

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
CENTRO MULTIDISCIPLINAR UFRJ-MACAÉ
CURSO DE ENGENHARIA MECÂNICA

LARA ARAUJO EVARISTO CARLOS

**MODELAGEM DE MOLA HELICOIDAL DE SMA E ANÁLISE DE SEU
COMPORTAMENTO EM UM SISTEMA DINÂMICO**

Macaé

2022

LARA ARAUJO EVARISTO CARLOS

MODELAGEM DE MOLA HELICOIDAL DE SMA E ANÁLISE DE SEU
COMPORTAMENTO EM UM SISTEMA DINÂMICO

Trabalho de Conclusão de Curso de graduação submetida ao Instituto Politécnico do CM Universidade Federal do Rio de Janeiro – Campus Macaé como parte dos requisitos necessários à obtenção do grau de bacharel em Engenharia Mecânica.

Orientador:

Prof. Dr. Lucas Lisbôa Vignoli

Macaé

2022

CIP - Catalogação na Publicação

C284

Carlos, Lara Araújo Evaristo

Modelagem de mola helicoidal de SMA e análise de seu comportamento em um sistema dinâmico / Lara Araújo Evaristo Carlos - Macaé, 2022.

116 f.

Orientador(a): Lucas Lisboa Vignoli.

Trabalho de conclusão de curso (graduação) - Universidade Federal do Rio de Janeiro, Instituto Politécnico, Bacharel em Engenharia Mecânica, 2022.

1. Ligas com Memória de Forma. 2. Método de Elementos Finitos.
3. Engenharia Mecânica. I. Vignoli, Lucas Lisboa, orient. II. Título.

CDD 621

LARA ARAUJO EVARISTO CARLOS

MODELAGEM DE MOLLA HELICOIDAL DE SMA E ANÁLISE DE SEU
COMPORTAMENTO EM UM SISTEMA DINÂMICO

Trabalho de Conclusão de Curso de graduação submetida ao Instituto Politécnico do CM Universidade Federal do Rio de Janeiro – Campus Macaé como parte dos requisitos necessários à obtenção do grau de bacharel em Engenharia Mecânica.

Aprovado em Macaé, 15 de Julho de 2022.

BANCA EXAMINADORA:

Prof. Lucas Lisbôa Vignoli, D.Sc. (UFRJ)

Prof. Vinicius de Castro Ecard, M.Sc. (UFRJ)

Prof. Anselmo Leal Carneiro, D.Sc. (UFRJ)

Prof. Arthur Adeodato, D.Sc. (IFF)

Este trabalho é dedicado à minha
família.

AGRADECIMENTOS

Primeiramente, agradeço a Deus por ter me permitido chegar até aqui, por toda a saúde, capacidade e bençãos com que tem me presenteado.

Agradeço aos meus pais e minha irmã que sempre me apoiaram e acreditaram em mim sem medir esforços, dando todo o amor, ensinamentos, motivação e suporte necessários para que as dificuldades fossem vencidas e conseguíssemos alcançar nossos sonhos. Também sou grata pela minha família que, mesmo de longe, sempre torceu por mim e me incentivou.

Agradeço ao meu orientador por toda disponibilidade, dedicação de seu tempo e conhecimento compartilhados comigo para que esse trabalho fosse realizado.

Agradeço aos amigos feitos durante a graduação por toda a parceria, pelas conversas, desabafos e aprendizados. Sem vocês teria sido muito mais difícil concluir esse curso. Também agradeço pelas atividades nos projetos de extensão e empresa júnior que realizamos juntos e tanto me acrescentaram.

Às minhas irmãs do coração, agradeço por sempre serem tão presentes em minha vida, sempre torcendo por mim e sendo como casa.

Agradeço aos meus professores que durante esse tempo me transmitiram tanto conhecimento, além de proporem desafios e motivação para que pudéssemos nos tornar melhores profissionais.

Aos colegas de trabalho nas duas empresas que fiz meus estágios, agradeço pela oportunidade, por todo o aprendizado e desenvolvimento que me ajudaram a alcançar, acreditando no meu potencial e me incentivando para que me tornasse uma melhor profissional e pessoa.

Agradeço aos professores da banca por terem aceitado o convite de participarem desse projeto e por todas as considerações feitas, que com certeza colaboraram para um melhor resultado final.

RESUMO

Avanços científicos e tecnológicos possuem relação com o desenvolvimento de materiais. Nesse contexto, estuda-se sobre materiais inteligentes, que são capazes de alterar suas propriedades de acordo com influências externas de temperatura e carregamento mecânico. Dentre eles, as Ligas com Memória de Formas (*Shape Memory Alloys - SMA*) apresentam complexo comportamento termomecânico que lhes dá a capacidade de retornar a seu estado original se submetidos às condições adequadas de temperatura ou carregamento mecânico. Esse trabalho tem o objetivo de avaliar o comportamento pseudoelástico de uma mola helicoidal de SMA, além de explorar a influência da alteração nos parâmetros geométricos da mola. Procurando entender as vantagens de aplicação do material inteligente, seu comportamento é comparado ao de uma mola elástica. Sendo assim, busca-se: modelar através da simulação computacional uma mola de SMA para obter as curvas de força-deslocamento; validar as simulações com dados experimentais e numéricos; estudar com implementação de um Runge-Kutta de 4ª ordem a aplicação da mola em um sistema dinâmico de massa-mola-amortecedor. As vantagens da aplicação de uma mola de SMA são a dissipação de energia e capacidade de alcançar maiores deslocamentos sobre as mesmas condições que uma mola linear. No geral, são obtidos resultados satisfatórios com a implementação dos métodos numéricos, sendo possível simular o complexo comportamento da mola de SMA.

Palavras-chave: Ligas com Memória de Forma, Mola de SMA, Método de Elementos Finitos, Pseudoelasticidade, Modelo Constitutivo.

ABSTRACT

Scientific and technological advances are related to the development of materials. In this context, smart materials have been studied; they are capable of changing their properties according to external influences of temperatures and mechanical loads. Among them, Shape Memory Alloys (SMA) have a complex thermomechanical behavior that gives them the ability to return to their original shape if subjected to adequate temperature or mechanical loading conditions. This work aims to investigate the pseudoelastic behavior of a SMA helical spring and to explore the influence of changing the geometric parameters of the spring. Seeking to understand the application advantages of the smart material, its behavior is compared to an helical spring's behavior. Thus, it is sought to model through computer simulation a SMA spring to obtain the force-displacement curves, to validate the simulations with experimental and numerical data and to study the application of the spring in a dynamic mass-spring-damper system with the implementation of a 4th order Runge-Kutta. The advantages of applying an SMA spring are energy dissipation and ability to achieve larger displacements under the same conditions as a linear spring. Overall, satisfactory results were obtained with the implementation of the numerical methods and it was possible to simulate the complex behavior of the SMA spring.

Key words: *Shape Memory Alloy, SMA Spring, Finit Element Method Pseudoelasticity, Constitutive Model.*

LISTA DE FIGURAS

Figura 1 - Quantidade de publicações sobre <i>Shape Memory Alloy</i> (Ligas com memória de forma) e <i>Smart Materials</i> (materiais inteligentes) desde 1960 (<i>Scopus</i> , 2022).....	2
Figura 2 - Quantidade de publicações sobre <i>Springs</i> (molas) desde 1950 (<i>Scopus</i> , 2022)	3
Figura 3 - Parâmetros geométricos de uma mola helicoidal (D_o , D_i , d , N e L_f) (Norton, 2013).....	7
Figura 4 - Parâmetros geométricos mola da helicoidal (D , d , α , l) (An, S., 2012, Adaptado)	8
Figura 5 - Forças e torques nas espiras da mola helicoidal (Norton, 2013)	8
Figura 6 - Gráfico tensão-deformação (Callister, 2007, Adaptado)	10
Figura 7 – Tensões de transformação de fase – Pseudoelasticidade (Júnior, 2012).....	11
Figura 8 - Transformação de fase – Efeito de Memória de Forma (Júnior, 2012).....	12
Figura 9 - Comportamento do (a) Fio e (b) Mola de SMA (Garcia, 2015).....	12
Figura 10 - <i>Stents</i> (Bandeira, 2017).....	13
Figura 11 - <i>Chevrons</i> em turbinas de aeronaves (Garcia, 2015)	14
Figura 12 - Diagrama esquemático de um sistema massa-mola-amortecedor	16
Figura 13 - Diagrama de corpo livre sistema massa-mola-amortecedor	16
Figura 14 - Ponte de Tacoma Narrows (Abreu, 2021)	20
Figura 15 - Malha de Elementos Finitos (Borgmann, 2012).....	23
Figura 16 - Pontos para calibração da solução analítica (adaptada de Garcia, 2015)	27
Figura 17 - Esquema de Massa-mola-amortecedor com 1gdl	27
Figura 18 - Parâmetros da Mola (<i>Ansys</i>)	30
Figura 19 - Malha escolhida - 9368 nós	31
Figura 20 - Condições de Contorno (<i>Ansys</i>).....	31
Figura 21 - <i>Steps</i> no <i>Ansys</i> e Definição do deslocamento – 120 mm.....	31
Figura 22 - Detalhes das Configurações da Análise (<i>Ansys</i>).....	32
Figura 23 - Tensão equivalente de von Mises referente a 120 mm de deslocamento máximo	32
Figura 24 - Deslocamento 120 mm: carregamento e descarregamento.....	33
Figura 25 – Diagrama Força x tempo referente a 120 mm de deslocamento máximo... 33	
Figura 26 – Comparação entre as malhas menos refinada e mais refinada.....	34
Figura 27 - Estudo de convergência de malha.....	35
Figura 28 - Soluções numérica e experimental - deslocamentos máximos de: 45 mm, 80 mm e 120 mm.....	35
Figura 29 - Calibração do modelo analítico	36
Figura 30 - Calibração com <i>subloops</i> [0 120 40 80 0]	37
Figura 31 - Soluções analítica, experimental e numérica – 45 mm.....	38
Figura 32 - Soluções analítica, experimental e numérica - 80 mm	38
Figura 33 - Soluções analítica, experimental e numérica - 120 mm	38
Figura 34 - Calibração do modelo analítico - Alteração do diâmetro da mola (D).....	39
Figura 35 - Alteração do diâmetro da mola no comportamento da mola - $C = 6$, $C = 9$ e $C = 12$	40
Figura 36 - Coeficientes de rigidez (k_a e k_m) e variação do índice da mola - (D)	41
Figura 37 - Forças de início da transformação para fase martensita por índice da mola 41	
Figura 38 - Forças de fim da transformação completa para martensita por índice da mola	42
Figura 39 - Forças de início da austenita por índice da mola	42
Figura 40 - Soluções Numérica e Analítica do Sistema dinâmico	43

Figura 41 - Solução do sistema para diferentes frequências de forçamento – mola linear	43
Figura 42 - Amplitude de acordo com $\Omega\omega n$	44
Figura 43 - Solução no tempo, Espaço de fase e Força-Deslocamento para $h = 0,07$ (F=40N)	45
Figura 44 - Solução no tempo, Espaço de fase e Força-Deslocamento para $h = 0,03$ (F=40N)	46
Figura 45 - Solução no tempo, Espaço de fase e Força-Deslocamento para $h = 0,005$ (F=40N)	47
Figura 46 - Sobreposição de Solução para Mola de SMA e Mola Linear.....	48
Figura 47 - Força-Deslocamento no sistema dinâmico	48
Figura 48 - Variação de Força-deslocamento com razão de frequências (a) $\Omega\omega n \sim 0,2$, (b) $\Omega\omega n \sim 0,6$, (c) $\Omega\omega n \sim 1,2$ e (d) $\Omega\omega n \sim 2$	49
Figura 49 - Solução SMA, F=40N, (a) $\Omega\omega n \sim 0,2$, (b) $\Omega\omega n \sim 0,6$, (c) $\Omega\omega n \sim 1,2$ e (d) $\Omega\omega n \sim 2$	50
Figura 50 - Amplitudes pela razão das frequências de forçamento e natural.....	51
Figura 51 - Transformação de fase e amplitude pela razão entre frequências (F=30 N)	52

LISTA DE TABELAS

Tabela 1 - Propriedades das ligas NiTi e CuZnAl (Coura, 2016)	11
Tabela 2 - Propriedades da Liga de Memória de Forma (Savi <i>et al.</i> 2015).....	25
Tabela 3 - Parâmetros da mola (Savi <i>et al.</i> 2015)	25
Tabela 4 – Parâmetros do sistema dinâmico	28
Tabela 5 - Parâmetros utilizados nas diferentes malhas e tempo de simulação	34
Tabela 6 - Pontos de transformação de fase – Deslocamento 120 mm	36
Tabela 7 - Pontos de transformação de fase - Alteração do diâmetro da mola (D).....	40
Tabela 8 - Tempo decorrido para o código gerar solução de uma frequência para cada passo	44

LISTA DE SIGLAS

MEF – Método de Elementos Finitos

SMA – *Shape Memory Alloy*

EDO – Equação Diferencial Ordinária

gdl – Grau de liberdade

CAD – *Computer-Aided Design*

CAE – *Computer-Aided engineering*

LISTA DE SIMBOLOS

α	Coeficiente de expansão térmica
E	Módulo de Elasticidade da Austenita (Módulo de Young)
E_s	Módulo de Elasticidade da Martensita
G	Módulo de cisalhamento
ν	Coeficiente de Poisson
M_s	Temperatura de início da transformação martensítica
M_f	Temperatura de fim da transformação martensítica
A_s	Temperatura de início da transformação austenítica
A_f	Temperatura de fim da transformação austenítica
σ^{Ms}	Tensão de início da transformação da austenita em martensita
σ^{Mf}	Tensão da completa transformação da austenita em martensita
σ^{As}	Tensão de início da transformação em austenítica.
σ^{Af}	Tensão de fim da transformação austenítica.
σ_s	Tensão de início da transformação para martensita não-maclada.
σ_f	Tensão da completa transformação para martensita não-maclada.
F	Força
u	Deslocamento
C	Índice da mola
D	Diâmetro da mola
d	Diâmetro do fio
N	Número de Espiras
ε_L	Deformação máxima de transformação
F_R	Força resistiva
F_D	Força dissipativa

k	Coeficiente de rigidez
c	Coeficiente dissipativo
m	Massa
$u(t)$	Equação de movimento
\dot{u}	Primeira derivada no tempo da equação de movimento
\ddot{u}	Segunda derivada no tempo da equação de movimento
u_h	Solução homogênea
u_p	Solução particular
ξ	Coeficiente de amortecimento
ω_n	Frequência natural
Ω	Frequência de forçamento
t	Tempo
φ_p	Ângulo de fase

SUMÁRIO

1	INTRODUÇÃO	1
1.1	CONTEXTUALIZAÇÃO	1
1.2	MOTIVAÇÃO E JUSTIFICATIVA	4
1.3	OBJETIVOS	5
1.3.1	Objetivo Geral	5
1.3.2	Objetivos Específicos	5
1.4	ESTRUTURA DO TRABALHO	6
2	FUNDAMENTAÇÃO TEÓRICA.....	7
2.1	MOLAS	7
2.2	MATERIAIS	9
2.2.1	Propriedades Mecânicas dos materiais	9
2.3	LIGAS COM MEMÓRIA DE FORMA	10
2.3.1	Aplicações	13
2.3.2	Modelo Constitutivo da Brinson	14
2.4	VIBRAÇÕES	16
2.4.1	Sistema massa-mola-amortecedor	16
2.4.2	Variáveis de Estado	19
2.4.3	Ressonância	20
2.4.4	Método Runge-Kutta	21
2.5	SIMULAÇÃO COMPUTACIONAL.....	22
3	MÉTODOS.....	25
4	RESULTADOS E DISCUSSÕES	30
4.1	SIMULAÇÕES COMPUTACIONAIS.....	30
4.2	ESTUDO DE CONVERGÊNCIA DE MALHA	33
4.3	VALIDAÇÃO DA SIMULAÇÃO DO ANSYS.....	35
4.3.1	Calibração do Modelo	36
4.3.2	Resultados da validação.....	37
4.4	INFLUÊNCIA DE ALTERAÇÃO NO DIÂMETRO DA MOLA	39
4.4.1	Diâmetro da mola	39
4.4.2	Forças de Transformação de Fase	41
4.5	MOLA DE SMA APLICADA A UM SISTEMA DINÂMICO	42
4.5.1	Validação da implementação.....	42
4.5.2	Estudo do Passo	44
4.5.3	Comparação entre Molas	47
4.5.4	Estudo Paramétrico.....	48
5	CONCLUSÕES.....	53
6	TRABALHOS FUTUROS.....	54
	REFERÊNCIAS BIBLIOGRÁFICAS.....	56
	ANEXO A	58
	ANEXO B	66

ANEXO C..... 68

1 INTRODUÇÃO

As molas estão presentes nas atividades mais variadas do dia-a-dia; desde as mais simples para utilização em canetas esferográficas, interruptores de luz ou pregadores de roupa, até em aplicações mais complexas como em aparelhos de academia, amortecedores de carros e robôs industriais. Sua principal característica está relacionada ao armazenamento de energia, onde ela tende a retornar ao seu estado original podendo ser usada para impulsionar ou resistir a algum movimento.

Molas tradicionais podem recuperar deformações desde que mantidas no limite elástico, pois quando atingem o regime plástico, não são capazes de retornar completamente ao estado anterior. Por outro lado, as ligas com memória de forma possuem complexo comportamento termomecânico e, se submetidas às condições adequadas, podem recuperar seu estado original mesmo após deformações inelásticas oriundas de transformação de fase.

Tais ligas podem apresentar efeito de memória de forma, com ciclos de carregamentos termomecânicos; ou pseudoelasticidade, também chamado de superelasticidade, onde apenas carregamentos mecânicos são necessários. Ambos são possíveis por conta da transformação ou reorientação de fase na microestrutura.

1.1 CONTEXTUALIZAÇÃO

Os avanços científicos e tecnológicos possuem relação com o desenvolvimento de materiais de forma que certas aplicações só se tornam possíveis por aprimoramentos realizados após estudos sobre materiais. Como exemplo, um estudo abrangendo fibra de carbono possibilitou a fabricação de uma órtese personalizada para utilização na perna, garantindo precisão, confiabilidade enquanto possui como vantagens a leveza e propriedades mecânicas favoráveis para a aplicação (Giracca, 2021).

Nesse contexto, estuda-se sobre materiais inteligentes, que são capazes de alterar suas propriedades de acordo com influências externas. Também são chamados de materiais multifuncionais ou adaptativos e podem ter diferentes formas de atuação como diferença de potencial, alteração de temperatura, imposição de cargas mecânicas, dentre outros. Sua classificação está relacionada ao fator que altera as propriedades (Oliveira e Savi, 2013).

As Ligas com Memória de Forma – ou *Shape Memory Alloys (SMA)* – enquadram-se em materiais inteligentes pois têm a capacidade de retornar a seu estado original se submetidos às condições adequadas de temperatura ou carregamento. Elas começaram a ser estudadas na década de 1930, mas foi em 1962 que o NOL – *U.S. Naval Ordenance Laboratory* descobriu o efeito de memória de forma na liga Ni-Ti que, posteriormente, passou a ser chamada de Nitinol como forma de homenagem ao laboratório (Garcia, 2015).

Desde 1960 houve um aumento significativo na quantidade de publicações relacionadas a materiais inteligentes (*Smart Materials*) e ligas com memória de forma (*Shape Memory Alloy*), o que demonstra a relevância de estudar mais sobre o tema. Com auxílio do banco de dados *Scopus*, é possível observar esse número em ascensão, principalmente a partir dos anos 2000, o que é ilustrado na Figura 1.

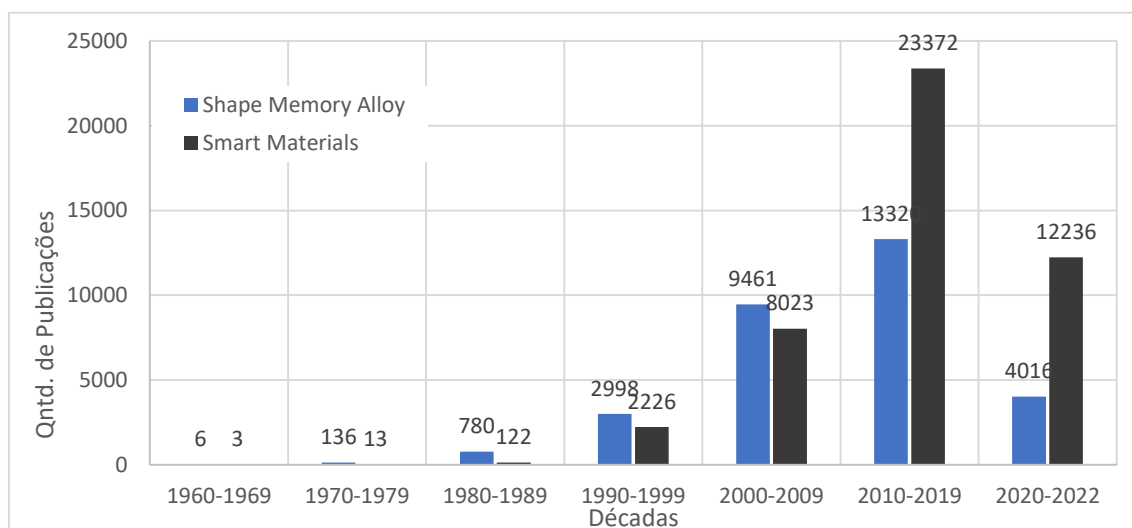


Figura 1 - Quantidade de publicações sobre *Shape Memory Alloy* (Ligas com memória de forma) e *Smart Materials* (materiais inteligentes) desde 1960 (*Scopus*, 2022)

Em relação a molas, existem pesquisas publicadas desde 1864, mas foi a partir da década de 1950 que o número de publicações passou a se acentuar significativamente. A Figura 2 – contendo gráfico construído com dados do *Scopus* - demonstra essa intensificação no estudo sobre o assunto *Springs* (molas).

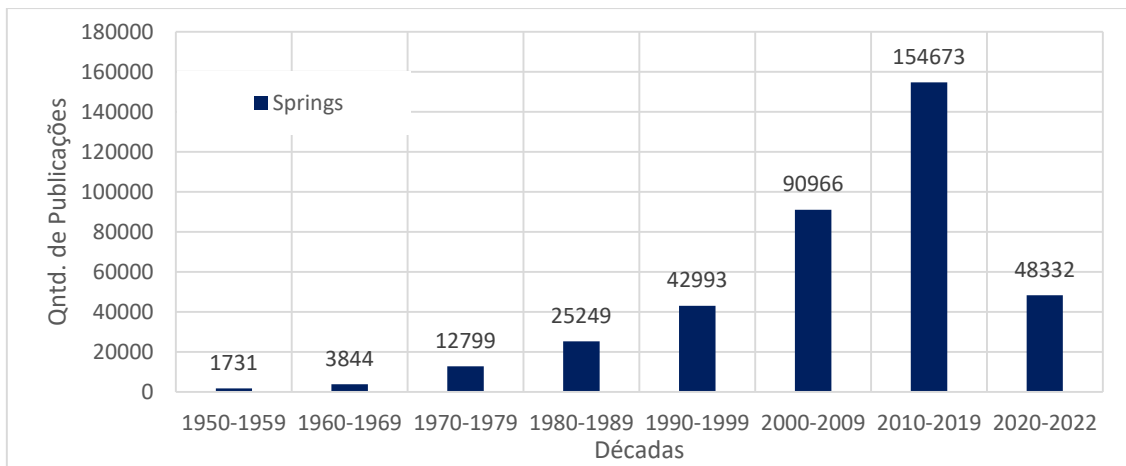


Figura 2 - Quantidade de publicações sobre *Springs* (molas) desde 1950 (*Scopus*, 2022)

Tal crescimento no número de publicações sobre materiais inteligentes, ligas com memória de forma e molas pode estar relacionado à propagação dos temas e, conseqüente, ao maior interesse em descobrir possibilidades de aplicações e soluções comerciais. É interessante ressaltar que a última coluna com os dados da presente década ainda está em construção, visto que esses dados foram coletados em Junho de 2022.

Dentre esses trabalhos, é interessante ressaltar alguns que foram importantes para compreensão do comportamento das SMA e aplicações do material.

Tobushi e Tanaka (1991) discutiram sobre as regiões de transformação de uma mola de SMA, enquanto observaram as relações de tensão e temperatura no material, e entre carga e deflexão da mola. Concluíram que a relação de carga e deflexão pode ser representada por um modelo simples associando tensão na mola e temperatura e, para certa deflexão máxima, é possível promover a recuperação de força e tensão pelo aumento de temperatura. Além disso, afirmam que a região transformada na seção transversal do fio aumenta da superfície para o centro durante a aplicação de uma carga, mas se diminui do centro para a superfície no processo de descarregamento.

Shaw e Kyriakides (1995) exploraram os aspectos termomecânicos da liga Ni-Ti, conduzindo experimentos em faixas de temperaturas e deformações as quais induziam o efeito de memória de forma e a pseudoelasticidade. Utilizou-se um fio de Ni-Ti em experimentos uniaxiais com temperaturas variando de 20 a 100 °C. Os autores perceberam que a taxa de carga tem importante influência no comportamento final. Ainda, a tensão e a temperatura podem ser usadas para identificar as transformações de fase a diferentes taxas de carregamento.

Paiva (2004) apresenta um modelo constitutivo para as ligas de memória de forma considerando assimetria e plasticidade do comportamento termomecânico dessas ligas. Os autores comparam soluções numéricas e experimentais buscando assegurar a validade do modelo, permitindo a descrição de características importantes como a pseudoelasticidade, efeito de memória de forma, transformações de fase por variações de temperatura, entre outras características.

An *et al.* (2012) realizam a determinação de uma estrutura para seleção dos parâmetros necessários para projeto de uma mola de SMA, a fim de se obter força e acionamento desejados para aplicação em um atuador. Eles consideram um modelo que representa a relação entre força e deflexão do atuador em dois estados: 100% martensítico ou 100% austenítico. Ao final realizaram um estudo de caso que demonstrou que a estrutura atende aos requisitos e pode ser útil para que engenheiros e pesquisadores possam utilizar atuadores com molas de SMA sem a necessidade de conhecer profundamente os modelos termomecânicos.

1.2 MOTIVAÇÃO E JUSTIFICATIVA

Existem diversas aplicações para molas de SMA que exploram o comportamento termomecânico tanto com o efeito de memória de forma, ligados à temperatura, como na pseudoelasticidade, relacionado a cargas mecânicas.

Analisando a aplicação da mola de SMA em uma máquina de calor, Monteiro (2012) desenvolve um modelo térmico e mecânico apropriado para utilizar o efeito de memória de forma da mola no acionamento da máquina. O rendimento global da máquina apresentada mostrou que o projeto seria viável se comparado a outras máquinas, mas com limitações ligadas ao coeficiente de atrito.

A partir da variação de temperatura, a mola de SMA é capaz de realizar o acionamento de um circuito elétrico, devido à sua transformação de fase. Tal aplicação é estudada por (Fernandes, 2014) no projeto de uma mola de comportamento reversível, com recuperação elástica de 53,06%, utilizado em um atuador linear termoativado.

