

DESENVOLVIMENTO E IMPLEMENTAÇÃO DO MÉTODO ERA NA IDENTIFICAÇÃO DE PARÂMETROS MODAIS USANDO A PLATAFORMA PYTHON

Daniel Awada Elarrat Canto, daniel.awadaec@gmail.com¹
Heraldo Nélio Cambraia, heraldo@ufpr.br¹

¹Departamento de Engenharia Mecânica – Setor de Tecnologia – UFPR Caixa Postal 19019, Jardim das Américas, CEP 81531-990, Curitiba - PR

Resumo. Este artigo apresenta o desenvolvimento e a implementação computacional do método ERA (Eigensystem Realization Algorithm) na identificação de parâmetros modais usando a plataforma Python. O método ERA é baseado na teoria da realização de sistemas lineares com parâmetros invariantes no tempo, isto é, no ajuste de um modelo na forma de estado em tempo discreto para uma quantidade de dados experimentais obtidos da medição de funções de respostas impulsivas (FRI) de um sistema dinâmico. Os parâmetros modais são obtidos resolvendo um problema de autovalor da matriz de estado do sistema, onde os autovalores fornecem as frequências naturais e os fatores de amortecimento e os autovetores as formas modais associadas. A matriz de estado é determinada da solução de um problema de mínimos quadrados envolvendo de duas submatrizes obtidas da matriz de observabilidade do sistema através de uma propriedade de invariância ao deslocamento, sendo que a estimativa da matriz de observabilidade, por sua vez, vem da aplicação da decomposição em valores singulares (SVD, singular value decomposition) sobre uma matriz de estrutura Hankel que contém os dados das FRIs. A ordem mínima do sistema é identificada através da inspeção dos valores singulares mais significativos da matriz dos dados usando SVD. A efetividade do presente método é demonstrada usando dados das FRIs obtidos de simulação numérica.

Palavras chave: Análise Modal, Método ERA, Análise de Sinais, Python

Abstract. The article presents the computational implementation of the Eigensystem Realization Algorithm (ERA) to be used in modal parameters identification via a Python platform. ERA is based on the realization theory consisting of fitting a discrete and invariant time first-order state-space mathematical model to a quantity of measured impulsive responses functions (IRF) taken from a structural system. Numerically, the identification scheme relies on eigenvalue and singular value decomposition (SVD) problems of appropriated matrices, such as: state-matrix of the invariant and discrete-time system and block-Hankel structured matrix of IRFs data. The effectiveness of present method is demonstrated using samples of IRF's obtained from numerical simulation.

Keywords: Modal Analysis, ERA Method, Signal Analysis, Python

1. INTRODUÇÃO

A identificação de parâmetros de sistemas lineares invariantes no tempo é uma área importante na ciência e na engenharia (Söderström e Stoica, 1989, Ljung, 1987 e Van Der Veen et al., 1993). A análise modal experimental (AME) trata com o processo da estimativa dos parâmetros modais de sistemas mecânicos (ou sistemas estruturais) – frequências naturais, fatores de amortecimento e formas modais - usando medições dos sinais de entrada e saída ou usando apenas os sinais de saída (Maia e Silva, 1997). A AME envolve uma vasta quantidade de diferentes aplicações tais como: testes modais, ajuste de modelos matemáticos por elementos finitos, *updating* modal, sub-estruturação modal, modificação modal, saúde estrutural, detecção de falhas e análise modal não linear. O objetivo deste trabalho consiste na aplicação do método ERA (Eigensystem Realization Algorithm) em AME implementado em uma plataforma Python (Borges, 2014). Segue uma descrição mais detalhada sobre o conteúdo do presente estudo.

O método ERA foi desenvolvido por Juang e Pappa, (1985) baseado em conceitos predominantemente originados da teoria de controle – mais especificamente da teoria da realização de sistemas lineares com parâmetros invariantes no tempo (Viberg, 1995). Apesar do ERA ter sido desenvolvido na década de 80, ele conta com várias aplicações e pesquisas atuais graças a sua grande precisão e robustez como destacam, por exemplo, os trabalhos de Kordkheili et al., (2018), e Hu et al., (2014).

Matematicamente, ERA é um método implementado no domínio do tempo (ou da frequência) e se baseia no ajuste de um modelo linear na forma de estado em tempo discreto para uma quantidade de dados experimentais obtidas de medições de funções de respostas impulsivas (FRI) em alguns pontos do sistema estrutural, sendo que os autovalores da matriz de estado do sistema (ou polos do sistema) fornecem as frequências naturais e fatores de amortecimento e os autovetores as formas modais associadas. É importante observar que o uso de dados experimentais do tipo FRIs confere ao método extrema versatilidade na realização dos testes modais, uma vez que o uso do martelo na excitação da estrutura pode trazer muito mais facilidades quando comparado com algumas montagens mais complexas exigidas pelo uso de excitadores.

