



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2017/08.18.16.12-TDI

VIBRATION-BASED DAMAGE IDENTIFICATION USING HYBRID OPTIMIZATION ALGORITHMS

Reynier Hernández Torres

Doctorate Thesis of the Graduate
Course in Applied Computing,
guided by Drs. Haroldo Fraga
de Campos Velho, and Leonardo
Dagnino Chiwiacowsky, approved
in September 04, 2017.

URL of the original document:

[<http://urlib.net/8JMKD3MGP3W34P/3PFB7K2>](http://urlib.net/8JMKD3MGP3W34P/3PFB7K2)

INPE
São José dos Campos
2017

PUBLISHED BY:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

E-mail: pubtc@inpe.br

**COMMISSION OF BOARD OF PUBLISHING AND PRESERVATION
OF INPE INTELLECTUAL PRODUCTION (DE/DIR-544):****Chairperson:**

Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação (CPG)

Members:

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

Dr. André de Castro Milone - Coordenação de Ciências Espaciais e Atmosféricas (CEA)

Dra. Carina de Barros Melo - Coordenação de Laboratórios Associados (CTE)

Dr. Evandro Marconi Rocco - Coordenação de Engenharia e Tecnologia Espacial (ETE)

Dr. Hermann Johann Heinrich Kux - Coordenação de Observação da Terra (OBT)

Dr. Marley Cavalcante de Lima Moscati - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Silvia Castro Marcelino - Serviço de Informação e Documentação (SID) **DIGITAL LIBRARY:**

Dr. Gerald Jean Francis Banon

Clayton Martins Pereira - Serviço de Informação e Documentação (SID)

DOCUMENT REVIEW:

Simone Angélica Del Ducca Barbedo - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

ELECTRONIC EDITING:

Marcelo de Castro Pazos - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2017/08.18.16.12-TDI

VIBRATION-BASED DAMAGE IDENTIFICATION USING HYBRID OPTIMIZATION ALGORITHMS

Reynier Hernández Torres

Doctorate Thesis of the Graduate
Course in Applied Computing,
guided by Drs. Haroldo Fraga
de Campos Velho, and Leonardo
Dagnino Chiwiacowsky, approved
in September 04, 2017.

URL of the original document:

[<http://urlib.net/8JMKD3MGP3W34P/3PFB7K2>](http://urlib.net/8JMKD3MGP3W34P/3PFB7K2)

INPE
São José dos Campos
2017

Cataloging in Publication Data

Hernández Torres, Reynier.

H43v Vibration-based damage identification using hybrid optimization algorithms / Reynier Hernández Torres. – São José dos Campos : INPE, 2017.

xxxiv + 169 p. ; (sid.inpe.br/mtc-m21b/2017/08.18.16.12-TDI)

Thesis (Doctorate in Applied Computing) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2017.

Guiding : Drs. Haroldo Fraga de Campos Velho, and Leonardo Dagnino Chiwiacowsky.

1. Vibration-based damage identification. 2. Nastran. 3. Inverse problem. 4. Multi-particle collision algorithm. 5. q-gradient method. I.Title.

CDU 004.4'416:519.254



Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](#).

This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](#).

Aluno (a): **Reynier Hernández Torres**

Título: "VIBRATION-BASED DAMAGE IDENTIFICATION USING HYBRID OPTIMIZATION ALGORITHMS"

Aprovado (a) pela Banca Examinadora
em cumprimento ao requisito exigido para
obtenção do Título de **Doutor(a)** em
Computação Aplicada

Dr. Ezzat Selim Chalhoub


Presidente / INPE / SJC Campos - SP

Dr. Haroldo Fraga de Campos Velho


Orientador(a) / INPE / São José dos Campos - SP

Dr. Leonardo Dagnino Chiwiacowsky


Orientador(a) / UCS / Bento Gonçalves - RS

Dr. Stephan Stephany


Membro da Banca / INPE / SJC Campos - SP

Dr. Fernando Manuel Ramos


Membro da Banca / INPE / SJC Campos - SP

Dr. Domingos Alves Rade


Convidado(a) / IITA / São José dos Campos - SP

Dr. Roberto Pinto Souto


Convidado(a) / LNCC / Petrópolis - RJ

Este trabalho foi aprovado por:

() maioria simples

☒ unanimidade

São José dos Campos, 04 de setembro de 2017

*Poner la ciencia en lengua diaria: he ahí
un gran bien que pocos hacen*

José Martí

“LAS LEYES DE LA HERENCIA”, 1884

*Science is knowledge which we
understand so well that we can teach it to
a computer; and if we don't fully
understand something, it is an art to deal
with it.*

Donald E. Knuth

“COMPUTER PROGRAMMING AS AN ART”, 1974

A mis padres Consuelo y Juan

ACKNOWLEDGEMENTS

First of all, I want to thank my advisors Dr. Haroldo F. de Campos Velho, from the *Instituto Nacional de Pesquisas Espaciais* (INPE) and Dr. Leonardo D. Chiwiacowsky, from the *Universidade de Caxias do Sul*, for giving me the opportunity to make science with them. Their guidance, patience, and friendship helped me in all the research and writing of this thesis.

I want to acknowledge the financial support provided by the *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq) – grant 159547/2013-0, and the *Coordenação de Aperfeiçoamento de Pessoal de Ensino Superior* (CAPES), both from Brazil.

Thanks to the INPE and its great researchers and professors, for all the knowledge and support, the access to the laboratory and research facilities.

Thanks to the *Universidade do Vale do Rio dos Sinos* (Unisinos) for receiving me in 2014, and allowing me to use their installations.

I also would like to thank my colleagues of the research group: Dra. Renata Sampaio, Dr. Eduardo da Luz, Dra. Juliana Anochi, Dr. Helaine Furtado, Dra. Sabrina Sambatti, Dr. Elcio Shiguemori, Me. Luis Fernando Salgueiro, Dr. Leonardo Santos, for the stimulating discussions, and for all the experiences and fun we have had in this four years.

Thanks to Dr. Fernando Ramos and its research group, in special to Dra. Marluce Scarabello, Dra. Aline Soterroni, and Dra. Érica Gouvêa, for giving me the opportunity of collaborating with them.

Thanks to the people with whom I have lived along these four years, offering their friendship, and helping to make the bad times not so bad, and make the good times excellent. In special, I want to mention the Cuban community at INPE, which made me feel like home is not so far away.

Finally, I want to express my gratitude to my family, in special to my parents, for their love, for their support, help, and for giving me the opportunities that have made me who I am.

Thanks to all!

ABSTRACT

The inverse problem of structural damage identification is addressed in this thesis. The inverse solution is obtained by solving an optimization problem using different hybrid algorithms. The forward structural model is solved by Finite Element codes. FORTRAN code developed by the research group of the Laboratório Associado de Computação e Matemática Aplicada (LAC) of the Instituto Nacional de Pesquisas Espaciais (INPE) was applied to some problems, and for other numerical experiments the NASTRAN software was employed. The acceleration, velocity or displacement time history could be used as experimental data in this methodology. The objective function is formulated as the sum of the squared difference between the measured displacement and the data calculated by the forward model. Different hybrid metaheuristics are tested, using a two-step approach. The first step performs the exploration, and the second one carries out the exploitation, starting from the best solution found in the first step. One optimization approach combines the Multi-Particle Collision Algorithm (MPCA) with the Hooke-Jeeves (HJ) direct search method. MPCA is improved using different mechanisms derived from the Opposition-Based Learning, such as Center-Based Sampling and Rotation-Based Learning. Other applied optimizer is the novel q-gradient, and it is also hybridized with HJ method. The methodology is tested on structures with different complexities. Time-invariant damage was assumed to generate the synthetic measurements. Noiseless and noisy data were considered in tests using models implemented in FORTRAN. Most of the experiments were performed using a full set of data, from all possible nodes, and an experiment was done using a reduced dataset with a low level of noise in data. Noiseless data were considered with experiments using NASTRAN. In this case, the experiments were performed using a full set of data. In general, good estimations for damage location and severity are achieved. Some false positives have appeared, but damages were well identified.

Keywords: Vibration-based Damage Identification. Nastran. Inverse Problem. Multi-Particle Collision Algorithm. q-Gradient Method. Hooke-Jeeves Direct Search Method.

IDENTIFICAÇÃO DE DANOS BASEADA EM VIBRAÇÃO USANDO ALGORITMOS DE OTIMIZAÇÃO HÍBRIDOS

RESUMO

O problema inverso da identificação de danos estruturais é abordado nesta tese. A solução inversa é obtida resolvendo um problema de otimização usando diferentes algoritmos híbridos. O modelo direto estrutural é resolvido pelo Método dos Elementos Finitos. Código FORTRAN desenvolvido pelo grupo de pesquisa do Laboratório Associado de Computação e Matemática Aplicada (LAC) do Instituto Nacional de Pesquisas Espaciais (INPE) foi aplicado em alguns problemas e, para outros experimentos numéricos, o software NASTRAN foi empregado. O histórico de tempo de aceleração, velocidade ou deslocamento pode ser usado como dados experimentais nesta metodologia. A função objetivo é formulada como a soma da diferença quadrática entre o deslocamento medido e os dados calculados pelo modelo direto. Diferentes metaheurísticas híbridas são testadas, usando uma abordagem em duas etapas. A primeira etapa realiza a exploração em todo o espaço de busca, e a segunda etapa realiza a intensificação a partir da melhor solução encontrada pela primeira etapa. Uma abordagem de otimização combina o Algoritmo de Colisão de Múltiplas Partículas (MPCA) com o método de busca direta Hooke-Jeeves (HJ). O MPCA é melhorado usando diferentes mecanismos derivados da Aprendizagem Baseada na Oposição, como são a Amostragem Baseada no Centro, e a Aprendizagem Baseada em Rotação. Outro otimizador aplicado é o novo q -gradiente, que também é hibridado com o método HJ. A metodologia é testada em estruturas com diferentes complexidades. Supõe-se que os danos são invariante no tempo para gerar as medidas experimentais sintéticas. Dados sem ruído e com diferentes níveis de ruído foram considerados em testes usando modelos implementados em FORTRAN. A maioria dos experimentos foi realizada usando um conjunto completo de dados, de todos os nós possíveis, e um dos experimentos foi feito usando um conjunto incompleto de dados, com um baixo nível de ruído. Para os experimentos utilizando o NASTRAN, foram considerados dados sintéticos sem ruído, e foi utilizado o conjunto completo de dados. Em geral, boas estimativas para localização e gravidade do dano foram alcançadas. Alguns falsos positivos apareceram nas estimativas, mas os danos presentes nos sistemas foram bem identificados.

Palavras-chave: Identificação de danos baseada em vibração. Nastran. Problema Inverso. Algoritmo de Colisão de Múltiplas Partículas. Método q -Gradiente. Hooke-Jeeves.

LIST OF FIGURES

	<u>Page</u>
1.1 Aircrafts developed by Embraer	1
1.2 Satellites developed by INPE	2
1.3 Space Stations	3
1.4 Overview of damage identification methods. The branch of vibration-based methods is detailed with different features and classifiers used in the literature	9
2.1 Local (s'_1 and s'_2) and global minima (s^*)	21
2.2 Schematic representation of the forward and inverse problems	22
2.3 Vibration-Based Damage Identification as optimization problem	24
3.1 Finite Element Method	32
3.2 Degrees of freedom	33
3.3 Spring element	34
3.4 Bar element	36
3.5 Representation of a bar element	36
3.6 Bar element in global coordinates	39
3.7 Global coordinates	40
3.8 Examples of different types of beams	41
3.9 Beam element	42
4.1 Classification of mono-objective optimization algorithms	48
4.2 Phenomena inside a nuclear reactor that inspire MPCA: (a) scattering; and (b) absorption.	52
4.3 Flowchart of the Multi-Particle Collision Algorithm	53
4.4 Graphical representation of the opposition number (s_o), quasi-opposite number (s_{qo}), quasi-reflected number (s_{qr}) and center-based sampled number (s_{cb}) from the original number s	59
4.5 Geometric interpretation of the Rotation (s_r) and the Rotation-based Sampling (s_{rbs}) numbers in 2D space	60
4.6 Flowchart of the q -gradient	68
4.7 Flowchart of the Hooke-Jeeves direct search method	69
4.8 Flowchart of the Exploratory function in the Hooke-Jeeves direct search method	69
4.9 Operating flow of the hybrid algorithms	72

5.1	Vibration-Based Damage Identification as optimization problem using models implemented in FORTRAN	74
5.2	Case study 1: Damped spring-mass system with 10-DOF	76
5.3	Case study 1: Load applied on the spring-mass system	76
5.4	Case study 1: Dynamic responses: deflection in the x -axis vs. time	77
5.5	Case study 1: Results using MPCA-HJ on different noise levels	77
5.7	Case study 2: Three-bay truss structure	78
5.6	Case study 1: Results with different configurations	79
5.8	Case study 2: Dynamic responses: deflection in the y -axis vs. time	80
5.9	Case study 2: Results using MPCA-HJ, CMPCA-HJ, RMPCA-HJ, and q G-HJ	82
5.10	International Space Station	83
5.11	Case study 3: Truss structure with 72 bars	84
5.12	Case study 3: Loads applied on the 8 th and 13 th -DOFs of the system	85
5.13	Case study 3: Dynamic responses: deflection vs. time	85
5.14	Case study 3: Average results for 50 experiments	86
5.15	Case study 4: Kabe's Problem	87
5.16	Case study 4: Load applied on the 6 th DOF of the system	88
5.17	Case study 4: Dynamic responses: deflection in the y -axis vs. time	88
5.18	Case study 4: Results using MPCA-HJ, CMPCA-HJ, RMPCA-HJ and q G-HJ - Mean for 50 experiments assuming noiseless and noisy data	90
5.19	Case study 4: Boxplot for 50 experiments on noisy data using RMPCA-HJ. Outliers are represented with the \times symbol.	91
5.20	Case study 5: Yang's Problem	92
5.21	Case study 5: Loads applied on the 8 th (—) and 13 th (—) DOFs of the system	93
5.22	Case study 5: Dynamic responses: deflection in the y -axis vs. time	93
5.23	Case study 5: Results using MPCA-HJ, CMPCA-HJ and RMPCA-HJ - Mean for 50 experiments on noiseless and noisy data	95
5.24	Case study 6: Cantilever Beam structure	96
5.25	Case study 6: Beam model with 20-DOF	96
5.26	Case study 6: Load applied on the cantilever beam	96
5.27	Case study 6: Dynamic responses: deflection in the y -axis vs. time	97
5.28	Case study 6: Results for the MPCA-HJ, CMPCA-HJ and RMPCA-HJ using a full dataset - Single damage in the fixed element	99
5.29	Case study 6: Results for the MPCA-HJ using entropy regularization and a full dataset - Single damage in the fixed element	100
5.30	Case study 6: Results for the MPCA-HJ, CMPCA-HJ and RMPCA-HJ using an incomplete dataset - Single damage in the fixed element	101

5.31	Case study 6: Results for MPCA-HJ, CMPCA-HJ and RMPCA-HJ using a full dataset - Configuration with mixed damages	103
5.32	Case study 6: Results for the MPCA-HJ and RMPCA-HJ using an incomplete dataset - Configuration with mixed damages	104
6.1	Vibration-Based Damage Identification as optimization problem using NASTRAN as a black-box model	110
6.2	Case study 1: Spring mass system with 2-DOF	111
6.3	Case study 1: Damage identification on the spring-mass system	113
6.4	Case study 1: Evolution of the MPCA and RMPCA using noiseless data	113
6.5	Case study 2: Spring mass system with 10-DOF	114
6.6	Case study 2: Damage identification on the spring-mass system	115
6.7	Case study 3: Two-bay truss structure	115
6.8	Case study 3: Damage identification on the truss structure	117
6.9	Case study 4: Cantilever beam structure	117
6.10	Case study 4: Cantilever beam with ten elements with applied loads on nodes 5 and 10. Damages are represented in red.	118
6.11	Case study 4: Loads applied on the cantilever beam	118
6.12	Case study 4: Damage identification on the cantilever beam	119
B.1	Structure of the NASTRAN Input file. The required sections are represented in bold.	158

LIST OF TABLES

	<u>Page</u>
1.1 Dynamics based NDE technologies. The sensibility to the damage, the data interpretation, and the applicability for SHM are evaluated on a scale of 1 (less) to 5 (more) points, shown with plus signs (+).	10
1.2 Overview of the vibration based damage features	11
1.3 Literature overview of Computational Intelligence methods used for SDI	11
3.1 Modal versus direct transient response (NASTRAN, 2004)	30
3.2 Types of elements	32
3.3 Some common sections and their area moment of inertia	44
5.1 Control parameters for the hybrid algorithms	75
5.2 Case study 1: Mean of the computing time (in seconds) spent by the algorithms	78
5.3 Case study 2: Material properties	80
5.4 Case study 2: Mean of the computing time (in second) spent by the algorithms	81
5.5 Case study 3: Material properties	84
5.6 Case study 4: Dimensionless mass and stiffness	88
5.7 Case study 4: Mean of the computing time (in seconds) spent by the algorithms	89
5.8 Case study 5: Parameters of the system. Masses in <i>kg</i> , stiffness in <i>kg/m</i> and damping in <i>kg s/m</i>	92
5.9 Case study 5: Mean of the computing time (in seconds) spent by the algorithms	94
5.10 Case study 6: Mean of the computing time (in seconds) spent by the algorithms	102
5.11 Summary of the false-positives (FP) and false-negatives (FN) obtained by the algorithms in the case studies	106
6.1 Case study 6: Control parameters for the hybrid algorithms	111
A.1 Literature overview of features and classification approaches for vibration-based damage identification	155
B.1 Some commonly used elastic elements in NASTRAN	159

LIST OF ALGORITHMS

	<u>Page</u>
4.1 Multi-Particle Collision Algorithm	54
4.2 Perturbation function	55
4.3 Exploitation function	56
4.4 Small Perturbation function	56
4.5 Small Perturbation in a single random dimension function	57
4.6 Scattering function	57
4.7 UpdateBlackboard function using MPI	58
4.8 Opposition function	62
4.9 Multi-Particle Collision Algorithm with Opposition	63
4.10 Perturbation function with Opposition	64
4.11 Exploitation function with Opposition	64
4.12 Scattering function with Opposition	65
4.13 q -gradient method	67
4.14 Hooke-Jeeves pattern search method	70
4.15 Exploratory function	70

LIST OF ABBREVIATIONS

ABC	– Artificial Bee Colony
ACO	– Ant Colony Optimization
AE	– Acoustic Emission
AIS	– Artificial Immune Systems
AU	– Acousto-ultrasonic
BBC	– Big Bang-Big Crunch
BBO	– Biogeography-based Optimization
BFO	– Bacterial Foraging Optimization
BMO	– Bird Mating Optimizer
BVID	– Barely Visible Impact Damage
CA	– Cultural Algorithms
CAE	– Computer-Aided Engineering
CBERS	– China-Brazil Earth Resources Satellite
CB-MPCA	– Center-based Sampling MPCA
CM	– Condition Monitoring
CoFE	– Compatible Finite Elements
CPU	– Central Processing Unit
DE	– Differential Evolution
DOF	– Degrees of freedom
DP	– Damage Prognosis
EMI	– Electromechanical Impedance
EMBRAER	– <i>Empresa Brasileira de Aeronáutica</i>
EP	– Evolutionary Programming
ES	– Evolution Strategies
FEA	– Finite Element Analysis
FEM	– Finite Element Method
GA	– Genetic Algorithm
GH	– Greedy Heuristic
GP	– Genetic Programming
HJ	– Hooke-Jeeves
HS	– Harmony Search
ILS	– Iterated Local Search
INPE	– <i>Instituto Nacional de Pesquisas Espaciais</i>
IP	– Inverse Problem
ISS	– International Space Station
IWS	– Intelligent Water Drops
KMA	– Kabe’s Stiffness Matrix Adjustment
LS	– Local Search
MA	– Memetic Algorithm
MPCA	– Multi-Particle Collision Algorithm
MSC	– MacNeal-Schendler Corporation
NASA	– National Aeronautics and Space Administration
NASTRAN	– NASA Structural Analysis System
NDE	– Non-Destructive Damage Evaluation

NDD	–	Non-Destructive Damage Detection
NM	–	Nelder-Mead algorithm
OBL	–	Opposition Based Learning
O-MPCA	–	Oppositional MPCA
PCA	–	Particle Collision Algorithm
PSO	–	Particle Swarm Optimization
qG	–	q -Gradient
QO-MPCA	–	Quasi-Oppositional MPCA
QR-MPCA	–	Quasi-Reflective MPCA
RB	–	Rotation-based Learning
RBS	–	Rotation-based Sampling
SA	–	Simulated Annealing
SCD	–	Data Collection Satellite
SDI	–	Structural Damage Identification
SHM	–	Structural Health Monitoring
SPC	–	Statistical Process Control
SS	–	Space Station
SV	–	Structural Vibration
TS	–	Tabu Search
UT	–	Ultrasonic Testing
VDI	–	Vibration-based Damage Identification

LIST OF SYMBOLS

Latin symbols

A	–	Cross-sectional area
c	–	$\cos \theta$
\mathbf{C}	–	Viscous damping
D	–	number of decision variables
e	–	Residual
E	–	Young's modulus (modulus of elasticity)
\mathbf{f}	–	Force
\mathbf{F}	–	External force
$g(\square)$	–	Inequality constraint
$h(\square)$	–	Equality constraint
h	–	Step size (HJ)
$J(\square)$	–	Objective function
\mathbf{k}	–	Stiffness vector
\mathbf{K}	–	Stiffness matrix
ℓ	–	Length
L	–	Largest distance within the search space
\mathbf{M}	–	Mass matrix
n_m	–	Number of measured displacements
N	–	Modal force
N_{FE}	–	Number of function evaluations
$N_{particles}$	–	Number of particles (MPCA)
$N_{processor}$	–	Number of processors (MPCA)
\mathcal{N}	–	Random number (normal distribution)
p_s	–	Scattering probability
q	–	q parameter (qG)
R	–	Random number (uniform distribution)
\mathbb{R}	–	Space of real numbers
s	–	$\sin \theta$
t	–	Time
\mathbf{T}	–	Transformation matrix
\mathbf{u}	–	Displacement vector
\mathbf{v}	–	Search direction (qG)
\mathbf{V}	–	Search direction matrix (HJ)
\mathbf{w}	–	Internal forces vector
x, y, z	–	Coordinate system axes

Greek symbols

α	–	Step length
β	–	Deflection angle (RB)
β	–	Reduction factor (qG)
δ	–	Deformation
ϵ	–	Axial strain
ζ	–	Modal damping ratio

θ	–	Rotation vector
Θ^d	–	Damage parameters
Θ^e	–	Environmental and operational conditions
ξ	–	Modal coordinate
ρ	–	Density
ρ	–	Reduction factor of the step (HJ)
σ	–	Axial stress
σ	–	Standard deviation
τ	–	Delay
Φ	–	Modal vector
$\Phi_{\mathbf{w}}$	–	Transformation matrix
ω	–	Modal frequency

Subscripts

\square_{cb}	–	Center-based sampling
\square_d	–	Dimension d
\square_i	–	Processor i
\square_j	–	Particle j
\square_m	–	Measurement m
\square_{min}	–	Minimum value
\square_o	–	Opposite
\square_q	–	Referent to q -gradient
\square_{qo}	–	Quasi-opposite
\square_{qr}	–	Quasi-reflected
\square_r	–	Rotated
\square_{rbs}	–	Rotation-based sampling
\square_0	–	Initial

Superscripts

\square^b	–	Best solution
\square^c	–	Current solution (HJ)
\square^d	–	Damaged
\square^i	–	Integral
\square^{inf}	–	Inferior limit
\square^l	–	Lower bound
\square^{mod}	–	Data obtained from the structural model
\square^n	–	New solution (MPCA)
\square^{obs}	–	Measured or observed data
\square^{sup}	–	Superior limit
\square^{\top}	–	Transpose operation
\square^u	–	Upper bound
\square^*	–	Globally optimal
\square'	–	Locally optimal
\square^{\diamond}	–	Best solution overall (MPCA)
\square^{\star}	–	Perturbed solution (MPCA)
\square^{\S}	–	Exploited solution (MPCA)

\square° – Pattern point (HJ)

Others

$\dot{\square}$ – First derivative

$\ddot{\square}$ – Second derivative

$\tilde{\square}$ – Optimal solution

$\hat{\square}$ – Noise added to the data

∇ – Gradient

$\|\square\|$ – Norm of \square for a vector or matrix

CONTENTS

	<u>Page</u>
1 INTRODUCTION	1
1.1 Damages	3
1.2 Structural Damage Identification	5
1.2.1 Classification of the Structural Damage Identification methods	6
1.3 Literature overview of techniques for Structural Damage Identification	8
1.3.1 Methods based on structural vibration for Structural Damage Identification	10
1.3.2 Metaheuristics applied to Structural Damage Identification	11
1.3.3 Gaps found in the literature	14
1.4 Research objective	16
1.4.1 Tasks	16
1.5 Outline	17
 2 VIBRATION-BASED DAMAGE IDENTIFICATION SOLVED AS OPTIMIZATION PROBLEM	 19
2.1 Basic concepts of optimization	19
2.2 Inverse problems in engineering	22
2.3 Vibration-based Damage Identification as an Inverse Problem	23
2.3.1 Damage parameters	24
2.3.2 Objective function	24
2.3.3 Regularization by maximum entropy principle	25
2.3.4 Morozov's discrepancy principle	25
2.4 Chapter Conclusions	26
 3 TRANSIENT RESPONSE ANALYSIS OF STRUCTURES AS DIRECT MODEL	 27
3.1 Transient response analysis	27
3.1.1 Direct transient response	27
3.1.2 Modal transient response	29
3.1.3 Modal Versus Direct Transient Response	30
3.2 Finite Element Method	31
3.2.1 Stiffness matrix	33
3.3 Spring element	33

3.4	Trusses	36
3.4.1	Bar element	36
3.4.1.1	Transformation into global coordinates	39
3.4.2	Beam element	41
3.5	Damping matrix	45
3.6	Chapter conclusions	45
4	HYBRID ALGORITHMS FOR SOLVING THE DAMAGE IDENTIFICATION PROBLEM	47
4.1	Optimization algorithms	47
4.1.1	Metaheuristic algorithms	48
4.1.2	Hybrid algorithms	51
4.2	Multi-Particle Collision Algorithm	52
4.2.1	Initial set of particles	54
4.2.2	Perturbation function	55
4.2.3	Exploitation function	55
4.2.3.1	Exploitation in a single random dimension each time	56
4.2.4	Scattering function	57
4.2.5	Blackboard updating function	58
4.2.6	Stopping criteria	58
4.3	Opposition-Based Optimization and some mechanisms derived	58
4.4	Multi-Particle Collision Algorithm with Opposition-Based Optimization derived mechanisms	62
4.4.1	Initialization using Opposition	63
4.4.2	Traveling in the search space using Opposition	64
4.5	q -Gradient Method	65
4.5.1	Stopping criteria	67
4.6	Hooke-Jeeves Pattern Search Method	68
4.6.1	Exploratory move	70
4.6.2	Pattern move	70
4.6.3	Stopping criteria	71
4.7	Hybrid approaches: OMPCA-HJ and q G-HJ	71
4.8	Chapter conclusions	72
5	VIBRATION-BASED DAMAGE IDENTIFICATION ON SYSTEMS/STRUCTURES MODELED WITH FORTRAN, AND USING <i>IN SILICO</i> EXPERIMENTAL DATA	73
5.1	Experimental design	73

5.1.1	Noise in data	75
5.2	Case study 1: Damage identification in a damped spring-mass system with 10-DOF	75
5.3	Case study 2: Damage identification in a truss structure with 12-DOF . .	78
5.4	Case study 3: Damage identification in a model with a similar shape to the International Space Station	81
5.5	Case study 4: Damage identification in the Kabe's problem	87
5.6	Case study 5: Damage identification in the Yang's problem	92
5.7	Case study 6: Damage identification in a cantilevered beam	96
5.7.1	Results using a full dataset, with a single damage	97
5.7.2	Results using a full dataset, with a single damage, applying regularization	98
5.7.3	Results using an incomplete dataset, with single damage	98
5.7.4	Results using a full dataset, with mixed multi-damage	100
5.7.5	Results using an incomplete dataset, with mixed multi-damage	102
5.8	Chapter conclusions	105
6	VIBRATION-BASED DAMAGE IDENTIFICATION ON SYSTEMS/STRUCTURES MODELED WITH NASTRAN, AND USING <i>IN SILICO</i> EXPERIMENTAL DATA	109
6.1	Experimental design	109
6.2	Case study 1: Damage identification in a spring-mass system with two degrees of freedom	111
6.3	Case study 2: Damage identification in a spring-mass system with ten degrees of freedom	113
6.4	Case study 3: Damage identification in a cantilever truss	115
6.5	Case study 4: Damage identification in a cantilever beam	117
6.6	Chapter conclusions	120
7	CONCLUSIONS AND RECOMMENDATIONS	121
7.1	Conclusions	121
7.1.1	Main contributions	122
7.2	Recommendations	123

REFERENCES	125
FEATURES AND CLASSIFICATION APPROACHES FOR VIBRATION-BASED DAMAGE IDENTIFICATION	155
USING NASTRAN FOR TRANSIENT RESPONSE ANALYSIS . .	157
B.1 NASTRAN files	157
B.1.1 Input file	157
B.1.2 Output files	159
B.2 Using NASTRAN for defining structures	159
B.2.1 Setting the structures geometry	159
B.2.2 Defining material properties	160
B.3 Using NASTRAN for transient response analysis	161
B.3.1 Integration time step	161
B.3.2 Time-dependent loads	161
B.3.3 Dynamic Load Tabular Function	162
B.3.4 Spatial distribution of loading	162
B.3.5 Initial conditions	163
B.3.6 Single-point constraints	163
B.3.7 Damping	163

1 INTRODUCTION

*Research is what I'm doing when I don't know what
I'm doing.*

WERNHER VON BRAUN

The Brazilian aerospace industry is strongly represented in the world by Embraer (in portuguese *Empresa Brasileira de Aeronáutica*). This company, through its aircraft, is one of the leading exporters of Brazil, with the highest added value product. Examples of aircraft produced by Embraer are the regional jet E190 (Figure 1.1(a)), the military KC-390 (Figure 1.1(b)) and the business jet Legacy 650 (Figure 1.1(c)).

Figure 1.1 - Aircrafts developed by Embraer: (a) E190; (b) KC390; (c) Legacy 650

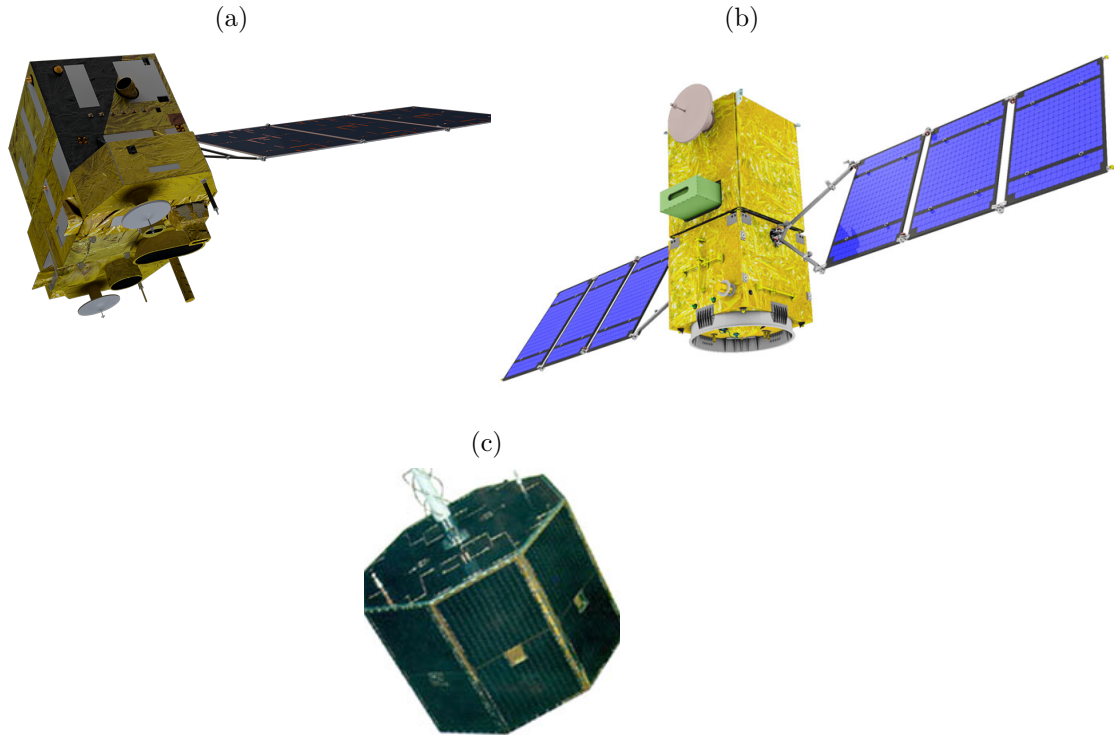


SOURCE: (a) Embraer (2017a) (b) Embraer (2017b) (c) Embraer (2017c)

Another institution that works with aerospace structures is the National Institute for Space Research (in Portuguese: *Instituto Nacional de Pesquisas Espaciais* (INPE)). INPE develops satellites for Earth observation in projects together with other countries. Examples of satellites developed by INPE are the China-Brazil Earth Re-

sources Satellite Program (CBERS, see Figure 1.2(a)), the Amazônia project (see Figure 1.2(b)) and the Data Collection Satellite (SCD, see Figure 1.2(c)).

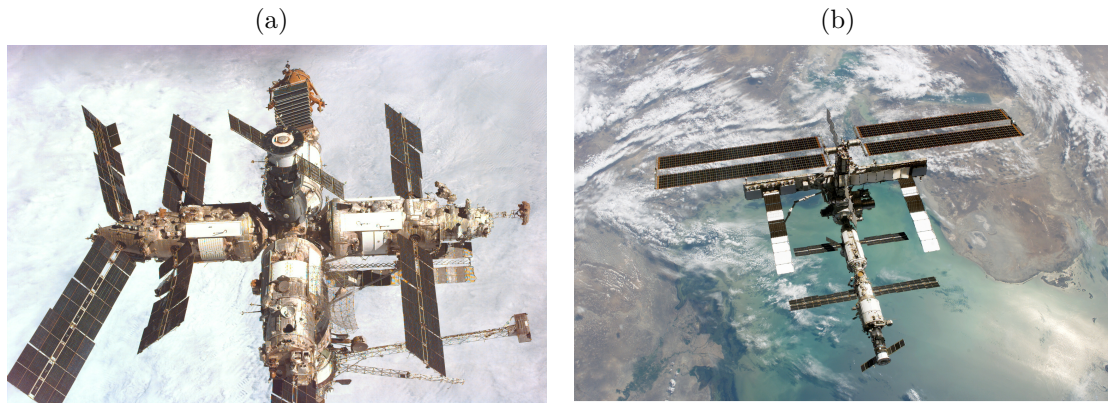
Figure 1.2 - Satellites developed by INPE: (a) CBERS-3; (b) Amazônia-1; (c) SCD-1



SOURCE: (a) INPE (2011) (b) INPE (2017a) (c) INPE (2017b)

Worldwide, Large Space Structures are developed and sent to outer space. A Space Station (SS) is a spaceship that is held for an extended period in orbit, supporting crew-members, and also allowing other spacecraft to dock. Example of SS are the Salyut programme from the former Soviet Union, Skylab from United States, MIR (see Figure 1.3(a)) from Soviet Union and Russia, the International Space Station (ISS, see Figure 1.3(b)), and the Tiangong programme from China. All these structures are composed of large parts including solar wings, radar, antennas, shields, and booms. Those elements are exposed to damage, caused by space debris or meteor strikes.

Figure 1.3 - Space Stations: (a) MIR; (b) ISS



SOURCE: (a) NASA (2017a) (b) NASA (2017b)

The design of an adequate computer model for the structural analysis is fundamental in the development and construction process of engineering structures. Computer-Aided Engineering (CAE) is the extensive use of computer software to assist in engineering analysis tasks, including Finite Element Analysis (FEA), Computational Fluid Dynamics, Multibody Dynamics, and optimization of products or processes. Two well known CAE software for solving problems by FEA are ANSYS¹, and MSC NASTRAN² (LOGAN, 2011). Other FEA software can be found in MECHANICAL ENGINEERING (2011), FEA Compare (2017). Those programs are widely used in the design and analysis processes of structures in aerospace industry (BIRTLES, 1983). For example, Embraer and INPE adopted NASTRAN, which performs different engineering analysis, such as static, dynamic, and thermal analyse.

Aerospace structures are critical systems. Failures in structures would cause catastrophic consequences, with human and material losses. Hence, it is essential to have the capability of detecting damages in an early stage with accuracy and safely. This detection should help to repair or rehabilitate a structure before it has major damages.

1.1 Damages

Damages in the structural context can be defined as changes to the material properties or geometry of a structure, including changes in the boundary conditions and the system connectivity, which affect its current or future performance (KHOSH-

¹<http://www.ansys.com/>

²<http://www.mscsoftware.com/product/msc-nastran>

NOUDIAN; ESFANDIARI, 2011). Generally speaking, damage produces a loss of local stiffness in a structure (JIMÉNEZ et al., 2004). The damage identification is linked to a comparison of two different states of a structure, one of which is the initial state, named undamaged, and the other is the current structural response under a forcing term. The existence of damage does not imply that the system/structure losses all its functionality, but rather that it does not operate in its standard manner.

Examples of damages are, among others, cracks, local plasticity, delamination and debonding in composite materials.

A damage can be categorized in accidental damage, fatigue damage, or environmental damage (BAARAN, 2009; VENTERINK, 2017). Damages associated with fatigue or corrosion in metals can accumulate incrementally over periods of time. For composites, inspection of fatigue and environmental damages are often not required, if proper design precautions are taken.

In short time-scales, accidental damages can be the result of scheduled discrete events such as aircraft landings, or unscheduled events, such as bird-strikes in aircraft or natural phenomena hazards.

As damages increase, the operation of the system or structure will no longer be acceptable to the user, *i.e.*, a failure will occur. In this case, the term user represents another system (physical or human) that interacts with the former. An unpredicted structural failure can cause catastrophic loss of economic value and human life.

Damages in aerospace structures can be categorized according to their severity (ILCEWICZ, 2006 apud BAARAN, 2009) (PRASAD; WANHILL, 2017) in:

Category 1 Allowable damage that may go undetected by scheduled or direct field inspection, for example, allowable manufacturing defects, but also non-allowable damage: Barely Visible Impact Damage, *e.g.*, minor environmental degradation, scratches, damage caused by debris or hailstones.

Category 2 Damage detected by scheduled or direct field inspection at specified intervals, *e.g.* major environmental degradation, exterior skin damage, or interior stringer blade damage.

Category 3 Obvious damage detected within a few flights by operations, *e.g.* accidental damage to the lower fuselage, or lost bonded repair patch.

Category 4 Discrete source damage immediately known by the pilot limiting flight maneuvers, *e.g.* rotor disk cut through the fuselage or severe rudder lightning damage.

Category 5 Severe damage created by anomalous ground or flight events, *e.g.* bird-strike, maintenance jacking incident, propeller mishap.

1.2 Structural Damage Identification

The Structural Damage Identification (SDI) process involves the conjunction of five disciplines ([FARRAR; WORDEN, 2010](#); [FARRAR; LIEVEN, 2007](#)):

Structural Health Monitoring Structural Health Monitoring (SHM) performs a global damage identification for aerospace, civil and mechanical engineering infrastructure, and it can be performed off-line as well as on-line. On-line refers to the monitoring during operation of the system or structure, while off-line refers to the monitoring during maintenance;

Condition Monitoring Condition Monitoring is analogous to SHM, but in rotating and reciprocating machinery;

Non-Destructive Evaluation Non-Destructive Evaluation (NDE) carried out off-line in a local manner after the damage has been located;

Statistical Process Control Statistical Process Control is based on process and uses a variety of sensors to monitor changes in a process;

Damage Prognosis Damage Prognosis evaluates the damage evolution and predicts the remaining useful life of a system.

All these processes involve the observation of a structure or a mechanical system for a period, by means of periodically spaced measurements, the extraction of damage-sensitive features from the measurements, and the analysis of these features to determine the current state of the structural health.

The process of damage identification involves four steps ([FARRAR et al., 2001](#); [FARRAR et al., 2009](#)):

Operational Evaluation Gives details of the implementation of the process, such as possible failure modes, operational and environmental conditions, and limitations related to data acquisition.

Data Acquisition Defines the quantities to be measured, the amount of data to be saved, and how the data will be acquired (which and how many sensors will be used, and where the sensors will be placed). Also, defines the data fusion and cleansing, that determine which data is necessary and useful in the feature extraction process.

Feature Extraction Identifies damage sensitive parameters from measured data.

Classification Performs the comparison between the damaged and undamaged states and quantifies the damage.

An ideal robust damage identification scheme should be able to detect a damage at a very early stage, to locate the damage within the sensor resolution, to provide an estimate of the extent or severity of the damage, and to predict the remaining useful life of the structural component when the damage is identified. The identification should be independent of changes in the operational and environmental conditions. The method should also be well suited to automation and should be independent of human judgment and ability (DOEBLING et al., 1996).

1.2.1 Classification of the Structural Damage Identification methods

Damage identification methods can be classified in different ways: according to their performance, and to whether they are local or global, model or non-model based, and whether they use a baseline or not.

Performance The damage identification process for a system of a structure can be classified attending five levels of performance (RYTTER, 1993; SOHN et al., 2004; WORDEN; DULIEU-BARTON, 2004; WORDEN et al., 2007):

Level 1: Detection Verification of the presence of damage in a structure;

Level 2: Location Determination of the location of the damage;

Level 3: Type Determination of the type of the damage;

Level 4: Extension Estimation of the extent and severity of the damage;

Level 5: Prognosis Prediction of the remaining service life of the structure.

The first four levels (detection, location, type, and extension) are related to the dam-

age diagnosis, while the fifth level is related to the damage prognosis. Increasing the level implies a higher degree of complexity and a greater necessity of mathematical models.

Local or global methods

This classification is related to the inspection domain concerning the structure (FRITZEN; KRAEMER, 2009):

Local Concentrated on a specific part of the structure.

Global The whole structure is analyzed.

Local methods are usually considered to be more sensitive than global methods. Local methods are capable of detecting small damages, while global ones can identify only relatively severe damages. However, local methods are applicable with a prior knowledge of the location of a damaged area of the structure.

Model based and non-model based approach

Methods can also be classified on model based and non-model based:

Non-model based The response of the structure in service is compared with results of a reference measurement previously acquired. Deviations in damage sensitive parameters are used to identify damage.

Model based The response is compared with the results obtained from an analytical or numerical model.

Advantages of model based over the non-model based techniques include the possibility of the former to be easily extended to provide information about the severity of the detected damage and can be used to account for environmental or operational variations, *e.g.* temperature and boundary conditions. Disadvantages include the difficulty in obtaining an accurate model representation of complex structures, and its limited applicability for *in situ* monitoring, due to their computational cost.

Baseline or non-baseline

The process of identifying a damage requires the comparison of two states, one of them being the undamaged reference. Therefore, methods can also be classified according to whether a baseline is used or not (WORDEN et al., 2007):

Baseline The response of the structure, measured at an earlier stage, is usually utilized as a baseline to distinguish between the damaged and undamaged states.

Non-baseline The response is compared with a state of the structure assumed to have a normal behavior, *e.g.* a smooth pattern or linear-elastic response. The system, in this case, is classified as damaged when the response deviates from the expected behavior.

SDI can be treated as two complementary approaches ([MONTANARI, 2014](#)):

Inverse problem This approach typically uses a model of the structure and tries to relate changes in measured data from the structure to changes in the model parameters ([FRISWELL, 2007](#)). Locally linearized models are sometimes used to make the analysis simpler. The behavior of the real structure is compared to the corresponding model, through optimization algorithms.

Pattern recognition problem This approach follows the concepts of the Pattern Recognition problem ([SCHALKOFF, 1992](#) apud [MONTANARI, 2014](#))

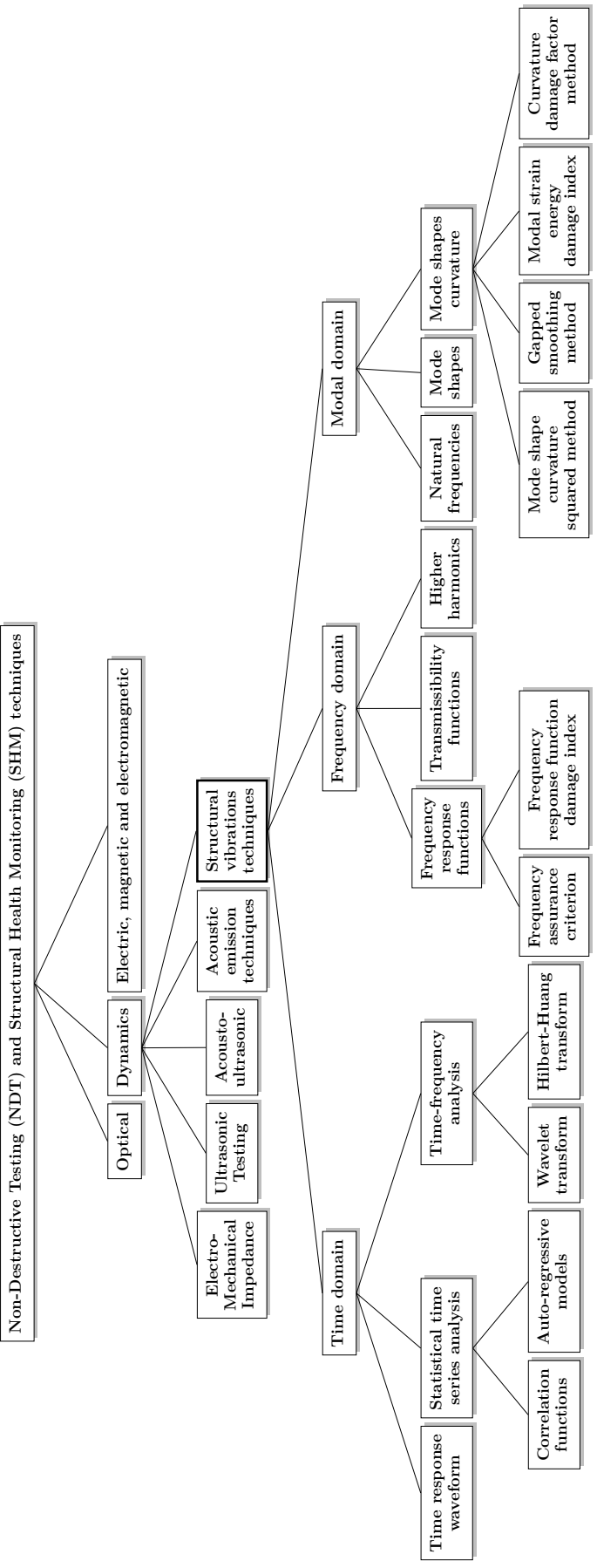
1.3 Literature overview of techniques for Structural Damage Identification

A variety of NDE techniques can be used for damage identification. Most of them can only be applied off-line. Thus, only a minor quantity of these techniques can be used in an SHM system.

[Figure 1.4](#) shows a scheme of these techniques (first three lines), that include the Structural Vibration (SV) and Electromechanical Impedance (EMI) techniques which primarily rely on standing wave patterns. Another group, including the higher frequency Acoustic Emission (AE), Acousto-Ultrasonic (AU), and Ultrasonic Testing (UT), employ traveling wave characteristics.

[Table 1.1](#) shows a comparison of the NDE techniques performances. SV and EMI provide results that are relatively easy to interpret. More complex structures can be analyzed with these methods, allowing to explore outright a relatively large area. However, in these methods the limited frequency range causes a limited resolution, and only relatively severe damage, such as delaminations, can be identified. AE, AU and UT can all detect small damages, such as cracks. For this reason, these wave

Figure 1.4 - Overview of damage identification methods. The branch of vibration-based methods is detailed with different features and classifiers used in the literature



SOURCE: Adapted from Ooijsaar (2014)

based technologies are more frequently used for aircraft applications (STASZEWSKI et al., 2009; DIAMANTI; SOUTIS, 2010). However, the interpretation of the data is more complicated than with SV and EMI, especially in the case of non-flat or composite structures (DALTON et al., 2001).

Table 1.1 - Dynamics based NDE technologies. The sensibility to the damage, the data interpretation, and the applicability for SHM are evaluated on a scale of 1 (less) to 5 (more) points, shown with plus signs (+).

