



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2015/04.23.19.03-TDI

UM ALGORITMO DE CLUSTERIZAÇÃO DE DADOS PARA AUXÍLIO À ANÁLISE DE COMPORTAMENTOS DE SISTEMAS

Lorena Pena Gayarre

Tese de Doutorado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle, orientada pelos Drs. Marcelo Lopes de Oliveira e Souza, e Guilherme Conceição Rocha, aprovada em 18 de maio de 2015.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3JCFC22>>

INPE

São José dos Campos

2015

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

COMISSÃO DO CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELECTUAL DO INPE (DE/DIR-544):

Presidente:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Membros:

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Dr. Amauri Silva Montes - Coordenação Engenharia e Tecnologia Espaciais (ETE)

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

Dr. Joaquim José Barroso de Castro - Centro de Tecnologias Espaciais (CTE)

Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr^a Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

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

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

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

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

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

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

EDITORAÇÃO ELETRÔNICA:

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

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



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2015/04.23.19.03-TDI

UM ALGORITMO DE CLUSTERIZAÇÃO DE DADOS PARA AUXÍLIO À ANÁLISE DE COMPORTAMENTOS DE SISTEMAS

Lorena Pena Gayarre

Tese de Doutorado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle, orientada pelos Drs. Marcelo Lopes de Oliveira e Souza, e Guilherme Conceição Rocha, aprovada em 18 de maio de 2015.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3JCFC22>>

INPE
São José dos Campos
2015

Dados Internacionais de Catalogação na Publicação (CIP)

Gayarre Peña, Lorena.

G254a Um algoritmo de clusterização de dados para auxílio à análise de comportamentos de sistemas / Lorena Pena Gayarre. – São José dos Campos : INPE, 2015.
xxviii + 157 p. ; (sid.inpe.br/mtc-m21b/2015/04.23.19.03-TDI)

Tese (Doutorado em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2015.

Orientadores : Dr. Marcelo Lopes de Oliveira e Souza, e Guilherme Conceição Rocha.

1. Clusterização. 2. Algoritmo. 3. Métodos baseados em dados. 4. Detecção de anomalias. 5. Análise de comportamentos. I.Título.

CDU 629.78:519.254



Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

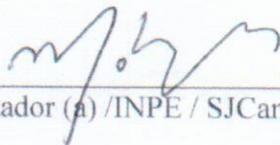
This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).

Aprovado pela Banca Examinadora em
cumprimento a requisito exigido para a
obtenção do Título de **Doutor(a)** em
ETE/Mecânica Espacial e Controle.

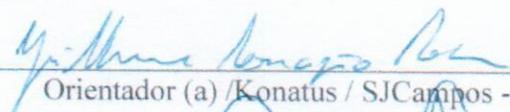
Dr. Antonio Fernando Bertachini de
Almeida Prado


Presidente /INPE /SJC Campos - SP

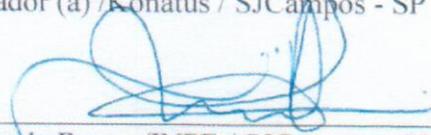
Dr. Marcelo Lopes de Oliveira e
Souza


Orientador (a) /INPE / SJC Campos - SP

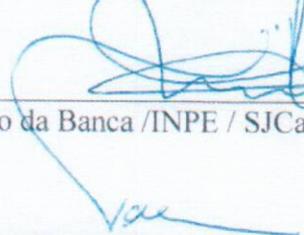
Dr. Guilherme Conceição Rocha


Orientador (a) /Konatus / SJC Campos - SP

Dr. Paulo Giácomo Milani


Membro da Banca /INPE / SJC Campos - SP

Dr. Valdemir Carrara


Membro da Banca / INPE / SJC Campos-SP

Dr. Fernando José de Oliveira Moreira


Convidado (a) /EMBRAER / SJC campos-SP

Dr. Henrique Mohallem Paiva


Convidado(a) /MECTRON / SJC Campos - SP

Este trabalho foi aprovado por:

maioria simples.

unanimidade

Título: "Um Algoritmo de Clusterização de Dados para Auxílio à Análise de Comportamento de Sistemas"

Aluno (a): Lorena Gayarre Peña

São José dos Campos, 18 de Maio de 2015.

Sempre tem um jeito e o Brasil me fez aprender isso.

No digas todo lo que piensas, pero piensa todo lo que dices.

A LAS QUE ESTÁN POR LLEGAR, QUE NOS HAGAN MUY FELICES Y A LA QUE SE FUE QUE ME ENSEÑÓ A SER FUERTE Y A LUCHAR POR MIS SUEÑOS (Te quiero y nunca te olvido).

27-05-2015, dia feliz minhas queridas filhas. Vocês mudaram nossa vida. Mamãe ama vocês.

AGRADECIMENTOS

Ao INPE/ETE/CMC e ao CNPq pela oportunidade e suporte para a realização deste trabalho.

Aos meus orientadores Marcelo Lopes e Guilherme Rocha, por me apoiarem e me indicarem como continuar nos momentos difíceis.

Ao professor Marcos Botelho, pelas sugestões feitas durante o desenvolvimento.

Ao Eloy, Danilo, Cassya, Liana e demais colegas que colaboraram para fazer este doutorado possível.

Ao Jairo Siqueira por dar o apoio na reta final do doutorado, e com quem espero continuar colaborando depois da apresentação.

Ao Pierre Bigot, por ler e reler, organizar e comentar linha a linha este documento, me ajudando e me apoiando a estruturar as ideias e esclarecer os conceitos.

Ao Brasil, por mudar-me a vida.

RESUMO

Os sistemas atuais se caracterizam pela variedade de incertezas humanas, ambientais e sistêmicas que afetam os comportamentos deles, influenciando nos **modos normais** de operação e fazendo com que estes se comportem de **modos anômalos** ou até de **modos falhos**. Como isto é inevitável, é mandatória a **detecção de anomalias e a análise de comportamentos** no projeto de sistemas altamente confiáveis como nas áreas espacial ou aeronáutica. Motivados por esta necessidade, o objetivo deste trabalho é propor, implementar e testar um algoritmo de clusterização de dados para auxílio à análise de comportamentos de sistemas.. Neste trabalho procura-se que o algoritmo desenvolvido seja de aplicação genérica, i.e., não se precise ter conhecimento prévio do sistema a ser estudado. Para isso, primeiro é feita uma apresentação dos métodos baseados em dados e dos métodos baseados em modelos para identificação de comportamentos, explicando por que foram escolhidos os métodos baseados em dados. Depois são apresentados os métodos baseados em dados que são utilizados no desenvolvimento do algoritmo, a saber, métodos de clusterização, métodos matemáticos, métodos estatísticos e métodos de filtragem. Fazendo uso dessas técnicas, é desenvolvido o algoritmo de análise de comportamentos, chamado de **BAbyLO-BR**. Em seguida, o algoritmo é validado utilizando quatro estudos de caso de diferentes características e áreas de aplicação. Esta diversidade de estudos de caso permite ilustrar a generalidade de aplicação do algoritmo. Por último, um estudo que enumera as vantagens e desvantagens de aplicação deste novo algoritmo é realizado.

A DATA CLUSTERING ALGORITHM TO ASSIST IN SYSTEM BEHAVIOURAL ANALYSIS

ABSTRACT

Current systems are characterized by the variety of human, environmental and system uncertainties which affect the normal operation of the system, causing anomalous or even failed behavior. This situation is not avoidable, so anomaly detection and behavioral analysis of the component or system is needed and even mandatory, especially in high reliability systems such as those from the space and aeronautical industry. Under this motivation, the aim of this thesis is to propose, implement and test a data clustering algorithm to assist in system behavioural analysis.. In this work, the algorithm is intended to be of general application, i.e., it will not require previous knowledge of the system under study. To do this, first a theoretical comparison between data based methods and model based methods is done, explaining why data based methods are chosen. Later, the used data based methods are presented: clustering methods, mathematical methods, statistical methods, and filtering methods. By using these methods, a behavioral analysis algorithm is developed; it is called BAbyLO-BR algorithm. Then, the algorithm is validated by its application to four study cases with diverse characteristics and belonging to different areas of research. This diversity allows illustrating the general application of the algorithm. Finally, a study enumerating the vantages and disadvantages of using the algorithm for anomaly detection and behavioral analysis is conducted.

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 2-1. Gêneros de classificação de técnicas de clusterização.....	15
Figura 2-2. Exemplo de dendograma.	16
Figura 2-3. a) Exemplo de variável aleatória com baixa dispersão. b) Exemplo de variável aleatória com alta dispersão.	21
Figura 2-4. Distribuição gaussiana.	22
Figura 2-5. Exemplo de máscara 3x3.	26
Figura 4-1. Sistema de três tanques.	36
Figura 4-2. Modelo em Simulink do sistema três tanques.	38
Figura 4-3. Representação dos quatro pixels aumentados possíveis.	44
Figura 4-4. Processo de <i>padding</i>	45
Figura 4-5. Cálculo do valor do pixel aumentado no algoritmo branco e preto.	49
Figura 4-6. Cálculo de variância em cada modo de aglutinação.	50
Figura 4-7. Processo de aglutinação aplicado a uma grade de tamanho 8×8	51
Figura 4-8. Dendograma completo após processo de aglutinação hierárquico segundo o algoritmo branco e preto.	51
Figura 4-9. Exemplo de identificação de <i>cluster</i> (em amarelo) no dendograma.	52
Figura 4-10. Exemplo de clusterização final.	52
Figura 4-11 a) Exemplo de histograma natural com um comportamento único; b) Exemplo de histograma natural com dois comportamentos.	57
Figura 4-12. Clusterização unidimensional do algoritmo BAbyLO-BR.	58
Figura 4-13. Clusterização multidimensional do algoritmo BAbyLO-BR.	59
Figura 4-14. Exemplo de dados de entrada do algoritmo BAbyLO-BR.	60
Figura 4-15. a) Exemplo de vale não desejado em vizinhança de subida, b) Exemplo de pico inesperado em vizinhança de subida e consequente vale não desejado.	64
Figura 4-16. Exemplo de histograma antes e depois do processo de suavização.	65
Figura 4-17. Exemplo de aplicação do corolário da primeira derivada para identificação dos <i>clusters</i>	67
Figura 4-18. Representação gráfica da combinação de resultados unidimensionais de clusterização para um caso bidimensional.	68
Figura 4-19. Exemplo de arquivo de saída "DadosClassificados.txt".	69

Figura 4-20. Exemplo de arquivo de saída "ClustersFormados.txt" .	70
Figura 5-1. Exemplo de amostras com e sem falha no estudo de caso de tubulações hidráulicas.	76
Figura 5-2. Amstras sem classificar.	77
Figura 5-3. Classificação das amostras usando a máscara de características [0 0.5 1 0.5 0] a) com visualização em duas dimensões e b) com visualização em três dimensões.	78
Figura 5-4. Classificação das amostras usando a máscara de características [1 2 2 2 1].	78
Figura 5-5. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de mau funcionamento.	79
Figura 5-6. Falha de entupimento em $t=0s$. a) Distribuição espacial das amostras em $t=30s$. b) Resultados da classificação segundo o algoritmo BAbyLO-BR.	80
Figura 5-7. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de normal e mau funcionamento.	81
Figura 5-8. Falha de entupimento em $t=15s$. a) Distribuição espacial das amostras em $t=30s$. b) Resultados da classificação segundo o algoritmo BAbyLO-BR.	82
Figura 5-9. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de 'não funcionamento: perda de sensor'.	83
Figura 5-10. Perda do sensor em $t=0s$. Distribuição espacial das amostras em $t=30s$.	83
Figura 5-11. Resultados de classificação segundo algoritmo BAbyLO-BR. a) usando máscara padrão [0 0.5 1 0.5 0] e b) usando máscara mais severa [1 2 2 2 1].	84
Figura 5-12. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de 'normal e não funcionamento: perda do sensor'.	85
Figura 5-13. Perda do sensor em $t=15s$. a) Distribuição espacial das amostras em $t=30s$. b) Resultados da classificação segundo o algoritmo BAbyLO-BR.	86
Figura 5-14. Modelo do SCAO da plataforma PMM em Modo Nominal.	88
Figura 5-15. Exemplo de dados de entrada no estudo de caso 'Identificação de falhas de sensores e atuadores da Plataforma Multi-Missão'.	94
Figura 5-16 Imagem recebida do satélite Landsat para identificação de corpos d'água.	99
Figura 5-17. Resultados de identificação de corpos d'água; a) Produzidos pelo algoritmo BAbyLO-BR e b) produzidos pelo analista da área usando classificador e processo manual.	101
Figura 5-18. Imagem exemplo de Arifin (1974); Imagem real, histograma da imagem e imagem classificada em Arifin (1974).	103

Figura 5-19. Resultados de classificação da imagem do peixe listrado; a) Imagem de entrada do algoritmo, b) Classificação da imagem usando o algoritmo BAbyLO-BR e c) Classificação da imagem realizada em Arifin (1974).	105
Figura B.1. Exemplo de histograma com variações bruscas onde x representa os valores do parâmetro discretizado e y o número de amostras por janela.	128
Figura B.2. a) Exemplo 1 de histograma; b) Exemplo 2 de histograma.	129
Figura C.1. Representação espacial das amostras de campo magnético em caso de simulação sem perturbação, a) sem classificação e b) após classificação pelo algoritmo.	134
Figura C.2. Representação espacial das amostras de campo magnético em caso de funcionamento com anomalia isolada, a) antes da classificação e b) depois da classificação.....	135
Figura C.3. Evolução temporal dos parâmetros Bx e By com perturbação perto de $t=100s$	136
Figura C.4. Representação espacial das amostras de campo magnético em caso de dois comportamentos, a) sem classificação e b) após classificação pelo algoritmo.	136
Figura C.5. Focos de calor no Brasil a) sem classificação e b) classificados pelo algoritmo BAbyLO-BR.....	139
Figura C.6. Distribuição dos focos de calor mês a mês, após a aplicação do algoritmo sobreposto a um mapa de chuva anual acumulada, derivada do TRMM.	140
Figura C.7. Distribuição de focos de calor na área de Mato Grosso, a) antes de aplicar o algoritmo e b) depois de aplicar o algoritmo.....	141
Figura D.1 Resposta em frequência, eixo x, linearizado e desacoplado, malha aberta	144
Figura D.2. Resposta em frequência, eixo y, linearizado e desacoplado, malha aberta	144
Figura D.3. Resposta em frequência, eixo z, linearizado e desacoplado, malha aberta	145
Figura E.1. Ilustração da manobra de de-tumble com R-LQR (parâmetros de sensores, valores angulares convertidos para graus, graus/s, graus/s ²)	147
Figura E.2. Ilustração da manobra de de-tumble com R-LQR (parâmetros de atuadores)	148
Figura E.3. Propagação da falha F1, eixo X, t = 30 segundos.....	148
Figura E.4. Propagação da falha F1, eixo X, t = 30 segundos.....	149
Figura E.5. Propagação da falha F2, eixo X, t = 30 segundos.....	149
Figura E.6. Propagação da falha F2, eixo X, t = 30 segundos.....	150
Figura E.7. Propagação da falha F3, eixo X, t = 30 segundos.....	150
Figura E.8. Propagação da falha F3, eixo X, t = 30 segundos.....	151
Figura E.9. Propagação da falha F4, eixo X, t = 30 segundos.....	151

Figura E.10. Propagação da falha F4, eixo X, t = 30 segundos.....	152
Figura E.11. Propagação da falha T1, eixo X, t = 30 segundos	152
Figura E.12. Propagação da falha T1, eixo X, t = 30 segundos	153
Figura E.13. Propagação da falha T2, eixo X, t = 30 segundos	153
Figura E.14. Propagação da falha T2, eixo X, t = 30 segundos	154
Figura E.15. Propagação da falha T3, eixo X, t = 30 segundos	154
Figura E.16. Propagação da falha T3, eixo X, t = 30 segundos	155
Figura E.17. Propagação da falha T4 eixo X, t = 30 segundos	155
Figura E.18. Propagação da falha T4, eixo X, t = 30 segundos	156
Figura E.19. Propagação da falha T4, eixo X, t = 250 segundos	156
Figura E.20. Propagação da falha T4, eixo X, t = 250 segundos	157

LISTA DE TABELAS

	<u>Pág.</u>
Tabela 2-1. Classificação dos métodos de clusterização.	14
Tabela 2-2. Comparativo de técnicas hierárquicas e técnicas particionais.....	17
Tabela 2-3 Possíveis tomadas de decisão de um classificador.	23
Tabela 3-1. Resumo de fases de desenvolvimento, técnicas utilizadas e algoritmo desenvolvidos até chegar ao algoritmo final.	33
Tabela 4-1. Valores utilizados para simulação do sistema três tanques.....	39
Tabela 4-2. Evolução da implementação do algoritmo BAbyLO-BR versão a versão. .	55
Tabela 5-1. Teste de hipóteses para caso de comportamento normal	79
Tabela 5-2. Teste de hipóteses para caso de entupimento em t=0s.	81
Tabela 5-3. Teste de hipóteses para caso de entupimento em t=15s.	82
Tabela 5-4. Teste de hipóteses para caso de perda de sensor em t=0s.	84
Tabela 5-5. Teste de hipóteses para caso de perda de sensor em t=15s.	86
Tabela 5-6. Tabela resumo de resultados de aplicação do algoritmo no caso Identificação de falhas em sensores e atuadores na PMM.	96
Tabela 5-7. Classificação das amostras segundo o nível de cinzas.	100
Tabela 5-8. Classificação das amostras segundo o nível de cinzas.	104
Tabela 5-9. Classificação de estudos de caso.	107
Tabela 6-1. Vantagens e desvantagens da utilização do algoritmo.	111
Tabela B-1. Estudo empírico do fator de porcentagem para determinação do critério de parada.	131
Tabela C-1. Identificação das informações disponibilizadas no produto MODIS MCD14ML.....	138

LISTA DE SIGLAS E ABREVIATURAS

BAbyLO-BR	Behavioral Analysis by Lore – Brasil
INPE	Instituto Nacional de Pesquisas Espaciais
MDA	Multiple Discriminant Analysis
MANOVA	Multivariate ANalysis Of VAriance and covariance
PCA	Principal Component Analysis
SDLC	Software Development Life Cycle
SID	Serviço de Informação e Documentação
SPG	Serviço de Pós-Graduação
TDI	Teses e Dissertações Internas

LISTA DE SÍMBOLOS

a_i	Amostra i de n_p parâmetros
a_{ij}	Parâmetro j da amostra i
$A \in \mathcal{M}^{n_a \times n_p}$	Matriz de dados de entrada do algoritmo BAbyLO-BR
A_{imagem}	Área da imagem
A_{pixel}	Área do pixel
A, B, C	Matrizes de caracterização do sistema no formato de espaço de estados
d	Medida de dissimilaridade
d_{eucl}	Distância euclidiana
d_{eucl2}	Distância euclidiana quadrada
d_{min}	Distância mínima entre duas amostras unidimensionais
$E(X)$	Esperança matemática
$f(x)$	Função densidade de probabilidade
$f(x, y)$	Imagem real
$g(x, y)$	Imagem processada após filtragem
h	Histograma
H_0	Hipótese nula
H_1	Hipótese alternativa
h_i	Altura [m]
h_{suav}	Histograma suavizado
$i_{compact}$	Iteração de compactação
k_{limiar}	Fator de porcentagem
l_{parada}	Limiar de parada
L_{pixel}	Tamanho do pixel
M_{prox}	Matriz de proximidade
$M \in \mathcal{R}^{n_m}$	Máscara
N	Tamanho do espaço amostral
$N(\mu, \sigma^2)$	Distribuição gaussiana de probabilidade
n_a	Número de amostras

n_i	Número inicial de janelas do histograma
n_m	Número de elementos da máscara
n_p	Número de parâmetros por amostra
n_{pixel_y}	Número de pixels do eixo y
n_{pixel_x}	Número de pixels do eixo x
n_s	Número de janelas do histograma
$n_{s_{10\%}}$	Número de janelas relevantes
n_v	Vizinhança da máscara
$p_{compact}$	Porcentagem de compactação
p_i	Pixel
p_j	Parâmetro j das n_a amostras
q_i	Vazão de líquido [m^3]
R_i	Resistência da tubulação [S/m^3]
$S = \{s_1, \dots, s_i\}$	Conjunto de janelas do histograma.
S_i	Seção
T	Operador de filtragem -> Máscara
U	Vetor de variáveis de entrada
v_{aglut}	Nível de cinza no pixel aumentado
X	Vetor de estado do sistema
Y	Vetor de variáveis de saída
α	Nível de significância
$1 - \beta$	Potência de contraste
μ	Média
σ^2	Variança
Ω	Espaço amostral

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
1.1. Contexto	1
1.2. Motivação e objetivos	2
1.3. Originalidade, generalidade e utilidade	3
1.4. Organização deste trabalho.....	4
2 CONCEITOS BÁSICOS E REVISÃO BIBLIOGRÁFICA	7
2.1. Detecção de anomalias	7
2.2. Análise de dados.....	8
2.3. Técnicas supervisionadas VS técnicas não supervisionadas	11
2.4. Natureza dos dados.....	12
2.5. Métodos baseados em dados	13
2.5.1. Técnica de clusterização	13
2.5.2. Os histogramas como método para estudo de comportamentos de amostras	19
2.5.3. Métodos estatísticos.....	20
2.5.4. Métodos matemáticos	24
2.5.5. Técnicas de filtragem.....	25
3 FORMULAÇÃO DO PROBLEMA	27

3.1. Definição de requisitos de projeto	27
3.2. Hipóteses do projeto	29
3.3. Técnica de desenvolvimento	32
4 DESENVOLVIMENTO DO ALGORITMO BAbyLO-BR.....	35
4.1. Detecção de anomalias em tubulações hidráulicas.....	35
4.2. FASE I – Técnicas hierárquicas com conhecimento prévio do sistema.....	39
4.2.1. Algoritmo de distância euclidiana	40
4.2.2. Algoritmo rastreável	41
4.2.3. Conclusões	42
4.3. FASE II – Técnicas hierárquicas sem conhecimento prévio do sistema.....	43
4.3.1. Algoritmo de cinzas	43
4.3.2. Algoritmo branco e preto	47
4.4. FASE III – Técnicas particionais sem conhecimento prévio do sistema; implementação do algoritmo BAbyLO-BR.....	53
4.4.1. PASSO 1: Ler dados de entrada	59
4.4.2. PASSO 2: Criar histograma unidimensional para cada parâmetro p	60
4.4.3. PASSO 3: Comprovar cumprimento de critério de parada.....	61
4.4.4. PASSO 4: Aplicar suavização do histograma	63
4.4.5. PASSO 5: Criar <i>clusters</i> unidimensionais	66
4.4.6. PASSO 6: Realizar combinação de resultados unidimensionais	67

4.4.7. PASSO 7 Criar arquivos de saída do algoritmo	69
4.4.8. Resumo das heurísticas utilizadas no algoritmo BAbyLO-BR	71
5 ESTUDOS DE CASO E RESULTADOS	73
5.1. Detecção de Anomalias em Tubulações Hidráulicas	75
5.1.1. Caso 1) Comportamento normal.....	77
5.1.2. Caso 2) Entupimento da tubulação em t=0s	79
5.1.3. Caso 3) Entupimento da tubulação em t=15s	81
5.1.4. Caso 4) Falha do sensor h_1 em t=0s.....	82
5.1.5. Caso 5) Falha do sensor h_1 em t=15s.....	85
5.1.6. Utilidade da aplicação do algoritmo	86
5.2. Identificação de falhas de sensores e atuadores da Plataforma Multi-Missão	87
5.2.1. Objetivos de detecção do algoritmo	88
5.2.2. Descrição da manobra simulada	89
5.2.3. Descrição das falhas simuladas.....	90
5.2.4. Aplicação do algoritmo BAbyLO-BR	93
5.2.5. Utilidade de aplicação do algoritmo	98
5.3. Identificação de corpos d'água em uma imagem Landsat.....	98
5.3.1. Objetivos de identificação do algoritmo.....	99
5.3.2. Descrição dos dados de entrada	100
5.3.3. Aplicação do algoritmo.....	100

5.3.4. Utilidade da aplicação do algoritmo	102
5.4. Comparação com outro algoritmo da área.....	102
5.4.1. Objetivos de identificação do algoritmo.....	103
5.4.2. Descrição dos dados de entrada.....	104
5.4.3. Aplicação do algoritmo.....	104
5.4.4. Utilidade da aplicação do algoritmo	105
5.5. Conclusões.....	106
6 AVALIAÇÃO DO IMPACTO DE UTILIZAÇÃO DO ALGORITMO NO PROCESSO DE DETECÇÃO DE ANOMALIAS	111
7 CONCLUSÕES E TRABALHOS FUTUROS.....	113
REFERÊNCIAS BIBLIOGRÁFICAS	117
APÊNDICE A - APÊNDICE A - PROPAGAÇÃO DE ERRO NA MEDIDA DE DISSIMILARIDADE	125
APÊNDICE B - APÊNDICE B - CÁLCULO DO LIMIAR DE PARADA DO ALGORITMO BABYLO-BR.....	127
APÊNDICE C - APÊNDICE C - OUTROS ESTUDOS DE CASO.....	133
APÊNDICE D - APÊNDICE D - DINÂMICA DA PMM.....	143
APÊNDICE E - COMPORTAMENTOS NORMAL E FALHADOS DA PMM.....	147

1 INTRODUÇÃO

1.1. Contexto

Na atualidade, muitas comunidades de pesquisa trabalham no estudo de sistemas, surgindo a necessidade de analisar e entender os diferentes comportamentos dos mesmos, sejam normais, anômalos ou até falhados.

Para realizar o estudo de comportamentos do sistema há **métodos baseados em modelos** (mais acurados, mas que dependem do conhecimento profundo do sistema e do ambiente) e **métodos baseados em dados** (menos acurados, mas que prescindem do conhecimento profundo do sistema e do ambiente).

Na Área de Controle, o interesse principal é realizar a detecção de anomalias. Nesta área, existe uma grande tendência a modelar os sistemas a serem controlados devido à relativa facilidade de projetar estes modelos e simulá-los. Na hora de se fazer a detecção de anomalias no sistema estudado, tende-se a utilizar técnicas de detecção baseadas em modelos, utilizando os **resíduos** (diferenças entre as respostas do modelo e as respostas reais) para **detectar** as falhas/anomalias do sistema. Precisa-se, então, de um modelo altamente confiável.

Em outras áreas de interesse no INPE, como o Sensoriamento Remoto ou a Meteorologia, o interesse é realizar a análise de dados, e o método de **tentativa-erro** é o mais utilizado. Cada vez mais, objetiva-se trocar esse método por outros mais sistemáticos que ofereçam os mesmos resultados, ou até mais robustos, visando a economia financeira, computacional e de tempo. A presença deste problema é devida, em grande parte, à dificuldade de criar um algoritmo automático eficiente que seja genérico o suficiente para abranger a grande variedade de estudos de caso destas áreas. Isto ocorre porque a maioria dos eventos que acontecem na superfície terrestre são altamente multidimensionais (amostras caracterizadas por vários parâmetros) e, por sua vez, são muito variáveis – ou seja, de difícil identificação de comportamentos bem caracterizados. Além disso, neste tipo de pesquisa, costuma-se realizar o estudo com um alto número de amostras (para descobrir padrões de comportamento anuais). Isto produz

conjuntos muito dispersos que precisam de um extensivo tratamento de dados feito pelo pesquisador, muitas vezes, de forma manual e mediante a técnica de tentativa-erro.

O processo de tratamento de dados manual pode ser dividido em três passos: 1) apontamento de **anomalias** ou valores que são considerados errados (difícil devido à quantidade de amostras a processar); 2) determinação de grupos de comportamento dos dados (frequentemente feito realizando gráficos ou filtrando as amostras pelos parâmetros característicos); e 3) determinação dos **limiares** que definem esses grupos de comportamentos (determinação empírica em estudos de caso específicos com validação prática da sensibilidade ao limiar).

1.2. **Motivação e objetivos**

Existem muitas áreas de trabalho, tanto dentro do INPE quanto na comunidade de pesquisa, dedicadas ao estudo da detecção de anomalias e análises de comportamentos de sistemas; dentro do INPE, podem-se nomear as Áreas de Controle, Sensoriamento Remoto e Meteorologia.

Os métodos utilizados diferem muito em função das áreas e são custosos e específicos para cada estudo de caso (métodos baseados em modelos, métodos de tentativa-erro ou algoritmos de aplicação específica).

Existem muitos trabalhos orientados a realizar a análise de comportamentos e detecção de anomalias ou falhas de sistemas. Especificamente, o trabalho realizado no MIT (LI, 2013) junto com o interesse cada vez maior das empresas da área de análise de comportamentos de sistemas de dispor de novos algoritmos de aplicação genérica, motivou-nos a desenvolver este trabalho.

O objetivo deste trabalho é propor, implementar e testar um algoritmo de clusterização de dados para auxílio à análise de comportamentos de sistemas. Isto permite atenuar as dificuldades dos pesquisadores no processo de tratamento de anomalias e análise de comportamentos mediante a implementação de um **algoritmo genérico** (aplicável a diferentes sistemas multidimensionais, com parâmetros de entrada de comportamento gaussiano e características intrínsecas diferentes) e **não supervisionado** (sem depender

do conhecimento prévio dos resultados esperados), utilizando uma extensão e combinação de métodos baseados em dados.

Este algoritmo visa auxiliar o pesquisador suportando as atividades de 1) determinação de comportamentos, 2) apontamento de anomalias, e 3) definição de limiares.