A característica dissipativa das Ligas de Memória de Forma em regime pseudoelástico as torna favoráveis para a aplicação em Absorvedores Dinâmicos de Vibrações (ADV), utilizados em sistemas buscando atenuação de vibrações indesejadas,

que poderiam vir a comprometer a operacionalidade, integridade física e eficiência de equipamentos. Bandeira (2017) apresenta um estudo dos tratamentos térmicos na mola de SMA, buscando a maior dissipação de energia.

Já Silva (2017) analisa numericamente dois sistemas contendo mola helicoidal de SMA, investigando a atenuação de vibrações resultantes das transformações de fase do material, onde são estudados modelos termomecânicos, assim como as facilidades e dificuldades de implementação do trabalho.

Sendo assim, observa-se na modelagem de molas helicoidais de SMA e análise de seu comportamento em sistema dinâmico a possibilidade de pesquisa sobre material inteligente com interesse comercial e aplicações no desenvolvimento tecnológico. Para isso, ainda são colocados em prática métodos e análises vistos durante a graduação, sendo possível aprimorá-los.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

O objetivo desse trabalho é verificar o comportamento pseudoelástico de uma mola helicoidal com liga de memória de forma (SMA - *Shape Memory Alloy*), além de explorar a influência da alteração nos parâmetros da mola. Tendo essa análise, ainda é possível comparar seu comportamento ao de uma mola elástica e entender as vantagens de aplicação do material inteligente.

1.3.2 Objetivos Específicos

- Modelar através da simulação computacional uma mola de SMA para obter as curvas de força-deslocamento;
- Obter o comportamento da mola através do modelo da Brinson;
- Validar as simulações numéricas com dados experimentais e analíticos;
- Explorar a influência do diâmetro da mola no comportamento da mola de SMA;
- Estudar o comportamento da mola de SMA em um sistema dinâmico de massa-mola-amortecedor, com implementação de um Runge-Kutta de 4ª ordem.

1.4 ESTRUTURA DO TRABALHO

O primeiro capítulo busca introduzir o tema, apresentando a justificativa para esse trabalho enquanto explica brevemente sobre o comportamento da mola de SMA, faz referência a aplicações e trabalhos relacionados. Ainda contém o objetivo geral e objetivos específicos que buscam ser alcançados ao decorrer do projeto.

O segundo capítulo apresenta a fundamentação teórica que será necessária compreender para o desenvolvimento e validação das próximas etapas, abordando os conceitos de mola, liga de memória de forma, modelo constitutivo, vibrações e outros temas relacionados à pesquisa.

No terceiro capítulo, está apresentada a metodologia onde é possível encontrar a maneira que a definição dos parâmetros iniciais, coleta e análise de dados serão realizadas.

Os resultados estão no quarto capítulo, assim como as discussões relacionadas a eles. Por fim, no quinto capítulo estão apresentadas as conclusões obtidas após o estudo realizado.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 MOLAS

Molas podem ser entendidas como elementos elásticos capazes de sofrer grandes deformações e recuperarem totalmente ou parcialmente seu estado original pois armazenam grande quantidade de energia. Foram feitas para resistir ou se defletir quando submetidas a cargas de tração, compressão ou torção (Budynas & Nisbett, 2011).

Elas são usadas em amortecimento, armazenagem de energia, distribuição de cargas e ainda como limitantes de vazão, dentre outras aplicações. Podem ser classificadas em helicoidais e planas. A mais usada em mecânica é a helicoidal, que pode ser cilíndrica ou cônica e sua seção pode ser circular, quadrada ou retangular (Norton, 2013).

Molas helicoidais podem transformar forças axiais em torção e tensão cisalhante na seção transversal do fio (Budynas & Nisbett, 2011). Geralmente, quatro parâmetros geométricos precisam ser considerados na modelagem de uma mola helicoidal, além da consideração sobre seu material. São eles: o diâmetro médio da mola, D , que considera a média entre o diâmetro externo, D_o , e o diâmetro interno, D_i ; o diâmetro do fio, d ; o número de espiras, N ; e o comprimento livre, L_f . A Figura 3 abaixo ilustra os parâmetros citados.

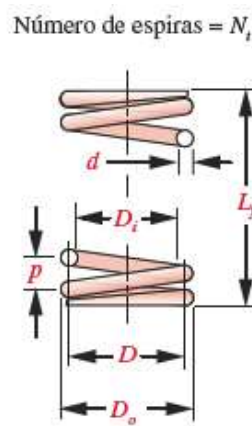


Figura 3 - Parâmetros geométricos de uma mola helicoidal (D_o , D_i , d , N e L_f) (Norton, 2013)

O passo (p) tem relação direta com o comprimento da mola e é referente à distância entre 2 espiras como na Figura 3. Tal comprimento também pode ser calculado com relação ao ângulo de *pitch*, o que é mostrado na Figura 4. O comprimento da mola está relacionado com a deflexão máxima da mola. (An, 2012)

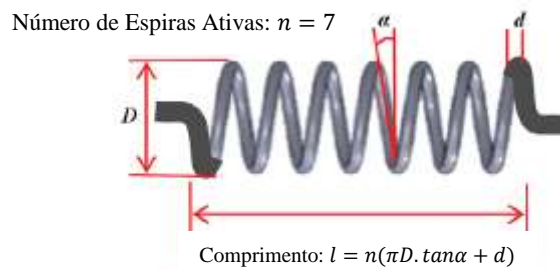


Figura 4 - Parâmetros geométricos mola da helicoidal (D, d, α , l) (An, S., 2012, Adaptado)

O índice da mola é a razão entre o diâmetro da espira, D, e o diâmetro do fio, d, como mostrado na Eq. (1). Ele possui relação com aspectos de fabricação da mola, de modo que um índice muito baixo caracteriza uma mola de difícil fabricação, pois podem surgir deformações indesejadas no fio. Por outro lado, uma mola com índice muito alto favorece a flambagem. (Norton, 2013)

$$C = \frac{D}{d} \quad (1)$$

O material em que a mola é construída tem forte influência na sua resistência à tração e no comportamento do elemento. Para construção de molas, é necessário que o material tenha alta resistência à tração, alta resistência ao escoamento e baixo módulo de cisalhamento se a aplicação exigir armazenamento de energia. Quando atuando em regime dinâmico, ainda é importante que o material seja resistente à fadiga (Norton, 2013).

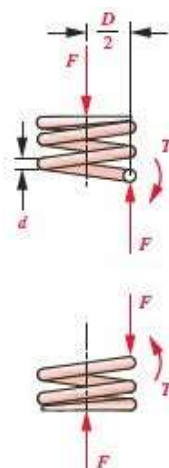


Figura 5 - Forças e torques nas espiras da mola helicoidal (Norton, 2013)

A Figura 5 ilustra que ao ser submetida a uma força de compressão, o fio da mola helicoidal sofre uma torção. Então, considerando que uma mola helicoidal é uma barra de

torção espiralada em formato de hélice, a deflexão dela com fio redondo pode ser considerada como a mostrada na Eq. (2):

$$u = \frac{8FD^3N}{d^4G} \quad (2)$$

onde, além dos parâmetros D, d e N citados anteriormente, estão presentes F sendo a força axial aplicada na mola e G como o módulo de cisalhamento do material.

Sendo assim, rearranjando a Eq. (2), o cálculo da constante da mola é feito como mostra a Eq. (3):

$$k = \frac{F}{u} = \frac{d^4G}{8D^3N} \quad (3)$$

Quando o material utilizado nas molas é SMA, a distribuição da tensão de cisalhamento é não-linear por conta da transformação de fase sofrida para grandes transformações. Esse comportamento tem impacto nos deslocamentos produzidos que são maiores nas molas de SMA em comparação com molas elásticas, sob mesma força (Garcia, 2015). An *et al.* (2012) propõem e validam um modelo que considera dois estados para as molas SMA: estado completamente martensítico (100% Martensita) ou estado completamente austenítico (100% Austenita).

Sendo assim, os fenômenos de transição podem ser desconsiderados e somente os casos de força máxima e deslocamento são analisados. Com isso, é possível afirmar a abordagem linear e, conseqüentemente que a força e o deslocamento possuem a relação linear citada anteriormente. Explicitando a força em função do deslocamento, a Eq. (4):

$$F = \frac{d^4G}{8D^3N} u \quad (4)$$

2.2 MATERIAIS

2.2.1 Propriedades Mecânicas dos materiais

Ensaio de tração podem ser realizados em materiais com o objetivo de se obter informações sobre suas propriedades mecânicas. Esses ensaios tracionam até a ruptura um corpo de prova padrão a uma carga constante, enquanto medem os alongamentos resultantes. O resultado deles é registrado em um gráfico da tensão em função da deformação (Callister, 2007). Na Figura 6 é possível observar a parte elástica desse gráfico.



Figura 6 - Gráfico tensão-deformação (Callister, 2007, Adaptado)

A partir do ensaio de tração é possível extrair informações sobre as propriedades mecânicas do material tais como resistência ao escoamento, módulo de elasticidade, tenacidade, entre outras. O módulo de elasticidade, E , é o coeficiente angular da parte elástica. A tensão e a deformação de cisalhamento são proporcionais uma à outra pelo módulo de cisalhamento, G , sendo a inclinação da região elástica linear da curva tensão-deformação de cisalhamento. O coeficiente de Poisson, ν , é a razão entre as deformações lateral e axial. Para materiais isotrópicos, isto é, que apresentam mesmo comportamento independente da direção, a relação entre essas propriedades é apresentada na Eq. (5) (Callister, 2007).

$$E = 2G(1 + \nu) \quad (5)$$

2.3 LIGAS COM MEMÓRIA DE FORMA

Em 1932, Arne Olander descobriu com a liga de AuCd que transformações de fase podem gerar comportamentos interessantes desconhecidos até então. Foi na década de 1960 que as ligas com memória de forma começaram a ganhar mais importância, com as ligas de NiTi e CuZnAl. Outras ligas ainda exibem o efeito de memória de forma, sendo elas: TiNb, CuAlNi, AgCd, FePt, dentre outras (Silva, 2016).

Dentre as ligas citadas, as ligas de Ni-Ti são as com maior interesse comercial uma vez que combinam memória de forma com boas propriedades mecânicas (Garcia, 2015). Por outro lado, a liga de CuZnAl, possui como vantagem o seu custo (Coura, 2016). Algumas das propriedades de ambas as ligas estão apresentadas na Tabela 1.

Tabela 1 - Propriedades das ligas NiTi e CuZnAl (Coura, 2016)

Propriedade	Liga NiTi	Liga CuZnAl
Deformação Recuperável	máx. 8%	máx. 4%
Tensão gerada na recuperação	máx.400 MPa	máx.200 MPa
Fadiga	10^5 ($\varepsilon = 0,02$) 10^7 ($\varepsilon = 0,005$)	10^2 ($\varepsilon = 0,02$) 10^5 ($\varepsilon = 0,005$)
Resistência a corrosão	Boa	Problemática, especialmente sob tensão
Trabalhabilidade	Pobre	Razoável
Conformação	Relativamente fácil	Razoavelmente difícil

As propriedades apresentadas pelas ligas com memória de forma são possíveis graças as transformações de fase que ocorrem por carregamento térmico ou mecânico (Garcia, 2015). A Figura 7 aproxima o comportamento das SMAs e ilustra as tensões que induzem transformações de fase na pseudoelasticidade. A Figura 8 ilustra as transformações de fase para o efeito de memória de forma, que além da tensão e da deformação, tem influência da temperatura. (Júnior, 2012)

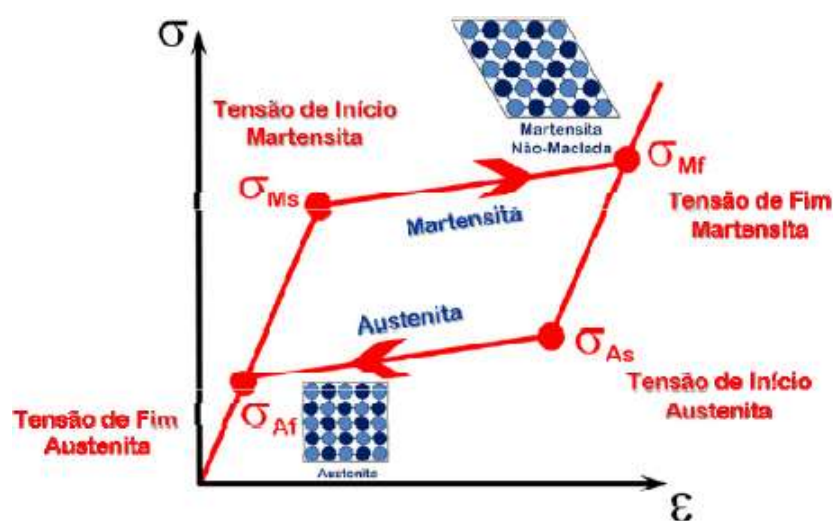


Figura 7 – Tensões de transformação de fase – Pseudoelasticidade (Júnior, 2012)

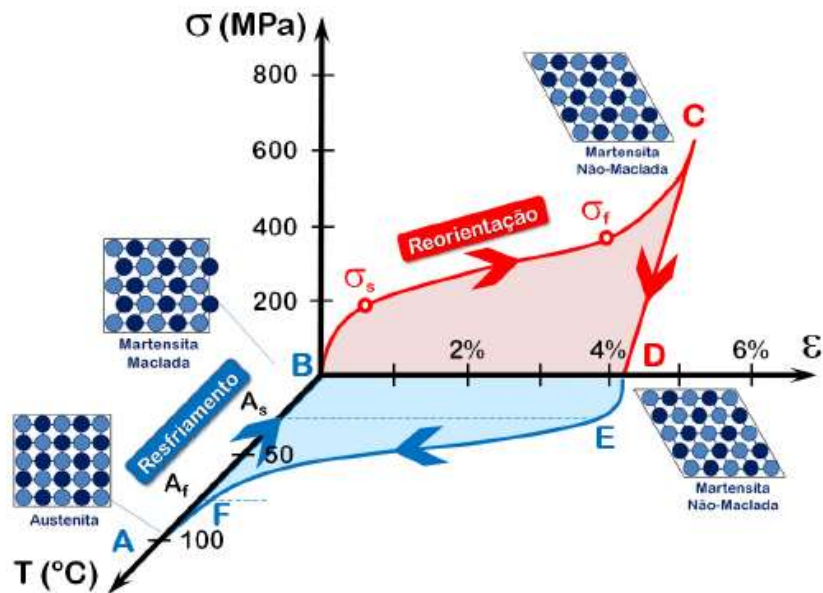


Figura 8 - Transformação de fase – Efeito de Memória de Forma (Júnior, 2012)

A martensita não-maclada é induzida por tensão, enquanto a martensita maclada é induzida pela temperatura. Na ausência de tensão, a martensita é estável a baixas temperaturas, enquanto para altas temperaturas a austenita é estável (Garcia, 2015). Mas, dependendo do histórico termomecânico da liga e de sua composição química, tanto a austenita como a martensita podem ser estáveis a temperatura ambiente (Coura, 2016).

O comportamento de um fio de SMA pode ser representado pela curva tensão-deformação, como representado pela Figura 9 (a), da qual é possível obter os módulos de elasticidade da fase austenítica e da fase martensítica, apresentados como E_A e E_M , respectivamente. O comportamento pseudoelástico (também conhecido como superelástico) de molas de SMA é melhor expressado pela curva força-deslocamento na Figura 9 (b), similar à curva tensão-deformação do fio. (Savi *et al.*, 2015)

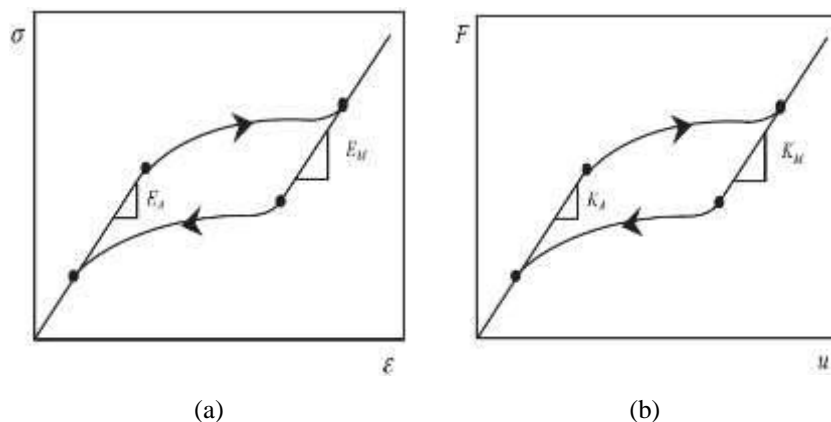


Figura 9 - Comportamento do (a) Fio e (b) Mola de SMA (Garcia, 2015)

Dessa segunda curva na Figura 9 (b) é possível extrair o elemento de rigidez de ambas as fases, apresentadas como k_A e k_M . O potencial dissipativo do material pode ser obtido através da área do laço de histerese (Bandeira, 2017).

2.3.1 Aplicações

As aplicações das ligas com memória de forma estão presentes nas mais diversas áreas, desde a área aeroespacial até a odontológica, estando muito presente em problemas de vibrações e dissipação de energia.

Na biomedicina, têm aplicação cardiovascular como procedimento de angioplastia e em tratamento de aneurismas como dispositivos conhecidos como *stents*, que ajudam a sustentar vasos sanguíneos e artérias. As malhas do material inteligente são inseridas no corpo com o diâmetro menor e, à medida que fica exposta à temperatura corporal, se expande de maneira que apoia as paredes do vaso. A diferença de diâmetros entre o *stent* antes ou depois da expansão pode ser vista na Figura 10 (Bandeira, 2017).

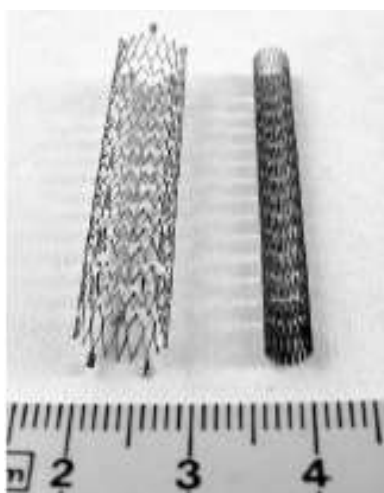


Figura 10 - *Stents* (Bandeira, 2017)

Pesquisadores da NASA, Boeing e GE desenvolveram *chevrons* com fitas de material de SMA para buscar reduzir os ruídos provocados pelas turbinas de aviões. Nesse caso, a diferença de temperatura é capaz de alterar o formato do dispositivo de forma que os ruídos conseguem ser amenizados (Silva, 2016). Na Figura 11 é possível ver o dispositivo que estão nas saídas das turbinas e atuam na mistura de gases, o que reduz o ruído (Garcia, 2015).



Figura 11 - Chevrons em turbinas de aeronaves (Garcia, 2015)

Pela grande capacidade dissipativa, tem sido utilizada em Absorvedores Dinâmicos de Vibrações (ADV) (Silva, 2017). Na robótica está sendo utilizada em músculos artificiais (Bandeira, 2017). Além dessas, existem muitas outras aplicações, evidenciando que as ligas com memória de forma podem trazer muitas vantagens.

2.3.2 Modelo Constitutivo da Brinson

Os modelos constitutivos são utilizados para descrever o comportamento de determinado material. Para SMAs, busca-se obter a relação entre tensões, deformações e temperatura. No caso de estudo desse trabalho, busca-se reproduzir a pseudoelasticidade; logo, considerando a temperatura constante. O modelo constitutivo unidimensional de Brinson para esse comportamento pode ser descrito como Eq. (6).

$$\sigma - \sigma_0 = (E\varepsilon - E_0\varepsilon_0) - \varepsilon_L(E\beta - E_0\beta_0) \quad (6)$$

Em que 0 indica o estado inicial e E é o módulo de elasticidade equivalente da liga, E_A é o módulo de elasticidade da austenita, E_M é o da martensita, β é a fração volumétrica da martensita por tensão e ε_L é a máxima deformação que a liga consegue recuperar através do carregamento térmico. O módulo de elasticidade equivalente é definido de acordo com a fração volumétrica de martensita pela Eq. (7):

$$E = E(\beta) = E_A + \beta(E_M - E_A) \quad (7)$$

Se $\beta = 0$, a microestrutura será completamente austenita. Por outro lado, se $\beta = 1$, a fase será completamente martensita. Considerando uma temperatura acima da temperatura de fim da transformação austenítica, as equações das transformações de fase podem ser divididas para dois casos, onde as Eq. (8) e Eq. (9) definem as equações da fração volumétrica β :

- (i) Transformação de austenita para martensita se $\sigma_s^{cr} + C_M(T - M_s) < \sigma < \sigma_f^{cr} + C_M(T - M_s)$

$$\beta = \frac{1-\beta_0}{2} \cos \left\{ \frac{\pi}{\sigma_s^{cr} - \sigma_f^{cr}} [\sigma - \sigma_f^{cr} - C_M(T - M_s)] \right\} + \frac{1+\beta_0}{2} \quad (8)$$

- (ii) Transformação da martensita para austenita se $C_A(T - A_f) < \sigma < C_A(T - A_s)$

$$\beta = \frac{\beta_0}{2} \left\{ \cos \left[\frac{\pi}{A_f - A_s} \left(T - A_f - \frac{\sigma}{C_A} \right) \right] + 1 \right\} \quad (9)$$

Pinho, 2022 propõe Modelo da Brinson modificado que continua sendo descrito pela Eq. (6) e as transformações de fase seguem os mesmos dois casos, mas com diferença na definição do β . As Eq. (10) e Eq. (14) definem o β para cada um dos casos e as Eq. (11), Eq. (12), Eq. (15) e Eq. (16) demonstram como calcular os coeficientes.

- (i) Transformação de austenita para martensita se $\sigma_s^{cr} + C_M(T - M_s) < \sigma < \sigma_f^{cr} + C_M(T - M_s)$

$$\beta = a\sigma + b \quad (10)$$

$$a = \frac{1-\beta_0}{\sigma_f^{cr} - \sigma_s^{cr}} \quad (11)$$

$$b = \frac{\beta_0 [\sigma_f^{cr} + C_M(T - M_s)] - [\sigma_s^{cr} + C_M(T - M_s)]}{\sigma_f^{cr} - \sigma_s^{cr}} \quad (12)$$

ou

$$\beta = \frac{(1-\beta_0)\sigma + \beta_0\sigma_f^{cr} - \sigma_s^{cr} - (1-\beta_0)C_M(T - M_s)}{\sigma_f^{cr} - \sigma_s^{cr}} \quad (13)$$

- (ii) Transformação da martensita para austenita se $C_A(T - A_f) < \sigma < C_A(T - A_s)$

$$\beta = c\sigma + d \quad (14)$$

$$c = \frac{\beta_0}{C_A(A_f - A_s)} \quad (15)$$

$$d = -\frac{\beta_0 C_A(T - A_f)}{C_A(A_f - A_s)} \quad (16)$$

$$\beta = \frac{\beta_0\sigma - \beta_0 C_A(T - A_f)}{C_A(A_f - A_s)} \quad (17)$$

Assim como utilizado em Pinho (2022), o modelo linear é interessante pois permite que o deslocamento seja imposto (utilizado como entrada) e, a partir dele, seja calculada a força. Sem a modificação linear, isso não seria possível. É possível reescrever a Eq. (6) considerando a força e o deslocamento, que serão utilizados na análise do comportamento da mola de SMA. Para isso, considera-se a Eq. (18).

$$F - F_0 = (Eu - E_0u_0) - u_r(E\beta - E_0\beta_0) \quad (18)$$

2.4 VIBRAÇÕES

2.4.1 Sistema massa-mola-amortecedor

O estudo de vibrações mecânicas está associado a movimentos oscilatórios. Um oscilador constituído por um sistema de massa-mola-amortecedor submetido a uma excitação externa em função do tempo é ilustrado na Figura 12.

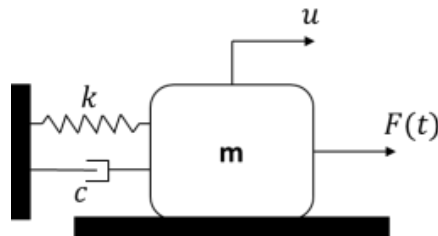


Figura 12 - Diagrama esquemático de um sistema massa-mola-amortecedor

O diagrama de corpo livre para o modelo proposto está apresentado na Figura 13, onde F_R é a força de restituição realizada pela mola, F_D é a força dissipativa que o amortecedor exerce, N indica a normal, P o peso e $F(t)$ indica o forçamento.

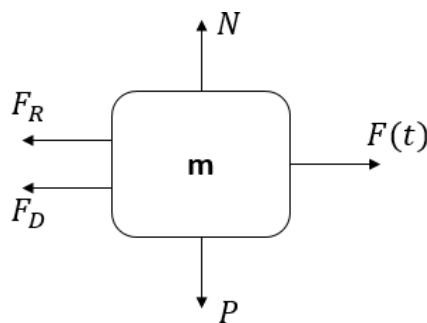


Figura 13 - Diagrama de corpo livre sistema massa-mola-amortecedor

A solução apresentada foi baseada no conteúdo apresentado em Savi & Paula (2017). A força restitutiva da mola linear é representada pela Eq. (19)

$$F_R = k u \quad (19)$$

Enquanto a força dissipativa do amortecedor está na Eq. (20)

$$F_D = c \dot{u} \quad (20)$$

Aplicando a 2ª Lei de Newton para somatório de forças na vertical, tem-se a Eq. (21)

$$\sum F_y = m \ddot{u}_y \quad (21)$$

Logo, $N - P = 0$. Assim, $N = P = mg$, sendo g a aceleração da gravidade.

Agora usando a 2ª Lei de Newton para a horizontal:

$$\sum F_x = m \ddot{u}_x \quad (22)$$

Então, $F(t) - F_R - F_D = m \ddot{u}$.

Rearranjando os termos da Eq. (22), tem-se a Eq. (23)

$$m \ddot{u} + c \dot{u} + k u = F(t) \quad (23)$$

Dividindo toda a equação pela massa, tem-se a Eq. (24)

$$\ddot{u} + \frac{c}{m} \dot{u} + \frac{k}{m} u = \frac{1}{m} F(t) \quad (24)$$

Assumindo que: $\frac{c}{m} = 2 \xi \omega_n$, $\frac{k}{m} = \omega_n^2$ e $\frac{F(t)}{m} = f(t)$, onde ω_n é a frequência natural, chega-se a Eq. (25)

$$\ddot{u} + 2 \xi \omega_n \dot{u} + \omega_n^2 u = f(t) \quad (25)$$

Se o forçamento é harmônico, pode-se descrevê-lo por uma função de seno ou cosseno, como apresentado na Eq. (26).

$$F(t) = k f_0 \cos(\Omega t) \quad (26)$$

A equação de movimento é uma equação diferencial ordinária (EDO) de segunda ordem. A solução geral, $u(t)$, é a soma da solução homogênea, u_h - resolvida sem considerar o forçamento - e a solução particular, u_p - relacionada ao forçamento. Considerando um forçamento harmônico, essas soluções são conhecidas.

