0]
1617.             zt = P[:,1]/(2*np.pi*fn)
1618.
1619.             #-----
1620.
1621.         if plot:
1622.
1623.             tf = np.linspace(0,t[-1],len(t)*100)

```

```

1624.
1625.     fig, ax = plt.subplots(1, len(PSD.pki), figsize=(4*len(PSD.pki),4),
1626.                          sharey=True,squeeze=False)
1627.
1628.     for i, (j, k) in enumerate(zip(idx[:,2],idx[1:,2])):
1629.         ax[0,i].plot(t[2*j:2*k],R[i,2*j:2*k], 'bo')
1630.         #ax[0,i].plot(te[j:k],env[i,j:k], 'ro')
1631.         ax[0,i].plot(tf,decay(tf, *P[i,:], *Q[i,:])) #fitted curve
1632.         ax[0,i].set_xlim(0,t[2*k])
1633.
1634.         ax[0,i].text(.99, .99, r'$f_n$ = {:.3f} Hz'.format(fn[i])
1635.                     +'\n'+ r'$\xi$ = {:.2f}%'.format(zt[i]*100),
1636.                     horizontalalignment='right',verticalalignment='top',
1637.                     transform=ax[0,i].transAxes,fontsize=11)
1638.
1639.
1640.     ax[0,i//2].set_xlabel("Tempo (s)")
1641.     ax[0,0].set_ylabel("Autocorrelação Normalizada")
1642.     fig.suptitle('Funções de Autocorrelação Normalizadas')
1643.     fig.tight_layout(rect=[0, 0.03, 1, 0.97])
1644.
1645.     return fn, zt, PSD
1646.
1647. #=====
1648. # Funções auxiliares: MAC e gráficos das formas modais
1649. #=====
1650.
1651. def MAC( $\Psi_i, \Psi_j$ , plot=False):
1652.     """
1653.     Calcula o Critério de Concordância Modal [1] a partir das formas modais
1654.     localizadas nas colunas de  $\Psi_i$  e  $\Psi_j$ .
1655.
1656.     Entradas
1657.     -----
1658.      $\Psi_i, \Psi_j$  : array_like
1659.         2D arrays com as formas modais nas colunas.
1660.     plot : bool, optional
1661.         Se verdadeiro, gera um gráfico dos MACs. O padrão é falso.
1662.
1663.     Saídas
1664.     -----
1665.     MAC : array_like
1666.
1667.     Referências
1668.     -----
1669.     .. [1] Allemang, R. J.; Brown, D. L. "A correlation coefficient for
1670.         modal vector analysis", In: 1st International Modal Analysis
1671.         Conference, p. 110-116, 1982.
1672.     """
1673.
1674.     MOMij =  $\Psi_i.T @ np.conj(\Psi_j)$ 
1675.     MOMii = np.diag( $\Psi_i.T @ np.conj(\Psi_i)$ )
1676.     MOMjj = np.diag( $\Psi_j.T @ np.conj(\Psi_j)$ )
1677.
1678.     MAC = np.abs(MOMij)**2 / np.outer(MOMii,MOMjj)
1679.
1680.     if plot:
1681.         plt.figure()
1682.         plt.pcolormesh(np.real(MAC), cmap='Blues', vmin=0, vmax=1,
1683.                       edgecolors='k', linewidth=.5)
1684.         cb = plt.colorbar()
1685.         cb.ax.set_title('MAC')
1686.         plt.xticks(np.arange(.5,MAC.shape[1] ,1),
1687.                   np.arange( 1,MAC.shape[1]+1,1))
1688.         plt.yticks(np.arange(.5,MAC.shape[0] ,1),

```

```

1689.         np.arange( 1,MAC.shape[0]+1,1))
1690.     plt.gca().invert_yaxis()
1691.     plt.gca().xaxis.tick_top()
1692.     plt.tight_layout()
1693.
1694.     return np.real(MAC)
1695.
1696. #-----
1697.
1698. def plot_1dshapes(fn,zt,vv,title,X,ref=False, fix=False):
1699.     """
1700.     Plota formas modais unidimensionais
1701.
1702.     Entradas
1703.     -----
1704.     fn : ndarray
1705.         Array com as frequências naturais.
1706.     zt : ndarray
1707.         Array com as razões de amortecimento.
1708.     vv : ndarray
1709.         Array 2D com as formas modais como colunas.
1710.     title : string
1711.         Título do gráfico.
1712.     X : ndarray
1713.         Coordenadas dos l sensores.
1714.     ref : tuple, list, optional
1715.         Especifica quais sensores são de referência.
1716.     fix : list, optional
1717.         Adiciona componente modal nula no gráfico para as coordenadas
1718.         especificadas. Por exemplo se [0,L], adiciona a componente nula na
1719.         coordenada zero e na coordenada L. O padrão é False.
1720.
1721.     Notas
1722.     -----
1723.     As primeiras componentes das formas modais são dos sensores de
1724.     referência. Por isso a necessidade de informar ref.
1725.     """
1726.     a_for = {'fontname':'Times New Roman','size':14}
1727.     l_for = {'fontname':'Times New Roman','size':12}
1728.     t_for = {'fontname':'Times New Roman','size':10}
1729.
1730.     if ref is not False:
1731.         X = np.hstack((X[ref,],np.delete(X,ref)))
1732.
1733.     if fix is not False:
1734.         X = np.hstack((np.array(fix),X))
1735.         vv = np.vstack((np.zeros((len(fix),vv.shape[1])),vv))
1736.
1737.     vv = np.sign(np.real(vv))*np.abs(vv)
1738.     idx = np.argsort(X)
1739.     it = np.argsort(fn)
1740.
1741.     plt.figure(figsize=(2*len(fn),5))
1742.
1743.     for i, k in enumerate(it):
1744.         plt.subplot(1,fn.shape[0],i+1)
1745.         plt.plot(0.97*vv[idx,k]/np.max(np.abs(vv[:,k])),X[idx])
1746.         plt.xlim((-1,1))
1747.         plt.ylim((0,X.max()))
1748.         plt.xticks(**t_for)
1749.         plt.yticks([])
1750.         plt.xlabel(r'$f_n$ = {:.3f} Hz'\n'r'$\zeta$ = {:.2f} %'
1751.                 .format(fn[k],zt[k]*100),**l_for)
1752.         if i == 0:
1753.             plt.yticks(np.linspace(0,X.max(),10),**t_for)

```

```

1754.
1755.     plt.suptitle(title + ' Modal Shapes',**a_for)
1756.     plt.tight_layout(rect=[0, -0.02, 1, 0.97])
1757.
1758.     return
1759.
1760. #-----
1761.
1762. def plot_3das1d(fn,zt,q,X,title,ref=False):
1763.     """
1764.     Plota as três formas modais de edifícios no formato unidimensional.
1765.
1766.     Entradas
1767.     -----
1768.     fn : ndarray
1769.         Array com as frequências naturais.
1770.     zt : ndarray
1771.         Array com as razões de amortecimento.
1772.     q  : ndarray
1773.         Array 2D com as formas modais como colunas.
1774.     title : string
1775.         Título do gráfico.
1776.     X  : ndarray
1777.         Array com as alturas.
1778.     title : string
1779.         Título do gráfico.
1780.     ref : tuple, list, optional
1781.         Especifica quais sensores são de referência.
1782.
1783.     Notas
1784.     -----
1785.     As primeiras componentes das formas modais são dos sensores de
1786.     referência. Por isso a necessidade de informar ref.
1787.     A componente modal nula na base é adicionada automaticamente.
1788.     """
1789.
1790.     a_for = {'fontname':'Times New Roman','size':14}
1791.     g_for = {'family'  :'Times New Roman','size':12}
1792.     l_for = {'fontname':'Times New Roman','size':12}
1793.     t_for = {'fontname':'Times New Roman','size':10}
1794.
1795.     if ref is not False:
1796.         X = np.hstack((X[ref],np.delete(X,ref)))
1797.
1798.     q  = np.sign(np.real(q))*np.abs(q)
1799.     it = np.argsort(fn)
1800.     X  = np.hstack((0,X))
1801.     q  = np.vstack((np.zeros((3,len(fn))),q))
1802.     idx = np.argsort(X)
1803.
1804.     plt.figure(figsize=(2*len(fn),5))
1805.
1806.     for ii, kk in np.ndenumerate(it):
1807.
1808.         q[:,kk] = q[:,kk]/np.max(np.abs(q[:,kk]))
1809.
1810.         plt.subplot(1,fn.shape[0],ii[0]+1)
1811.         plt.plot(q[0::3,kk][idx],X[idx],'k', linewidth=3)
1812.         plt.plot(q[1::3,kk][idx],X[idx],'r--',linewidth=2)
1813.         plt.plot(q[2::3,kk][idx],X[idx],'g:', linewidth=4)
1814.         plt.xticks(**t_for)
1815.         plt.yticks([])
1816.         plt.xlim((-1,1))
1817.         plt.ylim((0,X.max()))
1818.         plt.xlabel(r'$f_n$ = {:.3f} Hz'\n'r'$\zeta$ = {:.2f} %'

```

```
1819.         .format(fn[kk],zt[kk]*100,**l_for)
1820.     if ii[0] == 0:
1821.         plt.yticks(np.linspace(0,X.max(),10),**t_for)
1822.         plt.legend(('x', 'y',r'\theta_z$'), loc='lower left',
1823.                 prop=g_for, handlelength=1.2, handletextpad=0.4)
1824.
1825.     plt.suptitle(title + ' Modal Shapes',**a_for)
1826.     plt.tight_layout(rect=[0, 0.03, 1, 0.97])
1827.     plt.show()
1828.
1829.     return
1830.
```

APÊNDICE B – CESSIPy: guia ao usuário

A identificação de sistemas visa mensurar a descrição matemática de um sistema analisando as suas entradas e saídas. No contexto da Engenharia Civil, as estruturas como edifícios, torres e pontes são os sistemas e os parâmetros a serem identificados são as frequências naturais, razões de amortecimentos e formas modais. As entradas são os carregamentos aplicados e as saídas são as respostas estruturais, as quais podem ser medidas em termos de deslocamentos, velocidades ou acelerações.

A identificação de estruturas a partir de excitações ambientais é denominada identificação estocástica de sistemas ou ainda identificação modal estocástica. Nestas apenas a resposta (saída) é medida, sendo a excitação considerada como um processo estocástico multidimensional, representado tipicamente por um ruído branco.

O principal objetivo do CESSIPy é a identificação das propriedades dinâmicas de estruturas quando apenas as saídas são medidas. Programaram-se:

- a) Método de Detecção de Picos (*Peak Picking Method – PP*) também chamado de Método Básico no Domínio da Frequência (*Basic Frequency Domain Method – BFD*);
- b) Método Aperfeiçoado de Decomposição no Domínio da Frequência (*Enhanced Frequency Domain Decomposition – EFDD*);
- c) Método de Identificação Estocástica de Subespaços baseados nas Covariâncias (*Covariance-driven Stochastic Subspace Identification – SSI COV*);
- d) Método das Variáveis Instrumentais (*Instrumental Variable – IV*);
- e) Método de Identificação Estocástica de Subespaços baseados nas Séries Temporais (*Data-driven Stochastic Subspace Identification – SSI DATA*).

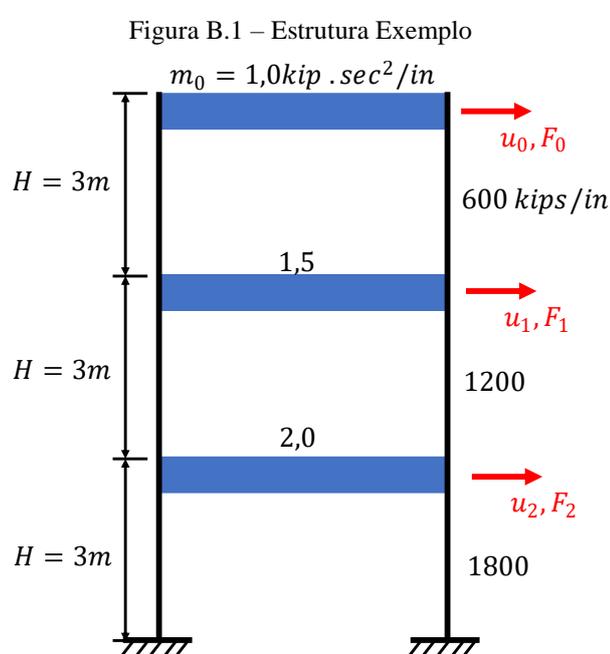
Neste Apêndice apresenta-se uma breve descrição dos métodos e o uso das rotinas programadas em Python para a identificação de uma estrutura simples. Este texto encontra-se disponível para download em < <https://github.com/MatheusCarini/CESSIPy> > podendo ser executado de forma interativa por meio do aplicativo Jupyter Notebook.

```
In [1]: import numpy          as np
import pandas            as pd
import matplotlib.pyplot as plt
import CESSIPy          as SSI
from MRPy import MRPy
%matplotlib inline
```

B.1 ESTRUTURA EXEMPLO

Visando facilitar a compreensão das rotinas, os métodos são empregados na identificação de uma estrutura simples submetida a forças aleatórias.

Trata-se de um pórtico plano com três pavimentos (Figura B.1) retirado de Clough e Penzien (2003, p. 203). As massas são concentradas nas lajes, enquanto a flexibilidade é atribuída exclusivamente à deformação por flexão dos pilares. Como consequência, a estrutura tem deformações horizontais em forma de corte (*shear building*), com contribuição da flexão global desprezível.



(fonte: adaptado de CLOUGH; PENZIEN, 2003, p. 203)

Inicialmente definem-se os parâmetros:

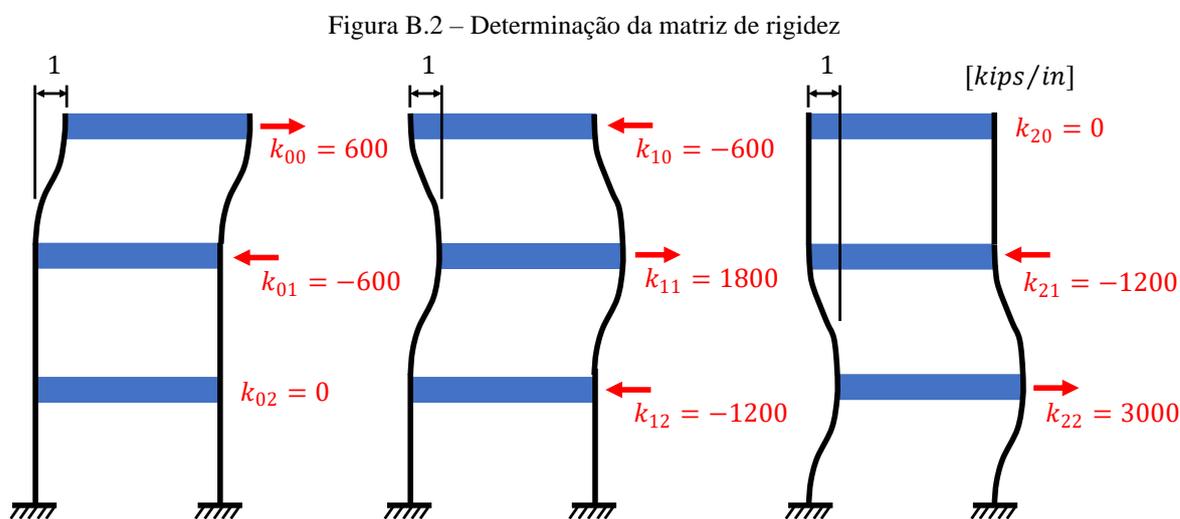
```
In [2]: H = 3 # altura do pavimento [m]
        ζ = 0.015 # razão de amortecimento crítico [-]
```

A matriz de massa \mathbf{M} é (CLOUGH; PENZIEN, 2003, p. 203):

