UNIVERSITY OF CAMPINAS
School of Electrical and Computer Engineering

Yara Quilles Marinho

Application of Meta-heuristics to Tune Vibration Spectrum Analyzers based on Microelectromechanical Systems

Aplicação de Meta-heurísticas para Afinação de Analisadores de Espectro de Vibração baseados em Sistemas Microeletromecânicos

CAMPINAS
2020

Yara Quilles Marinho

Application of Metaheuristics to Tune Vibration Spectrum Analyzers based on Microelectromechanical Systems

Aplicação de Meta-heurísticas para Afinação de Analisadores de Espectro de Vibração baseados em Sistemas Microeletromecânicos

Dissertation presented to the School of Electrical and Computer Engineering of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Electrical Engineering, in the area of Automation.

Dissertação apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestra em Engenharia Elétrica, na área de Automação.

Supervisor/Orientador: Prof. Dr. Mateus Giesbrecht

Co-Supervisor/Coorientador: Prof. Dr. Fabiano Fruett

This copy corresponds to the final version of the dissertation defended by student Yara Quilles Marinho, supervised by Prof. Dr. Mateus Giesbrecht and co-supervised by Prof. Dr. Fabiano Fruett.

CAMPINAS
2020

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Luciana Pietrosanto Milla - CRB 8/8129

Marinho, Yara Quilles, 1993-
M338a Application of meta-heuristics to tune vibration spectrum analyzers based on microelectromechanical systems / Yara Quilles Marinho. – Campinas, SP : [s.n.], 2020.

Orientador: Mateus Giesbrecht.
Coorientador: Fabiano Fruett.
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Sistemas microeletromecânicos. 2. Otimização. 3. Evolução diferencial. I. Giesbrecht, Mateus, 1984-. II. Fruett, Fabiano, 1970-. III. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Aplicação de meta-heurísticas para afinação de analisadores de espectro de vibração baseados em sistemas microeletromecânicos
Palavras-chave em inglês:
Microelectromechanical systems
Optimization
Differential evolution
Área de concentração: Automação
Titulação: Mestra em Engenharia Elétrica
Banca examinadora:
Mateus Giesbrecht [Orientador]
Fernando José Von Zuben
João Paulo Pereira do Carmo
Edmond Cretu
Data de defesa: 20-02-2020
Programa de Pós-Graduação: Engenharia Elétrica

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0000-0002-2009-5743
- Currículo Lattes do autor: http://lattes.cnpq.br/0240034136563197

COMISSÃO JULGADORA - DISSERTAÇÃO DE MESTRADO

Candidata: Yara Quilles Marinho, RA: 210333

Data da Defesa: 20 de fevereiro de 2020

Título da Dissertação: Aplicação de Meta-heurísticas para Afinação de Analisadores de Espectro de Vibração baseados em Sistemas Microeletromecânicos

Dissertation Title: Application of Meta-heuristics to Tune Vibration Spectrum Analyzers based on Microelectromechanical Systems

Prof. Dr. Mateus Giesbrecht (Presidente)

Prof. Dr. Fernando José Von Zuben

Prof. Dr. João Paulo Pereira do Carmo

Prof. Dr. Edmond Cretu

A ata de defesa, com as respectivas assinaturas dos membros da Comissão Julgadora, encontra-se no SIGA (Sistema de Fluxo de Dissertação/Tese) e na Secretaria de Pós-Graduação da Faculdade de Engenharia Elétrica e de Computação.

To Moly and Lanna

Acknowledgments

According to the Merriam-Webster Dictionary [1], the adjective "thankful" – besides expressing gratitude – refers to the consciousness of benefit received. And, aware of all the privileges I have experienced so far, firstly I would like to express my gratitude to all those whose work and dedication are fundamental to the development of this work and to the functioning of this university – and who, perhaps, could not enjoy the same opportunities and the same comfort: porters, security guards, secretaries, cleaners, technicians, administrative staff, and so many others.

This study was financed in part by CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil. To CNPq, I thank for the financial support.

To my advisor, Prof. Dr. Mateus Giebrescht: for his interest, teaching – and also learning – how to deal with microelectromechanical systems; for his trust, allowing me to develop my own ideas; for his advice, guiding these ideas in the right direction; for his heartening, celebrating each advance; for his comprehension, regarding me as a person, I am grateful.

To my co-advisor, Prof. Dr. Fabiano Fruett, for believing in me since the first meeting and for presenting to me the device which was responsible for increasing my particular passion in micro-technology. His patience – explaining in detail the microelectromechanical systems operation principles – and his trust in my methodology were primordial to the development of this work.

To my coworker, Iuri Monteiro, whom I am happy to call a friend. For his help during all this process – from technical to personal issues. His comforting presence and encouragement were essential to move me forward.

Finally, the most important – and difficult – acknowledgment: to my mother, Sandra. She is directly responsible for everything. I can hardly find words and stop the tears. Without her, I would never have been able to pursue my Master degree at Unicamp. Without her, I would never have gone to professors' Fabiano and Mateus office and would never have worked with my passion. Without her, I would never have found a reason to keep working, living, and dreaming.

[1] "Thankful". The Merriam-Webster.com Dictionary, Merriam-Webster Inc., https://www.merriam-webster.com/dictionary/thankfulness.

Abstract

The mechanical vibration spectrum is a frequency-domain characteristic used for monitoring various systems and is traditionally calculated by the Fast Fourier Transform (FFT) of a time series. Another possible alternative, with several operational advantages, is the use of twin-microaccelerometers to obtain the spectrum directly in the frequency domain. This strategy has its greatest limitation in the differences found in the accelerometers physical parameters – due to their manufacturing process –, such that the spectrum distortion level may be considerably higher than that found in the spectrum raised by the FFT. To overcome these differences, in this work the tuning of the spectrum analyzer microdevice is proposed by adjusting the accelerometers actuation voltages amplitudes. To perform the tuning, the Differential Evolution (DE) is used and the tuning problem is approached in two different optimization perspectives: a mono-objective and a multi-objective. For both optimization problems, the objective functions and constraints are based on the Fourier series components of the spectrum analyzer system closed-loop gain – a composition that depends on the excitation voltages. To solve the multi-objective optimization problem, the DE algorithm is properly adapted. The advantages and disadvantages of both tuning strategies are discussed in detail, as well as the results obtained for the Pareto-set approximation. The results – specially the distortion-sensitivity compromise – are demonstrated and discussed. The validity of the proposed tuning strategy is evidenced, since it is able to determine the voltages amplitudes to be applied to the micro spectrum analyzer to attend the distortion level and sensitivity requirements.

Key words: Microelectromechanical systems; Optimization; Differential evolution.

Resumo

O espectro de vibração mecânica é uma característica do domínio da frequência utilizada para o monitoramento de sistemas diversos e é, tradicionalmente, calculado pela Transformada Rápida de Fourier (FFT) – do termo em inglês *Fast Fourier Transform* – de uma série temporal. Uma alternativa viável, com diversas vantagens operacionais, é o uso de microacelerômetros gêmeos para a obtenção do espectro diretamente no domínio da frequência. Essa estratégia possui sua maior limitação nas diferenças encontradas nos parâmetros físicos de acelerômetros – devidas a seu processo de fabricação –, de tal forma que o nível de distorção do espectro pode ser consideravelmente superior àquele encontrado no espectro levantado pela FFT. Para contornar essas diferenças, neste trabalho a afinação do microdispositivo analisador de espectro é proposta através do ajuste das amplitudes das tensões de atuação dos acelerômetros. Para realizar a afinação, a Evolução Diferencial (DE, do termo em inglês *Differential Evolution*) é usada e o problema da afinação é abordado sob duas diferentes perspectivas de otimização: uma mono-objetivo e uma multi-objetivo. Para ambos os problemas de otimização, as funções objetivo e restrições são baseadas nas componentes da série de Fourier do ganho de malha fechada do sistema analisador de espectro – composição essa que depende das tensões de excitação. Para a solução do problema de otimização multi-objetivo, o algoritmo DE é devidamente adaptado. As vantagens e desvantagens de ambas as estratégias de afinação são discutidas em detalhe, bem como os resultados obtidos para a aproximação do conjunto de Pareto. Esses resultados – especialmente o compromisso distorção-sensibilidade – são demonstrados e discutidos. A validade da estratégia de afinação proposta é evidenciada, uma vez que é capaz de determinar as amplitudes das tensões a serem aplicadas ao micro analisador de espectro para atender os requisitos de nível de distorção e sensibilidade.

Palavras-chave: Sistemas microeletromecânicos; Otimização; Evolução diferencial.

List of Figures

2.1 Flow chart for the implementation of the Pareto-dominance selection rule of Equation 2.20. . . . 35

3.1 Operating principle of the inverted pendulum inserted in a gravitational field. . . . 41
3.2 Schematic representation of the microaccelerometer with zero-displacement (a) and with a $x(t)$ displacement (b). . . . 41
3.3 Block diagram of MEMS quasi-static model for common-mode actuation. . . . 46
3.4 Block diagram of the mechanical spectrum analyzer differential structure. . . . 51

5.1 (a) Real external acceleration spectrum. Spectra obtained by the simulation of the MEMS analyzer with excitation voltages $V_1 = V_2 = 7.6$ V: (b) single accelerometer, (c) a non-ideal twin-structure, (d) ideal twin-accelerometers. . . . 63
5.2 Closed-loop gain analysis for (1) single-accelerometer, (2) a non-ideal twin-structure, (3) ideal twin-accelerometers. . . . 64
5.3 Mapping results for (1) single-accelerometer, (2) a non-ideal twin-structure, (3) ideal twin-accelerometers. . . . 67
5.4 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, bounce-back boundary strategy. . . . 68
5.5 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, bounce-back boundary strategy. . . . 69
5.6 Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, bounce-back boundary strategy, test 1. . . . 74
5.7 Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, bounce-back boundary strategy, test 2. . . . 75
5.8 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, bounce-back boundary strategy. . . . 75
5.9 Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, bounce-back boundary strategy. . . . 78
5.10 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, bounce-back boundary strategy. . . . 79
5.11 Tuning validation results for a non-ideal twin-structure – bounce-back boundary strategy. . . . 81

A.1 Continuous-time periodic square wave. . . . 100

A.2 Envelope of Fourier series coefficients for the periodic square wave for several values of T_0 with T_1 fixed. (a)$T_0 = 4T_1$; (b)$T_0 = 8T_1$; (c)$T_0 = 16T_1$. 100
A.3 Aperiodic signal $x(t)$ (a). Periodic signal $\tilde{x}(t)$ constructed to be equal $x(t)$ over one period, with period $T_0 = 4T_1$ (b) and $T_0 = 8T_1$ (c). 101
E.1 Mapping results for (1) to (10) non-ideal twin-accelerometers. 127
E.2 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, brick-wall boundary strategy. 128
E.3 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, brick-wall boundary strategy. 128
E.4 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, random reinitialization boundary strategy. 129
E.5 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, random reinitialization boundary strategy. 129
E.6 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, resampling boundary strategy. 130
E.7 Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, resampling boundary strategy. 130
E.8 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, bounce-back boundary strategy. 131
E.9 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, bounce-back boundary strategy. 132
E.10 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, brick-wall boundary strategy. 133
E.11 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, brick-wall boundary strategy. 134
E.12 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, random reinitialization boundary strategy. 135
E.13 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, random reinitialization boundary strategy. 136
E.14 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, resampling boundary strategy. 137
E.15 Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, resampling boundary strategy. 138
E.16 Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, brick-wall boundary strategy. 139
E.17 Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, random reinitialization boundary strategy. 139
E.18 Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, resampling boundary strategy. 140
E.19 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, brick-wall boundary strategy. 140

E.20 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, random reinitialization boundary strategy. 141
E.21 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, resampling boundary strategy. 141
E.22 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, bounce-back boundary strategy. 142
E.23 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, brick-wall boundary strategy. 143
E.24 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, random reinitialization boundary strategy. 144
E.25 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, resampling boundary strategy. 145
E.26 Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, brick-wall boundary strategy. 146
E.27 Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, random reinitialization boundary strategy. 146
E.28 Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, resampling boundary strategy. 147
E.29 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, brick-wall boundary strategy. 147
E.30 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, random reinitialization boundary strategy. 148
E.31 Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, resampling boundary strategy. 148
E.32 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, bounce-back boundary strategy. 149
E.33 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, brick-wall boundary strategy. 150
E.34 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, random reinitialization boundary strategy. 151
E.35 Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, resampling boundary strategy. 152
E.36 Tuning validation results for a non-ideal twin-structure – brick-wall boundary strategy.153
E.37 Tuning validation results for a non-ideal twin-structure – random reinitialization boundary strategy. 154
E.38 Tuning validation results for a non-ideal twin-structure – resampling boundary strategy. 155
E.39 Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – bounce-back boundary strategy. 156
E.40 Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, brick-wall boundary strategy. 157

E.41 Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – random reinitialization boundary strategy. 158
E.42 Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – resampling boundary strategy. 159
E.43 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, bounce-back boundary strategy. 160
E.44 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, brick-wall boundary strategy. 161
E.45 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, random reinitialization boundary strategy. 162
E.46 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, resampling boundary strategy. 163
E.47 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, bounce-back boundary strategy. 164
E.48 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, brick-wall boundary strategy. 165
E.49 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, random reinitialization boundary strategy. 166
E.50 Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, resampling boundary strategy. 167

List of Tables

4.1 Micro-accelerometer rated parameters. 56
4.2 References to the expressions for the closed-loop gain and the composition of the Fourier series for single-accelerometer, non-ideal twin-accelerometers, and ideal twin-accelerometers. 58

5.1 Distortion and sensitivity results for single-accelerometer, non-ideal twin-accelerometers, and ideal twin-accelerometers MEMS spectrum analyzers. 65
5.2 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – maximum distortion level 0.8. . 70
5.3 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – maximum distortion level 0.2. . 70
5.4 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, bounce-back boundary strategy. 72
5.5 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, bounce-back boundary strategy. 72
5.6 Average behavior of mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – maximum distortion level 0.8. 73
5.7 Average behavior of mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – maximum distortion level 0.2. 73
5.8 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – approach 1. 76
5.9 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, bounce-back boundary strategy. 77

5.10 Average behavior of multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – approach 1. 77
5.11 Average behavior of multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – approach 2. 79

E.1 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, brick-wall boundary strategy. . 121
E.2 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, brick-wall strategy. 121
E.3 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, random reinitialization boundary strategy. 122
E.4 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, random reinitialization boundary strategy. 122
E.5 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, resampling boundary strategy. 123
E.6 Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, resampling boundary strategy. 123
E.7 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, brick-wall boundary strategy. 123
E.8 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, random reinitialization boundary strategy. 124
E.9 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, resampling boundary strategy. 124
E.10 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – approach 2. 124
E.11 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, bounce-back boundary strategy. 125

E.12 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, brick-wall boundary strategy. 125
E.13 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, random reinitialization boundary strategy. 125
E.14 Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, resampling boundary strategy. 126

Contents

1 Introduction **18**

1.1 Contributions 20

1.2 Dissertation organization 21

2 Optimization and Meta-heuristics **23**

2.1 Optimization basic concepts 23

2.1.1 Multi-objective optimization 25

2.1.1.1 Weighted sum of objective functions 27

2.1.1.2 ε-Constraint method 27

2.2 Meta-heuristics 27

2.2.1 Differential Evolution 28

2.2.1.1 Differential Evolution for multi-objective optimization 34

3 Vibration Mapping and Monitoring **37**

3.1 Microinstrumentation 39

3.1.1 Real-time MEMS-based spectrum analyzer 40

3.1.1.1 Spectral extraction 47

4 Tuning: Proposal and Methodology **52**

4.1 Tuning: sensitivity and distortion optimization 52

4.1.1 Mono-objective tuning 55

4.1.2 Multi-objective tuning 55

4.2 Methodology 56

4.2.1 Simulation model validation methodology 57

4.2.2 Mapping methodology 57

4.2.3 Tuning methodology 58

4.2.3.1 Mono-objective tuning methodology 60

4.2.3.2 Multi-objective tuning methodology: approach 1 60

4.2.3.3 Multi-objective tuning methodology: approach 2 60

4.2.4 Tuning validation methodology 61

5 MEMS-based Spectrum Analyzer Simulation and Tuning **62**

5.1 Simulation model validation results 62

5.2 Mapping results 66
5.3 Tuning results 67
5.3.1 Mono-objective tuning results 68
5.3.2 Multi-objective tuning results: approach 1 73
5.3.3 Multi-objective tuning results: approach 2 78
5.4 Tuning validation results 80

6 Conclusions and Future Steps **83**

References **85**

Appendix A – Fourier Analysis **94**
Fourier Series 94
Fourier Transform 99

Appendix B – Trigonometric Identities **103**

Appendix C – A Brief Table of Integrals **104**

Appendix D – Closed-loop Gain Fourier Series Decomposition **105**
A Brief Discussion on the Integration Period 119

Appendix E – Complementary Results **121**

Chapter 1

Introduction

Vibration has been a subject of interest since 4000 B.C. when various civilizations – such as Chinese, Hindu, Japanese, and Egyptian – started to appreciate music, although their knowledge has not reached a science level. It was Pythagoras (582-507 B.C.), in the ancient Greek civilization, who first investigated musical sounds on a scientific basis. The concept of pitch was developed during Pythagoras' time, but the relation between pitch and frequency was not understood until the 16^{th} century, the time of Galileo Galilei [1].

Another vibration that has intrigued society for centuries was the one generated by earthquakes. In A.D. 132, a Chinese astronomer developed an instrument with the shape of a wine jar to measure earthquakes. The world's first seismograph was capable of indicating the earthquake occurrence direction by means of pendulums and levers located inside the jar [1]. Nowadays, the effects of earthquakes are important for civil engineering, and the diagnosis of structural safety conditions is performed by analyzing the variation of the building's natural frequency [2, 3].

Vibration can be understood as any motion that repeats itself periodically. The vibration theory involves the study of oscillatory motions and the forces associated with them [1]. Currently, vibration sensing and monitoring are important in a myriad of applications in science and technology – such as aeronautical, mechanical, civil, and electrical systems – and are particularly profitable in economically critical or life-threatening systems [4]. In electric machines, the presence of moderate vibrations is very common; however, when the amplitudes rise, these vibrations may become harmful, result in premature fatigue failure due to large dynamic stresses, and damage the machine's mechanical components due to inertial forces [5]. Furthermore, as far as human health is concerned, depending on the existing interaction, the vibrations may cause physical or auditory discomfort [1, 6].

Currently, the most used solution for sensing the mechanical vibration spectrum is an accelerometer – to acquire the data of a time series – and a processor – to compute the Fast Fourier Transform (FFT). The algorithm to compute the FFT can be implemented on a computer [7], on a Digital Signal Processor (DSP) [8, 9], or even on an embedded circuit using Field Programmable Gate Array (FPGA) [10, 11]. These solutions have some drawbacks in using the FFT algorithm:

since the spectrum calculation does not occur in real time, it is possible that some information will be lost [12]. Besides, the algorithm adds some unexpected spectral components that are not present in the monitored physical vibration. However, the main disadvantage of the FFT solution is exactly the need for a digital processor, which consumes power and has a size that prevents its use in several applications, such as in the medical area [13, 14], where microinstrumentation is applied.

Microinstruments are of great value for many other applications besides medicine, not only due to their small size and low manufacturing costs, but also due to the advantages related to their operation – such as minimizing energy consumption and increasing sensitivity. The fast development of silicon manufacturing technologies since the 1960s and the knowledge of its excellent mechanical properties – coupled with electrical and thermal versatility – led to the expansion of its use in new technologies, such as Microelectromechanical Systems (MEMS) [15]. MEMS can be used for spectral analysis by amplifying the vibration signal in a small band around the resonance frequency, eliminating other spectral components [16]. With this selective approach, the signal-to-noise ratio is improved in the resonance frequency of the microsensor, simplifying the signal conditioning circuit – eliminating the FFT algorithm –, as can be seen in several studies since the late 1990s [17, 18]. This strategy, however, is limited to applications with vibration frequencies coincident with the resonance frequency of the sensor.

To overcome this constraint, resonance frequency tuning mechanisms may be employed, selecting the resonance frequency of the micro-oscillator – and, consequently, the sensor sensitivity – through electrostatic force feedback. This consists in the application of a continuous voltage in the device, allowing the electronic modulation of its spring constant [19]. The electrostatic force feedback allows the extraction of several spectral lines simultaneously by using an array of oscillators, each one tuned to a specific frequency [16, 20]. In spite of allowing greater flexibility, this strategy is still limited to the monitoring of a reduced number of vibration frequencies – corresponding to the number of micro-oscillators used.

Extending the electrostatic force feedback concept, a microaccelerometer can be made selectively responsive to a narrow frequency component of mechanical vibration by applying an alternating exciting signal to the device. As a result, the vibration components are sequentially obtained by varying the frequency of the driving signal. In this way, an electronically controlled spectral filtering can be performed, obtaining a real-time mechanical vibration spectrum analyzer, dispensing the FFT, through nonlinear electromechanical positive feedback. This was first proposed by Cretu, Bartek, and Wolffenbuttel [21], using twin-microaccelerometers with two inverted clamped pendulum capacitive structures and common-mode actuation voltages to perform electrostatic momentum feedback and obtain an output signal that represents the correlation between the external acceleration, $a_{ext}(t)$, and a cosine function, $\cos(\omega_d t)$. The working principle is general and can be applied to any microaccelerometer topology, as long as electrostatic driving is possible [19].

The combination of the capacitive transducer and the low-power reading circuitry has the potential for lower power consumption compared to a digital system running an FFT

algorithm and is well suited for low power and low cost applications. This advantage is even more prominent in applications where, based on prior knowledge of the failures of the mechanisms monitored, some specific spectral components are expected and can be monitored. The critical part is the matching of the twin-microaccelerometers parameters – such as mass and spring constant – since any difference between them leads to the appearance of spectral lines that do not exist in the monitored vibration signal [19, 22].

1.1 Contributions

Although the differences between the accelerometers can be corrected by adjusting the actuation voltages amplitudes, the critical part is the trade-off between the achievable sensitivity and the distortion level, which determines the amplitude of the voltage used for electrostatic actuation. In order to overcome the microaccelerometers differences and reduce the noise of the mechanical vibration spectrum, considering the sensitivity-distortion compromise, a tuning of the spectrum analyzer is proposed in the present work by means of the accelerometers excitation voltages.

In this work, the accelerometer structure used by Cretu, Rocha, and Wolffenbuttel [22] is chosen – although any other twin-accelerometers structure with separate electrodes for driving and sensing can be used –, and the simulation is based on a macro-model that considers the interaction between the electrical and the mechanical domains. All the simulations are performed with software MATLAB®İn order to make the simulation close to actual systems behavior, the rated values of the macro-model parameters correspond to the parameters of a real physical device, as described by Cretu, Rocha, and Wolffenbuttel [22]. In order to mimic the parameters differences inherent to the production process, the parameters are not set equal to their rated values; instead, they are randomly chosen from a range around these values. These assumptions about the accelerometers parameters allow the visualization of their influence in the mechanical vibration spectrum obtained.

To compensate the microaccelerometers differences, a tuning mechanism must be employed to select the excitation amplitudes that provide a better result in the final spectrum in terms of distortion and sensitivity. These features are proposed in the present work to be quantified in terms of the Fourier Series composition of the system closed-loop gain. The Fourier Series components are computed using MATLAB `fit` function applied to the system closed-loop gain when the spectrum analyzer is subjected to a known acceleration input.

The first methodology suggested to the spectrum analyzer tuning consists of a mono-objective optimization problem, since it is possible to maximize the sensitivity, while the distortion is kept below a maximum level. Therefore, the distortion represents a constraint and the cost function is based on the sensitivity. This methodology is suitable for applications where there is previous information about the spectrum to be mapped, so it is possible to determine the *a priori* required distortion level.

The other methodology applied relies on the fact that sensitivity and distortion are

concurrent features; therefore, this tuning can be performed by means of a multi-objective optimization problem. The tuning results, consequently, are an approximation to the Pareto-optimal set, so that it is possible to choose between the solutions in order to prioritize one feature over another.

To solve both the optimization problems, Differential Evolution (DE) is used. The algorithm is adapted to solve the tuning problems, using a properly proposed stopping criterion, and all the steps – initialization, reproduction process and competition – are performed by means of author's MATLAB codes. All necessary adaptations for the application of DE to multi-objective problems are implemented.

Using the results of both the mono- and multi-objective optimization problems, it is possible to determine the actuation voltages amplitudes that guarantee the requirements of the distortion-sensitivity compromise. The validation of these results occurs in two different perspectives: first comparing the optimization results with the mapped results, and then visually analyzing the final spectrum when the voltages combination is used and an external acceleration – different from that used during the tuning – is applied. The mentioned mapped results consist of an exhaustive search of the distortion and the sensitivity surfaces as functions of the voltages applied to the accelerometers. This exhaustive search provides the solutions for the tuning optimization problems based on analytical expressions for the Fourier components of the system closed-loop gain; these expressions depend on the accelerometers parameters and are also demonstrated in this work. In real cases, the accelerometers parameters are unknown, but the hypothesis assumed here is useful to understand the problem and to validate the proposed tuning strategies. The tuning, however, does not depend on the mapping to work properly.

Then, it is possible to apply the micro spectrum analyzer system, with all its advantages such as reduced size and without the need for the FFT algorithm, and now overcoming its main disadvantage, more specifically the differences found in the twin-accelerometers due to their production. The tuning performed has its characteristics, advantages, and disadvantages explained.

1.2 Dissertation organization

This work is divided as follows: a brief introduction to optimization problems is provided in Chapter 2, along with the multi-objective concepts of Pareto-optimality. In the same chapter, the use of meta-heuristics to solve optimization problems is reported and a Differential Evolution (DE) algorithm is described in detail, in addition to the adaptations found in literature to use DE to solve multi-objective problems. In Chapter 3, the application of the frequency analysis to the mechanical vibration spectrum is presented, with its possible instrumentation briefly described. In that chapter, a small history of microelectromechanical systems is presented, justifying the advantages of their use in the real-time spectrum analyzer, whose operation and particularities are thoroughly described later in this chapter. The methodologies for tuning the MEMS vibration analyzer are detailed described in Chapter 4. In Chapter 5, the results obtained

for the spectrum analyzer simulation and the tuning of the twin-microaccelerometers, validating their implementation, are presented. Finally, in Chapter 6, the conclusions and the future works are stated.

Chapter 2

Optimization and Meta-heuristics

In this chapter, basic optimization concepts are presented, together with the definition of multi-objective optimization. Next, the meta-heuristics – an alternative to solve optimization problems – are briefly described, with one meta-heuristic explained in detail: the Differential Evolution (DE) algorithm.

2.1 Optimization basic concepts

The concept of optimization underlies the analysis of many complex decision or allocation problems. Using this concept, a problem that involves the selection of values for a set of interrelated variables can be solved by focusing attention on a single objective, designed to quantify performance and measure the quality of the decision. The objective is maximized or minimized – depending on the problem formulation – subject to constraints that may limit the selection of the decision variable values [23].

A variety of practical problems can be stated as mathematical optimization problems, or some variation – as multi-criterion optimization problem –, so that mathematical optimization becomes a valuable tool in many areas. Optimization problems can be divided into continuous or discrete, constrained or unconstrained, mono- or multi-objective, static or dynamic [24].

The concept of optimization can be understood as the search for the best possible solution among all available. Mathematically, a general optimization problem has the form:

$$(P) \left| \begin{array}{lll} \textbf{minimize} & f(\mathbf{x}) & \\ \textbf{subject to} & h_i(\mathbf{x}) = 0, & 1 \leq i \leq m \\ & g_j(\mathbf{x}) \leq 0, & 1 \leq j \leq p \\ & \mathbf{x} \in S & \end{array} \right. , \tag{2.1}$$

where $\mathbf{x} = (x_1, \ldots, x_D)$ is the optimization variable vector, $f(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}$ is the objective or cost function, $h_i(\mathbf{x})$ and $g_j(\mathbf{x})$ are the equality and inequality constraint functions, the set S is a subset of the D-dimensional optimization decision space $\mathbb{R}^D$, and this set restriction is also a constraint [23]. For simplicity, it is assumed that the objective function is to be minimized. If an objective function f is to be maximized, it is equivalent to minimize the function $(-f_i)$ [25]. A

vector $\mathbf{x}^\star$ is a solution of the problem P if the objective function evaluated in $\mathbf{x}^\star$ has the smallest value among all feasible vectors, i.e., $f(\mathbf{z}) \geq f(\mathbf{x}^\star)$ for any $\mathbf{z}$ that satisfies all the constraints [26].

The form of the objective and constraint functions characterize families or classes of optimization problems. An important class is linear programming. The optimization problem (2.1) is called linear if the objective and all constraint functions are linear, i.e., if they satisfy

$$f(\alpha\mathbf{x} + \beta\mathbf{y}) = \alpha f(\mathbf{x}) + \beta f(\mathbf{y}), \tag{2.2}$$

$$h_i(\alpha\mathbf{x} + \beta\mathbf{y}) = \alpha h_i(\mathbf{x}) + \beta h_i(\mathbf{y}), \; 1 \leq i \leq m, \tag{2.3}$$

$$g_j(\alpha\mathbf{x} + \beta\mathbf{y}) = \alpha g_j(\mathbf{x}) + \beta g_j(\mathbf{y}), \; 1 \leq j \leq p \tag{2.4}$$

for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^D$ and all $\alpha, \beta \in \mathbb{R}$ [26].

Another important class of optimization problems is formed by convex optimization problems, in which S is a convex set and the objective and constraint functions are convex. This means that they satisfy the inequalities

$$f(\alpha\mathbf{x} + \beta\mathbf{y}) \leq \alpha f(\mathbf{x}) + \beta f(\mathbf{y}), \tag{2.5}$$

$$h_i(\alpha\mathbf{x} + \beta\mathbf{y}) \leq \alpha h_i(\mathbf{x}) + \beta h_i(\mathbf{y}), \; 1 \leq i \leq m, \tag{2.6}$$

$$g_j(\alpha\mathbf{x} + \beta\mathbf{y}) \leq \alpha g_j(\mathbf{x}) + \beta g_j(\mathbf{y}), \; 1 \leq j \leq p \tag{2.7}$$

for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^D$ and all $\alpha, \beta \in \mathbb{R}$ with $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$ [26]. Since convexity is more general than linearity, any linear program is therefore a convex optimization, and convex optimization is a generalization of linear programming.

The optimization problem (2.1) is an abstraction to the problem of choosing the best possible vector from a set of candidate choices. The decision vector represents the choice made, the constraints represent specifications that limit the possible choices, and the objective function value presents the cost of choosing the decision vector. A decision choice that leads to a minimum cost among all choices that meet the requirements is a solution to the optimization problem [26].

To compute a solution of an optimization problem, a solution method – an algorithm suitable to solve a class of optimization problems – can be used. Despite the effort to develop algorithms to solve various classes of optimization problems, the effectiveness of these algorithms varies considerably depending on the particular forms of the objective and constraint functions, how many variables and constraints exist, and other factors. Solving optimization problems can be a difficult task, presenting very long computation time, or the possibility of not finding the global optimal solution. Effective algorithms have been developed to solve some problem classes, such as least-square problems, linear program and convex problems [26].

When the objective or constraint functions are not linear, the optimization problem is called nonlinear optimization. To solve these problems, there are no effective general methods – problems with ten variables can be surprisingly challenging, and problems with few hundreds of variables can be impossible to solve –, and each different approach involves some compromises [26]. In local optimization, for example, the compromise is to quit the search for a variable

vector that minimizes the objective function over all feasible points, seeking for a point that is only locally optimal – i.e., a point that minimizes the objective function among feasible points that are near it. In global optimization, the true global solution – which is guaranteed to have a lower objective value than all other feasible points – is found, but the compromise is the efficiency, since even small problems, with a few tens of variables, can take a very long time – hours or days – to be solved, once fixed the computational resource available.

2.1.1 Multi-objective optimization

A multi-objective optimization problem has the form

$$(P) \left| \begin{array}{ll} \textbf{minimize} & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_L(\mathbf{x})\} \\ \textbf{subject to} & h_i(\mathbf{x}) = 0, \quad 1 \leq i \leq m \\ & g_j(\mathbf{x}) \leq 0, \quad 1 \leq j \leq p \\ & \mathbf{x} \in S \end{array} \right. , \tag{2.8}$$

where there are $L \geq 2$ objective functions $f_i(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}$. The objective functions vector is denoted as $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_L(\mathbf{x}))$ – or $\mathbf{z} = (z_1, z_2, \ldots, z_L)$, where $z_l = f_l(\mathbf{x})$ for all $l = 1, \ldots, L$ – and belongs to the objective space $\mathbb{R}^L$, and the optimization variable vector $\mathbf{x} = (x_1, \ldots, x_D)$ belongs to the decision space $\mathbb{R}^D$ [23, 25, 26]. The feasible objective space, denoted by Z, corresponds to the image of the feasible optimization decision space S. As the mono-objective case, it is assumed that all the objective functions are to be minimized, since it is possible to maximize an objective function f_l by minimizing the function $(-f_l)$ [25].

In mono-objective optimization problems, the focus is on the decision space, or design space; in that space, the constraints and the objective function contours are plotted as functions of the decision vector $\mathbf{x}$. In multi-objective optimization problems, on the other hand, the objective space is more evidenced, since the objective values are used in the definition of optimality. In the objective space – also called cost space or criterion space –, the axes represent different objective functions [25, 27].

If there is no conflict between the objective functions, a solution can be found in which every objective function achieves its optimum; this is a trivial case that requires no special methods. However, multi-objective problems generally present no single solution that is optimal with respect to every objective function, which means that the objective functions are at least partly conflicting [25].

Because of the conflict among the objectives and possible incommensurability – i.e., different units – of the objective functions, it is not possible to find a single solution that would be optimal for all the objectives simultaneously. Multi-objective problems are in a sense ill-defined, and there is no natural ordering in the objective space since it is only partially ordered – as is the case when vectors are compared in real spaces. Nonetheless, a dominance relation is defined when none of the components can be improved without deterioration to at least one of the other components. This definition, called Pareto-optimality or efficiency, is the predominant solution

concept in defining solutions for multi-objective optimization problems and is formally given by Definition 2.1 [25, 27].

Definition 2.1: Global Pareto-optimality

A decision vector $\boldsymbol{x}^\star \in S$ is Pareto-optimal if there is no other variable vector $\boldsymbol{x} \in S$ such that $f_l(\boldsymbol{x}) \leq f_l(\boldsymbol{x}^\star)$ for all $l = 1, 2, \ldots, L$ with $f_j(\boldsymbol{x}) < f_j(\boldsymbol{x}^\star)$ for at least one index j, $1 \leq j \leq L$.

A simple interpretation of this concept – in the case of minimizing all cost functions – indicates that a given solution a is better than a given solution b if all cost functions evaluated in a are less than or equal to the evaluation in b, with at least one of the cost functions being less in a than in b [26]. This definition is named after the economist and sociologist Vilfredo Pareto, who, in 1896, developed further the concept presented in 1881 by Francis Ysidro Edgeworth [25].

Similarly, an objective vector $\mathbf{z}^\star \in Z$ is non-dominated if there is no other objective vector $\mathbf{z} \in Z$ such that $z_l \leq z_l^\star$ for all $l = 1, 2, \ldots, L$ with $z_j < z_j^\star$ for at least one index j, $1 \leq j \leq L$; equivalently, $\mathbf{z}^\star$ is Pareto-optimal if the decision vector corresponding to it is Pareto-optimal [25].

The set of all Pareto-optimal vectors is called Pareto-optimal set [27].

The definitions of Pareto-optimality or efficiency, and dominance are similar; the only distinction is that efficiency refers to vectors in the decision space and dominance refers to vectors in the objective space. The hyper-surface within the objective functions space that is defined by the set of all efficient solutions is called the Pareto-front. With this set of solutions, it is possible to learn how much improving one objective worsens the others before choosing one solution from the non-dominated set. The Pareto-front divides the objective function space in two regions: one that contains non-optimal solutions – or dominated solutions –, and one that contains unfeasible solutions – i.e., the invalid points beyond the Pareto-front. In some cases, the Pareto-front may converge to a single point; this is the case when the objective functions are not conflicting or when the constraints restrict the problem [28].

Unlike the mono-objective optimization, where the only goal is to find the optimum, in a multi-objective optimization there are two goals: the convergence to the Pareto-optimal solutions; and the maintenance of a set of maximally spread Pareto-optimal solutions. These goals are independent of each other and an optimization algorithm should have specific properties for achieving each goal. The presence of two different spaces – the decision space and the objective functions space – allows flexibility in the design of search algorithms for multi-objective problems [29].

A fundamental characteristic of multi-objective optimization methods is the nature of the solutions provided by them. Some methods always provide Pareto-optimal solutions, but may not obtain all the points of the Pareto-optimal set. This characteristic is useful when the interest consists in obtaining just one solution point. Some other methods are able to provide all the Pareto-optimal points, but may also yield non-Pareto-optimal points. This quality is beneficial when the complete Pareto-set must be generated [27].

2.1.1.1 Weighted sum of objective functions

The weighted-sum method is a classical procedure to transform a set of objectives into a single objective by pre-multiplying each objective with a weight provided by the user. Although the idea is simple, the question is how to choose the weights [29]. Based on how the weights are assigned, the method can be classified as *a priori*, progressive, or *a posteriori*. The first group assumes that the objective preferences can be ordered and that the weights do not change during optimization. The second group is more flexible than the first, since the selection made at the beginning of the optimization can be altered, correcting the weights based on next level knowledge. Finally, in *a posteriori* methods, the preference is typically based on Pareto-optimality [28].

2.1.1.2 ε-Constraint method

Another classical method to solve multi-objective problems consists in reformulating the multi-objective optimization problem by keeping one of the objectives and constraining the rest of the objectives within specified values chosen by the user, based on the problem itself. This approach is the ε-constraint method [29]. The optimization problem of Equation 2.8 then becomes:

$$(P) \quad \left| \begin{array}{ll} \textbf{minimize} & f_\mu(\mathbf{x}) \\ \textbf{subject to} & f_n(\mathbf{x}) \leq \varepsilon_n, \quad 1 \leq n \leq L,\ n \neq \mu \\ & h_i(\mathbf{x}) = 0, \quad 1 \leq i \leq p \\ & g_j(\mathbf{x}) \leq 0, \quad 1 \leq j \leq p \\ & \mathbf{x} \in S \end{array} \right. , \tag{2.9}$$

where the parameter ε_n represents an upper bound of the value of f_n and it is not necessarily close to zero. Since the constants ε_n are user-defined, changing their values results in different single optimum solutions, and therefore different Pareto-optimal solutions can be found [30].

2.2 Meta-heuristics

In optimization problems, the central idea is to find the optimal solution within the decision domain. For a differentiable scalar cost function, its gradient gives the fastest increasing direction of the cost function. Thus, optimization problems seem to be easy, since theoretically it is possible to start from any initial point and use the gradient to guide the search to the optimum, where the gradient is null. However, this is only applicable for some types of problems. In combinatorial, multi-objective constrained optimization and even in differentiable unconstrained optimization, methods based on gradients might not work [31]. A valid alternative to gradient-based optimization algorithms are the heuristic-based search techniques [30].

Heuristics and meta-heuristics are approximation methods that can find a good enough solution in a reasonable time. The main difference between heuristics and meta-heuristics is that the first one is problem-dependent – heuristics can be efficient to a specific problem but not to other problems. A meta-heuristic, on the other hand, is a generic algorithm or a black-box

optimizer that can be applied to almost all optimization problems [32]. This characteristic of meta-heuristics associates with two search schemes: exploration and exploitation. These concepts can be understood as global and local searches, respectively. The exploration – also called diversification – refers to the ability of visiting many different regions of the search space. Exploitation, which is also called intensification, allows to obtain high-quality solutions within already detected promising regions [32, 33].

Meta-heuristics can be classified in different ways, considering the way of employing the exploration and the exploitation, and the metaphor of the search procedures, where metaphor refers to the simulation of some natural, human behavior, or mathematics phenomena [32]. There are meta-heuristics based on chemistry, music, mathematics, physics, social, and sport principles. There are also non-metaphor based meta-heuristics, that do not use any inspiration to determine their search strategy; such is the case of Tabu Search. However, many meta-heuristics are inspired by biological evolution principles, such as evolutionary, swarm, and immune systems.

To solve optimization problems, meta-heuristics can be based on a single solution or population of solutions; in both cases, they do not use the gradient nor the Hessian matrix of the objective function and present several meta-parameters to be adjusted according to the problem [24]. Meta-heuristics are widely recognized as efficient tools for several optimization problems that cannot be solved by deterministic methods in admissible time.

Among some of the most commonly used meta-heuristics [34] are Simulated Annealing, Tabu Search, and Genetic Algorithms, the latter encompassing several algorithms – such as the Differential Evolution, described in what follows.

2.2.1 Differential Evolution

The Differential Evolution (DE) is one of the most powerful tools for global optimization [24]. It consists in a small and simple mathematical model of a big and naturally complex process of evolution, realizing the evolution of a population of individuals in an intelligent manner – using differences between individuals, performed in a simple and fast linear operation, so-called differentiation, making DE unique [35]. Proposed by Storn and Price [36], it was developed to be a reliable, versatile, and easy-to-use function optimizer. In DE, like in nearly all other evolutionary algorithms, an initial population of candidate solutions is arbitrarily chosen. For each generation of the evolutionary process, new individuals are created through the reproductive process, which consists of mutation and crossover. Each member of the current population competes with a new individual generated by the reproductive process, so only the one that produces the most adequate solution to the problem in question is maintained in the next population. DE presents some variations [28] that can be named by the code *DE/x/y/z*; *x* indicates the base individual, being either random (*rand*) or the best within the population (*best*); *y* indicates the number of difference vectors used to disturb the base vector; the crossing scheme is indicated by *z*, and can be binomial (*bin*) or exponential (*exp*). The classical version of the DE algorithm, and also the most commonly used one, is denoted by *DE/rand/1/bin*.

In this section, the description of the DE is made based on the concepts, properties, and characteristics described by Price, Storn, and Lampinen [28]; thus, the notation used by those authors is also used here. The DE algorithm maintains a pair of vector populations, both of which contain N_p real vectors of dimension D. The population in the generation g is expressed as:

$$\boldsymbol{P}_{\boldsymbol{x},g} = \left(\boldsymbol{x}_{i,g}\right), \quad i = 1, \ldots, N_p, \quad g = 1, \ldots, g_{max} \tag{2.10}$$

and contains the N_p feasible vectors – called target vectors, $\boldsymbol{x}_{i,g}$. The i-th target vector of generation g is given by Equation 2.11.

$$\boldsymbol{x}_{i,g} = \left(x_{1,i,g}, x_{2,i,g}, \ldots, x_{D,i,g}\right) \tag{2.11}$$

After the initialization, DE mutates randomly chosen vectors to produce an intermediary population

$$\boldsymbol{P}_{\boldsymbol{v},g} = \left(\boldsymbol{v}_{i,g}\right), \quad i = 1, \ldots, N_p, \quad g = 1, \ldots, g_{max} \tag{2.12}$$

of N_p mutant vectors $\boldsymbol{v}_{i,g}$, given by Equation 2.13.

$$\boldsymbol{v}_{i,g} = \left(v_{1,i,g}, v_{2,i,g}, \ldots, v_{D,i,g}\right) \tag{2.13}$$

The relation between the target vectors and the mutant vectors is presented later in this text.

Each vector of the current population is then recombined with a mutant vector, producing a trial population

$$\boldsymbol{P}_{\boldsymbol{u},g} = \left(\boldsymbol{u}_{i,g}\right), \quad i = 1, \ldots, N_p, \quad g = 1, \ldots, g_{max} \tag{2.14}$$

of N_p trial vectors $\boldsymbol{u}_{i,g}$, given by Equation 2.15.

$$\boldsymbol{u}_{i,g} = \left(u_{1,i,g}, u_{2,i,g}, \ldots, u_{D,i,g}\right) \tag{2.15}$$

The relation of trial and mutant vectors is described later in this text; these vectors can occupy the same array, since the first one overwrites the last during recombination [28].

Before the population initialization, the limits for each variable x_i of the variable vector $\boldsymbol{x}$ must be specified, i.e., back to Equation 2.1, the upper and lower bounds of the set S must be defined in two D-dimensional initialization vectors – $\boldsymbol{b}_U$ and $\boldsymbol{b}_L$, respectively. After this specification, a random number generator $rand_j(0,1)$ – that returns a uniformly distributed random number from within the range $[0,1)$ and generates a random value for each parameter j – is used to assign each variable to a value within the correspondent range, according to Equation 2.16.

$$x_{j,i,0} = rand_j\left(0,1\right)\left(b_{j,U} - b_{j,L}\right) + b_{j,L} \tag{2.16}$$

All the variables are initialized with real values – even if they are discrete or integer –, since DE internally treats all variables as floating-point, despite of their type, to add diversity to their difference distributions.

The step after the initialization is the recombination, which involves mutation and crossover. The differential mutation is represented in Equation 2.17, where a randomly sampled vector, $\boldsymbol{x}_{r_0,g}$, is added to a difference vector, $(\boldsymbol{x}_{r_1,g} - \boldsymbol{x}_{r_2,g})$ – also randomly chosen –, generating the mutant vector $\boldsymbol{v}_{i,g}$.

$$\boldsymbol{v}_{i,g} = \boldsymbol{x}_{r_0,g} + F\left(\boldsymbol{x}_{r_1,g} - \boldsymbol{x}_{r_2,g}\right), \tag{2.17}$$

where the scale factor F, $F \in (0, 1+)$, multiplies the difference vector and controls the population evolution rate. The constant F is a control parameter, which manages the trade-off between exploitation and exploration of the space [35]. Even though there is no upper limit to F, it is rarely greater than 1.0. As mentioned, the vectors are randomly chosen, so the indexes r_0, r_1, and r_2 are randomly generated and must be distinct from one another and from the target vector index i, i.e, they must be mutually exclusives: $r_0 \neq r_1 \neq r_2 \neq i$.

The DE reproductive process also involves a uniform crossover, building trial vectors by copying parameter values from two different vectors.

$$u_{j,i,g} = \begin{cases} v_{j,i,g}\,, & \text{if } rand_j\,(0,1) \leq CR \text{ or } j = j_{rand} \\ x_{j,i,g}\,, & \text{otherwise}\,, \end{cases} \tag{2.18}$$

where the crossover probability, $CR \in [0, 1]$, controls the fraction of parameters copied from the mutant and the condition $j = j_{rand}$ ensures that the trial vector $\boldsymbol{u}_{i,g}$ is not a duplicate of the target vector $\boldsymbol{x}_{i,g}$.

The final step of DE is the selection. In the case of a minimization cost function, during this process, the trial vector replaces the target vector in the next generation only if the first has an equal or lower objective function than the later:

$$\boldsymbol{x}_{i,g+1} = \begin{cases} \boldsymbol{u}_{i,g} & \text{if } f\left(\boldsymbol{u}_{i,g}\right) \leq f\left(\boldsymbol{x}_{i,g}\right) \\ \boldsymbol{x}_{i,g} & \text{otherwise}\,. \end{cases} \tag{2.19}$$

It is important to highlight that, at this stage, the quantization of the variables is applied: the cost function is evaluated with the quantized parameters, although the real values are maintained in the populations.

The process of mutation, recombination and selection is repeated until the optimum is reached or a stopping criterion is satisfied.

Feoktistov [35] provides a clear graphical interpretation of Differential Evolution that simplifies the understanding of such algorithm, emphasizing three key-elements to DE success. The first key-element is the spontaneous self-adaptability to the function. This characteristic lies in the way of creating the mutant vector, given in Equation 2.17. In this equation, the scaled difference between two randomly chosen individuals defines the direction and length of the search step. This difference is added to a third randomly chosen individual, that consists in a base point of application, i.e., the current reference point. The basic idea of this equation – and of Differential Evolution in general – is to adapt the step length intrinsically throughout the evolutionary process. In the first generations, the step length is large, since individuals

are scattered all over the feasible decision space and far from each other. As the algorithm progresses, the population converges and the step length becomes smaller. The randomness of search directions and base points provides, in many cases, the convergence to the global optimum.

The second key-element for DE success, according to Feoktistov [35], is the diversity control, which occurs in the creation of the trial vector, according to Equation 2.18. In this equation, each parameter of the mutant vector is accepted by the trial individual with a probability CR, and the random number j_{rand} ensures that at least one element of the trial vector is coppied from the mutant vector – and, therefore, is different from the target vector. Even though the mutant vector could be used directly in the next step of the algorithm, the crossover used to create the trial vector increases the exploration and exploitation capabilities, acting as a diversity control; another side effect is the increase in the ability to handle some functions properties, resulting in better convergence of the algorithm.

The final element that ensures DE success is the continuous improvement, related with the selection of the best vector, as presented in Equation 2.19. The trial is compared with the target vector; if the trial vector presents an equal or better fitness value, it replaces the target vector in the next generation. Thus, the population fitness always improves or, at least, maintains the same values. The selection described is called elitist selection and is used successfully in many evolutionary algorithms [35].

In constrained optimizations problems, DE can deal with boundary constraints – that define the upper and lower bounds of the set S – separately and in a easier way than the inequality constraints. Since the current population already satisfies all the constraints, the boundaries only need to be checked when a mutant parameter is selected for the trial vector [28]. Basically, there are two distinct techniques to deal with parameters that violate the bounds: resetting and penalty methods.

Random reinitialization is an example of the resetting method, modifying the out-of-bounds parameters so that the trial vector satisfies all constraints. The bounce-back method is similar to the random reinitialization, but it takes into account the progress toward the optimum by selecting a parameter value between the base parameter and the violated boundary. Another option is the resampling, which randomly samples a new mutant vector, and therefore, a new trial vector. On the other hand, penalty methods deal with the out-of-bounds parameters using an objective function criterion. The brick-wall penalty is a simple example of penalty methods, assigning a value high enough to the objective function of the vector that violates the bounds, ensuring that it will not be selected. The techniques cited to deal with boundary constraints are described, as well as their pseudo-code, in Price, Storn, and Lampinen [28], except the resampling, which can be found in Arabas, Szczepankiewicz, and Wroniak [37].

In contrast, inequality constraints are commonly implemented as penalty functions that increase the objective function – considering that the optimization problems constitute on a minimization –, when constraints are violated. Then, penalty functions transform a constrained problem into an unconstrained one with the definition of a new evaluation function by adding

or subtracting a certain value from the objective function based on the amount of constraint violation present in a certain solution [38, 39].

In classical optimization, there are two kinds of penalty functions: exterior and interior. In the first method, it starts with an unfeasible solution and from there the algorithm moves to a feasible region. In the case of interior methods, the penalty term is chosen such that its value is small at feasible points away from the constraint boundaries, and tends to infinity as the constraint boundaries are approached. Then, if the algorithm starts in a feasible point, the subsequent points generated always lies within the feasible region since the constraint boundaries act as barriers [39].

Using penalty functions presents the risk that one penalty may dominate unless the weights are correctly adjusted. If the penalty weight is predefined by the user and does not change during the evolving process, the penalty function is called static. An extreme example of static penalty function is the death penalty. This is one of the easiest ways to handle constraints and assigns an infinite cost function to every solution that violates a constraint. Therefore, there is no need of further calculations to estimate the degree of unfeasibility of such solutions; besides, the cost function does not need to be evaluated for unfeasible points. This approach is taken recursively, generating a new point at each recursive call, until a feasible solution is found. One of the drawbacks of the death penalty is the possibility of stagnation when all the initial population is unfeasible; this approach works well when the feasible search space constitutes a reasonably large portion of the whole search space [39].

Price, Storn, and Lampinen [28] described a direct method – called Lampinen's criterion – that separates the objective value and the constraint violation, assigning to each population vector an array of objective values, containing the objective function value and also the constraint functions values. This criterion selects the trial vector $\mathbf{u}_{i,g}$ over the target vector $\mathbf{x}_{i,g}$ in the selection stage if:

- $\mathbf{u}_{i,g}$ satisfies all constraints and has a lower or equal objective function value than $\mathbf{x}_{i,g}$, or;
- $\mathbf{u}_{i,g}$ is feasible and $\mathbf{x}_{i,g}$ is not, or;
- $\mathbf{u}_{i,g}$ and $\mathbf{x}_{i,g}$ are both unfeasible, but $\mathbf{u}_{i,g}$ does not violate any constraint more than $\mathbf{x}_{i,g}$.

This criterion for direct constraint handling is based on Pareto-dominance.

Another important definition in the DE algorithm is the stopping criterion. Even though sometimes it is clear when the optimization has finished, it is not always evident when to stop the search to ensure a desirable solution. That is the case of multi-objective problems, where the objectives are conflicting and it is not obvious when to stop to guarantee a better compromise.

A popular strategy to deal with the stopping criterion in evolutionary algorithms consists in reaching a certain number of objective function evaluations or a certain number of generations. The disadvantage of this strategy is the need to perform tests to determine the maximum number of generations, since this information is not known *a priori*. This process can be avoided by using

information about the state of the optimization in the stopping criterion. The stopping criterion must ensure that the algorithm is executed long enough to converge to the optimum, without wasting computational resources [40, 41]. Different mechanisms can be used to conclude about the current state of the optimization. In principle, any phenomenon that shows a definite trend from the beginning to the end of the optimization can be used. As an example, the movement of the population individuals is large at the beginning and decreases as the algorithm converges.

Zielinski, Peters-Drolshagen, and Laur [40] classifies six different stopping criteria, which are briefly described as follows:

1. Reference criterion: when the optimum is known, the algorithm terminates when a certain percentage p of the population converged to the optimum.
2. Exhaustion-based criterion: due to limited computational resources, stopping criterion can be set when a certain computation time, number of generations, or objective function evaluations are reached.
3. Improvement-based criterion: if only small improvements are accomplished over some time, the optimization should finish.
4. Movement-based criterion: similar to the improvement, the movement of individuals can be used as stopping criterion.
5. Distribution-based criterion: usually all individuals converge to the optimum, so the algorithm can stop when the individuals are close to each other.
6. Combined criterion: the combination of several stopping criteria can benefit from all the different advantages of each one.

Within the improvement-based criterion, it is interesting to highlight the strategy called $NoAcc$ [41]. Since DE incorporates a greedy selection scheme, the acceptance of trial vectors means that the population is improving. Based on this, monitoring if trial vectors are selected over a specified number of generations can be a criterion to finish the optimization. This method has the advantage that only one parameter must be set, and it is not recommended to choose a too low value, since long periods without improvement may occur during the optimization process.

One of the great advantages of DE is its small number of search control meta-parameters: the population size N, the differentiation constant F, and the control parameter of the crossover rate CR [24]. Although it is not possible to determine *a priori* these meta-parameters, the literature [36, 42, 43] recommends some initial settings, such as $N = 10D$, $F = 0.9$, and $CR = 0.9$ – which are typically effective, even though the control parameters fine-tuning usually results in a considerably higher convergence rate. Several methodologies have been proposed for the control meta-parameters tuning [44], for example, perform an exhaustive grid search, varying the meta-parameters and analyzing the results in terms of mean time, best and worst solutions found, convergence rate and standard deviation [45].

2.2.1.1 Differential Evolution for multi-objective optimization

Several real-world optimization problems can be modeled using multiple conflicting objectives. The classical approach to solve such problems were focused on scalarizing multiple objectives into a single objective. Whereas, evolutionary algorithms are used to solve multi-objective problems as they are [29]. Such problems have a set of optimal solutions – the Pareto-optimal solutions described in Section 2.1 – instead of a single optimum solution; these solutions should be considered equivalent in the absence of information about the relevance of each objective [46]. Then, it is important to find not just one of these solutions, but as many as possible. Since any solution constitutes a compromise relation between the cost functions, a better choice can be made when several solutions are unveiled.

When using stochastic techniques, such as meta-heuristics, the goal is to obtain a Pareto-front approximation. Having this set, it is possible to choose a solution that privileges one cost function over another, depending on the need of the problem in question [47].

For most nonlinear multi-objective optimization problems, although it is practically impossible to find all the Pareto-optimal points, it is realistic to find a subset of the Pareto-set. Even a simple random search can locate Pareto-optimal points; however, evolutionary algorithm are a natural option for solving multi-objective optimization problems, since they deal with groups of candidate solutions. Evolutionary algorithms can find multiple non-dominated solutions within a single run, and that is the main advantage of this approach.

In numerical algorithms, genetic algorithms and random-search methods for multi-objective optimization, the idea of dominance in the objective space is used for a subset of points; that is, the objective function value of a new addition to the set of potential solutions is compared to the objective function value previously presented to determine if the new point is dominated. If it is non-dominated, the new point is kept in the set of potential solution points; this point, however, may not be Pareto-optimal [27], since the comparison occurs over a set that does not match the feasible set of points in the decision space.

Several authors [30, 48, 49] have proposed the use of differential evolution for multi-objective optimization problems.

According to Price, Storn, and Lampinen [28], by using Pareto-dominance as a selection criterion, a population can be driven toward the Pareto-front. This incorporation of the dominance-based selection into DE consists on comparing the trial and target vectors to determine which one is dominant, as given by Equation 2.20.

$$\boldsymbol{x}_{i,g+1} = \begin{cases} \boldsymbol{u}_{i,g} & \text{if } \forall n \in \{1, 2, \ldots, L\} : f_n\left(\mathbf{u}_{i,g}\right) \leq f_n\left(\mathbf{x}_{i,g}\right) \\ \boldsymbol{x}_{i,g} & \text{otherwise} \end{cases} \tag{2.20}$$

According to Equation 2.20, the trial vector $\mathbf{u}_{i,g}$ is selected if the trial vector dominates the target vector $\mathbf{x}_{i,g}$ or if the trial and the target vectors are non-dominated with regard to each other. This is because of the criterion used to select $\mathbf{u}_{i,g}$ is necessary, but not sufficient, to ensure that $\mathbf{u}_{i,g}$ dominates $\mathbf{x}_{i,g}$. In other words, the trial vector is not selected only in the cases where it

is not possible for it to dominate the target vector in the objective function space.