Para isto foram adotados os passos descritos a seguir:

- I. Analisar e selecionar métodos baseados em dados para a detecção de anomalias e até de falhas de comportamento do sistema pela revisão da literatura;
- II. Propor e desenvolver um novo algoritmo baseado em dados pela extensão e combinação de, pelo menos, dois métodos;
- III. Verificar e validar tal algoritmo pela sua aplicação e comparação a diferentes estudos de caso disponíveis na literatura e no INPE.
- IV. Avaliar os impactos da utilização do algoritmo no processo de análise de comportamentos.

É importante destacar que não é objetivo deste trabalho implementar um algoritmo com a capacidade de indicar quais são os agrupamentos de características normais e quais são as anomalias. Devido à generalidade da aplicação, é necessária a análise detalhada de um especialista na área estudada para identificar o significado dos agrupamentos no contexto e no ambiente em que as amostras foram tomadas.

1.3. Originalidade, generalidade e utilidade

A **originalidade** vem de que, até a data atual, e conforme revisão extensiva bibliográfica (vide Capítulo 2), não foram identificados algoritmos não supervisionados e genéricos de análise de agrupamentos como o desenvolvido neste trabalho.

A **generalidade** vem de que o algoritmo é aplicável a qualquer conjunto de amostras de caráter numérico, independentemente das características intrínsecas do sistema que as produziu, desde que os parâmetros de entrada tenham comportamento gaussiano. No caso de sistemas dinâmicos, faz-se necessário um pré-processamento dos dados.

A **utilidade** vem de ter sido aplicado a diferentes estudos de caso do INPE e da literatura dando resultados coerentes e esperados e até reveladores em todos eles. Ele está em processo de ser introduzido como melhoria da ferramenta DART da empresa Konatus, está sendo testado na detecção de anomalias da plataforma PMM (vide seção 5.2) e permanecerá à disposição na biblioteca do INPE para os alunos futuros e pesquisadores que possam estar interessados na utilização dele.

1.4. Organização deste trabalho

O presente trabalho está organizado como se segue:

Capítulo 2. Conceitos básicos e revisão bibliográfica.

Neste capítulo são apresentados primeiro os conceitos gerais relacionados com detecção de anomalias e análise de comportamentos. Depois, expõem-se os métodos baseados em dados que serão utilizados no desenvolvimento do algoritmo. Por último, são explicados os diferentes modelos de desenvolvimento que podem ser utilizados no ciclo de vida de um projeto.

Capítulo 3. Formulação do problema e abordagens para a sua solução.

Com base nos conceitos básicos estudados no Capítulo 2, neste capítulo definem-se os requisitos do algoritmo implementado neste trabalho. A técnica utilizada para o desenvolvimento do algoritmo (Capítulo 4) também é apresentada nesta seção.

Capítulo 4. Desenvolvimento do algoritmo BAbyLO-BR

Neste capítulo explica-se o algoritmo desenvolvido. Esta explicação será feita seguindo a linha de trabalho da autora, deduções, tomada de decisões, entre outros aspectos relevantes durante o desenvolvimento, até chegar ao algoritmo final. A implementação do algoritmo final utiliza as técnicas baseadas em dados apresentadas no Capítulo 2 e responde aos requisitos definidos no Capítulo 3. O algoritmo final desenvolvido nesta tese é conhecido pelo nome BAbyLO-BR (Behavioral Analysis by Lore – Brasil)

Capítulo 5. Validação do algoritmo pela aplicação em quatro estudos de caso

Neste capítulo é realizada a validação do algoritmo final BAbyLO-BR pela aplicação em quatro estudos de caso. Estes estudos de caso são escolhidos para mostrar validade, aplicabilidade e generalidade do algoritmo desenvolvido. O primeiro estudo de caso é uma amostra de conveniência; ele é simples, pelo que permite inserir falhas e conhecer os resultados esperados. O segundo é uma aplicação na Área de Controle do INPE (neste caso, tem-se também os resultados esperados, podendo assim avaliar a classificação feita pelo algoritmo), o terceiro estudo de caso, uma aplicação em outra área de interesse do INPE; Sensoriamento Remoto (este estudo de caso demonstra a generalidade de aplicação do algoritmo desenvolvido, sendo aplicável não somente em sistemas de controle, mas em qualquer outro sistema) e, por último, no quarto estudo de caso, realiza-se a classificação de uma imagem e compara a classificação obtida com o BAbyLO-BR e com outro algoritmo existente na literatura (ARIFIN, 1974). No final de cada estudo de caso se descreve a utilidade de aplicação do algoritmo; e, para fechar o Capítulo 5, enumeram-se as conclusões de aplicação do algoritmo nos diferentes estudos de caso.

Capítulo 6. Avaliação do impacto de utilização do algoritmo no processo de detecção de anomalias.

Neste capítulo, com base nos resultados obtidos e a experiência ganha com a aplicação do algoritmo em diversos estudos de caso (tanto os do Capítulo 5 quanto os do APÊNDICE C - , são avaliados os impactos da utilização do algoritmo no processo de detecção de anomalias.

Capítulo 7. Conclusões e trabalhos futuros

Neste capítulo descrevem-se as conclusões finais e enumeram-se vários possíveis trabalhos futuros.

No final deste trabalho é disponibilizado um **glossário** com os termos mais importantes deste trabalho e cinco **apêndices** que detalham conceitos ou deduções utilizadas durante o documento.

2 CONCEITOS BÁSICOS E REVISÃO BIBLIOGRÁFICA

Neste capítulo são apresentados cinco conceitos básicos, a saber, detecção de anomalias, análise de comportamentos de dados, natureza de dados em sistemas, métodos baseados em dados e modelos de ciclo de vida de um projeto de desenvolvimento.

2.1. Detecção de anomalias

A detecção de anomalias é a capacidade de achar padrões de comportamento que não correspondem ao considerado como esperado (comportamento normal) (CHANDOLA, 2009).

Conforme a classificação feita por Chandola (2009) existem anomalias de vários tipos:

- Anomalias pontuais: amostra individual cujo padrão de comportamento difere do conjunto total de amostras.
- Anomalias contextuais: conjunto de amostras que são consideradas anômalas pelo contexto onde estão acontecendo, sendo que uma mudança no contexto faria com que elas não fossem mais anômalas.
- Anomalias coletivas: conjunto de amostras cujo padrão de comportamento é considerado anormal em relação ao resto de amostras.

Devido à variedade de anomalias, todo classificador (algoritmo de classificação) capaz de distinguir entre anomalias e comportamento normal em um sistema deve ser treinado (seja mediante aprendizado prévio, seja mediante limiares definidos na definição do algoritmo, seja mediante definição de hipóteses, etc.) no espaço característico específico desse sistema (CHANDOLA, 2009).

O fato de definir o comportamento normal de um sistema faz com que o algoritmo perca generalidade e não seja aplicável a outro sistema de características diferentes.

Como um dos objetivos do algoritmo aqui proposto é diferenciar as anomalias dos comportamentos normais na classificação final (detecção de anomalias), revisaram-se as **técnicas de detecção de anomalias**, realizando a leitura da pesquisa feita por Chandola (2009).

Aquela pesquisa enumera as técnicas para realizar a detecção de anomalias, sendo 1) redes neurais, 2) redes bayesianas, 3) máquinas de vetores de suporte (SVM, do Inglês *Support Vector Machines*), 4) técnicas baseadas em regras, 5) técnicas baseadas no vizinho mais próximo e 6) técnicas de análise de agrupamentos.

Todas as técnicas de detecção de anomalias realizam uma fase de aprendizado prévio no ambiente de trabalho do sistema a ser estudado para “aprender” qual é o modo normal de comportamento e poder assim diferenciá-lo das anomalias. Este tipo de técnicas é conhecido como **técnicas supervisionadas**.

Porém, a técnica de análise de agrupamentos é, por natureza, uma técnica não supervisionada, embora esteja sendo utilizada também como técnica supervisionada para a detecção de anomalias (assume que as anomalias não formam agrupamentos, ficando isoladas uma vez feita a classificação) (CHANDOLA, 2009 e PADHY, 2012).

2.2. Análise de dados

A detecção de anomalias é uma particularização de um problema mais abrangente; a análise de dados.

Enquanto a detecção de anomalias refere-se à diferenciação entre o comportamento padrão ou normal e o comportamento anômalo, a análise de dados abrange todo o processo relacionado com o fato de tratar os dados para dotá-los de identificação e razão de ser no mundo racional.

Como descrito na introdução, o processo de tratamento de dados (análise de dados) é realizado como se segue:

- 1) Identificação de anomalias
- 2) Determinação de comportamentos
- 3) Definição de limiares para caracterizar matematicamente esses comportamentos.

Devido à natureza multidimensional do algoritmo que é desenvolvido neste trabalho, estudaram-se técnicas para realizar a análise de dados multidimensionais, realizando a leitura de Hair (2007)¹.

As técnicas para realizar a análise de dados multidimensionais são:

Análise de componentes principais e fatores comuns: (PCA, do termo Inglês *Principal Component Analysis*). Refere-se à técnica que consiste em encontrar a forma de condensar as informações contidas em um número de parâmetros originais em um conjunto menor de parâmetros finais, com perda mínima de informações.

Regressão múltipla: considera-se uma amostra formada por vários parâmetros métricos, sendo que um deles é dependente da variação dos outros. Esta técnica é utilizada quando se deseja conhecer e prever a variação do parâmetro dependente em função da variação de um ou mais parâmetros independentes.

Análise discriminante múltipla: (MDA, do termo Inglês *Multiple Discriminant Analysis*). Considera-se uma amostra formada por um parâmetro dependente não métrico categórico (por exemplo, alto/médio/baixo) e um ou mais parâmetros independentes métricos. Esta técnica procura conhecer a probabilidade que uma amostra tem de pertencer a uma categoria determinada do parâmetro não métrico em função dos parâmetros métricos que a caracterizam.

Análise multivariada de variância e covariância: (MANOVA, do termo Inglês *Multivariate ANalysis Of VAriance and covariance*). Esta técnica estuda as relações entre diversos parâmetros não métricos e dois ou mais parâmetros métricos de uma amostra.

¹ A terminologia utilizada por Hair (2007) difere da definida para este trabalho, sendo que Hair utiliza o termo 'variável' para se referir a cada um dos parâmetros que formam uma amostra. Neste trabalho modificaram-se os conhecimentos extraídos de Hair (2007) para ficar conforme a terminologia deste texto.

Análise conjunta: técnica emergente que consiste em, considerando uma amostra com vários parâmetros não métricos, realizar uma avaliação geral da amostra, de forma que o pesquisador consiga determinar a importância de cada parâmetro (não métrico) e até qual dos valores possíveis de cada parâmetro é mais destacável. Por exemplo, considera-se um carro (amostra) com os parâmetros (não métricos) cor, preço e qualidade; cada um destes parâmetros tem dois possíveis valores, por exemplo, para preço pode-se ter alto e baixo. Com esta técnica, o fabricante (pesquisador) consegue avaliar quais parâmetros são mais importantes e até a relação entre eles com base nas opiniões gerais dos clientes.

Correlação canônica: é uma generalização da regressão múltipla; o objetivo é correlacionar simultaneamente diversos parâmetros dependentes métricos com diversos parâmetros independentes métricos.

Análise de agrupamentos: técnica analítica para desenvolver subgrupos significativos de amostras. O objetivo é classificar um conjunto de amostras unidimensionais ou multidimensionais em grupos, com base nas similaridades entre elas. Esta técnica envolve três passos: 1) definição de como será feita a medida da similaridade, 2) classificação de amostras em grupos e 3) determinação do perfil de cada grupo para identificá-lo.

Escalonamento multidimensional: o objetivo desta técnica é transformar o julgamento do consumidor em distâncias apresentadas em um espaço multidimensional de forma que itens julgados como semelhantes fiquem mais próximos do que itens julgados como opostos.

Análise de correspondência: esta técnica fornece uma representação multivariada de interdependência para dados não métricos que não é possível fazer com outras técnicas.

Modelos lineares de probabilidades: São uma combinação de regressão múltipla e análise discriminatória múltipla, enquanto que o parâmetro dependente é não métrico e os parâmetros independentes podem ser métricos ou não métricos.

Modelagem de equações estruturais: consiste em formar modelos computacionais que relacionam os parâmetros dependentes com os independentes. O pesquisador se vale de teoria, experiência prévia ou outras orientações para saber quais parâmetros independentes prevêm cada parâmetro dependente.

De todas as técnicas acima explicadas, a única técnica que se aplica ao conjunto de amostras total, sem ter que predefinir parâmetros dependentes e independentes e sem ter que realizar um estudo prévio do espaço multidimensional de parâmetros que caracterizam as amostras é a técnica de análise de agrupamentos.

2.3. Técnicas supervisionadas VS técnicas não supervisionadas

Conforme descrito nas seções anteriores, os métodos de tratamento de dados podem ser separados em dois grandes grupos:

- 1) Técnicas supervisionadas ou baseadas em modelos; estas técnicas utilizam o conhecimento prévio do sistema para classificar os resultados do mesmo. Este conhecimento prévio pode resultar em um modelo do sistema ou pode resultar em um aprendizado baseado em dados de saída do sistema cujo comportamento é conhecido. Chamaram-se de MAPs (Técnicas de Aprendizado Prévio).
- 2) Técnicas que não precisam de aprendizado prévio (não supervisionadas): estas técnicas utilizam os dados de saída do sistema sem necessidade de conhecer o comportamento dos mesmos. Chamaram-se de MNAPs (Técnicas de Não Aprendizado Prévio).

Baseada na caracterização anterior, a seguinte lista enumera as diferenças entre as técnicas MAPs e MNAPs. A nomenclatura MAP/MNAP foi utilizada unicamente nesta seção para facilitar a leitura das seguintes linhas:

- MAPs precisam de conhecimento profundo prévio do sistema para serem implementadas. MNAPs não precisam de nenhum conhecimento do sistema, só requerem definir a que tipo de sistema vão ser aplicadas.

- MAPs são desenvolvidas para um sistema específico a ser estudado, MNAPs são desenvolvidas de forma genérica para aplicação em diversos sistemas.
- MAPs, quando bem modeladas, são mais acuradas. MNAPs, por serem elas de aplicação mais genérica, podem resultar menos acuradas.
- MAPs sabem discernir os comportamentos normais dos comportamentos anômalos. MNAPs separam os diferentes comportamentos, mas não sabem especificar qual é o normal e qual é o anômalo.
- MAPs requerem maior esforço para serem implementadas, por serem elas mais específicas. MNAPs requerem menor esforço para serem modeladas.
- MAPs, uma vez desenvolvidas, não são reaproveitáveis 100% para aplicação em outros sistemas. MNAP, devido à sua independência com os dados de entrada, podem ser reaproveitáveis até 100% para outros sistemas.

2.4. Natureza dos dados

Neste trabalho trabalhar-se-á com modelos baseados em dados; daí a necessidade de entender a natureza dos dados de saída de um sistema genérico que servirão como dados de entrada para o algoritmo BAbyLO-BR.

Com base na leitura de Chandola (2009), Li (2013) e Hair (2007), determina-se a definição do conceito de dados de entrada e as características mais importantes deles.

- 1) Os dados de entrada são um conjunto de instâncias (amostras) caracterizadas por atributos (parâmetros).
- 2) As amostras podem ser unidimensionais ou multidimensionais dependendo do número de parâmetros que as caracterizem.
- 3) Os parâmetros podem ter naturezas diferentes: métricos (por exemplo, binários ou contínuos) e não métricos (qualitativos ou categóricos).
- 4) A relação entre duas amostras pode ser sequencial (a ordem importa), espacial (a posição no espaço importa), espaço-temporal (importa ordem e posição), qualitativa (características similares importam).
- 5) Os dados de entrada podem ter comportamento estático (é esperado que todas as amostras tenham o mesmo valor, sendo que, valores diferentes referem-se a

comportamentos diferentes) ou dinâmico (as amostras evoluem assumindo diferentes valores, mas referindo-se ao mesmo comportamento).

2.5. Métodos baseados em dados

Nas seguintes subseções são repassados de forma teórica os métodos baseados em dados utilizados na implementação do algoritmo BAbyLO-BR.

2.5.1. Técnica de clusterização

A técnica de clusterização (chamada também de **técnica de análise de agrupamentos** nas seções anteriores) consiste em classificar amostras em grupos (*clusters*) em função das características próprias delas, de forma que esses grupos tenham significado no contexto em que foram criados (PANG-NING, T. 2006).

Obs: neste trabalho usar-se-á o termo Inglês *cluster* para se referir aos grupos formados pela aplicação da técnica de clusterização.

Para realizar a clusterização, precisa-se definir uma medida de dissimilaridade. Esta medida caracteriza a distância entre amostras e é utilizada para agrupar as amostras. A medida de dissimilaridade é, em muitos casos, uma distância entre amostras que se representa geralmente com uma matriz de proximidade que relaciona cada par de amostras com a distância entre elas (PANG-NING, 2006). Existem várias distâncias que são utilizadas habitualmente nos algoritmos de clusterização (SWAGATAM, 2008) como, por exemplo, a distância euclidiana, utilizada neste trabalho para estudo prévio do problema. Pode acontecer que a medida de dissimilaridade não seja um cálculo da distância, mas uma técnica, conjunto de deduções, etc. e não possa ser representada por uma matriz de dissimilaridade. Isto depende das características do método de clusterização utilizado (Tabela 2-1).

A Tabela 2-1 resume estas características e foi construída com base nas leituras de Pang-Ning, T (2006), Chandola (2009), Swagatam (2008) e Dubes (1988), entre outras.

Tabela 2-1. Classificação dos métodos de clusterização.

Tipos de clusterização	Comparação
<p>Exclusiva</p> <p>Sobreposta</p> <p>Fuzzy</p>	<p>Supõe que cada dado pode pertencer unicamente a um <i>cluster</i>.</p> <p>É utilizada em classificações onde os objetos podem pertencer a mais de um <i>cluster</i> como, por exemplo, os esportes praticados pelos alunos de uma escola.</p> <p>Designa cada dado em um <i>cluster</i> com uma probabilidade entre 0 e 1 dependendo da afinidade desse dado nesse <i>cluster</i>.</p>
<p>Intrínseca</p> <p>Extrínseca</p>	<p>Utiliza unicamente a medida de dissimilaridade para realizar a classificação. É chamada também de não-supervisionada.</p> <p>Utiliza informações de aprendizado em conjunto com a medida de dissimilaridade. É chamada também de supervisionada.</p>
<p>Hierárquica</p> <p>Particional</p>	<p>Supõe a classificação em partições sequenciais segundo um critério de proximidade entre amostras. O resultado é uma representação em forma de árvore (dendograma).</p> <p>Supõe a classificação dos dados em diferentes <i>clusters</i> diretamente desde o conjunto inicial, sem fases de criação de sub-<i>clusters</i>, atendendo a um critério escolhido.</p>
<p>Completa</p> <p>Parcial</p>	<p>Classifica todos os dados dentro de algum <i>cluster</i>.</p> <p>Supõe que existem dados que não pertencem a nenhum <i>cluster</i> específico, estes dados representam anomalias ou ruídos.</p>
<p>Aglomerativa</p>	<p>Começando com as amostras individualmente separadas em <i>clusters</i>, estas são agrupadas iterativamente em função de um critério de proximidade.</p>

Continua

Tabela 2.1 - Conclusão

Tipos de clusterização	Comparação
Divisiva	Começando com todas as amostras em um único <i>cluster</i> , estas são separadas iterativamente em função de um critério de proximidade.

Na Figura 2-1 se mostra o caminho definido em Dubes (1988) para escolher um método de clusterização em função dos gêneros definidos na Tabela 2-1.

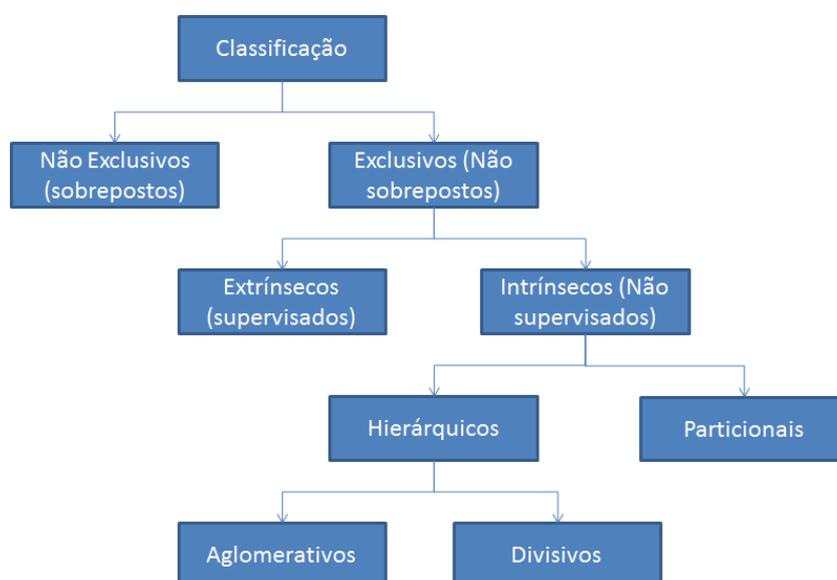


Figura 2-1. Gêneros de classificação de técnicas de clusterização.

Fonte: Adaptação de Dubes (1988).

A essência do método de clusterização é a classificação intrínseca, onde não existe supervisão nem aprendizado prévio para realizar agrupamento das amostras (CHANDOLA, 2009; PADHY, 2012).

Dentro dos métodos intrínsecos, deve-se escolher entre trabalhar com técnicas hierárquicas ou particionais.

No caso de métodos hierárquicos, existe ainda a diferenciação em dois subgrupos; métodos hierárquicos aglomerativos e divisivos. No primeiro, começa-se com cada amostra pertencendo a um *cluster* diferente e, utilizando a medida de dissimilaridade

(critério de proximidade), em cada iteração, as amostras mais próximas são agrupadas no mesmo *cluster*. As iterações continuam até atingir um número de *clusters* pré-definido pelo usuário ou até aglomerar todas as amostras em um *cluster*. No segundo método (divisivo) todas as amostras começam no mesmo *cluster* e o processo é aplicado ao contrário do caso anterior. Em ambos os casos, obtém-se como resultado da estratégia um gráfico conhecido como **dendograma**, que é a representação gráfica da evolução iterativa da agrupação das amostras (Figura 2-2).

No caso de métodos particionais, não é aplicada uma medida de dissimilaridade, mas uma técnica que produz diretamente a classificação final das amostras. A aplicação teórica de métodos particionais seria conforme se segue: 1) seleciona-se o critério para criação de *clusters*, 2) aplica-se este critério para todas as possíveis partições de dados e 3) escolhe-se a partição que otimiza o critério. Existem duas dificuldades na aplicação desta técnica; 1) a dificuldade de expressar de forma matemática as noções intuitivas que se tem do conceito de *cluster* (definição do critério) e 2) a impossibilidade de representar ou reproduzir todas as partições possíveis para avaliar o critério escolhido. (Dubes, 1988).

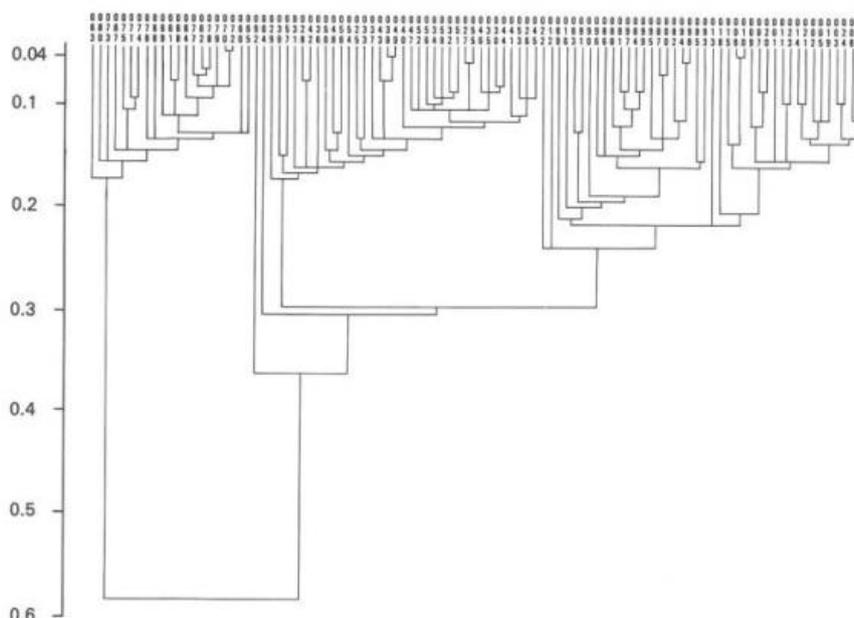


Figura 2-2. Exemplo de dendograma.

Fonte: Dubes (1988).

Dubes (1988) enumera as diferenças entre técnicas hierárquicas e técnicas particionais; a Tabela 2-2 resume as características das duas técnicas.

Tabela 2-2. Comparativo de técnicas hierárquicas e técnicas particionais.

Técnicas hierárquicas	Técnicas particionais
Classificação criada de forma sequencial, por iterações.	Classificação criada atendendo a uma técnica em uma única iteração.
Dendograma como resultado final que representa a evolução da classificação: o analista pode estudar a evolução mediante o dendograma.	Resultado final: Classificação final. Não existem classificações intermediárias.
A implementação de dendogramas é impraticável com grande número de amostras (rastreadabilidade é pesada).	Técnicas particionais são mais leves e mais adaptadas a serem aplicadas em grandes conjuntos de amostras.
Técnicas hierárquicas são populares nas ciências biológicas, sociais e de comportamento, onde é necessário conhecer a taxonomia dos dados.	Técnicas particionais são aplicadas frequentemente na área de engenharia onde é interessante obter partições exclusivas.

Existe uma grande variedade de métodos de clusterização e, em consequência, algoritmos de clusterização. É importante definir a diferença entre método (ou técnica) e algoritmo (DUBES, 1988);

Método de clusterização: é uma estratégia geral para agrupar padrões em *clusters* em função da medida de dissimilaridade escolhida.

Algoritmo de clusterização: é um programa computacional que implementa uma estratégia ou método de clusterização incorporando várias **heurísticas**.

As heurísticas são o conjunto de decisões empíricas tomadas durante a fase de desenvolvimento para transformar uma técnica teórica em uma aplicação prática. Elas caracterizam o algoritmo e o fazem único.

Nas leituras Ping-Ning (2006), Chandola (2009), Dubes (1988), Ankerst (1999), Budalakoti (2006) e Castro (2001) podem ser encontradas definições, explicações e implementações de métodos e algoritmos de clusterização.

A dificuldade achada em todas as leituras realizadas pela autora é sempre a mesma; definir o **critério de parada** (ANKERST, 1999). O critério de parada determina em que iteração/momento pode ser considerado que se chegou à solução ótima de clusterização. O critério de parada é chamado de diferentes formas dependendo do método de clusterização aplicado; pode ser feita a referência a ele como ‘número de *clusters* finais desejados’, ‘tamanho final do *cluster* desejado’, ‘distância máxima desejada entre *clusters*’, etc., mas, independentemente do nome associado, uma coisa sempre é comum: esse valor deve ser definido pelo usuário e influi fortemente na classificação final (ANKERST, 1999). Ankerst (1999) formula este problema explicando que não existem parâmetros globais que consigam caracterizar as características intrínsecas de qualquer conjunto de dados.

Para finalizar, enumeram-se as seguintes afirmações feitas em relação aos métodos de clusterização (Dubes, 1988):

1. Não existe critério único e ótimo para particionar os dados, pois não existe uma definição precisa do conceito de *cluster*. Cada critério de clusterização impõe uma estrutura aos dados que pode ou não coincidir com a estrutura real dos mesmos.
2. Só uma pequena parte de critérios de clusterização pode ser entendida tanto matematicamente quanto intuitivamente.
3. Os métodos de clusterização têm o mau costume de criar *clusters* onde não existem comportamentos naturais diferenciados dos dados; por isso, as classificações produzidas devem ser estudadas para conferi-las.
4. A estrutura hierárquica formada no dendograma pode mudar de forma notável se for mudada a forma de medir a dissimilaridade entre amostras (Matriz de proximidade).
5. É difícil fazer uma comparação teórica entre diferentes algoritmos de clusterização, pois os algoritmos de clusterização são habitualmente impossíveis de modelar matematicamente e, portanto, difíceis de comparar.

6. De qualquer forma, o objetivo da técnica de clusterização é investigar a estrutura dos dados; por isso, a imposição de qualquer critério prévio pode não ser apropriada.

2.5.2. Os histogramas como método para estudo de comportamentos de amostras

A utilização de histogramas como método para estudo de comportamento de amostras unidimensionais (com um único parâmetro característico) é uma técnica de clusterização particional utilizada há mais de 30 anos (PING-NING, 2006).

Esta técnica consiste na construção de um histograma que particiona o conjunto de amostras em comportamentos onde os máximos dos histogramas representam centróides dos *clusters* e os vales representam divisão entre dois *clusters* (PING-NING, 2006).

Construir um histograma consiste em dividir o espaço amostral em janelas onde o valor de cada janela corresponde ao número de amostras que se encontram dentro dos limites dessa janela.

Matematicamente, um histograma unidimensional h de um parâmetro p com n_a amostras pode ser definido como se segue:

$$\begin{aligned} h: S &\rightarrow \mathbb{N} \\ s &\rightarrow h(s) \end{aligned} \quad (2.1)$$

sendo $S = \{s_1, \dots, s_{n_s}\}$ o conjunto de janelas do histograma e n_s o número total de janelas. O comprimento das janelas l_{slot} é

$$l_{slot} = \frac{\max(p) - \min(p)}{n_s} \quad (2.2)$$

Cada janela s_i representa um intervalo tal que

$$s_i = [\min(p_j) + l_{slot}(i - 1), \min(p_j) + l_{slot}(i)] \quad (2.3)$$

Assim, a função h associa a cada intervalo s_i um número natural correspondente ao número de amostras de p dentro desse intervalo.