Technology	Frequency range [Hz]	Actuation approach	Sensitivity to damage	Ease of data interpretation	Applicability for SHM
SV	$10^0 - 10^4$	Active / passive	++	+++	++++
EMI	$10^3 - 10^5$	Active	+++	+++	++++
AE	$10^4 - 10^6$	Passive	++++	++	+++
AU	$10^4 - 10^6$	Active	++++	+	++++
UT	$10^5 - 10^7$	Active	+++++	++	+

SOURCE: Adapted from Ooijselaar (2014)

The aeronautic engineering field gives a great attention to Vibration-based SHM and other damage detection techniques (LIU; NAYAK, 2012). A good methodology for SHM can improve the efficiency of the structural maintenance, with the subsequent reduction of the cost and the increase of the reliability of the structures (MEDEIROS, 2016).

1.3.1 Methods based on structural vibration for Structural Damage Identification

The literature dealing with SV methods is vast. Different methods are classified depending on their approach, *e.g.*, frequency changes, mode shape changes, dynamically measured flexibility, matrix update methods, nonlinear methods, and Neural Networks-based methods. These kind of methods are the subject of books (BALAGEAS et al., 2006; BOLLER et al., 2009), doctoral theses (OOIJEVAAR, 2014; MEDEIROS, 2016), and review papers, some of them on specific topics (composite materials, strain based methods, nonlinear methods) (DOEBLING et al., 1996; DOEBLING et al., 1998; ZOU et al., 2000; SOHN et al., 2004; CARDEN; FANNING, 2004; MONTALVAO et al., 2006; FASSOIS; SAKELLARIOU, 2007; WORDEN et al., 2008; JHANG, 2009; FRITZEN; KRAEMER, 2009; LI, 2010; FAN; QIAO, 2011). Figure 1.4 summarizes the principal damage sensitive features and classifiers found in the literature on the

Vibration-based Damage Identification methods category. These methods can be divided into three principal areas: time, frequency and modal domains. Table 1.2 shows an overview of these features, while Table A.1 in Appendix A shows a more detailed list of classifiers associated with the features.

Doebling et al. (1998) affirmed that sufficient evidence exists to promote the use of measured vibration data for damage identification in structures, using forced-response tests and long-term monitoring of ambient signals.

Table 1.2 - Overview of the vibration based damage features

Time	Time-frequency	Frequency	Modal
Time response / waveform	Wavelet transform	Fourier / power spectra	Natural frequencies
Statistical time series analysis	Hilbert transform	Frequency response function	Mode shape
	Hilbert-Huang transform	Frequency response function curvature	Mode shape curvatures
		Mechanical impedance	Modal damping
		Transmissibility function	Dynamic stiffness
		Anti-resonances	Dynamic flexibility
		Higher harmonics (nonlinear)	Updating methods
		Modulation (nonlinear)	

SOURCE: Adapted from Ooijsaar (2014)

1.3.2 Metaheuristics applied to Structural Damage Identification

Table 1.3 shows a compilation of papers where the SDI problem is approached using different Computational Intelligence techniques. They can be divided into five groups: neural network techniques, support vector machine, fuzzy inference, metaheuristic algorithms, and hybrid techniques. Among them, the formulation of the SDI problem as a constrained optimization problem is typically considered the most effective strategy (YU; XU, 2011).

Table 1.3 - Literature overview of Computational Intelligence methods used for SDI

Neural networks	
Iterative Neural Network (INN)	(CHANG et al., 2000)
Backpropagation NN (BP-NN)	(ZANG; IMREGUN, 2001)
	(YAM et al., 2003)
	(PAWAR et al., 2006)
	(MEHRJOO et al., 2008)
	(CHIWIACOWSKY et al., 2008)
Multilayer Neural Network (MNN)	(HUANG; LOH, 2001)

Continued on next page

Table 1.3 – *Continued from previous page*

Extended Kalman Filter Trained Neural Network (EKFNN)	(JIN et al., 2016)
Support Vector Machine (SVM)	
Hyper-Solution SVM (HSVM)	(CANDELIERI et al., 2014)
Fuzzy	
Fuzzy Clustering (FC)	(SILVA et al., 2008)
Fuzzy Logic System (FLS)	(CHANDRASHEKHAR; GANGULI, 2009)
Metaheuristics	
Genetic Algorithm (GA)	(BOONLONG, 2014)
	(BORGES et al., 2007)
	(CHOU; GHABOUSSI, 2001)
	(GOMES; SILVA, 2008)
	(MARES; SURACE, 1996)
Improved PSO (IPSO)	(YU; WAN, 2008)
Tabu Search (TS)	(ARAFAT et al., 2010)
Continuous ACO (CnACO)	(YU; XU, 2011)
Ant Colony Optimization (ACO)	(MAJUMDAR et al., 2012)
	(YU; XU, 2011)
Swarm Intelligence (SI)	(YU et al., 2012)
Particle Swarm Optimization (PSO)	(MOHAN et al., 2013)
	(SEYEDPOOR, 2012)
Big Bang-Big Crunch (BB-BC)	(TABRIZIAN et al., 2013)
Simulated Annealing (SA)	(KOUREHLI et al., 2013)
Differential Evolution (DE)	(FU; YU, 2014)
	(SEYEDPOOR et al., 2015)
	(SEYEDPOOR; YAZDANPANA, 2015)
Charged System Search (CSS)	(KAVEH; ZOLGHADR, 2015)
Multi-Swarm Fruit Fly Optimization Algorithm (MFOA)	(LI; LU, 2015)
Global Artificial Fish Swarm Algorithm (GAFSA)	(YU; LI, 2014)
Rank-based Ant System (RAS)	(BRAUN et al., 2015)
Ant System with Heuristic Information (ASH)	(BRAUN et al., 2015)
Elitist Ant System (EAS)	(BRAUN et al., 2015)
Firefly Algorithm (FA)	(PAN; YU, 2015)
Cyclical parthenogenesis algorithm (CPA)	(KAVEH; ZOLGHADR, 2017)
Hybrid algorithms	
Genetic Algorithm + Conjugated Gradient Method (GA-CGM)	(CHIWIACOWSKY et al., 2006)
	(CAMPOS VELHO et al., 2006)
Genetic Algorithm + Levenberg-Maquardt (GA-LM)	(HE; HWANG, 2006)
Genetic Fuzzy System (GFS)	(PAWAR; GANGULI, 2007)
Simulated Annealing Genetic Algorithm (SAGA)	(KOKOT; ZEMBATY, 2009)
Genetic Algorithm and Particle Swarm Optimization (GA-PSO)	(SANDESH; SHANKAR, 2010)
Ant System with Heuristic Information + Hooke-Jeeves (ASH-HJ)	(BRAUN et al., 2015)
PSO with Nelder-Mead (PSO-NM)	(CHEN; YU, 2015)
Multi-Particle Collision Algorithm with Hooke-Jeeves (MPCA-HJ)	(HERNANDEZ TORRES et al., 2015)
q -Gradient with Hooke-Jeeves (q G-HJ)	(HERNÁNDEZ TORRES et al., 2015b)
Firefly Algorithm / Genetic Algorithm	(KHATIR et al., 2016)
Quick Artificial Bee Colony algorithm (QABC)	(DING et al., 2016)
Particle Swarm Inspired Multi-Elitist Artificial Bee Colony (PS-MEABC)	(FATAHI; MORADI, 2017)

Some authors have used the classic Genetic Algorithm (GA) for solving the Damage

Identification problem. For instance, [Mares and Surace \(1996\)](#) located and quantified damages in truss and beam structures; [Borges et al. \(2007\)](#) identified damages in a framed structure; and recently, [Boonlong \(2014\)](#) used Cooperative Coevolutionary Genetic Algorithm as the optimizer for the vibration-based damage detection in beams.

Other classical metaheuristics have also been used for this purpose. A continuous variant of Ant Colony Optimization (ACO) was used by [Yu and Xu \(2011\)](#) to detect single and multiple damages on 2 and 3-storey building models; they suggested the use of hybrid approaches of ACO to improve the results. In [Braun et al. \(2015\)](#), different versions of ACO, as well as hybrid variants using Hooke-Jeeves direct search method, were used to detect damages on a damped spring-mass system.

A combination of a self-configurable Particle Swarm Optimization (PSO) with the Nelder-Mead (NM) algorithm has also been used for optimal localization of sensors for health monitoring ([RAO; ANANDAKUMAR, 2007](#)). In [Abdalla \(2009\)](#), PSO was also applied in the solution of the inverse problem of SDI using a cantilever beam model. The hybrid PSO-simplex (PSOS) was used in two SDI problems with noisy and incomplete data, in the frequency domain. [Begambre and Laier \(2009\)](#) used Latin hypercube sampling and PSO for an experimental verification of the structural damage detection process. [Kang et al. \(2012\)](#) proposed an immunity enhanced particle swarm optimization (IEPSO) algorithm for damage detection, which was used for a simply supported beam, and on a plane truss structure. Also, [Baghmisheh et al. \(2012\)](#) hybridized PSO with Nelder-Mead (NM) for crack detection in cantilever beams.

Recently, the novel Bird Mating Optimizer (BMO) was used in [Zhu et al. \(2017\)](#) for identifying structural damages using the minimization of the differences between the measured and calculated natural frequencies and correlation function acceleration vector of damaged and undamaged structures. This method was used for identifying damages on a planar frame structure, and a three connected shear building.

Novel metaheuristics have also been presented for solving the task of identifying damages. For example, the Big Bang-Big Crunch (BBC), in [Tabrizian et al. \(2013\)](#), is used to identify damages in beam bridges, a Belgian truss (which contains 21 truss elements, 12 nodes and 21 nodal degrees-of-freedom), and a two-story two-bay frame. The proposed optimization algorithm minimizes an objective function using complete and incomplete modal data, with and without noise. In [Yu and Li \(2014\)](#), single and multiple damage cases are detected in a simulated 4-storey benchmark

frame structure, and in a 2-storey rigid frame, using the global Artificial Fish Swarm Algorithm (GAFSA). Also, an experimental laboratory study on damage detection was performed in a 3-storey building model, with four damage patterns. [Ding et al. \(2016\)](#) presented the quick ABC (QABC), an artificial bee colony (ABC) algorithm with hybrid search strategy based on modal data. QABC algorithm is applied in the damage identification of a truss and a plate.

1.3.3 Gaps found in the literature

As discussed in [Section 1.3.1](#), many damage identification techniques have been proposed in the literature. However, their practical application can be difficult ([DOEBLING et al., 1998](#); [OOLJEVAAR, 2014](#)).

There are uncertainties in damage identification ([HE JIA; XU, 2017](#)), mainly caused by:

- Methodology errors:
 - Limitations of the damage identification methods.
- Modeling errors:
 - Inaccuracy in the discretization of the FE model;
 - Uncertainties in the geometry and boundary conditions;
 - Variations in material properties;
 - Operational and environmental variability.
- Measurement errors:
 - Errors in the measured signals, caused by noise, or by procedures in the measurements;
 - Errors in the post-processing techniques.

Some issues that need to be solved to enable the actual practice of these methods are summarized below:

Lack of robustness None of the methods solves all problems for all structures.

Occurrence of False alarms One of the main challenges in damage identification is the development of methods that provide a high detection probability, without false alarms. There are two types of false alarms in

damage identification: *false-positive*, when an intact element is identified as damaged, and a *false-negative*, that is a failure in identifying damaged elements. False alarms cause a decrease in the reliability of the method. False-negative could cause serious life and safety consequences, while false-positive could increase in maintenance costs.

Low-performance level Most of the methods attends the first two levels of performance (detecting and locating damage), but they are limited in their capability to achieve a prognosis, as well as in classifying the type and severity of the damage (OOIJEVAAR, 2014). The prognosis is the prediction of the remaining lifetime of the structure through the analysis of the damage evolution. It represents an important step for the development of autonomous systems for monitoring the structural health.

Low complexity structures Most of the methods deals with relatively low complexity structures (simple concrete or metallic structures, or relatively simple composite beams and plates structures). These structures are typically well-defined, or the damage scenario is created artificially. It is necessary to focus on more specific applications, such as real life and industrial structures like air-frames, bridges, and other long life structures with life-safety or economic implications.

No integration with engineering software CAE software help in the task of modeling and analyzing structures. Among the papers reviewed, the use of these computer programs is scarcely mentioned.

No operational and environmental variability In real applications, the operational and environmental conditions, such as temperature and wind, could mask or modify the effects of the damages. Good approaches should have the capability of separating their effects.

Integrated sensor and network Another problem is related to the optimal quantity and location of sensors because they are expensive. Also, obtaining many signals increases the computational cost, and could add errors due to the inherent noise of the sensors and some possible external noises.

Cooperation Due to the magnitude of the projects, more cooperation is required between academia and industry.

More detailed discussions on these problems are provided by [Farrar and Worden](#)

(2007) and Boller (2013).

In this thesis, the SDI will be defined as an inverse problem and solved as an optimization problem.

1.4 Research objective

The main objective of this thesis is:

To develop hybrid metaheuristics for identifying damages on structures modeled both in FORTRAN language and NASTRAN system.

This work will contribute to the improvement of the Multi-Particle Collision Algorithm and the q -gradient method, to the creation of hybrid algorithms, and to their application in the development of a tool to assist the implementation of structural health monitoring technology in aerospace structures.

Technical contributions associated to this thesis are the refactoring of the FORTRAN code for Finite Element Model of structures, and the refactoring of the FORTRAN code of the Multi-Particle Collision Algorithm.

Another technical contribution is the handling of the input and output files of the NASTRAN software, using an interface with FORTRAN.

1.4.1 Tasks

Specific tasks and sub-tasks are defined to accomplish the main objective:

- a) Perform a literature review on methods for Damage Identification;
- b) Perform modifications in the Multi-Particle Collision Algorithm, for performance improvement;
 - Modify functions for improving the exploration and the exploitation of the algorithm.
- c) Develop hybrid algorithms based on MPCA and q -gradient, using concepts derived from the Opposition Based Learning, and the direct-search method Hooke-Jeeves;
- d) Identify damages on structures with different complexities implemented in

FORTRAN;

- Model structures with different types and complexities in FORTRAN;
- Identify damages on structures implemented using finite elements on FORTRAN using hybrid algorithms.

e) Identify damages on structures with different complexities modeled on NASTRAN;

- Model structures with different types and complexities in NASTRAN;
- Identify damages on structures modeled on NASTRAN using hybrid algorithms.

1.5 Outline

This thesis is structured into eight chapters, including introduction and conclusions, and three appendixes, organized as follow:

Chapter 1 INTRODUCTION (this chapter) This first chapter served as an introduction to the SHM-related concepts, and to give information about the state-of-the-art on the SDI methods. The research objectives, and main tasks were also addressed.

Chapter 2 VIBRATION-BASED DAMAGE IDENTIFICATION SOLVED AS OPTIMIZATION PROBLEM In this chapter the solution of the inverse problem is defined.

Chapter 3 TRANSIENT RESPONSE ANALYSIS OF STRUCTURES AS DIRECT MODEL In this chapter the transient response analysis of structures is presented as the forward model. The mathematical definition and the numerical solution of the forward method is then summarized. The Finite Element Method is introduced and presented the explanation of how it is applied in the task of building systems and structures using springs, bars and beams.

Chapter 4 HYBRID ALGORITHMS FOR SOLVING THE DAMAGE IDENTIFICATION PROBLEM The hybrid algorithms are presented. Initially the Multi-Particle Collision Algorithm is introduced.

Later, the Opposition mechanisms and variants are presented, and their application for the improvement of the MPCA is detailed. Other algorithm that is presented in this chapter is the novel q -gradient method. Thereafter, the Hooke-Jeeves direct search method is briefly presented. Finally, the hybrid scheme that is used in the thesis is shown.

Chapter 5 VIBRATION-BASED DAMAGE IDENTIFICATION ON SYSTEMS/STRUCTURES MODELED WITH FORTRAN, AND USING *IN SILICO* EXPERIMENTAL DATA This chapter presents the results obtained using the methodology on six case studies with different complexities implemented in Fortran.

Chapter 6 VIBRATION-BASED DAMAGE IDENTIFICATION ON SYSTEMS/STRUCTURES MODELED WITH NASTRAN, AND USING *IN SILICO* EXPERIMENTAL DATA This chapter presents the results obtained using the methodology on four case studies with different complexities modeled in NASTRAN.

Chapter 7 CONCLUSIONS AND RECOMMENDATIONS The last chapter presents a summary of the content of this thesis, as well as concluding remarks and recommendations for further works.

References

Appendix A FEATURES AND CLASSIFICATION APPROACHES FOR VIBRATION-BASED DAMAGE IDENTIFICATION This appendix presents the main features and classification approaches found in the literature, with an exhaustive bibliographic revision.

VITA Finally, the vita of the author of this thesis is presented, including full papers on journals, national and international conferences that directly or indirectly contributed to the doctoral thesis.

2 VIBRATION-BASED DAMAGE IDENTIFICATION SOLVED AS OPTIMIZATION PROBLEM

A problem well-stated is a problem half-solved.

JOHN DEWEY

In the first chapter, the main concepts on the SDI were introduced. A literature review was performed on the different methods for SDI, and the classification of those methods was presented.

In this chapter, the Vibration-based Damage Identification (VDI) will be formulated as an optimization problem.

Initially, the main concepts about inverse problems in engineering will be presented. Then, the VDI will be formulated as an optimization problem, and the damage parameters and the objective function will be defined.

2.1 Basic concepts of optimization

Optimization is the area of the Applied Mathematics that studies the theory and techniques to find the best available values to optimize (minimize or maximize) some objective function, also called error function or cost function. Typical applications of optimization are to find the minimal cost, maximal profit, minimal error, optimal design and optimal management. Many books handle the concepts of the optimization area, such as Rothlauf (2011).

Definition 2.1. An **optimization problem** is the problem of finding the best solution from all feasible solutions.

A solution is **feasible** if it satisfies all the constraints.

Definition 2.2. A **candidate solution** s is an element of the search space \mathcal{S} of a certain optimization problem.

Definition 2.3. The **search space** \mathcal{S} of an optimization problem is the set containing all elements s which could be its solution.

An optimization problem can be classified depending on the existence of constraints. In an **unconstrained** optimization problem, the objective function depends on variables with no restrictions, and in a **constrained** optimization problem, the objective function is subject to constraints on the variables.

Definition 2.4. The standard form of an unconstrained optimization problem is:

$$\begin{aligned} & \underset{\mathbf{s}}{\text{minimize}} && J_i(\mathbf{s}), \quad i = 1, \dots, n_j \\ & \text{subject to} && \mathbf{s} \in \mathcal{S} \end{aligned} \tag{2.1}$$

where $\mathbf{s} = (s_1, s_2, \dots, s_D)$ is a vector of D **decision variables**, $J_i : \mathbb{R}^D \rightarrow \mathbb{R}$ are the **objective functions** to be minimized over the \mathbf{s} , and $\mathcal{S} = \mathbb{R}^D$.

By convention, the standard framework is defined as the minimization problem. Some strategies can be used to define the maximization problem, such as taking the negative sign to the objective function.

Definition 2.5. The **globally optimal solution** $\mathbf{s}^* \in \mathcal{S}$ of an objective function J is defined as:

$$J(\mathbf{s}^*) \leq J(\mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S}. \tag{2.2}$$

Definition 2.6. A **locally optimal solution** $\mathbf{s}' \in \mathcal{S}$ of an objective function J with respect to a neighborhood function \mathcal{N} is defined as:

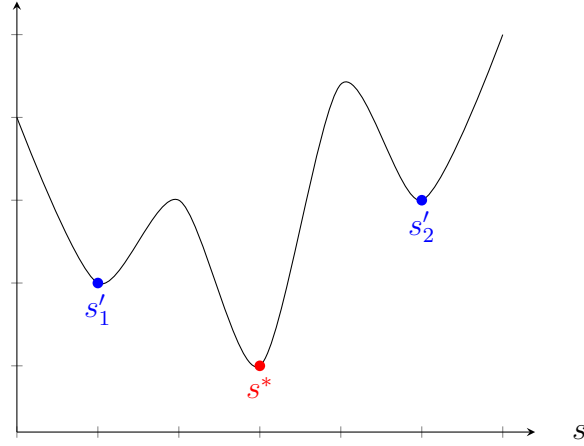
$$J(\mathbf{s}') \leq J(\mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{N}(\mathbf{s}'). \tag{2.3}$$

If $\mathcal{S} \subseteq \mathbb{R}^D$:

$$\forall \mathbf{s}', \quad \exists \varepsilon > 0 : J(\mathbf{s}') \leq J(\mathbf{s}) \forall \mathbf{s} \in \mathcal{S}, |\mathbf{s} - \mathbf{s}'| < \varepsilon \tag{2.4}$$

Figure 2.1 shows a graphical representation of some local and global minima for a problem with one variable.

Figure 2.1 - Local (s'_1 and s'_2) and global minima (s^*)
 $J(s)$



Optimization problems can be categorized according to the solution domain. They can be divided into **continuous** when the variables are allowed to get any value within a range; or **discrete**, when the variables are required to belong to a discrete set. This set could be a subset of integers (called *integer programming*) or a set of objects or combinatorial structures (called *combinatorial optimization*). Also, some problems exist where the domain is **mixed**.

The nature of the variables divides the optimization problems into two groups: **deterministic** when the data for the given problem are known accurately, and **stochastic** or non-deterministic when some or all the design variables are probabilistic.

The optimization problems can also be classified according to the number of objective functions. A **mono-objective** problem is when there is only one objective function, while a **multi-objective** problem have more than one objective function that can be minimized or maximized simultaneously.

The standard form of a constrained mono-objective optimization problem is:

$$\begin{aligned}
 & \underset{s}{\text{minimize}} && J(s) \\
 & \text{subject to} && g_j(s) \leq 0, \quad j = 1, \dots, n_g \\
 & && h_k(s) = 0, \quad k = 1, \dots, n_h
 \end{aligned} \tag{2.5}$$

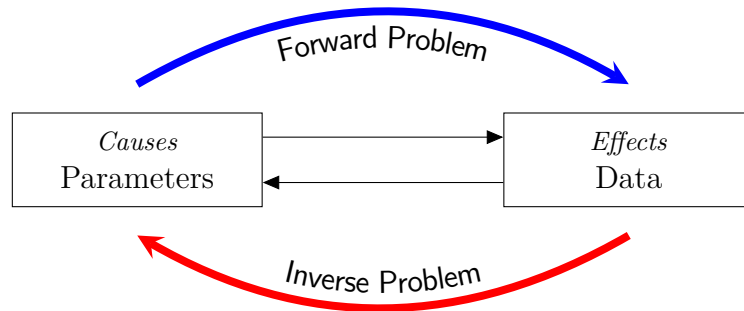
where g_j are called inequality constraints, and h_k are called equality constraints.

2.2 Inverse problems in engineering

A model can be simply described as the relationship between the parameters and the data, that is useful for simulating and predicting aspects of the behavior of a system. A model represents a real system as symbols using the language of other sciences. For instance, an engineering process may be modeled using mathematical symbols and computer sciences (ALAVALA, 2008).

The process of finding data based on a model using a set of parameters as input is called as **forward problem** while estimating a set of parameters of a model based on a set of data is known as **inverse problem** (IP) (CAMPOS VELHO, 2001; SILVA NETO; MOURA NETO, 2005; CAMPOS VELHO, 2008). A schematic representation of these processes is shown in Figure 2.2.

Figure 2.2 - Schematic representation of the forward and inverse problems



In an inverse problem, the **parameters** are the numerical values to be estimated, and the **data** are the observations or measurements made on the system of interest.

Inverse problems are found in almost every field of science and mathematics, such as optics, radar, acoustics, communication theory, signal processing, medical imaging, computer vision, geophysics, oceanography, astronomy, remote sensing, natural language processing, machine learning, nondestructive testing, among others.

Definition 2.7. (Hadamard (1925)) A problem is called **well-posed** if it meets the following conditions:

- a) Existence: A solution of the problem exists;
- b) Uniqueness: The solution is unique;

c) Stability: The solution depends continuously on the data.

Usually, inverse problems violate all the three Hadamard's conditions, being therefore **ill-posed**. For handling this issue, some special methods are used, called regularization methods (KAPIO; SOMERSALO, 2006; CAMPOS VELHO, 2017) (e.g., Tikhonov Regularization, Lavrentiev Regularization, Entropic Regularization and Asymptotic Regularization). These methods transform an ill-posed problem into a well-posed problem in sense of Hadamard, closer to the original problem. With the use of smooth or regular solutions, these techniques can control the influence of the noise, and prevent over-fitting.

Some other methods for solving inverse problems are Variational method, Least squares method, filtering methods, Neural Networks, and Monte Carlo methods (TARANTOLA, 2005; SINHA et al., 2017).

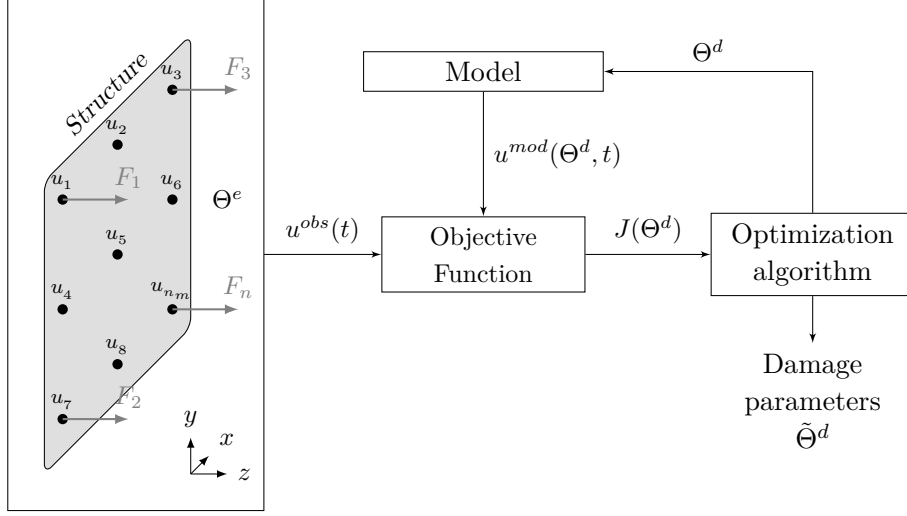
2.3 Vibration-based Damage Identification as an Inverse Problem

Dynamic processes modeling in mechanical vibration is characterized by knowing initial and boundary conditions, the geometry of the structure, material properties, and forcing terms. The system output response is the measurement of displacement, accelerations, strains, natural frequencies, and modes shapes. In vibration systems, modal parameters are a function of the physical properties of the structure (such as mass, damping, stiffness, and boundary conditions). Therefore, a change in the physical properties (caused by cracks, loosening of connections or another possible damage) will cause detectable variations in the modal properties.

The inverse problem of damage evaluation consists in detecting, localizing, and quantifying the damage severity. In this thesis, the damage severity is obtained by estimating the stiffness values from structural displacements and modal properties measurements.

The inverse problem can be formulated as an optimization problem. Figure 2.3 shows a graphical representation of the inverse solution for a generic structure. Parameter Θ^e represents the influence of environmental and operational conditions (e.g. temperature, and humidity), and Θ^d are the damage parameters (e.g. crack length, loss of stiffness, and loss of mass) (FRITZEN; KRAEMER, 2009). The displacements u^{mod} are obtained running the structural model with a stiffness vector k^d , and the measured displacements u^{obs} are acquired from the sensors in the vibration experiments.

Figure 2.3 - Vibration-Based Damage Identification as optimization problem



2.3.1 Damage parameters

The damage parameters Θ^d are defined as the percentage of loss of stiffness at each element to be monitored:

$$\Theta^d = \left(1 - \frac{\mathbf{k}^d}{\mathbf{k}^i}\right) \times 100\%, \quad (2.6)$$

where $\mathbf{k}^d = (k_1^d, k_2^d, \dots, k_n^d)$ and $\mathbf{k}^i = (k_1^i, k_2^i, \dots, k_n^i)$ are the estimated stiffness vector for the damaged system, and the stiffness vector of the undamaged system, respectively.

2.3.2 Objective function

The optimization problem can be stated as follows:

$$\begin{aligned} & \underset{\Theta^d}{\text{minimize}} \quad J(\Theta^d) = \sum_{m=1}^{n_m} e_m^2(\Theta^d), \\ & \text{subject to} \quad \Theta^{d^l} \leq \Theta^d \leq \Theta^{d^u}, \end{aligned} \quad (2.7)$$

where n_m is the number of measured displacements, and Θ^{d^l} and Θ^{d^u} are the lower and upper bounds for the damage parameters, respectively.

The residuals e_m are computed as:

$$e_m(\Theta^d) = \sum_{t=0}^{t_f} \left[u_m^{obs}(t) - u_m^{mod}(\Theta^d, t) \right], \quad (2.8)$$

in which t represents the time, and t_f is the final time.

2.3.3 Regularization by maximum entropy principle

The entropy principle was proposed by Jaynes (1957) on the basis of Shannon's axiomatic characterization of the amount of information. The maximum entropy regularization searches for global regularity, and yields the smoothest reconstructions which are consistent with the available data (MUNIZ et al., 2000).

The discrete entropy S of the parameter vector \mathbf{x} with non-negative components is defined by

$$S(\Theta^d) = \sum_{q=1}^n s_q \ln(s_q), \quad (2.9)$$

where $s_q = \frac{x_q}{\sum_{q=1}^n x_q}$.

The entropy function S gets its maximum when all x_q are the same, corresponding to a uniform distribution with a value of $S_{max} = \ln n$, while the lowest entropy level $S_{min} = 0$, is obtained when all elements x_q but one are set to zero (Dirac delta distribution).

Then, the optimization problem becomes an approach with regularization

$$J(\Theta^d) = \sum_{m=1}^{n_m} e_m^2(\Theta^d) + \alpha_\Omega \Omega(\Theta^d) \quad (2.10)$$

where α_Ω is the regularization parameter, and Ω is the regularization operator. In this case $\Omega \equiv S$

2.3.4 Morozov's discrepancy principle

Following the Morozov's discrepancy principle (MOROZOV, 1984), the regularization parameter δ_Ω is chosen such that

$$||\mathbf{u}^{mod}(\boldsymbol{\Theta}_{\alpha\Omega}^d) - \hat{\mathbf{u}}^{obs}||_2 = ||\sigma d||_2 \quad (2.11)$$

where $\hat{\mathbf{u}}^{obs}$ is the noisy data, σ is the known noise level.

2.4 Chapter Conclusions

In this chapter, the structural damage identification was presented as an inverse problem, solved as an optimization problem. The damage parameters will be used as the solution vector of the problem. The objective function to be minimized was defined as the square error between the measured data and the data obtained running the computational model of the structure. Some common models will be further discussed in the next chapter.

3 TRANSIENT RESPONSE ANALYSIS OF STRUCTURES AS DIRECT MODEL

All models are wrong, but some are useful.

GEORGE E. P. BOX

In the previous chapter, it was presented that the VDI will be solved as an optimization problem that uses data obtained through structural models.

In this chapter, the direct model used to obtain the transient response analysis of structures is presented. Initially, the transient response of a structure will be defined, and classified into the direct and the modal categories. The mathematical definition, and the numerical solution of both methods will be summarized. Then, the Finite Element Method will be introduced, showing how it is applied in computing the model of systems with springs, and structures with bars and beams.

3.1 Transient response analysis

In the transient analysis, a response to a time-varying input is computed. The excitation is defined in the time domain, and the responses, such as nodal displacements and accelerations, are functions of time.

There are two ways to compute the transient response analysis:

Direct transient response carries out a numerical integration of the complete set of equations of motion, and

Modal transient response uses some mode shapes, reducing the problem size. Uncouple the equations when no damping or only modal damping is used on the model.

Both methods will be detailed in the next sections.

3.1.1 Direct transient response

The dynamic response of motion of a continuous structure discretized in terms of Finite Elements under excitation is shown as:

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

where \mathbf{M} , \mathbf{C} and \mathbf{K} represent the mass, viscous damping and stiffness matrices, respectively. \mathbf{F} and \mathbf{u} are the external force and the displacement vectors, respectively. The initial conditions for the model are given by Equations 3.2 and 3.3.

$$\mathbf{u}(0) = \mathbf{u}_0, \quad (3.2)$$

$$\dot{\mathbf{u}}(0) = \dot{\mathbf{u}}_0. \quad (3.3)$$

The numerical solution for this model is obtained using the Newmark method (NEW-MARK, 1959) since no analytical solution exists for any arbitrary functions of \mathbf{M} , \mathbf{C} , \mathbf{K} and \mathbf{u} .

The equation of motion is approximated by a central finite difference representation. The velocity and the acceleration for the k -th time step are given by:

$$\dot{\mathbf{u}}_k = \frac{1}{2\Delta t} (\mathbf{u}_{k+1} - \mathbf{u}_{k-1}), \quad (3.4)$$

$$\ddot{\mathbf{u}}_k = \frac{1}{\Delta t^2} (\mathbf{u}_{k+1} - 2\mathbf{u}_k + \mathbf{u}_{k-1}). \quad (3.5)$$

respectively.

The equations of motion over three successive time instants can be reformulated as follow:

$$\begin{aligned} \frac{\mathbf{M}}{\Delta t^2} (\mathbf{u}_{k+1} - 2\mathbf{u}_k + \mathbf{u}_{k-1}) + \frac{\mathbf{C}}{2\Delta t} (\mathbf{u}_{k+1} - \mathbf{u}_{k-1}) + \frac{\mathbf{K}}{3} (\mathbf{u}_{k+1} + \mathbf{u}_k + \mathbf{u}_{k-1}) = \\ = \frac{1}{3} (\mathbf{F}_{k+1} + \mathbf{F}_k + \mathbf{F}_{k-1}). \end{aligned} \quad (3.6)$$

Regrouping the terms, the following equation is obtained:

$$\mathbf{A}_1 \mathbf{u}_{k+1} = \mathbf{A}_2 + \mathbf{A}_3 \mathbf{u}_k + \mathbf{A}_4 \mathbf{u}_{k-1}, \quad (3.7)$$

in which

$$\mathbf{A}_1 = \frac{\mathbf{M}}{\Delta t^2} + \frac{\mathbf{C}}{2\Delta t} + \frac{\mathbf{K}}{3}, \quad (3.8)$$

$$\mathbf{A}_2 = \frac{1}{3} (\mathbf{F}_{k+1} + \mathbf{F}_k + \mathbf{F}_{k-1}), \quad (3.9)$$

$$\mathbf{A}_3 = \frac{2\mathbf{M}}{\Delta t^2} - \frac{\mathbf{K}}{3}, \quad (3.10)$$

$$\mathbf{A}_4 = -\frac{\mathbf{M}}{\Delta t^2} + \frac{\mathbf{C}}{2\Delta t} - \frac{\mathbf{K}}{3}. \quad (3.11)$$

By decomposition of \mathbf{A}_1 and transfer to the right side of Equation (3.7), the transient solution displacement \mathbf{u}_{k+1} is obtained. It is recommended to keep Δt fixed, to avoid additional decomposition of \mathbf{A}_1 .

3.1.2 Modal transient response

For obtaining the modal transient response, the variables are transformed from physical coordinates \mathbf{u} to modal coordinates ξ :

$$\mathbf{u}(t) = \Phi \xi(t). \quad (3.12)$$

Substituting Equation (3.12) into Equation (3.13), is obtained:

$$\mathbf{M}\Phi\ddot{\xi}(t) + \mathbf{C}\Phi\dot{\xi}(t) + \mathbf{K}\Phi\xi(t) = \mathbf{F}(t). \quad (3.13)$$

Multiplying Equation (3.13) by Φ^\top , is obtained:

$$\Phi^\top \mathbf{M} \Phi \ddot{\xi}(t) + \Phi^\top \mathbf{C} \Phi \dot{\xi}(t) + \Phi^\top \mathbf{K} \Phi \xi(t) = \Phi^\top \mathbf{F}(t). \quad (3.14)$$

where $\Phi^\top \mathbf{M} \Phi$ is the modal mass matrix, $\Phi^\top \mathbf{C} \Phi$ is the modal damping matrix, $\Phi^\top \mathbf{K} \Phi$ is the modal stiffness matrix, and $\Phi^\top \mathbf{F}(t)$ is the modal force vector.

The coupled equations are then solved by the same method used in the Section 3.1.1, obtaining:

$$\mathbf{A}_1 \xi_{k+1} = \mathbf{A}_2 + \mathbf{A}_3 \xi_k + \mathbf{A}_4 \xi_{k-1}, \quad (3.15)$$

in which

$$\mathbf{A}_1 = \Phi^\top \left[\frac{\mathbf{M}}{\Delta t^2} + \frac{\mathbf{C}}{2\Delta t} + \frac{\mathbf{K}}{3} \right] \Phi, \quad (3.16)$$

$$\mathbf{A}_2 = \frac{1}{3} \Phi^\top (\mathbf{F}_{k+1} + \mathbf{F}_k + \mathbf{F}_{k-1}), \quad (3.17)$$

$$\mathbf{A}_3 = \Phi^\top \left[\frac{2\mathbf{M}}{\Delta t^2} - \frac{\mathbf{K}}{3} \right] \Phi, \quad (3.18)$$

$$\mathbf{A}_4 = \Phi^\top \left[-\frac{\mathbf{M}}{\Delta t^2} + \frac{\mathbf{C}}{2\Delta t} - \frac{\mathbf{K}}{3} \right] \Phi. \quad (3.19)$$

The direct integration of the equations with modal variables is not as costly as with physical variables, since the number of modes used in a solution is typically much

less than the number of physical variables (NASTRAN, 2004).

The uncoupled equations of motion for each mode present the following form:

$$m_i \ddot{\xi}_i(t) + c_i \dot{\xi}_i(t) + k_i \xi_i(t) = f_i(t), \quad (3.20)$$

where m_i is the i -th modal mass, c_i is the i -th modal damping, and k_i is the i -th modal stiffness

The dynamic equation can be expressed as follows:

$$\ddot{\xi}_i(t) + 2\zeta_i \omega_i \dot{\xi}_i(t) + \omega_i^2 \xi_i(t) = f_i(t)/m_i, \quad (3.21)$$

where ξ_i is the i -th modal coordinate, $\omega_i^2 = k_i/m_i$ is the i -th modal frequency, $\zeta_i = c_i/2m_i\omega_i$ is the i -th modal damping ratio, and N_i is the i -th modal force.

For the recovery of the physical response, the summation of the modal response is done.

3.1.3 Modal Versus Direct Transient Response

For selecting modal versus transient response analysis, there are some guidelines to be used (NASTRAN, 2004), which are summarized in Table 3.1.

Table 3.1 - Modal versus direct transient response (NASTRAN, 2004)

	Modal	Direct
Small Model		✓
Large Model	✓	
Few Time Steps		✓
Many Time Steps	✓	
High Frequency Excitation		✓
Normal Damping		✓
Higher Accuracy		✓
Nonlinearities		✓
Initial conditions	✓	✓

Large models are commonly solved more efficiently when using modal transient response. In this type of analysis, the problem is typically reduced in modal space. In addition, a modal transient response analysis is recommended when no damping or only modal damping is used. In these cases, the major computational effort is to calculate the modes. If the model is large with a large number of modes, the

modal transient response analysis can be as intensive as the direct transient response analysis, computationally speaking. In simulations with a fewer number of time steps, the direct transient response should be the most efficient because the equations are solved without computing the modes. Finally, the direct method is more accurate than the modal method, since it is not concerned with mode truncation. The direct transient response can only be used in systems with initial conditions ([NASTRAN, 2004](#)).

3.2 Finite Element Method

The Finite Element Method (FEM) or Finite Element Analysis (FEA) is a numerical method for investigating the behaviour of structures. Now, FEM is used as a general method for numerical integration of partial differential equations. The principal concepts in FEM ([LOGAN, 2011](#); [FELIPPA, 2014](#)) will be introduced below.

FEM breaks the structures into smaller simpler pieces called elements. The finite elements are connected to each other at nodes. The finite element model is the assembly of all elements and nodes in the structure.

FEM consists of three main steps ([MOAVENI, 2008](#)), shown in [Figure 3.1](#), and detailed as follow:

Preprocessing

- (i) Create the solution domain, and discretize into finite elements, creating nodes and elements;
- (ii) Represent the physical behaviour using a shape function;
- (iii) Develop equations for the elements;
- (iv) Assemble the element equations to create the global matrices;
- (v) Apply boundary conditions, initial conditions, and loading.

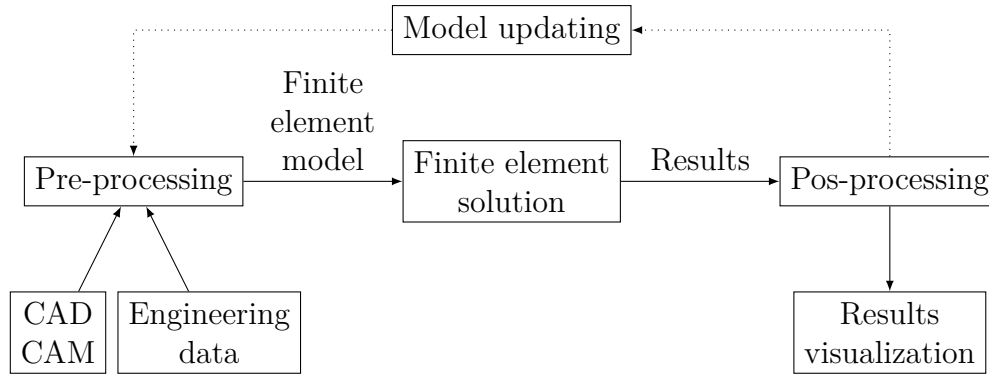
Finite Element Solution

- (vi) Solve the system of equations to obtain the results for each node.

Postprocessing

- (vii) Obtain other information using the results from the analysis.

Figure 3.1 - Finite Element Method



SOURCE: Adapted from Moaveni (2008)

An **element** is the basic building block, and it is the mathematical relationship that defines how the degrees of freedom of one node relate to the next node.

Elements can have different shapes, with intrinsic dimensionality of one (1D), two (2D), or three dimensions (3D). There are also special elements with zero dimensions (0D). Some basic types of finite elements are shown in Table 3.2.

The intrinsic dimensionality can be expanded as well. For example, a 1D element, such as a bar, can be used to build a model in 2D or 3D.

Table 3.2 - Types of elements

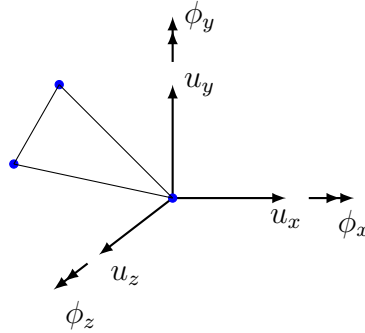
Dimension	Element type	Use	Example
0D	Point masses	Concentrated mass or weights	Brackets, pins or screws
	Lumped springs		
1D	Bars	Long and slender structural members	Communication towers
	Beams		
2D	Plates	Thin structural members	Aircraft fuselage skin
3D	Solids	Thick components	Piston head

Each element has points called **nodal points** or **nodes**. The nodes are used for defining the element geometry, and for placing of **degrees of freedom** (DOF). The geometry of an element is defined by the placement of the geometric nodes. DOFs are the values of a primary field variable, or independent directions, at connection

nodes.

In the real world, each point can move in six DOF, as shown in [Figure 3.2](#). The displacement vector $\mathbf{u} = [u_x \ \Phi_x \ u_y \ \Phi_y \ u_z \ \Phi_z]^\top$, where u_x , u_y , and u_z are translations, and Φ_x , Φ_y , and Φ_z are rotations. In the FEM, each element type defines the limitations in their motion.

Figure 3.2 - Degrees of freedom



3.2.1 Stiffness matrix

The elemental stiffness matrix $\mathbf{K}^{(e)}$ is a matrix such that:

$$\mathbf{f}^{(e)} = \mathbf{K}^{(e)} \mathbf{u}^{(e)} \quad (3.22)$$

in which $\mathbf{f}^{(e)}$ are the nodal forces and $\mathbf{u}^{(e)}$ are the nodal displacements of a single element. A simple element is the spring, which is described in the next section.

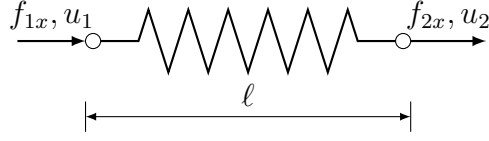
3.3 Spring element

The spring elements, like the one shown in the [Figure 3.3](#), are commonly used to model connectors and interfaces. In a single spring element, there are two nodes, with one single DOF per node, resulting in two DOFs. The element has two nodal displacements (u_1, u_2) , and two nodal forces (f_{1x}, f_{2x}) .

Since there are two degrees of freedom associated with the spring element, a linear displacement function u along the x -axis is assumed:

$$u = a_1 + a_2 x, \quad (3.23)$$

Figure 3.3 - Spring element



where a_1 and a_2 are constants. Evaluating u at each node, and solving for a_1 and a_2 :

$$u(0) = u_1 = a_1, \quad (3.24)$$

$$u(\ell) = u_2 = a_2\ell + u_1, \quad (3.25)$$

and

$$a_2 = \frac{u_2 - u_1}{\ell}. \quad (3.26)$$

Therefore, substituting [Equations \(3.24\) and \(3.26\)](#) in [Equation \(3.23\)](#)

$$u = \left(\frac{u_2 - u_1}{\ell} \right) x + u_1, \quad (3.27)$$

$$u = \left[1 - \frac{x}{\ell} \quad \frac{x}{\ell} \right] \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}, \quad (3.28)$$

$$u = \begin{bmatrix} N_1 & N_2 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}, \quad (3.29)$$

where $N_1 = 1 - \frac{x}{\ell}$ and $N_2 = \frac{x}{\ell}$ are called shape functions.

The tensile forces T produce a deformation δ of the spring, represented by:

$$\delta = u(L) - u(0) = u_2 - u_1. \quad (3.30)$$

The force/deformation relationship is expressed as:

$$T = k\delta = k(u_2 - u_1). \quad (3.31)$$

Using the sign convention for nodal forces and equilibrium:

$$f_{1x} = -T, \quad (3.32)$$

$$f_{2x} = T, \quad (3.33)$$

then

$$f_{1x} = k(u_1 - u_2), \quad (3.34)$$

$$f_{2x} = k(u_2 - u_1), \quad (3.35)$$

In a matrix form,

$$\begin{Bmatrix} f_{1x} \\ f_{2x} \end{Bmatrix} = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \quad (3.36)$$

The local stiffness matrix for the element e is:

$$\mathbf{K}^{(e)} = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix}. \quad (3.37)$$

For the entire FE model, the stiffness matrices $\mathbf{K}^{(e)}$ and the force vectors $\mathbf{f}^{(e)}$ of each element are assembled to form the global stiffness matrix \mathbf{K} and \mathbf{f} :

$$\mathbf{K} = \sum_{e=1}^n \mathbf{K}^{(e)}, \quad (3.38)$$

$$\mathbf{f} = \sum_{e=1}^n \mathbf{f}^{(e)}. \quad (3.39)$$

Then, the global stiffness matrix \mathbf{K} relates global-coordinate (x, y, z) nodal displacements to global load vector f , as show in [Equation \(3.40\)](#):

$$\mathbf{f} = \mathbf{K}\mathbf{u}. \quad (3.40)$$

The finite element model can be constrained by means of the boundary conditions. In practice, those rows and columns that correspond to the constrained DOF in the global matrix are removed.

3.4 Trusses

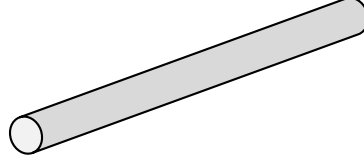
A truss is a structure composed of slender members connected at their ends by means of bolts, rivets, pins or welding. Metal bars or wooden struts are members that can be found in trusses (MOAVENI, 2008; HIBBELER, 2010).

In particular, a **plane truss** is a truss whose members lie in a single plane. The forces act also in this single plane. In trusses, two-force members are considered. In two-force members force is applied to only two points, and the internal forces act in equal and opposite directions along the members.

3.4.1 Bar element

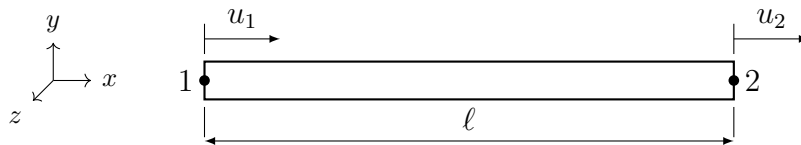
Bars are structural components characterized by having one preferred dimension, and by resisting an internal axial force along its longitudinal dimension. The **longitudinal dimension** or **axial dimension** is much larger than the other two dimensions, known as transverse dimensions, as shown in the Figure 3.4.