In many cases, the trial vector $\mathbf{u}_{i,g}$ can be rejected before all the L objective functions have been evaluated, which makes DE faster. The flowchart indicated in Figure 2.1 describes the Pareto-dominance selection criterion of Equation 2.20, considering the minimization of all the objective functions.

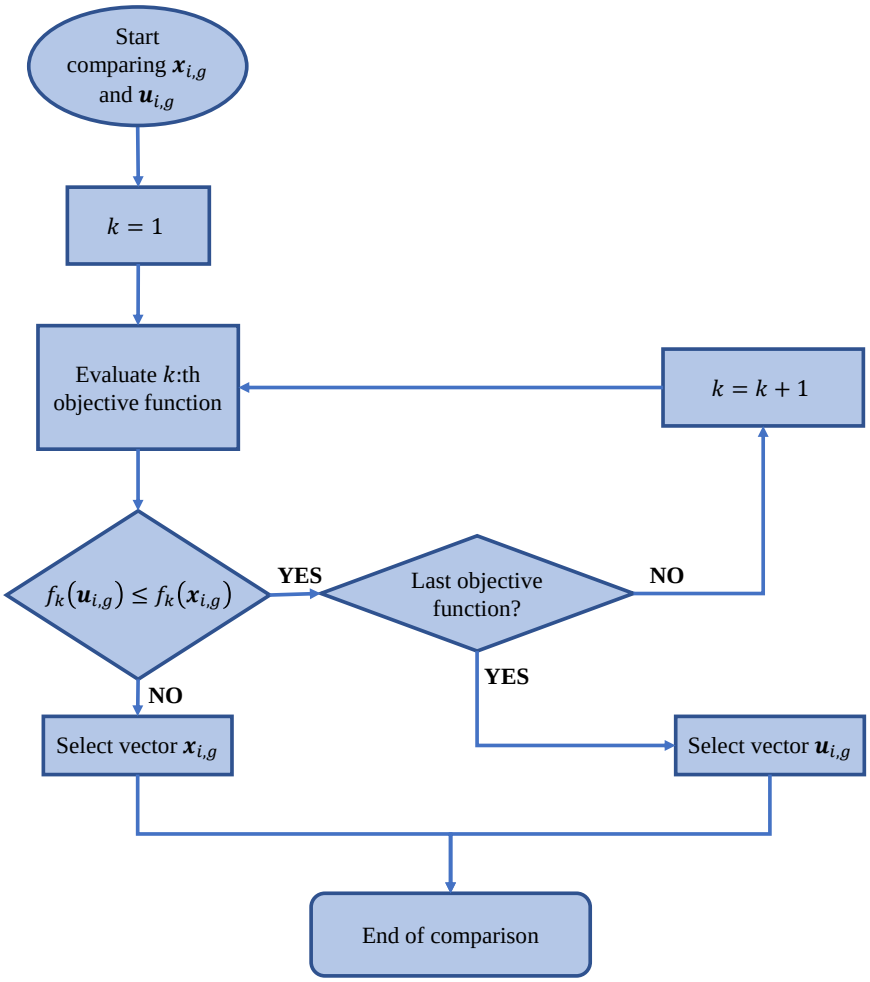

Figure 2.1: Flow chart for the implementation of the Pareto-dominance selection rule of Equation 2.20. Adapted from Price, Storn, and Lampinen [28, p. 251].

After a number of generations, some of the population vectors will be dominated, while others will be non-dominated. As a final step all the dominated points in the last generation should be removed; then, the remaining population approximates the Pareto-optimal set of solutions for the multi-objective problem [28].

The Pareto-DE approach described by Price, Storn, and Lampinen [28] is relatively easy to implement and should be effective on a wide range of problems. This approach, however, does not have a tool to deal with the second goal on multi-objective optimization problems, which is the attainment of a set of maximally spread Pareto-optimal solutions.

It is possible to find several different implementations of DE to multi-objective optimization problems. Mezura-Montes, Reyes-Sierra, and Coello [46] provide a good review on some DE multi-objective adaptations found in literature, besides classifying them into three categories according to the use of Pareto-optimality. The first category consists in non-Pareto-based approaches and considers combination of functions or problem transformation to deal with the multi-objective problem. The Pareto-based approaches use the Pareto concepts to deal with the multiple objectives by either using it as a criterion to select the best solution in the DE selection mechanism, or as a ranking procedure. The third class of methods listed by Mezura-Montes, Reyes-Sierra, and Coello [46] considers approaches where a set of schemes have been mixed in the DE-based multi-objective algorithm. These approaches consider Pareto concepts and population-based concepts in the same technique, or local and global search together.

One interesting adaptation of DE to multi-objective optimization found in literature is the proposal of Robič and Filipič [49], named Differential Evolution for Multi-Objective Optimization (DEMO). The algorithm modifies the selection criterion to decide when the trial vector replaces the target vector considering the concept of dominance. If the trial vector dominates the target vector, this last one is replaced; when the target vector dominates the trial vector, this last one is discarded; otherwise – i.e., when there is no dominance between the trial and target vectors –, the trial vector is simply added to the population. Therefore, the population is extended and the newly created vectors take part immediately in the creation of the following vectors, which emphasizes elitism within reproduction and helps achieving the first goal of multi-objective optimization, the convergence to the true Pareto-front. The enlarged population obtained due to the selection criterion is truncated to prepare it for the algorithm next step. This truncation is derived from Non-dominated Sorting Genetic Algorithm II (NSGA-II) and consists in sorting the individuals with non-dominated sorting and then evaluating the individuals of the same front considering the crowding distance. This process stimulates the uniform spread of solutions, which is a tool to achieve the multi-objective optimization second goal: finding diverse non-dominated solutions.

The tuning of MEMS mechanical vibration spectrum analyzers, adjusting the sensitivity-distortion relation, can be seen as an optimization problem. Therefore, the concepts and techniques presented in this chapter are essential do understand the tuning problem – formulated in Chapter 4 – and to solve it using DE algorithms – as presented in Chapter 5. Before the tuning proposal, however, it is fundamental to understand the spectrum analyzer operation, as described in the next chapter.

Chapter 3

Vibration Mapping and Monitoring

Even though the interest of ancient civilizations in music, their knowledge in vibration did not reach the level of a science. It was the philosopher and mathematician Pythagoras who first investigated music sounds using a scientific basis, investigating experimentally the behavior of a vibrating string. In the 16^{th} century, Galileo Galilei (1564–1642) studied the behavior of a simple pendulum, describing the dependence of the vibration frequency on the pendulum's length, along with the phenomenon of sympathetic vibrations, also called resonance. Galileo also indicates the understanding of the relation between frequency, length, tension and density of a vibrating stretched string. The possibility of such string to vibrate with several of its harmonics at the same time was argued on physical grounds by Daniel Bernoulli (1700–1782) in 1753. This characteristic – referred to as the principle of the coexistence of small oscillations, which is currently called the principle of superposition –, is valuable to the vibration theory development, allowing to express any arbitrary function using an infinite series of sines and cosines [1]. However, since Bernoulli did not pursued a mathematical proof, his ideas were not accepted. Leonard Euler (1707–1783) – who examined the motion of a vibrating string and noted that if this string configuration at some point in time is a linear combination of normal modes, so is the configuration at any subsequent time – discarded trigonometric series. In 1759, Joseph Lagrange (1736–1813) strongly criticized the use of trigonometric series in the examination of vibrating strings, arguing that trigonometric series were of very limited used [50].

Jean Baptiste Fourier (1768–1830) presented his ideas half a century after Bernoulli's proposition. In 1807, Fourier discovered that series of harmonically related sinusoids were useful to represent the temperature distribution through a body, and claimed that any periodic signal could be represented by such a series. Besides his imprecise mathematical arguments, many of the basic ideas behind Fourier's treatment had been discovered by others – since the concept of using sums of harmonically related sines and cosines or periodic complex exponentials to describe periodic phenomena goes back to Babylonians, who used it to predict astronomical events. It was Peter Lejeune Dirichlet (1805–1859) in 1829 who provided precise conditions that determine if a periodic signal can be represented by a Fourier series, i.e., as a weighted sum of harmonically related sinusoids. Thereby, Fourier did not contribute mathematically to the theory of the series that carries his name. However, he was able to see the potential for this series

representation. Besides, Fourier used this type of representation for aperiodic signals using weighted integrals of sinusoids that are not all harmonically related, going one step further than his predecessors [50].

The Fourier series representation of periodic signals by means of combinations of harmonic signals or sinusoids unfolds a perspective of periodic signals in the frequency domain in terms of their frequency content, or spectrum. The term spectral analysis, or harmonic analysis, is often used to refer to the analysis of a periodic signal by its Fourier series. The magnitude and phase spectra are plots of the magnitude and phase of each harmonic, plotted as discrete signals; one-sided spectra refer to plots which contain only positive frequencies, while two-sided spectra contain plots in all frequencies, positive and negative [51]. The Fourier Transform – an expansion of the Fourier series to aperiodic signals – provides a frequency-domain description of a time-domain signal; in this case, the spectrum becomes a continuous curve.

Adnani, Duplicy, and Philips [12] made an evaluation and a historical review of spectrum analyzers, classifying them as swept-based and FFT-based. A spectrum analyzer allows the study of the spectral composition of electrical, acoustic, optical, and mechanical waveforms and is an essential tool in many applications in science and technology, from medicine [13, 14], to civil engineering [2, 3]. The first category of spectrum analyzers consists in superheterodyne architecture and sweep across the frequency range of interest, displaying all the components present. The second category, as the name suggests, calculates and displays the FFT for blocks of the input signal data. FFT analyzers are limited by Analog-to-Digital Converter (ADC) technology in terms of frequency and dynamic range but, unlike swept analyzers, they are capable to capture not only the magnitude, but also the signal phase in the spectral domain.

The latest evolution in spectrum analyzers is the real-time processing, possible due to the advances in digital signal processing. In spectral analysis, real-time operation means that all signal samples are processed for some sort of measurement result or triggering operation. Real-time spectrum analyzers are gap-free, which means that no information is lost during the spectrum calculation [52].

The most popular approach for mechanical vibration spectral analysis is still the use of the FFT. This strategy is based on acceleration sensors – called accelerometers – to acquire the time-series data of the system under analysis. The hardware of the spectrum analyzer must also have a signal conditioner, ADCs, and a processor to compute the FFT algorithm; this algorithm is the responsible to translate the data acquired in the time domain to the frequency domain. The processors used to perform the FFT are diverse, such as computers [7, 53], DSP [8, 9], and FPGA [10, 11, 54]. The FFT algorithm was introduced in the 1960s and proved to be perfectly suited for efficient digital implementation, reducing the time required to compute transforms by orders of magnitude. With this tool, the ideas of Fourier series and transform – described in Appendix A – became practical. An important issue, however, is the introduction of nonexistent spectral components in the final computed spectral due to the FFT algorithm itself.

3.1 Microinstrumentation

The development of microsystems began with the advances of microelectronics in the 1960s; although, it was from 1980s that the greatest advances in accelerometers, pressure sensors, microactuators, and other electromechanical structures were conquered [55]. The current maturity level of Microelectromechanical Systems (MEMS) has led to their use in daily tasks of modern man. Present in accelerometers and pressure sensors in cars, micromirrors on plasma televisions, microphones in mobiles, and inertial sensors in videogames, MEMS are defined as the integration of mechanical elements, sensors, actuators, and electronics for signal processing on a common silicon substrate through the micromanufacturing technology [56].

MEMS involve electronic and non-electronic elements and can perform functions that include signals acquisition and processing, actuation and control; they may also serve as an environment for chemical and biochemical reactions. Basically, MEMS are sensors and actuators. In the first group, inertial sensors, such as accelerometers and gyroscopes can be mentioned. Among the examples of MEMS actuators are micromirrors to deflect light on flat panel televisions and force and displacement actuators used for microscopy [56].

Silicon is the most commonly used material for the base where the components are constructed and electrically connected, although recently other materials – such as conductive polymers – have been the subject of research [57]. In addition to its excellent thermal and mechanical resistance coupled with low thermal expansion and high melting point, silicon is the preferred material because of its well-established manufacturing processes due to previous microelectronics research. The micromanufacturing technology – extensively researched and optimized for each stage to achieve stability and reliability – allows the production of a large number of devices at the same time, so that the production of MEMS is given at a very low cost [55, 56].

In addition to the low cost, their low weight and reduced size are interesting and convenient for several applications, allowing new possibilities of implementation, impracticable with the conventional scale devices – as is the case of medicine. These applications are possible also due to the low power consumption, allowing the development of self-powered MEMS that drain the energy required for its operation from the environment [58], reducing its operational costs. Other advantages of MEMS are their superior performance, intelligent features, and the ability to perform complex tasks that would not be solved with other technologies [56].

The accelerometers are among the most used MEMS sensors. They are present in various applications, from the automotive industry – in the actuation of airbags and active suspensions –, to consumer electronics, as smartphones and video game consoles. MEMS accelerometers can be designed and manufactured to be sensitive to acceleration components on one, two, or three axes. Acceleration can be measured through the use of capacitive [59–61], piezoelectrics [62, 63], piezoresistive [64], optical [65], tunneling [66–68], thermal [69, 70], or inductive [71] transducers. However, capacitive accelerometers are the most popular because they have the advantages of high sensitivity and low noise, associated with low power consumption, low cost, and low

temperature dependence.

There are, however, some aspects that require caution in capacitive sensors application, such as their non-linearity and the vulnerability to electromagnetic interference [72], requiring attention during the devices modeling. The capacitive microaccelerometers present various topologies, such as masses suspended by single or several supports [73], or even V-shaped structures [74]. Each one of these topologies leads to specific characteristics of bandwidth, linearity, sensitivity, and natural frequency.

Although MEMS accelerometers can be used together with FFT algorithms to obtain the mechanical vibration spectrum, another possibility is to explore their resonance frequency. Then, by amplifying the vibration signal around this frequency, the signal-to-noise ratio is improved and, therefore, the spectral component in this frequency is obtained [17, 18]. This strategy has its main disadvantage in the fixed frequency mapped, that corresponds to the MEMS resonance frequency. Thus, it can be used only in specific applications that have known frequencies correspondent to the microdevice resonance.

To make the use of microinstruments more flexible, some method to select its resonance frequency can be employed. One of those methods is the electrostatic force feedback, which is based on the application of a continuous voltage excitation, allowing electronic modulation of the spring constant of the instrument [19]. The electrostatic force feedback allows the acquisition of several spectral lines simultaneously by using an array of oscillators, each one tuned to a specific frequency [16, 20].

The electrostatic force feedback concept can be further explored by applying an alternating excitation signal to the device. As a result, the vibration components are sequentially obtained by varying the driving signal frequency, as detailed in Subsection 3.1.1.

3.1.1 Real-time MEMS-based spectrum analyzer

The device operation proposed by Cretu, Bartek, and Wolffenbuttel [21] can be understood on the basis of a conventional inverted pendulum, clamped at one end, inserted in a vertical gravitational field $\overrightarrow{g}$, according to Figure 3.1.

The seismic mass m is assumed to be concentrated at the top, while the clamped beam has only elastic properties, disregarding its mass. Any external horizontal force $F_{x,ext}$ applied to the system causes a displacement φ from the vertical equilibrium position, until the reaction developed at the set point balances the external action. Without a gravitational field, this external action is solely determined by the horizontal force; however, in the presence of a gravitational field, the effect is magnified by the tangential component of the weight force, $F_{v,t}$, and, thus, results in a greater equilibrium deflection. The tangential component of the weight force, therefore, amplifies the rotational effect due to the external force, inducing a positive feedback.

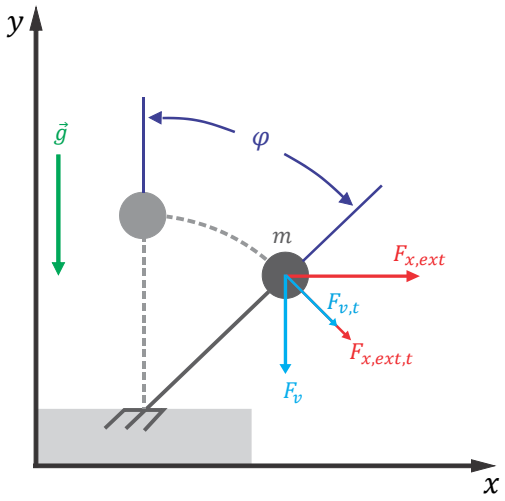

Figure 3.1: Operating principle of the inverted pendulum inserted in a gravitational field. Adapted from Cretu, Bartek, and Wolffenbuttel [21, p. 24].

Although the spectrum analyzer used in this project has a general operating principle, which can be extended to any capacitive microaccelerometer with separate electrodes for driving and sensing, the practical problems encountered in its use are not directly related with the chosen topology, but with the comparative differences of the accelerometers. Thus, in this work the topology originally used by Cretu, Rocha, and Wolffenbuttel [22], highly simplified in Figure 3.2, is considered. It consists of a vertical beam that supports two sets of horizontal arms that act as movable plates for the corresponding differential capacitors; there are separate differential capacitors for sensing and actuation, two springs, and the structure is clamped at both sides.

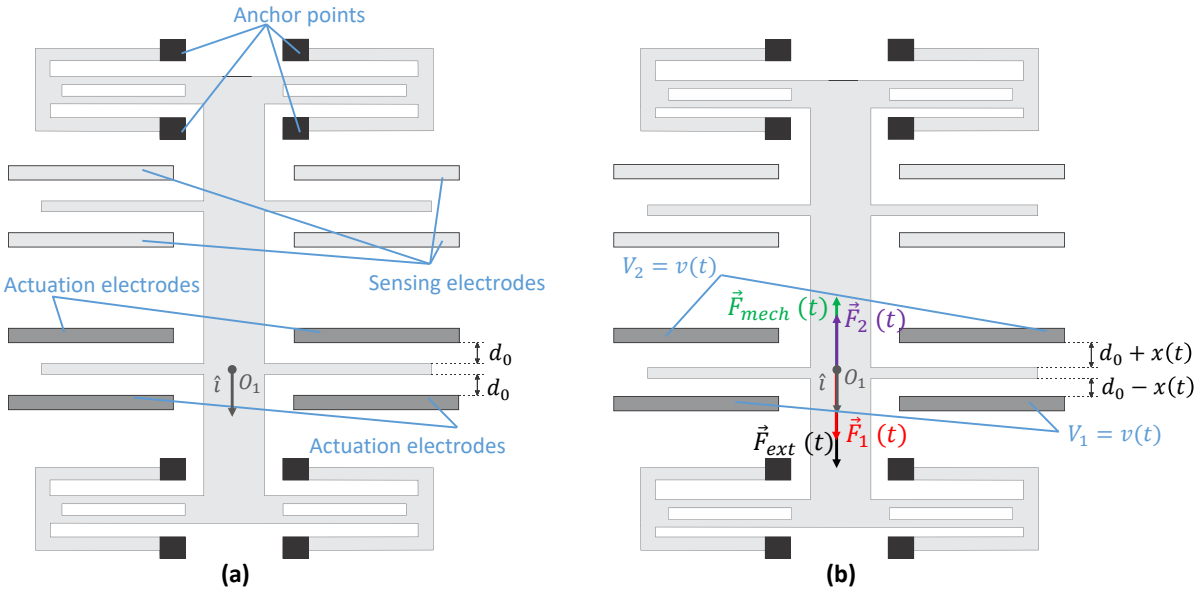

Figure 3.2: Schematic representation of the microaccelerometer with zero-displacement (a) and with a $x(t)$ displacement (b). Adapted from Cretu, Rocha, and Wolffenbuttel [22, p. 1407].

In the zero-displacement condition, corresponding to the equilibrium state when no external disturbances are applied – depicted in Figure 3.2(a) –, the distance between the movable plates and the corresponding electrodes is d_0.

When the system is forced to leave the equilibrium position by an external acceleration $\vec{a}_{ext}$, two effects can be observed: the mechanical and the electrostatic. Considering Figure 3.2(b), the origin of the reference system O_1, the movable plate mass m, and disregarding the movable plate thickness, the external force, $\vec{F}_{ext}$,

$$\begin{aligned} \vec{F}_{ext} &= m\vec{a}_{ext} \\ &= ma_{ext}\hat{i} \\ &= F_{ext}\hat{i} \end{aligned} \tag{3.1}$$

pushes the structure to move from the initial vertical position $\vec{x}_i$,

$$\vec{x}_i = 0\hat{i}\,, \tag{3.2}$$

to the final one $\vec{x}_f$,

$$\vec{x}_f = x(t)\hat{i}\,, \tag{3.3}$$

causing a vertical displacement $\vec{r}$,

$$\begin{aligned} \vec{r} &= \left(x_f - x_i\right)\hat{i} \\ &= x(t)\hat{i}\,, \end{aligned} \tag{3.4}$$

where $\hat{i}$ is the unit vector in the vertical direction.

The mechanical spring exerts a force, $\vec{F}_{mech}$, on the same direction of $\vec{r}$, but on the opposite orientation, forcing the system to restore the relaxed state, as stated by Hooke's law [75]. The spring force is given by Equation 3.5 and is related to the displacement, $\vec{r}$, and the spring constant, k, which depends on the spring stiffness.

$$\begin{aligned} \vec{F}_{mech} &= -k\vec{r} \\ &= -kx(t)\hat{i} \\ &= -F_{mech}\hat{i} \end{aligned} \tag{3.5}$$

On the other hand, the resultant electrostatic force acting upon the movable plate, $\vec{F}_{elec}$, is given by Equation 3.6, and the forces $\vec{F}_1$ and $\vec{F}_2$ correspond, respectively, to the forces that the inferior and the superior fixed electrodes perform upon the movable plate.

$$\vec{F}_{elec} = \vec{F}_1 + \vec{F}_2 \tag{3.6}$$

A suitable approach to obtain the electrostatic forces relies on the virtual work method – that express the forces by the derivative of the energy stored in the system [76]. Considering that the movable plate is able to move in the x-axis direction under the action of the electrostatic force $\vec{F}$ – while the excitation voltage V, applied to the fixed electrode, remains constant –,

any given elementary work dW of the electrostatic force induces a change dE_c in the potential energy stored in the capacitor – of capacitance C – and an energy loss dE_s spent by the source to maintain the voltage constant. Since V remains constant, dE_c is only due to the capacity chance, dC. The energy provided by the source compensates the work of the electrostatic force and the change in the potential energy, as given by Equation 3.7.

$$dE_s = dW + dE_c\,, \tag{3.7}$$

where

$$dE_s = dCV^2 \tag{3.8}$$

and

$$dE_c = \frac{1}{2}dCV^2\,. \tag{3.9}$$

Then, substituting equations 3.8 and 3.9 in Equation 3.7:

$$dW = \frac{1}{2}dCV^2\,. \tag{3.10}$$

Since $dW = \vec{F} \cdot d\vec{x}$, it is possible to write the general expression for the electrostatic force as Equation 3.11.

$$\vec{F} = \frac{1}{2}V^2\frac{dC}{dx}\hat{i} \tag{3.11}$$

For the parallel-plate capacitor, the expression of its capacitance [75] is given in Equation 3.12:

$$C = \frac{\varepsilon A}{d}\,, \tag{3.12}$$

where A is the plate area, d is the plate separation, and ε is the dielectric material permittivity constant.

Considering, firstly, the system 1 formed by the movable plate and the actuation capacitor inferior fixed electrode, the capacitance C_1 – given by Equation 3.13 – corresponds to the case when the system moves along the arrow of Figure 3.2(b), reducing the gap distance to $d_0 - x(t)$.

$$\begin{aligned} C_1 &= \frac{\varepsilon A}{d_0 - x(t)} \\ &= \frac{\varepsilon A}{d_0\left(1 - x(t)/d_0\right)} \\ &= \frac{C_0}{\left(1 - x(t)/d_0\right)}\,, \end{aligned} \tag{3.13}$$

where

$$C_0 = \frac{\varepsilon A}{d_0}\,. \tag{3.14}$$

So, applying the quotient differentiation rule [77]:

$$\begin{aligned} \frac{dC_1}{dx\,(t)} &= \frac{-C_0\left(-1/d_0\right)}{\left(1 - x\,(t)/d_0\right)^2} \\ &= \frac{C_0}{d_0}\frac{1}{\left(1 - x\,(t)/d_0\right)^2}\,. \end{aligned} \tag{3.15}$$

Back to Equation 3.11, using Equation 3.15, the electrostatic force $\vec{F}_1$ that the inferior fixed electrode performs on the movable plate is:

$$\vec{F}_1 = \frac{1}{2} V_1^2 \frac{C_0}{d_0} \frac{1}{\left(1 - x\left(t\right)/d_0\right)^2} \hat{i}\,, \tag{3.16}$$

where V_1 is the potential difference between the electrodes.

The same procedure can be applied to the system 2, formed by the superior fixed electrode and the movable plate, resulting in the force that the first exerts on the last one, as Equation 3.17, when the gap distance changes from d_0 to $d_0 + x(t)$, with a potential difference V_2 between the electrodes.

$$\vec{F}_2 = -\frac{1}{2} V_2^2 \frac{C_0}{d_0} \frac{1}{\left(1 + x\left(t\right)/d_0\right)^2} \hat{i} \tag{3.17}$$

In this way, the resulting electrostatic force of Equation 3.6 can be written as Equation 3.18:

$$\vec{F}_{elec}\left(x(t)\right) = \left[\frac{1}{2} V_1^2 \frac{C_0}{d_0} \frac{1}{\left(1 - x\left(t\right)/d_0\right)^2} - \frac{1}{2} V_2^2 \frac{C_0}{d_0} \frac{1}{\left(1 + x\left(t\right)/d_0\right)^2}\right] \hat{i}\,. \tag{3.18}$$

In the electrostatic force feedback that is used for counteracting the inertial force in null measurement systems, two different voltages V_1 and V_2 are applied across the electrodes of Figure 3.2 for actuation [21, 78]. However, if a common-mode voltage is applied, with $V_1 = V_2 = v(t)$, Equation 3.18 reduces to:

$$\begin{aligned} \vec{F}_{elec}\left(x(t)\right) &= \frac{1}{2} \frac{C_0}{d_0} v^2\left(t\right) \left[\frac{1}{\left(1 - x\left(t\right)/d_0\right)^2} - \frac{1}{\left(1 + x\left(t\right)/d_0\right)^2}\right] \hat{i} \\ &= \frac{1}{2} \frac{C_0}{d_0} v^2\left(t\right) \left[\frac{\left(1 + x\left(t\right)/d_0\right)^2 - \left(1 - x\left(t\right)/d_0\right)^2}{\left(1 - x\left(t\right)/d_0\right)^2 \left(1 + x\left(t\right)/d_0\right)^2}\right] \hat{i} \\ &= 2 \frac{C_0}{d_0^2} v^2\left(t\right) \frac{x\left(t\right)}{\left[1 - \left(x\left(t\right)/d_0\right)^2\right]} \hat{i}\,. \end{aligned} \tag{3.19}$$

For small displacements relative to the gap size, i.e. $x(t) \ll d_0$, Equation 3.19 can be simplified, resulting in a linear dependence between the electrostatic force and the displacement, as shown in Equation 3.20.

$$\begin{aligned} \vec{F}_{elec}\left(t\right) &\cong \frac{2 C_0 v^2\left(t\right)}{d_0^2} x\left(t\right) \hat{i} \\ &= F_{elec} \hat{i} \end{aligned} \tag{3.20}$$

Thereby, unlike the spring force, the resulting electrostatic force amplifies the effect of the external force; thus, and a positive feedback phenomenon takes place.

Since a MEMS capacitive accelerometer is a second order mechanical system, it can be described as a second order differential equation [79], as given in Equation 3.21, where F_{ext} and F_{elec} are the magnitude of the external and the electrostatic forces, respectively.

$$m \frac{d^2 x\left(t\right)}{dt^2} + b \frac{dx(t)}{dt} + k x(t) = F_{ext} + F_{elec}\,, \tag{3.21}$$

where b is the damping coefficient. The accelerometer resonance frequency, ω_n, when no voltage is applied to system, is

$$\omega_n = \sqrt{\frac{k}{m}} \tag{3.22}$$

and the DC sensitivity is

$$S = \frac{m}{k}\,. \tag{3.23}$$

When the system is operated in the base band, with $\omega \ll \omega_n$, the quasi-static regime takes place, and the dynamic behavior can be neglected $\left(\frac{dx(t)}{dt} \approx 0\right)$. As a result, the mechanical force balances the others, and Equation 3.21 reduces to:

$$kx(t) = F_{ext} + F_{elec}\,. \tag{3.24}$$

Using equations 3.1 and 3.20 to write the forces magnitude and replacing in Equation 3.24:

$$kx\,(t) = ma_{ext} + \frac{2C_0 v^2\,(t)}{d_0^2} x\,(t)\ . \tag{3.25}$$

Assuming that the microaccelerometer voltage actuation $v(t)$ is periodical, with amplitude V_0, and Δk is the change in the spring constant caused by the feedback,

$$\Delta k = \frac{2C_0 V_0^2}{d_0^2}\,, \tag{3.26}$$

it is possible to rearrange Equation 3.25 to obtain an equation for the MEMS accelerometer output signal, $x(t)$:

$$\begin{aligned} x\,(t) &= \frac{ma_{ext}(t)}{k - \Delta k\left(\frac{v(t)}{V_0}\right)^2} \\ &= \frac{m}{k}\frac{1}{1 - \beta\left(\frac{v(t)}{V_0}\right)^2} a_{ext}(t)\,, \end{aligned} \tag{3.27}$$

where β is the modulation constant:

$$\beta = \frac{\Delta k}{k}\,. \tag{3.28}$$

So, the closed-loop time-variable gain $G(t)$ is given by:

$$\begin{aligned} G(t) &= \frac{x\,(t)}{a_{ext}\,(t)} \\ &= \frac{m}{k}\frac{1}{1 - \beta\left(\frac{v(t)}{V_0}\right)^2}\,. \end{aligned} \tag{3.29}$$

Using Mason's gain formula [80], it is possible to schematically represent the system in a block diagram, as shown in Figure 3.3, in the case of the quasi-static operation – in which the operating frequency is much lower than the mechanical resonance frequency, as previously mentioned.

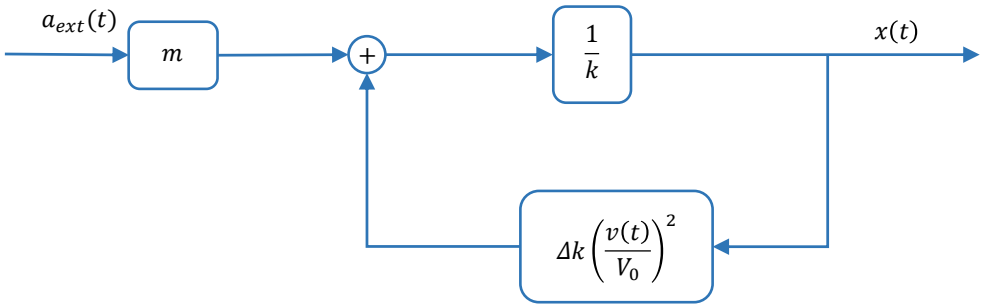

Figure 3.3: Block diagram of MEMS quasi-static model for common-mode actuation. Adapted from Cretu, Rocha, and Wolffenbuttel [22, p. 1408].

Considering Equation 3.27 for the system output, in the case of a cosine actuation voltage, $v_1(t) = V_1 \cos(\omega t)$, applied to an accelerometer 1 – which has mass m_1, elastic constant k_1, and modulation constant β_1 –, the accelerometer output $x_1(t)$ is:

$$\begin{aligned} x_1(t) &= \frac{m_1}{k_1} \frac{1}{1 - \beta_1 \left(\frac{V_1 \cos(\omega t)}{V_1}\right)^2} a_{ext}(t) \\ &= \frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(2\omega t)} a_{ext}(t) \\ &= \frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t)} a_{ext}(t), \end{aligned} \tag{3.30}$$

and the closed-loop gain – according to Equation 3.29 – is given by Equation 3.31:

$$\begin{aligned} G_1(t) &= \frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(2\omega t)} \\ &= \frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t)}, \end{aligned} \tag{3.31}$$

where the angular frequency ω_d is twice the excitation frequency ω, i.e., $\omega_d = 2\omega$.

The dependence of $G_1(t)$ in Equation 3.31 on the voltage suggests the possibility of a more advanced use of the structural coupling between the mechanical and electrostatic fields: by applying a correlation method and taking advantage of the positive feedback induced by the electrostatic forces, it is possible to calculate the spectral component of the input mechanical acceleration signal [22], as described in Subsubsection 3.1.1.1.

Based on Equation 3.31, it is possible to analyze the positive feedback stability [81] as a function of the system parameters. The system is unstable – with $G_1(t) \to \infty$ – if

$$1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t) = 0. \tag{3.32}$$

Rearranging the terms, Equation 3.32 can be solved to β_1 to ensure the stability:

$$\beta_1 \neq \frac{2}{1 + \cos(\omega_d t)}. \tag{3.33}$$

Besides, since the feedback is positive, the gain $G_1(t)$ of Equation 3.31 must be positive,

$$G_1(t) = \frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t)} \geq 0\,, \tag{3.34}$$

where m_1, k_1 and β_1 are all positive – since they relate to the accelerometer physical parameters. Then it is possible to reduce the analysis of the positive feedback to

$$1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t) \geq 0 \Rightarrow \beta_1 \leq \frac{2}{1 + \cos(\omega_d t)}\,. \tag{3.35}$$

Combining the conditions of stability – Equation 3.33 – and positive feedback – Equation 3.35 –, the modulation constant β_1 must satisfy:

$$\beta_1 < \frac{2}{1 + \cos(\omega_d t)}\,. \tag{3.36}$$

Considering the extreme values of $\cos(\omega_d t)$:

$$\cos(\omega_d t) = 1 \Rightarrow \beta_1 < 1\,; \tag{3.37}$$

$$\cos(\omega_d t) = 0 \Rightarrow \beta_1 < 2\,; \tag{3.38}$$

$$\cos(\omega_d t) = -1 \Rightarrow \beta_1 < \infty\,, \tag{3.39}$$

and, therefore,

$$\beta_1 < 1\,. \tag{3.40}$$

Using Equations 3.26 and 3.28, Equation 3.40 results in the stability and positive feedback conditions based on the actuation voltage amplitude:

$$\frac{2C_0 V_0^2}{k d_0^2} < 1 \Rightarrow V_0 < d_0\sqrt{\frac{k}{2C_0}}\,. \tag{3.41}$$

So, the critical value of the voltage amplitude is given by Equation 3.42.

$$V_{cr} = d_0\sqrt{\frac{k}{2C_0}} \tag{3.42}$$

3.1.1.1 Spectral extraction

To better understand how the spectral extraction scheme works on the MEMS analyzer, it is crucial to understand the Fourier analysis, whose fundamental aspects are described in Appendix A.

To obtain the desired spectral component $A(j\omega_d)$ from the mechanical acceleration input signal at a desired frequency ω_d, using Euler's formula [50] to write

$$e^{-j\omega_d t} = \cos(\omega_d t) - j\sin(\omega_d t)\,, \tag{3.43}$$

the Fourier integral – Equation A.52 – results in Equation 3.44.

$$\begin{aligned} A\left(j\omega_d\right) &= \int_{-\infty}^{\infty} a_{ext}\left(t\right) e^{-j\omega_d t} dt \\ &= \int_{-\infty}^{\infty} a_{ext}\left(t\right) \left[\cos\left(\omega_d t\right) - j\sin\left(\omega_d t\right)\right] dt \\ &= \int_{-\infty}^{\infty} a_{ext}\left(t\right) \cos\left(\omega_d t\right) dt - j\int_{-\infty}^{\infty} a_{ext}\left(t\right) \sin\left(\omega_d t\right) dt \\ &= A_c\left(\omega_d\right) - jA_s\left(\omega_d\right) , \end{aligned} \tag{3.44}$$

where $A_c\left(\omega_d\right)$ and $A_s\left(\omega_d\right)$ correspond, respectively, to the real and imaginary parts – also called in this text as cosine and sine portions – of the spectral component $A\left(j\omega_d\right)$ of the external acceleration in the Cartesian form.

In this case, the expression for the spectral component real part

$$A_c\left(\omega_d\right) = \int_{-\infty}^{\infty} a_{ext}\left(t\right) \cos\left(\omega_d t\right) dt \tag{3.45}$$

can be seen as a correlation method. The correlation function is defined as [50]:

$$\phi_{xy}\left(t\right) = \int_{-\infty}^{\infty} x\left(t+\tau\right) y\left(\tau\right) d\tau , \tag{3.46}$$

where τ is the time shift between the signals.

The correlation – or cross-correlation – function of two signals is a measure of the similarity between them as a function of the time shift [82, 83]. Besides, the cross-correlation of $x(t)$ and $y(t)$ in time domain corresponds to the product of both functions amplitude spectra in frequency domain [84]. If the functions are similar – i.e, if one is the time-shifted version of another –, the correlation produces a large positive number, while small correlation values are obtained if the functions are not alike.

Consequently, the cosine portion of the spectral component $A(j\omega_d)$ corresponds to the correlation between the external acceleration $a_{ext}(t)$ and a cosine signal $\cos(\omega_d t)$ with zero time lag. Similarly, the imaginary part of the spectral component, given by Equation 3.47 is the cross-correlation of the external acceleration and a sine signal, $\sin(\omega_d t)$.

$$A_s\left(\omega_d\right) = \int_{-\infty}^{\infty} a_{ext}\left(t\right) \sin\left(\omega_d t\right) dt \tag{3.47}$$

Then, it is possible to apply a correlation method to obtain both the real and imaginary parts of the spectral component $A(j\omega_d)$.

The MEMS spectrum analyzer closed-loop gain, however, is not a pure sine or cosine signal. In fact, Cretu, Rocha, and Wolffenbuttel [22] reported the presence of the DC and higher

order coefficients in the Fourier series of the closed-loop gain – which is closely investigated in Section 4.1 and Appendix D. This problem can be partially compensated by using a differential structure, with a second accelerometer actuated in quadrature [19, 22]. Therefore, back to Equation 3.27 the accelerometer 2, with mass m_2, elastic constant k_2, and modulation constant β_2, has the actuation signal $v_2(t) = V_2 \sin(\omega t)$ and an output signal $x_2(t)$ according to Equation 3.48.

$$\begin{aligned} x_2(t) &= \frac{m_2}{k_2} \frac{1}{1 - \beta_2 \left(\frac{V_2 \sin(\omega t)}{V_2}\right)^2} a_{ext}(t) \\ &= \frac{m_2}{k_2} \frac{1}{1 - \frac{\beta_2}{2} + \frac{\beta_2}{2}\cos(2\omega t)} a_{ext}(t) \\ &= \frac{m_2}{k_2} \frac{1}{1 - \frac{\beta_2}{2} + \frac{\beta_2}{2}\cos(\omega_d t)} a_{ext}(t), \end{aligned} \tag{3.48}$$

and, again, $\omega_d = 2\omega$.

Now, the closed-loop gain is

$$\begin{aligned} G_2(t) &= \frac{m_2}{k_2} \frac{1}{1 - \frac{\beta_2}{2} + \frac{\beta_2}{2}\cos(2\omega t)} \\ &= \frac{m_2}{k_2} \frac{1}{1 - \frac{\beta_2}{2} + \frac{\beta_2}{2}\cos(\omega_d t)}. \end{aligned} \tag{3.49}$$

Thereby, the differential output

$$\begin{aligned} x_{dif}(t) &= x_1(t) - x_2(t) \\ &= \left(\frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t)} - \frac{m_2}{k_2} \frac{1}{1 - \frac{\beta_2}{2} + \frac{\beta_2}{2}\cos(\omega_d t)} \right) a_{ext}(t) \end{aligned} \tag{3.50}$$

leads to a differential gain

$$\begin{aligned} G_{dif}(t) &= \frac{x_{dif}(t)}{a_{ext}(t)} \\ &= G_1(t) - G_2(t) \\ &= \left(\frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t)} - \frac{m_2}{k_2} \frac{1}{1 - \frac{\beta_2}{2} + \frac{\beta_2}{2}\cos(\omega_d t)} \right). \end{aligned} \tag{3.51}$$

Ideally, all the accelerometers parameters are equal – i.e., $m_1 = m_2 = m$, $k_1 = k_2 = k$ and $\beta_1 = \beta_2 = \beta$ – and the voltages amplitudes are the same – $V_1 = V_2 = V$ –, so Equation 3.50 is reduced to

$$x_{ideal}(t) = \frac{m\beta}{k} \frac{1}{\left(1 - \beta + \frac{\beta^2}{4} - \frac{\beta^2}{4}\cos^2(\omega_d t)\right)} \cos(\omega_d t)\, a_{ext}(t), \tag{3.52}$$

then the ideal differential gain is given by:

$$G_{ideal}(t) = \frac{m\beta}{k} \frac{1}{\left(1 - \beta + \frac{\beta^2}{4} - \frac{\beta^2}{4}\cos^2(\omega_d t)\right)} \cos(\omega_d t). \tag{3.53}$$

Thus, the differential output $x_{ideal}(t)$ in the ideal case of identical accelerometers contains a signal proportional to $a_{ext}(t)\cos(\omega_d t)$, where a_{ext} is the acceleration input, and $\omega_d = 2\omega$ and ω is the angular frequency of the electrostatic driving force.

Thereby, it is possible to obtain $A_c(\omega_d)$ through the integration of the differential output signal $x_{ideal}\,(t)$ of Equation 3.52. It should be emphasized that, since this last equation is proportional to $\cos(2\omega t)$, the real part of the spectral component, $A_c(\omega_d)$, will be obtained at a frequency ω_d corresponding to twice the frequency ω of the actuation signal.

Equation 3.53 presents two other important portions to be analyzed. First, the term $m\beta/k$ modifies the amplitude of the system differential gain, suggesting that, with no previous information about the system parameters, the spectrum obtained will present a scale that does not match the real absolute values of the spectral components. Consequently, in applications where the vibrations amplitudes are critical, this gain must be compensated. The other portion that requires attention is related to the modulation constant β and $\cos^2(\omega_d t)$, observed in the denominator of Equation 3.53.

As noted, the use of the two identical accelerometers in the differential configuration – also called twin-accelerometers structure – eliminates the DC and odd coefficients of the gain Fourier Series, making it closer of a cosine signal and, therefore, the spectral extraction scheme approximates the ideal correlation method. Consequently, the influence of the term proportional to $\cos^2(2\omega t)$ in the denominator of Equation 3.53 can be neglected in cases where $\beta < 1$, since it little interferes in the spectrum obtained with the microaccelerometers

To obtain $A_s(\omega_d)$, a similar quadrature actuation, with the same excitation angular frequency ω, but with a phase difference of $\pi/4$, can be used – i.e., $v_1(t) = V_1 \cos\left(\omega t - {}^{\pi}/_{4}\right)$ and $v_2(t) = V_2 \sin\left(\omega t - {}^{\pi}/_{4}\right)$. These actuation signals generate a differential output signal similar to Equation 3.52, but proportional to $a_{ext}\,(t)\sin(2\omega t)$, as shown in Equation 3.54.

$$x_{dif}\,(t) = \frac{m\beta}{k}\frac{1}{\left(1-\beta+\frac{\beta^2}{4}-\frac{\beta^2}{4}\sin^2(\omega_d t)\right)}a_{ext}\,(t)\sin(\omega_d t) \tag{3.54}$$

Similarly to the procedure to obtain the cosine component, the integration of Equation 3.54 leads to a reasonable approximation of the sinusoidal portion of the spectral component, $A_s(\omega_d)$, with frequency corresponding to twice the system excitation frequency, that is, $\omega_d = 2\omega$.

Since the phase difference between the external acceleration $a_{ext}(t)$ and the microaccelerometers excitation voltages are unknown, the amplitude $|A(\omega_d)|$ of the spectral component in the frequency ω_d can be calculated by using Equation 3.55:

$$|A(\omega_d)| = \sqrt{A_c^2(\omega_d) + A_s^2(\omega_d)}\,. \tag{3.55}$$

The strategy described can be implemented using a time-multiplexing method, as shown in the block diagram of in Figure 3.4 for the quadrature driven twin-accelerometers scheme. To accomplish this, the actuation voltages on the left, in darker tones, are first used, obtaining the real part $A(\omega_d)$ after the integration of the differential output signal; then, the voltages on the right, in lighter tones, are applied, obtaining the imaginary part of $A(\omega_d)$ with the integration of the output signal. In that way, by varying the frequency ω of the electric actuation, the mechanical vibration spectrum is mapped.

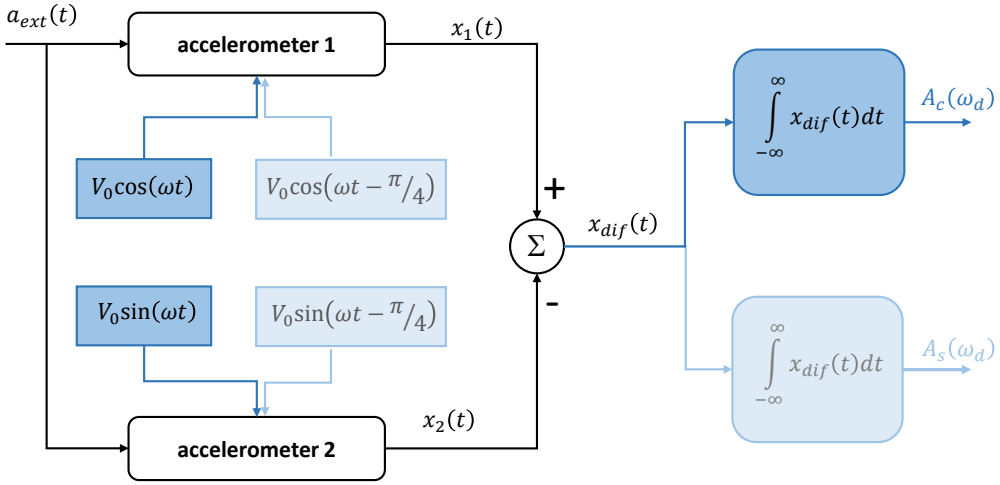

Figure 3.4: Block diagram of the mechanical spectrum analyzer differential structure. Adapted from Rocha, Cretu, and Wolffenbuttel [19, p. 1262].

The results obtained by Cretu, Rocha, and Wolffenbuttel [22] with the real-time spectrum analyzer are very close to the FFT for the same acquisition time. Similar to the FFT approach, the result of the real-time spectrum mapping depends on the integration time, and this characteristic is discussed on Appendix D. The obtained MEMS spectrum analyzer presents potential for low-power, low-cost applications, and this is its main advantage.

However, the authors highlight two critical factors [19, 22]. The first is the trade-off between the achievable sensitivity and the distortion level, which is determined by the actuation voltages amplitudes. The other issue is related to the matching of the accelerometers in the twin-structure, since any mismatch between the accelerometers can result in spurious frequencies in the spectrum obtained. Rocha, Cretu, and Wolffenbuttel [19] cite the possibility of correcting these parameters mismatches by using different voltage amplitude for each accelerometer of the twin-structure, settling their sensitivity; the authors, although, do not indicate how to choose the actuation voltages.

In this context, a tuning methodology to overcome the parameters differences, considering the distortion-sensitivity compromise and based on the system closed-loop gain is proposed in the next chapter.

Chapter 4

Tuning: Proposal and Methodology

In this chapter, methods to tune MEMS spectrum analyzers sensitivity and distortion are proposed. The tuning problem is stated as a multi-objective optimization problem, where sensitivity and distortion are the cost functions and the optimization variables are the amplitudes of the excitation voltages. Also, an alternative mono-objective approach to the tuning optimization problem is described. This chapter also contains a detailed description of the methodology used to solve the optimization problems. The algorithms used and all the assumptions and modifications performed are described.

4.1 Tuning: sensitivity and distortion optimization

According to Equation 3.44, the correlation method for obtaining the spectral component $A(\omega_d)$ occurs efficiently when the external acceleration signal is multiplied by cosine and sine signals of angular frequency ω_d – to obtain $A_c(\omega_d)$ and $A_s(\omega_d)$, respectively. In the case of the spectrum analyzer system used, however, the signals multiplying the external acceleration are not pure cosines and sines even for the ideal case, as shown in Equations 3.52 and 3.54.

In non-ideal cases the scenario is worst than the ideal case: since the accelerometers physical parameters are not matched due to manufacturing differences, the system output, according to Equation 3.50, presents the DC and higher coefficients of the Fourier Series. Consequently, the spectrum obtained with real MEMS accelerometers shows frequencies non-existent in the actual input signal. To eliminate the unwanted spectrum components, the mismatches between the accelerometers parameters can be corrected by using a different voltage amplitude for settling the sensitivity of the accelerometers, as suggested by Rocha, Cretu, and Wolffenbuttel [19]. The authors, however, do not indicate a method to perform this adjustment of the excitation voltages, which results in changes in the microaccelerometers modulation constants β_1 and β_2. The development of a method to perform this adjustment is the main contribution of this dissertation.

In order to develop a methodology to tune the spectrum analyzers, it is necessary to define a measure of the device sensitivity, as well as of its distortion, as a function of the actuation voltages amplitude. This work's proposal is to use the Fourier Series of the time-

variable gain $G_{dif}(t)$, since the accelerometers modulation constants depend on the actuation voltages amplitudes – $\beta_1 = f(V_1)$ and $\beta_2 = f(V_2)$.

Observing the system closed-loop gain of Equation 3.31, it is clear that the integration of the output $x(t)$ corresponds to an approximation of the correlation when a proper integration period, T, is chosen. Besides, the presence of the quadratic actuation voltage, $v(t)$, in the feedback path of Figure 3.3 leads to a correlation with terms of double actuation frequency, i.e., $\omega_d = 2\omega$. To better understand the effects of the closed-loop gain format on the computation of the spectrum, the Fourier Series (FS) decomposition of the system closed-loop gain can be performed [50].

Considering a single accelerometer configuration and the approximation $n = 4$, it is possible to deduce the DC component and the cosine coefficients for the Fourier Series of Equation 3.31, as Equations 4.1 to 4.5, with the sine coefficients, $c_1, \ldots, c_4$, all equal to zero. The deduction of these expressions is clearly described in Appendix D.

$$a_{single,0} = \frac{m_{single}}{k_{single}} \left(\frac{1}{\sqrt{1-\beta_{single}}} \right) \tag{4.1}$$

$$b_{single,1} = \frac{2m_{single}}{k_{single}\beta_{single}} \left(-2 + \frac{2-\beta_{single}}{\sqrt{1-\beta_{single}}} \right) \tag{4.2}$$

$$b_{single,2} = \frac{2m_{single}}{k_{single}\beta_{single}} \left(4 - \frac{8}{\beta_{single}} + \frac{8-8\beta_{single}+\beta_{single}^2}{\beta_{single}\sqrt{1-\beta_{single}}} \right) \tag{4.3}$$

$$b_{single,3} = \frac{2m_{single}}{k_{single}\beta_{single}} \left(-6 + \frac{32}{\beta_{single}} - \frac{32}{\beta_{single}^2} + \frac{32-48\beta_{single}+18\beta_{single}^2-\beta_{single}^3}{\beta_{single}^2\sqrt{1-\beta_{single}}} \right) \tag{4.4}$$

$$b_{single,4} = \frac{2m_{single}}{k_{single}\beta_{single}} \left(8 - \frac{80}{\beta_{single}} + \frac{192}{\beta_{single}^2} - \frac{128}{\beta_{single}^3} + \frac{128-256\beta_{single}+160\beta_{single}^2-32\beta_{single}^3+\beta_{single}^4}{\beta_{single}^3\sqrt{1-\beta_{single}}} \right) \tag{4.5}$$

where m_{single}, k_{single}, and β_{single} are the accelerometer parameters – and this single accelerometer can be the accelerometer 1 or 2 of the twin-structure –, $a_{single,0}$ is the DC component, and in the notation $b_{single,j}$, j indicates the harmonic order.

Considering the differential scheme, the differential closed-loop gain has a Fourier Series composition that can be expressed by Equations 4.6 to 4.10.

$$a_{dif,0} = \frac{m_1}{k_1} \left(\frac{1}{\sqrt{1-\beta_1}} \right) - \frac{m_2}{k_2} \left(\frac{1}{\sqrt{1-\beta_2}} \right) \tag{4.6}$$

$$b_{dif,1} = \frac{2m_1}{k_1\beta_1} \left(-2 + \frac{2-\beta_1}{\sqrt{1-\beta_1}} \right) + \frac{2m_2}{k_2\beta_2} \left(-2 + \frac{2-\beta_2}{\sqrt{1-\beta_2}} \right) \tag{4.7}$$

$$b_{dif,2} = \frac{2m_1}{k_1\beta_1} \left(4 - \frac{8}{\beta_1} + \frac{8-8\beta_1+\beta_1^2}{\beta_1\sqrt{1-\beta_1}} \right) - \frac{2m_2}{k_2\beta_2} \left(4 - \frac{8}{\beta_2} + \frac{8-8\beta_2+\beta_2^2}{\beta_2\sqrt{1-\beta_2}} \right) \tag{4.8}$$

$$\begin{aligned} b_{dif,3} = {} & \frac{2m_1}{k_1\beta_1} \left(-6 + \frac{32}{\beta_1} - \frac{32}{\beta_1^2} + \frac{32-48\beta_1+18\beta_1^2-\beta_1^3}{\beta_1^2\sqrt{1-\beta_1}} \right) + \\ & + \frac{2m_2}{k_2\beta_2} \left(-6 + \frac{32}{\beta_2} - \frac{32}{\beta_2^2} + \frac{32-48\beta_2+18\beta_2^2-\beta_2^3}{\beta_2^2\sqrt{1-\beta_2}} \right) \end{aligned} \tag{4.9}$$

$$b_{dif,4} = \frac{2m_1}{k_1\beta_1}\left(8 - \frac{80}{\beta_1} + \frac{192}{\beta_1^2} - \frac{128}{\beta_1^3} + \frac{128 - 256\beta_1 + 160\beta_1^2 - 32\beta_1^3 + \beta_1^4}{\beta_1^3\sqrt{1-\beta_1}}\right) - \\ - \frac{2m_2}{k_2\beta_2}\left(8 - \frac{80}{\beta_2} + \frac{192}{\beta_2^2} - \frac{128}{\beta_2^3} + \frac{128 - 256\beta_2 + 160\beta_2^2 - 32\beta_2^3 + \beta_2^4}{\beta_2^3\sqrt{1-\beta_2}}\right) \tag{4.10}$$

where the index j in the notation $b_{dif,j}$ represents the harmonic order, m_1, k_1, and β_1 are the parameters of the accelerometer 1 in the twin-structure, and m_2, k_2, and β_2 are the parameters of the accelerometer 2.

In the ideal case – i.e., when the accelerometers parameters match perfectly, with $m_1 = m_2 = m$, $k_1 = k_2 = k$ and $\beta_1 = \beta_2 = \beta$ –, simplifying Equations 4.6 to 4.10, the ideal differential Fourier Series is:

$$a_{ideal,0} = 0 \tag{4.11}$$

$$b_{ideal,1} = \frac{4m}{k\beta}\left(-2 + \frac{2-\beta}{\sqrt{1-\beta}}\right) \tag{4.12}$$

$$b_{ideal,2} = 0 \tag{4.13}$$

$$b_{ideal,3} = \frac{4m}{k\beta}\left(-6 + \frac{32}{\beta} - \frac{32}{\beta^2} + \frac{32 - 48\beta + 18\beta^2 - \beta^3}{\beta^2\sqrt{1-\beta}}\right) \tag{4.14}$$

$$b_{ideal,4} = 0\,. \tag{4.15}$$

As a result, it is possible to describe the device sensitivity directly by means of the fundamental component of the closed-loop gain Fourier series, as stated by Equation 4.16, where the index i can refer to any of the three configurations – single, differential and ideal – investigated.

$$s\left(V_1, V_2\right) = b_{i,1} \tag{4.16}$$

In general, an accelerometer sensitivity is defined as the output voltage signal generated per unit input acceleration, in g [85]. In the proposed measure, the sensitivity also corresponds to the relation between the system output and the input acceleration, but, in this case, the output corresponds to the displacement $x(t)$ – and, therefore, the dimensional analysis reveals that the sensitivity is measured in $\mathrm{m/(m/s^2)}$ or, alternatively, in $\mathrm{m/g}$.

The higher the sensitivity, the easier it is to detect external acceleration spectral components of low amplitude. Thus, a high sensitivity is desirable and, therefore, the problem of optimizing this cost function should be written in terms of its maximization:

$$(P_1) \left| \begin{array}{ll} \textbf{maximize} & s\left(V_1, V_2\right) \\ \textbf{subject to} & 0 < V_1 < V_{cr} \\ & 0 < V_2 < V_{cr} \end{array} \right. , \tag{4.17}$$

where the boundary constraints related to V_1 and V_2 are added in order to guarantee the positive feedback stability.

Now, the distortion level measurement can be associated with the absolute values of coefficients other than the fundamental, and the sum can be normalized with respect to the absolute value of the fundamental component, as pointed out in Equation 4.18, and, again, the

index i can refer to the single, differential or ideal configuration. The distortion level, therefore, is non-dimensional.

$$d\left(V_1, V_2\right) = \frac{|a_{i,0}| + |b_{i,2}| + |b_{i,3}| + |b_{i,4}|}{|b_{i,1}|} \tag{4.18}$$

Although the continuous component does not influence the emergence of spurious spectral lines in the calculated spectrum, this component is considered in the distortion measure as it can conceal small amplitude vibrations.