A matriz de estado do sistema é determinada através da solução de um problema de mínimos quadrados envolvendo duas submatrizes obtidas da matriz de observabilidade do sistema, usando uma importante propriedade de invariância ao deslocamento. Na prática, entretanto, a estimativa da matriz de observabilidade é obtida através de um procedimento de redução de posto de uma matriz de estrutura Hankel-bloco, formada por amostras das FRIs, usando decomposição em valores singulares (SVD), (Van Der Veen et al., 1993). Além disso, a ordem mínima do sistema é identificada pela inspeção dos valores singulares mais significativos desta matriz Hankel-bloco.

Os algoritmos desenvolvidos neste trabalho são implementados no domínio do tempo e na linguagem Python graças a sua grande eficiência e sua atual relevância na área de computação numérica. Por ser uma linguagem de fácil entendimento e ter cada vez mais usuários, o código apresentado nesse trabalho poderá ser usado como material de estudo e incentivar novos estudantes da área sobre o assunto.

O presente estudo é organizado da seguinte forma. A seção 2 apresenta a modelagem matemática de sistemas mecânicos lineares com parâmetros invariantes no tempo, a seção 3 apresenta noções básicas sobre a teoria de realização visando à determinação da matriz de observabilidade do sistema, a seção 4 trata da estimativa da matriz de observabilidade estendida usando SVD, a seção 5 apresenta exemplos de aplicação cujos dados das FRIs são obtidos por simulação numérica e a seção 6 apresenta a conclusão do trabalho. É apresentada também a listagem do código em Python usado nos exemplos de aplicação do presente trabalho.

2. MODELAGEM MATEMÁTICA DE SISTEMAS MECÂNICOS

Esta seção apresenta conceitos básicos sobre modelagem matemática de sistemas mecânicos lineares e invariantes no tempo (LIT). Este tipo de modelagem é usado por dois motivos: i) para incluir o amortecimento do tipo viscoso não proporcional e principalmente, ii) porque o desenvolvimento analítico do método ERA requer este tipo de modelagem no processo de identificação do sistema dinâmico.

2.1. Modelagem de sistemas na forma de estado para tempo contínuo

A equação de movimento de um sistema mecânico LIT de f graus de liberdade é representada pela seguinte equação diferencial matricial de segunda ordem em tempo contínuo,

$$\bar{\mathbf{M}}\ddot{\mathbf{q}}(t) + \bar{\mathbf{C}}\dot{\mathbf{q}}(t) + \bar{\mathbf{K}}\mathbf{q}(t) = \mathbf{f}(t) \quad (1)$$

onde, $\bar{\mathbf{M}}$, $\bar{\mathbf{C}}$, $\bar{\mathbf{K}}$ são, respectivamente, as matrizes de massa, amortecimento e rigidez, todas de dimensão $f \times f$. Os vetores $\mathbf{q}(t)$ e $\mathbf{f}(t)$, ambos de dimensão $f \times 1$, representam, respectivamente, as coordenadas generalizadas em termos de posição de pontos do sistema e as forças externas que atuam no sistema. A Eq. (1) pode ser convertida em um modelo de primeira ordem na forma de estado para tempo contínuo como,

$$\dot{\mathbf{x}}(t) = \bar{\mathbf{A}}\mathbf{x}(t) + \bar{\mathbf{B}}\mathbf{u}(t) \quad (2)$$

onde $\mathbf{x}(t) = \begin{Bmatrix} \mathbf{q}(t) \\ \dot{\mathbf{q}}(t) \end{Bmatrix}$ é o vetor de estado do sistema, de dimensão $n \times 1$, sendo $n = 2f$ a ordem mínima do modelo e

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_f \\ -\bar{\mathbf{M}}^{-1}\bar{\mathbf{K}} & -\bar{\mathbf{M}}^{-1}\bar{\mathbf{C}} \end{bmatrix} \quad \text{e} \quad \bar{\mathbf{B}} = \begin{bmatrix} \mathbf{0} \\ \bar{\mathbf{M}}^{-1}\mathbf{U}_f \end{bmatrix} \quad (3)$$

são matrizes de dimensões, respectivamente, iguais a $n \times n$ e $n \times m$, onde a chamada matriz de estado $\bar{\mathbf{A}}$ conecta a variável de estado $\mathbf{x}(t)$ com sua derivada temporal $\dot{\mathbf{x}}(t)$. O vetor $\mathbf{u}(t)$ de dimensão $m \times 1$ representa as m entradas não nulas do vetor $\mathbf{f}(t)$. A matriz \mathbf{U}_f , de dimensão $f \times m$, é a matriz de seleção de entrada onde, $\mathbf{f}(t) = \mathbf{U}_f \mathbf{u}(t)$, \mathbf{I}_f é a matriz de identidade e $\mathbf{0}$ denota matriz nula de dimensões apropriadas.

