



Instituto Superior de Engenharia de Lisboa (ISEL)

Área Departamental de Engenharia Mecânica (ADEM)

# Benchmarking de identificação de dano

Criação de um site para benchmarking de técnicas de identificação de dano

JOÃO PEDRO DA CONCEIÇÃO CASTANHEIRA  
Licenciado em Engenharia Mecânica

Dissertação para obtenção do grau de Mestre em Engenharia Mecânica na Área de Manutenção e Produção

***Orientador:***

Professor Doutor Rui Pedro Chedas Sampaio

***Júri:***

*Presidente - Professor Doutor Joaquim Infante Barbosa*

*Vogal - Professora Doutora Cláudia Dias Sequeira*

*Vogal - Professor Doutor Rui Pedro Chedas Sampaio*

Portugal, Lisboa  
3 de Dezembro de 2015

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>9</b>
1.1	Enquadramento e justificação do tema . . . . .	9
1.2	Objectivos da dissertação . . . . .	10
1.3	Estrutura da dissertação . . . . .	11
<b>2</b>	<b>Estado da Arte</b>	<b>12</b>
2.1	Dados científicos disponíveis . . . . .	12
2.2	Métodos de SHM por medição de vibrações . . . . .	16
<b>3</b>	<b>Fundamentos Teóricos de Vibrações</b>	<b>19</b>
3.1	Conceitos Básicos de Vibrações . . . . .	19
3.2	Medição de vibrações . . . . .	23
3.2.1	Conceito de Vibração Harmónica . . . . .	23
3.2.2	Conceito de FFT . . . . .	24
3.2.3	Conceito de FRF . . . . .	26
3.2.4	Cálculo de Respostas no Tempo . . . . .	29
3.3	Métodos de identificação de dano seleccionados . . . . .	31
3.3.1	Algoritmos seleccionados . . . . .	32
<b>4</b>	<b>Site de Benchmarking</b>	<b>35</b>
4.1	Estrutura e funcionamento . . . . .	35
4.1.1	Exemplo 1 . . . . .	37
4.1.2	Exemplo 2 . . . . .	40
4.2	Modelos e dados fornecidos . . . . .	41
<b>5</b>	<b>Conclusões</b>	<b>47</b>
<b>A</b>	<b>Códigos MATLAB</b>	<b>51</b>
A.1	Algoritmos de Detecção de Dano . . . . .	51
A.1.1	FRF DRQ - Modelo Espacial . . . . .	51
A.1.2	FRF Difference Index - Modelo Espacial . . . . .	55

A.1.3	FRF TDI - Modelo Espacial . . . . .	57
A.1.4	FRF DRQ - Respostas no tempo . . . . .	60
A.1.5	FRF Difference Index - Respostas no tempo . . . . .	63
A.1.6	FRF TDI - Respostas no tempo . . . . .	67
A.2	Algoritmos de Localização do Dano . . . . .	70
A.2.1	FRF BASED - MODE SHAPE CURVATURE - Modelo Es- pacial . . . . .	70
A.2.2	FRF DAMAGE INDEX - Modelo Espacial . . . . .	72
A.2.3	FRF BASED - MODE SHAPE CURVATURE - Modelo Res- postas no Tempo . . . . .	75
A.2.4	FRF DAMAGE INDEX - Modelo Respostas no Tempo . . . . .	77

## Resumo

Desde dos anos 80 que se tem vindo a estudar e desenvolver várias técnicas e algoritmos matemáticos para a identificação de dano em estruturas. Hoje em dia começa a ser cada vez mais importante avaliar estas técnicas e perceber o desempenho que cada uma delas pode oferecer para sistemas de monitorização da integridade estrutural (Structural Health Monitoring - SHM) para os mais variados tipos de estruturas, tais como, pontes, aviões, prédios, barcos, aerogeradores, entre outras. Este trabalho pretende ser uma ferramenta de auxílio a este tipo de avaliação.

Os investigadores desenvolvem determinado algoritmo/método e testam-no numericamente e experimentalmente. No primeiro caso recorrem a modelos FEM, entre outros, e no segundo caso recorrem a dados experimentais que obtêm em laboratório ou em estruturas civis como pontes ou edifícios. Para se compararem diferentes métodos é necessário que se utilizem os mesmos dados e que cada algoritmo/método seja programado pelo autor. Os modelos numéricos são relativamente fáceis de construir pelo próprio investigador mas as estruturas reais não. Normalmente o investigador mede estruturas em laboratório e para tal necessita de sensores e equipamento de aquisição/processamento de sinal normalmente muito dispendiosos. Quanto a medir as estruturas civis é extremamente raro haver autorização dos donos dessas estruturas para se danificar alguma parte e depois medir. A melhor altura para se medir uma estrutura civil é quando esta vai ser abatida ao serviço. Mas para se efetuarem medições é necessária a conjugação de uma série de factores, como o dono da estrutura estar sensibilizado para este tipo de investigação, haver orçamento que suporte os custos das medições e o atraso que as mesmas podem implicar no descomissionamento da estrutura, haver tempo tanto do dono como dos investigadores, etc... O resultado prático destes factores leva a uma quase inexistência deste tipo de dados.

Levando isso em consideração, no presente trabalho foi desenvolvido um site de acesso publico na Internet que permite a comparação entre vários algoritmos de identificação de dano.

Neste trabalho é apresentado um conjunto de exemplos que demonstram a funcionalidade do programa.

Foram disponibilizados no site alguns modelos FEM, criados para esse efeito neste TFM, e foram disponibilizadas medições numa viga em laboratório, que foram cedidas pelo Professor Chedas Sampaio. Também, foram disponibilizados no site alguns algoritmos de detecção e localização de dano, programados para esse efeito. De forma a efectuar uma demonstração do programa foram também adicionados alguns algoritmos de calculo.

Palavras Chave: SHM - Structural Health Monitoring; Identificação de dano em estruturas;

# Abstract

Since the 80's has been studied and developed several techniques and mathematical algorithms to identify damage in structures. It is very important to evaluate these techniques and algorithms in order to understand the performance that each one can provide for systems for monitoring the structural integrity (Structural Health Monitoring - SHM) for various types of structures such as bridges, aircraft, buildings, boats, wind turbines, among others. This work is intended to design and produce a web tool to aid assessment of these techniques.

A researcher develops the algorithm/method and test it numerically and experimentally. In the first case relies in a FEM models, or others models, and in the second case rely on experimental data obtained in the laboratory or in civil structures such as bridges or buildings. To compare different methods it is necessary to use the same data and each algorithm / method is set by the author. The numerical models are relatively easy to build by the investigator himself but a real structure is not. Usually the investigator measures structures in the laboratory however it requires sensors and signal processing equipments to do it, that are usually very costly. Measure a civil structure is extremely rare, and usually there is no permission to do it. The best time to measure a civil structure is when it will be decommissioned. But to take effective measurements is required a combination of several factors, such as the owner of the structure to be aware for this type of research, a budget to bear the costs of the measurements and the delay that they may involve in the decommissioning of the structure, etc ... The net result of these factors leads to a virtual absence of such data in today's days.

Therefore here is presented a development of a public access website that allows comparison between different damage identification algorithms for several structures (theoric and real).

This paper presents a set of examples that demonstrate the program's functionality.

On the website were made available some FEM models created for this purpose and measurements were of a beam in the laboratory, which were provided by Professor Chedas Sampaio. Also, some algorithms of detection and location of damage were programmed for this purpose.

Keywords: SHM - Structural Health Monitoring; Damage identification in structures;

# Agradecimentos

Quero deixar aqui os meus agradecimentos a todos aqueles que colaboraram e tornaram possível a elaboração da presente dissertação.

Ao meu orientador Doutor Rui Pedro Chedas Sampaio agradeço a oportunidade e confiança demonstrada para realização da dissertação, bem como, os conhecimentos transmitidos, a ajuda, motivação, disponibilidade e paciência demonstrados ao durante este longo tempo de trabalho.

Agradeço aos meus colegas de trabalho e amigos, nomeadamente ao Nuno Franco, pela ajuda, paciência e conhecimento demonstrado.

Agradeço a toda a minha família, pelo apoio, orientação e incentivo que sempre demonstraram durante todo o meu percurso de vida.

Por fim, agradeço ao Instituto Superior de Engenharia de Lisboa (incluindo professores e colegas de turma) pelo conhecimento multidisciplinar transmitido até hoje.

# Nomenclatura

1. *SHM* - Structural Health Monitoring
2. *CM* - Condition Monitoring
3. *FRF* - Frequency Response Function
4. *FEM* - Finite Element Model
5. *DOF* - Degree Of Freedom
6. *RVAC* - Response Vector Assurance Criterion
7. *DRQ* - Detection and Relative Damage Quatification
8. *MS* - Mode Shape
9. *TDI* - Transmissibility Damage Indicator
10. *DI* - Damage Index
11. *MSCW* - MATLAB Script Command Window
12. *TFM* - Trabalho Final de Mestrado
13. *SIMCES* - System Identification to Monitor Civil Engineering Structures

# Capítulo 1

## Introdução

### 1.1 Enquadramento e justificação do tema

Muitas das vezes a falha de uma infra-estrutura, tal como uma ponte, um avião, ou uma plataforma petrolífera, pode ter como consequência perdas humanas, perda de equipamento dispendioso ou poluição do ambiente com agentes nocivos, tais como petróleo ou outros produtos químicos. De forma a evitar estes acidentes a sociedade tem vindo a desenvolver e a prepara técnicas de gestão de manutenção preventiva que evitam a ocorrência destes acidentes. Contudo estas técnicas de manutenção preventiva são geralmente baseadas em regras e modelos temporais, não sendo claro perceber qual é a real condição do elemento/estrutura.

Um exemplo simples da manutenção preventiva é aquele caso em é necessário mudar o óleo do motor de um automóvel. Tipicamente muda-se o óleo uma vez por ano, contudo nunca temos noção por quanto tempo mais o óleo poderia ser usado, ou então o óleo já devia ter sido substituído a algum tempo pois apresenta sinais de desgaste.

De modo a ser mais eficaz na manutenção de máquinas, a industria tem vindo a aplicar o conceito de *controlo de condição*, isto é, é o processo de monitorização de um parâmetro de estado das máquinas (vibração, temperatura, etc.), com o objectivo de identificar uma alteração significativa que é indicativa de uma falha. Também é possível aplicar este conceito de monitorização em estruturas civis, aeroespaciais e industriais. A este tipo de monitorização dá-se o nome de Structural Health Monitoring (SHM). As principais vantagens destas técnicas são permitir a redução de perdas humanas, ambientais e financeiras.

Alguns exemplos:

1. **Aerogeradores** - O tempo de paragem de um aerogerador por avaria equivale a um prejuízo de algumas centenas de euros por hora. O dano ou avaria de um componente estrutural, tal como uma pá ou estrutura de suporte, podem levar alguns meses a serem reparados o que leva a um prejuízo de milhares de euros.
2. **Aeronáutica** - Descobrir um dano numa estrutura de um avião é uma tarefa complexa e dispendiosa. É necessário cerca de um mês para a realização de uma inspecção total em uma estrutura da dimensão de um Airbus A320.
3. **Espacial** - Um dano numa estrutura espacial, tal como um foguetão ou um outro veículo de transporte espacial, pode levar a uma falha catastrófica.

## 1.2 Objectivos da dissertação

O principal objectivo deste trabalho é a criação de um website público, na Internet, que sirva de ferramenta de calculo para benchmarking de técnicas de identificação de dano em estruturas mecânicas.

A necessidade da criação deste website/programa advém da dificuldade de implementar estas técnicas de identificação de dano em estruturas reais.

O investigador Finlandês Jyrki Kullaa [1] afirma que *”Em muitos casos, os analistas não possuem recursos para adquirir dados ou para executar simulação realistas. Assim sendo, seria interessante existir uma biblioteca com dados de SHM para testes e validar os algoritmos.”*

Uma entidade de investigação Americana, o laboratório SSTL (Smart Structures Technology Laboratory) da Universidade do Illinois também afirma no seu website institucional que é necessário reunir esforços para melhorar os algoritmos de SHM:

*”Muitos algoritmos e tecnologias têm sido propostos para o controlo de estruturas de engenharia civil, cada um com as suas próprias vantagens, dependendo da aplicação específica e efeito desejado. Ainda hoje, os estudos que são efectuados não conseguem demonstrar quais são os prós e contras das diferentes abordagens. Claramente, a capacidade de fazer comparações directas entre as estratégias que empregam vários algoritmos e tecnologias deve ser melhorada, pois é necessário concentrar esforços futuros nas direcções mais promissoras e para definir efectivamente metas e especificações de desempenho.”*

É evidente que o conhecimento e desenvolvimento tecnológico nesta área ainda é muito reduzido.

Sendo uma área abrangente e de modo a focar esforços num tema apenas, foi decidido criar e dedicar este programa apenas para as técnicas com base em vibrações mecânicas. O site desenvolvido no presente trabalho disponibilizará dados de modelos de elementos finitos (FEM) e de estruturas reais. Os investigadores poderão testar os seus métodos online, compará-los com métodos já disponibilizados no site e, posteriormente, fazer o upload para o site do seu próprio método.

### **1.3 Estrutura da dissertação**

Esta dissertação será dividida em quatro partes.

A primeira é referente ao enquadramento do tema. Aqui serão demonstradas e explicadas algumas das necessidades a que levaram criar este programa/web-site.

A segunda parte servirá para demonstrar o estado da arte e aprofundar o conhecimento nas técnicas usadas no tema de Structural Health Monitoring.

De seguida são aprofundados os conhecimentos fundamentais em Teoria de Vibrações. Esta passagem teórica é essencial para o uso correto do programa/web-site de benchmarking e para a correta interpretação dos dados. Também são fornecidos alguns modelos teóricos de métodos/algoritmos para identificação e localização de dano em estruturas.

Por fim é apresentado o web-site e as suas funcionalidades. Aqui são demonstrados o painel principal de controlo e os modelos de estruturas fornecidos no web-site.

## Capítulo 2

# Estado da Arte

### 2.1 Dados científicos disponíveis

Para o desenvolvimento de algoritmos no SHM são necessários dados relativos a medições/simulações sobre algumas propriedades das estruturas a analisar.

De seguida são apresentados algumas bases de dados públicas:

1. **Benchmark Comparisons** - É uma base de dados desenvolvido pelo *Smart Structure Technology Laboratory (SSTL)*, que possui dados experimentais e relatórios experimentais. Nesta base de dados existem vários tipos de estruturas sendo que são maioritariamente estruturas de engenharia Civil. Alguns exemplos de estruturas disponíveis:
  - (a) Edifício de 20 andares excitado por terramoto
  - (b) Edifício de 76 andares excitado pelo vento
2. **Experimental data - Los Alamos** - É um arquivo público criado pelo laboratório de Los Alamos. Esta base de dados possui a descrição das experiências, os dados adquiridos e a nomenclatura aplicada. Os dados são gravados em vários formatos tais como Matlab(.mat) e formato universal (.UFF e .UNV). Também são adicionados relatórios e alguns modelos FEM. Alguns exemplos de estruturas disponíveis:
  - (a) Alamosa Canyon Bridge
  - (b) I-40 Bridge
  - (c) UC-Irvine Bridge Column
  - (d) Sheraton Hotel

3. **SHM Benchmark Data Download Page** - O investigador Finlandês, Jyrki Kullaa, disponibiliza três casos de estudo na sua *webpage* [1] .