In order to approximate the system closed-loop gain to the ideal signal – which can be a cosine or a sine, depending on the time interval considered –, it is necessary to reduce all the Fourier Series components that are different from the fundamental. This implies that the distortion level should be minimized:

$$(P_2) \left| \begin{array}{ll} \textbf{minimize} & d\left(V_1, V_2\right) \\ \textbf{subject to} & 0 < V_1 < V_{cr} \\ & 0 < V_2 < V_{cr} \end{array} \right. . \tag{4.19}$$

Thus, solving the optimization problem of Equation 4.19, the differences in the parameters of the twin-accelerometers can be compensated, making the real system response closer to the ideal answer.

4.1.1 Mono-objective tuning

An interpretation to the problem of optimizing both the distortion and the sensitivity can be developed based on the fact that there is a quantification of the distortion level. So, based on a maximum acceptable distortion level d_{max}, it can be inserted into an optimization problem as an inequality constraint, maintaining the sensitivity maximization:

$$(P_3) \left| \begin{array}{ll} \textbf{maximize} & s\left(V_1, V_2\right) \\ \textbf{subject to} & d\left(V_1, V_2\right) < d_{max} \\ & 0 < V_1 < V_{cr} \\ & 0 < V_2 < V_{cr} \end{array} \right. . \tag{4.20}$$

This approach can be of great interest for problems where the vibration spectrum characteristics are well known, so it is possible to determine, previously, if the application requires lower or higher distortion. This is the case of harmonic vibration in electric machines, which occur in certain known frequencies; therefore, it is important to obtain the mechanical vibration spectrum with low distortion level.

4.1.2 Multi-objective tuning

Now, combining Equation 4.17 and Equation 4.19, a multi-objective optimization problem that maximizes the device sensitivity at the same time that minimizes its distortion level

can be stated as Equation 4.21.

$$(P_4) \left| \begin{array}{ll} \textbf{maximize} & s\left(V_1, V_2\right) \\ \textbf{minimize} & d\left(V_1, V_2\right) \\ \textbf{subject to} & 0 < V_1 < V_{cr} \\ & 0 < V_2 < V_{cr} \end{array} \right. \tag{4.21}$$

Minimizing the distortion level makes the system closed-loop gain closer to the ideal signal, which can be compared to the pitch in musical instruments [86]. On the other hand, maximizing the sensitivity can be compared to another musical parameter: the loudness [86].

This tuning strategy can be used in situations where there is no prior knowledge on the spectrum vibration characteristics. Therefore, it is important to obtain several Pareto-optimal solutions, that can be tested in order to choose the best one to the application. This is the case, for example, of a hydraulic leak, in which the spectrum, even before the leaking, presents components all over it, and the leak frequency cannot be predicted.

Besides, looking back to Equation 4.20 in Subsection 4.1.1, it is possible to associate this problem to the multi-objective problem of P4: Equation 4.20 consists on the ε-constraint method for Equation 4.21, as described in Subsubsection 2.1.1.2.

4.2 Methodology

For the real-time spectrum analyzer simulation, the software MATLAB® was chosen to implement the accelerometer model, according to the expressions obtained for the accelerometers output – Equations 3.30 and 3.48 – and the multiplexing strategy shown in Figure 3.4. The external acceleration was simulated using sines and cosines signals in specific frequencies and with desired amplitudes; the sampling frequency of 10 kHz was chosen to ensure behavior close to that of a real signal. The integration of the output was performed using the MATLAB function `trapz`, which approximates the integral through the trapezoidal method using a discretization defined by the user. The simulations use `double` variables, that constructs the double data type according to IEEE Standard 754 for double precision, with precision of 10^{308} [87].

The physical parameters of the accelerometer described by Cretu, Rocha, and Wolffenbuttel [22] were used as rated values, as summarized in Table 4.1.

Table 4.1: Micro-accelerometer rated parameters.

Parameter	Value
Mass (m)	2.3 ng
Mechanical spring constant (k)	2.877 N/m
Zero-displacement actuation capacitance (C_0)	157 fF
Zero-displacement gap (d_0)	3 μm
Resonance frequency (f_r)	5.7 kHz
Quality factor (Q)	0.707

To simulate non-ideal cases, the physical parameters m, k, C_0, and d_0 of both accelerometers of the twin-structure were randomly chosen considering a range of 20% of the respective rated value – which is a high tolerance for industrial purposes, but a realistic case for a prototype. Then, the parameters of each accelerometer of the twin-structure corresponds to a given chosen set within the 20% range.

4.2.1 Simulation model validation methodology

To validate the analyzer system simulation, three different scenarios were chosen: the first is the single-accelerometer, where only one accelerometer, with all its parameters correspondent to the respective rated values, is used; the second case corresponds to the differential scheme with non-ideal accelerometers, i.e., when the parameters of the accelerometers differ from each other; finally, the third scenario corresponds to the ideal differential structure, with all the accelerometers parameters correspondent to their rated values. The parameters chosen to simulate the non-ideal case are the rated parameters to the first accelerometer of the twin-structure, while the second accelerometer has $m_2 = 1.2m_{rated}$, $k_2 = 0.97k_{rated}$, $C_2 = 1.1C_{rated}$, and $d_2 = 0.9d_{rated}$.

The spectrum analyzer was used with spacing of 1 Hz between each point of the spectrum, since the chosen integration period is 1 s for each one of the points. To ensure that the operation is quasi-static, the actuation and acceleration frequencies are much less than 5.7 kHz – the resonance frequency. Therefore, the acceleration frequency range chosen is from 0 to 500 Hz, which corresponds to 0 to 250 Hz for the actuation frequency. An external validation acceleration signal with amplitudes 30 m/s^2 at 53 Hz, 45 m/s^2 at 127 Hz, 9 m/s^2 at 255 Hz and 21 m/s^2 at 432 Hz, covering all the frequency range investigated was used to observe the parameters influence on the final spectrum.

4.2.2 Mapping methodology

In real physical applications, the accelerometers real parameters are not known. However, if the accelerometer parameters were known, it would be possible to solve the optimization problems of Equations 4.20 and 4.21 analytically by using the expressions for the Fourier series obtained in Subsubsection 3.1.1.1, summarized in Table 4.2 together with the respective closed-loop gain.

Other possibility besides the analytical resolution of the optimization problems, using the same expressions, is to explore all voltage combinations to evaluate their influence on the accelerometers sensitivity and distortion, in an exhaustive search. When performing an exhaustive search, each possible solution is evaluated and the best of all is chosen. For small size problems, this is an acceptable strategy; however, for more complex and larger problems – the case of most real systems – this type of search becomes impracticable [29].

Since in this work the spectrum analyzer twin-accelerometers are simulated, their parameters are known and it is possible to map the distortion and the sensitivity surfaces as functions

of the voltages applied to the accelerometers. In real cases, those parameters are unknown, but the hypothesis assumed here is useful to understand the problem and to validate the tuning strategies. The proposed tuning, however, does not depend on the mapping to work properly.

Table 4.2: References to the expressions for the closed-loop gain and the composition of the Fourier series for single-accelerometer, non-ideal twin-accelerometers, and ideal twin-accelerometers.

Device configuration	**Closed-loop gain**	**Fourier Series**				
		a_0	b_1	b_2	b_3	b_4
Single-accelerometer	$G_1(t)$ (3.31)	$a_{single,0}$ (4.1)	$b_{single,1}$ (4.2)	$b_{single,2}$ (4.3)	$b_{single,3}$ (4.4)	$b_{single,4}$ (4.5)
Non-ideal twin-accelerometers	$G_{dif}(t)$ (3.51)	$a_{dif,0}$ (4.6)	$b_{dif,1}$ (4.7)	$b_{dif,2}$ (4.8)	$b_{dif,3}$ (4.9)	$b_{dif,4}$ (4.10)
Ideal twin-accelerometers	$G_{ideal}(t)$ (3.53)	$a_{ideal,0}$ (4.11)	$b_{ideal,1}$ (4.12)	$b_{ideal,2}$ (4.13)	$b_{ideal,3}$ (4.14)	$b_{ideal,4}$ (4.15)

The mapping was performed considering spacing of 0.1 V to both actuation voltages, V_1 and V_2, and minimum amplitude of 0.5 V. The maximum amplitude to each one of the voltages was calculated considering the positive feedback and its stability criterion, given in Equation 3.42. Besides the single-accelerometer, the non-ideal differential structure – with the same parameters described in Subsection 4.2.1 –, and the ideal differential scheme, the mapping was performed to other ten different non-ideal twin-accelerometers. Each accelerometer of these differential structures presents a given chosen set of parameters residing in the 20% margin around the rated values.

4.2.3 Tuning methodology

In this work, the solutions to the optimization problems are found through a DE algorithm, which was based on the concepts of Section 2.2 and adapted to solve the spectrum analyzer problem.

The reason for choosing to use a meta-heuristic to solve the problem is the same reason why the problem exists: due to the production process, accelerometers differ from each other, so their parameters vary and are unknown. Therefore, it is not possible to apply a deterministic strategy to solve the optimization problems. Besides, the optimization problems are highly non-linear, since they are non-linear on the modulation constants, β_1 and β_2, which are themselves non-linear on the excitation voltages amplitudes, V_1 and V_2, respectively – due to the quadratic relation shown in equations 3.26 and 3.28. Thus, it is necessary to look for alternatives, such as systems identification [88], or the use of meta-heuristics. This last option was chosen because of the operational advantages it presents – as previously described in Section 2.2.

This strategy can be considered as an initial calibration of the spectrum analyzer system: in a controlled environment, by subjecting the accelerometers to known test accelerations, it is possible to define the drive voltages to solve the optimization problem. Besides, an adjustment of the overall gain, w, of the system can be performed during this tuning, so the absolute value

of the spectral components can be computed.

The implementation of the DE algorithm to solve the optimization problems of equations 4.20 and 4.21 have particularities and differences between each other; however, some considerations are general and must be applied to all of them. This is the case of the memory. This feature was chosen to prevent the spectrum from being mapped multiple times to the same voltage amplitudes combination. This memory is a variable that contains each combination of voltages tested – properly quantized – and their respective sensitivity and distortion level. During the algorithm initialization, the memory is also initialized with the first generation elements that are distinct from each other. Thus, at the cost function evaluation stage for the trial vector, the spectrum is mapped only if the voltage combination has not been tested previously. This strategy reduces the number of spectrum evaluations – especially when the algorithm presents its population almost stable and, thus, close to the Pareto-set – and can be especially advantageous depending on the strategy used to handle boundary constraints.

To apply the proposed tuning in any of the stated optimization problems, attention should be given to the time interval chosen. Besides the fact that the strategy depicted in Figure 3.4 involves a time-multiplexing method to obtain the real and imaginary parts of the spectral component, the spectrum is obtained by varying the driving voltage frequency, ω, so it is important that the time interval to be used for the Fourier series calculation considers only one of these actuation frequencies and the computing of the real or the imaginary parts. Therefore, the acquisition time required to approximate the system closed-loop gain Fourier series corresponds to one integration period, since the tuning is performed considering just one frequency and during just one part of the time-multiplexing strategy used on the device.

An external acceleration signal, with a component of 10 $\mathrm{m/s^2}$ at 440 Hz, was used. The Fourier series of the closed-loop gain was computed in the interval in which the excitation voltage has a frequency of 220 Hz and the cosine portion of the spectral component is considered. However, unlike the validation simulation, only a small portion of the spectrum was raised, in 440 Hz, and only for the attainment of the spectrum real part. The integration period remained in 1 s; although the integration itself is not performed during the tuning, this period also corresponds to the acquisition interval and, therefore, interferes in the quality of the approximation of the gain Fourier series. This approximation was performed by means of MATLAB `fit` function, with order 4 approximation. Since the DE algorithm is not deterministic, it was chosen to perform five tests for each non-ideal twin-accelerometer structure.

The meta-parameters N, F, and CR of differential evolution algorithm were chosen according to the literature indications, and so $N = 20$, $F = 0.9$, and $CR = 0.9$. The feasible solution space for both voltages has been reduced to $[0.0, 9.0]$ – with the upper limit slightly smaller than the stability limit of the feedback loop for the rated accelerometer parameters, according to Equation 3.42 –, to the nearest decimal place. To deal with these boundary constraints, the four strategies briefly described in Subsection 2.2.1 – namely random reinitialization, bounce-back, resampling and brick-wall – were used in order of comparison.

The algorithm evolution and the stopping criteria consistency are evaluated using

the same non-ideal twin-structure, which parameters are given in Subsection 4.2.1. Then, the ten different non-ideal structures, that had their distortion and sensitivity mapped in Subsection 4.2.2, are used to assess the tuning average behavior.

4.2.3.1 Mono-objective tuning methodology

This approach deals with the inequality constraint by means of the death penalty described in Subsection 2.2.1.

The stopping criteria chosen combines the maximum number of generations – $g_{max} = 200$ – and the *NoAcc*, as described in Subsection 2.2.1. This last one monitors how many trial vectors are selected in each generation and the algorithm stops if the number of trial vectors selected is less than 10% of the population size in the last 10% of the maximum number of generations. This corresponds to say that at least 90% of the population is retained in the last generations, so it can be understood that the optimization converged. The final optimum answer corresponds to the best solution in the last generation, i.e., the feasible solution with maximum sensitivity.

4.2.3.2 Multi-objective tuning methodology: approach 1

In this first approach for the multi-objective optimization problem of Equation 4.21, the strategy to adapt the DE to a multi-objective problem proposed by Price, Storn, and Lampinen [28] and described in Subsubsection 2.2.1.1 was used to achieve an approximation of the Pareto-set. The idea of evaluating the Pareto-dominance between the elements of the last generation is also considered.

To this approach, the same stopping criteria of the mono-objective approach is used. However, unlike the mono-objective case – in which the result of the optimization converge to a single point –, here the population corresponds to a Pareto-set approximation.

4.2.3.3 Multi-objective tuning methodology: approach 2

The second approach used to the multi-objective problem resolution consists of an adaptation of Robič and Filipič [49] proposal of immediately replacing the selected vector in the current population. The other strategies in this proposal – as the truncation – are not implemented. So, here it is proposed to select the trial vector $\mathbf{u}_{i,g}$ if it dominates the target vector $\mathbf{x}_{i,g}$ or if there is no domination between them – which is the same selection criterion of Approach 1. The main difference is that, if selected, $\mathbf{u}_{i,g}$ takes the place of $\mathbf{x}_{i,g}$ in the next and in the current population. Like the multi-objective Approach 1, the stopping criteria described for the mono-objective problem are used, and the Pareto-dominance between the population of the last generation is evaluated.

4.2.4 Tuning validation methodology

Besides the comparison with the analytical optimum results obtained with the distortion and sensitivity mapping, to validate the results of the tuning performed in the spectral analyzer, an external acceleration signal different from that used for calibration was applied to the spectrum analyzer. The signal is the same used for the simulation validation – defined in Subsection 4.2.1 – and was applied to all non-ideal twin-accelerometers. The amplitudes of the excitation voltages were chosen for different relations between distortion and sensitivity considering the tuning results.

Chapter 5

MEMS-based Spectrum Analyzer Simulation and Tuning

In this chapter, the main results obtained during the simulations performed are presented. First, the simulation model is validated using an external acceleration signal; the spectra and the closed-loop gains are analyzed for three different cases: a single-accelerometer, a non-ideal twin-accelerometers, and an ideal twin-accelerometers spectrum analyzer. Next, assuming that the accelerometers parameters are known, an exhaustive search for the Pareto-set is performed, mapping the distortion and sensitivity as a function of the voltages not only for the three cases investigated previously, but for other ten different twin-accelerometers with randomly chosen parameters around the rated values.

In real cases, where the accelerometers parameters are not known, it is necessary to determine the combination of voltages which satisfy a given distortion-sensitivity criterion. The results of mono- and multi-objective approaches proposed in this dissertation are shown in Section 5.3, using the mapping results to compare their efficiency. Finally, the tunings are validated using an external acceleration signal and the voltages amplitudes that ensure different distortion-sensitivity compromises.

5.1 Simulation model validation results

To validate the simulation model, excitation voltages $V_1 = V_2 = 7.6$ V were chosen, since they result in a modulation factor $\beta = 0.7$ – according to equations 3.26 and 3.28 and the parameters presented in Table 4.1 – and, thus, a low distortion level [19]. The acceleration signal is depicted in Figure 5.1(a).

The simulation results of Figure 5.1 refer to the use of a single-accelerometer (b), a non-ideal differential scheme (c), and the ideal twin-accelerometers (d). The accelerometers parameters in the single-accelerometer and in the ideal differential structure are the accelerometers rated parameters, as given in Table 4.1.

In general, the results shown in Figure 5.1 validate the simulation of the MEMS spectrum

analyzer since all three configurations are able to reproduce the spectral components, keeping the frequencies and amplitude ratios adequate. The absolute values, however, are considerably small, which relates to the presence of the accelerometers parameters in the expressions for the fundamental Fourier component – equations 4.2, 4.7, and 4.12 – since these parameters are small in microsystems.

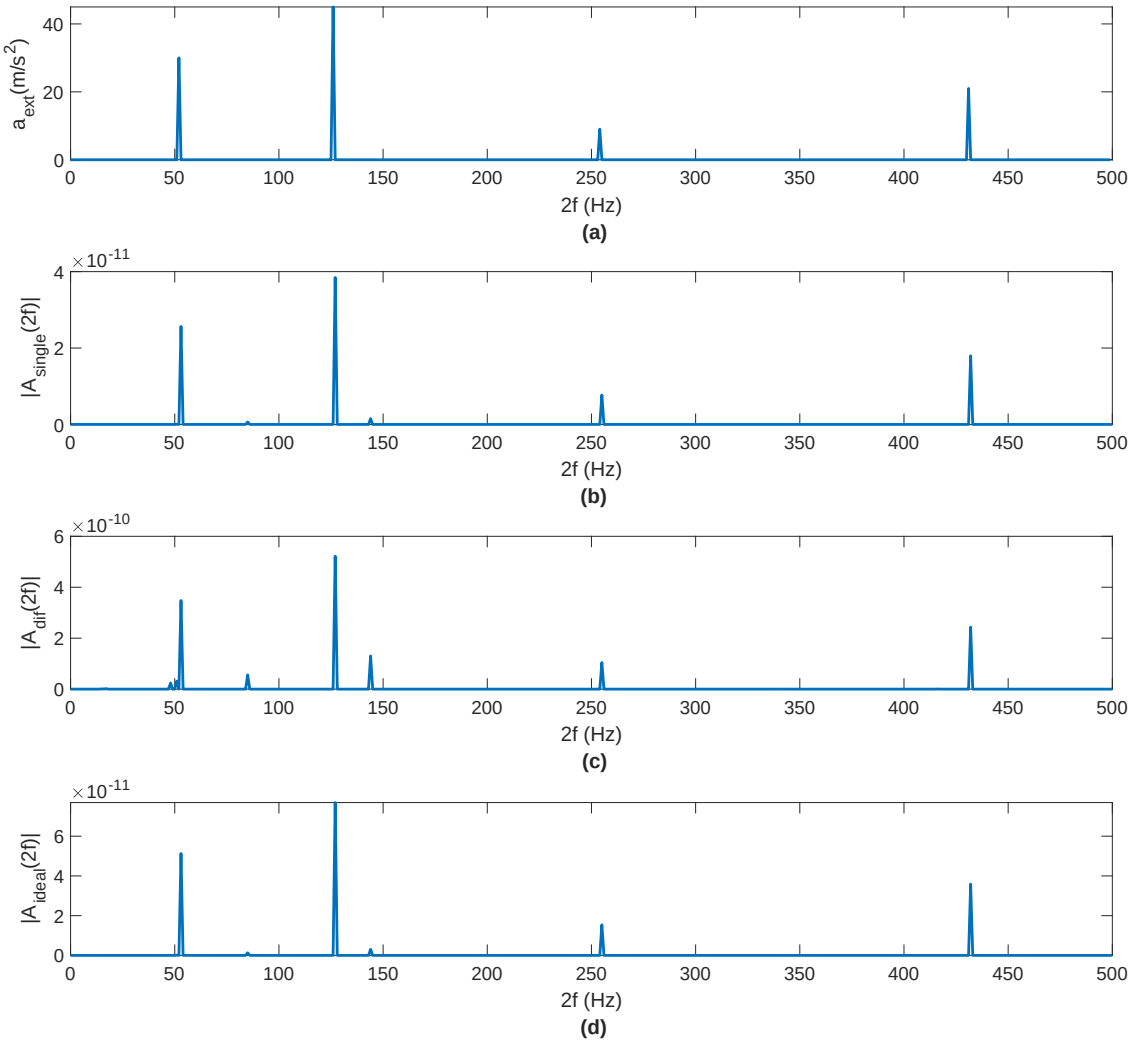

Figure 5.1: (a) Real external acceleration spectrum. Spectra obtained by the simulation of the MEMS analyzer with excitation voltages $V_1 = V_2 = 7.6$ V: (b) single accelerometer, (c) a non-ideal twin-structure, (d) ideal twin-accelerometers.

First, considering the single-accelerometer case (Figure 5.1(b)), the distortion observed in the spectrum is considerably small, presenting only an undesired component in 144 Hz, with an amplitude that represents only 3.99% of the spectrum largest component – that corresponds to the component at 127 Hz.

Now, the non-ideal twin-accelerometers of Figure 5.1(c) present more undesired components – at least four visible frequencies –, and the largest undesired amplitude in 144 Hz, representing 24.98% of the 127 Hz component.

Finally, the spectrum raised by the ideal twin-accelerometers – Figure 5.1(d) – shows the same distortion of 3.99% in 144 Hz as the single-accelerometer case, with the main difference in the amplitude scale.

In order to understand the cause of the spectrum distortions, the closed-loop gain of the system in the three cases under investigation can be used, as shown in Figures 5.2(a.1), (a.2), and (a.3), together with their respective coefficients of the Fourier series in (b.1), (b.2), and (b.3) – calculated using the MATLAB function `fit` to obtain the fourth order series approximation. Its important to emphasize that all these results are normalized with respect to the fundamental component. As discussed previously, any excitation frequency – within the proposed discretization of the spectrum frequencies – can be chosen for this analysis, and here the excitation frequency of 220 Hz was selected, resulting in a Fourier series fundamental frequency of 440 Hz. It is important to notice that, although only a small time interval is shown in Figures 5.2(a), the coefficients of the Fourier series were calculated using the time series of the entire acquisition period.

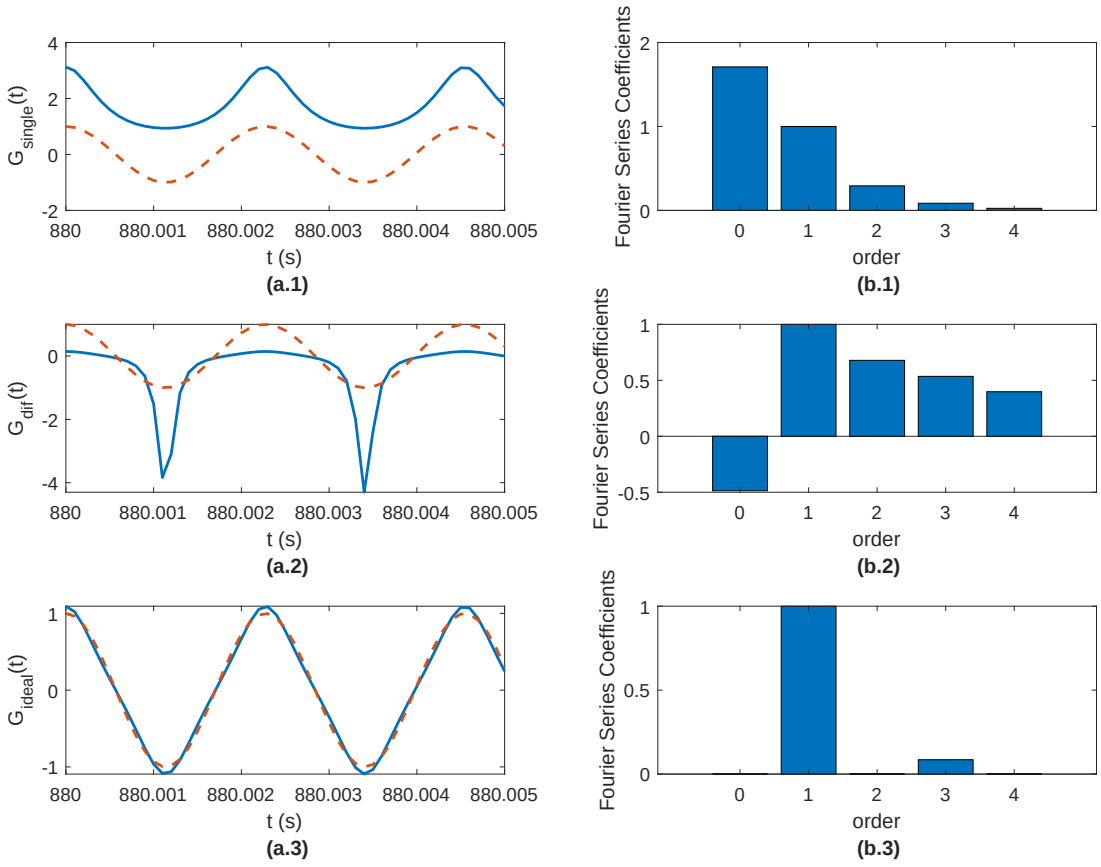

Figure 5.2: Closed-loop gain analysis for (1) single-accelerometer, (2) a non-ideal twin-structure, (3) ideal twin-accelerometers. (a) MEMS spectrum analyzer closed-loop gain for cosine component (solid line) and ideal cosine (dotted line). (b) Fourier series coefficients of closed-loop gain.

From Figure 5.2(a.1), it is clear that in the proposed time interval, referring to the cosine portion of the spectrum, the use of a single-accelerometer results in a closed-loop gain with a

shape similar to a cosine, despite the continuous component – which is the largest component of the Fourier series. The second and the third order coefficients – depicted in Figure 5.2(b.1) – are responsible by the shape deformations found in the closed-loop gain regarding the reference cosine.

Notwithstanding, in the non-ideal twin-accelerometers case, with differences between the accelerometers, the differential gain depicted by the continuous line in Figure 5.2(a.2) is considerably distorted in relation to the reference cosine. This fact is evidenced by the composition of the Fourier series, that presents the continuous component, besides non-zero coefficients of first to fourth order, as can be observed in the results of Figure 5.2(b.2). The Fourier series composition justifies the appearance of spurious harmonics in the spectrum of Figure 5.1(c).

On the other hand, the identical accelerometers result in a differential gain very close to the ideal cosine – as evidenced in Figure 5.2(a.3) –, confirmed by the Fourier series – which presents only the third order coefficient, with small amplitude, besides the fundamental component, as can be seen in Figure 5.2(b.3).

From the Fourier series composition, the results for the sensitivity (Equation 4.16) and the distortion level (Equation 4.18) for each case are obtained, as shown in Table 5.1.

Table 5.1: Distortion and sensitivity results for single-accelerometer, non-ideal twin-accelerometers, and ideal twin-accelerometers MEMS spectrum analyzers.

Configuration	Distortion	Sensitivity (m/(m/s^2))
Single-accelerometer	2.1117	8.54e-13
Non-ideal twin-accelerometers	2.0987	1.16e-11
Ideal twin-accelerometers	0.0859	1.71e-12

Even though the result shown in Table 5.1, the distortion does not visually influence the spectrum obtained with the single-accelerometer (Figure 5.1(b)), since the DC component is the responsible for the distortion value, as depicted in Figure 5.2(b.1).

For the non-ideal twin-accelerometers, despite the distortion slightly smaller than the single-accelerometer case, the spectrum is noticeably distinct from the actual input composition, as shown in Figure 5.1(c). This distortion associates with the Fourier series composition, that presents larger higher order coefficients (Figure 5.2(b.2)). However, the sensitivity measurement indicates a value higher than the single-accelerometer case.

In the ideal case, the distortion level reaches 4.07% of the single-accelerometer distortion and 4.09% of the non-ideal distortion. These results confirm the fact that the ideal differential configuration presents a lower distortion level when compared to the single-accelerometer topology and the non-ideal case of distinct accelerometers.

5.2 Mapping results

If the accelerometers parameters are assumed to be known, it is possible to map the distortion and the sensitivity surfaces as functions of the voltages applied to the accelerometers. In real cases, those parameters are unknown, but the hypothesis assumed in this section is useful to understand the problem and to validate the tuning strategies proposed in this dissertation.

The results for the distortion level (a) and the sensitivity (b) are shown in Figure 5.3 for the single-accelerometer structure (1), the non-ideal (2), and the ideal (3) differential structures, respectively, as a function of the excitation voltages amplitudes, V_1 and V_2. The Pareto-set is highlighted in purple in the decision space, which has the mapped points depicted in cyan (c); dominated (cyan), and non-dominated (purple) solutions – where this last one corresponds to the Pareto-front – are shown in the objective space (d). The accelerometers parameters used in the non-ideal twin-accelerometers are the same of Section 5.1.

From figures 5.3(a) and (b), it is possible to observe the conflicting characteristic of the objective functions, at least in part of the solution space. Although it is difficult to visualize this conflict between the distortion (Figure 5.3(a)) and the sensitivity (Figure 5.3(b)) to the differential cases (2 and 3), the resulting Pareto-front of Figure 5.3(d) for each case makes the compromise relation much more obvious. It is interesting to note the variation of the Pareto-set and the Pareto-front in Figures 5.3(c) and (d) for each differential case. Since the optimization problem depends on the accelerometer parameters, their values strongly influence the resulting dominance. It is interesting to note that the influence of the parameters is more pronounced in the distortion, which visually varies from the ideal to the non-ideal case, as can be seen from the Figures 5.3(a.2) and (a.3). The sensitivity, however, maintains a similar form in figures 5.3(b.2) (b.3), and it is clear that higher excitation amplitudes implies higher sensitivity.

A similar characteristic, with different formats for the distortion level as a function of the actuation voltages, can be noticed for ten different non-ideal twin-accelerometers, as depicted in Figure E.1(a) of Appendix E. In the same figure, the sensitivity (b) shows resembling appearance to the results of Figure 5.3(b). The Pareto-set (c) and the Pareto-front (d) vary in all the ten twin-accelerometers – since they present different parameters. Another interesting observation is the maximum voltages amplitudes for each differential structure: since the critical amplitude that ensures the feedback stability depends on the accelerometers parameters, the allowed voltages vary to each twin-accelerometers considered.

It is important to emphasize that the mapping results are presented here to demonstrate the optimization surface complexity, besides to validate the tuning results presented later in this text. In real physical cases, however, it is not possible to perform this mapping, since the accelerometers parameters vary among the rated values due to the manufacturing process and, therefore, are not known.

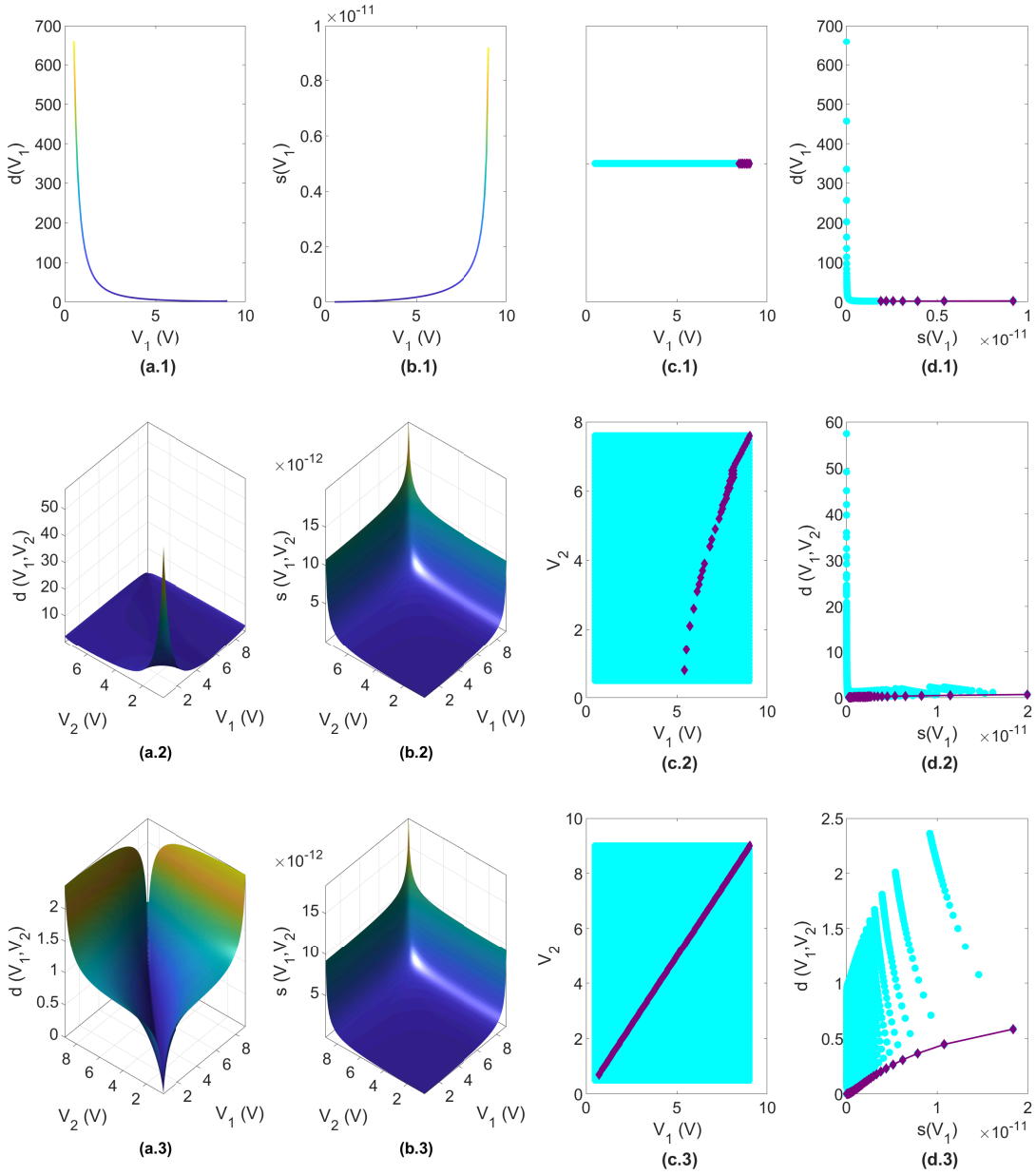

Figure 5.3: Mapping results for (1) single-accelerometer, (2) a non-ideal twin-structure, (3) ideal twin-accelerometers. (a) Mapped distortion. (b) Mapped sensitivity. (c) Mapped actuation voltages amplitudes V_1 and V_2 (cyan) and Pareto-set (purple) in the decision space. (d) Mapped dominated (cyan) and non-dominated (purple) solutions in the objective space for the MEMS spectrum analyzer.

5.3 Tuning results

The results of the spectrum analyzer tuning are provided, considering the methodology described in Subsection 4.2.3. First, the mono-objective methodology is investigated and the DE algorithm convergence is validated using the previously explored non-ideal twin-structure. The tuning is then applied to different non-ideal twin-accelerometers, which are also used to analyze the algorithm average behavior. Next, the two multi-objective approaches are evaluated considering the same steps of the mono-objective case.

5.3.1 Mono-objective tuning results

To apply the mono-objective tuning of Equation 4.20, it is first necessary to validate the algorithm convergence. So, the non-ideal twin-accelerometers – mapped in Section 5.2 – are used to evaluate the tuning evolution. This comparison is possible since, according to Equation 2.8, the mono-objective optimization problem of Equation 4.20 can be seen as the ε-constrained version of Equation 4.21. In Figure 5.4, the results for the bounce-back strategy and maximum distortion level of 0.8 are shown.

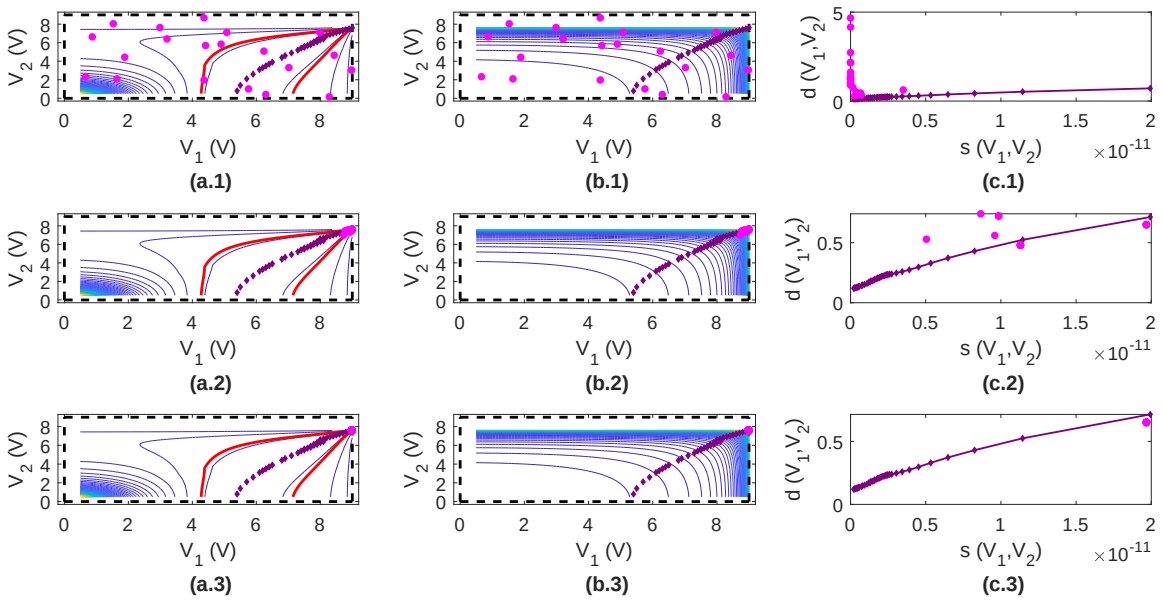

Figure 5.4: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

In that figure, the generations (magenta points) in the decision space are plotted over the distortion (a) and sensitivity (b) level curves obtained during the mapping of Section 5.2, with the feasible region for the algorithm limited by the black dotted lines; in purple, the results obtained by the mapping for the Pareto-set are highlighted. In the objective space, the evolution results are presented in magenta, together with the results of the mapping for the Pareto-front in purple. In the first column, the red level curve delimits the maximum distortion level considered in the constraint. The results refer to the first (1), the intermediate (2), and the last (3) generation.

In the first generation, the population spreads all over the decision space within the boundaries and, therefore, some population individuals present distortion level higher then the 0.8 constraint, as shown in Figure 5.4(a.1). The evolution over the decision space clearly shows that the populations search for the feasible region where the distortion level is smaller

than 0.8, as shown in Figures 5.4(a.2) and (a.3). On the other hand, observing the evolution over the sensitivity curves – Figures 5.4(b.1) to (b.3) –, it is clear that the algorithm tends to the higher level curves, solving the maximization problem. Since this is a mono-objective problem, the main idea is to stop the iterations when all the population converge to the same position on the decision space, which, in fact, occurs in this case – validating the proposed stopping criteria as well. Now, another interesting confirmation is that, indeed, this mono-objective approach is able to approximate a Pareto-optimal solution, as can be seen in Figure 5.4(c.3).

Analogous results are presented in Figure 5.5, but with maximum distortion level of 0.2. The main difference between these results and those for distortion level 0.8 is that the constraint region is smaller – which can be a problem to the algorithm convergence.

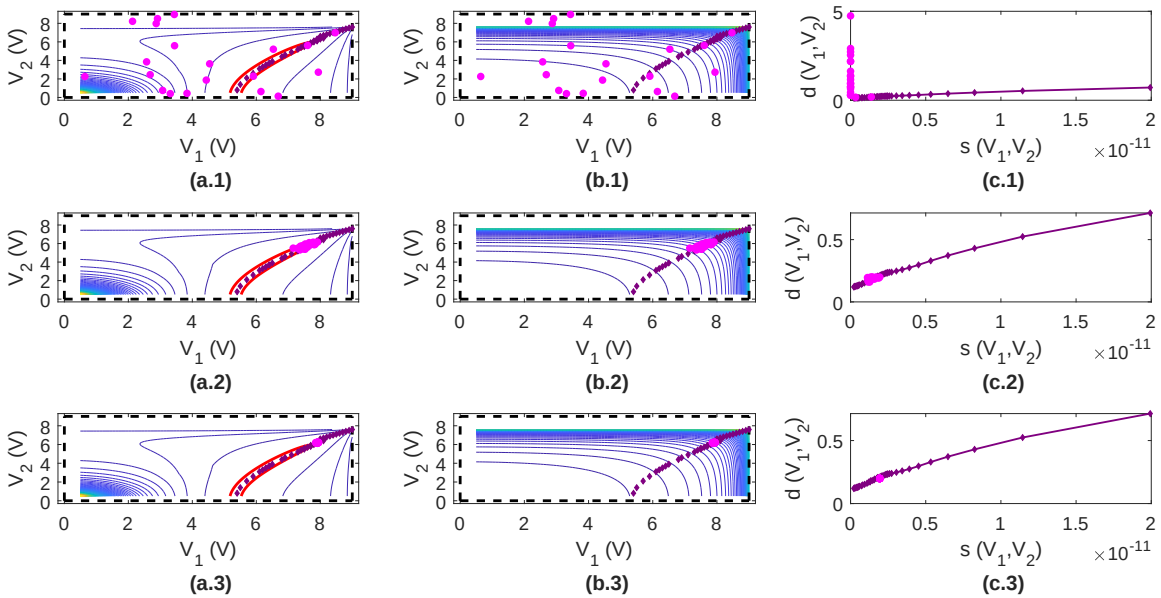

Figure 5.5: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

The results for the other strategies are shown in Appendix E. In figures E.2 and E.3, the evolution results for maximum distortion of 0.8 and 0.2, respectively, are shown for the brick-wall boundary strategy. For random reinitialization, the mono-objective evolution considering the same distortion constraints of 0.8 an 0.2 has its results presented in figures E.4 and E.5. Finally, in figures E.6 and E.7, are shown the results for maximum distortion levels of 0.8 and 0.2, respectively, for resampling boundary strategy.

It is important to evidence that this tuning results in only one point in the Pareto-optimal space – making analogy, again, with the multi-objective problem. Hence, this tuning is adequate

when the characteristics of spectrum monitored are well know and it is possible to determine one single distortion level that meets this mechanical vibration spectrum requirements. When these requirements change, a new tuning must be performed.

Again due to the non-deterministic behavior, to analyze the convergence results, it is necessary to use the average values and their respective standard deviations, as shown in Tables 5.2 and 5.3 for the convergence time, the number of generations and the number of spectrum evaluations to the four different boundary handling strategies. In Table 5.2 the results considering constraint of 0.8 distortion level are presented, while in Table 5.3 are shown the results for maximum distortion level of 0.2.

Table 5.2: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – maximum distortion level 0.8.

Strategy	Time (s)	Generations	Spectrum Eval.
Bounce-back	35.63 ± 7.99	51.20 ± 17.12	411.40 ± 75.08
Brick-wall	31.54 ± 11.83	62.20 ± 23.85	353.20 ± 84.59
Random reinit.	40.27 ± 11.80	61.80 ± 24.22	513.20 ± 115.49
Resampling	35.02 ± 5.48	58.60 ± 21.10	415.40 ± 31.63

Table 5.3: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – maximum distortion level 0.2.

Strategy	Time (s)	Generations	Spectrum Eval.
Bounce-back	39.27 ± 11.88	50.20 ± 27.77	520.00 ± 168.29
Brick-wall	27.97 ± 14.53	44.80 ± 27.93	344.60 ± 138.47
Random reinit.	51.17 ± 15.38	64.40 ± 25.39	637.00 ± 195.14
Resampling	39.54 ± 10.89	59.80 ± 22.60	513.60 ± 96.54

It is interesting to note that the convergence time is similar to bounce-back and resampling strategies, but the random reinitialization presents a larger convergence time and the brick-wall strategy, a smaller one. Besides, the average number of generations is smaller than the maximum allowed in the stopping criterion, indicating that the algorithm in fact converges – since the other stopping criterion, associated with the population stability, is the responsible to stop the iterations.

Looking to the number of spectral evaluations, since it is less than the product of the generations number and the individuals in the population, the approach of using the memory is adequate to reduce the total number of times that the device is tested. Since the tuning proposed is based on the MEMS analyzer system closed-loop gain, each time a new voltage combination must be evaluated it is necessary to test the system – which consists of applying the voltages signals with the test amplitudes, measuring the system output and computing the Fourier series of the closed-loop gain. Therefore, each test takes a time equivalent to the integration period to be performed, and the number of spectral evaluations is a critical factor to be analyzed.

As a result, the convergence time does not correspond to a realistic measure of the total time to perform the optimization. Since the spectrum must be mapped for each combination of the excitation amplitudes not tested before, and since each of this tests take the integration period – settled to 1 s –, the tuning in fact takes approximately the number of spectrum evaluations, in seconds, to converge.

Once the algorithm convergence is validated, it is applied to diverse non-ideal twin-accelerometers, choosing different limits for the maximum distortion. The final results obtained for the Pareto-set and the Pareto-front for maximum distortion levels of 0.8 – similar to those of figures 5.4(a.3), (b.3), and (c.3) – and 0.2 – similar to those of figures 5.5(a.3), (b.3) and (c.3) – are presented in Appendix E for ten different non-ideal twin-accelerometers. For the bounce-back boundary strategy, the mentioned results are shown in figures E.8 and E.9, respectively; for the brick-wall strategy, in figures E.10 and E.11; the results for random reinitialization boundary handling technique are presented in E.12 and E.13; and for resampling, in E.14 and E.15.

In these figures, there are some interesting results to highlight. The first one is the difference between the boundary limits for the optimization variables – in dotted black lines – and the mapping space, that considers the maximum voltages to ensure the positive feedback stability. Since the maximum voltage amplitude depends on the accelerometer parameters, this value is different for each accelerometer of the twin-structure. Consequently, it is possible to observe that, among the ten twin-structures, the maximum voltages can be larger or smaller than the boundary limits for each voltage V_1 and V_2 independently. This characteristic influences the resulting voltage combination obtained by the tuning. In some cases, the tuned voltages can appear outside the mapped decision space – as in figures E.10(a.8) and (b.8) for the twin-structure number 8, for example –, resulting in a unfeasible solution – as depicted in Figure E.10(c.8), with the result beyond the Pareto-front.

Another interesting characteristic is that the accelerometer parameters also strongly influence the format of the distortion level curves – as previously shown in the mapping results of Figure E.1. As a result, specially for smaller distortion levels, the constrained region can represent a small portion of the decision space – as the case of the twin-accelerometer 2 –, making difficult to the tuning algorithm to converge. The non-convergence is clear, for example, in Figure E.15(a.2), (b.2), and (c.2) for the resampling strategy using the twin-strutuctre 2 and the maximum distortion level of 0.2.

The mono-objective tuning does not consider the dominance between the objective function, but only the maximum sensitivity that resides inside the distortion constraint. Therefore, in some cases, the tuned voltages can be outside the Pareto-set – as in Figure E.8(a.3) and (b.3) –, resulting in a dominated solution – as evidenced in Figure E.8(c.3).

Considering maximum distortion of 0.8 and 0.2 for ten twin-accelerometers, the mean and standard deviation of time, number of generations, and number of spectrum evaluations are shown in tables 5.4 and 5.5, respectively, when bounce-back is applied to handle the boundary constraints. The analogous results for the brick-wall, random reinitialization and resampling for maximum distortion level of 0.8 are presented in tables E.1, E.3, and E.5, respectively, in

Appendix E. In this appendix, tables E.2, E.4, and E.6 present the results considering the same boundary strategies, but a constraint of 0.2 for the distortion level.

Table 5.4: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, bounce-back boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	20.63	±	4.98	34.60	±	7.64	252.20	±	79.87
(2)	21.28	±	4.94	38.80	±	10.33	283.00	±	88.27
(3)	25.12	±	3.32	44.00	±	13.38	349.20	±	54.37
(4)	24.17	±	1.24	44.00	±	13.17	326.80	±	42.39
(5)	45.09	±	12.16	71.60	±	29.02	620.40	±	145.96
(6)	21.51	±	2.80	40.40	±	11.04	284.20	±	64.71
(7)	20.34	±	3.09	35.80	±	8.53	274.20	±	73.61
(8)	41.78	±	11.09	71.40	±	29.90	568.60	±	118.81
(9)	30.39	±	5.46	53.40	±	18.80	412.80	±	63.42
(10)	27.14	±	3.05	46.40	±	14.48	364.60	±	50.73
mean	**27.75**	±	**5.21**	**48.04**	±	**15.63**	**373.60**	±	**78.21**

Table 5.5: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, bounce-back boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	30.05	±	6.26	43.60	±	13.39	367.60	±	58.75
(2)	33.40	±	16.12	27.40	±	14.31	511.40	±	246.90
(3)	36.38	±	8.84	53.20	±	19.18	505.80	±	106.27
(4)	34.36	±	7.26	57.60	±	21.72	470.60	±	89.30
(5)	46.47	±	13.57	67.00	±	27.46	641.00	±	192.46
(6)	42.80	±	10.26	61.80	±	22.84	588.60	±	116.21
(7)	37.33	±	8.17	56.40	±	20.17	525.60	±	81.35
(8)	41.35	±	12.10	59.80	±	23.35	588.00	±	156.33
(9)	38.99	±	9.27	58.60	±	21.31	559.00	±	109.78
(10)	38.17	±	8.10	59.80	±	21.89	529.00	±	113.62
mean	**37.93**	±	**9.99**	**54.52**	±	**20.56**	**528.66**	±	**127.10**

Now, the average behavior of the boundary strategies can be compared using the last line of Table 5.4, as well as the other strategies for the maximum distortion level of 0.8. This last line corresponds to the mean value for all the accelerometers of the mean between the tests performed, given in Table 5.6. Analogous results are presented for maximum distortion 0.2 in Table 5.7.

These results present the dependence on the boundary handling strategy. Since each one of them uses a different way to handle the points beyond the boundaries, each one influences differently the convergence time and, specially, the number of spectrum evaluations – depending on the strategy, it is more or less probable that diverse points in the feasible decision space are evaluated.

Table 5.6: Average behavior of mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – maximum distortion level 0.8.

Strategy	Table	Time (s)	Generations	Spectrum Eval.
Bounce-back	5.4	27.75 ± 5.21	48.04 ± 15.63	373.60 ± 78.21
Brick-wall	E.1	28.05 ± 9.19	61.52 ± 23.83	369.32 ± 106.35
Random reinit.	E.3	37.90 ± 8.56	58.92 ± 21.81	532.14 ± 95.84
Resampling	E.5	29.66 ± 5.87	54.20 ± 20.17	408.86 ± 77.42

Table 5.7: Average behavior of mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – maximum distortion level 0.2.

Strategy	Table	Time (s)	Generations	Spectrum Eval.
Bounce-back	5.5	37.93 ± 9.99	54.52 ± 20.56	528.66 ± 127.10
Brick-wall	E.2	28.72 ± 9.94	56.58 ± 22.09	381.36 ± 119.33
Random reinit.	E.4	38.53 ± 9.44	54.90 ± 21.21	543.56 ± 108.56
Resampling	E.6	32.50 ± 6.50	52.00 ± 18.13	454.40 ± 67.69

From both Tables 5.6 and 5.7, the random reinitialization strategy presents a higher convergence time and higher number of spectrum evaluations than the other strategies. The higher number of evaluations is due to the probability of generating new vectors that have not been mapped before, which also increases the convergence time. On the other hand, the brick-wall strategy results in lower convergence time and smaller number of spectrum evaluations, despite the greater number of generations. In the brick-wall, the higher number of generations can be associated with the fact that the variables that correspond to the Pareto-optimum can lie near the bounds, so it is improbable to generate solutions that do not violate the boundaries, slowing the population progress. The smaller number of spectrum evaluations for the brick-wall can be associated with the penalty applied – since the target vector is chosen whenever the boundaries are violated by the trial vector. So it is probable that the trial vectors generated by brick-wall have been mapped before, consequently reducing the total number of evaluations and the convergence time. These characteristics are the same of tables 5.2 and 5.3 for the twin-structure which parameters are known. Again, it is important to point that the algorithm convergence time does not correspond to the tuning total time, since each spectrum evaluation requires data acquisition during 1 s.

Since this tuning is non-deterministic – a fact associated with the DE –, it is important to perform it in a loop and choose the best results among all the tests to tune the MEMS spectrum analyzer. The best result can be understood as the voltage combination, between those that meet the distortion requirement, that results in maximum sensitivity.

5.3.2 Multi-objective tuning results: approach 1

First, to validate the implementation of the DE algorithm for the problem of Equation 4.21, the same non-ideal twin-accelerometers of the previous section was used, since their parameters are known. The results of one test for the populations evolution, together with the

distortion and sensitivity solutions are shown in Figure 5.6 to the bounce-back strategy used to deal with the boundary constraints. In Figure 5.6, the generations in the decision space are plotted over the sensitivity and distortion level curves, while in the objective space are presented, besides the results of the evolution, the results of the mapping of Section 5.2.

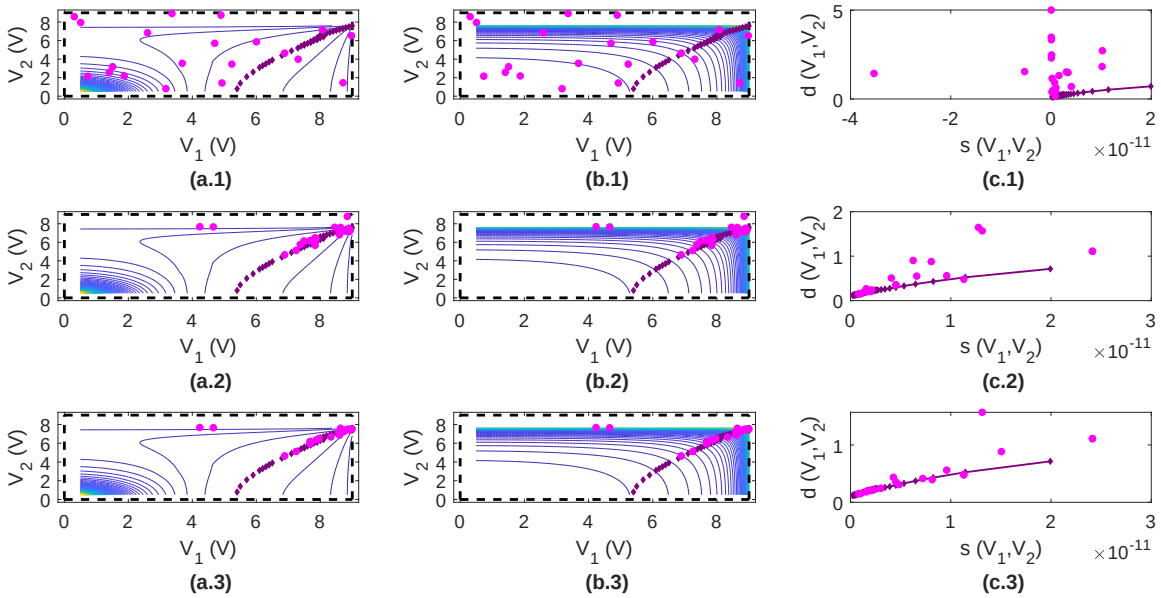

Figure 5.6: Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, bounce-back boundary strategy, test 1. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

It is possible to observe that the proposed tuning algorithm evolves to the approximation of the Pareto-solutions. In the decision space, it is possible to observe that the populations evolve to regions with lower distortion level and higher sensitivity, approximating the Pareto-set. In the objective space, the evolution to the Pareto-front is even easier to observe. The evolution results for one test of the other boundary strategies are similar and shown in Appendix E, in figures E.16, E.17 and E.18 for the brick-wall, random reinitialization, and resampling.

Since the meta-heuristic used to solve the optimization problem is non-deterministic, as previously stated, the results of different tests can be slightly diverse. In order to visualize those variations and compare the results, the evolution results are shown in Figure 5.7 for a test different from that of Figure 5.6.

Comparing the plots in the last line of Figure 5.6 with Figure 5.7, it is possible to note that the approximation of the Pareto-front is slightly different, specially concerning the sampling of the Pareto-front. Since the DE multi-objective adaptation used does not have a tool to ensure diversity between the points in the Pareto-front, the points can be randomly more or less spaced on the different tests.