Seja $\mathbf{y}(t)$ um vetor de dimensão $l \times 1$ usado para descrever as l variáveis observadas do sistema definido como,

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (4)$$

onde, \mathbf{C} e \mathbf{D} são matrizes de dimensões, respectivamente, iguais a $l \times n$ e $l \times m$.

O problema de autovalor da matriz $\bar{\mathbf{A}}$ pode ser descrito como,

$$\bar{\mathbf{A}} = \boldsymbol{\Theta} \boldsymbol{\Lambda} \boldsymbol{\Theta}^{-1} \quad (5)$$

onde $\boldsymbol{\Lambda}$ e $\boldsymbol{\Theta}$ são, respectivamente, as matrizes dos autovalores (diagonal) e dos autovetores de $\bar{\mathbf{A}}$, ambas de dimensão $n \times n$. Os n autovalores de $\bar{\mathbf{A}}$, denotados por λ_r , fornecem os polos do sistema modelado em tempo contínuo. Para o caso de sistemas subamortecidos e com polos distintos eles aparecem em pares complexos conjugados como,

$$\lambda_r = -\xi_r \omega_r \mp i \omega_r \sqrt{1 - \xi_r^2} \quad (6)$$

onde $i = \sqrt{-1}$, ω_r são as frequências naturais e ξ_r são fatores de amortecimento que podem ser calculados como,

$$\omega_r = \sqrt{(Re(\lambda_r))^2 + (Im(\lambda_r))^2} \quad \text{e} \quad \xi_r = -\frac{Re(\lambda_r)}{\omega_r} \quad (7)$$

Os n autovetores de $\bar{\mathbf{A}}$, denotados por $\boldsymbol{\theta}_r$, de dimensão $n \times 1$, também aparecem em pares complexos conjugados e podem ser organizados como,

$$\boldsymbol{\theta}_r = \begin{Bmatrix} \boldsymbol{\phi}_r \\ \lambda_r \boldsymbol{\phi}_r \end{Bmatrix} \quad \text{com} \quad \boldsymbol{\phi}_r = \begin{Bmatrix} \phi_{1r} \\ \vdots \\ \phi_{fr} \end{Bmatrix} \quad (8)$$

onde $\boldsymbol{\phi}_r$ representa a r -ésima forma modal do sistema descrito pelas Eqs. (1) e (2), com $r = 1, \dots, n$.

2.2. Modelagem de sistemas na forma de estado para tempo discreto

A Eq. (2) constitui o modelo matemático na forma de estado para tempo contínuo. Entretanto, os dados experimentais de entrada e saída usados nos testes modais são obtidos na forma discreta em intervalos de tempo constantes Δt . A descrição do sistema mecânico na forma de estado para tempo discreto k , sendo $t = k\Delta t$, fica como,

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \quad (9)$$

onde a matriz de estado do sistema \mathbf{A} e a matriz \mathbf{B} são relacionadas com $\bar{\mathbf{A}}$ e $\bar{\mathbf{B}}$ como (Luenberger, 1979),

$$\mathbf{A} = e^{\bar{\mathbf{A}}\Delta t} \quad \text{e} \quad \mathbf{B} = \int_0^{\Delta t} e^{\bar{\mathbf{A}}\bar{t}} d\bar{t} \bar{\mathbf{B}} \quad (10)$$

Além disso, com base na Eq. (4), o vetor das variáveis observadas do sistema para tempo discreto é definido como,

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) \quad (11)$$

O problema de autovalor da matriz \mathbf{A} pode ser expressa como $\mathbf{A} = \boldsymbol{\Psi} \mathbf{Z} \boldsymbol{\Psi}^{-1}$. Os n autovalores de \mathbf{A} , denotados por z_r , fornecem os polos do sistema modelado em tempo discreto cuja relação com os polos para tempo contínuo é a seguinte,

$$\lambda_j = \log_e(z_j)/\Delta t \quad (12)$$

Entretanto, as formas modais $\boldsymbol{\phi}_r$ obtidas dos autovetores das matrizes de estado dos modelos tanto para tempo contínuo como para tempo discreto são iguais, pois trata se do mesmo sistema dinâmico.

3. REALIZAÇÃO DE SISTEMAS LINEARES INVARIANTES NO TEMPO

A teoria da realização (Viberg, 1995) de sistemas lineares com parâmetros invariantes no tempo consiste no ajuste de um modelo matemático de ordem mínima na forma de estado a partir de um conjunto de dados experimentais obtidos das FRIs. Esta seção apresenta uma importante estrutura de invariância ao deslocamento da matriz de observabilidade do sistema com relação matriz de estado \mathbf{A} , através da qual permite se o cálculo das matrizes \mathbf{A} e \mathbf{C} que serão usadas na implementação computacional do método ERA para AME.