- (a) O primeiro caso de estudo é referente a uma viga bi-apoiada nas suas extremidades e com uma mola presa numa zona da viga. Também são adicionadas as variações ambientais e operacionais tais como: temperatura ambiente e excitações aleatórias.
- (b) O segundo caso de estudo é a realização de simulações FEM numa viga com uma fenda. O comportamento de propagação da fenda simulada foi de dois tipos, de comportamento linear e não-linear. Foram extraídos 15 pontos de medição, de forma a simular acelerómetros.
- (c) Por último foi modelada uma viga bi-apoiada com cargas móveis, o objectivo é simular o tráfico automóvel em pontes.

4. **SMC Benchmark Problems for Condition Assessment and Damage Detection** - - No centro de estudos SMC (Structural Monitoring and Control) do Instituto Tecnológico de Harbin, na China, foi preparado um estudo de análise estrutural de uma ponte estaiada à escala real. No *website* [2] do instituto é possível encontrar os seguintes dados:

- (a) Estudo da condição dos cabos de suporte da ponte: São fornecidos ficheiros com as tensões no cabos ao longo do tempo.
- (b) Detecção de dano na Viga-Mestra da ponte: São fornecidos dois ficheiros diferentes, um ficheiro relativo ao sinal no tempo das acelerações medidas na viga mestra e outro ficheiro é relativo às condições ambientais (temperatura ambiente e condições do vento).

5. **Projecto SIMCES** - - SIMCES foi um projecto europeu que decorreu entre 1997 e 1999, que tinha como objectivo testar técnicas de SHM numa estrutura real. O departamento de Engenharia Civil da Universidade Católica de Leuven foi responsável pelo projecto. O acesso aos dados e às publicações científicas pode ser feito na sua *webpage* [3].

Começa a aparecer uma grande variedade de dados dedicadas ao tema de SHM. O processamento dos dados de todos os exemplos anteriores é um processo de grande complexidade e envolve o conhecimento de diferentes áreas científicas. O professor Charles Farrar afirma [4] que "*A educação não está a evoluir de forma a colmatar as necessidades multi-disciplinares que a integração de tecnologias exige*".

Assim sendo, nos dias de hoje pelo mundo inteiro decorrem muitas actividades associadas ao tema de SHM, com o objectivo de melhorar os conhecimentos dos

alunos e investigadores com as técnicas/algoritmos. Algumas destas actividades correm em ambiente académico e outras em ambiente profissional, vejamos alguns exemplos:

1. **Dynamics Summer School** - É um curso de verão do laboratório *Los Alamos National Laboratory (LANL)* e foca-se em agrupar um conjunto de alunos licenciados do 1º ano de licenciatura num campus de engenharia multidisciplinar durante as férias de verão. É pretendido neste curso construir sistemas com sensores, controladores e tecnologia da informação para todos os tipos de sistemas de engenharia.  
*Website* - [5]
2. **Smart Structure Technology Laboratory** - Este laboratório pertence há Universidade do Illinois. A sua pesquisa está focada nas áreas de SHM, controle estrutural e tecnologias de sensores inteligentes.  
*Website* - [6]
3. **Asia-Pacific Summer School on Smart Structures Technology** - Este programa tem a intenção de unir alunos licenciados dos EUA, Coreia do Sul, Japão, China, Índia e Tailândia. O Encontro dura três semanas e tem actividades focadas na área de tecnologias para estruturas inteligentes. Os alunos participantes serão orientados no seu currículo de forma a colmatar algumas lacunas existentes na sua educação ligadas a esta temática, que são necessários para o avanço das actividades. Um dos objectivos do programa é colocar alunos em contacto com vários investigadores internacionais, de modo a inspirar e a seguir uma carreira na área da investigação.

Alguns destes laboratórios, tais como, *Los Alamos National Laboratory (LANL)* e *Smart Structure Technology Laboratory (SSTL)* têm vindo a desenvolver softwares, estruturas e metodologias para o SHM, tais com:

***Softwares desenvolvidos:***

1. **SHMtools** - É um programa open-source e desenvolvido por *Los Alamos National Laboratory (LANL)* para MATLAB. Este programa facilita a construção de processos de SHM. O pacote oferece um conjunto de funções organizadas em módulos de acordo com os três estágios primários de Monitorização Estrutural: Aquisição de Dados, extracção de características e classificação de funções. No programa são incluídos vários algoritmos com códigos fonte, juntamente com os dados estruturais para servir como pontos de referência para a avaliação dos algoritmos. O SHMTools pode funcionar com uma interface gráfica, usando a aplicação *mFUSE* desenvolvido em Java. Este programa é bastante completo, permitindo realizar vários, tipos de análises aos dados, comparação de algoritmos entre outros. Contudo por ser um programa muito complexo e extenso, pode-se tornar difícil de perceber, instalar e usar, requerendo muito tempo e estudo de preparação.
2. **Diamond Tools** - é uma ferramenta desenvolvido por *Los Alamos National Laboratory (LANL)* para MATLAB que implementa várias técnicas recentes para a análise experimental modal, identificação de dano e correlação modelo de elementos finitos.
3. **ProDia** - Este programa é proprietário de uma empresa portuguesa, a *Critical Materials*. O principio de funcionamento do *ProDia* divide-se em dois, ou seja, o primeiro é desenvolver um modelo em FEM muito aproximado da estrutura em análise. Depois de desenvolvido o modelo espacial da estrutura real, são adicionados alguns sensores pela estrutura. Os sensores de vibração são colocados em pontos estratégicos. Com estes dois processos implementados o *ProDia* é usado como ferramenta de previsão de dano na estrutura em análise. Pouca informação existe sobre este programa, pois é um programa proprietário.

Apenas estão apresentadas alguns exemplos de grupos, centros, bases de dados e programas associados à temática SHM.

## 2.2 Métodos de SHM por medição de vibrações

A ideia fundamental dos métodos de SHM baseadas em medição de vibrações é obter a resposta dinâmica da estrutura em diferentes localizações. A resposta da estrutura será tipicamente medida através de acelerómetros e/ou extensómetros. O dano existente na estrutura irá alterar as características dinâmicas da própria estrutura.

Assim sendo é necessário realizar um processo de diagnóstico da estrutura em causa. Este processo de identificação do dano pode ser dividido em 5 etapas/processos [7]:

1. *Detecção* - Existe dano na estrutura?
2. *Localização* - Qual é a localização do dano na estrutura?
3. *Classificação* - Que tipo de dano está presente?
4. *Quantificação* - Qual é a extensão do dano na estrutura?
5. *Previsão* - Quanto resta de vida útil?

Em cada uma das etapas de identificação de dano tem-se desenvolvido diferentes técnicas/algoritmos de cálculo. Um dos livros mais populares sobre SHM, *Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in Their Vibration Characteristics: A Literature Review* [8], possui uma extensa e resumida lista dessas técnicas de SHM para as diferentes etapas do processo de identificação de dano.

A classificação dos métodos divide-se essencialmente em dois tipos, em **Modelos** e **Medições reais**:

1. *Modelos* - Normalmente a identificação do dano faz-se comparando a medição actual com a medição prevista pelo modelo. Altera-se o modelo até se obter a medição actual. As diferenças entre o modelo da estrutura e o modelo alterado permitem identificar o dano.

### **Vantagens:**

- Permitem ir mais longe na identificação, seja na quantificação ou na previsão.

### **Desvantagens:**

- Necessidade de um modelo rigoroso da estrutura.

- Dificuldade em representar no modelo as condições de fronteira (ex: fundações, marés,...).
  - Diferentes modelos podem dar o mesmo resultado.
2. *Medições reais* - A identificação do dano é feita comparando duas medições, a medição da estrutura sã (referência) e a medição actual.

**Vantagens:**

- Não necessitam de um modelo da estrutura.

**Desvantagens:**

- Dificuldade em quantificar e prever o tempo de vida restante da estrutura.

Quer seja através de **Modelos** ou **Medições reais** é necessário extrair indicadores quantitativos para poder avaliar o estado das estruturas.

A próxima lista mostra alguns tipos de indicadores com uma breve descrição [9]:

1. **Frequências Naturais** - Cada estrutura possui um espectro de frequências único e ao longo do tempo esta identidade única da estrutura vai-se modificando. A identificação do dano é feita comparando duas medições: a medição da estrutura sã (referência) e a medição actual. O dano altera as frequências naturais e estas podem ser medidas em qualquer ponto da estrutura (necessitam apenas de um sensor). Estes métodos permitem detectar e localizar o dano, contudo as variações operacionais (ex: trânsito) e ambientais (ex: temperatura) e a impossibilidade de detectar múltiplos danos ao longo da estrutura têm levado a que comunidade científica tenha perdido o interesse nestes métodos nos últimos anos.
2. **Modos** - Estes métodos comparam os modos de vibração da estrutura sã (referência) com os modos da estrutural actual. O cálculo dos modos de vibração requerem análises modais complexas. Permitem detectar e localizar o dano, contudo necessitam de medir varios pontos da estrutura, requerem um excitação de uma shaker ou martelo de impacto e na prática apenas se consegue medir os primeiros modos.
3. **Energia de deformação** - Para estruturas tipo viga, a deformação está relacionada com a curvatura e com a rigidez da estrutura. A deformação

pode medir-se directamente com extensómetros ou indirectamente com acelerómetros através da curvatura.

4. **Deformadas Operacionais** - Baseiam-se no facto de que as deformadas da estrutura (deformadas a todas as outras frequências que não as naturais) contém mais informação que os modos. Todos os métodos baseados nos modos podem ser implementados usando as deformadas em vez dos modos. Podem ser obtidas a partir do domínio do tempo ou da frequência. Estes métodos não necessitam de identificação modal e são mais sensíveis ao dano que os *modos*. Também possuem as suas desvantagens pois requerem medições em muitos pontos da estrutura, só se conseguem excitar os primeiros modos e necessitam de uma excitação que seja mensurável em unidades de força (Newton).

Neste trabalho foram apenas apresentados alguns exemplos de indicadores usados em SHM.

O laboratório de *Los Alamos* publicou um livro [8], que ainda hoje é uma referência bibliográfica mundial sobre a classificação dos indicadores de identificação de dano em estruturas, onde apresenta cerca de 90 técnicas e indicadores. Todos estes métodos/indicadores são usados para manipular matematicamente apenas 3 tipos de dados existentes nos modelos de uma estrutura, que são:

1. **Modelo Espacial** - Representa descrição espacial das propriedades dinâmicas das estrutura.  $\mathbf{M}\ddot{\vec{x}}(t) + \mathbf{K}\vec{x}(t) = \vec{f}(t)$
2. **Modelo Modal** - Este modelo fornece os modos de vibração das estrutura para a respectiva frequência natural.  $(\mathbf{K} - \omega_r^2\mathbf{M})\Psi_r = 0$
3. **Modelo Resposta** - Através deste modelo consegue-se obter as repostas no domínio do tempo de qualquer estrutura, para qualquer excitação.  $(\mathbf{K} - \omega^2\mathbf{M})X(\omega) = F(\omega)$

Onde:

$\mathbf{M}$  - matriz massa

$\mathbf{K}$  - matriz rigidez

$\Psi$  - matriz modal

$\vec{x}(t)$  - vetor deslocamentos

$\ddot{\vec{x}}(t)$  - vetor acelerações

$\vec{f}(t)$  - vetor de forças aplicadas

$\omega_r$  - frequências naturais

## Capítulo 3

# Fundamentos Teóricos de Vibrações

### 3.1 Conceitos Básicos de Vibrações

Nesta secção serão descritos alguns fundamentos teóricos de vibrações para uma melhor compreensão das técnicas e modelos apresentados. Para desenvolver e trabalhar com técnicas de identificação de dano em estruturas é necessário compreender o tema das *Vibrações*. A *Vibração* pode ser classificada das seguintes formas [10]:

1. **Vibração Livre** - É um sistema que vibra continuamente, sem a atuação de forças externas.
2. **Vibração forçada** - É um sistema que vibra continuamente quando é sujeito a atuação de uma força externa periódica.

Qualquer sistema que vibre no mundo real não vibra continuamente, existe sempre atenuação do seu movimento ao longo do tempo. Diz-se então que a vibração é *Amortecida*.

Podemos descrever as vibrações mecânicas com três propriedades físicas:

1. **Massa ou Inércia**- É o armazenamento da energia cinética. (kg)
2. **Rigidez** - É o armazenamento da energia potencial. (N/m)
3. **Amortecimento** - É a dissipação de energia. (N.s/m)

A estas três propriedades físicas chama-se o *Modelo Espacial*, podendo com ele definir qualquer sistema oscilatório.

As próximas três imagens são a representação gráfica de um modelo de 1 grau de liberdade (1 DOF) para o Modelo Espacial (  $m$ ,  $k$  e  $c$ ), vibração livre e vibração forçada, respetivamente. Onde  $m$  é a massa,  $k$  é a rigidez e o  $c$  é o amortecimento.

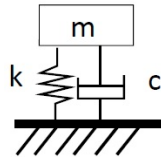


Figura 3.1: Modelo Espacial

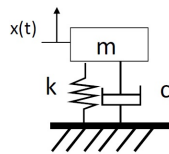


Figura 3.2: Modelo Espacial - Vibração livre

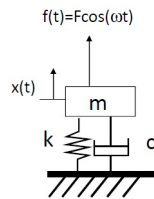


Figura 3.3: Modelo Espacial - Vibração forçada

A vibração pode ser livre ou forçada, tal como foi afirmado anteriormente. Vejamos então a representação matemática destas duas formas de vibração para o sistema de 1 grau de liberdade (1 DOF).

De acordo com a Segunda Lei de Newton, Lei Fundamental da Dinâmica:

”A força resultante do conjunto de forças que atuam num corpo produz nele uma aceleração com a mesma direção e o mesmo sentido da força resultante, que é tanto maior quanto maior for a intensidade da força resultante.”

Podemos aplicar esta Lei aos vetores da figura anterior, obtendo assim a seguinte expressão do equilíbrio de forças:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = 0 \quad (3.1)$$

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = f(t) \quad (3.2)$$

A equação 3.1 é relativa a uma vibração livre e a equação 3.2 é relativa a uma vibração forçada periódica.

As técnicas de identificação de dano aqui apresentadas são para sistemas que possuem vários *Graus de Liberdade (NDOF)*.

Assim sendo é necessário compreender como são construídos os *Modelos Espaciais de N graus de Liberdade*.

A construção de um modelo de n graus de liberdade inicia-se quando é necessário discretizar uma estrutura real em várias partes, vejamos a próxima figura como exemplo.

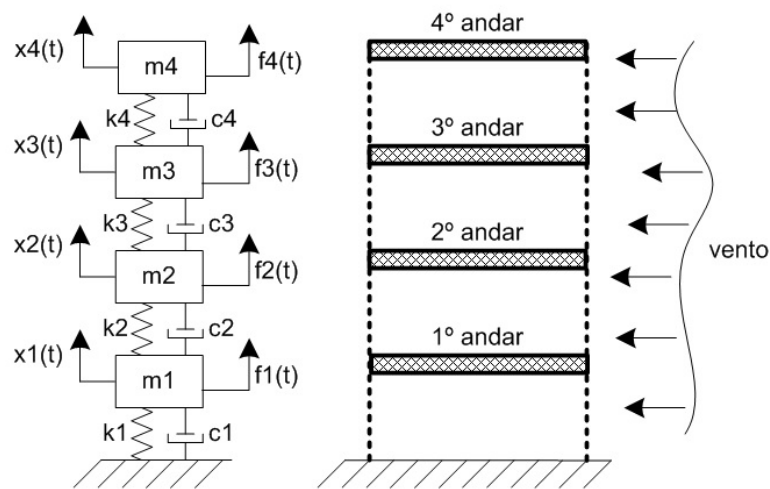


Figura 3.4: Modelo Espacial de um edifício de 4 andares

A representação do Modelo Espacial de um edifício de quatro andares pode ser feita usando um sistema de multi-corpo, 4 massas ( $m_1$ ,  $m_2$ ,  $m_3$  e  $m_4$ ) interligadas por quatro molas ( $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ ) e quatro amortecedores ( $c_1$ ,  $c_2$ ,  $c_3$ ,  $c_4$ ), tal como é representada na figura 3.4.