As Eq. (27) e Eq. (28) são relacionadas com a solução homogenia conhecida, análoga à solução para vibração livre. Onde as constantes A_1 e A_2 são obtidas a partir das condições iniciais $u(0) = u_0$ e $\dot{u}(0) = v_0$.

$$u_h = e^{-\xi \omega_n t} A \cos(\omega_d t - \varphi) = e^{-\xi \omega_n t} [A_1 \cos(\omega_d t) + A_2 \sin(\omega_d t)] \quad (27)$$

$$\omega_d = \omega_n \sqrt{1 - \xi^2} \quad (28)$$

Já para a solução particular, é apresentada de acordo com a Eq. (29)

$$u_p = K_1 \cos(\Omega t) + K_2 \sin(\Omega t) \quad (29)$$

Onde Ω representa a frequência do forçamento. Somando as duas soluções para chegar à solução geral, chega-se a Eq. (30).

$$(\ddot{u}_h + \ddot{u}_p) + 2\xi\omega_n(\dot{u}_h + \dot{u}_p) + \omega_n^2(u_h + u_p) = \omega_n^2 f_0 \cos(\Omega t) \quad (30)$$

Agrupando os termos relacionados à cada uma das soluções, tem-se a Eq. (31).

$$(\ddot{u}_h + 2\xi\omega_n\dot{u}_h + \omega_n^2 u_h) + (\ddot{u}_p + 2\xi\omega_n\dot{u}_p + \omega_n^2 u_p) = \omega_n^2 f_0 \cos(\Omega t) \quad (31)$$

Assim, obtém-se duas Equações Diferenciais Ordinárias (EDOs), onde a primeira é igual a zero, pois está relacionada à solução homogênea e a segunda – ligada ao forçamento – será analisada abaixo. Tomando como base a Eq. (29) como solução, inicialmente olha-se para as derivadas dela, como mostrado no sistema abaixo (Eq. (32)).

$$\begin{cases} \dot{u}_p = K_1\Omega \cos(\Omega t) - K_2\Omega \sin(\Omega t) \\ \ddot{u}_p = -K_1\Omega^2 \sin(\Omega t) - K_2\Omega^2 \cos(\Omega t) \end{cases} \quad (32)$$

Substituindo a Eq. (32) na parte da solução particular na Eq. (31), tem-se a Eq. (33).

$$[-K_1\Omega^2 \sin(\Omega t) - K_2\Omega^2 \cos(\Omega t) + 2\xi\omega_n[K_1\Omega \cos(\Omega t) - K_2\Omega \sin(\Omega t)] + \omega_n^2[K_1 \cos(\Omega t) + K_2 \sin(\Omega t)] = \omega_n^2 f_0 \cos(\Omega t) \quad (33)$$

Colocando todos os termos com $\sin(\Omega t)$ e $\cos(\Omega t)$ em evidência, chega-se à Eq. (34).

$$(-K_1\Omega^2 - 2\xi\omega_n K_2\Omega + K_1\omega_n^2) \sin(\Omega t) + (-K_2\Omega^2 + 2\xi\omega_n K_1\Omega + K_2\omega_n^2) \cos(\Omega t) = \omega_n^2 f_0 \cos(\Omega t) \quad (34)$$

Sendo assim, para a solução ser válida em qualquer instante de tempo, o sistema da Eq. (35) é montado.

$$\begin{cases} -K_1\Omega^2 - 2\xi\omega_n K_2\Omega + K_1\omega_n^2 = 0 \\ -K_2\Omega^2 + 2\xi\omega_n K_1\Omega + K_2\omega_n^2 = \omega_n^2 f_0 \end{cases} \quad (35)$$

Então,

$$\begin{cases} K_1 = \left[\frac{2\xi\omega_n^2\Omega}{(\omega_n^2 - \Omega^2)^2 - (2\xi\omega_n\Omega)^2} \right] f_0 \\ K_2 = \left[\frac{\omega_n^2(\omega_n^2 - \Omega^2)}{(\omega_n^2 - \Omega^2)^2 - (2\xi\omega_n\Omega)^2} \right] f_0 \end{cases} \quad (36)$$

Conhecendo a Eq. (29) e as constantes na Eq. (36), é possível reescrevê-la, como mostra a Eq. (37).

$$u_p = U_p \cos(\Omega t - \varphi_p) \quad (37)$$

Onde, as Eq. (38) e Eq. (39) demonstram os termos inicialmente desconhecidos.

$$U_p = \frac{f_0}{\left\{ \left[1 - (\Omega/\omega_n)^2 \right]^2 + \left[(2\xi\Omega)/\omega_n \right]^2 \right\}^{1/2}} \quad (38)$$

$$\varphi_p = tg^{-1} \frac{2\xi\Omega/\omega_n}{1 - (\Omega/\omega_n)^2} \quad (39)$$

Dessa forma, a solução geral da equação de movimento está na Eq. (40).

$$u(t) = e^{-\xi\omega_n t} [A_1 \cos(\omega_d t) + A_2 \sin(\omega_d t)] + U_p \cos(\Omega t - \varphi_p) \quad (40)$$

Derivando a expressão da solução em relação ao tempo, chega-se a Eq. (41).

$$\dot{u}(t) = (-\xi\omega_n) e^{-\xi\omega_n t} [A_1 \cos(\omega_d t) + A_2 \sin(\omega_d t)] + e^{-\xi\omega_n t} [-A_1 \omega_d \sin(\omega_d t) + A_2 \omega_d \cos(\omega_d t)] - U_p \Omega \sin(\Omega t - \varphi_p) \quad (41)$$

Definindo as condições iniciais como $u(0) = u_0 = 0$ e $\dot{u}(0) = v_0 = 0$, tem-se a Eq. (42) e, conseqüentemente, a Eq. (43).

$$u(0) = A_1 + U_p \cos(\varphi_p) = 0 \quad (42)$$

Então,

$$A_1 = -U_p \cos(\varphi_p) \quad (43)$$

Para segunda condição, onde a velocidade – primeira derivada – é zero, apresenta-se a Eq. (44).

$$\dot{u}(0) = (-\xi\omega_n)A_1 + A_2\omega_d + U_p\Omega \sin(\varphi_p) = 0 \quad (44)$$

Substituindo o resultado da Eq. (43) na Eq. (44),

$$(\xi\omega_n)U_p \cos(\varphi_p) + A_2\omega_d + U_p\Omega \sin(\varphi_p) = 0 \quad (45)$$

Assim, chega-se a Eq. (46).

$$A_2 = - \frac{U_p [(\xi\omega_n) \cos(\varphi_p) + \Omega \sin(\varphi_p)]}{\omega_d} \quad (46)$$

2.4.2 Variáveis de Estado

Em um sistema dinâmico, a equação diferencial é utilizada para modelar o problema. A solução dessa equação diferencial representa uma evolução quadro a quadro do que acontece no sistema e, assim, a evolução das variáveis de estado. Sua definição

consiste em definir a equação diferencial de forma que a derivada da variável seja igual a uma função que só dependa das variáveis de estado e do tempo, como mostra a Eq. (47).

$$\{\dot{X}\} = \{g(\{X\}, t)\}, \quad \{X\} \in \mathbb{R}^n, \quad t \in \mathbb{R}^1 \quad (47)$$

Para o sistema massa-mola-amortecedor de 1 gdl, as variáveis de estado podem ser definidas como mostra a Eq. (48).

$$x_1 = u \quad \text{e} \quad x_2 = \dot{u} \quad (48)$$

Aplicando as variáveis na equação de movimento Eq. (30), chega-se as Eq. (49) e Eq. (50).

$$\dot{x}_2 + 2\xi\omega_n x_2 + \omega_n^2 x_1 = \omega_n^2 f_0 \cos(\Omega t) \quad (49)$$

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{Bmatrix} = \begin{Bmatrix} x_2 \\ -2\xi\omega_n x_2 - \omega_n^2 x_1 + \omega_n^2 f_0 \cos(\Omega t) \end{Bmatrix} \quad (50)$$

Dessa forma, divide-se a EDO de 2ª ordem em duas EDOs de 1ª ordem. Sendo assim, é possível utilizar-se de métodos numéricos para a resolução do sistema dinâmico.

O espaço de fase é a representação do deslocamento pela velocidade e é útil para avaliar o comportamento das amplitudes.

2.4.3 Ressonância

As frequências naturais são de grande importância para o estudo de vibrações em estruturas visto que o fenômeno de ressonância pode levar a desastres. Um dos exemplos mais conhecidos desse fenômeno é o da ponte de Tacoma Narrows que entrou em colapso depois de ser atingida por ventos com frequência da força coincidindo com suas frequências naturais. A Figura 14 ilustra essa ponte em ressonância e depois colapsada. (Abreu, 2021)

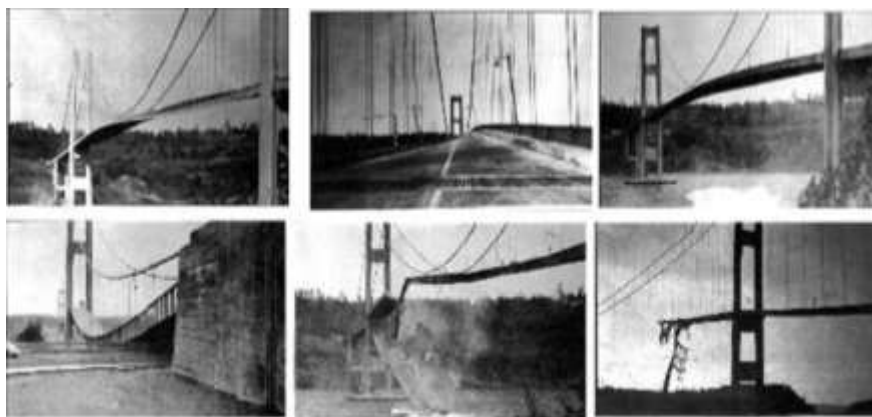


Figura 14 - Ponte de Tacoma Narrows (Abreu, 2021)

A ressonância pode então ser entendida como a frequência de forçamento que resulta na maior amplitude de vibração do sistema. A razão entre a frequência natural e a frequência de forçamento pode ser útil para analisar a ressonância em um sistema dinâmico, pois aparece na amplitude e no ângulo de fase da solução particular, como mostram as Eq. (38) e Eq. (39).

Para encontrar o ponto máximo, pode-se usar a derivada igual a zero, então Eq. (51)

$$\frac{du_p}{dx} = \frac{dU_p}{dx} \cos(\Omega t - \varphi_P) = [f_0 \cos(\Omega t - \varphi_P)] \frac{d}{dx} \left\{ [(1 - x^2)^2 + (2\xi x)^2]^{-\frac{1}{2}} \right\} = 0 \quad (51)$$

Em que,

$$\frac{d}{dx} \left\{ [(1 - x^2)^2 + (2\xi x)^2]^{-\frac{1}{2}} \right\} = -\frac{1}{2} [(1 - x^2)^2 + (2\xi x)^2]^{-\frac{3}{2}} [(-4x + 4x^3) + (8\xi^2 x)] = 0 \quad (52)$$

$$[(-4x + 4x^3) + (8\xi^2 x)] = 4x[x^2 - 1 + 2\xi^2] = 0 \quad (53)$$

A solução que é interessante é com a razão diferente de zero, logo tem-se Eq. (54)

$$[x^2 - 1 + 2\xi^2] = 0 \rightarrow x^2 = 1 - 2\xi^2 \rightarrow x = \sqrt{1 - 2\xi^2} = \frac{\Omega_r}{\omega_n} \rightarrow \Omega_r = \omega_n \sqrt{1 - 2\xi^2} \quad (54)$$

2.4.4 Método Runge-Kutta

Carl David Runge – físico e matemático - e M. Wilhelm Kutta – matemático - propuseram o método de Runge-Kutta para obter soluções aproximadas de valor inicial. O método consiste na comparação de um polinômio de Taylor apropriado para eliminar o cálculo das derivadas, podendo ser entendido como um aperfeiçoamento do método de Euler (Sterza, 2016).

O método de Runge-Kutta descrito abaixo foi baseado em Sterza (2016) e, de modo geral, é dado pela Eq. (55)

$$y_{n+1} = y_n + h \quad (55)$$

onde h é a inclinação, obtida através do cálculo da inclinação em n pontos, onde n é a ordem do método. Essa ordem indica o número de pontos usados para determinar a inclinação. A expansão da série de Taylor pode ser escrita de maneira geral como a Eq. (56)

$$y(x) = y(a) + \sum_{k=1}^n \frac{y^{(k)}(a)}{k!} (x - a)^k + O((x - a)^{n+1}) \quad (56)$$

Aplicando na Eq. (56) $x = x_{n+1} = x_n + h$ e $a = x_n$, com $n = 4$, chega-se a Eq. (57)

$$y(x_{n+1}) = y(x_n + h) = y(x_n) + hy' + \frac{h^2}{2!} y''(x_n) + \frac{h^3}{3!} y'''(x_n) + \frac{h^4}{4!} y^{(4)}(x_n) + \frac{h^5}{5!} y^{(5)}(\zeta) \quad (57)$$

E ζ é um número entre x_n e x_{n+1} . Para Runge Kutta de 4ª ordem, busca-se encontrar as constantes apropriadas que permitam a Eq. (58)

$$y_{n+1} = y_n + h(c_1 K_1 + c_2 K_2 + c_3 K_3 + c_4 K_4) \quad (58)$$

Uma solução particular muito usada consiste em assumir Eq. (59)

$$\begin{cases} c_1 = c_4 = \frac{1}{6} \\ c_2 = c_3 = \frac{1}{3} \end{cases} \quad (59)$$

Sendo assim, o método de Runge Kutta de 4ª ordem fica definido como a Eq. (60)

$$y_n = y_{n-1} + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) \quad (60)$$

onde as constantes K_1, K_2, K_3 e K_4 estão definidas nas Eq. (61), Eq. (62), Eq. (63) e (64). Assim, para calcular o instante de tempo seguinte, só é necessário o tempo anterior.

$$\{K_1\} = \{f(\{X\}_{n-1}, t_{n-1})\} \Delta t \quad (61)$$

$$\{K_2\} = \left\{ f \left(\{X\}_{n-1} + \frac{1}{2} \{K_1\}, t_{n-1} + \frac{1}{2} \Delta t \right) \right\} \Delta t \quad (62)$$

$$\{K_3\} = \left\{ f \left(\{X\}_{n-1} + \frac{1}{2} \{K_2\}, t_{n-1} + \frac{1}{2} \Delta t \right) \right\} \Delta t \quad (63)$$

$$\{K_4\} = \left\{ f(\{X\}_{n-1} + \{K_3\}, t_{n-1} + \Delta t) \right\} \Delta t \quad (64)$$

Nessas equações o adicional de tempo - conhecido como o passo - está diretamente relacionado à convergência da solução, uma vez que um passo muito grande pode não ser suficiente para calcular a solução adequadamente.

2.5 SIMULAÇÃO COMPUTACIONAL

Os sistemas CAD (*Computer-Aided design*) e CAE (*Computer-Aided Engineering*) são recursos tecnológicos que permitem se obter resultados satisfatórios no

processo de desenvolvimento de produtos, se utilizados de forma adequada. Enquanto com o CAD é possível projetar um produto de forma eficiente, com CAE podem ser realizadas previsões e avaliação de funcionalidades futuras, através da simulação do desempenho do produto. (Giracca, 2021)

Segundo o Instituto ESSS, CAE pode ser entendido como um “conjunto de tecnologias de simulação computacional que permitem os engenheiros analisarem as mais diferentes opções para resolver um problema sem qualquer custo ou risco a que estaria associado normalmente”. Sendo assim, pode ser usado para validação e otimização de produtos e soluções. Uma das técnicas usada para a simulação CAE, está o Método dos Elementos Finitos, que tem se tornado cada vez mais popular devido sua versatilidade e precisão (Giracca, 2021).

O método dos elementos finitos (MEF) baseia-se na discretização do meio contínuo, o que permite aproximar os modelos contínuos (muito complexos) por modelos discretizados através da soma de problemas mais simples. Dessa forma, é realizada a divisão em um número finito de elementos com comportamento simplificado e à medida que aumenta o número de elementos no sistema, a solução se aproxima da solução complexa. A malha consiste no conjunto dos elementos finitos e dos pontos nodais, o que está apresentado na Figura 15. (Borgmann, 2012)

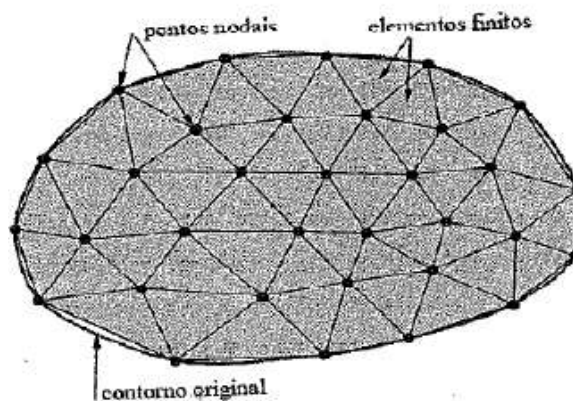


Figura 15 - Malha de Elementos Finitos (Borgmann, 2012).

O *Ansys* é um *software* comercial de simulação de engenharia e *design* 3D, que proporciona soluções acuradas em diversas áreas da física. Ele foi escolhido para realização desse trabalho pois permite a análise pelo método dos elementos finitos de ligas com memória de forma, através do modelo de Auricchio implementado em seu *software*. Segundo Mtili (2021) o comportamento superelástico desse modelo considera o material completamente isotrópico e, assim, duas transformações de fase: de Austenita

(A) para Martensita (M) e de Martensita (M) para Austenita (A). A soma das frações volumétricas de cada uma das fases (A ou M) é igual a 1.

3 MÉTODOS

Inicialmente, é realizada uma revisão bibliográfica, analisando trabalhos que abordam o tema de estudo de Ligas com Memória de Forma para compreender o que se espera do comportamento de tal material e como obtê-lo. Após, é definido que o estudo será realizado com auxílio do *software* comercial *Ansys*, de forma que o comportamento da mola helicoidal de SMA possa ser avaliado.

Os parâmetros e propriedades a serem utilizados nas simulações computacionais são definidos com base no estudo da influência da não-linearidade geométrica no comportamento mecânico de molas helicoidais com ligas de memória de forma de Savi *et al.* (2015) e a análise experimental do comportamento termomecânico das ligas com memória de forma de Garcia (2015), para ser possível realizar uma posterior validação. Os dados utilizados estão apresentados a seguir na Tabela 2 e na Tabela 3.

Tabela 2 - Propriedades da Liga de Memória de Forma (Savi *et al.* 2015).

Propriedade	Valor	Unidade
E_A	46	GPa
ν	0,3	-
σ_s^{AM}	297	MPa
σ_f^{AM}	772	MPa
σ_s^{MA}	306	MPa
σ_f^{MA}	78	MPa
ε_L	0,07	-
α	0	-
E_M	22	GPa

Tabela 3 - Parâmetros da mola (Savi *et al.* 2015)

Parâmetro	Valor	Unidade
Diâmetro do fio (d)	1,7	mm
Diâmetro da mola (D)	13,8	mm
Número de Espiras (N)	5	-

Com base nisso, a mola pode ser construída no *software Ansys* e as simulações computacionais podem ser realizadas de forma que uma das extremidades da mola se mantém fixa e sobre a outra é aplicado um deslocamento em duas etapas:

1. Carregamento: partindo de zero até X;
2. Descarregamento: de X a zero.

Tal procedimento será realizado três vezes para avaliar o comportamento da mola em menores e em maiores deslocamentos, onde os valores de X serão: 45 mm, 80 mm e 120 mm.

A partir do resultado dessas simulações será possível analisar o comportamento da mola helicoidal de SMA através dos gráficos de Força x Deslocamento para cada um dos casos

Um estudo de convergência da malha é realizado para que seja ponderado se o resultado obtido pode ser melhorado ou se está suficientemente próximo do esperado. Nele, algumas simulações são realizadas alterando o tamanho da malha e os resultados do comportamento da mola é comparado em todos os casos.

Buscando assegurar a validade dos resultados obtidos nas simulações, é realizada uma comparação entre três soluções:

1. Solução numérica: gerada pelo *Ansys*;
2. Solução analítica: implementada no *software Octave*;
3. Dados Experimentais: retirados do artigo de Savi *et al.* (2015) com auxílio do *software Get Data*.

A solução analítica é desenvolvida considerando um modelo constitutivo (Pinho, 2022) onde primeiro é realizada uma calibração das propriedades da mola de SMA. Nela, é gerado um gráfico dos dados experimentais; a partir dele são traçadas retas abrangendo a maior quantidade de pontos possíveis e, assim, são extraídos três pontos indicados na Figura 16 como 1, 2 e 3, sendo eles, respectivamente, início da transformação para martensita, 100% martensita e início da transformação para austenita. Com esses dados, a solução é implementada e é possível realizar a comparação entre as três soluções.

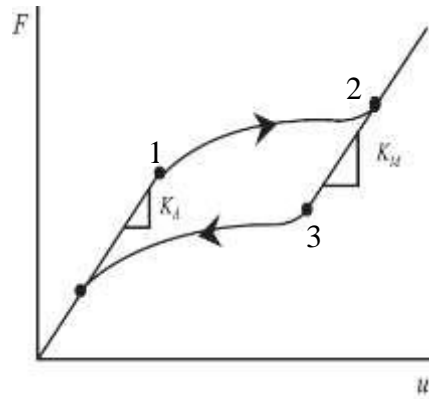


Figura 16 - Pontos para calibração da solução analítica (adaptada de Garcia, 2015)

Após a validação do modelo, a influência do diâmetro da mola helicoidal é estudada para três casos: índice da mola mínimo ($C = 6$), índice da mola médio ($C = 9$) e índice da mola máximo ($C = 12$), onde o diâmetro do fio e demais parâmetros são mantidos constantes.

Através do resultado das simulações com esses parâmetros alterados, o comportamento da mola para esses casos é observado. Para isso, o processo de calibração citado anteriormente também será realizado, com o objetivo de comparar as soluções analítica e numérica. Para finalizar, será realizada análise da influência do índice da mola nas forças de início e fim de transformações de fase.

Como o objetivo de analisar a aplicação da mola de SMA em um sistema dinâmico, inicialmente é necessário implementar o código para uma mola linear elástica, para que o mesmo possa ser validado com a solução analítica. Utiliza-se um sistema massa-mola-amortecedor, conforme Figura 17, com 1 grau de liberdade (gdl). As equações utilizadas baseando-se na Seção 2.4.1, os parâmetros inseridos no código são apresentados na Tabela 4.

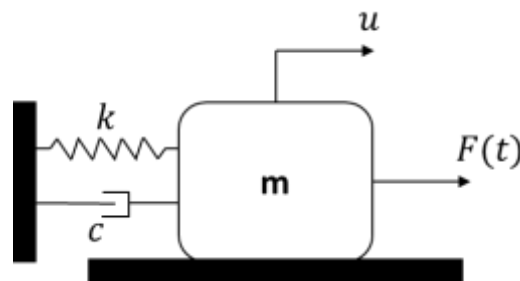


Figura 17 - Esquema de Massa-mola-amortecedor com 1gdl

Tabela 4 – Parâmetros do sistema dinâmico

Parâmetros	Valor	Unidade
m	1	kg
k_A	1193,77	N/m
ξ	0,1	-
ω_n	$\sqrt{k/m}$	
c	$2m\xi\omega_n$	
Ω	$2\omega_n$	
ω_d	$\omega_n\sqrt{1-\xi^2}$	
F	40	N
f_0	F/k	
$x_1 = u(0)$	0	m
$x_2 = \dot{u}(0)$	0	m/s

O coeficiente de rigidez da mola é utilizado como resultado da modelagem e análise do comportamento da mola SMA na fase martensítica. Para a etapa de validação utiliza-se um forçamento harmônico, conforme apresenta a Eq. (25). As soluções analíticas para a mola linear elástica são obtidas através das Eq. (37), Eq. (38), Eq. (39), Eq. (43) e Eq. (46). Já para a solução numérica, o Método Runge-Kutta de 4ª ordem é implementado, em que as variáveis de estado estão definidas na Eq. (48), para resolver a Eq. (49).

Nesse caso será avaliado como a solução $u(t)$ muda com a diminuição da frequência de forçamento e como a amplitude muda de acordo com a frequência de forçamento, além de investigar como a força altera a resposta de amplitude nas diferentes frequências.

Tendo a validação do código, é possível alterar o mesmo para a mola de SMA, onde é necessário alterar a definição da equação de movimento. Relembrando as Eq. (49) e Eq. (50) que são referentes à mola elástica:

$$\dot{x}_2 + 2\xi\omega_n x_2 + \omega_n^2 x_1 = \omega_n^2 f_0 \cos(\Omega t) \quad (49)$$

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{Bmatrix} = \begin{Bmatrix} x_2 \\ -2\xi\omega_n x_2 - \omega_n^2 x_1 + \omega_n^2 f_0 \cos(\Omega t) \end{Bmatrix} \quad (50)$$

Primeiramente é realizada uma alteração no termo referente à força da mola $\omega_n^2 x_1$, passando a ser F_m/m , onde F_m é calculada para cada caso através do Modelo da Brinson modificado implementado. Dessa maneira, a equação das variáveis de estado pode ser escrita como:

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{Bmatrix} = \begin{Bmatrix} x_2 \\ -2 \xi \omega_n x_2 - F_m/m + \omega_n^2 f_0 \cos(\Omega t) \end{Bmatrix} \quad (65)$$

Mantendo a frequência de forçamento constante, realiza-se um estudo para averiguar qual o melhor passo para realizar o estudo paramétrico, visto que um passo menor resulta em uma solução mais precisa, só que com uma demanda computacional mais alta; por outro lado, um passo maior gera uma solução mais rápida, mas com menor precisão. Pode-se analisar a diferença entre os comportamentos linear da mola elástica e não-linear da mola de SMA.

Tendo o valor do passo definido, analisa-se a partir de qual momento a solução está em regime permanente. Assim, são observadas as soluções da equação de movimento no tempo, o espaço de fase e a curva força-deslocamento para a razão de Ω/ω_n variando entre 0 e 2, para três forças diferentes. Esse intervalo para a razão entre as frequências é escolhido pois acima da razão de 2, o comportamento de ambas as molas é muito semelhante.

4 RESULTADOS E DISCUSSÕES

4.1 SIMULAÇÕES COMPUTACIONAIS

No *software* comercial *Ansys* o material foi definido no campo de *Engineering Data* conforme Tabela 2, na categoria de *Shape memory alloy - Superelasticity*. Após essa etapa, foi definida a geometria da mola através do *Design Modeler*. O método utilizado foi o *Sweep*, onde foram definidos o raio da mola ($D/2$), o diâmetro do fio (d) e o número de espiras (N), representados na Figura 18 por H3 e D2, respectivamente. Os valores desses parâmetros estão apresentados na Tabela 3. Além desses, também foi necessário inserir a altura da mola, representada na Figura 18 como $V1 = 12$ mm.