É importante notar que a eficácia desta técnica depende de vários **fatores de êxito** (DUBES, 1988), a saber, 1) o tamanho da discretização, i.e., o número de janelas n_s criadas no histograma (janelas muito pequenas produzem uma estimação da densidade ‘ruidosa’; e janelas muito grandes produzem uma estimação da densidade ‘suavizada’ e 2) o procedimento para localizar picos e vales no histograma deve ser realizado levando em conta uma vizinhança de tamanho conhecido. Segundo Dubes (1988), estes fatores são difíceis de controlar quando se trabalha em várias dimensões.

Existem vários trabalhos utilizando métodos baseados em histogramas, como Arifin (1974) e Chhikara (1979).

Arifin (1974) aplica estratégias de clusterização hierárquica para definir o limiar que separa dois comportamentos (*clusters*) no histograma mediante um dendograma, mas ele precisa definir manualmente o número de *clusters* iniciais para construir o dendograma. Arifin (1974) aplica seu algoritmo unicamente ao tratamento de imagens em escala de cinzas, trabalhando unidimensionalmente com um parâmetro único; o nível de cinza, e diferenciando dois comportamentos (branco e preto).

Chhikara (1979) propõe um algoritmo em que o usuário recebe o histograma e ele mesmo deve, de forma manual, escolher o limiar que separa dois comportamentos no histograma. A técnica é aplicada a um único estudo de caso.

2.5.3. Métodos estatísticos

Seja $X: \Omega \rightarrow \mathbb{R}$ uma variável aleatória onde Ω é o espaço amostral e $\{x_1, \dots, x_N\}$ N amostras.:

A **média** μ_X ou esperança matemática $E[X]$ de X é uma medida que aponta o valor ao redor do qual se concentram o maior número de amostras de uma distribuição. Ela é calculada como:

$$\mu_X = E(X) = \frac{1}{N} \sum_{i=1}^N x_i \quad (2.4)$$

A **variança** σ_X^2 de uma variável aleatória X é uma medida que representa a dispersão ou espalhamento das amostras em relação à média μ_X (MORETTIN, 1995). Ela é calculada como:

$$\sigma_X^2 = E((x - \mu_X)^2) \text{ com } \mu_X = E(X) \quad (2.5)$$

Assim, como pode ser deduzido com a Equação (2.5), uma variança alta representa uma grande dispersão das amostras em relação à média μ_X e uma variança pequena representa amostras pouco espalhadas em relação à média μ_X (MORETTIN, 1995).

A Figura 2-3 mostra um exemplo desta dedução: na Figura 2-3a, a variança é pequena, pois as amostras estão pouco espalhadas em relação à media; Por outro lado, na Figura 2-3b a variança é grande pois as amostras não estão agrupadas ao redor da média.

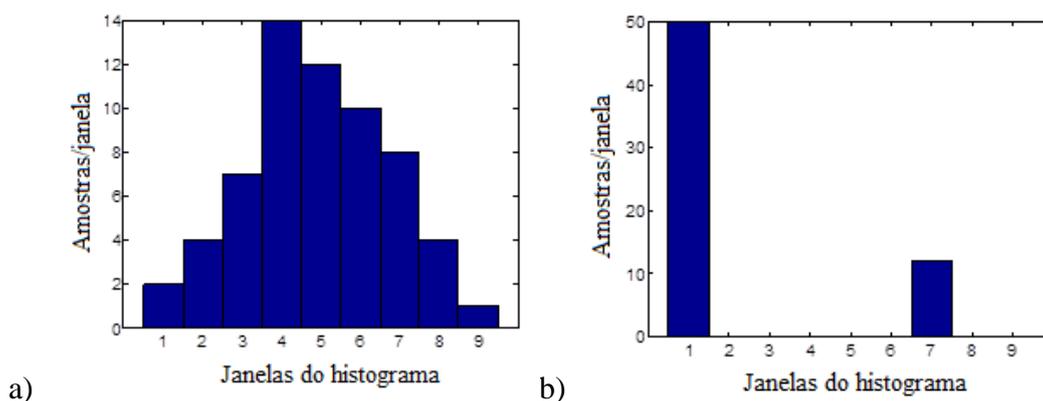


Figura 2-3. a) Exemplo de variável aleatória com baixa dispersão. b) Exemplo de variável aleatória com alta dispersão².

A variável aleatória X tem **distribuição gaussiana de probabilidade** se sua função densidade de probabilidade $f(x)$ é representada pela Figura 2-4:

² Neste documento são utilizadas representações de histogramas em múltiplas ocasiões. Em toda representação de histograma, considera-se que o eixo x representa as janelas do histograma e o eixo y representa o número de amostras pertencentes a essa janela (conforme equações (2.1), (2.2) e (2.3)). Assim, todos os histogramas deste trabalho são apresentados sem legenda dos eixos, sendo que eles seguem a legenda aqui descrita.

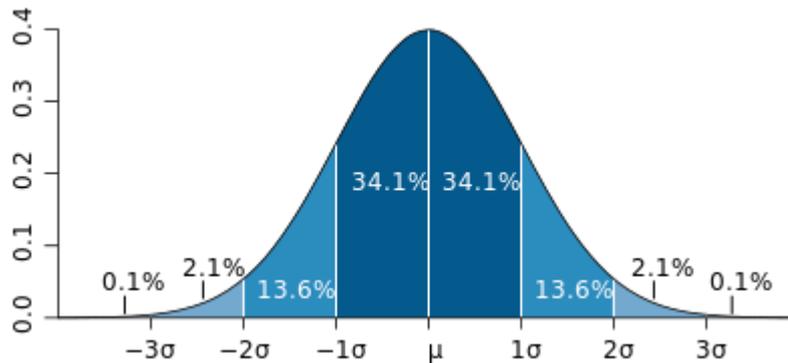


Figura 2-4. Distribuição gaussiana.

As principais características desta função são:

- 1) o ponto máximo de $f(x)$ é ponto $X = \mu$,
- 2) os pontos de inflexão da função são: $X = \mu + \sigma$ e $X = \mu - \sigma$,
- 3) a curva é simétrica em relação a μ , e
- 4) a média de X é μ e sua variância é σ^2 .

Para representar este tipo de distribuição é utilizada a seguinte notação:

$$X: N(\mu, \sigma^2) \quad (2.6)$$

onde X representa uma distribuição gaussiana com média μ e variância σ^2 .

Esta distribuição também é chamada de distribuição normal.

O **Teste de Hipóteses** (ISSERMAN, 2006 e MORETTIN, 2000) estuda os procedimentos para aceitar ou rejeitar uma hipótese estatística emitida por uma população. Para escolher a hipótese certa, precisa-se estudar toda a população. Como normalmente isto é inviável (devido ao tamanho da população, etc.), procede-se a estudar uma ou várias amostras representativas da população e tomar uma decisão sobre a verdade ou falsidade desta hipótese. As dificuldades deste processo estão em: 1) delimitar até que ponto cada amostra é representativa da população 2) estimar qual a probabilidade de que a decisão tomada seja certa.

Esta técnica supõe que a distribuição de cada variável de interesse é normal; $N(\mu, \sigma^2)$ conforme Figura 2-4 e equação (2.6).

A hipótese que se deseja contrastar é conhecida como hipótese nula H_0 , que é a probabilidade de que a amostra pertença a essa distribuição normal $N(\mu, \sigma^2)$. Ao contrário, se esta hipótese for rejeitada, diz-se que está acontecendo a hipótese alternativa ou H_1 que é a probabilidade de que essa amostra não pertença à $N(\mu, \sigma^2)$.

Um classificador pode então decidir atendendo à seguinte tabela (Tabela 2-3):

Tabela 2-3 Possíveis tomadas de decisão de um classificador.

		Hipótese detectada	
		H_0	H_1
Hipótese real	H_0	Correto / Verdadeiro Negativo (VN)	Erro tipo I / Falso Positivo (FP)
	H_1	Erro tipo II / Falso Negativo (FN)	Correto / Verdadeiro Positivo (VP)

O erro de tipo I é conhecido também como Falso Alarme (*False Alarm* em Inglês), pois representa o caso em que é detectada uma condição anômala da amostra sendo esta normal. O erro de tipo II é conhecido como Perda de Alarme (*Missing Alarm* em Inglês), pois o comportamento anômalo da amostra não foi detectado. Na teoria da detecção, o erro de tipo I é o chamado de crítico porque produz custos desnecessários.

Frente a estas possibilidades, definem-se dois conceitos:

- **Nível de Significância:** A probabilidade de se rejeitar a hipótese verdadeira sendo esta verdadeira, i.e., a probabilidade de cometer um erro do tipo I ou Falsos Positivo. Conhece-se como α .

$$\alpha = P\left(\frac{H_1}{H_0}\right) \quad (2.7)$$

- **Potência de Contraste:** A probabilidade de rejeitar a hipótese nula e que realmente era falsa, i.e. a probabilidade de Verdadeiros Negativos; é chamada de $1 - \beta$. Se a Potência de Contraste estiver perto da unidade, então isto quer dizer que as amostras que pertencem à hipótese H_0 são bem diferenciadas pelo classificador das que pertencem à H_1 . Também, se o valor de $1 - \beta$ estiver perto da unidade, β estará perto de zero e o número de erros de tipo II (β) será pequeno.

$$1 - \beta = P\left(\frac{H_1}{H_1}\right) \quad (2.8)$$

2.5.4. Métodos matemáticos

Neste trabalho foram consideradas as seguintes deduções matemáticas (WEIR, 2009):

Primeira derivada

Seja $f: \mathbb{R} \rightarrow \mathbb{R}$ uma função contínua em $[a, b]$ (intervalo fechado) e derivável em (a, b) (intervalo aberto):

- 1) $\forall x \in [a, b], f'(x) > 0 \Leftrightarrow f$ é estritamente crescente em (a, b)
- 2) $\forall x \in [a, b], f'(x) < 0 \Leftrightarrow f$ é estritamente decrescente em (a, b)

Considerando 1) e 2), $\forall c \in (a, b)$:

- 3) Se $f'(c) = 0 \Rightarrow c$ é um ponto crítico de $f(x)$.
- 4) Supõe-se que c é um ponto crítico de $f(x)$, i.e., $f'(c) = 0$ então:
 - a. Se $f''(c) > 0 \Rightarrow f$ têm um mínimo local em c .
 - b. Se $f''(c) < 0 \Rightarrow f$ têm um máximo local em c .
- 5) $f(x)$ tem mínimo local em c se $f(c) \leq f(x_0)$ sendo x_0 uma vizinhança de c .
- 6) $f(x)$ tem máximo local em c se $f(c) \geq f(x_0)$ sendo x_0 uma vizinhança de c .

Considerando estas definições pode-se concluir que:

- 7) $\forall c \in (a, b)$ a mudança no sinal da derivada f' em c indica a existência de um extremo nesse ponto, sendo esse extremo um mínimo local quando o sinal muda

de negativo para positivo e um máximo local quando o sinal muda de positivo para negativo.

A conclusão 7) é chamada de **corolário da primeira derivada**.

Segunda derivada:

Sensibilidade à variação:

Quando uma pequena variação em x provoca uma mudança significativa no valor $f(x)$, diz-se que a função f é relativamente sensível à variação em x . A segunda derivada f'' é uma medida dessa sensibilidade.

O conceito de medida de sensibilidade usando a segunda derivada será chamado de **corolário da segunda derivada**.

2.5.5. Técnicas de filtragem

A filtragem digital consiste na aplicação de técnicas de transformação (operadores-máscaras) com o objetivo de corrigir, suavizar ou realçar determinadas características de uma imagem dentro de uma aplicação específica (THOMÉ, 2014).

Os métodos de filtragem de imagens baseiam-se em técnicas no domínio da frequência (operam sobre a Transformada de Fourier da imagem original) ou no domínio espacial (operam diretamente sobre os pixels da imagem na sua forma original) (THOMÉ, 2014). Neste trabalho serão aplicadas técnicas no domínio espacial.

O termo domínio espacial refere-se ao agregado de pixels que compõem uma imagem e métodos no domínio espacial são procedimentos que operam diretamente sobre estes pixels. As funções de processamento no domínio espacial podem ser expressadas como (GONZALEZ, 2000):

$$g(x, y) = T[f(x, y)] \quad (2.9)$$

onde (x, y) são as coordenadas de um pixel da imagem, f a função que define a imagem de entrada (por exemplo o nível de cinza ou as cores), g a função que define a imagem processada e T um operador sobre f , definido sobre alguma vizinhança de (x, y) .

A abordagem principal para definir uma vizinhança consiste em usar uma subimagem quadrada ou retangular centrada em (x, y) . O centro da subimagem é movido pixel a pixel percorrendo toda a imagem. Esta subimagem é chamada comumente de máscara. Um exemplo de máscara é mostrado na Figura 2-5:

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Figura 2-5. Exemplo de máscara 3x3.

Baseando-se nestes conceitos, a abordagem básica de filtragem espacial consiste em somar os produtos entre os coeficientes da máscara e as intensidades dos pixels sob a máscara numa posição específica da imagem. Assim, a resposta da máscara linear é conforme a Equação (2.9) e aplicando a máscara da Figura 2-5, o valor processado do pixel i localizado em (x_i, y_i) é:

$$g(x_i, y_i) = w_1 f(x_i - 1, y_i - 1) + w_2 f(x_i, y_i - 1) + \dots + w_9 f(x_i + 1, y_i + 1) \quad (2.10)$$

Considerando que as intensidades dos pixels $f(x_i - 1, y_i - 1)$ até $f(x_i + 1, y_i + 1)$ correspondem a z_1, \dots, z_9 , e que a máscara é posicionada de forma que o peso w_5 esteja situado no pixel de coordenadas (x_i, y_i) , então a equação (2.10) pode ser reescrita como:

$$g(x_i, y_i) = w_1 z_1 + w_2 z_2 + \dots + w_9 z_9 \quad (2.11)$$

Nota-se que $f(x_i, y_i)$ representa o valor do pixel i da imagem inicial. Para uma imagem em preto e branco, $f: \mathbb{R}^2 \rightarrow \mathbb{R}$ é uma função com valores escalares que representa o nível de cinza; no caso de uma imagem colorida, $f: \mathbb{R}^2 \rightarrow \mathbb{R}^3$ é uma função com valores vetoriais que representa um conjunto RGB (proporções de vermelho, verde e azul).

3 FORMULAÇÃO DO PROBLEMA

A seguir, apresenta-se o problema a resolver em três seções; requisitos (3.1), hipóteses (3.2) e técnica (3.3).

3.1. Definição de requisitos de projeto

Nesta seção são mostrados os requisitos iniciais definidos para o desenvolvimento do algoritmo BAbyLO-BR. Durante o desenvolvimento do projeto, surgirão novos requisitos derivados do aprendizado e da evolução do trabalho.

- 1) Com o objetivo de não perder a generalidade do algoritmo desenvolvido neste trabalho, a proposta é realizar as diferenciações entre comportamentos normais ou esperados e anomalias, mas não determinar qual dos dois comportamentos corresponde à anomalia. Desta forma, o usuário final poderá estudar os comportamentos reconhecidos pelo algoritmo e definir o que é considerado como comportamento inesperado e o que é considerado como comportamento normal. Por este motivo, este algoritmo pretende não somente permitir a detecção de anomalias em sistema de controle, mas também ser capaz de fazer análise de comportamentos de sistema. Como resultados do estudo das técnicas de detecção de anomalias (Capítulo 2), surgiram os seguintes requisitos para definição do BAbyLO-BR:

Req_1: o algoritmo BAbyLO-BR deve agrupar os dados em função de suas características.

Req_2: o algoritmo BAbyLO-BR deve agrupar em coleções diferentes os dados de comportamentos diferentes.

Obs: Não é necessário que o algoritmo BAbyLO-BR indique qual é o comportamento normal e qual é o anômalo.

- 2) Sabendo da diversidade de dados de entrada possíveis, é necessário definir a natureza dos dados de entrada que será aceita pelo algoritmo BAbyLO-BR.

Req_3: o algoritmo BAbyLO-BR deve ser implementado para receber como dados de entrada a matriz $A \in \mathcal{M}^{n_a \times n_p}$ onde n_a é o número de amostras e n_p é o número de parâmetros de cada amostra.

Req_4: todos os parâmetros que formam os dados de entrada devem ser de natureza quantitativa (métricos) e pertencer ao domínio dos números reais.

Req_5: cada uma das amostras deve possuir informações de todos os parâmetros que a formam.

Req_6: os dados de entrada devem ter comportamento estático.

Obs: os sistemas estudados poderão ser estáticos ou dinâmicos. No caso de sistemas estáticos será feita uma estatística no tempo (temporal) para criar as amostras de entrada do algoritmo, enquanto que para sistemas dinâmicos, será necessário realizar uma estatística no conjunto (conjuntal).

Veja-se que, devido ao requisito Req_3, o algoritmo desenvolvido ganha generalidade (abrange os sistemas multidimensionais), mas devido aos requisitos Req_4, Req_5 e Req_6 a perde (não abrange dados de entrada com parâmetros qualitativos nem comportamentos dinâmicos).

3) Levando em conta que pretende ser feita uma combinação de métodos baseados em dados, precisa-se definir um requisito para cumprir com essa restrição.

Req_7: o algoritmo desenvolvido deve ser uma combinação de métodos baseados em dados.

4) Dentre todas as técnicas de detecção de dados e análise de comportamentos apresentadas no Capítulo 2, observa-se que a técnica de análise de agrupamentos e, concretamente, os métodos exclusivos e intrínsecos, são os únicos que podem ser aplicados de forma não supervisionada (sem conhecimento prévio do sistema). Além disso, considerando a definição de método e algoritmo de clusterização (no fim da seção 2.5.1), percebe-se que o algoritmo deve incorporar heurísticas pré-configuradas pela autora.

Req_8: o algoritmo BAbyLO-BR deve ser um algoritmo baseado em métodos de clusterização exclusivos e intrínsecos.

Req_9: o algoritmo BAbyLO-BR deve ser um programa computacional que incorpore heurísticas pré-configuradas pela autora.

Req_10: o algoritmo BAbyLO-BR deve realizar clusterização completa.

Obs: Como já foi falado, neste trabalho será utilizado o termo clusterização para se referir à técnica de análise de agrupamentos e o termo *cluster* para se referir a cada um dos agrupamentos resultantes da aplicação da técnica.

3.2. Hipóteses do projeto

O algoritmo de classificação BAbyLO-BR está desenvolvido se baseando nas seguintes hipóteses:

Hipótese H1: se n_a testes forem realizados com um sistema dinâmico sob as mesmas condições e se seus resultados $y_j(t)$ com $j \in [1, n_a]$ forem registrados em m instantes t_k com $k \in [1, m]$, então, para cada instante t_k , tem-se n_a amostras com valores $y_j(t_k)$.

Hipótese H2: se n_a amostras foram tomadas com um sistema estático sob as mesmas condições sendo seus resultados $y(t_k)$ com $k \in [1, n_a]$, então, tem-se n_a amostras com valores $y(t_k)$.

Hipótese H3: supõe-se que o comportamento nominal de uma população verificando a hipótese H1 ou H2 segue uma distribuição normal $N(\mu, \sigma)$, assim, os valores das n_a amostras vão se acumular em torno de uma média amostral $m \xrightarrow[n_a \rightarrow \infty]{} \mu$ com dispersão medida pelo desvio padrão amostral $s \xrightarrow[n_a \rightarrow \infty]{} \sigma$. Estes valores dependem do tamanho da população (número de amostras n_a) e dos valores tomados pela população.

Hipótese H4: supõe-se que o comportamento anômalo de uma população que verifica as hipóteses H1 ou H2 segue uma distribuição normal $N(\mu_a \neq \mu, \sigma_a \neq \sigma)$, assim, os valores das n_a amostras vão se acumular em torno de uma média amostral

$m_a \xrightarrow{n_a \rightarrow \infty} \mu_a \neq \mu$ com dispersão medida pelo desvio padrão amostral $s_a \xrightarrow{n_a \rightarrow \infty} \sigma_a \neq \sigma$. Estes valores dependem do tamanho da população (número de amostras n_a) e dos valores tomados pela população.

Hipótese H5: supõe-se que o comportamento anômalo da população pode ser uma ou mais amostras isoladas.

Com essas cinco hipóteses, o algoritmo BAbyLO-BR fica contextualizado em um ambiente onde podem ser recebidas amostras ao longo do tempo de um sistema estático (estatística temporal) (H2) ou amostras no mesmo instante de tempo de diferentes repetições experimentais de um sistema dinâmico (estatística conjuntal) (H1). Espera-se que o comportamento nominal seja aproximadamente gaussiano (H3) e que as amostras anômalas apareçam como um conjunto aproximadamente gaussiano de amostras (H4) ou como elementos isolados (H5).

Da aplicação do método de clusterização para implementação do algoritmo de classificação surgem as seguintes definições:

Conforme o Req_3, a **variável de entrada** do algoritmo BAbyLO-BR será a matriz $A \in \mathcal{M}^{n_a \times n_p}$ onde n_a representa o número de amostras e n_p representa o número de parâmetros de cada amostra de forma que:

- O vetor $a_i = [a_{i1} \dots a_{in_p}]$ com $i \in [1 \dots n_a]$ representa a amostra i de n_p parâmetros.
- O vetor $p_j = [a_{j1} \dots a_{jn_a}]^T$ com $j \in [1 \dots n_p]$ representa todas as n_a amostras do parâmetro p_j .
- O parâmetro j da amostra i é representado como a_{ij} .

Para aplicar a técnica de clusterização usando técnicas hierárquicas precisa-se definir uma **matriz de proximidade** que associa a cada par de amostras uma **medida de dissimilaridade** d :

$$M_{prox} \in \mathcal{M}^{n_a \times n_a} \setminus m_{prox_{ik}} = d(a_i, a_k) \quad (3.1)$$

Além disso, precisa-se definir um **critério de parada** para decidir em que iteração é considerado que se atingiu a clusterização ótima.

Para aplicar a técnica de clusterização usando técnicas particionais, precisa-se definir um **procedimento que gere a partição final de forma direta**, sem iterações. O critério de parada das técnicas hierárquicas é comparável ao procedimento definido nas técnicas particionais.

O **objetivo do algoritmo de classificação** é agrupar o conjunto das n_a amostras em grupos de comportamento em função dos n_p parâmetros que as caracterizam de forma que as amostras classificadas em diferentes comportamentos sejam menos similares do que as amostras classificadas no mesmo comportamento.

Seja qual for a técnica utilizada, segundo pesquisa realizada, os algoritmos de classificação mediante clusterização (ou mediante outras técnicas) desenvolvidos até agora podem ser classificados em dois grupos:

- 1) Métodos ou algoritmos **genéricos** aplicáveis a diversos estudos de caso, mas que precisam da **definição do critério de parada por parte do usuário**, por exemplo, a estratégia *K-means* (SWAGATAM, 2008 e ARIFIN, 2006).
- 2) Métodos ou algoritmos **específicos**, aplicáveis unicamente a um ou vários estudos de caso que realizam a clusterização de forma automática (**critério de parada automático**) graças ao conhecimento prévio das características intrínsecas dos dados de entrada (algoritmos extrínsecos) (PORTER, 1996).

O problema que se quer resolver é conseguir implementar um algoritmo utilizando uma **combinação de técnicas baseadas em dados**. Gostar-se-ia conseguir também que este algoritmo seja de aplicação **genérica**, i.e., aplicável a diversos estudos de caso onde as características dos dados de entrada são desconhecidas (métodos intrínsecos) cujo **critério de parada seja calculado** com base nas informações do sistema inerentes aos dados de entrada, sem necessidade de conhecimento prévio do sistema no qual foram gerados. Isto resulta um desafio conforme testemunhado na seção 2.5.1.

3.3. Técnica de desenvolvimento

Para realizar o projeto de desenvolvimento do algoritmo BAbyLO-BR, utilizar-se-á uma combinação do modelo de protótipo evolutivo e do modelo incremental (GAYARRE, 2013) devido à condição de pesquisa que caracteriza uma tese de doutorado: inicialmente, conhece-se em grandes linhas o problema que se quer ser resolvido e, só realizando um estudo de métodos já existentes, desenvolvendo protótipos e aprofundando no problema que se quer resolver, chega-se ao produto final da tese.

Sabendo-se que será utilizada como base do desenvolvimento a técnica de análise de comportamentos, com os métodos exclusivos e intrínsecos, falta definir se serão utilizados métodos hierárquicos ou métodos particionais (Tabela 2-2).

Para isto, na **primeira fase de desenvolvimento** (seção 4.2), desenvolveram-se vários algoritmos já existentes na literatura para entender de que forma a definição do critério de parada é dependente do conhecimento dos dados de entrada do algoritmo.

Uma vez entendida esta dependência, na **segunda fase de desenvolvimento** (seção 4.3), foram implementados vários algoritmos utilizando técnicas hierárquicas cujo critério de parada era independente do conhecimento prévio dos dados de entrada, mas percebeu-se a inviabilidade de aplicação destas técnicas (hierárquicas) em sistemas com várias variáveis ou muitas amostras de entrada (conforme definido por Dubes (1988)).

Por último, na **terceira fase de desenvolvimento** (seção 4.4), decidiu-se aplicar técnicas de clusterização particionais (em concreto, uma técnica particional baseada em histogramas) combinadas com outras técnicas baseadas em dados (matemáticas, estatísticas e de filtragem) para implementar o algoritmo final BAbyLO-BR. Mesmo tendo realizado as fases anteriores, foi necessário realizar 6 iterações de implementação para chegar à versão final do algoritmo.

Todos os algoritmos desenvolvidos neste trabalho, pela própria definição do termo algoritmo (seção 2.5.1), contêm heurísticas que serão devidamente indicadas.

A Tabela 3-1 é um resumo do projeto de desenvolvimento realizado até chegar ao algoritmo final BAbyLO-BR.

Tabela 3-1. Resumo de fases de desenvolvimento, técnicas utilizadas e algoritmo desenvolvidos até chegar ao algoritmo final.

FASE de desenvolvimento	Técnicas utilizadas	Algoritmos desenvolvidos
FASE I	Hierárquicas Com conhecimento prévio do sistema.	1-Algoritmo de distância euclidiana 2-Algoritmo rastreável
FASE II	Hierárquicas Sem conhecimento prévio do sistema.	1-Algoritmo de cinzas 2-Algoritmo branco e preto
FASE III	Particionais Sem conhecimento prévio do sistema.	1 - BAbyLO-BR V01 2 - BAbyLO-BR V02 3 - BAbyLO-BR V03 4 - BAbyLO-BR V04 5 - BAbyLO-BR V05 6 - BAbyLO-BR V06

4 DESENVOLVIMENTO DO ALGORITMO BAbyLO-BR

O desenvolvimento do algoritmo BAbyLO-BR será explicado conforme foi desenvolvido sequencialmente no tempo; enumerando as dificuldades e explicando as deduções que levaram às tomadas de decisão e mudanças na implementação até chegar ao algoritmo final.

Todos os algoritmos desenvolvidos até chegar ao algoritmo final foram testados e validados utilizando o estudo de caso chamado de amostra de conveniência, disponível na literatura (SHI, JOHN Z. 2007). Por este motivo, primeiramente será explicado este estudo de caso para poder seguir de forma coerente às deduções durante as diferentes fases do projeto.

Os resultados obtidos da classificação das amostras deste estudo de caso com os diferentes protótipos do algoritmo desenvolvidos não vão ser mostrados, pois estes resultados não agregam informações relevantes para entender a dedução que levou à implementação final (sendo que o interesse das fases não é comprovar a confiabilidade do algoritmo implementado, mas entender os problemas e dificuldades que querem ser resolvidos na implementação do algoritmo final).

Este capítulo está dividido em quatro seções, correspondendo à definição do estudo de caso e às três fases de desenvolvimento. A primeira seção (4.1) detalha o estudo de caso utilizado durante o projeto de desenvolvimento e as seções 4.2, 4.3 e 4.4 estão organizadas segundo os algoritmos desenvolvidos em cada fase. Para cada algoritmo explicam-se as características dele (heurísticas) e a técnica de implementação. No final de cada fase apresentam-se as conclusões.

4.1. Detecção de anomalias em tubulações hidráulicas

O estudo de caso chamado de amostra de conveniência consiste na detecção de anomalias em tubulações hidráulicas. Este estudo de caso foi escolhido por ser um estudo de caso simples, utilizado com exemplo na disciplina de Modelagem e Detecção de Falhas em Tempos Virtual e Real (CMC-211-4) e por estar disponível na literatura (SHI, JOHN Z. 2007).

A Figura 4-1 mostra o princípio de operação do sistema de três tanques; duas torneiras situadas nos tanques 1 e 3 regulam independentemente duas entradas de água no sistema com vazões respectivas q_{i1} e q_{i2} . Os três tanques estão conectados mediante duas tubulações com resistências R_1 e R_3 para controlar as vazões de água q_1 e q_3 . A água do sistema pode sair unicamente mediante o sumidouro situado no tanque 2. A resistência do sumidouro é R_2 e a vazão é q_2 . Neste sistema analisa-se a evolução do nível de água nos 3 tanques; h_1, h_2 e h_3 . Assim, essas 3 variáveis serão as entradas do algoritmo de classificação.

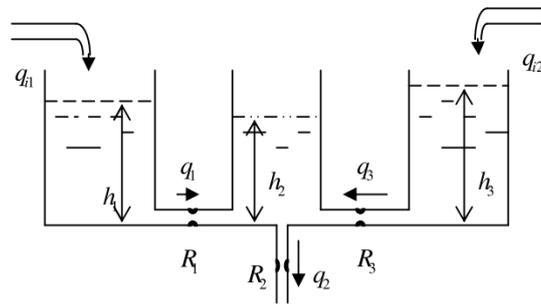


Figura 4-1. Sistema de três tanques.