```
In [3]: M = 1*175127*np.array([[ 1.0,  0,  0],
                               [  0, 1.5,  0],
                               [  0,  0, 2.0]])
```

Sendo utilizada a conversão $1 \text{ kip} \cdot \text{sec}^2 / \text{in} = 175127 \text{ kg}$.

A determinação da matriz de rigidez \mathbf{K} da estrutura é apresentada em Clough e Penzien (2003, p. 203). A montagem da matriz de rigidez é feita aplicando-se deslocamentos unitários em cada pavimento isoladamente, conforme Figura B.2.



(fonte: adaptado de CLOUGH; PENZIEN, 2003, p. 203)

Resultando na matriz:

```
In [4]: K = 600*175127*np.array([[ 1, -1,  0],
                               [-1,  3, -2],
                               [ 0, -2,  5]])
```

Sendo utilizada a conversão 1 kips / in = 175127 N/m.

Deseja-se estudar a determinação das frequências naturais (autovalores) e das formas modais (autovetores) de uma estrutura com n graus de liberdade. Parte-se do caso mais simples, correspondente a um sistema não amortecido em vibração livre:

$$\mathbf{M}\ddot{\vec{u}}(t) + \mathbf{K}\vec{u}(t) = 0 \quad (\text{B.1})$$

Sendo $\ddot{\vec{u}}(t)$ e $\vec{u}(t)$ os vetores deslocamentos e acelerações generalizados, respectivamente.

No método da superposição modal, escreve-se o vetor deslocamento como a soma dos deslocamentos modais:

$$\vec{u}(t) = \sum_{i=1}^n u_i(t) \vec{\phi}_i = \Phi \vec{u}_m(t) \quad (\text{B.2})$$

Sendo $u_i(t)$ a resposta modal, $\vec{\phi}_i$ a forma modal, Φ uma matriz cujas colunas são as formas modais e $\vec{u}_m(t)$ o vetor das respostas modais.

As formas modais $\vec{\phi}_i$ e as respectivas frequências naturais ω_i são determinadas a partir da solução do problema de autovalores e autovetores:

$$\mathbf{K} \vec{\phi}_i = \omega_i^2 \mathbf{M} \vec{\phi}_i \quad (\text{B.3})$$

O qual pode ser escrito na forma convencional:

$$\mathbf{K}^{-1} \mathbf{M} \vec{\phi}_i = \frac{1}{\omega_i^2} \vec{\phi}_i \quad (\text{B.4})$$

```
In [5]: Λ, Q = np.linalg.eig(np.linalg.inv(K) @ M)
```

Os autovetores são agrupados, um em cada coluna, formando a matriz Φ . Tornando a componente modal no topo unitária, é possível compará-los com os calculados por Clough e Penzien (2003, p. 207).

```
In [6]: pd.DataFrame(Q/Q[0,:], columns=['Modo 1', 'Modo 2', 'Modo 3'])
```

```
Out [6]:
```

	Modo 1	Modo 2	Modo 3
0	1.000000	1.000000	1.000000
1	0.648535	-0.606599	-2.541936
2	0.301850	-0.678977	2.439628

As frequências naturais são determinadas a partir dos autovalores Λ :

```
In [7]: wn = (1/Λ)**.5 # frequências naturais circulares [rad/s]
fn = wn/(2*np.pi) # frequências naturais [Hz]

for j, (k,l) in enumerate(zip(wn,fn)):
    print('w{0} = {1:3.3f} rad/s    f{0} = {2:2.4f} Hz'.format(j+1,k,l))
```

```
Out [7]:  ω1 = 14.522 rad/s    f1 = 2.3112 Hz
          ω2 = 31.048 rad/s    f2 = 4.9414 Hz
          ω3 = 46.099 rad/s    f3 = 7.3370 Hz
```

As quais são iguais as calculadas por Clough e Penzien (2003, p. 204).

É conveniente ajustar a escala das formas modais de modo a obter massas modais unitárias:

$$\Phi^T \mathbf{M} \Phi = \mathbf{I}_n \quad (\text{B.5})$$

Sendo \mathbf{I}_n a matriz identidade de dimensões $n \times n$.

```
In [8]:  Mk = np.diag(Q.T @ M @ Q) # massa modal
         Q = Q/Mk**0.5           # normaliza forma modal de modo que as massas modais
                                sejam unitárias
```

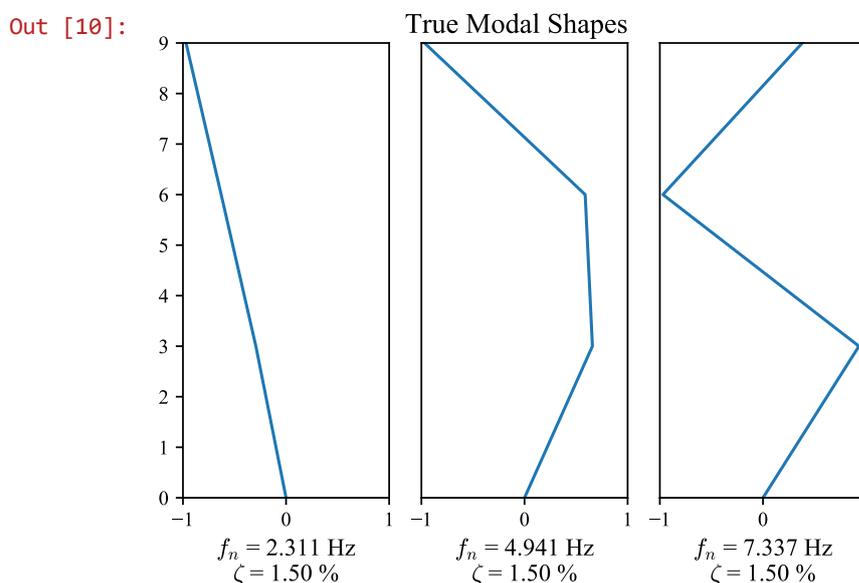
Demonstrando que as massas modais são unitárias:

```
In [9]:  Mk = Q.T @ M @ Q
         Mk
```

```
Out [9]:  array [[ 1.00000000e+00,  5.52118756e-16, -3.28436546e-16],
               [ 5.36755073e-16,  1.00000000e+00,  1.79685031e-16],
               [-3.29470757e-16,  1.46502042e-16,  1.00000000e+00]])
```

Por fim, define-se o vetor com as alturas de cada pavimento x e plotam-se as propriedades dinâmicas da estrutura simulada.

```
In [10]:  x = np.linspace(3*H,H,3)
          SSI.plot_1dshapes(fn,ζ*np.ones(3),Q,'True',x,fix=[0])
```



O parâmetro $fix=[0]$ adiciona uma componente nula à forma modal na altura 0.

Uma descrição detalhada das funções programadas pode ser obtida escrevendo o nome da função acompanhado de `?`.

Visando determinar a resposta estrutural, definem-se o número de pontos no tempo N e a frequência de amostragem f_s :

```
In [11]: N = 2**15      # número de pontos no tempo
         fs = 25       # frequência de amostragem [Hz]
```

As forças aplicadas em cada pavimento são consideradas aleatórias com distribuição gaussiana:

```
In [12]: F = MRPy(5E3*np.random.randn(3,N),fs=fs) # forças nodais [N]
```

A resposta dinâmica de um sistema com n graus de liberdade é determinada através da solução da equação matricial de equilíbrio dinâmico:

$$\mathbf{M}\ddot{\vec{u}}(t) + \mathbf{C}_f\dot{\vec{u}}(t) + \mathbf{K}\vec{u}(t) = \vec{F}(t) \quad (\text{B.6})$$

Sendo \mathbf{C}_f a matriz de amortecimento, $\dot{\vec{u}}(t)$ o vetor velocidades e $\vec{F}(t)$ o vetor das cargas.

No método da superposição modal, a equação matricial de equilíbrio dinâmico é desacoplada em equações diferenciais isoladas, uma para cada modo. Pré-multiplicando-se a equação anterior por Φ^T e utilizando a definição (B.2):

$$\Phi^T \mathbf{M} \Phi \ddot{\vec{u}}_m(t) + \Phi^T \mathbf{C}_f \Phi \dot{\vec{u}}_m(t) + \Phi^T \mathbf{K} \Phi \vec{u}_m(t) = \Phi^T \vec{F}(t) \quad (\text{B.7})$$

Sendo $\Phi^T \mathbf{M} \Phi \ddot{\vec{u}}_m(t)$ a matriz diagonal com as massas modais m_i , $\Phi^T \mathbf{C}_f \Phi$ a matriz dos amortecimentos modais, $\Phi^T \mathbf{K} \Phi$ a matriz diagonal com as rigidezes modais k_i e $\Phi^T \vec{F}(t) = \vec{F}_m(t)$ o vetor das forças modais.

As forças modais são calculadas:

```
In [13]: Fk = MRPy(np.dot(Q.T,F),fs=fs) # forças modais
```

Admitindo-se amortecimento proporcional, assume-se que os autovetores diagonalizam a matriz \mathbf{C}_f :

$$\Phi^T \mathbf{C}_f \Phi = \text{diag}(c_i) = \text{diag}(2\xi_i \omega_i m_i) \quad (\text{B.8})$$

Sendo c_i os amortecimentos modais e ξ_i a razão de amortecimento crítico.

Assim, a equação matricial (B.6) resulta em equações diferenciais desacopladas, uma para cada modo:

$$m_i \ddot{u}_i(t) + c_i \dot{u}_i(t) + k_i u_i(t) = F_i(t) \quad (\text{B.9})$$

As quais são resolvidas separadamente no domínio da frequência.

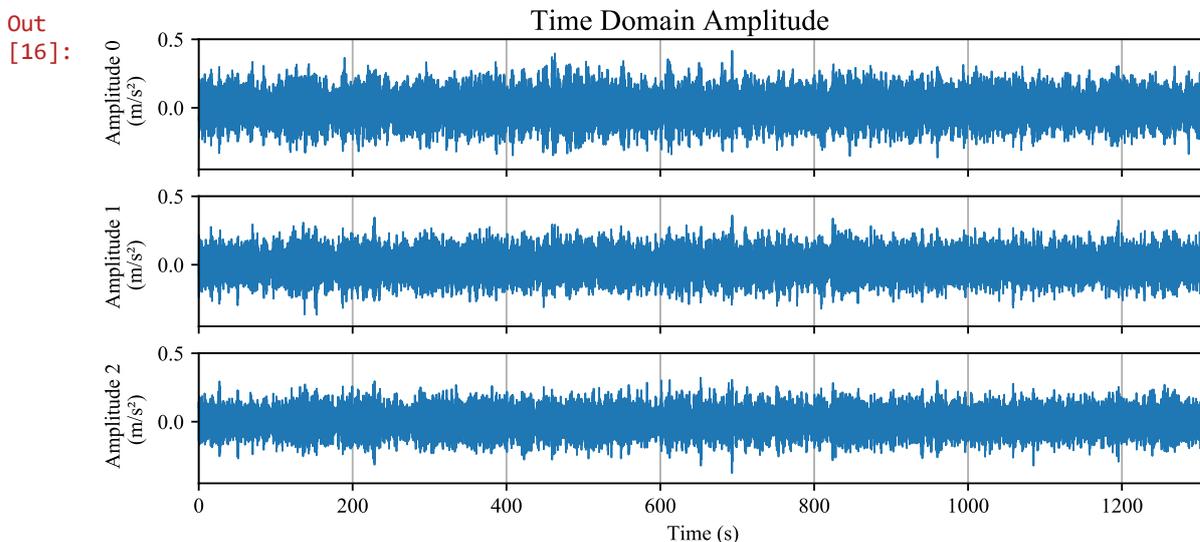
```
In [14]: Xk = MRPy.s dof_Fourier(Fk,fn,ζ) # respostas modais
```

Com a equação B.2, obtém-se os deslocamentos e, derivando-os duas vezes, as acelerações.

```
In [15]: X = MRPy(np.dot(Q,Xk),fs=fs) # deslocamentos
A = MRPy.differentiate(MRPy.differentiate(X)) # acelerações
```

As séries temporais das acelerações são as únicas informações que os métodos de identificação empregam. Um gráfico pode ser facilmente plotado.

```
In [16]: h1 = A.plot_time(unit=' (m/s²)')
```



B.2 IDENTIFICAÇÃO DE SISTEMAS NO DOMÍNIO DA FREQUÊNCIA

Os métodos de identificação podem ser divididos de acordo com o tipo de dados que empregam: séries temporais, covariâncias ou densidades espectrais. Inicia-se a apresentação pelos métodos no domínio da frequência para corresponder à ordem histórica da identificação estocástica de sistemas. Apesar das medições estarem no domínio do tempo, é muito útil analisar as suas representações no domínio da frequência.

B.2.1 Densidades espectrais

Os métodos BFD e EFDD assumem que a estrutura está sujeita a excitações aleatórias do tipo ruído branco e empregam a matriz das densidades espectrais da resposta $\mathbf{G}(f)$:

$$\mathbf{G}(f) = E[\mathbf{Y}(f)\mathbf{Y}^H(f)] \quad (\text{B.10})$$

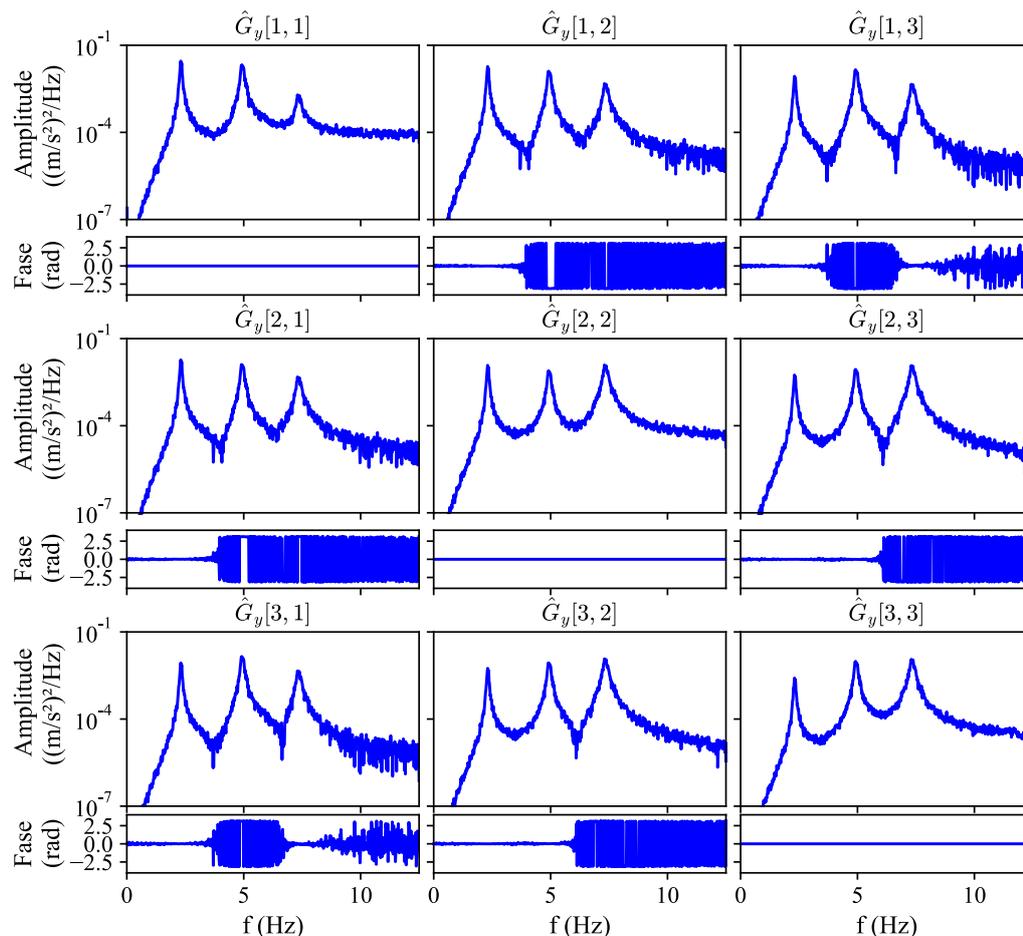
Sendo \mathbf{Y} a resposta da estrutura no domínio da frequência. O superescrito $(.)^H$ denota complexo conjugado transposto e $E[.]$ valor esperado.

As estimativas das funções de densidades espectrais são calculadas e plotadas pela função `SDM`, a qual emprega a função `scipy.signal.csd`. Os termos da diagonal são chamados de autoespectros e os demais de espectros cruzados. Os sinais são divididos em segmentos com `nperseg` pontos visando obter estimativas suavizadas e, por padrão, são multiplicados por uma janela de dados do tipo Hanning.

Uma forma de visualização da matriz espectral é apresentada a seguir.