Figure 3.4 - Bar element



The model of a bar element is shown in Figure 3.5. It is considered linear-elastic, and constant cross-sectional area (prismatic). The bar element has a constant cross-section area A , length ℓ , and a modulus of elasticity E (also known as Young's modulus). The nodal degrees of freedom are the local axial displacements u_1 and u_2 at the ends of the bar. T is the tensile force directed along the axis at nodes 1 and 2, and x is the local coordinate system directed along the length of the bar.

Figure 3.5 - Representation of a bar element



Assuming that the material of the bar is linearly elastic and obeys the Hooke's law, and that the strain/displacement are small, their relationship can be expressed as:

$$\sigma_x = E\epsilon_x \quad (3.41)$$

$$\epsilon_x = \frac{du}{dx} \quad (3.42)$$

where σ_x is the axial stress, and ϵ_x is the axial strain.

From force equilibrium, $A\sigma_x = T = \text{constant}$.

Combining the equations above, and differentiating with respect to x , the following differential equation governing the linear-static bar behaviour can be obtained:

$$\frac{d}{dx} \left(AE \frac{du}{dx} \right) = 0 \quad (3.43)$$

The bar element stiffness matrix can be derived considering the following assumptions:

- a) The bar can not sustain the shear force: $f_{1y} = f_{2y} = 0$;
- b) Any effect of transverse displacement is ignored;
- c) Hookes law applies; stress is related to strain: $\sigma_x = E\epsilon_x$.

There are two degrees of freedom associated with the bar element. Thus, a linear displacement function u is assumed: $u = a_1 + a_2x$.

Applying the boundary conditions and solving for the unknown coefficients, is obtained that:

$$u = \frac{u_2 - u_1}{\ell}x + u_1 \quad (3.44)$$

Or, in a matrix representation:

$$u = \begin{bmatrix} 1 - \frac{x}{\ell} & \frac{x}{\ell} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \quad (3.45)$$

$$u = \begin{bmatrix} N_1 & N_2 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \quad (3.46)$$

The stress-displacement relationship is given by:

$$\epsilon_x = \frac{du}{dx} = \frac{u_2 - u_1}{\ell} \quad (3.47)$$

The stiffness matrix of the bar element can be derived as follows:

$$T = A\sigma_x = AE \frac{u_2 - u_1}{\ell} \quad (3.48)$$

The nodal force sign convention defined is:

$$f_{1x} = -T \quad (3.49)$$

$$f_{2x} = T. \quad (3.50)$$

Therefore,

$$f_{1x} = AE \frac{u_1 - u_2}{\ell} \quad (3.51)$$

$$f_{2x} = AE \frac{u_2 - u_1}{\ell}. \quad (3.52)$$

In matrix form:

$$\begin{Bmatrix} f_{1x} \\ f_{2x} \end{Bmatrix} = \frac{AE}{\ell} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}. \quad (3.53)$$

The local stiffness matrix for the element e is defined as:

$$\mathbf{K}^{(e)} = \frac{AE}{\ell} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \quad (3.54)$$

The term AE/ℓ is analogous to the spring constant k for a spring element.

The global stiffness matrix \mathbf{K} and the global force vector \mathbf{f} are assembled using the nodal force equilibrium equations, and force/deformation and compatibility equations:

$$\mathbf{K} = \sum_{e=1}^n \mathbf{K}^{(e)} \quad (3.55)$$

$$\mathbf{f} = \sum_{e=1}^n \mathbf{f}^{(e)}. \quad (3.56)$$

The elemental mass matrix is computed as:

$$\mathbf{M}^{(e)} = \frac{\rho A \ell}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad (3.57)$$

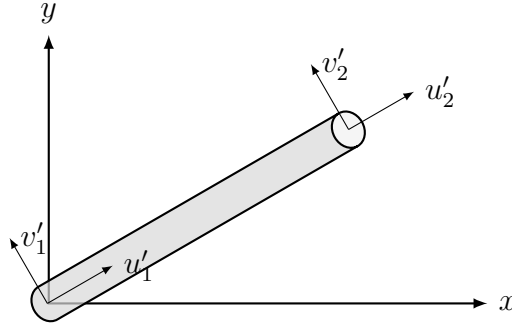
where ρ is the density of the member.

3.4.1.1 Transformation into global coordinates

A planar/spatial truss structure can be modeled using bar elements with two or three degrees of freedom at each node.

As shown in Figure 3.6, u_1 and u_2 are the displacement of nodes 1 and 2 in local coordinate system. In global coordinate system, each node has two degrees of freedom. The nodal displacement at nodes 1 and 2 are denoted as u'_1 , v'_1 and u'_2 , v'_2 , respectively along x and y directions.

Figure 3.6 - Bar element in global coordinates



The new coordinates u'_1 , v'_1 , u'_2 and v'_2 are represented in Figure 3.7, and calculated as:

$$u'_1 = \cos u_1 + \sin v_1, \quad (3.58)$$

$$u'_2 = \cos u_2 + \sin v_2, \quad (3.59)$$

$$v'_1 = -\sin u_1 + \cos v_1, \quad (3.60)$$

$$v'_2 = -\sin u_2 + \cos v_2. \quad (3.61)$$

In matrix form, can be expressed as:

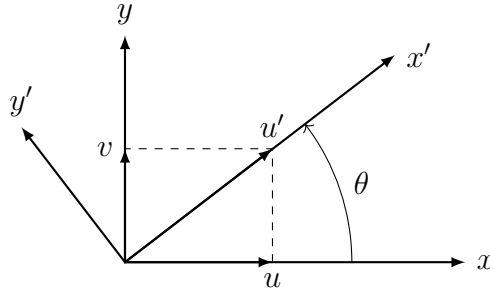
$$\begin{bmatrix} u'_1 \\ v'_1 \\ u'_2 \\ v'_2 \end{bmatrix} = \begin{bmatrix} c & -s & 0 & 0 \\ s & c & 0 & 0 \\ 0 & 0 & c & -s \\ 0 & 0 & s & c \end{bmatrix} \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \end{bmatrix}, \quad (3.62)$$

or

$$\mathbf{u}' = \mathbf{T}\mathbf{u}, \quad (3.63)$$

in which $c = \cos \theta$, and $s = \sin \theta$, being θ the angle between the element and the global x -axis, as shown in Figure 3.7. \mathbf{T} is the 4×4 matrix called the displacement transformation matrix.

Figure 3.7 - Global coordinates



The local stiffness matrix is modified to include the additional degrees-of-freedom.

$$\mathbf{K}^{(e)} = \frac{AE}{\ell} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.64)$$

The associated matrices can be transformed into global coordinates. using the following relations:

$$\mathbf{K}'^{(e)} = \mathbf{T}^T \mathbf{K}^{(e)} \mathbf{T}, \quad (3.65)$$

$$\mathbf{M}'^{(e)} = \mathbf{T}^T \mathbf{M}^{(e)} \mathbf{T}, \quad (3.66)$$

The nodal forces are then calculated as:

$$\begin{Bmatrix} f'_{1x} \\ f'_{2x} \end{Bmatrix} = \frac{AE}{\ell} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u'_1 \\ u'_2 \end{Bmatrix}. \quad (3.67)$$

The elemental stiffness matrix for a bar that is arbitrarily oriented in the $x - y$ plane, is expressed as follows:

$$\mathbf{K}^{(e)} = \frac{AE}{\ell} \begin{bmatrix} c^2 & cs & c^2 & cs \\ cs & s^2 & cs & s^2 \\ c^2 & cs & c^2 & cs \\ cs & s^2 & cs & s^2 \end{bmatrix}. \quad (3.68)$$

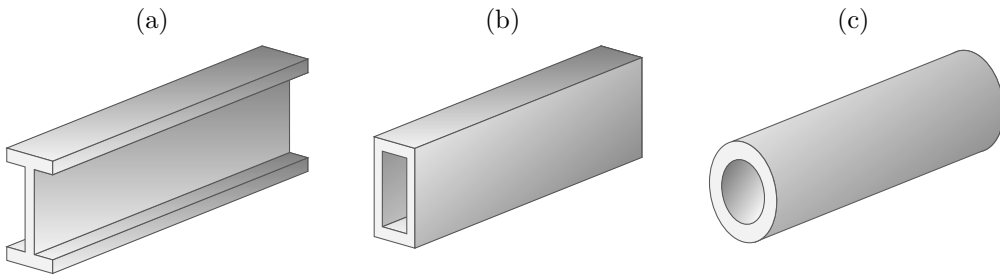
The elemental mass in the global reference system is defined as:

$$\mathbf{M}^{(e)} = \frac{\rho A \ell}{6} \begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 \end{bmatrix}. \quad (3.69)$$

3.4.2 Beam element

Beams are a simple but common type of structural component. They are used mostly in Civil and Mechanical Engineering. These structural members have the main function of supporting transverse loading and carry it to the supports. Their shape is bar-like, being one dimension larger than the others, and they deform only in the directions perpendicular to the x -axis. Figure 3.8 shows examples of different types of beams.

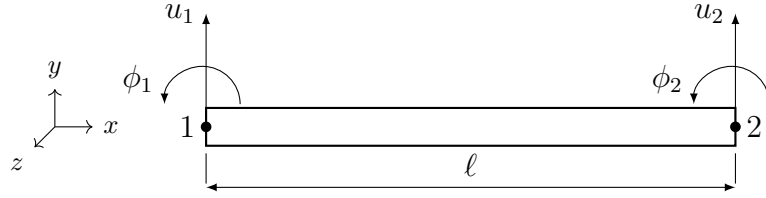
Figure 3.8 - Examples of different types of beams: (a) American wide flange beam; (b) Rectangular hollow section; (c) Circular hollow section (pipe)



The experiments will be performed on a plane beam. This type of beam resists primarily transverse loading on a preferred longitudinal plane.

A planar beam finite element is obtained by subdividing beam members longitudinally. This type of element have two nodes (1 and 2), with two degrees of freedom at each node: deflection in the y axis, u , and rotation in the $x - y$ plane $\Phi = du(x)/dx$. At the left node, the degrees of freedom are called u_1 , Φ_1 , and at the right node, are called u_2 , Φ_2 . At an arbitrary location x , the vertical displacement is called $u(x)$ and the rotation is called $\Phi(x)$.

Figure 3.9 - Beam element



There are four shape functions, one of each DOF for a beam element. In order to obtain them, the transverse displacement function u is assumed:

$$u = a_1x^3 + a_2x^2 + a_3x + a_4, \quad (3.70)$$

where a_1 , a_2 , a_3 and a_4 are integration constants.

There are four coefficients in the displacement function, corresponding to the total number of DOF associated to the beam element.

The boundary conditions are:

$$u(x = 0) = u_1, \quad (3.71)$$

$$u(x = \ell) = u_2, \quad (3.72)$$

$$\left. \frac{du}{dx} \right|_{x=0} = \Phi_1, \quad (3.73)$$

$$\left. \frac{du}{dx} \right|_{x=\ell} = \Phi_2. \quad (3.74)$$

Applying the boundary conditions and solving for the unknown coefficients:

$$u(x=0) = u_1 = a_4, \quad (3.75)$$

$$u(x=\ell) = u_2 = a_1\ell^3 + a_2\ell^2 + a_3\ell + a_4, \quad (3.76)$$

$$\left. \frac{du}{dx} \right|_{x=0} = \Phi_1 = a_3, \quad (3.77)$$

$$\left. \frac{du}{dx} \right|_{x=\ell} = \Phi_2 = 3a_1\ell^2 + 2a_2\ell + a_3. \quad (3.78)$$

Then, solving the equations for a_1 , a_2 , a_3 and a_4 , is obtained:

$$u = \underbrace{\left[\frac{2}{\ell^3}(u_1 - u_2) + \frac{1}{\ell^2}(\Phi_1 + \Phi_2) \right]}_{a_1} x^3 + \underbrace{\left[-\frac{3}{\ell^2}(u_1 - u_2) - \frac{1}{\ell}(2\Phi_1 + \Phi_2) \right]}_{a_2} x^2 + \underbrace{\Phi_1}_{a_3} x + \underbrace{u_1}_{a_4}. \quad (3.79)$$

In matrix form, this equation is expressed as:

$$u = \mathbf{N}\mathbf{u}, \quad (3.80)$$

in which $\mathbf{N} = \begin{bmatrix} N_1 & N_2 & N_3 & N_4 \end{bmatrix}$ and $\mathbf{u} = \begin{bmatrix} u_1 & \Phi_1 & u_2 & \Phi_2 \end{bmatrix}^\top$, and

$$N_1 = \frac{1}{\ell^3} (2x^3 - 3x^2\ell + \ell^3), \quad (3.81)$$

$$N_2 = \frac{1}{\ell^3} (x^3\ell - 2x^2\ell^2 + x\ell^3), \quad (3.82)$$

$$N_3 = \frac{1}{\ell^3} (-2x^3 + 3x^2\ell), \quad (3.83)$$

$$N_4 = \frac{1}{\ell^3} (x^3\ell - x^2\ell^2). \quad (3.84)$$

In a beam, forces and displacements are positive in the positive y direction, while moments and rotations are positive in the counter-clockwise direction. The nodal end forces vector \mathbf{p} is found as:

$$\mathbf{p} = \begin{bmatrix} f_{1y} & m_1 & f_{2y} & m_2 \end{bmatrix}^\top, \quad (3.85)$$

where, m_1 and m_2 are the moments at the left and the right nodes, respectively, and f_{1y} and f_{2y} are the vertical forces at the left and the right nodes, respectively.

$$f_{1y} = EI \frac{d^3 u}{dx^3} \Big|_{x=0} = \frac{EI}{\ell^3} (12u_1 + 6\ell\Phi_1 - 12u_2 + 6\ell\Phi_2) , \quad (3.86)$$

$$f_{2y} = -EI \frac{d^3 u}{dx^3} \Big|_{x=\ell} = \frac{EI}{\ell^3} (-12u_1 - 6\ell\Phi_1 + 12u_2 - 6\ell\Phi_2) , \quad (3.87)$$

$$m_1 = -EI \frac{d^2 u}{dx^2} \Big|_{x=0} = \frac{EI}{\ell^3} (6\ell u_1 + 4\ell^2\Phi_1 - 6\ell u_2 + 2\ell^2\Phi_2) , \quad (3.88)$$

$$m_2 = EI \frac{d^2 u}{dx^2} \Big|_{x=\ell} = \frac{EI}{\ell^3} (6\ell u_1 + 2\ell^2\Phi_1 - 6\ell u_2 + 4\ell^2\Phi_2) . \quad (3.89)$$

In matrix form, these equations are expressed as:

$$\begin{Bmatrix} f_{1y} \\ m_1 \\ f_{2y} \\ m_2 \end{Bmatrix} = \frac{EI}{\ell^3} \begin{bmatrix} 12 & 6\ell & -12 & 6\ell \\ 6\ell & 4\ell^2 & -6\ell & 2\ell^2 \\ -12 & -6\ell & 12 & -6\ell \\ 6\ell & 2\ell^2 & -6\ell & 4\ell^2 \end{bmatrix} \begin{Bmatrix} u_1 \\ \Phi_1 \\ u_2 \\ \Phi_2 \end{Bmatrix} , \quad (3.90)$$

in which where I is the area moment of inertia. Some area moment of inertia are shown in [Table 3.3](#).

Table 3.3 - Some common sections and their area moment of inertia

Description	Area moment of inertia
Filled circular area of radius r	$I = \pi/4r^4$
Anulus of inner radius r_1 and outer radius r_2	$I = I = \frac{\pi}{4} (r_2^4 - r_1^4)$
Filled rectangular	$I = \frac{Ah^2}{12}$

Then, the stiffness matrix $\mathbf{K}^{(e)}$ of a beam element is defined as:

$$\mathbf{K}^{(e)} = \frac{EI}{\ell^3} \begin{bmatrix} 12 & 6\ell & -12 & 6\ell \\ 6\ell & 4\ell^2 & -6\ell & 2\ell^2 \\ -12 & -6\ell & 12 & -6\ell \\ 6\ell & 2\ell^2 & -6\ell & 4\ell^2 \end{bmatrix} . \quad (3.91)$$

The mass matrix $\mathbf{M}^{(e)}$ of a beam element is defined as:

$$\mathbf{M}^{(e)} = \frac{\rho A \ell}{420} \begin{bmatrix} 156 & 22l & 54 & -13l \\ 22l & 4l^2 & 13l & -3l^2 \\ 54 & 13l & 156 & -22l \\ -13l & -3l^2 & -22l & 4l^2 \end{bmatrix}. \quad (3.92)$$

Both $\mathbf{K}^{(e)}$ and $\mathbf{M}^{(e)}$ are symmetric and positive semi-definite matrices.

3.5 Damping matrix

In this thesis, the damping matrix is evaluated according to the Rayleigh damping, as a linear combination of the mass and stiffness matrices,

$$\mathbf{C} = \alpha_M \mathbf{M} + \beta_K \mathbf{K}, \quad (3.93)$$

where α_M and β_K are constants with units s^{-1} and s , respectively.

3.6 Chapter conclusions

In this chapter, the main differences between the direct and the modal transient response of a structure were shown. Then, the FEM was presented, and the mathematical model of springs, bars, and beams were summarized.

The next chapter will present the optimization algorithms used to solve the identification problem.

4 HYBRID ALGORITHMS FOR SOLVING THE DAMAGE IDENTIFICATION PROBLEM

Veni, vidi, vici

GAIUS JULIUS CAESAR

In the last chapter, the FEM method was briefly presented. Also, were shown some 0D and 1D elements that are used for modeling structures. The way for calculating their stiffness was summarized.

This chapter will present the hybrid algorithms that will be used for solving the optimization problem. Initially, it will be presented a classification of the optimization algorithms, as well as the concepts of metaheuristic and hybrid algorithms, and the classification of these methods. Then, all the algorithms used in the solution of the problem will be introduced: Multi-Particle Collision Algorithm, q -gradient, Opposition-based Optimization, and Hooke-Jeeve direct search method.

4.1 Optimization algorithms

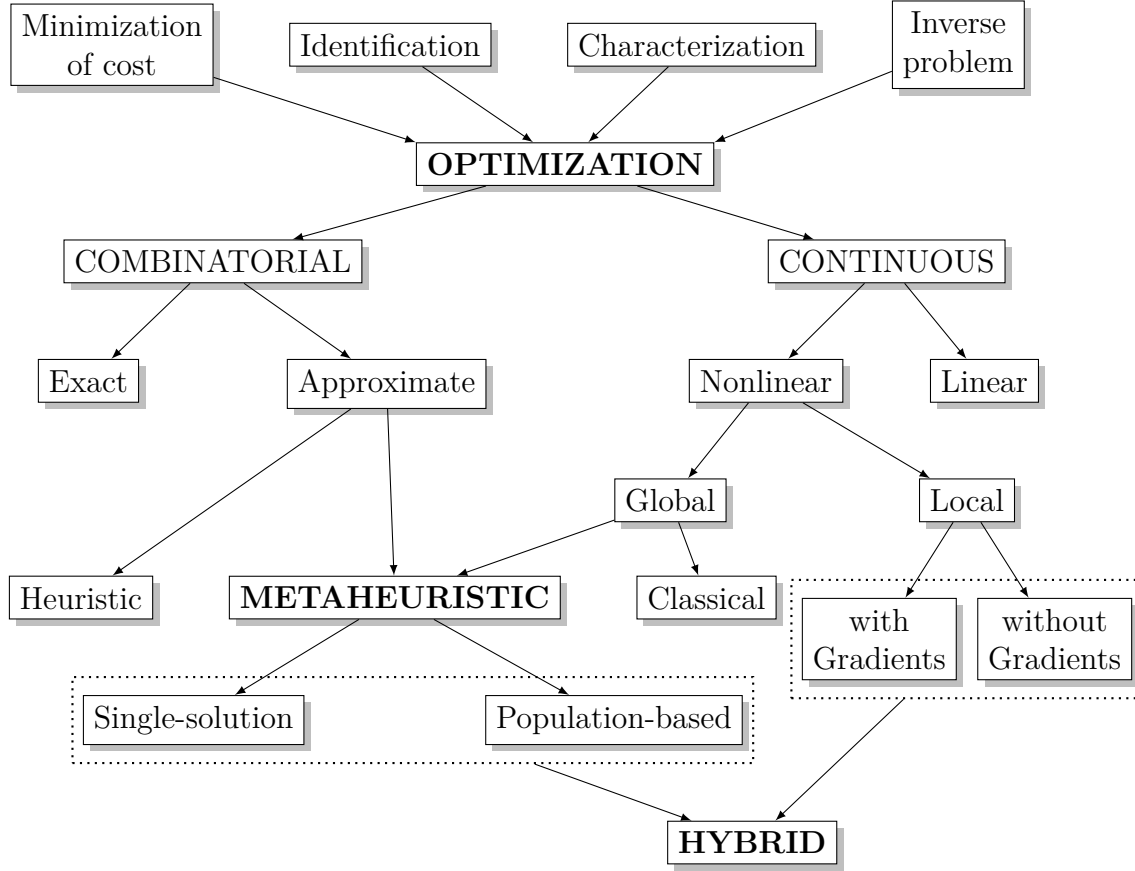
The choice of an appropriate optimization algorithm depends on the optimization problem.

Figure 4.1 shows a classification for the mono-objective optimization algorithms (SIARRY, 2016). Following this classification, optimization algorithms can be distinguished in two main branches: **combinatorial** or **continuous**.

For combinatorial optimization, there exist exact and approximate methods. For difficult optimization problems, there are specialized heuristics dedicated to specific problems, and metaheuristics.

For continuous optimization, there exists a **linear** approach (with linear programming), distinguished from the **nonlinear** case. If there is a low number of local minima, a **local** method should be used, which may or may not use the gradient for searching the objective function. If a high number of local minima exists, a **global** method should be used, in which are included the traditional methods and the metaheuristics.

Figure 4.1 - Classification of mono-objective optimization algorithms



SOURCE: Adapted from (SIARRY, 2016)

4.1.1 Metaheuristic algorithms

Stochastic optimization has become an important tool to solve multi-modal optimization problems. Sometimes it is hard to compute the gradient of the objective function, and even and even there might not exist a derivative of such a function. Most of the stochastic methods do not need the gradient information, or other internal information of the process/system, to be applied. Stochastic methods use random processes to generate candidate solutions, and facilitate the **exploration** (global search) in the search space, at the same time that **exploitation** (local search) is done. The exploration gives to the algorithm the ability of visiting various regions of the search space seeking promising good solutions. The exploitation performs a search in a smaller area in order to improve a good solution found. In other words, these algorithms have the capacity of visiting almost the entire search space by gen-

erating new randomly candidate solutions, while an intense search is done in the neighborhood of promising candidate solutions.

With recent advances in the computer science area, many techniques have been developed in the sub-area of stochastic optimization methods. Those algorithms are called to improve the exploration of the search space, making it more efficient, expectedly converging more quickly to the global optimum.

Wolpert and Macready (1997), in their *No Free Lunch* Theorem, established that “for any algorithm, any elevated performance over one class of problems is offset by performance over another class”.

Metaheuristic algorithms are powerful tools within the approximated methods that can solve hard optimization problems which could not be solved by deterministic optimization algorithms in a reasonable time (YANG, 2010; LIN et al., 2012).

The term metaheuristic was first used by Glover (1986), and comes from the composition of two Greek words: meta and *heuriskein*. A good definition of metaheuristic is given by Osman and Laporte (1996):

“A metaheuristic is formally defined as an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space, learning strategies are used to structure information in order to find efficiently near-optimal solutions.”

The two main features of the metaheuristic algorithms are the **intensification**, also called exploitation, and the **diversification**, also called exploration. The exploration phase is responsible for efficiently exploring the search space, while the exploitation phase searches within the current best solutions neighborhood, and selects the best solutions (BLUM; ROLI, 2003; GANDOMI et al., 2013; YANG, 2014).

Blum and Roli (2003) summarize some fundamental properties which characterize metaheuristics:

- “Metaheuristics are strategies that ‘guide’ the search process.”
- “The goal is to efficiently explore the search space in order to find (near-) optimal solutions.”

- “Techniques which constitute metaheuristic algorithms range from simple local search procedures to complex learning processes.”
- “Metaheuristic algorithms are approximate and usually non-deterministic.”
- “They may incorporate mechanisms to avoid getting trapped in confined areas of the search space.”
- “The basic concepts of metaheuristics permit an abstract level description.”
- “Metaheuristics are not problem-specific.”
- “Metaheuristics may make use of domain-specific knowledge in the form of heuristics that are controlled by the upper level strategy.”
- “Today’s more advanced metaheuristics use search experience (embodied in some form of memory) to guide the search.”

A huge number of metaheuristics can be found in the literature (DU; SWAMY, 2016; JR. et al., 2013; SORENSEN et al., 2017). They can be classified in different ways. Among these methods, there are two main categories:

Single-solution based does the search by using only one solution at a time; and

Population-based uses a set of solutions for exploring the search space.

Other classification attends their inspiration:

Evolution Evolutionary Programming (EP) (FOGEL, 1999), Evolution Strategies (ES) (BEYER, 2013), Genetic Algorithms (GA) (HOLLAND, 1992), Genetic Programming (GP) (LANGDON; GUSTAFSON, 2010), Differential Evolution (DE) (DAS; SUGANTHAN, 2011), Cultural Algorithms (CA) (REYNOLDS, 1994), and Biogeography-Based Optimization (BBO) (SIMON, 2008);

Swarm intelligence Particle Swarm Optimization (PSO) (EBERHART; KENNEDY, 1995; KENNEDY, 2010), Ant Colony Optimization (ACO) (DORIGO; BIRATTARI, 2010; DORIGO et al., 2006), Artificial Bee Optimization (ABC) (KARABOGA; BASTURK, 2007), Bacterial foraging optimization (BFO) (DAS et al., 2009), Intelligent Water Drops (IWS) (SHAH-HOSSEINI,

2009), and Artificial Immune Systems (AIS) (CASTRO; TIMMIS, 2002);

Human-based Memetic Algorithms (MA) (MOSCATO, 1989), and Harmony search (HS) (GEEM et al., 2001).

Sciences-based Simulated Annealing (SA) (KIRKPATRICK et al., 1983), Particle Collision Algorithm (PCA) (SACCO; OLIVEIRA, 2005) and Multi-Particle Collision Algorithm (MPCA) (LUZ et al., 2008).

Not inspired in nature Local Search (LS), Greedy Heuristic (GH), Scatter Search (SS), Tabu Search (TS), and Iterated Local Search (ILS).

4.1.2 Hybrid algorithms

Hybrid metaheuristics are methods that combine a metaheuristic with other optimization approaches, such as exact methods (JOURDAN et al., 2009), algorithms from mathematical programming, constraint programming, machine learning, or even artificial intelligence (RAIDL, 2006; RAIDL et al., 2010).

This cooperation can be done in an easy way, where the local method refines the solution obtained from the metaheuristic. Also, a more complex way of hybrid algorithms can be found, in which the single methods are intermingled.

Hybridizing different algorithmic concepts allows obtaining a better performance, exploiting and combining the advantages of single strategies (BLUM et al., 2010).

Hybrid algorithms can be classified into two main types (TALBI, 2002; JOURDAN et al., 2009): **low-level**, with a functional composition, where a given function of a metaheuristic is replaced by another method or **high-level**: there is no composition of the different algorithms, retaining their own identities; and working as a **relay**, where one algorithm takes as its inputs the output of the previous algorithms, working in series, like a pipeline, or a **teamwork**, using cooperative optimization models.

Another classification was presented by Talbi (2002), where the hybrid algorithm was divided in **homogeneous**, when all the combined algorithms use the same metaheuristic, or **heterogeneous**, in which different metaheuristics are used; in **global**, with all the algorithms searching in the whole search space, or **partial**, when the problem is divided into sub-spaces, and the algorithms perform the search in its search space; and in **general**, when all the algorithms solve the same optimization

problem, or **specialist**, when each algorithm solve a different problem.

A new classification was made by [Ting et al. \(2015\)](#), separating the hybrid algorithms into two groups according to their taxonomy:

- Collaborative hybrids combine two or more algorithms that could work in three ways:
 - Multi-stage, combining two stages: a global search followed by a local search;
 - Sequential, running both algorithms alternatively until a stopping criterion is met; or,
 - Parallel, where the algorithms run simultaneously over the same population.
- Integrative hybrids, where a master algorithm has other algorithm embedded working in two possible ways:
 - with a full manipulation of the population at every iteration, or
 - with the manipulation of a portion of the population.

4.2 Multi-Particle Collision Algorithm

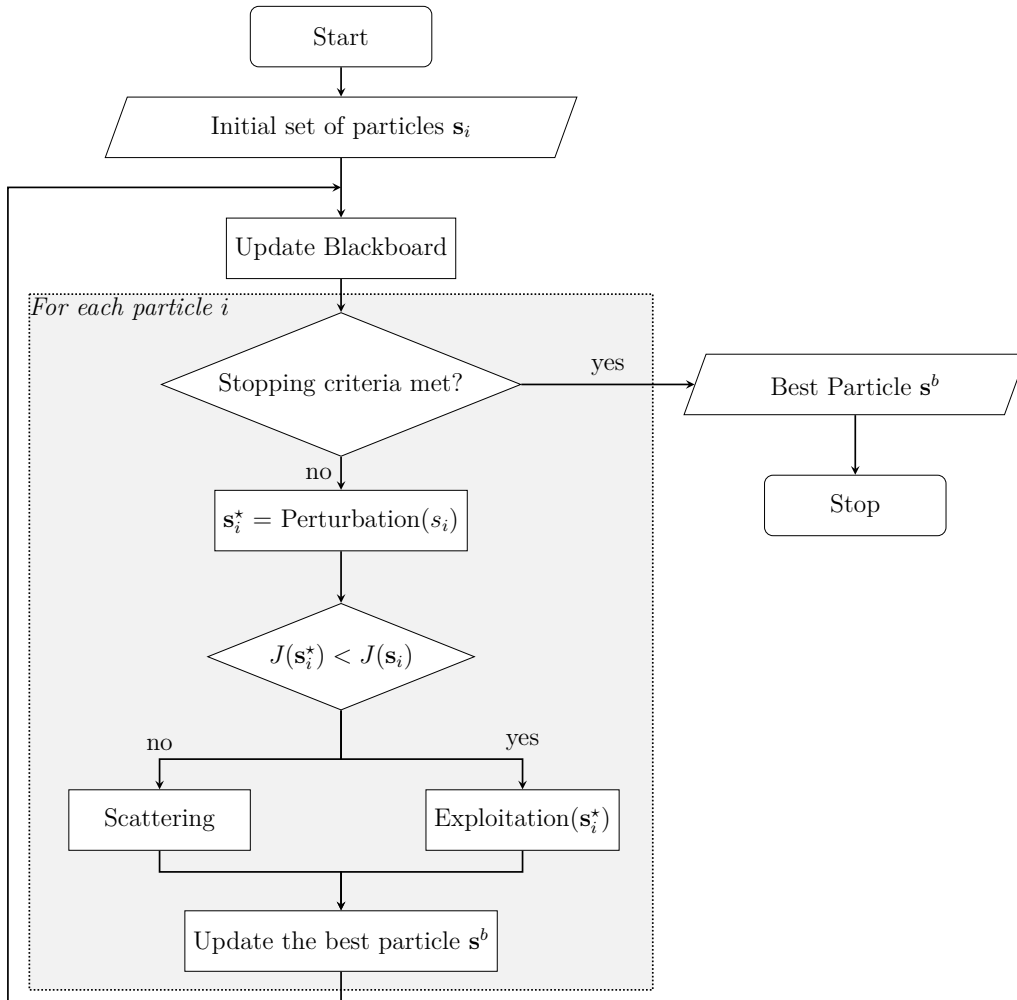
Multi-Particle Collision Algorithm (MPCA) is a metaheuristic inspired by the physics of nuclear particle collision reactions created by [Luz et al. \(2008\)](#) based on the Particle Collision Algorithm (PCA) from [Sacco and Oliveira \(2005\)](#). It is based on the scattering and absorption phenomena that occur inside the nuclear reactor. Scattering is when an incident particle is scattered by a target nucleus, while absorption is when an incident particle is absorbed by the target nucleus, as depicted in [Figure 4.2](#).

Figure 4.2 - Phenomena inside a nuclear reactor that inspire MPCA: (a) scattering; and (b) absorption.



In MPCA, a set of particles (i.e. candidate solutions), travels through the search space. There are three primary functions: perturbation, exploitation, and scattering. New solutions are created perturbing the particles. After the perturbation is applied, if the new position of the particle is better than the previous one, an intensification is made in its neighborhood, looking for improving, even more, the solution found. If the new particle position is worse than the previous one, two possible processes can be done, depending on a predefined probability: a new random position is generated, or a local search is performed in the neighborhood of the particle. The flowchart of the MPCA is shown in Figure 4.3, and the pseudo-code is presented in Algorithm 4.1.

Figure 4.3 - Flowchart of the Multi-Particle Collision Algorithm



Algorithm 4.1 Multi-Particle Collision Algorithm

```
1: Set MPCA control parameters ( $N_{processors}$ ,  $N_{particles}$ ,  $N_{FE}^{mpca}$ ,  $N_{FE}^{blackboard}$ ,  $s^l$ ,  $s^u$ ,  $R^{inf}$ ,  $R^{sup}$ )
2: for  $i \leftarrow 1$  to  $N_{processors}$  do ▷ Initial set of particles
3:    $N_{FE_i} = 0$ ,  $N_{FE_i}^{lastUpdate} = 0$ 
4:   for  $j \leftarrow 1$  to  $N_{particles}$  do
5:      $s_{i,j} = \text{RANDOM SOLUTION}$ 
6:      $N_{FE_i} = N_{FE_i} + 1$ 
7:      $s_i^b = \text{UPDATEBLACKBOARD}$  ▷ Initial blackboard
8:   while  $N_{FE_{total}} < N_{FE}^{mpca}$  do ▷ Stopping criteria
9:      $N_{FE_{total}} = 0$ 
10:    for  $i \leftarrow 1$  to  $N_{processors}$  do
11:      for  $j \leftarrow 1$  to  $N_{particles}$  do
12:         $s_{i,j}^* = \text{PERTURBATION}(s_{i,j})$ 
13:        if  $J(s_{i,j}^*) < J(s_{i,j})$  then
14:           $s_{i,j} = s_{i,j}^*$ 
15:           $s_{i,j} = \text{EXPLORATION}(s_{i,j})$ 
16:        else
17:           $s_{i,j} = \text{SCATTERING}(s_{i,j}, s_{i,j}^*, s_i^b)$ 
18:          if  $J(s_{i,j}) < J(s_i^b)$  then
19:             $s_i^b = s_{i,j}$ 
20:          if  $N_{FE_i} - N_{FE_i}^{lastUpdate} > N_{FE}^{blackboard}$  then
21:            for  $i \leftarrow 1$  to  $N_{processors}$  do ▷ Blackboard
22:               $s_i^b = \text{UPDATEBLACKBOARD}$ 
23:               $N_{FE_i}^{lastUpdate} = N_{FE_i}$ 
24:             $N_{FE_{total}} = N_{FE_{total}} + N_{FE_i}$ 
25:    for  $i \leftarrow 1$  to  $N_{processors}$  do ▷ Final blackboard
26:       $s_i^b = \text{UPDATEBLACKBOARD}$ 
27: return  $s_1^b$ 
```

MPCA has been used in the solution of some optimization problems such as fault diagnosis (ECHEVARRÍA et al., 2012), automatic configuration of neural networks applied to atmospheric temperature profile identification (SAMBATTI et al., 2012), data assimilation (ANOCHI et al., 2015) and climate prediction (ANOCHI; CAMPOS VELHO, 2014), as well as in the solution of an inverse radiative problem (HERNÁNDEZ TORRES et al., 2015a), obtaining good results.

A parallel version of MPCA using OpenMPI¹ was implemented in FORTRAN 90, in a multiprocessor architecture with distributed memory machine.

4.2.1 Initial set of particles

The initial set of particles is constituted by $N_{particles}$ particles. The generation of candidate solutions in the initial set of particles can be performed in two ways:

¹<https://www.open-mpi.org/>

using good solutions known so far, or creating the solution randomly within the search space.

For generating a random solution, each coordinate d of a solution $\mathbf{s} = (s_1, \dots, s_D)$ is found as:

$$s_d = s_d^l + (s_d^u - s_d^l) \times rand(0, 1). \quad (4.1)$$

4.2.2 Perturbation function

The **Perturbation** function performs a random variation of a particle within a defined range. A perturbed particle \mathbf{s}^* is completely defined by its coordinates:

$$s_d^* = s_d + (s_d^u - s_d) R - (s_d - s_d^l) (1 - R), \quad (4.2)$$

where \mathbf{s} is the particle to be perturbed, s_d^u and s_d^l are the upper and the lower limits of the defined search space, respectively, and $R = rand(0, 1)$.

The pseudo-code of this function is shown in [Algorithm 4.2](#).

Algorithm 4.2 Perturbation function

```

1: function PERTURBATION( $s$ )
2:   for  $d \leftarrow 1$  to  $D$  do
3:      $R = rand(0, 1)$ 
4:      $s_d^* = s_d + (s_d^u - s_d) R - (s_d - s_d^l) (1 - R)$ 
5:     if  $s_d^* > s_d^u$  then
6:        $s_d^* = s_d^u$ 
7:     else if  $s_d^* < s_d^l$  then
8:        $s_d^* = s_d^l$ 
9:    $N_{FE} = N_{FE} + 1$ 
10:  return  $\mathbf{s}^*$ 

```

If $J(\mathbf{s}^*) < J(\mathbf{s})$, then the particle \mathbf{s} is replaced with \mathbf{s}^* , and the **EXPLOITATION** function (see [Section 4.2.3](#)) is activated, performing an exploitation in the neighborhood of the particle. If the new perturbed particle \mathbf{s}^* is worse than the current particle \mathbf{s} , the **SCATTERING** function (see [Section 4.2.4](#)) is launched.

4.2.3 Exploitation function

In this stage, the algorithm performs a series of $N_{FE}^{exploitation}$ small perturbations, computing a new particle \mathbf{s}^s each time, using the following equation for each coor-

dinate:

$$s_d^{\S} = s_d + (u - s_d) R - (s_d - l) (1 - R) , \quad (4.3)$$

where $u = s_d \times \text{rand}(1, R^{sup})$ is the small upper limit, and $l = s_d \times \text{rand}(R^{inf}, 1)$ is the small lower limit, with a superior value R^{sup} and a inferior value R^{inf} for the generation of the random number. R^{sup} and R^{inf} are both defined empirically.

The pseudo-code of this function is shown in [Algorithm 4.3](#) and [Algorithm 4.4](#).

Algorithm 4.3 Exploitation function

```

1: function EXPLOITATION( $s$ )
2:   for  $n \leftarrow 1$  to  $N_{FE}^{exploitation}$  do
3:      $s^{\star} = \text{SMALLPERTURBATION}(s)$ 
4:      $N_{FE} = N_{FE} + 1$ 
5:     if  $J(s^{\star}) < J(s)$  then
6:        $s = s^{\star}$ 
7:   return  $s$ 

```

Algorithm 4.4 Small Perturbation function

```

1: function SMALLPERTURBATION( $s$ )
2:   for  $d \leftarrow 1$  to  $D$  do
3:      $u = s_d \times \text{rand}(1, R^{sup})$ 
4:      $l = s_d \times \text{rand}(R^{inf}, 1)$ 
5:      $R = \text{rand}(0, 1)$ 
6:     if  $u > s_d^u$  then
7:        $u = s_d^u$ 
8:     if  $l < s_d^l$  then
9:        $l = s_d^l$ 
10:     $s_d^{\S} = s_d + (u - s_d) R - (s_d - l) (1 - R)$ 
11:   return  $s^{\S}$ 

```

4.2.3.1 Exploitation in a single random dimension each time

The exploitation function is modified, performing a small perturbation in only one dimension chosen randomly, as shown in [Algorithm 4.5](#).

Algorithm 4.5 Small Perturbation in a single random dimension function

```
1: function SMALLPERTURBATION( $s$ )
2:    $d = \text{rand}(0, 1) \times (D - 1) + 1$ ;
3:    $u = s_d \times \text{rand}(1, R^{sup})$ 
4:    $l = s_d \times \text{rand}(R^{inf}, 1)$ 
5:    $R = \text{rand}(0, 1)$ 
6:   if  $u > s_d^u$  then
7:      $u = s_d^u$ 
8:   if  $l < s_d^l$  then
9:      $l = s_d^l$ 
10:   $s_d^{\S} = s_d + (u - s_d)R - (s_d - l)(1 - R)$ 
11:  return  $s_d^{\S}$ 
```

4.2.4 Scattering function

The SCATTERING function works as a Metropolis scheme. The pseudo-code is presented in [Algorithm 4.6](#).

Algorithm 4.6 Scattering function

```
1: function SCATTERING( $\mathbf{s}, \mathbf{s}^*, \mathbf{s}^b$ )
2:   Calculate  $p^s$ 
3:   if  $p^s > \text{rand}(0, 1)$  then
4:      $\mathbf{s} = \text{RANDOMSOLUTION}$ 
5:      $N_{FE} = N_{FE} + 1$ 
6:   else
7:      $\mathbf{s} = \text{EXPLORATION}(\mathbf{s})$ 
8:   return  $\mathbf{s}$ 
```

The particle is replaced by a new random solution \mathbf{s} using [Equation \(4.1\)](#), or the exploitation is made, with a given probability.

The scattering probability p^s can be found in some different ways, such as:

- Truncated exponential distribution:

$$p^s = 1 - \frac{J(s^b)}{J(s)}. \quad (4.4)$$

- Cauchy distribution:

$$p^s = \frac{1}{\pi\gamma \left(1 + \left(\frac{J(s) - J(s^b)}{\gamma}\right)^2\right)}, \quad \gamma = 1. \quad (4.5)$$

4.2.5 Blackboard updating function

A mechanism called *blackboard updating* allows sending the best solution overall reached in certain moments of the algorithm to the other particles. That moment is determined by the number of function evaluations. Each $N_{FE}^{blackboard}$ function evaluations, the best particle overall s^\diamond is selected and sent to all the particles, updating the reference s^b .

Algorithm 4.7 UpdateBlackboard function using MPI

```

1: function UPDATEBLACKBOARD( $s^b$ )
2:   if is master processor then
3:      $s^\diamond = s^b$ 
4:     for  $i \leftarrow 1$  to  $N_{processors}$  do
5:        $s = \text{MPI\_RECEIVE}(i)$       ▷ Receive best particle from each processor
6:       if  $J(s) < J(s^\diamond)$  then
7:          $s^\diamond = s$                 ▷ Update the best particle overall
8:        $\text{MPI\_BROADCAST}(s^\diamond)$     ▷ Send best particle overall to the other processors
9:   else                                ▷ Other processors
10:     $\text{MPI\_SEND}(s^b)$               ▷ Send the self best particle to the master processor
11:     $\text{MPI\_BROADCAST}(s^\diamond)$     ▷ Wait for the best particle overall
12:   return  $s^\diamond$ 

```

4.2.6 Stopping criteria

As for stopping criterion, a maximum number of function evaluations (N_{FE}^{mpca}) is defined.

4.3 Opposition-Based Optimization and some mechanisms derived

The Opposition-based Learning (OBL) mechanism was created by Tizhoosh (2005) in 2005. The idea of OBL is to consider the opposite of a candidate solution, which has a certain probability of being closer to the global optimum.

Some mechanisms derived from OBL have been developed, such as **Quasi-opposition** (QO), **Quasi-reflection** (QR), **Center-based** sampling (CB), and

Rotation-based Learning (RBL) (ERGEZER et al., 2009; RAHNAMAYAN et al., 2007; TIZHOOSH; VENTRESCA, 2008). QO reflects a point to a random point between the center of the domain and the opposite point. QR projects the point to a random point between the center of the domain and itself. CB creates a point between itself and its opposite.

In a short amount of time, these mechanisms have been utilized in different soft computing areas, improving the performance of various techniques of Computational Intelligence, such as metaheuristics, artificial neural networks, fuzzy logic, and other applications (XU et al., 2014).

For a better understanding of the mechanisms, it is necessary to define the concept of some specific numbers.

Definition 4.1. Let $s \in [s^l, s^u]$ be a real number, and $c = (s^l + s^u)/2$. The opposite number s_o , the quasi-opposite number s_{qo} , the quasi-reflected number s_{qr} , and the center-based sampled number s_{cb} are defined as:

$$s_o = s^l + s^u - s; \quad (4.6)$$

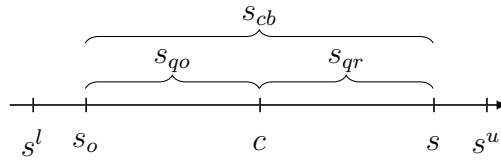
$$s_{qo} = rand(s_o, c); \quad (4.7)$$

$$s_{qr} = rand(c, s); \quad (4.8)$$

$$s_{cb} = rand(s_o, s). \quad (4.9)$$

Figure 4.4 shows a graphical representation of these numbers.

Figure 4.4 - Graphical representation of the opposition number (s_o), quasi-opposite number (s_{qo}), quasi-reflected number (s_{qr}) and center-based sampled number (s_{cb}) from the original number s



Definition 4.2. Let $\mathbf{s} = (s_1, \dots, s_D) \in \mathbb{R}^D$ be a point, $s_d \in [s_d^l, s_d^u], \forall d \in (1, \dots, D)$. The opposite point \mathbf{s}_o , the quasi-opposite point \mathbf{s}_{qo} , the quasi-reflected point \mathbf{s}_{qr} , and the center-based sampled point \mathbf{s}_{cb} are completely defined by their

coordinates:

$$s_{o_d} = s_d^l + s_d^u - s_d; \quad (4.10)$$

$$s_{qo_d} = rand(s_{o_d}, c_d); \quad (4.11)$$

$$s_{qr_d} = rand(c_d, s_d); \quad (4.12)$$

$$s_{cb_d} = rand(s_{o_d}, s_d). \quad (4.13)$$

The Rotation-Based Learning (RB) mechanism is another extension of the OBL (LIU et al., 2014), and the Rotation-Based Sampling (RBS) is a combination of the Center-Based Sampling and RBL mechanisms.

Definition 4.3. Let $s \in [s^l, s^u]$ be a real number, and $c = (s^l + s^u)/2$ be the center. Draw a circle with center c and radius $c - s^l$. The point $(s, 0)$ is projected on the circle. Defining the quantity from the original number to the center $u = s - c$, the length from the original number to the corresponding intersection point l on the circle $v = \sqrt{(s - s^l)(s^u - s)}$, and the deflection angle $\beta = \beta_0 \mathcal{N}(1, \delta)$, with mean β_0 and standard deviation δ . The rotation number s_r is defined as:

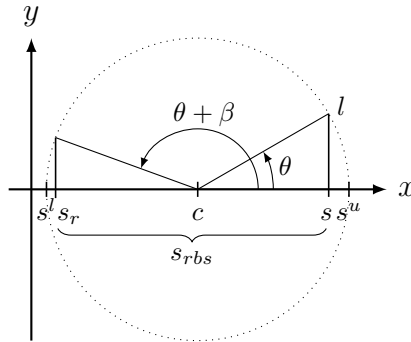
$$s_r = c + u \times \cos \beta - v \times \sin \beta. \quad (4.14)$$

Definition 4.4. The rotation-based sampling number s_{rbs} is defined as:

$$s_{rbs} = rand(s_r, s). \quad (4.15)$$

The geometric representation in a 2D-space of the Rotation and the Rotation-Based Sampling numbers is shown in Figure 4.5.

Figure 4.5 - Geometric interpretation of the Rotation (s_r) and the Rotation-based Sampling (s_{rbs}) numbers in 2D space



Similarly to [Definition 4.2](#), the concepts of rotation and rotation-based sampling points are enunciated:

Definition 4.5. Let $\mathbf{s} = (s_1, \dots, s_D) \in \mathbb{R}^D$ be a point, and $s_d \in [s_d^l, s_d^u], \forall d \in (1, \dots, D)$. The rotation point s_r , is completely defined by their coordinates:

$$s_{r_d} = c_d + u_d \times \cos \beta - v_d \times \sin \beta. \quad (4.16)$$

Definition 4.6. Let $\mathbf{s} = (s_1, \dots, s_D) \in \mathbb{R}^D$ be a point, and $s_d \in [s_d^l, s_d^u], \forall d \in (1, \dots, D)$. The rotation-based sampling number s_{rbs} is completely defined by their coordinates:

$$s_{rbs_d} = rand(s_{r_d}, s_d). \quad (4.17)$$

Definition 4.7. Opposition-based Optimization – Let $\mathbf{s} \in \mathbb{R}^D$ be a point (*i.e.*, candidate solution), and \mathbf{s}_o a opposite point of \mathbf{s} (*i.e.*, opposite candidate solution). If $J(\mathbf{s}_o) \leq J(\mathbf{s})$, then the point \mathbf{s} can be replaced with \mathbf{s}_o , which is better, otherwise it will maintain its current value.

The solution and the opposite solution are evaluated simultaneously, and the optimization process will continue with the better one.