Modelagem do sistema (SHI, JOHN Z. 2007).

O modelo linearizado do sistema é:

$$\left\{ \begin{array}{l} q_{i1} - q_1 = S_1 \frac{d}{dt} h_1 = q_{i1} - \frac{h_1 - h_2}{R_1} \\ q_{i2} - q_3 = S_3 \frac{d}{dt} h_3 = q_{i2} - \frac{h_3 - h_2}{R_3} \\ q_1 + q_3 - q_2 = S_2 \frac{d}{dt} h_2 = \frac{h_1 - h_2}{R_1} + \frac{h_3 - h_2}{R_3} - \frac{h_2}{R_2} \end{array} \right. \quad (4.1)$$

onde S_1, S_2 e S_3 são as seções dos 3 tanques, medidas em $[m^2]$.

Quando um sistema é modelado por equações diferenciais lineares, é sempre possível colocar as equações do sistema no formato de espaço de estados:

$$\begin{cases} \dot{X}(t) = AX(t) + BU(t) \\ Y(t) = CX(t) \end{cases} \quad (4.2)$$

sendo U o vetor das variáveis de entrada do sistema (atuadores), Y vetor das variáveis de saída do sistema (sensores), X vetor dos estados do sistema e A, B e C três matrizes caracterizando o sistema.

Como as equações diferenciais do sistema de três tanques (Equação (4.1)) são lineares é possível reescrever essas equações no formato do espaço de estados definido no sistema de Equações (4.2):

$$\dot{X} = \begin{bmatrix} -\frac{1}{S_1 R_1} & \frac{1}{S_1 R_1} & 0 \\ \frac{1}{S_2 R_1} & -\frac{1}{S_2} \left(\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} \right) & \frac{1}{S_2 R_3} \\ 0 & \frac{1}{S_2 R_3} & -\frac{1}{S_3 R_3} \end{bmatrix} X + \begin{bmatrix} \frac{1}{S_1} & 0 \\ 0 & 0 \\ 0 & \frac{1}{S_3} \end{bmatrix} U \quad (4.3)$$

$$Y = X$$

onde

- As variáveis de entrada do sistema são $U^T = [q_{i1}, q_{i2}]^T$
- As variáveis de estado do sistema são $X^T = [h_1, h_2, h_3]^T$
- As variáveis de saída do sistema são $Y^T = [h_1, h_2, h_3]^T$

Simulação do sistema (adaptação de SHI, JOHN Z. 2007).

O sistema acima descrito foi simulado utilizando a ferramenta *Simulink* conforme mostrado na Figura 4-2 (SHI, JOHN Z. 2007). Para este trabalho foi adicionado um ruído branco gaussiano de característica $N(0, 0.01)$ [m] (ver seção 2.5.3) independente em cada variável de saída Y do sistema para simular ruído nos sensores.

Note-se que o nível de ruído gaussiano simulado é baixo em comparação com os valores finais dos parâmetros de saída (sistema muito bem comportado). Isto é feito intencionalmente para poder observar com facilidade o comportamento do algoritmo implementado.

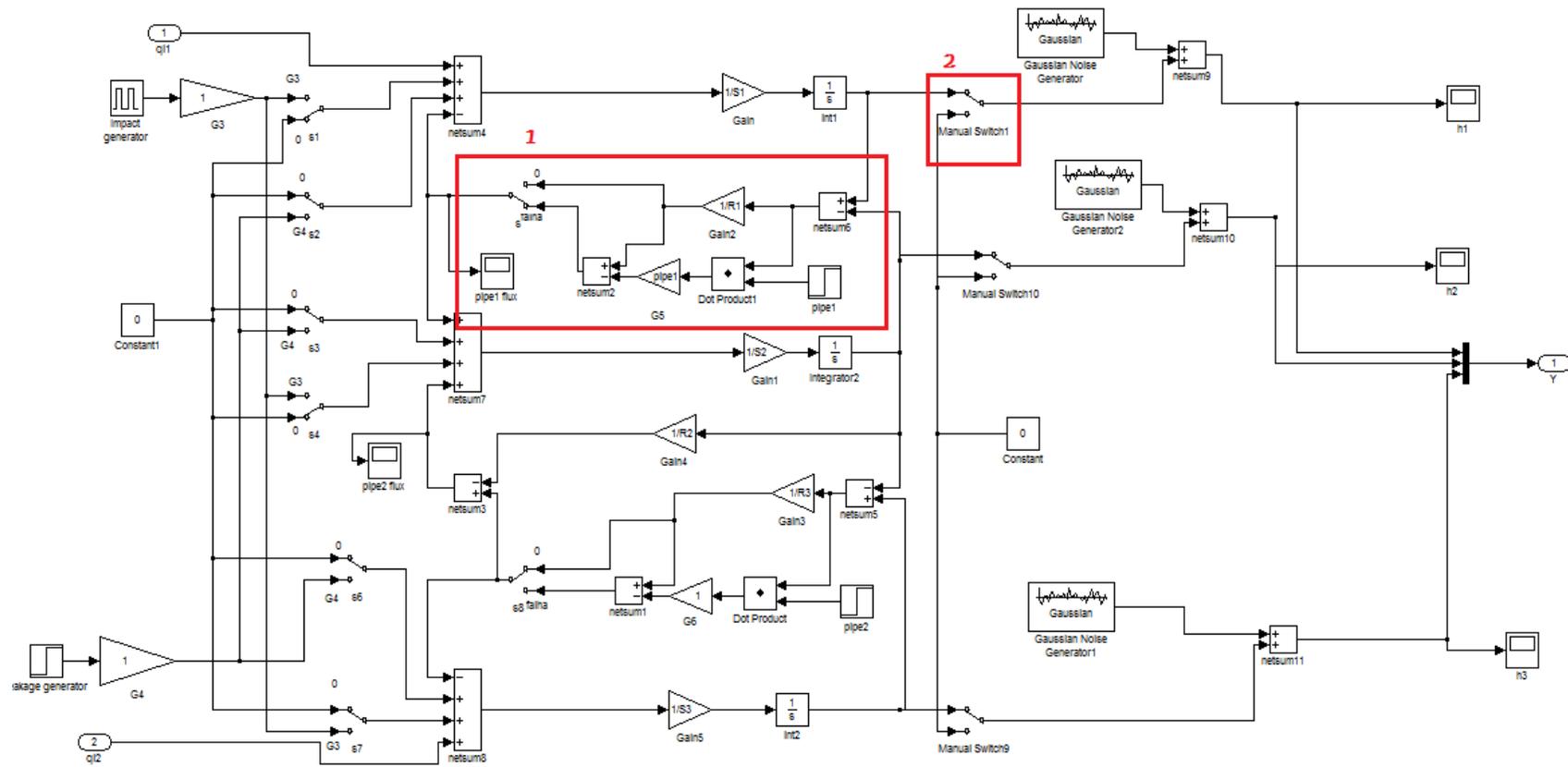


Figura 4-2. Modelo em Simulink do sistema três tanques.

As falhas simuladas são (modelagem indicada na Figura 4-2 com quadro vermelho numerado 1 e 2 respectivamente):

- 1) Entupimento parcial ou total da tubulação de resistência R_1 . O entupimento pode variar entre 0 e 100%, sendo 100% o entupimento total.
- 2) Perda do sensor de medida do nível de água no tanque 1 (h_1). A falha pode tomar valores 1 ou 0, isto é, sensor funcionando ou não.

Os valores dos parâmetros característicos do sistema utilizados na simulação são definidos na Tabela 4-1:

Tabela 4-1. Valores utilizados para simulação do sistema três tanques.

Parâmetros	S_1 (m²)	S_2 (m²)	S_3 (m²)
Valores	1	1	1
Parâmetros	R_1(s/m²)	R_2(s/m²)	R_3(s/m²)
Valores	0.5	2	1

4.2. FASE I – Técnicas hierárquicas com conhecimento prévio do sistema

Esta fase tem como objetivo entender a dependência que os algoritmos de clusterização têm do conhecimento prévio do sistema a ser estudado no momento de definir o **critério de parada**.

Começou-se pelos métodos hierárquicos devido às características deles (algoritmos iterativos onde é possível visualizar o processo de clusterização passo a passo graças ao dendograma).

Para implementar um algoritmo hierárquico, precisa-se definir a **medida de dissimilaridade**, criar a **matriz de proximidade** e determinar o **critério de parada**.

A medida de dissimilaridade escolhida foi a distância euclidiana d_{eucl} e a gêmea distância euclidiana quadrada d_{eucl2} . Para duas amostras quaisquer a_i e a_k essas duas normas são definidas como:

$$d_{eucl}(a_i, a_k) = \left(\sum_j^{n_p} |a_{ij} - a_{kj}|^2 \right)^{\frac{1}{2}} \quad (4.4)$$

$$d_{eucl2}(a_i, a_k) = \left(\sum_j^{n_p} |a_{ij} - a_{kj}|^2 \right) \quad (4.5)$$

O **critério de parada** foi calculado baseado no conhecimento prévio do sistema estudado. As amostras normais apresentam um comportamento aproximadamente gaussiano de característica $N(0, 0.01)$ devido ao ruído simulado nas três saídas do sistema. A propagação do erro devido ao ruído no cálculo da distância de dissimilaridade é calculada conforme o APÊNDICE A - (erro acumulado). Assim, a distância máxima entre duas amostras normais é, como máximo, duas vezes o erro acumulado.

A seguir, são apresentados os dois algoritmos implementados nesta fase, o algoritmo de distância euclidiana e o algoritmo rastreável.

4.2.1. Algoritmo de distância euclidiana

Como falado no começo desta FASE, este algoritmo pode utilizar a distância euclidiana e a distância euclidiana quadrada; e o critério de parada (erro acumulado) é o raio do *cluster*, calculado em função das características conhecidas previamente do sistema.

Técnica:

A técnica é apresentada na forma de pseudocódigo:

- 1) Escolhe-se a medida de dissimilaridade (pode ser usada *deucl* ou *deucl2*)
- 2) Gera-se a matriz de proximidades (*Mprox*), usando a distância euclidiana ou a distância euclidiana quadrada como medida de dissimilaridade.

While não é a última amostra

- 3) Determinam-se os dois elementos *elem_j* e *elem_k* com distância mínima usando *Mprox*.
- 4) Escolhe-se um dos elementos (*elem_j*) e calcula-se o erro acumulado segundo ele.
- 5) Gera-se um *cluster* com centroide em *elem_j* e raio erro acumulado.

```

6) Marcam-se todos os elementos classificados neste primeiro cluster para não serem
    escolhidos de novo como distância mínima na próxima iteração.

end while

7) Fim do processo. Desenha-se o gráfico com os resultados da classificação.

```

A classificação de dados obtida foi a mesma independentemente da distância utilizada, o que valida a técnica usada para o cálculo do raio do *cluster* em função da propagação do erro na saída. Esta classificação não foi boa, pois as amostras de comportamento aproximadamente normal foram às vezes divididas em dois *clusters* diferentes. Acredita-se que isto acontece pela forma em que foi escolhido o centróide do *cluster* (a amostra com distância de dissimilaridade menor não tem por que ser o centróide do *cluster*).

4.2.2. Algoritmo rastreável

Como o algoritmo anterior, o algoritmo rastreável pode utilizar a distância euclidiana e a distância euclidiana quadrada.

Com a implementação do algoritmo rastreável, buscou-se achar outra heurística para formar o *cluster*, sem definir um centróide fixo. Assim, procurou-se achar sempre os elementos mais próximos ao elemento atual, fazendo um tipo de rastreio e tentando chegar à área com maior densidade de amostras.

Para levar a cabo este rastreio, a matriz de proximidade é calculada de forma iterativa como a distância entre o elemento atual e os outros elementos. Considerando a_i como amostra atual e a_k qualquer outra amostra não classificada, o critério de parada é $d(a_i, a_k) > \text{Erro acumulado}$.

Técnica:

A técnica é apresentada na forma de pseudocódigo:

- 1) Escolhe-se a medida de dissimilaridade (pode ser usada *deucl* ou *deucl2*)
- 2) Calcula-se a matriz de proximidades *Mprox*
- 3) Procura-se o maior elemento de *Mprox*: *Max*
- 4) $\text{Diag}(\text{Mprox}) = \max(\text{Mprox}) + 10$ %evitar $\text{Diag}(\text{Mprox}0) = 0$

```

1° While distância entre clusters < Max

5) Escolhem-se os elementos com dist mínima entre eles (elem_j e elem_k)

2° While dist mínima < erro acumulado %cria-se cluster

6) Começando com elem_j, procura-se o elemento mais perto dele (na primeira iteração
vai ser elem_k)
7) Inclui-se elem_k no cluster
8) elem_j = elem_k
9) Atualizam-se distâncias na matriz de proximidades
10) Marca-se elem_k como classificado para não ser escolhido de novo

End do 2° while % Quando termina o 2°while acabou-se com o primeiro cluster, mesma
coisa para os seguintes.

End do 1° while

11) Realiza-se clusterização dos clusters para agrupar aqueles cuja distância entre
centróides é menor do que o raio esperado do cluster.
12) Cria-se a matriz de clusters de saída e desenha-se o gráfico com os resultados da
classificação.

```

A classificação atingida por este algoritmo foi melhor do que a do algoritmo de distância euclidiana, devido possivelmente ao método seguido para escolher as amostras com características similares para formar o *cluster*. Além disso, o passo 11, definido para agrupar *clusters* entre eles uma vez que já foram formados, evita a classificação de amostras normais em *clusters* finais diferentes. Mesmo assim, ele depende ainda do conhecimento prévio dos dados para calcular o **erro acumulado** que é utilizado como critério de parada.

4.2.3. Conclusões

Ambos os algoritmos foram aplicados ao sistema de três tanques para realizar a detecção das falhas simuladas, utilizando as duas medidas de dissimilaridade, e as conclusões foram as seguintes:

- 1) O critério de parada é definido pelo usuário utilizando as informações conhecidas da variância do ruído branco gaussiano simulado no sistema. (precisa-se de conhecimento prévio da natureza das amostras para definir esse valor).

- 2) Existem decisões de implementação como o processo de formação do *cluster* que devem ser resolvidas de forma heurística; essa heurística diferencia os algoritmos de clusterização baseados nos mesmos métodos, produzindo uma classificação final diferente.
- 3) Pode acontecer que as amostras de comportamento normal sejam divididas em vários *clusters* diferentes devido à estratégia utilizada para formação de *clusters*.
- 4) Existem decisões de implementação que são difíceis de escolher, pois dependem de muitos fatores característicos das amostras, como, por exemplo, a definição do critério de parada.
- 5) Na próxima fase, deve-se procurar um processo de clusterização que não dependa do conhecimento prévio do sistema.

4.3. FASE II – Técnicas hierárquicas sem conhecimento prévio do sistema

Na fase anterior foram usados métodos hierárquicos, pois eles são simples de implementar e permitem ver a evolução do algoritmo por iterações, para assim entender melhor as dificuldades que apareceram durante a classificação. Por esse motivo, nesta fase continua-se usando métodos hierárquicos.

Porém, nesta fase procura-se a independência do conhecimento prévio do sistema, para generalizar o algoritmo e permitir sua aplicação a sistemas com características diferentes. Assim, procurou-se uma nova técnica, baseada em grades.

Esta técnica consiste em trabalhar com a distribuição espacial das amostras (desenhar as amostras no plano/espço e tratá-las como se fossem uma figura de pontos); as amostras são agrupadas em função da posição.

4.3.1. Algoritmo de cinzas

O algoritmo de cinzas é um algoritmo bidimensional (trabalha unicamente com dois parâmetros), pelo que aplica uma técnica básica PCA para escolher os dois parâmetros de entrada com mais informações que caracterizam as amostras. Os valores desses parâmetros são utilizados para desenhar o conjunto de amostras de entrada como se fossem uma imagem em duas dimensões.

Técnica:

- 1) Constrói-se uma grade com base nos valores da imagem de amostras, designando a cada pixel da grade o valor correspondente ao número de amostras contidas nesse pixel. Esse valor é chamado de **nível de cinza**.
- 2) Procura-se o pixel com maior nível de cinza. Ao redor dele, criam-se os quatro possíveis pixels 2×2 (chamados de **pixels aumentados**) conforme Figura 4-3. O nível de cinza v_{aglut} de cada **pixel aumentado** é a média dos níveis de cinza dos 4 pixels (p_1, p_2, p_3, p_4) que o formam.

$$v_{aglut} = \frac{p_1 + p_2 + p_3 + p_4}{4} \quad (4.6)$$

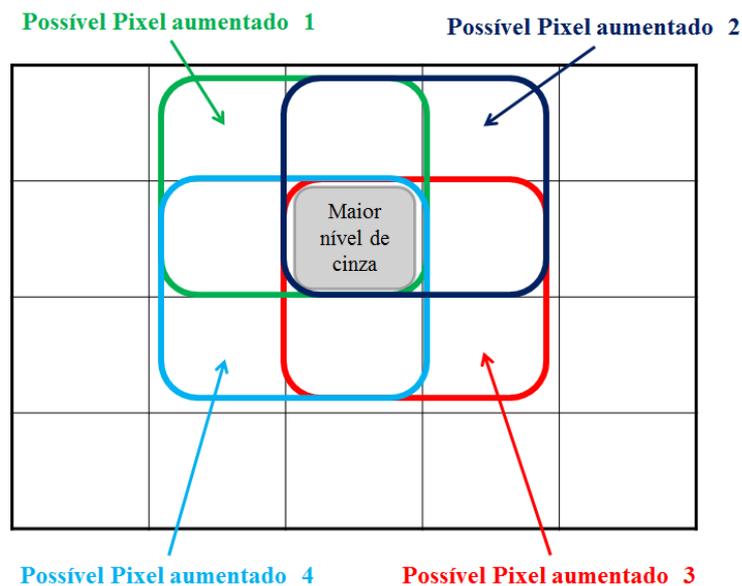


Figura 4-3. Representação dos quatro pixels aumentados possíveis.

- 3) Escolhe-se o pixel aumentado com maior nível de cinza v_{aglut} (**pixel aumentado de referência**) e aplica-se então um procedimento hierárquico aglomerativo (**processo de aglutinação**) de forma que toda grade é dividida em grupos de pixels aumentados 2×2 ao redor do pixel aumentado de referência conforme a Figura 4-4.

O pixel aumentado de referência acumula o maior número de amostras; assim, seguindo este método, favorece-se a formação do *cluster* maior, desfavorecendo a formação dos *clusters* menores.

Nota-se que o processo de aglutinação pode precisar da aplicação do processo de *padding* (adição de zeros); caso não exista um número par de filas ou colunas ao redor do pixel aumentado de referência, adiciona-se uma fila ou coluna com valores nulos para completar. A Figura 4-4 também mostra um exemplo do processo de *padding*.

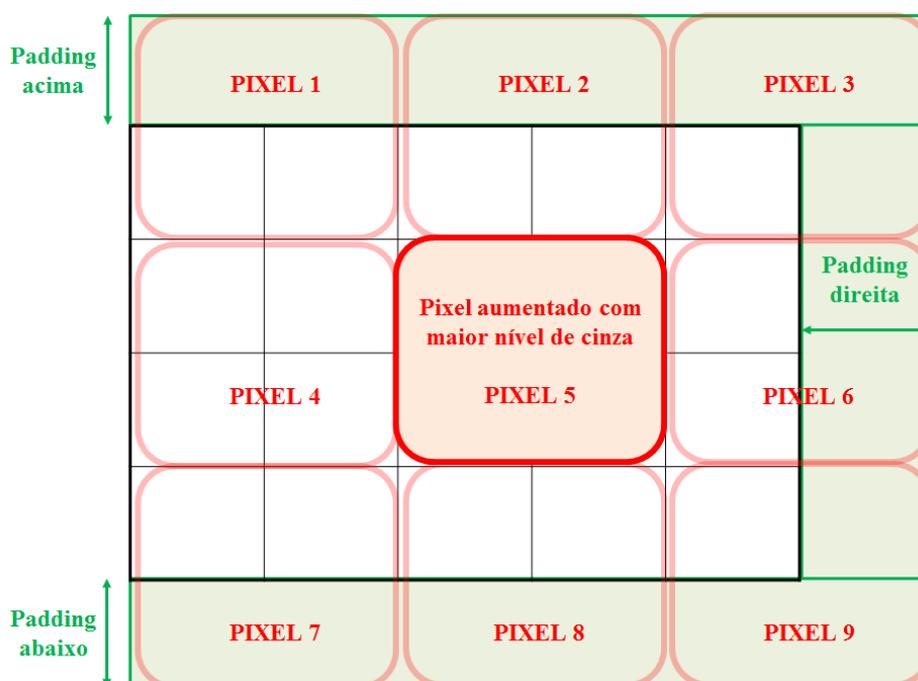


Figura 4-4. Processo de *padding*.

O pixel aumentado de referência é o pixel número 5 da Figura 4-4. Assim, para aglutinar os pixels da fila superior (criar os pixels aumentados 1, 2 e 3), precisa-se adicionar uma fila acima; a mesma coisa acontece com os pixels situados na última coluna (para criar os pixels aumentados 3, 6 e 9) e na última fila (para criar os pixels aumentados 7, 8 e 9).

- 4) Cada pixel aumentado criado (no exemplo da Figura 4-4, os pixels aumentados de 1 até 9) deve adquirir um nível de cinza associado. Para isso, primeiro é avaliado se o pixel aumentado é **próprio para** clusterização conforme os

critérios de parada (definidos abaixo). Se ele for próprio para clusterização, o nível de cinza associado corresponde ao v_{aglut} calculado. Se não, o nível de cinza é 0.

Os dois critérios de parada são os que se seguem:

- a) No pixel aumentado, devem existir no mínimo dois pixels dos quatro que o compõem tal que $p_i > v_{aglut}$.

Matematicamente, seja $f: \{p_1, p_2, p_3, p_4\} \rightarrow \mathbb{R}$ uma função definida tal que, para cada nível de cinza p dos pixels contidos em um pixel aumentado, ela associa um valor y em função do nível de cinza do pixel aumentado v_{aglut} :

$$y = f(p) = \begin{cases} 0 & \text{se } v_{aglut} > p \\ 1 & \text{se } v_{aglut} \leq p \end{cases} \quad (4.7)$$

Assim, o primeiro critério de parada considera um pixel aumentado como próprio para clusterização se e somente se:

$$y_1 + y_2 + y_3 + y_4 \geq 2 \quad (4.8)$$

- b) A soma das diferenças entre o nível de cinza do pixel aumentado v_{aglut} e os níveis de cinza de cada pixel p_i que o formam deve ser sempre maior ou igual a zero.

$$\sum_{i=1}^4 (v_{aglut} - p_i) \geq 0 \quad (4.9)$$

- 5) Uma vez atribuído um nível de cinza para cada pixel aumentado, repetem-se os passos 2) até 4) onde os pixels são os pixels aumentados da iteração anterior, até que não existam pixels aumentados próprios para clusterização (avaliados com zero). Neste instante o dendograma está completo.
- 6) Para calcular os *clusters* finais, procura-se no dendograma a iteração em que cada pixel é zerado. Nesta iteração, considera-se que o pixel representa um

cluster e procuram-se os pixels da grade inicial que formam esse *cluster* voltando atrás no dendograma.

Observação: O processo realizado nos passos 5 e 6 é similar ao realizado nos passos 4 e 5 do algoritmo branco e preto que é explicado na seguinte seção (seção 4.3.2) à diferença do valor atribuído a cada pixel. Porém, a representação gráfica do processo iterativo de aglutinação e do dendograma resulta mais simples no algoritmo branco e preto, por ter uma medida de dissimilaridade mais simples. Assim, nesta seção faz-se referência às Figura 4-7, Figura 4-8, Figura 4-9 e Figura 4-10 da seguinte seção (4.3.2) para melhor compreensão dos passos 5 e 6.

Conclusões

O algoritmo foi aplicado ao estudo de caso de três tanques e as conclusões são como se segue:

- 1) O critério de parada aplicado neste algoritmo é muito heurístico.
- 2) Este algoritmo funcionou com o estudo de caso de três tanques, mas, possivelmente não vai funcionar em diversos casos de uso devido ao empirismo do critério de parada.
- 3) O fato de se atribuir a cada pixel um valor de **nível de cinzas** dificulta a definição de um critério de parada menos empírico (o nível de cinzas pode variar muito de um estudo de caso a outro, pois depende do número de amostras de entrada e do tamanho da grade inicial entre outros parâmetros próprios dos dados de entrada).
- 4) Deve-se implementar um algoritmo com um critério de parada mais genérico que dependa menos das características dos dados do sistema estudado.
- 5) O fato do algoritmo ser bidimensional limita a aplicabilidade prática do mesmo.

4.3.2. Algoritmo branco e preto

Visando seguir as conclusões do algoritmo de cinzas, foram pensadas as seguintes modificações, resultando na implementação do algoritmo branco e preto:

O algoritmo de cinzas pode ser simplificado associando a cada pixel um valor binário $\{0,1\}$ (branco ou preto) em função de se ter ou não amostras nesse pixel, independentemente do número de amostras. Fazendo esta simplificação, o critério de parada ficará mais simples assim como os cálculos e armazenamento de informações em cada iteração.

Antes de entrar na técnica do algoritmo, definem-se os seguintes conceitos.

Diferentemente do algoritmo de cinzas, no algoritmo preto e branco criam-se as quatro possíveis grades de pixels aumentados 2×2 em cada iteração. Para escolher a grade final da iteração, aplica-se um procedimento de avaliação baseado na variância.

O tamanho dos pixels iniciais (grade) deve ser calculado de forma automática em função do número de amostras e da dispersão delas na imagem criada. Para isso, aplica-se o seguinte processo; considerando que a imagem bidimensional formada corresponde aos dois parâmetros (p_i, p_j) escolhidos pela técnica PCA, a imagem tem as dimensões seguintes:

$$x = \max(p_i) - \min(p_i) \quad (4.10)$$

$$y = \max(p_j) - \min(p_j) \quad (4.11)$$

$$A_{imagem} = xy \quad (4.12)$$

onde A_{imagem} representa a área da imagem.

Sabendo que se tem n_a amostras, o tamanho e número de pixels que formam a grade são calculados como:

$$A_{pixel} = \frac{A_{imagem}}{n_a} \quad (4.13)$$

$$L_{pixel} = \sqrt{A_{pixel}} \quad (4.14)$$

$$n_{pixels_x} = \left\lceil \frac{x}{L_{pixel}} \right\rceil \quad (4.15)$$

$$n_{pixels_y} = \left\lceil \frac{y}{L_{pixel}} \right\rceil \quad (4.16)$$

assim, a grade tem pixels de tamanho L_{pixel} . Nota-se que os números de pixels nos dois eixos calculados nas Equações (4.15) e (4.16) são aproximados até o próximo número inteiro (operador $\lceil \cdot \rceil$).

Resumindo, o tamanho da grade é escolhido de tal maneira que tenha, em média, uma amostra por pixel.

Técnica

- 1) Para cada pixel é associado o valor 1 quando contenha uma ou mais amostras; e o valor 0 quando não contenha amostras (elimina-se assim o nível de cinzas).
- 2) Em cada iteração são calculadas as 4 possíveis grades 2×2 de forma que a cada pixel aumentado se associa o valor 1 sempre que mais de dois pixels que o formam tenham valor 1. A Figura 4-5 mostra um exemplo deste cálculo; no pixel aumentado azul, existem unicamente 2 pixels unitários pelo que o valor pixel aumentado é 0. Porém, o pixel aumentado vermelho tem valor final 1 pois existem três pixels com valor 1.

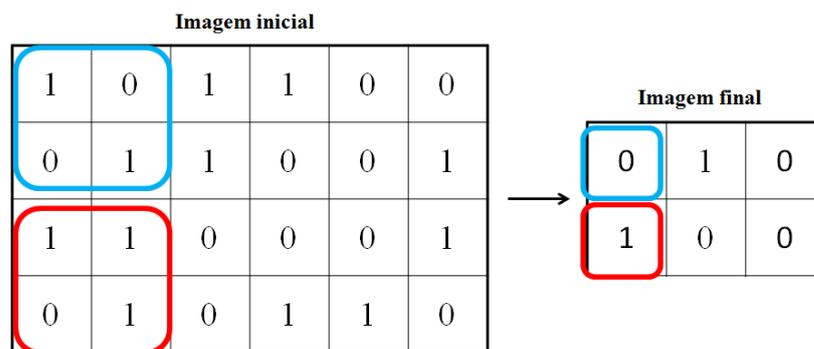


Figura 4-5. Cálculo do valor do pixel aumentado no algoritmo branco e preto.

- 3) Das 4 grades possíveis, escolhe-se uma conforme o seguinte procedimento:
 - a. Somam-se as linhas e as colunas da imagem (conforme Figura 4-6).
 - b. Calcula-se a variância nas colunas (σ_x^2) e nas linhas (σ_y^2) conforme a Equação (2.5).

c. Calcula-se a variância total conforme:

$$\sigma_{total}^2 = \sigma_x^2 + \sigma_y^2 \quad (4.17)$$

d. A grade escolhida é aquela com maior variância total.

		σ_x^2					
		┌───────────┐					
		2 3 2 2 1 2					
σ_y^2	3	1	0	1	1	0	0
	3	0	1	1	0	0	1
	3	1	1	0	0	0	1
	3	0	1	0	1	1	0

Figura 4-6. Cálculo de variância em cada modo de aglutinação.