*Significado de cada propriedade:*

1. **m** - Representa a massa de cada andar, incluindo laje, paredes e recheio.(kg)
2. **k** - É a rigidez das estruturas que suportam cada andar, normalmente estão associados aos pilares de betão armado entre lajes.(N/m)
3. **c** - Caracteriza a dissipação de energia inerente aos materiais e estrutura.(N.s/m)
4. **f(t)** - Força aplicada.(N)

5.  $\mathbf{x}(t)$  - Deslocamento.(m)

Tal como nas figuras 3.2 e 3.3 que possuem uma representação matemática do equilíbrio de forças, equação 3.1 e 3.2, respetivamente, um sistema de N graus de liberdade também pode ser representado matematicamente.

A expressão do equilíbrio de forças será:

$$\mathbf{M}\ddot{\vec{x}}(t) + \mathbf{C}\dot{\vec{x}}(t) + \mathbf{K}\vec{x}(t) = \vec{f}(t) \quad (3.3)$$

Onde:

$\mathbf{M}$  - matriz massa

$\mathbf{K}$  - matriz rigidez

$\mathbf{C}$  - matriz amortecimento viscoso

$\vec{x}(t)$  - vetor deslocamentos

$\dot{\vec{x}}(t)$  - vetor velocidades

$\ddot{\vec{x}}(t)$  - vetor acelerações

$\vec{f}(t)$  - vetor de forças aplicadas

Após aplicar a Lei Fundamental da Dinâmica a matriz massa, rigidez e amortecimento viscoso do modelo representado na figura 3.4 ficam com a seguinte forma:

$$\mathbf{M} = \begin{bmatrix} m1 & 0 & 0 & 0 \\ 0 & m2 & 0 & 0 \\ 0 & 0 & m3 & 0 \\ 0 & 0 & 0 & m4 \end{bmatrix} \quad (3.4)$$

$$\mathbf{K} = \begin{bmatrix} k1 + k2 & -k2 & 0 & 0 \\ -k2 & k2 + k3 & -k3 & 0 \\ 0 & -k3 & k3 + k4 & -k4 \\ 0 & 0 & -k4 & k4 \end{bmatrix} \quad (3.5)$$

$$\mathbf{C} = \beta.\mathbf{M} + \gamma.\mathbf{K} \quad (3.6)$$

$\beta$  - coeficiente de amortecimento proporcional da massa

$\gamma$  - coeficiente de amortecimento proporcional da rigidez

Nesta secção foi feita uma explicação fundamental sobre a teoria de vibrações, na próxima serão introduzidos outros temas essenciais à medição de vibrações e que foram usados nos algoritmos apresentados no software desenvolvido.

## 3.2 Medição de vibrações

Nesta secção serão abordados alguns conceitos de vibração. O intuito da abordagem destes temas é clarificar e fornecer o conhecimento necessário para a correta utilização do site de benchmarking. O website terá disponível um conjunto de dados de estruturas modeladas numericamente, modelo numérico, e outras estruturas testadas em laboratório, modelo real.

### 3.2.1 Conceito de Vibração Harmónica

Qualquer sistema de vibração possui algum movimento, velocidade e aceleração. Estas propriedades quando medidas não se mantêm estáticas no tempo, sendo sempre possível verificar oscilações ou variações ao longo do tempo.

Desta forma podemos introduzir o conceito de *Vibração Harmónica*. A *Vibração Harmónica* é o movimento de um ponto oscilando em torno de um ponto de referência. O número de vezes que ocorre o movimento completo em determinado tempo é denominado por Frequência.

A matemática pode descrever este movimento através da seguinte equação.

$$x(t) = A.\cos(2\pi ft + \alpha) \quad (3.7)$$

Onde:

$x(t)$  - Função que descreve o movimento

$A$  - Amplitude máxima do movimento oscilatório

$f$  - Frequência de oscilação (em Hertz)

$t$  - Tempo (em segundos)

$\alpha$  - Ângulo de desfasamento do oscilação (em radianos)

### 3.2.2 Conceito de FFT

Este movimento oscilatório quando é captado por um sensor pode ser processado por computador de forma a descobrir quais são as suas propriedades, Amplitude, frequência, ângulo de defasamento. Uma das formas mais eficazes de descobrir estas propriedades é utilizar a Transformada Rápida de Fourier (FFT). Esta função permite representar no domínio da frequência, as contribuições ou amplitudes de várias frequências num sinal composto. Permite por exemplo compreender qual a frequência que mais contribui para um sinal. Vejamos o seguinte exemplo da próxima figura 3.5. Assumindo que temos uma oscilação harmónica no domínio do tempo com uma determinada Amplitude e Frequência e aplicamos a função FFT. Essa oscilação é agora representada não no domínio do tempo mas sim no domínio da frequência. Contudo uma estrutura real vibra com varias frequências simultaneamente, tal como demonstra a figura 3.6.

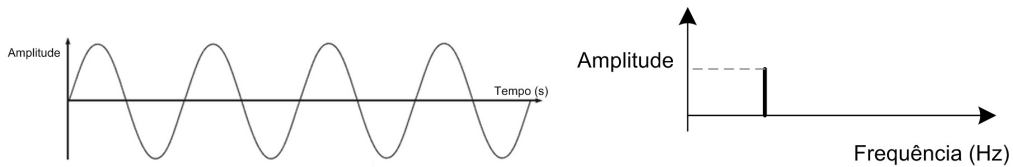


Figura 3.5: Oscilação harmónica em frequência

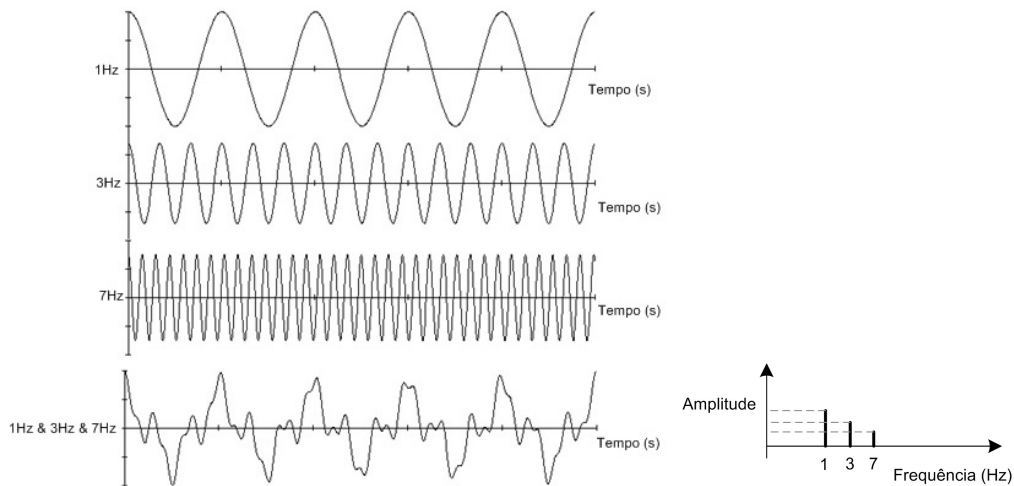


Figura 3.6: Oscilação harmónica composta

Até agora verificamos que podemos transformar uma estrutura real num modelo, podemos dividir este modelo em varias partes de forma a obter uma representação mais realista da estrutura em estudo. Sabemos também que ao discretizar uma estrutura podemos saber qual é o comportamento, deslocamento, velocidade e aceleração em cada nó.

Com a introdução destes conceitos podemos imaginar e criar muitas formas de análise a estruturas, tais como identificação de dano e localização de dano.

### 3.2.3 Conceito de FRF

Outro tema importante para este trabalho é o conhecimento da *Função de Resposta em Frequência (FRF)*. Esta função é explicada neste trabalho pois o seu conhecimento é fundamental para a compreensão dos algoritmos demonstrados.

Das secções anteriores neste capítulo foram explicados os conceitos de oscilação harmónica no tempo e de como se obtém essa oscilação no domínio da frequência. Numa estrutura de N graus de liberdade, tal como mostra a próxima figura, podemos obter um sinal no tempo para cada nó (N1, N2... N6).

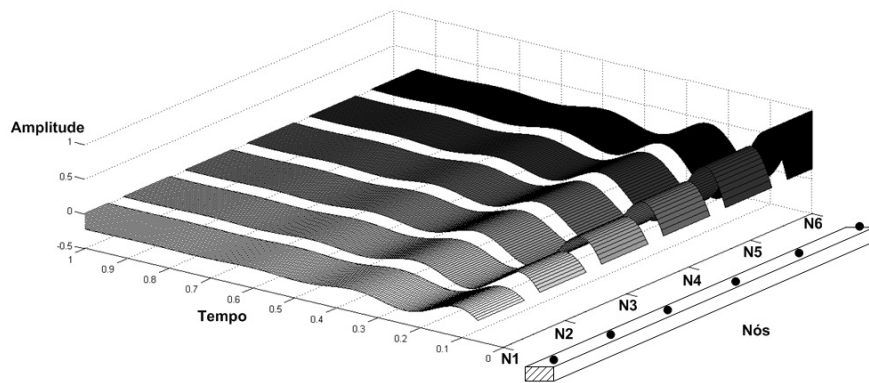


Figura 3.7: Modelo de viga discretizada vs sinal no tempo

Após a aquisição de um sinal para cada nó podemos executar a função FFT e obter uma representação no domínio da frequência, tal como mostra a próxima figura.

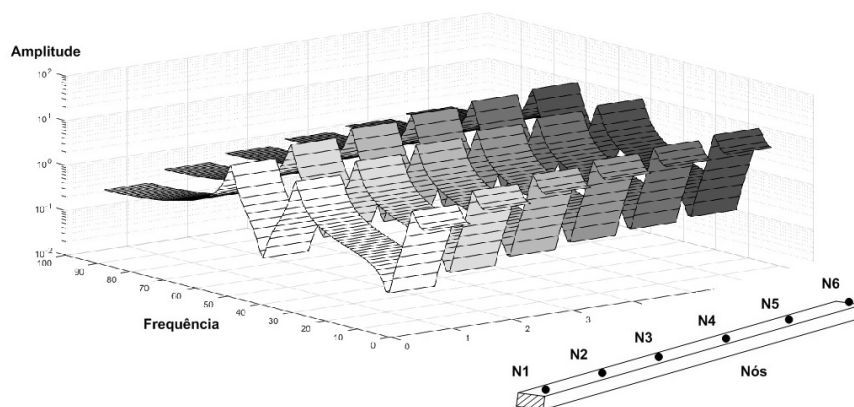


Figura 3.8: Modelo de viga discretizada vs resposta em frequência

Podemos verificar que cada reposta no tempo de cada nó possui uma identidade única e que subsequentemente também tem uma resposta em frequência de identidade única.

Contudo as amplitudes e as frequências naturais das repostas de cada nó podem variar bastante consoante seja a *Amplitude*, *Frequência* e *localização* da força de excitação aplicada na estrutura, deixando de ser possível executar determinados algoritmos de identificação e localização de dano em estruturas . Por esta razão é introduzido o conceito de *Função de Resposta em Frequência (FRF)*.

Esta função FRF ou matriz receptância constitui o modelo de resposta de um sistema.

Cada elemento da FRF ou matriz receptância representa a resposta na coordenada  $i$  a uma força unitária aplicada na coordenada  $j$ .

Vejamos as seguintes equações que demonstram a sua execução matemática. Consideremos o seguinte sistema forçado por uma força harmónica:

$$\mathbf{M}\ddot{\vec{x}}(t) + \mathbf{C}\dot{\vec{x}}(t) + \mathbf{K}\vec{x}(t) = \vec{f}(t) \quad (3.8)$$

após a aplicação de calculo diferencial sabemos que:

$$\vec{x}(t) = \vec{\mathbf{X}}e^{j\omega t} \quad (3.9)$$

$$\dot{\vec{x}}(t) = j\omega\vec{\mathbf{X}}e^{j\omega t} \quad (3.10)$$

$$\ddot{\vec{x}}(t) = (j\omega)^2\vec{\mathbf{X}}e^{j\omega t} = -\omega^2\vec{\mathbf{X}}e^{j\omega t} \quad (3.11)$$

$$\vec{f}(t) = \vec{\mathbf{F}}e^{j\omega t} \quad (3.12)$$

então resolvendo a equação 3.8 obtemos a seguinte expressão final da matriz da FRF:

$$\alpha_{i,j}(\omega) = \frac{\vec{\mathbf{X}}}{\vec{\mathbf{F}}} = [\mathbf{K} + j\omega\mathbf{C} - \omega^2\mathbf{M}]^{-1} \quad (3.13)$$

Onde:

$\mathbf{M}$  - matriz massa

$\mathbf{K}$  - matriz rigidez

$\mathbf{C}$  - matriz amortecimento viscoso

$\omega$  - Frequência angular

$\vec{\mathbf{X}}$  - Resposta em frequência do deslocamento, velocidade ou aceleração

$\vec{\mathbf{F}}$  - Resposta em frequência da força de excitação

$\alpha_{i,j}$  - Matriz receptância/FRF

A FRF é uma das funções mais usadas na Engenharia, pois os resultados medidos são geralmente bastante fiéis para que se possa efetuar estudos e análises em estruturas, não sendo necessário criar modelos muito complexos que requerem muitos recursos, humanos e tecnológicos, e tempos de execução muito longos.

### 3.2.4 Cálculo de Respostas no Tempo

Foram incluídos alguns sinais no tempo de modelos FEM de geometria simples de forma a que o utilizador possa testar os seus métodos sem os típicos erros associados ao mau funcionamento do sensor de medida ou equipamento de teste deficitário. A razão pela qual foram adicionados estes sinais no tempo prende-se com o objetivo primordial deste software, ou seja, *permitir a comparação de diferentes técnicas de identificação de dano em estruturas reais*.

Para tal são necessárias medições com vários tipos de medidas, isto é, deslocamento, velocidade, aceleração, força, entre outras medições.

Foram calculados e simulados *Sinais no Tempo* dos modelos FEM fornecidos. Isto permite que a validação dos algoritmos seja feita sequencialmente, desde um modelo numérico até a um modelo real.

#### Métodos de cálculo:

O cálculo das repostas no tempo de um modelo de N graus de liberdade, requer a utilização de um método numérico de solução de equações diferenciais.

A próxima equação, que define o modelo espacial de um sistema harmónico amortecido forçado de 1 grau de liberdade, é uma Equação Diferencial Ordinária (ODE). Entenda-se que uma ODE é uma equação diferencial que envolve uma ou mais derivadas da função desconhecida e é ordinária pois todas as derivadas são definidas relativamente a uma só variável.

Portanto transformemos a ODE escalar de 2ª ordem(3.14), numa ODE vectorial de 1ª ordem.