The same idea of the Opposition-based Optimization is applicable for the other mechanisms. To abbreviate, when the text refers to all these mechanisms, the terms OBO or Opposition will be used without distinction. [Algorithm 4.8](#) shows a pseudo-code of a possible computational implementation of the Opposition.

Algorithm 4.8 Opposition function

```
1: function OPPOSITION(s)
2:   for  $d \leftarrow 1$  to  $D$  do
3:     Obtain  $s_d^l, s_d^u$ 
4:      $c_d = \frac{(s_d^l + s_d^u)}{2}$  ▷ Calculate center of the search space
5:      $s_{od} = s_d^l + s_d^u - s_d$  ▷ Calculate opposite point
6:      $R = \text{rand}(0, 1)$ 
7:     switch type do
8:       case quasi-opposition
9:         if  $s_d < c_d$  then
10:            $s_{od} = c_d + R(s_{od} - c_d)$ 
11:         else
12:            $s_{od} = s_{od} + R(c_d - s_{od})$ 
13:       case quasi-reflected
14:         if  $s_d < c_d$  then
15:            $s_{od} = s_d + R(c_d - s_d)$ 
16:         else
17:            $s_{od} = c_d + R(s_d - c_d)$ 
18:       case center-based sampling
19:         if  $s_d < c_d$  then
20:            $s_{od} = s_d + R(s_{od} - s_d)$ 
21:         else
22:            $s_{od} = s_{od} + R(s_d - s_{od})$ 
23:       case rotation-based sampling
24:          $u_d = s_d - c_d$ 
25:          $v_d = \sqrt{(s_d - s_d^l)(s_d^u - s_d)}$ 
26:          $s_{rd} = c_d + u_d \times \cos \beta - v_d \times \sin \beta$ 
27:         if  $s_d < c_d$  then
28:            $s_{od} = s_d + R(s_{rd} - s_d)$ 
29:         else
30:            $s_{od} = s_{rd} + R(s_d - s_{rd})$ 
31:       default ▷ Opposition
32:          $s_{od} = s_{od}$ 
33:      $N_{FE} = N_{FE} + 1$ 
34:     if  $J(s_o) < J(s)$  then
35:       return  $s_o$ 
36:     else
37:       return  $s$ 
```

4.4 Multi-Particle Collision Algorithm with Opposition-Based Optimization derived mechanisms

The hybrid of MPCA with Opposition is defined as follows. The pseudo-code of the computational implementation of the generic algorithm is shown in [Algorithm 4.9](#).

Algorithm 4.9 Multi-Particle Collision Algorithm with Opposition

```
1: Set MPCA control parameters ( $N_{processors}$ ,  $N_{particles}$ ,  $N_{FE}^{mpca}$ ,  $N_{FE}^{blackboard}$ ,  $s^l$ ,  $s^u$ ,  $R^{inf}$ ,  $R^{sup}$ )
2: for  $i \leftarrow 1$  to  $N_{processors}$  do ▷ Initial set of particles
3:    $N_{FE_i} = 0$ ,  $N_{FE_i}^{lastUpdate} = 0$ 
4:   for  $j \leftarrow 1$  to  $N_{particles}$  do
5:      $s_{i,j}^* = \text{RANDOM SOLUTION}$ 
6:      $N_{FE_i} = N_{FE_i} + 1$ 
7:      $s_{i,j}^* = \text{OPPOSITION}(s_{i,j}^*)$ 
8: for  $i \leftarrow 1$  to  $N_{processors}$  do ▷ Initial blackboard
9:    $s_i^b = \text{UPDATEBLACKBOARD}$ 
10: while  $N_{FE_i} < N_{FE}^{mpca}$  do ▷ Stopping criteria
11:   for  $i \leftarrow 1$  to  $N_{processors}$  do
12:     for  $j \leftarrow 1$  to  $N_{particles}$  do
13:        $s_{i,j} = \text{PERTURBATION}(s_{i,j}^*)$ 
14:       if  $J(s_{i,j}) < J(s_{i,j}^*)$  then
15:          $s_{i,j}^* = s_{i,j}$ 
16:          $s_{i,j}^* = \text{EXPLORATION}(s_{i,j}^*)$ 
17:       else
18:          $s_{i,j}^* = \text{SCATTERING}(s_{i,j}^*, s_{i,j}, s_i^b)$ 
19:         if  $J(s_{i,j}^*) < J(s_i^b)$  then
20:            $s_i^b = s_{i,j}^*$ 
21:            $s_i^b = \text{OPPOSITION}(s_i^b)$ 
22:       if  $N_{FE_i} - N_{FE_i}^{lastUpdate} > N_{FE}^{blackboard}$  then
23:         for  $i \leftarrow 1$  to  $N_{processors}$  do ▷ Blackboard
24:            $s_i^b = \text{UPDATEBLACKBOARD}$ 
25:            $N_{FE_i}^{lastUpdate} = N_{FE_i}$ 
26: for  $i \leftarrow 1$  to  $N_{processors}$  do ▷ Final blackboard
27:    $s_i^b = \text{UPDATEBLACKBOARD}$ 
28: return  $s_1^b$ 
```

4.4.1 Initialization using Opposition

Random number generation is commonly the most used choice to create an initial population. The use of opposition working together with randomness permits to obtain better-starting candidates even when there is no *a priori* knowledge about the solution (TIZHOOSH, 2005).

In the proposed hybrid versions, the first step is to create the initial solution for each particle as usual. Next, the opposite solution is calculated within the search space $[s_d^l, s_d^u]$. The original solution is substituted by the opposite if the latter has a better fitness (see Algorithm 4.9, line 7).

4.4.2 Traveling in the search space using Opposition

The application of Opposition on the traveling of the particles in the search space is dependent on the MPCa function being called.

When the **Perturbation** function is applied on a particle, the opposite particle is calculated at the same time. The best particle among them will be maintained as the new particle (see [Algorithm 4.10](#), line 10). The bounds to create the opposite particle is dynamically reduced to $[s_d^l, s_d^u]$, where s_d^l and s_d^u are the minimum and maximum values for each dimension in all the population of particles.

Algorithm 4.10 Perturbation function with Opposition

```

1: function PERTURBATION(s)
2:   for  $d \leftarrow 1$  to  $D$  do
3:      $R = \text{rand}(0, 1)$ 
4:      $s_d^* = s_d + (s_d^u - s_d) R - (s_d - s_d^l) (1 - R)$ 
5:     if  $s_d^* > s_d^u$  then
6:        $s_d^* = s_d^u$ 
7:     else if  $s_d^* < s_d^l$  then
8:        $s_d^* = s_d^l$ 
9:    $N_{FE} = N_{FE} + 1$ 
10:   $\mathbf{s}^* = \text{OPPOSITION}(\mathbf{s}^*)$ 
11:  return  $\mathbf{s}^*$ 

```

When the **Exploration** is performed, an opposite particle is also calculated, based on a probability, called jumping rate $J_r \in [0, 1]$ (see [Algorithm 4.11](#), lines 6). This implies that the opposition is applied only few times. The value of J_r is chosen empirically ([SIMON, 2013](#)).

The bounds to create the opposite particle is dynamically reduced to $[s_d^l, s_d^u]$, as was done in the **Perturbation** function.

Algorithm 4.11 Exploitation function with Opposition

```

1: function EXPLOITATION( $s$ )
2:   for  $n \leftarrow 1$  to  $N_{FE_{\text{exploitation}}}$  do
3:      $s^* = \text{SMALLPERTURBATION}(s)$ 
4:      $N_{FE} = N_{FE} + 1$ 
5:     if  $\text{rand}(0, 1) < J_r$  then
6:        $s^* = \text{OPPOSITION}(s^*)$ 
7:     if  $J(s^*) < J(s)$  then
8:        $s = s^*$ 
9:   return  $s$ 

```

In the **Scattering** function, if a random particle is created, then the opposite particle is also created using the original bounds $[s_d^l, s_d^u]$. The best particle among them will be maintained (see [Algorithm 4.12](#), line 6).

Algorithm 4.12 Scattering function with Opposition

```

1: function SCATTERING( $s, s^*, s^b$ )
2:   Find  $p^s$ 
3:   if  $p^s > rand(0, 1)$  then
4:      $s^n = \text{RANDOMSOLUTION}$ 
5:      $N_{FE} = N_{FE} + 1$ 
6:      $s = \text{OPPOSITION}(s^n)$ 
7:   else
8:      $s = \text{EXPLORATION}(s)$ 
9:   return  $s$ 

```

After applying the **Perturbation**, the **Exploration**, and the **Scattering** functions to generate the new solution, according to the [Algorithm 4.9](#), the opposite of the best particle heretofore is calculated using the computed limits $[s_d^l, s_d^u]$ (see [Algorithm 4.9](#), line 21).

4.5 q -Gradient Method

The first concepts of the q -calculus were developed by [Jackson \(1908\)](#), [Jackson \(1910a\)](#), [Jackson \(1910b\)](#). At the beginning of the 20th century appeared the q -analogs of functions, series, special numbers, and the q -derivative concepts, including the q -gradient vector.

The q -gradient (qG) method can be described as a q -analog of the *steepest descent* method that reduces to its classical version whenever the parameter $q = 1$. For $q \neq 1$, the search direction is likely to be either descent or not descent, and the method performs a global search.

In the qG method, the search process gradually shifts from global in the beginning to almost local search in the end ([SOTERRONI et al., 2012](#)).

Definition 4.8. Let J be a differentiable (objective) function of D variables, the D first-order partial q -derivatives of J with respect to the variable s_d are given by Equation (4.1) ([SOTERRONI et al., 2011](#); [SOTERRONI et al., 2012](#); [SOTERRONI et al.,](#)

2013).

$$D_{q_d, s_d} J(s) = \begin{cases} \frac{J(s_1, \dots, q_d s_d, \dots, s_D) - J(s_1, \dots, s_D)}{q_d s_d - s_d}, & \text{if } s_d \neq 0 \text{ and } q_d \neq 1, \\ \frac{\partial J(s)}{\partial s_d}, & \text{otherwise,} \end{cases} \quad (4.1)$$

where $\mathbf{q} = (q_1, \dots, q_D) \in \mathbb{R}^D$.

When $s_d = 0$ or $q_d = 1$, $\forall d$, the first-order partial q -derivative is the classical first-order partial derivative.

Definition 4.9. The q -gradient is the vector of the n first-order partial q -derivatives of f (SOTERRONI et al., 2011; SOTERRONI et al., 2012; SOTERRONI et al., 2013)

$$\nabla_q J(s) = [D_{q_1, s_1} J(s) \cdots D_{q_d, s_d} J(s) \cdots D_{q_D, s_D} J(s)]. \quad (4.2)$$

The q G method uses an iterative procedure that, starting from an initial point \mathbf{s} , generates a sequence \mathbf{s} given by:

$$\mathbf{s} = \mathbf{s} + \alpha \mathbf{v}, \quad (4.3)$$

where v is the search direction, and α is the step length or the distance moved along v . Thus, the search direction v in the q G method is the negative of the q -gradient of the objective function $-\nabla_q J(s)$ (Equation (4.2)).

The parameters q_d are generated drawing their values from a Gaussian probability distribution, such that $q_d s_d$ has a standard deviation σ that decreases as the iterative search proceeds (SOTERRONI et al., 2012; SOTERRONI et al., 2013). The process starts from a given σ , that is decreased by the reduction factor $\beta : 0 < \beta < 1$. When σ goes to 0, q_d tend to unity. The q G method starts as a global search method, and gradually becomes a local search, with a similar behavior to the *steepest descent* method (SOTERRONI et al., 2013)

The step length α is computed by the geometric recursion $\alpha = \beta \cdot \alpha$, where β is the same for updating σ (SOTERRONI et al., 2012; SOTERRONI et al., 2013).

The values of σ and α are normalized by the largest distance within the search space L , calculated by:

$$L = \sqrt{\sum_{d=1}^D (s_d^u - s_d^l)^2}. \quad (4.4)$$

4.5.1 Stopping criteria

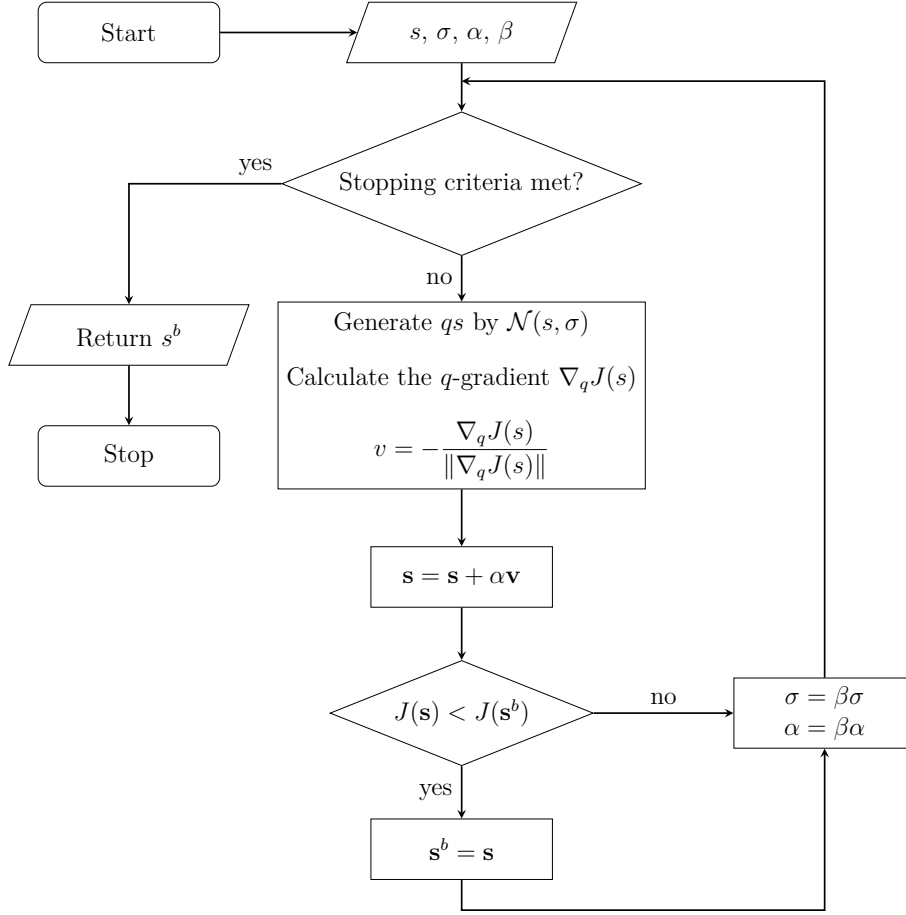
The stopping criterion is the maximum number of the function evaluations (N_{FE}^{qG}).

The pseudo-code for the qG method is summarized in the [Algorithm 4.13](#), and its flowchart is shown in [Figure 4.6](#).

Algorithm 4.13 q -gradient method

```
1: Given initial point  $s$ ,  $\sigma > 0$ ,  $\alpha > 0$  and  $0 < \beta < 1$ 
2:  $s^b = s$ 
3: while  $N_{FE} < N_{FE}^{qG}$  do ▷ Stopping criterion
4:   Generate  $qs$  by a Gaussian distribution with  $\sigma$ , and  $\mu = s$ 
5:   Calculate the  $q$ -gradient  $\nabla_q J(s)$ 
6:    $d = -\nabla_q J(s) / \|\nabla_q J(s)\|$ 
7:    $s = s + \alpha d$ 
8:   if  $J(s) < J(s^b)$  then
9:      $s^b = s$ 
10:     $N_{FE} = N_{FE} + 1$ 
11:     $\sigma = \beta \sigma$ 
12:     $\alpha = \beta \alpha$ 
13: return  $s^b$ 
```

Figure 4.6 - Flowchart of the q -gradient



4.6 Hooke-Jeeves Pattern Search Method

The direct search method proposed by [Hooke and Jeeves \(1961\)](#) consists of the repeated application of exploratory moves around a base point s defined in a n -dimensional space. If these steps are successful, they are followed by pattern moves.

Hooke-Jeeves (HJ) method has been used in some hybrid algorithms, such as HJPCA ([RIOS-COELHO et al., 2010](#)), HJMA ([MOSER; CHIONG, 2009](#)), PSO/HJ ([KÄMPF et al., 2010](#)), GAHJ ([LONG; WU, 2014](#)), AS+HJ ([BRAUN et al., 2015](#)), qG-HJ ([HERNÁNDEZ TORRES et al., 2015b](#)), and MPCA-HJ ([HERNANDEZ TORRES et al., 2015](#)).

Figures 4.7 and 4.8, and Algorithms 4.14 and 4.15 show the flowchart and the pseudo-code of the HJ method, respectively.

Figure 4.7 - Flowchart of the Hooke-Jeeves direct search method

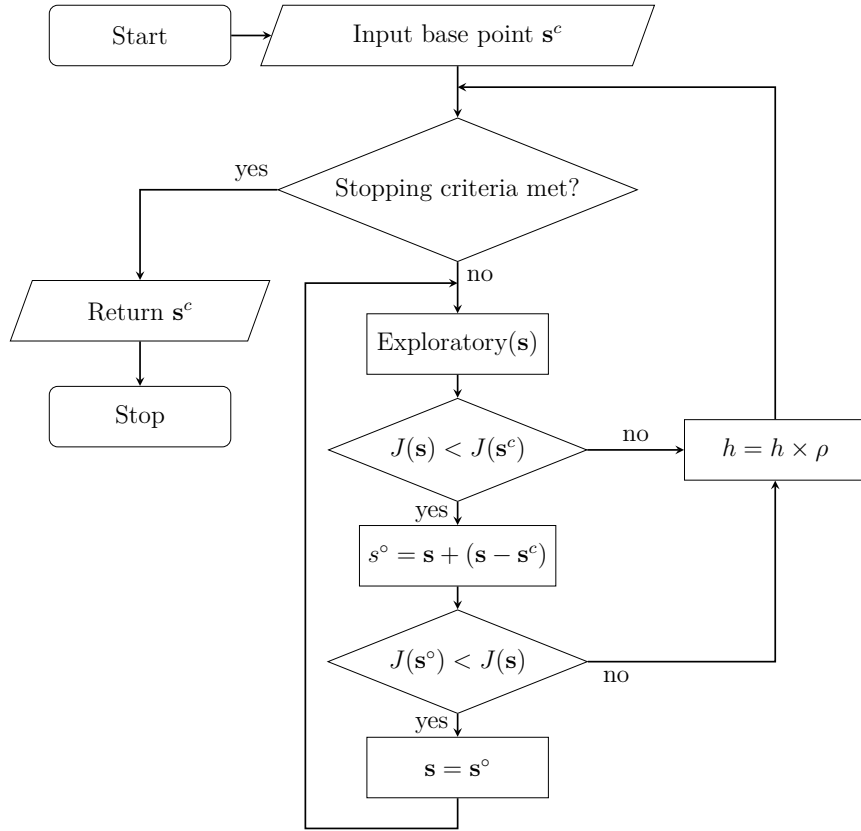
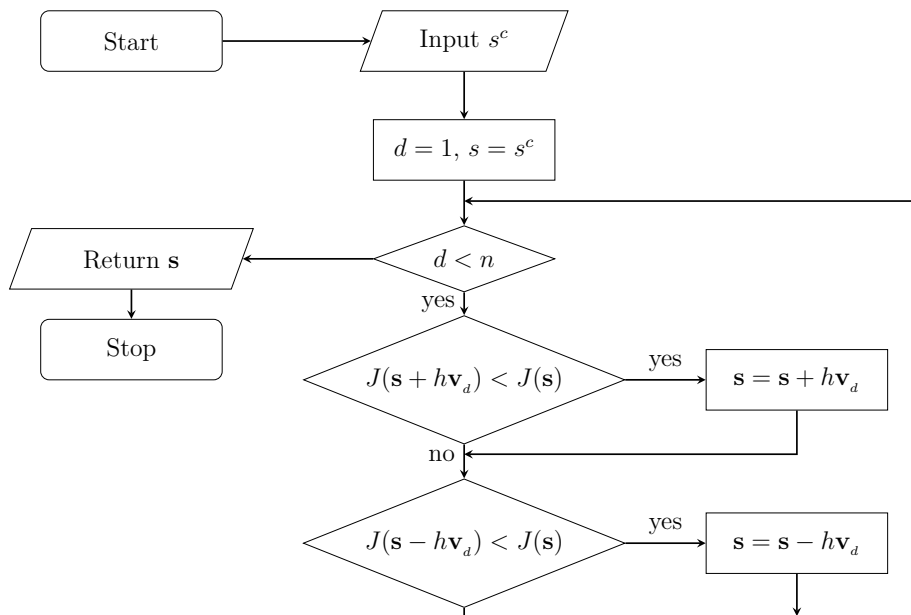


Figure 4.8 - Flowchart of the Exploratory function in the Hooke-Jeeves direct search method



Algorithm 4.14 Hooke-Jeeves pattern search method

```
1: Choose  $s^c, \rho, h_{\min}$ 
2: while  $N_{FE} < N_{FE}^{hj}$  and  $h > h_{\min}$  do ▷ Stopping criteria
3:    $s = \text{EXPLORATORY}(s^c, h)$ 
4:   if  $J(s) < J(s^c)$  then
5:      $s^\circ = s + (s - s^c)$ 
6:     if  $J(s^\circ) < J(s)$  then
7:        $s^c = s^\circ$ 
8:     else
9:        $s^c = s$ 
10:     $N_{FE} = N_{FE} + 1$ 
11:  else
12:     $h = h \times \rho$ 
13: return  $s^c$ 
```

Algorithm 4.15 Exploratory function

```
1: function EXPLORATORY( $s^c, h$ )
2:    $s = s^c$ 
3:   for  $d \leftarrow 1$  to  $D$  do
4:     if  $J(s + hv_d) < J(s)$  then
5:        $s = s + hv_d$ 
6:      $N_{FE} = N_{FE} + 1$ 
7:     if  $J(s - hv_d) < J(s)$  then
8:        $s = s - hv_d$ 
9:      $N_{FE} = N_{FE} + 1$ 
10:  return  $s$ 
```

4.6.1 Exploratory move

In the exploratory move, the solution is changed adding and subtracting each time, the column vector \mathbf{v}_d of the search direction matrix \mathbf{V} , scaled by a step size h . When \mathbf{V} is the identity matrix, the modification is performed on the d -th element of the solution \mathbf{s} each time. This process is done for all the dimensions of the solution. The original solution is compared with both solutions ($\mathbf{s} + h\mathbf{v}_d$ and $\mathbf{s} - h\mathbf{v}_d$) created, and the best among them will be returned.

4.6.2 Pattern move

A new pattern point s° is calculated as follows:

$$\mathbf{s}^\circ = \mathbf{s} + (\mathbf{s} - \mathbf{s}^c) , \quad (4.5)$$

where \mathbf{s} is the solution obtained from the exploratory move, and \mathbf{s}^c is the current solution. If the pattern point \mathbf{s}° is better than \mathbf{s}^c , \mathbf{s}^c will be replaced with \mathbf{s}° . If there is no improvement, the step size h is reduced in ρ times.

4.6.3 Stopping criteria

A minimum step size h_{min} and a maximum number of function evaluations N_{FE}^{hj} are defined as stopping criteria.

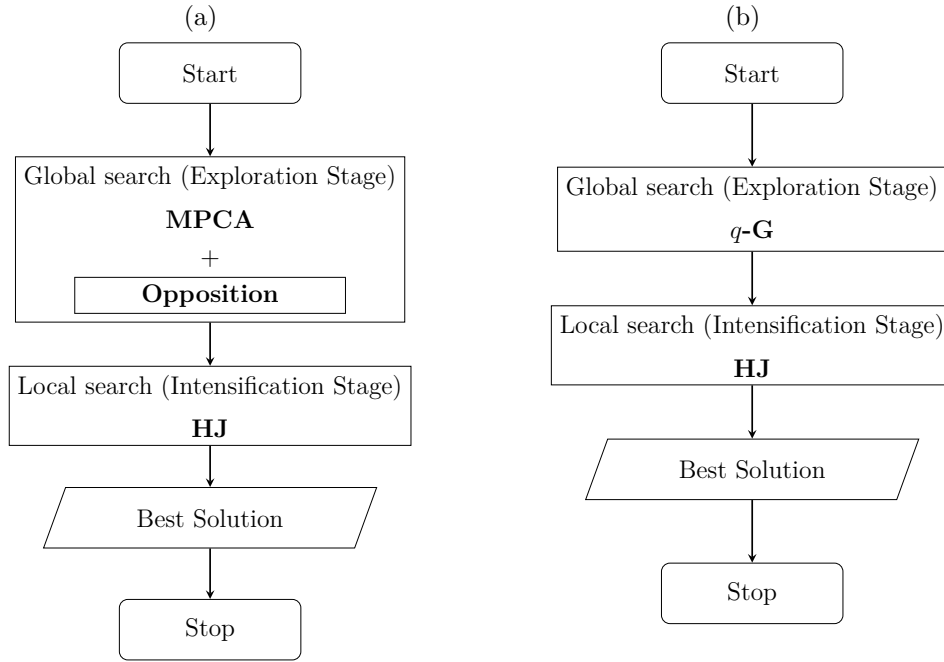
4.7 Hybrid approaches: OMPCA-HJ and qG-HJ

The hybrid algorithms Multi-Particle Collision Algorithm with Hooke-Jeeves (MPCA-HJ) and its variants use an integration scheme of the MPCA with Opposition (OMPCA-HJ), with Center-based Sampling (CMPCA-HJ), or with Rotation-based Sampling (RMPCA-HJ), for improving the global search stage. The first phase of exploration is followed by an intensification stage performed by HJ.

Similarly, the q -gradient method with Hooke-Jeeves (q G-HJ) uses the same approach of the MPCA-HJ, with an exploration phase with q G and an exploitation phase with HJ.

Figure 4.9 presents the operating flow of the hybrid algorithms.

Figure 4.9 - Operating flow of the hybrid algorithms: (a) OMPCA-HJ; (b) q G-HJ;



4.8 Chapter conclusions

The chapter presents the optimization algorithms applied to hybrid approaches used to solve the inverse problem of damage identification. The description of algorithms includes pseudo-code, flowchart, and an explanation on the operation of MPCA, and its variants using Opposition, q G method, and HJ direct search method. The last section presented the scheme to hybridize these algorithms, with an operation in two stage: the global (exploration) phase, and the local (exploitation) phase.

The next chapter shows a compilation of case studies using the damage identification methodology presented, where the model is implemented in FORTRAN.

5 VIBRATION-BASED DAMAGE IDENTIFICATION ON SYSTEMS/STRUCTURES MODELED WITH FORTRAN, AND USING *IN SILICO* EXPERIMENTAL DATA

FORTRAN is not a flower but a weed – it is hardy, occasionally blooms, and grows in every computer.

ALAN J. PERLIS

The methodology presented in the previous chapters for identifying structural damages will be applied for some structures, whose models were implemented in FORTRAN.

Initially, the steps of the experimental design will be defined. Next, some study cases will be presented: a damped spring-mass system with 10-DOF, the Kabe's problem, the Yang's problem, a truss system with 12-DOF, a truss model for the International Space Station, and a cantilevered beam.

5.1 Experimental design

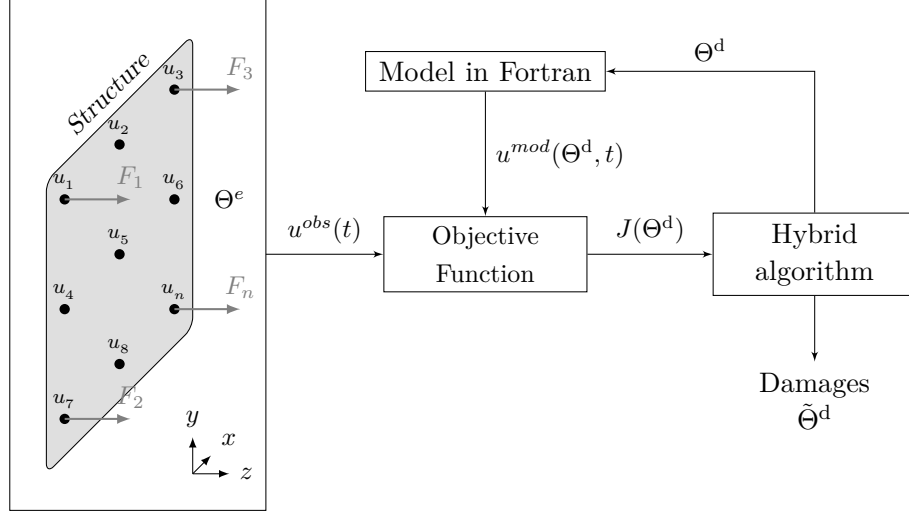
The experimental design to identify damage in structures consists of the following steps:

- (1) Implement the solution of the model of the structure using FORTRAN.
- (2) Create *in silico* observed data u^{obs} using the damage configuration as input of the model, *i.e.* modifying the stiffness of the elements ($\mathbf{k}^{d*} = \mathbf{k}^i$).
- (3) Add noise to the synthetic measurements.
- (4) Set-up the hybrid algorithm depending on the characteristics of the structure.
- (5) Run the hybrid algorithm at least 50 times, and tabulate the output damage parameters.
- (6) Compute some statistics, depending on the analysis to be performed.

The lower and the upper bounds of the search space are defined between -5% and 50% of damage, respectively. A negative damage, although it is not allowed physically, is set as lower bound. This configuration facilitates the exploration to the hybrid optimization algorithms, because the most solutions are located in the

value of 0%.

Figure 5.1 - Vibration-Based Damage Identification as optimization problem using models implemented in FORTRAN



All the experiments in this chapter were done on a PC with 8× Intel® Core™ i7-4790 CPU @ 3.60GHz, with 8 GB of memory, operating with Ubuntu 16.04.3 LTS.

The configuration common to all experiments is shown in [Table 5.1](#). Those parameters were set empirically.

Table 5.1 - Control parameters for the hybrid algorithms

Algorithm	Parameter	Value
MPCA	$N_{particles}$	10
	N_{FE}^{mpca}	10000
	$N_{FE}^{blackboard}$	1000
	$N_{FE}^{exploitation}$	200
	R^{inf}	0.7
	R^{sup}	1.1
	Initial population p^s	Random + One solution is set in the best known Cauchy
CBS & RBS	J_r	0.01
RBS	β_0	3.14 rad
	δ	0.25
qG	N_{FE}^{qG}	100000
	σ^0	$0.2L$
	α^0	$0.1L$
	β	0.999
HJ	N_{FE}^{hj}	10000
	h_{min}	1×10^{-10}
	ρ	0.8

The way to add noise to the synthetic data will be presented in the next section.

5.1.1 Noise in data

In real life, measurements are commonly corrupted by noise that affects the instrumentation channel, including sensors, wires, and other electronic devices. In another hand, the solution of an inverse problem can be affected by small perturbations or noise on the measurement.

For testing the robustness of the algorithm in the solution of the inverse problem, some noise is added to the synthetic measured data (NICHOLS; MURPHY, 2016):

$$\hat{\mathbf{u}}(t) = \mathbf{u}(t) + \mathcal{N}(0, \sigma^2), \quad (5.1)$$

where $\hat{\mathbf{u}}(t)$ is the noisy data, and $\mathcal{N}(0, \sigma^2)$ is the Gaussian distribution with mean 0 and variance σ^2 .

5.2 Case study 1: Damage identification in a damped spring-mass system with 10-DOF

The results in this section were presented at the 10th International Conference on Composite Science and Technology – (HERNANDEZ TORRES et al., 2015).

The first case in which the methodology will be tested is a damped spring-mass system with 10-DOF, as shown in Figure 5.2. All the system components are assumed as $m = 10.0 \text{ kg}$, the springs have a stiffness $k = 2 \times 10^5 \text{ N/m}$, and the damping matrix is assumed proportional to the stiffness matrix $\mathbf{C} = 5.0 \times 10^{-3} \mathbf{K}$. Constant external force $F = 500 \sin(5t) \text{ N}$ was assumed over the 10th mass, as shown in Figure 5.3. The numerical integration was performed assuming $t_f = 5 \text{ s}$, and a time step $\Delta t = 5 \times 10^{-3} \text{ s}$.

Figure 5.2 - Case study 1: Damped spring-mass system with 10-DOF

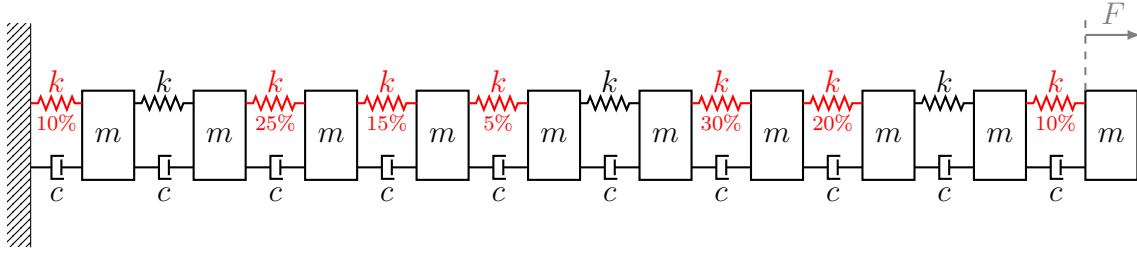
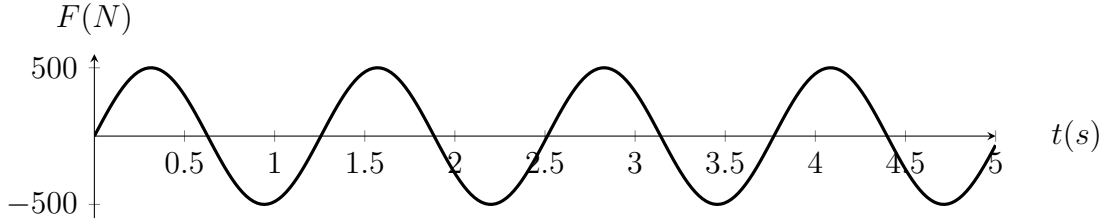


Figure 5.3 - Case study 1: Load applied on the spring-mass system



The following damage parameters configuration was assumed in the system: a reduction of 10% on the 1st spring, 25% on the 3rd, 15% on the 4th, 5% on the 5th, 30% on the 7th, 20% on the 8th, and 10% on the 10th, as represented in Figure 5.2. Red color indicates damage. The other elements have been assumed without damage.

Figure 5.4 shows the dynamic response for the displacements masses 1, 5 and 10 for the system with and without damages.

For this first experiment set, the hybrid algorithm MPCA-HJ was configured as follows: $N_{FE}^{mpca} = 10000$, $N_{FE}^{blackboard} = 1000$, and $N_{FE}^{hj} = 10000$.

Figure 5.4 - Case study 1: Dynamic responses: deflection in the y -axis vs. time from 0 to 5s of the element 1 (—), element 5 (---), and element 10 of the system with damages (.....), and element 1 (—), element 5 (---), and element 10 (.....) of the system with no damages.

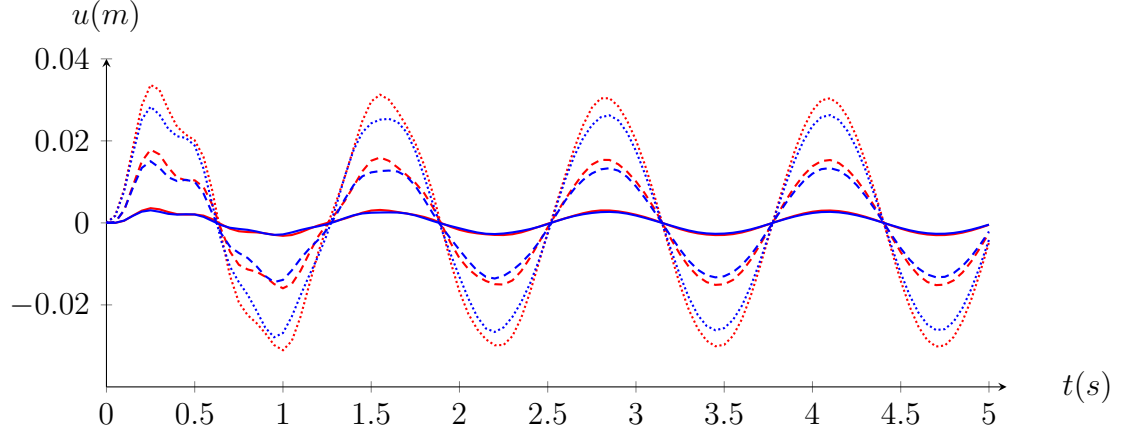
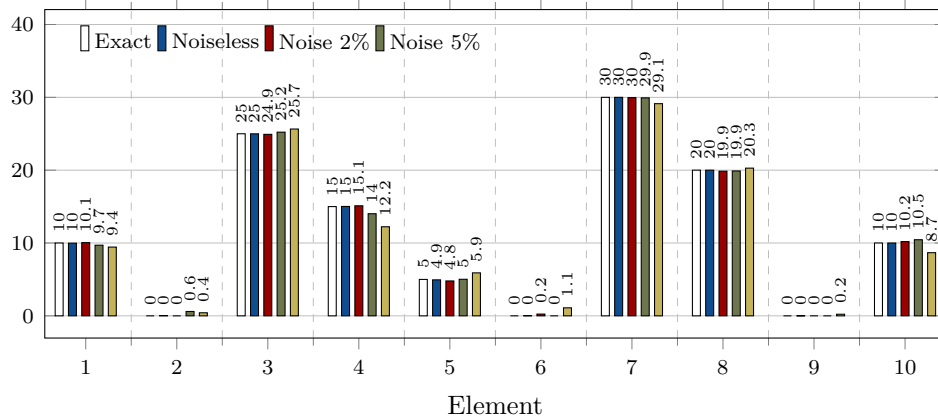


Figure 5.5 shows the results for the estimated damage parameters. The main columns, each one containing five colored bars, represent the spring elements in the system. White bars indicate the exact damage values, and blue, red and gold bars represent the estimated damages for noiseless, noise with 2% and 5%, respectively. The results demonstrate that perfect estimation of damage was obtained for the case with noiseless data. Good results, with a low relative error, were obtained using 2% and 5% noisy experimental data.

Figure 5.5 - Case study 1: Results using MPCA-HJ on different noise levels Θ^d



In the second part of this case study, the same damage configuration was used. Now, the MPCA-HJ is compared with CMPCA-HJ and RMPCA-HJ. In Figure 5.6, the average of 50 independent runs of the system is shown, using different noise levels in the data.

The damage parameter was well estimated for all cases. Using the noiseless data, results are very similar to the exact values. The time spent by the algorithms is shown in Table 5.2. The times are similar for all the cases, with a difference of few seconds.

Table 5.2 - Case study 1: Mean of the computing time (in seconds) spent by the algorithms

Noise level	MPCA-HJ	CMPCA-HJ	RMPCA-HJ
0%	16.2	19.0	15.0
2%	14.8	14.2	16.1
5%	14.7	16.1	17.1
10%	14.2	14.1	14.9

5.3 Case study 2: Damage identification in a truss structure with 12-DOF

Results on the same structure were presented in the 10th International Conference on Composite Science and Technology – (HERNANDEZ TORRES et al., 2015), and in the XXXVI Iberian Latin-American Congress on Computational Methods in Engineering (HERNÁNDEZ TORRES et al., 2015b).

The methodology is now tested on a three-bay truss structure modeled with 12 bars and 12-DOF, as shown in Figure 5.7.

Figure 5.7 - Case study 2: Three-bay truss structure

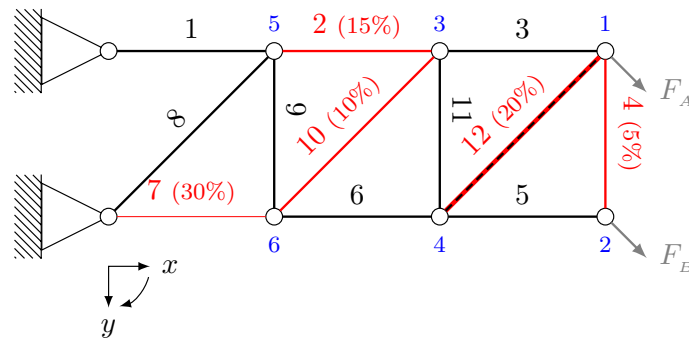
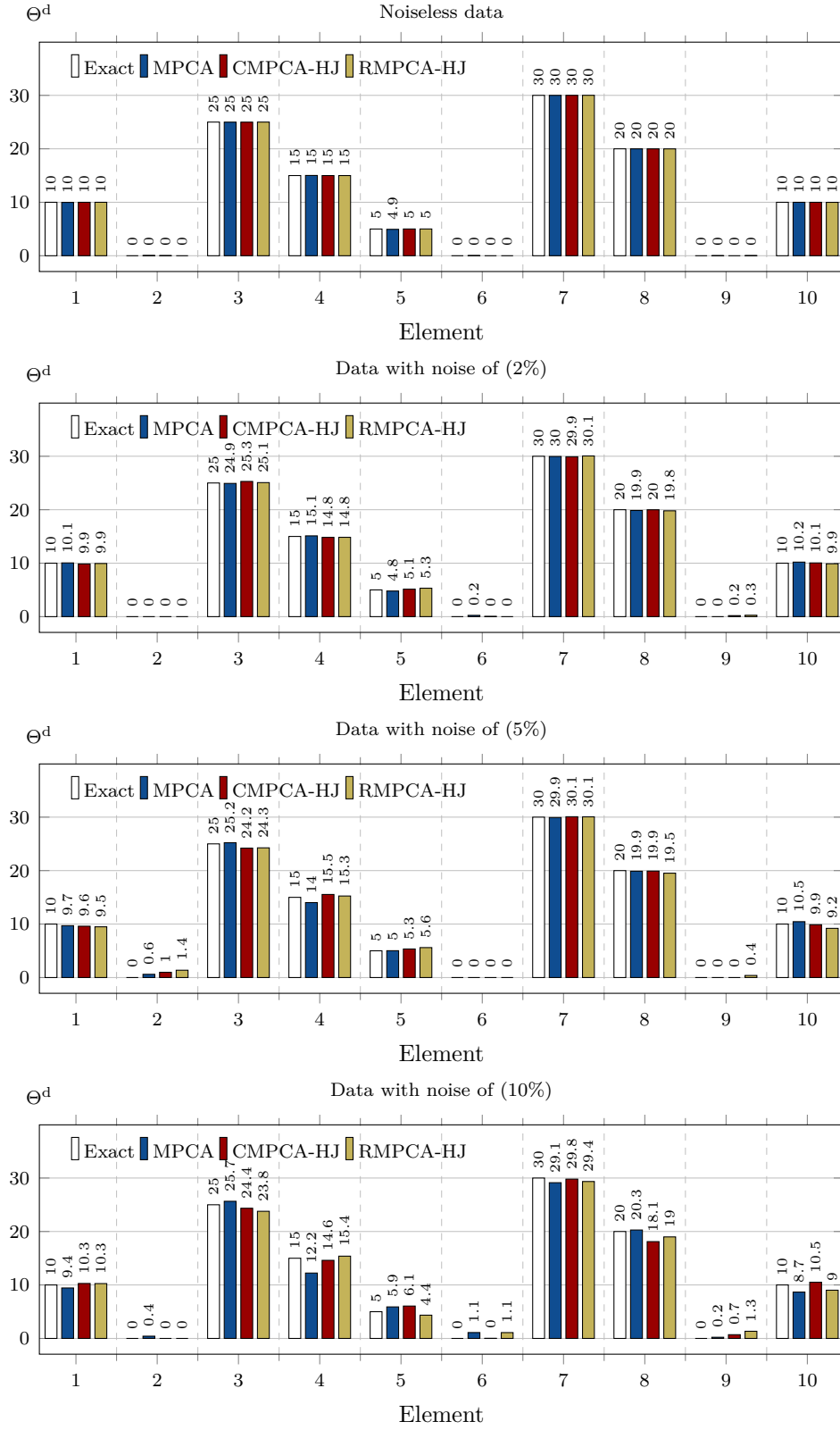


Figure 5.6 - Case study 1: Results with different configurations



Properties and dimensions of the structure are shown in Table 5.3. The damping matrix is assumed proportional to the stiffness matrix ($C = 10^{-5}K$). External forces in the positive diagonal direction and constants over the time are imposed over the nodes 1 and 2. Initial conditions for displacement and velocity are equal to zero ($u(0) = 0, \dot{u}(0) = 0$). The final time for all the numerical simulations was assumed as $t_f = 5 \times 10^{-2} s$, with a time step of $5 \times 10^{-4} s$.

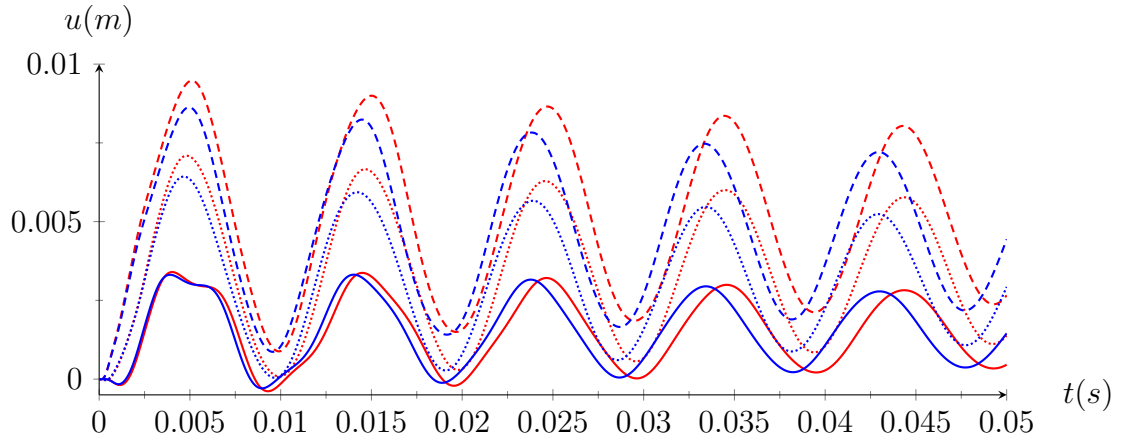
Table 5.3 - Case study 2: Material properties

Property	Value
Element type	Bars
Material	Aluminum
Youngs modulus (E)	70 GPa
Material density (ρ)	2700 kg/m ³
Square cross section area (A)	$2.5 \times 10^{-5} m^2$
Non-diagonal elements length ($l_{non-diagonal}$)	1.0 m
Diagonal elements length ($l_{diagonal}$)	1.414 m

A damage configuration of 15% over the 2nd element, 5% over the 4th, 30% over the 7th, 10% over the 10th and 20% over the 12th element was considered, as represented in Figure 5.7. All the others elements have been assumed as undamaged.

Figure 5.8 shows the dynamic for the variable ν in the first element for the first eight seconds.

Figure 5.8 - Case study 2: Dynamic responses: deflection in the y -axis vs. time from 0 to 0.05s of the node 1 (.....), node 3 (- - -), and node 5 of the system with damages (—), and node 1 (.....), node 3 (- - -), and node 5 (—) of the system with no damages.



A comparison is done between qG -HJ, MPCA-HJ, CMPCA-HJ, and RMPCA-HJ. The algorithms were configured as follows: $N_{FE}^{mpca} = 100000$, $N_{FE}^{blackboard} = 1000$, $N_{FE}^{qG} = 70000$, $N_{FE}^{hj} = 100000$.

Figure 5.9 shows the results for the estimated damage parameters in four cases: using noiseless data and noisy data with level of 2%, 5%, and 10%. White bars show the simulated damage in the elements. In these experiments, all the algorithms obtained similar results. All the damages were identified, and almost perfect damage estimation was obtained for noiseless data. In the experiments that were performed over the noisy data, the results were getting worse. For the noisy data with 5% and 10%, false positives appeared in the 9th and 11th elements.

The mean of the computing time is shown in Table 5.4. In all cases, except with noise level of 10%, the RMPCA-HJ obtained the lowest times. qG -HJ obtained computing times more than 10 times greater than those got with the variations of MPCA-HJ.

Table 5.4 - Case study 2: Mean of the computing time (in second) spent by the algorithms

Noise level	MPCA-HJ	CMPCA-HJ	RMPCA-HJ	qG -HJ
0%	29.5	27.8	28.5	245.73
2%	29.1	29.9	27.1	249.27
5%	27.4	28.2	26.4	309.27
10%	26.4	23.9	24.1	253.53

5.4 Case study 3: Damage identification in a model with a similar shape to the International Space Station

The third case studies a truss structure that has the simplified shape of the International Space Station (ISS). ISS is a space station in low Earth orbit. Figure 5.10 shows a photo of the ISS taken from the Space Shuttle Endeavour on December, 2000, after completing the mission STS-97 which installed new solar arrays.