3.1. Matrizes de Observabilidade e Controlabilidade. Invariância ao Deslocamento

Definindo a matriz \mathbf{H} de estrutura Hankel-bloco com i blocos de linhas e j colunas, de dimensão $il \times j$, formadas por dados discretos de FRIs $\mathbf{h}_i(k) = \{h_{1i}(k) \dots h_{li}(k)\}^T$, de dimensão $l \times 1$, obtém-se o seguinte (Juang e Pappa, 1985),

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_i(1) & \mathbf{h}_i(2) & \dots & \mathbf{h}_i(j) \\ \mathbf{h}_i(2) & \mathbf{h}_i(3) & \dots & \mathbf{h}_i(j+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}_i(i) & \mathbf{h}_i(i+1) & \dots & \mathbf{h}_i(i+j-1) \end{bmatrix} = \mathbf{\Gamma}_i \mathbf{\Omega}_j \quad (13)$$

onde

$$\mathbf{\Gamma}_i = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{i-1} \end{bmatrix} \quad (14)$$

é a matriz de observabilidade do sistema de dimensão $il \times n$, e $\mathbf{\Omega}_j = [\mathbf{b}_i \quad \mathbf{Ab}_i \quad \mathbf{A}^2\mathbf{b}_i \quad \dots \quad \mathbf{A}^{j-1}\mathbf{b}_i]$ é a matriz de controlabilidade do sistema de dimensão $n \times j$.

Da Eq.(14) observa-se uma estrutura de invariância ao deslocamento da matriz de observabilidade $\mathbf{\Gamma}_i$ como,

$$\mathbf{\Gamma}_i^{(1)} \mathbf{A} = \mathbf{\Gamma}_i^{(2)} \quad (15)$$

onde as sub-matrizes $\mathbf{\Gamma}_i^{(1)}$ e $\mathbf{\Gamma}_i^{(2)}$ são definidas como,

$$\mathbf{\Gamma}_i^{(1)} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{i-2} \end{bmatrix} \quad \text{e} \quad \mathbf{\Gamma}_i^{(2)} = \begin{bmatrix} \mathbf{CA} \\ \mathbf{CA}^2 \\ \mathbf{CA}^3 \\ \vdots \\ \mathbf{CA}^{i-1} \end{bmatrix} \quad (16)$$

Assumindo que a realização do sistema seja de ordem mínima n , o que leva a matriz de observabilidade $\mathbf{\Gamma}_i$ possuir posto completo igual a n , resulta,

$$\mathbf{A} = (\mathbf{\Gamma}_i^{(1)})^+ \mathbf{\Gamma}_i^{(2)} \quad (17)$$

onde o símbolo "+" denota pseudo-inversa de Moore-Penrose de $\mathbf{\Gamma}_i^{(1)}$, isto é, $(\mathbf{\Gamma}_i^{(1)})^+ = ((\mathbf{\Gamma}_i^{(1)})^T \mathbf{\Gamma}_i^{(1)})^{-1} (\mathbf{\Gamma}_i^{(1)})^T$.

A matriz \mathbf{C} pode ser obtida diretamente do primeiro bloco de $\mathbf{\Gamma}_i$ como mostra a Eq. (14).

As formas modais relacionadas com a parte observável do sistema $\mathbf{\Phi}_{obs}$ são trivialmente calculadas como,

$$\mathbf{\Phi}_{obs} = \mathbf{C}\mathbf{\Psi} \quad (18)$$

Uma importante observação é que este método pode ser generalizado para o caso multivariável, onde são consideradas p entradas impulsivas atuando não simultaneamente em q pontos distintos do sistema estrutural (Cabraia e Kurka, 2011).

4. MATRIZ DE OBSERVABILIDADE EXTENDIDA E O ALGORITMO ERA

A seção anterior mostrou o importante papel da matriz de observabilidade $\mathbf{\Gamma}_i$ na estimativa das matrizes \mathbf{A} e \mathbf{C} do sistema dinâmico. Esta seção apresenta a determinação de uma versão da matriz $\mathbf{\Gamma}_i$ a partir de dados experimentais das FRIs, através de uma técnica de redução de posto da matriz \mathbf{H} , usando SVD. Além disso, é mostrada a identificação da ordem mínima n do sistema através da simples inspeção dos valores singulares mais significativos de \mathbf{H} .

4.1. Versão extendida matriz de observabilidade usando SVD

Com base na Eq. (13) e considerando $n < il$ e $n < j$, o posto de ambas as matrizes $\mathbf{\Gamma}_i$ e $\mathbf{\Omega}_j$ é obviamente igual a n , igual a ordem mínima do sistema. Isso implica que o produto $\mathbf{H} = \mathbf{\Gamma}_i \mathbf{\Omega}_j$, de dimensão $il \times j$ também possui posto deficiente igual a n . Além disso, as n colunas da matriz $\mathbf{\Gamma}_i$ geram o espaço coluna de \mathbf{H} e as n linhas de $\mathbf{\Omega}_j$ geram o espaço linha de \mathbf{H} (Van Der Veen et al., 1993). Então $\mathbf{\Gamma}_i$ pode ser obtida pela base que gera o espaço coluna de \mathbf{H} .