4) Os passos 2) e 3) são aplicados de forma iterativa até que a variância total seja nula (critério de parada). Neste momento o dendograma está completo.

A Figura 4-7 mostra o processo completo de aglutinação de uma grade de tamanho 8 × 8. Como exemplo, em cada iteração (vermelho, verde e laranja respectivamente), mostram-se exemplos de pixels aumentados formados e como eles evoluíram para a iteração seguinte. Precisou-se de 4 iterações em total.

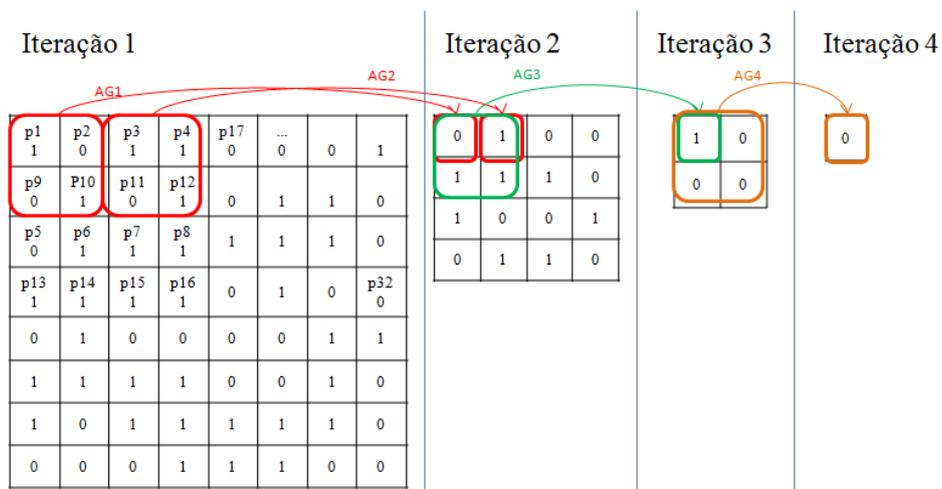


Figura 4-7. Processo de aglutinação aplicado a uma grade de tamanho 8×8 .

A Figura 4-8 mostra o dendograma completo final resultado da aglutinação realizada na Figura 4-7. Em vermelho, verde e laranja estão representados os pixels aumentados conforme a Figura 4-7.

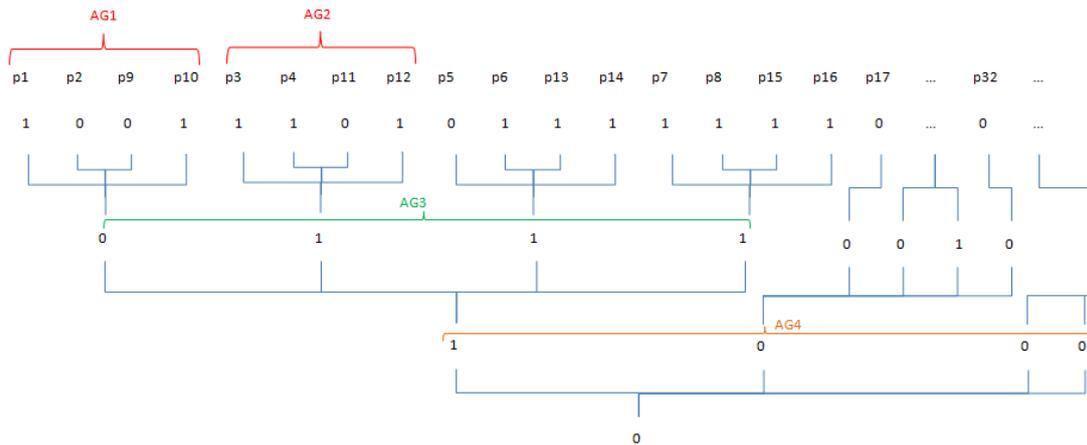


Figura 4-8. Dendograma completo após processo de aglutinação hierárquico segundo o algoritmo branco e preto.

- 5) Para calcular os *clusters* finais, procura-se no dendograma a iteração em que cada pixel é zerado. Nesta iteração, considera-se que o pixel representa um *cluster* e procuram-se os pixels da grade inicial que formam esse *cluster* voltando atrás no dendograma.

Como exemplo, na Figura 4-9 pode-se observar que na terceira iteração achou-se um pixel com valor 1 (em amarelo) que foi zerado na quarta iteração. Assim esse **pixel zerado** forma um *cluster*. Fazendo uso da rastreabilidade do dendograma, volta-se até a primeira iteração e localiza-se o **cluster formado** (também em amarelo).

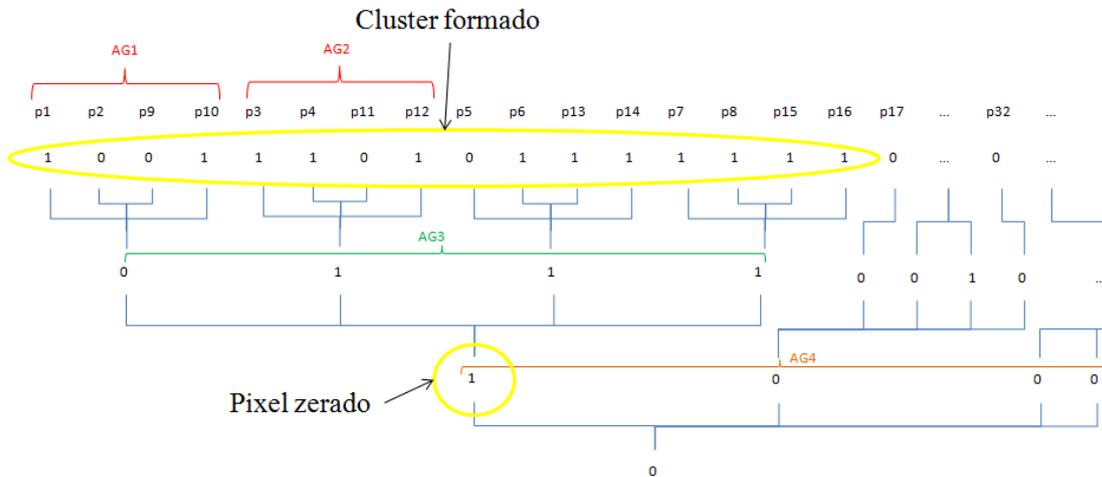


Figura 4-9. Exemplo de identificação de *cluster* (em amarelo) no dendograma.

Para finalizar, a Figura 4-10 mostra o processo de identificação de *clusters* e a clusterização final da grade inicial (iteração 1). A cada iteração são achados os pixels de valor 1 que evoluíram para zero (por exemplo, em amarelo na iteração 3) e retorna-se até a iteração 1 para formar o *cluster* final. Da mesma forma, em laranja, têm-se os *clusters* formados na iteração 2 e em azul os *clusters* formados na iteração 1. Para terminar, se observa então que, no total, formaram-se 1 *cluster* de tamanho 4×4 (em amarelo), 5 *clusters* de tamanho 2×2 (em laranja) e 8 *clusters* de tamanho 1×1 (em azul).

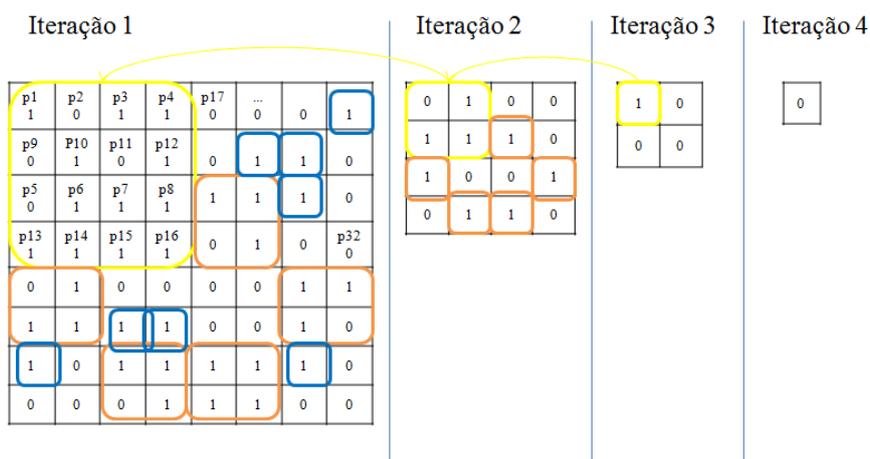


Figura 4-10. Exemplo de clusterização final.

Conclusões

- 1) O critério de parada é muito mais simples, genérico e independente do número de amostras e do valor delas.
- 2) As técnicas hierárquicas precisam de rastreabilidade entre iterações (dendograma) para poder identificar os *clusters*. Esta rastreabilidade é muito pesada de ser feita e requer um grande armazenamento de dados em memória. Isto dificulta muito a aplicação em sistemas onde é preciso realizar um grande número de iterações (DUBES, 1988).
- 3) Encarar e resolver um problema unidimensional é simples, aplicar o método a um problema bidimensional resulta mais complexo; e estendê-lo a um problema multidimensional é tarefa que requer uma capacidade de abstração muito alta. Na próxima fase, deve-se pensar em uma técnica que resolva a classificação de forma unidimensional fazendo depois uma combinação multidimensional para obter diretamente a classificação final (técnicas particionais).

4.4. FASE III – Técnicas particionais sem conhecimento prévio do sistema; implementação do algoritmo BAbyLO-BR

As Fases I e II serviram para 1) entender as dificuldades intrínsecas aos métodos de clusterização (definição de heurísticas, falta de uma definição matemática de *cluster*, diversidade de métodos, entre outras); 2) entender a dificuldade de extensão de um método hierárquico para aplicação em sistemas genéricos com independência do conhecimento prévio dos dados de entrada; 3) ganhar experiência e contato com as diversas técnicas; e, com esta experiência ganha, 4) sentir-se capaz de escolher uma técnica particional que permita atingir o objetivo deste trabalho; e 5) Definir mais dois requisitos derivados que serão utilizados na fase III.

Req_11: O algoritmo BAbyLO-BR deve ser implementado utilizando técnicas de clusterização particionais baseadas em histogramas.

Req_12: O algoritmo BAbyLO-BR deve ser implementado em ferramenta computacional para prova de conceito, na qual o usuário seja capaz de visualizar e editar as pré-configurações das heurísticas.

Conhecendo a dificuldade de visualizar técnicas multidimensionais (o cérebro humano está pronto para trabalhar em três dimensões, as demais dimensões ficam abstratas) pensou-se em um método que comportasse duas fases; 1) primeiro clusterizar de forma unidimensional cada parâmetro p_j ; para depois 2) juntar os resultados e obter a clusterização multidimensional. Para fazer isto, foi escolhido o método particional baseado em histogramas (vide seção 2.5.2).

O algoritmo criado nesta FASE III é o algoritmo BAbyLO-BR. Ele é implementado utilizando uma combinação de métodos baseados em dados e procura usar heurísticas genéricas para aplicação em sistemas com características diferentes. Este algoritmo foi validado utilizando quatro estudos de caso diferentes (o detalhamento dos estudos de caso e os resultados obtidos são mostrados no Capítulo 5).

Os resultados foram seis versões do algoritmo BAbyLO-BR. Estas seis versões são consequências do aperfeiçoamento do algoritmo conforme ia sendo validado com os diferentes estudos de caso.

Estas versões diferem unicamente em pequenos detalhes de implementação e não na técnica geral utilizada, pelo que as diferenças serão apresentadas em forma de tabela e unicamente a versão final V06 será explicada com detalhe na seção de técnica.

Assim, a Tabela 4-2 apresenta a evolução da implementação do algoritmo BAbyLO-BR e as modificações realizadas entre as versões.

Tabela 4-2. Evolução da implementação do algoritmo BAbyLO-BR versão a versão.

Versão	Problema	Implementação
V01	Implementação inicial.	Primeira versão do algoritmo.
V02	Saturação de memória quando os dados de entrada são grandes.	Armazenamento de dados em arquivos e não em variáveis do <i>Matlab</i> .
V03	Classificação pouco eficiente e não aplicável a diferentes sistemas.	Implementação de heurística para suavizar o histograma (técnicas de filtragem) e implementação de heurística para definição de <i>clusters</i> (corolário da primeira derivada).
V04	Os histogramas formados não são bons, devem ser aperfeiçoados conforme fatores de êxito definidos no Capítulo 2.	Implementação de heurística para definir número de janelas no histograma.
V05	O processo de formação de <i>clusters</i> pode ser melhorado.	Aprimoramento para formação de <i>clusters</i> (corolário da segunda derivada e conceito de variância).
V06	Os <i>clusters</i> não representam padrões de comportamento. O algoritmo não oferece bons resultados em amostras com parâmetros discretos nem com parâmetros com baixo desvio padrão.	Implementação de nova técnica para calcular valores máximo e mínimo de cada <i>cluster</i> mais ajustados ao comportamento das amostras que o formam. Aprimoramento de heurística para definir número de janelas inicial do histograma.

Técnica

Nesta seção explica-se de forma detalhada a última (6ª) versão do algoritmo BAbyLO-BR, considerada como produto final deste trabalho. A técnica escolhida foi particional baseada em histogramas (conforme Req_11). Esta técnica leva consigo algumas dificuldades que são apresentadas a seguir para entender os passos que formam o algoritmo BAbyLO-BR

Devido às hipóteses H3 e H4 definidas na formulação do problema (seção 3.2), considera-se que um comportamento formado por um conjunto de amostras (seja normal ou anômalo) tem distribuição aproximadamente gaussiana (seção 2.5.3). Estas hipóteses são necessárias para a aplicação da técnica escolhida baseada em histogramas e as heurísticas definidas.

No caso real, devido a que se trabalha com amostras e não com populações completas, a representação destas amostras disponíveis, resultado de um estudo empírico, não garante a obtenção de uma distribuição gaussiana perfeita sendo que podem existir pequenas variações (picos ou vales) em função de onde, como, quando, etc. foram tomadas as amostras. Este problema, explicado em forma de hipóteses, equivale aos fatores de êxito levantados por Ping-Ning (2006) e relatados na seção 2.5.2 deste documento.

Os métodos aplicados na implementação deste algoritmo e as heurísticas utilizadas pretendem lidar com este problema conseguindo criar histogramas de forma que os vales do histograma representem separações naturais entre comportamentos dos dados. Estes histogramas são chamados neste trabalho de **histogramas naturais**. A Figura 4-11 mostra dois exemplos de histogramas considerados naturais.

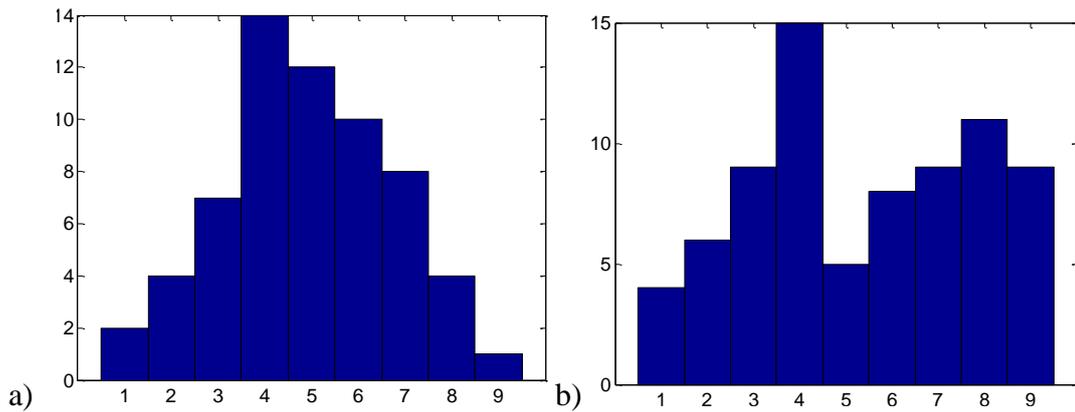


Figura 4-11 a) Exemplo de histograma natural com um comportamento único; b) Exemplo de histograma natural com dois comportamentos.

Depois de conseguir histogramas naturais, o reconhecimento de *clusters* é feito utilizando os vales do histograma, pois é esperado que, depois do tratamento, os vales delimitem os diversos comportamentos de um parâmetro.

A implementação lógica do algoritmo desenvolvido pode ser representada mediante um fluxograma. Este fluxograma geral pode ser dividido em duas partes: a clusterização unidimensional representada na Figura 4-12 e clusterização multidimensional representada na Figura 4-13.

A clusterização unidimensional (Figura 4-12) faz-se por meio de histogramas: primeiro leem-se os dados de entrada (formato “txt” separados por tabulação) e depois para cada parâmetro p_j aplica-se uma malha (passos 2, 3, 4 e 5) de criação de histograma natural. Em um histograma natural, os vales são utilizados como separadores naturais entre *clusters*.

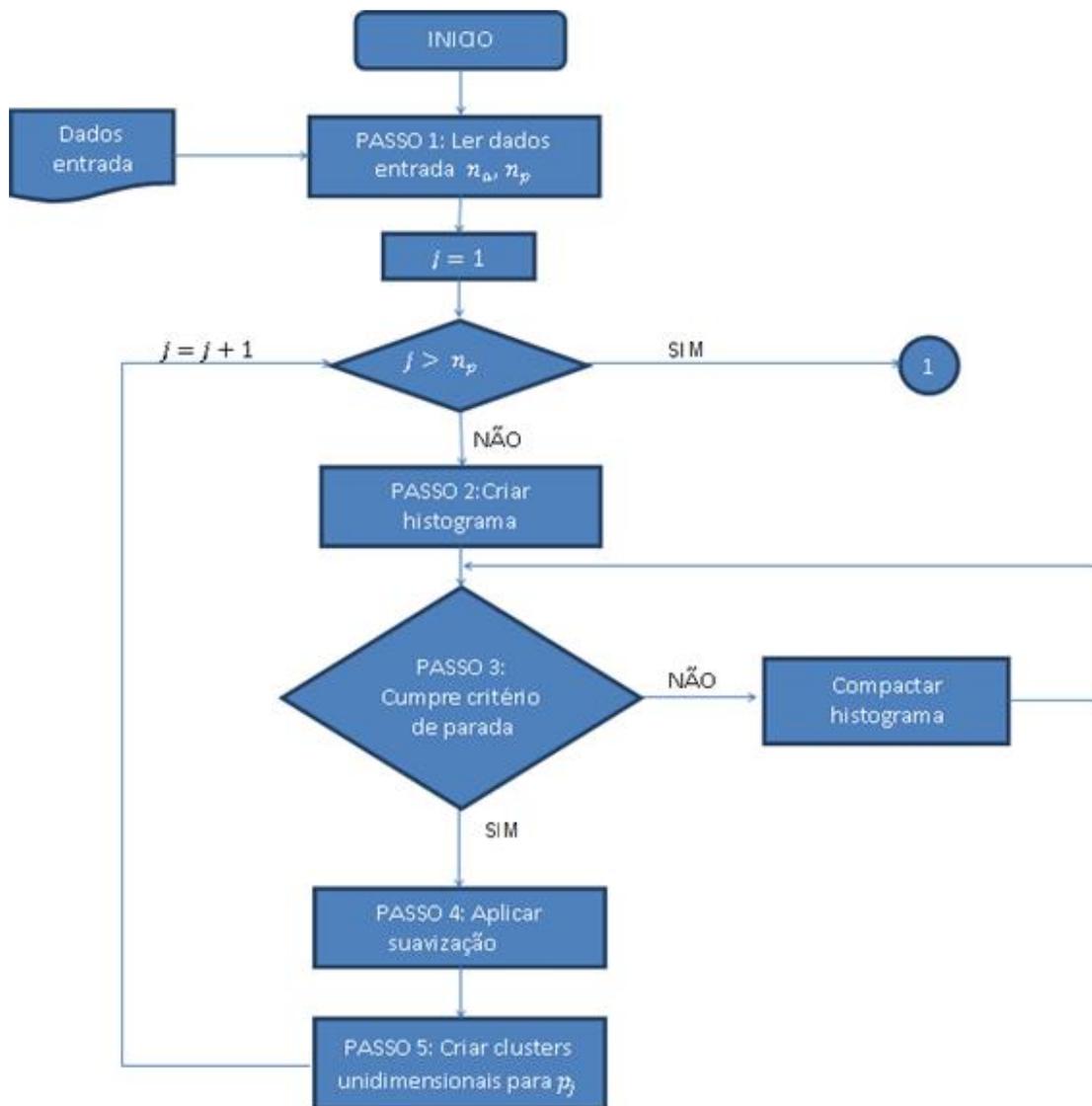


Figura 4-12. Clusterização unidimensional do algoritmo BABYLO-BR.

A clusterização multidimensional (Figura 4-13) consiste na combinação dos resultados da clusterização unidimensional para formar os *clusters* finais.

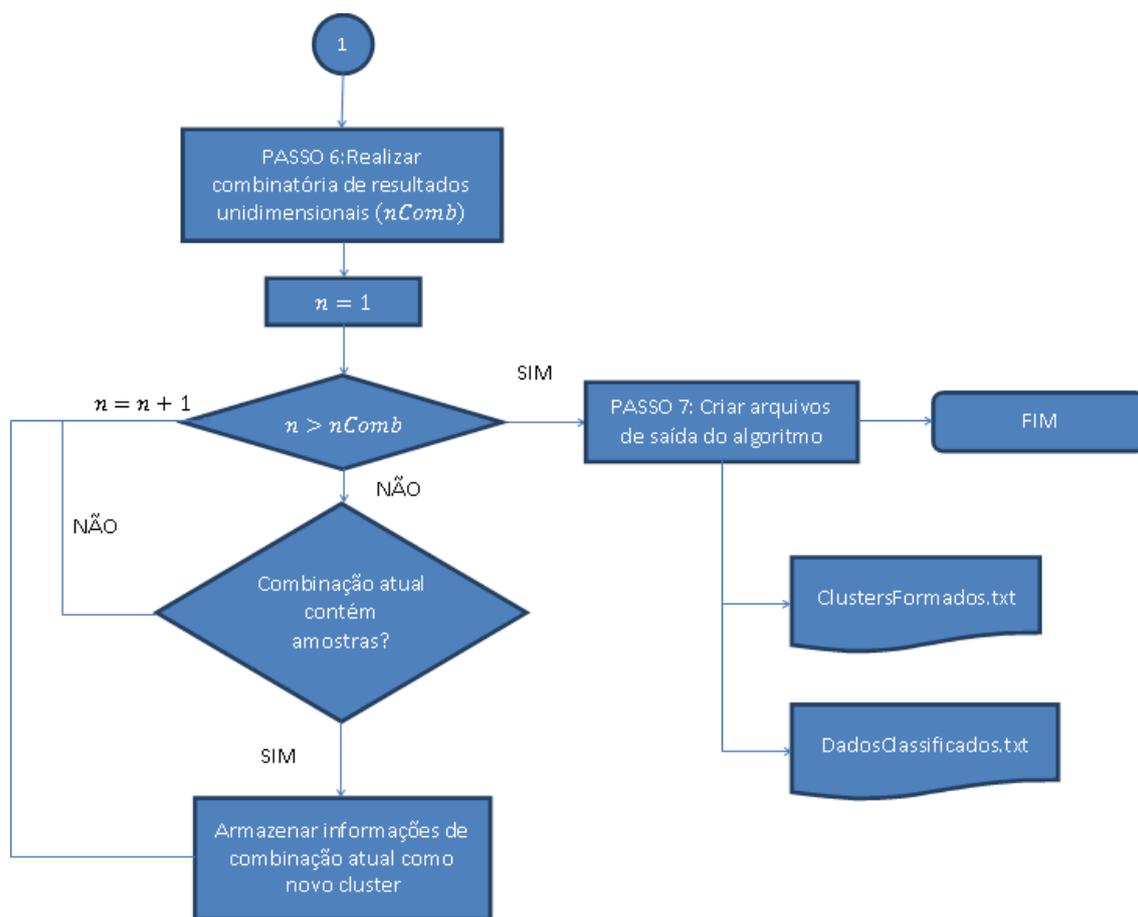


Figura 4-13. Clusterização multidimensional do algoritmo BABYLO-BR.

4.4.1. PASSO 1: Ler dados de entrada

O algoritmo BABYLO-BR recebe um conjunto de dados de entrada conforme os requisitos Req_3, Req_4, Req_5 e Req_6 (seção 3.1) com o seguinte formato:

- Extensão “.txt” com colunas separadas por tabulações.
- Cada linha contém uma amostra e cada coluna um parâmetro.
- A primeira coluna é um identificador das amostras.
- A primeira linha de cada coluna é um identificador das informações dessa coluna.

A Figura 4-14 mostra um exemplo de dados de entrada; a primeira coluna representa o identificador da amostra, no exemplo, as letras ID seguido de um número ascendente; e

a primeira linha contém o identificador dos parâmetros. A matriz de dados está formada por $n_a = 14$ amostras ($ID1, \dots, ID14$) e $n_p = 3$ parâmetros (h_1, h_2, h_3).

	A	B	C	D
1		h1	h2	h3
2	ID1	44.4861	39.5258	49.4414
3	ID2	44.4714	39.5265	49.4102
4	ID3	44.4869	39.5023	49.4311
5	ID4	44.4553	39.5252	49.4292
6	ID5	44.4678	39.5103	49.4268
7	ID6	44.5051	39.5163	49.4124
8	ID7	44.4811	39.4875	49.4298
9	ID8	44.647	39.4774	49.3729
10	ID9	44.4757	39.4875	49.4211
11	ID10	44.4881	39.5166	49.4096
12	ID11	44.4791	39.5155	49.4383
13	ID12	44.4756	39.4983	49.4295
14	ID13	44.4664	39.5265	49.4268
15	ID14	44.4869	39.5252	49.4134

Figura 4-14. Exemplo de dados de entrada do algoritmo BAbyLO-BR.

4.4.2. PASSO 2: Criar histograma unidimensional para cada parâmetro p

TÉCNICA:

Para cada parâmetro $p \in \{p_1, \dots, p_{n_p}\}$ (sendo n_p o número de parâmetros) cria-se um histograma unidimensional h conforme seção 2.5.2. Segundo os fatores de êxito definidos por Dubes (1988), o histograma criado depende fortemente do número de janelas n_s escolhido.

HEURÍSTICAS:

O **número de janelas** n_s é calculado em função das características dos dados de entrada. Primeiro, escolhe-se um número de janelas inicial $n_{inicial} = 50$ e calcula-se o número de janelas n_s como:

$$n_s = \begin{cases} \frac{\max(p) - \min(p)}{d_{min}}, & \text{se } \frac{\max(p) - \min(p)}{d_{min}} < n_{inicial} \\ n_{inicial}, & \text{se } \frac{\max(p) - \min(p)}{d_{min}} \geq n_{inicial} \end{cases} \quad (4.18)$$

sendo d_{min} a distância mínima entre duas amostras i e k quaisquer do parâmetro p :

$$d_{min} = \min (p(i) - p(k)) / \{i \neq k\} \in [1, n_a] \quad (4.19)$$

Depois, o número de janelas n_s será adaptado no PASSO 3 de acordo com as características do histograma formado.

4.4.3. PASSO 3: Comprovar cumprimento de critério de parada

TÉCNICA:

Seguindo a idéia de achar o número de janelas n_s que permitem obter o histograma natural (fator de êxito número 1 definido em Dubes (1988)), este passo é uma malha que consiste em identificar se as amostras estão suficientemente compactadas no histograma ou não.

A escolha inicial do número de janelas no histograma pode não ser adequada; por esse motivo aplicar-se-á um “critério de parada” baseado no corolário da segunda derivada (seção 2.5.4) que consiste em estudar se as variações entre janelas adjacentes da segunda derivada do histograma são menores do que um determinado limiar de parada l_{parada} . O limiar de parada é considerado o critério de parada (ANKERST, 1999) deste algoritmo.

$$h''(s_{i-1}) - h''(s_i) < l_{parada} \quad \forall i \in [2, n_s] \quad (4.20)$$

Se a diferença entre duas janelas adjacentes é maior do que o limiar, considera-se que as amostras estão muito espalhadas no histograma e que ele deve ser compactado (criar outro histograma com menos janelas).

O processo de compactação do histograma é aplicado de forma iterativa até cumprir com o critério de parada.

HEURÍSTICAS:

limiar de parada

Escolhe-se o valor do limiar de parada l_{parada} conforme a seguinte heurística dependente da variância de h'' :

$$l_{parada} = \begin{cases} \frac{n_a}{n_{s\ 10\%}} k_{limiar}, & \text{se } 2\sigma_{real}^2 > \sigma_{max}^2 \\ \frac{n_a}{n_{s\ 10\%}}, & \text{se } 2\sigma_{real}^2 \leq \sigma_{max}^2 \end{cases} \quad (4.21)$$

sendo $n_{s\ 10\%}$ o número de janelas consideradas relevantes para o cálculo:

$$n_{s\ 10\%} = s \in S / h(s) > 0.1 \max(h) \quad (4.22)$$

k_{limiar} um fator decrescente com o número de iterações de compactação ($i_{compact}$):

$$k_{limiar} = \begin{cases} \frac{8}{5} - 0.1 * i_{compact}, & \text{se } i_{compact} < 6 \\ \frac{8}{5}, & \text{se } i_{compact} \geq 6 \end{cases} \quad (4.23)$$

σ_{real}^2 o valor da variância (MORETTIN, 1995) de h'' e σ_{max}^2 o valor da variância de h'' assumindo que as n_a amostras estão acumuladas em uma única janela (ver APÊNDICE B -).

A definição de k_{limiar} foi escolhida heurísticamente com base nos dados disponíveis dos estudos de caso que foram estudados neste trabalho. Essa escolha está explicada no APÊNDICE B - deste documento.