```
In [17]: nps = A.N//16
         PSD = SSI.SDM(A, nperseg=nps, plot=True)
```

Out [17]:



As densidades espectrais são salvas no objeto PSD como um *array* tridimensional. Nele, os dois primeiros índices correspondem aos sinais e o terceiro à frequência. Por exemplo, as densidades espectrais para a frequência nula são:

In [18]: `PSD[:, :, 0]`

```
Out [18]: auxclass([[2.93663986e-07+0.j, 8.80953600e-08+0.j, 6.42401499e-08+0.j],
                    [8.80953600e-08+0.j, 1.60110918e-07+0.j, 7.05961834e-08+0.j],
                    [6.42401499e-08+0.j, 7.05961834e-08+0.j, 1.21324584e-07+0.j]])
```

Note que PSD é um objeto com as frequências no atributo PSD.f e o número de pontos de cada segmento PSD.nperseg.

B.2.2 Método Básico no Domínio da Frequência (BFD)

A abordagem mais simples para estimar as características dinâmicas da estrutura submetida a ações ambientais é o Método de Detecção de Picos (*Peak Picking method* – PP) também chamado de Método Básico no Domínio da Frequência (*Basic Frequency Domain method* –

BFD). Conforme o próprio nome sugere, a identificação das frequências naturais é feita selecionando-se os picos das densidades espectrais das respostas. Para estruturas pouco amortecidas e com modos bem separados, tanto os autoespectros quanto os espectros cruzados atingem um valor máximo na frequência de ressonância. Na vizinhança de uma frequência de ressonância f_r , a matriz espectral pode ser aproximada como:

$$\mathbf{G}(f_r) \approx \alpha_r \vec{\phi}_r \vec{\phi}_r^H \quad (\text{B.11})$$

Sendo α_r uma constante que depende da frequência natural, do amortecimento e da participação modal e $\vec{\phi}_r$ uma estimativa da forma modal.

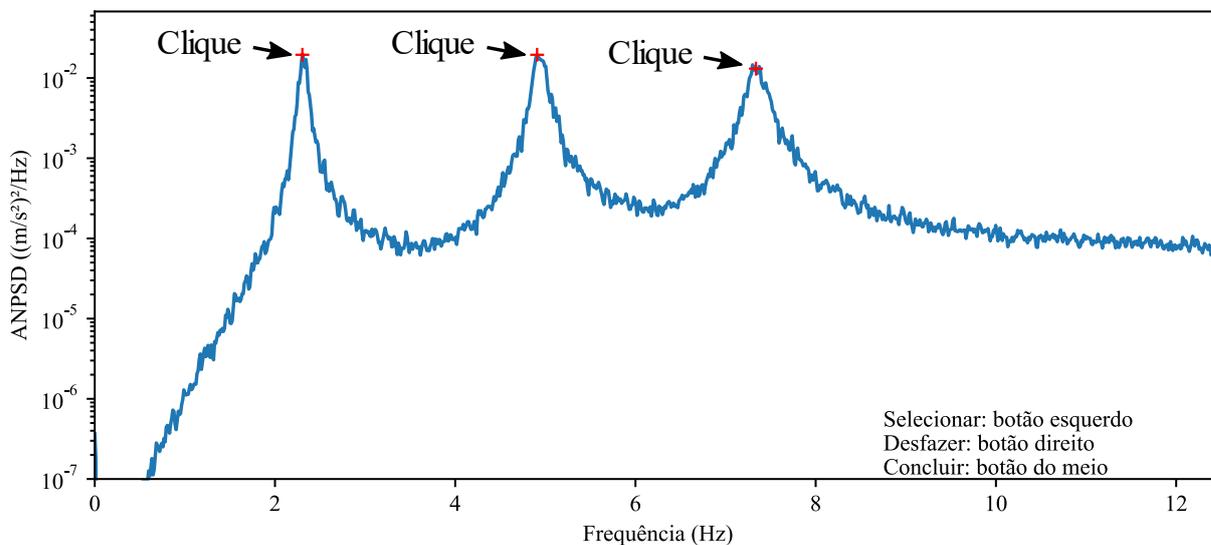
O espectro normalizado médio (ANPSD), proposto por Felber (1993), sintetiza em um único gráfico a informação contida em todos os autoespectros e nele inicia-se o processo de identificação detectando seus picos.

A função programada possui dois modos: 'interactive' e 'batch'. Inicia-se a apresentação pelo modo padrão 'interactive'.

Ao chamar a função ANPSD_from_SDM, uma janela gráfica surgirá. Nela, a seleção dos picos deve ser feita clicando-se nos picos com o botão esquerdo do mouse (Figura B.3). Uma vez que todos os picos estejam marcados, clica-se no botão central para finalizar. Caso seja necessário desconsiderar o ponto selecionado, deve-se pressionar o botão direito. É possível empregar a ferramenta nativa de zoom, desde que após o seu uso seja desfeita a ação com o botão direito do mouse pois o Python entende que o clique do zoom é uma seleção de pico. A Figura B.3 apresenta um exemplo de uso.

Figura B.3 – Seleção dos picos do ANPSD

Clique nos picos com o botão esquerdo
Finalize clicando com o botão central



(fonte: elaborado pelo autor)

Na célula abaixo, pressione Enter e selecione os picos no gráfico que aparecerá na tela.

```
In [19]: input ('Pressione Enter para continuar')

%matplotlib qt
PSD = SSI.ANPSD_from_SDM(PSD)
```

Out [19]: Pressione Enter para continuar

O objeto PSD recebe o atributo PSD.ANPSD e PSD.pki contendo o espectro normalizado médio e os índices dos picos selecionados pelo usuário, respectivamente.

No método BFD, as frequências naturais estimadas correspondem aos picos das densidades espectrais:

```
In [20]: for k, j in enumerate(PSD.pki):
          print('fn{0} = {1:7.4f} Hz'.format(k+1, PSD.f[j]))
```

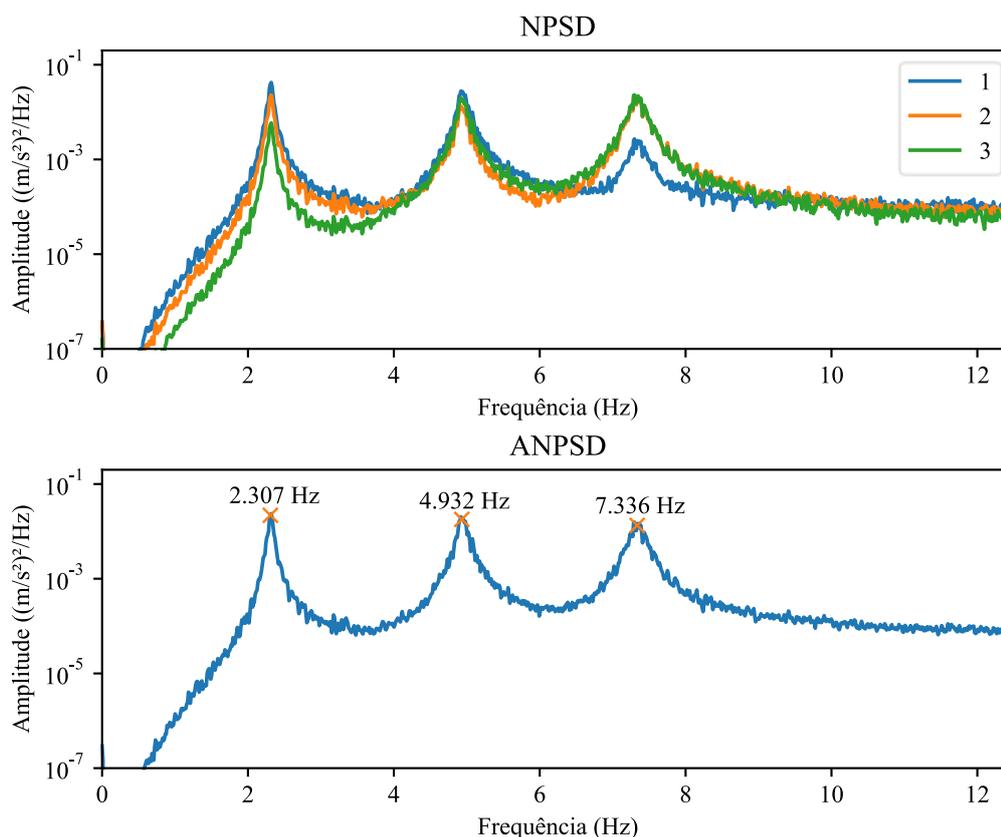
```
Out [20]: fn1 = 2.3071 Hz
          fn2 = 4.9316 Hz
          fn3 = 7.3364 Hz
```

Conforme mencionado, o modo padrão de ANPSD_from_SDM é 'interactive', no qual os picos são selecionados com o mouse. O usuário pode alterar as frequências especificadas modificando diretamente o atributo PSD.pki.

O outro modo é chamado de 'batch' e permite a execução do código de forma automatizada, sem a interação do usuário. Nele, o atributo `PSD.pki` deve conter os índices das frequências dos picos.

```
In [21]: %matplotlib inline
PSD = SSI.ANPSD_from_SDM(PSD,mode='batch',plot=True)
```

Out [21]:



A identificação das formas modais e razões de amortecimento é feita pela função BFD. O amortecimento é estimado por dois métodos: o método de meia-potência e ajuste ao espectro analítico.

No método da meia potência, a razão de amortecimento é estimada a partir da frequência do pico f_r e das frequências com ordenada espectral igual a metade da ordenada máxima f_a e f_b (EWINS, 2000):

$$\xi_i = \frac{f_b^2 - f_a^2}{4f_r^2} \quad (\text{B.12})$$

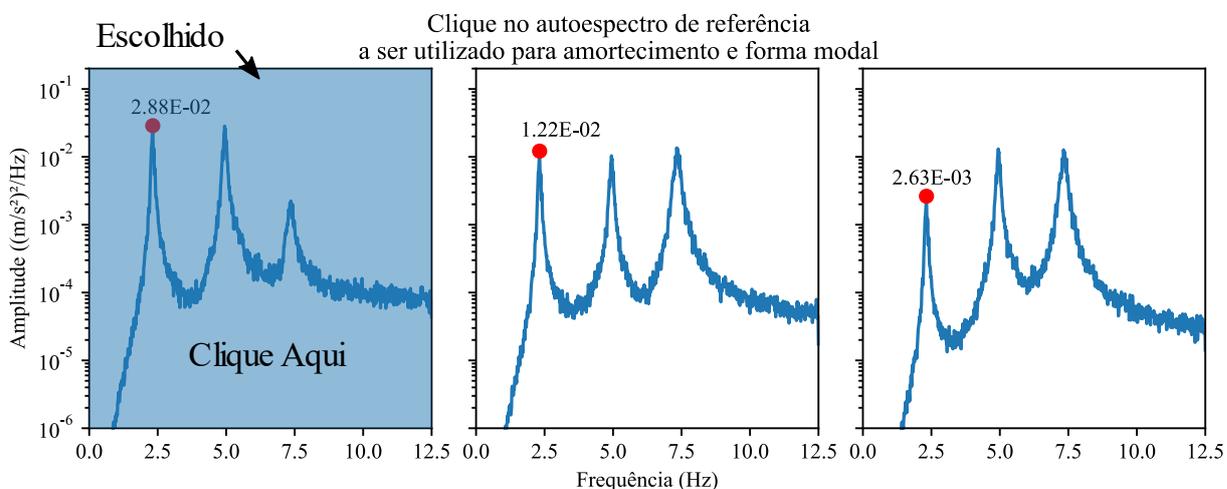
No método de ajuste ao espectro analítico, seleciona-se um trecho das densidades espectrais na vizinhança do pico e ajusta-se a curva corresponde ao espectro da resposta em aceleração de um sistema de um grau de liberdade sujeito a uma entrada de densidade espectral constante:

$$G_y(f) = C_1 \left| \frac{f^2}{1 - (f/f_n)^2 + j2\xi(f/f_n)} \right|^2 + C_2 \quad (\text{B.13})$$

Sendo C_1 uma constante que depende da rigidez do sistema e da densidade espectral da entrada, f_n a frequência natural não amortecida, ξ a razão de amortecimento, j a unidade imaginária e C_2 uma constante que leva em conta a influência dos modos fora da zona do pico analisado.

Uma vez chamada a função, por padrão ela emprega o modo 'interactive'. Primeiramente deve-se clicar no autoespectro a ser utilizado para estimativa do amortecimento (normalmente escolhe-se aquele em que a densidade espectral possui maior amplitude), conforme Figura B.4.

Figura B.4 – Seleção do autoespectro utilizado na estimativa do amortecimento

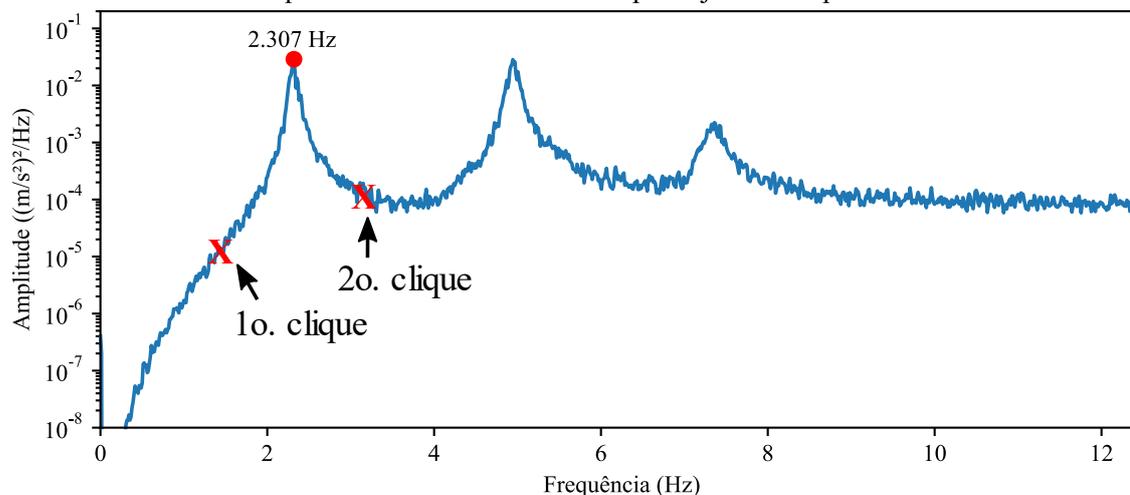


(fonte: elaborado pelo autor)

Em seguida, deve-se selecionar o extremo inferior e o superior do intervalo a ser empregado para ajuste ao espectro analítico. A Figura B.5 apresenta um exemplo.

Figura B.5 – Seleção do intervalo para ajuste ao espectro analítico

Clique nos extremos do intervalo para ajuste ao espectro analítico



(fonte: elaborado pelo autor)

O código abaixo apresenta a chamada da função. Pressione Enter para realizar a seleção gráfica.