Na prática, considerando amostras experimentais das FRIs, a matriz \mathbf{H} possui posto completo igual a $\min(il, j)$. Seja, portanto, um sistema dinâmico de ordem mínima igual a n , sendo $n < j < il$, a aplicação da SVD sobre matriz \mathbf{H} resulta no seguinte (Van Der Veen et al., 1993),

$$\mathbf{H} = \mathbf{Q}\mathbf{S}\mathbf{V}^T = [\hat{\mathbf{Q}}_s \quad \hat{\mathbf{Q}}_r] \begin{bmatrix} \hat{\mathbf{S}}_s & \mathbf{0}_{j-n} \\ \mathbf{0}_{j-n} & \hat{\mathbf{S}}_r \\ \mathbf{0}_{il-j \times j} & \end{bmatrix} \begin{bmatrix} \hat{\mathbf{V}}_s^T \\ \hat{\mathbf{V}}_r^T \end{bmatrix} = \hat{\mathbf{Q}}_s \hat{\mathbf{S}}_s \hat{\mathbf{V}}_s^T + \hat{\mathbf{Q}}_r \hat{\mathbf{S}}_r \hat{\mathbf{V}}_r^T \quad (19)$$

onde \mathbf{Q} , \mathbf{S} e \mathbf{V} são matrizes de dimensões, respectivamente, iguais a $il \times il$, $il \times j$ e $j \times j$. A matriz diagonal $\hat{\mathbf{S}}_s$ possui dimensão $n \times n$ e contém os valores singulares de \mathbf{H} relacionados com a dinâmica do sistema. A matriz diagonal $\hat{\mathbf{S}}_r$ possui dimensão $(j - n) \times (j - n)$ e contém os valores singulares de \mathbf{H} não relacionados com a dinâmica do sistema – modos espúrios ou ruído presente nas medições das FRIs. Os valores singulares mais significativos de \mathbf{H} são, portanto, aqueles agrupados na matriz $\hat{\mathbf{S}}_s$.

As matrizes $\hat{\mathbf{Q}}_s$ e $\hat{\mathbf{V}}_s$ da Eq. (19) são ortogonais com dimensões, respectivamente, iguais a $il \times n$ e $j \times n$. Além disso, as n colunas das matrizes $\hat{\mathbf{Q}}_s$ e $\hat{\mathbf{V}}_s$ geram, respectivamente, o espaço coluna e o espaço linha da matriz \mathbf{H} . Portanto, uma estimativa da matriz de observabilidade estendida $\hat{\mathbf{F}}_s$ pode ser obtida como,

$$\hat{\mathbf{F}}_s = \hat{\mathbf{Q}}_s \quad \text{ou} \quad \hat{\mathbf{F}}_s = \hat{\mathbf{Q}}_s \hat{\mathbf{S}}_s^{1/2} \quad (20)$$

4.2. Algoritmo para identificação de parâmetros modais usando o ERA

- i) partindo de valores de j que resulta em posto sobredeterminado de \mathbf{H} , identificar a ordem mínima n do sistema pela inspeção do número dos valores singulares mais significativos de \mathbf{H} agrupados em $\hat{\mathbf{S}}_s$ - Eq. (19)
- ii) estimativa da matriz de observabilidade estendida $\hat{\mathbf{F}}_s$ usando SVD - Eqs. (19) e (20),
- iii) estimativa da matriz de estado \mathbf{A} e dos autovalores z_i e autovetores Ψ_i de \mathbf{A} , para $i = 1, 2, \dots, n$ - Eq. (17),
- iv) cálculo dos n polos $\lambda_j = \log_e(z_j)/\Delta t$, n frequências naturais ω_r e n fatores de amortecimento ξ_r - Eq. (7), e
- v) determinação da matriz \mathbf{C} obtida do primeiro bloco de $\hat{\mathbf{F}}_s$ e das formas modais $\phi_i = \mathbf{C}\Psi_i$ conforme Eq. (18).

5. EXEMPLO DE APLICAÇÃO

Esta seção apresenta dois testes de aplicação do método ERA em AME para dados das FRIs obtidos numericamente de um sistema massa-mola-amortecedor com amortecimento proporcional de cinco graus como mostrado na Fig. 1. Os valores adotados para as massas dos blocos, rigidezes das molas e amortecimento são: $m_1 = m_2 = m_3 = m_4 = m_5 = 0,5 \text{ kg}$, $k_1 = k_2 = k_3 = k_4 = k_5 = 2000 \text{ N/m}$ e $b_1 = b_2 = b_3 = b_4 = b_5 = 20 \text{ Ns/m}$.