Figure 5.9 - Case study 2: Results using MPCA-HJ, CMPCA-HJ, RMPCA-HJ, and q G-HJ

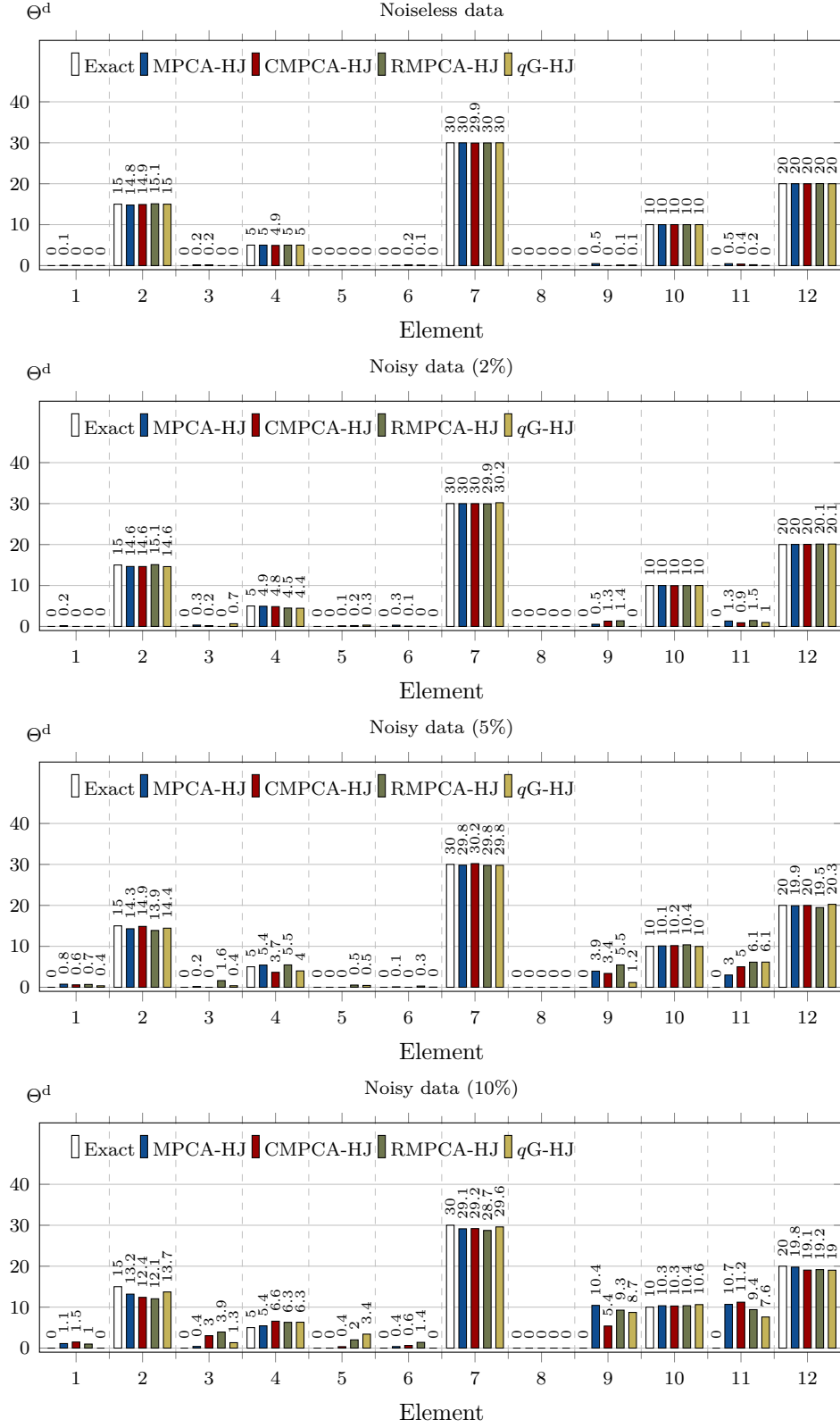


Figure 5.10 - International Space Station



SOURCE: Taken from [NASA](#) (2012)

The structure model has 72 bars, and the inferior extreme of the structure is fixed, resulting in a total of 68-DOF, as shown in [Figure 5.11](#).

Figure 5.11 - Case study 3: Truss structure with 72 bars

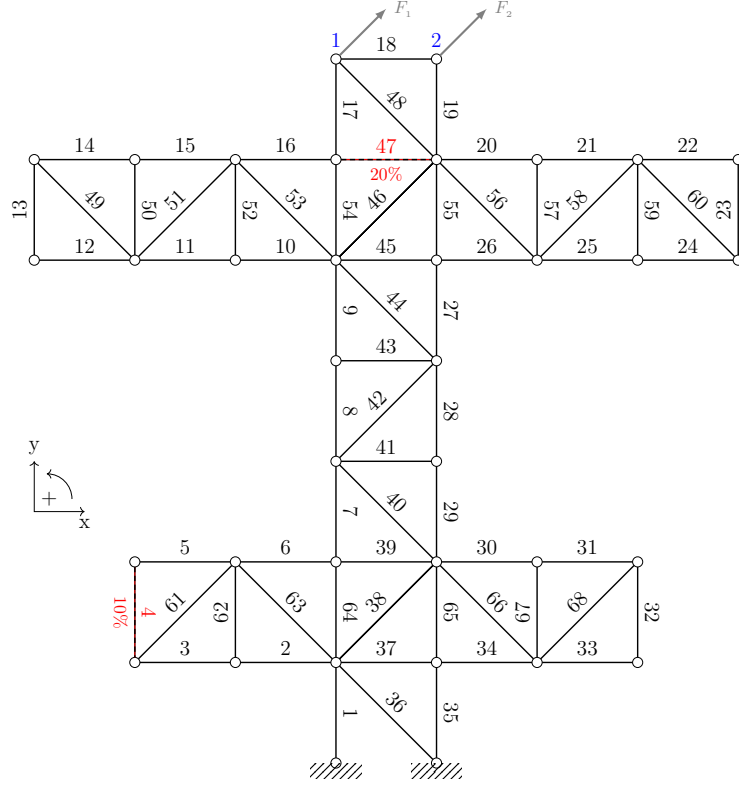
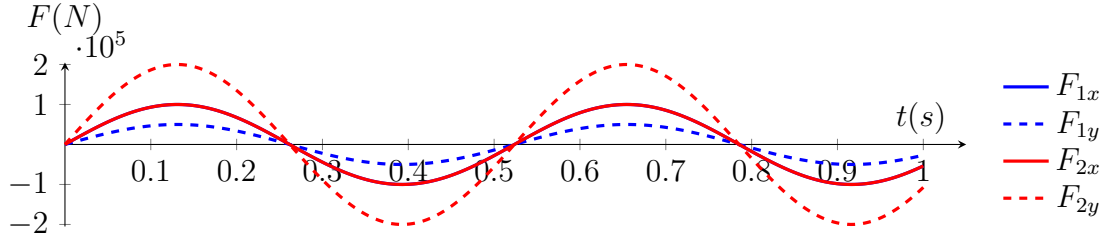


Table 5.5 presents the material properties of the structure. The damping matrix is assumed proportional to the mass matrix $\mathbf{C} = 1.7 \times 10^{-1} \mathbf{M}$. External forces are applied to the nodes 1 and 2 (in blue), in the positive diagonal direction with components $F_{1_x} = 10^5 \sin(12t) \text{ N}$, $F_{1_y} = 5 \times 10^4 \sin(12t) \text{ N}$, $F_{2_x} = 10^5 \sin(12t) \text{ N}$ and $F_{2_y} = 2 \times 10^5 \sin(12t) \text{ N}$, as shown in Figure 5.12. Initial conditions for displacement and velocity are equal to zero ($u(0) = 0, \dot{u}(0) = 0$). The time step is assumed as $\Delta t = 5 \times 10^{-2} \text{ s}$, and the final time is $t_f = 5.0 \text{ s}$ for the numerical simulations.

Table 5.5 - Case study 3: Material properties

Property	Value
Element type	Bars
Material	Aluminium
Youngs modulus (E)	70 GPa
Material density (ρ)	2700 kg/m ³
Square cross section area (A)	$8.0 \times 10^{-3} \text{ m}^2$
Non-diagonal elements length ($l_{non-diagonal}$)	6.0 m
Diagonal elements length ($l_{diagonal}$)	8.485 m

Figure 5.12 - Case study 3: Loads applied on the 8th and 13th-DOFs of the system



The maximum number of function evaluations for each hybrid approach is set to 1×10^6 , with 9×10^5 evaluations reserved for the global optimization methods. For the MPCA, the blackboard update was set each 1000 function evaluations. In these experiments, the best solution known, *i.e.* the undamaged configuration, is introduced in the initial population. One candidate solution is initialized with all values in 100%, while the others solutions are initialized randomly.

Damages in the 4th and the 47th bars were simulated. Figure 5.13 shows the dynamic response in some nodes of the structure.

All damages were well estimated, as shown in Figure 5.14. A little false positive, with value of about 5%, appeared using RMPCA-HJ in the 33th element, while for the *q*G-HJ was less than 2%. The average time spent by RMPCA-HJ was equal 896 seconds, while by *q*G-MPCA was equal to 2384 seconds.

Figure 5.13 - Case study 3: Dynamic responses: deflection vs. time from 0 to 0.05s of the DOF 33 (—), DOF 35 (---), DOF 37 (.....) and DOF 39 (-.-) of the system with damages, and DOF 33 (—), DOF 35 (---), DOF 37 (.....) and DOF 39 (-.-) of the system with no damages.

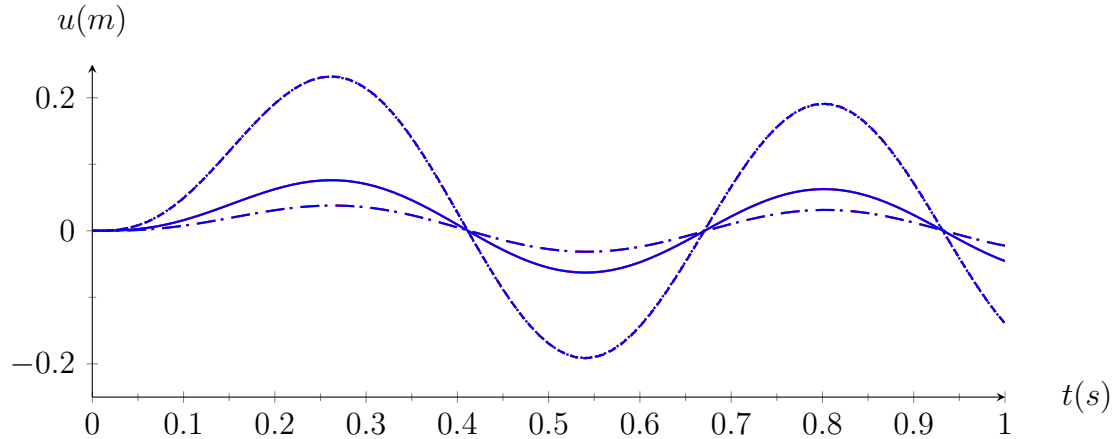
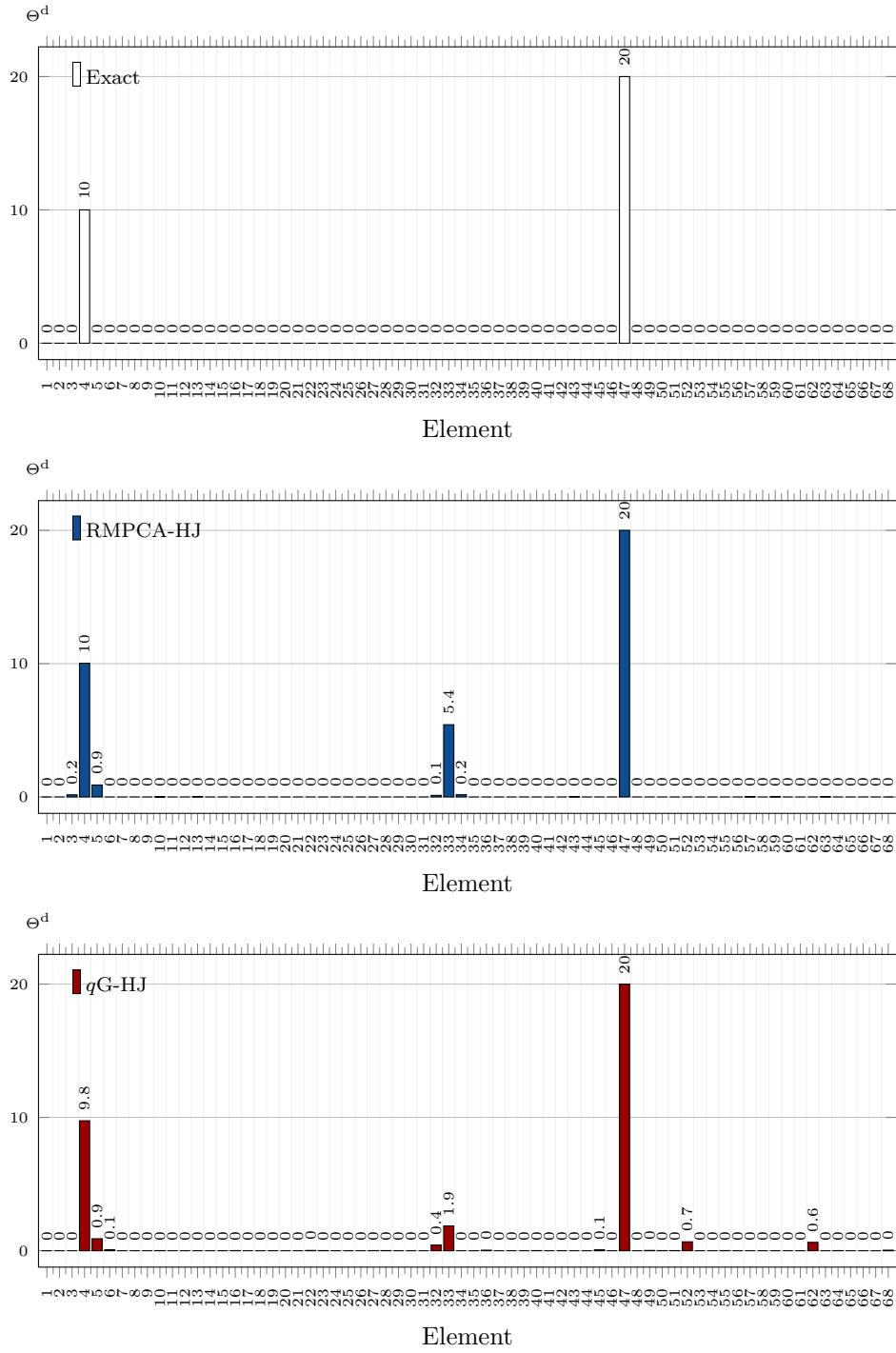


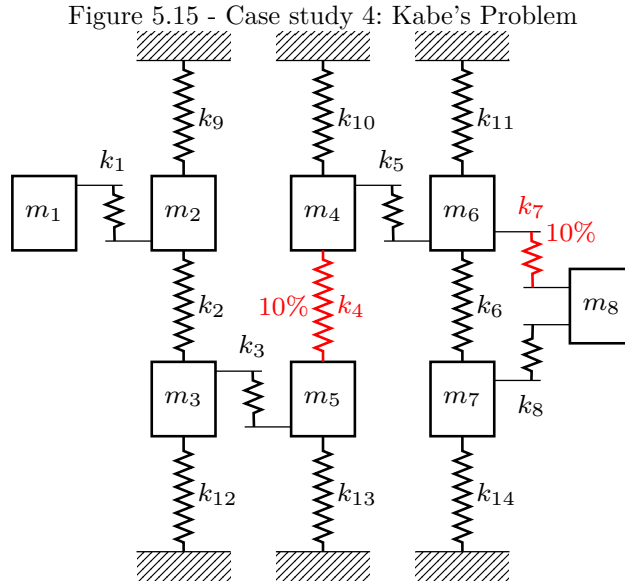
Figure 5.14 - Case study 3: Average results for 50 experiments



5.5 Case study 4: Damage identification in the Kabe's problem

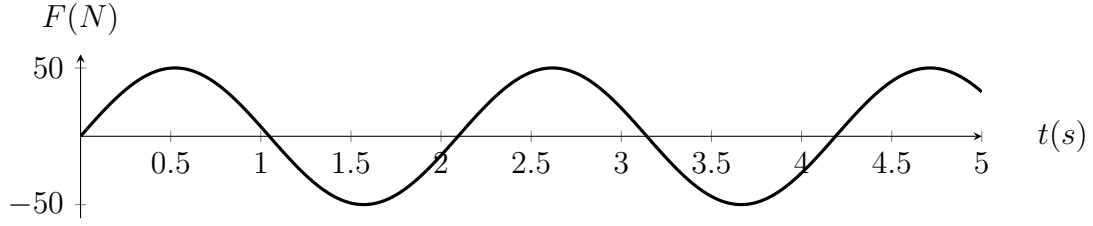
Results using the same system will be presented in the *XXXVII Congresso Nacional de Matemática Aplicada e Computacional* (CNMAC 2017) - (HERNÁNDEZ TORRES et al., 2017 in press).

In this case study, the methodology is tested on a mass-spring system called Kabe's Problem (KABE, 1985). This system has 8-DOF, and 14 springs with a distribution as shown in Figure 5.15. This problem was first used as a test using Kabe's Stiffness Matrix Adjustment (KMA) method (ATALLA, 1996). This system has some peculiarities that makes it a challenge: there are large differences between the magnitude of the stiffness of the elements (from 1.5 to 1000), and the system has a very high modal density. All eight natural frequencies have a small difference between them (ATALLA, 1996).



A load $f = 50\sin(3t)$ N is applied on the sixth DOF, as shown in Figure 5.16.

Figure 5.16 - Case study 4: Load applied on the 6th DOF of the system



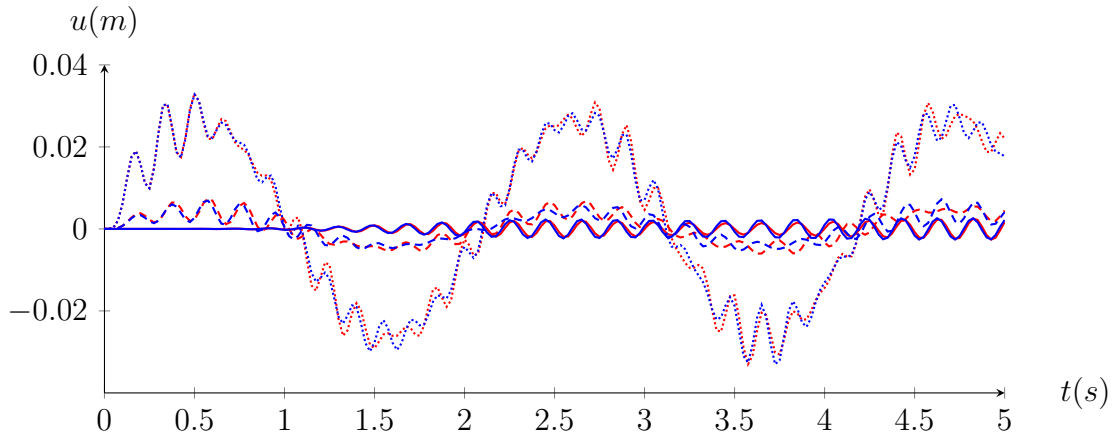
Dimensionless values for the masses and stiffness are shown in Table 5.6.

Table 5.6 - Case study 4: Dimensionless mass and stiffness

$k_{1,8}$	$k_{2,6}$	k_{3-5}	k_7	$k_{9,11,12,14}$	$k_{10,13}$	m_1	m_{2-7}	m_8
1.5	10	100	2	1000	900	0.001	1	0.002

The fourth and seventh elements were simulated as damaged, each one with 10% of stiffness reduction. Figure 5.17 shows the dynamic for the displacement in the first, fifth and eighth DOFs for the system with and without damages.

Figure 5.17 - Case study 4: Dynamic responses: deflection in the y -axis vs. time from 0 to 5s of the mass 1 (—), mass 5 (---), and mass 8 of the system with damages (.....), and mass 1 (—), mass 5 (---), and mass 8 (.....) of the system with no damages.



Experiments were performed assuming noiseless data, and noisy data with a level of 2%, 5% and 10%, all of them generated synthetically running the direct model.

The number of runs was set in 50.

Figure 5.18 shows the average of the damage parameters from a set of 50 runs. Both damages were well estimated. The estimation of the damage in the fourth damage is deteriorated as noise increases. Also, false positives appeared for the 5th and 6th springs for all algorithms.

Figure 5.19 shows the boxplot resuming the results obtained from the noiseless and noisy experimental data. The values of the standard deviation is very high for the 5th and the 6th springs, indicated by the size of the boxplots, being less for the 12th spring.

Table 5.7 shows the mean of the time spent by the algorithms. Times are very similar for all algorithms, but RMPCA-HJ is faster than the others.

Table 5.7 - Case study 4: Mean of the computing time (in seconds) spent by the algorithms

Noise level	MPCA-HJ	CMPCA-HJ	RMPCA-HJ
0%	19.7	18.9	17.2
2%	18.3	16.8	16.9
5%	17.6	18.7	17.7
10%	17.9	17.5	16.2

Figure 5.18 - Case study 4: Results using MPCA-HJ, CMPCA-HJ, RMPCA-HJ and qG-HJ - Mean for 50 experiments assuming noiseless and noisy data

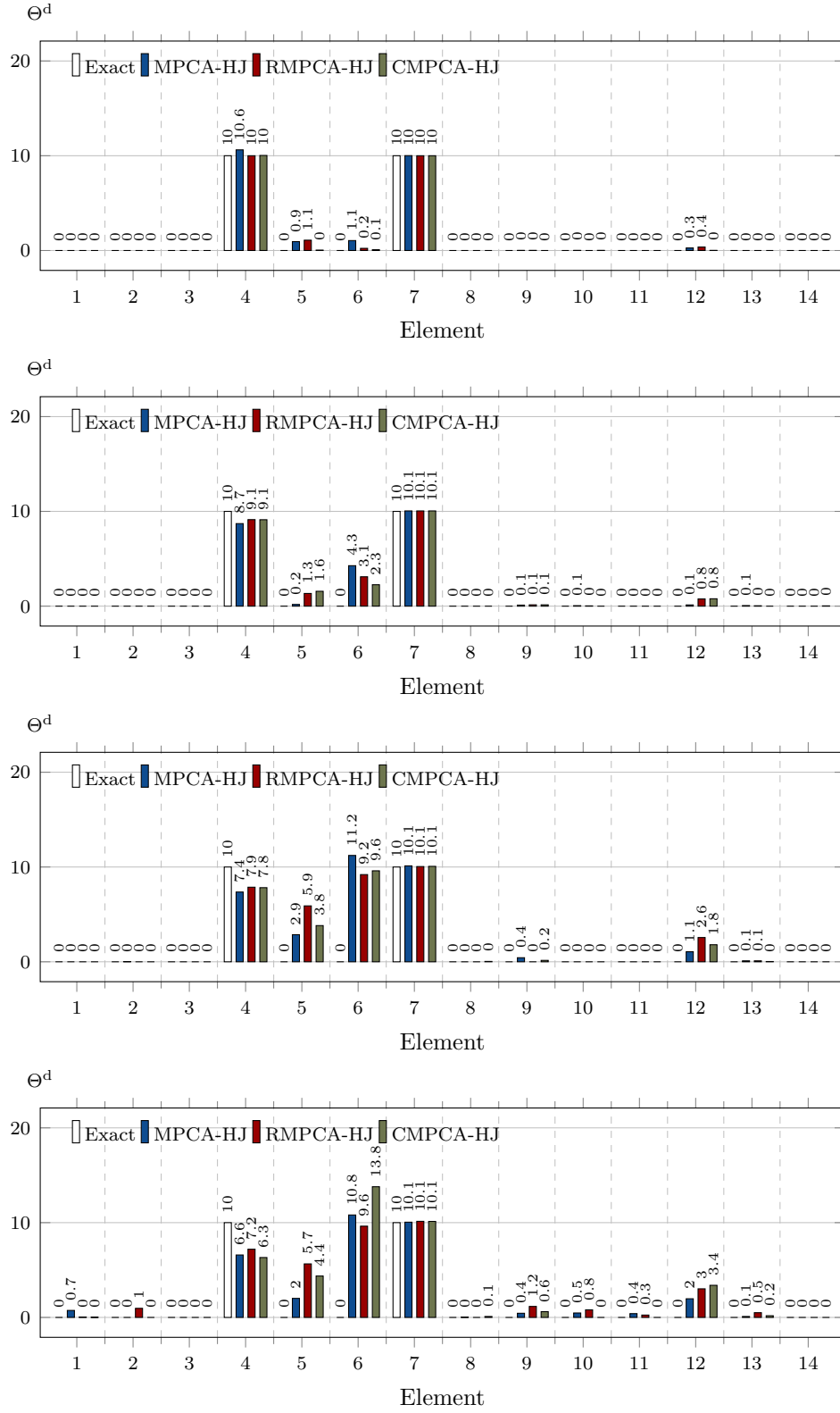
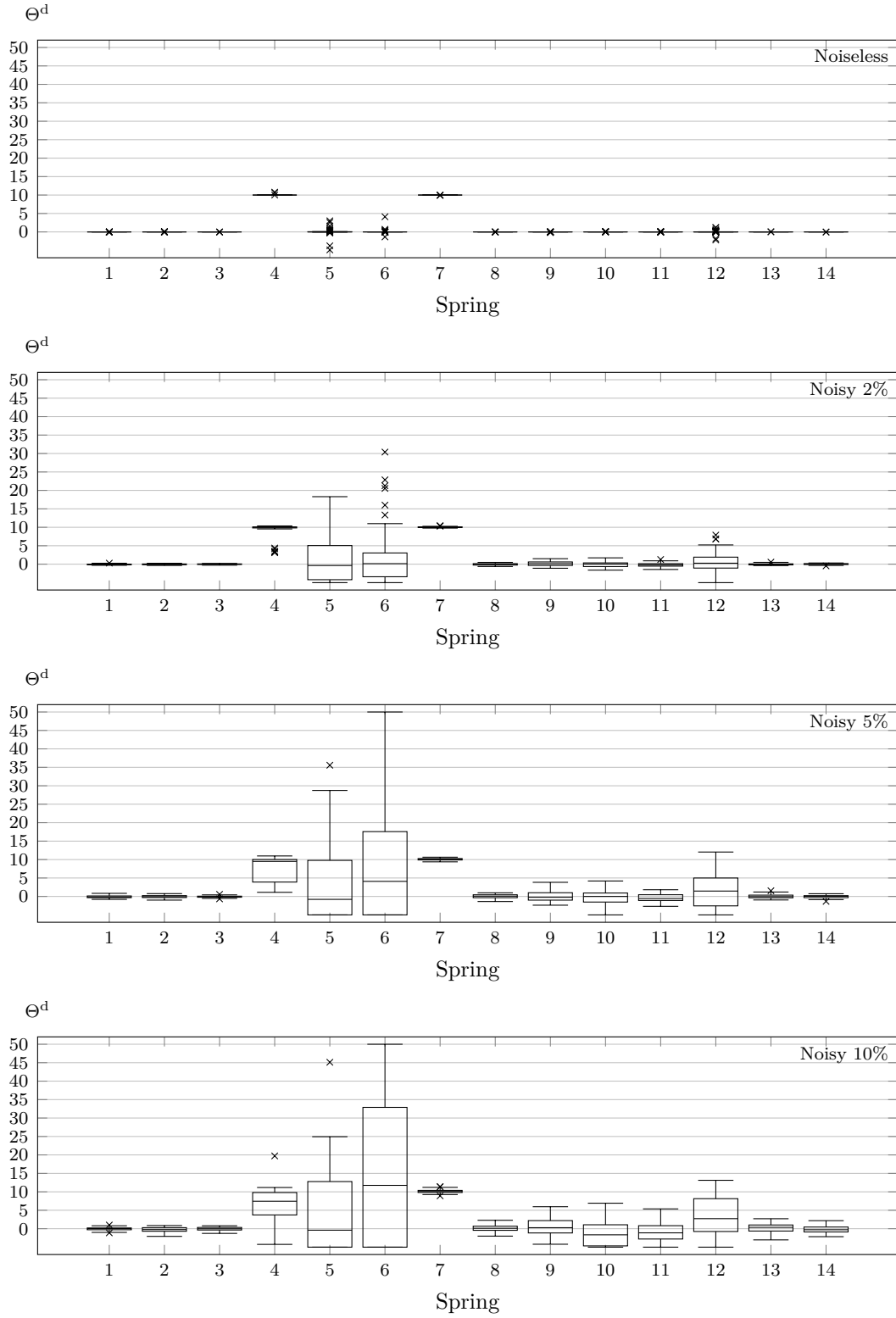


Figure 5.19 - Case study 4: Boxplot for 50 experiments on noisy data using RMPCA-HJ. Outliers are represented with the \times symbol.



5.6 Case study 5: Damage identification in the Yang's problem

The method is also tested over a 15-DOF damped discrete system called Yang's Problem (ATALLA; INMAN, 1998 apud YANG; BROWN, 1996), shown in Figure 5.20. The system has repeated modes and a high modal density, becoming a complex system. Table 5.8 shows the nominal parameters for the model, taken from Atalla and Inman (1998) and converted to the International System of Units.

In the system, loads were applied on the 8th, and the 13th masses, as shown in Figure 5.21, and described mathematically in Equation (5.2).

$$F_8 = 100 \sin(4t), F_{13} = 50 \sin(3t) \quad (5.2)$$

Figure 5.20 - Case study 5: Yang's Problem

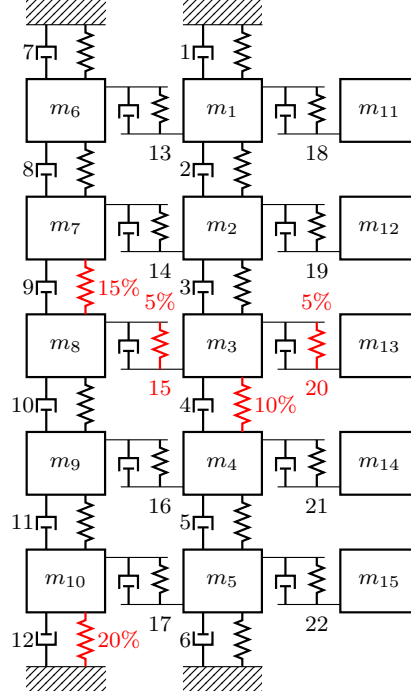
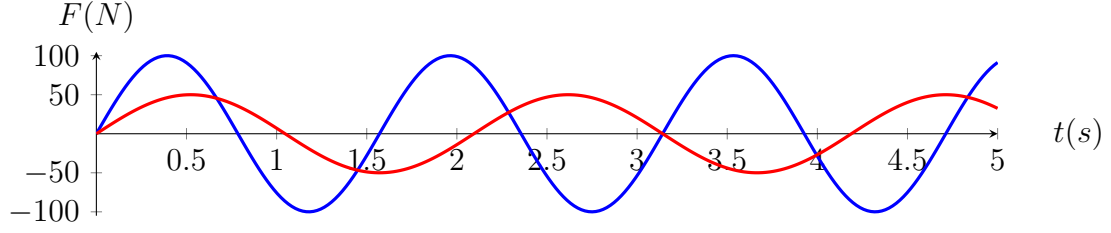


Table 5.8 - Case study 5: Parameters of the system. Masses in *kg*, stiffness in *kg/m* and damping in *kg s/m*.

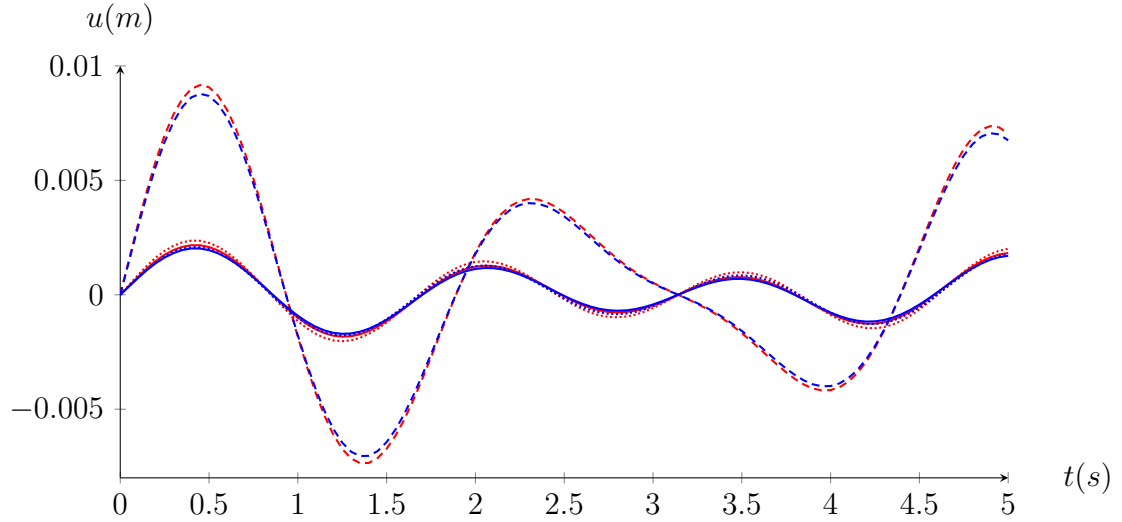
k_{1-18}	k_{19}	k_{20}	k_{21}	k_{22}	m_{1-10}	m_{11-15}	c_{1-12}	c_{13-22}
175127	192639	157614	210152	140101	0.01175	0.00059	17.5127	1.57127

Figure 5.21 - Case study 5: Loads applied on the 8th (—) and 13th (—) DOFs of the system



Damages were simulated in the 4th element, with 10% of stiffness reduction, in the 9th spring, with 15%, in the 12th spring, with 20%, and in the 15th and 20th springs, with 5% each one. Figure 5.22 shows the dynamic for the displacement in the 10th, 13th and 15th masses for the system with and without damages.

Figure 5.22 - Case study 5: Dynamic responses: deflection in the y -axis vs. time from 0 to 5s of the element 15 (—), element 13 (---), and element 10 of the system with damages (.....), and element 15 (—), element 13 (---), and element 10 (.....) of the system with no damages.



The configuration of the hybrid algorithm is the same that was set in Section 5.5.

Figure 5.23 shows the mean of the damage parameters for 50 runs using noiseless synthetic data. In this case, the algorithm estimated all the damages in the system, and appeared some false-positives on the springs 18, 21 and 22. In this case, RMPCA-HJ obtained the best results.

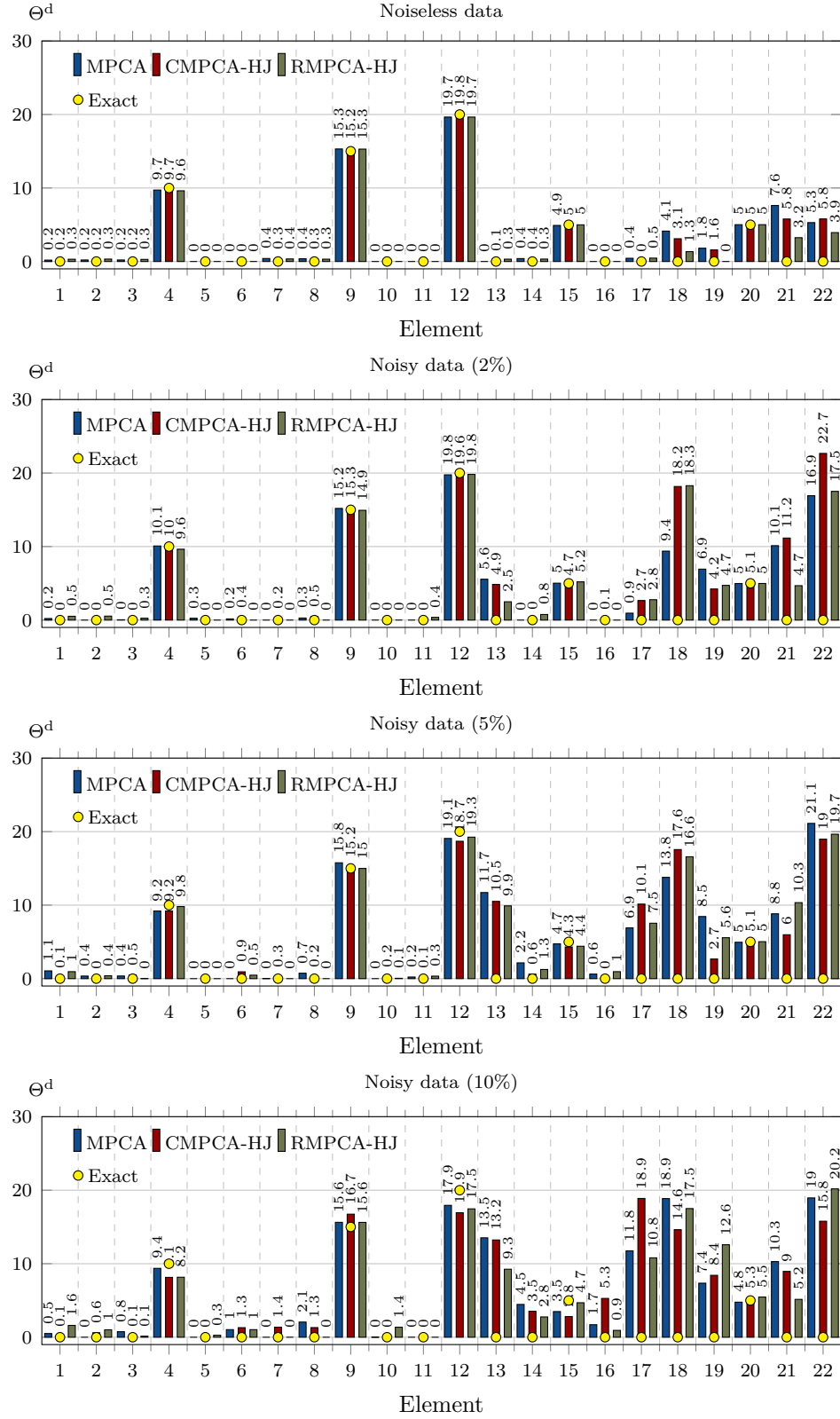
Also, in Figure 5.23 are shown the results with noisy synthetic data. The method detected false damages in all the springs that join the *free* salient masses with the rest of the system (springs from 18 through 22). More false-positives appeared in the spring 13 and 17. As noise increases, the values of the false-positives increase considerably. With a noise level of 10% appeared a false negative for the damage located in the 15th spring, except with RMPCA-HJ.

Table 5.9 shows the mean of the computing time for the 50 experiments with each algorithm. RMPCA-HJ got the best times in the experiments with noiseless data and with noise level of 2%, while CMPCA-HJ was faster in the experiments with noise level of 5% and 10%.

Table 5.9 - Case study 5: Mean of the computing time (in seconds) spent by the algorithms

Noise level	MPCA-HJ	CMPCA-HJ	RMPCA-HJ
0%	82.7	67.7	54.5
2%	60.8	65.0	59.8
5%	54.1	50.9	60.9
10%	49.5	40.6	43.7

Figure 5.23 - Case study 5: Results using MPCA-HJ, CMPCA-HJ and RMPCA-HJ - Mean for 50 experiments on noiseless and noisy data



5.7 Case study 6: Damage identification in a cantilevered beam

The results in this section will be published in the book Computational Intelligence in Engineering Problems (HERNÁNDEZ TORRES et al., 2017 forthcoming).

The cantilevered beam shown in Figure 5.24 is modeled with ten beam finite elements. It is clamped at the left end, and each aluminum beam element, with $\rho = 2700 \text{ kg/m}^3$ and $E = 70 \text{ GPa}$, has a constant rectangular cross section area with $b = 15 \times 10^{-3} \text{ m}$ and $h = 6 \times 10^{-3} \text{ m}$, a total length $l = 0.43 \text{ m}$, and an inertial moment $I = 3.375 \times 10^{-11} \text{ m}^4$. The damping matrix is assumed proportional to the undamaged stiffness matrix $\mathbf{C} = 10^{-3} \mathbf{K}$. An external varying force $F(t) = 5.0 \sin(3t) \text{ N}$ is applied to the tenth element, in the free extreme of the beam, as shown in Figure 5.26.

Figure 5.24 - Case study 6: Cantilever Beam structure

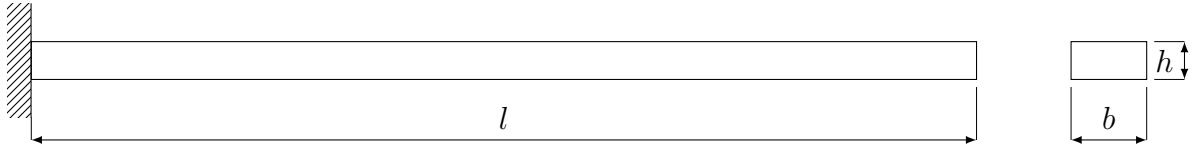


Figure 5.25 - Case study 6: Beam model with 20-DOF

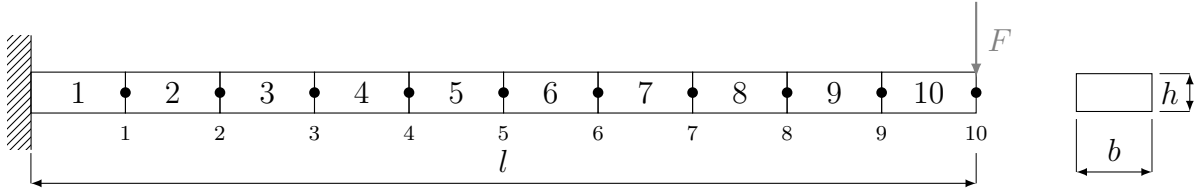
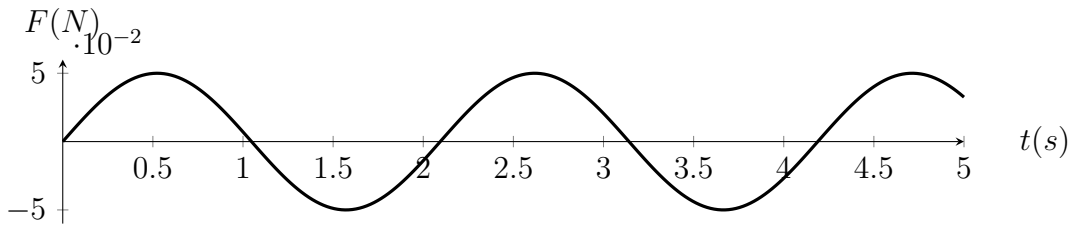


Figure 5.26 - Case study 6: Load applied on the cantilever beam



Initial conditions for displacement and velocity in all degrees of freedom are equal

to zero ($u(0) = 0, \dot{u}(0) = 0$).

For the experiments, the numerical simulation was performed assuming $t_f = 5$ s, with a time step $\Delta t = 4 \times 10^{-3}$ s.

Synthetic data were taken from the nodes of the structure, by simulating the forward model of the beam structure.

The algorithms were configured as follows: the number of particles $N_{\text{particles}} = 20$, $N_{FE}^{mpca} = 200000$, $N_{FE}^{blackboard} = 10000$, and $N_{FE}^{hj} = 100000$.

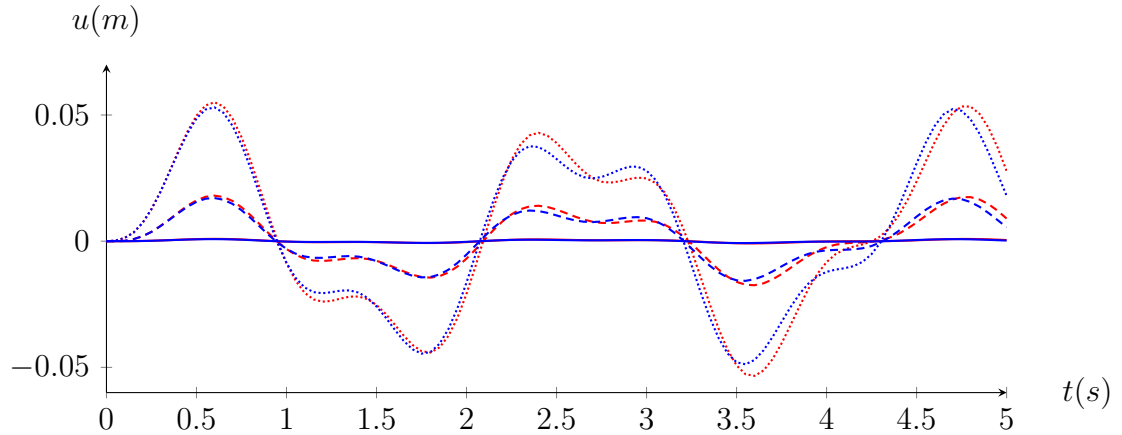
5.7.1 Results using a full dataset, with a single damage

In the subcase 1, a single damage was simulated on the first element of the structure, reducing its initial stiffness in 10%. The others elements remain without changes.

Observed data were taken from all nodes, with a total of 20 time-series with 500 points in each one.

Figure 5.27 shows the dynamic response for the displacement in the nodes 1, 5 and 10 of the system with and without damages.

Figure 5.27 - Case study 6: Dynamic responses: deflection in the y -axis vs. time from 0 to 5s of the node 1 (—), node 5 (---), and node 10 (·····) of the structure with damages (·····), and node 1 (—), node 5 (---), and node 10 (·····) of the structure with no damages.



Four cases are tested: with noiseless data, and noisy data with 2%, 5% and 10%, respectively. For the analysis in each case, the mean of the results of 50 experiments was calculated.

Figure 5.28 shows results for the identification process. The damage was well estimated in all experiments. No false-positives appeared using noiseless data. In the experiments with noisy data, all algorithms obtained a false-positive in the 10th element, being higher as the noise is increased.

5.7.2 Results using a full dataset, with a single damage, applying regularization

The maximum entropy regularization operator was used for eliminating the false-positive in this case. Figure 5.29 shows a comparison of the mean of 50 experiments using MPCA-HJ with and without regularization, in the cases with noisy data with a level of 2% and 5%, respectively. The regularization parameter was calculated using the Morozov's discrepancy principle, with $\alpha_{\Omega}^{2\%} = 0.02 \times 20 = 0.4$ and $\alpha_{\Omega}^{5\%} = 0.05 \times 20 = 1$. As shown, the false-positive is totally eliminated in both cases.

5.7.3 Results using an incomplete dataset, with single damage

Observed data were taken from some degrees of freedom: displacements from node 2 and node 10, and rotation from node 5 and node 10. In total, in this case, there are four time-series with 500 points in each one. For the analysis in each case, the mean of 50 runs of the inverse solution was calculated.

Similar to the experiments performed with a full dataset in Section 5.7.1, a single damage of 10% was simulated on the first element, maintaining the other elements undamaged.

Figure 5.30 shows the means of the estimated damage parameters. For the noiseless data, the damage was well identified and no false-positives appeared.

For the experiments using 2% and the 5% noisy experimental data, the algorithms launched false-positives for the 2nd and the 10th elements.

For the experiments using 5% noisy experimental data, it is noticeable that MPCA-HJ and CMPCA-HJ estimated the damage at the 1st element with the half the its exact value.

For the experiments using 10% noisy, there is a false-negative obtained by the MPCA-HJ. All algorithms detected false-positives for the 10th element. MPCA also got false-positives in the 5th and the 6th elements. CMPCA-HJ and RMPCA-HJ detect a false-positive in the 9th element.