Processo de compactação do histograma

Quando o histograma da iteração atual não cumpre com o critério de parada (equação (4.20)), o histograma é compactado, i.e., é criado outro histograma com uma porcentagem ($p_{compact}$) menor de janelas.

$$n_s = (1 - p_{compact}) n_s \quad (4.24)$$

Foi escolhido $p_{compact} = 5\%$ devido a que, com a compactação do histograma, pretende-se que os limiares que definem cada janela mudem de posição levemente, conseguindo assim uma nova distribuição das amostras nas janelas e procurando evitar mudanças bruscas entre elas.

4.4.4. PASSO 4: Aplicar suavização do histograma

Embora o processo de compactação do histograma e o critério de parada aplicado (PASSO 3) tenham sido definidos para obter o histograma natural, cabe esperar que ainda existam mudanças entre janelas que provoquem vales não esperados.

Como exemplo, veja-se a Figura 4-15. Na Figura 4-15a existe um vale não desejado em uma vizinhança de subida. Realizando a suavização do histograma espera-se que esse vale desapareça ficando uma subida regular. Na Figura 4-15b, observa-se o caso oposto, um pico inesperado em uma vizinhança de subida que provoca um vale não desejado. Do mesmo modo, aplicando suavização do histograma pretende-se eliminar o pico de forma a obter uma subida regular. Em ambos os casos essa irregularidade estaria separando em dois *clusters* um comportamento único, o que não é desejado.

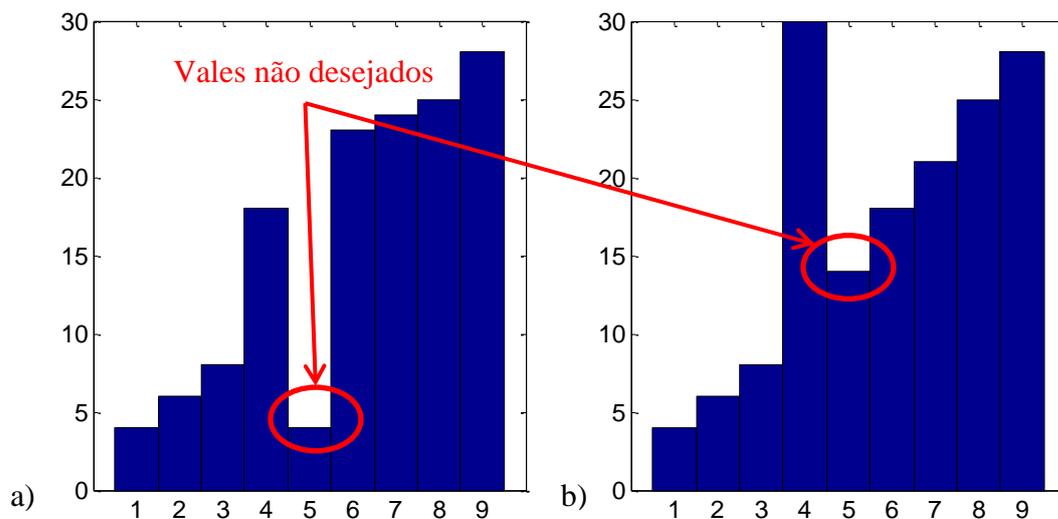


Figura 4-15. a) Exemplo de vale não desejado em vizinhança de subida, b) Exemplo de pico inesperado em vizinhança de subida e consequente vale não desejado.

TÉCNICA:

É utilizada a técnica de filtragem apresentada na Seção 2.5.5 para suavizar o contorno do histograma mediante máscaras de forma a evitar mudanças não desejadas entre janelas adjacentes.

Realizando uma similitude entre uma imagem e um histograma, pode-se considerar que um histograma h é uma imagem unidimensional (com 1 linha e n_s colunas). A máscara utilizada para suavizar o histograma é então unidimensional.

A máscara aplicada está representada pelo vetor $M \in \mathbb{R}^{n_M}$ sendo $n_M \in 2N + 1$ o número de elementos da máscara (valor ímpar). A vizinhança de aplicação da máscara é:

$$n_v = \frac{n_M - 1}{2} \quad (4.25)$$

e representa o número de janelas anteriores e posteriores que influem no valor final da janela sendo suavizada.

O histograma suavizado h_{suav} é calculado conforme definido em Gonzalez (2000) e descrito na equação (2.10) como segue:

$$h_{suav}(s_i) = \sum_{j=1}^{n_M} h(s_{i-n_v+j}) M(j) \quad (4.26)$$

O histograma resultante do processo de suavização é considerado histograma natural.

Um exemplo real do resultado do processo de suavização se mostra na Figura 4-16; antes de aplicar suavização, existem muitos vales que estariam identificando início/final de *clusters* diferentes e que não separam comportamentos naturais. Depois de aplicar o processo de suavização os vales não desejados são eliminados, ficando unicamente o vale que diferencia entre comportamentos.

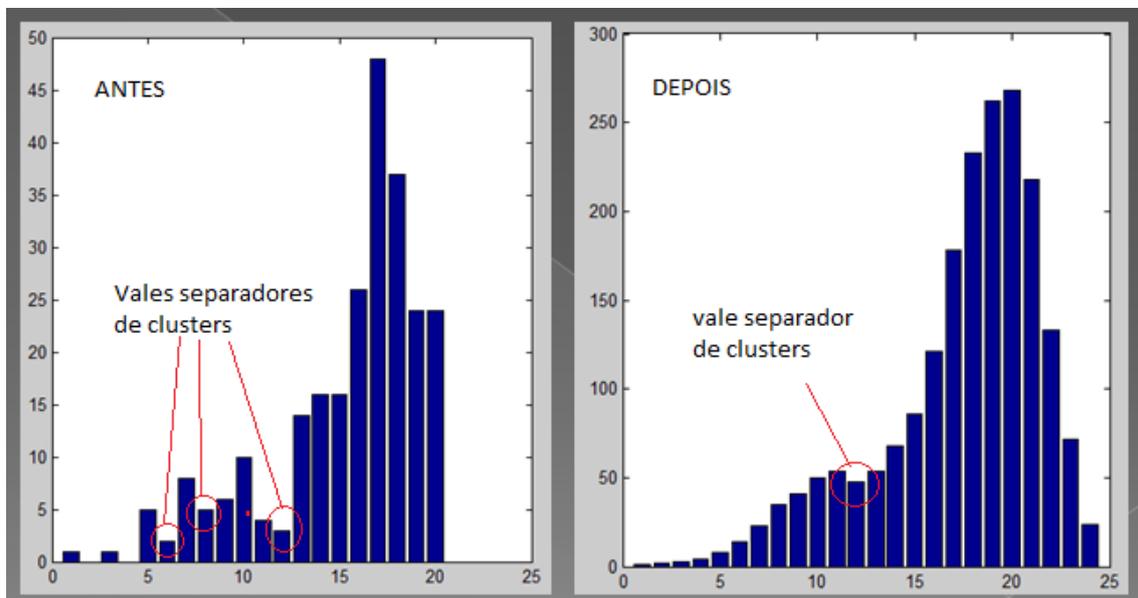


Figura 4-16. Exemplo de histograma antes e depois do processo de suavização.

HEURÍSTICA:

As máscaras escolhidas são:

$$M_1 = [0.5 \quad 1 \quad 0.5]^T \quad (4.27)$$

O número de elementos desta máscara é $n_M = 3$, pois se considera que uma vizinhança imediata ($n_v = 1$) é suficientemente abrangente em casos onde é considerado que o conjunto de amostras é confiável (fator de êxito 2 conforme visto na Seção 2.5.2).

$$M_2 = [1 \ 2 \ 2 \ 2 \ 1]^T \quad (4.28)$$

O número de elementos desta máscara é $n_M = 5$, pois se considera que, em casos onde se tem um número de amostras menor ou onde existe mais ruído/incertezas no valor delas, o histograma pode não representar o comportamento natural das amostras precisando de uma suavização maior (fator de êxito 2 conforme visto na Seção 2.5.2).

O valor das máscaras foi escolhido de forma heurística, comprovando o funcionamento da maioria dos estudos de caso onde foi aplicado.

4.4.5. PASSO 5: Criar *clusters* unidimensionais

TÉCNICA:

Uma vez criado o histograma natural, procede-se a localizar os vales para identificar os diferentes *clusters* em que serão classificadas as amostras do parâmetro em estudo.

Para realizar a identificação dos vales utiliza-se o corolário da primeira derivada definido na Seção 2.5.4, sendo que cada **mudança de sinal de negativo para positivo em h'** indica a existência de um mínimo ou vale do histograma.

Matematicamente isto é:

$$\forall s_i \in S, i > 1, \exists \text{ vale em } s_i \Leftrightarrow h'(s_i) > 0 \text{ e } h'(s_{i-1}) < 0 \quad (4.29)$$

onde a derivada do histograma h' é calculada como:

$$h'(s_i) = h(s_{i+1}) - h(s_i) \quad (4.30)$$

Um vale define o final de um *cluster*. Após isso, o seguinte valor distinto de zero define o começo de um novo *cluster*. A Figura 4-17 exemplifica este procedimento de identificação dos *clusters*.

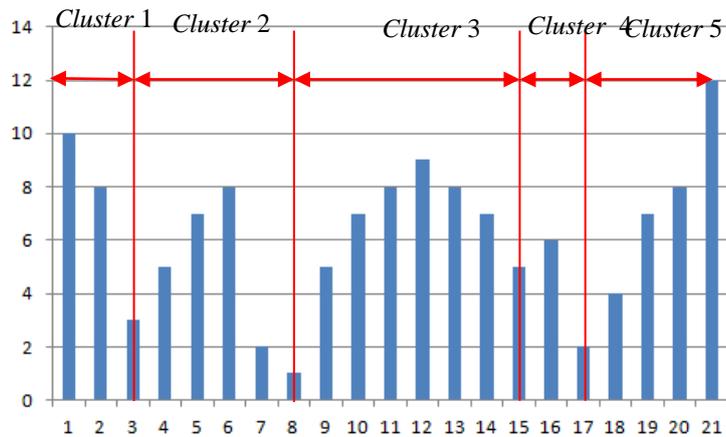


Figura 4-17. Exemplo de aplicação do corolário da primeira derivada para identificação dos *clusters*.

Como resultado desta clusterização unidimensional obtém-se o número de *clusters* (comportamentos) para cada parâmetro e os limiares (valor mínimo e máximo) que caracterizam esses *clusters*.

Este procedimento é aplicado aos n_p parâmetros que caracterizam as amostras conforme refletido no fluxograma da Figura 4-12.

HEURÍSTICA:

Não é aplicada nenhuma heurística neste passo.

4.4.6. PASSO 6: Realizar combinação de resultados unidimensionais

Este passo corresponde à clusterização multidimensional detalhada no fluxograma da Figura 4-13.

TÉCNICA:

Nas seguintes linhas detalha-se a técnica aplicada para obter os *clusters* finais em um caso bidimensional (Figura 4-18). Depois, generaliza-se esta técnica para qualquer caso multidimensional.

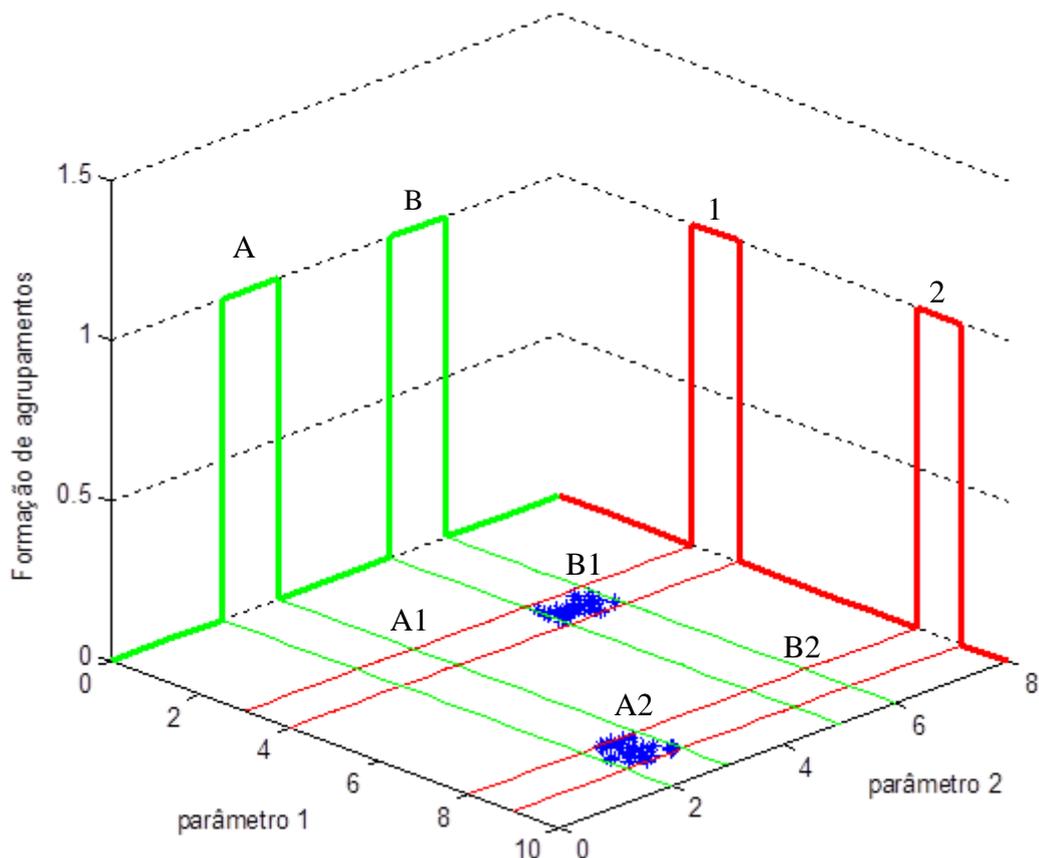


Figura 4-18. Representação gráfica da combinação de resultados unidimensionais de clusterização para um caso bidimensional.

- 1) Considera-se n_a amostras e 2 parâmetros p_1 e p_2 (amostras bidimensionais). As amostras a_i que formam a matriz de dados de entrada estão representadas na Figura 4-18 em azul com coordenadas $a_i = [p_1(i), p_2(i)]$.
- 2) Para cada parâmetro aplica-se a clusterização unidimensional (PASSOS 1 até 5 do algoritmo). Assim, obtêm-se os *clusters* 1 e 2 para p_1 e os *clusters* A e B para p_2 .
- 3) Supondo que a situação das amostras a_i é desconhecida, mas sabendo que foram formados os *clusters* A, B e 1, 2, pode-se deduzir que as amostras estão obrigatoriamente situadas em algumas das quatro possíveis combinações

bidimensionais de *clusters* (A1, A2, B1, B2), mas não necessariamente todas as combinações possuem amostras.

- 4) Assim, para terminar determina-se quais dos *clusters* (A1, A2, B1, B2) combinados possuem amostras; eles são os *clusters* finais multidimensionais (No caso da Figura 4-18, os *clusters* B1 e A2).

Generalizando este exemplo, para mais de dois parâmetros, a técnica aplicada consiste em formar todas as possíveis combinações multidimensionais de *clusters* e, em seguida, determinar quais dessas combinações possuem amostras. Aquelas que contêm amostras são declaradas *clusters* finais.

HEURÍSTICA:

Não é aplicada nenhuma heurística neste PASSO.

4.4.7. PASSO 7 Criar arquivos de saída do algoritmo

Neste passo, o algoritmo BAbyLO-BR cria os dois arquivos de saída:

	A	B	C	D	E
1		h1	h2	h3	C
2	ID1	44.4861	39.5258	49.4414	9
3	ID2	44.4714	39.5265	49.4102	9
4	ID3	44.4869	39.5023	49.4311	9
5	ID4	44.4553	39.5252	49.4292	9
6	ID5	44.4678	39.5103	49.4268	9
7	ID6	44.5051	39.5163	49.4124	9
8	ID7	44.4811	39.4875	49.4298	9
9	ID8	44.647	39.4774	49.3729	1
10	ID9	44.4757	39.4875	49.4211	9
11	ID10	44.4881	39.5166	49.4096	9
12	ID11	44.4791	39.5155	49.4383	9
13	ID12	44.4756	39.4983	49.4295	9
14	ID13	44.4664	39.5265	49.4268	9
15	ID14	44.4869	39.5252	49.4134	9

Figura 4-19. Exemplo de arquivo de saída "DadosClassificados.txt".

DadosClassificados.txt: É uma modificação do arquivo de entrada do algoritmo. Utilizando o arquivo de dados de entrada, adiciona-se uma coluna que identifica, para cada amostra, o *cluster* que a contém (identificador do *cluster*). A Figura 4-19 é o arquivo de saída formado a partir do arquivo de entrada da Figura 4-14. Nota-se que a Figura 4-14 e Figura 4-19 representam uma parte dos dados de entrada total (250 amostras no arquivo completo), sendo que nas figuras se mostram unicamente 14 amostras. As amostras da Figura 4-19 foram classificadas nos *clusters* 1 e 9.

ClustersFormados.txt: É um arquivo onde são armazenadas as características de cada *cluster*, a saber, 1) identificador do *cluster* (para relacioná-lo com o arquivo ‘DadosClassificados.txt’), 2) o número de amostras contidas nele e 3) os valores mínimo e máximo de cada parâmetro p_j que delimitam o *cluster*. A Figura 4-20 mostra o arquivo “ClustersFormados.txt” correspondente à classificação da Figura 4-19; foram criados 12 *clusters* para as 250 amostras, estando entre eles o *cluster* 1 e o *cluster* 9 referenciados anteriormente.

	A	B	C	D	E	F	G	H
1		Namostra	h1Vmin	h1Vmax	h2Vmin	h2Vmax	h3Vmin	h3Vmax
2	_1	3	44.4439	44.6678	39.4415	39.5319	49.3379	49.3935
3	_2	4	44.6678	44.9365	39.4415	39.5319	49.3379	49.3935
4	_3	7	44.9365	45.3395	39.4415	39.5319	49.3379	49.3935
5	_4	8	45.5186	46.2798	39.4415	39.5319	49.3379	49.3935
6	_5	3	46.2798	46.6381	39.4415	39.5319	49.3379	49.3935
7	_6	2	46.6381	47.3545	39.4415	39.5319	49.3379	49.3935
8	_7	4	47.3545	47.8471	39.4415	39.5319	49.3379	49.3935
9	_8	7	48.0262	48.9218	39.4415	39.5319	49.3379	49.3935
10	_9	204	44.4439	44.6678	39.4415	39.5319	49.3935	49.4438
11	10	2	45.5186	46.2798	39.4415	39.5319	49.3935	49.4438
12	11	2	46.2798	46.6381	39.4415	39.5319	49.3935	49.4438
13	12	4	46.6381	47.3545	39.4415	39.5319	49.3935	49.4438

Figura 4-20. Exemplo de arquivo de saída "ClustersFormados.txt".

O algoritmo BAbyLO-BR termina sua execução com o armazenamento desses arquivos.

Resumindo, o algoritmo BAbyLO-BR cria agrupamentos, independentemente do número de amostras contidas neles; assim, é de se esperar que existam agrupamentos

com baixo número de amostras. Essas amostras são candidatas a serem anomalias (conforme hipótese H5 definida na Seção 3.2). Da mesma forma, podem existir agrupamentos com um número grande de amostras que também estejam identificando um comportamento anômalo (conforme hipótese H4 definida na Seção 3.2). Este algoritmo não conhece a definição de comportamento normal, por isso não vai fazer diferença entre agrupamentos de comportamentos normais e anômalos (Seção 1.2).

4.4.8. **Resumo das heurísticas utilizadas no algoritmo BAbyLO-BR**

As heurísticas utilizadas neste algoritmo foram definidas com base na validação do algoritmo e nos estudos de caso onde foi aplicado.

- 1) Número de janelas n_s (Equação (4.18))
- 2) Limiar de parada l_{parada} (Equação (4.21))
- 3) Porcentagem de compactação do histograma $p_{compact}$ (Equação (4.24))
- 4) Máscara de suavização do histograma M (Equação (4.27))

5 ESTUDOS DE CASO E RESULTADOS

A definição de estudos de caso é um método muito utilizado para levantamento de requisitos, verificação, e validação de algoritmos. Este método ajuda a visualizar quais problemas podem ser resolvidos com o algoritmo em questão e até onde se quer abranger com o projeto, pois nunca é possível resolver tudo.

Os estudos de caso para aplicação deste algoritmo dependem de:

- 1) Tipos de sistemas, segundo o funcionamento deles:
 - a. Estáticos: a saída do sistema não varia com o tempo (Ex: máxima tensão esperada numa tomada).
 - b. Dinâmicos: a saída do sistema varia com o tempo (Ex: nível de água em um tanque com fontes e sumidouros ativos).
- 2) Tipos de anomalia que se pretende detectar:
 - a. Não funcionamento: o sistema não devolve valor de saída.
 - b. Mau funcionamento: o sistema devolve um valor de saída, mas ele é errado.
 - c. Intermitente: o sistema devolve um valor de saída, mas aleatoriamente ele pode ser errado.
- 3) Quantidade de dados disponíveis tanto normais quanto anômalos: refere-se à porcentagem de valores normais em relação a valores anômalos e à quantidade total de valores.

Conforme o Req_6 (seção 3.1), este algoritmo foi pensado para achar anomalias em dados estáticos, isto é, o valor esperado não varia de uma amostra para outra. Levando isto em conta, podemos aplicar o algoritmo a dois tipos de dados:

- 1) Dados provenientes de um sistema estático cujas amostras foram tomadas de uma única simulação ao longo do tempo (estatística temporal).
- 2) Dados provenientes de um sistema dinâmico cujas amostras foram tomadas de várias simulações no mesmo instante de tempo (estatística conjuntal).

Devido a esta característica, este algoritmo não diferencia anomalias do tipo intermitente de comportamentos anômalos, pois elas acontecem ao longo do tempo.

Neste capítulo são apresentados quatro estudos de caso com o objetivo de validar o algoritmo BAbyLO-BR_V06 (produto final deste trabalho): Primeiro, é apresentado um estudo de caso simples chamado de amostra de conveniência; (Detecção de Anomalias em Tubulações Hidráulicas). Depois é apresentado um estudo de caso utilizado como exemplo de aplicação na Área de Controle (Identificação de Falhas de Sensores e Atuadores da Plataforma Multi-Missão), em seguida um estudo de caso utilizado como exemplo de aplicação na Área de Sensoriamento Remoto (Identificação de Corpos d'Água em uma Imagem Landsat) e por último um estudo de caso onde o BAbyLO-BR será aplicado em uma imagem e os resultados são comparados com outro algoritmo da literatura (ARIFIN, 1974).

O estudo de caso 1 foi utilizado por ser simples e fácil de modelar e inserir falhas. Ele foi utilizado durante todas as fases de desenvolvimento para comprovar o funcionamento dos diferentes algoritmos 'protótipo' desenvolvidos antes de chegar ao algoritmo BAbyLO-BR.

O estudo de caso 2 foi escolhido como exemplo de aplicação na Área de Controle. Este estudo de caso demonstra que, usando um algoritmo genérico como é o algoritmo BAbyLO-BR, podem ser levantados padrões de comportamento.

O estudo de caso 3 foi escolhido por várias razões, a saber; 1) para testar a generalidade de aplicação do algoritmo, 2) por serem as características das amostras de entrada diferentes às dos outros estudos de caso (unidimensionais, grande número de amostras, dados do tipo integral, entre outros).

O estudo de caso 4 foi escolhido para reforçar a validação do algoritmo mediante a comparação com outro algoritmo da literatura que utiliza a mesma técnica (ARIFIN, 1974).

O algoritmo foi aplicado a mais dois estudos de caso reais, mas dos quais não se tem conhecimento prévio dos resultados esperados, e que foram adicionados neste trabalho

no APÊNDICE C - (Estes estudos de caso não aportam mais informações ao trabalho, por esse motivo foram adicionados em um apêndice).

Cabe destacar que foram estudadas unicamente falhas simples (não se realizou estudo de comportamento dos sistemas sujeitos a várias falhas). Em todos os estudos de caso aqui apresentados foi utilizada a máscara da equação (4.27), exceto para dois deles (indicado de forma expressa no estudo de caso específico) que foi utilizada a máscara da equação (4.28).

5.1. Detecção de Anomalias em Tubulações Hidráulicas

O estudo de caso aqui apresentado foi introduzido na Seção 4.1 deste documento. O sistema é simulado aplicando as seguintes condições iniciais: em $t = 0s$ os tanques estão vazios $[h_1, h_2, h_3] = [0, 0, 0]$ e são ativadas as duas fontes $[q_{i1}, q_{i2}]$ com vazões constantes de $10 m^3/s$.

Cabe lembrar que se simulam dois tipos de falhas: o entupimento da tubulação de resistência R_1 e a perda do sensor de medida da altura de água do tanque 1 (h_1). Essas falhas são simuladas em dois instantes de tempo diferentes, a saber, $t = 0s$ e $t = 15s$. O objetivo é comprovar se a propagação da falha poderia afetar o resultado da classificação.

Assim, estudam-se 5 casos de comportamento:

- Caso 1) Comportamento normal.
- Caso 2) Entupimento da tubulação em $t = 0s$.
- Caso 3) Entupimento da tubulação em $t = 15s$.
- Caso 4) Falha do sensor h_1 em $t = 0s$.
- Caso 5) Falha do sensor h_1 em $t = 15s$.

O Caso 1 representa o funcionamento normal, os Casos 2 e 3 representam mau funcionamento, e os Casos 4 e 5 representam não funcionamento.

Para cada caso, são realizadas 250 simulações de 30 segundos de duração com 20% de probabilidade de falha.

Por ser um modelo dinâmico, aplicou-se a primeira hipótese H1 (ver seção 3.2) para tratar o sistema de forma estática, escolhendo o instante de tempo $t_k = 30s$. Assim, para cada simulação, extraíram-se os valores de $[h_1, h_2, h_3]$ em $t_k = 30s$.

A matriz de dados de entrada do algoritmo de classificação é a matriz $A \in \mathcal{M}^{n_a \times n_p}$ com $n_a = 250$ (uma amostra por simulação) e $n_p = 3$ (h_1, h_2, h_3).

Um exemplo de amostras com e sem falha é mostrado na Figura 5-1 onde foi simulada a falha de entupimento na tubulação R_1 ; na segunda coluna chamada de “falha” mostra-se a porcentagem de entupimento, sendo que o valor zero representa uma amostra sem falha e um valor distinto de zero representa uma amostra com falha. Pode-se observar que quanto maior for o entupimento, maior será quantidade de água que fica no tanque 1 (h_1) e menor a que fica nos tanques 2 e 3 (h_2 e h_3).

	A	B	C	D	E
1		falha	h1	h2	h3
2	ID1	0	44.4861	39.5258	49.4414
3	ID2	0	44.4714	39.5265	49.4102
4	ID3	0	44.4869	39.5023	49.4311
5	ID4	0	44.4553	39.5252	49.4292
6	ID5	0	44.4678	39.5103	49.4268
7	ID6	0.009015	44.5051	39.5163	49.4124
8	ID7	0	44.4811	39.4875	49.4298
9	ID8	0.08618	44.647	39.4774	49.3729
10	ID9	0	44.4757	39.4875	49.4211
11	ID10	0	44.4881	39.5166	49.4096
12	ID11	0	44.4791	39.5155	49.4383
13	ID12	0	44.4756	39.4983	49.4295
14	ID13	0	44.4664	39.5265	49.4268
15	ID14	0	44.4869	39.5252	49.4134
16	ID15	0	44.4838	39.5252	49.428
17	ID16	0	44.485	39.5265	49.4181
18	ID17	0.516643	46.1746	39.4934	49.3854

Figura 5-1. Exemplo de amostras com e sem falha no estudo de caso de tubulações hidráulicas.

5.1.1. Caso 1) Comportamento normal

Neste caso, durante toda a simulação, o sistema funciona corretamente, sem que falhas sejam introduzidas. Espera-se que o algoritmo classifique todas as amostras num mesmo agrupado. A Figura 5-2 mostra o conjunto de amostras de entrada sem classificar, todas elas representadas na cor azul por não existirem falhas.

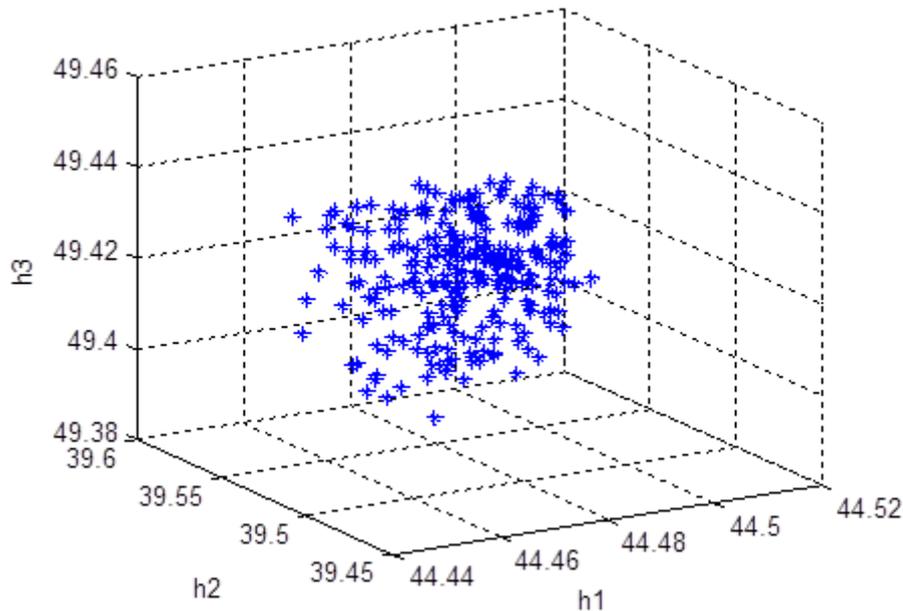


Figura 5-2. Amostras sem classificar.