Uma quantidade de N_p amostras das FRIs para tempo discreto são calculadas como (Maia e Silva, 1997),

$$h_{ij}(k) = 2\text{Real}\left\{\sum_{l=1}^f \phi_{il} \phi_{jl} e^{\lambda_l k \Delta t}\right\} \quad k = 1, \dots, N_p \quad (21)$$

onde ϕ_{ij} s são termos da forma modal $\boldsymbol{\phi}_r$ dada pela Eq. (9) e o termo $r_{ij} = \phi_{il} \phi_{jl}$ da equação acima é chamado resíduo.

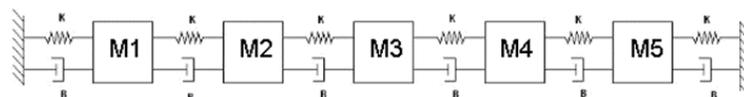


Figura 1. Sistema mecânico de cinco graus de liberdade

5.1. Teste 1 – Dados das FRIs sem ruído

Para uma quantidade de $N_p=500$ pontos e um intervalo de discretização $\Delta t=1/40$ segundo são calculadas as seguintes FRIs $h_{11}(k)$, $h_{21}(k)$, $h_{31}(k)$, $h_{41}(k)$ e $h_{51}(k)$ - excitação impulsiva no bloco 1 e medidas simultâneas nos cinco blocos.

Partindo de um valor sobredeterminado $j=40$ para uma quantidade de blocos $i = N_p - j + 1 = 461$, a matriz dos dados \mathbf{H} , segundo a Eq. (13), possui dimensão igual a 2305×40 .

A ordem mínima do sistema é identificada pela inspeção dos valores singulares mais significativos da matriz \mathbf{H} com base na Fig. 2, mostrando que a ordem mínima do sistema é identificada como $n=10$.

Usando o valor da ordem mínima $n=10$ as frequências naturais, os fatores de amortecimento identificados bem como as formas modais identificadas são exatamente iguais às obtidas da simulação numérica, como pode ser comprovado rodando o código apresentado no apêndice do trabalho.

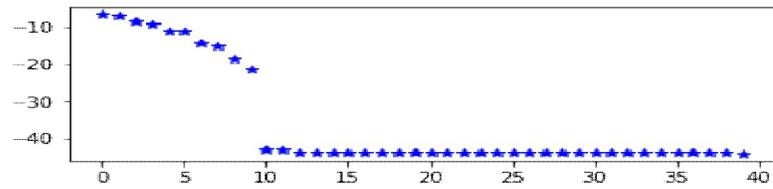


Figura 2. Logaritmo natural dos valores singulares de \mathbf{H}

5.1. Teste 2 - Dados das FRIs com ruído

A fim de obter uma situação mais realista, neste teste é somado a cada FRI um sinal aleatório Gaussiano de média zero e amplitude igual a 10^{-8} . Adotando o mesmo procedimento do teste 1, a Fig. 3 mostra o logaritmo natural dos valores singulares, de modo que a ordem mínima é identificada como $n=10$. Devido a presença do ruído apenas quatro modos do sistema foram identificados. Os parâmetros modais identificados são comparados com seus valores exatos e apresentados na Tab. 1 e as partes real e imaginária das quatro formas modais identificadas são mostradas na Fig. 4. O erro é calculado como

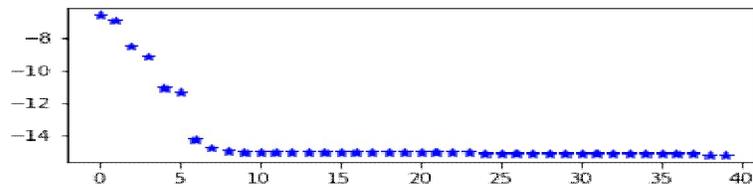


Figura 3. Logaritmo natural dos valores singulares de \mathbf{H}

Tabela 1. Comparação dos parâmetros exatos do sistema mecânico com os identificados pelo ERA.

r	Exata (Hz)	ERA (Hz)	Erro (%)	Exato (%)	ERA (%)	Erro (%)
1	5.2105	5.2105	0	0.1637	0.1637	0
2	10.0658	10.0647	0.0109	0.3162	0.3162	0
3	14.2353	14.2205	0.1040	0.4472	0.4470	0.0447
4	17.4346	-	-	0.5477	-	-
5	19.4457	19.4684	-0.1167	0.6109	0.5589	8.5120