Figure 5.28 - Case study 6: Results for the MPCA-HJ, CMPCA-HJ and RMPCA-HJ using a full dataset - Single damage in the fixed element

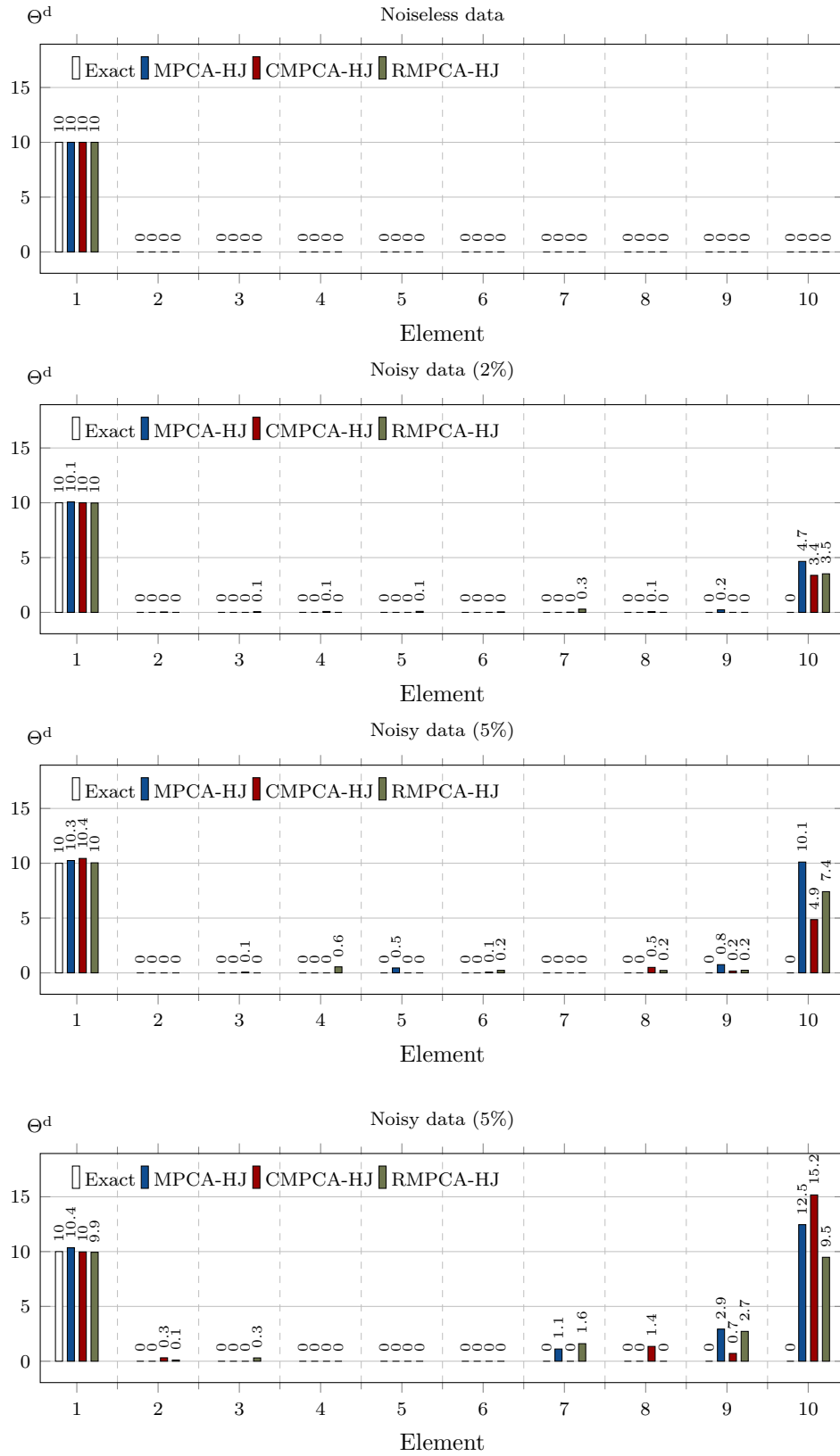
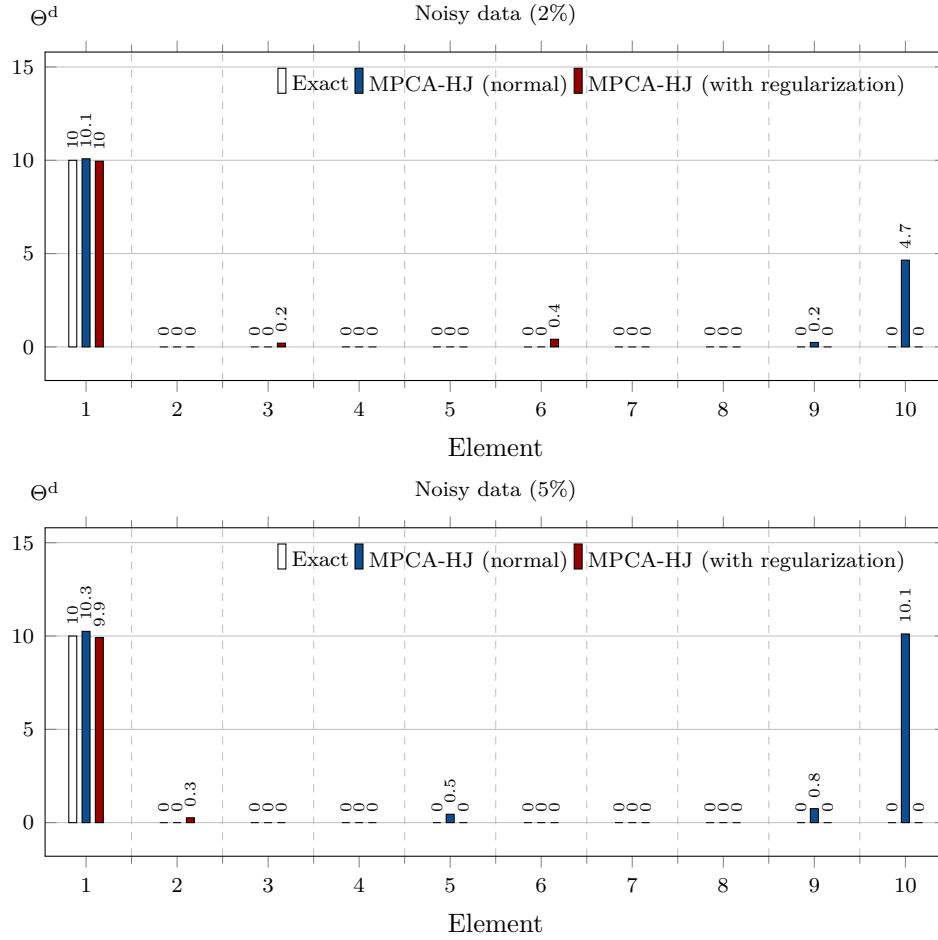


Figure 5.29 - Case study 6: Results for the MPCA-HJ using entropy regularization and a full dataset - Single damage in the fixed element



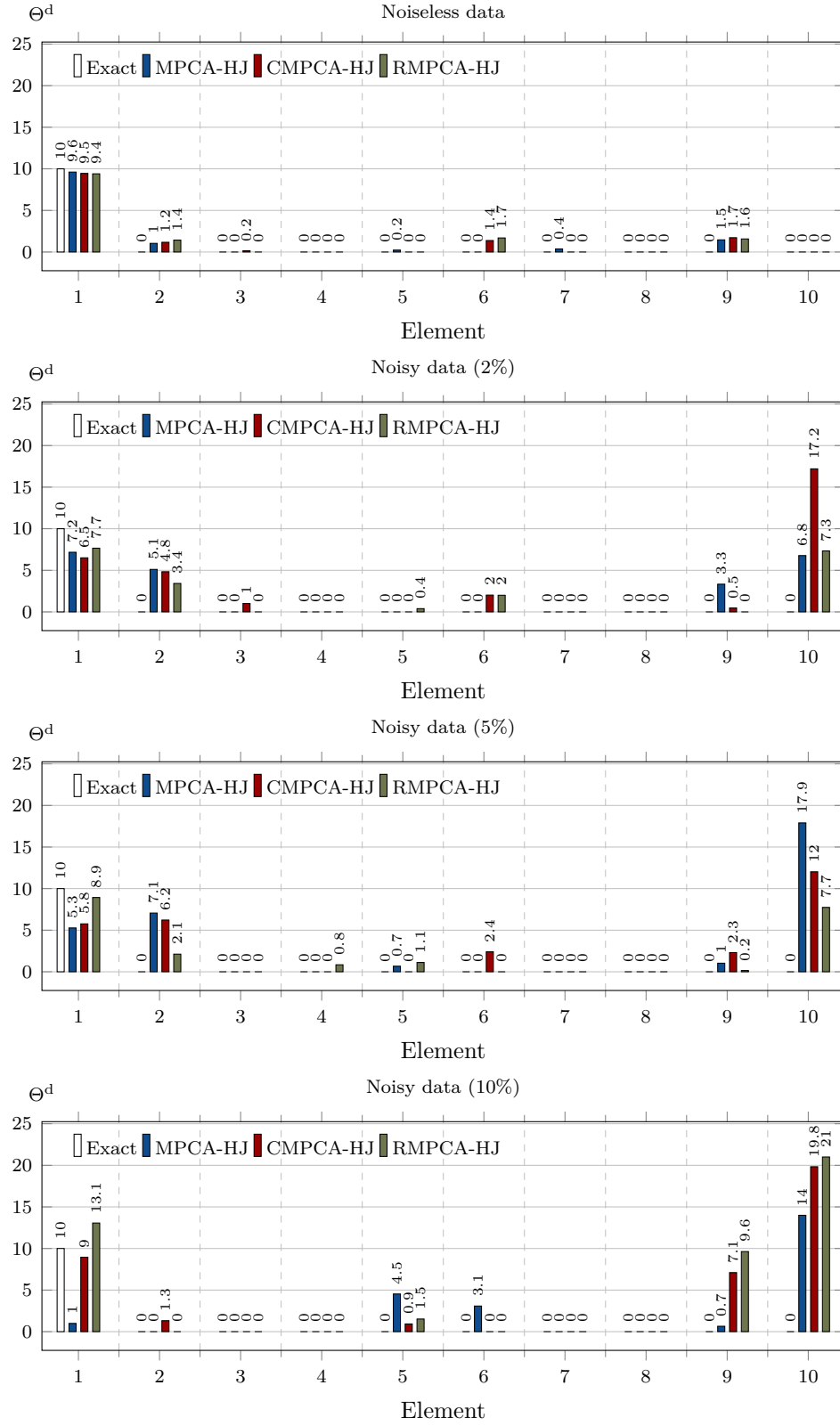
It is important to highlight the increasing level of variation for the results, caused by noise.

5.7.4 Results using a full dataset, with mixed multi-damage

In the subcase 2, a damage configuration of 10% stiffness reduction on the 2nd element, 20% on the 4th, 30% on the 6th, 5% on the 9th element, and 10% on the 10th element was set. The remaining elements are assumed as undamaged. Figure 5.31 shows the results of the mean of 50 experiments for each algorithm. The results were almost perfect in the case with noiseless data.

In the case using noisy data with a level of 2%, a false-negative appeared in the 9th element, for the CMPCA-HJ and the RMPCA-HJ, while in the 10th element the CMPCA-HJ got a value almost the half of the exact values, while RMPCA-HJ

Figure 5.30 - Case study 6: Results for the MPCA-HJ, CMPCA-HJ and RMPCA-HJ using an incomplete dataset - Single damage in the fixed element



over-estimated the value.

In the case with noisy data with levels of 5% and 10%, the effect of the damage in the 9th element is overshadowed by the noise, and some algorithms cannot estimate it correctly. The rest of the damages are well located. For the RMPCA-HJ, a false-positive appeared in the 7th element.

Table 5.10 shows the mean of the computing time for the 50 experiments with each algorithm. MPCA-HJ got the best times in the experiments with noiseless data and with noise level of 2% and 5%, while RMPCA-HJ was faster in the experiments with noise level of 10%.

Table 5.10 - Case study 6: Mean of the computing time (in seconds) spent by the algorithms

Noise level	MPCA-HJ	CMPCA-HJ	RMPCA-HJ
0%	25.1	29.1	26.8
2%	26.3	31.1	27.7
5%	26.6	29.8	28.3
10%	28.3	29.9	26.8

5.7.5 Results using an incomplete dataset, with mixed multi-damage

In this experiment, the same damage configuration of Section 5.7.4 was used. The observed data present the same characteristics than those used in Section 5.7.3.

Figure 5.32 shows the mean of 50 experiments of damage identification for each algorithm.

When assuming noiseless experimental data, false-negatives appeared in the 9th and the 10th elements. The damages in the second and the fourth elements are estimated with a difference of more than 5%. False-positives appeared in the 3rd and the 8th elements. In the 6th and 9th elements, the damage values were well estimated, with a small error. The estimated damage in the 2nd, 4th, and 10th have an error of about 5% around the real value. In the 3rd element appeared a false alarm with a damage of about 5%.

For the experiments using 2% noisy experimental data, the damage at the 2th and the 9th elements were not well identified, being confused with those errors that appeared caused by the presence of the noise. The damage in the 4th element was detected, but with an error greater than 5%. On the other hand, the damages at

Figure 5.31 - Case study 6: Results for MPCA-HJ, CMPCA-HJ and RMPCA-HJ using a full dataset - Configuration with mixed damages

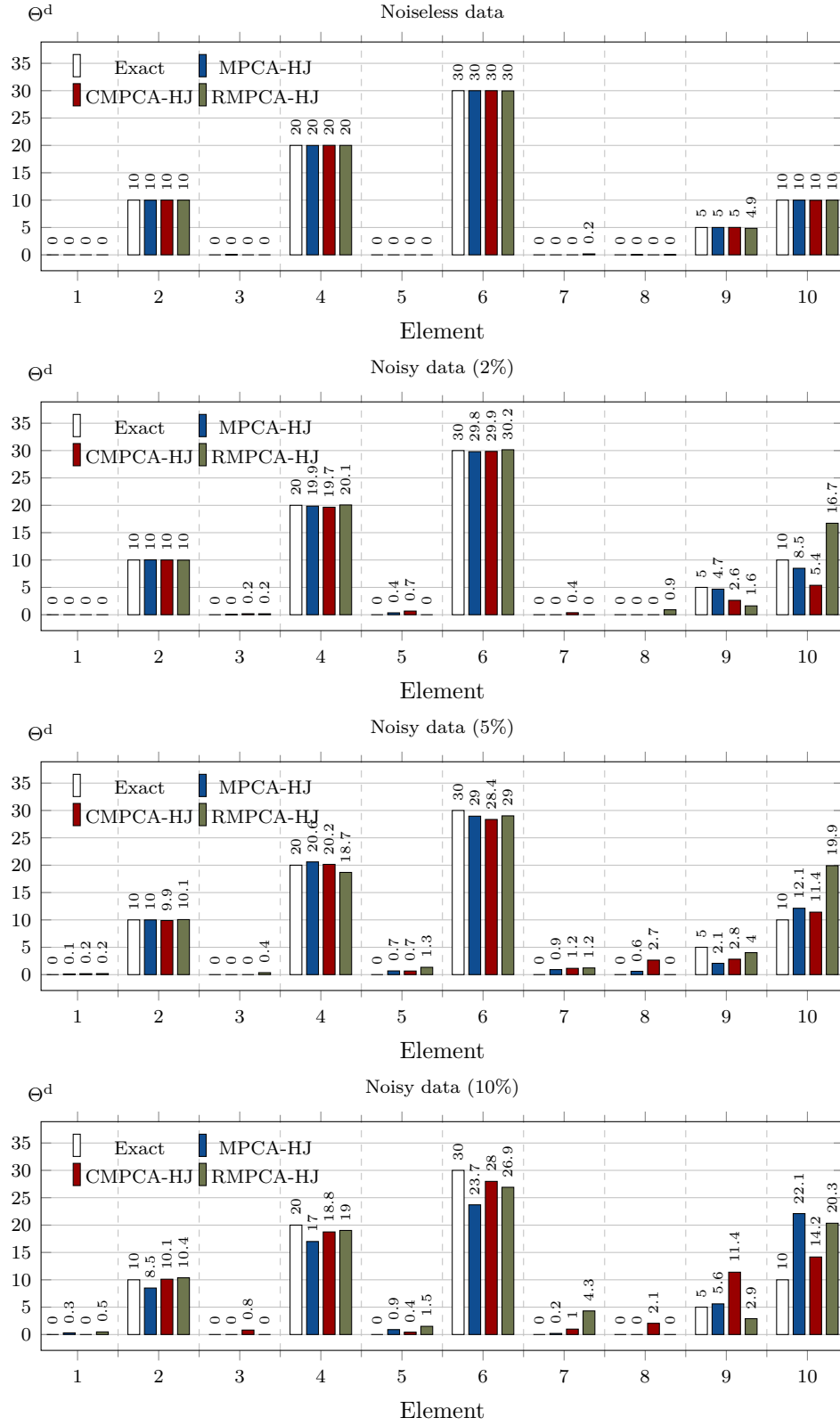
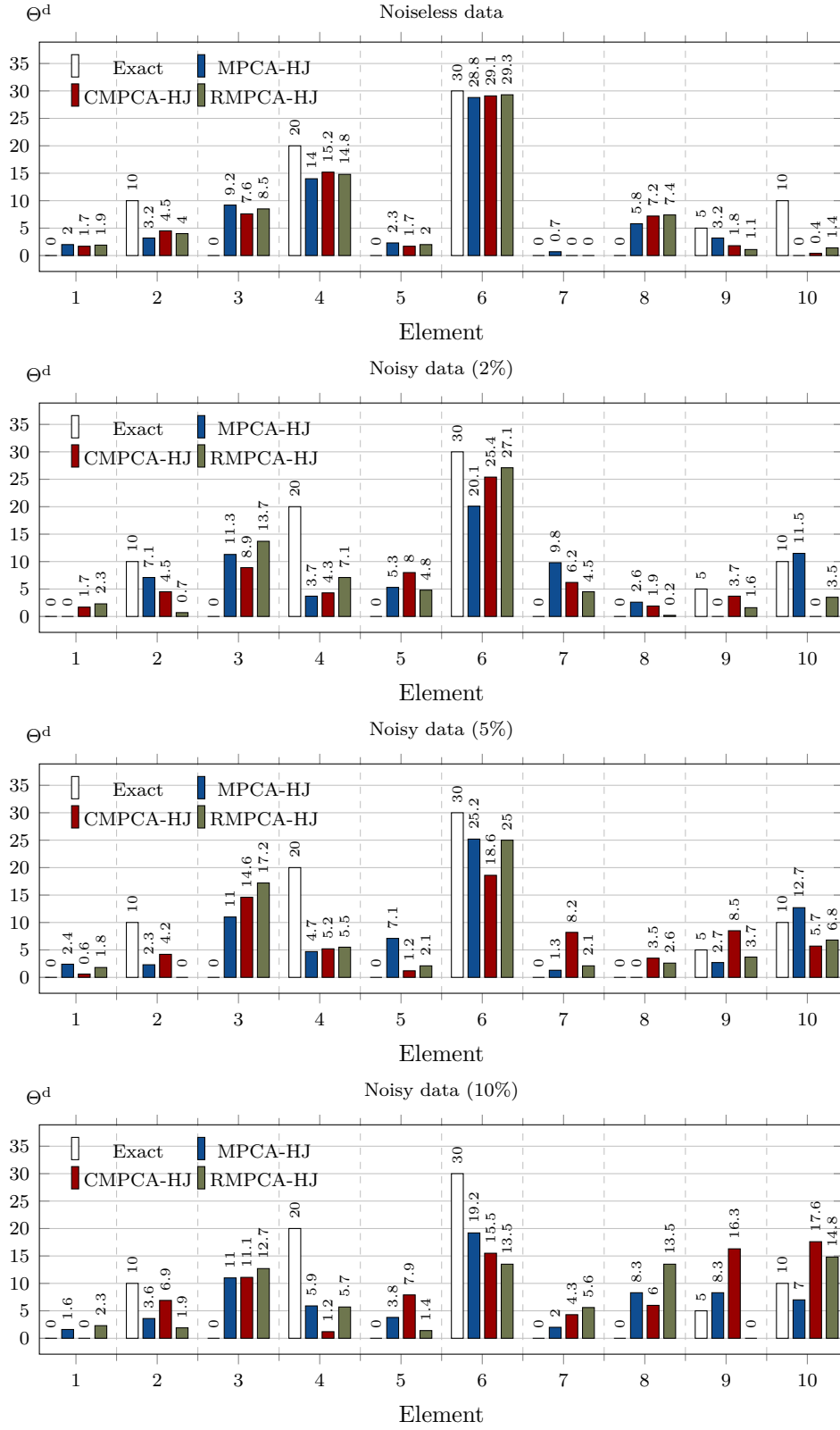


Figure 5.32 - Case study 6: Results for the MPCA-HJ and RMPCA-HJ using an incomplete dataset
- Configuration with mixed damages



the 6th and 10th were detected, and the values have an error of less than 5%. Again, a false alarm appeared at the 3rd element.

For the experiments using 5% noisy experimental data, the results are equivalent to those achieved with noise level of 2%. The damage at the 9th element was overshadowed by the effects of the noise. In the 3rd element, which is undamaged, damage equal to the estimated in the 4th element was estimated, which has a damage of 20%. Also, false alarms appeared for the 5th and the 7th elements.

5.8 Chapter conclusions

In this chapter, the methodology of SDI defined in the last chapters was applied on some benchmark systems and structures. A comparison of different hybrid approaches was done, over cases without noise, and with different levels of noise. A summary of the false-positives and false-negatives in all cases is shown in [Table 5.11](#). RMPCA-HJ obtained less false-positives than the other algorithms, and tied with MPCA-HJ in false-negatives.

Table 5.11 - Summary of the false-positives (FP) and false-negatives (FN) obtained by the algorithms in the case studies

Structure	Noise	MPCA-HJ		CMPCA-HJ		RMPCA-HJ		q G-HJ	
		FP	FN	FP	FN	FP	FN	FP	FN
Spring-mass Truss	All	0	0	0	0	0	0		
	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0
	5	2	0	2	0	2	0	1	0
ISS Kabe	10	2	0	3	0	3	0	3	0
	0					1	0	0	0
	0	0	0	0	0	0	0		
	2	1	0	1	0	0	0		
Yang	5	2	0	2	0	2	0		
	10	1	0	3	0	3	0		
	0	3	0	3	0	2	0		
	2	5	0	5	0	4	0		
Cantilever 1	5	6	0	6	0	6	0		
	10	7	0	8	1	6	0		
	0	0	0	0	0	0	0		
	2	1	0	1	0	1	0		
Cantilever 2	5	1	0	1	0	1	0		
	10	2	0	1	0	2	0		
	0	0	0	0	0	0	0		
	2	3	0	2	0	2	0		
Cantilever 3	5	2	0	2	0	1	0		
	10	2	1	2	0	2	0		
	0	0	0	0	1	0	1		
	2	0	1	0	1	0	0		
Cantilever 4	5	0	1	0	1	0	0		
	10	0	0	0	0	1	1		
	0	1	2	1	2	1	2		
	2	3	2	3	1	3	2		
Total	5	2	2	3	0	1	1		
	10	3	0	4	1	3	2		
Total		49	9	53	8	47	9	4	0

For the damped spring-mass system with 10-DOF, a configuration of seven damages was first used to test the MPCA-HJ, with different noise levels in the synthetic experimental data. Then, a configuration of three damages was tested to compare the performance of the MPCA-HJ and two variants, CMPCA-HJ and RMPCA-HJ.

In the second case, the damage identification was done on a truss with 12-DOFs, with five damaged elements. Results obtained from MPCA-HJ, CMPCA-HJ, RMPCA-HJ, and the q G-HJ were compared. The best results were obtained by RMPCA-HJ. q G-HJ was slower than the others.

In the third case, the methodology is applied on a structure similar to the ISS, with 68-DOFs. Two elements were simulated as damaged on the structure. The RMPCA-HJ and q G-HJ were compared in this case study. RMPCA-HJ obtained less false positives than the other method. q G-HJ was much slower than the others.

The methodology was also tested on the Kabe's problem, with 8-DOF and 14 elements. Two damages were simulated. For all noise levels, actual damages were identified. In experiments with noiseless data, the identification was successful, but not perfect. With noisy data, as noise increases, more false positives appeared in the identification.

A complex system, called Yang's problem was also tested assuming damages in five elements. The results for the experiments with noiseless synthetic data were almost perfect. In the noisy cases, damage over 10% were identified, but a false negative appeared for the 15th element for the experiments assuming noise level of 5% and 10%. Also, false positive appeared for springs numbered from 17 to 22, being worst as noise increases.

The sixth case simulates two different damage configuration in a cantilevered beam: a single damage located close to the fixed end, and mixed damages distributed in five elements. Same experiments were done with both full set of data, and incomplete set of data. A comparison was done between results with MPCA-HJ, CMPCA-HJ and RMPCA-HJ. All damages were identified, and almost perfect results were obtained for the single damage configuration. A false positive appeared in the 10th element. This false positive was eliminated applying maximum entropy regularization in the solution of the problem. For the multi-damage configuration all damages were well located, and no false positive appeared with the full dataset. For noisy data, the methods failed to estimate the damages in the 2th and, in some cases in the 9th and 10th elements.

In the next chapter, the same methodology will be applied on structures modeled in NASTRAN.

6 VIBRATION-BASED DAMAGE IDENTIFICATION ON SYSTEMS/STRUCTURES MODELED WITH NASTRAN, AND USING *IN SILICO* EXPERIMENTAL DATA

In theory, theory and practice are the same.

In practice, theyre not.

YOGI BERRA

In the last chapter, the defined methodology was applied in the damage identification of structures on models implemented in FORTRAN.

In this chapter, the experimental design will be defined for damage identification of structures with models in NASTRAN. Next, some case studies will be presented: a 2-DOF model, a 10-DOF model, a cantilevered truss, and a cantilevered beam.

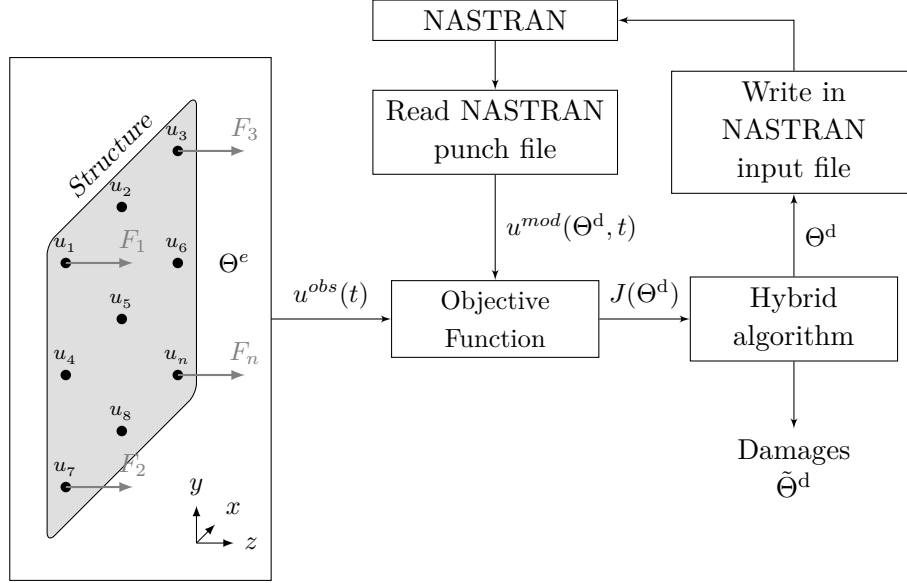
NASTRAN (NASA Structural Analysis System) is a Finite Element Analysis program for use in Computer-Aided Engineering (MSC, 2001; MSC, 2014), that was originally developed by National Aeronautics and Space Administration (NASA) in the mid-1960. The MacNeal-Schendler Corporation (MSC) is one of the principal developers of the public domain code.

NASTRAN is used to perform static, dynamic, and thermal analysis in linear and nonlinear systems. Also, it permits to perform automated structural optimization and embedded fatigue analysis technologies, and uses high-performance computing.

6.1 Experimental design

The proposed damage identification approach takes NASTRAN as the simulator of the structural model, and the hybrid metaheuristics as optimizer, as shown in Figure 6.1.

Figure 6.1 - Vibration-Based Damage Identification as optimization problem using NASTRAN as a black-box model



The experiment design for identifying damages with structures modeled in NASTRAN consists of the following steps:

- (1) Create the NASTRAN input file of the structure, using the solution 109 (direct solution) or 112 (modal solution).
- (2) Create *in silico* observed data u^{obs} using the damage configuration as input of the model, *i.e.* modifying the stiffness of each damaged element.
- (3) Add noise to the synthetic measurements.
- (4) Set-up the hybrid algorithm depending on the characteristics of the structure (dimension number, magnitude of the error, etc.).
- (5) Run the hybrid algorithm at least 50 times, and tabulate the output damage parameters.
- (6) Compute some statistics, depending on the analysis to be performed.

The main program modifies the NASTRAN input file. The stiffness values must be written in the appropriate element position with the correct format. Then the `nastran` application is called, to simulate the performance of the structure with

the stiffness values configured. NASTRAN returns the requested data (such as displacement, frequency response, modal data) written in a punch file. In the next step, the program takes the computed model response, and compares it with the experimentally measured response.

All the experiments in this chapter were done on a PC with 8× Intel® Core™ i7-4790 CPU @3.60GHz, with 8 GB of memory, operating with Ubuntu 16.04.3 LTS.

The configuration common to all experiments is shown in [Table 6.1](#).

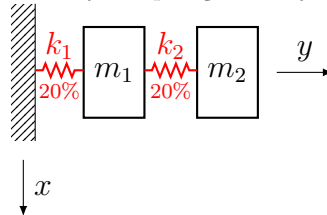
Table 6.1 - Case study 6: Control parameters for the hybrid algorithms

Algorithm	Parameter	Value
MPCA	$N_{particles}$	10
	N_{FE}^{mpca}	10000
	$N_{FE}^{blackboard}$	1000
	$N_{FE}^{exploitation}$	200
	R_{inf}^{inf}	0.7
	R^{sup}	1.1
	Initial population	Random + One solution is set in the best known
HJ	p^s	Cauchy
	N_{FE}^{hj}	10000
	h_{min}	1×10^{-10}
	ρ	0.8

6.2 Case study 1: Damage identification in a spring-mass system with two degrees of freedom

The approach is tested over a spring-mass system with 2-DOF, as shown in [Figure 6.2](#). In this system, $k_1 = 100 \text{ N/m}$, $k_2 = 1 \times 10^4 \text{ N/m}$, $m_1 = 0.1 \text{ kg}$ and $m_2 = 10 \text{ kg}$.

Figure 6.2 - Case study 1: Spring mass system with 2-DOF



A fragment of the bulk section in the NASTRAN input file that models the system is shown in [Listing 6.1](#).

Listing 6.1 - Case study 1: NASTRAN input file

```

$.....2.....3.....4.....5.....6.....7.....8.....9.....10.....$
TIC      777      2      2      0.1
TSTEP    888      1000    0.01    1
GRID     1              0.      2.      0.
GRID     2              0.      1.      0.
GRID     3              0.      0.      0.
GRDSET                                13456
CONM2     1      1              0.1
CONM2     2      2              10.0
CELAS2    11      100.0    1      2      2      2
CELAS2    12      1.0E4    2      2      3      2
SPC       996      3      2

```

An initial displacement of 0.1 m was set at grid 2. Then, the system is left in free vibration. The integration time step was set in $\Delta t = 0.01 \text{ s}$, with a total integration time $t = 10 \text{ s}$. Damping was assumed as negligible.

For the experiments, the optimization algorithm should modify the third column in the CELAS2 entries, which represent each spring in the system. The third column defines the scalar spring stiffness.

Damage of 20% stiffness reduction was assumed in both springs, as represented in [Figure 6.2](#) with red color. For this experiments, the hybrid algorithms MPCA-HJ and RMPCA-HJ were configured with a maximum number of function evaluations $N_{FE}^{mpca} = 10000$ and $N_{FE}^{hj} = 10000$, while the blackboard is updated each $N_{FE}^{blackboard} = 1000$ evaluations.

[Figure 6.3](#) shows the results for the estimated damage parameters. The main columns, each one containing three bars, represent the spring elements. White bars indicate the exact damage values, blue bars represent the estimation by the MPCA-HJ, and red bars, the estimation by the RMPCA-HJ. The graphic shows a perfect damage identification using noiseless data, while some errors appear on the estimation using 5% noisy data.

In all these experiments, a value near to zero of the objective function was reached by the first stage of the hybrid metaheuristics, or exploration phase. It was not necessary to activate the second stage, or intensification phase, performed by the HJ method. The zero value was achieved at the evaluation number 827 for the RMPCA, and 2684 for the MPCA, more than three times higher, as shown in [Figure 6.4](#).

Figure 6.3 - Case study 1: Damage identification on the spring-mass system

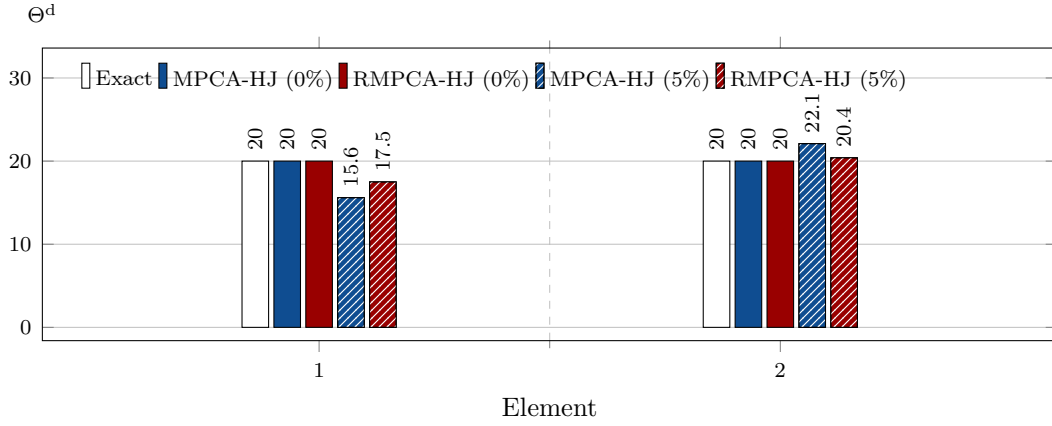
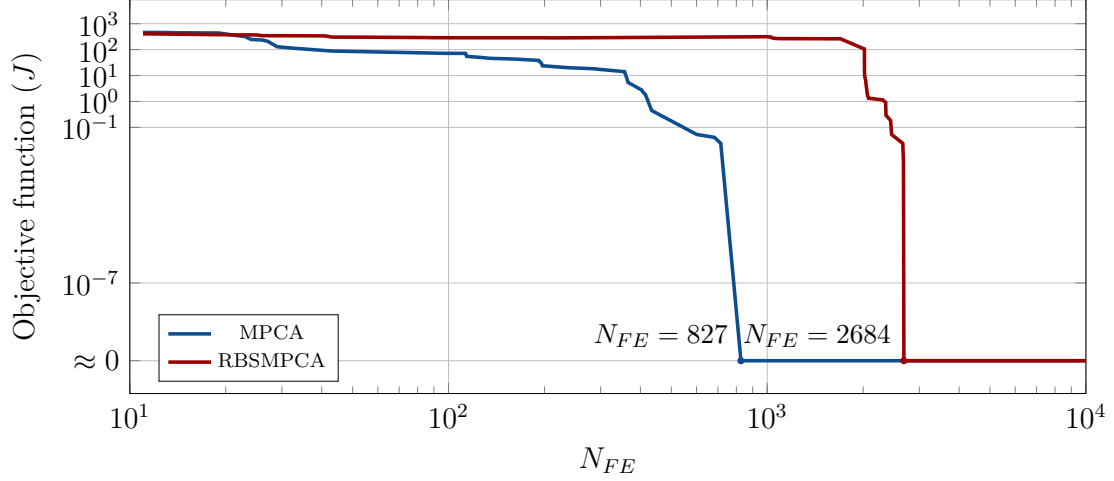


Figure 6.4 - Case study 1: Evolution of the MPCA and RMPCA using noiseless data

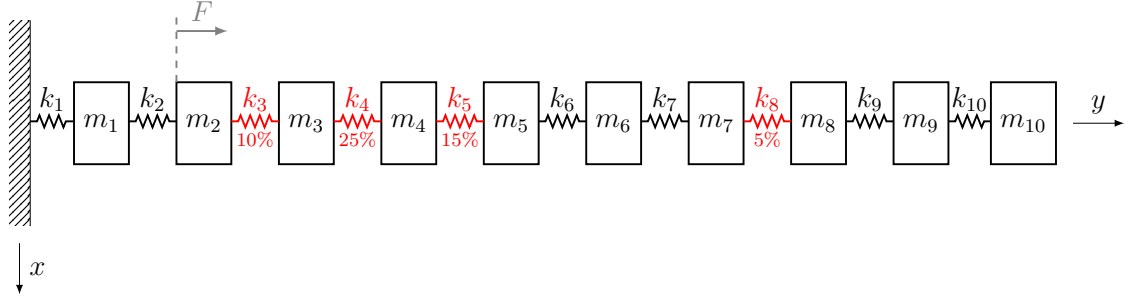


6.3 Case study 2: Damage identification in a spring-mass system with ten degrees of freedom

The results in this section were presented in the Conference of Computational Interdisciplinary Science (CCIS 2016) - (HERNÁNDEZ TORRES et al., 2016).

A spring-mass system with 10-DOF, as shown in Figure 6.5, is tested. In this case, $k_{1-9} = 100 \text{ N/m}$, $k_{10} = 1 \times 10^3 \text{ N/m}$, $m_{1-9} = 0.1 \text{ kg}$ and $m_{10} = 10 \text{ kg}$, and the system is excited with an external force $F(t) = 200 \text{ N}$ applied over the element m_2 .

Figure 6.5 - Case study 2: Spring mass system with 10-DOF



A fragment of the NASTRAN input file that models the system is shown in [Listing 6.2](#). In the bulk data, the entries to be manipulated are those that begin with CELAS2, each one representing the springs in the system. The third column in the card represents the stiffness of the scalar spring.

Listing 6.2 - Case study 2: NASTRAN input file for the spring mass system: segment of the bulk section that defines the spring properties

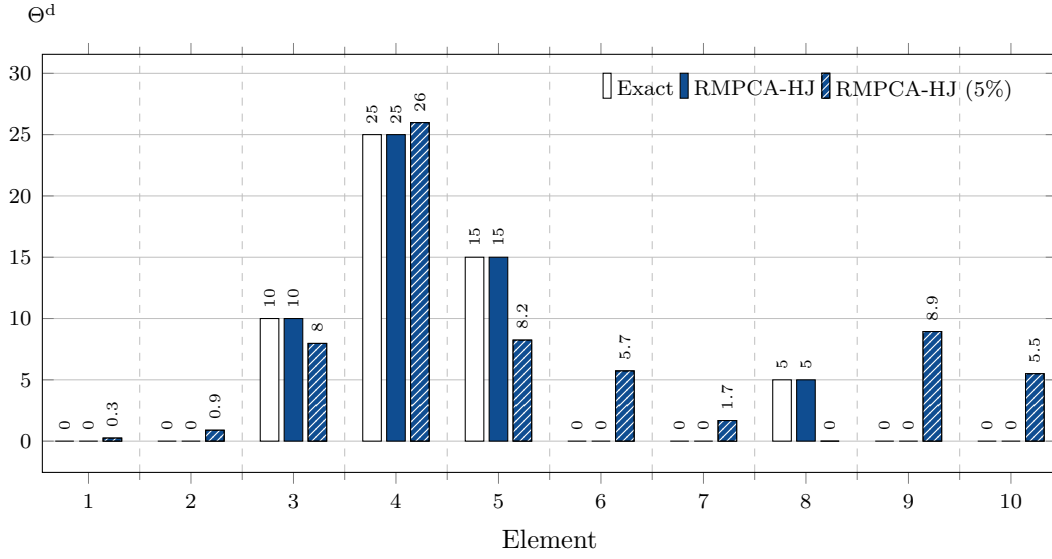
\$	2	3	4	5	6	7	8	9	10	\$
CELAS2		11		100		1		2		2		2								
CELAS2		12		100		2		2		3		2								
CELAS2		13		100		3		2		4		2								
CELAS2		14		100		4		2		5		2								
CELAS2		15		100		5		2		6		2								
CELAS2		16		100		6		2		7		2								
CELAS2		17		100		7		2		8		2								
CELAS2		18		100		8		2		9		2								
CELAS2		19		100		9		2		10		2								
CELAS2		20		1000		10		2		11		2								

Damage of 10% was assumed in the 3rd element, 25% in the 4th element, 15% in the 5th element, and 5% in the 8th element, as represented in [Figure 6.5](#) with red color.

For this experiments, the hybrid algorithm RMPCA-HJ was configured with $N_{FE}^{mpca} = 10000$ and $N_{FE}^{hj} = 10000$, and the blackboard updates each 1000 function evaluations.

[Figure 6.6](#) shows the results of the damage identification. The white bars represent the value of the simulated damage, and the blue bars represent the damage parameters estimated by RMPCA-HJ. Results are almost perfect using noiseless data. False-positives over the 6th, 9th and 10th elements, and a false-negative for the 8th element appeared using noisy data with a level of 5%.

Figure 6.6 - Case study 2: Damage identification on the spring-mass system

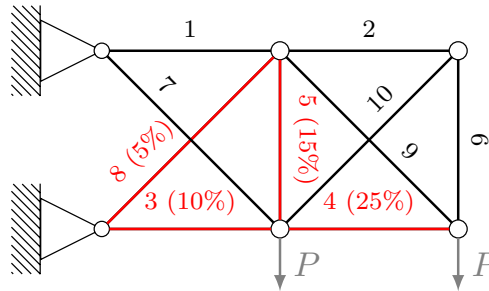


6.4 Case study 3: Damage identification in a cantilever truss

The results in this section were presented in the 4th Conference of Computational Interdisciplinary Science (CCIS 2016) - (HERNÁNDEZ TORRES et al., 2016).

The third case study is a cantilever truss (VENKAYYA, 1971) with 10 bars, as shown in Figure 6.7. The structure configuration was taken from the NASTRAN Compatible Finite Elements (CoFE)¹ (RICCIARDI et al., 2016). The unit were converted to the International System of Units. In this structure, $L = 9.144 \text{ m}$, $P = 444.82 \text{ kN}$, Young's modulus $E = 6.895 \times 10^{10} \text{ Pa}$, and a specific mass equal to $\rho = 27679.9 \text{ kg/m}^3$, with area of $A = 0.00064516 \text{ m}^2$.

Figure 6.7 - Case study 3: Two-bay truss structure



¹http://vtpasquale.github.io/NASTRAN_CoFE/tenBarTrussOptimization.html

The NASTRAN entries to be manipulated are those that begin with **MAT1**, which define the material of single elements, in this case, bars elements. Each line in [Listing 6.3](#) represents a single bar in the structure. The Youngs modulus is modified writing in the third column of the entries.

Listing 6.3 - Case study 3: NASTRAN input file for the cantilever truss: segment of the bulk section that defines the material properties

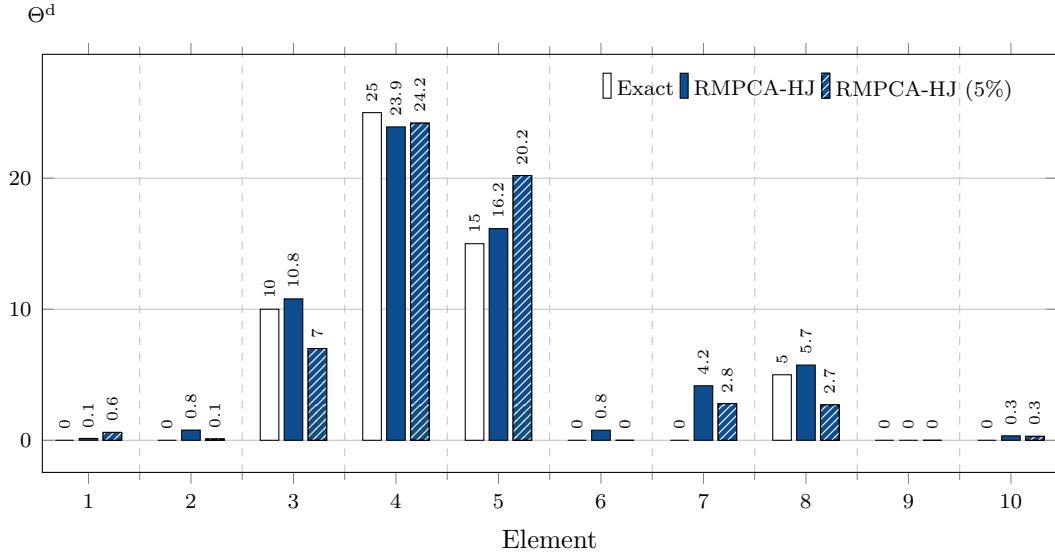
```
$ 1-----2-----3-----4-----5-----6-----7-----8-----
MAT1    101    1.E7          .33    2.59E-4
MAT1    201    1.E7          .33    2.59E-4
MAT1    301    1.E7          .33    2.59E-4
MAT1    401    1.E7          .33    2.59E-4
MAT1    501    1.E7          .33    2.59E-4
MAT1    601    1.E7          .33    2.59E-4
MAT1    701    1.E7          .33    2.59E-4
MAT1    801    1.E7          .33    2.59E-4
```

Damage of 10% was assumed in the 3rd element, 25% in the 4th element, 15% in the 5th element, and 5% in the 8th element, as represented in [Figure 6.7](#) with a red color.

For the experiments, the hybrid algorithm RMPCA-HJ was configured with the same parameters than the [Section 6.2](#).

[Figure 6.8](#) shows the results of the damage identification. In this case, results are worse than in the previous cases. Damages in the 3rd, 4th and 5th elements were well located, but the method misses the severity in the 3rd and 5th elements. The damage located at the 8th element was barely detected using noisy data, with an estimated value of about 3%. In the 7th element, that is crossed with the latter damaged element, appeared a false-positive.

Figure 6.8 - Case study 3: Damage identification on the truss structure

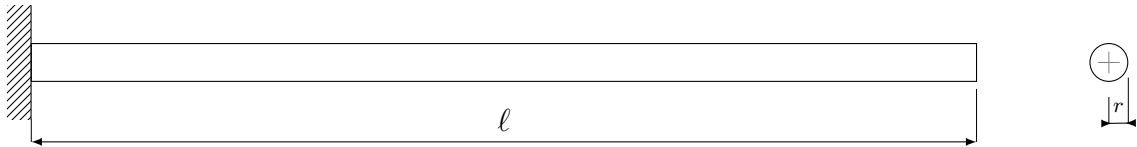


6.5 Case study 4: Damage identification in a cantilever beam

In the last case, the methodology is applied on a cantilever beam model, as shown in Figure 6.9.

The beam has a length $\ell = 3m$, with a radius $r = 0.014m$, torsional constant $j = 6.0 \times 10^{-8} m^4$, area $A = 6.158 \times 10^{-4} m^2$, area moments of inertia $I = 3.0 \times 10^{-8} m^4$, the weight density $\rho_w = 2.65 \times 10^4 N/m^3$, the Young's modulus $E = 7.1 \times 10^{10} N/m^2$, and Poisson's ratio $\nu = 0.33$. This model comes from the modified example in the "Chapter 6 - Transient Response Analysis" of the book "Basic Dynamic Analysis User's Guide" (NASTRAN, 2004).

Figure 6.9 - Case study 4: Cantilever beam structure



The damage identification is performed using the modal transient response (SOL 112). Loads are applied to grids 5 and 10, as shown in Figure 6.10. The waveform of

each load is shown in Figure 6.11 The analysis is run with a time step $\Delta t = 0.001s$, and $t_f = 2s$. Modal damping of 5% critical damping is used for all modes.

The loads are set using the DLOAD entry, that references two TLOAD2 entries. Each TLOAD2 entry references a DAREA, and one of them (the one that defines the load applied at grid 5) references a DELAY entry.

Damage of 10% was assumed in the 2nd and the 4th elements, and a 15% damage was simulated in the 9th element, as represented in Figure 6.10 with a red color.

Figure 6.10 - Case study 4: Cantilever beam with ten elements with applied loads on nodes 5 and 10. Damages are represented in red.