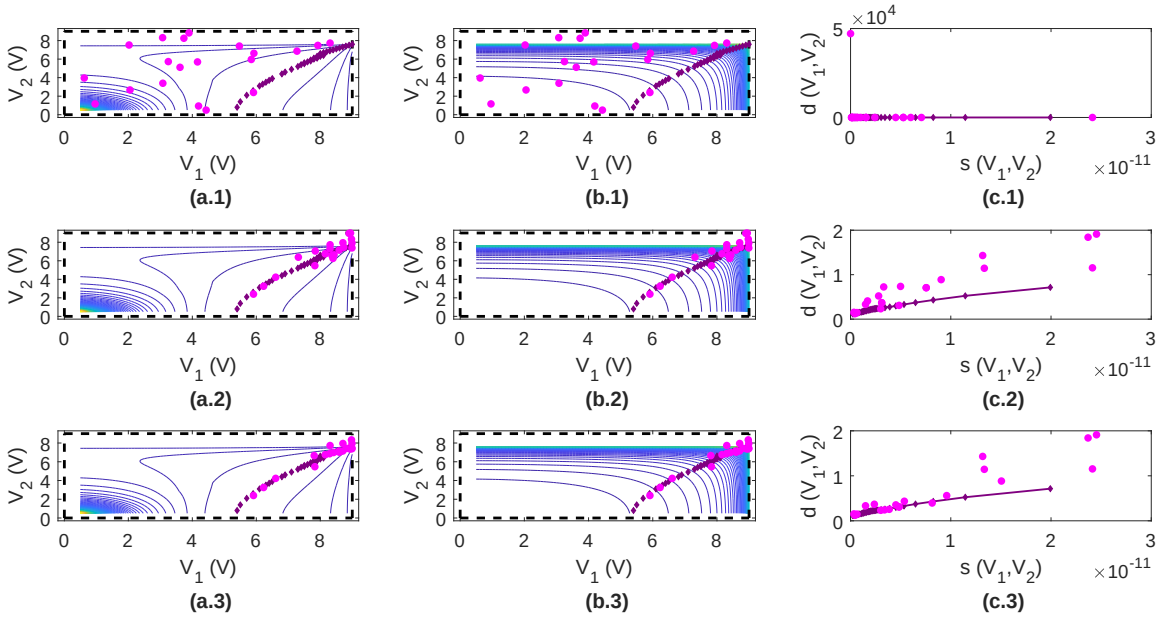

Figure 5.7: Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, bounce-back boundary strategy, test 2. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

Given this non-deterministic behavior of the DE algorithm in – with different number of elements and spacing in the Pareto-set for each trial –, to obtain a better sampling of the Pareto-set, it is possible to combine the final results of all the tests. Besides, as previously stated, it is important to use mean and standard deviation to evaluate the algorithm average convergence. The results obtained combining all the tests are shown in Figure 5.8 for the bounce-back boundary strategy using the non-ideal twin-structure. The analogous results for brick-wall, random reinitialization and resampling strategies are presented in figures E.19, E.20, and E.21.

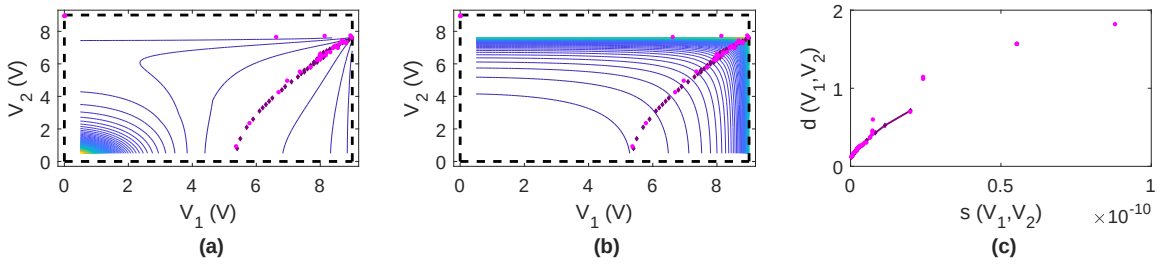

Figure 5.8: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

Now, the mean and standard deviation results obtained to convergence time, number of spectrum evaluations and number of generations by using each one of the different strategies described to deal with the boundaries constraints are shown in Table 5.8 for the same non-ideal differential structure.

Table 5.8: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – approach 1.

Strategy	Time (s)			Generations			Spectrum Eval.		
Bounceback	50.21	±	8.23	63.80	±	28.37	686.80	±	132.45
Brickwall	44.17	±	24.75	66.00	±	45.15	589.60	±	274.87
Random reinit.	64.47	±	7.25	90.80	±	61.90	857.60	±	172.46
Resampling	57.97	±	4.39	60.60	±	11.59	805.40	±	96.11

The first result that can be noticed from Table 5.8 is the average time. Despite the complexity of the search for the optimization variables – associated not only with the presence of two optimization parameters, but also with its quadratic relation in the accelerometer model – and the process of obtaining the spectrum itself, the convergence time remains below 60 s since only a small portion of the spectrum is computed and used to the tuning for each combination of the operating voltages. It can be highlighted that the standard deviation found in this result is a consequence of the non-deterministic characteristic of the DE algorithm. As the mono-objective tuning, the convergence time is not a real measure of the total tuning time, that must consider the number of spectrum evaluations.

Another important characteristic is that the average number of spectrum evaluations is less than the number of generations multiplied by the number of the members in each population. This is due to use of the memory – as described in Chapter 4 – to avoid the spectrum evaluation in points previously mapped.

After the validation of the proposed multi-objective tuning, it is possible to apply the strategy to different non-ideal twin-accelerometers. In Table 5.9, the results obtained for ten different twin-accelerometer structures are shown, considering the bounce-back strategy. The mean values of the convergence time, as well as the number of spectrum evaluations and the number of generations, are presented along with their respective standard deviation values. Similarly, tables E.7 to E.9 in Appendix E present the mean results for the ten twin-accelerometers, considering the other strategies to deal with the bounds, and the same discussion made to the results of Table 5.8 remain valid.

Since Table 5.8 shows the behavior of the strategies subjected to a single twin-structure, these results do not represent the mean behavior of the strategies themselves. Now, in order to compare the average behavior of the boundary strategies, it is possible to use the last line of Table 5.9 – as well as the last line of the results for the other boundary strategies shown in Appendix E in tables E.7 to E.9, as summarized in Table 5.10. As well as the mono-objective approach, the average behavior of the multi-objective algorithm is influenced by the boundary

strategy used.

Table 5.9: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, bounce-back boundary strategy.

Structure	Time (s)	Generations	Spectrum Eval.
(1)	59.73 ± 10.16	175.20 ± 42.02	670.40 ± 236.91
(2)	55.35 ± 7.28	175.40 ± 55.01	612.60 ± 223.15
(3)	63.62 ± 8.86	131.60 ± 63.41	874.00 ± 56.00
(4)	59.45 ± 10.65	108.20 ± 55.09	870.80 ± 219.88
(5)	52.72 ± 8.82	55.00 ± 13.42	884.20 ± 137.24
(6)	41.81 ± 6.13	67.80 ± 19.15	645.20 ± 99.08
(7)	37.51 ± 13.23	96.00 ± 73.83	489.80 ± 217.00
(8)	55.48 ± 12.06	84.00 ± 66.77	856.60 ± 149.49
(9)	55.46 ± 9.14	90.00 ± 64.05	844.20 ± 192.26
(10)	47.19 ± 4.58	87.60 ± 63.08	693.60 ± 103.50
mean	**52.83 ± 9.09**	**107.08 ± 51.58**	**744.14 ± 163.45**

Table 5.10: Average behavior of multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – approach 1.

Strategy	Table	Time (s)	Generations	Spectrum Eval.
Bounce-back	5.9	52.83 ± 9.09	107.08 ± 51.58	744.14 ± 163.45
Brick-wall	E.7	47.94 ± 11.61	87.72 ± 42.68	717.00 ± 155.26
Random reinit.	E.8	70.11 ± 18.84	83.84 ± 46.33	978.46 ± 229.77
Resampling	E.9	62.46 ± 11.17	103.18 ± 54.12	862.62 ± 154.60

The combined results obtained for the Pareto-optimal approximation for the ten non-ideal twin-accelerometers, similarly to the results of Figure 5.8, are shown in figures E.22 to E.25 in Appendix E, for bounce-back, brick-wall, random reinitialization, and resampling boundary strategies, respectively. The combined results considers the dominance between them.

In all the boundary strategies, it is possible to observe – similarly to the mono-objective tuning – the influence of the accelerometers parameters in the results obtained, even though all the structures present some good approximations for the Pareto-optimum in some regions. In structures which the maximum amplitude voltages that ensure the stability of the positive feedback are higher than the DE boundaries – as the case of structures 1 and 2 –, the tuning is capable of approximating the Pareto-set that lies in the DE voltages boundaries; however, the Pareto-set that lies outside these boundaries is not appropriately approximated, and the tuning result corresponds to the voltage boundary. This highly influences the aspect of the Pareto-front, which approximates the mapping results just for a few points, returning dominated solutions – when comparing to the mapping results – for the others. In structures that present critical voltages to stability lower than the DE boundaries, the effect is the opposite, resulting in unfeasible solutions in the Pareto-front when the tuning results are compared to the mapping results; this is the case of structures 5 and 8. An interesting feature is observed in structures 3,

4, and 6, which present one of the maximum actuation voltages lower than the DE boundary and the other one higher; consequently, the results for the Pareto-optimum approximation present unfeasible and dominated solutions, besides the Pareto-set and Pareto-front correct approximations for some points. In the meantime, in cases where the DE boundaries are similar to the maximum voltages to the system stability, the results for the Pareto-set and the Pareto-front are good approximations to the mapping results, as depicted in the results of structures 7, 9, and 10. The unfeasible and dominated solutions found, although, do not invalidate the tunings, and their effects are explored in Section 5.4.

5.3.3 Multi-objective tuning results: approach 2

Similarly to the first approach, the results for the populations evolution, together with the distortion and sensitivity solutions are shown in Figure 5.9 for the bounce-back strategy. The evolution results for the other boundary strategies are presented in Appendix E, in Figure E.26 for brick-wall, in Figure E.27 for random reinitialization, and in Figure E.28 for resampling boundary strategy.

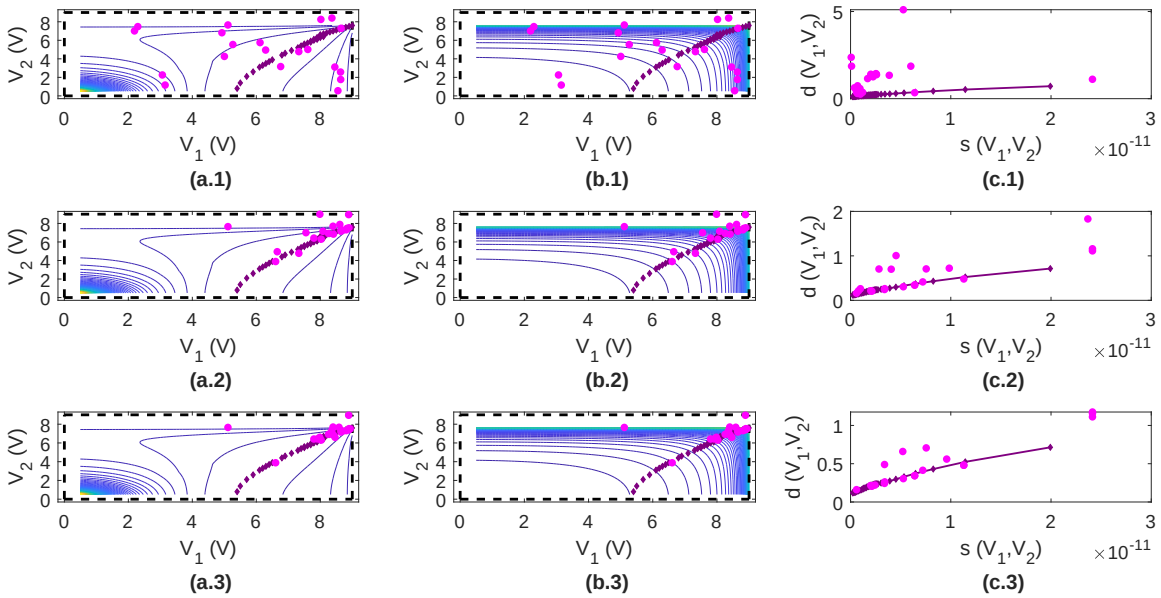

Figure 5.9: Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

The behavior of the populations evolution is analogous to that presented in Figure 5.7 for the approach 1: considering the distortion, the population seeks the minimum levels; considering sensitivity, the population tends to its maximum; considering the objective space results, the population approaches the Pareto-front appropriately. These results validate the proposed

approach to the selection, evidencing that the algorithm can converge when the alterations described in Subsubsection 4.2.3.3 are implemented in the differential evolution to solve the multi-objective tuning.

The mean and standard deviation results obtained to convergence time, number of spectrum evaluations, and number of generations by using each one of the different strategies described to deal with the boundaries constraints are shown in Table E.10 for the same twin-accelerometer structure. These results also validate the algorithm implementation, especially the proposed stopping criteria.

The final result to the optimization problem via this approach, similarly with the multi-objective approach 1, can combine the results of some trials to sample the Pareto-front in a greater number of points. The results for this twin-structure are given in Figure 5.10 for bounce-back strategy and in figures E.29, E.30, and E.31 for brick-wall, random reinitialization and resampling. The results are quite similar to those obtained with approach 1, since both the approaches are able to approximate the Pareto-optimal set.

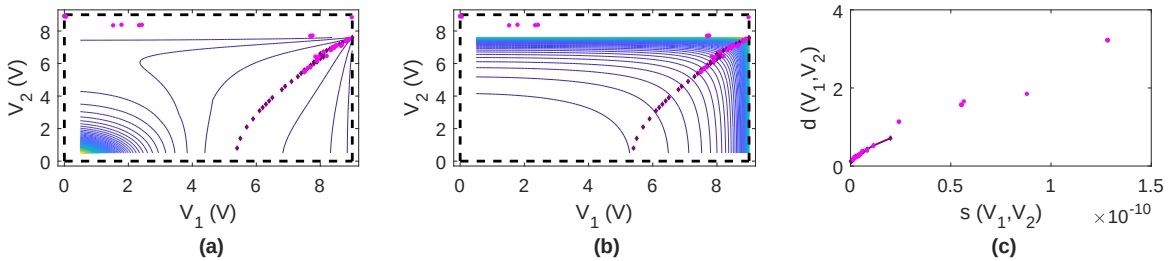

Figure 5.10: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

The analysis of the average behavior – presented in tables E.11 to E.14 – is similar to that presented in the mono-objective tuning and in the other multi-objective approach – varying according to the strategy used to handle the boundary constraints. However, it is possible to evaluate if this approach is capable of reducing the convergence time. The average behavior, given in Table 5.11, must be analyzed in comparison with those of the first approach – Table 5.10.

Table 5.11: Average behavior of multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers with different boundary strategies – approach 2.

Strategy	Table	Time (s)			Generations			Spectrum Eval.		
Bounce-back	E.11	54.04	±	8.83	107.50	±	57.30	737.62	±	161.89
Brick-wall	E.12	49.88	±	11.82	93.14	±	39.92	708.66	±	166.31
Random reinit.	E.13	66.33	±	19.61	83.62	±	44.58	975.58	±	268.45
Resampling	E.14	58.75	±	9.68	92.36	±	48.48	794.20	±	140.47

Generally speaking, approach 2 presents an average convergence time and number of generation very similar to the approach 1 behavior, with some small percentage reductions and increments, which can be possibly associated with the DE algorithm non-deterministic characteristic. However, the most important reduction is in the spectrum evaluation. Despite the low percentage difference – with the smallest reduction of 0.29% and the largest of 7.93%, when compared to approach 1 –, since this number highly influences the total tuning time – as previously discussed –, even this small improvement is advantageous. Since this proposal is based on Robič and Filipič [49], but the original proposal has other modifications and characteristics, it is possible to conclude that this characteristics may be determinant for a more prominent reduction of the total processing time. This fact, however, does not invalidate the results of this approach. The combined results are shown in Appendix E for ten different twin-accelerometers in figures E.32, E.33, E.34, and E.35 for the four boundary strategies considered.

5.4 Tuning validation results

The tuning results validation can be done by subjecting the twin-accelerometers to an acceleration signal and choosing the voltage combinations according to the requirement on the maximum distortion level – or on the minimum sensitivity, in the case of the multi-objective tuning results. This choice of the sensitivity, however, does not seem to be an easy definition, since its absolute value varies considerably as a function of the accelerometers parameters. The distortion level, on the other hand, has a more direct quantification, since it is normalized, according to the proposal of Section 4.1.

Hence, the maximum distortion level is the requirement chosen to guide the voltages selection. Besides, the choice on the maximum distortion allows the validation of the mono- and multi-objective tunings altogether.

The results from the left-hand column of Figure 5.11 show the spectra without tuning, with $V_1 = V_2 = 7.6$ V, while on the other two columns are the spectra raised using the tuned voltages for the same non-ideal twin-accelerometers used in the previous sections. The results in the middle column refer to the choice of the excitation voltages that ensure maximum sensitivity with distortion level less than 0.8, i.e., in the trade-off between sensitivity and distortion, the first one is preferred. On the other hand, the results on the right-hand column refer to the choice of the maximum sensitivity with distortion level less than 0.2; in this case, the preference is the distortion. The external acceleration signal used is the same of Section 5.1.

According to the previous sections, both results of the multi-objective approaches are capable to approximate the Pareto-front; besides, the mono-objective tuning can map into a single point of the Pareto-set each time a different maximum distortion is chosen. This fact is evidenced by Figure 5.11, since all the three tunings are able to adjust the raised spectrum so the distortion-sensitivity requirements are reached. Comparing figures 5.11(a) and 5.11(b), it is possible to visually note, for the three tunings, only a small reduction of the distortion in the final tuned spectra. However, in 5.11(b), the spectra presents adequate amplitude, since the

system overall gain is also adjusted during the proposed tuning. In 5.11(c) it is also possible to see the correct spectrum amplitudes, but the appearance of the final spectrum is more clean – since the maximum distortion level is lower –, presenting only the expected components. The analogous results for brick-wall, random reinitialization, and resampling boundary strategies are shown in figures E.36, E.37, and E.38, respectively. Like the results in Figure 5.11, the results of all the three tunings compare well for the other boundary strategies.

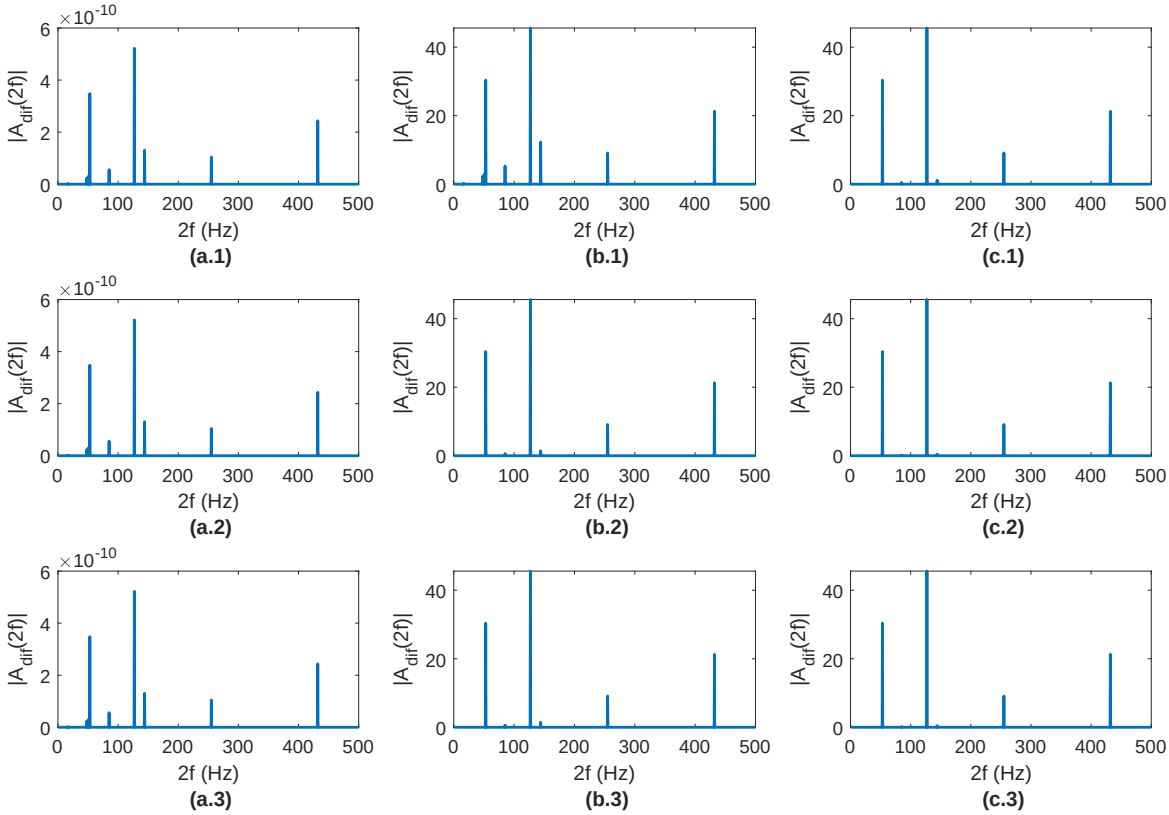

Figure 5.11: Tuning validation results for a non-ideal twin-structure – bounce-back boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2. (1) Mono-objective tuning. (2) Multi-objective tuning, approach 1. (3) Multi-objective tuning, approach 2.

For the ten twin-structures tuned previously, the validation results are shown in Appendix E. For the mono-objective tuning, considering bounce-back, brick-wall, random reinitialization and resampling, the results are presented in figures E.39 to E.42. For the multi-objective tuning, the results considering the same boundary strategies are presented in figures E.43 to E.46 for the approach 1, and in figures E.47 to E.50 for the approach 2.

In these figures, it is interesting to note the differences between each twin-structure. The twin-accelerometers 5 and 9, for example, present completely polluted spectra before the tuning, indicating that the voltages initially used are superior than the critical value to positive feedback stability – a fact evidenced by voltages limits in the distortion and sensitivity mapped previously, given in Figure E.1. The other structures present, before the tuning, diverse spectra appearance, with more or less undesired spectral components depending on the twin-

accelerometers parameters. Considering the tuned spectra to maximum distortion 0.8, in general, it is possible to note that the spectra present the adequate amplitudes for the expected frequencies, besides some spurious components that may present amplitudes similar to the desired components. Those undesired components, however, are mostly eliminated for the twin-structures when the maximum allowed distortion is 0.2; in these cases, the spectra amplitudes are also corrected and correspond to the external acceleration real amplitudes. This general behavior can be found even for the structures with before-tuning completely polluted spectrum.

However, some special cases can be found in all the three tuning methodologies for all the boundary strategies. These cases correspond to spectra that, even after the tuning, present components over all the frequencies analyzed, and, therefore, there is no possibility of distinction between the desired frequency components. These cases are found both to the maximum distortion levels 0.8 and 0.2, and to structures with before-tuning spectrum completely polluted, but also to structures with clean before-tuning spectrum. This behavior associates with the unfeasible solutions found in some of the tunings – as previously discussed – due to the differences in accelerometer parameters and the corresponding maximum voltages to the system stability. The dominated solutions found with the tunings, on the other hand, do not visually influence the tuned spectra.

The dominated and unfeasible solutions found during the tunings – where these last result in the polluted tuned spectra special cases – do not invalidate the tunings for some reasons. First, the tunings are non-deterministic, which means that the unfeasible and dominated solutions are randomly found and, therefore, there is also the possibility of finding non-dominated solutions for twin-structures with any tuning methodology. Besides, the twin-accelerometers simulated have their parameters varied 20% around the respective rated values, which is a tolerance realistic for prototypes, but a high value for devices produced in industrial scale. This means that, with lower tolerance, the differences between the DE variables boundaries and the maximum voltages to positive feedback stability are smaller, reducing the unfeasible and dominated problems, with a better approximation of the Pareto-optimal. Still, if the problems persist in the twin-accelerometers tunings, it is possible to adjust the DE variables boundaries in order to reduce – or amplify – the algorithm borders to match the maximum voltages to system stability.

Chapter 6

Conclusions and Future Steps

The simulations performed for the real-time spectrum analyzer showed the spectrum distortion when accelerometers with different parameters are used in the differential configuration. To overcome these problems, the proposed strategy for the spectral analyzers tuning based on DE proved itself to be adequate.

The mono-objective tuning strategy presented good results considering the comparison with the exhaustive search for a known accelerometer twin-structure, with optimal results obtained via the DE algorithm close to the results via the objective space mapping. These results were also validated to different non-ideal twin-accelerometers when the voltages obtained by the DE were applied to the twin-structures and an acceleration signal distinct to the one used for the tuning was used. It was evidenced that the value chosen for the maximum distortion level highly influenced the resulting mechanical vibration spectrum.

The proposed multi-objective tuning was able to approximate the Pareto-optimal set. With this variables set, it was possible to choose between different Pareto-optimal solutions that increased the importance of one cost function over another. Thus, in applications where there is no previous information about the mechanical vibration spectrum to be mapped, several different combinations may be tested before choosing the most appropriated. The main advantage of this strategy consists in eliminating noise even in extreme cases, provided that proper compromise relation between distortion and sensitivity is chosen.

In both mono- and multi-objective optimization, although the voltage adjustment presented considerably high complexity – due to the quadratic factor of the actuation voltage and its presence in the feedback of the accelerometer model – the convergence time was considerably small, since just a small part of the spectrum was used. The main drawback is that the spectrum needs to be mapped each time an optimization vector not tested before is considered, increasing the total time, that does not correspond only to the algorithm convergence time. Considering, however, that the tuning strategies are proposed for initial calibration of twin-accelerometers in a controlled environment, the total processing time is not a limiting factor.

Considering this fact and comparing the mono- and multi-objective average convergence time and number of spectrum evaluations, since both of them are capable of obtaining the

appropriated voltage combination that results in the required trade-off between distortion level and sensitivity, the multi-objective scenario is advantageous. This is due to the fact that this approach is capable of obtaining several Pareto-optimal results within the same tuning and, therefore, is more flexible. Even when the behavior of the spectrum to be mapped is known, it seems to be a good option to apply the multi-objective tuning; so, if for any reason the application suffers a variation, there are already other possible solutions available to adjust the characteristics of the mapped spectrum. Hence, the multi-objective tuning can be understood as an application-free approach: provided the Pareto-optimal points, any requirements for mechanical vibration spectrum can be obtained.

Besides, the mono-objective tuning presents another disadvantage: since the accelerometers parameters are unknown and influence the shape of the objective space, it is not possible to know *a priori* if the feasible region is large or small – or even if there is a feasible region that satisfies the distortion level chosen, given the trend to restrict the feasible region when lower distortions levels are chosen.

To future works, an optimization method that ensures a good approximation of the Pareto-set and the Pareto-front can be explored. However, the most important future investigation to be performed is the application of the proposed tuning methodology to real physical microelectromechanical spectrum analyzers. Although the simulation of the spectrum analyzer system was performed to mimic the real physical system – considering the variation of the system parameter among the values of the rated parameters of a physical microaccelerometer, as well as a quasi-static model that considers the interaction between the mechanical and electrical domains –, only a physical test with the device can really ensure the reliability of the proposed tuning.

References

1 RAO, S. **Mechanical Vibrations in SI Units**. [S.l.]: Pearson HigherEducation, 2017. 1152 pp. ISBN 9781292178615.

2 KAVITHA, S.; DANIEL, R. J.; SUMANGALA, K. Design and Analysis of MEMS Comb Drive Capacitive Accelerometer for SHM and Seismic Applications. **Measurement**, Elsevier BV, v. 93, p. 327–339, Nov. 2016. ISSN 0263-2241. DOI: 10.1016/j.measurement.2016.07.029.

3 _____. High performance MEMS accelerometers for concrete SHM applications and comparison with COTS accelerometers. **Mechanical Systems and Signal Processing**, Elsevier BV, v. 66-67, p. 410–424, Jan. 2016. ISSN 0888-3270. DOI: 10.1016/j.ymssp.2015.06.005.

4 CEMPEL, C. Multidimensional condition monitoring of mechanical systems in operation. **Mechanical Systems and Signal Processing**, Elsevier BV, v. 17, n. 6, p. 1291–1303, Nov. 2003. ISSN 0888-3270. DOI: 10.1006/mssp.2002.1573.

5 SRINIVAS, K. N.; ARUMUGAM, R. Static and Dynamic Vibration Analyses of Switched Reluctance Motors Including Bearings, Housing, Rotor Dynamics, and Applied Loads. **IEEE Transactions on Magnetics**, Institute of Electrical and Electronics Engineers (IEEE), v. 40, n. 4, p. 1911–1919, July 2004. ISSN 0018-9464. DOI: 10.1109/tmag.2004.828034.

6 CASTELLANOS, J. C.; FRUETT, F. Embedded system to evaluate the passenger comfort in public transportation based on dynamical vehicle behavior with user's feedback. **Measurement**, Elsevier BV, v. 47, p. 442–451, Jan. 2014. ISSN 0263-2241. DOI: 10.1016/j.measurement.2013.08.068.

7 DOS SANTOS PEDOTTI, L. A.; ZAGO, R. M.; FRUETT, F. Fault diagnostics in rotary machines through spectral vibration analysis using low-cost MEMS devices. **IEEE Instrumentation & Measurement Magazine**, Institute of Electrical and Electronics Engineers (IEEE), v. 20, n. 6, p. 39–44, Dec. 2017. ISSN 1094-6969. DOI: 10.1109/mim.2017.8121950.

8 BETTA, G.; LIGUORI, C.; PIETROSANTO, A. A multi-application FFT analyzer based on a DSP architecture. **IEEE Transactions on Instrumentation and Measurement**, Institute of Electrical and Electronics Engineers (IEEE), v. 50, n. 3, p. 825–832, June 2001. ISSN 0018-9456. DOI: 10.1109/19.930461.

9 BETTA, G.; LIGUORI, C.; PAOLILLO, A.; PIETROSANTO, A. A DSP-based FFT-analyzer for the fault diagnosis of rotating machine based on vibration analysis. **IEEE Transactions on Instrumentation and Measurement**, Institute of Electrical and Electronics Engineers (IEEE), v. 51, n. 6, p. 1316–1322, Dec. 2002. ISSN 0018-9456. DOI: 10.1109/tim.2002.807987.

10 CONTRERAS-MEDINA, L. M.; ROMERO-TRONCOSO, R. J.; MILLAN-ALMARAZ, J. R.; RODRIGUEZ-DONATE, C. FPGA based multiple-channel vibration analyzer embedded system for industrial applications in automatic failure detection. In: 2008 International Symposium on Industrial Embedded Systems. [S.l.]: IEEE, June 2008. P. 229–232. DOI: 10.1109/sies.2008.4577705.

11 SHAHZAD, K.; OELMANN, B. An FPGA-based high-performance wireless vibration analyzer. In: 2013 NORCHIP. [S.l.]: IEEE, Nov. 2013. P. 1–5. DOI: 10.1109/norchip.2013.6702038.

12 ADNANI, A. A.; DUPLICY, J.; PHILIPS, L. Spectrum analyzers today and tomorrow: part 1 towards filterbanks-enabled real-time spectrum analysis. **IEEE Instrumentation & Measurement Magazine**, Institute of Electrical and Electronics Engineers (IEEE), v. 16, n. 5, p. 6–11, Oct. 2013. ISSN 1941-0123. DOI: 10.1109/mim.2013.6616284.

13 BIEFELD, V.; CLASBRUMMEL, B.; BINDER, J. Implantable low-g accelerometer for the telemetric monitoring of micro-movements in fracture zones. In: 1ST Annual International IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology. Proceedings (Cat. No.00EX451). [S.l.]: IEEE, Oct. 2000. P. 497–501. DOI: 10.1109/mmb.2000.893834.

14 PUERS, R.; CATRYSSE, M.; VANDEVOORDE, G.; COLLIER, R.; LOURIDAS, E.; BURNY, F.; DONKERWOLCKE, M.; MOULART, F. A telemetry system for the detection of hip prosthesis loosening by vibration analysis. **Sensors and Actuators A: Physical**, Elsevier BV, v. 85, n. 1, p. 42–47, Aug. 2000. ISSN 0924-4247. DOI: 10.1016/s0924-4247(00)00320-4.

15 BAO, M.-H. Introduction to micro mechanical transducers. In: _____. **Micro Mechanical Transducers - Pressure Sensors, Accelerometers and Gyroscopes**. Ed. by Min-Hang Bao. [S.l.]: Elsevier, 2000. v. 8. (Handbook of Sensors and Actuators). P. 1–21. DOI: 10.1016/S1386-2766(00)80015-8.

16 SCHEIBNER, D.; MEHNER, J.; REUTER, D.; GESSNER, T.; DÖTZEL, W. A spectral vibration detection system based on tunable micromechanical resonators. **Sensors and Actuators A: Physical**, Elsevier BV, v. 123, p. 63–72, Sept. 2005. DOI: 10.1016/j.sna.2005.03.034.

17 PEINER, E.; MIKUTA, R.; IWERT, T.; FRITSCH, H.; HAUPTMANN, P.; FRICKE, K.; SCHLACHETZKI, A. Micromachined resonator for cavitation sensing. **Sensors and Actuators A: Physical**, Elsevier BV, v. 76, n. 1, p. 266–272, Aug. 1999. ISSN 0924-4247. DOI: 10.1016/s0924-4247(99)00048-5.

18 FRITSCH, H.; LUCKLUM, R.; IWERT, T.; HAUPTMANN, P.; SCHOLZ, D.; PEINER, E.; SCHLACHETZKI, A. A low-frequency micromechanical resonant vibration sensor for wear monitoring. **Sensors and Actuators A: Physical**, Elsevier BV, v. 62, n. 1, p. 616–620, July 1997. ISSN 0924-4247. DOI: 10.1016/s0924-4247(97)01609-9.

19 ROCHA, L. A.; CRETU, E.; WOLFFENBUTTEL, R. F. MEMS-based mechanical spectrum analyzer. **IEEE Transactions on Instrumentation and Measurement**, Institute of Electrical and Electronics Engineers (IEEE), v. 54, n. 3, p. 1260–1265, June 2005. ISSN 0018-9456. DOI: 10.1109/tim.2005.847168.

20 BANG LEE, K.; CHO, Y.-H. A triangular electrostatic comb array for micromechanical resonant frequency tuning. **Sensors and Actuators A: Physical**, Elsevier BV, v. 70, n. 1-2, p. 112–117, Oct. 1998. DOI: 10.1016/s0924-4247(98)00122-8.

21 CRETU, E.; BARTEK, M.; WOLFFENBUTTEL, R. Spectral analysis through electromechanical coupling. **Sensors and Actuators A: Physical**, Elsevier BV, v. 85, n. 1-3, p. 23–32, Aug. 2000. DOI: 10.1016/S0924-4247(00)00414-3.

22 CRETU, E.; ROCHA, L. A.; WOLFFENBUTTEL, R. F. Electro-mechanical feedback for realization of a mechanical spectrum analyzer. In: TRANSDUCERS '03. 12th International Conference on Solid-State Sensors, Actuators and Microsystems. Digest of Technical Papers (Cat. No.03TH8664). [S.l.]: IEEE, June 2003. v. 2, 1407–1410 vol.2. DOI: 10.1109/SENSOR.2003.1217038.

23 LUENBERGER, D. G.; YE, Y. **Linear and Nonlinear Programming**. [S.l.]: Springer International Publishing, 2015. 560 pp. ISBN 3319188410.

24 BOUSSAÏD, I.; LEPAGNOT, J.; SIARRY, P. A survey on optimization metaheuristics. **Information Sciences**, Elsevier BV, v. 237, p. 82–117, July 2013. Prediction, Control and Diagnosis using Advanced Neural Computations. ISSN 0020-0255. DOI: 10.1016/j.ins.2013.02.041.

25 MIETTINEN, K. **Nonlinear Multiobjective Optimization**. [S.l.]: Springer US, 1998. 324 pp. (International Series in Operations Research & Management Science). ISBN 0792382781.

26 BOYD, S.; VANDENBERGHE, L. **Convex Optimization**. New York, NY, USA: Cambridge University Press, 2004. 732 pp. ISBN 0521833787.

27 ARORA, J. **Introduction to Optimum Design**. London, UK: Elsevier Science, 2016. ISBN 9780128009185.

28 PRICE, K.; STORN, R.; LAMPINEN, J. **Differential Evolution: A Practical Approach to Global Optimization**. [S.l.]: Springer, 2005. (Natural Computing Series). ISBN 3540209506.

29 BURKE, E. K.; KENDALL, G. **Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques**. 2nd ed. [S.l.]: Springer Publishing Company, Incorporated, 2013. ISBN 9781461469391.

30 DEB, K. D. **Multiobjective Optimization Using Evolutionary Algorithms. Wiley, New York**. [S.l.]: John Wiley & Sons, 2001. 536 pp. ISBN 978-0-471-87339-6.

31 YU, X.; GEN, M. **Introduction to Evolutionary Algorithms**. [S.l.]: Springer London, 2010. (Decision Engineering). ISBN 9781849961295.

32 ABDEL-BASSET, M.; ABDEL-FATAH, L.; SANGAIAH, A. K. Chapter 10 - Metaheuristic Algorithms: A Comprehensive Review. In: SANGAIAH, A. K.; SHENG, M.; ZHANG, Z. (Eds.). **Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications**. [S.l.]: Elsevier, 2018. (Intelligent Data-Centric Systems). P. 185–231. ISBN 978-0-12-813314-9. DOI: 10.1016/b978-0-12-813314-9.00010-4.

33 XU, J.; ZHANG, J. Exploration-exploitation tradeoffs in metaheuristics: Survey and analysis. In: PROCEEDINGS of the 33rd Chinese Control Conference. [S.l.]: IEEE, July 2014. P. 8633–8638. DOI: 10.1109/chicc.2014.6896450.

34 GENDREAU, M.; POTVIN, J.-Y. **Handbook of Metaheuristics**. 2nd ed. [S.l.]: Springer-Verlag New York Inc., 30 Sept. 2010. 648 pp. ISBN 1441916636.

35 FEOKTISTOV, V. **Differential Evolution: In Search of Solutions (Springer Optimization and Its Applications)**. Berlin, Heidelberg: Springer US, 17 Oct. 2006. 208 pp. ISBN 0387368957.

36 STORN, R.; PRICE, K. Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. **Journal of Global Optimization**, Springer Science and Business Media LLC, v. 11, n. 4, p. 341–359, Jan. 1997. DOI: 10.1023/A:1008202821328.

37 ARABAS, J.; SZCZEPANKIEWICZ, A.; WRONIAK, T. Experimental Comparison of Methods to Handle Boundary Constraints in Differential Evolution. In: _____. **Parallel Problem Solving from Nature, PPSN XI**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. P. 411–420. ISBN 9783642158711.

38 WAZIR, H.; JAN, M. A.; MASHWANI, W.; SHAH, T. A Penalty Function Based Differential Evolution Algorithm for Constrained Optimization. **Nucleus**, v. 53, p. 155–162, June 2016. ISSN 2306-6539.

39 COELLO, C. A. C. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. **Computer Methods in Applied Mechanics and Engineering**, Elsevier BV, v. 191, n. 11, p. 1245–1287, Jan. 2002. ISSN 0045-7825. DOI: 10.1016/s0045-7825(01)00323-1.

40 ZIELINSKI, K.; PETERS-DROLSHAGEN, D.; LAUR, R. Stopping criteria for single-objective optimization, Jan. 2005.

41 ZIELINSKI, K.; LAUR, R. Stopping Criteria for Differential Evolution in Constrained Single-Objective Optimization. In: **Advances in Differential Evolution**. Ed. by Uday K. Chakraborty. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. P. 111–138. ISBN 978-3-540-68830-3. DOI: 10.1007/978-3-540-68830-3_4.

42 LAMPINEN, J.; STORN, R. Differential Evolution. In: NEW Optimization Techniques in Engineering. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. P. 123–166. ISBN 978-3-540-39930-8. DOI: 10.1007/978-3-540-39930-8_6.

43 STORN, R. On the usage of differential evolution for function optimization. In: PROCEEDINGS of North American Fuzzy Information Processing. [S.l.]: IEEE, July 1996. P. 519–523. ISBN 0-7803-3225-3. DOI: 10.1109/nafips.1996.534789.

44 KERR, A.; MULLEN, K. A comparison of genetic algorithms and simulated annealing in maximizing the thermal conductance of harmonic lattices. **Computational Materials Science**, Elsevier BV, v. 157, p. 31–36, Feb. 2019. ISSN 0927-0256. DOI: 10.1016/j.commatsci.2018.10.007.

45 ZIELINSKI, K.; WEITKEMPER, P.; LAUR, R.; KAMMEYER, K. .-. Parameter Study for Differential Evolution Using a Power Allocation Problem Including Interference Cancellation. In: 2006 IEEE International Conference on Evolutionary Computation. [S.l.]: IEEE, July 2006. P. 1857–1864. DOI: 10.1109/cec.2006.1688533.

46 MEZURA-MONTES, E.; REYES-SIERRA, M.; COELLO, C. A. C. Multi-objective Optimization Using Differential Evolution: A Survey of the State-of-the-Art. In: **Advances in Differential Evolution**. Ed. by Uday K. Chakraborty. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. P. 173–196. ISBN 978-3-540-68830-3. DOI: 10.1007/978-3-540-68830-3_7.

47 NEBRO, A.; ZAVALA, G.; LUNA, F.; COELLO, C. A survey of multi-objective metaheuristics applied to structural optimization. **Structural and Multidisciplinary Optimization**, Springer Science and Business Media LLC, v. 0, n. 4, p. 0, Apr. 2014. DOI: 10.1007/s00158-013-0996-4.

48 BABU, B. V.; JEHAN, M. M. L. Differential evolution for multi-objective optimization. In: THE 2003 Congress on Evolutionary Computation, 2003. CEC '03. [S.l.]: IEEE, Dec. 2003. v. 4, 2696–2703 vol.4. DOI: 10.1109/cec.2003.1299429.

49 ROBIČ, T.; FILIPIČ, B. DEMO: Differential Evolution for Multiobjective Optimization. In: _____. **Evolutionary Multi-Criterion Optimization**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. P. 520–533. ISBN 978-3-540-31880-4.

50 OPPENHEIM, A.; WILLSKY, A.; NAWAB, S. **Signals and Systems**. Upper Saddle River, N.J: Prentice Hall, 1997. (Prentice-Hall signal processing series). ISBN 9780138147570.

51 AMBARDAR, A. **Analog and Digital Signal Processing**. Pacific Grove, CA: Brooks Cole Pub. Co, 1999. ISBN 9780534954093.

52 ADNANI, A. A.; DUPLICY, J.; PHILIPS, L. Spectrum analyzers today and tomorrow: Part 2. **IEEE Instrumentation Measurement Magazine**, Institute of Electrical and Electronics Engineers (IEEE), v. 16, n. 6, p. 36–40, Dec. 2013. ISSN 1941-0123. DOI: 10.1109/MIM.2013.6704970.

53 IWANIEC, M.; HOLOVATYY, A.; TESLYUK, V.; LOBUR, M.; KOLESNYK, K.; MASHEVSKA, M. Development of vibration spectrum analyzer using the Raspberry Pi microcomputer and 3-axis digital MEMS accelerometer ADXL345. In: 2017 XIIIth International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH). [S.l.]: IEEE, Apr. 2017. P. 25–29. DOI: 10.1109/memstech.2017.7937525.

54 IGLESIAS, V.; GRAJAL, J.; SANCHEZ, M.-A.; LOPEZ-VALLEJO, M. Implementation of a Real-Time Spectrum Analyzer on FPGA Platforms. **IEEE Transactions on Instrumentation and Measurement**, Institute of Electrical and Electronics Engineers (IEEE), v. 64, n. 2, p. 338–355, Feb. 2015. DOI: 10.1109/tim.2014.2344411.

55 SENTURIA, S. D. **Microsystem Design**. Norwell, MA, USA: Springer US, 8 Dec. 2004. 720 pp. ISBN 0-7923-7246-8.

56 YOUNIS, M. **MEMS Linear and Nonlinear Statics and Dynamics**. [S.l.]: Springer US, 2011. (Microsystems). ISBN 1441960201.

57 MAHMOOD, M. S.; CELIK-BUTLER, Z.; BUTLER, D. P. Design, fabrication and characterization of flexible MEMS accelerometer using multi-Level UV-LIGA. **Sensors and Actuators A: Physical**, Elsevier BV, v. 263, p. 530–541, Aug. 2017. ISSN 0924-4247. DOI: https://doi.org/10.1016/j.sna.2017.07.007.

58 BEEBY, S. P.; TUDOR, M. J.; WHITE, N. M. Energy harvesting vibration sources for microsystems applications. **Measurement Science and Technology**, IOP Publishing, v. 17, n. 12, r175–r195, Oct. 2006. DOI: 10.1088/0957-0233/17/12/r01.

59 SANYAL, K.; BISWAS, K. Structural design and optimization of MEMS based capacitive accelerometer. In: 2017 Devices for Integrated Circuit (DevIC). [S.l.]: IEEE, Mar. 2017. P. 294–298. DOI: 10.1109/devic.2017.8073955.

60 TEZ, S.; TORUNBALCI, M. M.; AKIN, T. A novel method for fabricating MEMS three-axis accelerometers using low temperature Au-Sn eutectic bonding. In: 2016 IEEE SENSORS. [S.l.]: IEEE, Oct. 2016. P. 1–3. DOI: 10.1109/icsens.2016.7808705.

61 ZHOU, X.; CHE, L.; SHENGLIN, L.; LIN, Y.; LI, X.; WANG, Y. Design and fabrication of a MEMS capacitive accelerometer with fully symmetrical double-sided H-shaped beam structure. **Microelectronic Engineering**, Elsevier BV, v. 131, p. 51–57, Jan. 2015. DOI: 10.1016/j.mee.2014.10.005.

62 KOBAYASHI, T.; OKADA, H.; MASUDA, T.; MAEDA, R.; ITOH, T. A digital output accelerometer using MEMS-based piezoelectric accelerometers and arrayed CMOS inverters with satellite capacitors. **Smart Materials and Structures**, IOP Publishing, v. 20, n. 6, p. 065017, May 2011. DOI: 10.1088/0964-1726/20/6/065017.

63 LI-PENG WANG; WOLF, R. A.; YU WANG; DENG, K. K.; ZOU, L.; DAVIS, R. J.; TROLIER-MCKINSTRY, S. Design, fabrication, and measurement of high-sensitivity piezoelectric microelectromechanical systems accelerometers. **Journal of Microelectromechanical Systems**, Institute of Electrical and Electronics Engineers (IEEE), v. 12, n. 4, p. 433–439, Aug. 2003. ISSN 1057-7157. DOI: 10.1109/jmems.2003.811749.

64 ROY, A.; BHATTACHARYYA, T. Design, fabrication and characterization of high performance SOI MEMS piezoresistive accelerometers. **Microsystem Technologies**, Springer Science and Business Media LLC, v. 21, n. 1, p. 55–63, Jan. 2015. DOI: 10.1007/s00542-013-1904-y.

65 SHEIKHALEH, A.; ABEDI, K.; JAFARI, K. A Proposal for an Optical MEMS Accelerometer Relied on Wavelength Modulation With One Dimensional Photonic Crystal. **Journal of Lightwave Technology**, Institute of Electrical and Electronics Engineers (IEEE), v. 34, n. 22, p. 5244–5249, Nov. 2016. ISSN 0733-8724. DOI: 10.1109/jlt.2016.2597539.

66 CUI, T.; WANG, J. Polymer-based wide-bandwidth and high-sensitivity micromachined electron tunneling accelerometers using hot embossing. **Journal of Microelectromechanical Systems**, Institute of Electrical and Electronics Engineers (IEEE), v. 14, n. 5, p. 895–902, Oct. 2005. ISSN 1057-7157. DOI: 10.1109/jmems.2005.851865.

67 LIU, C.-H.; KENNY, T. W. A high-precision, wide-bandwidth micromachined tunneling accelerometer. **Journal of Microelectromechanical Systems**, Institute of Electrical and Electronics Engineers (IEEE), v. 10, n. 3, p. 425–433, Sept. 2001. ISSN 1057-7157. DOI: 10.1109/84.946800.

68 VOPILKIN, E. A.; KLIMOV, A. Y.; ROGOV, V. V.; PRYAKHIN, D. A.; GUSEV, S. A.; SKOROHODOV, E. V.; SHULESHOVA, I. Y.; SHASHKIN, V. I. MEMS Tunneling Sensor Without the Feedback Loop. **IEEE Sensors Journal**, Institute of Electrical and Electronics Engineers (IEEE), v. 14, n. 6, p. 1831–1835, June 2014. ISSN 1530-437X. DOI: 10.1109/jsen.2014.2305307.

69 MAILLY, F.; MARTINEZ, A.; GIANI, A.; PASCAL-DELANNOY, F.; BOYER, A. Design of a micromachined thermal accelerometer: Thermal simulation and experimental results. **Microelectronics Journal**, Elsevier BV, v. 34, n. 4, p. 275–280, Apr. 2003. DOI: 10.1016/s0026-2692(02)00194-5.

70 MEZGHANI, B.; TOUNSI, F.; REKIK, A.; MAILLY, F.; MASMOUDI, M.; NOUET, P. Sensitivity and power modeling of CMOS-MEMS single axis convective accelerometers. **Microelectronics Journal**, Elsevier BV, v. 44, n. 12, p. 1092–1098, Dec. 2013. DOI: 10.1016/j.mejo.2013.06.006.

71 TAVAKKOLI, H.; MOMEN, H. G.; SANI, E. A.; YAZGI, M. An inductive MEMS accelerometer. In: 2017 10th International Conference on Electrical and Electronics Engineering (ELECO). [S.l.: s.n.], Nov. 2017. P. 459–463.

72 MOHAMMED, Z.; DUSHAQ, G.; CHATTERJEE, A.; RASRAS, M. An optimization technique for performance improvement of gap-changeable MEMS accelerometers. **Mechatronics**, Elsevier BV, v. 54, p. 203–216, Oct. 2018. ISSN 0957-4158. DOI: 10.1016/j.mechatronics.2017.10.011.

73 KAMPEN, R. van; WOLFFENBUTTEL, R. Modeling the mechanical behavior of bulk-micromachined silicon accelerometers. **Sensors and Actuators A: Physical**, Elsevier BV, v. 64, n. 2, p. 137–150, Jan. 1998. ISSN 0924-4247. DOI: 10.1016/s0924-4247(98)80007-1.

74 XIAO, D.; LI, Q.; HOU, Z.; XIA, D.; XU, X.; WU, X. A double differential torsional micro-accelerometer based on V-shape beam. **Sensors and Actuators A: Physical**, Elsevier BV, v. 258, p. 182–192, May 2017. ISSN 0924-4247. DOI: 10.1016/j.sna.2017.03.011.

75 HALLIDAY, D.; RESNICK, R.; WALKER, J. **Fundamentals of Physics**. [S.l.]: John Wiley & Sons, 2010. ISBN 9780470469118.

76 COLLARD, D.; FUJITA, H.; TOSHIYOSHI, H.; LEGRAND, B.; BUCHAILLOT, L. Electrostatic Micro-actuators. In: BOUSSEY, J. (Ed.). **Microsystems technology: fabrication, test & reliability**. London Sterling, VA: Kogan Page Science, 2003. chap. 4, p. 75–116. ISBN 1903996473.

77 THOMAS, G.; WEIR, M. **Thomas' Calculus**. Boston: Pearson Education, 2015. ISBN 0321878965.

78 CRETU, E.; BARTEK, M.; WOLFFENBUTTEL, R. F. Analytical Modelling for Accelerometers with Electrically Tunable Sensitivity. In: TECHNICAL Proceedings of the 1999 International Conference on Modeling and Simulation of Microsystems. Cambridge, MA: Computational Publications, Apr. 1999. P. 601–604. ISBN 0966613546.

79 ROCHA, L. A.; DIAS, R. A.; CRETU, E.; MOL, L.; WOLFFENBUTTEL, R. F. Autocalibration of capacitive MEMS accelerometers based on pull-in voltage. **Microsystem Technologies**, Springer Science and Business Media LLC, v. 17, n. 3, p. 429–436, Feb. 2011. DOI: 10.1007/s00542-011-1252-8.

80 GOLNARAGHI, M. F.; KUO, B. C. **Automatic Control Systems**. 9 ed. Hoboken, NJ: Wiley, 2010. ISBN 9780470048962.

81 OGATA, K. **Modern Control Engineering**. Boston: Prentice Hall, 2010. (Instrumentation and controls series). ISBN 9780136156734.

82 ANSTEY, N. A. Correlation techniques - a review. **Geophysical Prospecting**, Wiley, v. 12, n. 4, p. 355–382, Dec. 1964. DOI: 10.1111/j.1365-2478.1964.tb01911.x.

83 DERRICK, T. Time Series Analysis: The Cross-Correlation Function. In: [s.l.]: Human Kinetics Publishers, 29 July 2003. P. 189–205. ISBN 0736044671.

84 DONDURUR, D. Chapter 4 - Fundamentals of Data Processing. In: _____. **Acquisition and Processing of Marine Seismic Data**. Ed. by Derman Dondurur. [S.l.]: Elsevier, 13 Mar. 2018. P. 211–239. ISBN 0128114908. DOI: https://doi.org/10.1016/B978-0-12-811490-2.00004-9.

85 MOHAMMED, Z.; ELFADEL, I.; RASRAS, M. Monolithic Multi Degree of Freedom (MDoF) Capacitive MEMS Accelerometers. **Micromachines**, MDPI AG, v. 9, n. 11, p. 602, Nov. 2018. DOI: 10.3390/mi9110602.

86 GUNTHER, L. **The Physics of Music and Color**. 1st ed ed. New York: Springer New York, 2012. DOI: 10.1007/978-1-4614-0557-3.

87 DOUBLE. [S.l.: s.n.]. https://la.mathworks.com/help/matlab/ref/double.html. Accessed: 2020-01-29.

88 LJUNG, L. **System Identification - Theory for the user**. Ed. by Thomas Kailath. 2nd ed. [S.l.]: Prentice Hall, 1999. 658 pp. ISBN 0136566952.

Appendix A – Fourier Analysis

In this appendix, the basis of the Fourier analysis – which enables the evaluation of signals in the frequency domain – is described. Initially, the idea of the Fourier series for periodic signals is explored, and, later, this representation is enlarged to the Fourier transform. The approach used here is mainly based on the book of Oppenheim, Willsky, and Nawab [1].

Fourier Series

A signal is periodic if, for some positive value T_0,

$$x(t) = x(t + T_0)\,, \tag{A.1}$$

for all t.

The fundamental period of $x(t)$, T_0, in seconds, is the minimum positive, nonzero value for which Equation A.1 is satisfied. Therefore, the fundamental frequency ω_0, in rad/s, is given by

$$\omega_0 = \frac{2\pi}{T_0}\,. \tag{A.2}$$

The periodic complex exponential,

$$x\,(t) = e^{j\omega_0 t}\,, \tag{A.3}$$

has a set of harmonically related complex exponentials associated:

$$\phi_n\,(t) = e^{jn\omega_0 t} = e^{jn(2\pi/T_0)t},\ n = 0, \pm 1, \pm 2, \ldots\,. \tag{A.4}$$

Each of these signals has a fundamental frequency $n\omega_0$ – i.e., a multiple of ω_0 – and a fundamental period T_0/n – i.e, a fraction of T_0. Thus, each signal $\phi_n\,(t)$ is also periodic with period T_0. A linear combination of harmonically related complex exponentials of the form

$$x\,(t) = \sum_{n=-\infty}^{+\infty} a_n e^{jn\omega_0 t} = \sum_{n=-\infty}^{+\infty} a_n e^{jn(2\pi/T_0)t} \tag{A.5}$$

is also periodic with period T_0. In Equation A.5, the term for $n = 0$ is constant and the components for $n = \pm N$ are referred to as the N^{th} harmonic components – specifically, the

[1] OPPENHEIM, A.; WILLSKY, A.; NAWAB, S. **Signals and Systems**. [S.l.]: Prentice Hall, 1997. (Prentice-Hall signal processing series). ISBN 9780138147570.

first harmonic components are also called fundamental components. The representation of a periodic signal in the form of Equation A.5 is the signal Fourier Series (FS) representation in the exponential form.

The process to obtain the expressions for the coefficients a_n of the Fourier series begins with the multiplication of both sides of Equation A.5 by $e^{-jm\omega_0 t}$, where $m = 0, \pm 1, \pm 2, \ldots$

$$x(t)\, e^{-jm\omega_0 t} = \sum_{n=-\infty}^{+\infty} a_n e^{jn\omega_0 t} e^{-jm\omega_0 t} \tag{A.6}$$

Integrating both sides from 0 to $T_0 = 2\pi/\omega_0$, results in Equation A.7.

$$\int_0^{T_0} x(t)\, e^{-jm\omega_0 t} dt = \int_0^{T_0} \sum_{n=-\infty}^{+\infty} a_n e^{jn\omega_0 t} e^{-jm\omega_0 t} dt \tag{A.7}$$

Switching the order of integration and summation:

$$\int_0^{T_0} x(t)\, e^{-jm\omega_0 t} dt = \sum_{n=-\infty}^{+\infty} a_n \left[\int_0^{T_0} e^{j(n-m)\omega_0 t} dt \right]. \tag{A.8}$$

Using Euler's formula,

$$e^{j\theta} = \cos\theta + j \sin\theta\,, \tag{A.9}$$

the integral between the brackets can be written as:

$$\int_0^{T_0} e^{j(n-m)\omega_0 t} dt = \int_0^{T_0} \cos(n-m)\,\omega_0 t\, dt + j \int_0^{T_0} \sin(n-m)\,\omega_0 t\, dt\,. \tag{A.10}$$

For $n \neq m$, $\cos(n-m)\omega_0 t$ and $\sin(n-m)\omega_0 t$ are periodic sinusoids with fundamental period $(T_0/\,|n-m|)$. Thus, in Equation A.10, the integration is performed over an interval that is an integral number of periods of the signals $\cos(n-m)\omega_0 t$ and $\sin(n-m)\omega_0 t$, and therefore the integrals of these signals are zero.