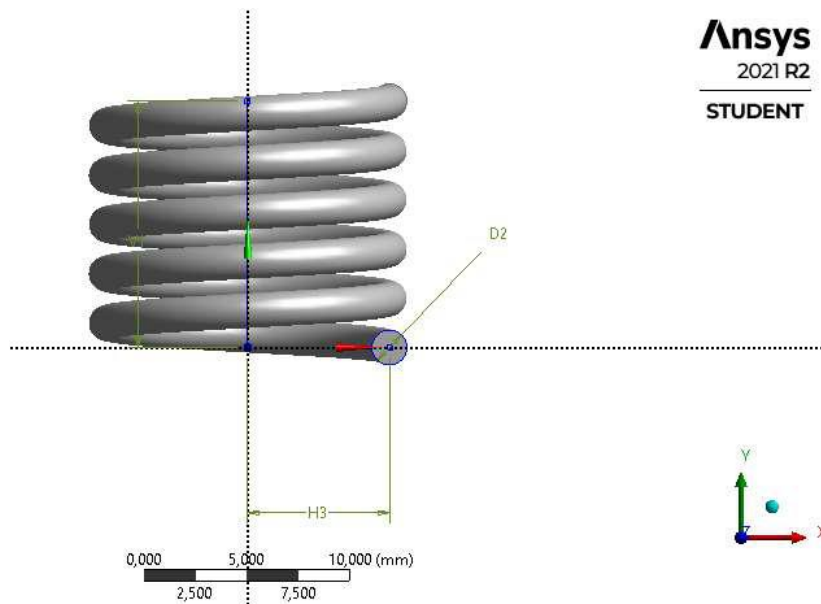


Figura 18 - Parâmetros da Mola (*Ansys*)

Para a definição da malha dois parâmetros são utilizados: o tamanho da face e o método. O elemento utilizado foi o Solid 186 – elemento 3D, quadrático de 20 nós – pois ele permite modelar o comportamento complexo do material. O tamanho do elemento foi configurado como 0,5 mm na face da mola. No método, foi escolhido o *Sweep Method* com 100 divisões, onde *source* e *target* são as faces de extremidade da mola. Essa malha

gerou 9368 nós (Figura 19) e no item 4.2 será apresentado o estudo de convergência da mesma.

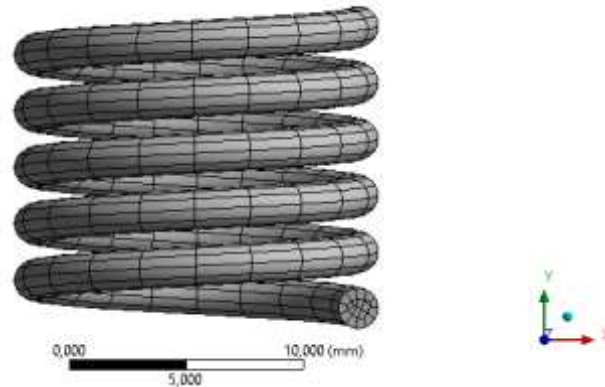


Figura 19 - Malha escolhida - 9368 nós

As condições de contorno são definidas mantendo uma das extremidades fixa (A) e sobre a outra foi aplicado um deslocamento (B) em Y, como pode ser observado na Figura 20.

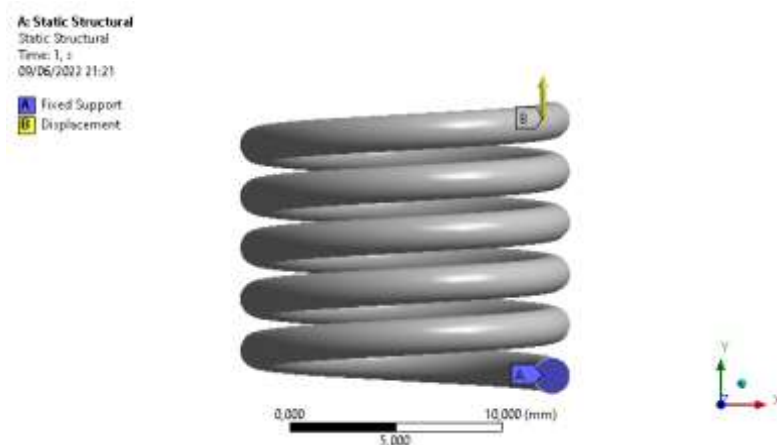


Figura 20 - Condições de Contorno (Ansys)

O deslocamento é aplicado em duas etapas: a primeira com o carregamento partindo de zero até o máximo alcançando 120 mm; depois, partindo de 120 mm e chegando a zero. O deslocamento nas outras direções X e Z é mantido em zero. Essas definições podem ser visualizadas na Figura 21.

Steps	Time [s]	<input checked="" type="checkbox"/> X [mm]	<input checked="" type="checkbox"/> Y [mm]	<input checked="" type="checkbox"/> Z [mm]
1	1	= 0,	0,	= 0,
2	1	0,	120,	0,
3	2	= 0,	0,	= 0,

Figura 21 - Steps no Ansys e Definição do deslocamento – 120 mm

Algumas configurações são alteradas na análise, como o número de steps, e a quantidade mínima e máxima de substeps, para melhorar a convergência da simulação.

Além desses, o campo referente a grandes deformações foi habilitado, como mostra a Figura 22.

Details of "Analysis Settings"	
Step Controls	
Number Of Steps	2,
Current Step Number	1,
Step End Time	1, s
Auto Time Stepping	On
Define By	Substeps
Initial Substeps	10,
Minimum Substeps	10,
Maximum Substeps	1000,
Solver Controls	
Solver Type	Program Controlled
Weak Springs	Program Controlled
Solver Pivot Checking	Program Controlled
Large Deflection	On
Inertia Relief	Off
Quasi-Static Solution	Off

Figura 22 - Detalhes das Configurações da Análise (Ansys)

Como resultados dessa simulação, foram retirados a Tensão Equivalente de von Mises, tendo alcançado 606,43 MPa nas áreas em vermelho na Figura 23. A maior tensão está sendo observada na parte interna da mola.

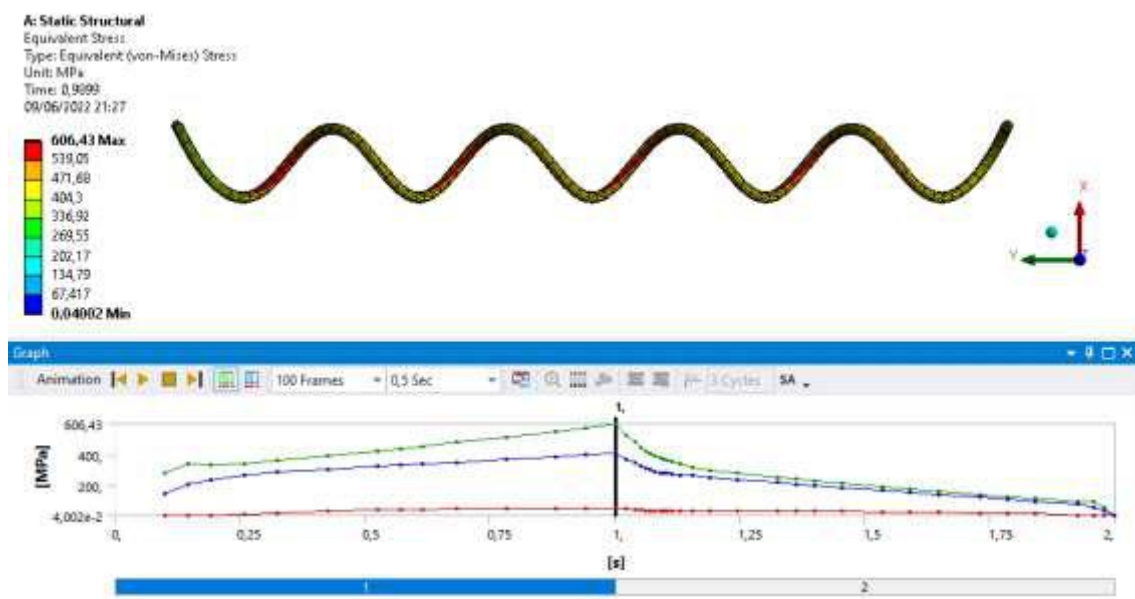


Figura 23 - Tensão equivalente de von Mises referente a 120 mm de deslocamento máximo

A Figura 24 mostra o gráfico de deslocamento aplicado sobre uma das extremidades da mola em relação ao tempo normalizado. O deslocamento foi aplicado em 2 passos.

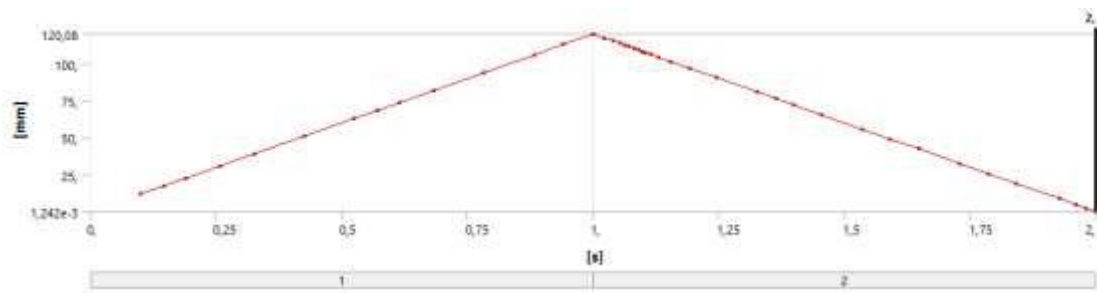


Figura 24 - Deslocamento 120 mm: carregamento e descarregamento

As reações de força enquanto o deslocamento foi aplicado foram coletadas pelo *software* e estão apresentadas em função do tempo e é mostrado na Figura 25, a seguir.

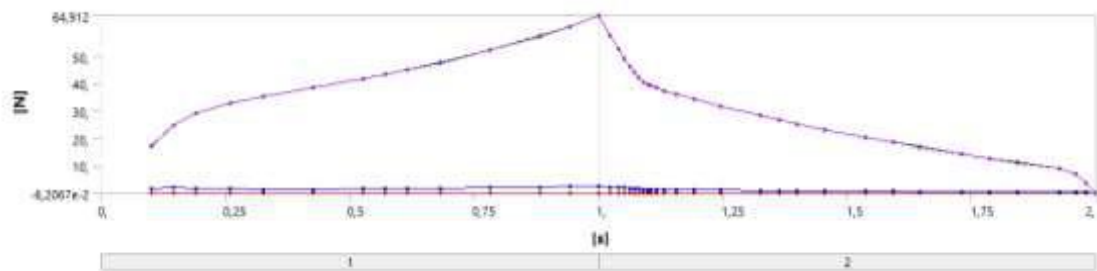
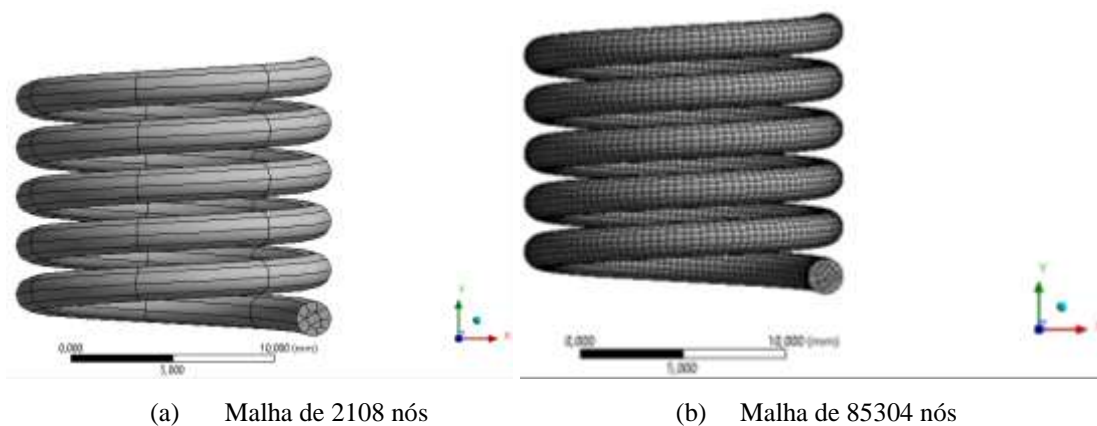


Figura 25 – Diagrama Força x tempo referente a 120 mm de deslocamento máximo

Com os dados da Força (Figura 25) e deslocamento (Figura 24) é possível construir a relação entre força e deslocamento para visualizar o comportamento da mola helicoidal para esse caso. A Figura 9 ilustra o comportamento esperado.

4.2 ESTUDO DE CONVERGÊNCIA DE MALHA

A Figura 15 demonstra uma malha de elementos finitos e a seção 2.5 explica o impacto de se refinar a malha na precisão do resultado. Sendo assim, para avaliar se a malha utilizada poderia ser melhorada para obter um resultado mais preciso, o estudo de convergência de malha é realizado. Na Figura 26 é possível observar a diferença entre (a) malha de 2108 nós, onde os elementos estão maiores, e (b) malha de 85304 nós, mais refinada e com elementos menores. No primeiro caso (a), o resultado pode se distanciar da solução real, enquanto o segundo (b) pode oferecer uma informação mais precisa. Porém, também existe uma diferença na demanda computacional, uma vez que no segundo caso o computador precisará processar muito mais informação.



(a) Malha de 2108 nós

(b) Malha de 85304 nós

Figura 26 – Comparação entre as malhas menos refinada e mais refinada

A Tabela 5 abaixo resume os parâmetro que foram alterados para cada uma das malhas geradas, onde o tamanho do elemento no *Ansys* está em *Face sizing* e número de divisões é definido em *Sweep Method*. O tempo é referente a quantos minutos o *software* demorou para gerar a solução numérica.

Tabela 5 - Parâmetros utilizados nas diferentes malhas e tempo de simulação

	Tamanho do Elemento	Nº de Divisões	Tempo
2108 nós	0,65 mm	35	1 m 52 s
3602 nós	0,60 mm	50	2 m 11 s
9368 nós	0,50 mm	100	6 m 9 s
11228 nós	0,50 mm	120	9 m 45 s
25092 nós	0,45 mm	200	21 m 37 s
59587 nós	0,40 mm	500	52 m 15 s
85304 nós	0,35 mm	600	1 h 31 m

A Figura 27 mostra como foi descrito o comportamento da mola em cada uma das malhas.

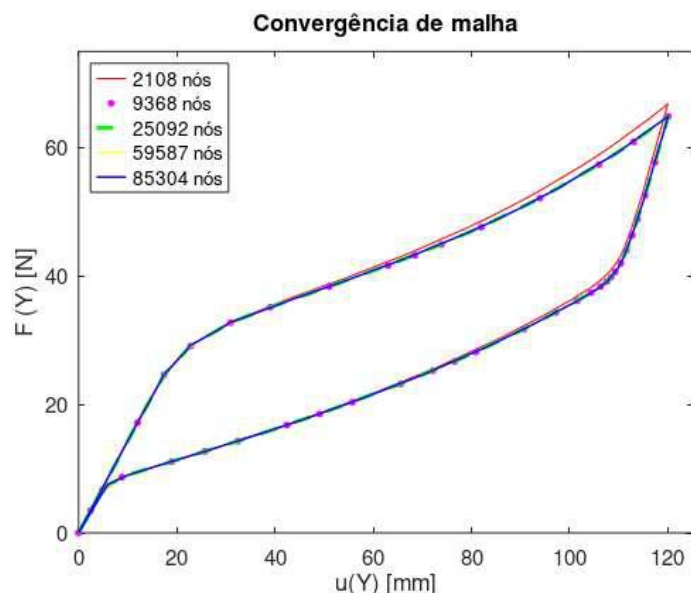


Figura 27 - Estudo de convergência de malha

A malha de 2108 nós se distanciou um pouco do comportamento gerado pelas outras malhas. A malha utilizada (9368 nós) não possui diferenças significativas no comportamento quando comparada à malha de 85304 nós e, além disso, possui uma demanda computacional menor se comparado à malha mais refinada. Portanto, a malha de 9368 representa bem o comportamento e não serão necessárias alterações nesse quesito.

4.3 VALIDAÇÃO DA SIMULAÇÃO DO ANSYS

Com o objetivo de assegurar a validade das simulações realizadas, os dados gerados pelo *Ansys* foram comparados com dados experimentais retirados de Savi *et al.* (2015) pelo *software GetData*. A comparação inicial está apresentada na Figura 28, onde os resultados para os três deslocamentos (45mm, 80 mm e 120 mm) estão sobrepostos.

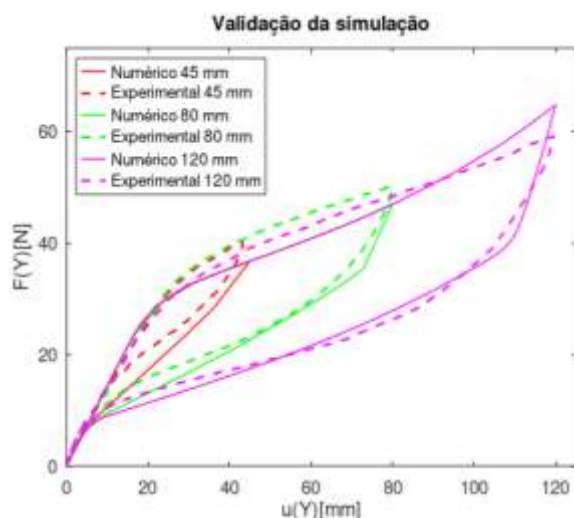


Figura 28 - Soluções numérica e experimental - deslocamentos máximos de: 45 mm, 80 mm e 120 mm.

Buscando aprofundar essa análise, a comparação da solução numérica gerada pelo *Ansys* também foi comparada com a solução analítica implementada no *software Octave*, utilizando o modelo constitutivo, onde a implementação do Modelo da Brinson Modificado foi baseada em Pinho, 2022, com adaptações. Os códigos foram utilizados alterando o deslocamento no que está disponível no Anexo A.

4.3.1 Calibração do Modelo

Para garantir que a solução analítica está bem implementada, é realizada a calibração do modelo, onde são traçadas retas para extração dos pontos 1, 2 e 3, como explicado na seção de Métodos. A Figura 29 mostra as retas construídas como aproximação para selecionar os pontos de transformação de fase para o deslocamento de 120 mm.

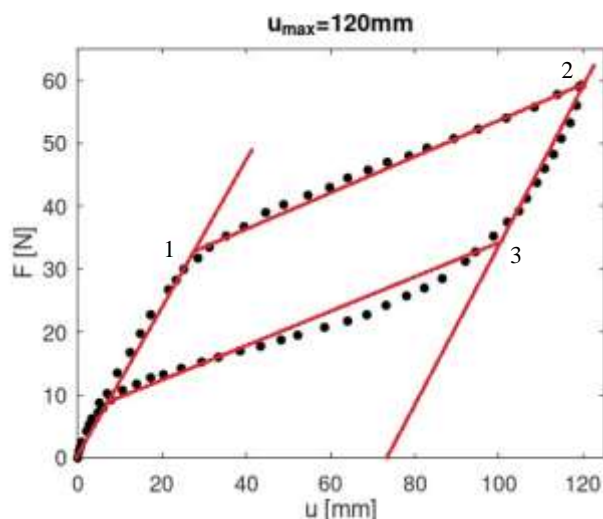


Figura 29 - Calibração do modelo analítico

Depois dessa etapa, o *software Get Data* foi utilizado para obter os valores correspondentes de força e deslocamento para cada um desses pontos, o que está apresentado na Tabela 6.

Tabela 6 - Pontos de transformação de fase – Deslocamento 120 mm

Deslocamento	120 mm	
	F [N]	u [mm]
Ponto 1	33,08	27,71
Ponto 2	59,16	119,80
Ponto 3	34,25	100,17

Ainda buscando averiguar se o modelo analítico está respondendo de acordo com o esperado, foi realizada uma comparação entre a solução numérica (solução do *Ansys*) e a analítica (utilizando o modelo constitutivo) para a sequência de deslocamento carregando de zero a 120 mm, descarregando para 40 mm, carregando novamente até 80 mm e, por fim, voltando a zero. O resultado está apresentado na Figura 30.

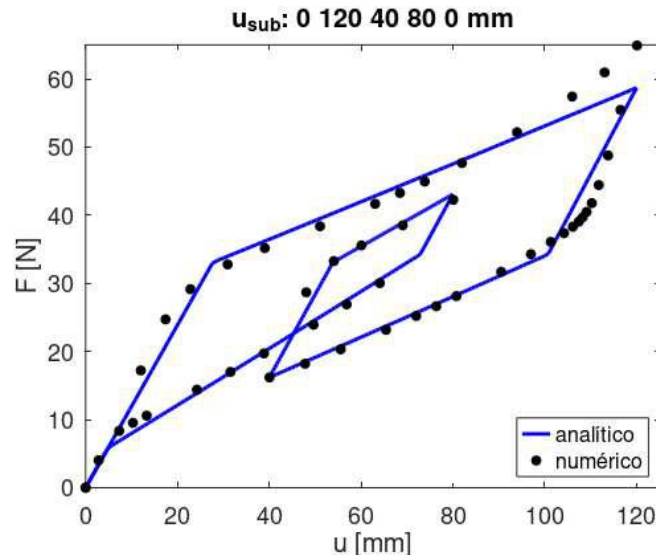


Figura 30 - Calibração com *subloops*[0 120 40 80 0]

Observando os dados apresentados é possível afirmar que o modelo utilizado para obter a solução analítica é eficaz ao reproduzir o comportamento da mola de SMA, ainda sendo capaz de representar o comportamento em casos intermediários com transformação de fase incompleta.

4.3.2 Resultados da validação

A Figura 31 mostra a comparação entre esses dados para o deslocamento máximo de 45 mm, aplicado na direção y, da mesma forma que explicado na seção 4.1. Para gerá-lo foram inseridos no programa os pontos de transformação de fase apresentados na Tabela 6, bem como o deslocamento residual foi ajustado para cada um dos casos: deslocamento de 45mm, de 80 mm e de 120 mm.

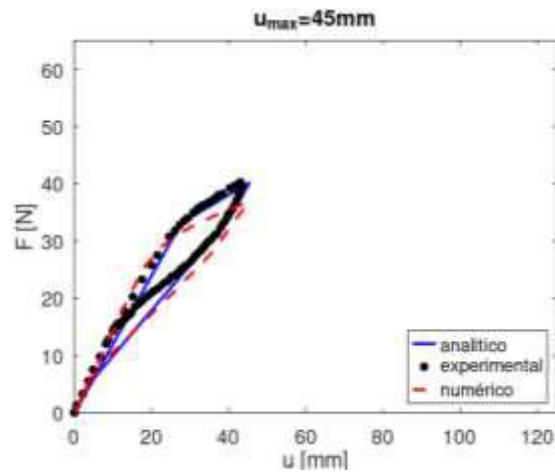


Figura 31 - Soluções analítica, experimental e numérica – 45 mm

De forma semelhante, a Figura 32 apresenta a sobreposição das soluções para o deslocamento máximo de 80 mm.

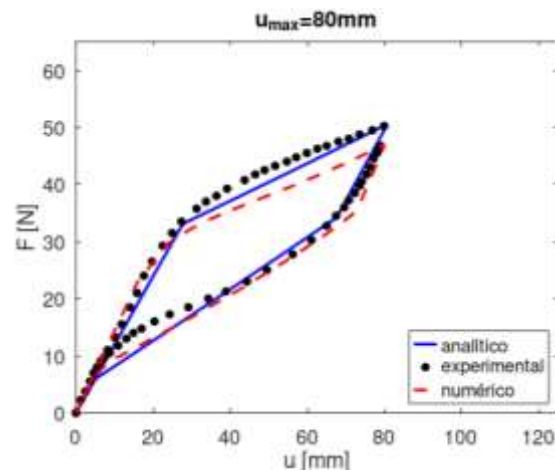


Figura 32 - Soluções analítica, experimental e numérica - 80 mm

O último deslocamento analisado foi o de 120mm, presente na Figura 33.

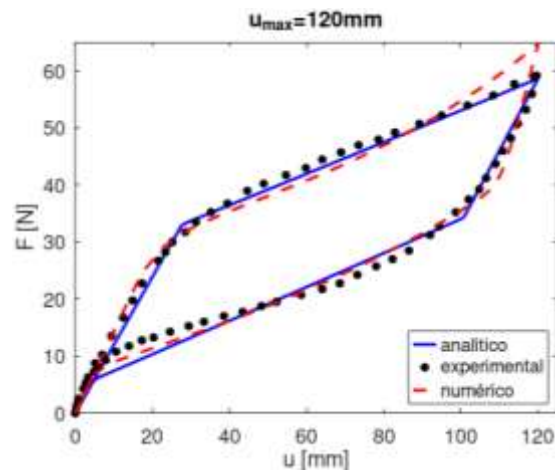


Figura 33 - Soluções analítica, experimental e numérica - 120 mm

A partir dessa análise, é possível afirmar que o modelo constitutivo utilizado na solução analítica está suficientemente próximo do resultado experimental e que a solução numérica está próxima deles, o que comprova a validade das simulações realizadas e do modelo empregado.

4.4 INFLUÊNCIA DE ALTERAÇÃO NO DIÂMETRO DA MOLA

Tendo o comportamento da mola de SMA apresentado pelas curvas de histerese nas Figura 31, Figura 32 e Figura 33, agora é interessante avaliar como uma alteração no diâmetro da mola pode influenciar no seu comportamento. Retomando a Eq. (1) que define o índice da mola como sendo a razão entre o diâmetro da mola e do fio, este parâmetro é alterado mantendo o índice da mola em 6, 9 e 12, sendo esses valores respectivamente, o mínimo, o médio e o máximo recomendados.

4.4.1 Diâmetro da mola

Buscando avaliar a influência da alteração do diâmetro da mola no seu comportamento, o diâmetro do fio – 1,7 mm – e demais parâmetros foram mantidos sem alteração, enquanto o diâmetro da mola, D , foi alterado para chegar aos índices de mola 6, 9 e 12.

Assim como apresentado na seção 4.3.1, foi realizada a calibração para cada um dos casos, o que está apresentado na Figura 34.