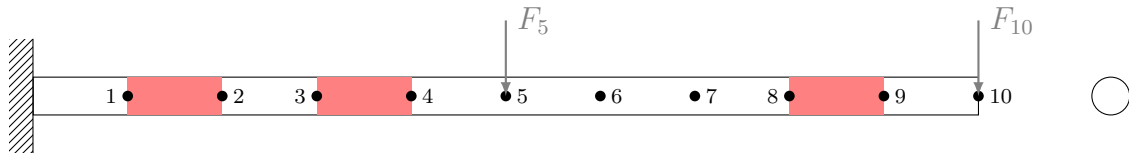
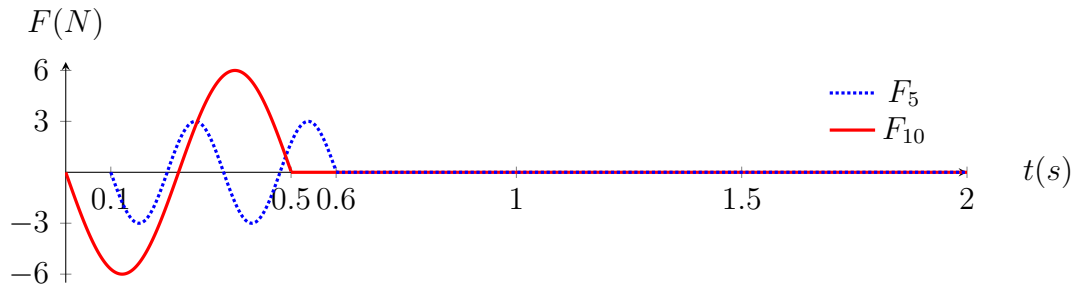


Figure 6.11 - Case study 4: Loads applied on the cantilever beam



Listing 6.4 - Case study 4: NASTRAN input file for the cantilever beam: bulk section

```

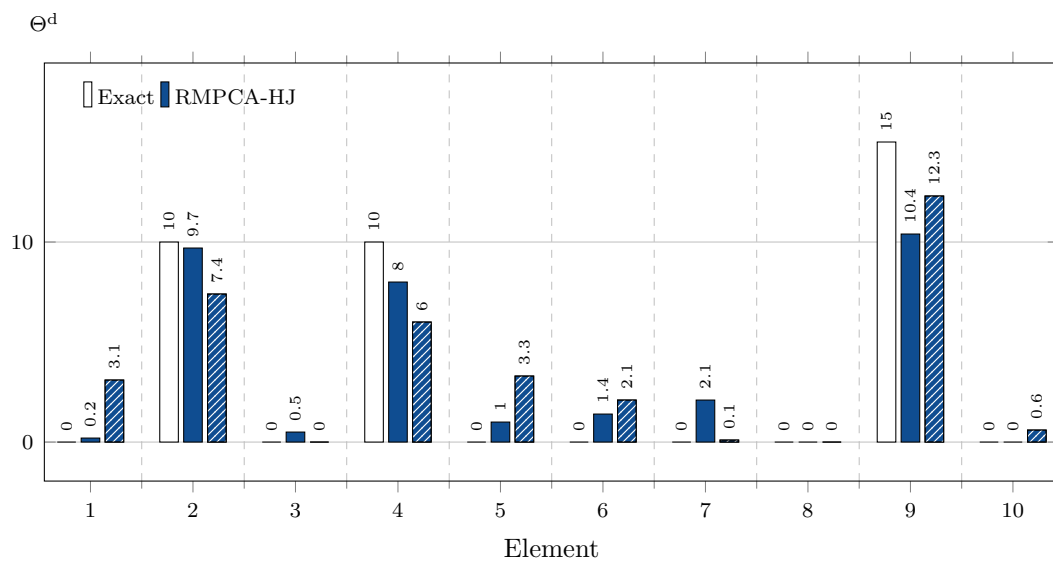
EIGRL 10 -0.1 3000. 0
TSTEP 27 2000 0.001 1
TABDMP1 25 CRIT +TABD
+TABD 0. 0.05 1000. 0.05 ENDT
DLOAD 22 1.0 1.0 231 1.0 232
TLOAD2 231 241 0 0.0 0.50 2.0 90.
TLOAD2 232 242 262 0 0.0 0.50 4.0 90.
DAREA 241 11 2 6.0
DAREA 242 6 2 3.0
DELAY 262 6 2 0.1
GRDSET 345
MAT1 1 7.1+10 0.33 2.65+4
PARAM WTMASS 0.102

```

PBAR	1	1	6.158-4	3.0-8	3.0-8	6.-8	2.414
CBAR	1	1	1	2	0.	1.	0.
CBAR	2	1	2	3	0.	1.	0.
CBAR	3	1	3	4	0.	1.	0.
CBAR	4	1	4	5	0.	1.	0.
CBAR	5	1	5	6	0.	1.	0.
CBAR	6	1	6	7	0.	1.	0.
CBAR	7	1	7	8	0.	1.	0.
CBAR	8	1	8	9	0.	1.	0.
CBAR	9	1	9	10	0.	1.	0.
CBAR	10	1	10	11	0.	1.	0.
GRID	1		0.0	0.	0.		
GRID	2		0.3	0.	0.		
GRID	3		0.6	0.	0.		
GRID	4		0.9	0.	0.		
GRID	5		1.2	0.	0.		
GRID	6		1.5	0.	0.		
GRID	7		1.8	0.	0.		
GRID	8		2.1	0.	0.		
GRID	9		2.4	0.	0.		
GRID	10		2.7	0.	0.		
GRID	11		3.0	0.	0.		
SPC	21	1	126				

Figure 6.12 shows the results of the damage identification. Using noiseless data, all damages were well located. No false-positive appears, but the results were worse in the 5th to the 7th elements. The estimated damage in the 2nd element was very close to the real value, while the results for the 4th and the 9th elements have a difference of about 5% from the exact damage. Using noisy data, false-positives appeared at the 1st, the 5th and the 6th elements. and all the damages were located.

Figure 6.12 - Case study 4: Damage identification on the cantilever beam



6.6 Chapter conclusions

In this chapter, the methodology was tested on some benchmark systems and structures modeled on NASTRAN. A comparison of different hybrid approaches was made over cases without noise added to the synthetic data.

The first case study was a 2-DOF model. Both springs were simulated with a damage of ten percent. MPCA-HJ and RMPCA-HJ were applied to the damage identification, obtaining perfect results.

In the second case, four springs were simulated as damaged in a 10-DOF model. All damages were identified almost perfectly.

RMPCA-HJ was also used to identify damages in a truss structure with ten bars, and fixed in two nodes. Damages were simulated in four bars, and all of them were well identified. A false positive appeared in the seventh bar.

The last experiments were done over a cantilever beam, with three damages which were well located. The damage in the second element, the closest to the fixed end, was well identified. Compared to the real value, the estimated damage in the fourth bar had a difference of two percent, while for the ninth element, the difference was about five percent. In this case, no false positive appeared.

In all cases, after adding noise in the experimental data, the methodology estimates all the damages, but some deviations and false-positives appeared in the damage parameters.

7 CONCLUSIONS AND RECOMMENDATIONS

*I have no special talents. I am only passionately
curious.*

ALBERT EINSTEIN

7.1 Conclusions

The inverse problem of structural damage identification was solved as an optimization problem, using different hybrid algorithms.

The structures were modeled with Finite Elements. For some problems, the models were implemented in FORTRAN language, while for others were designed in the NASTRAN system.

For the experiments, the acceleration, and displacement time series can be used as experimental data.

The objective function to be minimized was formulated as the sum of the square difference between the measured displacement and the data calculated by the forward model.

Some hybrid metaheuristics were tested. These algorithms have a multi-stage approach. The first phase performs the global search, and it was done by MPCA or qG method. The second phase, done by the HJ method, performs the local intensification starting from the best solution found by the first stage. The exploration skill of MPCA was improved using different mechanisms derived from Opposition.

The methodology was tested on structures with different complexities, with different levels of damage. Synthetic data was created to simulate the behaviour of the damaged structures.

Noiseless and noisy data were considered in tests using models implemented in FORTRAN. The experiments were performed using a full set of data, or using a reduced dataset, where data was obtained from some nodes.

The maximum entropy regularization was applied on the damage estimation on the cantilever beam with one damage. This approach obtained good results, eliminating the false-positive that appeared in the solution.

In general, good estimations of damage location and severity were achieved. In some

cases appeared false positives and false-negatives, but in general all large damages were well identified.

Noiseless data and noisy data with level of 5% were considered in tests using models in NASTRAN. In this case, the experiments were performed using a full set of data.

Also, good estimations of damage location and severity were achieved. In some cases appeared false positives, but all damages were well identified.

7.1.1 Main contributions

The main technical and scientific contributions of this thesis are itemized as follows:

- 1) A hybrid algorithm with MPCA and HJ was developed, with a multi-stage approach (HERNANDEZ TORRES et al., 2015; HERNÁNDEZ TORRES et al., 2015b);
- 2) A hybrid algorithm with qG and HJ was developed, with a multi-stage approach (HERNÁNDEZ TORRES et al., 2015b);
- 3) MPCA was improved as follows (HERNÁNDEZ TORRES et al., 2017 in press; HERNÁNDEZ TORRES et al., 2017 forthcoming):
 - a) Some mechanisms derived from Opposition were used to improve the exploration.
 - b) A new Exploitation function was implemented, modifying a single random dimension at time.
 - c) A new way to calculate the scattering probability was tested.
- 4) In Opposition, a new mechanism derived from RBL and CBS, called Rotation-based Sampling was tested (HERNÁNDEZ TORRES et al., 2017 forthcoming).
- 5) The hybrid algorithms were applied to the damage identification on structures with different complexities implemented in FORTRAN (HERNANDEZ TORRES et al., 2015; HERNÁNDEZ TORRES et al., 2015b; HERNÁNDEZ TORRES et al., 2017 in press).

- 6) The hybrid algorithms were applied to the damage identification on structures with different complexities modeled in NASTRAN ([HERNÁNDEZ TORRES et al., 2016](#)).

7.2 Recommendations

A list of recommendations is provided below:

- To test other hybrid configurations, for improving the results, decreasing the computing costs.
- To extend the scope of the methodology, adapting the code to interface with other CAE software;
- To test the methodology on structures with a high number of DOF;
- To perform controlled experiments for different structures.

REFERENCES

- ABDALLA, M. O. Particle swarm optimization (PSO) for structural damage detection. In: INTERNATIONAL CONFERENCE ON APPLIED MATHEMATICS, SIMULATION, MODELLING, CIRCUITS, SYSTEMS AND SIGNALS, 3., 2009, Houston, Texas. **Proceedings...** Houston: IEEE, 2009. p. 29–31. 13
- ABDO, M. A. B.; HORI, M. A numerical study of structural damage detection using changes in the rotation of mode shapes. **Journal of Sound and Vibration**, v. 251, n. 2, p. 227–239, 2002. ISSN 0022460X. 156
- ALAVALA, C. R. **Finite element methods**: basic concepts and applications. [S.l.]: PHI Learning Pvt. Ltd., 2008. 22
- ALLEMANG, R. The modal assurance criterion – Twenty years of use and abuse. **Sound and Vibration**, v. 1, n. August, p. 14–21, 2003. ISSN 00092541. Available from: <<http://www.sandv.com/downloads/0308alle.pdf>>. 156
- ANOCHI, J. A.; CAMPOS VELHO, H. F. Optimization of feedforward neural network by multiple particle collision algorithm. In: IEEE SYMPOSIUM SERIES ON COMPUTATIONAL INTELLIGENCE., 9-12 Dec., Orlando, FL. **Proceedings...** [S.l.]: IEEE, 2014. p. 128–134. Access in: 25 set. 2017. 54
- ANOCHI, J. A.; CAMPOS VELHO, H. F.; FURTADO, H. C. M.; LUZ, E. F. P. Self-configuring two types of neural networks by MPCA. **Journal of Mechanics Engineering and Automation**, David Publishing, v. 5, p. 112–120, 2015. 54
- ARAFA, M.; YOUSSEF, A.; NASSEF, A. A modified continuous reactive tabu search for damage detection in beams. In: AMERICAN SOCIETY OF MECHANICAL ENGINEERS. **ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference**. [S.l.], 2010. p. 1161–1169. 12
- ATALLA, M.; INMAN, D. On model updating using neural networks. **Mechanical Systems and Signal Processing**, v. 12, n. 1, p. 135 – 161, 1998. ISSN 0888-3270. Available from: <<http://www.sciencedirect.com/science/article/pii/S0888327097901382>>. 92

ATALLA, M. J. **Model updating using neural networks**. PhD Thesis (PhD) — Virginia Polytechnic Institute and State University, 1996. 87

AUWERAER, H. Van der; IADEVAIA, M.; EMBORG, U.; GUSTAVSSON, M.; TENGZELIUS, U.; HORLIN, N. Linking test and analysis results in the medium frequency range using principal field shapes. In: INTERNATIONAL SEMINAR ON MODAL ANALYSIS, 3., 1999, Houston, Texas. **Proceedings...** Katholieke Universiteit Leuven, 1999. v. 2, p. 823–830. 155

AVITABILE, P.; PECHINSKY, F. Coordinate orthogonality check (CORTHOG). In: SPIE INTERNATIONAL SOCIETY FOR OPTICAL. **Proceedings...** [S.l.], 1994. p. 753–753. 155

BAARAN, J. **Visual Inspection of Composite Structures**. Braunschweig, Germany, 07 2009. 4

BAGHMISHEH, M. V.; PEIMANI, M.; SADEGHI, M. H.; ETTEFAGH, M. M.; TABRIZI, A. F. A hybrid particle swarmeldermead optimization method for crack detection in cantilever beams. **Applied Soft Computing**, v. 12, n. 8, p. 2217 – 2226, 2012. ISSN 1568-4946. Available from: <<http://www.sciencedirect.com/science/article/pii/S1568494612001287>>. 13

BALAGEAS, D.; FRITZEN, C.-P.; GEMES, A. (Ed.). **Structural Health Monitoring**. London, UK: ISTE, 2006. 1–11 p. ISBN 9780470612071. Available from: <<http://doi.wiley.com/10.1002/9780470612071.fmatter><http://doi.wiley.com/10.1002/9780470612071>>. 10

BANERJEE, S.; RICCI, F.; MONACO, E.; MAL, A. A wave propagation and vibration-based approach for damage identification in structural components. **Journal of Sound and Vibration**, v. 322, n. 1-2, p. 167–183, 2009. ISSN 0022460X. Available from: <<GotoISI>://WOS:000264895800011>. 155

BEGAMBRE, O.; LAIER, J. E. A hybrid particle swarm optimization–simplex algorithm (psos) for structural damage identification. **Advances in Engineering Software**, Elsevier, v. 40, n. 9, p. 883–891, 2009. 13

BEYER, H.-G. **The theory of evolution strategies**. [S.l.]: Springer Science & Business Media, 2013. 50

- BIRTLES, P. Introducing cae into the aerospace industry. **Electronics and Power**, v. 29, n. 1, p. 59–62, January 1983. ISSN 0013-5127. 3
- BLUM, C.; PUCHINGER, J.; RAIDL, G.; ROLI, A. A brief survey on hybrid metaheuristics. In: **4th International Conference on Bioinspired Optimization methods and their applications**. [S.l.: s.n.], 2010. p. 3–16. 51
- BLUM, C.; ROLI, A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. **ACM Computing Surveys (CSUR)**, ACM, v. 35, n. 3, p. 268–308, 2003. 49
- BOLLER, C. Structural health monitoring—its association and use. In: **New Trends in Structural Health Monitoring**. [S.l.]: Springer, 2013. p. 1–79. 16
- BOLLER, C.; CHANG, F.-K.; FUJINO, Y. (Ed.). **Encyclopedia of Structural Health Monitoring**. Chichester, UK: John Wiley & Sons, Ltd, 2009. 2960 p. ISBN 9780470058220. Available from:
<<http://doi.wiley.com/10.1002/9780470061626>>. 10
- BOONLONG, K. Vibration-Based Damage Detection in Beams by Cooperative Coevolutionary Genetic Algorithm. **Advances in Mechanical Engineering**, v. 2014, p. 1–13, 2014. ISSN 1687-8132. Available from:
<<http://www.scopus.com/inward/record.url?eid=2-s2.0-84899563776&partnerID=tZ0tx3y1>>. 12, 13
- BORGES, C. C. H.; BARBOSA, H. J. C.; LEMONGE, A. C. C. A structural damage identification method based on genetic algorithm and vibrational data. **International Journal for Numerical Methods in Engineering**, v. 69, n. 13, p. 2663–2686, 2007. ISSN 0029-5981. 12, 13
- BRAUN, C. E.; CHIWIACOWSKY, L. D.; GOMEZ, A. T. Variations of Ant Colony Optimization for the Solution of the Structural Damage Identification Problem. **Procedia Computer Science**, v. 51, p. 875–884, 2015. ISSN 18770509. Available from:
<<http://www.sciencedirect.com/science/article/pii/S1877050915010261>>. 12, 13, 68
- CAMPOS VELHO, H. F. **Problemas inversos: Conceitos Básicos e Aplicações**. São José dos Campos, 2001. 22

_____. **Introdução aos Problemas Inversos: Aplicações em Pesquisa Espacial.** São José dos Campos, 2008. 22

_____. Inverse problems and regularization. In: **Thermal Measurements and Inverse Techniques.** [S.l.]: CRC Press, 2017. ISBN 9781138113862. 23

CAMPOS VELHO, H. F. de; CHIWIACOWSKY, L. D.; SAMBATTI, S. B. Structural damage identification by a hybrid approach: variational method associated with parallel epidemic genetic algorithm. **Scientia: Interdisciplinary Studies in Computer Science**, v. 17, n. 1, p. 10 – 18, 2006. 12

CANDELIERI, A.; SORMANI, R.; AROSIO, G.; GIORDANI, I.; ARCHETTI, F. A hyper-solution SVM classification framework: application to on-line aircraft structural health monitoring. **Procedia - Social and Behavioral Sciences**, v. 108, p. 57–68, jan 2014. ISSN 18770428. Available from: <http://www.sciencedirect.com/science/article/pii/S1877042813054608>>. 12

CARDEN, E. P.; FANNING, P. Vibration Based Condition Monitoring: A Review. **Structural Health Monitoring**, v. 3, n. 4, p. 355–377, dec 2004. ISSN 1475-9217. Available from: <http://shm.sagepub.com/cgi/doi/10.1177/1475921704047500>>. 10

CASTRO, L. N. D.; TIMMIS, J. **Artificial immune systems: a new computational intelligence approach.** [S.l.]: Springer Science & Business Media, 2002. 51

CAWLEY, P.; ADAMS, R. D. The location of defects in structures from measurements of natural frequencies. **The Journal of Strain Analysis for Engineering Design**, v. 14, n. 2, p. 49–57, apr 1979. ISSN 0309-3247. Available from: <http://sdj.sagepub.com/lookup/doi/10.1243/03093247V142049>>. 156

CHANDRASHEKHAR, M.; GANGULI, R. Structural damage detection using modal curvature and fuzzy logic. **Structural Health Monitoring**, v. 8, n. 4, p. 267–282, jul 2009. ISSN 1475-9217. Available from: <http://shm.sagepub.com/cgi/doi/10.1177/1475921708102088>>. 12, 156

CHANG, C. C.; CHANG, T. Y. P.; XU, Y. G.; WANG, M. L. Structural Damage Detection Using an Iterative Neural Network. **Journal of Intelligent Material Systems and Structures**, v. 11, n. 1, p. 32–42, jan 2000. ISSN 1045389X.

Available from:

<<http://jim.sagepub.com/cgi/doi/10.1106/XU88-UW1T-A6AM-X7EA>>. 11

CHEN, B.; ZHAO, S.-l.; LI, P.-y. Application of Hilbert-Huang Transform in Structural Health Monitoring: A State-of-the-Art Review. **Mathematical Problems in Engineering**, v. 2014, p. 1–22, 2014. ISSN 1024-123X. Available from: <<http://www.hindawi.com/journals/mpe/2014/317954/>>. 155

CHEN, Z.; YU, L. An improved PSO-NM algorithm for structural damage detection. In: INTERNATIONAL CONFERENCE IN ADVANCES IN SWARM AND COMPUTATIONAL INTELLIGENCE (ICSI 2015), 6., 2015. **Proceedings...** Springer International Publishing, 2015. p. 124–132. ISBN 978-3-319-20466-6. Available from: <https://doi.org/10.1007/978-3-319-20466-6_14>. 12

CHIWIACOWSKY, L. D.; CAMPOS VELHO, H. F.; GASBARRI, P. A variational approach for solving an inverse vibration problem. **Inverse Problems in Science and Engineering**, v. 14, n. 5, p. 557–577, jul 2006. ISSN 1741-5977. Available from: <<http://www.tandfonline.com/doi/abs/10.1080/17415970600574237>>. 12

CHIWIACOWSKY, L. D.; SHIGUEMORI, E. H.; VELHO, H. F. C.; GASBARRI, P.; SILVA, J. D. S. A comparison of two different approaches for the damage identification problem. **Journal of Physics: Conference Series**, v. 124, n. 1, p. 012017, jul 2008. ISSN 1742-6596. Available from: <<http://stacks.iop.org/1742-6596/124/i=1/a=012017>>. 11

CHOI, S.; PARK, S.; YOON, S.; STUBBS, N. Nondestructive damage identification in plate structures using changes in modal compliance. **NDT & E International**, v. 38, n. 7, p. 529–540, oct 2005. ISSN 09638695. Available from: <<http://www.sciencedirect.com/science/article/pii/S0963869505000320>>. 156

CHOU, J.-H.; GHABOUESSI, J. Genetic algorithm in structural damage detection. **Computers & Structures**, v. 79, n. 14, p. 1335–1353, jun 2001. ISSN 00457949. Available from: <<http://www.sciencedirect.com/science/article/pii/S004579490100027X>>. 12

CHRYSOCHOIDIS, N. A.; SARAVANOS, D. A. Assessing the effects of

delamination on the damped dynamic response of composite beams with piezoelectric actuators and sensors. **Smart Materials & Structures**, v. 13, n. 4, p. 733–742, 2004. 156

DALTON, R. P.; CAWLEY, P.; LOWE, M. J. S. The Potential of Guided Waves for Monitoring Large Areas of Metallic Aircraft Fuselage Structure. **Journal of Nondestructive Evaluation**, Kluwer Academic Publishers-Plenum Publishers, v. 20, n. 1, p. 29–46, 2001. ISSN 1573-4862. Available from: <http://link.springer.com/article/10.1023/A%3A1010601829968>. 10

DAS, S.; BISWAS, A.; DASGUPTA, S.; ABRAHAM, A. Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications. In: **Foundations of Computational Intelligence Volume 3**. [S.l.]: Springer, 2009. p. 23–55. 50

DAS, S.; SUGANTHAN, P. N. Differential evolution: a survey of the state-of-the-art. **Evolutionary Computation, IEEE Transactions on**, IEEE, v. 15, n. 1, p. 4–31, 2011. 50

DERAEMAERKER, A.; PREUMONT, A. Vibration based damage detection using large array sensors and spatial filters. **Mechanical Systems and Signal Processing**, v. 20, n. 7, p. 1615–1630, 2006. ISSN 08883270. 156

DIAMANTI, K.; SOUTIS, C. Structural health monitoring techniques for aircraft composite structures. **Progress in Aerospace Sciences**, v. 46, n. 8, p. 342–352, nov 2010. ISSN 03760421. Available from: <http://www.sciencedirect.com/science/article/pii/S0376042110000369>. 10

DILENA, M.; MORASSI, A. The use of antiresonances for crack detection in beams. **Journal of Sound and Vibration**, v. 276, n. 1-2, p. 195–214, 2004. ISSN 0022460X. 156

DING, Z.; HUANG, M.; LU, Z. Structural damage detection using artificial bee colony algorithm with hybrid search strategy. **Swarm and Evolutionary Computation**, v. 28, p. 1 – 13, 2016. ISSN 2210-6502. Available from: <http://www.sciencedirect.com/science/article/pii/S2210650215000991>. 12, 14

DOEBLING, S.; FARRAR, C.; PRIME, M.; SHEVITZ, D. **Damage**

identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: A literature review. Los Alamos, NM, may 1996. 133p p. Available from: http://www.osti.gov/energycitations/product.biblio.jsp?osti_{id}=249299http://www.osti.gov/servlets/purl/249299-n4r7Vr/webviewable/. 6, 10

DOEBLING, S. W.; FARRAR, C. R.; PRIME, M. B. **A Summary Review of Vibration-Based Damage Identification Methods**. 1998. 91–105 p. 10, 11, 14

DONSKOY, D. M.; SUTIN, A. M. Vibro-Acoustic Modulation Nondestructive Evaluation Technique. **Journal of Intelligent Material Systems and Structures**, v. 9, n. 9, p. 765–771, sep 1998. ISSN 1045-389X. Available from: <http://jim.sagepub.com/content/9/9/765.abstract>. 156

DORIGO, M.; BIRATTARI, M. Ant colony optimization. In: **Encyclopedia of machine learning**. [S.l.]: Springer, 2010. p. 36–39. 50

DORIGO, M.; BIRATTARI, M.; STÜTZLE, T. Ant colony optimization. **Computational Intelligence Magazine, IEEE**, IEEE, v. 1, n. 4, p. 28–39, 2006. 50

DU, K.-L.; SWAMY, M. **Search and Optimization by Metaheuristics**. [S.l.]: Springer, 2016. ISBN 9783319411927. 50

EBERHART, R. C.; KENNEDY, J. A new optimizer using particle swarm theory. In: INTERNATIONAL SYMPOSIUM ON MICRO MACHINE AND HUMAN SCIENCE, 6., New York, NY. **Proceedings...** [S.l.], 1995. v. 1, p. 39–43. 50

ECHEVARRÍA, L. C.; SANTIAGO, O. L.; SILVA NETO, A. J. Aplicación de los algoritmos Evolución Diferencial y Colisión de Partículas al diagnóstico de fallos en sistemas industriales. **Revista Investigación Operacional**, v. 33, n. 2, p. 160–172, 2012. 54

EMBRAER. **Embraer Commercial Aviation**. [S.l.], 2017. Accessed: 2017-09-07. Available from: <http://www.embraercommercialaviation.com>. 1

_____. **Embraer Defense & Security**. [S.l.], 2017. Accessed: 2017-09-07. Available from: <http://kc-390.com>. 1

_____. **Embraer Executive Jets**. [S.l.], 2017. Accessed: 2017-09-07. Available

from: <<http://www.embraerexecutivejets.com/>>. 1

ERGEZER, M.; SIMON, D.; DU, D. Oppositional biogeography-based optimization. In: IEEE. **Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on**. [S.l.], 2009. p. 1009–1014. 59

FAN, W.; QIAO, P. Vibration-based Damage Identification Methods: A Review and Comparative Study. **Structural Health Monitoring**, v. 10, n. 1, p. 83–111, jan 2011. ISSN 1475-9217. Available from: <<http://shm.sagepub.com/cgi/doi/10.1177/1475921710365419>>. 10

FARRAR, C. R.; DOEBLING, S. W.; NIX, D. a. Vibration-based structural damage identification. **Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences**, v. 359, n. 1778, p. 131–149, jan 2001. ISSN 1364-503X. Available from: <<http://rsta.royalsocietypublishing.org/cgi/doi/10.1098/rsta.2000.0717>>. 5

FARRAR, C. R.; LIEVEN, N. a. J. Damage prognosis: the future of structural health monitoring. **Philosophical transactions. Series A, Mathematical, physical, and engineering sciences**, v. 365, n. 1851, p. 623–632, 2007. ISSN 1364-503X. 5

FARRAR, C. R.; PARK, G.; ANGHEL, M.; BEMENT, M. T.; SALVINO, L. **Structural health monitoring for ship structures**. [S.l.], 2009. 5

FARRAR, C. R.; WORDEN, K. An introduction to structural health monitoring. **Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences**, The Royal Society, v. 365, n. 1851, p. 303–315, 2007. 16

_____. _____. In: **New Trends in Vibration Based Structural Health Monitoring**. [S.l.]: Springer, 2010. p. 1–17. 5

FASSOIS, S. D.; SAKELLARIOU, J. S. Time-series methods for fault detection and identification in vibrating structures. **Philosophical transactions. Series A, Mathematical, physical, and engineering sciences**, v. 365, n. 1851, p. 411–448, 2007. ISSN 1364-503X. 10, 155

FATAHI, L.; MORADI, S. Multiple crack identification in frame structures using a hybrid bayesian model class selection and swarm-based optimization methods.

Structural Health Monitoring, v. 0, n. 0, p. 1475921716683360, 2017. Available from: <<http://dx.doi.org/10.1177/1475921716683360>>. 12

FEA Compare. **FEA Compare**. 2017. Visited on: 2017-08-01. Available from: <<http://feacompare.com/>>. 3

FELIPPA, C. A. **Introduction to finite element methods**. [S.l.]: Department of Aerospace Engineering Sciences, University of Colorado at Boulder, 2014. 31

FOGEL, L. J. **Intelligence Through Simulated Evolution: Forty Years of Evolutionary Programming**. New York, NY, USA: John Wiley & Sons, Inc., 1999. ISBN 0-471-33250-X. 50

FOTSCH, D.; EWINS, D. Further applications of the FMAC. **Rolls Royce PLC-Report-PNR**, Citeseer, 2001. 155

FRISWELL, M. I. Damage identification using inverse methods. **Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences**, The Royal Society, v. 365, n. 1851, p. 393–410, 2007. 8

FRITZEN, C.-P.; KRAEMER, P. Self-diagnosis of smart structures based on dynamical properties. **Mechanical Systems and Signal Processing**, v. 23, n. 6, p. 1830–1845, aug 2009. ISSN 08883270. Available from: <<http://linkinghub.elsevier.com/retrieve/pii/S0888327009000119><http://www.sciencedirect.com/science/article/pii/S0888327009000119>>. 7, 10, 23

FU, Y. M.; YU, L. A DE-Based Algorithm for Structural Damage Detection. **Advanced Materials Research**, v. 919-921, p. 303–307, apr 2014. ISSN 1662-8985. Available from: <<http://www.scientific.net/AMR.919-921.303>>. 12

GANDOMI, A. H.; YANG, X.-S.; TALATAHARI, S.; ALAVI, A. H. **Metaheuristic applications in structures and infrastructures**. [S.l.]: Newnes, 2013. 49

GEEM, Z. W.; KIM, J. H.; LOGANATHAN, G. A new heuristic optimization algorithm: harmony search. **Simulation**, SAGE Publications, v. 76, n. 2, p. 60–68, 2001. 51

GIURGIUTIU, V. **Damage Detection in Thin Plates and Aerospace**

Structures with the Electro-Mechanical Impedance Method. 2005.
99–118 p. 155

GLOVER, F. Future paths for integer programming and links to artificial intelligence. **Computers & operations research**, Elsevier, v. 13, n. 5, p. 533–549, 1986. 49

GOMES, H.; SILVA, N. Some comparisons for damage detection on structures using genetic algorithms and modal sensitivity method. **Applied Mathematical Modelling**, v. 32, n. 11, p. 2216–2232, nov 2008. ISSN 0307904X. Available from: <<http://www.sciencedirect.com/science/article/pii/S0307904X07001850>>. 12

GUILD, F. J. The detection of cracks in damaged composite materials. **Journal of Physics D: Applied Physics**, v. 14, n. 8, p. 1561–1573, aug 1981. ISSN 0022-3727. Available from: <<http://stacks.iop.org/0022-3727/14/i=8/a=023>>. 156

HADAMARD, J. **Lectures on Cauchy's problem in linear partial differential equations**. [S.l.]: Courier Corporation, 1925. 22

HADJILEONTIADIS, L. J.; DOUKA, E.; TROCHIDIS, A. Fractal dimension analysis for crack identification in beam structures. **Mechanical Systems and Signal Processing**, v. 19, n. 3, p. 659–674, 2005. ISSN 08883270. 156

HE JIA; XU, Y.-L. **Smart Civil Structures**. 1. ed. [S.l.]: CRC Press, Taylor & Francis, 2017. ISBN 1498743986,978-1-4987-4398-3,9781498743990,1498743994. 14

HE, R.-S.; HWANG, S.-F. Damage detection by an adaptive real-parameter simulated annealing genetic algorithm. **Computers & Structures**, v. 84, n. 31-32, p. 2231–2243, dec 2006. ISSN 00457949. Available from: <<http://linkinghub.elsevier.com/retrieve/pii/S0045794906002641>>. 12

HERNANDEZ TORRES, R.; CHIWIACOSKY, L. D.; VELHO, H. F. d. C. Multi-particle collision algorithm with Hooke-Jeeves for solving a structural damage detection problem. In: INTERNATIONAL CONFERENCE ON COMPOSITE SCIENCE AND TECHNOLOGY, 10. (ICCST), 2-4 Sept., Lisboa, Portugal. **Proceedings...** Instituto Superior Técnico, 2015. v. 1, p. 1–12. Available from: <http://www.dem.ist.utl.pt/iccst10/files/ICCST10_Proceedings/pdf/WEB_PAPERS/ICCST10_Upload_195.pdf>. Access in: 03 nov. 2017. 12, 68, 75, 78,

HERNÁNDEZ TORRES, R.; CAMPOS VELHO, H. F.; CHIWIACOWSKY, L. D. Vibration-based damage identification using the multi-particle collision algorithm with Hooke-Jeeves coupled with NASTRAN. In: CONFERENCE OF COMPUTATIONAL INTERDISCIPLINARY SCIENCE (CCIS 2016), 4., 2016, São José dos Campos, SP, Brazil. **Proceedings...** Instituto Nacional de Pesquisas Espaciais, 2016. 113, 115, 123

_____. Rotation-based multi-particle collision algorithm with Hooke-Jeeves approach applied to the structural damage identification. In: PLATT, G. M.; YANG, X.-S.; SILVA NETO, A. J. (Ed.). **Computational Intelligence in Engineering Problems**. [S.l.]: Springer, 2017 forthcoming. 96, 122

HERNÁNDEZ TORRES, R.; CHIWIACOWSKY, L. D.; CAMPOS VELHO, H. F. Multi-particle collision algorithm with hooke jeeves applied to the damage identification in a Kabe antenna. In: **Congresso Nacional de Matemática Aplicada e Computacional (CNMAC 2017)**. [S.l.: s.n.], 2017 in press. 87, 122

HERNÁNDEZ TORRES, R.; LUZ, E. F. P.; CAMPOS VELHO, H. F. Multi-particle collision algorithm for solving an inverse radiative problem. In: CONSTANDA, C.; KIRSCH, A. (Ed.). **Integral Methods in Science and Engineering. Theoretical and Computational Advances**. [S.l.]: Birkhäuser Basel, 2015. 54

HERNÁNDEZ TORRES, R.; SCARABELLO, M. C.; CAMPOS VELHO, H.; CHIWIACOWSKY, L. D.; SOTERRONI, A. C.; RAMOS, F. M. A hybrid method usign q-gradient to identify structural damage. In: IBERIAN LATIN-AMERICAN CONGRESS ON COMPUTATIONAL METHODS IN ENGINEERING, XXXV., 2015, Rio de Janeiro, RJ, Brazil. **Proceedings...** [S.l.], 2015. 12, 68, 78, 122

HIBBELER, R. **Engineering Mechanics: Statics**. [S.l.]: Prentice Hall, 2010. (Engineering mechanics). ISBN 9780136077909. 36

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**. Cambridge, MA, USA: MIT Press, 1992. ISBN 0-262-58111-6. 50

HOOKE, R.; JEEVES, T. A. “direct search” solution of numerical and statistical problems. **Journal of the ACM (JACM)**, ACM, v. 8, n. 2, p. 212–229, 1961. 68

HUANG, C.-C.; LOH, C.-H. Nonlinear Identification of Dynamic Systems Using Neural Networks. **Computer-Aided Civil and Infrastructure Engineering**, v. 16, n. 1, p. 28–41, jan 2001. ISSN 1093-9687. Available from: <http://doi.wiley.com/10.1111/0885-9507.00211>>. 11

HUANG, N. E.; SHEN, Z.; LONG, S. R.; WU, M. C.; SHIH, H. H.; ZHENG, Q.; YEN, N.-C.; TUNG, C. C.; LIU, H. H. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. In: THE ROYAL SOCIETY. **Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences**. [S.l.], 1998. v. 454, n. 1971, p. 903–995. 155

ILCEWICZ, L. Composite damage tolerance and maintenance safety issues. In: **FAA Damage Tolerance and Maintenance Workshop, Rosemont, IL**. [S.l.: s.n.], 2006. 4

INPE. **Lançamento CBERS-4**. [S.l.], 2011. Accessed: 2017-09-07. Available from: http://www.cbers.inpe.br/sobre_satelite/lancamento_cbers4.php>. 2

_____. **Missão Amazonia-1**. [S.l.], 2017. Accessed: 2017-09-07. Available from: http://www.inpe.br/amazonia-1/sobre_satelite/>. 2

_____. **Satélite SCD-1**. [S.l.], 2017. Accessed: 2017-09-07. Available from: http://www.inpe.br/scd1/site_scd/scd1/osatelite.htm>. 2

JACKSON, F. H. On q -functions and a certain difference operator. **Trans. Roy Soc. Edin.**, v. 46, p. 253–281, 1908. 65

_____. On q -definite integrals. **Quart. J. Pure and Appl. Math.**, v. 41, p. 193–203, 1910. 65

_____. q -difference equations. **American Journal of Mathematics**, v. 32, p. 307–314, 1910. 65

JAISHI, B.; REN, W.-X. Damage detection by finite element model updating using modal flexibility residual. **Journal of Sound and Vibration**, v. 290, n. 1-2, p. 369–387, feb 2006. ISSN 0022460X. Available from: <http://www.sciencedirect.com/science/article/pii/S0022460X05002919>>. 156

JAYNES, E. T. Information theory and statistical mechanics. **Physical review**, APS, v. 106, n. 4, p. 620, 1957. 25

JHANG, K.-Y. Nonlinear ultrasonic techniques for nondestructive assessment of micro damage in material: A review. **International Journal of Precision Engineering and Manufacturing**, v. 10, n. 1, p. 123–135, jan 2009. ISSN 1229-8557. Available from:
<<http://link.springer.com/10.1007/s12541-009-0019-y>>. 10

JIMÉNEZ, R.; SANTOS, L.; KUHL, N.; EGANA, J.; SOTO, R. An inverse eigenvalue procedure for damage detection in rods. **Computers & Mathematics with Applications**, Elsevier, v. 47, n. 4-5, p. 643–657, 2004. 4

JIN, C.; JANG, S.; SUN, X.; LI, J.; CHRISTENSON, R. Damage detection of a highway bridge under severe temperature changes using extended kalman filter trained neural network. **Journal of Civil Structural Health Monitoring**, v. 6, n. 3, p. 545–560, 2016. ISSN 2190-5479. Available from:
<<http://dx.doi.org/10.1007/s13349-016-0173-8>>. 12

JOHNSON, T. J.; ADAMS, D. E. Transmissibility as a Differential Indicator of Structural Damage. **Journal of Vibration and Acoustics**, v. 124, n. 4, p. 634, 2002. ISSN 07393717. Available from:
<<http://vibrationacoustics.asmedigitalcollection.asme.org/article.aspx?articleid=1470461>>. 155

JOURDAN, L.; BASSEUR, M.; TALBI, E.-G. Hybridizing exact methods and metaheuristics: A taxonomy. **European Journal of Operational Research**, Elsevier, v. 199, n. 3, p. 620–629, 2009. 51

JR., I. F.; YANG, X.-S.; FISTER, I.; BREST, J.; FISTER, D. A brief review of nature-inspired algorithms for optimization. **CoRR**, abs/1307.4186, 2013. Available from: <<http://dblp.uni-trier.de/db/journals/corr/corr1307.html#FisterYFBF13>>. 50

KABE, A. M. Stiffness matrix adjustment using mode data. **AIAA journal**, v. 23, n. 9, p. 1431–1436, 1985. 87

KAIPPIO, J.; SOMERSALO, E. **Statistical and computational inverse problems**. [S.l.]: Springer Science & Business Media, 2006. 23

KÄMPF, J. H.; WETTER, M.; ROBINSON, D. A comparison of global optimization algorithms with standard benchmark functions and real-world applications using energyplus. **Journal of Building Performance Simulation**, Taylor & Francis, v. 3, n. 2, p. 103–120, 2010. 68

KANG, F.; LI, J.-J.; XU, Q. Damage detection based on improved particle swarm optimization using vibration data. **Applied Soft Computing**, Elsevier, v. 12, n. 8, p. 2329–2335, 2012. 13

KARABOGA, D.; BASTURK, B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. **Journal of global optimization**, Springer, v. 39, n. 3, p. 459–471, 2007. 50

KAVEH, A.; ZOLGHADR, A. An improved css for damage detection of truss structures using changes in natural frequencies and mode shapes. **Advances in Engineering Software**, Elsevier, v. 80, p. 93–100, 2015. 12

_____. Cyclical parthenogenesis algorithm for guided modal strain energy based structural damage detection. **Applied Soft Computing**, p. –, 2017. ISSN 1568-4946. Available from:
<<http://www.sciencedirect.com/science/article/pii/S1568494617301813>>. 12

KENNEDY, J. Particle swarm optimization. In: **Encyclopedia of Machine Learning**. [S.l.]: Springer, 2010. p. 760–766. 50

KESSLER, S. Damage detection in composite materials using frequency response methods. **Composites Part B: Engineering**, v. 33, n. 1, p. 87–95, 2002. ISSN 13598368. Available from:
<[http://dx.doi.org/10.1016/S1359-8368\(01\)00050-6](http://dx.doi.org/10.1016/S1359-8368(01)00050-6)>. 155

KHATIR, A.; TEHAMI, M.; KHATIR, S.; WAHAB, M. A. Multiple damage detection and localization in beam-like and complex structures using co-ordinate modal assurance criterion combined with firefly and genetic algorithms. **Journal of Vibroengineering**, v. 18, n. 8, 2016. 12

KHOO, L. M.; MANTENA, P. R.; JADHAV, P. **Structural Damage Assessment Using Vibration Modal Analysis**. 2004. 177–194 p. 156

KHOSHNOUDIAN, F.; ESFANDIARI, A. Structural damage diagnosis using

modal data. **Scientia Iranica**, v. 18, n. 4, p. 853 – 860, 2011. ISSN 1026-3098.
Available from:

<<http://www.sciencedirect.com/science/article/pii/S102630981100126X>>. 4

KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. et al. Optimization by simulated annealing. **Science**, World Scientific, v. 220, n. 4598, p. 671–680, 1983. 51

KOKOT, S.; ZEMBATY, Z. Damage reconstruction of 3D frames using genetic algorithms with Levenberg–Marquardt local search. **Soil Dynamics and Earthquake Engineering**, v. 29, n. 2, p. 311–323, feb 2009. ISSN 02677261.
Available from:

<<http://linkinghub.elsevier.com/retrieve/pii/S0267726108000304>>. 12

KOPSAFTOPOULOS, F. P.; FASSOIS, S. D. Identification of stochastic systems under multiple operating conditions: the vector dependent FP-ARX parametrization. In: IEEE. **Control and Automation, 2006. MED'06. 14th Mediterranean Conference on**. [S.l.], 2006. p. 1–6. 155

KOUREHLI, S. S.; BAGHERI, A.; AMIRI, G. G.; GHAFORY-ASHTIANY, M. Structural damage detection using incomplete modal data and incomplete static response. **KSCE Journal of Civil Engineering**, v. 17, n. 1, p. 216–223, jan 2013. ISSN 1226-7988. Available from:

<<http://link.springer.com/10.1007/s12205-012-1864-2>>. 12

KROHN, N.; STOESSEL, R.; BUSSE, G. Acoustic non-linearity for defect selective imaging. In: **Ultrasonics**. [S.l.: s.n.], 2002. v. 40, n. 1-8, p. 633–637. ISBN 0041-624X (Print)\r0041-624X (Linking). ISSN 0041624X. 156

LANGDON, W. B.; GUSTAFSON, S. M. Genetic programming and evolvable machines: ten years of reviews. **Genetic Programming and Evolvable Machines**, v. 11, n. 3/4, p. 321–338, sep. 2010. ISSN 1389-2576. Tenth Anniversary Issue: Progress in Genetic Programming and Evolvable Machines.
Available from:

<<http://www.cs.ucl.ac.uk/staff/W.Langdon/ftp/papers/gppubs10.pdf>>. 50

LI, S.; LU, Z. Multi-swarm fruit fly optimization algorithm for structural damage identification. **Struct Eng Mech**, v. 56, n. 3, p. 409–22, 2015. 12

- LI, Y. Hypersensitivity of strain-based indicators for structural damage identification: A review. **Mechanical Systems and Signal Processing**, v. 24, n. 3, p. 653–664, apr 2010. ISSN 08883270. Available from: <<http://www.sciencedirect.com/science/article/pii/S0888327009003732><http://linkinghub.elsevier.com/retrieve/pii/S0888327009003732>>. 10
- LIN, M.-H.; TSAI, J.-F.; YU, C.-S. A review of deterministic optimization methods in engineering and management. **Mathematical Problems in Engineering**, Hindawi Publishing Corporation, v. 2012, 2012. 49
- LIU, H.; WU, Z.; LI, H.; WANG, H.; RAHNAMAYAN, S.; DENG, C. Rotation-based learning: A novel extension of opposition-based learning. In: _____. **PRICAI 2014: Trends in Artificial Intelligence: 13th Pacific Rim International Conference on Artificial Intelligence, Gold Coast, QLD, Australia, December 1-5, 2014. Proceedings**. Cham: Springer International Publishing, 2014. p. 511–522. ISBN 978-3-319-13560-1. Available from: <https://doi.org/10.1007/978-3-319-13560-1_41>. 60
- LIU, X.; LIEVEN, N. A. J.; ESCAMILLA-AMBROSIO, P. J. Frequency response function shape-based methods for structural damage localisation. **Mechanical Systems and Signal Processing**, v. 23, n. 4, p. 1243–1259, 2009. ISSN 08883270. 155
- LIU, Y.; NAYAK, S. Structural health monitoring: state of the art and perspectives. **JOM**, v. 64, n. 7, p. 789–792, 2012. ISSN 1543-1851. Available from: <<http://dx.doi.org/10.1007/s11837-012-0370-9>>. 10
- LOGAN, D. L. **A first course in the finite element method**. [S.l.]: Cengage Learning, 2011. 3, 31
- LONG, Q.; WU, C. A hybrid method combining genetic algorithm and hooke-jeeves method for constrained global optimization. **Journal of industrial and management optimization**, v. 10, n. 4, p. 1279–1296, 2014. 68
- LU, Q.; REN, G.; ZHAO, Y. Multiple damage location with flexibility curvature and relative frequency change for beam structures. **Journal of Sound and Vibration**, v. 253, n. 5, p. 1101–1114, jun 2002. ISSN 0022460X. Available from: <<http://www.sciencedirect.com/science/article/pii/S0022460X01940921>>. 156

LUZ, E. F. P.; BECCENERI, J. C.; CAMPOS VELHO, H. F. A new multi-particle collision algorithm for optimization in a high performance environment. **Journal of Computational Interdisciplinary Sciences**, v. 1, n. 1, p. 3–10, 2008. 51, 52

MAHDAVI, S. H.; Abdul Razak, H. A wavelet-based approach for vibration analysis of framed structures. **Applied Mathematics and Computation**, v. 220, p. 414–428, 2013. ISSN 00963003. 155

MAIA, N.; SILVA, J.; ALMAS, E.; SAMPAIO, R. Damage detection in structures: from mode shape to frequency response function methods. **Mechanical Systems and Signal Processing**, v. 17, n. 3, p. 489–498, 2003. ISSN 08883270. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0888327002915062>>. 155, 156

MAJUMDAR, A.; MAITI, D. K.; MAITY, D. Damage assessment of truss structures from changes in natural frequencies using ant colony optimization.

Applied Mathematics and Computation, Elsevier Inc., v. 218, n. 19, p. 9759–9772, jun 2012. ISSN 00963003. Available from:

<<http://linkinghub.elsevier.com/retrieve/pii/S0096300312002639><http://dx.doi.org/10.1016/j.amc.2012.03.031>>. 12

MARES, C.; SURACE, C. An application of genetic algorithms to identify damage in elastic structures. **Journal of Sound and Vibration**, v. 195, n. 2, p. 195–215, 1996. ISSN 0022460X. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0022460X96904162>>. 12, 13

MECHANICAL ENGINEERING. **List of FEA Softwares**. 2011. Visited on:

2017-08-01. Available from: <<http://www.mechanicalengineeringblog.com/1757-finite-element-analysis-fea-list-of-fea-softwares-list-of-open-source-softwares-list-of-commercial-softwares/>>. 3

MEDEIROS, R. D. **SHM system for composite aeronautic structures using combination of different methods: A contribution for residual strength prediction**. PhD Thesis (PhD) — KU Leuven, 2016. 10, 156

MEHRJOO, M.; KHAJI, N.; MOHARRAMI, H.; BAHREININEJAD, A. Damage detection of truss bridge joints using Artificial Neural Networks. **Expert Systems with Applications**, v. 35, n. 3, p. 1122–1131, oct 2008. ISSN 09574174. Available

from: <<http://linkinghub.elsevier.com/retrieve/pii/S0957417407003302>>.