```
In [22]: input('Pressione Enter para continuar')

%matplotlib qt
FBFD, ZBFD, VBFD, PSD = SSI.BFD(A, PSD, plot=False)
```

```
Out [22]: Pressione Enter para continuar
          Selecione o autoespectro de referência
```

As razões de amortecimento estimadas pelo método da meia-potência são ZBFD[0] e pelo ajuste ao espectro analítico ZBFD[1].

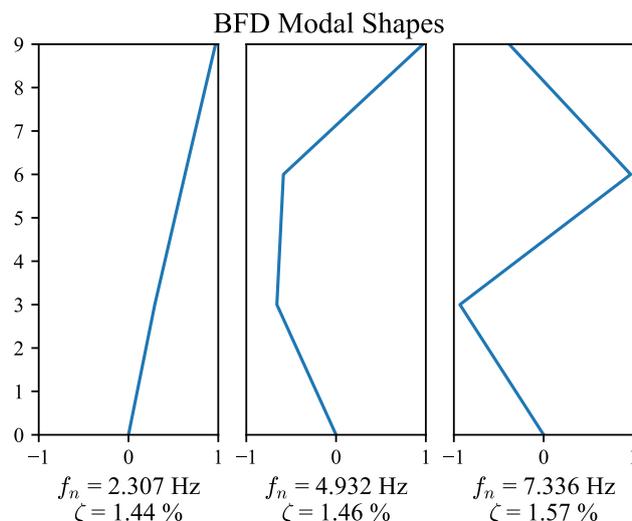
```
In [23]: print('Meia Potência Espectro Analítico')
          for j, (k,l) in enumerate(zip(ZBFD[0],ZBFD[1])):
              print('ζ{0} = {1:4.2f}%      ζ{0} = {2:4.2f}%'.format(j+1,k*100,l*100))
```

```
Out [23]: Meia Potência Espectro Analítico
          ζ1 = 1.41%      ζ1 = 1.44%
          ζ2 = 1.31%      ζ2 = 1.46%
          ζ3 = 1.67%      ζ3 = 1.57%
```

E as propriedades dinâmicas identificadas:

```
In [24]: %matplotlib inline
          SSI.plot_1dshapes(FBFD,ZBFD[1],VBFD, 'BFD',x, fix=[0])
```

Out [24]:



Após o uso da função BFD no modo 'interactive', o objeto PSD recebe vários atributos:

In [25]: `PSD.pki # índice dos picos das densidades espectrais`

Out [25]: `array([189, 404, 601], dtype=int64)`

In [26]: `PSD.MGi # índice dos autoespectros empregados para determinação do amortecimento`

Out [26]: `array([0, 0, 1])`

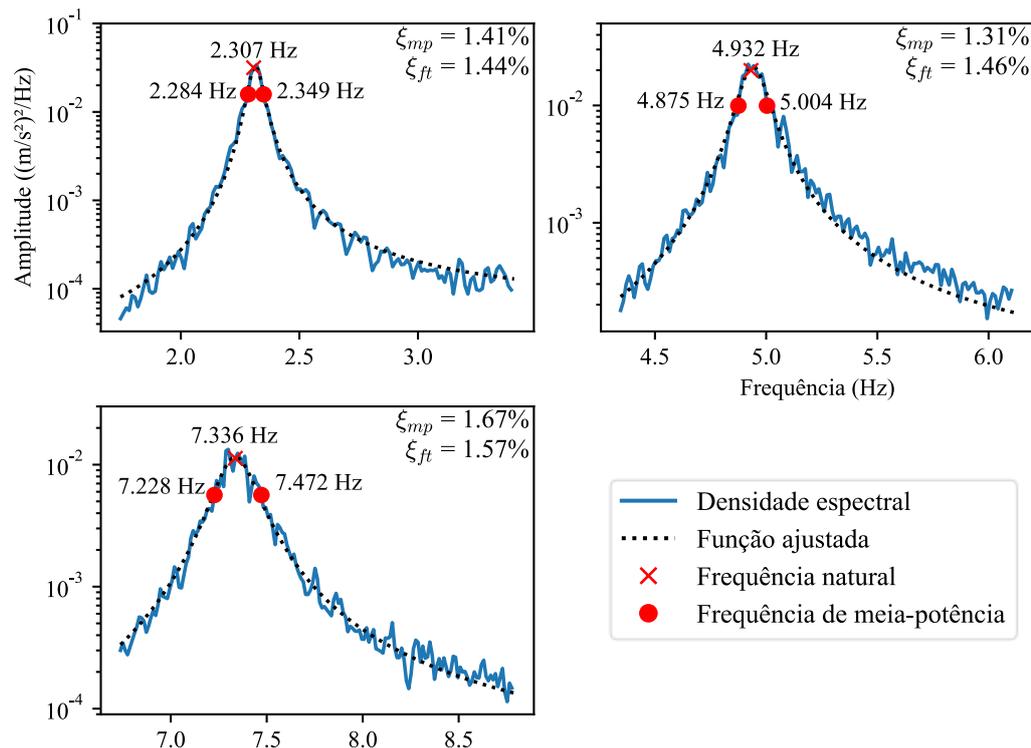
In [27]: `PSD.fint # frequências inferior e superior empregadas para ajuste ao espectro analítico`

Out [27]: `array([1.75, 3.40, 4.35, 6.12, 6.74, 8.79])`

Esses atributos podem ser alterados para a realização de ajustes aos dados informados com o mouse. Também podem ser definidos antes da chamada da função, permitindo o uso do modo 'batch'. A entrada `plot=True` permite verificar graficamente as frequências de meia-potência e o espectro analítico ajustado.

In [28]: `FBFD, ZBFD, VBFD, PSD = SSI.BFD(A, PSD, plot=True, mode='batch')`

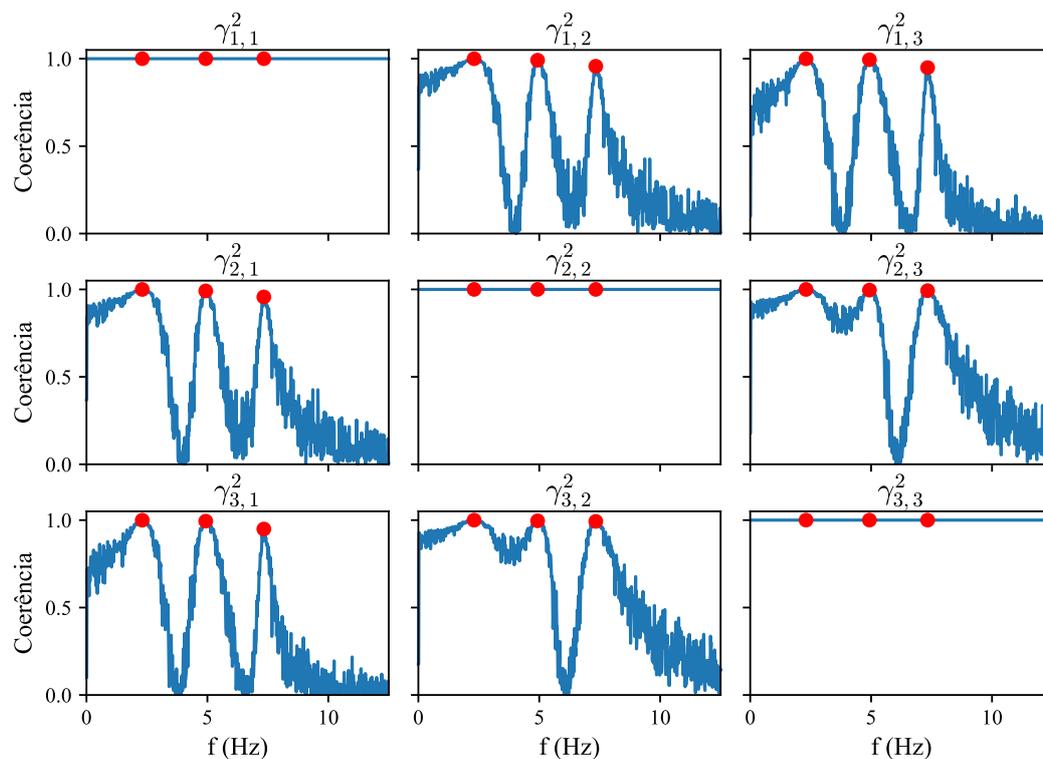
Out [28]:



As funções de coerência permitem confirmar se as estimativas das formas modais são confiáveis. Na frequência de ressonância, a função de coerência deve ser próxima a 1.

In [29]: `γ = SSI.coherence(A, PSD, nps, plot=True)`

Out [29]:



B.2.3. Método Aperfeiçoado de Decomposição no Domínio da Frequência (EFDD)

O método EFDD, formalizado por Brincker et al. (2000, 2001b), aplica a decomposição em valores singulares (SVD) à matriz espectral para cada frequência, visando decompor o sistema estrutural em uma série de sistemas com um grau de liberdade. A SVD da matriz espectral é:

$$\mathbf{G}(f) = \mathbf{U}(f)\mathbf{\Sigma}(f)\mathbf{U}^H(f) \quad (\text{B.14})$$

Sendo $\mathbf{\Sigma}(f)$ uma matriz diagonal com os valores singulares em ordem decrescente e $\mathbf{U}(f)$ uma matriz complexa com os vetores singulares. Os valores singulares não nulos representam fisicamente o espaço do sinal enquanto os valores próximos a zero indicam o espaço do ruído. Se apenas um modo é importante numa certa frequência f_r , a matriz espectral pode ser aproximada por uma matriz com posto um com o valor singular máximo:

$$\mathbf{G}(f) \cong \sigma_1(f_r)\vec{u}_1(f_r)\vec{u}_1^H(f_r) \quad (\text{B.15})$$

O vetor singular $\vec{u}_1(f_r)$ é uma estimativa para a forma modal $\vec{\phi}_r$. No caso de modos pouco espaçados, mais de um valor singular atingirá um máximo nas vizinhanças de uma certa frequência, e todos os vetores singulares correspondentes a valores singulares não nulos são estimativas das formas modais (HU et al., 2009).

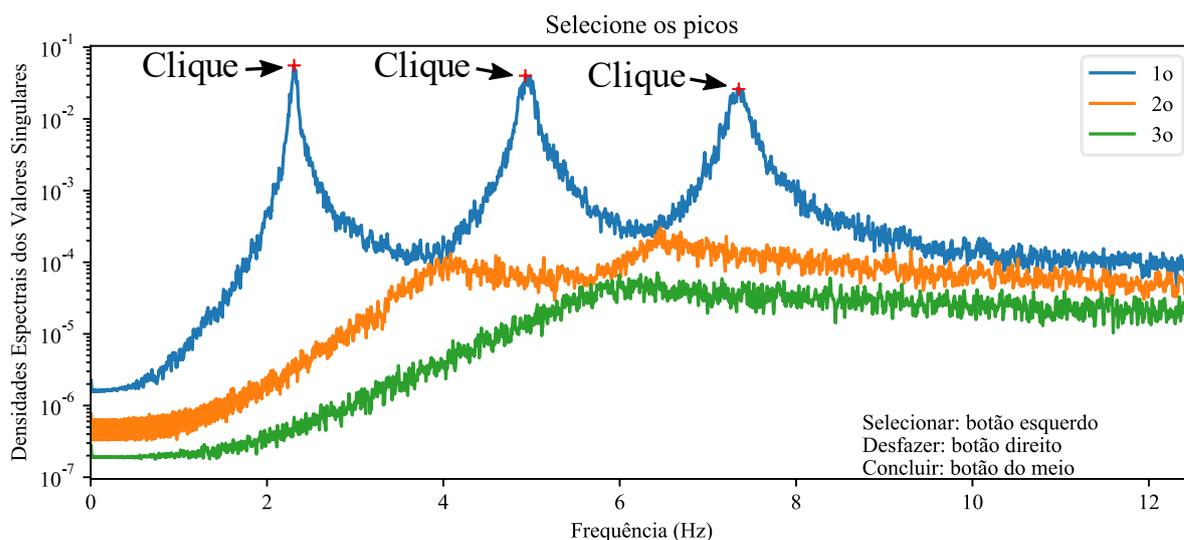
Na avaliação das densidades espectrais, Bendat e Piersol (2010) não recomendam empregar janela de dados, o que pode ser feito especificando `window='boxcar'`. Os autores também recomendam aumentar cada segmento de `nps` valores com `nps` zeros, o que pode ser feito especificando `nfft=2*nps`.

In [30]: `P2SD = SSI.SDM(A, nperseg=nps, plot=False, window='boxcar', nfft=2*nps)`

Programou-se o método na rotina EFDD, cujo modo padrão é 'interactive'.

Uma vez chamada, clicam-se nos picos das densidades espectrais dos valores singulares, conforme Figura B.6.

Figura B.6 – Seleção dos picos dos valores singulares

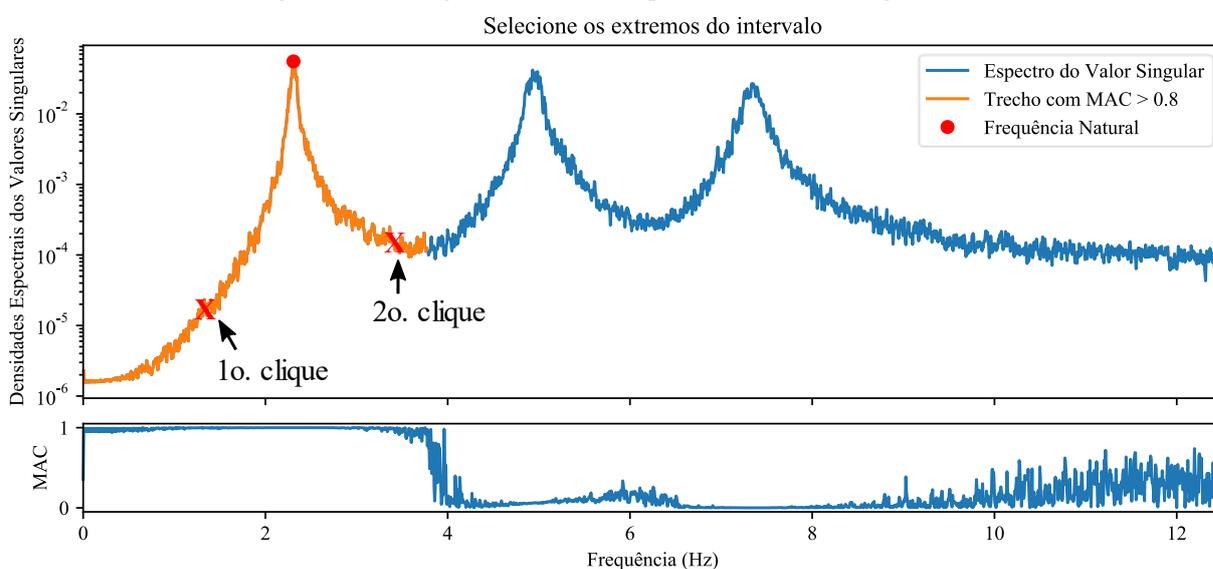


(fonte: elaborado pelo autor)

Em seguida, seleciona-se o trecho da densidade espectral no domínio da frequência. Fora desse trecho, o espectro terá valores nulos (Figura B.7).

Visando auxiliar na sua definição, calcula-se o Critério de Concordância Modal (MAC) entre o vetor singular do topo e os demais. Ele é uma medida de correlação entre duas formas modais. O seu valor varia entre 0 e 1, sendo igual zero para modos não correlacionados e um para modos perfeitamente correlacionados (para mais detalhes, ver item B.4.2). O trecho no qual $MAC > 0,8$ é indicado na cor laranja.

Figura B.7 – Seleção do trecho do espectro dos valores singulares



(fonte: elaborado pelo autor)

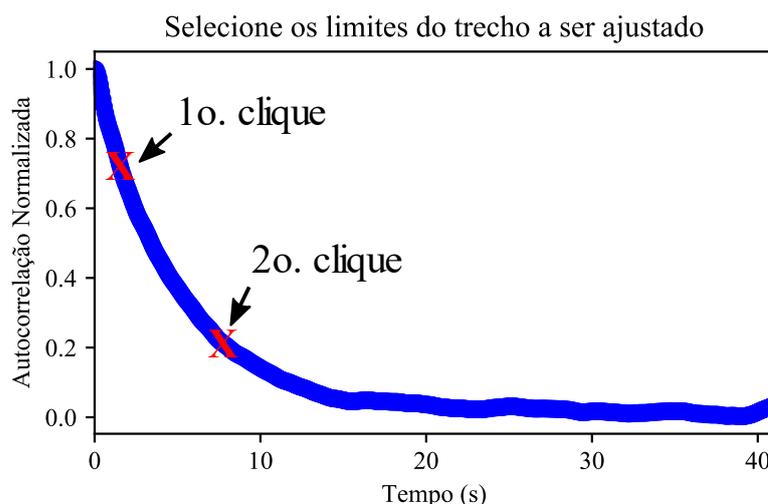
Transformando para o domínio do tempo a função de densidade espectral de cada modo de vibração, obtém-se a função de autocorrelação. A partir desta, determinam-se a frequência natural e a razão de amortecimento ajustando-se a expressão da resposta em vibração livre amortecida:

$$y(t) = y_p \exp(-\eta t) \cos(\omega_d t) \quad (\text{B.16})$$

Sendo y_p a amplitude, $\eta = \omega_n \xi$ e ω_d a frequência natural amortecida $\omega_d = \omega_n \sqrt{1 - \xi^2}$.

No gráfico, seleciona-se o trecho da função de autocorrelação a ser empregado para ajuste da função teórica (Figura B.8). Os valores são plotados em módulo visando facilitar a definição do trecho.

Figura B.8 – Seleção do trecho da função de autocorrelação utilizado para ajuste da função teórica



O ajuste é feito em dois passos. No primeiro, ajusta-se o envelope $y(t) = y_p \exp(-\eta t)$ obtendo-se η e y_p . No segundo passo, ajusta-se a resposta em vibração livre amortecida obtendo-se f_n e ξ .

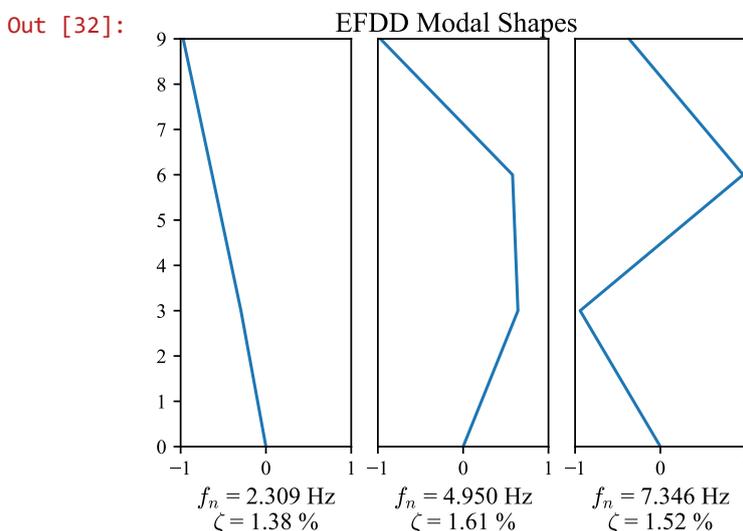
```
In [31]: input('Pressione Enter para continuar')