Considering now $n = m$, the left-hand side of Equation A.10 can be used:

$$\begin{aligned} \int_0^{T_0} e^{j(n-m)\omega_0 t} dt &= \int_0^{T_0} e^0 dt \\ &= \int_0^{T_0} dt \\ &= T_0\,. \end{aligned} \tag{A.11}$$

In summary,

$$\int_0^{T_0} e^{j(n-m)\omega_0 t} dt = \begin{cases} T_0\,, & \text{if } n = m \\ 0\,, & \text{if } n \neq m\,. \end{cases} \tag{A.12}$$

Consequently, when $n = m$, Equation A.8 is reduced to:

$$\int_0^{T_0} x(t)\, e^{-jm\omega_0 t} dt = \sum_{n=-\infty}^{+\infty} a_n T_0 \tag{A.13}$$

and the coefficient a_n is given by

$$a_n = \frac{1}{T_0} \int_0^{T_0} x(t)\, e^{-jn\omega_0 t} dt\,. \tag{A.14}$$

Since during the evaluation of Equation A.10, the integration occurs over an interval of length T_0, the same result is valid if different limits to the integral are chosen, as long as the interval length T_0 is maintained. Then, denoting integration over any interval length T_0 by $\int_{T_0}$, equations A.12 and A.14 can be rewritten as equations A.15 and A.16, respectively.

$$\int_{T_0} e^{j(n-m)\omega_0 t} dt = \begin{cases} T_0\,, & \text{if } n = m \\ 0\,, & \text{if } n \neq m \end{cases} \tag{A.15}$$

$$a_n = \frac{1}{T_0} \int_{T_0} x(t)\, e^{-jn\omega_0 t} dt \tag{A.16}$$

Thereby, the Fourier series of a periodic continuous-time signal is given by Definition 6.1.

Definition 6.1: Fourier Series of a periodic continuous-time signal

If $x(t)$ has a Fourier series representation – i.e., if it can be expressed as a linear combination of harmonically related complex exponentials in the form of Equation A.17 –, then the coefficients are given by Equation A.18.

$$x(t) = \sum_{n=-\infty}^{+\infty} a_n e^{jn\omega_0 t} = \sum_{n=-\infty}^{+\infty} a_n e^{jn(2\pi/T_0)t} \tag{A.17}$$

$$a_n = \frac{1}{T_0} \int_{T_0} x(t)\, e^{-jn\omega_0 t} dt = \frac{1}{T_0} \int_{T_0} x(t)\, e^{-jn(2\pi/T_0)t} dt \tag{A.18}$$

Equation A.17 is called the synthesis equation, while Equation A.18 is the analysis equation, which defines the set of coefficients a_n – called the Fourier series coefficients or spectral coefficients of $x(t)$. These coefficients measure the portion of the signal $x(t)$ that is at each harmonic of the fundamental component.

The coefficient a_0 is the DC or constant component of $x(t)$ – given by Equation A.18 with $n = 0$, resulting in Equation A.19 –, and represents the average value of $x(t)$ over one period.

$$a_0 = \frac{1}{T_0} \int_{T_0} x(t)\, dt \tag{A.19}$$

To real periodic signals, it is possible to write the Fourier series alternatively in the polar and trigonometric forms. To obtain the trigonometric representation – used later in this

text –, it is supposed that $x(t)$ is real and can be expressed in the form of Equation A.17. Since $x(t) = x^*(t)$ – where $x^*(t)$ is the complex conjugate of $x(t)$ –, then:

$$x(t) = \sum_{n=-\infty}^{\infty} a_n^* e^{-jn\omega_0 t}. \tag{A.20}$$

Replacing n by $-n$ in the summation:

$$x(t) = \sum_{n=-\infty}^{\infty} a_{-n}^* e^{jn\omega_0 t}, \tag{A.21}$$

which, by comparison with Equation A.17 requires that

$$a_n = a_{-n}^* \tag{A.22}$$

or, equivalently, that

$$a_n^* = a_{-n}. \tag{A.23}$$

Now, rearranging the summation of Equation A.17 as

$$x(t) = a_0 + \sum_{n=1}^{\infty} \left[a_n e^{jn\omega_0 t} + a_{-n} e^{-jn\omega_0 t} \right]. \tag{A.24}$$

Substituting Equation A.23 in Equation A.24:

$$x(t) = a_0 + \sum_{n=1}^{\infty} \left[a_n e^{jn\omega_0 t} + a_n^* e^{-jn\omega_0 t} \right] \tag{A.25}$$

Then, Equation A.25 can be written in function of the real part of $a_n e^{jn\omega_0 t}$, since the terms inside the summation of Equation A.25 are complex conjugates of each other.

$$x(t) = a_0 + \sum_{n=1}^{\infty} 2\Re \left\{ a_n e^{jn\omega_0 t} \right\} \tag{A.26}$$

The trigonometric form is obtained by writing a_n in the rectangular form

$$a_n = B_n + jC_n \tag{A.27}$$

where B_n and C_n are real. Substituting Equation A.27 in Equation A.26 and using Euler's formula, the Fourier series trigonometric form is obtained, as Equation A.28.

$$\begin{aligned} x(t) &= a_0 + \sum_{n=1}^{\infty} 2\Re \left\{ (B_n + jC_n) e^{jn\omega_0 t} \right\} \\ &= a_0 + \sum_{n=1}^{\infty} 2\Re \left\{ (B_n + jC_n) (\cos n\omega_0 t + j \sin n\omega_0 t) \right\} \\ &= a_0 + \sum_{n=1}^{\infty} 2\Re \left\{ (B_n \cos n\omega_0 t - C_n \sin n\omega_0 t) + j (B_n \sin n\omega_0 t + C_n \cos n\omega_0 t) \right\} \\ &= a_0 + \sum_{n=1}^{\infty} 2 (B_n \cos n\omega_0 t - C_n \sin n\omega_0 t) \end{aligned} \tag{A.28}$$

Equation A.28 can also be represented as

$$x(t) = a_0 + \sum_{n=1}^{\infty} \left(b_n \cos n\omega_0 t + c_n \sin n\omega_0 t\right). \tag{A.29}$$

where

$$b_n = 2B_n \tag{A.30}$$

and

$$c_n = -2C_n. \tag{A.31}$$

Now, to obtain the trigonometric Fourier series coefficients b_n and c_n, the analysis equation – Equation A.18 – can be used together with the Euler's formula:

$$\begin{aligned} a_n &= \frac{1}{T_0}\int_{T_0} x(t)\, e^{-jn\omega_0 t} dt \\ &= \frac{1}{T_0}\int_{T_0} x(t)\left(\cos n\omega_0 t - j \sin n\omega_0 t\right) dt \\ &= \frac{1}{T_0}\int_{T_0} x(t) \cos n\omega_0 t\, dt - j\frac{1}{T_0}\int_{T_0} x(t) \sin n\omega_0 t\, dt \end{aligned} \tag{A.32}$$

Comparing Equation A.27 with Equation A.32, the expressions for the coefficients B_n and C_n are obtained, according to equations A.33 and A.34.

$$B_n = \frac{1}{T_0}\int_{T_0} x(t) \cos n\omega_0 t\, dt \tag{A.33}$$

$$C_n = -\frac{1}{T_0}\int_{T_0} x(t) \sin n\omega_0 t\, dt \tag{A.34}$$

Finally, the trigonometric coefficients are given by equations A.35 and A.36.

$$b_n = \frac{2}{T_0}\int_{T_0} x(t) \cos n\omega_0 t\, dt \tag{A.35}$$

$$c_n = \frac{2}{T_0}\int_{T_0} x(t) \sin n\omega_0 t\, dt \tag{A.36}$$

Summarizing, in the trigonometric form, the Fourier series synthesis equation is given by Equation A.29, while the analysis equations are Equation A.19, Equation A.35 and Equation A.36, rewritten as equations A.37 to A.40.

$$x(t) = a_0 + \sum_{n=1}^{\infty} \left(b_n \cos n\omega_0 t + c_n \sin n\omega_0 t\right) \tag{A.37}$$

$$a_0 = \frac{1}{T_0}\int_{T_0} x(t)\, dt \tag{A.38}$$

$$b_n = \frac{2}{T_0}\int_{T_0} x(t) \cos n\omega_0 t\, dt \tag{A.39}$$

$$c_n = \frac{2}{T_0}\int_{T_0} x(t) \sin n\omega_0 t\, dt \tag{A.40}$$

Fourier Transform

The concepts used to represent periodic signals in a Fourier Series – as a linear combinations of complex exponentials – can be extended to non-periodic signals. While for periodic signals the complex exponential are harmonically related, for aperiodic signals the complex exponentials become infinitesimally close in frequency and their linear combination takes the form of an integral.

The development of the Fourier Transform (FT) representation of aperiodic signals is based on the fact that an aperiodic signal can be viewed as a periodic one with infinite period. The attainment of such representation begins with the continuous-time periodic square wave, which is given in Equation A.41 over one period and periodically repeats with period T_0, as depicted in Figure A.1.

$$x(t) = \begin{cases} 1\,, & \text{if } |t| < T_1 \\ 0\,, & \text{if } T_1 < |t| < T_0/2 \end{cases} \tag{A.41}$$

The Fourier series coefficients a_n for this square wave, according to Equation A.18 are:

$$\begin{aligned} a_n &= \frac{1}{T_0}\int_{T_0} x\,(t)\, e^{-jn\omega_0 t} dt \\ &= \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x(t) e^{-jn\omega_0 t} dt \\ &= \frac{1}{T_0}\int_{-T_1}^{T_1} e^{-jn\omega_0 t} dt \\ &= -\frac{1}{jn\omega_0 T_0} e^{-jn\omega_0 t}\Big|_{-T_1}^{T_1} \\ &= \frac{2}{n\omega_0 T_0}\left[\frac{e^{jn\omega_0 T_1} - e^{-jn\omega_0 T_1}}{2j}\right] \\ &= \frac{2\sin\left(n\omega_0 T_1\right)}{n\omega_0 T_0}\,, \end{aligned} \tag{A.42}$$

while the constant component a_0 is

$$\begin{aligned} a_0 &= \frac{1}{T_0}\int_{T_0} x\,(t)\, dt \\ &= \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x(t) dt \\ &= \frac{1}{T_0}\int_{-T_1}^{T_1} dt \\ &= \frac{2T_1}{T_0}\,. \end{aligned} \tag{A.43}$$

Equation A.42 can be used to write an envelope function, as Equation A.44.

$$T_0 a_n = \frac{2\sin\left(\omega T_1\right)}{\omega}\Big|_{\omega = n\omega_0} \tag{A.44}$$

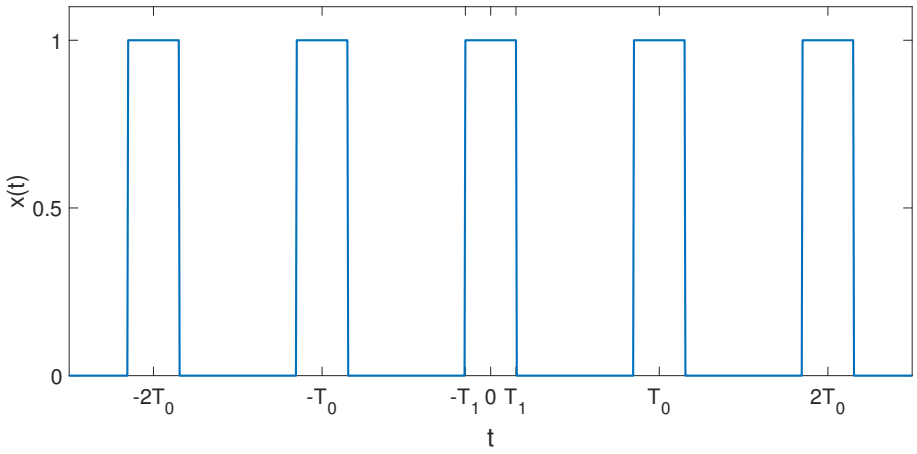

Figure A.1: Continuous-time periodic square wave. Adapted from Oppenheim, Willsky, and Nawab [p. 285].

In Equation A.44, ω is considered as a continuous variable and the coefficients a_n are equally spaced samples of the envelope $T_0 a_n$, and this envelope is independent of T_0. In Figure A.2 the envelope $T_0 a_n$ of the periodic square wave is shown, highlighting that as the fundamental period T_0 increases – i.e., as $\omega_0 = 2\pi/T_0$ decreases –, the envelope is sampled with a closer spacing. As T_0 becomes arbitrarily large, the set of Fourier series coefficients approaches the envelope function as $T_0 \to \infty$. Also, with an arbitrarily large value for T_0, the square wave comes close to a rectangular pulse – an aperiodic signal corresponding to one period of the square wave.

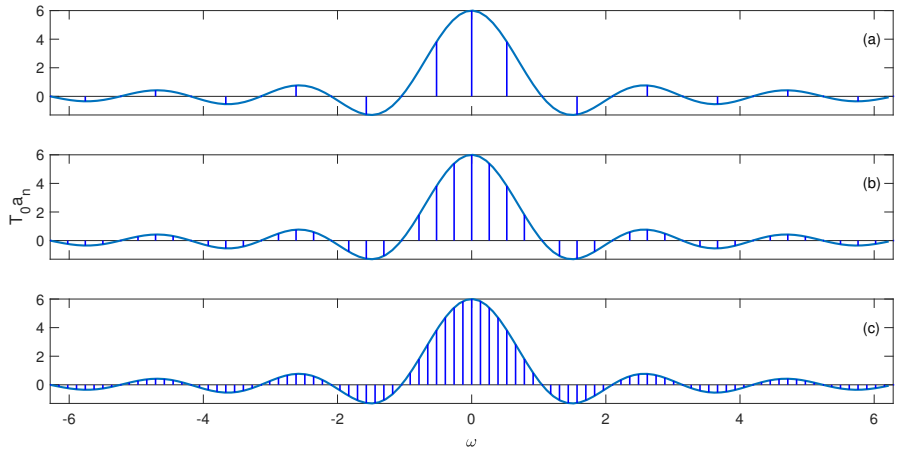

Figure A.2: Envelope of Fourier series coefficients for the periodic square wave for several values of T_0 with T_1 fixed. (a)$T_0 = 4T_1$; (b)$T_0 = 8T_1$; (c)$T_0 = 16T_1$. Adapted from Oppenheim, Willsky, and Nawab [p. 286].

This is the basic idea of the Fourier representation for aperiodic signals: an aperiodic signal is considered as the limit of a periodic signal as the period becomes arbitrarily large, and the limiting behavior of the Fourier series representation for this signal is examined. Particularly, considering an aperiodic signal $x(t)$ with finite duration, i.e., $x(t) = 0$ for $|t| > T_1$, as depicted

in Figure A.3(a). A periodic signal $\tilde{x}(t)$ can be constructed from the aperiodic signal $x(t)$, taking $x(t)$ as one period, as illustrated in Figure A.3(b). As the period T_0 is increased, $\tilde{x}(t)$ is equal to $x(t)$ over a longer interval – as indicated in Figure A.3(c), and as $T_0 \to \infty$, $\tilde{x}(t)$ is equal to $x(t)$ for any finite value of t.

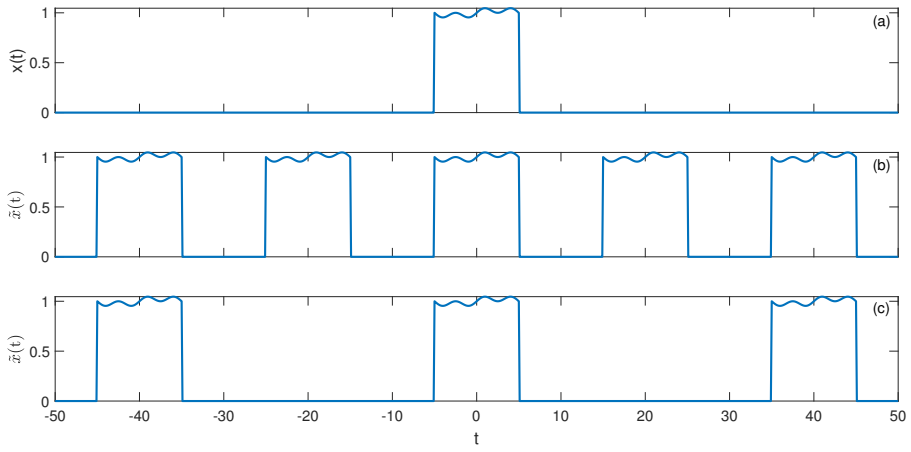

Figure A.3: Aperiodic signal $x(t)$ (a). Periodic signal $\tilde{x}(t)$ constructed to be equal $x(t)$ over one period, with period $T_0 = 4T_1$ (b) and $T_0 = 8T_1$ (c). Adapted from Oppenheim, Willsky, and Nawab [p.287].

This idea reflects on the Fourier series representation of $\tilde{x}(t)$. Back to equations A.17 and A.18, integrating between $-T_0/2$ and $T_0/2$:

$$\tilde{x}\,(t) = \sum_{n=-\infty}^{+\infty} a_n e^{jn\omega_0 t} = \sum_{n=-\infty}^{+\infty} a_n e^{jn(2\pi/T_0)t}\,, \tag{A.45}$$

$$a_n = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} \tilde{x}\,(t)\, e^{-jn\omega_0 t} dt\,, \tag{A.46}$$

where $\omega_0 = 2\pi/T_0$. Since $\tilde{x}(t) = x(t)$ for $|t| < T_0/2$ and $x(t) = 0$ outside this interval, Equation A.46 becomes:

$$\begin{aligned} a_n &= \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} x\,(t)\, e^{-jn\omega_0 t} dt \\ &= \frac{1}{T_0} \int_{-\infty}^{\infty} x\,(t)\, e^{-jn\omega_0 t} dt\,. \end{aligned} \tag{A.47}$$

Defining the envelope $X(j\omega)$ of $T_0 a_n$ as

$$X\,(j\omega) = \int_{-\infty}^{\infty} x\,(t)\, e^{-j\omega t} dt\,, \tag{A.48}$$

the coefficients a_n are:

$$a_n = \frac{1}{T_0} X\left(jn\omega_0\right) . \tag{A.49}$$

Combining Equations A.45 and A.49:

$$\begin{aligned} \tilde{x}\left(t\right) &= \sum_{n=-\infty}^{\infty} \frac{1}{T_0} X\left(jn\omega_0\right) e^{jn\omega_0 t} \\ &= \frac{1}{2\pi} \sum_{n=-\infty}^{\infty} X\left(jn\omega_0\right) e^{jn\omega_0 t} \omega_0 . \end{aligned} \tag{A.50}$$

Graphically, each term of the summation on the right-hand side is the area of a rectangle of height $X\left(jn\omega_0\right) e^{jn\omega_0 t}$ and width ω_0. When $T_0 \to \infty$, $\omega_0 \to 0$, so the right-hand side of Equation A.50 becomes an integral. Besides, as $T_0 \to \infty$, $\hat{x}(t)$ approaches $x(t)$ and, in the limit, Equation A.50 represents $x(t)$. Therefore, in the limit, equations A.50 and A.48 become, respectively, equations A.51 and A.52.

$$x\left(t\right) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X\left(j\omega\right) e^{j\omega t} d\omega \tag{A.51}$$

$$X\left(j\omega\right) = \int_{-\infty}^{\infty} x\left(t\right) e^{-j\omega t} dt \tag{A.52}$$

Equations A.51 and A.52 are called the Fourier transform pair, where Equation A.52 is referred to as the Fourier transform or Fourier integral of $x(t)$, while Equation A.51 is the inverse Fourier transform equation. Equation A.51 is a synthesis equation and, similarly to the synthesis equation of the Fourier series – Equation A.17 –, represents a signal as a linear combination of complex exponentials. These complex exponentials have amplitudes a_n, as given by Equation A.18, and occur at a discrete set of harmonically related frequencies $n\omega_0$, $n = 0, \pm 1, \pm 2, \ldots$ – as previously described. On the other hand, for aperiodic signals, the complex exponentials occur at a continuum of frequencies and have amplitudes

$$X(j\omega)\left(\frac{d\omega}{2\pi}\right) . \tag{A.53}$$

The Fourier transform $X(j\omega)$ is often referred to as the spectrum of $x(t)$ since it provides the information to describe $x(t)$ as a linear combination of sinusoidal signals at different frequencies.

Although the argument used to derive the Fourier transform pair of Equations A.51 and A.52 assumed that $x(t)$ was of arbitrary but finite duration, these equations remain valid for a myriad class of signals of infinite duration, that must satisfy a set of conditions so the Fourier transform converges.

Appendix B
Trigonometric Identities

$$\sin^2 A + \cos^2 A = 1 \tag{B.1}$$

$$\sin(A + B) = \sin A \cos B + \sin B \cos A \tag{B.2}$$

$$\sin(A - B) = \sin A \cos B - \sin B \cos A \tag{B.3}$$

$$\cos(A + B) = \cos A \cos B - \sin A \sin B \tag{B.4}$$

$$\cos(A - B) = \cos A \cos B + \sin A \sin B \tag{B.5}$$

Appendix C – A Brief Table of Integrals

$$\int \alpha dx = \alpha x + C \tag{C.1}$$

$$\int \sin(ax)\, dx = -\frac{1}{a}\cos(ax) + C \tag{C.2}$$

$$\int \cos(ax)\, dx = \frac{1}{a}\sin(ax) + C \tag{C.3}$$

$$\int \frac{1}{a^2 + x^2} dx = \frac{1}{a}\arctan\left(\frac{x}{a}\right) + C \tag{C.4}$$

$$\int \frac{1}{x} dx = \ln|x| + C \tag{C.5}$$

Appendix D
Closed-Loop Gain Fourier Series Decomposition

It is possible to deduce the expressions for the Fourier series coefficients of the MEMS spectrum analyzer system closed-loop gain, derived in Subsection 3.1.1. Back to Equation 3.51, rewritten here as

$$\begin{aligned} G_{dif}(t) &= \frac{x_{dif}(t)}{a_{ext}(t)} \\ &= G_1(t) - G_2(t) \\ &= \left(\frac{m_1}{k_1} \frac{1}{1 - \frac{\beta_1}{2} - \frac{\beta_1}{2}\cos(\omega_d t)} - \frac{m_2}{k_2} \frac{1}{1 - \frac{\beta_2}{2} + \frac{\beta_2}{2}\cos(\omega_d t)} \right), \end{aligned} \tag{D.1}$$

and to the Fourier series coefficients in the trigonometric form, given by equations A.38 to A.40, also rewritten here.

$$a_0 = \frac{1}{T_0} \int_{T_0} x(t)\, dt \tag{D.2}$$

$$b_n = \frac{2}{T_0} \int_{T_0} x(t) \cos n\omega_0 t\, dt \tag{D.3}$$

$$c_n = \frac{2}{T_0} \int_{T_0} x(t) \sin n\omega_0 t\, dt \tag{D.4}$$

Considering the fundamental angular frequency $\omega_0 = \omega_d$ – and, therefore, $T_0 = T_d = 2\pi/\omega_d$ –, replacing $x(t)$ by the second expression of Equation D.1 for $G_{dif}(t)$ in equations D.2, D.3 and D.4 and defining the integration between $-T_d/2$ and $T_d/2$:

$$\begin{aligned} a_{dif,0} &= \frac{1}{T_d} \int_{-T_d/2}^{T_d/2} [G_1(t) - G_2(t)]\, dt \\ &= \frac{1}{T_d} \int_{-T_d/2}^{T_d/2} G_1(t) dt - \frac{1}{T_d} \int_{-T_d/2}^{T_d/2} G_2(t) dt \\ &= a_{1,0} - a_{2,0}\,, \end{aligned} \tag{D.5}$$

$$
\begin{aligned}
b_{dif,n} &= \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}[G_1(t)-G_2(t)]\cos n\omega_d t dt \\
&= \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}G_1(t)\cos n\omega_d t dt - \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}G_2(t)\cos n\omega_d t dt \\
&= b_{1,n} - b_{2,n}\,,
\end{aligned}
\tag{D.6}
$$

$$
\begin{aligned}
c_{dif,n} &= \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}[G_1(t)-G_2(t)]\sin n\omega_d t dt \\
&= \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}G_1(t)\sin n\omega_d t dt - \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}G_2(t)\sin n\omega_d t dt \\
&= c_{1,n} - c_{2,n}\,.
\end{aligned}
\tag{D.7}
$$

Then, the trigonometric Fourier coefficients for the closed-loop gains $G_1(t)$ and $G_2(t)$ can be obtained using Equation 3.31 and Equation 3.49, rewritten as equations D.8 and D.9.

$$
\begin{aligned}
G_1(t) &= \frac{m_1}{k_1}\frac{1}{1-\frac{\beta_1}{2}-\frac{\beta_1}{2}\cos(\omega_d t)} \\
&= \frac{2m_1}{k_1}\frac{1}{2-\beta_1-\beta_1\cos(\omega_d t)}
\end{aligned}
\tag{D.8}
$$

$$
\begin{aligned}
G_2(t) &= \frac{m_2}{k_2}\frac{1}{1-\frac{\beta_2}{2}+\frac{\beta_2}{2}\cos(\omega_d t)} \\
&= \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}a_{ext}(t)
\end{aligned}
\tag{D.9}
$$

The Fourier series can be computed separately for each gain. For the closed-loop gain $G_1(t)$, considering the first four harmonics:

$$
a_{1,0} = \frac{1}{T_d}\int_{-T_d/2}^{T_d/2}\frac{2m_1}{k_1}\frac{1}{2-\beta_1-\beta_1\cos(\omega_d t)}dt \tag{D.10}
$$

$$
b_{1,1} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}\frac{2m_1}{k_1}\frac{1}{2-\beta_1-\beta_1\cos(\omega_d t)}\cos(\omega_d t)dt \tag{D.11}
$$

$$
b_{1,2} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}\frac{2m_1}{k_1}\frac{1}{2-\beta_1-\beta_1\cos(\omega_d t)}\cos(2\omega_d t)dt \tag{D.12}
$$

$$
b_{1,3} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}\frac{2m_1}{k_1}\frac{1}{2-\beta_1-\beta_1\cos(\omega_d t)}\cos(3\omega_d t)dt \tag{D.13}
$$

$$
b_{1,4} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2}\frac{2m_1}{k_1}\frac{1}{2-\beta_1-\beta_1\cos(\omega_d t)}\cos(4\omega_d t)dt \tag{D.14}
$$

$$c_{1,1} = \frac{2}{T_d} \int_{-T_d/2}^{T_d/2} \frac{2m_1}{k_1} \frac{1}{2 - \beta_1 - \beta_1 \cos(\omega_d t)} (\sin \omega_d t) dt \tag{D.15}$$

$$c_{1,2} = \frac{2}{T_d} \int_{-T_d/2}^{T_d/2} \frac{2m_1}{k_1} \frac{1}{2 - \beta_1 - \beta_1 \cos(\omega_d t)} \sin(2\omega_d t) dt \tag{D.16}$$

$$c_{1,3} = \frac{2}{T_d} \int_{-T_d/2}^{T_d/2} \frac{2m_1}{k_1} \frac{1}{2 - \beta_1 - \beta_1 \cos(\omega_d t)} \sin(3\omega_d t) dt \tag{D.17}$$

$$c_{1,4} = \frac{2}{T_d} \int_{-T_d/2}^{T_d/2} \frac{2m_1}{k_1} \frac{1}{2 - \beta_1 - \beta_1 \cos(\omega_d t)} \sin(4\omega_d t) dt \,. \tag{D.18}$$

To solve equations D.10 to D.18, some individual variables replacements and considerations must be made, but there are some common considerations to several of them. During the deduction developing these substitutions will be more clear.

The first variable replacement is

$$u_1 = \omega_d t \,, \tag{D.19}$$

with

$$\frac{du_1}{dt} = \omega_d = \frac{2\pi}{T_d} \,. \tag{D.20}$$

The second substitution is:

$$u_2 = \tan\left(u_1/2\right) \,. \tag{D.21}$$

Now, to obtain du_2/du_1, the derivative quotient and the chain rules can be used.

$$\begin{aligned} \frac{du_2}{du_1} &= \frac{d\left[\tan\left(u_1/2\right)\right]}{du_1} \\ &= \frac{d\left[\frac{\sin(u_1/2)}{\cos(u_1/2)}\right]}{du_1} \\ &= \frac{1}{2}\left[\frac{\cos^2\left(u_1/2\right) + \sin^2\left(u_1/2\right)}{\cos^2\left(u_1/2\right)}\right] \\ &= \frac{1}{2}\frac{1}{\cos^2\left(u_1/2\right)} \end{aligned} \tag{D.22}$$

To obtain an expression for $\cos^2\left(u_1/2\right)$ as a function of u_2, the trigonometric identity of Equation B.1 can be used with $A = u_1/2$.

$$\sin^2\left(u_1/2\right) + \cos^2\left(u_1/2\right) = 1 \tag{D.23}$$

Dividing both sides of Equation D.23 by $\cos^2\left(u_1/2\right)$:

$$\frac{\sin^2\left(u_1/2\right)}{\cos^2\left(u_1/2\right)} + 1 = \frac{1}{\cos^2\left(u_1/2\right)} \,. \tag{D.24}$$

Rearranging the terms and using Equation D.21:

$$\begin{aligned}\cos^2(u_1/2) &= \frac{1}{1+\tan^2(u_1/2)} \\ &= \frac{1}{1+u_2^2}\,,\end{aligned} \tag{D.25}$$

so Equation D.22 becomes:

$$\frac{du_2}{du_1} = \frac{1+u_2^2}{2}\,. \tag{D.26}$$

Now, to obtain an expression for $\cos u_1$ as a function of u_2, Equation B.4 can be used considering $A = B = u_1/2$.

$$\cos(u_1/2 + u_1/2) = \cos^2(u_1) - \sin^2(u_1) \tag{D.27}$$

Rearranging Equation D.23 and substituting in Equation D.27:

$$\cos u_1 = 2\cos^2(u_1/2) - 1\,. \tag{D.28}$$

Using Equation D.25 in Equation D.28:

$$\begin{aligned}\cos u_1 &= \frac{2}{1+u_2^2} - 1 \\ &= \frac{1-u_2^2}{1+u_2^2}\,.\end{aligned} \tag{D.29}$$

The general considerations used are:

$$v_1 = \frac{2-\beta_1}{\beta_1}\,, \tag{D.30}$$

and

$$v_2^2 = \frac{v_1 - 1}{v_1 + 1} = 1 - \beta_1\,. \tag{D.31}$$

Back to Equation D.10, using equations D.19 and D.20:

$$\begin{aligned}a_{1,0} &= \frac{1}{T_d}\int_{u_{1i}}^{u_{1f}} \frac{2m_1}{k_1}\frac{1}{2-\beta_1-\beta_1\cos u_1}\frac{T_d}{2\pi}du_1 \\ &= \frac{m_1}{\pi k_1}\int_{-\pi}^{\pi}\frac{1}{2-\beta_1-\beta_1\cos u_1}du_1\,.\end{aligned} \tag{D.32}$$

Replacing Equation D.30 in Equation D.32:

$$a_{1,0} = \frac{m_1}{\pi k_1 \beta_1}\int_{-\pi}^{\pi}\frac{1}{v_1 - \cos u_1}du_1\,. \tag{D.33}$$

Equations D.26 and D.29 can be used to rewrite Equation D.33 as:

$$\begin{aligned} a_{1,0} &= \frac{m_1}{\pi k_1 \beta_1} \int_{u_{2i}}^{u_{2f}} \frac{1}{v_1 - \left(\frac{1-u_2^2}{1+u_2^2}\right)} \frac{2}{1+u_2^2} du_2 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{(v_1-1)+(v_1+1)\,u_2^2} du_2 \,. \end{aligned} \tag{D.34}$$

Using Equation D.31 in Equation D.34:

$$a_{1,0} = \frac{2m_1}{\pi k_1 \beta_1 \,(v_1+1)} \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{v_2^2 + u_2^2} du_2 \,. \tag{D.35}$$

According to Equation C.4, if $v_2^2 > 0$ – i.e., $\beta_1 < 1$, which is the same condition for the stability of the positive feedback loop given in Equation 3.40 – Equation D.35 can be solved.

$$a_{1,0} = \frac{2m_1}{\pi k_1 \beta_1 \,(v_1+1)} \frac{1}{v_2} \arctan\left(\frac{u_2}{v_2}\right) \Big|_{\tan(-\pi/2)}^{\tan(\pi/2)} \tag{D.36}$$

Using equations D.30 and D.31 in Equation D.36:

$$a_{1,0} = \frac{m_1}{\pi k_1} \frac{1}{\sqrt{1-\beta_1}} \arctan\left(\frac{u_2}{\sqrt{1-\beta_1}}\right) \Big|_{\tan(-\pi/2)}^{\tan(\pi/2)} . \tag{D.37}$$

Since, at the limit,

$$\frac{\tan(\pi/2)}{\sqrt{1-\beta_1}} = \tan(\pi/2) \,, \tag{D.38}$$

$$\frac{\tan(-\pi/2)}{\sqrt{1-\beta_1}} = \tan(-\pi/2) \,, \tag{D.39}$$

Equation D.37 can be computed.

$$\begin{aligned} a_{1,0} &= \frac{m_1}{\pi k_1} \frac{1}{\sqrt{1-\beta_1}} \left[\frac{\pi}{2} - \left(-\frac{\pi}{2}\right)\right] \\ &= \frac{m_1}{k_1} \frac{1}{\sqrt{1-\beta_1}} \end{aligned} \tag{D.40}$$

Now, referring to Equation D.11, the procedure is similar to that used with Equation D.10,

so its steps are condensed.

$$
\begin{aligned}
b_{1,1} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\cos u_1}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\cos u_1}{v_1 - \cos u_1} du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left(-1 + \frac{v_1}{v_1 - \cos u_1} \right) du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \left[-\int_{-\pi}^{\pi} du_1 + 2v_1 \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{(v_1 - 1) + (v_1 + 1) u_2^2} du_2 \right] \\
&= \frac{2m_1}{\pi k_1 \beta_1} \left[-\int_{-\pi}^{\pi} du_1 + \frac{2v_1}{(v_1 + 1)} \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{v_2^2 + u_2^2} du_2 \right] \\
&= \frac{2m_1}{\pi k_1 \beta_1} \left[-u_1 \Big|_{u_1 = -\pi}^{\pi} + \frac{2v_1}{(v_1 + 1)} \frac{1}{v_2} \arctan\left(\frac{u_2}{v_2} \right) \Big|_{u_2 = \tan(-\pi/2)}^{\tan(\pi/2)} \right] \\
&= \frac{2m_1}{k_1 \beta_1} \left[-2 + \frac{2 - \beta_1}{\sqrt{1 - \beta_1}} \right]
\end{aligned}
\tag{D.41}
$$

Similarly, to $b_{1,2}$, the method take the same steps starting from Equation D.12:

$$
\begin{aligned}
b_{1,2} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\cos(2u_1)}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\cos(2u_1)}{v_1 - \cos u_1} du_1 \,.
\end{aligned}
\tag{D.42}
$$

Now $\cos(2u_1)$ can be expressed using Equation B.4 with $A = B = u_1$:

$$
\cos(2u_1) = \cos^2 u_1 - \sin^2 u_1 \,. \tag{D.43}
$$

Considering $A = u_1$ in Equation B.1:

$$
\sin^2 u_1 + \cos^2 u_1 = 1 \Rightarrow \sin^2 u_1 = 1 - \cos^2 u_1 \,. \tag{D.44}
$$

Replacing Equation D.44 in Equation D.43:

$$
\cos(2u_1) = 2\cos^2 u_1 - 1 \,. \tag{D.45}
$$

Using Equation D.45 in Equation D.42:

$$
\begin{aligned}
b_{1,2} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{2\cos^2 u_1 - 1}{v_1 - \cos u_1} du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left(-2\cos u_1 - 2v_1 + \frac{2v_1^2 - 1}{v_1 - \cos u_1} \right) du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \left[\int_{-\pi}^{\pi} \left(-2\cos u_1 - 2v_1 \right) du_1 + 2\left(2v_1^2 - 1\right) \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{(v_1 - 1) + (v_1 + 1)\, u_2^2} du_2 \right] \\
&= \frac{2m_1}{\pi k_1 \beta_1} \left[\int_{-\pi}^{\pi} \left(-2\cos u_1 - 2v_1 \right) du_1 + \frac{2\left(2v_1^2 - 1\right)}{(v_1 + 1)} \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{v_2^2 + u_2^2} du_2 \right] \\
&= \frac{2m_1}{\pi k_1 \beta_1} \left[\left(-2\sin u_1 - 2v_1 u_1 \right) \Big|_{u_1 = -\pi}^{\pi} + \frac{2\left(2v_1^2 - 1\right)}{(v_1 + 1)} \frac{1}{v_2} \arctan\left(\frac{u_2}{v_2} \right) \Big|_{u_2 = \tan(-\pi/2)}^{\tan(\pi/2)} \right] \\
&= \frac{2m_1}{\pi k_1 \beta_1} \left[-4v_1 \pi + \frac{2\left(2v_1^2 - 1\right)}{(v_1 + 1)} \frac{1}{\sqrt{1 - \beta_1}} \pi \right] \\
&= \frac{2m_1}{k_1 \beta_1} \left[4 - \frac{8}{\beta_1} + \frac{8 - 8\beta_1 + \beta_1^2}{\beta_1 \sqrt{1 - \beta_1}} \right] .
\end{aligned}
\tag{D.46}
$$

Now, Equation D.13 leads to:

$$
\begin{aligned}
b_{1,3} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\cos(3u_1)}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\cos(3u_1)}{v_1 - \cos u_1} du_1 .
\end{aligned}
\tag{D.47}
$$

Again, using Equation B.4 of Appendix B, now with $A = u_1$ and $B = 2u_1$:

$$
\cos\left(u_1 + 2u_1\right) = \cos u_1 \cos\left(2u_1\right) - \sin u_1 \sin\left(2u_1\right) . \tag{D.48}
$$

Equation D.45 can be used to express $\cos(2u_1)$. On the other hand, using Equation B.2 with $A = B = u_1$ allows to write $\sin(2u_1)$:

$$
\sin\left(2u_1\right) = 2\sin u_1 \cos u_1 . \tag{D.49}
$$

Replacing equations D.45 and D.49 in Equation D.48:

$$
\begin{aligned}
\cos\left(3u_1\right) &= \cos u_1 \left(2\cos^2 u_1 - 1\right) - \sin u_1 \left(2\sin u_1 \cos u_1\right) \\
&= 2\cos^3 u_1 - \cos u_1 - 2\sin^2 u_1 \cos u_1 .
\end{aligned}
\tag{D.50}
$$

Using Equation D.44:

$$
\begin{aligned}
\cos\left(3u_1\right) &= 2\cos^3 u_1 - \cos u_1 - 2\left(1 - \cos^2 u_1\right) \cos u_1 \\
&= 4\cos^3 u_1 - 3\cos u_1 .
\end{aligned}
\tag{D.51}
$$

Substituting Equation D.51 in Equation D.47:

$$\begin{aligned} b_{1,3} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{4\cos^3 u_1 - 3\cos u_1}{v_1 - \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left[-4\cos^2 u_1 - 4v_1 \cos u_1 - \left(4v_1^2 - 3\right) + \frac{4v_1^3 - 3v_1}{v_1 - \cos u_1} \right] du_1 . \end{aligned} \tag{D.52}$$

Equation D.45 can be used, rearranging its terms, so:

$$\begin{aligned} b_{1,3} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left[-2\left[\cos\left(2u_1\right) + 1\right] - 4v_1 \cos u_1 - \left(4v_1^2 - 3\right) + \frac{4v_1^3 - 3v_1}{v_1 - \cos u_1} \right] du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left[-2\cos\left(2u_1\right) - 4v_1 \cos u_1 - \left(4v_1^2 - 1\right) + \frac{4v_1^3 - 3v_1}{v_1 - \cos u_1} \right] du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left\{ \int_{-\pi}^{\pi} \left[-2\cos\left(2u_1\right) - 4v_1 \cos u_1 - \left(4v_1^2 - 1\right) \right] du_1 \right\} + \\ &\quad + \frac{2m_1}{\pi k_1 \beta_1} \left\{ 2\left(4v_1^3 - 3v_1\right) \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{\left(v_1 - 1\right) + \left(v_1 + 1\right) u_2^2} du_2 \right\} \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left\{ \int_{-\pi}^{\pi} \left[-2\cos\left(2u_1\right) - 4v_1 \cos u_1 - \left(4v_1^2 - 1\right) \right] du_1 \right\} + \\ &\quad + \frac{2m_1}{\pi k_1 \beta_1} \left\{ \frac{2\left(4v_1^3 - 3v_1\right)}{\left(v_1 + 1\right)} \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{v_2^2 + u_2^2} du_2 \right\} \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left[-\sin\left(2u_1\right) - 4v_1 \sin u_1 - \left(4v_1^2 - 1\right) u_1 \right] \Big|_{u_1 = -\pi}^{\pi} + \\ &\quad + \frac{2m_1}{\pi k_1 \beta_1} \left[\frac{2\left(4v_1^3 - 3v_1\right)}{\left(v_1 + 1\right)} \frac{1}{v_2} \arctan\left(\frac{u_2}{v_2}\right) \right] \Big|_{u_2 = \tan(-\pi/2)}^{\tan(\pi/2)} \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left[-2\left(4v_1^2 - 1\right)\pi + \frac{2\left(4v_1^3 - 3v_1\right)}{\left(v_1 + 1\right)} \frac{1}{\sqrt{1 - \beta_1}} \pi \right] \\ &= \frac{2m_1}{k_1 \beta_1} \left[-6 + \frac{32}{\beta_1} - \frac{32}{\beta_1^2} + \frac{32 - 48\beta_1 + 18\beta_1^2 - \beta_1^3}{\beta_1^2 \sqrt{1 - \beta_1}} \right] . \end{aligned} \tag{D.53}$$

Finally, back to Equation D.14:

$$\begin{aligned} b_{1,4} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\cos(4u_1)}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\cos(4u_1)}{v_1 - \cos u_1} du_1 . \end{aligned} \tag{D.54}$$

Using Equation B.4 with $A = B + 2u_1$:

$$\cos\left(2u_1 + 2u_1\right) = \cos^2\left(2u_1\right) - \sin^2\left(2u_1\right) . \tag{D.55}$$

Considering equations D.45 and D.49:

$$\begin{aligned}\cos(4u_1) &= \left(2\cos^2 u_1 - 1\right)^2 - (2\sin u_1 \cos u_1)^2 \\ &= 4\cos^4 u_1 - 4\cos^2 u_1 + 1 - 4\sin^2 u_1 \cos^2 u_1 \,.\end{aligned} \tag{D.56}$$

Replacing Equation D.44 in Equation D.56:

$$\begin{aligned}\cos(4u_1) &= 4\cos^4 u_1 - 4\cos^2 u_1 + 1 - 4\left(1 - \cos^2 u_1\right)\cos^2 u_1 \\ &= 8\cos^4 u_1 - 8\cos^2 u_1 + 1 \,.\end{aligned} \tag{D.57}$$

Back to Equation D.54, considering Equation D.57:

$$\begin{aligned}b_{1,4} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{8\cos^4 u_1 - 8\cos^2 u_1 + 1}{v_1 - \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left[-8\cos^3 u_1 - 8v_1 \cos^2 u_1 - \left(8v_1^2 - 8\right)\cos u_1 - \left(8v_1^3 - 8v_1\right) + \frac{8v_1^4 - 8v_1^2 + 1}{v_1 - \cos u_1}\right] du_1 \,.\end{aligned} \tag{D.58}$$

Now, to obtain a expression for $\cos^3 u_1$, it is possible to use Equation D.44, rearranged.

$$\begin{aligned}\cos^3 u_1 &= \cos u_1 \cos^2 u_1 \\ &= \cos u_1 \left(1 - \sin^2 u_1\right) \\ &= \cos u_1 - \cos u_1 \sin^2 u_1\end{aligned} \tag{D.59}$$

Substituting Equations D.45 – rearranged – and D.59 in Equation D.58:

$$b_{1,4} = \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left[8\cos u_1 \sin^2 u_1 - 4v_1 \cos(2u_1) - 8v_1^2 \cos u_1 - \left(8v_1^3 - 4v_1\right) + \frac{8v_1^4 - 8v_1^2 + 1}{v_1 - \cos u_1}\right] du_1 \,. \tag{D.60}$$

It is possible to evaluate each of the five parts of the integral of Equation D.60 separately. The first one is:

$$I_1 = \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} 8\cos u_1 \sin^2 u_1 du_1 \,. \tag{D.61}$$

Substituting $u_3 = \sin u_1$, $du_3 = \cos u_1 du_1$:

$$\begin{aligned}I_1 &= \frac{2m_1}{\pi k_1 \beta_1} \int_{\sin(-\pi)}^{\sin \pi} 8u_3^2 du_3 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \frac{8u_3^3}{3}\Big|_{\sin(-\pi)}^{\sin(\pi)} \\ &= 0 \,.\end{aligned} \tag{D.62}$$

The second part of Equation D.60:

$$\begin{aligned}I_2 &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} -4v_1 \cos(2u_1)\, du_1 \\ &= \frac{-4m_1 v_1}{\pi k_1 \beta_1} \sin(2u_1)\Big|_{-\pi}^{\pi} \\ &= 0 \,.\end{aligned} \tag{D.63}$$

The third portion, I_3, is:

$$\begin{aligned} I_3 &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} -8v_1^2 \cos u_1 du_1 \\ &= \frac{-16m_1 v_1^2}{\pi k_1 \beta_1} \sin u_1 \Big|_{-\pi}^{\pi} \\ &= 0\,. \end{aligned} \tag{D.64}$$

The fourth part of Equation D.60 is given by Equation D.65.

$$\begin{aligned} I_4 &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} -\left(8v_1^3 - 4v_1\right) du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left(-8v_1^3 + 4v_1\right) u_1 \Big|_{-\pi}^{\pi} \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left(-8v_1^3 + 4v_1\right) 2\pi \\ &= \frac{2m_1}{k_1 \beta_1} \left(8 - \frac{80}{\beta_1} + \frac{192}{\beta_1^2} - \frac{128}{\beta_1^3}\right) \end{aligned} \tag{D.65}$$

Finally, the last part I_5 is:

$$\begin{aligned} I_5 &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{8v_1^4 - 8v_1^2 + 1}{v_1 - \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} 2\left(8v_1^4 - 8v_1^2 + 1\right) \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{(v_1 - 1) + (v_1 + 1)\, u_2^2} du_2 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \frac{2\left(8v_1^4 - 8v_1^2 + 1\right)}{(v_1 + 1)} \int_{\tan(-\pi/2)}^{\tan(\pi/2)} \frac{1}{v_2^2 + u_2^2} du_2 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \frac{2\left(8v_1^4 - 8v_1^2 + 1\right)}{(v_1 + 1)} \frac{1}{v_2} \arctan\left(\frac{u_2}{v_2}\right) \Big|_{\tan(-\pi/2)}^{\tan(\pi/2)} \\ &= \frac{2m_1}{\pi k_1 \beta_1} \frac{2\left(8v_1^4 - 8v_1^2 + 1\right)}{(v_1 + 1)} \frac{1}{\sqrt{1 - \beta_1}} \pi \\ &= \frac{2m_1}{k_1 \beta_1} \frac{\left(128 - 256\beta_1 + 160\beta_1^2 - 32\beta_1^3 + \beta_1^4\right)}{\beta_1^3 \sqrt{1 - \beta_1}}\,. \end{aligned} \tag{D.66}$$

So, the coefficient $b_{1,4}$:

$$b_{1,4} = \frac{2m_1}{k_1 \beta_1} \left(8 - \frac{80}{\beta_1} + \frac{192}{\beta_1^2} - \frac{128}{\beta_1^3} + \frac{128 - 256\beta_1 + 160\beta_1^2 - 32\beta_1^3 + \beta_1^4}{\beta_1^3 \sqrt{1 - \beta_1}}\right)\,. \tag{D.67}$$

Now, back to Equation D.15, to compute the coefficient $c_{1,1}$:

$$\begin{aligned} c_{1,1} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\sin u_1}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\sin u_1}{v_1 - \cos u_1} du_1\,. \end{aligned} \tag{D.68}$$

To solve Equation D.68, a third variable replacement is needed, as given by Equation D.69, which derivative is given by Equation D.70.

$$u_3 = v_1 - \cos u_1 \tag{D.69}$$

$$\frac{du_3}{du_1} = \sin u_1 \tag{D.70}$$

Back to Equation D.68:

$$\begin{aligned} c_{1,1} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{u_{3,i}}^{u_{3,f}} \frac{\sin u_1}{u_3} \frac{1}{\sin u_1} du_3 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{v_1+1}^{v_1+1} \frac{-1}{v_1 - u_3} du_3 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \ln |u_3| \Big|_{v_1+1}^{v_1+1} \\ &= 0\,. \end{aligned} \tag{D.71}$$

Now referring to Equation D.16:

$$\begin{aligned} c_{1,2} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\sin(2u_1)}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\sin(2u_1)}{v_1 - \cos u_1} du_1\,. \end{aligned} \tag{D.72}$$

Using Equation D.49 and variable replacement

$$u_4 = \cos u_1\,, \tag{D.73}$$

$$\frac{du_4}{du_1} = -\sin u_1 \tag{D.74}$$

in Equation D.72:

$$
\begin{aligned}
c_{1,2} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{2 \sin u_1 \cos u_1}{v_1 - \cos u_1} du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \int_{u_{4,i}}^{u_{4,f}} \frac{2u_4 \sin u_1}{v_1 - u_4} \frac{-1}{\sin u_1} du_4 \\
&= \frac{4m_1}{\pi k_1 \beta_1} \int_{-1}^{-1} \frac{u_4}{u_4 - v_1} du_4 \\
&= \frac{4m_1}{\pi k_1 \beta_1} \int_{-1}^{-1} \left[1 + \frac{v_1}{u_4 - v_1} \right] du_4 \\
&= \frac{4m_1}{\pi k_1 \beta_1} \left[\int_{-1}^{-1} du_4 + \int_{-1}^{-1} \frac{v_1}{u_4 - v_1} du_4 \right] \\
&= \frac{4m_1}{\pi k_1 \beta_1} \left[u_4 \Big|_{-1}^{-1} + \int_{-1}^{-1} \frac{v_1}{u_4 - v_1} du_4 \right] \\
&= \frac{4m_1}{\pi k_1 \beta_1} \int_{-1}^{-1} \frac{v_1}{u_4 - v_1} du_4 \, .
\end{aligned}
\tag{D.75}
$$

Replacing

$$u_5 = u_4 - v_1 \, , \tag{D.76}$$

$$\frac{du_5}{du_4} = 1 \tag{D.77}$$

in Equation D.75:

$$
\begin{aligned}
c_{1,2} &= \frac{4m_1}{\pi k_1 \beta_1} \int_{u_{5,i}}^{u_{5,f}} \frac{v_1}{u_5} du_5 \\
&= \frac{4m_1 v_1}{\pi k_1 \beta_1} \int_{-1-v_1}^{-1-v_1} \frac{1}{u_5} du_5 \\
&= \frac{4m_1 v_1}{\pi k_1 \beta_1} \ln |u_5| \Big|_{-1-v_1}^{-1-v_1} \\
&= 0 \, .
\end{aligned}
\tag{D.78}
$$

To compute the coefficient $c_{1,3}$, according to Equation D.17:

$$
\begin{aligned}
c_{1,3} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\sin(3u_1)}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\
&= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\sin(3u_1)}{v_1 - \cos u_1} du_1 \, .
\end{aligned}
\tag{D.79}
$$

Using Equation B.2 with $A = u_1$ and $B = 2u_1$:

$$\sin(3u_1) = \sin u_1 \cos(2u_1) + \sin(2u_1) \cos u_1 \,. \tag{D.80}$$

Replacing equations D.45 and D.49 in Equation D.80:

$$\begin{aligned} \sin(3u_1) &= \sin u_1 \left(2\cos^2 u_1 - 1\right) + 2 \sin u_1 \cos u_1 \cos u_1 \\ &= 4 \sin u_1 \cos^2 u_1 - \sin u_1 \,. \end{aligned} \tag{D.81}$$

Back to Equation D.79:

$$\begin{aligned} c_{1,3} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{4 \sin u_1 \cos^2 u_1 - \sin u_1}{v_1 - \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left[-4 \sin u_1 \cos u_1 - 4 v_1 \sin u_1 + \frac{\left(4v_1^2 - 1\right) \sin u_1}{v_1 - \cos u_1} \right] du_1 \,. \end{aligned} \tag{D.82}$$

The first part of Equation D.82 can be solved using the variable substitution of Equation D.73, the second part can be solved without further variable substitution, and the third part can be solved with the substitution of Equation D.69. Performing this procedure:

$$\begin{aligned} c_{1,3} &= \frac{2m_1}{\pi k_1 \beta_1} \left[4 \int_{-1}^{-1} u_4 du_4 - 4 v_1 \int_{-\pi}^{\pi} \sin u_1 du_1 + \left(4v_1^2 - 1\right) \int_{v_1+1}^{v_1+1} \frac{1}{u_3} du_3 \right] \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left[2u_4^2 \Big|_{-1}^{-1} + 4 v_1 \cos u_1 \Big|_{-\pi}^{\pi} + \left(4v_1^2 - 1\right) \ln |u_3| \Big|_{v_1+1}^{v_1+1} \right] \\ &= 0 \,. \end{aligned} \tag{D.83}$$

Finally, Equation D.18 allows the calculation of $c_{1,4}$:

$$\begin{aligned} c_{1,4} &= \frac{2m_1}{\pi k_1} \int_{-\pi}^{\pi} \frac{\sin(4u_1)}{2 - \beta_1 - \beta_1 \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{\sin(4u_1)}{v_1 - \cos u_1} du_1 \,. \end{aligned} \tag{D.84}$$

The expression for $\sin(4u_1)$ can be obtained using Equation B.2 with $A = B = 2u_1$, besides Equations D.45 and D.49:

$$\begin{aligned} \sin(4u_1) &= 2 \sin(2u_1) \cos(2u_1) \\ &= 4 \sin u_1 \cos u_1 \left(2\cos^2 u_1 - 1\right) \\ &= 8 \sin u_1 \cos^3 u_1 - 4 \sin u_1 \cos u_1 \,. \end{aligned} \tag{D.85}$$

Back to Equation D.84:

$$\begin{aligned} c_{1,4} &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \frac{8 \sin u_1 \cos^3 u_1 - 4 \sin u_1 \cos u_1}{v_1 - \cos u_1} du_1 \\ &= \frac{2m_1}{\pi k_1 \beta_1} \int_{-\pi}^{\pi} \left[-8 \sin u_1 \cos^2 u_1 - 8 v_1 \sin u_1 \cos u_1 - \left(8v_1^2 - 4\right) \sin u_1 + \frac{\left(8v_1^3 - 4v_1\right) \sin u_1}{v_1 - \cos u_1} \right] du_1 . \end{aligned} \tag{D.86}$$

The first two parts of Equation D.86 can be solved using Equation D.73, the third part can be solved directly, and the last part can be solved using Equation D.69:

$$\begin{aligned} c_{1,4} &= \frac{2m_1}{\pi k_1 \beta_1} \left[8 \int_{-1}^{-1} u_4^2 du_4 + 8 v_1 \int_{-1}^{-1} u_4 du_4 - \left(8v_1^2 - 4\right) \int_{-\pi}^{\pi} \sin u_1 du_1 + \left(8v_1^3 - 4v_1\right) \int_{v_1+1}^{v_1+1} \frac{1}{u_3} du_3 \right] \\ &= \frac{2m_1}{\pi k_1 \beta_1} \left[\frac{8}{3} u_4^3 \Big|_{-1}^{-1} + 4 v_1 u_4^2 \Big|_{-1}^{-1} + \left(8v_1^2 - 4\right) \cos u_1 \Big|_{-\pi}^{\pi} + \left(8v_1^3 - 4v_1\right) \ln |u_3| \Big|_{v_1+1}^{v_1+1} \right] \\ &= 0 . \end{aligned} \tag{D.87}$$

Summarizing, for the closed-loop gain $G_1(t)$, the Fourier composition is given by:

$$a_{1,0} = \frac{m_1}{k_1} \frac{1}{\sqrt{1 - \beta_1}} \tag{D.88}$$

$$b_{1,1} = \frac{2m_1}{k_1 \beta_1} \left[-2 + \frac{2 - \beta_1}{\sqrt{1 - \beta_1}} \right] \tag{D.89}$$

$$b_{1,2} = \frac{2m_1}{k_1 \beta_1} \left[4 - \frac{8}{\beta_1} + \frac{8 - 8\beta_1 + \beta_1^2}{\beta_1 \sqrt{1 - \beta_1}} \right] \tag{D.90}$$

$$b_{1,3} = \frac{2m_1}{k_1 \beta_1} \left[-6 + \frac{32}{\beta_1} - \frac{32}{\beta_1^2} + \frac{32 - 48\beta_1 + 18\beta_1^2 - \beta_1^3}{\beta_1^2 \sqrt{1 - \beta_1}} \right] \tag{D.91}$$

$$b_{1,4} = \frac{2m_1}{k_1 \beta_1} \left(8 - \frac{80}{\beta_1} + \frac{192}{\beta_1^2} - \frac{128}{\beta_1^3} + \frac{128 - 256\beta_1 + 160\beta_1^2 - 32\beta_1^3 + \beta_1^4}{\beta_1^3 \sqrt{1 - \beta_1}} \right) \tag{D.92}$$

$$c_{1,1} = 0 \tag{D.93}$$

$$c_{1,2} = 0 \tag{D.94}$$

$$c_{1,3} = 0 \tag{D.95}$$

$$c_{1,4} = 0 . \tag{D.96}$$

The procedure to obtain the Fourier composition of the gain $G_2(t)$ is analogous to that adopted for the gain $G_1(t)$, and, therefore, the steps are omitted in the following.

$$\begin{aligned} a_{2,0} &= \frac{1}{T_d} \int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2} \frac{1}{2 - \beta_2 + \beta_2 \cos(\omega_d t)} dt \\ &= \frac{m_2}{k_2} \frac{1}{\sqrt{1 - \beta_2}} \end{aligned} \tag{D.97}$$

$$b_{2,1} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}\cos(\omega_d t)dt \\ = \frac{2m_2}{k_2\beta_2}\left[2-\frac{2-\beta_2}{\sqrt{1-\beta_2}}\right] \tag{D.98}$$

$$b_{2,2} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}\cos(2\omega_d t)dt \\ = \frac{2m_2}{k_2\beta_2}\left[4-\frac{8}{\beta_2}+\frac{8-8\beta_2+\beta_2^2}{\beta_2\sqrt{1-\beta_2}}\right] \tag{D.99}$$

$$b_{2,3} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}\cos(3\omega_d t)dt \\ = \frac{2m_2}{k_2\beta_2}\left[6-\frac{32}{\beta_2}+\frac{32}{\beta_2^2}-\frac{32-48\beta_2+18\beta_2^2-\beta_2^3}{\beta_2^2\sqrt{1-\beta_2}}\right] \tag{D.100}$$

$$b_{2,4} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}\cos(4\omega_d t)dt \\ = \frac{2m_2}{k_2\beta_2}\left(8-\frac{80}{\beta_2}+\frac{192}{\beta_2^2}-\frac{128}{\beta_2^3}+\frac{128-256\beta_2+160\beta_2^2-32\beta_2^3+\beta_2^4}{\beta_2^3\sqrt{1-\beta_2}}\right) \tag{D.101}$$

$$c_{2,1} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}(\sin\omega_d t)dt \\ = 0 \tag{D.102}$$

$$c_{2,2} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}\sin(2\omega_d t)dt \\ = 0 \tag{D.103}$$

$$c_{2,3} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}\sin(3\omega_d t)dt \\ = 0 \tag{D.104}$$

$$c_{2,4} = \frac{2}{T_d}\int_{-T_d/2}^{T_d/2} \frac{2m_2}{k_2}\frac{1}{2-\beta_2+\beta_2\cos(\omega_d t)}\sin(4\omega_d t)dt \\ = 0 \tag{D.105}$$

A Brief Discussion on the Integration Period

Considering the development shown in Appendix A to the envelope $T_0 a_n$, for an aperiodic signal defined for $t \geq 0$ the deduction here is analogous. According to the definition of Fourier series in Appendix A, the coefficient a_n can be obtained by computing the integral of

Equation A.18 in any interval T_0. Since the vibration signals considered are defined to $t \geq 0$, it is interesting to choose the integration interval between 0 and T_0, which is the period of the correspondent periodic signal formed repeating the aperiodic signal. Then,

$$a_n = \frac{1}{T_0} \int_0^{T_0} a_{ext}(t)\, e^{-jn\omega_0 t} dt\,. \tag{D.106}$$

In the limit, $T_0 \to \infty$, so $\omega_d = n\omega_0 = 2\pi n/T_0$ is a continuous variable and the Fourier transform $A(j\omega)$ of the external acceleration $a_{ext}(t)$ is:

$$A(j\omega_d) = \lim_{T_0 \to \infty} \int_0^{T_0} a_{ext}(t)\, e^{-j(2\pi n/T_0)t} dt\,. \tag{D.107}$$

Cretu, Rocha, and Wolffenbuttel [22] indicate that, similarly to the FFT approach, the selectivity of the MEMS spectrum analyzer depends on the integration time, as it is confirmed in Equation D.107. Since the integration period is not infinite, the variable ω_d is not continuous; instead, the spectral components are sampled with $\omega_0 = 2\pi/T_0$ spacing. This is equivalent to say that the excitation voltage angular frequency ω must be varied with π/T_0 steps.