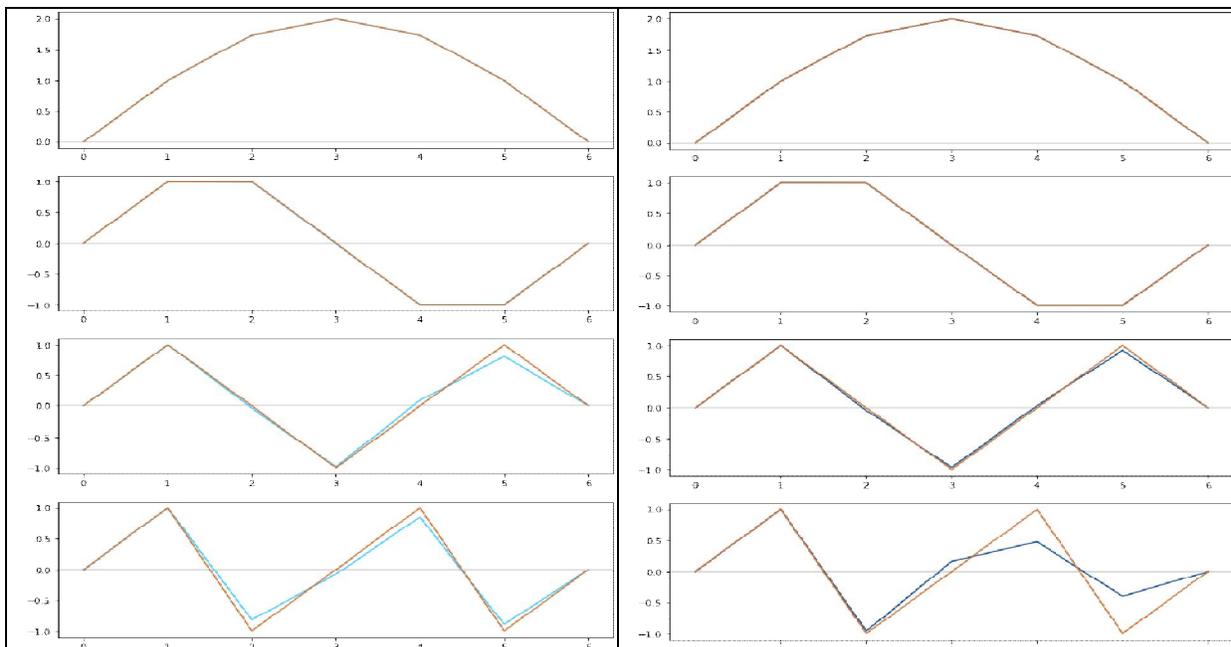


Figura 4. Partes real e imaginária das formas modais exatas (laranja) e identificadas (azul)

6. CONCLUSÃO

Este trabalho apresenta a aplicação do método ERA na identificação paramétrica em AME usando a plataforma Python para dados do tipo FRIs. Os dados das FRIs usados neste estudo são obtidos por simulação numérica de um sistema massa-mola-amortecedor de cinco graus de liberdade com amortecimento proporcional. Os resultados comprovam a excelente efetividade do código na identificação paramétrica. É importante observar que o algoritmo apresentado possui a mesma excelente precisão para identificação de frequências naturais e fatores de amortecimento para sistemas mecânicos com amortecimento não proporcional, porém ele não é o mais apropriado para o cálculo das formas modais. Nesse caso, sugere-se o uso da matriz de Vandermonde para o cálculo dos resíduos e a partir destes realizar o cálculo das formas modais como pode ser consultado, por exemplo, em Van Der Veen et al., (1993). O ruído tende a diminuir muito a precisão dos resultados, especialmente para as formas modais. Finalmente, o trabalho traz a listagem do código usado nos exemplos aqui apresentados, que poderá ser usado como material de estudo e incentivar novos estudantes da área sobre o assunto.

7. REFERÊNCIAS

- Borges, L.E., 2014, *Python para Desenvolvedores*, Novatec.
- Cambráia, H.N. e Kurka, P.R.G., 2011. "Time Domain Multivariable Subspace-Based Method for Modal Parameters Identification Using Impulse Excitation". In *Proceedings of the International Symposium on Dynamic Problems of Mechanics – DINAME 2011*. São Sebastião, Brazil.
- Hu, S.J., Yang, W.L., Liu, F.S. and Li, H.J., 2014. "Fundamental Comparison of Time-Domain Experimental Modal Analysis Methods Based on High and First Order Matrix Models", *Journal of Sound and Vibration*, Vol. 333, p. 6869-6884.
- Juang, J.N. and Pappa, R.S., 1985. "An Eigensystem Realization Algorithm for Modal Parameters Identification", *Journal of Guidance, Control and Dynamics*, Vol. 8(5), p. 620-627.
- Kordkheili, S.A., Massouleh, S.H, Hajirezayi, S. and Bahai, 2018. "Experimental Identification of Closely Spaced Modes Using NExT-ERA", *Journal of Sound and Vibration*, Vol. 412, p. 116-129.
- Ljung, L., 1999, *System Identification: Theory for the User*, Prentice Hall, 2ª edição.
- Luenberger, D.G., 1979, *Introduction to Dynamic System: Theory, Model and Applications*, John Wiley & Sons.
- Maia, N.M.M. and Silva, J.M.M., 1997, *Theoretical and Experimental Modal Analysis*, Research Studies Press.
- Söderström, T. and Stoica, P., 1989, *System Identification*, Prentice Hall.
- Viberg, M., 1995, "Subspace-Based Methods for Identification of Linear Time-Invariant System", *Automatica*, Vol. 31(12), p. 1835-1851.
- Van Der Veen, A.J., Deprettere, E.F. and Swindlehurst, 1993, "Subspace Based Signal Analysis Using Singular Value Decomposition", *Proceeding of the IEEE*, Vol. 81(9), p. 1277-1308.