%matplotlib qt
FFDD, ZTFDD, VFDD, P2SD = SSI.EFDD(A,P2SD,plot=False)
```

```
Out [31]: Pressione Enter para continuar
Selecione os picos no gráfico
```

As propriedades dinâmicas identificadas são apresentadas abaixo.

```
In [32]: %matplotlib inline
        SSI.plot_1dshapes(FFDD,ZTFDD,VFDD,'EFDD',x,fix=[0])
```



Após o uso da função EFDD no modo 'interactive', o objeto P2SD recebe vários atributos:

```
In [33]: P2SD.pki # índice dos picos das densidades espectrais
```

```
Out [33]: array([379, 811, 1201], dtype=int64)
```

```
In [34]: P2SD.svi # índice dos valores singulares dos picos
```

```
Out [34]: array([0, 0, 0], dtype=int64)
```

```
In [35]: P2SD.fint # frequências inferior e superior empregadas para ajuste ao espectro analítico
```

```
Out [35]: array([1.3, 3.2, 4.4, 5.9, 6.7, 8.6])
```

```
In [36]: P2SD.tint # tempo inferior e superior empregados para ajuste da função de autocorrelação
```

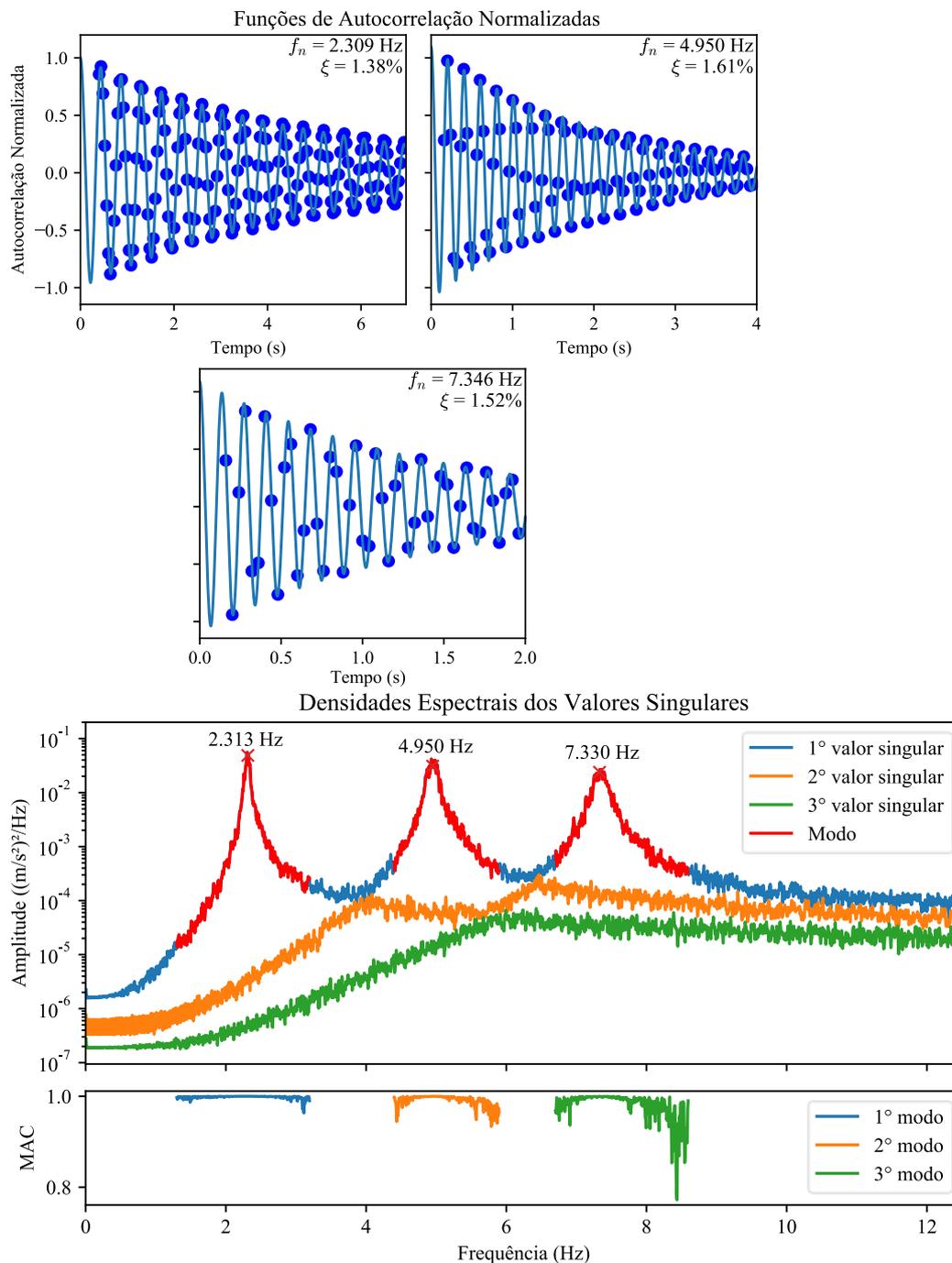
```
Out [36]: array([0.4, 7, 0.16, 4, 0.16, 2])
```

Esses atributos podem ser definidos antes da chamada da função. Eles só são empregados no modo 'batch'.

Os gráficos das funções de autocorrelação apresentam os pontos obtidos a partir da transformada inversa de Fourier das densidades espectrais e as curvas ajustadas para a obtenção das frequências naturais e razões de amortecimento.

```
In [37]: %matplotlib inline
         FFDD, ZTFDD, VFDD, P2SD = SSI.EFDD(A,P2SD,plot=True, mode='batch')
```

Out [37]:



B.3 IDENTIFICAÇÃO DE SISTEMAS NO DOMÍNIO DO TEMPO

Os métodos descritos a seguir são o Método das Variáveis Instrumentais (IV), o Método de Identificação Estocástica de Subespaços baseada nas Covariâncias (SSI COV) e baseada nas séries Temporais (SSI DATA), os quais empregam as funções de covariâncias ou diretamente

as séries temporais. A formulação dos métodos não segue exatamente a forma original com que foram propostos, sendo adotada uma formulação unificada bastante próxima da apresentada por Peeters (2000).

B.3.1 Séries temporais

As l saídas medidas são divididas em um conjunto de r sinais ou sensores de referência escolhidos adequadamente. O outro conjunto de resposta é composto de $(l - r)$ sensores. Rearranjando os dados de modo a posicionar as saídas de referência nas primeiras linhas:

$$\vec{y}_k = \begin{bmatrix} \vec{y}_k^{ref} \\ \vec{y}_k^{\sim} \end{bmatrix} \quad (\text{B.17})$$

Sendo \vec{y}_k^{ref} as saídas de referência e \vec{y}_k^{\sim} as demais saídas.

A função que rearranja os dados chama-se `rearrange_data` e recebe as séries temporais e a especificação dos sensores de referência:

```
In [38]: ref = (0,1,2) # todos os três sensores são de referência
         yk = SSI.rearrange_data(A,ref)
```

Algumas saídas são classificadas como saídas de referência: aquelas que contém informação modal relevante ou capturam uma resposta estrutural específica. A escolha dos sensores de referência corresponde a escolha das localizações das excitações em uma análise modal tradicional. O emprego de sensores de referência foi introduzido por Peeters (2000) e permite reduzir as matrizes do sistema, levando a algoritmos mais rápidos e com menor custo computacional. Caso o tempo de processamento seja de pouca importância, pode-se empregar todos os sensores como sensores de referência. Se, por outro lado, apenas o primeiro sensor for de referência, especifica-se `ref=(0)`.

Agrupam-se as saídas em uma matriz bloco $\mathbf{H}^{ref} \in \mathbb{R}^{(r+l)i \times N}$ (matriz cujos elementos da antidiagonal são iguais) composta por $2i$ linhas blocos e N colunas. As primeiras i linhas blocos possuem r linhas cada e as últimas i linhas bloco possuem l linhas. Essa matriz pode ser dividida em uma parte de saídas chamadas passadas (subscrito p) e uma parte de saídas futuras (subscrito f), conforme equação B.18:

$$\mathbf{H}^{ref} = \frac{1}{\sqrt{N}} \begin{bmatrix} \vec{y}_0^{ref} & \vec{y}_1^{ref} & \cdots & \vec{y}_{N-1}^{ref} \\ \vec{y}_1^{ref} & \vec{y}_2^{ref} & \cdots & \vec{y}_N^{ref} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{y}_{i-1}^{ref} & \vec{y}_i^{ref} & \cdots & \vec{y}_{i+N-2}^{ref} \\ \vec{y}_i & \vec{y}_{i+1} & \cdots & \vec{y}_{i+N-1} \\ \vec{y}_{i+1} & \vec{y}_{i+2} & \cdots & \vec{y}_{i+N} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{y}_{2i-1} & \vec{y}_{2i} & \cdots & \vec{y}_{2i+N-1} \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_{0|i-1}^{ref} \\ \mathbf{Y}_{i|2i-1} \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_p^{ref} \\ \mathbf{Y}_f \end{bmatrix} \begin{matrix} \updownarrow ri & \text{"passado"} \\ \updownarrow li & \text{"futuro"} \end{matrix} \quad (\text{B.18})$$

B.3.2 Estimativas das funções de covariâncias

As estimativas das funções de covariâncias são agrupadas em uma matriz Toeplitz

$\mathbf{T}_{1|i}^{ref} \in \mathbb{R}^{li \times ri}$ calculada com os dados da matriz Hankel:

$$\mathbf{T}_{1|i}^{ref} = \mathbf{Y}_f (\mathbf{Y}_p^{ref})^T = \begin{bmatrix} \mathbf{R}_i^{ref} & \mathbf{R}_{i-1}^{ref} & \cdots & \mathbf{R}_2^{ref} & \mathbf{R}_1^{ref} \\ \mathbf{R}_{i+1}^{ref} & \mathbf{R}_i^{ref} & \cdots & \mathbf{R}_3^{ref} & \mathbf{R}_2^{ref} \\ \mathbf{R}_{i+2}^{ref} & \mathbf{R}_{i+1}^{ref} & \cdots & \mathbf{R}_4^{ref} & \mathbf{R}_3^{ref} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{R}_{2i-1}^{ref} & \mathbf{R}_{2i-2}^{ref} & \cdots & \mathbf{R}_{i+1}^{ref} & \mathbf{R}_i^{ref} \end{bmatrix} \quad (\text{B.19})$$

O parâmetro i é escolhido pelo usuário e está associado à duração das funções de covariâncias, note que elas são estimadas até $2i - 1$ atrasos no tempo. Uma escolha usual encontrada na literatura técnica é tomar $i = 40$:

In [39]: i = 40

B.3.3 Métodos de Identificação Estocástica de Subespaços (SSI)

Os métodos de identificação estocástica de subespaços são formulados no espaço de estado em tempo discreto, no qual as matrizes do sistema são modeladas apenas em função das saídas:

$$\vec{x}_{k+1} = \mathbf{A}\vec{x}_k + \vec{w}_k \quad (\text{B.20})$$

$$\vec{y}_k = \mathbf{C}\vec{x}_k + \vec{z}_k \quad (\text{B.21})$$

Sendo $\vec{x}_k = [\vec{u}_k \quad \ddot{\vec{u}}_k]^T$ o vetor de estado no instante k ($k = 0, 1, \dots, N - 1$ com N o número de amostras no tempo), \vec{w}_k o vetor de perturbações e imprecisões no modelo, \vec{y}_k o vetor de saídas, \vec{z}_k o vetor de ruídos nas medições, \mathbf{A} a matriz dinâmica do sistema e \mathbf{C} a matriz de influências nas saídas.

Estimando as matrizes \mathbf{A} e \mathbf{C} com algoritmos empregando as covariâncias (SSI COV) ou as séries temporais (SSI DATA), as propriedades dinâmicas da estrutura são identificadas (VAN OVERSCHEE; DE MOOR, 1996; PEETERS, 2000). As frequências naturais f_i , as razões de amortecimento ξ_i e as formas modais $\vec{\phi}_i$ são obtidas:

$$\mathbf{A} = \Psi \mathbf{\Lambda}_D \Psi^{-1}, \quad \mathbf{\Lambda}_D = \text{diag}(\mu_i) \in \mathbb{C}^{n_2 \times n_2} \quad (\text{B.22a})$$

$$\lambda_i = \frac{\ln \mu_i}{\Delta t} \quad (\text{B.22b})$$

$$f_i = \frac{|\lambda_i|}{2\pi} \quad (\text{B.22c})$$

$$\xi_i = -\frac{\text{Re}(\lambda_i)}{|\lambda_i|} \quad (\text{B.22d})$$

$$\mathbf{V} = \mathbf{C}\Psi, \quad \mathbf{V} = [\vec{\phi}_1 \quad \dots \quad \vec{\phi}_n] \quad (\text{B.22e})$$

Sendo Ψ e $\mathbf{\Lambda}_D$ os autovetores e os autovalores de \mathbf{A} , μ_i e λ_i os autovalores em tempo discreto e contínuo, respectivamente. Δt o período de amostragem e $|\cdot|$ denota o módulo complexo.

O número de linhas do vetor de estado n_2 é chamado de ordem do modelo ou número de estados. O processo de identificação é realizado para modelos com ordens crescentes e a identificação dos modos de vibração verdadeiros é feita com auxílio do diagrama de estabilização. Para estruturas reais, não é possível prever a ordem que melhor se ajusta aos dados experimentais e que fornece propriedades dinâmicas mais realísticas. Assim, a forma mais apropriada é superestimá-la. O uso de modelos de alta ordem leva à introdução de modos

numéricos, os quais não têm importância física. Uma prática comum para separar os modos físicos dos espúrios é a criação de diagramas de estabilização. Nestes diagramas, os parâmetros modais estimados para todos os modelos são representados juntos, permitindo a identificação de modos estáveis para modelos de ordem crescente. Modos que aparecem com frequências, razões de amortecimento e formas modais semelhantes são classificados como estáveis e provavelmente são modos físicos.

Essa classificação depende da definição de limites de variação para as propriedades. As tolerâncias são especificadas em um *array* bidimensional *tol*. A primeira linha se refere às frequências, a segunda às razões de amortecimento e a terceira ao MAC entre as formas modais. Na primeira coluna encontram-se as variações relativas limites para um modo ser classificado como estável, na segunda coluna o limite inferior e na terceira o superior. Por exemplo, `[0.01, 0, 15]` indica que a tolerância para frequências é de 1%, a frequência natural mínima é de 0 Hz e a máxima é de 15 Hz. Valores fora desse intervalo são associados com modos espúrios.