11

MESSINA, A.; WILLIAMS, E. J.; CONTURSI, T. Structural Damage Detection by A Sensitivity and Statistical-Based Method. **Journal of Sound and Vibration**, v. 216, n. 5, p. 791–808, 1998. ISSN 0022460X. 156

MOAVENI, S. **Finite Element Analysis Theory and Application with ANSYS**, 3/e. [S.l.]: Pearson Education India, 2008. 31, 32, 36

MOHAN, S.; MAITI, D.; MAITY, D. Structural damage assessment using FRF employing particle swarm optimization. **Applied Mathematics and Computation**, Elsevier Inc., v. 219, n. 20, p. 10387–10400, jun 2013. ISSN 00963003. Available from: <<http://www.sciencedirect.com/science/article/pii/S0096300313004165>

<http://linkinghub.elsevier.com/retrieve/pii/S0096300313004165>>. 12

MONTALVAO, D.; MAIA, N.; RIBEIRO, A. A Review of Vibration-based Structural Health Monitoring with Special Emphasis on Composite Materials. **The Shock and Vibration Digest**, v. 38, n. 4, p. 295–324, jul 2006. ISSN 0583-1024. Available from:

<<http://svd.sagepub.com/cgi/doi/10.1177/0583102406065898>>. 10

MONTANARI, L. **Vibration-based damage identification in beam structures through wavelet analysis**. PhD Thesis (PhD) — University of Parma, 2014. 8

MONTANARI, L.; SPAGNOLI, A.; BASU, B.; BRODERICK, B. On the effect of spatial sampling in damage detection of cracked beams by continuous wavelet transform. **Journal of Sound and Vibration**, v. 345, p. 233 – 249, 2015. ISSN 0022-460X. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0022460X15000954>>. 155

MORASSI, A. Damage detection and generalized Fourier coefficients. **Journal of Sound and Vibration**, v. 302, n. 1-2, p. 229–259, apr 2007. ISSN 0022460X. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0022460X06008686>>. 155

MOROZOV, V. A. **Methods for solving incorrectly posed problems**. [S.l.]: Springer, 1984. 25

MOSCATO, P. **On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts - Towards Memetic Algorithms**. 1989. 51

MOSER, I.; CHIONG, R. A hooke-jeeves based memetic algorithm for solving dynamic optimisation problems. In: **HAIS**. [S.l.: s.n.], 2009. 68

MSC. **Getting Started with MSC.Nastran User's Guide**. 2001. 109, 157

_____. **MSC Nastran 2014.1. Installation and Operations Guide**. [S.l.: s.n.], 2014. 109, 157

MUNIZ, W.; RAMOS, F.; CAMPOS VELHO, H. Entropy- and tikhonov-based regularization techniques applied to the backwards heat equation. **Computers & Mathematics with Applications**, v. 40, n. 8, p. 1071 – 1084, 2000. ISSN 0898-1221. Available from:
<<http://www.sciencedirect.com/science/article/pii/S0898122100850178>>. 25

NASA. **STS-97 Shuttle Mission Imagery**. [S.l.], 2012. Accessed: 2017-09-07. Available from: <<https://spaceflight.nasa.gov/gallery/images/shuttle/sts-97/html/sts097-704-080.html>>. 83

_____. **NASA - Astronomy Picture of the Day**. [S.l.], 2017. Accessed: 2017-09-07. Available from: <<https://apod.nasa.gov/apod/ap980703.html>>. 3

_____. **NASA - STS-114 Picture of ISS**. [S.l.], 2017. Accessed: 2017-09-07. Available from: <https://www.nasa.gov/mission_pages/station/main/s114e7221_feature.html>. 3

NASTRAN, M. Basic dynamic analysis user's guide. **MSC. Software Corporation. US A**, 2004. xix, 30, 31, 117, 161

NEFSKE, D. J.; SUNG, S. H. Correlation of a coarse-mesh finite element model using structural system identification and a frequency response assurance criterion. In: **SPIE - THE INTERNATIONAL SOCIETY FOR OPTICAL ENGINEERING. Proceedings...** [S.l.], 1996. p. 597–602. ISBN 9780912053493 [0912053496]. 155

NEWMARK, N. M. A method of computation for structural dynamics. **Journal**

of the **Engineering Mechanics Division**, ASCE, v. 85, n. 3, p. 67–94, 1959. 28

NICHOLS, J. M.; MURPHY, K. D. **Modeling and estimation of structural damage**. [S.l.]: John Wiley & Sons, 2016. 75

OOIJEEVAAR, T. H. **Vibration based structural health monitoring of composite skin-stiffener structures**. 171 p. PhD Thesis (PhD) — University of Twente, 2014. Available from: <<http://doc.utwente.nl/89656/>>. 9, 10, 11, 14, 15, 156

OSMAN, I. H.; LAPORTE, G. **Metaheuristics: A bibliography**. [S.l.]: Springer, 1996. 49

PAI, P. F. Time–frequency characterization of nonlinear normal modes and challenges in nonlinearity identification of dynamical systems. **Mechanical Systems and Signal Processing**, Elsevier, v. 25, n. 7, p. 2358–2374, 2011. 155

PAN, C.; YU, L. Structural damage detection and moving force identification based on firefly algorithm. In: _____. **Advances in Swarm and Computational Intelligence: 6th International Conference, ICSI 2015, held in conjunction with the Second BRICS Congress, CCI 2015, Beijing, China, June 25-28, 2015, Proceedings, Part I**. Cham: Springer International Publishing, 2015. p. 57–64. ISBN 978-3-319-20466-6. Available from: <http://dx.doi.org/10.1007/978-3-319-20466-6_6>. 12

PANDEY, A.; BISWAS, M. Damage Detection in Structures Using Changes in Flexibility. **Journal of Sound and Vibration**, v. 169, n. 1, p. 3–17, 1994. ISSN 0022460X. Available from: <<http://www.sciencedirect.com/science/article/pii/S0022460X84710029>>. 156

PANDEY, A.; BISWAS, M.; SAMMAN, M. **Damage detection from changes in curvature mode shapes**. 1991. 321–332 p. 156

PARK, G.; SOHN, H.; FARRAR, C. R.; INMAN, D. J. **Overview of Piezoelectric Impedance-Based Health Monitoring and Path Forward**. 2003. 451–463 p. 155

PARLOO, E.; GUILLAUME, P.; OVERMEIRE, M. van. Damage assessment using mode shape sensitivities. **Mechanical Systems and Signal Processing**,

v. 17, n. 3, p. 499–518, 2003. ISSN 08883270. Available from:

<<http://d.wanfangdata.com.cn/NSTLQK{ }NSTL{ }QK6932358.aspx>>. 156

PASCUAL, R.; GOLINVAL, J.-C.; RAZETO, M. A frequency domain correlation technique for model correlation and updating. In: SPIE INTERNATIONAL SOCIETY FOR OPTICAL. **Proceedings-SPIE The International Society for Optical Engineering**. [S.l.], 1997. p. 587–592. 155

_____. On-line Damage Assessment Using Operating Deflection Shapes. feb 1999. Available from: <<http://orbi.ulg.ac.be/handle/2268/19937>>. 155

PAWAR, P. M.; GANGULI, R. Genetic fuzzy system for online structural health monitoring of composite helicopter rotor blades. **Mechanical Systems and Signal Processing**, v. 21, n. 5, p. 2212–2236, jul 2007. ISSN 08883270. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0888327006001944>>. 12

PAWAR, P. M.; Venkatesulu Reddy, K.; GANGULI, R. Damage Detection in Beams using Spatial Fourier Analysis and Neural Networks. **Journal of Intelligent Material Systems and Structures**, v. 18, n. 4, p. 347–359, 2006. ISSN 1045-389X. 11

PRASAD, N. E.; WANHILL, R. **Aerospace Materials and Material Technologies**. [S.l.]: Springer, 2017. 4

RAHNAMAYAN, S.; TIZHOOSH, H. R.; SALAMA, M. Quasi-oppositional differential evolution. In: IEEE. **Evolutionary Computation, 2007. CEC 2007. IEEE Congress on**. [S.l.], 2007. p. 2229–2236. 59

RAIDL, G. R. A unified view on hybrid metaheuristics. In: SPRINGER. **International Workshop on Hybrid Metaheuristics**. [S.l.], 2006. p. 1–12. 51

RAIDL, G. R.; PUCHINGER, J.; BLUM, C. Metaheuristic hybrids. In: **Handbook of metaheuristics**. [S.l.]: Springer, 2010. p. 469–496. 51

RAO, A. R. M.; ANANDAKUMAR, G. Optimal placement of sensors for structural system identification and health monitoring using a hybrid swarm intelligence technique. **Smart materials and Structures**, IOP Publishing, v. 16, n. 6, p. 2658, 2007. 13

RATCLIFFE, C. Damage detection usign a modified laplacian operator on mode shape data. **Journal of Sound and Vibration**, v. 204, n. 3, p. 505–517, 1997. ISSN 0022460X. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0022460X97909615>>.

156

REYNOLDS, R. G. An introduction to cultural algorithms. In: ANNUAL CONFERENCE ON EVOLUTIONARY PROGRAMMING, 3., 1994, Singapore. **Proceedings...** [S.l.], 1994. p. 131–139. 50

RICCIARDI, A. P.; CANFIELD, R. A.; PATIL, M. J.; LINDSLEY, N. Nonlinear aeroelastic scaled-model design. **Journal of Aircraft**, American Institute of Aeronautics and Astronautics, v. 53, n. 1, p. 20–32, 2016. 115

RIOS-COELHO, A.; SACCO, W.; HENDERSON, N. A metropolis algorithm combined with hookejeeves local search method applied to global optimization.

Applied Mathematics and Computation, v. 217, n. 2, p. 843 – 853, 2010.

ISSN 0096-3003. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0096300310007125>>.

68

ROTHLAUF, F. **Design of modern heuristics: principles and application**. [S.l.]: Springer Science & Business Media, 2011. 19

RUCKA, M. DAMAGE DETECTION IN BEAMS USING WAVELET TRANSFORM ON HIGHER VIBRATION MODES. **Journal of Theoretical and Applied Mechanics**, v. 49, n. 2, p. 399–417, 2011. 155

RYTTER, A. **Vibrational Based Inspection of Civil Engineering Structures**. PhD Thesis (PhD) — Aalborg University, 1993. 6

SACCO, W. F.; OLIVEIRA, C. R. E. A new stochastic optimization algorithm based on a particle collision metaheuristic. In: WORLD CONGRESSES OF STRUCTURAL AND MULTIDISCIPLINARY OPTIMIZATION, 6., 2005, Rio de Janeiro. RJ, Brazil. **Proceedings...** [S.l.], 2005. 51, 52

SAMBATTI, S. B. M.; ANOCHI, J. A.; LUZ, E. F. P.; CARVALHO, A. R.; SHIGUEMORI, E. H.; CAMPOS VELHO, H. F. Automatic configuration for neural network applied to atmospheric temperature profile identification. In: **3rd International Conference on International Conference on Engineering**

Optimization. [S.l.: s.n.], 2012. p. 1–9. 54

SAMPAIO, R.; MAIA, N. A Simple Correlation Factor as an Effective Tool for Detecting Damage. In: **Modeling, Simulation and Control of Nonlinear Engineering Dynamical Systems**. Dordrecht: Springer Netherlands, 2009. p. 233–242. Available from:

<http://link.springer.com/10.1007/978-1-4020-8778-3_{_}21>. 155

SAMPAIO, R.; MAIA, N.; SILVA, J. Damage detection using the frequency-response-function curvature method. **Journal of Sound and Vibration**, v. 226, n. 5, p. 1029–1042, 1999. ISSN 0022460X. Available from:

<<http://linkinghub.elsevier.com/retrieve/pii/S0022460X99923404>>. 155

_____. The frequency domain assurance criterion as a tool for damage detection. In: TRANS TECH PUBL. **Key Engineering Materials**. [S.l.], 2003. v. 245, p. 69–76. 155

SANDESH, S.; SHANKAR, K. Application of a hybrid of particle swarm and genetic algorithm for structural damage detection. **Inverse Problems in Science and Engineering**, Taylor & Francis Group, v. 18, n. 7, p. 997–1021, oct 2010. ISSN 1741-5977. Available from: <<http://www.tandfonline.com/doi/abs/10.1080/17415977.2010.500381{#}.VbgHBq0VhBd>>. 12

SCHALKOFF, R. J. **Pattern recognition**. [S.l.]: Wiley Online Library, 1992. 8

SEYEDPOOR, S. A two stage method for structural damage detection using a modal strain energy based index and particle swarm optimization. **International Journal of Non-Linear Mechanics**, v. 47, n. 1, p. 1–8, jan 2012. ISSN 00207462. Available from:

<<http://www.sciencedirect.com/science/article/pii/S0020746211001818>>. 12, 156

SEYEDPOOR, S.; SHAHBANDEH, S.; YAZDANPANA, O. An efficient method for structural damage detection using a differential evolution algorithm-based optimisation approach. **Civil Engineering and Environmental Systems**, p. 1–21, jun 2015. ISSN 1028-6608. Available from:

<<http://www.tandfonline.com/doi/full/10.1080/10286608.2015.1046051>>. 12

SEYEDPOOR, S. M.; YAZDANPANA, O. Structural Damage Detection by Differential Evolution as a Global Optimization Algorithm. **Iranian Journal of**

Structural Engineering, v. 1, n. MAY, 2015. 12

SHAH-HOSSEINI, H. The intelligent water drops algorithm: a nature-inspired swarm-based optimization algorithm. **International Journal of Bio-Inspired Computation**, Inderscience Publishers, v. 1, n. 1-2, p. 71–79, 2009. 51

SHI, Z. Y.; LAW, S. S.; ZHANG, L. M. Structural Damage Localization from Modal Strain Energy Change. **Journal of Sound and Vibration**, v. 218, n. 5, p. 825–844, 1998. ISSN 0022460X. 156

SIARRY, P. (Ed.). **Metaheuristics**. Springer International Publishing, 2016. Available from: <<https://doi.org/10.1007%2F978-3-319-45403-0>>. 47, 48

SILVA NETO, A. J.; MOURA NETO, F. D. **Problemas inversos: conceitos fundamentais e aplicações**. [S.l.]: EdUERJ, 2005. 22

SILVA, S.; Dias Júnior, M.; Lopes Junior, V.; BRENNAN, M. J. Structural damage detection by fuzzy clustering. **Mechanical Systems and Signal Processing**, v. 22, n. 7, p. 1636–1649, oct 2008. ISSN 08883270. Available from: <<http://www.sciencedirect.com/science/article/pii/S0888327008000034>>. 12, 155

SIMON, D. Biogeography-based optimization. **IEEE transactions on evolutionary computation**, IEEE, v. 12, n. 6, p. 702–713, 2008. 50

_____. **Evolutionary optimization algorithms**. [S.l.]: John Wiley & Sons, 2013. 64

SINHA, A. T.; LEE, J.; LI, S.; BARBASTATHIS, G. Solving inverse problems using residual neural networks. In: **Digital Holography and Three-Dimensional Imaging**. Optical Society of America, 2017. p. W1A.3. Available from: <<http://www.osapublishing.org/abstract.cfm?URI=DH-2017-W1A.3>>. 23

SMITH, C.; AKUJUOBI, C. M.; HAMORY, P.; KLOESEL, K. An approach to vibration analysis using wavelets in an application of aircraft health monitoring. **Mechanical Systems and Signal Processing**, v. 21, n. 3, p. 1255–1272, 2007. ISSN 08883270. 155

SOHN, H.; FARRAR, C. R.; HEMEZ, F.; CZARNECKI, J. A Review of Structural Health Monitoring Literature 1996 – 2001. In: **Third World**

Conference on Structural Control. [S.l.: s.n.], 2004. p. 1–7. 6, 10

SOHN, H.; FARRAR, C. R.; HUNTER, N. F.; WORDEN, K. Structural health monitoring using statistical pattern recognition techniques. **Journal of dynamic systems, measurement, and control**, v. 123, n. 4, p. 706—711, 2001. Available from: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.2259>>. 155

SORENSEN, K.; SEVAUX, M.; GLOVER, F. A history of metaheuristics. **arXiv preprint arXiv:1704.00853**, 2017. 50

SOTERRONI, A. C.; GALSKI, R. L.; RAMOS, F. M. The q-gradient vector for unconstrained continuous optimization problems. In: HU, B.; MORASCH, K.; PICKL, S.; SIEGLE, M. (Ed.). **Operations Research Proceedings 2010**. Springer Berlin Heidelberg, 2011, (Operations Research Proceedings). p. 365–370. ISBN 978-3-642-20008-3. Available from: <http://dx.doi.org/10.1007/978-3-642-20009-0_58>. 65, 66

_____. The q -gradient method for global optimization. **arXiv:1209.2084**, math.OC, 2012. 65, 66

_____. The q -gradient method for continuous global optimization. In: Simos, T.; Psihoyios, G.; Tsitouras, C. (Ed.). **American Institute of Physics Conference Series**. [S.l.: s.n.], 2013. (American Institute of Physics Conference Series, v. 1558), p. 2389–2393. 65, 66

STASZEWSKI, W.; MAHZAN, S.; TRAYNOR, R. Health monitoring of aerospace composite structures – Active and passive approach. **Composites Science and Technology**, v. 69, n. 11-12, p. 1678–1685, sep 2009. ISSN 02663538. Available from: <<http://www.sciencedirect.com/science/article/pii/S0266353808003539>>. 10

STUBBS, N.; FARRAR, C. Field verification of a nondestructive damage localization and severity estimation algorithm. In: INTERNATIONAL MODAL ANALYSIS CONFERENCE (IMAC XIII), 13., 1995. **Proceedings...** [S.l.], 1995. p. 210–218. ISBN 0361-0748. 156

SULAIMAN, M. S. A.; YUNUS, M. A.; BAHARI, A. R.; RANI, M. N. A. Identification of damage based on frequency response function (frf) data. **MATEC**

Web Conf., v. 90, p. 01025, 2017. Available from:

<<https://doi.org/10.1051/mateconf/20179001025>>. 155

TABRIZIAN, Z.; AFSHARI, E.; AMIRI, G. G.; Ali Beigy, M. H.; NEJAD, S. M. P. A new damage detection method: Big Bang-Big Crunch (BB-BC) algorithm. **Shock and Vibration**, v. 20, n. 4, p. 633–648, 2013. ISSN 10709622. 12, 13

TAHA, M. M. R. Wavelet Transform for Structural Health Monitoring: A Compendium of Uses and Features. **Structural Health Monitoring**, v. 5, n. 3, p. 267–295, sep 2006. ISSN 1475-9217. Available from: <<http://shm.sagepub.com/content/5/3/267.abstract>>. 155

TALBI, E.-G. A taxonomy of hybrid metaheuristics. **Journal of heuristics**, Kluwer Academic Publishers, v. 8, n. 5, p. 541–564, 2002. 51

TARANTOLA, A. **Inverse problem theory and methods for model parameter estimation**. [S.l.]: SIAM, 2005. 23

TING, T.; YANG, X.-S.; CHENG, S.; HUANG, K. Hybrid metaheuristic algorithms: Past, present, and future. In: YANG, X.-S. (Ed.). **Recent Advances in Swarm Intelligence and Evolutionary Computation**. [S.l.]: Springer International Publishing, 2015, (Studies in Computational Intelligence, v. 585). p. 71–83. ISBN 978-3-319-13825-1. 52

TIZHOOSH, H.; VENTRESCA, M. **Oppositional Concepts in Computational Intelligence**. Springer Berlin Heidelberg, 2008. (Studies in Computational Intelligence). ISBN 9783540708292. Available from: <<https://books.google.com.br/books?id=-35rCQAAQBAJ>>. 59

TIZHOOSH, H. R. Opposition-Based Learning: A New Scheme for Machine Intelligence. In: **International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)**. IEEE, 2005. v. 1, p. 695–701. ISBN 0-7695-2504-0. Available from: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1631345>>. 58, 63

VENKAYYA, V. Design of optimum structures. **Computers & Structures**, v. 1, n. 1, p. 265 – 309, 1971. ISSN 0045-7949. Available from:

<<http://www.sciencedirect.com/science/article/pii/0045794971900137>>. 115

VENTERINK, M. **Frequency optimisation for damage identification using the Vibro-Acoustic Modulation method**. Master Thesis (Mestrado) — University of Twente, 2017. 4

WAHAB, M. A.; ROECK, G. D. Damage detection in bridges using modal curvatures: Application to a real damage scenario. **Journal of Sound and Vibration**, v. 226, n. 2, p. 217–235, 1999. ISSN 0022460X. Available from: <<http://www.sciencedirect.com/science/article/pii/S0022460X99922952>>. 156

WAHL, F.; SCHMIDT, G.; FORRAI, L. On the significance of antiresonance frequencies in experimental structural analysis. **Journal of Sound and Vibration**, Elsevier, v. 219, n. 3, p. 379–394, 1999. 156

WANG, J.; QIAO, P. **Improved Damage Detection for Beam-type Structures using a Uniform Load Surface**. 2007. 99–110 p. 156

WOLPERT, D. H.; MACREADY, W. G. No free lunch theorems for optimization. **IEEE transactions on evolutionary computation**, IEEE, v. 1, n. 1, p. 67–82, 1997. 49

WORDEN, K.; DULIEU-BARTON, J. M. **An Overview of Intelligent Fault Detection in Systems and Structures**. 2004. 85–98 p. 6

WORDEN, K.; FARRAR, C. R.; HAYWOOD, J.; TODD, M. A review of nonlinear dynamics applications to structural health monitoring. **Structural Control and Health Monitoring**, v. 15, n. 4, p. 540–567, jun 2008. ISSN 15452255. Available from: <<http://doi.wiley.com/10.1002/stc.215>>. 10

WORDEN, K.; FARRAR, C. R.; MANSON, G.; PARK, G. The fundamental axioms of structural health monitoring. v. 463, n. 2082, p. 1639–1664, 2007. ISSN 1364-5021. 6, 7

WU, D.; LAW, S. S. Damage localization in plate structures from uniform load surface curvature. **Journal of Sound and Vibration**, v. 276, n. 1-2, p. 227–244, 2004. ISSN 0022460X. 156

XU, Q.; WANG, L.; WANG, N.; HEI, X.; ZHAO, L. A review of opposition-based

learning from 2005 to 2012. **Engineering Applications of Artificial Intelligence**, Elsevier, v. 29, p. 1–12, 2014. 59

YAM, L.; YAN, Y.; JIANG, J. Vibration-based damage detection for composite structures using wavelet transform and neural network identification. **Composite Structures**, v. 60, n. 4, p. 403–412, jun 2003. ISSN 02638223. Available from: <http://www.sciencedirect.com/science/article/pii/S0263822303000230>>. 11

YANG, M.; BROWN, D. An improved procedure for handling damping during finite element model updating. In: **International Modal Analysis Conference(IMAC)**, 14 th, Dearborn, MI. [S.l.: s.n.], 1996. p. 576–584. 92

YANG, X.-S. **Nature-inspired metaheuristic algorithms**. [S.l.]: Luniver press, 2010. 49

_____. **Nature-Inspired Optimization Algorithms**. [s.n.], 2014. 99–110 p. ISSN 0302-9743. ISBN 9780124167438. Available from: <http://www.sciencedirect.com/science/article/pii/B9780124167438000075>>. 49

YU, L.; LI, C. A Global Artificial Fish Swarm Algorithm for Structural Damage Detection. **Advances in Structural Engineering**, v. 17, n. 3, p. 331–346, mar 2014. ISSN 1369-4332. Available from: <http://multi-science.atypon.com/doi/10.1260/1369-4332.17.3.331>>. 12, 13

YU, L.; WAN, Z.-y. An Improved PSO Algorithm and Its Application to Structural Damage Detection. In: **2008 Fourth International Conference on Natural Computation**. IEEE, 2008. v. 1, p. 423–427. ISBN 978-0-7695-3304-9. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4666881>>. 12

YU, L.; XU, P. Structural health monitoring based on continuous ACO method. **Microelectronics Reliability**, v. 51, n. 2, p. 270–278, feb 2011. ISSN 00262714. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0026271410005019>>. 11, 12, 13

YU, L.; XU, P.; CHEN, X. A SI-Based Algorithm for Structural Damage Detection. In: **Advances in Swarm Intelligence**. [s.n.], 2012. p. 21–28. Available from: http://link.springer.com/10.1007/978-3-642-30976-2_{_}3>. 12

ZANG, C.; FRISWELL, M. I.; IMREGUN, M. Structural health monitoring and damage assessment using frequency response correlation criteria. **Journal of Engineering Mechanics-Asce**, v. 133, n. 9, p. 981–993, 2007. 155

ZANG, C.; IMREGUN, M. Structural damage detection using artificial neural networks and measured frf data reduced via principal component projection. **Journal of Sound and Vibration**, v. 242, n. 5, p. 813–827, may 2001. ISSN 0022460X. Available from:
<<http://linkinghub.elsevier.com/retrieve/pii/S0022460X0093390X>>. 11

ZHANG, Z.; AKTAN, A. E. Application of Modal Flexibility and Its Derivatives in Structural Identification. **Research in Nondestructive Evaluation**, v. 10, n. 1, p. 43–61, 1998. ISSN 0934-9847. Available from:
<<http://www.springerlink.com/index/10.1007/PL00003899>>. 156

ZHU, J.; HUANG, M.; LU, Z. Bird mating optimizer for structural damage detection using a hybrid objective function. **Swarm and Evolutionary Computation**, p. –, 2017. ISSN 2210-6502. Available from:
<<http://www.sciencedirect.com/science/article/pii/S2210650216302978>>. 13

ZOU, Y.; TONG, L.; STEVEN, G. Vibration-based model-dependent damage (delamination) identification and health monitoring for composite structures – a review. **Journal of Sound and Vibration**, v. 230, n. 2, p. 357–378, feb 2000. ISSN 0022460X. Available from:
<<http://linkinghub.elsevier.com/retrieve/pii/S0022460X9992624X><http://www.sciencedirect.com/science/article/pii/S0022460X9992624X>>. 10

FEATURES AND CLASSIFICATION APPROACHES FOR VIBRATION-BASED DAMAGE IDENTIFICATION

Table A.1 - Literature overview of features and classification approaches for vibration-based damage identification

Time domain	
Time response / waveform	
Statistical time series analysis	
Correlation functions (COR)	(FASSOIS; SAKELLARIOU, 2007)
Autoregressive models (AR)	(SOHN et al., 2001)
	(FASSOIS; SAKELLARIOU, 2007)
Auto-Regressive Moving Average (ARMA)	(SILVA et al., 2008)
Vector-dependent functionally pooled (VFP)	(KOPSAFTOPOULOS; FASSOIS, 2006)
Time-frequency analysis	
Wavelet transform based methods (WA)	(TAHA, 2006)
	(SMITH et al., 2007)
	(RUCKA, 2011)
	(MAHDAVI; Abdul Razak, 2013)
	(MONTANARI et al., 2015)
Hilbert(-Huang) transform based methods (HT/HHT)	(HUANG et al., 1998)
	(CHEN et al., 2014)
Empirical mode decomposition (EMD)	(HUANG et al., 1998)
Conjugate-pair decomposition (CPD)	(PAI, 2011)
Frequency domain	
Fourier / power spectra	
Fourier coefficients (FC)	(MORASSI, 2007)
Frequency response function / operational deflection shapes	
Frequency response function change (FRFCH)	(KESSLER, 2002)
	(SULAIMAN et al., 2017)
Frequency response function damage index (FRFDI)	(BANERJEE et al., 2009)
Frequency response function shape method (FRFSHP)	(LIU et al., 2009)
Detection and relative damage quantification indicator (DRQ)	(SAMPAIO; MAIA, 2009)
Frequency response assurance criterion (FRAC)	(NEFSKE; SUNG, 1996)
Coordinate orthogonality check (CORTHOG)	(AVITABILE; PECHINSKY, 1994)
Complex correlation coefficient (CCF)	(AUWERAER et al., 1999)
Modal FRF assurance criterion (MFAC)	(FOTSCH; EWINS, 2001)
Frequency domain assurance criterion (FDAC)	(PASCUAL et al., 1997)
Response vector assurance criterion (RVAC)	(SAMPAIO et al., 2003)
Operational deflection shape change (ODSC)	(PASCUAL et al., 1999)
Global shape correlation (GSC)	(ZANG et al., 2007)
Global amplitude correlation (GAC)	(ZANG et al., 2007)
Frequency response function curvature	
Frequency response curvature (FRFC)	(SAMPAIO et al., 1999)
Strain frequency response damage index (SFRFDI)	(MAIA et al., 2003)
Mechanical impedance (Z)	
Root mean square deviation (RMSD)	(PARK et al., 2003)
Mean absolute percentage deviation (MAPD)	(GIURGIUTIU, 2005)
Correlation coefficient deviation (CCD)	(GIURGIUTIU, 2005)
Transmissibility function (T)	
Transmissibility damage index (TDI)	(JOHNSON; ADAMS, 2002)
Antiresonances	

Continued on next page

Table A.1 – Continued from previous page

Antiresonance shift (AFN)	(WAHL et al., 1999) (DILENA; MORASSI, 2004)
Higher harmonics (nonlinear effect)	
Higher harmonic imaging (HH)	(KROHN et al., 2002)
Modulations (nonlinear effect)	
Vibro-acoustic modulations (VAM)	(DONSKOY; SUTIN, 1998)
Modal domain	
Natural frequencies	
Cawley-Adams criteria (CAC)	(CAWLEY; ADAMS, 1979)
Damage location assurance criteria (DLAC)	(MESSINA et al., 1998)
Multiple damage location assurance criteria (MDLAC)	(MESSINA et al., 1998)
Mode shape	
Mode shape rotation change (MSRC)	(ABDO; HORI, 2002)
Mode shape amplitude change (MS)	(MAIA et al., 2003)
Squared mode shape slope change (MSS)	(MAIA et al., 2003)
Modal assurance criteria (MAC)	(PARLOO et al., 2003)
Inverse modal assurance criteria (IMAC)	(ALLEMANG, 2003)
Co-ordinate modal assurance criteria (COMAC)	(PARLOO et al., 2003)
Relative mode shape change (RMS)	(KHOO et al., 2004)
Fractal dimension method (FD)	(HADJILEONTIADIS et al., 2005)
Generalized fractal dimension (GFD)	(WANG; QIAO, 2007)
Mode shape curvature	
Mode shape curvature method (MSC)	(PANDEY et al., 1991) (CHANDRASHEKHAR; GANGULI, 2009)
Modal strain energy damage index method (MSEDI)	(STUBBS; FARRAR, 1995) (SEYEDPOOR, 2012)
Gapped smoothing method (GSM)	(RATCLIFFE, 1997)
Modal strain energy change ratio (MSECR)	(SHI et al., 1998)
Curvature damage factor method (CDF)	(WAHAB; ROECK, 1999)
Mode shape curvature squared method (MSC)	(MAIA et al., 2003)
Modal damping	
Specific damping capacity (SDC)	(GUILD, 1981)
Dynamic stiffness (combination of modal parameters)	
Effective stiffness change (ESC)	(KHOO et al., 2004)
Dynamic flexibility (combination of modal parameters)	
Modal flexibility method (MFL)	(PANDEY; BISWAS, 1994)
Uniform load surface method (ULS)	(ZHANG; AKTAN, 1998)
Modal flexibility curvature method (MFLC)	(LU et al., 2002)
Uniform load surface curvature (ULSC)	(WU; LAW, 2004)
Modal compliance index method (MCI)	(CHOI et al., 2005)
Updating methods	
Modal flexibility residual (MFR)	(JAISHI; REN, 2006)
Other methods	
Modal peak density (MD)	(CHRYSOCHOIDIS; SARAVANOS, 2004)
Modal filters (MF)	(DERAEMAERKER; PREUMONT, 2006)

SOURCE: Adapted from (OOIJEVAAR, 2014; MEDEIROS, 2016)

USING NASTRAN FOR TRANSIENT RESPONSE ANALYSIS

NASTRAN (NASA Structural Analysis System) is a potent general purpose finite element analysis program for use in computer-aided engineering ([MSC, 2001](#); [MSC, 2014](#)).

Originally, NASTRAN was developed by National Aeronautics and Space Administration (NASA) in the mid-1960. The MacNeal-Schendler Corporation (MSC) was one of the principal developers of the public domain code.

B.1 NASTRAN files

The most important interface files in NASTRAN are the input and output files. Default extensions for the input file are: **dat**, **nas**, or **bdf**. When the structural analysis is performed, additional files are created. Most common output files extensions are ([MSC, 2001](#); [MSC, 2014](#)):

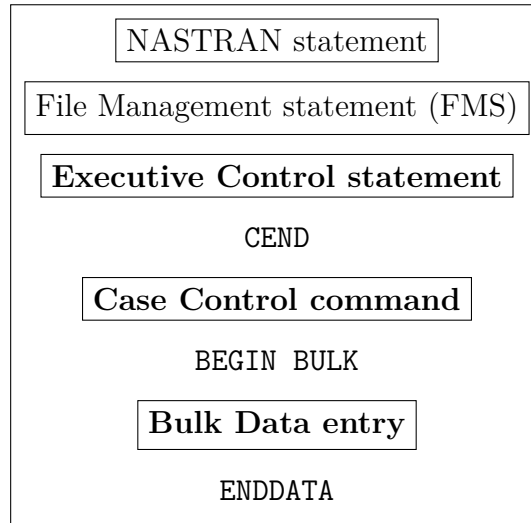
- f04** Contains the start and stop time for each module executed as well as the size of the database file;
- f06** Contains the results of the analysis, such as the displacements, stresses, etc., as well as any diagnostic messages;
- log** Contains system information, such as the name of the computer in which the program is running on, and any system error encountered.

B.1.1 Input file

The NASTRAN input file contains a complete description of the finite element model, including the type of analysis to be performed, the model geometry, the definition of the type and material of the finite elements, and the boundary conditions.

The input file is comprised of five distinct sections (NASTRAN Statement (optional), File Management Section (optional), Executive Control Section, Case Control Section and Bulk Data Section), and three required one-line delimiter (**CEND**, **BEGIN BULK** and **ENDDATA**), as shown in [Figure B.1](#). The three obligatory sections are briefly described below:

Figure B.1 - Structure of the NASTRAN Input file. The required sections are represented in bold.



Executive Control Section This section provides the overall control for the solution. In this section, the following options can be specified: an optional ID statement to identify a job, the type of analysis to be performed, an optional **TIME** statement to set limits on the maximum allowable CPU time for the run, and various diagnostics requested. The **CEND** delimiter ends this section.

Case Control Section This section follows the Executive Control section and precedes the Bulk Data section, separated by the **BEGIN BULK** delimiter. It contains commands that allow to:

- define/select analysis sub-cases (multiple loadings in a single job execution);
- define **SETS** to specify and control the type of analysis output produced (e.g., forces, stresses, and displacements);
- select output requests for printing, punching and plotting;
- define titles, subtitles and labels for documenting the analysis.

Bulk Data Section This section starts just after the **BEGIN BULK** delimiter. The Bulk Data Section constitutes the majority of the content of the input file and contains all the information necessary to define the finite element model, such as geometry, coordinate systems, finite elements,

element properties, loads, boundary conditions, material properties and so on.

B.1.2 Output files

The most common output file is the **f06** file. NASTRAN writes this file to the unit 6 of FORTRAN. The **f06** file contains all the requested results from the analysis, such as displacements, stresses and element forces, as well as any diagnostic messages. This file can be used in the process of model checkout and debugging. Also, NASTRAN can create a punch file (**pch**) that contains the analysis results in a fixed-length text table format.

The output from data recovery and plot modules is optional and controlled by commands in the Case Control section.

B.2 Using NASTRAN for defining structures

NASTRAN bulk data entries may have different formats: a fixed field format with ten fields per line and eight characters per field, or a free field format, with periods separating the fields.

Table B.1 - Some commonly used elastic elements in NASTRAN

Dimensionality	Entry	Use
0D	CELAS	Simple spring connecting 2-DOF
	CBUSH	Frequency-dependent spring/damper connecting up to 6-DOF
1D	ROD	Pin-ended rod
	BAR	Prismatic beam
	BEAM	Straight beam with warping
	BEND	Curved beam, pipe or elbow
2D	TRIA3 / TRIAR	Triangular plate
	QUAD4 / QUADR	Quadrilateral plate
	SHEAR	Four sided shear panel

B.2.1 Setting the structures geometry

The **GRID** bulk data entry is used to define a grid point.

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
GRID   ID      CP      X1      X2      X3      CD      PS      SEID
```

in which **ID** is the grid point identification number; **CP** is the identification number of coordinate system in which the location of the grid point is defined; **Xi** specify the

location of the grid point in coordinate system, **CD** is the identification number of coordinate system in which the displacements, degrees of freedom, constraints, and solution vectors are defined at the grid point; **PS** is the permanent single-point constraints associated with the grid point, and **SEID** is the super-element identification number.

Another entry, **GRDSET**, is useful, for example, in the case of such problems as trusses where is needed to remove all the rotational DOF. This entry is set as follow:

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
GRDSET          CP                      CD      PS      SEID
```

in which **CP** is the identification number of coordinate system in which the location of the grid point is defined, **CD** is the identification number of coordinate system in which the displacements, degrees of freedom, constraints, and solution vectors are all defined at the grid point, **PS** is the permanent single-point constraints associated with grid point, and **SEID** set a superelement identification number.

B.2.2 Defining material properties

There are different ways of doing the mass input: using the **MATi** entries, which define the material density, using the scalar mass entries **CMASSi** and **PMASS**, or by using the grid point mass definition **CONM1** and **CONM2**.

For example, **MAT1** entry defines a homogeneous¹, isotropic² material using the following format:

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
MAT1      MID      E      G      NU      RHO      A      TREF      GE
```

where **MID** is the material identification number, **E** is the Young's modulus, **G** is the Shear modulus, **NU** is the Poisson's ratio, **RHO** is the Mass density, **A** is the coefficient of thermal expansion, **TREF** is reference temperature for the calculation of thermal loads, and **GE** is the structural element damping coefficient.

Other entry used in NASTRAN is the **CONM2** entry:

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
CONM2     EID      G      CID      M      X1      X2      X3
          I11      I21      I22      I31      I32      I33
```

where **EID** is the element identification number, **G** is the grid point identification

¹Homogeneous: The material is the same nature, regardless of location within the material.

²Isotropic: The material properties do not change with the direction of the material.

number, **CID** is the coordinate system identification number, **M** is the mass value, **X1**, **X2**, **X3** are the offset distances from the grid point to the center of gravity of the mass in the basic coordinate system, and **Iij** is the mass moment of inertia measured at the mass center of gravity in the coordinate system defined by field **CID**.

B.3 Using NASTRAN for transient response analysis

In NASTRAN, the **SOL 109** is used to perform direct transient response analysis. The **SOL 112** performs modal transient response analysis (NASTRAN, 2004).

B.3.1 Integration time step

The integration time step is defined by the **TSTEP** entry card. The time step must be small enough to represent both the variation in the loading and the maximum frequency of interest.

```
$-----2-----3-----4-----5-----6-----7-----8-----
TSTEP   SID      N1      DT1      NO1
          N2      DT2      NO2
          -etc-
```

In the card **TSTEP**, **SID** represents the set identification number, **Ni** are the number of time steps, **DTi** is the integration time step Δt , and **NOi** control the output rates, with output at every **NOi** time step.

B.3.2 Time-dependent loads

For defining force as a function of time, NASTRAN has two methods: **TLOAD1** and **TLOAD2**

TLOAD1 is a brute force method, that uses ordered time, load pairs in a table input $f(t)$, as follows:

$$F(t) = A \times f(t - \tau) . \quad (\text{B.1})$$

TLOAD1 entry is defined as follows:

```
$-----2-----3-----4-----5-----6-----7-----8-----
TLOAD1  SID      EXCITEDID DELAY  TYPE    TID      US0      VS0
```

In the card **TLOAD1**, **SID** represents the set identification number, **EXCITEDID** is the identification number of **DAREA** or **SPCD** entry set, **DELAY** is the time delay τ , **TYPE** defines the type of the dynamic excitation, **TID** is the identification number of **TABLEDi** entry that defines $f(t)$, and **US0** and **VS0** are the factor for initial displacements and

velocities of the enforced DOF, respectively.

TLOAD2 is an analytical-type load with time, defined a dynamic loading of the form:

$$P(t) = \begin{cases} 0 & x < (T1 + \tau) \\ A\tilde{t}^{-B}e^{C\tilde{t}}\cos(2\pi Ft + P) & (T1 + \tau) \leq t \leq (T2 + \tau) \\ 0 & x > (T2 + \tau) \end{cases} \quad (\text{B.2})$$

in which $\tilde{t} = t - T1 - \tau$, $T1$ and $T2$ are time constants ($T2 > T1$), F is the frequency in cycles per unit time, P is the phase angle in degrees, C is the exponential coefficient, and B is the growth coefficient.

TLOAD2 entry is formatted as follows:

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
TLOAD2  SID      EXCITEDID DELAY  TYPE      T1        T2        F        P
          C        B          US0    VS0
```

In the card TLOAD2, SID, EXCITEDID, DELAY, TYPE, US0 and VS0 have the same definition that in TLOAD1.

B.3.3 Dynamic Load Tabular Function

There exists four types of TABLEDi entries, which define a tabular function for generating time-dependent dynamic loads.

The TABLED1 entry can perform linear or logarithmic interpolations between points. The format for this entry has the following form:

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
TABLED1 TID      XAXIS    YAXIS
          X1      Y1      X2      Y2      -etc-  ENDT
```

where TID is the identification number of the table; XAXIS and YAXIS specify a linear or logarithmic interpolation for the x -axis and the y -axis, respectively; and x_i and y_i are the tabular values, in which the values of x are frequency in cycles per unit time. The term ENDT ends the table input.

B.3.4 Spatial distribution of loading

The DAREA card defines the spatial distribution of a dynamic loading, using the following format:

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
```

DAREA SID P1 C1 A1 P2 C2 A2

in which **SID** is the identification number, **Pi** is the ID of the grid, extra, or scalar point, **Ci** is the component number, and **Ai** is the scale factor.

B.3.5 Initial conditions

For imposing the initial conditions of displacement and velocity, the **TIC** bulk data entry is used, and selected by the **IC** case control command.

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
TIC      SID
```

B.3.6 Single-point constraints

A single point constraint applies a fixed value to a translational or rotational component at an individual grid point or to a scalar point. This restriction is used for fixing a structure to the ground, or for removing some DOF that are not utilized in the structural analysis, for example.

The **SPC** entry format is defined as follows:

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
SPC      SID      G1      C1      D1      G2      C2      D2
```

where **SID** is the identification number of the single-point constraint set, **Gi** is the grid or scalar point identification number, **Ci** is the component number, and **Di** is the value of enforced displacement for all degrees of freedom that were designated.

B.3.7 Damping

The damping in NASTRAN can be modeled using different entries. For viscous damping, the cards **CVISC**, **CDAMPi** and **CBUSH** are used. The way for defining structural damping varies from element to element, and it is uniform for the entire model.

For establishing damping versus frequency in modal transient response, the **TABDMP1** entry is used. In the case control, the **SDAMPING** command will set the damping.

```
$-----2-----3-----4-----5-----6-----7-----8-----9-----
TABDMP1 ID      TYPE
          F1      G1      F2      G2      F3      G3      -etc-  ENDT
```


VITA

MSc. Eng. Reynier Hernández Torres

Born in Havana, Cuba, on February 2nd, 1984.

ORCID: 0000-0003-1554-5145

Education

Engineer in Automation (Ingeniero en Automática, Instituto Superior Politécnico José Antonio Echeverría, La Habana, Cuba) 2002 - 2007

University specialist in Automation and Control (Especialista Universitario en Automatización y Control, Universidad Politécnica de Valencia, Valencia, España) 2011

Master in Industrial Informatic and Automation (Master en Informática Industrial y Automatización, Instituto Superior Politécnico José Antonio Echeverría, La Habana, Cuba) 2008 - 2011

Professional experience

Instructor

Departamento de Automática y Computación

Instituto Superior Politécnico José Antonio Echeverría, La Habana, Cuba

2007 – 2010

Assistant

Departamento de Automática y Computación

Instituto Superior Politécnico José Antonio Echeverría, La Habana, Cuba

2010 – 2013

Scientific Publications

FULL PAPERS AT SCIENTIFIC CONFERENCES

Not related to the thesis

Conference: 7th International Conference on Inverse Problems in Engineering: Theory and Practice

City: Orlando, Florida, USA

Date: May 2011

Title: Using of metaheuristics methods for solving inverse problems in liquid

chromatography

Conference: *Congresso de Matemática Aplicada e Computacional CMAC-Nordeste 2012*

City: Natal, RN - Brazil

Date: November 2012

Title: Parameter estimation of liquid chromatography model using quasi-oppositional differential evolution

Conference: Inverse Problems, Design and Optimization Symposium (IPDO 2013)

City: Albi, France

Date: June 2013

Title: Parameter Estimation of the Liquid Chromatography General Rate Model using different stochastic methods

Related to the thesis

Conference: 13th International Conference on Integral Methods in Science and Engineering – IMSE 2014

City: Karlsruhe, Germany

Date: July 2014

Title: Multi-Particle Collision Algorithm for solving an inverse radiative problem

Conference: XXXV Congresso Nacional de Matemática Aplicada e Computacional – CNMAC 2014

City: Natal, RN - Brazil

Date: September 2014

Title: Multi-Particle Collision Algorithm with Reflected Points

Conference: 10th International Conference on Composite Science and Technology – ICCST/10

City: Lisbon, Portugal

Date: September 2015

Title: Multi-Particle Collision Algorithm with Hooke-Jeeves for solving a Structural Damage Detection Problem

Conference: XXXVI Ibero-Latin American Congress on Computational Methods in Engineering – CILAMCE 2015

City: Rio de Janeiro, RJ - Brazil

Date: November 2015

Title: A Hybrid Method using q-Gradient to identify Structural Damage

Conference: *XXXVI Congresso Nacional de Matemática Aplicada e Computacional* – CNMAC 2016

City: Gramado, RS - Brazil

Date: September 2016

Title: Rotation-Based Multi-Particle Collision Algorithm with Hooke-Jeeves

Conference: 5th International Conference on Engineering Optimization – EngOpt 2016

City: Iguassu Falls, PR - Brazil

Date: June 2016

Title: Rotation-based sampling Multi-Particle Collision Algorithm with Hooke-Jeeves for Vibration-based Damage Identification

Conference: 4th Conference of Computational Interdisciplinary Science – CCIS 2016

City: São José dos Campos, SP - Brazil

Date: November 2016

Title: Vibration-based Damage Identification using the Multi-Particle Collision Algorithm with Hooke-Jeeves coupled with NASTRAN

Conference: 4th Conference of Computational Interdisciplinary Science – CCIS 2016

City: São José dos Campos, SP - Brazil

Date: November 2016

Title: Climate precipitation prediction using optimal neural network architecture in Southeast Region of Brazil

Conference: *XXXVII Congresso Nacional de Matemática Aplicada e Computacional* – CNMAC 2017

City: São José dos Campos, SP - Brazil

Date: September 2017

Title: Multi-Particle Collision Algorithm with Hooke Jeeves applied to the damage identification in the Kabe Problem

FULL PAPERS IN SCIENTIFIC JOURNALS

Not related to the thesis

Journal: *Revista Ingeniería Electrónica, Automática y Comunicaciones*
Title: Application of Genetic Algorithms for Parameter Estimation in Liquid Chromatography
Date: November 2011
DOI: 10.1234/rielac.v32i3.89
ISSN: 1815-5928

Journal: *Ingeniare. Revista chilena de ingeniería*
Title: Comparison between different metaheuristic algorithms in parameter estimation of the general relational model of column liquid chromatography (in spanish)
Date: January 2014
DOI: 10.4067/S0718-33052014000100003
ISSN: 0718-3305

Related to the thesis

There are no papers published so far.

BOOK CHAPTERS

Book: Integral Methods in Science and Engineering - Theoretical and Computational Advances

Chapter: Multi-Particle Collision Algorithm for solving an inverse radiative problem

ISBN: 978-3-319-16727-5

URL: <http://www.springer.com/us/book/9783319167268>

Book: Computational Intelligence in Engineering Problems

Chapter: Rotation-Based Multi-Particle Collision Algorithm with Hooke-Jeeves Approach Applied to the Structural Damage Identification

ISBN: –

URL: –

Event organization

- *XIV Workshop de Computação Aplicada* (WorCAP 2014)
- *XV Workshop de Computação Aplicada* (WorCAP 2015)

- *16 Workshop de Computação Aplicada (WorCAP 2016)* coordinator
- *4th Conference of Computational Interdisciplinary Science (CCIS 2016)*

Skills

- Programming in C, C++, Fortran, MatLab, Simulink, LabView, Java, Android, VHDL, HTML, JavaScript, CSS, PHP
- Using of Eclipse/Netbeans/Codebreaks/Qt
- NASTRAN / PATRAN / AUTOCAD
- UNIX
- Experience with \LaTeX and *TikZ*

Languages

- Spanish – Native speaker
- Portuguese – Advanced speaker
- English – Advanced speaker
- French – Basic speaker

On the Web

https://www.researchgate.net/profile/Reynier_Hernandez_Torres

<http://br.linkedin.com/in/reynierhdez>

PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE

Teses e Dissertações (TDI)

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

Manuais Técnicos (MAN)

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

Notas Técnico-Científicas (NTC)

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programas de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

Relatórios de Pesquisa (RPQ)

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

Propostas e Relatórios de Projetos (PRP)

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

Publicações Didáticas (PUD)

Incluem apostilas, notas de aula e manuais didáticos.

Publicações Seriadas

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Constam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

Programas de Computador (PDC)

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. Aceitam-se tanto programas fonte quanto os executáveis.

Pré-publicações (PRE)

Todos os artigos publicados em periódicos, anais e como capítulos de livros.