According to Equation A.53, the amplitudes of the spectral components are given by $A(j\omega)(d\omega/2\pi)$ in the limit $T_0 \to \infty$. In the case of the micro spectrum analyzer used, however, these amplitudes are

$$2A(j\omega)\left(\frac{\omega_0}{2\pi}\right) = 2\frac{A(j\omega)}{T_0}\,, \tag{D.108}$$

where the factor 2 is used since only the positive frequencies are of interest.

Therefore, the MEMS spectrum analyzer approximates the spectral component by Equation D.107 in discrete frequencies $\omega = n2\pi/T_0$.

As the angular frequency of the excitation voltage ω raises, its period reduces. Therefore, to represent the signal as aperiodic, it is necessary a smaller integration period. The choice of the integration period must consider that characteristic, so T_0 must be chosen to be much larger than the period correspondent to the smaller frequency that is present in the spectrum mapping. The expected result of this choice is that, as the excitation frequency enlarges, the approximation of the spectral component by the MEMS spectrum analyzer is more accurate. The choice of the integration period must also consider the resulting frequency discretization of the mapped spectrum.

Appendix E
Complementary Results

In this appendix, the complementary simulation results obtained for mapping, tuning, and validation are presented.

Table E.1: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, brick-wall boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	24.58	±	6.95	59.80	±	22.16	290.00	±	63.32
(2)	24.51	±	6.61	60.20	±	21.99	317.80	±	57.37
(3)	27.06	±	9.07	61.20	±	23.51	351.40	±	114.10
(4)	27.46	±	9.79	61.40	±	23.97	370.00	±	120.99
(5)	33.48	±	11.28	62.60	±	25.10	448.00	±	132.50
(6)	24.52	±	6.81	57.20	±	20.75	330.20	±	76.17
(7)	26.21	±	7.32	60.80	±	22.73	344.20	±	77.25
(8)	35.58	±	14.08	68.40	±	31.00	473.40	±	183.16
(9)	29.58	±	9.78	58.60	±	21.66	398.00	±	120.88
(10)	27.54	±	10.17	65.00	±	25.42	370.20	±	117.75
mean	**28.05**	±	**9.19**	**61.52**	±	**23.83**	**369.32**	±	**106.35**

Table E.2: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, brick-wall strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	32.40	±	11.90	66.80	±	26.35	393.20	±	118.68
(2)	14.90	±	1.60	21.00	±	0.00	217.40	±	23.58
(3)	30.04	±	11.00	70.20	±	27.84	390.60	±	127.22
(4)	29.40	±	9.51	60.40	±	22.39	382.80	±	112.52
(5)	23.19	±	12.58	40.20	±	26.33	333.80	±	162.96
(6)	34.87	±	14.81	66.80	±	27.94	481.80	±	209.99
(7)	30.82	±	9.06	61.20	±	22.61	418.60	±	110.47
(8)	29.43	±	8.86	58.00	±	21.68	390.60	±	102.38
(9)	32.36	±	10.80	62.80	±	23.75	414.40	±	125.38
(10)	29.82	±	9.26	58.40	±	21.97	390.40	±	100.13
mean	**28.72**	±	**9.94**	**56.58**	±	**22.09**	**381.36**	±	**119.33**

Table E.3: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, random reinitialization boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	41.26	±	10.37	58.60	±	21.27	571.40	±	94.02
(2)	38.31	±	8.34	59.00	±	21.53	536.60	±	96.58
(3)	36.97	±	8.93	57.60	±	21.09	528.80	±	92.42
(4)	38.20	±	7.81	59.20	±	21.61	539.60	±	82.52
(5)	37.40	±	9.79	61.00	±	23.70	526.20	±	139.88
(6)	33.82	±	5.71	54.60	±	19.26	485.40	±	50.36
(7)	36.96	±	7.48	59.00	±	22.01	522.40	±	94.55
(8)	37.57	±	7.08	58.80	±	21.59	525.00	±	95.06
(9)	36.83	±	8.54	59.00	±	21.99	503.00	±	80.50
(10)	41.69	±	11.60	62.40	±	24.02	583.00	±	132.53
mean	**37.90**	±	**8.56**	**58.92**	±	**21.81**	**532.14**	±	**95.84**

Table E.4: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, random reinitialization boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	42.59	±	10.17	58.60	±	21.17	552.00	±	83.71
(2)	26.56	±	0.98	21.00	±	7.16	406.40	±	4.88
(3)	49.05	±	15.75	66.80	±	26.32	700.60	±	202.05
(4)	37.50	±	6.64	57.40	±	20.45	516.80	±	61.16
(5)	41.90	±	15.61	54.00	±	30.53	606.00	±	213.85
(6)	39.38	±	10.62	58.20	±	21.10	565.20	±	125.38
(7)	39.19	±	11.38	61.20	±	23.45	548.80	±	121.78
(8)	32.92	±	7.53	54.80	±	19.83	461.40	±	72.23
(9)	40.57	±	9.07	59.60	±	21.73	577.60	±	114.48
(10)	35.63	±	6.60	57.40	±	20.40	500.80	±	86.05
mean	**38.53**	±	**9.44**	**54.90**	±	**21.21**	**543.56**	±	**108.56**

Table E.5: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, resampling boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	24.79	±	2.56	47.20	±	14.97	311.80	±	46.01
(2)	26.39	±	3.81	50.80	±	16.84	363.80	±	59.36
(3)	27.47	±	4.98	50.40	±	17.08	376.20	±	72.74
(4)	31.28	±	6.89	56.60	±	20.40	439.00	±	75.32
(5)	39.15	±	13.45	67.40	±	32.83	544.00	±	170.03
(6)	26.96	±	1.59	51.00	±	17.04	371.80	±	16.77
(7)	24.91	±	0.94	48.00	±	15.38	342.20	±	35.63
(8)	38.32	±	16.88	63.40	±	29.37	542.20	±	243.24
(9)	30.06	±	4.98	55.40	±	20.48	422.40	±	21.82
(10)	27.31	±	2.65	51.80	±	17.28	375.20	±	33.24
mean	**29.66**	±	**5.87**	**54.20**	±	**20.17**	**408.86**	±	**77.42**

Table E.6: Mono-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, resampling boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	32.64	±	5.88	51.80	±	17.64	408.60	±	26.61
(2)	25.46	±	0.73	21.00	±	0.00	396.00	±	1.22
(3)	32.69	±	8.35	54.80	±	20.54	451.80	±	88.06
(4)	31.13	±	4.15	53.00	±	18.07	426.00	±	25.53
(5)	33.97	±	7.16	54.80	±	19.80	479.80	±	88.47
(6)	35.07	±	6.79	56.60	±	20.16	499.60	±	70.92
(7)	32.97	±	7.77	54.20	±	19.77	467.60	±	81.21
(8)	31.24	±	6.14	54.40	±	19.63	435.60	±	74.76
(9)	39.52	±	14.14	62.00	±	24.73	567.20	±	188.11
(10)	30.34	±	3.92	57.40	±	20.96	411.80	±	31.95
mean	**32.50**	±	**6.50**	**52.00**	±	**18.13**	**454.40**	±	**67.69**

Table E.7: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, brick-wall boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	42.24	±	8.72	80.80	±	33.02	605.80	±	69.04
(2)	48.80	±	14.89	136.60	±	68.11	674.60	±	206.70
(3)	49.11	±	13.47	88.20	±	43.26	751.40	±	187.28
(4)	56.27	±	6.55	104.60	±	53.48	829.40	±	66.60
(5)	49.42	±	14.14	65.80	±	17.92	778.00	±	211.17
(6)	44.54	±	10.37	71.20	±	22.35	672.80	±	140.42
(7)	47.88	±	11.27	93.40	±	61.24	707.80	±	121.12
(8)	42.55	±	13.89	60.60	±	21.76	664.80	±	203.11
(9)	49.71	±	10.09	96.40	±	62.88	715.80	±	161.91
(10)	48.91	±	12.72	79.60	±	42.83	769.60	±	185.22
mean	**47.94**	±	**11.61**	**87.72**	±	**42.68**	**717.00**	±	**155.26**

Table E.8: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, random reinitialization boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
1	72.80	±	22.54	102.80	±	60.34	1009.40	±	267.31
2	63.97	±	25.41	85.00	±	66.48	976.80	±	296.19
3	71.41	±	17.85	101.20	±	65.32	1063.40	±	248.86
4	70.65	±	13.30	91.20	±	62.32	917.40	±	107.64
5	68.05	±	15.93	65.20	±	28.36	986.20	±	204.49
6	57.77	±	13.13	53.40	±	3.21	788.40	±	155.10
7	74.22	±	23.64	93.60	±	61.49	1042.20	±	276.36
8	73.88	±	19.99	68.20	±	23.51	1079.60	±	260.35
9	73.44	±	17.53	96.20	±	61.67	941.80	±	289.85
10	74.96	±	19.09	81.60	±	30.64	979.40	±	191.52
mean	**70.11**	±	**18.84**	**83.84**	±	**46.33**	**978.46**	±	**229.77**

Table E.9: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 1, resampling boundary strategy.

Structure	Time (s)			Generations			Spectrum Eval.		
(1)	66.38	±	9.74	124.40	±	60.98	828.00	±	22.10
(2)	75.06	±	7.01	174.00	±	58.14	884.00	±	178.66
(3)	61.03	±	13.05	89.00	±	64.23	877.00	±	203.76
(4)	62.61	±	12.71	97.00	±	62.29	875.80	±	90.49
(5)	60.20	±	13.89	88.20	±	65.21	865.60	±	274.44
(6)	53.13	±	14.36	88.40	±	45.10	740.80	±	175.70
(7)	52.59	±	8.05	63.00	±	11.98	801.80	±	115.56
(8)	72.23	±	14.92	106.20	±	57.66	1034.80	±	167.80
(9)	55.94	±	6.36	95.40	±	59.20	788.00	±	224.21
(10)	65.44	±	11.61	106.20	±	56.44	930.40	±	93.24
mean	**62.46**	±	**11.17**	**103.18**	±	**54.12**	**862.62**	±	**154.60**

Table E.10: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for a non-ideal twin-structure with different boundary strategies – approach 2.

Strategy	Time (s)			Generations			Spectrum Eval.		
Bounceback	50.37	±	14.21	57.80	±	18.14	820.00	±	245.56
Brickwall	47.70	±	9.25	72.00	±	18.67	734.80	±	132.85
Random reinit.	46.74	±	8.94	50.60	±	14.17	747.80	±	144.38
Resampling	55.99	±	13.05	81.40	±	39.96	841.40	±	156.29

Table E.11: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, bounce-back boundary strategy.

Structure	Time			Generations			Spectrum Eval.		
1	61.41	±	12.54	173.20	±	59.93	625.20	±	135.14
2	49.53	±	10.14	172.00	±	62.61	496.20	±	190.85
3	58.47	±	7.00	99.20	±	57.70	846.80	±	161.46
4	54.44	±	5.77	87.00	±	63.93	802.80	±	164.05
5	65.31	±	13.82	90.00	±	62.67	998.40	±	160.16
6	46.46	±	9.32	70.00	±	18.17	699.20	±	156.70
7	39.77	±	5.59	86.20	±	64.86	541.40	±	174.11
8	48.03	±	3.50	76.00	±	69.43	716.80	±	122.09
9	61.02	±	14.65	115.20	±	60.00	856.20	±	213.22
10	55.99	±	5.96	106.20	±	53.69	793.20	±	141.09
mean	**54.04**	±	**8.83**	**107.50**	±	**57.30**	**737.62**	±	**161.89**

Table E.12: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, brick-wall boundary strategy.

Structure	Time			Generations			Spectrum Eval.		
1	45.85	±	9.07	91.40	±	41.87	580.20	±	107.36
2	54.16	±	6.01	145.40	±	59.33	710.60	±	13.65
3	41.50	±	18.00	64.80	±	28.52	620.60	±	274.28
4	50.21	±	14.62	92.80	±	63.60	697.60	±	184.07
5	47.82	±	13.89	63.80	±	18.05	729.00	±	198.16
6	54.46	±	5.28	96.40	±	20.53	798.20	±	90.56
7	44.49	±	7.95	91.80	±	62.07	622.80	±	69.94
8	51.71	±	24.07	72.20	±	31.48	795.80	±	342.20
9	59.31	±	11.51	124.40	±	50.08	808.00	±	268.30
10	49.30	±	7.79	88.40	±	23.67	723.80	±	114.62
mean	**49.88**	±	**11.82**	**93.14**	±	**39.92**	**708.66**	±	**166.31**

Table E.13: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, random reinitialization boundary strategy.

Structure	Time			Generations			Spectrum Eval.		
1	68.31	±	20.19	74.00	±	18.33	970.80	±	221.83
2	79.14	±	26.33	109.80	±	50.54	1126.80	±	331.97
3	71.36	±	17.14	95.80	±	60.25	1045.60	±	301.57
4	72.70	±	29.81	93.40	±	63.18	1072.40	±	391.26
5	55.60	±	20.93	56.20	±	28.49	860.40	±	308.58
6	52.66	±	9.30	58.20	±	13.10	798.00	±	139.04
7	67.10	±	32.29	86.40	±	66.36	992.80	±	414.41
8	68.95	±	21.22	91.00	±	63.89	1004.80	±	250.58
9	67.23	±	9.13	92.00	±	60.93	985.60	±	184.54
10	60.26	±	9.76	79.40	±	20.71	898.60	±	140.71
mean	**66.33**	±	**19.61**	**83.62**	±	**44.58**	**975.58**	±	**268.45**

Table E.14: Multi-objective tuning mean and standard deviation simulation results of convergence time, number of generations, and spectrum evaluations for (1) to (10) non-ideal twin-accelerometers – approach 2, resampling boundary strategy.

Structure	Time			Generations			Spectrum Eval.		
1	60.62	±	11.96	95.20	±	63.08	779.00	±	140.10
2	69.98	±	12.83	169.60	±	67.98	807.80	±	154.76
3	64.65	±	10.49	99.80	±	39.26	901.40	±	142.54
4	61.90	±	7.90	95.80	±	58.48	845.00	±	230.68
5	59.87	±	15.26	60.40	±	22.43	894.80	±	213.62
6	51.21	±	6.69	58.20	±	14.38	768.80	±	97.22
7	47.94	±	10.19	83.80	±	66.02	635.60	±	115.15
8	64.50	±	6.54	88.80	±	62.55	861.80	±	91.91
9	52.38	±	9.75	98.60	±	65.30	672.40	±	143.84
10	54.44	±	5.17	73.40	±	25.29	775.40	±	74.86
mean	**58.75**	±	**9.68**	**92.36**	±	**48.48**	**794.20**	±	**140.47**

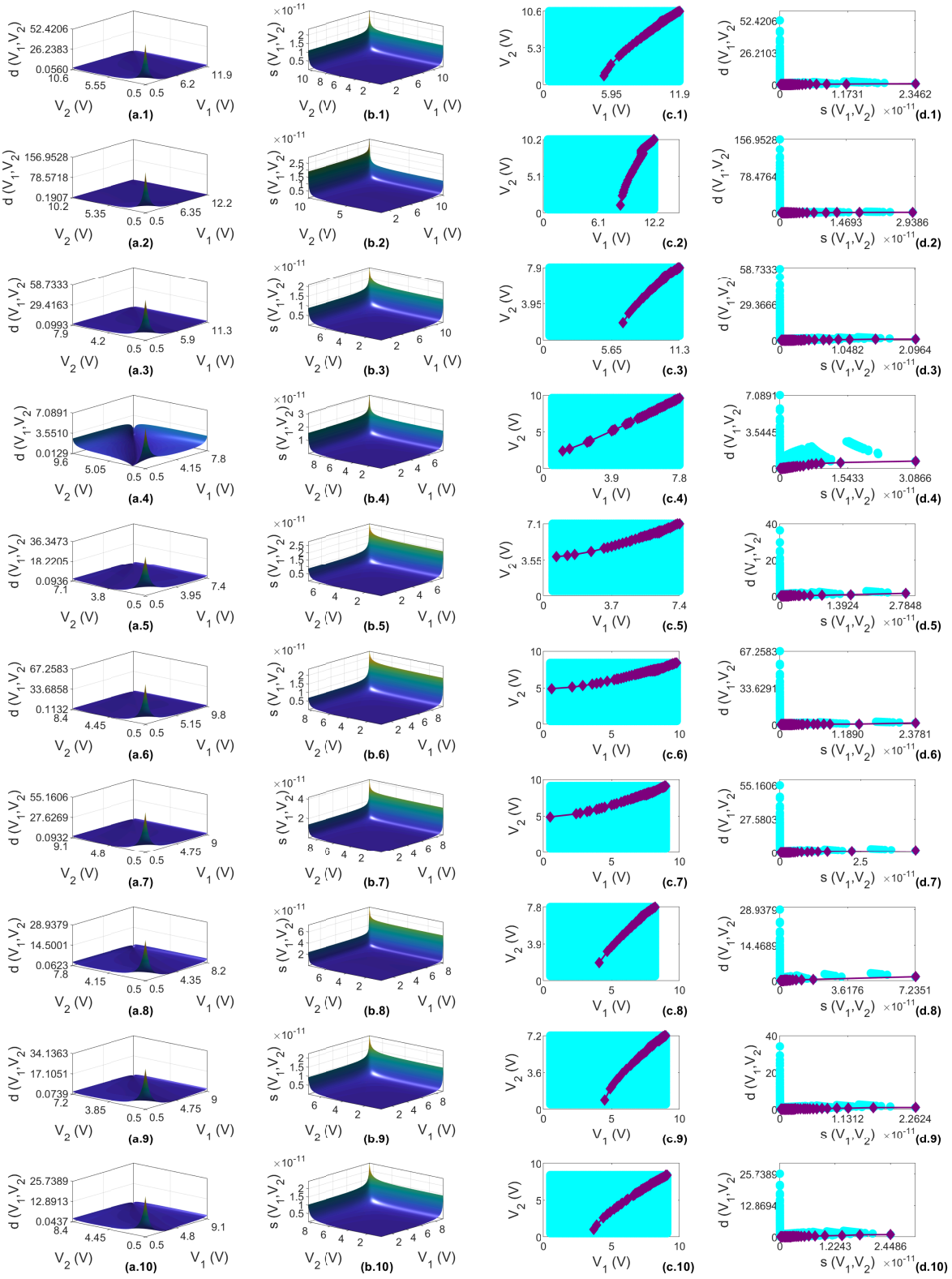

Figure E.1: Mapping results for (1) to (10) non-ideal twin-accelerometers. (a) Mapped distortion. (b) Mapped sensitivity. (c) Mapped actuation voltages amplitudes V_1 and V_2 (cyan) and Pareto-set (purple) in the decision space. (d) Mapped dominated (cyan) and non-dominated (purple) solutions in the objective space for the MEMS spectrum analyzer.

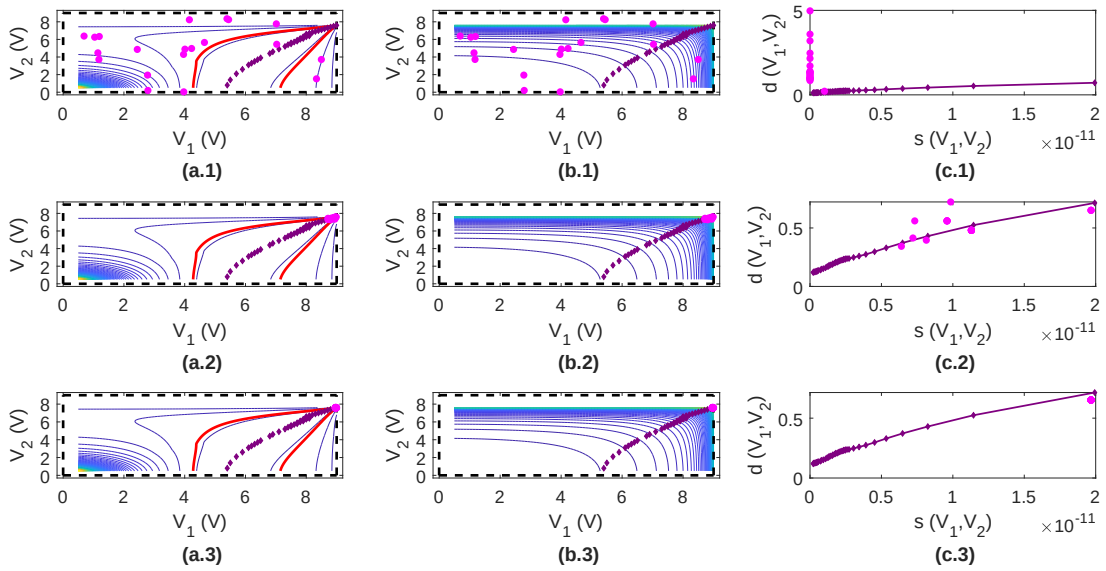

Figure E.2: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

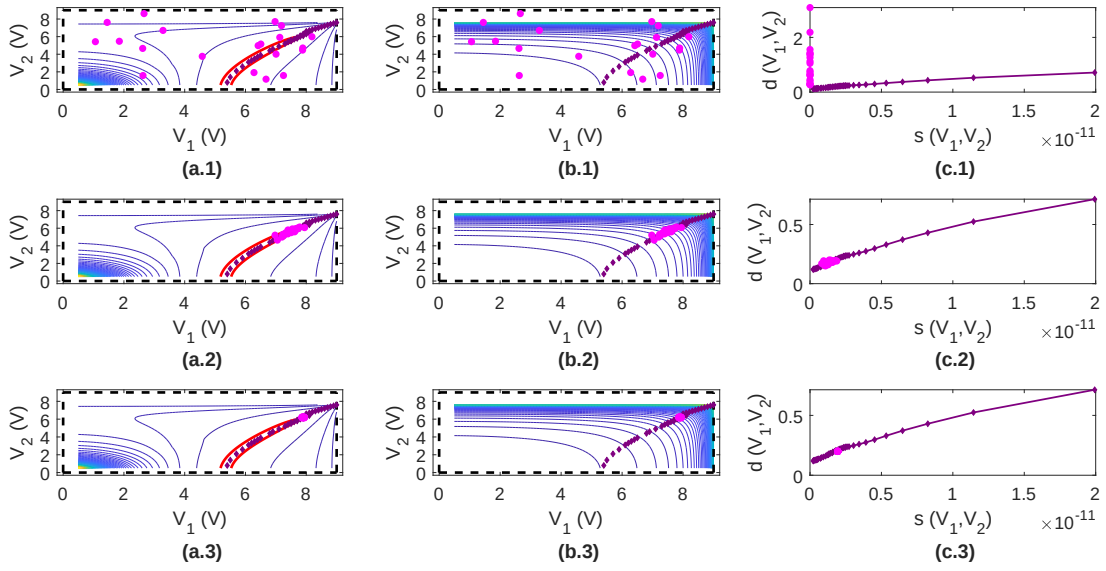

Figure E.3: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

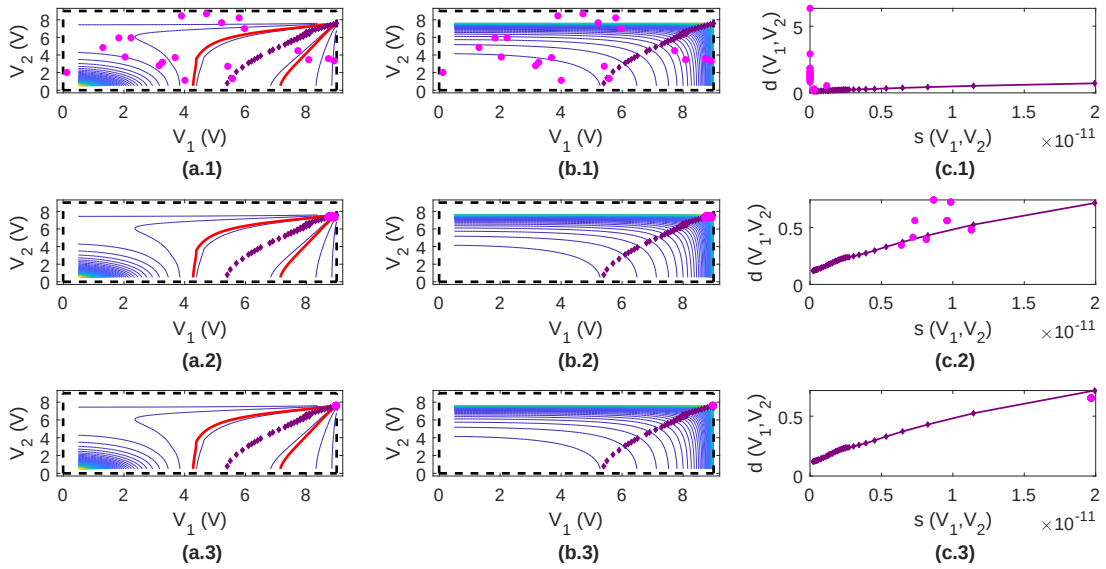

Figure E.4: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

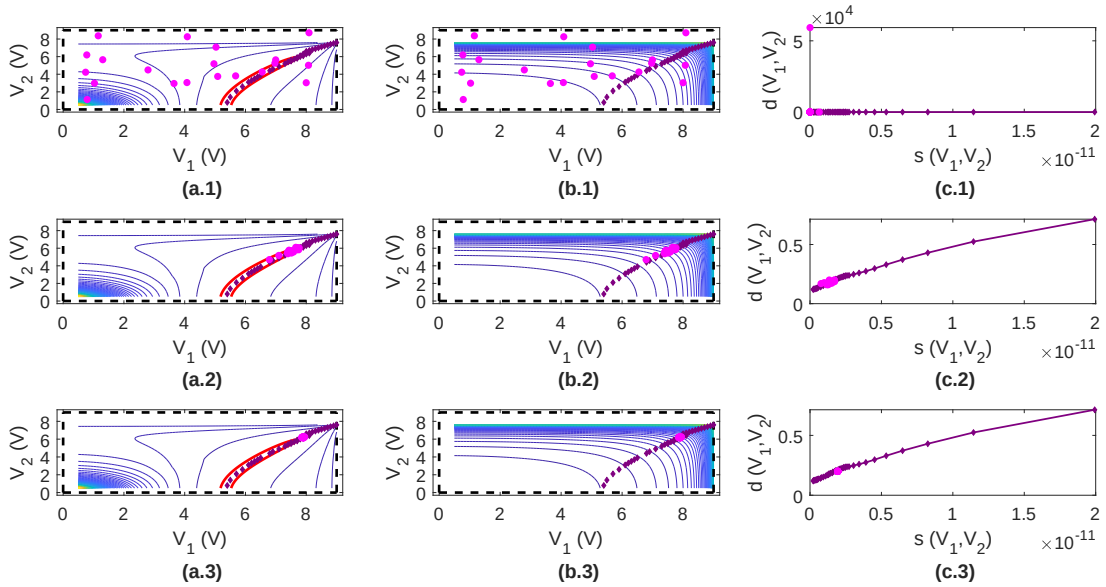

Figure E.5: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

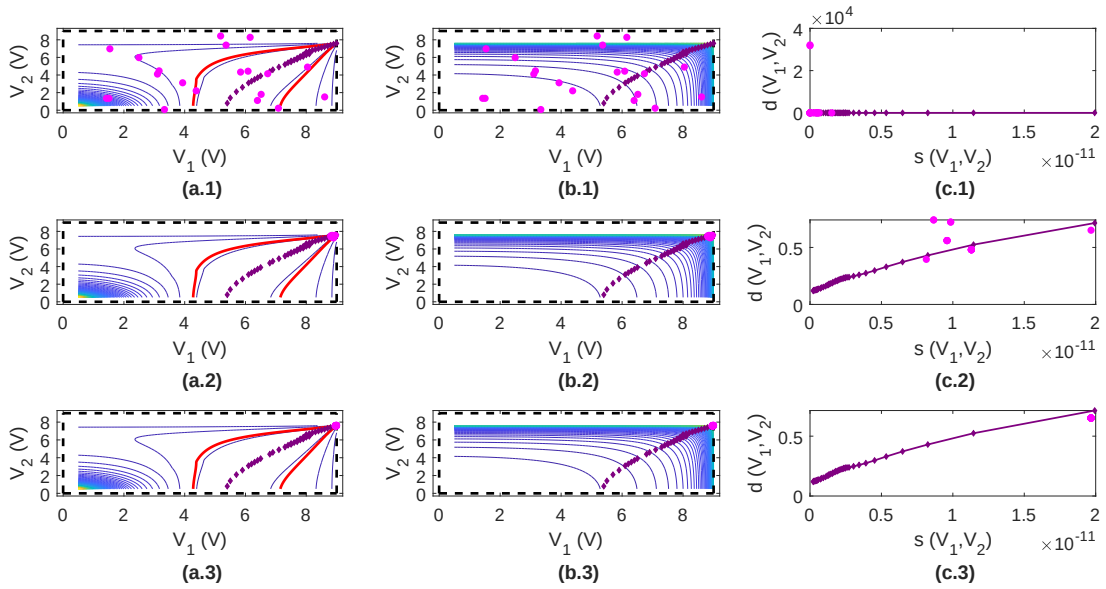

Figure E.6: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.8, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

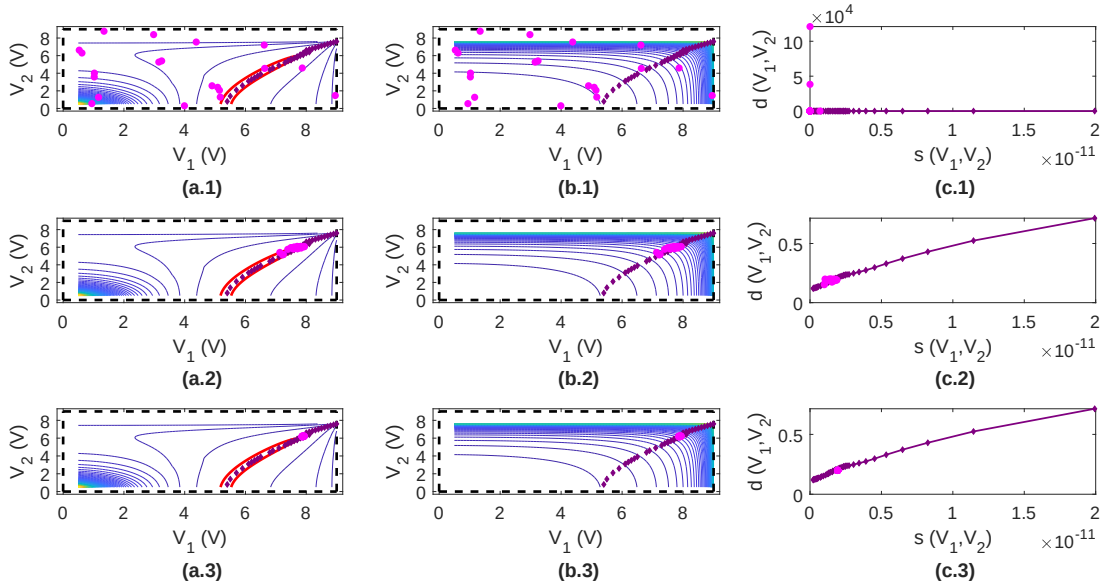

Figure E.7: Mono-objective tuning evolution for a non-ideal twin-structure – maximum distortion level 0.2, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation. (3) Last generation.

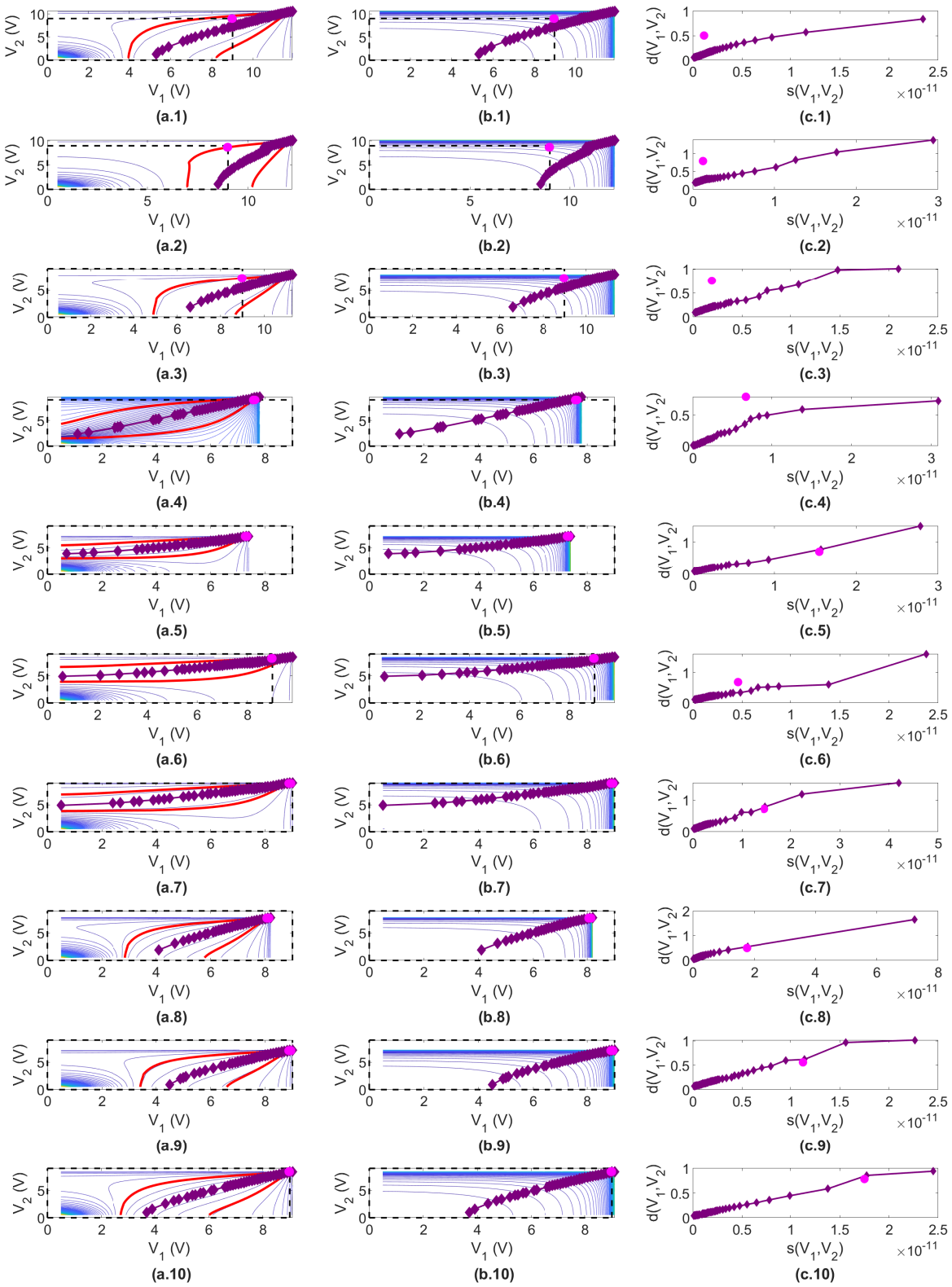

Figure E.8: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

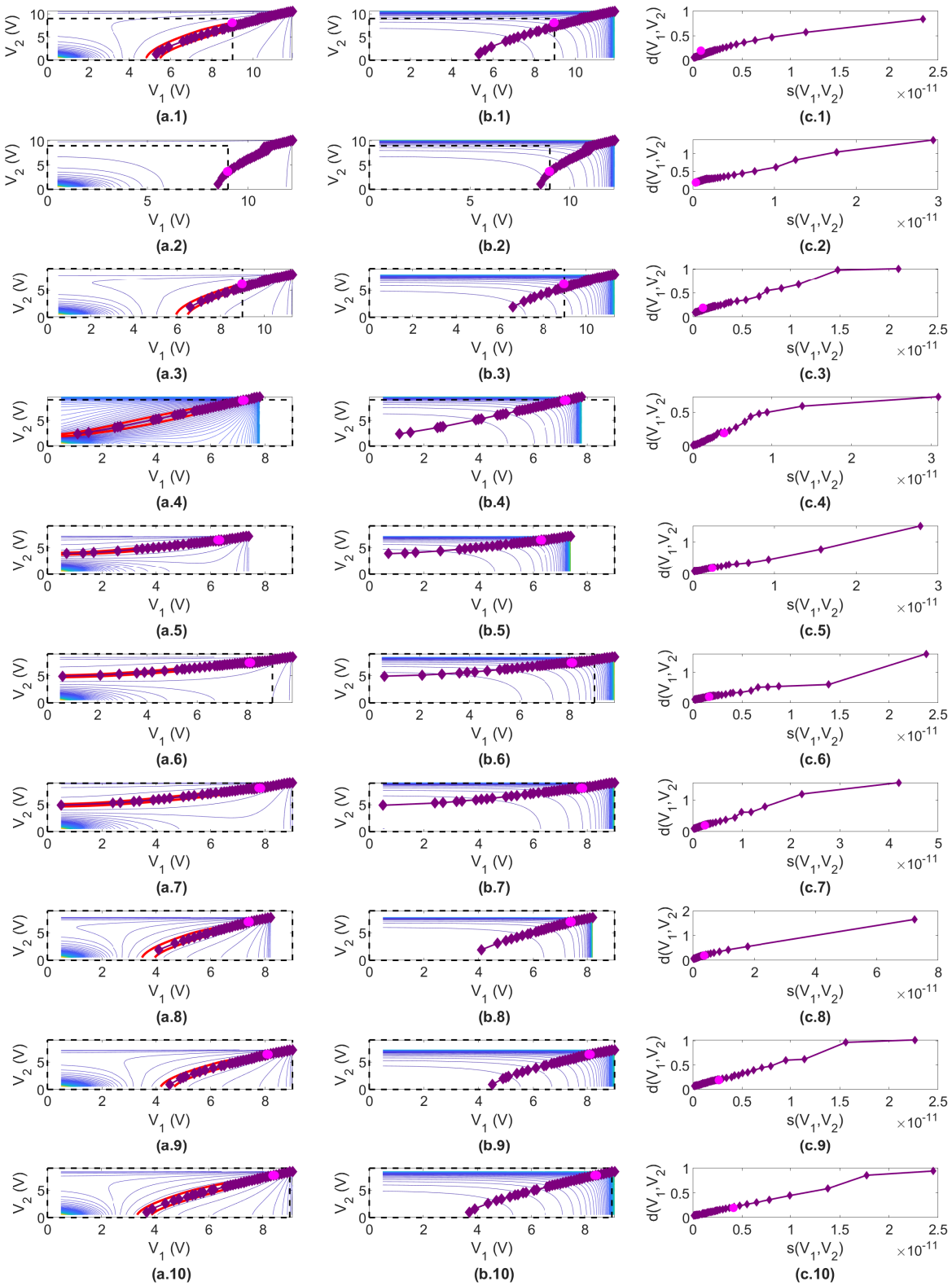

Figure E.9: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

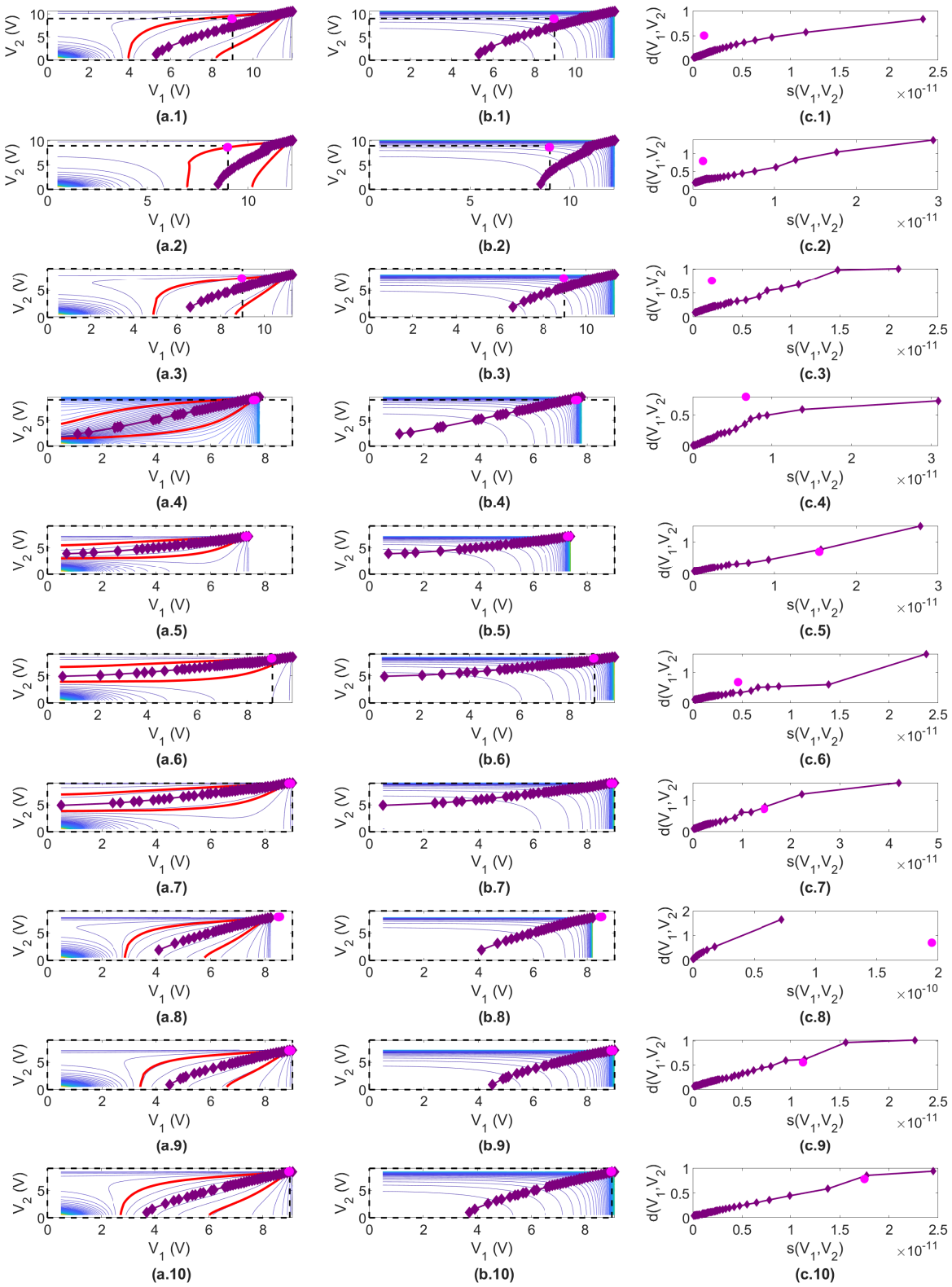

Figure E.10: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

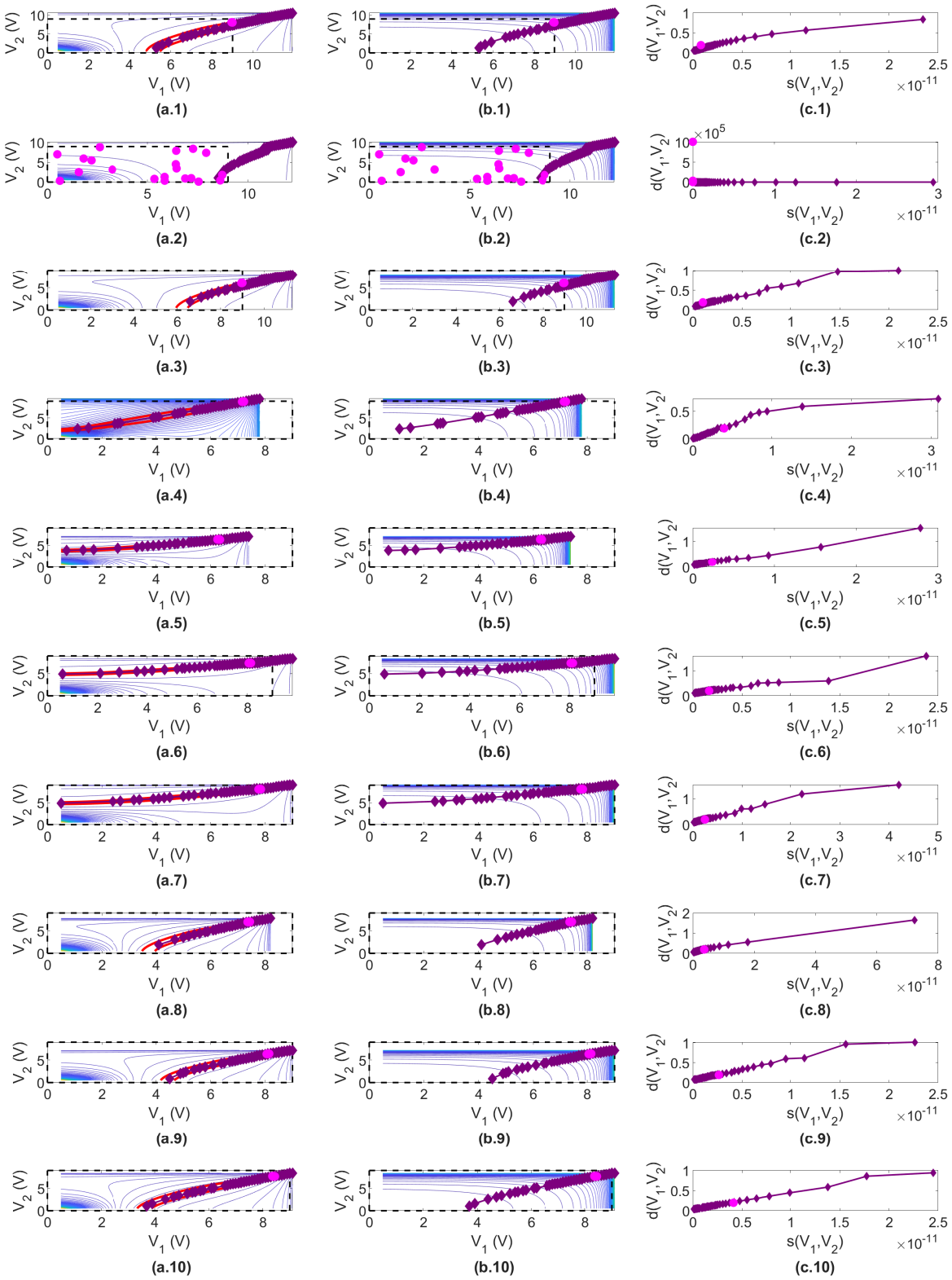

Figure E.11: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

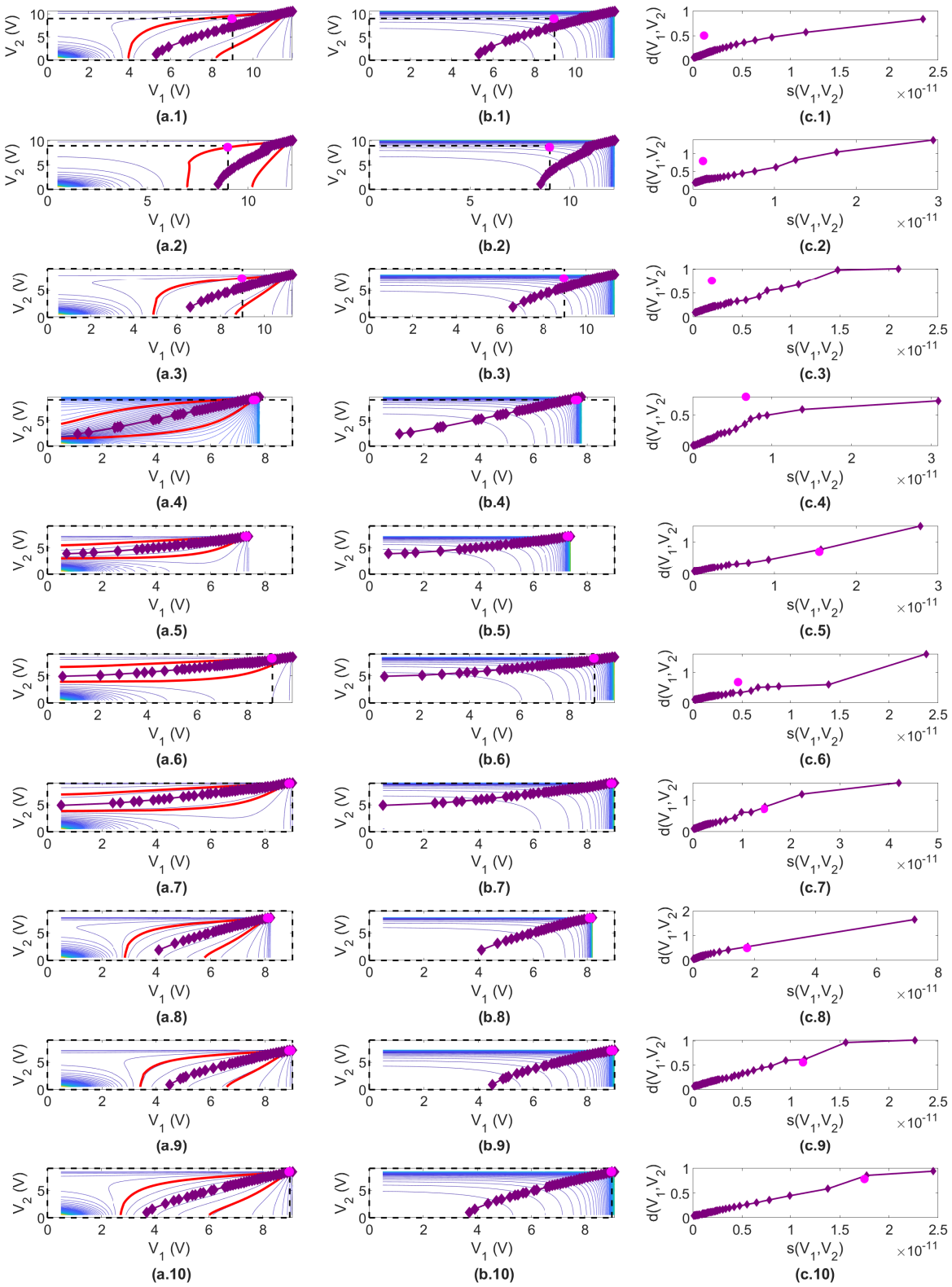

Figure E.12: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

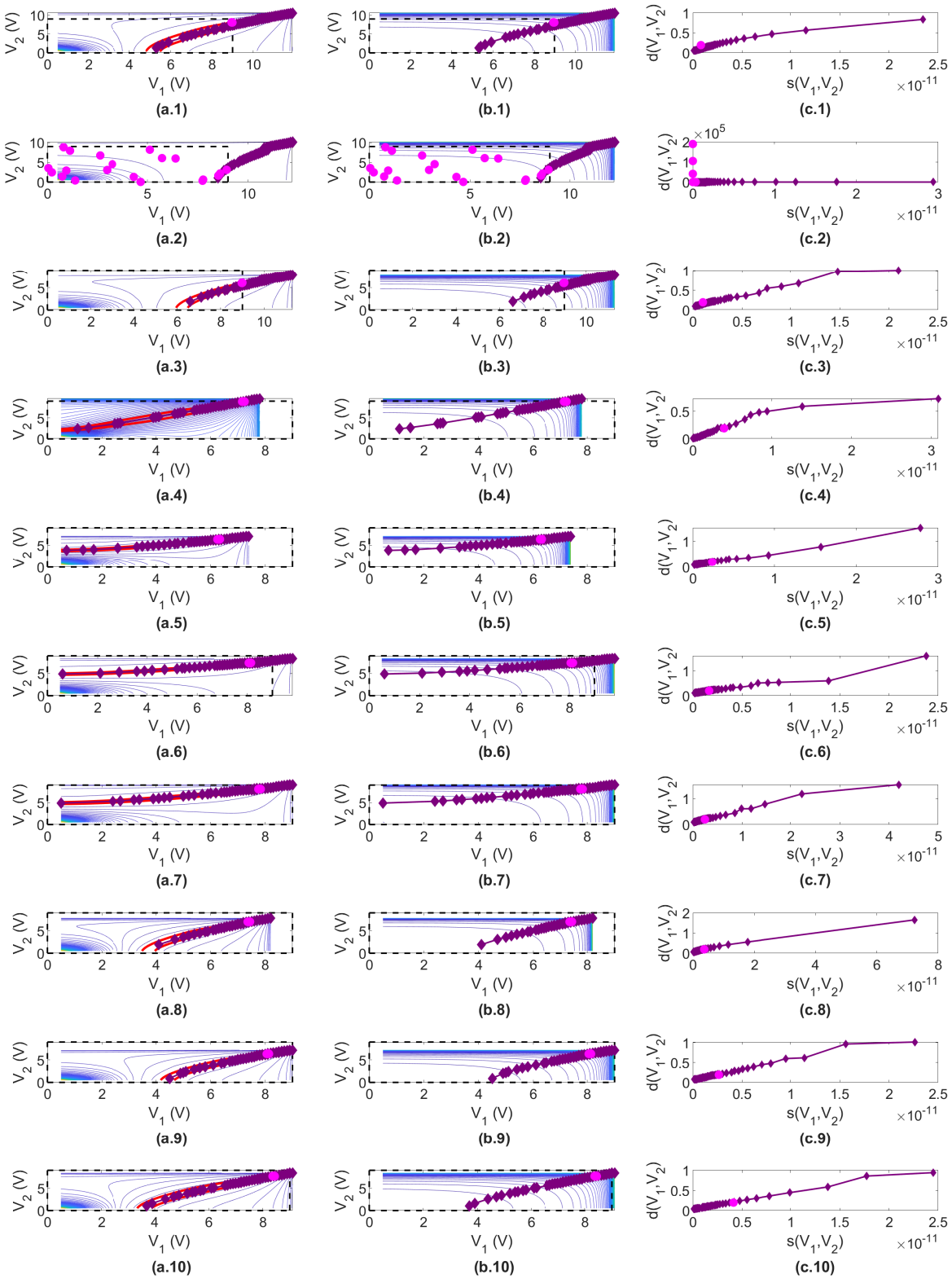

Figure E.13: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

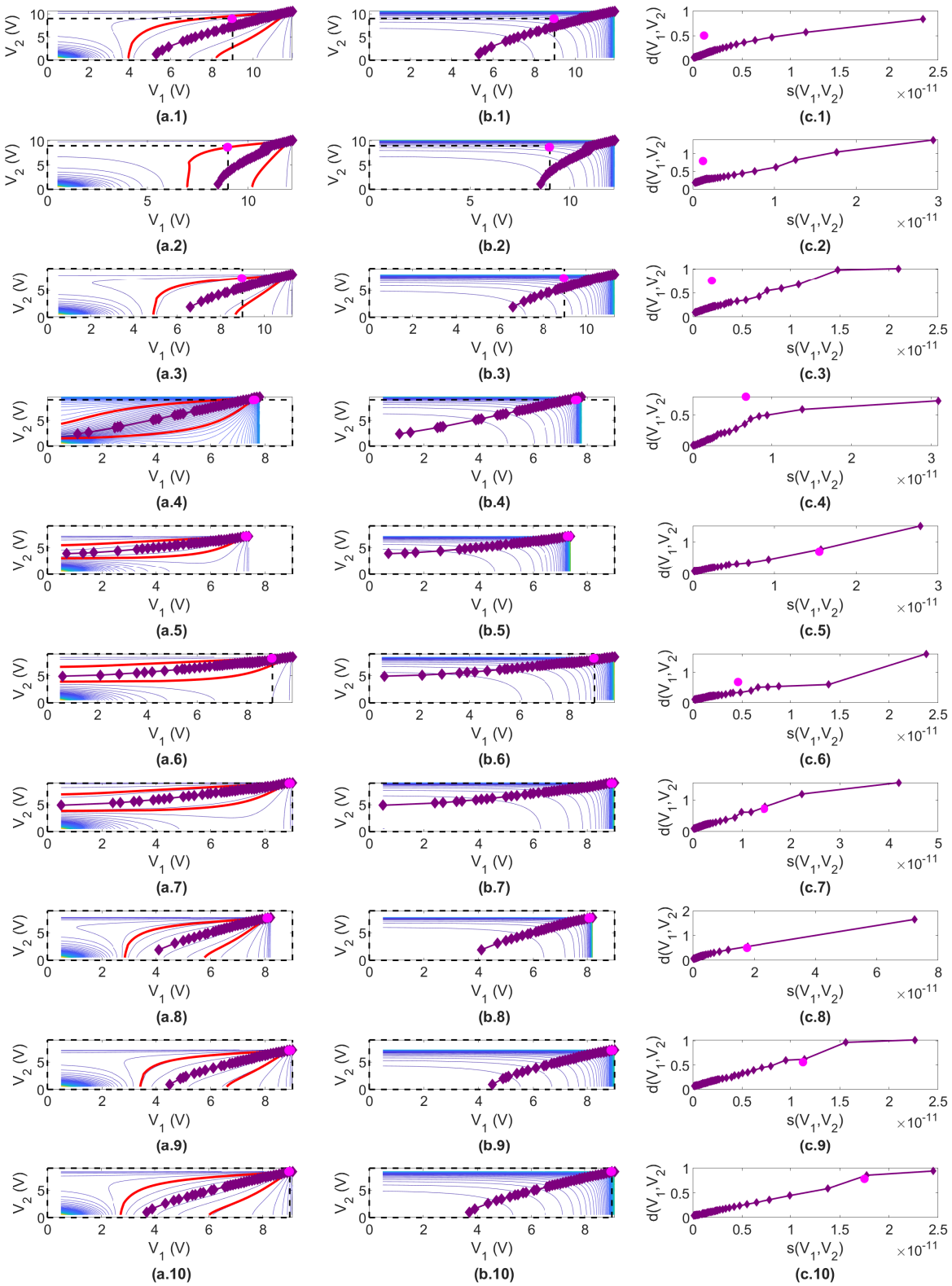

Figure E.14: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.8, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