```
In [40]: # tolerâncias: variação, limite inferior e limite superior
tol = np.array([[0.01, 0, 15],      # frequência
               [0.05, 0, .03],    # amortecimento
               [0.10, 0, 1]])     # MAC
```

B.3.3.1 Identificação baseada nas funções de covariâncias (SSI COV)

O método de Identificação Estocástica de Subespaços baseado nas Covariâncias aborda o problema de realização estocástica, ou seja, o problema de identificar um modelo estocástico em espaço de estado com apenas saídas medidas. Este método baseia-se na decomposição em valores singulares da matriz de covariâncias das saídas (B.19). Uma descrição detalhada é apresentada por Peeters (2000).

A função `SSI_COV` identifica o sistema de uma certa ordem em espaço de estado informada e a função `SSI_COV_iterator` itera a função `SSI_COV` para modelos de ordens crescentes. A matriz Toeplitz e a sua decomposição em valores singulares são calculadas uma única vez, devendo-se respeitar a inequação $ri \geq n_{2,max}$ sendo $n_{2,max}$ a ordem máxima do modelo em espaço de estado. Modelos de ordens inferiores $n_2 < n_{2,max}$ são facilmente obtidos simplesmente selecionando-se menos valores singulares.

Considerando um modelo de ordem mínima igual a 2, máxima igual a 24 e com incrementos padrão de 2 em 2:

```
In [41]: FNC, ZTC, VVC = SSI.SSI_COV_iterator(yk,i,2,24)
```

FNC e ZTC são *arrays* bidimensionais com as frequências e razões de amortecimento identificados. Cada linha se refere a um modelo. As propriedades do modelo de ordem $n_2 = 2$ encontram-se na primeira linha e de ordem $n_2 = 24$, na última. Por exemplo, as frequências e razões de amortecimento para o modelo de ordem $n_2 = 16$ são:

```
In [42]: df = pd.DataFrame(np.array((FNC[7, :], ZTC[7, :])))
df.style.format('{:.3f}')
```

```
Out [42]:
```

	0	1	2	3	4	5
0	2.311	2.311	2.833	2.833	4.943	4.943
1	0.016	0.016	-0.008	-0.008	0.014	0.014
6	7	8	9	10	11	12
4.877	4.877	5.379	5.379	7.800	7.800	7.340
-0.010	-0.010	-0.014	-0.014	-0.005	-0.005	0.015
13	14	15	16	17	18	19
7.340	7.208	7.208	0.000	0.000	0.000	0.000
0.015	-0.003	-0.003	0.000	0.000	0.000	0.000
20	21	22	23			
0.000	0.000	0.000	0.000			
0.000	0.000	0.000	0.000			

E as respectivas formas modais:

```
In [43]: df = pd.DataFrame(VVC[7, :, :])
df.style.format('{:.3f}')
```

```
Out [43]:
```

	0	1	2	3	4
0	-4.006-1.199j	-4.006+1.199j	0.095-0.167j	0.095+0.167j	-2.814+3.639j
1	-2.594-0.777j	-2.594+0.777j	0.039-0.098j	0.039+0.098j	1.694-2.219j
2	-1.207-0.361j	-1.207+0.361j	0.024-0.037j	0.024+0.037j	1.901-2.482j
5	6	7	8	9	
-2.814-3.639j	-0.673+0.361j	-0.673-0.361j	-0.006-0.208j	-0.006+0.208j	
1.694+2.219j	0.385-0.213j	0.385+0.213j	-0.000+0.100j	-0.000-0.100j	
1.901+2.482j	0.445-0.256j	0.445+0.256j	-0.008+0.037j	-0.008-0.037j	
10	11	12	13	14	
0.048+0.154j	0.048-0.154j	1.456-1.023j	1.456+1.023j	-0.019-0.259j	
-0.141-0.204j	-0.141+0.204j	-3.686+2.469j	-3.686-2.469j	-0.352+0.307j	
0.206+0.122j	0.206-0.122j	3.532-2.329j	3.532+2.329j	0.410-0.237j	
15	16	17	18	19	
-0.019+0.259j	0.000+0.000j	0.000+0.000j	0.000+0.000j	0.000+0.000j	
-0.352-0.307j	0.000+0.000j	0.000+0.000j	0.000+0.000j	0.000+0.000j	
0.410+0.237j	0.000+0.000j	0.000+0.000j	0.000+0.000j	0.000+0.000j	

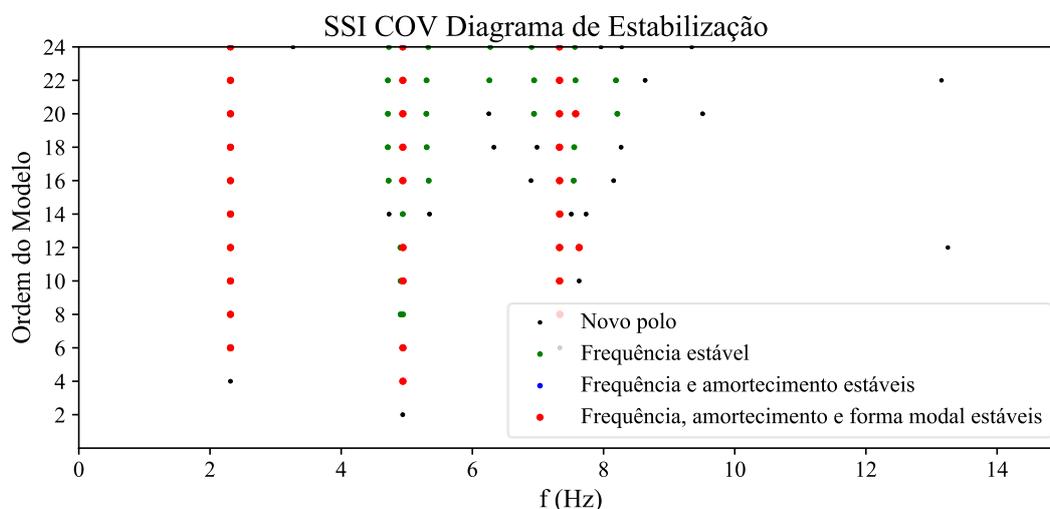
20	21	22	23
0.000+0.000j	0.000+0.000j	0.000+0.000j	0.000+0.000j
0.000+0.000j	0.000+0.000j	0.000+0.000j	0.000+0.000j
0.000+0.000j	0.000+0.000j	0.000+0.000j	0.000+0.000j

Nota-se que as propriedades dinâmicas aparecem aos pares. Isso acontece por causa da definição do vetor de estado. Como consequência, o mesmo modo é representado por dois estados.

Desenha-se um diagrama de estabilização e os modos físicos são identificados a partir dos polos estáveis. Um novo modo detectado é representado por um ponto na cor preta. Caso apenas a sua frequência estabilize (respeitando os limites de $\text{tol}[0]$), representa-se o ponto com a cor verde. Se além da frequência, a razão de amortecimento respeitar $\text{tol}[1]$, o ponto recebe a cor azul. Por fim, caso a frequência, razão de amortecimento e forma modal sejam estáveis, ou seja, se elas respeitam todos os limites especificados em tol , desenha-se o ponto na cor vermelha.

In [44]: `stbC = SSI.stabilizationdiagram(FNC,ZTC,VVC,'SSI COV', tol=tol)`

Out [44]:



Os modos físicos são aqueles que apresentam propriedades dinâmicas estáveis para modelos de diferentes ordens. Os pontos vermelhos, formando uma reta vertical, provavelmente estão associadas a modos físicos.

`stbC` é um *array* booleano indicando os modos estáveis. É possível selecionar as frequências naturais e as razões de amortecimento desses modos:

In [45]: `df = pd.DataFrame(np.array((FNC[stbC],ZTC[stbC])))
df.style.format('{:.4f}')`

Out [45]:

	0	1	2	3	4	5
0	4.9455	4.9455	4.9454	4.9454	2.3107	2.3107
1	0.0163	0.0163	0.0164	0.0164	0.0163	0.0163
6	7	8	9	10	11	12
2.3107	2.3107	7.3439	7.3439	4.9367	4.9367	2.3107
0.0163	0.0163	0.0148	0.0148	0.0157	0.0157	0.0163
13	14	15	16	17	18	19
2.3107	4.9367	4.9367	7.3469	7.3469	2.3107	2.3107
0.0163	0.0155	0.0155	0.0151	0.0151	0.0163	0.0163
20	21	22	23	24	25	26
4.9367	4.9367	7.3436	7.3436	2.3107	2.3107	4.9343
0.0155	0.0155	0.0145	0.0145	0.0163	0.0163	0.0163
27	28	29	30	31	32	33
4.9343	7.3443	7.3443	2.3108	2.3108	7.3402	7.3402
0.0163	0.0146	0.0146	0.0163	0.0163	0.0150	0.0150
34	35	36	37	38	39	40
2.3108	2.3108	4.9433	4.9433	7.3425	7.3425	2.3108
0.0163	0.0163	0.0144	0.0144	0.0147	0.0147	0.0163
41	42	43	44	45	46	47
2.3108	4.9429	4.9429	7.3456	7.3456	2.3108	2.3108
0.0163	0.0144	0.0144	0.0144	0.0144	0.0163	0.0163
48	49	50	51	52	53	
7.3427	7.3427	2.3108	2.3108	4.9437	4.9437	
0.0142	0.0142	0.0163	0.0163	0.0160	0.0160	

E as respectivas formas modais:

```
In [46]: VE = [[],[],[ ]]
for j in range(stbC.shape[0]):
    VE = np.hstack((VE,VVC[j,:],stbC[j]).T)
df = pd.DataFrame(VE)
df.style.format('{:.3f}')
```

Out [46]:

	0	1	2	3	4
0	-3.925-2.889j	-3.925+2.889j	-3.957-2.892j	-3.957+2.892j	-4.010-1.198j
1	2.334+1.740j	2.334-1.740j	2.413+1.748j	2.413-1.748j	-2.595-0.779j
2	2.771+1.967j	2.771-1.967j	2.696+1.958j	2.696-1.958j	-1.208-0.362j
5	6	7	8	9	
-4.010+1.198j	4.010+1.200j	4.010-1.200j	1.490-1.036j	1.490+1.036j	
-2.595+0.779j	2.595+0.778j	2.595-0.778j	-3.701+2.535j	-3.701-2.535j	
-1.208+0.362j	1.208+0.361j	1.208-0.361j	3.537-2.398j	3.537+2.398j	
10	11	12	13	14	
-2.984+3.672j	-2.984-3.672j	-4.010-1.200j	-4.010+1.200j	2.974-3.656j	
1.792-2.241j	1.792+2.241j	-2.595-0.778j	-2.595+0.778j	-1.791+2.229j	
2.020-2.502j	2.020+2.502j	-1.208-0.361j	-1.208+0.361j	-2.005+2.497j	
15	16	17	18	19	
2.974+3.656j	1.484-1.040j	1.484+1.040j	-4.010-1.200j	-4.010+1.200j	
-1.791-2.229j	-3.671+2.567j	-3.671-2.567j	-2.596-0.778j	-2.596+0.778j	
-2.005-2.497j	3.497-2.434j	3.497+2.434j	-1.207-0.362j	-1.207+0.362j	

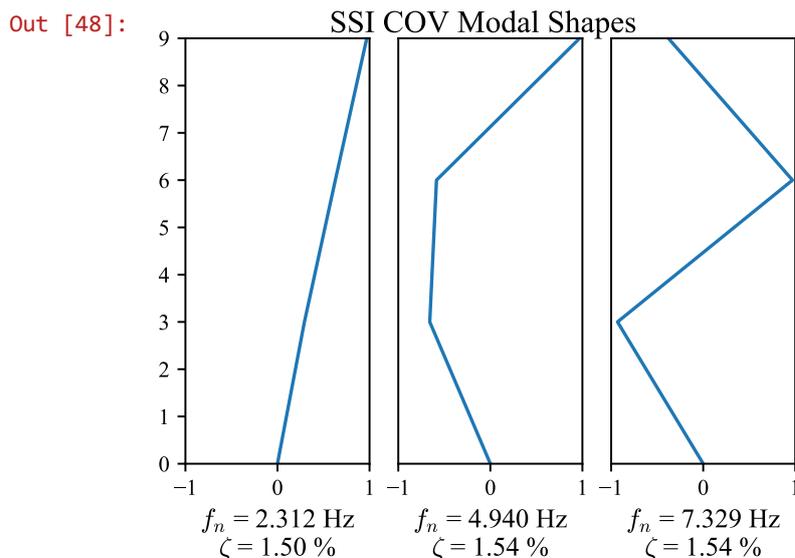
20	21	22	23	24
-2.973+3.654j	-2.973-3.654j	1.472-1.049j	1.472+1.049j	-4.010-1.200j
1.792-2.228j	1.792+2.228j	-3.641+2.499j	-3.641-2.499j	-2.596-0.777j
2.003-2.496j	2.003+2.496j	3.469-2.347j	3.469+2.347j	-1.207-0.361j
25	26	27	28	29
-4.010+1.200j	3.013-3.628j	3.013+3.628j	-1.480+1.053j	-1.480-1.053j
-2.596+0.777j	-1.815+2.223j	-1.815-2.223j	3.642-2.511j	3.642+2.511j
-1.207+0.361j	-2.031+2.486j	-2.031-2.486j	-3.469+2.360j	-3.469-2.360j
30	31	32	33	34
-4.006-1.199j	-4.006+1.199j	1.456-1.023j	1.456+1.023j	-4.006-1.199j
-2.594-0.777j	-2.594+0.777j	-3.686+2.469j	-3.686-2.469j	-2.594-0.777j
-1.207-0.361j	-1.207+0.361j	3.532-2.329j	3.532+2.329j	-1.207-0.361j
35	36	37	38	39
-4.006+1.199j	-2.813+3.641j	-2.813-3.641j	1.461-1.041j	1.461+1.041j
-2.594+0.777j	1.698-2.213j	1.698+2.213j	-3.645+2.489j	-3.645-2.489j
-1.207+0.361j	1.896-2.486j	1.896+2.486j	3.493-2.335j	3.493+2.335j
40	41	42	43	44
-4.006-1.199j	-4.006+1.199j	2.816-3.634j	2.816+3.634j	1.458-1.052j
-2.595-0.777j	-2.595+0.777j	-1.700+2.207j	-1.700-2.207j	-3.592+2.505j
-1.207-0.361j	-1.207+0.361j	-1.898+2.484j	-1.898-2.484j	3.431-2.346j
45	46	47	48	49
1.458+1.052j	4.006+1.199j	4.006-1.199j	1.438-1.032j	1.438+1.032j
-3.592-2.505j	2.594+0.777j	2.594-0.777j	-3.566+2.443j	-3.566-2.443j
3.431+2.346j	1.207+0.361j	1.207-0.361j	3.422-2.286j	3.422+2.286j
50	51	52	53	
-4.006-1.199j	-4.006+1.199j	-2.957+3.834j	-2.957-3.834j	
-2.595-0.777j	-2.595+0.777j	1.776-2.328j	1.776+2.328j	
-1.207-0.361j	-1.207+0.361j	1.996-2.605j	1.996+2.605j	

A função `stable_modes` agrupa os polos estáveis. A tolerância relativa em frequência `tol=0.01` deve ser especificada para que os polos sejam considerados do mesmo modo (neste exemplo é de 1%). Também deve-se especificar a quantidade mínima de modelos para que um modo seja classificado como estável. Por exemplo, `spo=8` indica que devem ter no mínimo oito polos estáveis com uma mesma frequência para que ela seja considerada um modo físico.