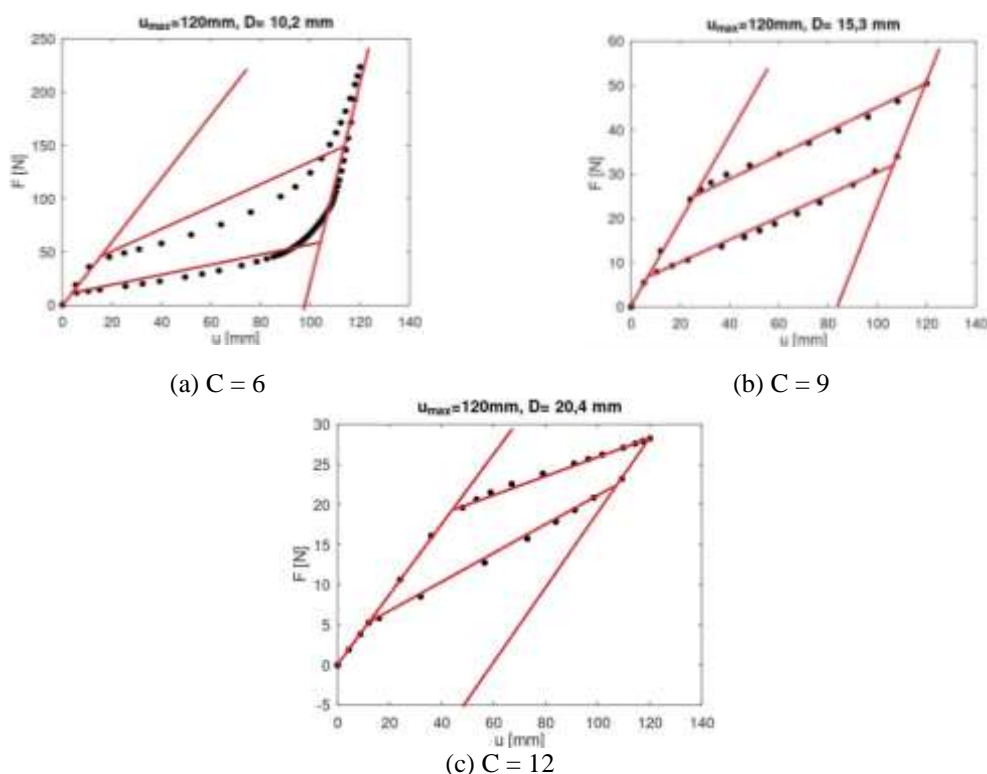


Figura 34 - Calibração do modelo analítico - Alteração do diâmetro da mola (D)

Dessa calibração são extraídos os pontos de transformação de fase, sendo a ponto 1 o início da transformação de austenita para martensita, o ponto 2 o fim da transformação para martensita e o ponto 3 o início da transformação para austenita novamente. Os valores extraídos da Figura 34 são apresentados na Tabela 7 para cada um dos índices de mola, resultantes de uma alteração no diâmetro da mola.

Tabela 7 - Pontos de transformação de fase - Alteração do diâmetro da mola (D)

Diâmetro da mola	C = 6 D = 10,2 mm		C = 9 D = 15,3 mm		C = 12 D = 20,4 mm	
	F [N]	u [mm]	F [N]	u [mm]	F [N]	u [mm]
Ponto 1	47,90	16,45	25,33	26,13	19,41	44,516
Ponto 2	148,95	113,5	50,30	119,36	28,11	119,36
Ponto 3	58,383	103,87	31,98	106,45	22,34	107,10
Ka [N/mm]	2,91		0,97		0,44	
Km [N/mm]	9,36		1,42		0,47	

Esses pontos foram extraídos para serem usados como dados de entrada no modelo que reproduz o comportamento da mola. Assim, foi possível obter a Figura 35, que ilustra as soluções numérica e analítica para os três casos.

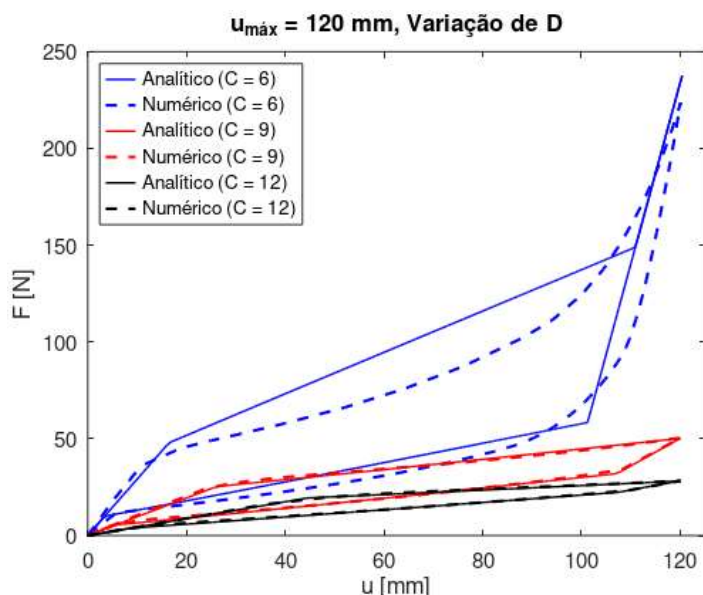


Figura 35 - Alteração do diâmetro da mola no comportamento da mola - C = 6, C = 9 e C = 12

É possível observar que na alteração do diâmetro da mola, quanto menor o índice da mola – e, conseqüentemente, o menor diâmetro da mola – maior é a área do gráfico, mas o comportamento se afasta da aproximação linear.

A Figura 36 mostra como o coeficiente de rigidez da mola na fase austenita, K_a , e na fase martensita, K_m , variam quanto o diâmetro da mola é alterado.

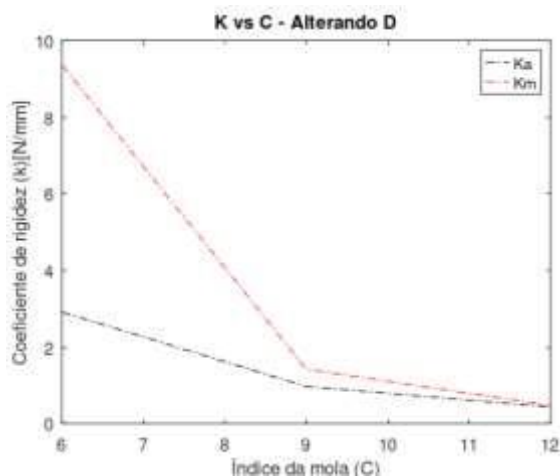


Figura 36 - Coeficientes de rigidez (k_a e k_m) e variação do índice da mola - (D)

À medida que o diâmetro da mola cresce e assim o índice da mola aumenta, ambos os coeficientes diminuem, resultando em uma mola menos rígida. O coeficiente da martensita apresenta uma diferença muito grande entre os índices 6 e 9. Já o da austenita apresenta uma diferença bem menor.

4.4.2 Forças de Transformação de Fase

De forma geral, é interessante avaliar como as forças de início de transformação de fase variam de acordo com o índice da mola para a alteração do diâmetro da mola.

As forças necessárias para transformação de fase inicial de austenita para martensita têm sua representação na Figura 37.

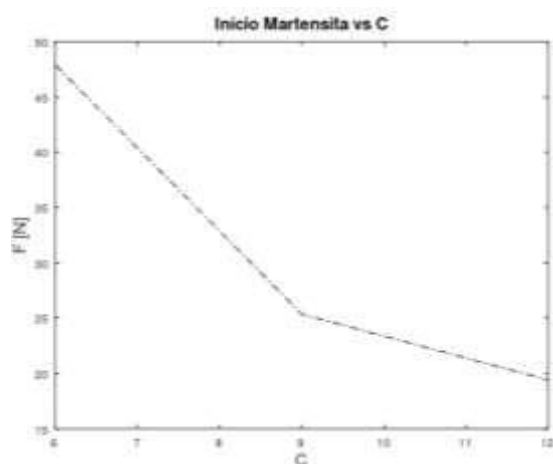


Figura 37 - Forças de início da transformação para fase martensita por índice da mola

Para análise do fim da transformação completa para martensita, houve uma maior diferença entre os valores de força, o que está na Figura 38.

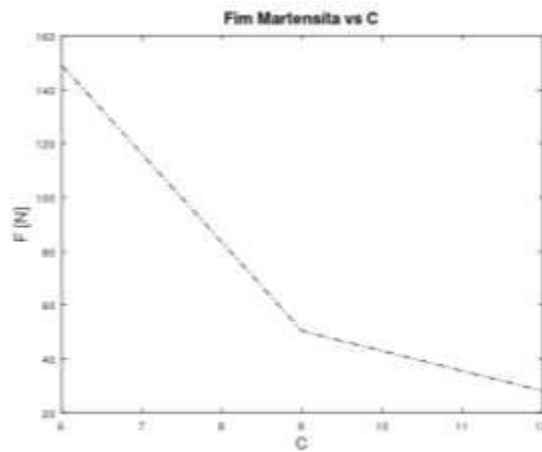


Figura 38 - Forças de fim da transformação completa para martensita por índice da mola
 Por fim, considerando o início da transformação para austenita, a Figura 39.

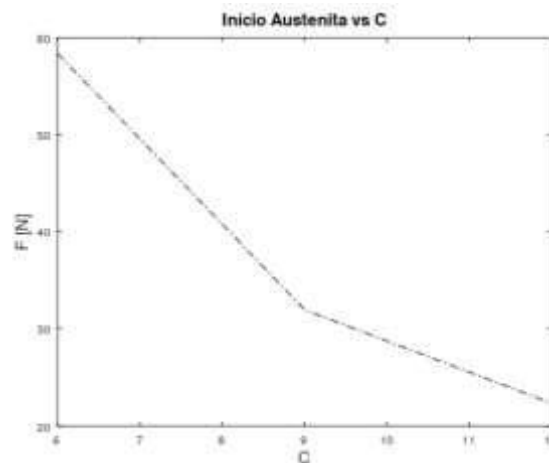


Figura 39 - Forças de início da austenita por índice da mola

Para os três gráficos é possível ver que o ponto que sofreu maior impacto da alteração do índice da mola foi o fim da transformação de fase para martensita.

4.5 MOLA DE SMA APLICADA A UM SISTEMA DINÂMICO

4.5.1 Validação da implementação

As soluções numérica e analítica foram implementadas como indicado nos Métodos e o código está disponível no Anexo B, desenvolvido pela autora. O coeficiente de rigidez da mola utilizado foi o mesmo da austenita obtido nas simulações do *Ansys*, pois a mola linear não possui transformação de fase. Essas soluções para o sistema dinâmico estão apresentadas na Figura 40.

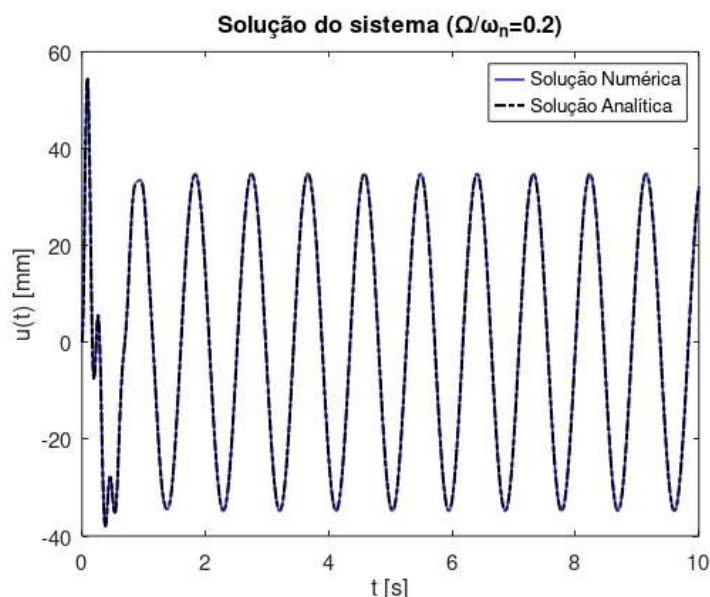


Figura 40 - Soluções Numérica e Analítica do Sistema dinâmico

As soluções analítica e numérica sobrepostas não apresentam diferenças perceptíveis, o que indica que foram implementadas corretamente.

Buscando visualizar como a frequência de forçamento influencia a solução geral, a Figura 41 foi construído, alterando o valor de Ω em Ω/ω_n para que essa razão varie de 0 a 2.

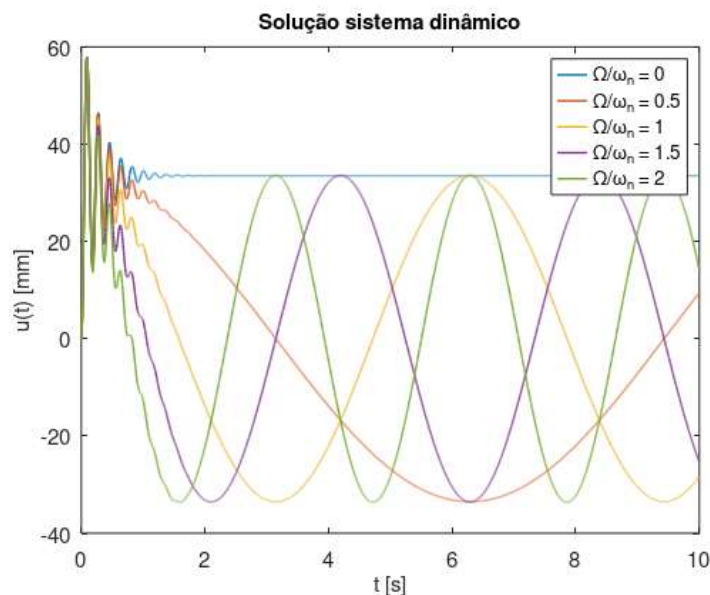


Figura 41 - Solução do sistema para diferentes frequências de forçamento – mola linear

A amplitude foi avaliada quanto à frequência de forçamento para três forças diferentes. Os resultados para a mola elástica estão apresentados na Figura 42, onde observa-se que quanto maior a força, maior a amplitude.

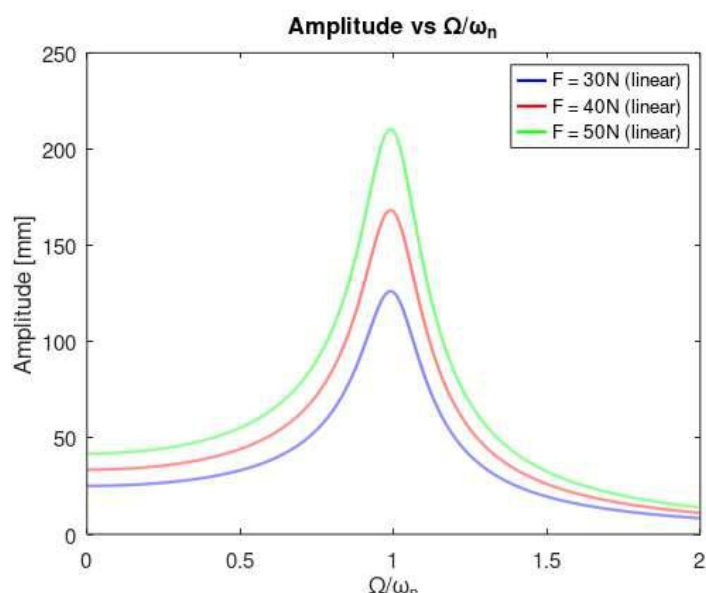


Figura 42 - Amplitude de acordo com Ω/ω_n

A ressonância nesse caso aconteceria no ponto em que a razão Ω/ω_n se aproxima de 1, como esperado de acordo com a Eq. (54). Pela equação, a ressonância aconteceria para razão $\Omega/\omega_n = \sqrt{1 - 2\xi^2} = 0,98995$.

4.5.2 Estudo do Passo

Mantendo a Frequência de forçamento em $\Omega = \omega_n$, o código foi testado para diferentes valores de passo, com o objetivo de alcançar o melhor passo – que tenha certa precisão, mas com menor demanda computacional possível. A Tabela 8 contém as informações de quanto tempo o código demorou para terminar de rodar com o passo h.

Tabela 8 - Tempo decorrido para o código gerar solução de uma frequência para cada passo

h	Tempo [min]
0,07	0,64
0,05	1,06
0,04	2,07
0,03	3,18
0,02	6,41
0,015	10,23
0,01	28,40
0,005	106,90

É possível observar que à medida que o passo diminui, o tempo aumenta. Por exemplo, uma pequena diminuição no passo de $h=0,01$ para $h=0,005$ resultou em um

aumento de mais de 78,5 minutos (1,3h) para chegar à solução. Por outro lado, para passos maiores essa diferença no tempo para solução não é tão grande, como na diminuição do passo de $h=0,05$ para $h=0,04$ que causou um aumento de apenas 1,01 minutos.

A força de 40N foi escolhida para comparar as Figura 43, Figura 44 e Figura 45, que representam a resposta do código para diferentes valores de passo, por ser a intermediária entre as três usadas. Na Figura 43 é possível observar que a solução não aparenta estar muito precisa.

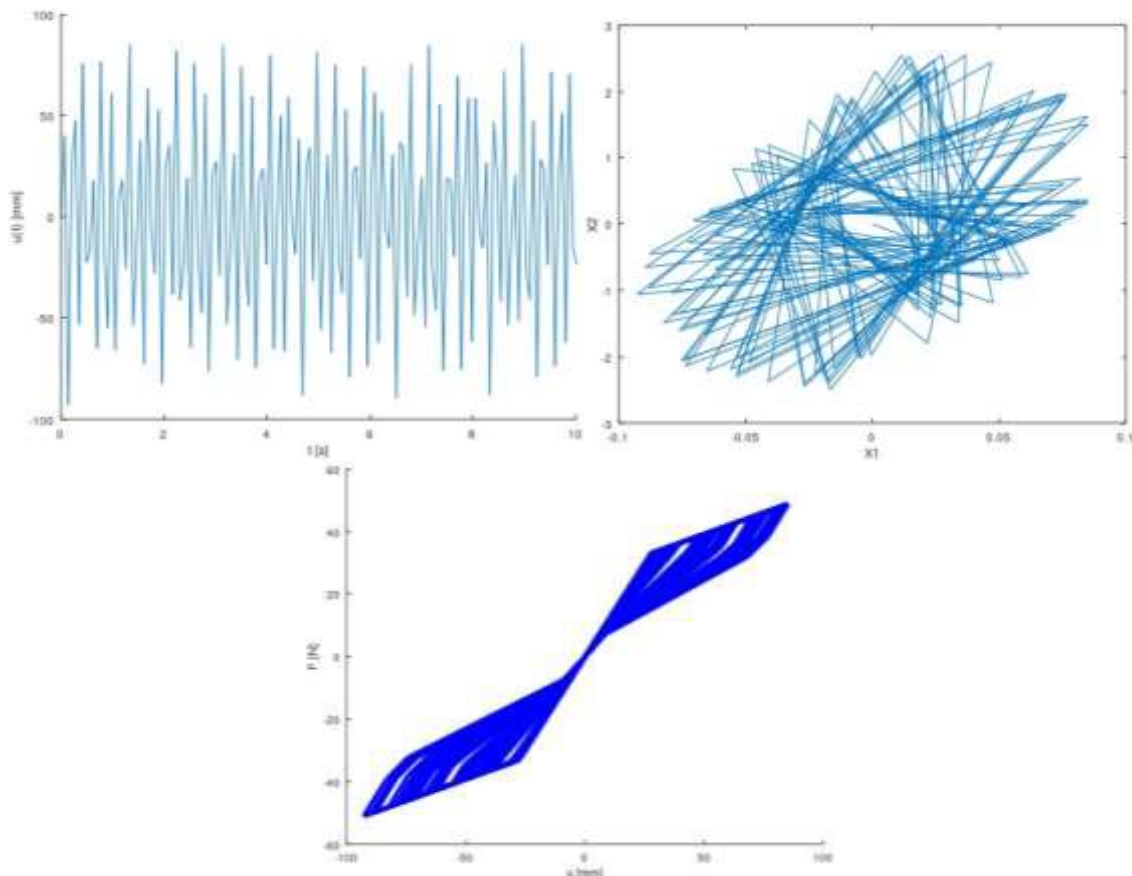


Figura 43 - Solução no tempo, Espaço de fase e Força-Deslocamento para $h = 0,07$ ($F=40N$)

Já na Figura 44, o espaço de fase se aproxima mais de uma espiral, o resultado esperado para esse caso de sistema massa-mola-amortecedor. Mas ainda se observa linhas um pouco acentuadas.

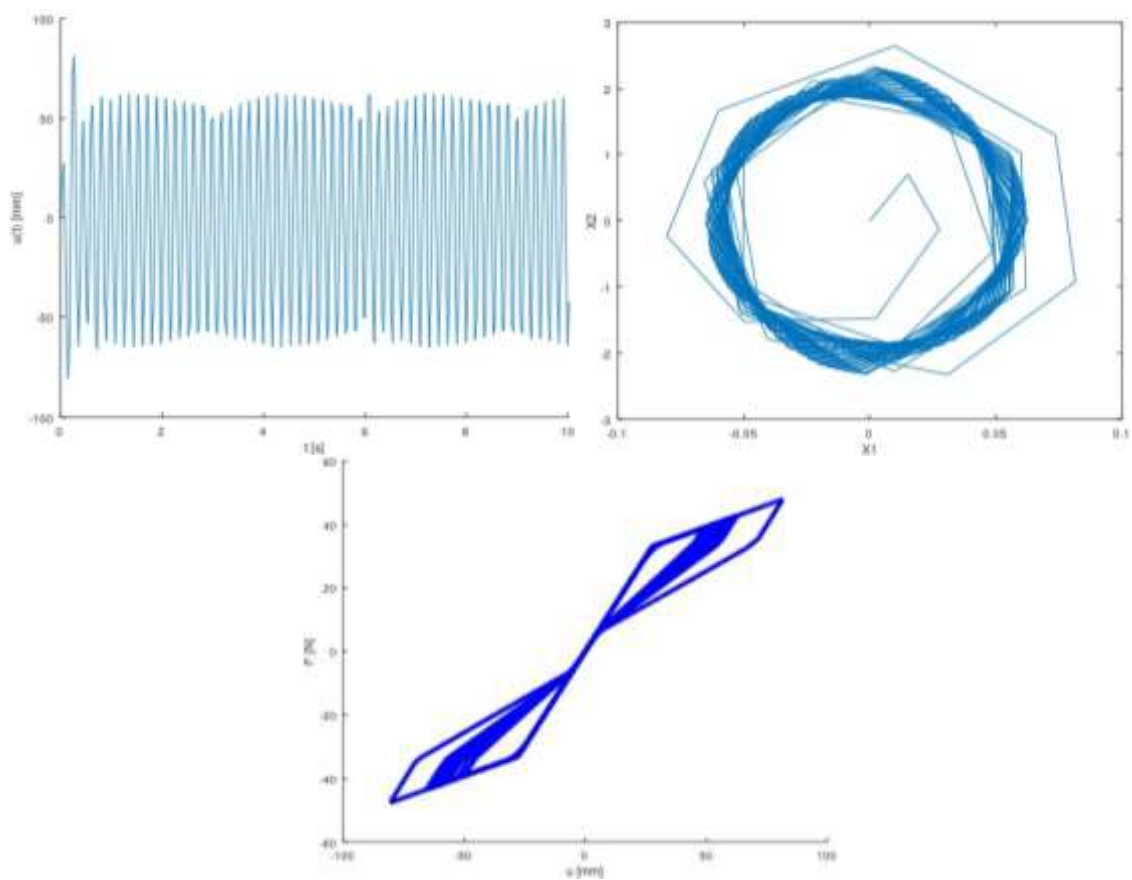


Figura 44 - Solução no tempo, Espaço de fase e Força-Deslocamento para $h = 0,03$ ($F=40N$)

A Figura 45 representa o passo mais preciso testado e exigiu a maior demanda computacional. Vê-se que as curvas no espaço de fase estão mais suaves, demonstrando que a convergência e precisão da solução.

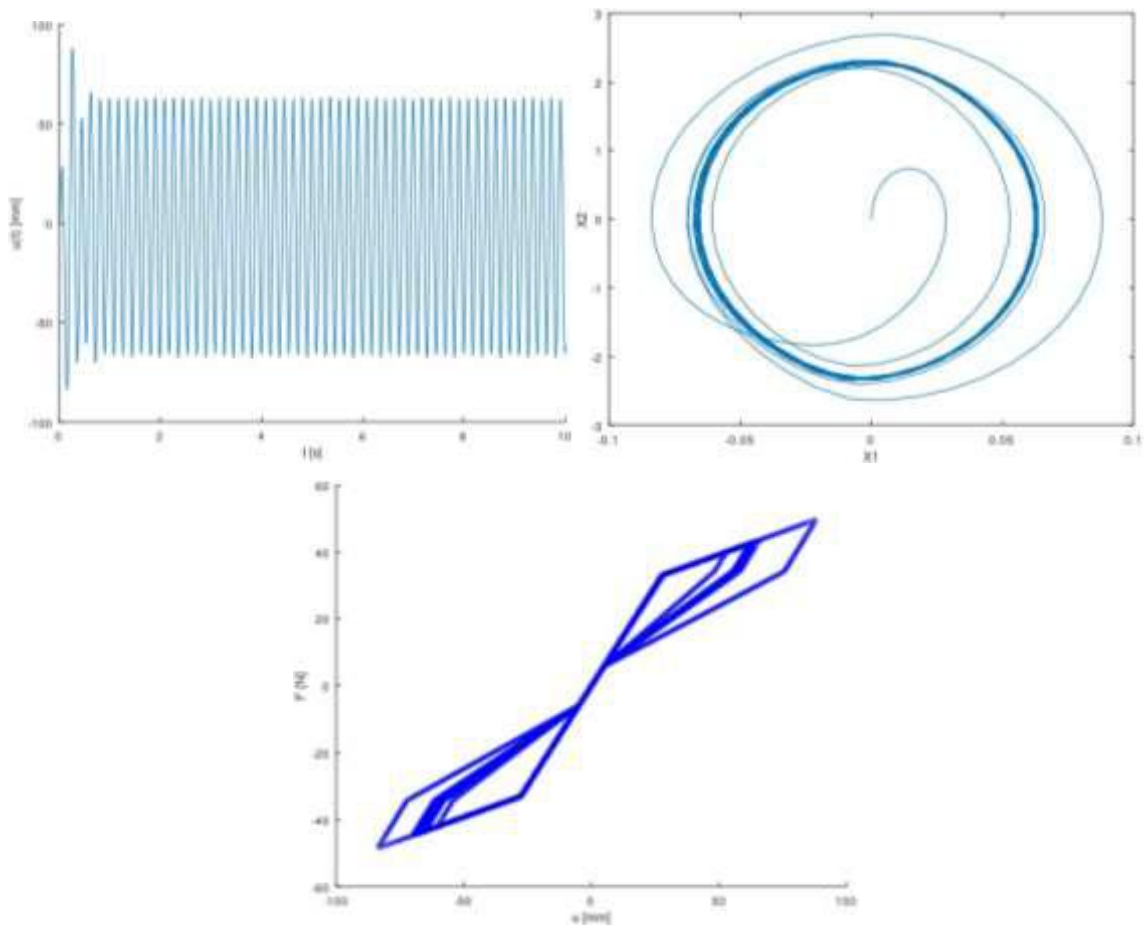


Figura 45 - Solução no tempo, Espaço de fase e Força-Deslocamento para $h = 0,005$ ($F=40N$)

Todos os tempos citados na Tabela 8 são referentes à solução para apenas uma frequência de forçamento. No estudo paramétrico serão analisadas para 10 valores de frequências de forçamento para cada força e o passo $h=0,005$ foi o escolhido porque apresenta maior precisão.