Aplicando o algoritmo com a máscara padrão (vide equação (4.27)) o resultado da classificação é conforme mostrado na Figura 5-3. Pode-se observar que houve uma separação em dois *clusters* diferentes na dimensão que caracteriza a altura do tanque h_2 .

Assim, foram criados dois *clusters* para um único comportamento esperado. Para evitar esta separação aplicou-se o algoritmo utilizando um filtro mais severo (vide equação (4.28)) com o objetivo de evitar o vale inesperado que surgiu na dimensão de característica h_2 . A Figura 4-15 e Figura 4-16 representam dois exemplos teóricos do acontecido neste estudo de caso. Lembre-se que a suavização foi implementada precisamente para evitar este problema.

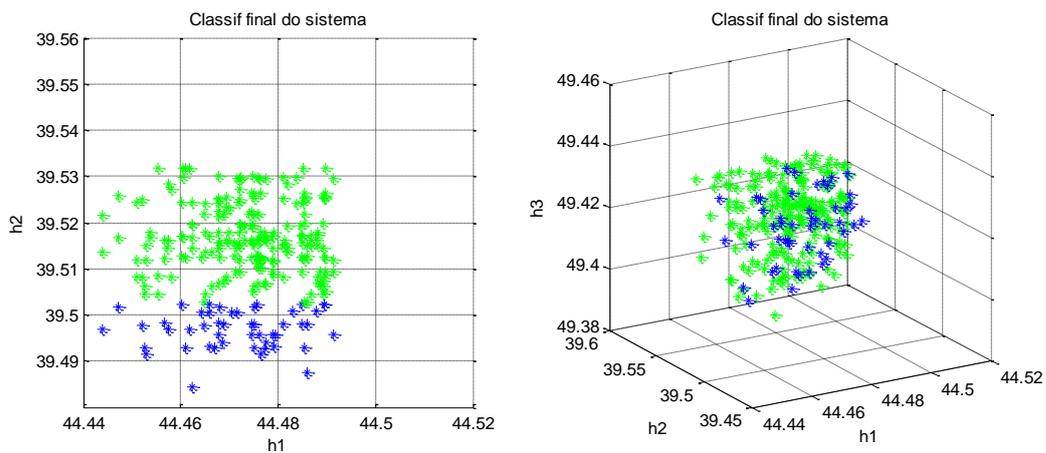


Figura 5-3. Classificação das amostras usando a máscara de características $[0 \ 0.5 \ 1 \ 0.5 \ 0]$ a) com visualização em duas dimensões e b) com visualização em três dimensões.

Assim, aplicando a máscara mais severa obteve-se a classificação esperada, Figura 5-4, com as 250 amostras agrupadas em um único cluster. A Tabela 5-1 representa o número de acertos e falhas do algoritmo.

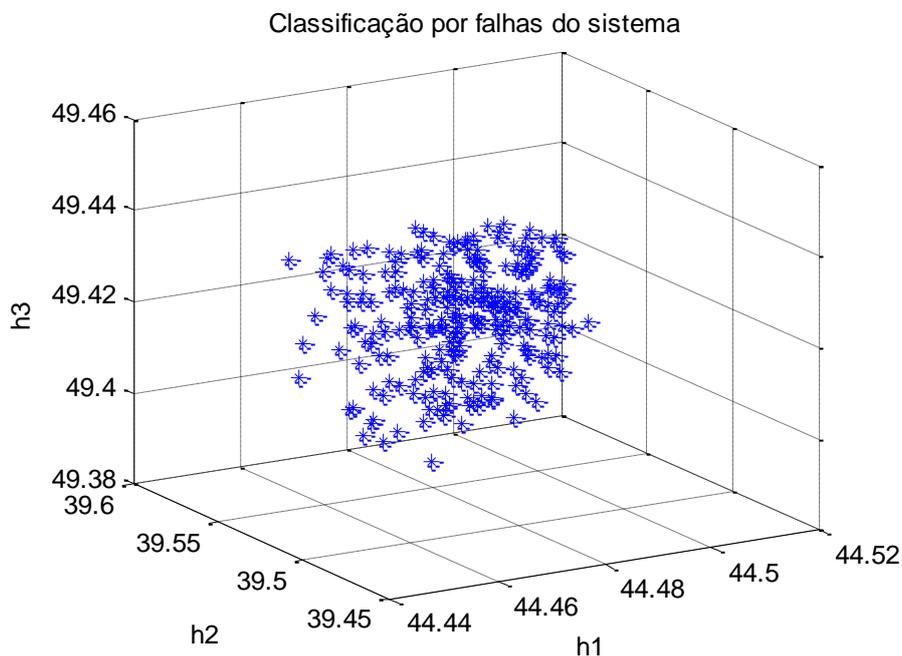


Figura 5-4. Classificação das amostras usando a máscara de características $[1 \ 2 \ 2 \ 2 \ 1]$.

Tabela 5-1. Teste de hipóteses para caso de comportamento normal

	Existe Falha	Não existe Falha
O algoritmo localizou falha	0	0
O algoritmo não localizou falha	0	250

5.1.2. Caso 2) Entupimento da tubulação em $t=0s$

Neste caso, desde o início das simulações, existe 20% de probabilidade de falha na tubulação de característica R_1 . Lembre-se que o grau de entupimento da tubulação é aleatório pelo que em $t = 30s$ tem-se vários comportamentos anômalos com valores diferentes nas saídas devido a que o entupimento foi diferente para cada um dos casos falhados.

Neste caso, espera-se que o algoritmo crie um comportamento abrangendo as amostras sem falha na tubulação. As falhas podem ou não ser agrupadas em vários comportamentos, devido à distribuição aleatória delas.

A Figura 5-5 representa a propagação da falha no parâmetro h_1 entre uma amostra normal e uma falhada.

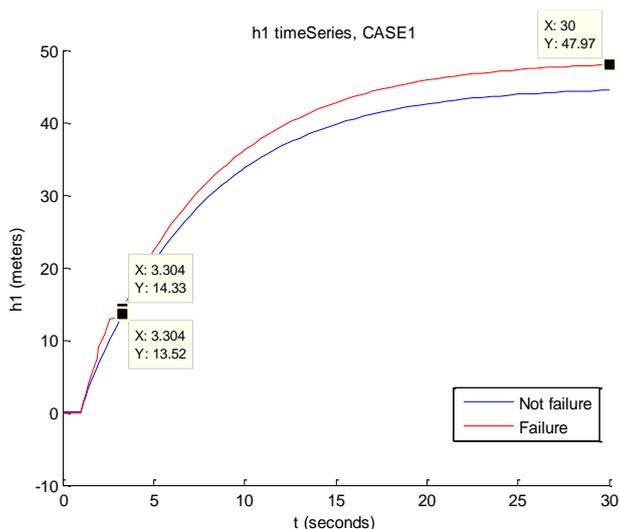


Figura 5-5. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de mau funcionamento.

A Figura 5-6a representa a distribuição das amostras com falha (vermelho) e sem falha (azul) em $t=30s$. A Figura 5-6b representa a classificação feita pelo algoritmo, onde cada cor representa um *cluster*³.

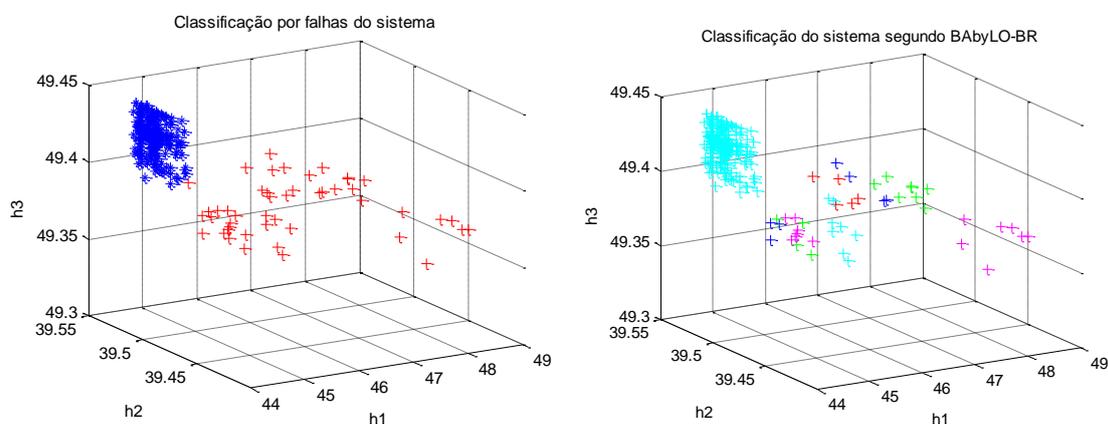


Figura 5-6. Falha de entupimento em $t=0s$. a) Distribuição espacial das amostras em $t=30s$. b) Resultados da classificação segundo o algoritmo BABYLO-BR.

A Tabela 5-2, representa o número de acertos e falhas do algoritmo⁴. Pode-se observar que o algoritmo cria corretamente o *cluster* de amostras sem falha e separa as amostras falhadas em vários *clusters* devido à aleatoriedade da falha simulada. Segundo a Tabela 5-2 o algoritmo classifica duas amostras falhadas no *cluster* normal, visíveis também na comparação entre os gráficos a e b da Figura 5-6. Isto acontece pois a aleatoriedade da falha faz com que existam falhas com porcentagem de entupimento muito pequeno (entupimento ≈ 0) o que quer dizer que a magnitude de propagação da falha é

³ Note-se que, devido à falta de quantidade de cores diferentes na representação da figura, pode acontecer que seja utilizado a mesma cor para dois *clusters* diferentes. O leitor deverá interpretar a mesma cor como representante de dois ou mais *clusters* diferentes, quando as amostras apareçam distanciadas entre elas. Esta nota é aplicável a todo o capítulo de resultados.

⁴ O algoritmo não tem a capacidade de discernir entre as amostras consideradas falhadas e as que não o são. Esta tabela é calculada após a aplicação do algoritmo estudando o arquivo de saída “clustersFormados.txt” em conjunto com a caracterização de amostras falhadas ou não falhadas (conhecido graças à simulação do sistema em Simulink). Assim, o *cluster* considerado como *cluster* de comportamento normal é aquele com maior número de amostras, pois existe 20% de porcentagem de falha. Os outros *clusters* são agrupados considerando que todos eles representam amostras falhadas. Esta nota é aplicável a todos os estudos de teste de hipóteses realizados neste capítulo.

comparável ao próprio ruído modelado no sensor, fazendo com que o algoritmo não a localize e a considere normal.

Tabela 5-2. Teste de hipóteses para caso de entupimento em $t=0s$.

	Existe Falha	Não existe Falha
O algoritmo localizou falha	46	0
O algoritmo não localizou falha	2	202

5.1.3. Caso 3) Entupimento da tubulação em $t=15s$

No início da simulação o sistema funciona corretamente até $t = 15s$, quando começa a funcionar de forma errada sofrendo entupimento na tubulação de característica R_1 com 20% de probabilidade conforme a Figura 5-7. A porcentagem de entupimento é aleatória.

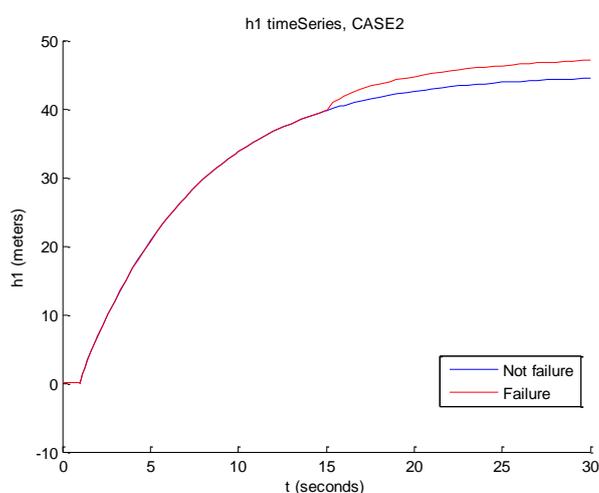


Figura 5-7. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de normal e mau funcionamento.

Como no caso anterior, espera-se que o algoritmo separe em *clusters* diferentes elementos com e sem falha. Os elementos falhados podem ser agrupados em um ou vários *clusters* devido à aleatoriedade da porcentagem de entupimento.

A Figura 5-8a representa a distribuição das amostras com falha (vermelho) e sem falha (azul) em $t=30s$. A Figura 5-8b representa a classificação feita pelo algoritmo, onde cada cor corresponde a um *cluster*.

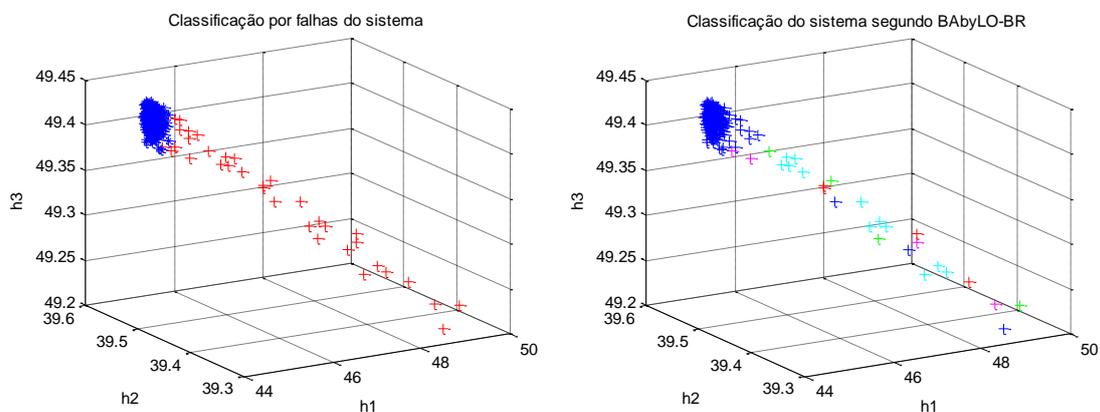


Figura 5-8. Falha de entupimento em $t=15s$. a) Distribuição espacial das amostras em $t=30s$. b) Resultados da classificação segundo o algoritmo BAbyLO-BR.

A Tabela 5-3 apresenta o número de acertos e falhas do algoritmo. Observa-se que, neste caso, existe um maior número de falsos negativos. Isto é esperado, pois a falha, se acontecer, começa a se propagar no instante $t=15s$, pelo que, para entupimentos pequenos, no instante $t=30s$ a diferença entre o valor normal e o valor falhado é muito pequena e o algoritmo não consegue diferenciá-la, pois a confunde com o ruído modelado no sensor de medida da saída. Comparando as deduções feitas no caso anterior e neste caso, pode-se confirmar que, sendo $t = 30s$ o instante de medida, quanto mais perto de $t = 30s$ acontecer a falha, menor capacidade terá o algoritmo de discernir entre amostras falhadas e não falhadas, devido à diferença pequena no valor final da saída ($[h_1, h_2, h_3]_{falhado} \cong [h_1, h_2, h_3]_{sem\ falha}$).

Tabela 5-3. Teste de hipóteses para caso de entupimento em $t=15s$.

	Existe Falha	Não existe Falha
O algoritmo localizou falha	27	0
O algoritmo não localizou falha	10	213

5.1.4. Caso 4) Falha do sensor h_1 em $t=0s$

Desde o início da simulação existem 20% de probabilidade de que o sensor que mede a saída h_1 não funcione. Este caso representa a perda do sensor. A Figura 5-9 representa a propagação da falha no parâmetro h_1 para uma amostra normal e uma falhada.

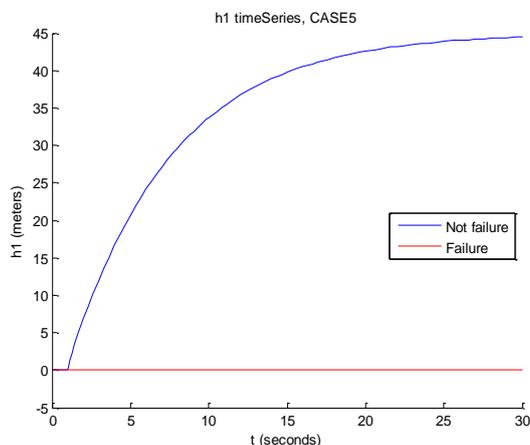


Figura 5-9. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de ‘não funcionamento: perda de sensor’.

A Figura 5-10 representa a distribuição das amostras com falha (vermelho) e sem falha (azul) em $t=30s$. Devido a que a falha simulada não tem comportamento aleatório (sensor inativo), elas criam um conjunto diferenciável. Por isso, é esperado que o algoritmo, além de separar as amostras falhadas do comportamento normal (*cluster* de cor azul na Figura 5-10), agrupe elas em um único comportamento (*cluster* de cor vermelha na Figura 5-10).

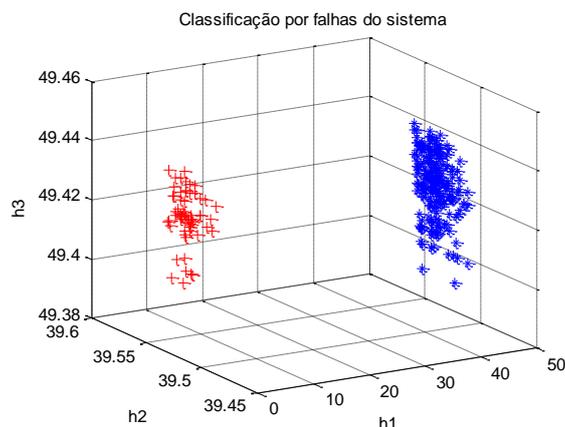


Figura 5-10. Perda do sensor em $t=0s$. Distribuição espacial das amostras em $t=30s$.

Neste caso, acontece a mesma situação que a percebida no caso 1; a aplicação do algoritmo com a máscara padrão $[0 \ 0.5 \ 1 \ 0.5 \ 0]$ produz a separação do mesmo comportamento em dois clusters diferentes (Figura 5-11a). Para conseguir a classificação esperada (conforme Figura 5-10), precisou-se aplicar o algoritmo com a máscara de suavização mais severa $[1 \ 2 \ 2 \ 2 \ 1]$, representada na equação (4.28) (Figura 5-11b).

Assim, a Figura 5-11 representa a classificação feita pelo algoritmo BAbyLO-BR aplicando as duas máscaras. No gráfico Figura 5-11a, as amostras de ambos os comportamentos são separadas em dois clusters (rosa e azul para amostras falhadas e ciano e verde para amostras sem falha). Já na Figura 5-11b, o algoritmo consegue separar as amostras falhadas do comportamento normal conforme esperado.

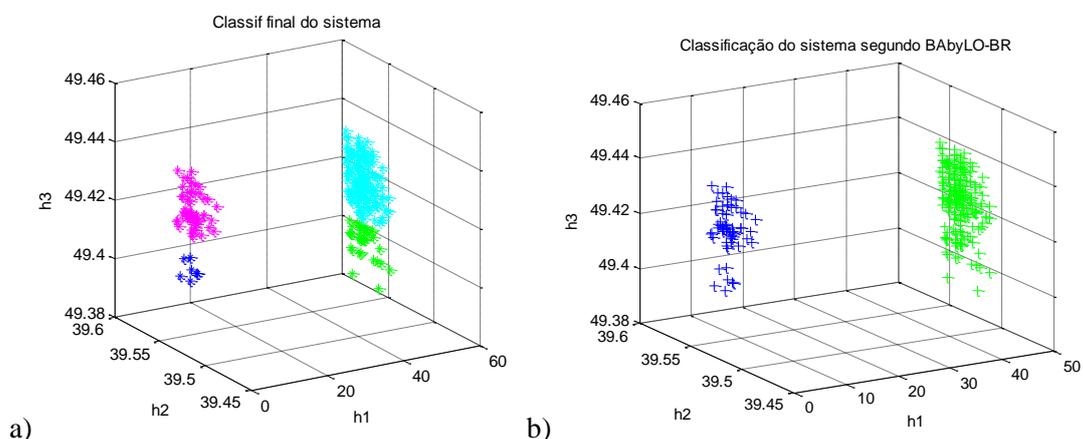


Figura 5-11. Resultados de classificação segundo algoritmo BAbyLO-BR. a) usando máscara padrão $[0 \ 0.5 \ 1 \ 0.5 \ 0]$ e b) usando máscara mais severa $[1 \ 2 \ 2 \ 2 \ 1]$.

A Tabela 5-4 apresenta o número de acertos e falhas do algoritmo usando a máscara de característica $[1 \ 2 \ 2 \ 2 \ 1]$. O algoritmo não comete nenhum erro na classificação e cria dois *clusters* agrupando as amostras com e sem falha.

Tabela 5-4. Teste de hipóteses para caso de perda de sensor em $t=0s$.

	Existe Falha	Não existe Falha
O algoritmo localizou falha	61	0
O algoritmo não localizou falha	0	189

5.1.5. Caso 5) Falha do sensor h_1 em $t=15s$

No início da simulação o sistema funciona corretamente até $t = 15s$ quando acontece a perda do sensor que mede o valor h_1 . A Figura 5-12 representa a evolução do parâmetro h_1 entre uma amostra normal e uma falhada ao longo do tempo.

Como aconteceu no caso 4, como a falha simulada não tem comportamento aleatório, espera-se que o algoritmo separe amostras com falha e amostras sem falha formando dois *clusters*.

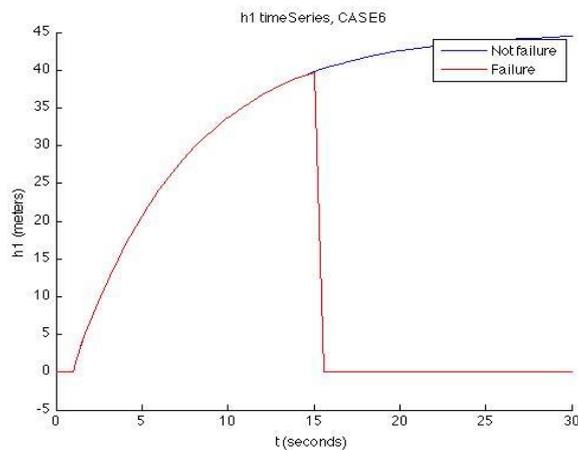


Figura 5-12. Evolução temporal do parâmetro h_1 de uma amostra normal e uma falhada em caso de 'normal e não funcionamento: perda do sensor'.

A Figura 5-13a representa a distribuição das amostras com falha (vermelho) e sem falha (azul) em $t=30s$. A Figura 5-13b representa a classificação feita pelo algoritmo, com dois comportamentos em azul e em verde conforme esperado e repetindo o resultado do caso 5.

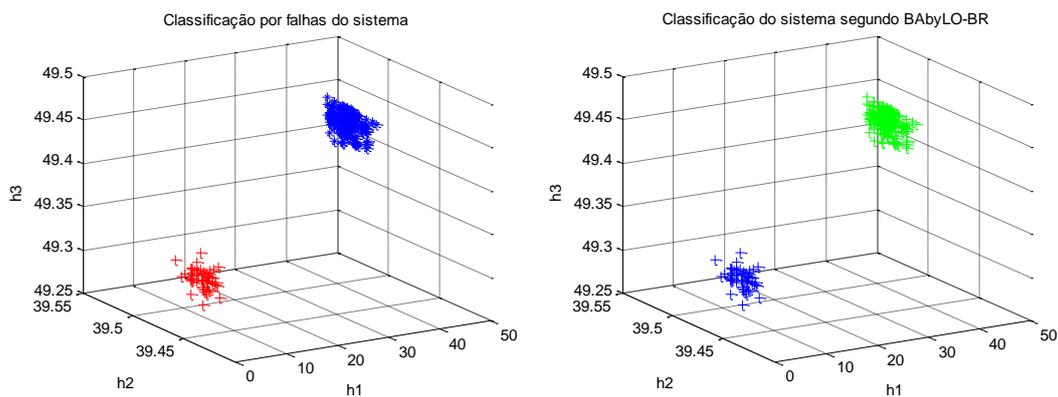


Figura 5-13. Perda do sensor em $t=15s$. a) Distribuição espacial das amostras em $t=30s$. b) Resultados da classificação segundo o algoritmo BAbYLO-BR.

Para finalizar, a Tabela 5-5 representa o número de acertos e falhas do algoritmo neste caso. O algoritmo não comete nenhum erro na classificação e cria dois *clusters* agrupando as amostras com e sem falha conforme esperado. Neste caso, mesmo a falha acontecendo em $t=15s$, por ter ela um comportamento característico, o algoritmo consegue discernir amostras com e sem falha, sem errar.

Tabela 5-5. Teste de hipóteses para caso de perda de sensor em $t=15s$.

	Existe Falha	Não existe Falha
O algoritmo localizou falha	61	0
O algoritmo não localizou falha	0	189

5.1.6. Utilidade da aplicação do algoritmo

Este estudo de caso foi utilizado como validação inicial do algoritmo. Observou-se que o algoritmo teve a capacidade de separar corretamente os casos falhados dos casos corretos em quase todas as falhas simuladas.

Nos casos 1) e 4) foi necessário utilizar uma máscara mais severa (equação (4.28)) para conseguir a separação nos clusters esperados. Assim, este estudo de caso serviu para validar principalmente as heurísticas do algoritmo percebendo que a heurística que mais afeta aos resultados é a definição da máscara para criação do histograma natural.

5.2. Identificação de falhas de sensores e atuadores da Plataforma Multi-Missão

A Plataforma Multi-Missão (PMM) foi concebida para ser uma plataforma modular, capaz de servir como base para várias missões científicas, como aquelas de comunicação e de observação da Terra segundo baixas órbitas (terrestres). A PMM é primariamente constituída de subsistemas que fornecem o essencial para o funcionamento do satélite, assim como meios de suporte para a integração de carga útil que será escolhida de acordo com a missão que o satélite irá desempenhar (LEITE, 2007).

O Subsistema de Controle de Atitude e de Órbita – SCAO, parte do Subsistema de Controle de Atitude e Tratamento de Dados (*Attitude Control and Data Handling – ACDH*), será o foco deste estudo de caso. Este subsistema é constituído por classes de componentes:

- Computador de bordo;
- Conjunto de sensores: 2 magnetômetros, 1 unidade de medida inercial (equipada com redundância interna), 8 sensores solares, 2 sensores de estrelas e 2 receptores GPS;
- Conjunto de atuadores: composto por 3 bobinas magnéticas, 1 para cada eixo de manobra, e 4 rodas de reação, 3 para cada eixo de manobra e 1 redundante, disposta simetricamente entre os eixos de manobra.

O SCAO da PMM tem sido e continua sendo objeto de muitos trabalhos (LOPES et al., 2001; MOREIRA, 2005; GOBATO, 2006; LEITE, 2007; (LUSTOSA, 2008; MANELLI NETO, 2011; AMARAL, 2013). Atualmente, é desenvolvido na própria DMC/CMC um trabalho de doutorado (SIQUEIRA, 2014) que visa tratar toda a cadeia para tolerância a falhas, desde a detecção e o diagnóstico de falhas (*Fault Detection and Diagnosis – FDD*), até a reconfiguração (*Fault Reconfiguration*) automática (ou, em “tempo real”) do modo falhado. Nele, são modeladas as dinâmicas em modo normal e falhado, para a plataforma, considerando-se um subconjunto dos seus sensores (giroscópio e sensor de estrelas) e atuadores (rodas de reação).

O modelo simulado do SCAO da PMM em Modo Nominal é conforme mostrado na Figura 5-14, com as seguintes anotações:

- O modelo da planta é (de dinâmica) não-linear e contempla os acoplamentos entre a planta e os atuadores (rodas de reação);
- Os ganhos K-LQR são calculados por meio de modelo de dinâmica linearizada (eixos desacoplados);
- A referência de entrada do controle é de horizonte finito, de acordo com o requisito de tempo para a manobra de de-tumble simulada (vide Seção 5.2.2);
- Os giroscópios são sujeitos a ruído, a *bias* e a fator de escala;
- Os sensores de estrelas são sujeitos a *ruído* e a fator de escala;
- Os torques de perturbação são modelados como constantes.

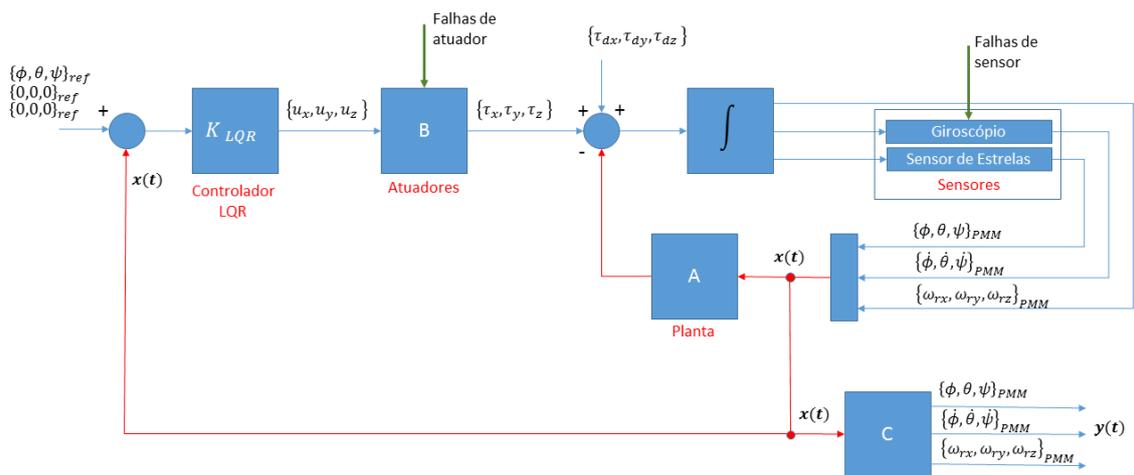


Figura 5-14. Modelo do SCAO da plataforma PMM em Modo Nominal.