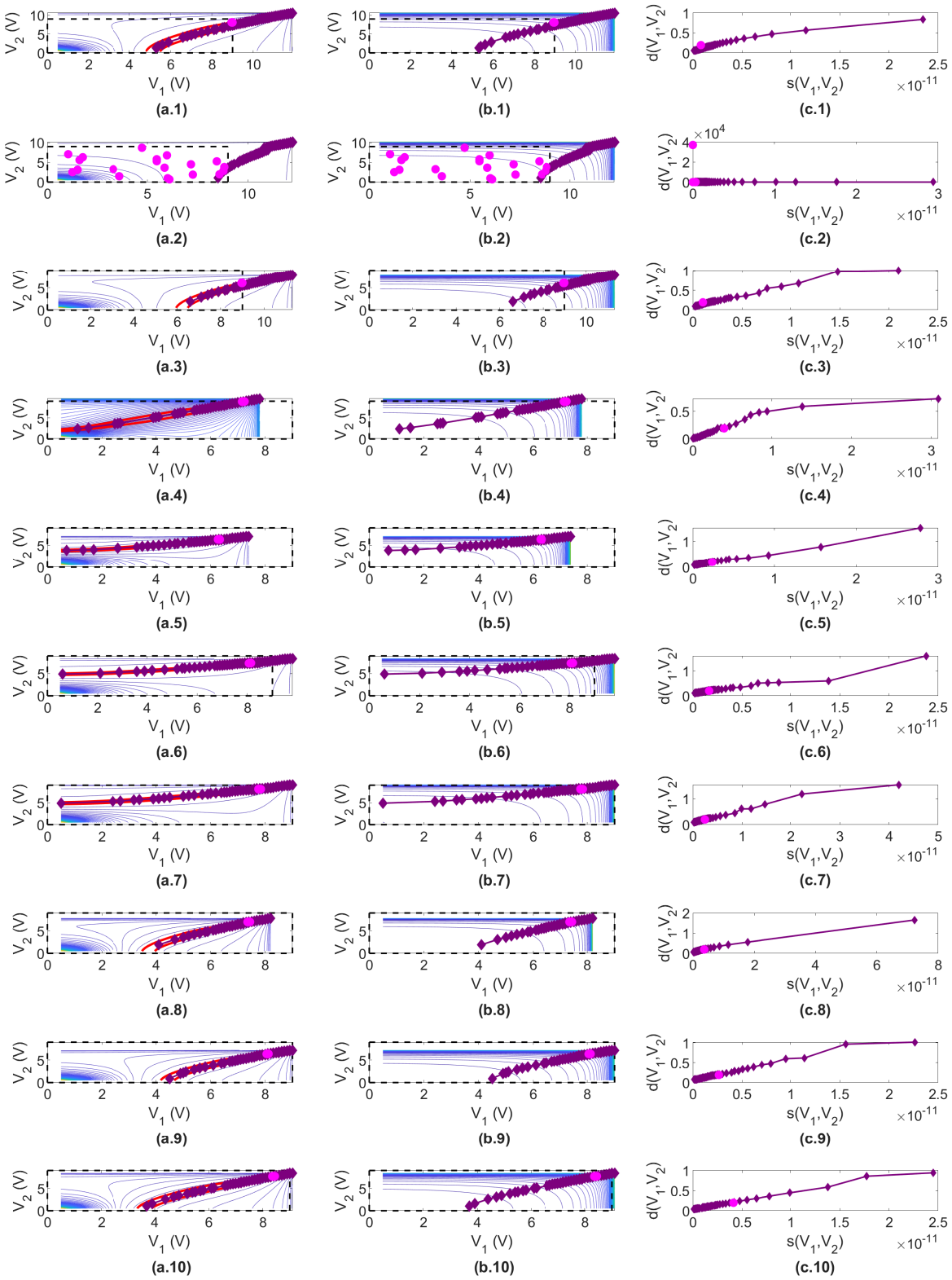

Figure E.15: Mono-objective tuning final results for (1) to (10) non-ideal twin-accelerometers – maximum distortion level 0.2, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space; in red, the distortion constraint. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

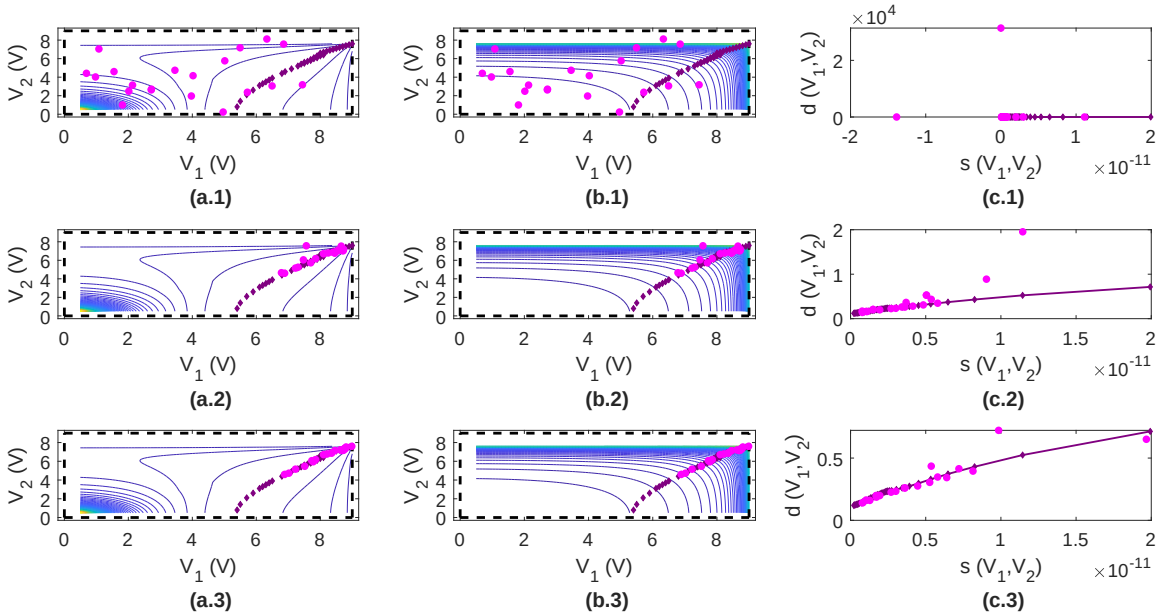

Figure E.16: Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

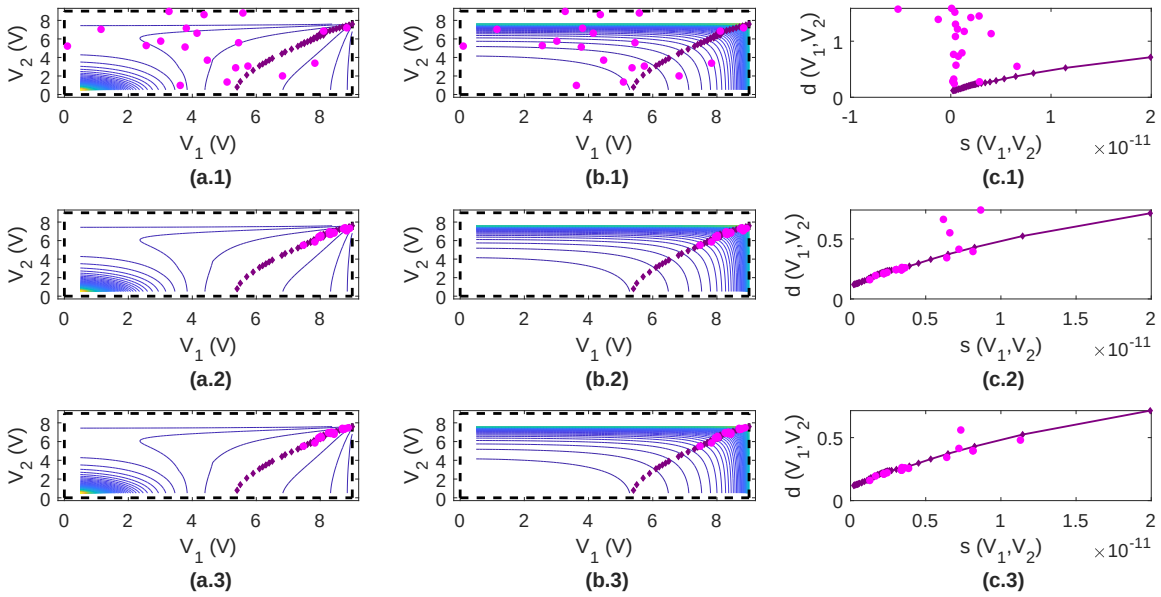

Figure E.17: Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

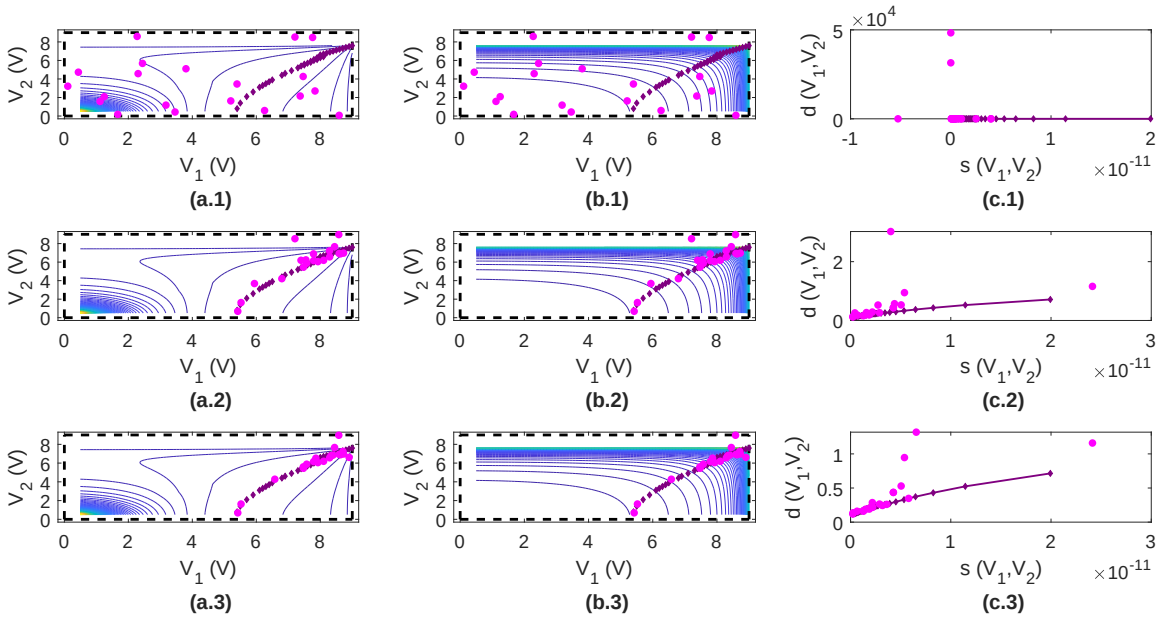

Figure E.18: Multi-objective tuning evolution for a non-ideal twin-structure – approach 1, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

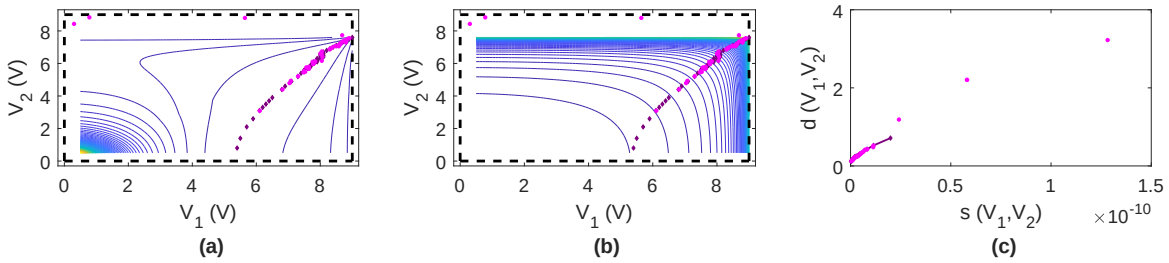

Figure E.19: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

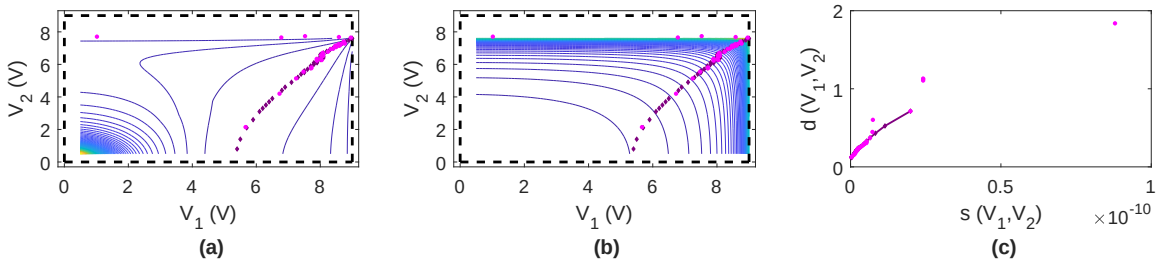

Figure E.20: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

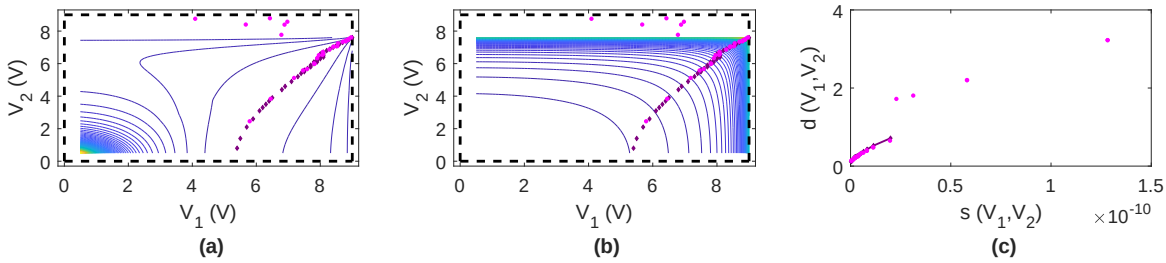

Figure E.21: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 1, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

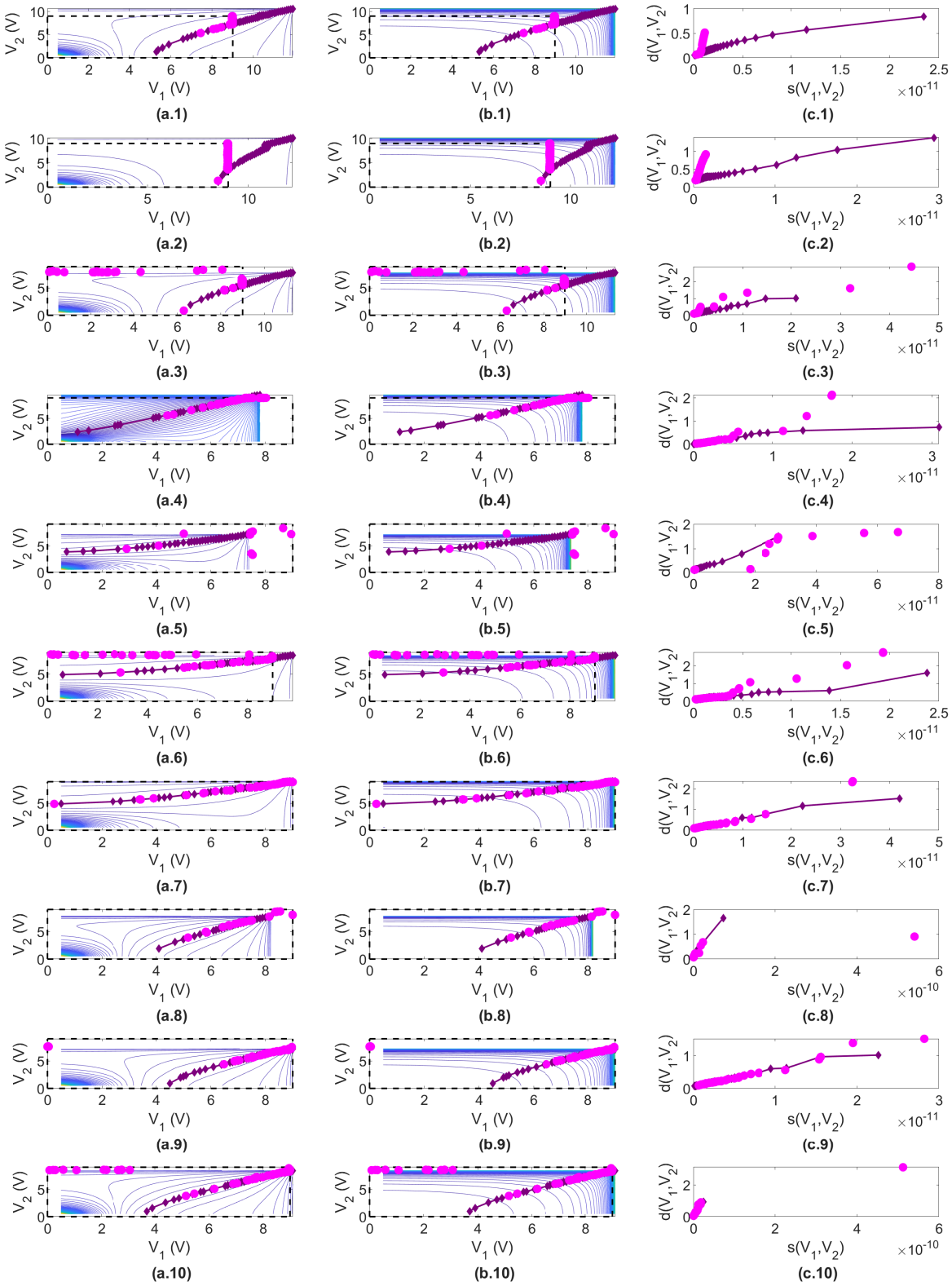

Figure E.22: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

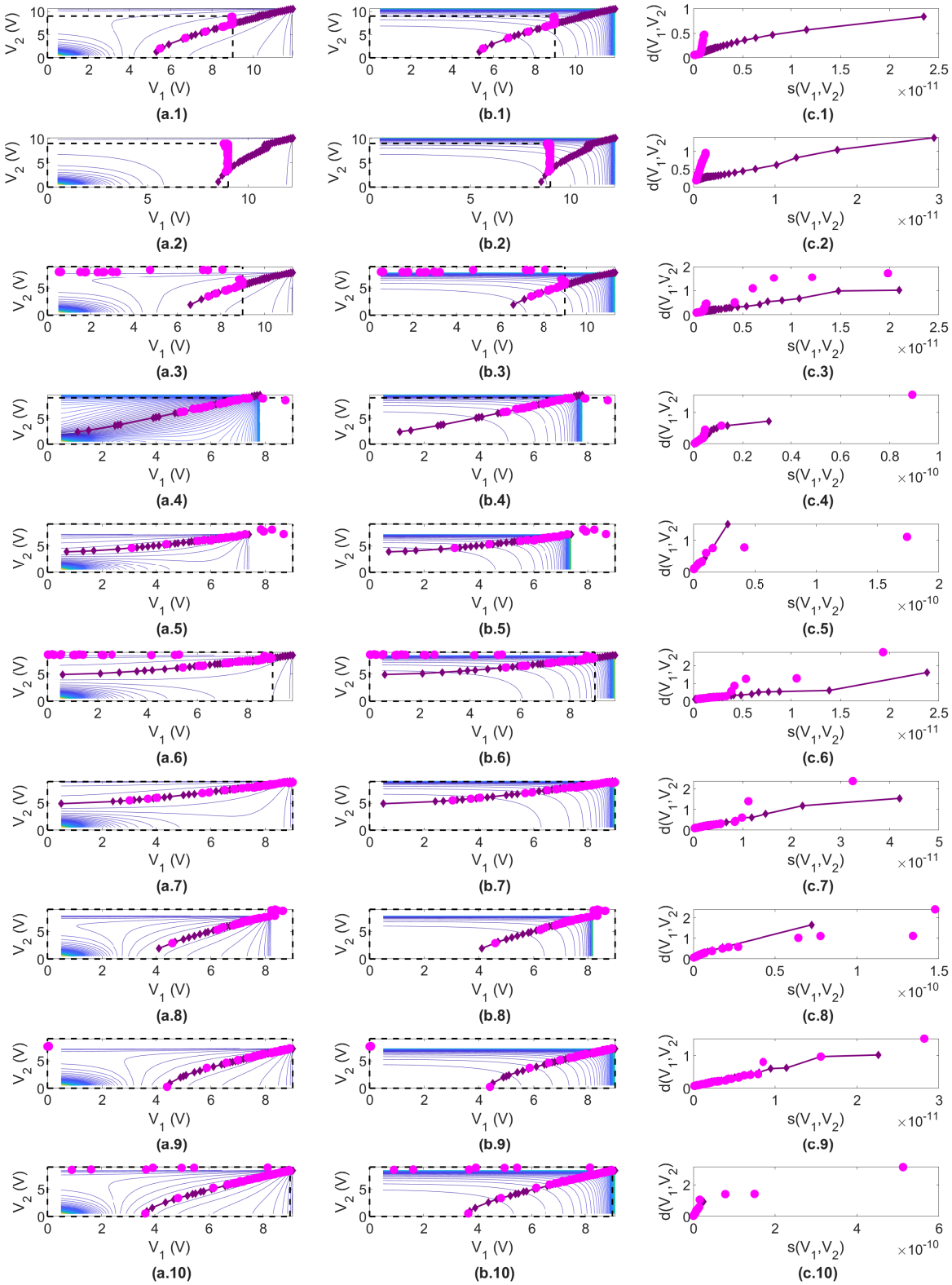

Figure E.23: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

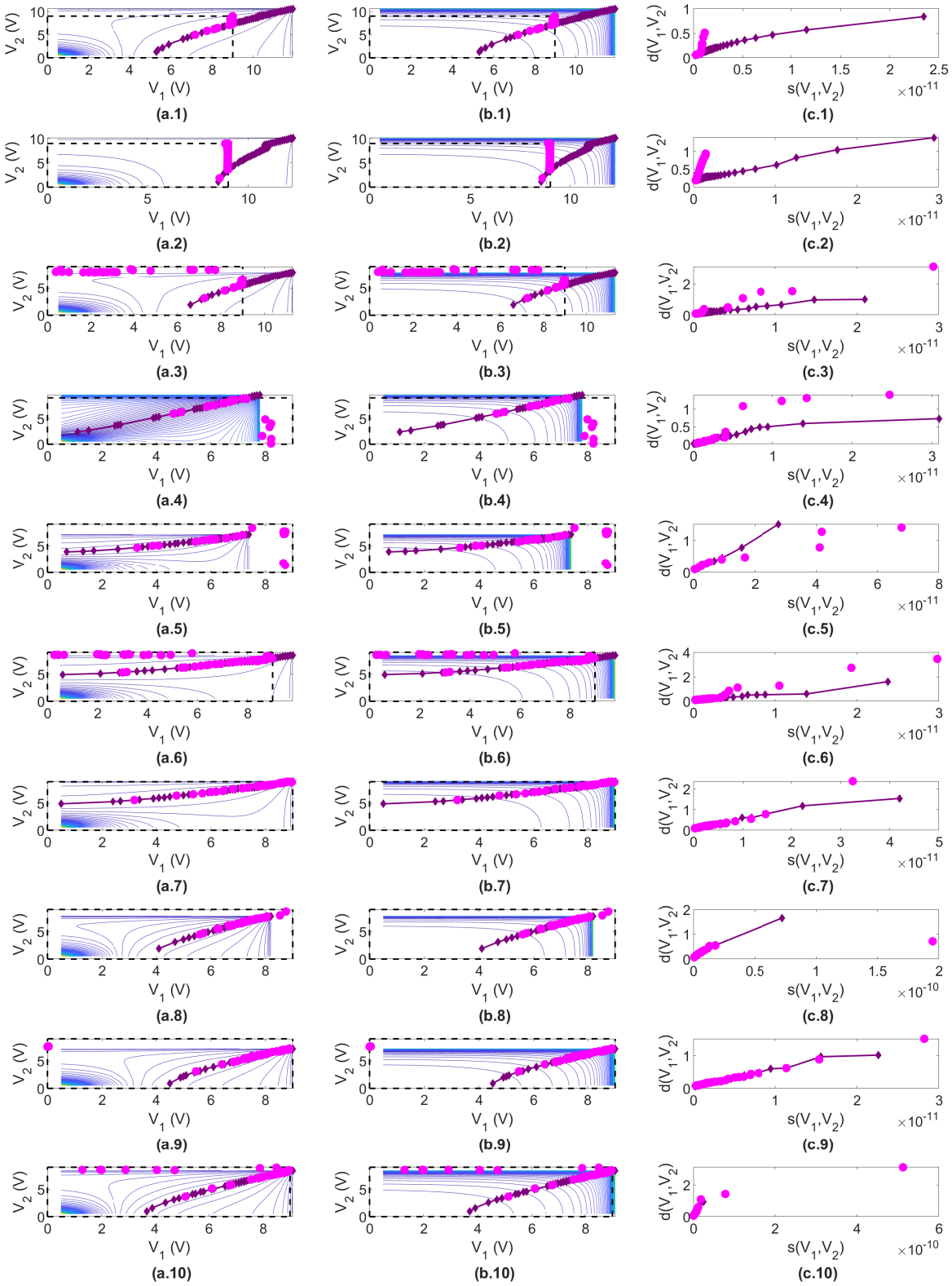

Figure E.24: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

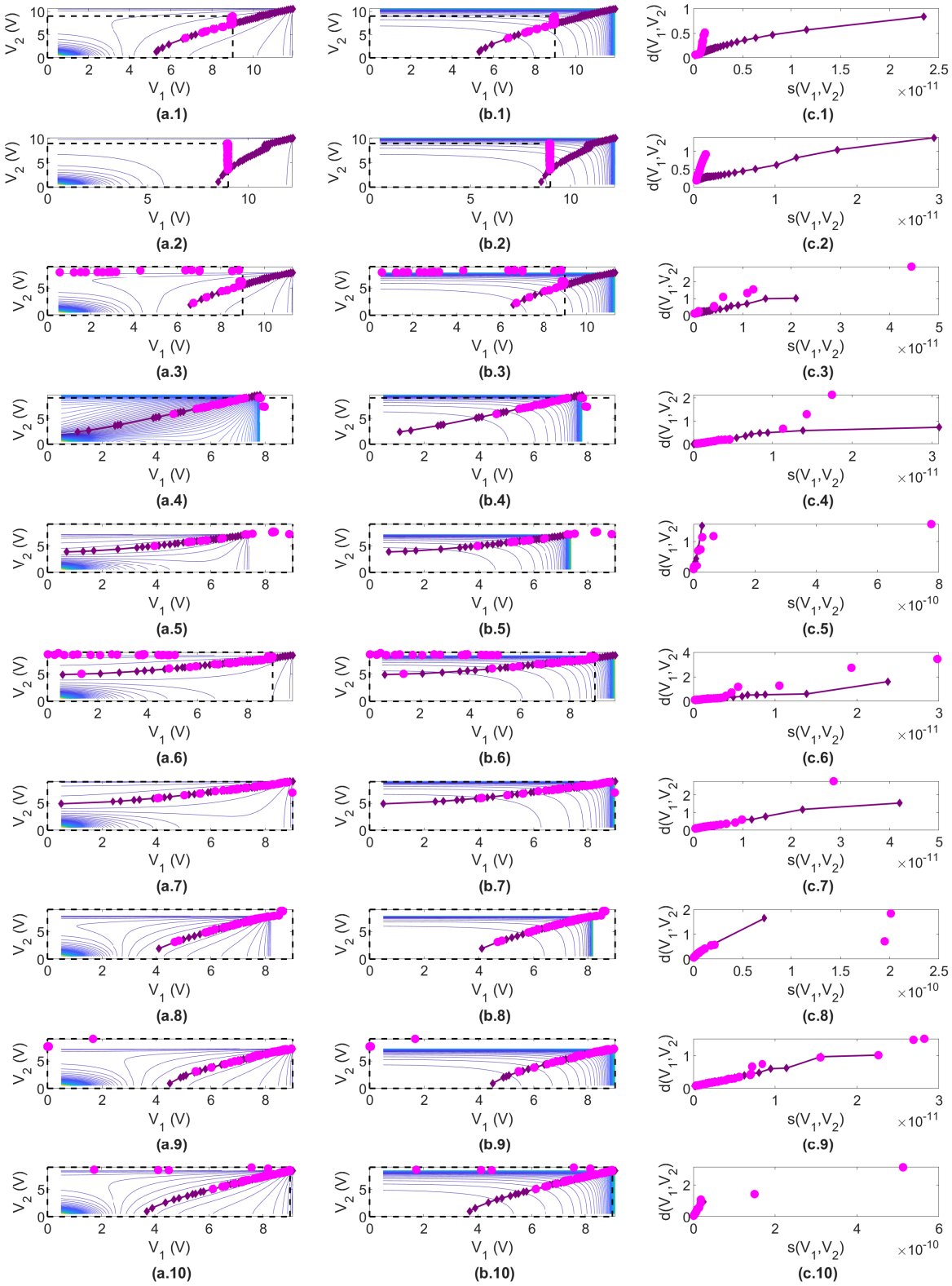

Figure E.25: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 1, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

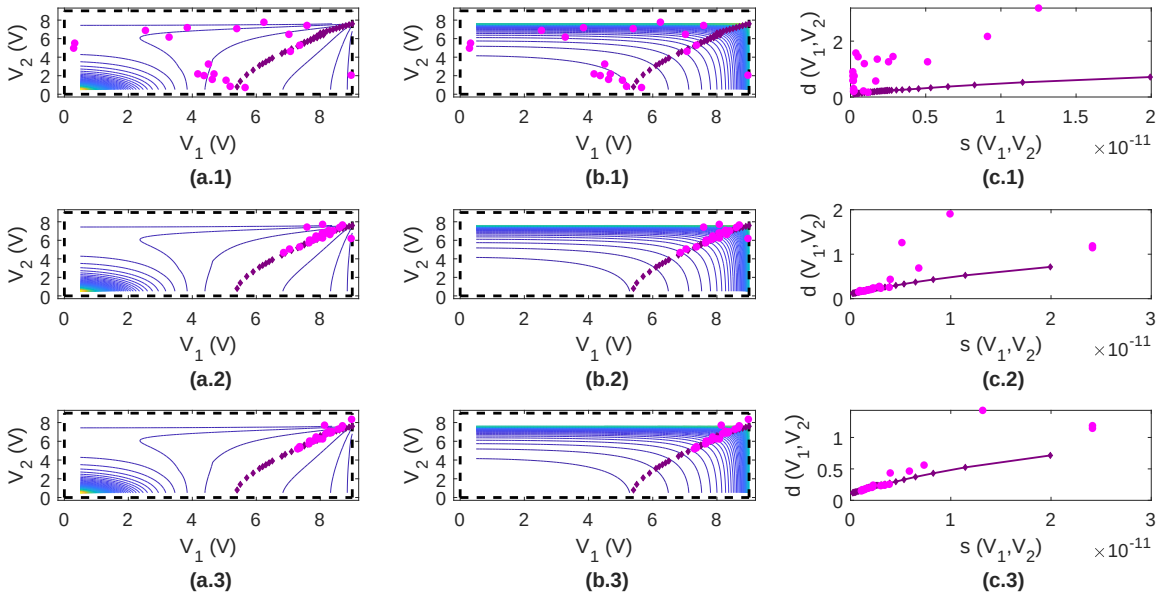

Figure E.26: Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

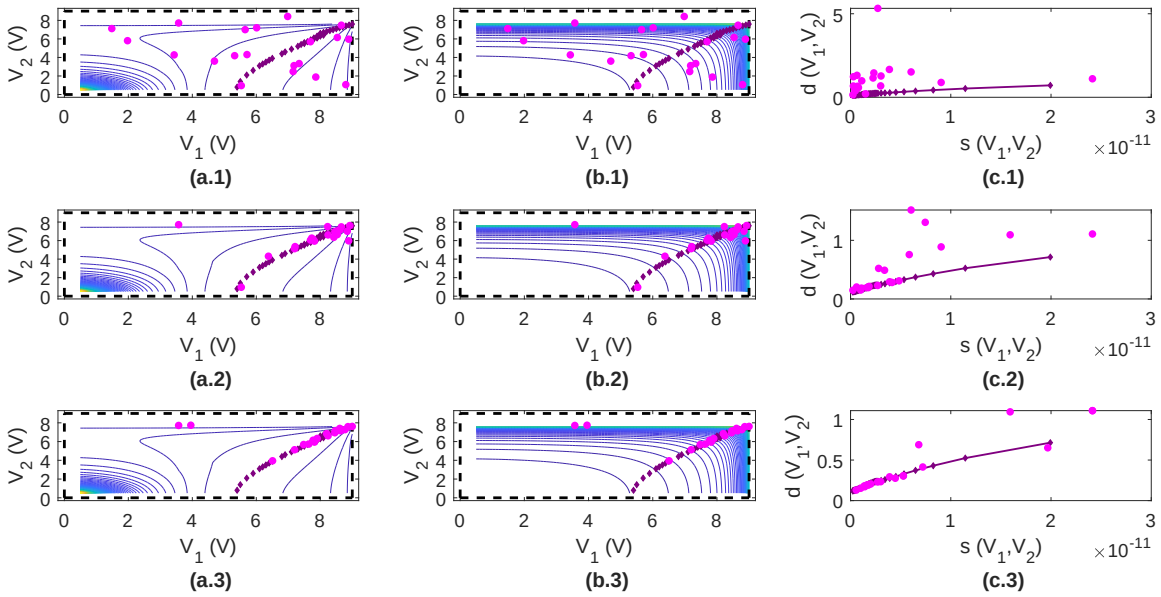

Figure E.27: Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

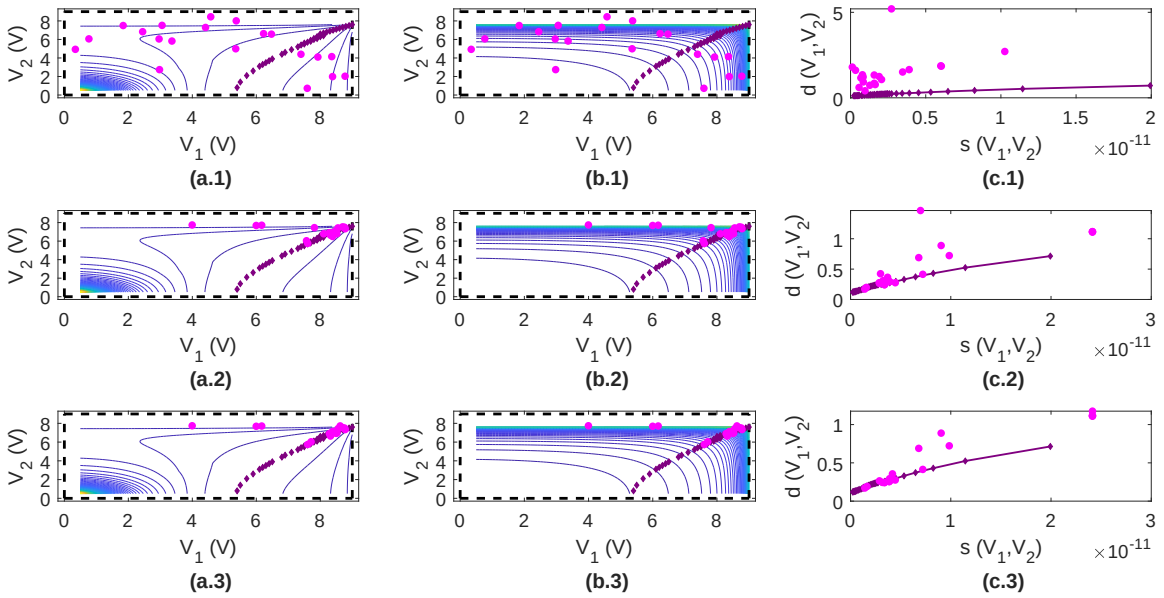

Figure E.28: Multi-objective tuning evolution for a non-ideal twin-structure – approach 2, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space. (1) First generation. (2) Intermediate generation (3) Last generation.

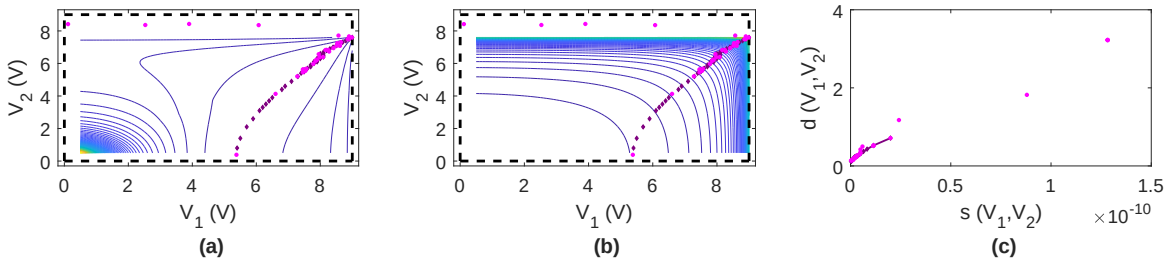

Figure E.29: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

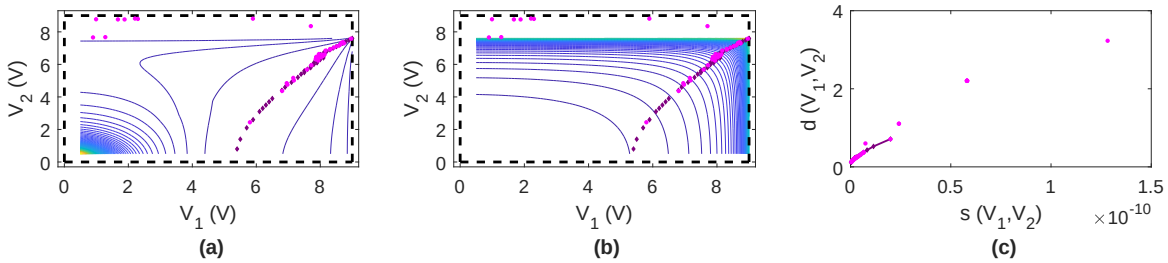

Figure E.30: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

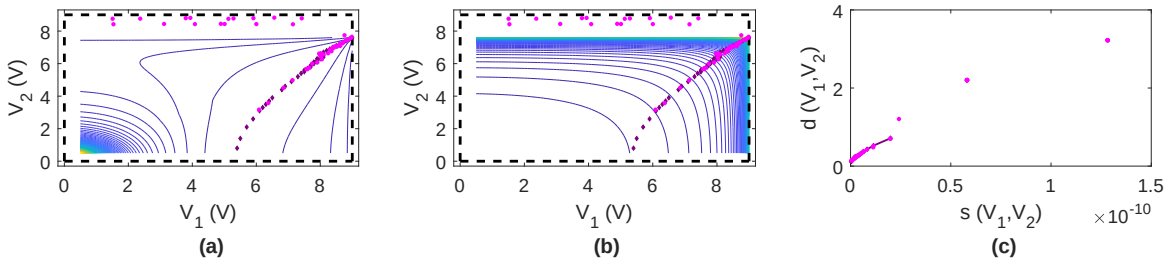

Figure E.31: Multi-objective tuning combined final results for a non-ideal twin-structure – approach 2, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

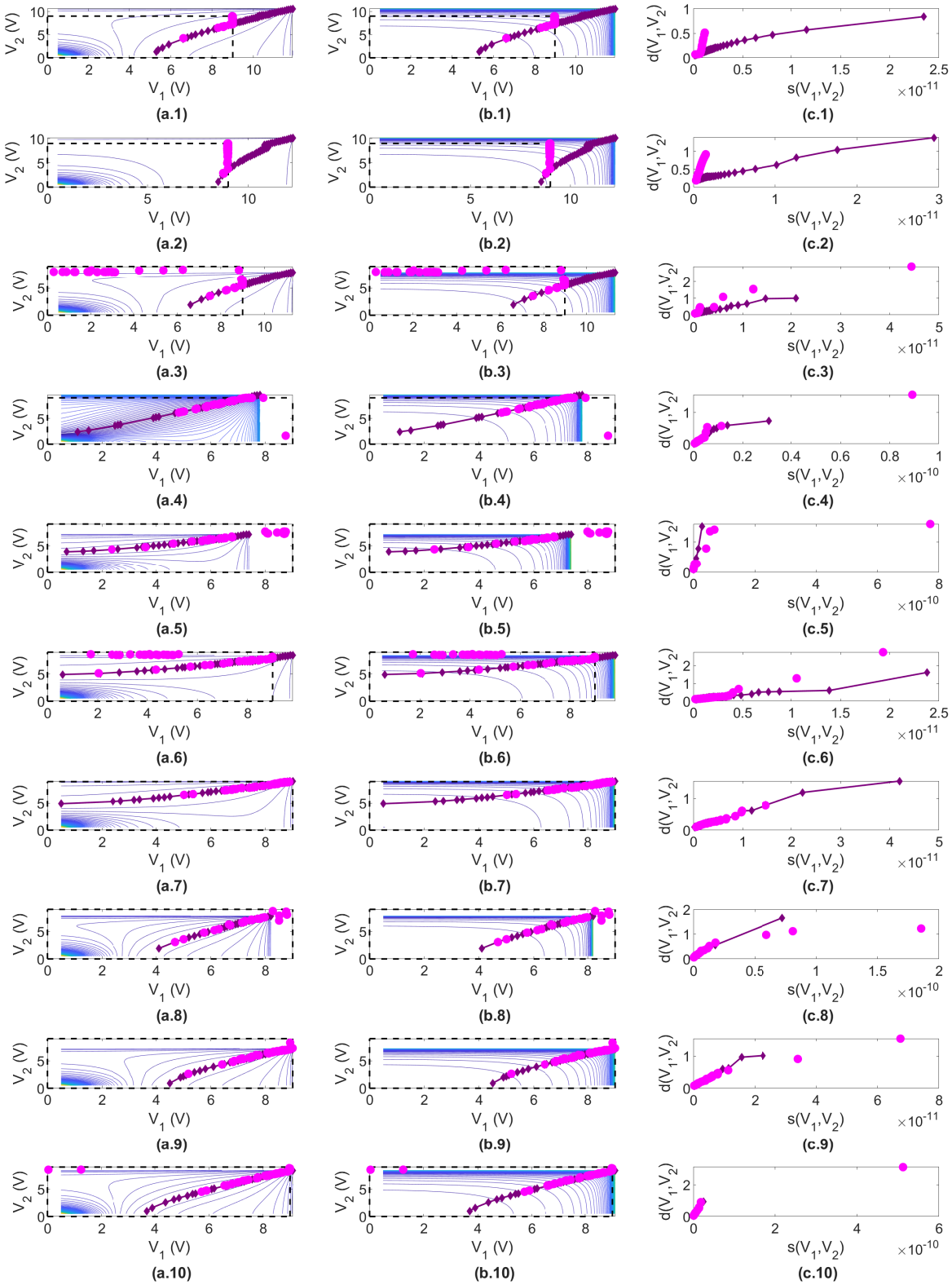

Figure E.32: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, bounce-back boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

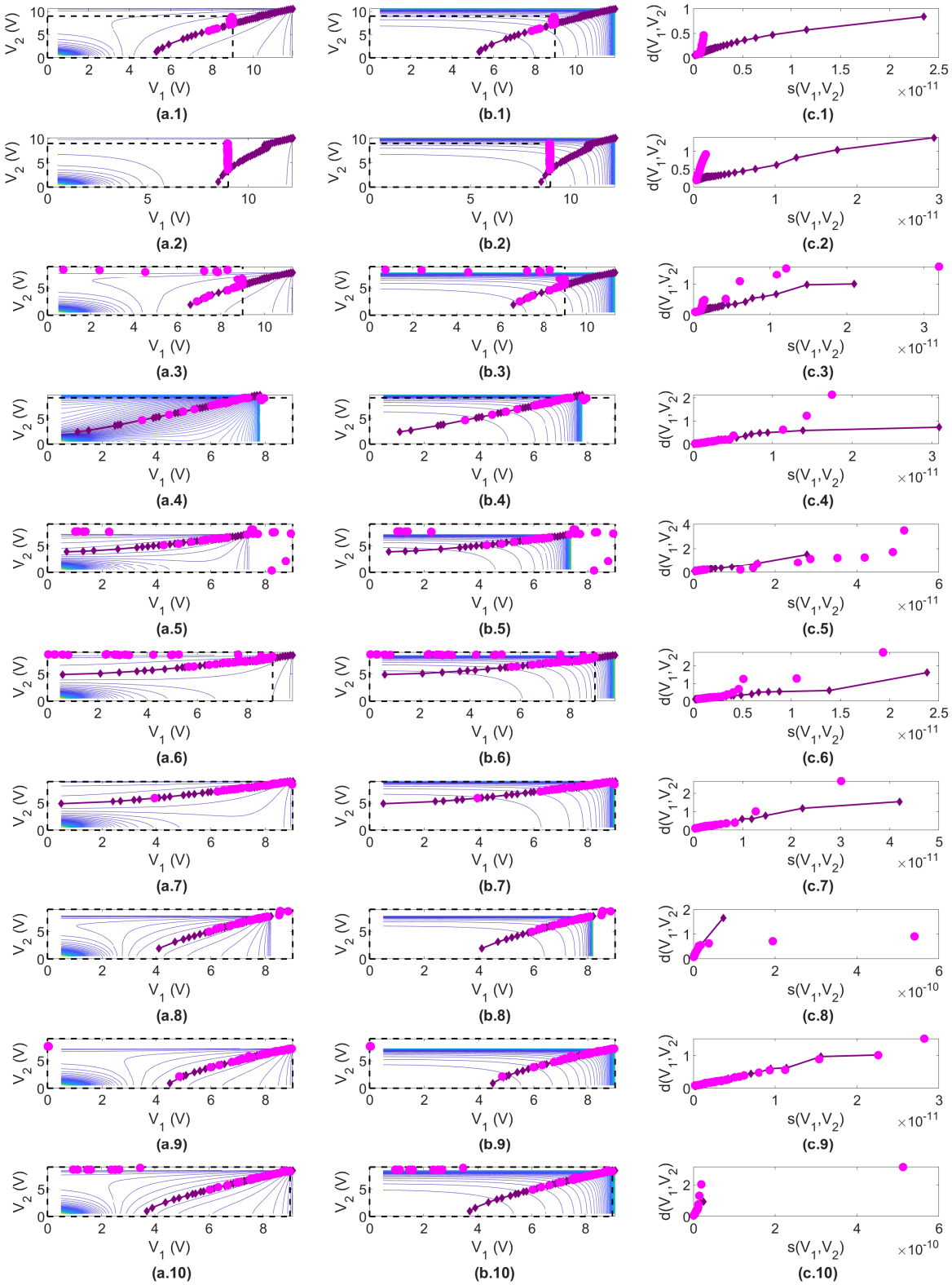

Figure E.33: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, brick-wall boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

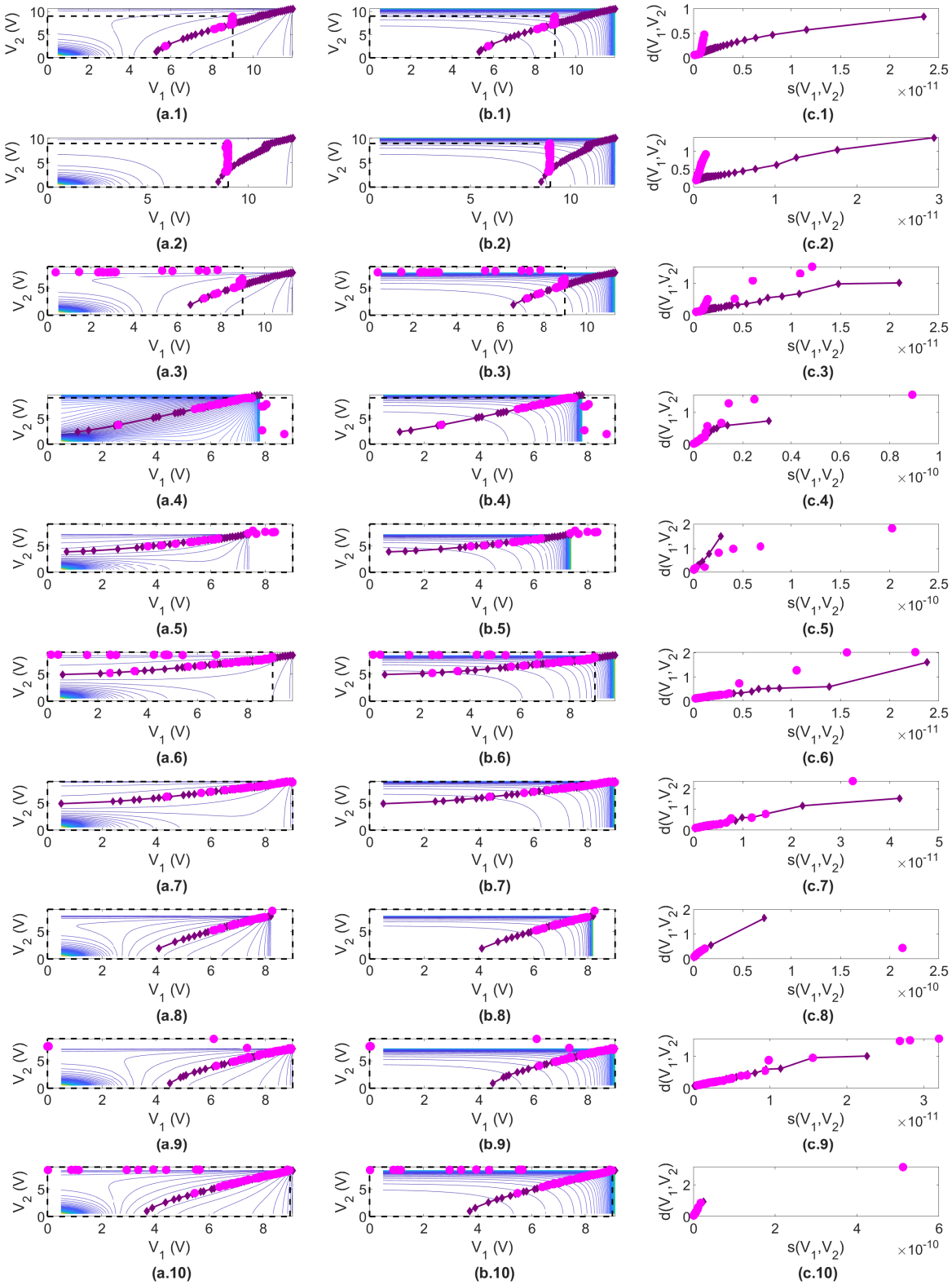

Figure E.34: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, random reinitialization boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

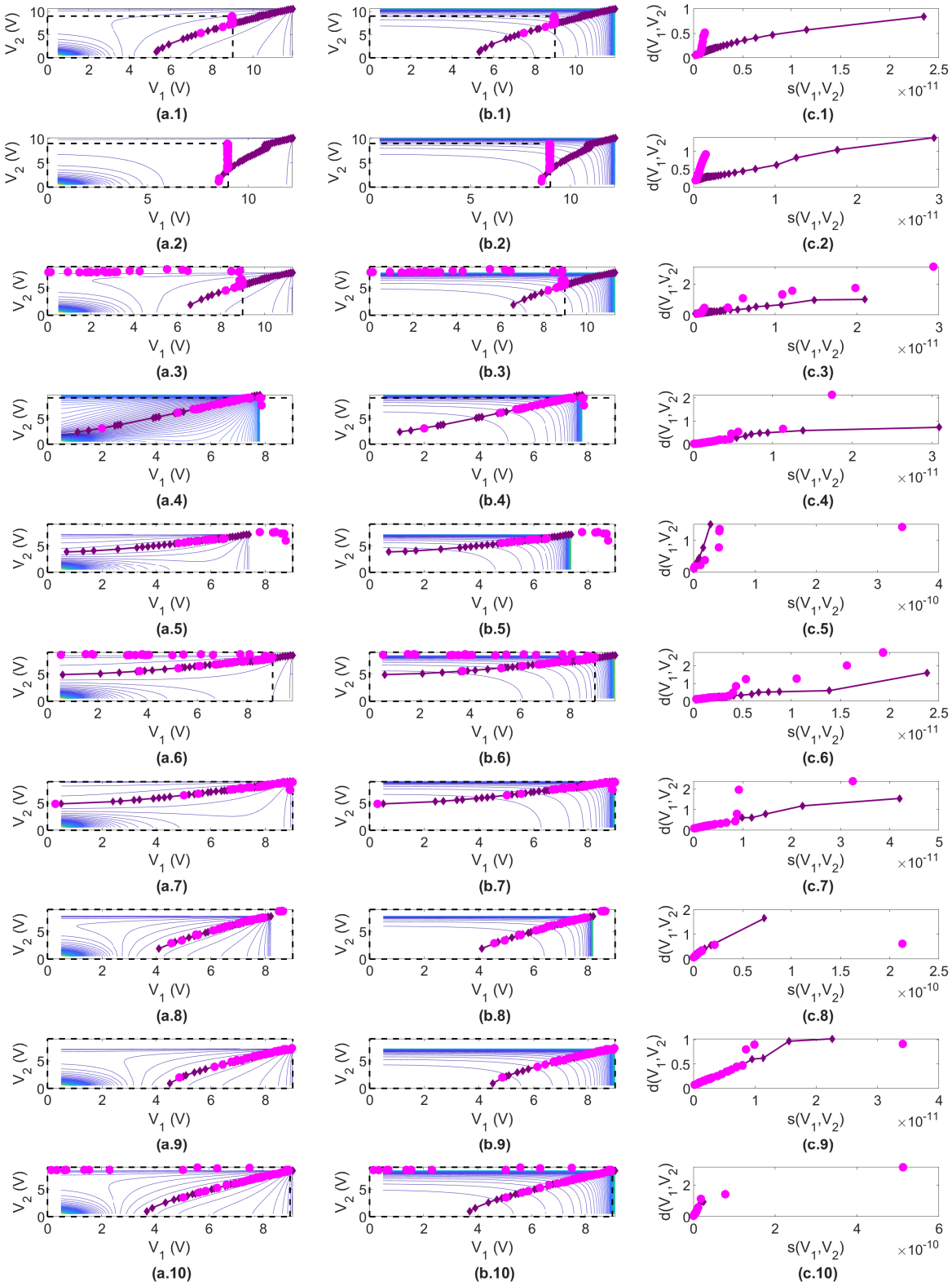

Figure E.35: Multi-objective tuning combined final results for (1) to (10) non-ideal twin-accelerometers – approach 2, resampling boundary strategy. (a) DE population (magenta), DE optimization variables boundaries (black), and mapped Pareto-set (purple) over distortion level curves in the decision space. (b) The same as (a), over sensitivity level curves. (c) DE cost function (magenta) and mapped Pareto-front (purple) in the objective space.