4.5.3 Comparação entre Molas

Sobrepondo a solução da mola linear e da mola de SMA que estão sobre as mesmas condições de força ($F=50 N$) e frequência de forçamento ($\Omega = \omega_n$) tem-se a Figura 46. Nota-se que o deslocamento da mola de SMA é maior do que da mola linear elástica.

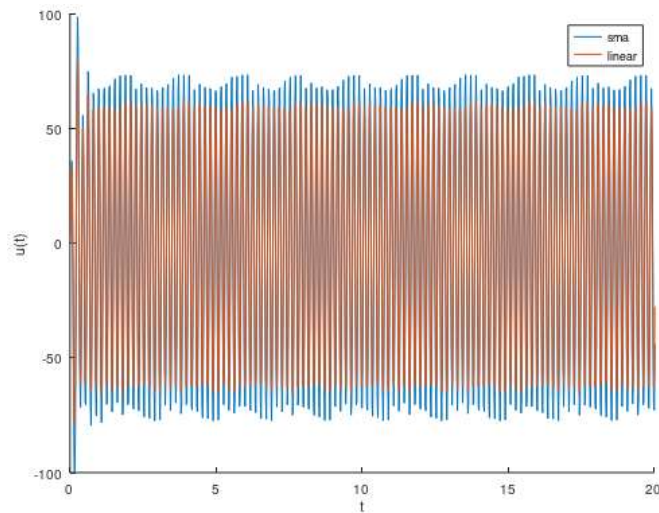


Figura 46 - Sobreposição de Solução para Mola de SMA e Mola Linear

Esse comportamento também pode ser observado na Figura 47, onde a mola alcança deslocamentos maiores com o mesmo nível de força aplicado. Nesse gráfico ainda se identifica a histerese formada, caracterizando a dissipação de energia.

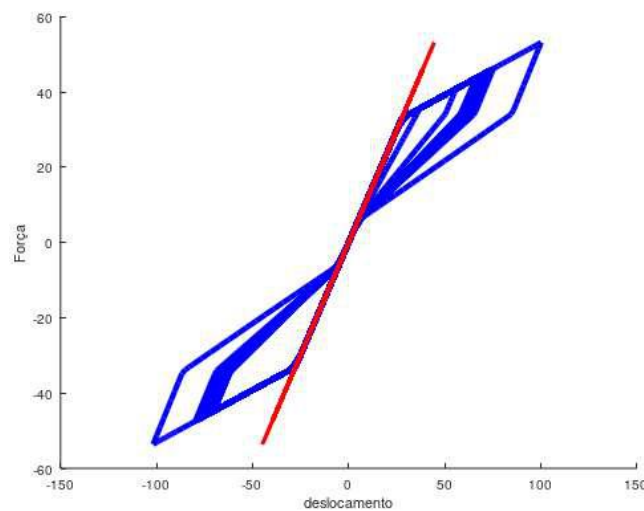


Figura 47 - Força-Deslocamento no sistema dinâmico

Após a aplicação de carga mecânica, inicia-se a transformação de fase próximo de 40N, chega-se até aproximadamente 100 mm de deslocamento e começa o descarregamento, que também passa por uma mudança de fase, voltando para o comportamento linear. Assim, a mola de SMA segue dissipando energia e diminuindo a área de histerese.

4.5.4 Estudo Paramétrico

No estudo paramétrico busca-se determinar o comportamento de ambas as molas para diferentes frequências de forçamento, através da análise da variação da amplitude em função da razão entre a frequência de forçamento e a frequência natural.

Para isso, foi implementado o código apresentado no Anexo C, utilizando-se um passo de $h=0,005$ s, calculando a solução para três forças - 30 N, 40 N e 50 N – e a razão entre as frequências variando de 0 a 2. A parte do código que descreve o modelo da Brinson Modificado foi baseado em Pinho, 2022. Na mola linear, são consideradas 200 frequências enquanto para a mola de SMA são calculadas para 10 frequências, para reduzir a demanda computacional. O tempo total que esse código demorou para rodar foi de aproximadamente 16 horas.

A Figura 48 mostra como o gráfico de Força-Deslocamento para a força de 30N varia com a razão entre as frequências. Vê-se que quanto maior a razão entre as frequências, menor é a diferença entre o comportamento da mola linear e a mola de SMA, pois a mola de SMA passa a se aproximar do comportamento linear, dissipando menos energia. Além disso, é vê-se que na Figura 48(b) ocorre a transformação de fase completa.

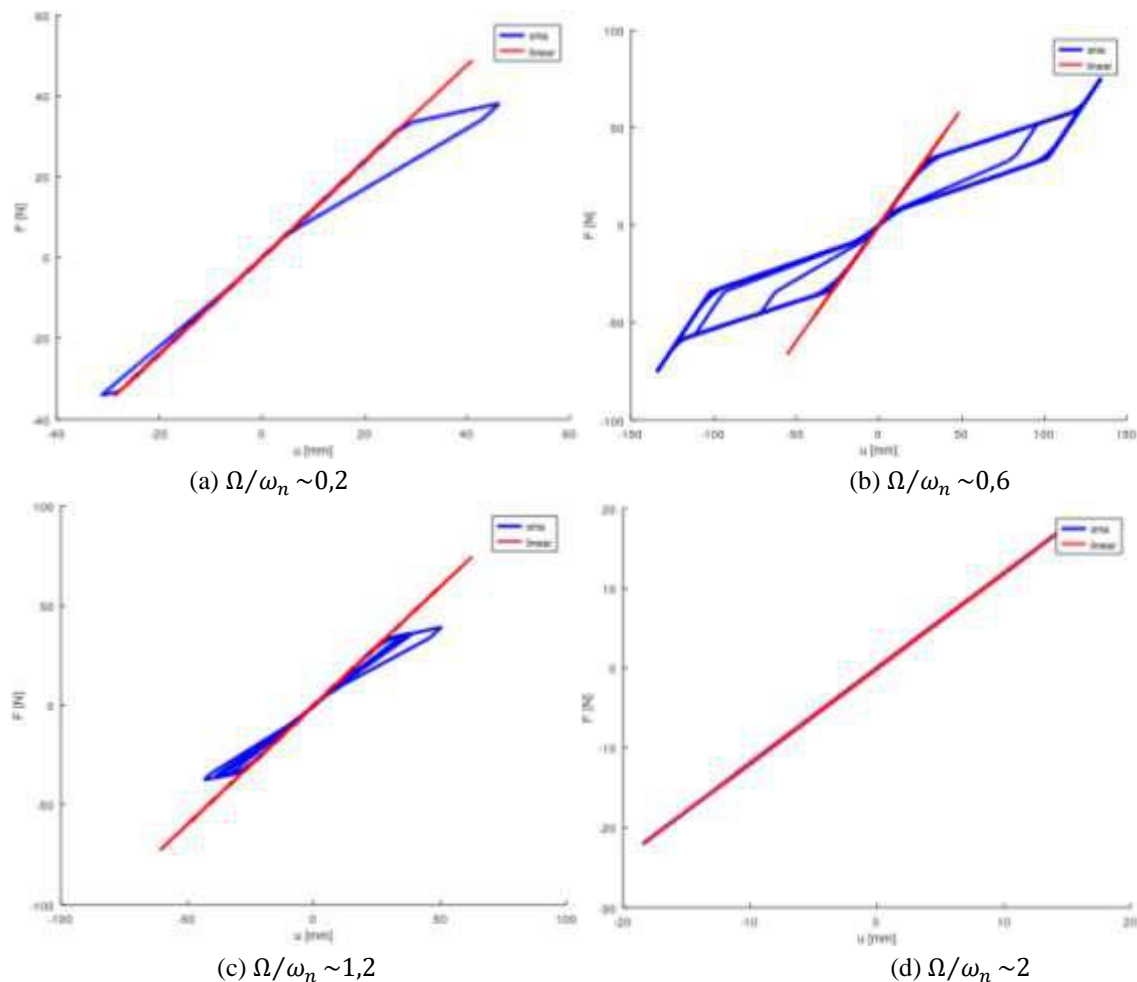


Figura 48 - Variação de Força-deslocamento com razão de frequências (a) $\Omega/\omega_n \sim 0,2$, (b) $\Omega/\omega_n \sim 0,6$, (c) $\Omega/\omega_n \sim 1,2$ e (d) $\Omega/\omega_n \sim 2$

Também pode-se analisar como a solução do deslocamento varia de acordo com alteração na frequência de forçamento. Essa comparação está representada na Figura 49, que ilustra essa diferença para quatro razões diferentes. Está sendo apresentado apenas para 1 das forças aplicadas pois o resultado é semelhante para as outras forças, alterando apenas a amplitude das soluções.

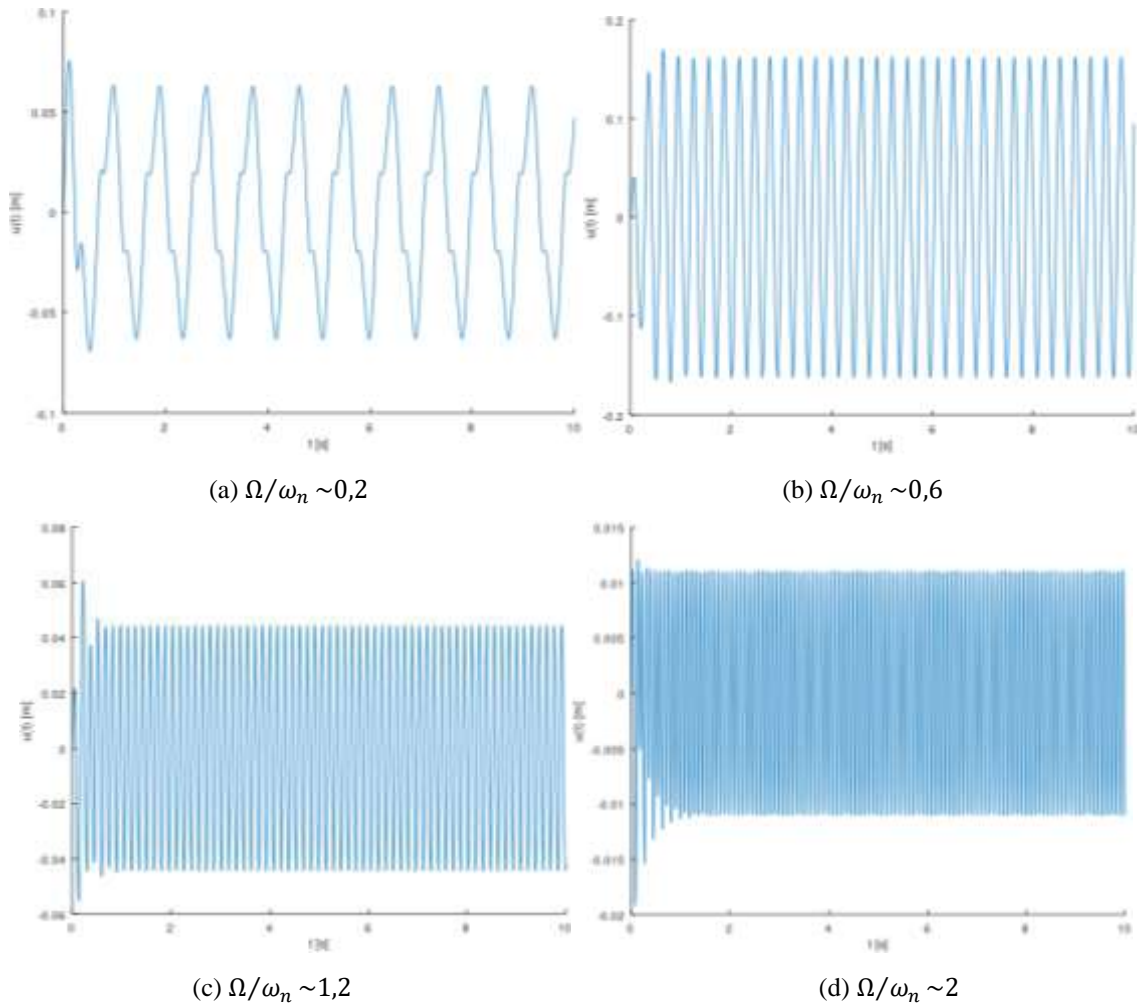


Figura 49 - Solução SMA, $F=40N$, (a) $\Omega/\omega_n \sim 0,2$, (b) $\Omega/\omega_n \sim 0,6$, (c) $\Omega/\omega_n \sim 1,2$ e (d) $\Omega/\omega_n \sim 2$

Nas soluções apresentadas, tem-se nos primeiros instantes de tempo a solução transiente (soma da solução homogênea e particular) e, com o passar do tempo, resta a solução permanente, somente com a solução particular.

A Figura 50 representa a maior amplitude da solução permanente para cada uma das razões entre as frequências. A mola linear tem sua frequência de ressonância como mostrado na validação, próximo do valor de 1. Então a amplitude cresce até chegar a esse valor e depois começa a decrescer.

À medida que a razão se aproxima de dois, a diferença entre as amplitudes da mola linear e da mola de SMA fica menor. Já a ressonância da mola de SMA acontece

para uma frequência menor, ficando entre de $X=0,7$ e $X=0,9$ – dependendo da força que está sendo aplicada – alcançando uma amplitude maior que a mola linear.

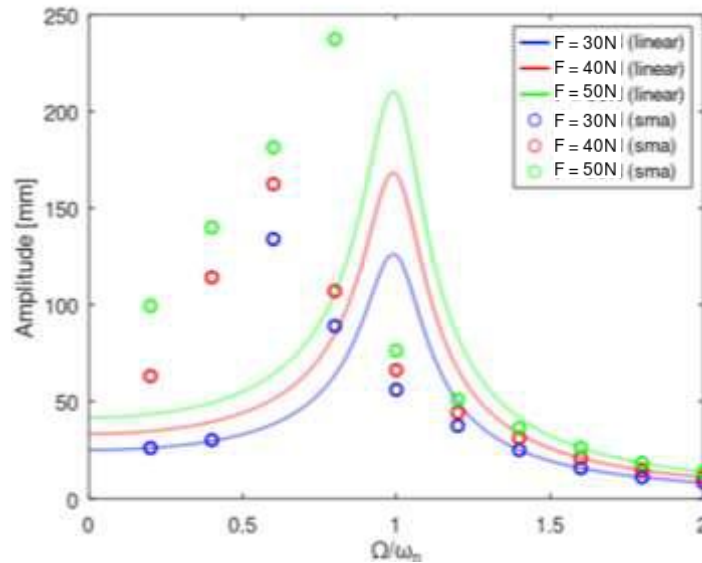


Figura 50 - Amplitudes pela razão das frequências de forçamento e natural.

Observa-se uma diferença entre o comportamento das molas de SMA e linear, onde a amplitude da mola de SMA dá um salto a partir de determinada frequência e, após, volta a coincidir com a mola linear.

Para avaliar melhor o que resulta esse comportamento, foi realizada nova simulação para forças menores e considerando 20 frequências de forçamento, para ter a solução para mais frequências e ser possível investigar em um intervalo com menor diferença entre as frequências o que pode ter influenciado esse resultado.

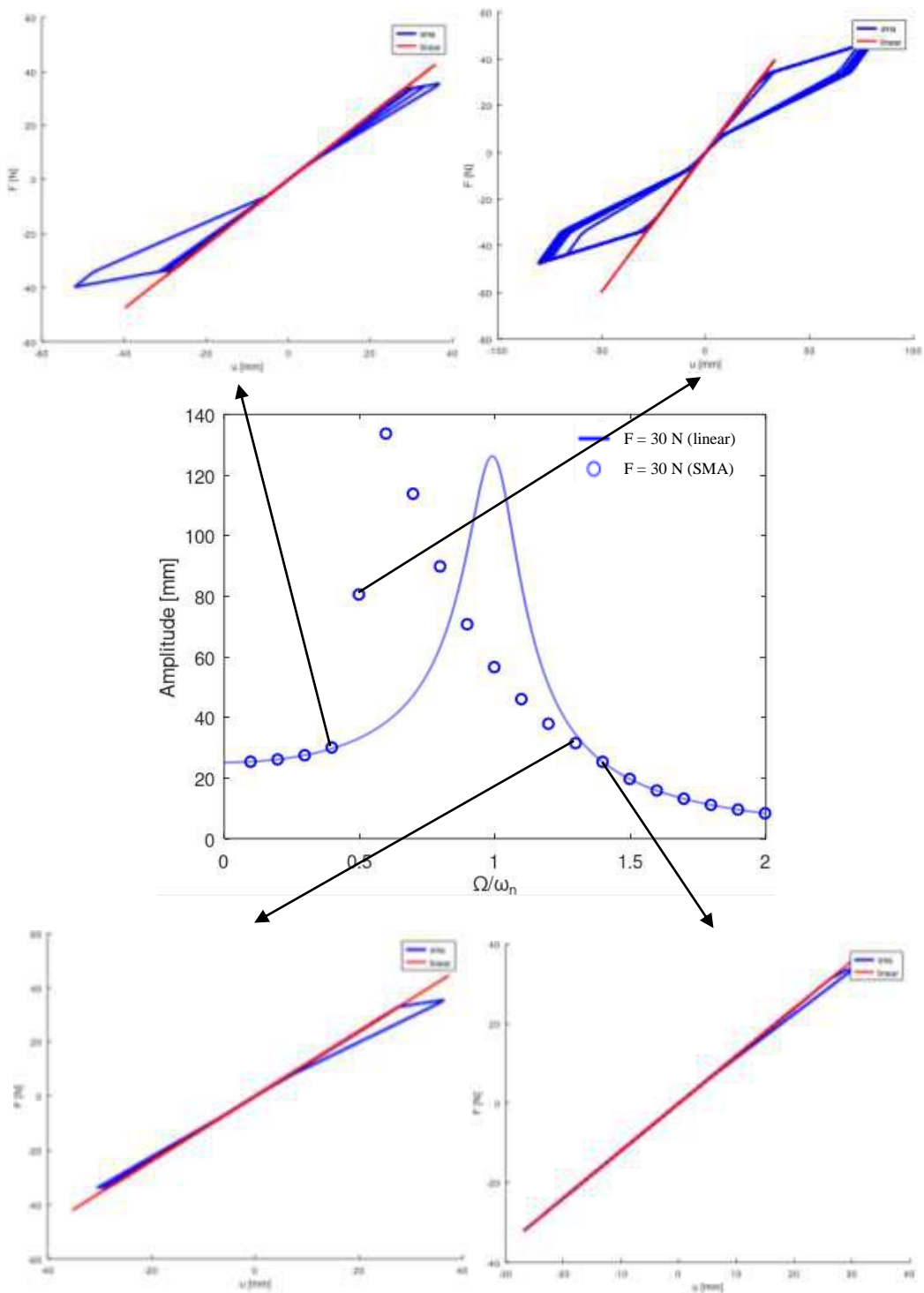


Figura 51 - Transformação de fase e amplitude pela razão entre frequências (F=30 N)

Analisando a Figura 51 nota-se que o salto na amplitude ocorre no momento em que se inicia a transformação de fase, como consequência da não-linearidade, o que está de acordo com Oliveira, 2014.

5 CONCLUSÕES

Por meio desse trabalho foi possível estudar o comportamento de uma mola helicoidal de liga com memória de forma através de sua modelagem e aplicação em um sistema dinâmico massa-mola-amortecedor, com forçamento harmônico. O *software* Ansys permitiu realizar a simulação computacional para o material inteligente, possibilitando simular o comportamento complexo através das curvas de força e deslocamento.

Os resultados das simulações computacionais puderam ser validados por dados experimentais retirados de Savi (2015), comprovando que o *software* comercial utilizado na versão de estudante foi eficiente para reproduzir as curvas características da pseudoelasticidade da liga Ni-Ti, com transformação de fase entre a austenita e a martensita para as condições apresentadas. As malhas ainda foram testadas e o tempo que cada uma levou para completar a simulação foi avaliado, permitindo comparar a demanda computacional de cada uma delas.

O modelo da Brinson modificado foi implementado no *software* Octave para reproduzir o comportamento da mola. O modelo constitutivo utilizado reproduziu bem o que se esperava, ainda sendo capaz de captar etapas intermediárias. É importante ressaltar a importância de se obter os pontos de transformação de fase para calibração do modelo. A partir disso, realizou-se a comparação entre os resultados numéricos (Ansys), analíticos (Brinson modificado) e experimentais (Savi, *et al*, 2015) para diferentes carregamentos.

Tendo comprovado a eficiência das simulações, a influência dos diâmetros da mola no comportamento da mola foi explorada, através da variação do índice da mola. Concluiu-se que na alteração do diâmetro da mola, quanto menor o índice da mola – e, conseqüentemente, o menor diâmetro da mola – maior é a área do gráfico, mas o comportamento se afasta da aproximação linear.

Sobre o impacto no coeficiente de rigidez da mola, à medida que o diâmetro da mola cresce, os coeficientes de rigidez da mola nas fases martensita e austenita diminuem, resultando em uma mola menos rígida.

Através da implementação de um Runge-Kutta de 4ª ordem com o modelo de Brinson modificado foi possível representar a aplicação da mola de SMA em um sistema dinâmico de massa-mola-amortecedor. A validação do RK foi feita através da implementação inicial para uma mola linear e, após constatar a boa funcionalidade, para

uma mola de SMA. Foi possível observar o impacto da diminuição do passo na demanda computacional para solução do sistema com uma frequência de forçamento.

No estudo paramétrico, viu-se que quanto maior a razão entre frequência de forçamento e natural, menor é a diferença entre o comportamento da mola linear e a mola de SMA. Para razões próximas de zero, o comportamento de ambas as molas também é semelhante dependendo do valor da força, pois para maiores forças, pode haver transformação de fase mesmo com frequência de forçamento nula.

Na análise das amplitudes pelas razões, percebe-se que para a mola linear, a amplitude aumenta a razão próximo de 1 e depois começa a diminuir, enquanto para a mola de SMA a ressonância acontece para diferentes valores da razão entre as frequências, onde constatou-se que essa alteração é devido a transformação de fase.

Como vantagens de aplicação do material inteligente, podem-se citar a dissipação de energia e capacidade de alcançar maiores deslocamentos sobre as mesmas condições que uma mola linear. Considerando mesma força e razão entre frequências de forçamento, a transformação de fase permite que o deslocamento da mola de SMA seja maior do que ao da mola linear, além da energia dissipada pela área de histerese.

De forma geral, através dos métodos numéricos e analíticos implementados em *softwares* em versões gratuitas foi possível realizar o estudo de um material de complexo comportamento termomecânico, obtendo resultados satisfatórios. Assim, constata-se a relevância dos mesmos para o desenvolvimento de soluções na engenharia, podendo ser usados para otimizar produtos, avaliar diferentes cenários, dentre outras aplicações, com baixo custo financeiro.

6 TRABALHOS FUTUROS

Como sugestão de trabalhos futuros sugere-se analisar a influência de outros parâmetros como diâmetro do fio e passo da mola na influência de seu comportamento. Além disso, verificar na modelagem na mola no *Ansys* a fração volumétrica durante o carregamento e o descarregamento, assim como realizar a análise de gráficos tensão x deformação poderia complementar as análises feitas nesse trabalho.

No sistema dinâmico seria interessante explorar a limitação do modelo da Brinson modificado utilizado em relação à representação da completa transformação de fase. A completa transformação não é possível ser alcançada em uma mola devido à força axial

que causa uma tensão de cisalhamento na sua seção transversal - que é mínima no interior do fio. Então, na coordenada em que o raio é zero (interior do fio), não ocorre a transformação de fase.

Outro aspecto a ser explorado seria em relação a limitações de aplicação do SMA, devido seu custo mais elevado; ainda podendo ser avaliada a fadiga funcional do material, considerando um número de ciclos muito alto onde o material tenderia a se comportar como linear.

Além desses tópicos, o efeito de memória de forma poderia ser explorado, avaliando também a influência de alterações de temperatura em mola de Ligas com Memória de Forma.

REFERÊNCIAS BIBLIOGRÁFICAS

- Abreu, A. L. E. L., 2021. “Análise Modal e Estática de um *Chassis Spaceframe* FSAE”. Projeto de Graduação, Universidade Federal do Rio de Janeiro, Brasil.
- An, S., et al, 2012. “*Engineering design framework for a shape memory alloy coil spring actuator using a static two-state model*”. *Paper , IOP Science, Smart Materials and Structures*. 21 055009 (16 pp).
- Bandeira, S. A., 2017. “Efeitos de tratamentos térmicos em molas pseudoelásticas de ligas com memória de forma”. Projeto de graduação, Universidade de Brasília, Brasil.
- Borgmann, I. C., 2012. “Análise por Elementos Finitos de um Carro Transportador de Plataforma de Corte”. Trabalho Final de Curso, Fator – Faculdade Horizontina, Curso de Engenharia Mecânica, Brasil.
- Budynas, R. G. e Nisbett, J. K., 2011. “Elementos de Máquinas de Shigley – Projeto de Engenharia Mecânica - 8ª edição”. Livro AMGH Editora Ltda. Obra originalmente publicada sob o título “*Shigley’s Mechanical Engineering Design, 8th edition in SI Units*”.
- Callister, W. D., 2007. “Ciência e Engenharia de Materiais: Uma introdução – 7ª edição”. Livros Técnicos e Científicos Editora S.A., uma editora integrante do GEN (grupo Editorial Nacional). Obra originalmente publicada sob o título “*Materials Science and Engineering: na introduction – Seventh Edition*”.
- Coura, E. A. N., 2016. “Estudo da Aplicação da Liga NiTi com Efeito Memória de Forma em Atuadores de Aeronaves”. Tese de mestrado, Escola de Engenharia, Universidade Federal de Minas Gerais, Brasil.
- Fernandes, G. F., 2014. “Construção e caracterização de mola helicoidal com memória de forma aplicada em um atuador linear termoativado”. Projeto de graduação, Escola de Engenharia de Lorena, Universidade de São Paulo, Brasil.
- Garcia, M. S., 2015. “Análise experimental do comportamento termomecânico das ligas com memória de forma”. Tese de doutorado, COPPE, Universidade Federal do Rio de Janeiro, Brasil.
- Giracca, C. N., 2021. “Gestão e tecnologia Órtese de fibra de carbono com base nas pessoas, projetos, processos e procedimentos”. Dissertação de mestrado, Centro de Comunicação e Expressão, Universidade Federal de Santa Catarina, Brasil.
- Júnior, W. L. P., et al, 2012. “Avaliação de Absorvedores Dinâmicos de Vibrações Sintonizáveis Baseados em Ligas com Memória de Forma”. CONEM 2012. VII Congresso Nacional de Engenharia Mecânica. São Luís, Maranhão, Brasil.
- Monteiro, L. L., 2012. “Desenvolvimento De Um Modelo Térmico Mecânico De Uma Mola Helicoidal De Ni-Ti Com Efeito Memória De Forma Para Aplicação Numa Máquina De Calor”. Tese de doutorado, Universidade Federal de Pernambuco, Brasil.
- Mtili, et al, 2021. “*Force-displacement relationships for NiTi alloy helical springs by using ANSYS: Superelasticity and shape memory effect*”. *International Review of Applied Sciences and Engineering*. DOI: [10.1556/1848.2021.00389](https://doi.org/10.1556/1848.2021.00389).
- Norton, R. L., 2013. “Projeto de Máquinas – Uma abordagem integrada – 4ª edição”. Bookman Editora Ltda. Obra originalmente publicada sob o título “*Machine Design, 4th edition*”.
- Paiva, A., et al, 2004. “*A constitutive model for shape memory alloys considering tensile-compressive asymetry and plasticity*”. *International Journal of Solids and Structures* 42 (2005) 3439–3457.
- Pinho, J. F., 2022. “Implementação do Modelo de Brinson Modificado para Ligas com Memória de Forma”. Projeto de Graduação, Universidade Federal do Rio de Janeiro, Brasil.
- Savi, M.A. e Paula, A.S., 2017. “Vibrações Mecânicas”, LTC.
- Savi, et al, 2015. “*Nonlinear geometric influence on the mechanical behavior of shape memory alloy helical springs*”. *Paper, IOP Science, Smart Materials and Structures*. 24 035012 (13 pp).
- Shaw, J. A. e Kyriakides, S., 1995. “*Thermomechanical aspects od NiTi*”. *Journal of the Mechanics and Physics of Solids*, Vol. 43, No. 8, pp. 1243-1281.
- Silva, I. A., 2016. “Concepção e Projeto de Uma Bancada Para Caracterização de Fadiga Funcional Em Ligas Com Memória De Forma”. Projeto de graduação, Universidade de Brasília, Brasil.
- Silva, R. O., 2017. “Atenuação de Vibrações em Sistemas que utilizam Molas de Liga de Memória de Forma”. Tese de Mestrado, Faculdade de Engenharia, Universidade Estadual Paulista Julio de Mesquita Filho, Brasil.
- Silva, R. S., 2016. “Análise De Estruturas Inteligentes Com Memória De Forma Utilizando O Método Dos Elementos Finitos”. Projeto de graduação, Escola Politécnica, Universidade Federal do Rio de Janeiro, Brasil.
- Sterza, R. L., 2016. “Comparação entre métodos numéricos: Runge-Kutta de quarta ordem e previsor corretor”. Revista Eletrônica Paulista de Matemática. ISSN 2316-9664. Volume 7, Edição ERMAC.
- Tobushi, H. e Tanaka, K., 1991. “*Deformation of a Shape Memory Alloy Helical Spring: Analysis Based on Stress-Strain-Temperature Relation*”. *JSME International Journal. Series I*, Vol. 34, No. 1. Pages 83-89. DOI: https://doi.org/10.1299/jsmea1988.34.1_83.