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = F(t) \quad (3.14)$$

$$\ddot{x}(t) = -\frac{c}{m}\dot{x}(t) - \frac{k}{m}x(t) + \frac{F(t)}{m} \quad (3.15)$$

Será necessário efectuar uma mudança de variavel de forma a transformar uma ODE de ordem  $n$  numa ODE de 1ª ordem. Vejamos:

$$u_1(t) = x(t)$$

$$u_2(t) = \dot{x}(t)$$

Passando para forma de vector:

$$\begin{bmatrix} \dot{u}_1(t) \\ \dot{u}_2(t) \end{bmatrix} = \begin{bmatrix} u_2(t) \\ -\frac{c}{m}u_2(t) - \frac{k}{m}u_1(t) + \frac{F(t)}{m} \end{bmatrix} \quad (3.16)$$

A expressão 3.16 pode ser simplificada através da criação da matriz estado  $[A]$  e

das variáveis estado  $[\vec{u}(t)]$ .

$$\begin{bmatrix} \dot{u}_1(t) \\ \dot{u}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \cdot \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{F(t)}{m} \end{bmatrix} \equiv [\dot{\vec{u}}(t)] = [\mathbf{A}][\vec{u}(t)] + \begin{bmatrix} 0 \\ \frac{F(t)}{m} \end{bmatrix} \quad (3.17)$$

Para resolver um sistema de  $N$  graus a Matriz Estado  $[A]$  e o vector das Variáveis Estado  $[\vec{u}(t)]$ , podem ser rescritos da seguinte forma:

$$[A] = \begin{bmatrix} \emptyset & I \\ -\frac{K}{M} & -\frac{C}{M} \end{bmatrix} \quad (3.18)$$

$$[\vec{u}(t)] = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \dots \\ u_{N \times 2}(t) \end{bmatrix} \quad (3.19)$$

Assim sendo podemos dizer que a equação 3.17 é do tipo,  $\vec{u}(t) = f(t, \vec{u}(t))$ , podendo ser resolvida por qualquer algoritmos de resolução de equações diferenciais tais como, Runge-Kutta, Adams, BDF, entre outros.

As repostas no tempo foram obtidas através do algoritmo híbrido *AdamsBDF*, disponível no software de cálculo *MathCad*<sup>®</sup> da *PTC*.

### 3.3 Métodos de identificação de dano selecionados

Durante o desenvolvimento do site foram adicionados alguns métodos de identificação de dano. Estes métodos permitiram testar e validar a funcionalidade do site. As técnicas selecionadas dividem-se em dois tipos:

1. **Deteção de dano** - Os métodos de identificação de dano são usados para avaliar o nível global da estrutura. Estas técnicas avaliam a estrutura na sua globalidade, não sendo permitido saber o local exato do dano. Estes métodos são geralmente usados numa fase inicial da avaliação da integridade estrutural de uma estrutura.

Os algoritmos são:

- (a) FRF Relative Damage Quantification Indicator [11]
- (b) FRF Difference Index [12]
- (c) FRF Transmissibility Damage Indicator

2. **Localização do dano** - Os métodos de localização do dano servem, tal como o nome indica, para identificar dano numa localização geométrica da estrutura.

Os algoritmos são:

- (a) FRF Based Mode Shape Curvature [13]
- (b) FRF Damage Index [13]

### 3.3.1 Algoritmos selecionados

Nesta secção serão apresentados os algoritmos selecionados. A sua selecção foi feita tendo em conta os conhecimentos aprofundados nestes algoritmos por parte do orientador de mestrado Prof. Rui Pedro Chedas Sampaio.

#### Identificação de dano

##### 1. FRF Relative Damage Quantification Indicator

Neste método é usada a função transferência (FRF) de uma estrutura para executar o critério RVAC (Response Vector Assurance Criterion). Após o cálculo do RVAC é aplicado o DRQ (Detection and Relative Damage Quantification).

Na secção 2 do documento [11] é enunciado o modelo teórico. Após extrair as FRF's (matrizes  $^* \alpha_{ij}(\omega) = \frac{X_{ij}}{F_{ij}}$ ) da estrutura sem dano e com dano, ou seja,  $\alpha_{ij}(\omega)$  e  $^d \alpha_{ij}(\omega)$ , respetivamente. É necessário executar o critério RVAC (Response Vector Assurance Criterion) para todas as estruturas:

$$RVAC_d(\omega) = \frac{|\sum_{i=1}^N [^d \alpha_{ij}(\omega) \cdot \overline{\alpha_{ij}(\omega)}]|^2}{\sum_{i=1}^N [^d \alpha_{ij}(\omega) \cdot \overline{^d \alpha_{ij}(\omega)}] \cdot \sum_{i=1}^N [\alpha_{ij}(\omega) \cdot \overline{\alpha_{ij}(\omega)}]} \quad (3.20)$$

Após o calculo do RVAC é aplicado o DRQ (Detection and Relative Damage Quantification):

$$DRQ_d = \frac{RVAC_d(\omega)}{N_\omega} \quad (3.21)$$

O valor de  $DRQ_d$  está compreendido entre 0 e 1.

Nomenclatura:

$RVAC_d(\omega)$  - Response Vector Assurance Criterium

$\alpha_{ij}(\omega)$  - Matriz receptância da estrutura saudavel.

$^d \alpha_{ij}(\omega)$  - Matriz receptância da estrutura danificada.

$i$  - coordenada da resposta medida

$j$  - coordenada da força medida

$\omega$  - frequência

$d$  - nível de dano

$N_\omega$  - número de frequências

## 2. FRF Difference Index

Na secção 3 do documento [12] são referidos vários métodos de identificação de dano, o método fornecido no site FRF Difference Index (ou Based Mode Shape, FRF MS) foi o proposto, pois a sua aplicação é relativamente simples quando comparado com os restantes aqui apresentados.

Na secção 3.1 do documento [12] é enunciado o modelo teórico. Este método também é baseado na matriz receptância (FRF- $\alpha_{ij}(\omega) = \frac{X_{ij}}{F_{ij}}$ ) da estrutura com e sem dano ( $\alpha_{ij}(\omega)$  e  ${}^d\alpha_{ij}(\omega)$ ).

O método é definido com as seguintes expressões matemáticas:

$$\Delta\alpha_{ij}(\omega) = |{}^d\alpha_{ij}(\omega) - \alpha_{ij}(\omega)| \quad (3.22)$$

$$FRFMS_i = \sum_{\omega} \sum_j \Delta\alpha_{ij}(\omega) \quad (3.23)$$

Nomenclatura:

$FRFMS_i$  - Frequency Response Function Based - Mode Shape

$\alpha_{ij}(\omega)$  - Matriz receptância da estrutura saudável.

${}^d\alpha_{ij}(\omega)$  - Matriz receptância da estrutura danificada.

$i$  - coordenada da resposta medida

$j$  - coordenada da força medida

$\omega$  - frequência

$d$  - nível de dano

$\Delta\alpha_{ij}(\omega)$  - variação dos modos entre estrutura sã e danificada

## 3. FRF Transmissibility Damage Indicator (TDI)

Este método aqui apresentado foi fornecido pelo professor Doutor Rui Pedro Chedas Sampaio. O seu modelo teórico tem como base o critério RVAC usado no método FRF Relative Damage Quantification Indicator. São usadas as matrizes receptância (FRF) para executar o algoritmo.

$$TDI = \frac{\sum_{\omega} \frac{|\sum_{i=1}^N [\frac{{}^d\alpha_{i,j}(\omega)}{d_{\alpha_{i+1,j}(\omega)}} \cdot \frac{\overline{\alpha_{i,j}(\omega)}}{\alpha_{i+1,j}(\omega)}]|^2}{\sum_{i=1}^N [\frac{{}^d\alpha_{i,j}(\omega)}{d_{\alpha_{i+1,j}(\omega)}} \cdot \frac{\overline{\alpha_{i,j}(\omega)}}{\alpha_{i+1,j}(\omega)}] \cdot \sum_{i=1}^N [\frac{\alpha_{i,j}(\omega)}{\alpha_{i+1,j}(\omega)} \cdot \frac{\overline{\alpha_{i,j}(\omega)}}{\alpha_{i+1,j}(\omega)}]}}{\omega_{max} + 1} \quad (3.24)$$

## Localização do dano

### 1. FRF Based Mode Shape Curvature

Este algoritmo de localização de dano é o chamado "*Curvature Method*". Este modelo foi inicialmente proposto por Pandey [13] e ele afirma que é possível saber a localização do dano numa estrutura através da sua a curvatura. O algoritmo aqui apresentado usa as matrizes FRF para o calculo, vejamos então:

A próxima expressão possibilita o calculo da curvatura da estrutura.

$$\alpha''(\omega)_i = \frac{(\alpha(\omega)_i - 2\alpha(\omega)_i + \alpha(\omega)_i)}{h} \quad (3.25)$$

Os resto do modelo é calculado com a seguinte expressão:

$$S_i = \Delta\alpha''_i = \Sigma_{\omega} |\alpha''(\omega)_i - \alpha''_d(\omega)_i| \quad (3.26)$$

Onde  $i$  representa a localização de cada sensor. [Para mais detalhes sobre o modelo ver o documento [13]]

### 2. FRF Damage Index

O ultimo indicador fornecido é o "*Damage Index*". Foi inicialmente proposto por Stubbs and Kim [13].

Tal como no algoritmo anterior é necessário calcular previamente a curvatura da estrutura através da equação 3.25.

O indicador é definido com as seguintes expressões:

$$\beta_{ij}(\omega) = \frac{({}^d\alpha''_{ij}(\omega) + \Sigma_{i=1}^N ({}^d\alpha''_{ij}(\omega)))\Sigma_{i=1}^N (\alpha''_{ij}(\omega))}{(\alpha''_{ij}(\omega) + \Sigma_{i=1}^N (\alpha''_{ij}(\omega)))\Sigma_{i=1}^N ({}^d\alpha''_{ij}(\omega))} \quad (3.27)$$

$$FRFDI_i = \Sigma_{\omega} \Sigma_j (\beta_{ij}(\omega)) \quad (3.28)$$

No apêndice A são fornecidos os códigos de programação em código MATLAB para cada um dos métodos/algoritmos referidos nesta dissertação.

# Capítulo 4

## Site de Benchmarking

### 4.1 Estrutura e funcionamento

Nesta secção será apresentada a estrutura do site e os seus menus para a execução das técnicas/algoritmos. O site está dividido em 6 áreas, tal como mostra a figura seguinte.

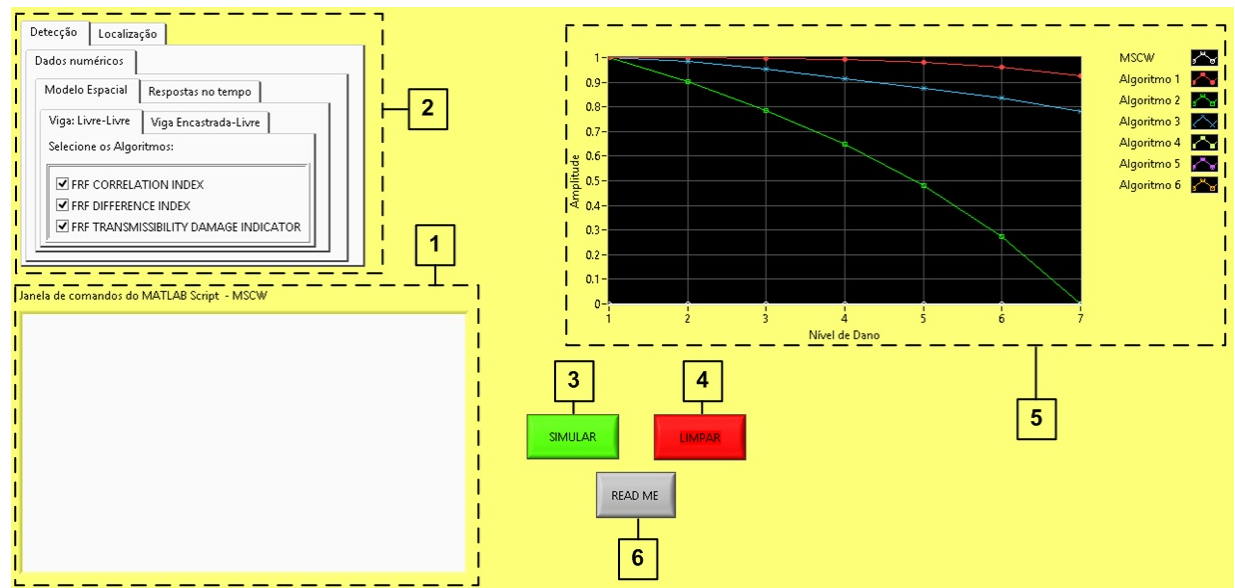


Figura 4.1: painel frontal do site

Cada uma das áreas(1 a 6) tem a sua função. Estas estão explicadas a seguir:

**Área:**

1. **MSCW** - O Utilizador poderá executar o seu código de programação nesta janela, desenvolvido na linguagem *MATLAB*. Se o código de programação estiver incorreto é lançado uma mensagem de erro " *O seu código de MATLAB Script não está correto. Para mais informação ler o Manual*". O programa que permite correr o código é o *MATLAB Script* da *National Instruments*. Esta aplicação que permite executar código *MATLAB*, contudo poderá não ter algumas das funções e funcionalidades do software proprietário original da *MathWorks*. *MSCW* é o acrónimo para *Matlab Script Command Window*.
2. **Seleção de algoritmos** - Nesta área é permitido seleccionar os algoritmos que pretendemos comparar e avaliar. Podemos seleccionar o tipo de análise (detecção ou localização), tipo de dados (numéricos ou experimentais) e o tipo de estrutura mecânica. Existem vários e diferentes algoritmos para cada um dos sub menus.
3. **Simular** - Quando o botão "Simular" é pressionado, o software executa o código dos algoritmos seleccionados do *Área 2* e o código introduzido no *Área 1*.
4. **Limpar** - Este botão serve para limpar os resultados apresentados no gráfico do *Área 5*.
5. **Gráfico** - Aqui são apresentados os resultados da janela de comandos (*MSCW*) com os resultados do algoritmos seleccionados no *Área 2*. Todos os valores apresentados aqui, calculados pelos algoritmos do *Área 1* e *2*, são normalizados em relação ao seu valor máximo, passando para a unidade, ou seja, 1.
6. **READ ME** - Quando o *Área 6* é pressionado o software lança um Manual em formato PDF com instruções para o utilizador poder esclarecer as suas dúvidas. Aqui são também fornecidos alguns exemplos de algoritmos executados no site.

### 4.1.1 Exemplo 1

Vejam os um exemplo da utilização do site.

O primeiro passo é requisitar o controlo da *VI*, pois o website foi desenvolvido com a ferramenta "Web Publisher", que pertence ao software *Labview* da *National Instruments*.

Para requisitar controlo da *VI*, tal como mostra a proxima figura, é necessário pressionar no painel frontal com botão direito do rato e seleccionar a opção "*Request Control of VI*". Após adquirir controlo do website teremos de escolher que

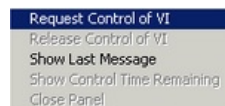


Figura 4.2: Requesitar controlo da VI

tipo de análise de *identificação de dano* pretendemos fazer, isto é, ***deteção de dano*** ou ***localização do dano***.

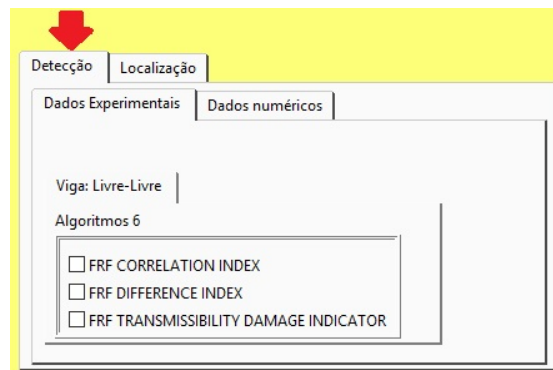


Figura 4.3: Seleccionar tipo de análise

Após seleccionar o tipo de análise pretendida (ver imagem anterior), neste exemplo a ***deteção de dano***, teremos de escolher que tipo de dados queremos avaliar e existem dois tipo:

1. **Dados Experimentais** - Dados obtidos através de medições em estruturas reais.
2. **Dados Numéricos** - Dados obtidos através da criação de modelos teóricos.