```
In [47]: FNCr, ZTCr, VVCR = SSI.stable_modes(FNC, ZTC, VVC, stbC, tol=0.01, spo=8)
```

As propriedades dinâmicas identificadas são:

```
In [48]: SSI.plot_1dshapes(FNCr, ZTCr, VVCR, 'SSI COV', x, ref, fix=[0])
```



B.3.3.2 SSI DATA

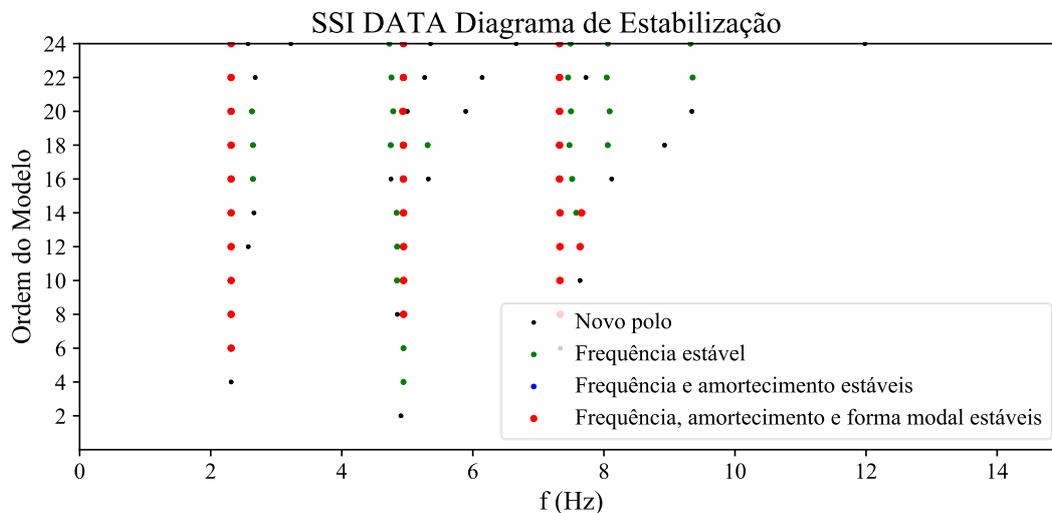
O método SSI DATA evita o cálculo das funções de covariância das saídas. Ele é substituído pela projeção das linhas das saídas futuras nas linhas das saídas passadas, a qual é calculada a partir da fatoração QR da matriz bloco Hankel com séries temporais das saídas (B.18). De fato, o conceito de covariâncias e projeções estão intimamente ligados. Ambos visam cancelar o ruído. A ideia da projeção é reter toda a informação passada útil para prever o futuro. O método emprega a decomposição em valores singulares da matriz com as projeções para obter a matriz de observabilidade e a sequência de estado no filtro Kalman. As matrizes do sistema são obtidas a partir da sequência de estado do filtro Kalman (VAN OVERSCHEE; DE MOOR, 1996).

As mesmas observações feitas no método SSI COV sobre a determinação da ordem do modelo em espaço de estado n_2 se aplicam ao SSI DATA. A função `SSI_DATA_iterator` itera a função `SSI_DATA` para modelos de ordens crescentes. Desenha-se um diagrama de estabilização e os modos verdadeiros são identificados a partir dos polos estáveis. A decomposição QR, as projeções e a sua decomposição em valores singulares são calculadas uma única vez, devendo-se respeitar a inequação $r_i \geq n_{2,max}$ sendo $n_{2,max}$ a ordem máxima do modelo em espaço de estado. Modelos de ordens inferiores $n_2 < n_{2,max}$ são facilmente obtidos simplesmente selecionando-se menos valores singulares.

Considerando um modelo de ordem mínima igual a 2, máxima igual a 24 e com incrementos padrão de 2 em 2.

```
In [49]: FND, ZTD, VVD = SSI.SSI_DATA_iterator(yk,i,2,24,2)
         stbD = SSI.stabilizationdiagram(FND,ZTD,VVD,'SSI DATA', tol=tol)
```

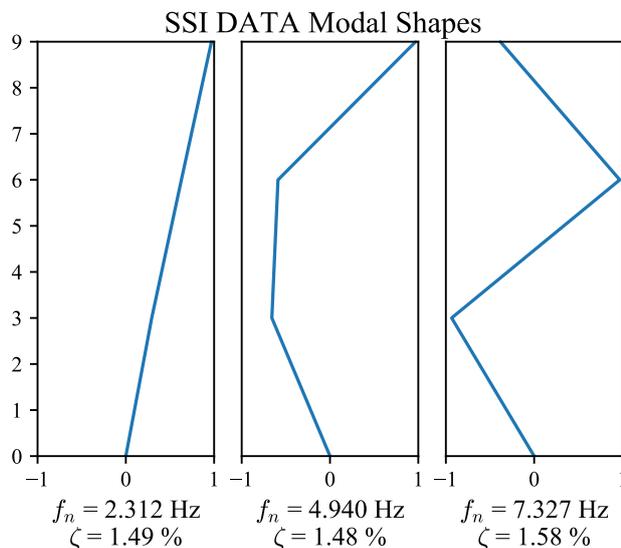
Out [49]:



Em seguida, agrupam-se os polos estáveis e plotam-se as propriedades dinâmicas identificadas.

```
In [50]: FNDR, ZTDR,VVDR = SSI.stable_modes(FND, ZTD, VVD, stbD, tol=0.01, spo=8)
         SSI.plot_1dshapes(FNDR,ZTDR,VVDR,'SSI DATA',x,ref,fix=[0])
```

Out [50]:



B.3.4 Método das Variáveis Instrumentais (IV)

O IV emprega a representação autorregressiva com média móvel (ARMA) e identifica os coeficientes da parte AR apenas, evitando a não-linearidade causada pela parte MA. Pode-se mostrar que para a determinação das características dinâmicas da estrutura, apenas os coeficientes da parte AR são necessários. A ideia é ajustar os coeficientes para os dados das

medições, resolvendo um sistema de equações sobredeterminado por mínimos quadrados (PEETERS, 2000).

O método programado emprega a formulação com os sensores de referência proposta por Peeters (2000). Ela baseia-se no modelo autorregressivo com média móvel reverso no tempo:

$$\vec{y}_k^{ref} + \alpha_1^b \vec{y}_{k+1}^{ref} + \dots + \alpha_p^b \vec{y}_{k+p}^{ref} = \vec{e}_k^b + \gamma_1^b \vec{e}_{k+1}^b + \dots + \gamma_p^b \vec{e}_{k+p}^b \quad (\text{B.23})$$

Sendo $\vec{y}_k^{ref} \in \mathbb{R}^r$ o vetor das saídas de referência e $\vec{e}_k^b \in \mathbb{R}^r$ um vetor ruído branco. Os termos ao lado esquerdo da igualdade são a parte autorregressiva (AR) e os termos ao lado direito são a parte com média móvel (MA). As matrizes $\alpha_1^b \in \mathbb{R}^{r \times r}$ contém os parâmetros da parte AR e as matrizes $\gamma_1^b \in \mathbb{R}^{r \times r}$ contém os parâmetros da parte MA.

A relação entre a ordem p do modelo ARMA e seu modelo equivalente no espaço de estado é $p \times r = n_2$.

Os coeficientes da parte AR reversa são determinados resolvendo um sistema de equações por mínimos quadrados com partes da matriz Toeplitz. Estes coeficientes são agrupados em uma matriz chamada de matriz de estado na forma companheira $\mathbf{A}_{\text{comp}}^{b,p}$. As matrizes com os autovalores e autovetores da matriz companheira são calculadas, Λ_D e Ψ_{comp} , respectivamente. A partir dos autovalores μ_i calculam-se as frequências naturais e razões de amortecimento.

As formas modais são determinadas em um segundo passo. A matriz \mathbf{G}_m^{ref} é obtida a partir das primeira r linhas da matriz com os autovetores de $\mathbf{A}_{\text{comp}}^{b,p}$:

$$\mathbf{G}_m^{ref} = (\Psi_{\text{comp}}[:, r, :])^T \quad (\text{B.24})$$

A matriz $\Gamma_{p,m}^{ref}$ é calculada:

$$\Gamma_{p,m}^{ref} = [\Lambda_D^{p-1} \mathbf{G}_m^{ref} \quad \Lambda_D^{p-2} \mathbf{G}_m^{ref} \quad \dots \quad \Lambda_D \mathbf{G}_m^{ref} \quad \mathbf{G}_m^{ref}] \quad (\text{B.25})$$

E finalmente as formas modais:

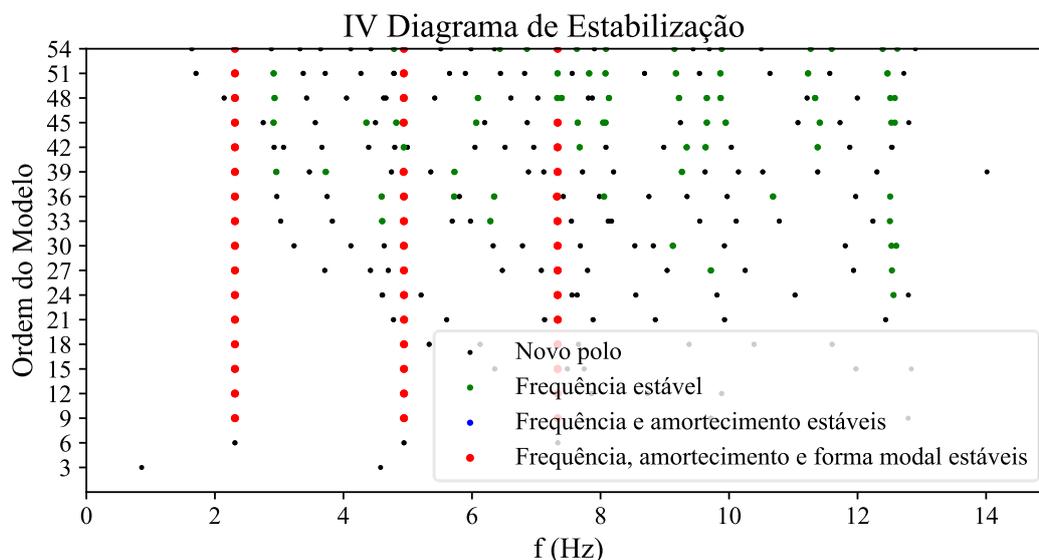
$$\mathbf{V} = [\mathbf{R}_p^{ref} \quad \mathbf{R}_{p-1}^{ref} \quad \dots \quad \mathbf{R}_1^{ref}] (\mathbf{\Gamma}_{p,m}^{ref})^{-1} \quad (\text{B.26})$$

Semelhante aos métodos anteriores, o processo de identificação é realizado para modelos com ordens crescentes e a identificação dos modos de vibração verdadeiros é feita com auxílio do diagrama de estabilização. A função `IV_iterator` faz esse processo. Nela, especifica-se uma ordem máxima p_{max} do modelo ARMA, a partir da qual a ordem máxima do modelo em espaço de estado $n_{2,max}$ é obtida fazendo-se $n_{2,max} = p_{max} \times r$ e a matriz Toeplitz é calculada uma única vez. Modelos de ordens inferiores $p < p_{max}$ são facilmente obtidos simplesmente selecionando-se menos elementos da matriz Toeplitz.

Para um modelo de ordem ARMA mínima igual a 1, máxima igual a 18 com incrementos de 1 em 1:

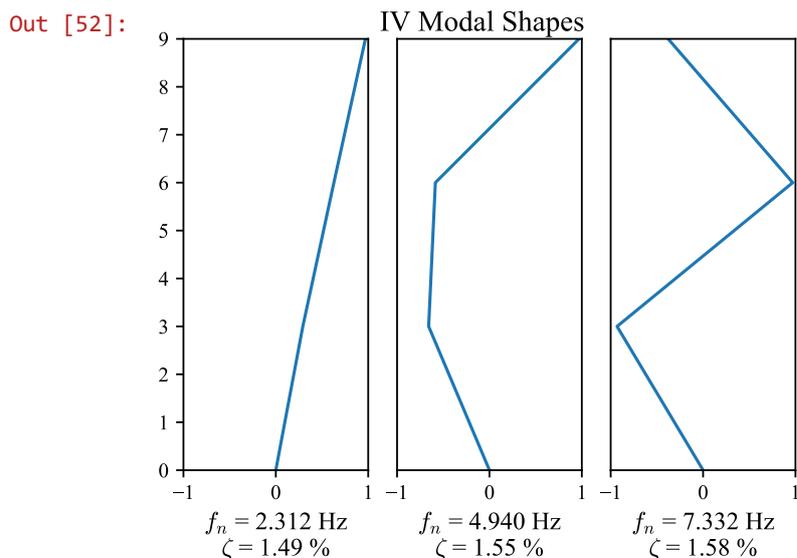
```
In [51]: FNI, ZTI, VVI = SSI.IV_iterator(yk,i,1,18,1)
         stbI = SSI.stabilizationdiagram(FNI, ZTI, VVI, 'IV', tol=tol)
```

Out [51]:



As propriedades dinâmicas identificadas:

```
In [52]: FNIR, ZTIR, VVIR = SSI.stable_modes(FNI, ZTI, VVI, stbI, tol=0.01, spo=8)
         SSI.plot_1dshapes(FNIR, ZTIR, VVIR, 'IV', x, ref, fix=[0])
```



B.4 OUTRAS ROTINAS

Nesta seção apresentam-se as rotinas para geração de gráficos das propriedades dinâmicas de edifícios, cálculo do Critério de Concordância Modal e geração de gráfico dos valores singulares.

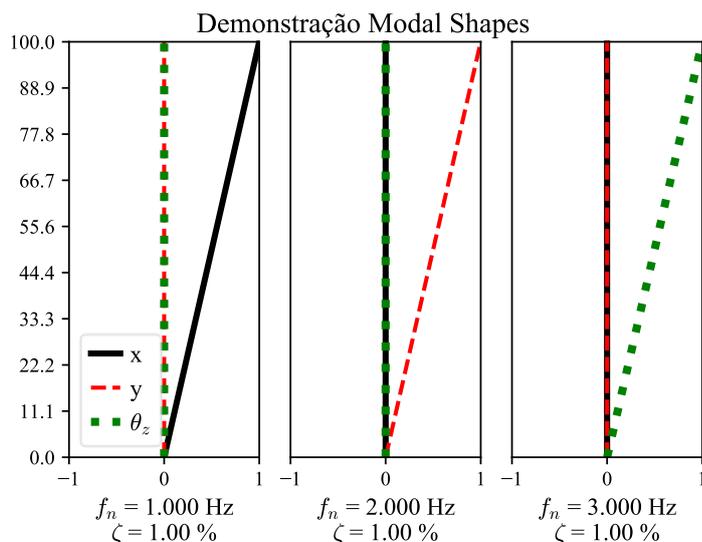
B.4.1 Gráfico das propriedades dinâmicas de edifícios

Elaborou-se a rotina `plot_3das1d` para representação das propriedades dinâmicas de edifícios. Cada forma modal deve estar nas colunas da matriz informada, sendo as linhas com índices $0:3$ para as componentes na direção x , as linhas com índices $1:3$ para componentes na direção y e $2:3$ para as componentes em θ_z .