- Oliveira, S. A. e Savi, M. A., 2013. “Os materiais inteligentes e suas aplicações”. Revista Marítima Brasileira v. 133 n.10/12. P. 39. Marinha do Brasil.
- Oliveira, H. S., et al, 2014. “*Dynamical Jumps in a Shape Memory Alloy Oscillator*”. Hindawi Publishing Corporation. *Shock and Vibration*. Volume 2014, Article ID 656212, 10 pages. DOI: <http://dx.doi.org/10.1155/2014/656212>.

ANEXO A

Modelo Constitutivo

```
1 function SMApolii
2
3 cla
4 close all
5 clear all
6
7 %Transformação de Unidades
8 GPa = 10^9;
9 MPa = 10^6;
10 mm = 10^-3;
11
12 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
13 %DADOS EXPERIMENTAIS - BAVI, RETIRADOS COM GET DATA
14
15 mola_exp120 = [0 0
16 0.3037 1
17 0.9111 2.5
18 2.7332 5.25
19 5.1627 8.75
20 6.9848 10.25
21 8.4183 13.5
22 12.4512 16.75
23 14.8807 19.75
24 17.3102 22.75
25 21.5618 26.75
26 23.3839 28.25
27 25.2061 30
28 28.5466 31.75
29 31.2798 33.5
30 35.2278 35.25
31 39.4784 36.75
32 44.6421 39
33 48.6937 40.25
34 54.6638 41.75
35 59.8265 43
36 64.0781 44.5
37 68.9371 45.75
38 73.4924 47
39 78.6551 48
40 82.9067 49.25
41 89.2842 50.75
42 95.0542 52.25
43 101.7350 54
44 108.4160 55.75
45 113.8830 57.75
46 119.0460 59
47 119.6530 59.25
48 118.4380 56
49 116.9200 53.25
50 114.7940 50.75
51 112.9720 48.25
52 110.8460 46
53 109.0240 43.75
54 106.5940 41.25
55 104.7720 39.25
56 102.0380 37.5
57 98.6985 35.25
58 94.4469 32.75
59 92.0174 31.25
60 86.5510 28.5
61 82.2983 27
62 78.0477 25.75
63 73.1887 24.25
64 68.6304 22.75
65 64.0781 21.75
66 58.6117 20.75
67 52.2343 19.5
68 48.2863 18.75
69 43.4273 17.75
70 38.5683 17
71 33.4056 16
72 29.4577 15.25
73 24.5987 14.25
74 20.3471 13.25
75 17.3102 12.75
```

```

76 13.9898 11.75
77 10.6291 10.75
78 7.8959 9.25
79 6.0738 8
80 4.8590 7.25
81 3.3406 6.25
82 2.1258 4.25
83 0.8074 1.75
84 0 0];
85
86 *****
87 %DADOS NUMÉRICOS - ANSYS
88
89 mola_num120 = [0 0
90 12.01 17.163
91 17.414 24.648;
92 22.818 29.102
93 30.925 32.741
94 39.031 35.15
95 51.041 38.356
96 63.048 41.647
97 68.453 43.243
98 73.654 44.922
99 81.959 47.628
100 93.964 52.143
101 105.97 57.381
102 113.02 60.913
103 120.08 64.867
104 117.38 57.745
105 115.35 52.56
106 113.87 48.881
107 112.69 46.355
108 111.56 44.011
109 110.42 41.992
110 109.28 40.642
111 108.43 39.844
112 107.57 39.167
113 106.29 38.388
114 104.37 37.421
115 101.99 36.144;
116 97.16 34.337
117 90.67 31.739
118 80.933 28.18
119 76.551 26.69
120 72.169 25.261
121 65.596 23.219
122 55.735 20.352
123 49.079 18.53
124 42.423 16.786
125 32.438 14.291
126 25.697 12.676
127 18.957 11.101
128 8.8459 8.6731
129 4.8651 6.7235
130 2.4326 3.4799
131 1.24E-03 4.41E-02];
132
133 *****
134 %Calibração - Pontos 1, 2 e 3
135
136 f1 = 33.0838;
137 f2 = 59.1617;
138 f3 = 34.2517;
139 u1 = 27.7138*mm;
140 u2 = 119.804*mm;
141 u3 = 100.173*mm;
142
143 k_A = f1/u1
144 k_M = (f2-f3)/(u2-u3)
145
146 e1 = f1
147 e2 = f2
148 e3 = f3
149
150 u_r = 75*mm; % ajuste

```

```

151
152 ~~~~~
153
154 As = 16.7+273;
155 Af = 30.2+273;
156 Ma = 24.8+273;
157 Mf = 9.1+273;
158 T = 33+273;
159
160 C_M = 5;
161 sigma_cr_s = f1 - C_M*(T-Ms);
162 sigma_cr_f = f2 - C_M*(T-Ms);
163 C_A = s37*(T-As);
164 mat_prop = [u_r k_A k_M C_A C_M sigma_cr_s sigma_cr_f As Af Ms Mf T];
165
166 ~~~~~
167 %
168 sigma0 = 0;
169 epsilon0 = 0;
170 beta0 = 0;
171 %
172 previous_state = [sigma0 epsilon0 beta0];
173 results = previous_state;
174 %
175 dir = 1;
176 %
177 sig_cos = 0;
178 eps_cos = 0;
179 bet_cos = 0;
180
181 div = 100;
182
183 u = [0 0.12 0];
184
185 teste120 = Brinson_poli_uF_mola(u,mat_prop,previous_state,div);
186 F_poli_t120 = teste120(:,1);
187 u_poli_t120 = teste120(:,2);
188
189 figure
190 plot(u_poli_t120*(10^3),F_poli_t120,'b','linewidth',2) %Soluçao octave
191
192 ~~~~~
193
194 figure
195 hold on
196 plot(u_poli_t120*(10^3),F_poli_t120,'b','linewidth',2) %Soluçao octave
197 plot(mola_exp120(:,1),mola_exp120(:,2),'ok','MarkerEdgeColor','k','MarkerFaceColor','k','MarkerSize',5,'color','k') %Soluçao Savi
198 plot(mola_num120(:,1),mola_num120(:,2),'r-','linestyle','--','linewidth',2) %Soluçao ansys
199 hold off
200 box on
201 title('u [mm]=120mm','FontSize',20)
202 xlabel('u [mm]','FontSize',20)
203 ylabel('F [N]','FontSize',20)
204 leg = legend('analytical','experimental','numeric');
205 set(leg,'FontSize',25)
206 set(leg,'Location','southeast');
207 set(gcf,'color','white')
208 set(gca,'FontSize',18)
209 ylim([0 45])
210 xlim([0 125])
211
212 end
213
214 function Brinson_poli_strain = Brinson_poli_strain(mat_prop,previous_state,epsilon,dir)
215
216 epsilon_r = mat_prop(1);
217 E_A = mat_prop(2);
218 E_M = mat_prop(3);
219 C_A = mat_prop(4);
220 C_M = mat_prop(5);
221 sigma_cr_s = mat_prop(6);
222 sigma_cr_f = mat_prop(7);
223 As = mat_prop(8);
224 Af = mat_prop(9);

```

```

225 Ms = mat_prop(10);
226 Mf = mat_prop(11);
227 T = mat_prop(12);
228
229 a_M = pi/(Ms-Mf);
230 a_A = pi/(Af-As);
231
232 sigma0 = previous_state(1);
233 epsilon0 = previous_state(2);
234 beta0 = previous_state(3);
235 E0 = E_A + beta0*(E_M-E_A);
236
237
238 sigma1 = (sigma_cr_s+C_M*(T-Ms));
239 sigma2 = (sigma_cr_f+C_M*(T-Ms));
240 sigma3 = (C_A*(T-Af));
241 sigma4 = (C_A*(T-As));
242
243
244 epsilon1_min = (sigma1)/E_A;
245 if epsilon<epsilon1_min
246     epsilon1 = epsilon1_min;
247 else
248     epsilon1 = ((sigma1-sigma0) + E0*epsilon0)/E0;
249 end
250 epsilon2 = (sigma2 + epsilon_r*E_M)/E_M;
251
252
253 epsilon3 = ((sigma3-sigma2) + E_M*epsilon2 + epsilon_r*(-E_M))/E_A;
254
255 epsilon4_max = ((sigma4-sigma2) + E_M*epsilon2)/E_M;
256 if epsilon>epsilon4_max
257     epsilon4 = epsilon4_max;
258 else
259     epsilon4 = ((sigma4-sigma0) + E0*epsilon0)/E0;
260 end
261
262 if dir==1
263     if epsilon>epsilon1
264         if epsilon<epsilon2
265             a1 = 1 + ((1-beta0)/(sigma_cr_f-sigma_cr_s))*(-sigma_cr_f-C_M*(T-Ms));
266             a2 = ((1-beta0)/(sigma_cr_f-sigma_cr_s));
267             %
268             a = epsilon_r*(E_M-E_A)*(a2^2);
269             b = 1 + (epsilon_r*E_A-(E_M-E_A)*epsilon)*a2 + epsilon_r*(E_M-E_A)*2*a1*a2;
270             c = (epsilon_r*E_A-(E_M-E_A)*epsilon)*a1 + epsilon_r*(E_M-E_A)*(a1^2) +
                (-sigma0+(-E_A*epsilon+E0*epsilon0)-epsilon_r*E0*beta0);
271             %
272             delta = (b^2)-4*a*c;
273             sigma = (-b+sqrt(delta))/(2*a);
274
275             beta = 1 + ((1-beta0)/(sigma_cr_f-sigma_cr_s))*(sigma-sigma_cr_f-C_M*(T-Ms));
276             E = E_A + beta*(E_M-E_A);
277             if sigma<sigma0
278                 [epsilon0 epsilon_r]
279                 [sigma0 sigma (-b+sqrt(delta))/(2*a)]
280                 (sigma0 + E0*(epsilon-epsilon0) -
                epsilon_r*E0*(a1-beta0))/(1+epsilon_r*E0*a2)
                [beta0 beta]
                [E0 E]
281
282
283
284                 sigma0
285                 E*epsilon - E0*epsilon0
286                 epsilon_r*(E*beta-E0*beta0)
287                 epsilon
288                 sigma
289                 sigma0 + (E*epsilon - E0*epsilon0) - epsilon_r*(E*beta-E0*beta0)
290             endif
291         else
292             beta = 1;
293             E = E_A + beta*(E_M-E_A);
294             sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
295         end
296     end
297

```

```

298     else
299         beta = beta0;
300         E = E_A + beta*(E_M-E_A);
301         sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
302
303     end
304 elseif dir==1
305     if epsilon>epsilon3
306         if beta0>0 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
307             if epsilon<epsilon4
308                 a1 = (beta0/(Af-As))*(Af-T);
309                 a2 = (beta0/(Af-As))*(1/C_A);
310                 %
311                 a = epsilon_r*(E_M-E_A)*(a2^2);
312                 b = 1 + (epsilon_r*(E_A-(E_M-E_A)*epsilon)*a2 + epsilon_r*(E_M-E_A)*a1*a2;
313                 c = (epsilon_r*(E_A-(E_M-E_A)*epsilon)*a1 + epsilon_r*(E_M-E_A)*(a1^2) + %
314                     (-sigma0+(-E_A*epsilon+E0*epsilon0)-epsilon_r*(E0*beta0));
315                 delta = (b^2)-4*a*c;
316                 sigma = (-b+sqrt(delta))/(2*a);
317                 beta = (beta0/(Af-As))*((Af-T)*(sigma/C_A));
318             else
319                 beta = beta0;
320                 E = E_A + beta*(E_M-E_A);
321                 sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
322             end
323         else
324             beta = 0;
325             E = E_A + beta*(E_M-E_A);
326             sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
327         end
328     else
329         beta = 0;
330         E = E_A + beta*(E_M-E_A);
331         sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
332     end
333 end
334 end
335
336 Brinson_poli_strain = [sigma epsilon beta];
337 end
338
339
340 function Brinson_poli_deslocamento_mola = %
341     Brinson_poli_uF_mola(U,mat_prop,previous_state,div) %
342
343 L = 1;
344 A = 1;
345 EPSILON = U/L;
346
347 resultado_poli_d = feval(@Brinson_poli_epsilon_load,mat_prop,previous_state,EPSILON,div);
348 sig_poli_d = resultado_poli_d(:,1)*(10^6);
349 eps_poli_d = resultado_poli_d(:,2)/100;
350 bet_poli_d = resultado_poli_d(:,3);
351
352 F_poli = sig_poli_d*A;
353 u_poli = eps_poli_d*L;
354
355
356 Brinson_poli_deslocamento_mola = [F_poli u_poli bet_poli_d];
357
358 end
359
360
361 function Brinson_poli_epsilon_load = %
362     Brinson_poli_epsilon_load(mat_prop,previous_state,EPSILON,div) %
363
364 MPa = 10^6;
365
366 [linha,coluna] = size(EPSILON);
367 if linha>coluna
368     I = linha;
369 else
370     I=coluna;

```

```

370 end
371 %
372 %
373 dir = 1;
374 %
375 sig_poli = 0;
376 eps_poli = 0;
377 bet_poli = 0;
378 results = previous_state;
379
380 i = 1;
381 j = 1;
382 epsilon = EPSILON(i);
383 while i<I
384     i = i+1;
385     EPSILON(i)
386     if 1>0
387         EPSILON(i)/EPSILON(i-1)
388     end
389     if EPSILON(i)>=0
390         if EPSILON(i)>EPSILON(i-1)
391             if EPSILON(i-1)<0
392                 dir = -1;
393                 sigma0 = results(1);
394                 epsilon0 = results(2);
395                 beta0 = results(3);
396                 previous_state = [sigma0 epsilon0 beta0];
397                 depsilon = (0-EPSILON(i-1))/(div);
398                 while epsilon<0
399                     j = j+1;
400                     epsilon = epsilon+depsilon;
401                     results =
402                         feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
403                     sig_poli(j) = -results(1)/MPa;
404                     eps_poli(j) = -results(2)*100;
405                     bet_poli(j) = -results(3);
406                 end
407                 dir = 1;
408                 sigma0 = 0;
409                 epsilon0 = 0;
410                 beta0 = 0;
411                 previous_state = [sigma0 epsilon0 beta0];
412                 depsilon = (EPSILON(i))/(div);
413                 while epsilon<EPSILON(i)
414                     j = j+1;
415                     epsilon = epsilon+depsilon;
416                     results =
417                         feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
418                     sig_poli(j) = results(1)/MPa;
419                     eps_poli(j) = results(2)*100;
420                     bet_poli(j) = results(3);
421                 end
422             else
423                 if dir==1
424                     dir=-1;
425                     sigma0 = results(1);
426                     epsilon0 = results(2);
427                     beta0 = results(3);
428                     previous_state = [sigma0 epsilon0 beta0];
429                 end
430                 depsilon = (EPSILON(i)-EPSILON(i-1))/div;
431                 while epsilon<EPSILON(i)
432                     j = j+1;
433                     epsilon = epsilon+depsilon;
434                     results =
435                         feval(@Brinson_poli_strain,mat_prop,previous_state,epsilon,dir);
436                     results(1)/MPa;
437                     sig_poli(j) = results(1)/MPa;
438                     eps_poli(j) = results(2)*100;
439                     bet_poli(j) = results(3);
440                 end
441             end
442         else
443             if dir==1
444                 dir=-1;

```

```

442         sigma0 = results(1);
443         epsilon0 = results(2);
444         beta0 = results(3);
445         previous_state = [sigma0 epsilon0 beta0];
446     end
447     depsilon = (EPSILON(i)-EPSILON(i-1))/div;
448     while epsilon>1.001*EPSILON(i)
449         j = j+1;
450         epsilon = epsilon+depsilon;
451         results = feval(@Brinson_poli_strain,mat_prop,previous_state,epsilon,dir);
452         sig_poli(j) = results(1)/MPa;
453         eps_poli(j) = results(2)*100;
454         bet_poli(j) = results(3);
455     end
456 end
457 else
458     if EPSILON(i-1)>0
459         if dir==1
460             dir = -1;
461             sigma0 = results(1);
462             epsilon0 = results(2);
463             beta0 = results(3);
464             previous_state = [sigma0 epsilon0 beta0];
465         end
466
467         depsilon = {0-EPSILON(i-1)}/(div);
468         while epsilon>0
469             j = j+1;
470             epsilon = epsilon+depsilon;
471             results = feval(@Brinson_poli_strain,mat_prop,previous_state,epsilon,dir);
472             sig_poli(j) = results(1)/MPa;
473             eps_poli(j) = results(2)*100;
474             bet_poli(j) = results(3);
475         end
476         dir = 1;
477         sigma0 = 0;
478         epsilon0 = 0;
479         beta0 = 0;
480         previous_state = [sigma0 epsilon0 beta0];
481         depsilon = (EPSILON(i))/div;
482         while epsilon>EPSILON(i)
483             j = j+1;
484             epsilon = epsilon+depsilon;
485             results =
486                 feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
487             sig_poli(j) = -results(1)/MPa;
488             eps_poli(j) = -results(2)*100;
489             bet_poli(j) = -results(3);
490         end
491     else
492         if EPSILON(i)<=-EPSILON(i-1)
493             if dir==1
494                 depsilon = (EPSILON(i)-EPSILON(i-1))/div;
495                 if depsilon==0
496                     results = previous_state;
497                     sig_poli(j) = -results(1)/MPa;
498                     eps_poli(j) = -results(2)*100;
499                     bet_poli(j) = -results(3);
500                 endif
501                 while epsilon>EPSILON(i)
502                     j = j+1;
503                     epsilon = epsilon+depsilon;
504                     results =
505                         feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
506                     sig_poli(j) = -results(1)/MPa;
507                     eps_poli(j) = -results(2)*100;
508                     bet_poli(j) = -results(3);
509                 end
510             else
511                 dir = 1;
512                 sigma0 = abs(results(1));
513                 epsilon0 = abs(results(2));

```



```

514         beta0 = abs(results(3));
515         previous_state = [sigma0 epsilon0 beta0];
516         depsilon = (EPSILON(i)-EPSILON(i-1))/div;
517         if depsilon==0
518             results = previous_state;
519             sig_poli(j) = -results(1)/MPa;
520             eps_poli(j) = -results(2)*100;
521             bet_poli(j) = -results(3);
522         else
523             while epsilon>EPSILON(i)
524                 j = j+1;
525                 epsilon = epsilon+depsilon;
526                 results =
527                     feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),
528                         dir);
529                 sig_poli(j) = -results(1)/MPa;
530                 eps_poli(j) = -results(2)*100;
531                 bet_poli(j) = -results(3);
532             end
533         end
534     else
535         if dir==)
536             dir = -1;
537             sigma0 = abs(results(1));
538             epsilon0 = abs(results(2));
539             beta0 = abs(results(3));
540             previous_state = [sigma0 epsilon0 beta0];
541         end
542         depsilon = (EPSILON(i)-EPSILON(i-1))/div;
543         while epsilon<EPSILON(i)
544             j = j+1;
545             epsilon = epsilon+depsilon;
546             results =
547                 feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
548             sig_poli(j) = -results(1)/MPa;
549             eps_poli(j) = -results(2)*100;
550             bet_poli(j) = -results(3);
551         end
552     end
553 end
554 end
555 Brinson_poli_epsilon_load = [sig_poli' eps_poli' bet_poli'];
556
557 end
558 ~~~~~

```

ANEXO B

Validação Runge-Kutta – mola linear – Solução numérica x analítica

```

1  function RK
2
3      clc
4      close all
5      clear all
6
7      %Condições iniciais
8      %Linear
9      x1 = 0;           % Posicao inicial
10     x2 = 0;          % Velocidade inicial (x2 = u.)
11
12     %Entrada
13     m = 1;           % massa [Kg] - deixar por unidade de massa
14     k = 1193.77;     % Coef. Eig. [N/m] - Austenita
15     E = 0.1;        %csi
16     wn = sqrt(k/m)   %frequência natural
17     c = 2*m*E*wn;    % Coef. Diss. [Ns/m]
18
19
20     wd = wn*sqrt(1-E^2);
21     W = 0.2*wn
22
23     F = 40;          %Força
24     f0 = F/k;
25
26
27     %%Solução analitica%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
28
29     Up = {f0}/(((1-(W/wn)^2)^2)+{(2*E*W/wn)^2})^(1/2)}
30     phi_p = atan(2*E*W/wn/(1-(W/wn)^2));
31
32     %Constantes da solução
33     A1 = -(Up*cos(phi_p));
34     A2 = -(Up*(E*wn*cos(phi_p)+W*sin(phi_p))/wd);
35     K1 = {(2*E*(wn^2)*W)/(((wn^2)-(W^2))^2)-{(2*E*wn*W)^2}}*f0;
36     K2 = {(wn^2)*{(wn^2)-(W^2)}/(((wn^2)-(W^2))^2)-{(2*E*wn*W)^2}}*f0;
37     %A parte de solução a cada t está no while da solução numérica
38
39     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
40     %Para o gráfico
41     ti = 0;          % Tempo inicial
42     tf = 10;        % Tempo final
43     h = 0.005;      % Passo do gráfico
44
45     N = (tf - ti)/h; % N° de iterações
46     t = ti;
47
48     %%Solução numérica%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
49     i = 1;
50     t1(i) = ti;
51     X1(i) = x1;
52     X2(i) = x2;
53
54     while t<tf
55         i = i+1;
56         t = t+h;
57
58         k1 = {feval(@var_estado, t1(i-1), X1(i-1), X2(i-1), E, wn,f0, W))*h;
59         k2 = {feval(@var_estado, t1(i-1)+0.5*h, X1(i-1)+0.5*k1(1), X2(i-1)+0.5*k1(2), E,
60         wn,f0, W))*h;
61         k3 = {feval(@var_estado, t1(i-1)+0.5*h, X1(i-1)+0.5*k2(1), X2(i-1)+0.5*k2(2), E,
62         wn,f0, W))*h;
63         k4 = {feval(@var_estado, t1(i-1)+h, X1(i-1)+k3(1), X2(i-1)+k3(2), E, wn,f0, W))*h;
64
65         X0 = [X1(i-1); X2(i-1)];
66         X = X0 + {(k1/6) + (k2/3) + (k3/3) + (k4/6)};
67
68         u_analitico = feval(@sol_analitica, t, E, wn, wd, W, phi_p, A1, A2, Up); %Da solução
69         analitica
70         U(i) = u_analitico; %Solução Analitica
71
72         X1(i) = X(1);
73         X2(i) = X(2);
74
75         t1(i) = t;

```

```

73
74 end
75
76
77 hold on
78 plot(tl,X1*10^3,'linewidth',1,'color','b') %Solução Numérica
79 plot(tl,U*10^3,'-.','linewidth',1.5,'color','k') %Solução analítica
80 title('Solução do sistema (%Omega/omega_0=0.2)','FontSize',10)
81 xlabel('t [s]','FontSize',15)
82 ylabel('oit) [mm]','FontSize',15)
83 hold off
84 box on
85 leg = legend('Solução Numérica', 'Solução Analítica')
86 set(gcf,'color','white')
87 set(gca,'FontSize',15)
88 xlim([0 10])
89
90 end
91
92
93 function y = var_estado(t, xa, xb, E, wn, f0, W) %Variáveis de Estado para RK da
Numérica
94 x_a = xb;
95 x_aa = (-2*E*wn*xb - (wn^2)*xa + (wn^2)*f0*cos(W*t));
96
97 y = [x_a; x_aa];
98 end
99
100 function z = sol_analitica(t, E, wn, wd, W, phi_p, A1, A2, Up) %Equação da Solução
analítica
101 u = (e^(-E*wn*t))*(A1*cos(wd*t)+A2*sin(wd*t))+Up*cos(W*t-phi_p);
102 z = [u];
103 end
104