Após seleccionar o tipo de dados, neste exemplo os "***Dados numéricos***", verificamos que existem dois tipos de dados numéricos:

1. **Modelo Espacial** - São fornecidas as matrizes Massa, Rigidez e Amortecimento de uma estrutura.
2. **Sinal no tempo** - São fornecidas as respostas no tempo de vários pontos da estrutura quando esta é excitada por uma força externa. O sinal no tempo da força de excitação também é fornecido.

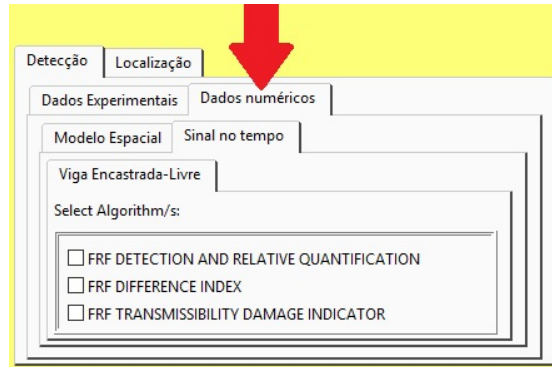


Figura 4.4: Selecionar tipo de dados

Após seleccionar a opção **Sinal no Tempo**, verificamos que tipos de estruturas existem, neste caso existe apenas uma estrutura (**Viga Encastrada-Livre**).

A partir deste momento podemos dar início ao processo de comparação de técnicas de **Identificação de Dano**.

Na **área 2** figura 4.1 podemos seleccionar alguns algoritmos nativos do site e na **área 1** podemos introduzir o algoritmos que queremos testar. Este tipo de abordagem possui várias vantagens para o investigador que está a usar o website:

1. Compara vários algoritmos ao mesmo tempo.
2. A confidencialidade dos algoritmos não é posta em causa.
3. O tempo de computação dos algoritmos é relativamente baixo.

Vejamos o exemplo da próxima figura.

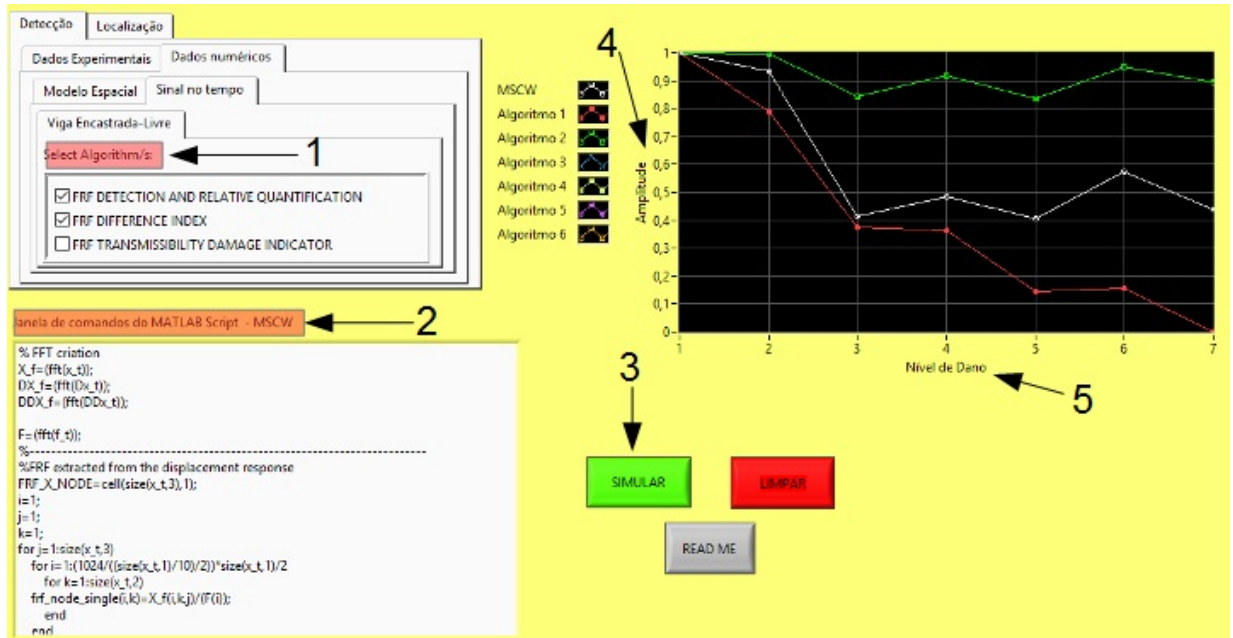


Figura 4.5: Exemplo 1 - painel frontal

Após selecionar dois algoritmos nativos (área 1 - fig. 4.5) do website, neste caso o "FRF detection and relative quantification" (algoritmo 1) e "FRF difference index" (algoritmo 2) e introduzir o algoritmo de teste na janela de comandos MSCW (área 2 - fig. 4.5) selecionamos o botão verde (área 3 - fig. 4.5) para *Simular*. Após alguns segundos de processamento de dados verificamos que três indicadores branco (MSCW), vermelho (algoritmo 1) e verde (algoritmo 2) aparecem no gráfico, sendo o branco relativo ao algoritmo de teste, o vermelho respectivo ao algoritmo "FRF detection and relative quantification" e o verde do algoritmo "FRF difference index". No eixo das abscissas (área 5 - fig. 4.5) encontra-se o nível de dano da estrutura e no eixo das ordenadas (área 4 - fig. 4.5) é apresentada a quantificação numérica do indicador de cada algoritmo.

Neste exemplo torna-se evidente que os três algoritmos apresentam resultados bastante diferentes, sendo o algoritmo 1 aquele que aparenta ser mais sensível ao dano.

### 4.1.2 Exemplo 2

Vejam os um exemplo relativo à *Localização do Dano*.

Tal como no exemplo anterior é necessário escolher o tipo de dados e a estrutura a analisar. Após a selecção dos algoritmos (**área 1** - fig. 4.6) é pressionado o botão para "Simular" (**área 3** - fig. 4.6). De seguida são apresentados os dois indicadores no gráfico à direita, a verde o *algoritmo 1* e a vermelho o *algoritmo 2*. Estes indicadores representam as localizações dos danos existentes na estrutura, sendo o eixo das abcissas (**área 5** - fig. 4.6) o eixo que define a localização do dano na estrutura e no eixo das ordenadas (**área 4** - fig. 4.6) é representada a severidade do dano na respectiva localização.

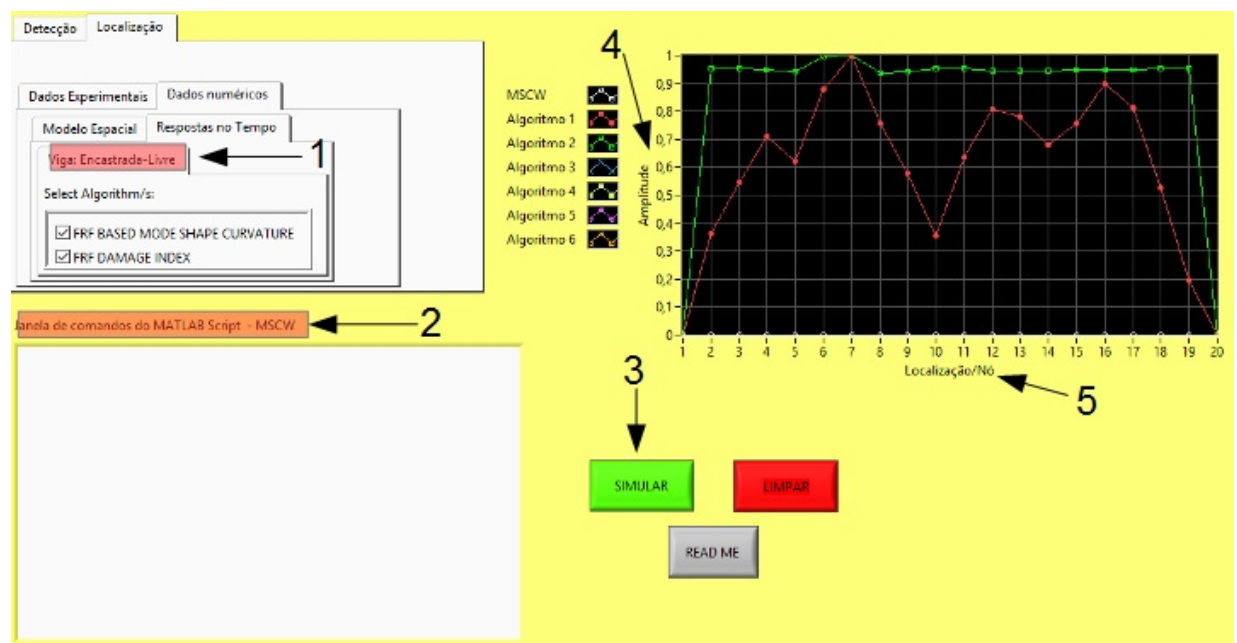


Figura 4.6: Exemplo 2 - painel frontal

No gráfico da figura 4.6 é possível verificar que os resultados dos dois algoritmos são bastante díspares entre si. O *algoritmo 1* aparenta ter uma maior sensibilidade ao dano, contudo, sabendo que o dano se encontra entre o *nó 6* e *7*, o *algoritmo 2* aparenta ser mais assertivo, pois apresenta apenas um pico entre o *nó 6* e *7*.

## 4.2 Modelos e dados fornecidos

Os cálculos e comparações entre métodos são efetuados em consola usando o motor de cálculo do *MATLAB*. A aplicação de cálculo é o "MATLAB Script" do fabricante de software da *LABVIEW* da *National Instruments*.

*Dados MATLAB:*

### 1. Modelo Numérico

- (a) **Modelo Espacial:** São fornecidos três (3) arrays multidimensionais, **M**, **K** e **C**, para o *Modelo Espacial*. Cada array multidimensional, é constituído por sete(7) matrizes quadradas de dimensão  $D$ .

Cada estrutura tem o seu numero de nós  $n$ , em cada nó existem três (3) graus de liberdade que definem, deslocamento em  $x$ , deslocamento e  $y$  e rotação em  $z$ , resultado em uma matriz quadrada de dimensão  $D = n \times 3$ .

Definição dos índices dos arrays multidimensionais [L,C,E]:

- **L** - A primeira dimensão da matriz é a **Linha**.
- **C** - A segunda dimensão da matriz é a **Coluna**.
- **E** - A terceira dimensão da matriz define a estrutura selecionada, ou seja, 1 é igual há estrutura sem dano e 7 é referente a estrutura com 60% de dano.

- (b) **Resposta no Tempo:** Foram simulados sinais no tempo para cada nó, em **i**, quando a estrutura é sujeita a um impacto num determinado nó, em **j**.

São disponibilizadas três arrays multidimensionais e um vetor, isto é, deslocamento de cada nó ao longo do tempo  $\mathbf{x}_i(t)$ , velocidade de cada nó ao longo do tempo  $\dot{\mathbf{x}}_i(t)$ , aceleração de cada nó ao longo do tempo  $\ddot{\mathbf{x}}_i(t)$  e a força (Newton) de impacto num nó específico ao longo do tempo  $\mathbf{f}_j(t)$ , respetivamente.

Definição dos índices dos arrays multidimensionais para, descolamento, velocidade e aceleração [L,C,E]:

- **L** - A primeira dimensão da matriz é a **Linha**. Aqui é selecionado o instante da resposta no tempo.
- **C** - A segunda dimensão da matriz é a **Coluna**. Aqui é definido o nó da estrutura simulada.

- **E** - A terceira dimensão da matriz define a estrutura selecionada, ou seja, 1 é igual há estrutura sem dano e 7 é referente a estrutura com 60% de dano.

Os dados disponibilizados no site são:

## 1. Modelo Numérico

### (a) Modelo Espacial

#### i. Viga Livre-Livre

- N° de Elementos - 19
- N° de Nós - 20
- Elemento com dano - 3 (terceiro)
- Níveis de dano - 6 (10%, 20%, 30%, 40%, 50%, 60%)
- Comprimento - 500 mm
- Secção Transversal - 35 mm(z) por 5 mm(y)
- Densidade relativa - 2.7
- Modulo de Young - 69 GPa

Dados disponíveis (constantes MATLAB):

- **M** : Matriz Massa - [60x60x7]
- **K** : Matriz Rigidez - [60x60x7]
- **C** : Matriz Amortecimento - [60x60x7]

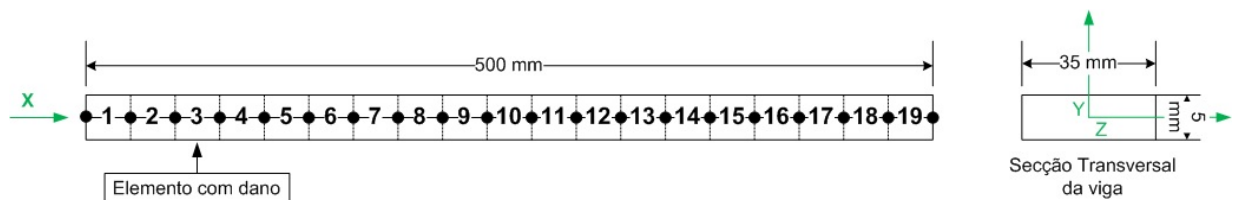


Figura 4.7: Representação gráfica da Viga Livre - Livre

ii. **Viga Encastrada-Livre**

- N° de Elementos - 20
- N° de Nós - 21
- Elemento com dano - 7 (Sétimo)
- Níveis de dano - 6 (10%, 20%, 30%, 40%, 50%, 60%)
- Comprimento - 500 mm
- Secção Transversal - 35 mm(z) por 5 mm(y)
- Densidade relativa - 2.7
- Modulo de Young - 69 GPa

Dados disponíveis (constantes MATLAB):

- **M** : Matriz Massa - [63x63x7]
- **K** : Matriz Rigidez - [63x63x7]
- **C** : Matriz Amortecimento - [63x63x7]

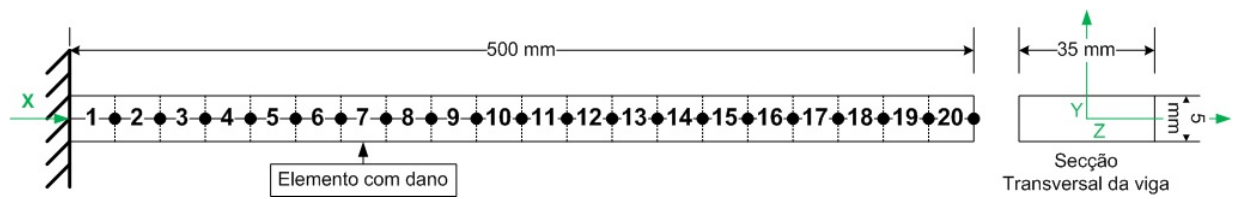


Figura 4.8: Representação gráfica da Viga Encastrada - Livre

(b) **Modelo Respostas no Tempo**

i. **Viga Encastrada-Livre**

- N° de Elementos - 20
- N° de Nós - 21
- Elemento com dano - 7 (Sétimo)
- Níveis de dano - 6 (10%, 20%, 30%, 40%, 50%, 60%)
- Comprimento - 500 mm
- Secção Transversal - 35 mm(z) por 5 mm(y)
- Densidade relativa - 2.7
- Modulo de Young - 69 GPa
- Força aplicada:
  - Nó : 19 (ver próxima figura)
  - Tipo de impulso: Meio-seno
  - Amplitude máxima: 100 g
  - Duração do impulso: 0.00098 segundos
- Sinal no tempo:
  - Duração do sinal - 10 segundos
  - Numero de pontos: 32768

Dados disponíveis (constantes MATLAB), sinais no tempo:

- $x.t$  : Deslocamento - [32768x21x7]
- $Dx.t$  : Velocidade - [32768x21x7]
- $DDx.t$  : Aceleração - [32768x21x7]
- $f.t$  : Força aplicada - [32768x1]

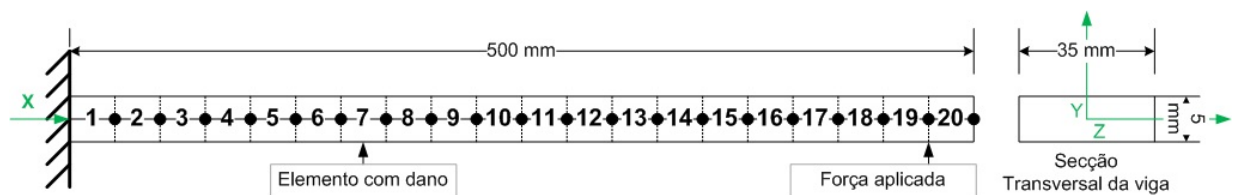


Figura 4.9: Representação gráfica da Viga Encastrada - Livre, do modelo resposta no tempo

## 2. Dados Experimentais

### (a) Modelo Respostas no Tempo

#### i. Viga Livre-Livre

Um grupo de investigadores portugueses realizou um teste experimental a uma viga de aço. [14]

A viga foi suspensa por dois cabos rígidos de forma a simular as condições de fronteira do tipo "Livre-Livre". Foram adicionados vinte e três (23) acelerómetros igualmente espaçados ao longo da estrutura, tal como mostra a próxima figura.