```
In [53]: fnt = np.array((1,2,3))      # frequências naturais
          ztt = 0.01*np.ones(3)     # razões de amortecimento
          qt = np.array([[1,0,0],   # componentes x das formas modais
                        [0,1,0],   # componentes y das formas modais
                        [0,0,1]])  # componentes  $\theta_z$  das formas modais
          xt = np.array((100,))     # altura do sensor

          SSI.plot_3das1d(fnt,ztt,qt,xt,title='Demonstração')
```

Out [53]:



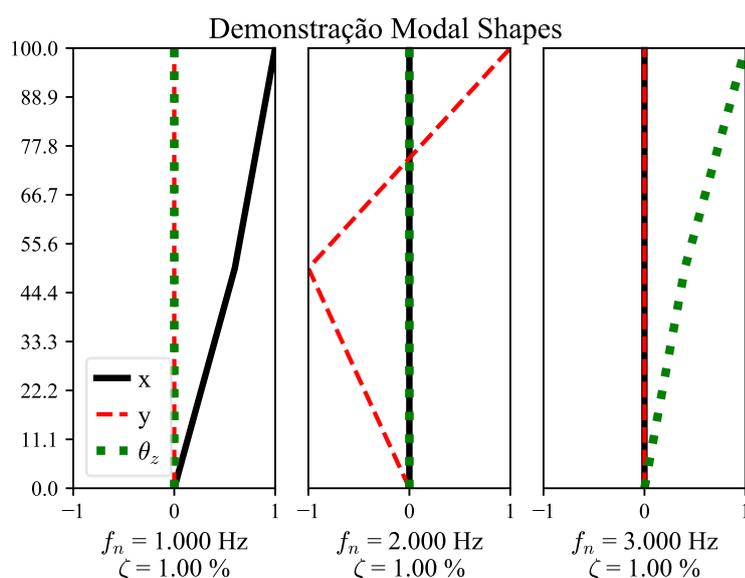
Na formulação com sensores de referência, as componentes modais desses sensores são posicionadas nas primeiras linhas das formas modais identificadas. Como consequência, é necessário rearranjar as formas modais. Isso é feito especificando uma tupla `ref`. Se os sensores de referência estiverem em `xt[1]`, especifica-se `ref=(1)`.

In [54]:

```
fnt = np.array((1,2,3))
ztt = 0.01*np.ones(3)
qt = np.array([[ 1,0,0],[0, 1,0],[0,0, 1], # componentes modais para xt[0]
               [0.6,0,0],[0,-1,0],[0,0,0.4]]) # componentes modais para xt[1]
xt = np.array((100,50))

SSI.plot_3das1d(fnt,ztt,qt,xt,title='Demonstração',ref=(0))
```

Out [54]:



B.4.2 Critério de Concordância Modal (MAC)

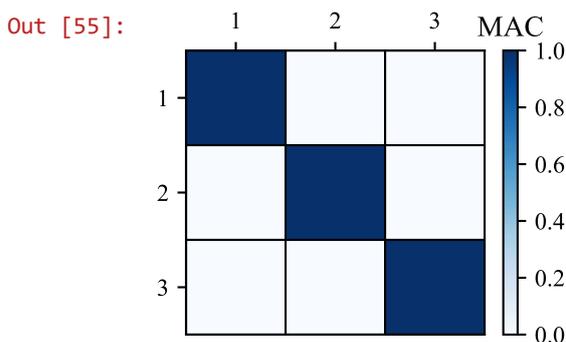
O Critério de Concordância Modal (*Modal Assurance Criterion* – MAC), proposto por Allemang e Brown (1982), é uma medida de correlação entre duas formas modais. O seu valor varia entre 0 e 1, sendo igual zero para modos não correlacionados e um para modos perfeitamente correlacionados. Sejam as formas modais complexas $\vec{\psi}_i$ e $\vec{\psi}_j$, o MAC é calculado:

$$\text{MAC}(i,j) = \frac{|\vec{\psi}_i^T \vec{\psi}_j^*|^2}{(\vec{\psi}_i^T \vec{\psi}_i^*)(\vec{\psi}_j^T \vec{\psi}_j^*)} \quad (\text{B.27})$$

Na qual $(.)^*$ denota complexo conjugado. Os vetores $\vec{\psi}_i$ e $\vec{\psi}_j$ podem ser substituídos por matrizes cujas colunas são as formas modais. Neste caso, o emprego da expressão B.27 resultará em uma matriz.

Uma forma útil de visualizar os valores do MAC é de forma gráfica, permitindo que mais dados sejam apresentados e compreendidos.

In [55]: `MAC = SSI.MAC(qt,qt,plot=True)`



B.4.3 Gráfico dos valores singulares

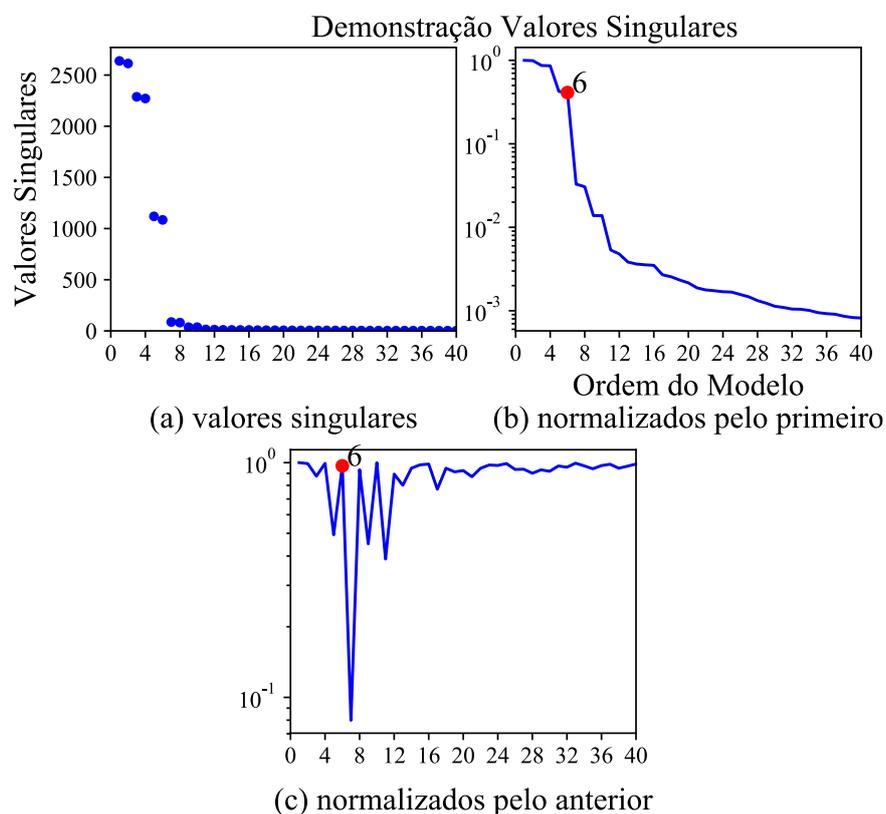
A contagem dos valores singulares não nulos fornece uma estimativa do número de modos físicos presentes nos sinais. Quando a ordem verdadeira do modelo for excedida, os valores singulares irão teoricamente para zero. Na figura abaixo é possível identificar seis valores singulares não nulos, indicando que se trata de um modelo com ordem no espaço de estado

$n_2 = 6$. Para dados experimentais, os valores singulares não são exatamente zero, mas muito pequenos. Assim, a taxa de variação dos valores singulares em função da ordem do modelo é utilizada como indicador. Uma abordagem é normalizar os valores singulares pelo primeiro (e maior), conforme gráfico (b). Outra abordagem é dividir cada valor singular pelo imediatamente anterior (c) formando uma curva com valores próximos da unidade se os valores não mudarem de magnitude. A ordem do modelo é identificada quando a relação cai subitamente.

A função `plot_singular_values` plota os gráficos dos valores singulares.

```
In [56]: T = SSI.Toeplitz(yk, i)
T.method = 'Demonstração'
SSI.plot_singular_values(T, figsize=(13,4))
```

Out [56]:



**APÊNDICE C – Frequências naturais e razões de amortecimento
da Ponte Z24**

C.1 MÉTODO SSI COV

Quadro C.1 – Estimativas das frequências naturais da Ponte Z24 pelo método SSI COV

Modo	Frequência natural (Hz)										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	3,87	3,87	3,85	3,85	3,86	3,85	3,86	3,86	3,85	3,86	0,009
2	4,91	4,91	4,90	4,88	4,89	4,89	4,91	4,92	4,88	4,90	0,014
3	9,76	9,77	9,76	9,74	9,76	9,77	9,71	9,74	9,75	9,75	0,018
4	10,25	10,24	10,28	10,28	10,26	10,24	10,32	10,31	10,40	10,29	0,050
5	12,50	12,45	12,48	12,01	12,39	12,48	12,47	12,43	12,43	12,40	0,151
6	13,63	13,43	13,14	13,23	13,21	13,43	13,21	13,19	13,16	13,29	0,168
8	19,08	19,04	19,14	19,08	19,13	19,11	19,15	19,59	19,18	19,16	0,165

(fonte: elaborado pelo autor)

Quadro C.2 – Estimativas das razões de amortecimento da Ponte Z24 pelo método SSI COV

Modo	Razão de amortecimento										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	0,94%	0,95%	0,98%	0,43%	0,80%	0,83%	0,83%	0,92%	0,86%	0,84%	0,17%
2	1,22%	1,20%	1,23%	1,27%	1,23%	0,91%	1,49%	1,41%	1,40%	1,26%	0,17%
3	1,52%	1,31%	1,28%	1,31%	1,70%	1,26%	1,27%	1,27%	1,33%	1,36%	0,15%
4	1,45%	1,08%	1,51%	1,40%	1,69%	1,19%	1,44%	0,99%	2,38%	1,46%	0,41%
5	3,40%	3,08%	3,34%	2,31%	2,27%	3,47%	2,67%	2,64%	2,90%	2,90%	0,46%
6	5,81%	4,02%	4,65%	2,68%	3,75%	5,10%	4,06%	2,08%	5,12%	4,14%	1,20%
8	2,02%	2,01%	1,94%	2,02%	1,66%	2,35%	1,86%	3,89%	1,95%	2,19%	0,66%

(fonte: elaborado pelo autor)

C.2 MÉTODO SSI DATA

Quadro C.3 – Estimativas das frequências naturais da Ponte Z24 pelo método SSI DATA

Modo	Frequência natural (Hz)										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	3,87	3,86	3,85	3,85	3,86	3,85	3,85	3,85	3,85	3,86	0,006
2	4,91	4,93	4,91	4,89	4,89	4,90	4,92	4,94	4,88	4,91	0,019
3	9,75	9,75	9,76	9,74	9,75	9,78	9,71	9,73	9,75	9,75	0,018
4	10,24	10,26	10,28	10,27	10,26	10,23	10,32	10,37	10,40	10,29	0,059
5	12,46	12,50	12,42	12,00	12,40	12,56	12,48	12,49	12,42	12,42	0,162
6	13,16	13,44	13,27	13,24	13,27	13,33	13,37	13,31	13,59	13,33	0,126
8	19,04	19,04	19,04	18,99	19,17	19,05	19,00	19,01	19,20	19,06	0,074

(fonte: elaborado pelo autor)

Quadro C.4 – Estimativas das razões de amortecimento da Ponte Z24 pelo método SSI DATA

Modo	Razão de amortecimento										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	1,06%	1,12%	1,04%	0,63%	0,83%	1,06%	0,85%	1,05%	0,89%	0,95%	0,16%
2	1,33%	1,44%	1,27%	1,16%	1,20%	0,80%	1,39%	1,22%	1,39%	1,24%	0,19%
3	1,53%	1,21%	1,37%	1,29%	1,67%	1,08%	1,18%	1,32%	1,45%	1,34%	0,18%
4	1,41%	1,43%	1,41%	1,46%	1,71%	1,12%	1,61%	1,65%	2,40%	1,58%	0,35%
5	3,38%	3,07%	3,05%	2,68%	2,44%	3,32%	2,30%	2,78%	2,71%	2,86%	0,37%
6	3,64%	3,40%	3,78%	2,36%	4,11%	3,98%	3,60%	2,35%	5,77%	3,67%	1,02%
8	1,93%	1,98%	2,45%	2,56%	3,22%	2,55%	2,02%	1,52%	2,29%	2,28%	0,49%

(fonte: elaborado pelo autor)

C.3 MÉTODO IV

Quadro C.5 – Estimativas das frequências naturais da Ponte Z24 pelo método IV

Modo	Frequência natural (Hz)										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	3,87	3,87	3,85	3,85	3,86	3,85	3,85	3,86	3,85	3,86	0,007
2	4,89	4,91	4,89	4,89	4,89	4,89	4,91	4,93	4,88	4,90	0,015
3	9,79	9,77	9,76	9,74	9,75	9,76	9,70	9,77	9,75	9,76	0,025
4	10,25	10,24	10,29	10,28	10,26	10,22	10,31	10,32	10,40	10,29	0,055
5	12,46	12,43	12,51	11,94	12,39	12,51	12,46	12,48	12,41	12,40	0,177
6	13,19	13,46	13,10	13,23	13,17	13,37	13,12	13,40	13,27	13,26	0,129
8	19,01	18,97	19,01	19,00	19,14	19,12	19,13	18,77	19,18	19,04	0,125

(fonte: elaborado pelo autor)

Quadro C.6 – Estimativas das razões de amortecimento da Ponte Z24 pelo método IV

Modo	Razão de amortecimento										
	Setup									Média	Desvio padrão
	1	2	3	4	5	6	7	8	9		
1	0,91%	1,01%	1,06%	0,65%	0,78%	1,04%	0,84%	0,77%	0,88%	0,88%	0,14%
2	1,55%	1,42%	1,31%	1,31%	1,31%	1,30%	1,59%	1,62%	1,39%	1,42%	0,13%
3	1,61%	1,12%	1,28%	1,36%	1,71%	1,38%	1,39%	1,39%	1,31%	1,39%	0,17%
4	1,49%	1,03%	1,49%	1,41%	1,75%	1,08%	1,56%	1,21%	2,46%	1,50%	0,43%
5	3,23%	2,93%	3,06%	2,06%	2,21%	3,38%	2,59%	3,11%	3,08%	2,85%	0,46%
6	3,91%	3,69%	3,42%	2,11%	4,69%	3,02%	2,46%	3,33%	3,98%	3,40%	0,79%
8	2,20%	1,92%	2,67%	2,30%	1,51%	2,28%	2,36%	1,93%	2,04%	2,14%	0,33%

(fonte: elaborado pelo autor)

C.4 QUADROS COMPARATIVOS

Quadro C.7 – Estimativas das frequências naturais da Ponte Z24

Modo	Frequências naturais (Hz)					
	SSI COV		SSI DATA		IV	
	Média	Desvio padrão	Média	Desvio padrão	Média	Desvio padrão
1	3,86	0,009	3,86	0,006	3,86	0,007
2	4,90	0,014	4,91	0,019	4,90	0,015
3	9,75	0,018	9,75	0,018	9,76	0,025
4	10,29	0,050	10,29	0,059	10,29	0,055
5	12,40	0,151	12,42	0,162	12,40	0,177
6	13,29	0,168	13,33	0,126	13,26	0,129
8	19,16	0,165	19,06	0,074	19,04	0,125

(fonte: elaborado pelo autor)

Quadro C.8 – Estimativas das razões de amortecimento da Ponte Z24

Modo	Razões de amortecimento					
	SSI COV		SSI DATA		IV	
	Média	Desvio padrão	Média	Desvio padrão	Média	Desvio padrão
1	0,84%	0,17%	0,95%	0,16%	0,88%	0,14%
2	1,26%	0,17%	1,24%	0,19%	1,42%	0,13%
3	1,36%	0,15%	1,34%	0,18%	1,39%	0,17%
4	1,46%	0,41%	1,58%	0,35%	1,50%	0,43%
5	2,90%	0,46%	2,86%	0,37%	2,85%	0,46%
6	4,14%	1,20%	3,67%	1,02%	3,40%	0,79%
8	2,19%	0,66%	2,28%	0,49%	2,14%	0,33%

(fonte: elaborado pelo autor)