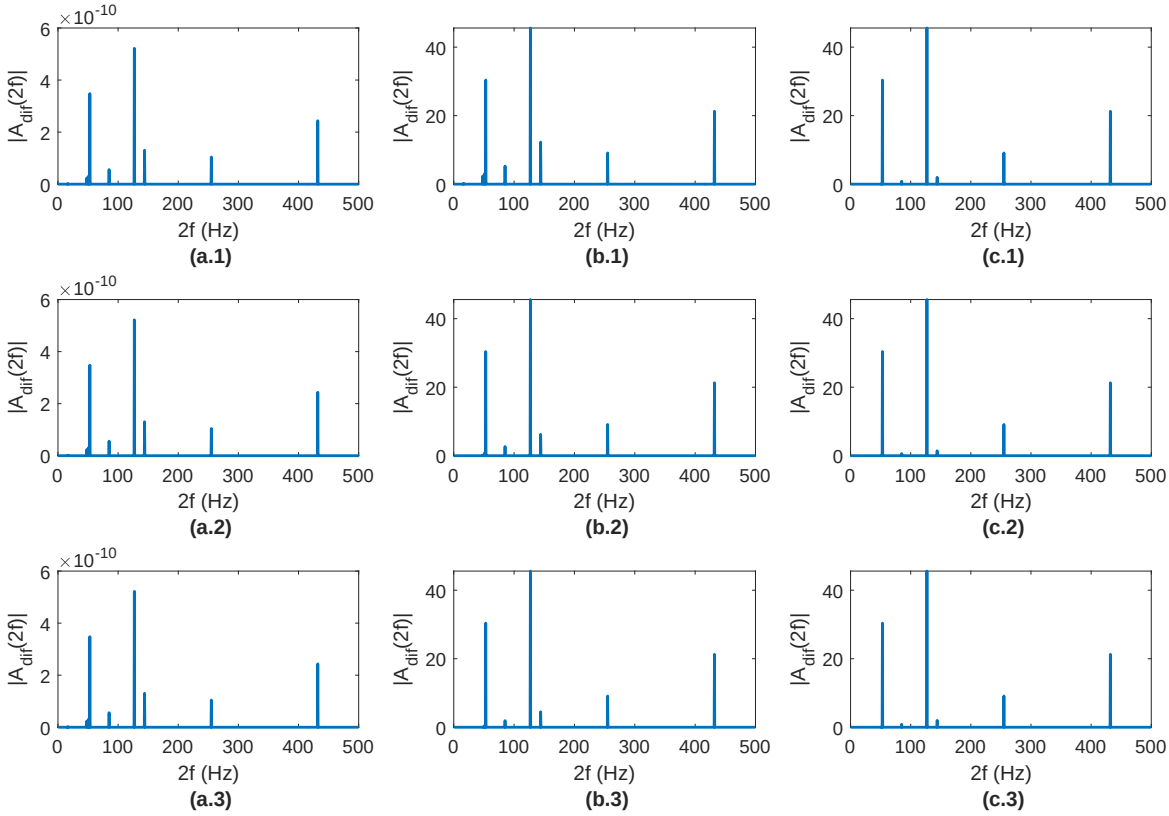

Figure E.36: Tuning validation results for a non-ideal twin-structure – brick-wall boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2. (1) Mono-objective tuning. (2) Multi-objective tuning, approach 1. (3) Multi-objective tuning, approach 2.

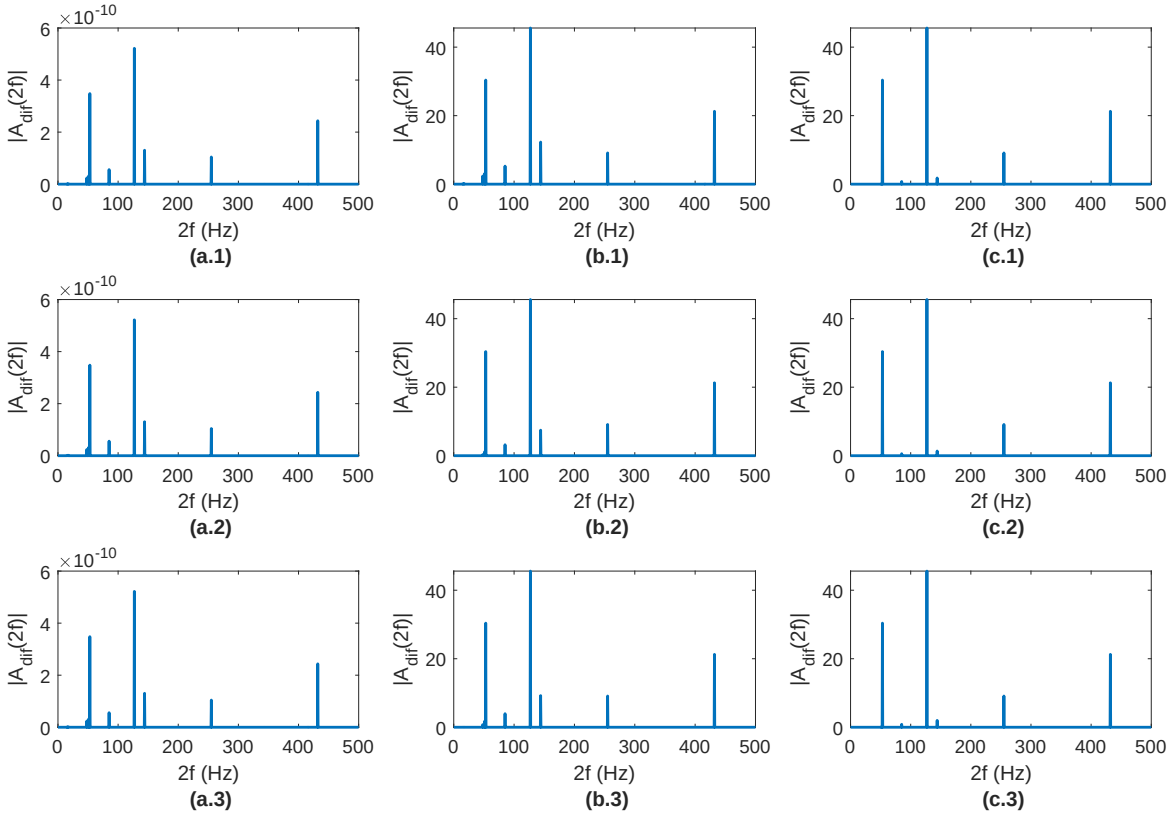

Figure E.37: Tuning validation results for a non-ideal twin-structure – random reinitialization boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2. (1) Mono-objective tuning. (2) Multi-objective tuning, approach 1. (3) Multi-objective tuning, approach 2.

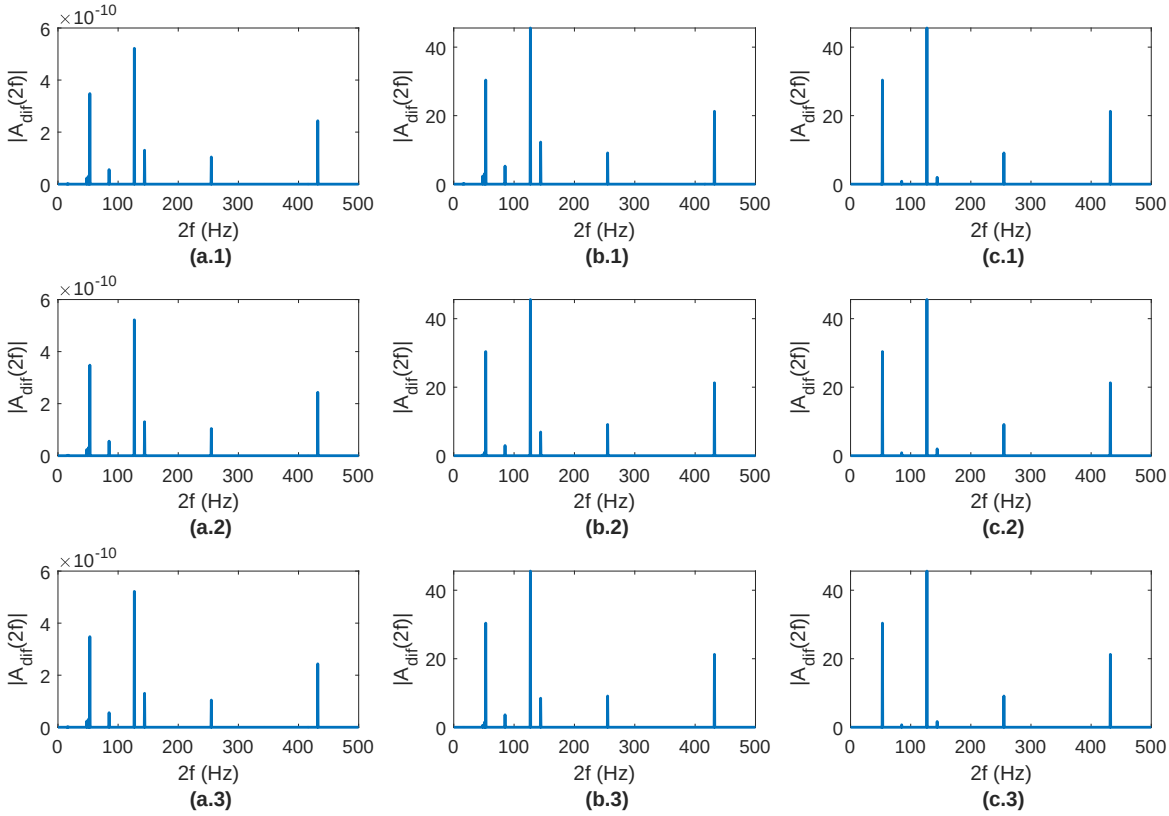

Figure E.38: Tuning validation results for a non-ideal twin-structure – resampling boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2. (1) Mono-objective tuning. (2) Multi-objective tuning, approach 1. (3) Multi-objective tuning, approach 2.

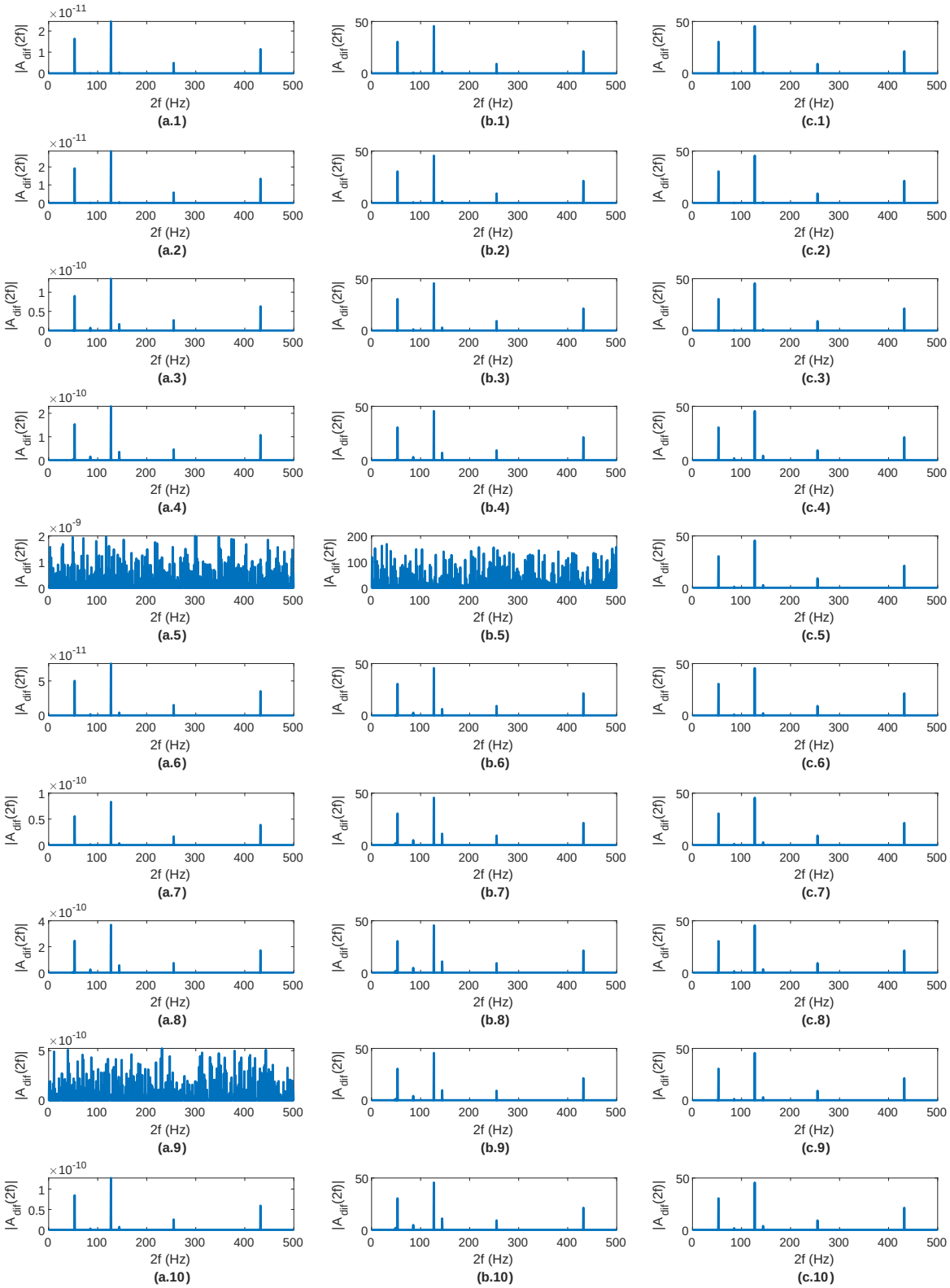

Figure E.39: Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, bounce-back boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

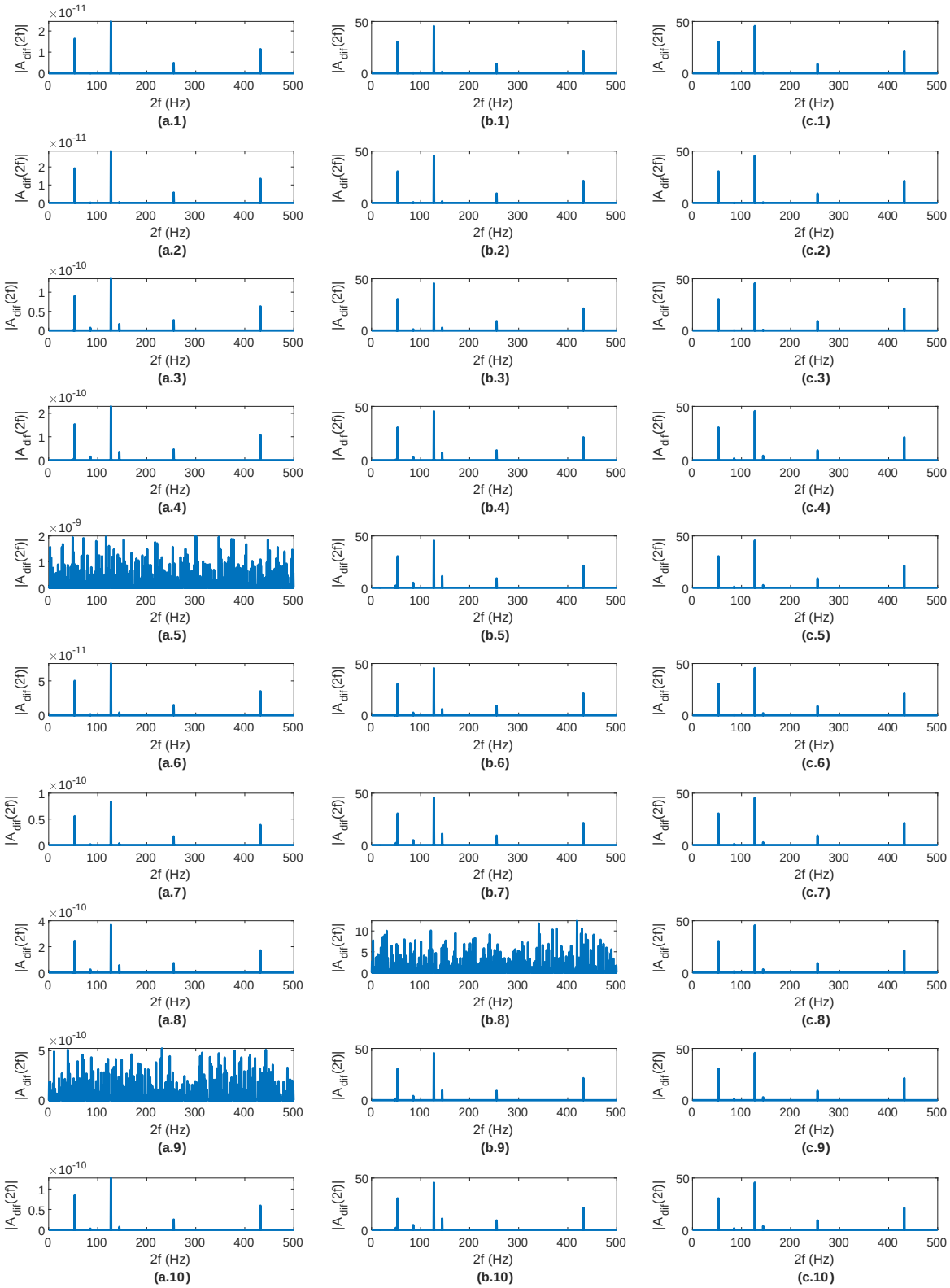

Figure E.40: Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – brick-wall boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

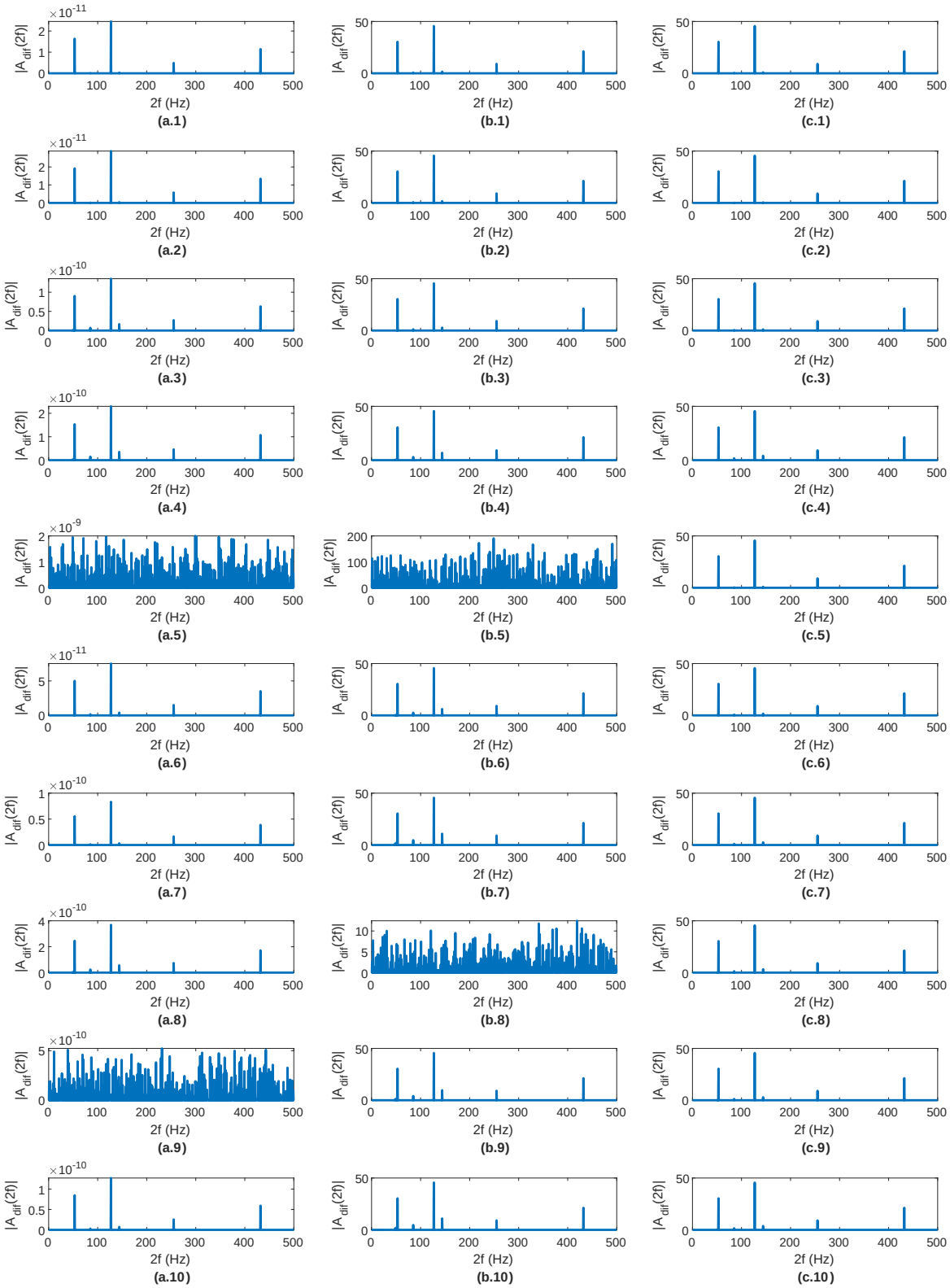

Figure E.41: Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, random reinitialization boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

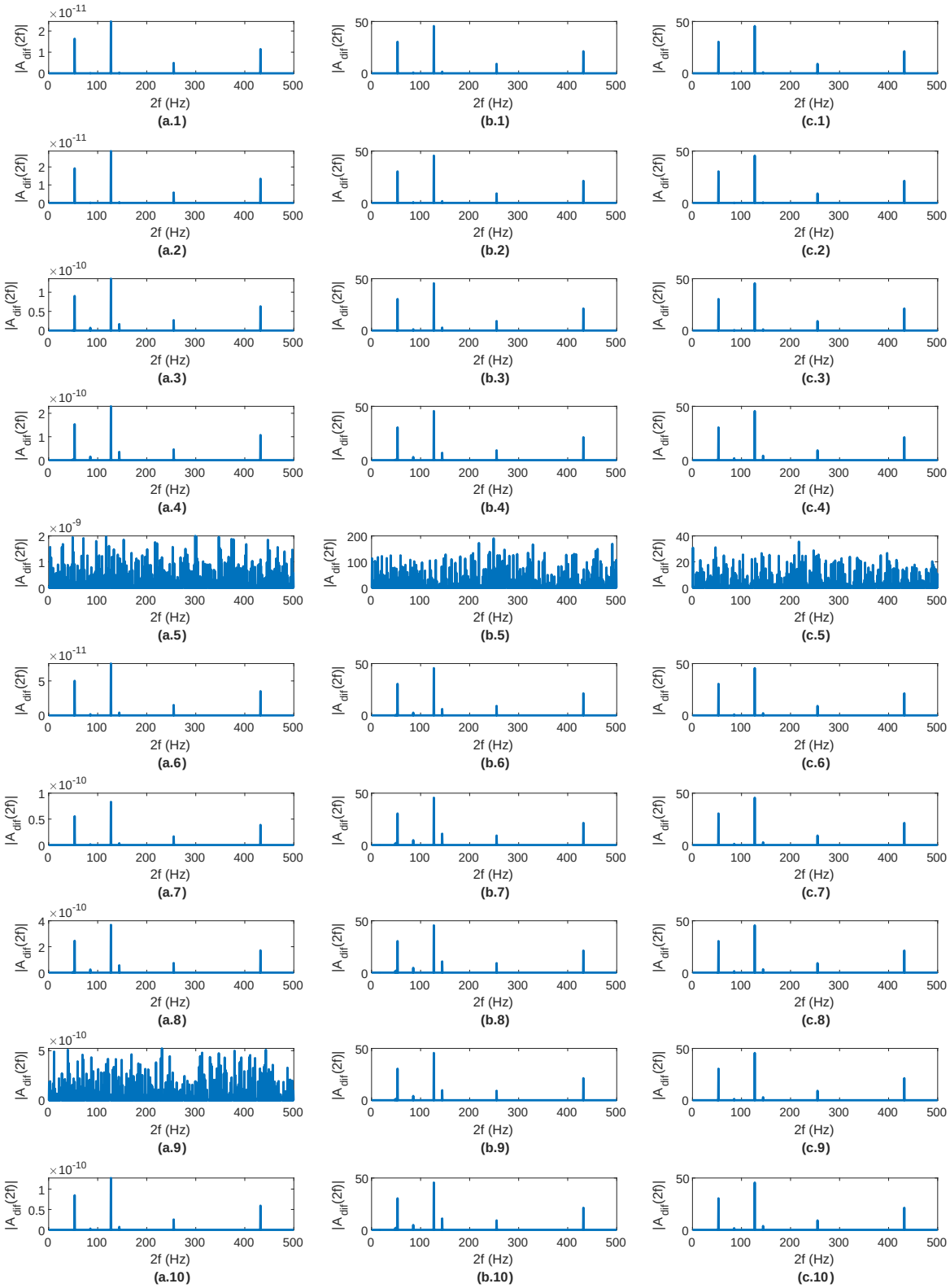

Figure E.42: Mono-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, resampling boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

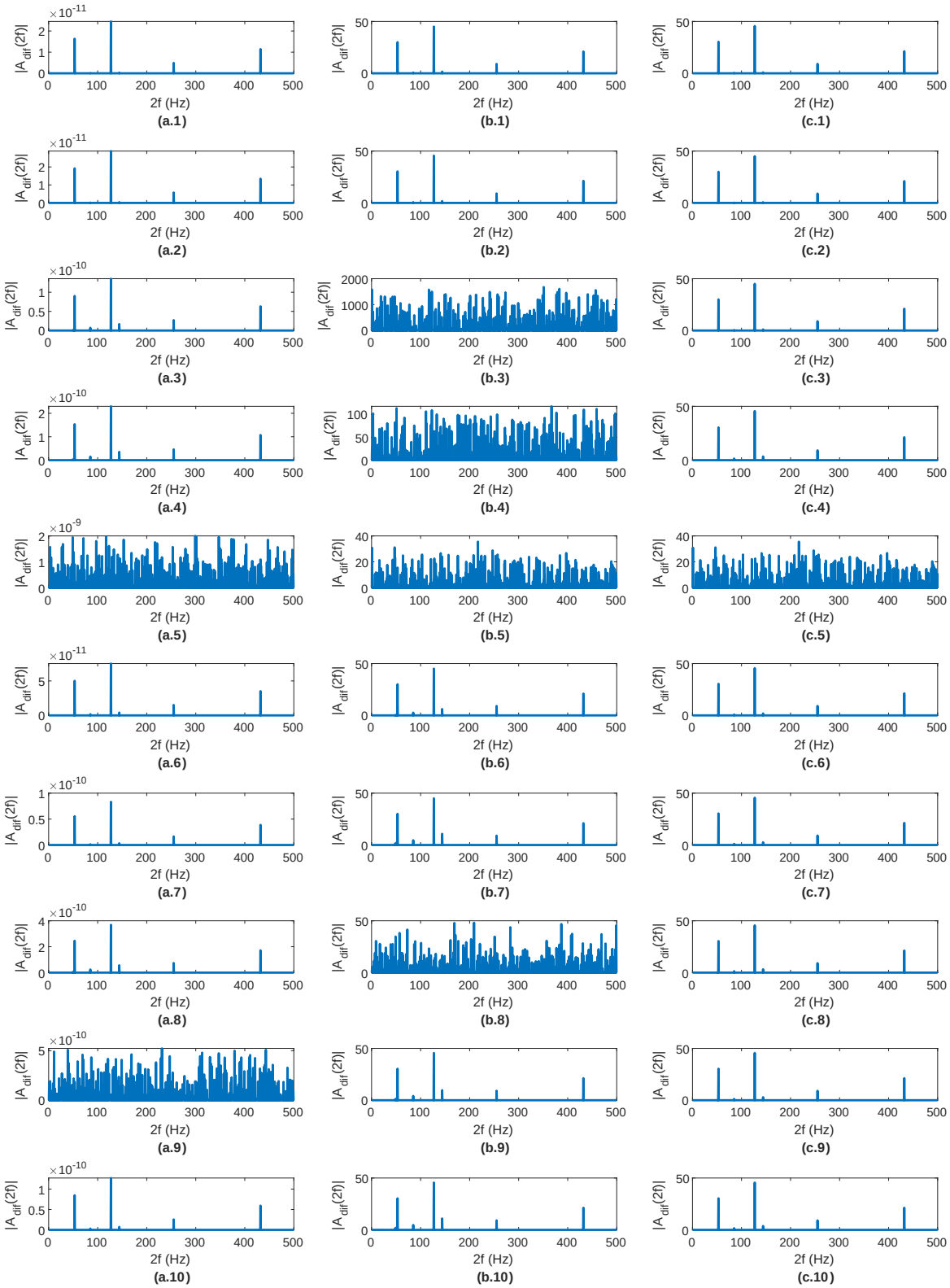

Figure E.43: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, bounce-back boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

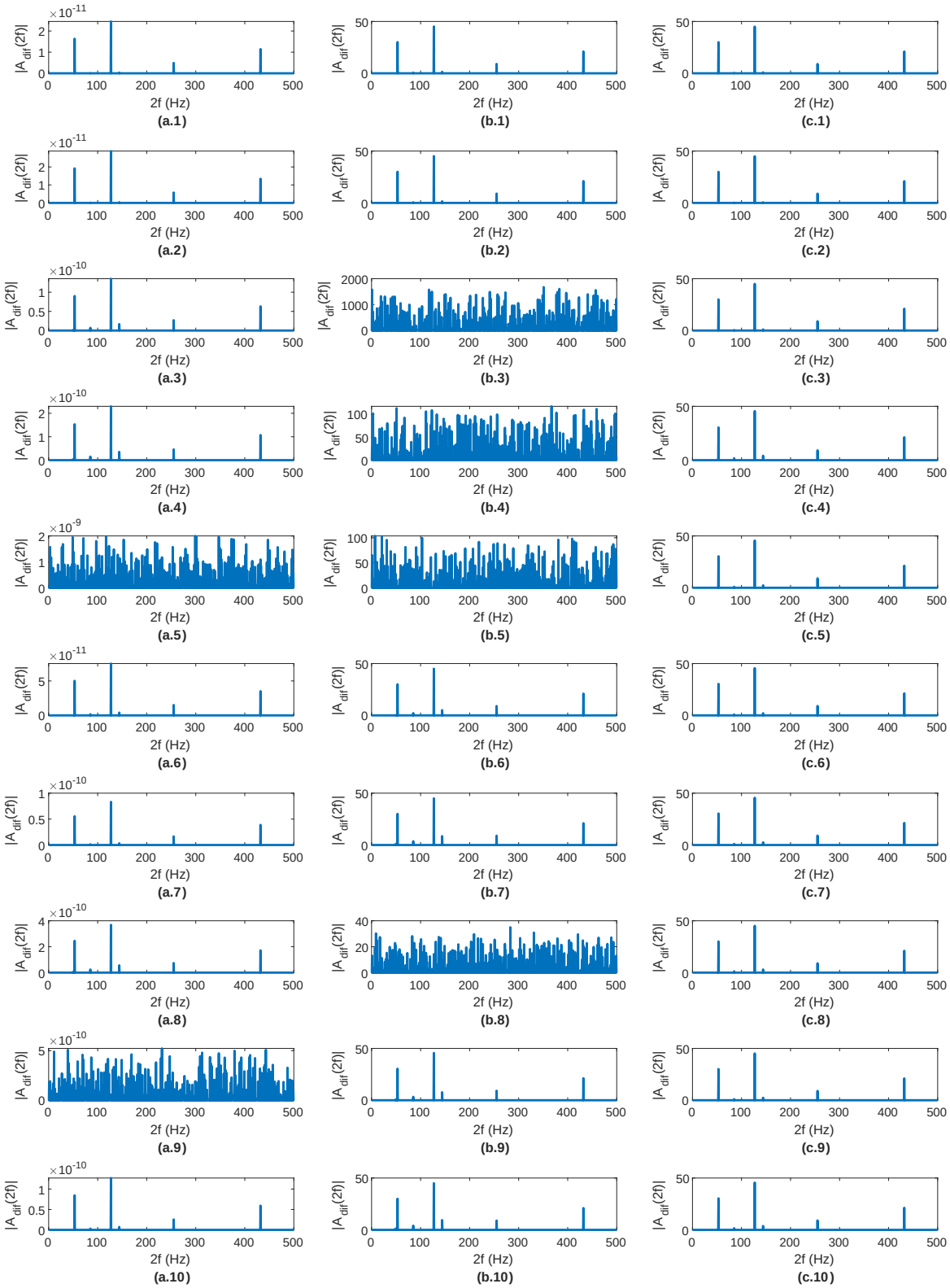

Figure E.44: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, brick-wall boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

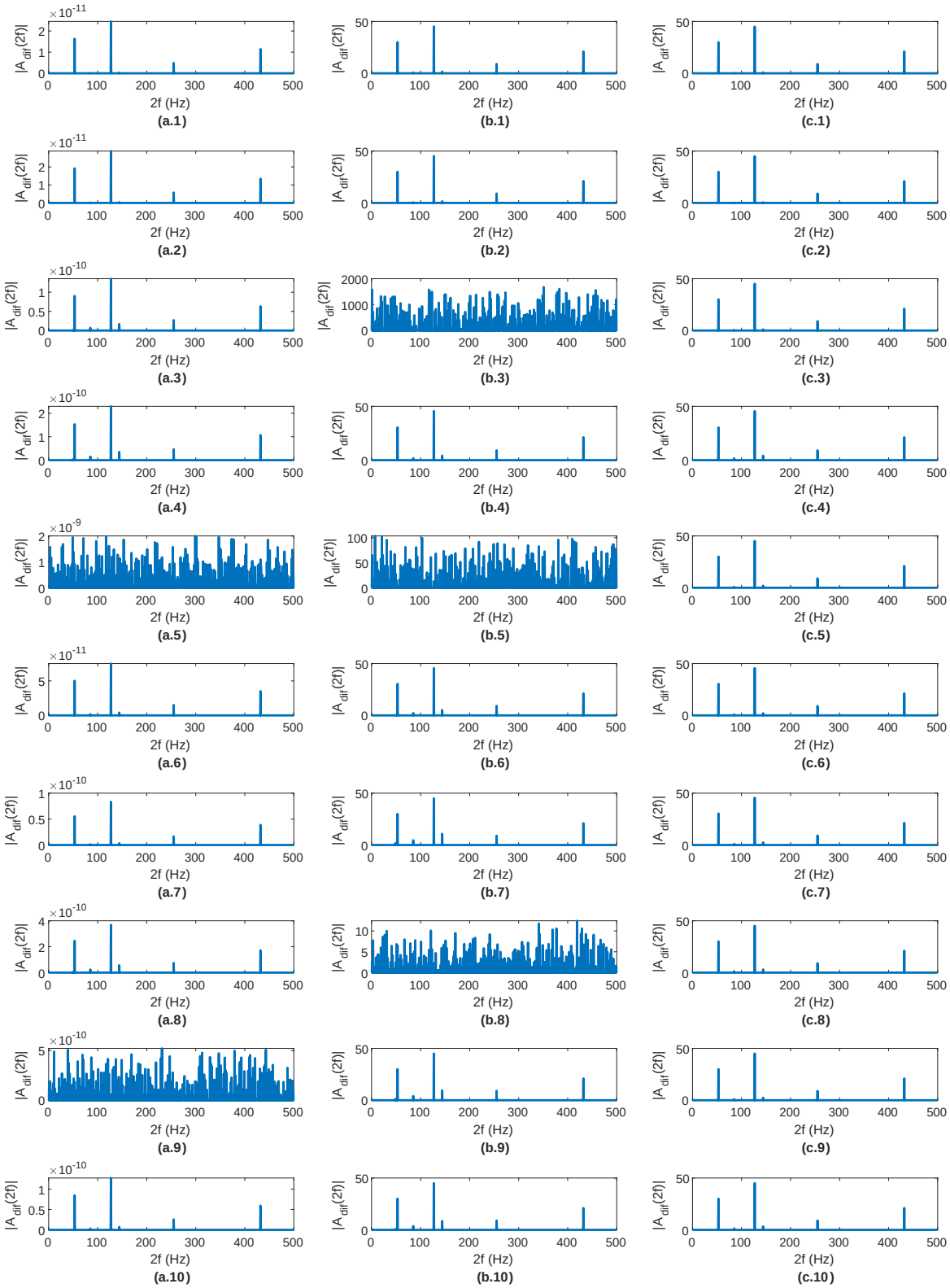

Figure E.45: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, random reinitialization boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

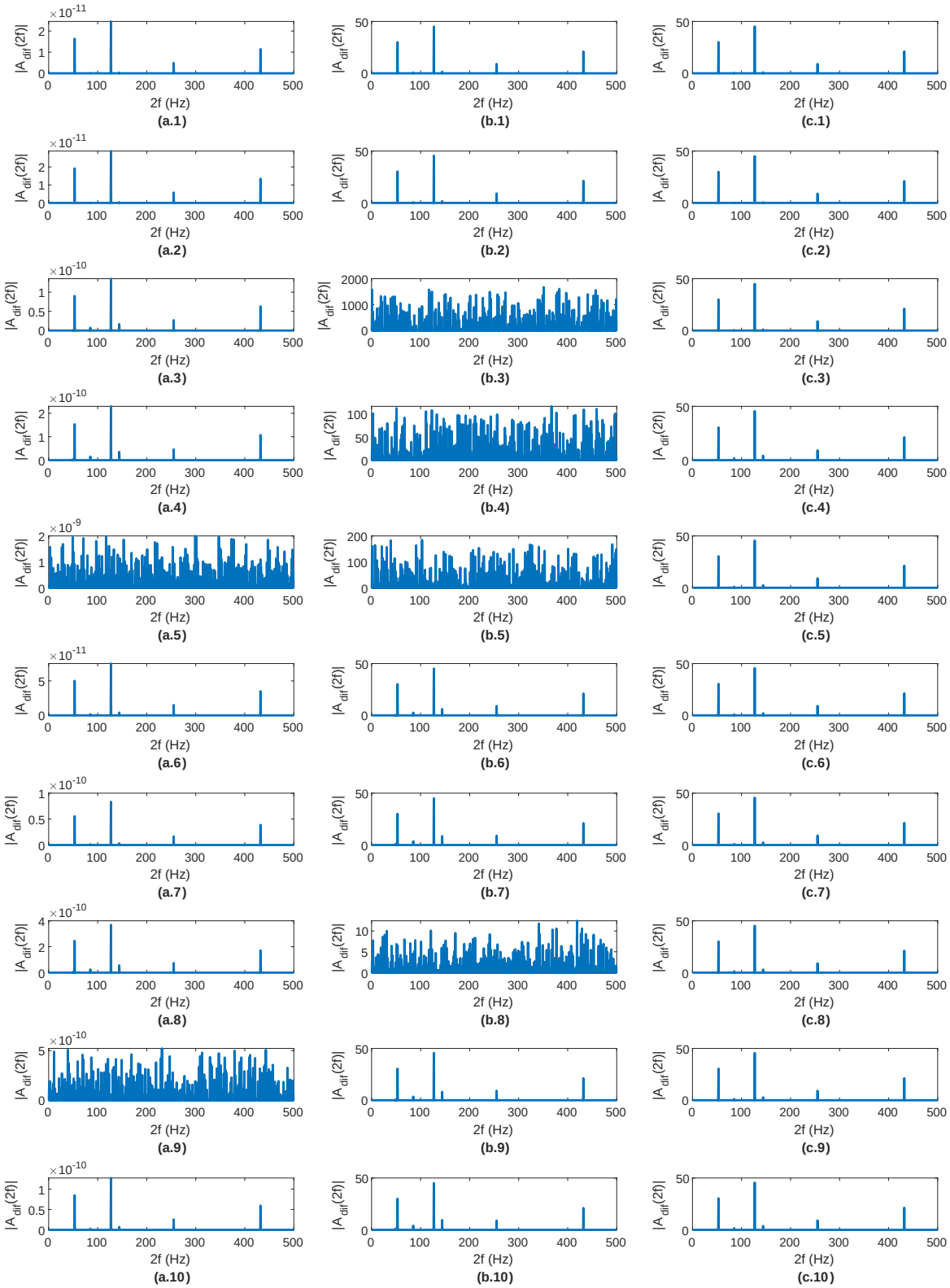

Figure E.46: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 1, resampling boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

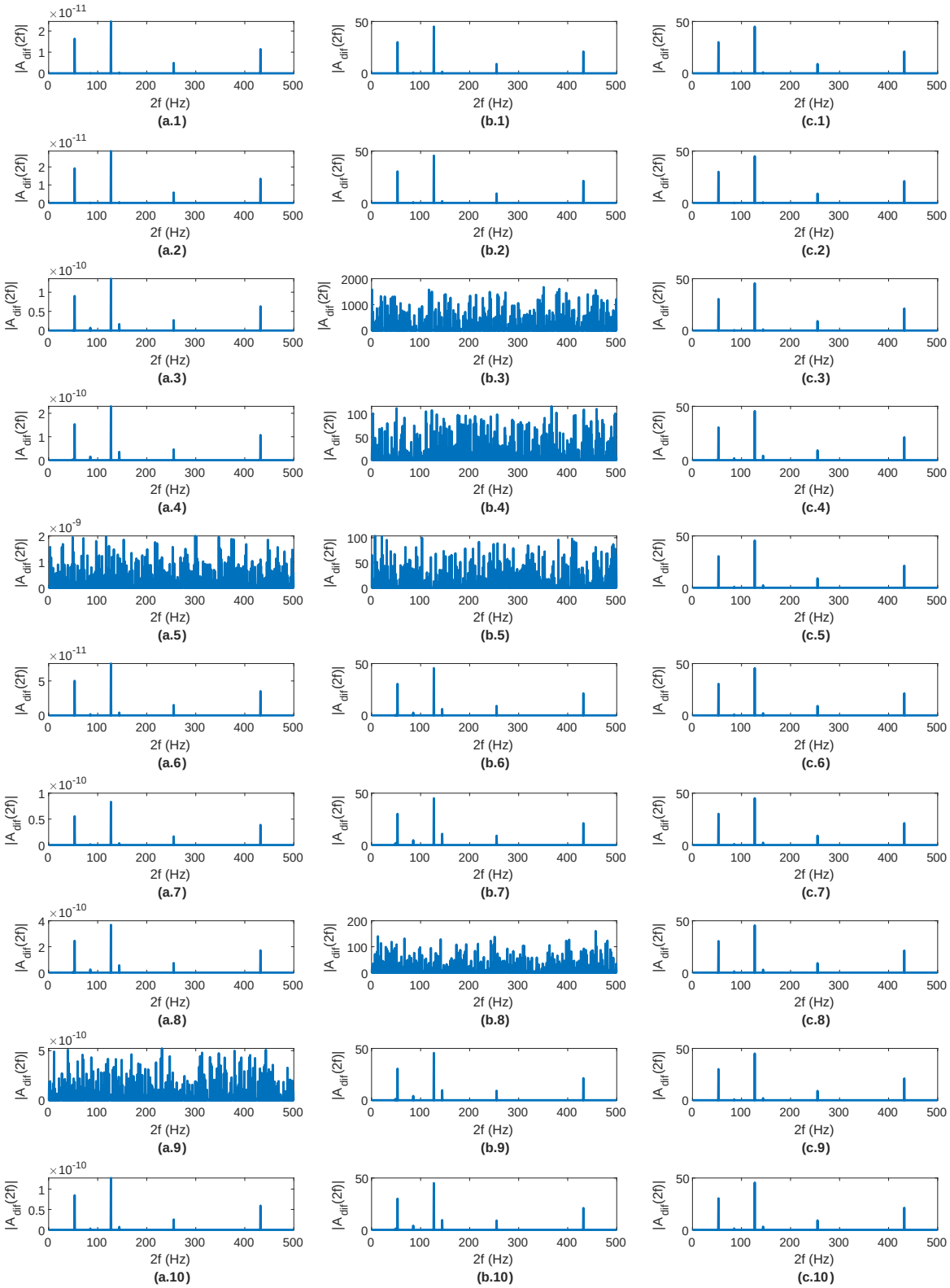

Figure E.47: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, bounce-back boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

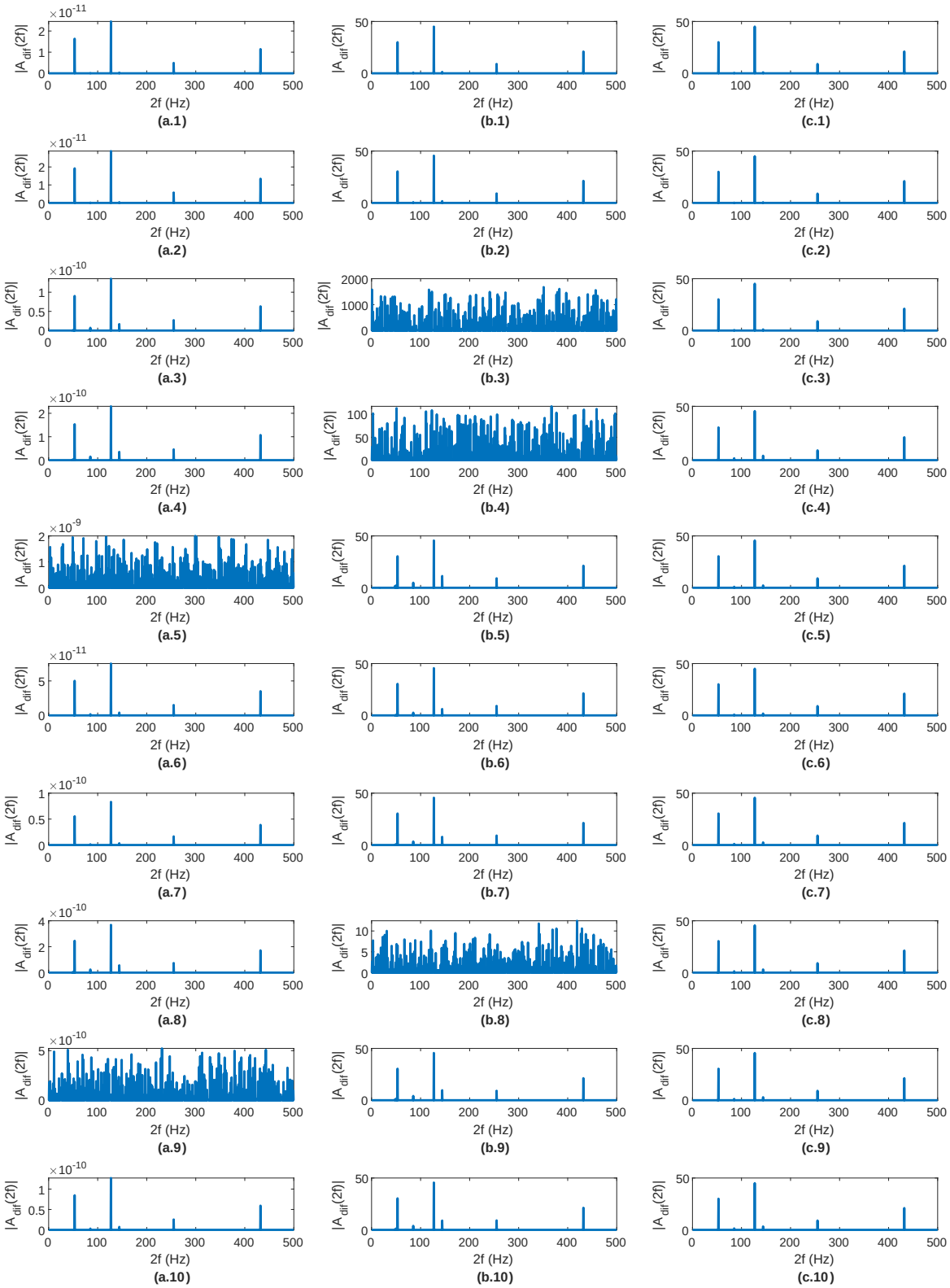

Figure E.48: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, brick-wall boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

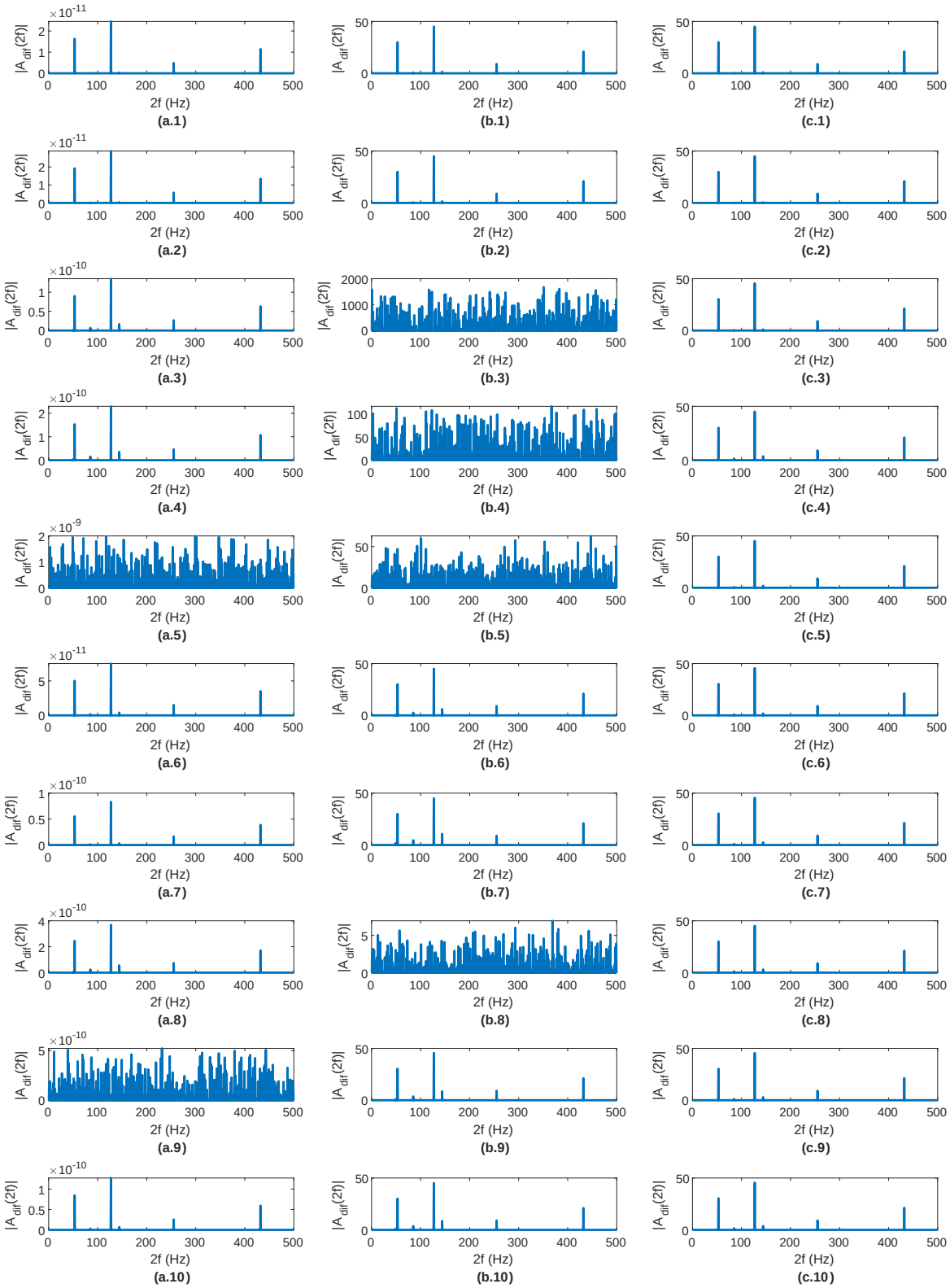

Figure E.49: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, random reinitialization boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.

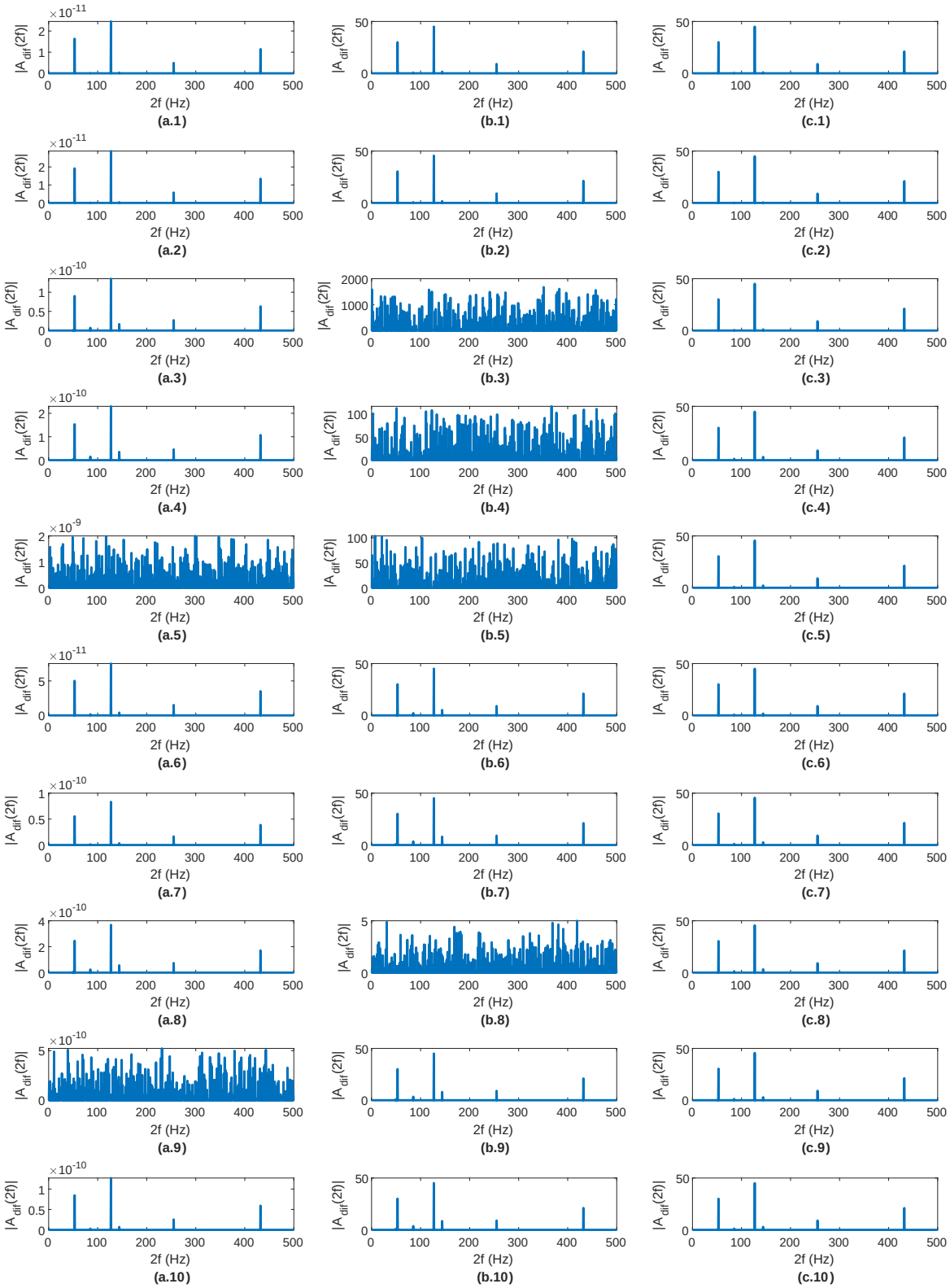

Figure E.50: Multi-objective tuning validation results for (1) to (10) non-ideal twin-accelerometers – approach 2, resampling boundary strategy. (a) Spectrum before tuning. (b) Tuned spectrum, maximum distortion level 0.8. (c) Tuned spectrum, maximum distortion level 0.2.