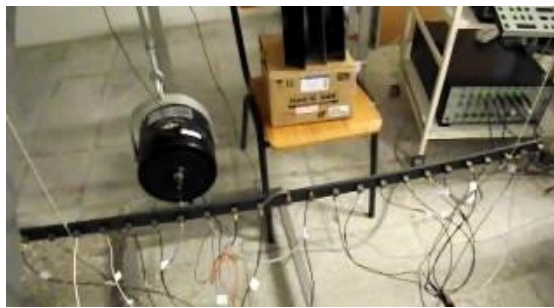


Figura 4.10: setup de testes [14]

A estrutura foi excitada por um vibrador (shaker) da *Bruel&Kjaer*. O dano na estrutura foi simulado com a redução de secção transversal, tal como mostra a proxima figura.



Figura 4.11: redução de secção transversal [14]

- N° de acelerómetros - 23
- Elemento com dano - Entre acelerómetros 15 e 16
- Níveis de dano - 9
- Comprimento - 1002 mm
- Secção Transversal - 35 mm(z) por 6 mm(y)
- Massa - 1.740 Kg
- Material - Aço carbono comum
- Força aplicada:
  - Localização : Acelerómetro 3
  - Tipo de vibração: pseudo-random
- Sinal no tempo:
  - Duração do sinal - 4 segundos
  - Numero de pontos: 8192

Dados disponíveis (constantes MATLAB), sinais no tempo:

- $DDx.t$  : Aceleração - [8192x24x10]
- $f.t$  : Força aplicada - [8192x1]

## Capítulo 5

### Conclusões

Este trabalho teve como o objectivo de criação de uma ferramenta multimédia, para a Internet, para comparação de algoritmos de identificação de dano em estruturas no âmbito do tema de SHM. O público alvo desta ferramenta é toda a comunidade científica, isto é, alunos, investigadores e revistas científicas.

Este website permite comparar várias estruturas e diferentes algoritmos simultaneamente. Irá também permitir demonstrar quais são as técnicas mais promissoras para cada tipo de estrutura.

Uma das grandes vantagens deste website/software relativamente a outros é a facilidade de utilização e obtenção de resultados, isso deve-se essencialmente devido à arquitectura de processamento do website e a simplicidade gráfica do painel de controlo. O utilizador apenas tem de se preocupar em desenvolver o seu algoritmo/técnica, implementar o seu algoritmo em código matlab e interpretar os resultados fornecidos pelo website.

O software com mais semelhanças desenvolvido por outras entidades é o SHM-Tools, é sem dúvida um software mais completo e mais robusto, contudo é mais complexo de operar, requer alguma formação e tempo de estudo.

A maturação das técnicas e dos algoritmos em SHM irá despoletar o desenvolvimento de novas tecnologias para manutenção e inspecção de estruturas. Estas técnicas permitem prolongar o tempo de operação das estruturas, sem que estas sejam um risco para a sociedade.

O desenvolvimento do Website foi feito com o software *Labview*. Também se criaram modelos espaciais, modelos de resposta de várias estruturas recorrendo às técnicas de modelação FEM. Toda a programação dos 5 algoritmos de detecção e localização foi feita em Matlab.

### **Trabalho Futuro**

Um dos objectivos deste trabalho era disponibilizar esta ferramenta na *WEB*, o que não foi cumprido. A publicação desta ferramenta na internet revelou-se ser uma tarefa mais complexa que o esperado.

Este website que foi desenvolvido funciona como uma base de dados de algoritmos e estruturas medidas, o que leva a necessidade de actualizações constantes ao longo do tempo, assim sendo será normal que a estrutura do website sofra alterações.

## Bibliografia

- [1] Kullaa, Jyrki; "SHM Benchmark Data Download Page."; Web. 02 Dec. 2015; <http://users.metropolia.fi/kullj/>
- [2] Li, Shunlong ; Li, Hui; Liu, Yang; Lan, Chengming; Zhou, Wensong; Ou, Jinping ; Structural Control and Health Monitoring; 2014; "*SMC structural health monitoring benchmark problem using monitored data from an actual cable-stayed bridge*"; Web. 02 Dec. 2015; <http://smc.hit.edu.cn>
- [3] SAMCO Final Report 2006; F04 Case Studies; "*Bridge Z24 - Switzerland*"; Web. 02 Dec. 2015; <http://bwk.kuleuven.be/bwm//IMAC/index.html>
- [4] Farrar, Charles R. ; February 19, 2010; The Bromilow Lecture; "*Structural Health Monitoring: An Engineering Grand Challenge for the 21st Century*"
- [5] "Science LANL Institutes Engineering Institute." LANL Institutes; Web. 29 Nov. 2015, <http://institute.lanl.gov/ei/dynamics-summer-school/>
- [6] Smart Structures Technology Laboratory. Web. 29 Nov. 2015, <http://sstl.cpanel.engr.illinois.edu/>
- [7] Farrar, Charles R. ; Worden, Keith ; 12 de Dezembro 2006; Philosophical Transactions of the Royal Society "*An introduction to structural health monitoring*"
- [8] Doebling, Scott W. ; Farrar, Charles R. ; Prime, Michael B.; Shevitz, Daniel W. ; Maio 1996; Literature Review Book; "*Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in Their Vibration Characteristics: A Literature Review*"
- [9] Sampaio, R.P.C.; Maio 2012; "*Structural Health Monitoring baseado na Medição de Vibrações*"
- [10] Rao, Singiresu S; "1.5 Classification of Vibration." Mechanical Vibrations; 5th ed. Reading, MA: Addison-Wesley, 1990. 16-17. Print.

- [11] R.P.C. Sampaio, N.M.M. Maia; Mechanical Systems and Signal Processing 23(2009)1855–1869; ”*Strategies for an efficient indicator of structural damage*”
- [12] Sampaio, R.P.C.; Maia, N.M.M.; Silva, J.M.M. ; E.A.M. Almas/Mechanical Systems and Signal Processing (2003) 17(3) 489–498; ”*DAMAGE DETECTION IN STRUCTURES: FROM MODE SHAPE TO FREQUENCY RESPONSE FUNCTION METHODS*”
- [13] Sampaio, R.P.C.; Maia, N.M.M.; Silva, J.M.M.; Journal of Sound and Vibration (1999) 226(5), 1029-1042; ”*DAMAGE DETECTION USING THE FREQUENCYRESPONSE-FUNCTION CURVATURE METHOD*”
- [14] Sampaio, R.P.C.; Maia, N.M.M.;Almeida, Raquel Albuquerque Soares Brás ;Urgueira, António V ; Elsevier Editorial System(tm) for Mechanical Systems and Signal Processing; MSSP14-347; 18 de Junho 2014; ”*A Simple Damage Detection Indicator using Operational Deflection Shapes*”

# Apêndice A

## Códigos MATLAB

### A.1 Algoritmos de Detecção de Dano

#### A.1.1 FRF DRQ - Modelo Espacial

```

1 % DETECTION AND RELATIVE QUANTIFICATION – DRQ (FRF ODS
   Correlation index)
2 %FRF calculation
3 %DEFINITION PARAMETERS
4 n_sample=2^10;%Number of samples
5 freq=n_sample/1;%Maximum Frequency definition (1024 Hz
   seems to be a very large frequency for the kind of
   structures that are analysed in this methods)
6 dfreq=freq/n_sample;%Frequency resolution
7 node=3;%Force Node for FRF
8 n=20;%Nr of nodes of the FEM structure
9 %-----
10 %FRF calculation for seven different structures , using the
   following matrices :
11 %M: Mass Matrix; K: Stiffness Matrix; C–Damping Matrix
12 %(1–no damage;2–10% damage;3–20% damage;
13 %4–30% damage;5–40% damage;6–50% damage;7–60% damage;)
14 FRF=cell ( size (M,3) ,1) ;
15 i=1;
16 j=1;
17 for j=1:size (M,3)
18     for i=1:n_sample
19         f(i)=dfreq*i ;

```

```

20     frf_single (:, :, i)=(inv((-2*pi*dfreq*i)^2*M(:, :, j))+
21         sqrt(-1)*(2*pi*dfreq*i)*C(:, :, j))+K(:, :, j));
22     end
23     FRF{j,1}=frf_single;
24 end
25 %Now we extract single FRF matrix for an exciting force at
26 %Node 3 for YY axis
27 FRF_NODE=cell(size(M,3),1);
28 i=1;
29 j=1;
30 k=1;
31 for j=1:size(M,3)
32     for i=1:n_sample
33         for k=1:n
34             frf_node_single(i,k)=(FRF{j,1}(k*3-1,node*3-1,i));
35         end
36     end
37     FRF_NODE{j,1}=frf_node_single;
38 end
39 %Aplicationg of FRF ODS Correlation Index
40 i=1;
41 j=1;
42 k=1;
43 for i=1:size(M,3)
44     for j=1:n_sample
45         for k=1:n
46             DRQ_1(i,j,k)=(((FRF_NODE{i,1}(j,k))*conj((FRF_NODE
47                 {1,1}(j,k)))));
48             DRQ_2(i,j,k)=((FRF_NODE{i,1}(j,k))*conj((FRF_NODE{i,1}(
49                 j,k)))));
50             DRQ_3(i,j,k)=((FRF_NODE{1,1}(j,k))*conj((FRF_NODE{1,1}(
51                 j,k)))));
52         end
53     end
54 end
55 i=1;
56 j=1;
57 for i=1:size(M,3)

```

```

55     for j=1:n_sample
56 DRQ_1_sum_f(i , j)=sum(DRQ_1(i , j , : ) ) ;
57     end
58 end
59 DRQ_1_sum=abs ( DRQ_1_sum_f ) . ^ 2 ;
60
61 i =1;
62 j =1;
63 for i =1: size (M, 3)
64     for j =1:n_sample
65 DRQ_2_sum ( i , j )=sum ( DRQ_2 ( i , j , : ) ) ;
66     end
67 end
68
69 i =1;
70 j =1;
71 for i =1: size (M, 3)
72     for j =1:n_sample
73 DRQ_3_sum ( i , j )=sum ( DRQ_3 ( i , j , : ) ) ;
74     end
75 end
76 %
77 i =1;
78 j =1;
79 for i =1: size (M, 3)
80     for j =1:n_sample
81 DRQ_4 ( i , j )=DRQ_1_sum ( i , j ) / (( DRQ_2_sum ( i , j ) * DRQ_3_sum ( i ,
82     j ) ) ) ;
83     end
84 end
85 i =1;
86 for i =1: size (M, 3)
87 DRQ_4_sum ( i )=sum ( DRQ_4 ( i , : ) ) ;
88 end
89 DRQ=DRQ_4_sum / max ( DRQ_4_sum ) ;
90
91
92 i =1;
93 for i =1: size (M, 3)

```

```
94 x(i)=DRQ(i);  
95 end
```

### A.1.2 FRF Difference Index - Modelo Espacial

```

1 % DETECTION AND RELATIVE QUANTIFICATION – (FRF ODS
   Difference index)
2 %FRF calculation
3 %DEFINITION PARAMETERS
4 n_sample=2^10;%Number of samples
5 freq=n_sample/1;%Maximum Frequency definition (1024 Hz
   seems to be a very large frequency for the kind of
   structures that are analysed in this methods)
6 dfreq=freq/n_sample;%Frequency resolution
7 node=3;%Force Node for FRF
8 n=20;%Nr of nodes of the FEM structure
9 %-----
10 %FRF calculation for seven different structures , using the
   following matrices :
11 %M: Mass Matrix; K: Stiffness Matrix; C–Damping Matrix
12 %(1–no damage;2–10% damage;3–20% damage;
13 %4–30% damage;5–40% damage;6–50% damage;7–60% damage;)
14 FRF=cell ( size (M,3) ,1) ;
15 i=1;
16 j=1;
17 for j=1:size (M,3)
18     for i=1:n_sample
19         f(i)=dfreq*i ;
20         frf_single (: , , i)=(inv((-2*pi*dfreq*i)^2*M(: , , j))+
           sqrt(-1)*(2*pi*dfreq*i)*C(: , , j)+(K(: , , j)))));
21     end
22     FRF{j,1}=frf_single ;
23 end
24 %Now we extract single FRF matrix for an exciting force at
   Node 3 for YY axis
25 FRF_NODE=cell ( size (M,3) ,1) ;
26 i=1;
27 j=1;
28 k=1;
29 for j=1:size (M,3)
30
31     for i=1:n_sample
32         for k=1:n
33             frf_node_single (i ,k)=(FRF{j ,1}(k*3-1,node*3-1,i));

```

```

34     end
35     end
36     FRF_NODE{j,1}=frf_node_single;
37 end
38 %Applicationg of FRF ODS Difference Index
39 dif=cell(size(M,3)-1,1);
40 i=1;
41 for i=1:size(M,3)-1
42     dif{i,1}=abs(FRF_NODE{i+1,1}-FRF_NODE{1,1});
43 end
44
45 i=1;
46 j=1;
47 for i=1:size(M,3)-1
48     for j=1:n
49         Sdif(i,j)=sum(dif{i,1}(:,j));
50     end
51 end
52
53 i=1;
54 j=1;
55 for i=1:size(M,3)-1
56     Mdif(i)=sum(Sdif(i,:));
57 end
58 i=1;
59 DI=1;
60 for i=2:size(M,3)
61     DI(i)=1-(Mdif(i-1)/max(Mdif));
62 end
63
64 i=1;
65 for i=1:size(M,3)
66     x(i)=DI(i);
67 end