```

ANEXO C

Código completo – RK mola SMA e mola linear, F= 30, 40 e 50N

```

1  function RK
2
3      clc
4      close all
5      clear all
6      mm = 10^-3;
7      %Condições iniciais
8      %Linear
9      x1 = 0;           % Posicao inicial
10     x2 = 0;          % Velocidade inicial (x2 = u.)
11
12     %Entrada
13
14     %~~~~~
15
16     f1 = 33.0838;
17     f2 = 59.1617;
18     f3 = 34.2517;
19     u1 = 27.7136*mm;
20     u2 = 119.804*mm;
21     u3 = 100.173*mm;
22
23     k_A = f1/u1
24     k_M = (f2-f3)/(u2-u3)
25
26     s1 = f1
27     s2 = f2
28     s3 = f3
29
30     u_r = 75*mm; % ajuste
31
32     %~~~~~
33
34     As = 16.7+273;
35     Af = 30.2+273;
36     Ms = 24.8+273;
37     Mf = 9.1+273;
38     T = 33+273;
39
40     C_M = 5;
41     sigma_cr_s = f1 - C_M*(T-Ms);
42     sigma_cr_f = f2 - C_M*(T-Ms);
43     C_A = s3/(T-As);
44     mat_prop = [u_r k_A k_M C_A C_M sigma_cr_s sigma_cr_f As Af Ms Mf T];
45
46     %~~~~~
47     %
48     sigma0 = 0;
49     epsilon0 = 0;
50     beta0 = 0;
51     %
52     previous_state = [sigma0 epsilon0 beta0];
53
54     div = 10; %aqui
55
56     m = 1;           % massa [Kg] - deixar por unidade de massa
57     k = k_A;         % Coef. Eig. [N/m] - Austenita
58     E = 0.1; %alterei %aqui
59     wn = sqrt(k/m); %frequência natural
60     c = 2*m*E*wn;   % Coef. Diss. [Ns/m]
61
62     W = 0;
63     wd = wn*sqrt(1-E^2);
64
65     %~~~~~
66     %Para o grafico
67     ti = 0;         % Tempo inicial
68     tf = 10;       % Tempo final
69     h = 0.005;     % Passo do grafico
70
71     N = (tf - ti)/h; %Num de interacoes
72     t = ti;
73
74     %%Solucao numerica%~~~~~
75     i = 1;

```

```

76 j = 0;
77 t1(i) = ti;
78 X1(i) = x1;
79 X2(i) = x2;
80
81 dW = 0.01*wn;
82
83 Forca = [30 40 30];
84 p = 0;
85 while p<3
86     p = p+1;
87     F = Forca(p);
88     f0 = F/k;
89     W = -0.01;
90     j = 0;
91     %figure
92     %hold on
93     while (W)<2*wn
94         j = j + 1;
95         W = W + 0.01
96         Amp_1(p,j) = (feval(@amplitude, f0, W, wn, E));
97         x_1(p,j) = W/wn;
98     end
99     %hold off
100 end
101
102 tic
103 p = 0;
104 while p<3
105     p = p+1;
106     F = Forca(p);
107     f0 = F/k;
108     W = 0;
109     j = 0;
110
111     while W<2*wn
112         j = j + 1;
113         W = W + 0.2*wn;
114         i = 1;
115         t1 = 0;
116         X1 = 0;
117         X2 = 0;
118         u = 0;
119         %
120         X11 = 0;
121         X21 = 0;
122         u1 = 0;
123         %
124         t=ti;
125         div = 1;
126        clc
127         tic
128         while t<10
129
130             i = i+1;
131             t = t+h
132
133             %u(i) = X1(i-1)
134             teste = Brinson_poli_uF_mola(u,mat_prop,previous_state,div);
135             [L,-] = size(teste);
136             %u
137             %u teste = teste(L,2)
138             Fm1 = teste(L,1);
139             Fm1a = k*X1(i-1);
140             k1 = (feval(@var_estado, t1(i-1), X1(i-1), X2(i-1), E, wn,f0, W, m, Fm1))*h;
141
142
143             u(i) = X1(i-1)+0.5*k1(1);
144             teste = Brinson_poli_uF_mola(u,mat_prop,previous_state,div);
145             [L,-] = size(teste);
146             Fm2 = teste(L,1);
147             Fm2a = k*(X1(i-1)+0.5*k1(1));
148             k2 = (feval(@var_estado, t1(i-1)+0.5*h, X1(i-1)+0.5*k1(1), X2(i-1)+0.5*k1(2), E,
149                 wn,f0, W, m, Fm2))*h;

```

```

150     u(i) = X1(i-1)+0.5*k2(1);
151     teste = Brinson_poli_uF_mola(u,mat_prop,previous_state,div);
152     [L,~] = size(teste);
153     Fm3 = teste(L,1);
154     Fm3a = k*(X1(i-1)+0.5*k2(1));
155     k3 = (feval(@var_estado, t1(i-1)+0.5*h, X1(i-1)+0.5*k2(1), X2(i-1)+0.5*k2(2), E, wn,f0, W, m, Fm3))*h;
156
157     u(i) = X1(i-1)+k3(1);
158     teste = Brinson_poli_uF_mola(u,mat_prop,previous_state,div);
159     [L,~] = size(teste);
160
161     Fm4 = teste(L,1);
162     Fm4a = k*(X1(i-1)+k3(1));
163     k4 = (feval(@var_estado, t1(i-1)+h, X1(i-1)+k3(1), X2(i-1)+k3(2), E, wn,f0, W, m, Fm4))*h;
164
165     X0 = [X1(i-1); X2(i-1)];
166     X = X0 + ((k1/6) + (k2/3) + (k3/3) + (k4/6));
167
168     X1(i) = X(1);
169     X2(i) = X(2);
170     t1(i) = t;
171     u(i) = X1(i);
172
173
174
175     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
176
177
178     Fm11 = k*X11(i-1);
179     k11 = (feval(@var_estado, t1(i-1), X11(i-1), X21(i-1), E, wn,f0, W, m, Fm11))*h;
180
181     Fm21 = k*(X11(i-1)+0.5*k11(1));
182     k21 = (feval(@var_estado, t1(i-1)+0.5*h, X11(i-1)+0.5*k11(1), X21(i-1)+0.5*k11(2), E, wn,f0, W, m, Fm21))*h;
183
184     Fm31 = k*(X11(i-1)+0.5*k21(1));
185     k31 = (feval(@var_estado, t1(i-1)+0.5*h, X11(i-1)+0.5*k21(1), X21(i-1)+0.5*k21(2), E, wn,f0, W, m, Fm31))*h;
186
187
188     %X11(i-1)+k31(1)
189     Fm41 = k*(X11(i-1)+k31(1));
190     k41 = (feval(@var_estado, t1(i-1)+h, X11(i-1)+k31(1), X21(i-1)+k31(2), E, wn,f0, W, m, Fm41))*h;
191
192     X01 = [X11(i-1); X21(i-1)];
193     X1 = X01 + ((k11/6) + (k21/3) + (k31/3) + (k41/6));
194
195     X11(i) = X1(1);
196     X21(i) = X1(2);
197     ul(i) = X11(i);
198
199     end
200     toc
201     %i
202     X1;
203
204     figure
205     hold on
206     plot(t1,X1)
207     xlabel('t [s]')
208     ylabel('u(t) [m]')
209     xlim([0 tf])
210     ylim([-0.25 0.25])
211     title(F)
212
213     figure
214     plot(X1,X2)
215     xlabel('X1')
216     ylabel('X2')
217     title(W)
218     %xlim([-0.2 0.2])
219     %ylim([-4 4])

```

```

220
221     sigma0 = 0;
222     epsilon0 = 0;
223     beta0 = 0;
224     %
225     previous_state = [sigma0 epsilon0 beta0];
226     %
227     dir = 1;
228     %
229
230     div = 1; %agui
231
232     figure
233     hold on
234     plot(tl,u,'b')
235     plot(tl,ul,'r')
236     leg3 = legend('sma', 'linear')
237     hold off
238
239     testel20 = Brinson_poli_uF_mola(u,mat_prop,previous_state,div);
240     F_poli_t120 = testel20(:,1);
241     u_poli_t120 = testel20(:,2);
242     F1 = k*u1;
243     [linhas,-]= size(testel20);
244     figure
245     hold on
246     plot(u_poli_t120*(10^3),F_poli_t120,'b','linewidth',2) %Solucao octave
247     plot(ul*(10^3),F1,'r','linewidth',1.5) %Solucao octave
248     leg2 = legend('sma', 'linear')
249     xlabel('u [mm]')
250     ylabel('F [N]')
251     title(F)
252     hold off
253
254     x(p,j) = W/wn;
255     %[l,c] = size(X1);
256     %b=round(2*(L/3))
257     %Amp_sma(p,j) = max(abs(X1(b:c)));
258     Amp_sma(p,j) = max(abs(X1(i-1000:i)));
259
260     end
261
262     hold off
263
264     figure
265     hold on
266     plot(x_l(p,:),Amp_l(p,:)*1000,'b','linewidth',1.5) %Solucao Numerica
267     plot(x_l(p,:),Amp_sma(p,:)*1000,'rb','linewidth',1.5) %Solucao Numerica
268     %title('Amplitude vs W/wn','FontSize',15)
269     hold off
270     box on
271     xlabel('\Omega/\omega_n','FontSize',15)
272     ylabel('Amplitude [mm]','FontSize',15)
273     leg = legend('F = 40N (linear)','F = 50N (linear)','F = 60N (linear)','F = 40N
(sma)','F = 50N (sma)','F = 60N (sma)')
274     set(gcf,'color','white')
275     set(gca,'FontSize',15)
276     %ylim([0 0.30])
277     xlim([0 2])
278
279     end
280     toc
281
282     figure
283     hold on
284     plot(x_l(1,:),Amp_l(1,:)*1000,'b','linewidth',1.5) %Solucao Numerica
285     plot(x_l(2,:),Amp_l(2,:)*1000,'r','linewidth',1.5) %Solucao Numerica
286     plot(x_l(3,:),Amp_l(3,:)*1000,'g','linewidth',1.5) %Solucao Numerica
287     plot(x_l(1,:),Amp_sma(1,:)*1000,'rb','linewidth',1.5) %Solucao Numerica
288     plot(x_l(2,:),Amp_sma(2,:)*1000,'or','linewidth',1.5) %Solucao Numerica
289     plot(x_l(3,:),Amp_sma(3,:)*1000,'og','linewidth',1.5) %Solucao Numerica
290     %title('Amplitude vs W/wn','FontSize',15)
291     hold off
292     box on
293     xlabel('\Omega/\omega_n','FontSize',15)

```

```

294 ylabel('Amplitude [mm]', 'FontSize', 15)
295 leg = legend('F = 40N (linear)', 'F = 50N (linear)', 'F = 40N (nonlinear)', 'F = 40N (nonlinear)', 'F = 50N (nonlinear)', 'F = 40N (nonlinear)');
296 set(gcf, 'color', 'white')
297 set(gca, 'FontSize', 15)
298 ylim([0 0.30])
299 xlim([0 2])
300
301
302 figure
303 hold on
304 plot(x_l(1,:), Amp_l(1,:)*1000, 'b', 'linewidth', 1.5) %Solucao Numerica
305 plot(x_l(1,:), Amp_sma(1,:)*1000, 'ob', 'linewidth', 1.5) %Solucao Numerica
306 title('Amplitude vs W/wn', 'FontSize', 15)
307 hold off
308 box on
309 xlabel('\Omega/\omega_n', 'FontSize', 15)
310 ylabel('Amplitude [mm]', 'FontSize', 15)
311 leg = legend('F = 40N (linear)', 'F = 50N (linear)', 'F = 40N (linear)', 'F = 40N (nonlinear)', 'F = 50N (nonlinear)', 'F = 40N (nonlinear)');
312 set(gcf, 'color', 'white')
313 set(gca, 'FontSize', 15)
314 ylim([0 0.30])
315 xlim([0 2])
316
317
318 end
319
320 function y = var_estado(t, xa, xb, E, wn, f0, W, m, Fm) %Variaveis de Estado para RK
321 da Numerica
322 x_a = xb;
323 x_aa = (-2*E*wn*xb - (Fm/m) + (wn^2)*f0*cos(W*t));
324 y = [x_a; x_aa];
325 end
326
327 function z = sol_analitica(t, E, wn, wd, W, phi_p, A1, A2, Up) %Equacao da Solucao
328 analitica
329 u = (e^(-E*wn*t)) * (A1*cos(wd*t) + A2*sin(wd*t)) + Up*cos(W*t - phi_p);
330 z = [u];
331 end
332 function a = f(F, k)
333 f0 = F/k;
334 a = f0;
335 end
336 function b = amplitude(f0, W, wn, E)
337 Up = (f0) / (((1 - (W/wn)^2)^2) + ((2*E*W/wn)^2)^(1/2));
338 b = Up;
339 end
340 function c = phi_p(W, wn, E)
341 phi_p = atan((2*E*W/wn) / (1 - ((W/wn)^2)));
342 c = phi_p;
343 end
344
345
346 function Brinson_poli_strain = Brinson_poli_strain(mat_prop, previous_state, epsilon, dir)
347
348 epsilon_r = mat_prop(1);
349 E_A = mat_prop(2);
350 E_M = mat_prop(3);
351 C_A = mat_prop(4);
352 C_M = mat_prop(5);
353 sigma_cr_s = mat_prop(6);
354 sigma_cr_f = mat_prop(7);
355 A_s = mat_prop(8);
356 A_f = mat_prop(9);
357 M_s = mat_prop(10);
358 M_f = mat_prop(11);
359 T = mat_prop(12);
360
361 a_M = pi / (M_s - M_f);
362 a_A = pi / (A_f - A_s);
363
364 sigma0 = previous_state(1);

```



```

365 epsilon0 = previous_state(2);
366 beta0 = previous_state(3);
367 E0 = E_A + beta0*(E_M-E_A);
368
369
370 sigma1 = (sigma_cr_s+C_M*(T-Ms));
371 sigma2 = (sigma_cr_f+C_M*(T-Ms));
372 sigma3 = (C_A*(T-Af));
373 sigma4 = (C_A*(T-As));
374
375 %
376 epsilon1_min = (sigma1)/E_A;
377 if epsilon<epsilon1_min
378     epsilon1 = epsilon1_min;
379 else
380     epsilon1 = ((sigma1-sigma0) + E0*epsilon0)/E0;
381 end
382 epsilon2 = (sigma2 + epsilon_r*E_M)/E_M;
383
384
385 epsilon3 = ((sigma3-sigma2) + E_M*epsilon2 + epsilon_r*(-E_M))/E_A;
386
387 epsilon4_max = ((sigma4-sigma2) + E_M*epsilon2)/E_M;
388 if epsilon>epsilon4_max
389     epsilon4 = epsilon4_max;
390 else
391     epsilon4 = ((sigma4-sigma0) + E0*epsilon0)/E0;
392 end
393
394 if dir==1
395     if epsilon>epsilon1
396         if epsilon<epsilon2
397             a1 = 1 + ((1-beta0)/(sigma_cr_f-sigma_cr_s))*(-sigma_cr_f-C_M*(T-Ms));
398             a2 = ((1-beta0)/(sigma_cr_f-sigma_cr_s));
399             %
400             a = epsilon_r*(E_M-E_A)*(a2^2);
401             b = 1 + (epsilon_r*E_A-(E_M-E_A)*epsilon)*a2 + epsilon_r*(E_M-E_A)*a1*a2;
402             c = (epsilon_r*E_A-(E_M-E_A)*epsilon)*a1 + epsilon_r*(E_M-E_A)*(a1^2) +
403                 (-sigma0+(-E_A*epsilon+E0*epsilon0)-epsilon_r*E0*beta0);
404             %
405             delta = (b^2)-4*a*c;
406             sigma = (-b+sqrt(delta))/(2*a);
407
408             beta = 1 + ((1-beta0)/(sigma_cr_f-sigma_cr_s))*(sigma-sigma_cr_f-C_M*(T-Ms));
409             E = E_A + beta*(E_M-E_A);
410             %if sigma<sigma0
411             % [epsilon0 epsilon epsilon_r]
412             % [sigma0 sigma (-b+sqrt(delta))/(2*a)]
413             % (sigma0 + E0*(epsilon-epsilon0) -
414             % epsilon_r*E0*(a1-beta0))/(1+epsilon_r*E0*a2)
415             % [beta0 beta]
416             % [E0 E]
417             %
418             % sigma0
419             % E*epsilon - E0*epsilon0
420             % epsilon_r*(E*beta-E0*beta0)
421             % epsilon
422             % sigma
423             % sigma0 + (E*epsilon - E0*epsilon0) - epsilon_r*(E*beta-E0*beta0)
424             %endif
425         else
426             beta = 1;
427             E = E_A + beta*(E_M-E_A);
428             sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
429         end
430     else
431         beta = beta0;
432         E = E_A + beta*(E_M-E_A);
433         sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
434     end
435 elseif dir==1
436     if epsilon>epsilon3

```

```

438     if beta0>0 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
439         if epsilon<epsilon4
440             a1 = (beta0/(Af-As))*(Af-T);
441             a2 = (beta0/(Af-As))*(1/C_A);
442             %
443             a = epsilon_r*(E_M-E_A)*(a2^2);
444             b = 1 + (epsilon_r*E_A-(E_M-E_A)*epsilon)*a2 + epsilon_r*(E_M-E_A)*a1*a2;
445             c = (epsilon_r*E_A-(E_M-E_A)*epsilon)*a1 + epsilon_r*(E_M-E_A)*(a1^2) + %
               (-sigma0+(-E_A*epsilon+E0*epsilon0)-epsilon_r*(E0*beta0));
446             %
447             delta = (b^2)-4*a*c;
448             sigma = (-b+sqrt(delta))/(2*a);
449             beta = (beta0/(Af-As))*((Af-T)+(sigma/C_A));
450         else
451             beta = beta0;
452             E = E_A + beta*(E_M-E_A);
453             sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
454         end
455     else
456         beta = 0;
457         E = E_A + beta*(E_M-E_A);
458         sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
459     end
460
461     else
462         beta = 0;
463         E = E_A + beta*(E_M-E_A);
464         sigma = sigma0 + (E*epsilon-E0*epsilon0) - epsilon_r*(E*beta-E0*beta0);
465     end
466 end
467
468 Brinson_poli_strain = [sigma epsilon beta];
469 end
470
471
472 function Brinson_poli_deslocamento_mola = %
Brinson_poli_uF_mola(U,mat_prop,previous_state,div) %
473
474 L = 1;
475 A = 1;
476
477 EPSILON = U/L;
478
479 resultado_poli_d = feval(@Brinson_poli_epsilon_load,mat_prop,previous_state,EPSILON,div);
480 sig_poli_d = resultado_poli_d(:,1)*(10^6);
481 eps_poli_d = resultado_poli_d(:,2)/100;
482 bet_poli_d = resultado_poli_d(:,3);
483
484 F_poli = sig_poli_d*A;
485 u_poli = eps_poli_d*L;
486
487
488 Brinson_poli_deslocamento_mola = [F_poli u_poli bet_poli_d];
489
490 end
491
492 function Brinson_poli_epsilon_load = %
Brinson_poli_epsilon_load(mat_prop,previous_state,EPSILON,div) %
493 MPA = 10^6;
494
495 %EPSILON
496
497 [linha,coluna] = size(EPSILON);
498 if linha>coluna
499     I = linha;
500 else
501     I = coluna;
502 end
503
504 %I
505 t
506 dir = 1;
507 %
508 sig_poli = 0;
509 eps_poli = 0;

```

```

510 bet_poli = 0;
511 results = previous_state;
512
513 i = 1;
514 j = 1;
515 epsilon = EPSILON(i);
516 while i<I
517     i = i+1;
518     %dir
519     %entrada = EPSILON(i);
520     %if 1>2
521     %EPSILON(i)/EPSILON(i-1);
522     %end
523     if EPSILON(i)>=0
524         if EPSILON(i)>=EPSILON(i-1)
525             if EPSILON(i-1)<0
526                 %EPSILON(i)
527                 if dir==1
528                     dir = -1;
529                     sigma0 = results(1);
530                     epsilon0 = results(2);
531                     beta0 = results(3);
532                     previous_state = [sigma0 epsilon0 beta0];
533                 end
534                 depsilon = (0-EPSILON(i-1))/(div);
535                 if depsilon==0
536                     results = previous_state;
537                     sig_poli(j) = -results(1)/MPa;
538                     eps_poli(j) = -results(2)*100;
539                     bet_poli(j) = -results(3);
540                 else
541                     while epsilon<0
542                         j = j+1;
543                         epsilon = epsilon+depsilon;
544                         results =
545                             feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir)
546                             ;
547                         sig_poli(j) = -results(1)/MPa;
548                         eps_poli(j) = -results(2)*100;
549                         bet_poli(j) = -results(3);
550                     end
551                     dir = 1;
552                     sigma0 = 0;
553                     epsilon0 = 0;
554                     beta0 = 0;
555                     epsilon = 0;
556                     sig_poli(j) = 0;
557                     eps_poli(j) = 0;
558                     bet_poli(j) = 0;
559                     previous_state = [sigma0 epsilon0 beta0];
560                     depsilon = (EPSILON(i))/(div);
561                     if depsilon==0
562                         results = previous_state;
563                         sig_poli(j) = results(1)/MPa;
564                         eps_poli(j) = results(2)*100;
565                         bet_poli(j) = results(3);
566                     else
567                         while epsilon<0.999999*EPSILON(i)
568                             j = j+1;
569                             epsilon = epsilon+depsilon;
570                             results =
571                                 feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir)
572                                 ;
573                             sig_poli(j) = results(1)/MPa;
574                             eps_poli(j) = results(2)*100;
575                             bet_poli(j) = results(3);
576                         end
577                     end
578                     depsilon = (EPSILON(i)-EPSILON(i-1))/div;
579                 if depsilon==0

```

```

581         results = previous_state;
582         sig_poli(j) = results(1)/MPa;
583         eps_poli(j) = results(2)*100;
584         bet_poli(j) = results(3);
585     else
586         if dir==1
587             dir=1;
588             sigma0 = results(1);
589             epsilon0 = results(2);
590             beta0 = results(3);
591             previous_state = [sigma0 epsilon0 beta0];
592         end
593         while epsilon<0.999999*EPSILON(i)
594             j = j+1;
595             epsilon = epsilon+depsilon;
596             results =
597                 feval(@Brinson_poli_strain,mat_prop,previous_state,epsilon,dir);
598             results(1)/MPa;
599             sig_poli(j) = results(1)/MPa;
600             eps_poli(j) = results(2)*100;
601             bet_poli(j) = results(3);
602         end
603     end
604 else
605     if dir==1
606         dir=-1;
607         sigma0 = results(1);
608         epsilon0 = results(2);
609         beta0 = results(3);
610         previous_state = [sigma0 epsilon0 beta0];
611     end
612     depsilon = (EPSILON(i)-EPSILON(i-1))/div;
613     if depsilon==0
614         results = previous_state;
615         sig_poli(j) = -results(1)/MPa;
616         eps_poli(j) = -results(2)*100;
617         bet_poli(j) = -results(3);
618     else
619         while epsilon>1.000001*EPSILON(i)
620             j = j+1;
621             epsilon = epsilon+depsilon;
622             results =
623                 feval(@Brinson_poli_strain,mat_prop,previous_state,epsilon,dir);
624             sig_poli(j) = results(1)/MPa;
625             eps_poli(j) = results(2)*100;
626             bet_poli(j) = results(3);
627         end
628     end
629 end
630 if EPSILON(i-1)>0
631     if dir==1
632         dir = -1;
633         sigma0 = results(1);
634         epsilon0 = results(2);
635         beta0 = results(3);
636         previous_state = [sigma0 epsilon0 beta0];
637     end
638     %clc
639     depsilon = (0-EPSILON(i-1))/(div);
640     if depsilon==0
641         results = previous_state;
642         sig_poli(j) = -results(1)/MPa;
643         eps_poli(j) = -results(2)*100;
644         bet_poli(j) = -results(3);
645         erroaqui
646     else
647         while epsilon>abs(depsilon/(10^6))
648             j = j+1;
649             epsilon = epsilon+depsilon;
650             results =
651                 feval(@Brinson_poli_strain,mat_prop,previous_state,epsilon,dir);
652             sig_poli(j) = results(1)/MPa;
653             eps_poli(j) = results(2)*100;

```

```

653         bet_poli(j) = results(3);
654     end
655 end
656
657 epsilon = 0;
658 sig_poli(j) = 0;
659 eps_poli(j) = 0;
660 bet_poli(j) = 0;
661 dir = 1;
662 sigma0 = 0;
663 epsilon0 = 0;
664 beta0 = 0;
665 previous_state = [sigma0 epsilon0 beta0];
666 depsilon = (EPSILON(i))/(div);
667 if depsilon==0
668     results = previous_state;
669     sig_poli(j) = -results(1)/MPa;
670     eps_poli(j) = -results(2)*100;
671     bet_poli(j) = -results(3);
672 else
673
674 %EPSILON(i)
675 while epsilon>0.999999*EPSILON(i)
676     j = j+1;
677     epsilon = epsilon+depsilon;
678     results =
679         feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
680     sig_poli(j) = -results(1)/MPa;
681     eps_poli(j) = -results(2)*100;
682     bet_poli(j) = -results(3);
683     %if j>500
684     % aaaaaaaaaaaaaaaaaaaaaa
685     %endif
686 end
687 end
688
689 else
690 if EPSILON(i)<=EPSILON(i-1)
691     %dir
692     if dir==1
693         depsilon = (EPSILON(i)-EPSILON(i-1))/(div);
694         if depsilon==0
695             results = previous_state;
696             sig_poli(j) = -results(1)/MPa;
697             eps_poli(j) = -results(2)*100;
698             bet_poli(j) = -results(3);
699         else
700             while epsilon>0.999999*EPSILON(i)
701                 j = j+1;
702                 epsilon = epsilon+depsilon;
703                 results =
704                     feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
705                 sig_poli(j) = -results(1)/MPa;
706                 eps_poli(j) = -results(2)*100;
707                 bet_poli(j) = -results(3);
708             end
709         end
710     end
711
712 else
713     dir = 1;
714     sigma0 = abs(results(1));
715     epsilon0 = abs(results(2));
716     beta0 = abs(results(3));
717     previous_state = [sigma0 epsilon0 beta0];
718     depsilon = (EPSILON(i)-EPSILON(i-1))/(div); %aqui
719     if depsilon==0
720         results = previous_state;
721         sig_poli(j) = -results(1)/MPa;
722         eps_poli(j) = -results(2)*100;
723         bet_poli(j) = -results(3);
724     else
725         while epsilon>0.999999*EPSILON(i)
726             j = j+1;

```

```

725         epsilon = epsilon+depsilon;
726         results =
            feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
727         sig_poli(j) = -results(1)/MPa;
728         eps_poli(j) = -results(2)*100;
729         bet_poli(j) = -results(3);
730     end
731 end
732 end
733 end
734 else
735     %dir
736     if dir==1
737         dir = -1;
738         sigma0 = abs(results(1));
739         epsilon0 = abs(results(2));
740         beta0 = abs(results(3));
741         previous_state = [sigma0 epsilon0 beta0];
742     end
743     depsilon = (EPSILON(i)-EPSILON(i-1))/dir;
744     if depsilon==0
745         results = previous_state;
746         sig_poli(j) = -results(1)/MPa;
747         eps_poli(j) = -results(2)*100;
748         bet_poli(j) = -results(3);
749     else
750         while epsilon<1.000001*EPSILON(i)
751             j = j+1;
752             epsilon = epsilon+depsilon;
753             results =
                feval(@Brinson_poli_strain,mat_prop,previous_state,abs(epsilon),dir);
                sig_poli(j) = -results(1)/MPa;
                eps_poli(j) = -results(2)*100;
                bet_poli(j) = -results(3);
754             end
755         end
756     end
757 end
758 end
759 end
760 end
761 end
762 %saida = eps_poli(j)
763 end
764
765 %j
766
767
768 Brinson_poli_epsilon_load = [sig_poli' eps_poli' bet_poli'];
769
770 end
771
772 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```