8. RESPONSABILIDADE PELAS INFORMAÇÕES

Os autores são os únicos responsáveis pelas informações incluídas neste trabalho.

9. APÊNDICE – CÓDIGO PYTHON

Aqui apresenta o código usado na simulação das FRIs e identificação dos parâmetros modais usando o método ERA.

```
import numpy as np
import numpy.linalg as LA
import matplotlib.pyplot as plt

np.set_printoptions(precision=4, suppress=True)
#GERAÇÃO DE DADOS
print("\033[4mGERAÇÃO DE DADOS\033[0m")
m = [0.5, 0.5, 0.5, 0.5, 0.5]
M = np.diag(m)
k = np.array([2000, 2000, 2000, 2000, 2000, 2000])

K = np.array([[k[0]+k[1], -k[1], 0, 0, 0],
              [-k[1], k[1]+k[2], -k[2], 0, 0],
              [0, -k[2], k[2]+k[3], -k[3], 0],
              [0, 0, -k[3], k[3]+k[4], -k[4]],
              [0, 0, 0, -k[4], k[4]+k[5]]])
C = K*0.01
ZR = np.zeros((5,5))
I = np.eye(5)
```

```
Minv = np.linalg.inv(M)
A = np.block([[ZR      , I      ],
              [-Minv@K, -Minv@C]])
LAMBDA, FI = np.linalg.eig(A)
lam = LAMBDA[10:0:-2].T
fi = FI[0:5, 10:0:-2].T
wn_ST = np.abs(lam)
qsi = -np.real(lam) / np.abs(lam)
fnl = np.abs(lam)/(2*np.pi)
sort = np.argsort(fnl)
print("\nFrequências exatas (em Hz):\n", fnl)
print("\nFatores de amortecimento exatos:\n", qsi)
fi_norm = np.array([fi.imag[i]/fi.imag[i,0] for i in range(5)])
modo_exato = np.block([np.zeros((5,1)), fi_norm, np.zeros((5,1))])
print("\nParte imaginária das formas modais:")
plt.figure(figsize=(6,10))
modos = [[plt.subplot(511+i), plt.axhline(color='darkgray'),\
          plt.plot(modo_exato[sort[i]])] for i in range(5)]
plt.show()
fi_norm_2 = np.array([fi.real[i]/fi.real[i,0] for i in range(5)])
modo_exato_2 = np.block([np.zeros((5,1)), fi_norm_2, np.zeros((5,1))])
print("\nParte real das formas modais:")
plt.figure(figsize=(6,10))
modos = [[plt.subplot(511+i), plt.axhline(color='darkgray'),\
          plt.plot(modo_exato_2[sort[i]])] for i in range(5)]
plt.show()
amp_ruido = float(input("Dê a amplitude do ruído: (separe os decimais com ponto (.)\n"))
dt = 1/(2*20)
Np = 500
t = np.array(np.arange(0, Np, 1)*dt

r = np.array([[fi[i, j] * fi[i, 0] for i in range(5)] for j in range(5)])
h = np.array([[2 * np.real(r[j] @ np.exp(lam*t[i])) + np.random.uniform(-amp_ruido, amp_ruido) for i in range(Np)] for j in range(5)]) #Resíduos exatos

#IDENTIFICAÇÃO
print("\n\033[4mIDENTIFICAÇÃO\033[0m")
No = 5
j = 40
i = Np - j + 1
H = np.block([[h[:, k+1] for k in range(i)] for l in range(j)].T)
u, s, v = LA.svd(H)
print("\nLogaritmo dos Valores Singulares:")
plt.figure(figsize=(6,2))
plt.plot(np.log(s), 'b*')
plt.show()
order = int(input("\nDê a ordem mínima do sistema:\n"))
u1 = u[0:No*(i-1), 0:order]
u2 = u[No:No*i, 0:order]
F = LA.pinv(u1) @ u2
Z, fi_ident = LA.eig(F)
lamb_ident = np.log(Z) / dt
fn_ident = np.abs(lamb_ident[0:order:2]) / (2*np.pi)
damp_factor = -np.real(lamb_ident[0:order:2]) / np.abs(lamb_ident[0:order:2])
print("\nFrequências identificadas (em Hz):\n", fn_ident)
print("\nFatores de amortecimentos identificados:\n", damp_factor)
MtC = u[0:No, 0:order] #Matriz de Observabilidade C
fi4SID = MtC @ fi_ident
print("\nAutovetores da matriz de estado para tempo discreto:\n", fi4SID)
```