```

### A.1.3 FRF TDI - Modelo Espacial

```

1  % DETECTION AND RELATIVE QUANTIFICATION – TDI (FRF
    Transmissibility Damage Indicator)
2  %FRF calculation
3  %DEFINITION PARAMETERS
4  n_sample=2^10;%Number of samples
5  freq=n_sample/1;%Maximum Frequency definition (1024 Hz
    seems to be a very large frequency for the kind of
    structures that are analysed in this methods)
6  dfreq=freq/n_sample;%Frequency resolution
7  node=3;%Force Node for FRF
8  n=20;%Nr of nodes of the FEM structure
9  %FRF calculation for seven different structures, using the
    following matrices :
10 %M: Mass Matrix; K: Stiffness Matrix; C-Damping Matrix
11 %(1-no damage;2-10% damage;3-20% damage;
12 %4-30% damage;5-40% damage;6-50% damage;7-60% damage;)
13 FRF=cell(size(M,3),1);
14 i=1;
15 j=1;
16 for j=1:size(M,3)
17     for i=1:n_sample
18         f(i)=dfreq*i;
19         frf_single(:, :, i)=(inv((-2*pi*dfreq*i)^2*M(:, :, j))+
                sqrt(-1)*(2*pi*dfreq*i)*C(:, :, j)+(K(:, :, j))));
20     end
21     FRF{j,1}=frf_single;
22 end
23 %Now we extract single FRF matrix for an exciting force at
    Node 3 for YY axis
24 FRF_NODE=cell(size(M,3),1);
25 i=1;
26 j=1;
27 k=1;
28 for j=1:size(M,3)
29
30     for i=1:n_sample
31         for k=1:n
32             frf_node_single(i,k)=(FRF{j,1}(k*3-1,node*3-1,i));
33         end

```

```

34     end
35     FRF_NODE{j,1}=frf_node_single;
36 end
37 %Applicationg of FRF Transmissibility Damage Indicator
38 i=1;
39 j=1;
40 k=1;
41 for i=1:size(M,3)
42     for j=1:n_sample
43         for k=1:n-1
44             TDI_1(i,j,k)=((FRF_NODE{i,1}(j,k))*conj((FRF_NODE{1,1}(
45                 j,k))))/((FRF_NODE{i,1}(j,k+1))*conj((FRF_NODE{1,1}(
46                 j,k+1))));
47             TDI_2(i,j,k)=((FRF_NODE{i,1}(j,k))*conj((FRF_NODE{i,1}(
48                 j,k))))/((FRF_NODE{i,1}(j,k+1))*conj((FRF_NODE{i,1}(
49                 j,k+1))));
50             TDI_3(i,j,k)=((FRF_NODE{1,1}(j,k))*conj((FRF_NODE{1,1}(
51                 j,k))))/((FRF_NODE{1,1}(j,k+1))*conj((FRF_NODE{1,1}(
52                 j,k+1))));
53         end
54     end
55 end
56 TDI_1_sum=abs(sum(TDI_1,3)).^2;
57 TDI_2_sum=sum(TDI_2,3);
58 TDI_3_sum=sum(TDI_3,3);
59 %
60 i=1;
61 j=1;
62 for i=1:size(M,3)
63     for j=1:n_sample
64         TDI_4(i,j)=TDI_1_sum(i,j)/(TDI_2_sum(i,j)*TDI_3_sum(i,j
65             ));
66     end
67 end
68 TDI_4_sum=sum(TDI_4,2);
69 TDI=TDI_4_sum/max(TDI_4_sum);
70
71 i=1;
72 for i=1:size(M,3)
73     x(i)=TDI(i);

```

67 end

### A.1.4 FRF DRQ - Respostas no tempo

```

1
2 % FFT criation
3 X_f=(fft(x_t));
4 DX_f=(fft(Dx_t));
5 DDX_f=(fft(DDx_t));
6
7 F=(fft(f_t));
8 %

```

---

```

9 %FRF extracted from the displacement response
10 FRF_X_NODE=cell(size(x_t,3),1);
11 i=1;
12 j=1;
13 k=1;
14 for j=1:size(x_t,3)
15     for i=1:(1024/((size(x_t,1)/10)/2))*size(x_t,1)/2
16         for k=1:size(x_t,2)
17             frf_node_single(i,k)=X_f(i,k,j)/(F(i));
18         end
19     end
20     FRF_X_NODE{j,1}=frf_node_single;
21 end
22
23 %FRF extracted from the velocity response
24 FRF_DX_NODE=cell(size(Dx_t,3),1);
25 i=1;
26 j=1;
27 k=1;
28 for j=1:size(Dx_t,3)
29     for i=1:(1024/((size(Dx_t,1)/10)/2))*size(Dx_t,1)/2
30         for k=1:size(Dx_t,2)
31             frf_node_single(i,k)=DX_f(i,k,j)/(F(i));
32         end
33     end
34     FRF_DX_NODE{j,1}=frf_node_single;
35 end
36
37 %FRF extracted from the aceleration response

```

```

38 FRF_DDX_NODE=cell ( size (DDx_t,3) ,1) ;
39 i=1;
40 j=1;
41 k=1;
42 for j=1:size (DDx_t,3)
43     for i=1:(1024/(( size (DDx_t,1) /10) /2)) * size (DDx_t,1) /2
44         for k=1:size (DDx_t,2)
45             frf_node_single (i ,k)=DDX.f(i ,k ,j) / (F (i) ) ;
46         end
47     end
48     FRF_DDX_NODE{j ,1}=frf_node_single ;
49 end
50 %

```

---

```

51 %

```

---

```

52 %

```

---

```

53 %Application of FRF ODS Correlation Index for FRF
    aceleration
54 dif=cell ( size (DDx_t,3) -1,1) ;
55 i=1;
56 for i=1:size (DDx_t,3)-1
57     dif {i ,1}=abs (FRF_DDX_NODE {i +1,1} -FRF_DDX_NODE {1 ,1} ) ;
58 end
59
60 i=1;
61 j=1;
62 for i=1:size (DDx_t,3)-1
63     for j=1:size (DDx_t,2)
64         Sdif (i ,j)=sum ( dif {i ,1} (: ,j) ) ;
65     end
66 end
67
68 i=1;
69 j=1;
70 for i=1:size (DDx_t,3)-1

```

```
71     Mdif(i)=sum(Sdif(i ,:));
72 end
73 i=1;
74 DI=1;
75 for i=2:size(DDx_t,3)
76     DI(i)=1-(Mdif(i-1)/max(Mdif));
77 end
78
79 i=1;
80 for i=1:size(DDx_t,3)
81     x(i)=DI(i);
82 end
```

### A.1.5 FRF Difference Index - Respostas no tempo

```

1
2 % FFT criation
3 X_f=(fft(x_t));
4 DX_f=(fft(Dx_t));
5 DDX_f=(fft(DDx_t));
6
7 F=(fft(f_t));
8 %

```

---

```

9 %FRF extracted from the displacement response
10 FRF_X_NODE=cell(size(x_t,3),1);
11 i=1;
12 j=1;
13 k=1;
14 for j=1:size(x_t,3)
15     for i=1:(1024/((size(x_t,1)/10)/2))*size(x_t,1)/2
16         for k=1:size(x_t,2)
17             frf_node_single(i,k,j)/(F(i));
18         end
19     end
20     FRF_X_NODE{j,1}=frf_node_single;
21 end
22
23 %FRF extracted from the velocity response
24 FRF_DX_NODE=cell(size(Dx_t,3),1);
25 i=1;
26 j=1;
27 k=1;
28 for j=1:size(Dx_t,3)
29     for i=1:(1024/((size(Dx_t,1)/10)/2))*size(Dx_t,1)/2
30         for k=1:size(Dx_t,2)
31             frf_node_single(i,k,j)/(F(i));
32         end
33     end
34     FRF_DX_NODE{j,1}=frf_node_single;
35 end
36
37 %FRF extracted from the aceleration response

```

```

38 FRF_DDX_NODE=cell(size(DDx_t,3),1);
39 i=1;
40 j=1;
41 k=1;
42 for j=1:size(DDx_t,3)
43     for i=1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
44         for k=1:size(DDx_t,2)
45             frf_node_single(i,k)=DDX_f(i,k,j)/(F(i));
46         end
47     end
48     FRF_DDX_NODE{j,1}=frf_node_single;
49 end
50 %

```

---

```

51 %

```

---

```

52 %

```

---

```

53 i=1;
54 j=1;
55 k=1;
56 for i=1:size(DDx_t,3)
57     for j=1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
58         for k=1:size(DDx_t,2)
59             DRQ_1(i,j,k)=(((FRF_DDX_NODE{i,1}(j,k))*conj((
60                 FRF_DDX_NODE{1,1}(j,k)))));
61             DRQ_2(i,j,k)=((FRF_DDX_NODE{i,1}(j,k))*conj((
62                 FRF_DDX_NODE{i,1}(j,k)))));
63             DRQ_3(i,j,k)=((FRF_DDX_NODE{1,1}(j,k))*conj((
64                 FRF_DDX_NODE{1,1}(j,k)))));
65         end
66     end
67 end
68 i=1;
69 j=1;
70 for i=1:size(DDx_t,3)

```

```

69     for j = 1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
70 DRQ_1_sum_f(i,j)=sum(DRQ_1(i,j,:));
71     end
72 end
73 DRQ_1_sum=abs(DRQ_1_sum_f).^2;
74
75 i = 1;
76 j = 1;
77 for i = 1:size(DDx_t,3)
78     for j = 1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
79 DRQ_2_sum(i,j)=sum(DRQ_2(i,j,:));
80     end
81 end
82
83 i = 1;
84 j = 1;
85 for i = 1:size(DDx_t,3)
86     for j = 1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
87 DRQ_3_sum(i,j)=sum(DRQ_3(i,j,:));
88     end
89 end
90 %
91 i = 1;
92 j = 1;
93 for i = 1:size(DDx_t,3)
94     for j = 1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
95 DRQ_4(i,j)=DRQ_1_sum(i,j)/((DRQ_2_sum(i,j)*DRQ_3_sum(i,
96     j)));
96     end
97 end
98
99 i = 1;
100 for i = 1:size(DDx_t,3)
101 DRQ_4_sum(i)=sum(DRQ_4(i,:));
102 end
103 DRQ=DRQ_4_sum/max(DRQ_4_sum);
104
105
106 i = 1;
107 for i = 1:size(DDx_t,3)

```

```
108 x(i)=DRQ(i);  
109 end
```

### A.1.6 FRF TDI - Respostas no tempo

```

1
2 % FFT criation
3 X_f=(fft(x_t));
4 DX_f=(fft(Dx_t));
5 DDX_f=(fft(DDx_t));
6
7 F=(fft(f_t));
8 %

```

---

```

9 %FRF extracted from the displacement response
10 FRF_XNODE=cell(size(x_t,3),1);
11 i=1;
12 j=1;
13 k=1;
14 for j=1:size(x_t,3)
15     for i=1:(1024/((size(x_t,1)/10)/2))*size(x_t,1)/2
16         for k=1:size(x_t,2)
17             frf_node_single(i,k,j)/(F(i));
18         end
19     end
20     FRF_XNODE{j,1}=frf_node_single;
21 end
22
23 %FRF extracted from the velocity response
24 FRF_DXNODE=cell(size(Dx_t,3),1);
25 i=1;
26 j=1;
27 k=1;
28 for j=1:size(Dx_t,3)
29     for i=1:(1024/((size(Dx_t,1)/10)/2))*size(Dx_t,1)/2
30         for k=1:size(Dx_t,2)
31             frf_node_single(i,k,j)/(F(i));
32         end
33     end
34     FRF_DXNODE{j,1}=frf_node_single;
35 end
36
37 %FRF extracted from the aceleration response

```

```

38 FRF_DDX_NODE=cell(size(DDx_t,3),1);
39 i=1;
40 j=1;
41 k=1;
42 for j=1:size(DDx_t,3)
43     for i=1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
44         for k=1:size(DDx_t,2)
45             frf_node_single(i,k)=DDX_f(i,k,j)/(F(i));
46         end
47     end
48     FRF_DDX_NODE{j,1}=frf_node_single;
49 end
50 %

```

---

```

51 %

```

---

```

52 %

```

---

```

53 %Applicationg of FRF Transmissibility Damage Indicator
54 i=1;
55 j=1;
56 k=1;
57 for i=1:size(DDx_t,3)
58     for j=1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
59         for k=1:size(DDx_t,2)-1
60             TDI_1(i,j,k)=((FRF_DDX_NODE{i,1}(j,k))*conj((
                FRF_DDX_NODE{1,1}(j,k))))/((FRF_DDX_NODE{i,1}(j,k+1)
                )*conj((FRF_DDX_NODE{1,1}(j,k+1))));
61             TDI_2(i,j,k)=((FRF_DDX_NODE{i,1}(j,k))*conj((
                FRF_DDX_NODE{i,1}(j,k))))/((FRF_DDX_NODE{i,1}(j,k+1)
                )*conj((FRF_DDX_NODE{i,1}(j,k+1))));
62             TDI_3(i,j,k)=((FRF_DDX_NODE{1,1}(j,k))*conj((
                FRF_DDX_NODE{1,1}(j,k))))/((FRF_DDX_NODE{1,1}(j,k+1)
                )*conj((FRF_DDX_NODE{1,1}(j,k+1))));
63         end
64     end
65 end

```

```

66 TDI_1_sum=abs(sum(TDI_1,3)).^2;
67 TDI_2_sum=sum(TDI_2,3);
68 TDI_3_sum=sum(TDI_3,3);
69 %
70 i=1;
71 j=1;
72 for i=1:size(DDx_t,3)
73     for j=1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
74         TDI_4(i,j)=TDI_1_sum(i,j)/(TDI_2_sum(i,j)*TDI_3_sum(i,j)
75             ));
76     end
77 end
78 TDI_4_sum=sum(TDI_4,2);
79 TDI=TDI_4_sum/max(TDI_4_sum);
80 i=1;
81 for i=1:size(DDx_t,3)
82     x(i)=TDI(i);
83 end

```

## A.2 Algoritmos de Localização do Dano

### A.2.1 FRF BASED - MODE SHAPE CURVATURE - Modelo Espacial

```

1  %No Damage Matrix 's
2  Mnd=M(:, :, 1);
3  Knd=K(:, :, 1);
4  Cnd=C(:, :, 1);
5  %
6  %60% Damaged Matrix 's
7  Md=M(:, :, size(M,3));
8  Kd=K(:, :, size(K,3));
9  Cd=C(:, :, size(C,3));
10 %
11 %DEFINITION PARAMETERS
12 n_sample=2^10;%Number of samples
13 freq=n_sample/1;%Maximum Frequency definition (1024 Hz
    seems to be a very large frequency for the kynd of
    structures that are analysed in this methods)
14 dfreq=freq/n_sample;%Frequency resolution
15 node=3;%Force Node for FRF
16 n=20;%Nr of nodes of the FEM structure
17 %
18 %FRF calculation for no damage structure by using the
    matrix :
19 %Mnd: Mass Matrix with no damage (nd)
20 %Knd: Stifness Matrix with no damage (nd)
21 %Cnd: Damping Matrix with no damage (nd)
22 i=1;
23 j=1;
24 for i=1:n_sample
25     f(i)=dfreq*i;
26     FRF_nd(:, :, i)=(inv((-2*pi*dfreq*i).^2*Mnd)+(sqrt(-1)
        *(2*pi*dfreq*i)*Cnd)+(Knd)));
27 end
28 %Now we extract single FRF matrix for an exciting force at
    Node 3 for YY axis
29 i=1;
30 j=1;

```

```

31 for i=1:n_sample
32     for j=1:n
33 FRF_node_nd(i , j)=(FRF_nd(j*3-1,node*3-1,i));
34     end
35 end
36 %FRF calculation for 60% damage structure by using the
    matrix :
37 %Md:Mass Matrix with damage (d)
38 %Kd:Stifness Matrix with damage (d)
39 %Cd:Damping Matrix with damage (d)
40 i=1;
41 j=1;
42 for i=1:n_sample
43     f(i)=dfreq*i;
44     FRF_d(:, :, i)=(inv((-2*pi*dfreq*i)^2*Md)+(sqrt(-1)*(2*
        pi*dfreq*i)*Cd)+(Kd)));
45 end
46 %Now we extract single FRF matrix for an exciting force at
    Node 3 for YY axis
47 i=1;
48 j=1;
49 for i=1:n_sample
50     for j=1:n
51 FRF_node_d(i , j)=(FRF_d(j*3-1,node*3-1,i));
52     end
53 end
54 %Calculating the second derivatives from a FRF with no
    damaged
55 i=1;
56 j=1;
57 d2a_ND=zeros(size(FRF_node_nd,1),size(FRF_node_nd,2));
58 for i=1:max(size(FRF_node_nd,1))
59     for j=2:(size(FRF_node_nd,2)-1)
60         d2a_ND(i , j)=((FRF_node_nd(i , j+1))-2*FRF_node_nd
            (i , j)+FRF_node_nd(i , j-1));
61     end
62 end
63 %Calculating the second derivatives from a FRF with damaged
64 i=1;
65 j=1;

```

```

66 d2a_D7=zeros( size(FRF_node_nd,1) , size(FRF_node_nd,2) );
67 for i=1:max( size(FRF_node_nd,1) )
68     for j=2:( size(FRF_node_nd,2) -1)
69         d2a_D7(i , j)=((FRF_node_d(i , j+1))-2*FRF_node_d(i
            , j)+FRF_node_d(i , j-1));%/(2*dh) ^ 2;
70     end
71 end
72 %Applicationg of FRF BASED MODE SHAPE CURVATURE METHOD (from
    DAMAGE DETECTIONIN STRUCTURES: FROM MODE SHAPE TO
    FREQUENCY RESPONSE FUNCTION METHODS; N. M. M. Maia, J. M
    . M. Silva , E. A. M. Almas and R. P. C. Sampaio)
73 i=1;
74 j=1;
75 delta_d2a_ND_D7=zeros( size(FRF_node_nd,1) , size(FRF_node_nd
    ,2) );
76 for i=1:max( size(FRF_node_nd,1) )
77     for j=2:( size(FRF_node_nd,2) -1)
78         delta_d2a_ND_D7(i , j)=abs( d2a_D7(i , j)-d2a_ND(i , j
            ) );
79     end
80 end
81
82 s=sum(delta_d2a_ND_D7);
83 %the "x" variable must be applied in order to output the
    result into the
84 %comparison plot
85 x=s/max(s);

```

## A.2.2 FRF DAMAGE INDEX - Modelo Espacial

```

1 % Matrix 's with No Damage
2 Mnd=M(:, :, 1);
3 Knd=K(:, :, 1);
4 Cnd=C(:, :, 1);
5 %
6 % Matrix 's with 60% Damaged
7 Md=M(:, :, size(M,3) );
8 Kd=K(:, :, size(K,3) );
9 Cd=C(:, :, size(C,3) );
10
11 %

```

```

12 %DEFINITION PARAMETERS
13 n_sample=2^10;%Number of samples
14 freq=n_sample/1;%Maximum Frequency definition (1024 Hz
    seems to be a very large frequency for the kynd of
    structures that are analysed in this methods)
15 dfreq=freq/n_sample;%Frequency resolution
16 node=3;%Force Node for FRF
17 n=20;%Nr of nodes of the FEM structure
18 %
19 %FRF calculation for no damage structure by using the
    matrix :
20 %Mnd:Mass Matrix with no damage (nd)
21 %Knd:Stifness Matrix with no damage (nd)
22 %Cnd:Damping Matrix with no damage (nd)
23 i=1;
24 j=1;
25 for i=1:n_sample
26     f(i)=dfreq*i;
27     FRF_nd(:, :, i)=(inv((- (2*pi*dfreq*i).^2*Mnd)+(sqrt(-1)
        *(2*pi*dfreq*i)*Cnd)+(Knd))));
28 end
29 %Now we extract single FRF matrix for an exciting force at
    Node 3 for YY axis
30 i=1;
31 j=1;
32 for i=1:n_sample
33     for j=1:n
34 FRF_node_nd(i , j)=(FRF_nd(j*3-1,node*3-1,i));
35     end
36 end
37 %FRF calculation for 60% damage structure by using the
    matrix :
38 %Md:Mass Matrix with damage (d)
39 %Kd:Stifness Matrix with damage (d)
40 %Cd:Damping Matrix with damage (d)
41 i=1;
42 j=1;
43 for i=1:n_sample
44     f(i)=dfreq*i;
45     FRF_d(:, :, i)=(inv((- (2*pi*dfreq*i).^2*Md)+(sqrt(-1)*(2*

```

```

        pi*dfreq*i)*Cd)+(Kd));
46 end
47 %Now we extract single FRF matrix for an exciting force at
    Node 3 for YY axis
48 i=1;
49 j=1;
50 for i=1:n_sample
51     for j=1:n
52 FRF_node_d(i , j)=(FRF_d(j*3-1,node*3-1,i));
53     end
54 end
55 %Calculating the second derivatives from a FRF with no
    damaged
56 i=1;
57 j=1;
58 d2a_ND=zeros(size(FRF_node_nd,1),size(FRF_node_nd,2));
59 for i=1:max(size(FRF_node_nd,1))
60     for j=2:(size(FRF_node_nd,2)-1)
61         d2a_ND(i , j)=((FRF_node_nd(i , j+1))-2*FRF_node_nd
            (i , j)+FRF_node_nd(i , j-1));
62     end
63 end
64 %Calculating the second derivatives from a FRF with damaged
65 i=1;
66 j=1;
67 d2a_D7=zeros(size(FRF_node_nd,1),size(FRF_node_nd,2));
68 for i=1:max(size(FRF_node_nd,1))
69     for j=2:(size(FRF_node_nd,2)-1)
70         d2a_D7(i , j)=((FRF_node_d(i , j+1))-2*FRF_node_d(i
            , j)+FRF_node_d(i , j-1));%/(2*dh)^2;
71     end
72 end
73 %Aplicação de FRF-BASED DAMAGE INDEX (from DAMAGE
    DETECTION IN STRUCTURES: FROM MODE SHAPE TO FREQUENCY
    RESPONSE FUNCTION METHODS; N. M. M. Maia, J. M. M. Silva
    , E. A. M. Almas and R. P. C. Sampaio)
74 i=1;
75 j=1;
76 BetaFRF_node_nd_D7=zeros(n_sample , size(FRF_node_d,2));
77 for i=1:max(size(FRF_node_d,1))

```

```

78     for j=2:(size(FRF_node_nd,2)-1)
79         BetaFRF_node_nd_D7(i,j)=((d2a_D7(i,j).^2 + sum(
                d2a_D7(i,:) .^2))*sum(d2a_ND(i,:) .^2))/((d2a_ND(i
                ,j).^2 + sum(d2a_ND(i,:) .^2))*sum(d2a_D7(i,:)
                .^2));
80     end
81 end
82 s=abs(sum(BetaFRF_node_nd_D7));
83 %the "x" variable must be applied in order to output the
    result into the
84 %comparison plot
85 x=s/max(s);

```

### A.2.3 FRF BASED - MODE SHAPE CURVATURE - Modelo Respostas no Tempo

```

1
2 % FFT criation
3 X_f=(fft(x_t));
4 DX_f=(fft(Dx_t));
5 DDX_f=(fft(DDx_t));
6
7 F=(fft(f_t));
8 %

```

---

```

9 %FRF extracted from the displacement response
10 FRF_X_NODE=cell(size(x_t,3),1);
11 i=1;
12 j=1;
13 k=1;
14 for j=1:size(x_t,3)
15     for i=1:(1024/((size(x_t,1)/10)/2))*size(x_t,1)/2
16         for k=1:size(x_t,2)
17             frf_node_single(i,k)=X_f(i,k,j)/(F(i));
18         end
19     end
20     FRF_X_NODE{j,1}=frf_node_single;
21 end
22
23 %FRF extracted from the velocity response

```

```

24 FRF_DX_NODE=cell(size(Dx_t,3),1);
25 i=1;
26 j=1;
27 k=1;
28 for j=1:size(Dx_t,3)
29     for i=1:(1024/((size(Dx_t,1)/10)/2))*size(Dx_t,1)/2
30         for k=1:size(Dx_t,2)
31             frf_node_single(i,k)=DX.f(i,k,j)/(F(i));
32         end
33     end
34     FRF_DX_NODE{j,1}=frf_node_single;
35 end
36
37 %FRF extracted from the acceleration response
38 FRF_DDX_NODE=cell(size(DDx_t,3),1);
39 i=1;
40 j=1;
41 k=1;
42 for j=1:size(DDx_t,3)
43     for i=1:(1024/((size(DDx_t,1)/10)/2))*size(DDx_t,1)/2
44         for k=1:size(DDx_t,2)
45             frf_node_single(i,k)=DDX.f(i,k,j)/(F(i));
46         end
47     end
48     FRF_DDX_NODE{j,1}=frf_node_single;
49 end
50 %
51 %
52 %
53 FRF_node_nd=FRF_DDX_NODE{1,1};
54 FRF_node_d=FRF_DDX_NODE{7,1};
55 %Calculating the second derivatives from a FRF with no
    damaged
56 i=1;

```

```

57 j=1;
58 d2a_ND=zeros(size(FRF_node_nd,1),size(FRF_node_nd,2));
59 for i=1:max(size(FRF_node_nd,1))
60     for j=2:(size(FRF_node_nd,2)-1)
61         d2a_ND(i,j)=((FRF_node_nd(i,j+1))-2*FRF_node_nd
62             (i,j)+FRF_node_nd(i,j-1));
63     end
64 end
65 %Calculating the second derivatives from a FRF with damaged
66 i=1;
67 j=1;
68 d2a_D7=zeros(size(FRF_node_nd,1),size(FRF_node_nd,2));
69 for i=1:max(size(FRF_node_nd,1))
70     for j=2:(size(FRF_node_nd,2)-1)
71         d2a_D7(i,j)=((FRF_node_d(i,j+1))-2*FRF_node_d(i
72             ,j)+FRF_node_d(i,j-1));%/(2*dh)^2;
73     end
74 end
75 %Applicationg of FRF BASED MODE SHAPE CURVATURE METHOD (from
76     DAMAGE DETECTIONIN STRUCTURES: FROM MODE SHAPE TO
77     FREQUENCY RESPONSE FUNCTION METHODS; N. M. M. Maia, J. M
78     . M. Silva , E. A. M. Almas and R. P. C. Sampaio)
79 i=1;
80 j=1;
81 delta_d2a_ND_D7=zeros(size(FRF_node_nd,1),size(FRF_node_nd
82     ,2));
83 for i=1:max(size(FRF_node_nd,1))
84     for j=2:(size(FRF_node_nd,2)-1)
85         delta_d2a_ND_D7(i,j)=abs(d2a_D7(i,j)-d2a_ND(i,j
86             ));
87     end
88 end
89 s=sum(delta_d2a_ND_D7);
90 %the "x" variable must be applied in order to output the
91     result into the
92 %comparison plot
93 x=s/max(s);

```

#### A.2.4 FRF DAMAGE INDEX - Modelo Respostas no Tempo

```

1  %
2
3  % FFT criation
4  X_f=(fft(x_t));
5  DX_f=(fft(Dx_t));
6  DDX_f=(fft(DDx_t));
7
8  F=(fft(f_t));
9  %
10 %FRF extracted from the displacement response
11 FRF_X_NODE=cell(size(x_t,3),1);
12 i=1;
13 j=1;
14 k=1;
15 for j=1:size(x_t,3)
16     for i=1:(1024/((size(x_t,1)/10)/2))*size(x_t,1)/2
17         for k=1:size(x_t,2)
18             frf_node_single(i,k)=X_f(i,k,j)/(F(i));
19         end
20     end
21     FRF_X_NODE{j,1}=frf_node_single;
22 end
23
24 %FRF extracted from the velocity response
25 FRF_DX_NODE=cell(size(Dx_t,3),1);
26 i=1;
27 j=1;
28 k=1;
29 for j=1:size(Dx_t,3)
30     for i=1:(1024/((size(Dx_t,1)/10)/2))*size(Dx_t,1)/2
31         for k=1:size(Dx_t,2)
32             frf_node_single(i,k)=DX_f(i,k,j)/(F(i));
33         end
34     end
35     FRF_DX_NODE{j,1}=frf_node_single;
36 end

```

```

37
38 %FRF extracted from the acceleration response
39 FRF_DDX_NODE=cell ( size (DDx_t,3) ,1) ;
40 i=1;
41 j=1;
42 k=1;
43 for j=1:size (DDx_t,3)
44     for i=1:(1024/(( size (DDx_t,1) /10) /2)) * size (DDx_t,1) /2
45         for k=1:size (DDx_t,2)
46             frf_node_single (i ,k)=DDX_f(i ,k ,j) / (F(i) ) ;
47             end
48         end
49         FRF_DDX_NODE{j ,1}=frf_node_single ;
50     end
51 %
52 %
53 %
54 FRF_node_nd=FRF_DDX_NODE{1 ,1} ;
55 FRF_node_d=FRF_DDX_NODE{7 ,1} ;
56 %Calculating the second derivatives from a FRF with no
    damaged
57 i=1;
58 j=1;
59 d2a_ND=zeros ( size (FRF_node_nd ,1) , size (FRF_node_nd ,2) ) ;
60 for i=1:max ( size (FRF_node_nd ,1) )
61     for j=2:( size (FRF_node_nd ,2) -1)
62         d2a_ND (i ,j) = ((FRF_node_nd (i ,j+1)) -2*FRF_node_nd
            (i ,j) +FRF_node_nd (i ,j-1)) ;
63     end
64 end
65 %Calculating the second derivatives from a FRF with damaged
66 i=1;
67 j=1;
68 d2a_D7=zeros ( size (FRF_node_nd ,1) , size (FRF_node_nd ,2) ) ;

```

```

69 for i=1:max(size(FRF_node_nd,1))
70     for j=2:(size(FRF_node_nd,2)-1)
71         d2a_D7(i,j)=((FRF_node_d(i,j+1))-2*FRF_node_d(i
72             ,j)+FRF_node_d(i,j-1));%/(2*dh)^2;
73     end
74 end
75 %Application of FRF-BASED DAMAGE INDEX (from DAMAGE
76     DETECTION IN STRUCTURES: FROM MODE SHAPE TO FREQUENCY
77     RESPONSE FUNCTION METHODS; N. M. M. Maia, J. M. M. Silva
78     , E. A. M. Almas and R. P. C. Sampaio)
79 i=1;
80 j=1;
81 BetaFRF_node_nd_D7=zeros((1024/((size(DDx_t,1)/10)/2))*size
82     (DDx_t,1)/2,size(FRF_node_d,2));
83 for i=1:max(size(FRF_node_d,1))
84     for j=2:(size(FRF_node_nd,2)-1)
85         BetaFRF_node_nd_D7(i,j)=((d2a_D7(i,j).^2 + sum(
86             d2a_D7(i,:) .^2))*sum(d2a_ND(i,:) .^2))/((d2a_ND(i
87                 ,j).^2 + sum(d2a_ND(i,:) .^2))*sum(d2a_D7(i,:)
88                     .^2));
89     end
90 end
91 s=abs(sum(BetaFRF_node_nd_D7));
92 %the "x" variable must be applied in order to output the
93     result into the
94 %comparison plot
95 x=s/max(s);

```