5.2.1. Objetivos de detecção do algoritmo

O objetivo deste estudo de caso é estudar as dinâmicas em modo normal e falhado modeladas em Siqueira (2014) e comprovar se o algoritmo BAByLO-BR é capaz de:

- 1) Detectar a falha = discernir **entre o comportamento devido à condição normal e aqueles devidos a condições falhadas** da PMM: se o algoritmo for bem sucedido, pode-se calcular um *cluster* que caracterize o comportamento normal

da manobra, gerando assim, uma assinatura do modo normal. Os parâmetros desta assinatura permitirão a detecção de comportamento anômalo, devido a falhas de sensor ou de atuador. Desta forma, saber-se-á se ocorreu falha durante a manobra.

- 2) Isolar e identificar a falha = discernir **entre diferentes tipos de falha**: se o algoritmo for bem sucedido, pode-se calcular um *cluster* que caracterize cada tipo de falha, seja ela de sensor ou de atuador (gerando assim, uma assinatura para cada falha contida no repertório). Isto permitirá não somente detectar e isolar a falha, mas também identificá-la, para posterior reconfiguração do sistema.

5.2.2. Descrição da manobra simulada

A manobra simulada visa reproduzir uma condição hipotética de *de-tumble* (WERTZ, 1978), que consiste em que o satélite adquira a atitude (orientação espacial, inercial) necessária para sua entrada em operação, após ter ocorrido a separação do veículo lançador.

Para este estudo de caso, arbitra-se que o *de-tumble* seja uma rotação *quase*-simultânea nos 3 eixos de controle do satélite (“*quase*” porque tanto na simulação quanto no mundo real, estão presentes discretização no tempo, quantização em amplitude e atraso nas relações de entrada/saída do sistema, característicos dos sistemas digitais).

Neste trabalho, todos os eixos partem de 30° (em $t = 0s$) e devem, por requisito (GOBATO, 2006), terem finalizado a manobra (isto é, terem apontamento de 0° nos 3 eixos e entrarem em modo de manutenção de apontamento e atitude) num tempo máximo de 180s (GOBATO, 2006).

Podem ser descritas 3 fases importantes neste “*de-tumble*”:

- $0 < t < 100s$: a PMM acelera até atingir as velocidades angulares máximas;
- $100 s < t < 180s$: a PMM é desacelerada para estabilizar as velocidades angulares e seus eixos principais ao redor de zero (inercial);
- $t > 180s$: a PMM passa a atuar com manutenção de apontamento e atitude.

As simulações contemplam um tempo total de 500s e utilizam como estratégia de controle um controlador ótimo LQT (*Linear Quadratic Tracker* = Rastreador Linear Quadrático).

Conforme pode ser observado nos trabalhos de Gobato (2006), Leite (2007), Lustosa (2008), Manelli Neto (2011) e Amaral (2013), a PMM é um sistema físico de dinâmica lenta. Pode-se esperar que suas respostas não sejam abruptas, se comparadas numericamente a respostas de sistemas mais rápidos, como aeronaves e veículos atmosféricos autônomos. No APÊNDICE D - é explicada esta dinâmica.

Estas características de “bom comportamento”, especialmente na fase de manutenção de órbita, tornam interessante o estudo do algoritmo. Os parâmetros de saída da simulação que caracterizam a manobra realizada são 12, a saber:

- $[\varphi, \theta, \psi]$ – são os ângulos de atitude entre o referencial orbital da PMM e o referencial “inercial” localizado no centro da Terra (*Earth Centered “Inertial”* – ECI), obtidas por meio da indicação do sensor de estrelas, em [rad]. Por hipótese, assume-se que as coordenadas medidas são internamente convertidas para o ECI;
- $[\dot{\varphi}, \dot{\theta}, \dot{\psi}]$ – são as taxas angulares entre o referencial orbital da PMM, obtidas por indicação do giroscópio, diretamente no referencial “inercial” (ECI), em [rad/s];
- $[V_x, V_y, V_z]$ – são os valores dados pela saída do controlador para cada uma das 3 rodas de reação, em [V]. Por hipótese, a saída do controlador é diretamente utilizada, sem que haja uma unidade de potência entre ele e os atuadores em questão;
- $[\omega_x, \omega_y, \omega_z]$ – são as velocidades das rodas de reação em [rpm].

5.2.3. Descrição das falhas simuladas

Para os fins deste estudo de caso, foram simuladas as falhas no canal do eixo X (rolamento) do giroscópio, assim como na roda de reação do mesmo eixo. O comportamento dos demais eixos, Y e Z, devido à propagação das falhas neles ocorridas, seria análogo àquele obtido para o eixo X (tanto no caso de sensores, como

no caso de atuadores). A diferença estaria na velocidade e na resultante de acoplamento das perturbações resultantes entre eixos.

A fim de entender como a propagação da falha ocorreria e assim, também como seriam formados os diferentes *clusters*, cada uma das falhas foi injetada nos instantes $t = \{15, 30, 50, 250\}$ s, tanto para sensores quanto para atuadores.

As falhas simuladas no giroscópio, conforme o desenvolvimento de Siqueira (2014) são:

- **Indicação de valor constante, fixado em zero:** é reconhecida neste trabalho como “F1”. Consiste na perda da leitura de um determinado canal de sensor, sendo o “zero” modelado como um ruído branco de intensidade (potência) aumentada com relação ao ruído presente em modo normal.
- **Indicação de valor constante, congelado no último valor válido:** é reconhecida como “F2”. É um caso análogo ao “F1”, mas o ruído do modo falhado é acrescido da última leitura válida realizada - valor constante, congelado na última leitura válida.
- **Indicação de valor sujeito a deriva de offset:** é reconhecida como “F3”. Consiste em uma falha de natureza aditiva, na qual um valor indesejado é adicionado ao valor real medido, de maneira linearmente crescente com o tempo.
- **Indicação de valor sujeito a deriva de fator de escala:** é reconhecida como “F4”. Consiste em uma falha de natureza multiplicativa, na qual o fator de escala do sensor é linearmente crescente com o tempo.

As falhas simuladas nas rodas de reação, conforme o desenvolvimento de Siqueira (2014) são:

- **Saturação do comando de potência:** é reconhecida como “T1”. Consiste de, a despeito do sinal de controle dado pela saída do controlador, apresentar sinal de comando (i.e., potência) saturado em um dos extremos de tensão, +10V ou -10V.

- **Falência da roda de reação:** é reconhecida como “T2”. Como explicado em LEITE (2007), nesta falha a roda simplesmente não responde mais ao sinal de controle e passa a trabalhar erraticamente ao redor e proximalmente da velocidade nula.
- **Atrito excessivo nos mancais das rodas:** é reconhecida como “T3”. Não é mérito do presente trabalho – analogamente ao trabalho feito por Leite (2007) e estendido em Siqueira (2014) – tratar da causa desta falha, mas do efeito da mesma. Assim, o comportamento é que a roda tenha considerável perda de efetividade e que a saída do controlador sature rapidamente, assim como a velocidade da roda de reação (esta, porém, para um valor inversamente proporcional ao coeficiente de atrito global aumentado).
- **Disparo de comando em regime permanente:** é reconhecida como “T4”. Esta falha é considerada mais provável de ocorrer em condição de regime permanente. Neste caso de falha, a roda de reação passa a atuar por comando de sinais não-determinísticos próximos dos limites de saturação (-10V e +10V).

Cada falha (F1 a F4 e T1 a T4), a cada instante de tempo, foi simulada 2 vezes para que fosse garantida a consistência do modelo de falha sobre a dinâmica do satélite. Assim, o total de simulações falhadas é de **4** modelos de falhas * **4** tempos de injeção * **2** simulações de casos = **32** casos de falha de sensores e de atuadores, tendo um total de **64**.

As figuras mostrando o comportamento normal da PMM e a propagação das falhas injetadas estão no APÊNDICE E - .

Finalmente, o comportamento de modo normal consiste de 20 simulações isentas de quaisquer anomalias, realizando a manobra de *de-tumble* com o controlador R-LQR.

No total tem-se 84 casos de simulações: 32 de falhas de sensores, 32 de falhas de atuadores e 20 simulações de manobra sem falha.

5.2.4. Aplicação do algoritmo BAbyLO-BR

Lembrando que o algoritmo BAbyLO-BR trabalha unicamente com dados estáticos e vendo que este estudo de caso é dinâmico, trataram-se os dados como se segue.

Conforme explicado no começo desta seção, para o modelo de simulação utilizado e até o tempo de 180 segundos sabe-se que a PMM manobra (acelera-desacelera-estabiliza), e, a partir disso, entra em manutenção de órbita e apontamento. Fazendo-se uma comprovação empírica com base nas 20 simulações de modo normal, percebeu-se que seria conveniente realizar uma coleta dos valores entre 180-500 segundos e calcular a média nesse tempo para cada um dos parâmetros de saída da simulação.

Os identificadores das amostras são como se seguem:

- **N_x**: Indica amostra normal sendo 'x' um número único;
- **T_{x_y_z}**: Indica uma falha de atuador (T) sendo 'Tx' o identificador da falha (T1 até T4), 'y' o instante de tempo em que aconteceu a falha (15, 30, 50 e 250) e 'z' o número identificador da amostra (1 ou 2, pois têm-se duas simulações para cada falha);
- **F_{x_y_z}**: Indica uma falha de sensor (F) sendo 'Fx' o identificador da falha (F1 até F4), 'y' o instante de tempo em que aconteceu a falha e 'z' o número identificador da amostra.

Conforme a explicação anterior, um exemplo de arquivo de dados de entrada é apresentado na Figura 5-15:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1		phi_estre (rad)	theta_estre (rad)	psi_estre (rad)	phidot_giro (rad)	thetadot_giro (rad)	psidot_giro (rad)	x_saida (V)	y_saida (V)	z_saida (V)	Vel_roda_X (rpm)	Vel_roda_Y (rpm)	Vel_roda_Z (rpm)
2	T1_10_1	50.5382	0.73361	0.13033	2.3056	0.091537	0.010509	561.7031	45.192	45.6448	45261.1	2726.283	2781.164
3	T1_10_2	50.5382	0.73353	0.13023	2.3056	0.091532	0.010494	561.7038	45.1795	45.6278	45261.1	2724.153	2778.953
4	T1_30_1	50.9123	0.7292	0.066624	2.3071	0.091216	0.007948	561.7165	34.7097	35.1515	45260.7	2164.856	2083.665
5	T1_30_2	50.9121	0.72929	0.066639	2.3071	0.091241	0.007938	561.7151	34.7391	35.184	45260.7	2166.311	2084.787
6	T1_50_1	52.6513	0.73841	0.066156	2.3072	0.091502	0.00807	561.7116	36.197	34.3631	45257.59	2133.593	2164.484
7	T1_50_2	52.6513	0.73844	0.066179	2.3072	0.091515	0.008072	561.7117	36.1716	34.359	45257.59	2133.305	2164.039
8	T1_250_1	44.3354	0.7167	0.095602	1.9119	0.092151	0.007329	494.5335	37.8851	35.0762	37519.66	2438.067	2234.017
9	T1_250_2	44.3353	0.71675	0.095648	1.912	0.092154	0.007337	494.5352	37.8974	35.0696	37519.68	2438.165	2233.942
10	N_1	0.000463	0.72306	0.000468	0	0.09109	0	25.0287	92.4313	88.591	1994.738	7386.97	7094.863
11	N_2	0.000469	0.72306	0.000469	0	0.09109	0	25.0286	92.431	88.5909	1994.726	7386.973	7094.875
12	N_3	0.000465	0.72306	0.000468	0	0.09109	0	25.0288	92.4321	88.5918	1994.733	7386.976	7094.876
13	N_4	0.000468	0.72306	0.000468	0	0.09109	0	25.0282	92.4309	88.5914	1994.733	7386.976	7094.862
14	N_5	0.000468	0.72306	0.000464	0	0.09109	0	25.0283	92.4314	88.5912	1994.725	7386.97	7094.857
15	N_6	0.000468	0.72306	0.000469	0	0.09109	0	25.0276	92.4306	88.5914	1994.723	7386.97	7094.857
16	N_7	0.000462	0.72306	0.000468	0	0.09109	0	25.0283	92.4319	88.5923	1994.735	7386.972	7094.869
17	N_8	0.000464	0.72306	0.000466	0	0.09109	0	25.0284	92.4309	88.5912	1994.731	7386.974	7094.884

Figura 5-15. Exemplo de dados de entrada no estudo de caso 'Identificação de falhas de sensores e atuadores da Plataforma Multi-Missão'.

O algoritmo foi aplicado nos seguintes conjuntos de amostras:

- Caso 1) Falha F1 + comportamento normal
- Caso 2) Falha F2 + comportamento normal
- Caso 3) Falha F3 + comportamento normal
- Caso 4) Falha F4 + comportamento normal
- Caso 5) Falhas F1 + F2 + F3 + F4 + comportamento normal
- Caso 6) Falha T1 + comportamento normal
- Caso 7) Falha T2 + comportamento normal
- Caso 8) Falha T3 + comportamento normal
- Caso 9) Falha T4 + comportamento normal
- Caso 10) Falhas T1 + T2 + T3 + T4 + comportamento normal

Devido ao grande número de casos estudados, os resultados são mostrados em forma de tabela definindo para cada simulação os seguintes parâmetros de validação:

- 1) Falhas e não falhas foram diferenciadas em *clusters* diferentes
- 2) Número de falhas associadas a comportamento normal (Falsos Negativos, FN)
- 3) Número de comportamentos normais associados a *cluster* de falhas (Falsos Positivos, FP).
- 4) Falhas em diferentes instantes de tempo foram diferenciadas

- 5) O algoritmo permitiu calcular a assinatura do comportamento normal
- 6) O algoritmo permitiu calcular a assinatura da falha
- 7) Observações

Tabela 5-6. Tabela resumo de resultados de aplicação do algoritmo no caso Identificação de falhas em sensores e atuadores na PMM.

Caso de estudo	Falha VS não falha	FN	FP	Falhas temporais.	Assinatura comportamento normal	Assinatura comportamento falhado	Observações
1	SIM	0	0	Parcialmente	SIM	Parcialmente	As falhas em 50s e 250s não são diferenciadas.
2	Parcialmente	1	0	Parcialmente	Parcialmente	Não	Nota(1)
3	SIM	0	0	SIM	SIM	SIM	Falha em t=30s e t=50s classificadas no mesmo agrupamento.
4	SIM	0	0	SIM	SIM	SIM	Falha em t=30s e t=50s classificadas no mesmo agrupamento.
5	SIM	0	0	Parcialmente	SIM	Parcialmente	Nota(3)
6	SIM	0	0	SIM	SIM	SIM	N/A
7	SIM	0	0	Parcialmente	SIM	Parcialmente	Nota (2)
8	SIM	0	0	NÃO	SIM	NÃO	Falhas temporais agrupadas conforme aconteceram temporalmente.
9	SIM	0	0	NÃO	SIM	NÃO	Falhas iguais classificadas em diferentes agrupamentos.
10	SIM	0	0	SIM parcialmente	SIM	SIM parcialmente	Nota (4)

Nota(1): As falhas acontecidas em diferentes instantes de tempo não são diferenciadas. Existe um FN, Falha acontecida em t=250s.

Nota (2): Falha temporal em t=15s e t=30s misturadas. Falha temporal em t=50s e t=250s misturadas.

Nota(3): As falhas são classificadas em agrupamentos que misturam diferentes tipos de falhas e instantes mas nunca separam falhas iguais.

Nota (4): As falhas T2, T3, T4 acontecidas em 10s ou 30s são classificadas no mesmo agrupamento. Igualmente as falhas T2, T3 e T4 acontecidas em t=50s e t=250s são classificadas o mesmo agrupamento. A falha T1 é corretamente separada em 4 agrupamentos conforme os tempos em que aconteceram.

Estudando a Tabela 5-6 podem-se deduzir as seguintes afirmações:

- 1) Em todo momento o algoritmo é capaz de diferenciar o caso normal do caso falhado (conforme objetivo 1) apresentado na Seção 5.2.1).
- 2) É possível criar uma assinatura de comportamento normal sem necessidade de ter um conhecimento prévio do sistema (modelo) ou sem utilizar um algoritmo de aprendizado prévio. A assinatura é criada com um algoritmo genérico e de forma rápida, o que economiza tempo.
- 3) O algoritmo não é sempre capaz de diferenciar entre falhas acontecidas em instantes de tempo diferentes (a mesma falha acontecida em instantes diferentes é classificada no mesmo agrupamento). Isto se pode dever à lenta propagação da falha simulada (acontece em um instante, mas demora em se propagar e causar um efeito na saída do sistema). Lembre-se que a Plataforma Multi-Missão tem uma dinâmica lenta.
- 4) O algoritmo, aplicado com as heurísticas padrão, não é sempre capaz de diferenciar entre tipos de falha, não podendo criar uma assinatura de cada tipo de falha para identificação e reconfiguração da PMM (conforme o objetivo 2 apresentado na Seção 5.2.1). Estudando os valores dos parâmetros das amostras falhadas, observa-se que são muito similares, o que quer dizer que as falhas não se manifestam de forma diferente na fase de manutenção e apontamento (entre 180 e 500s), pelo que é difícil achar uma assinatura utilizando essa fase.
- 5) Precisa-se realizar um estudo detalhado da dinâmica e propagação de falhas no sistema para entender até que ponto as falhas que acontecem em instantes diferentes provocam perturbações diferentes nos parâmetros de saída. Lembrando que o algoritmo é diretamente utilizável para sistemas com comportamento estático; os sistemas com comportamento dinâmico devem ser tratados previamente para escolher a estática que será estudada. Este processo não faz parte do processo de implementação do algoritmo e deve ser feito pelo analista de dados, usuário do algoritmo.

5.2.5. Utilidade de aplicação do algoritmo

A aplicação do algoritmo neste estudo de caso permitiu validar que as heurísticas são suficientemente genéricas para aplicar o algoritmo a sistemas diferentes.

Ficou demonstrado que o algoritmo, neste estudo de caso, também conseguiu diferenciar o comportamento normal dos comportamentos falhados, independentemente do tipo de falha acontecido, tendo um único falso negativo no conjunto total de todos os casos estudados. Os resultados do algoritmo vão permitir criar uma assinatura do comportamento normal e aplicar esta assinatura no trabalho de doutorado em desenvolvimento (SIQUEIRA, 2014) para diagnosticar o acontecimento de falhas no sistema em tempo real, identificar onde aconteceu a falha, isolá-la e reconfigurar o controlador para terminar a manobra com sucesso em tempo real no sistema PMM. Esta atividade está atualmente em desenvolvimento.

5.3. Identificação de corpos d'água em uma imagem Landsat

A separação entre solo e água é fundamental para o bom uso de imagens de Sensoriamento Remoto em Limnologia. Os especialistas de ambientes aquáticos incentivam o desenvolvimento de novas técnicas que aperfeiçoem o trabalho de identificação de corpos d'água em imagens digitais. Por isso, um integrante do grupo de estudos de ambientes aquáticos da DSR do INPE cedeu a imagem da Represa de Furnas em Minas Gerais, com área inundada de 1440 km². A Figura 5-16 mostra a imagem cedida; esta imagem corresponde ao satélite Landsat 8, órbita 222 ponto 76, data de aquisição da imagem 08 de fevereiro de 2014, resolução espacial de 30 metros, Sistema de Projeção UTM, Zona 23 sul e datum WGS 84.

No estudo de ambientes aquáticos, especificamente na identificação de corpos d'água, torna-se importante que o algoritmo identifique a faixa de níveis de cinza que representam a água na imagem. Esperava-se encontrar dois comportamentos, correspondentes à superfície terrestre e água.

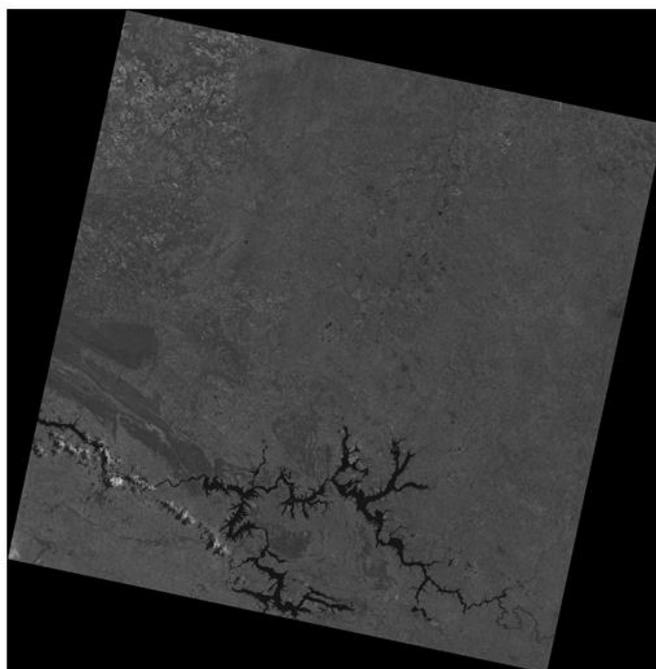


Figura 5-16 Imagem recebida do satélite Landsat para identificação de corpos d'água.

Atualmente, na área de Sensoriamento Remoto, o tratamento destes dados se realiza utilizando um algoritmo de classificação que utiliza técnicas fuzzy orientadas a objetos. Porém, a capacidade de inserção do conhecimento possibilitada por este algoritmo tende a tornar o processo mais complexo, exigindo uma grande especialização e participação mais decisiva por parte do analista (PINHO et al., 2005).

5.3.1. **Objetivos de identificação do algoritmo**

O objetivo deste estudo de caso é aplicar o algoritmo BAbyLO-BR na imagem digital utilizada para identificação de corpos d'água para comprovar se ele é capaz de:

- 1) Criar dois comportamentos diferenciando entre superfície terrestre e água: Se o algoritmo for bem sucedido, poder-se-á identificar um padrão de comportamento para água e outro para superfície terrestre e, talvez, reaproveitar esse padrão para classificar outras imagens.
- 2) Testar a generalidade de aplicação do algoritmo aplicando-o em um estudo de caso totalmente diferente aos estudos de caso da Área de Controle.

5.3.2. Descrição dos dados de entrada

Foi necessário converter a imagem digital (7761×7621 pixels) em um arquivo ASCII, em três colunas, contendo a latitude, a longitude e o valor do nível de cinza (0-65535) de cada pixel da imagem. Nota-se que neste estudo de caso trabalha-se com dados provenientes de um sistema estático.

A imagem recebida do satélite foi reduzida por questão de memória, aplicando uma redução de 0.1 e ficando um tamanho de 777×763 pixels. Os dados de entrada foram os níveis de cinza da imagem tratada (sistema unidimensional) sem levar em conta longitude e latitude.

5.3.3. Aplicação do algoritmo

O algoritmo classificou os dados de entrada em três *clusters* segundo a Tabela 5-7 criada a partir do arquivo “clustersFormados.txt”.

Tabela 5-7. Classificação das amostras segundo o nível de cinzas.

ID clusters	Namostras	cinzas Vmin	cinzas Vmax
1	11114	3	10907.2
2	407632	10907.2	45800.64
3	27	45800.64	54524

Era esperado encontrar unicamente dois comportamentos correspondentes a água e superfície terrestre, mas o algoritmo realizou a separação das amostras reconhecendo três comportamentos. Depois do estudo dos dados, percebeu-se que o terceiro *cluster* formado corresponde às nuvens da imagem, que não tinham sido levadas em conta na análise prévia à aplicação do algoritmo, mas que foram captadas (em parte) pelo algoritmo.

Para realizar a validação dos resultados obtidos, foi reproduzida a imagem do LANDSAT associando a cada pixel uma cor correspondente ao *cluster* onde tinha sido

classificado. A Figura 5-17a mostra a imagem reproduzida segundo esses três níveis de cinza, a saber; branco (nuvens), cinza (superfície terrestre) e preto (água) e a Figura 5-17b mostra a classificação produzida pelo analista utilizando o algoritmo de classificação específico para esse tratamento de dados e um processo manual, onde são reconhecidos quatro comportamentos (água em azul, nuvens em preto, superfície terrestre em cinza e desconhecido em branco).

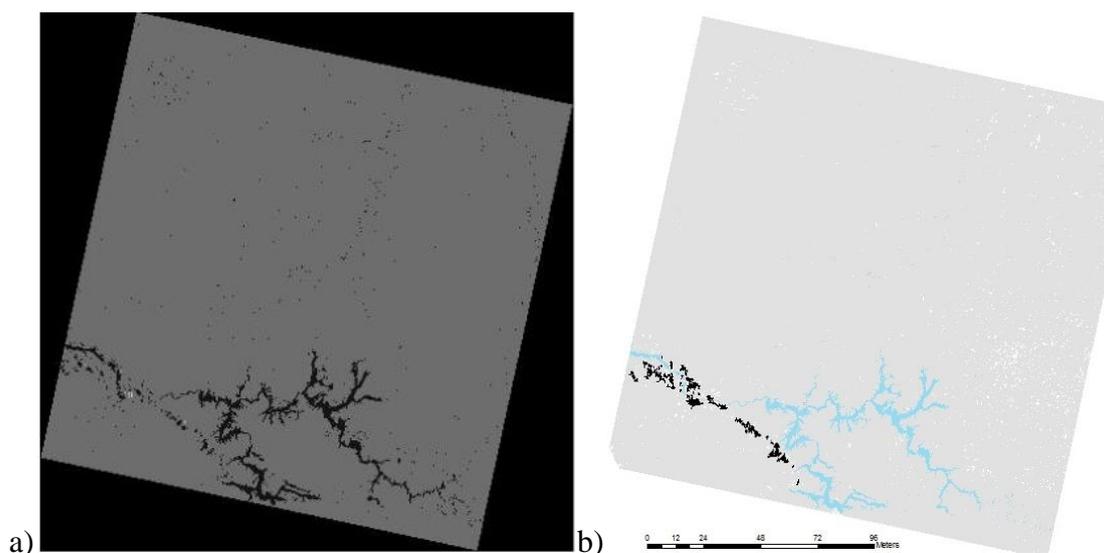


Figura 5-17. Resultados de identificação de corpos d'água; a) Produzidos pelo algoritmo BAbyLO-BR e b) produzidos pelo analista da área usando classificador e processo manual.

Pode-se observar que existe um conjunto de pixels correspondentes às sombras das nuvens que o algoritmo BAbyLO-BR classificou como água (falsos positivos). Também pode-se observar que existem alguns pixels classificados como não definido pelos analistas que o algoritmo BAbyLO-BR classifica como superfície terrestre. Isto é devido a que a câmera do satélite associa níveis de cinza iguais a elementos diferentes.

Estudando os resultados obtidos podem-se deduzir as seguintes afirmações:

- 1) O algoritmo conseguiu criar os padrões de comportamento esperados, conforme o objetivo 1) deste estudo de caso, achando ainda um comportamento que não tinha sido previsto pelos analistas.

- 2) O algoritmo BAbyLO-BR não é ótimo; encontram-se diferenças entre a classificação feita por ele e a dos analistas. Porém, estas diferenças podem ser resolvidas realizando um pós-tratamento dos *clusters* por parte do analista que leve em conta as latitudes e longitudes e que classifique manualmente esses pixels duvidosos. Isto leva um trabalho manual por parte do analista, mas focado unicamente nesses pixels duvidosos e não na imagem completa, conforme o algoritmo de classificação usado na área.
- 3) O tempo utilizado para aplicar o algoritmo e identificar os três *clusters* foi de 20 minutos. 10 minutos dedicados a formatar os dados segundo o formato aceito pelo algoritmo, e 10 minutos para aplicar o algoritmo, obter os *clusters* e reproduzir a figura classificada. Esses tempos são mínimos quando comparados ao tempo de processamento do algoritmo de classificação dos analistas.
- 4) O algoritmo poupa tempo de processamento e dedicação de recursos humanos.
- 5) Para finalizar, os resultados obtidos neste estudo de caso podem ser aplicados a outras imagens (conforme o objetivo 2 deste estudo de caso) sendo que, se os limites não funcionarem, não é necessário que o pesquisador realize um estudo para ajustá-los manualmente pois, como é rápido, o algoritmo pode ser aplicado novamente sobre a nova imagem.

5.3.4. Utilidade da aplicação do algoritmo

A aplicação do algoritmo neste estudo de caso e a consequente classificação exitosa dos dados permite demonstrar a generalidade do algoritmo e a abrangência de aplicação em outras áreas de interesse do INPE.

5.4. Comparação com outro algoritmo da área

Validar um algoritmo de aplicação genérica resulta difícil e pode ser feita conforme neste trabalho mediante aplicação em diversos estudos de caso. Além disso, a validação pode ser reforçada comparando os resultados obtidos em um estudo de caso com outro algoritmo da área que já esteja desenvolvido e publicado.

Em Arifin (1974) desenvolveu-se um algoritmo específico para classificação de imagens em branco e preto, usando a técnica de clusterização particional baseada em histogramas; a mesma técnica que a utilizada neste trabalho. Em Arifin (1974), à diferença do BAbyLO-BR, é necessário que o usuário do algoritmo defina um limiar para separar o histograma em clusters; ele não é genérico.

A Figura 5-18 mostra um exemplo de uma imagem classificada pelo algoritmo desenvolvido em Arifin (1974). A figura da esquerda mostra a imagem real, no centro o histograma em nível de cinzas e a direita a classificação realizada em Arifin (1974).

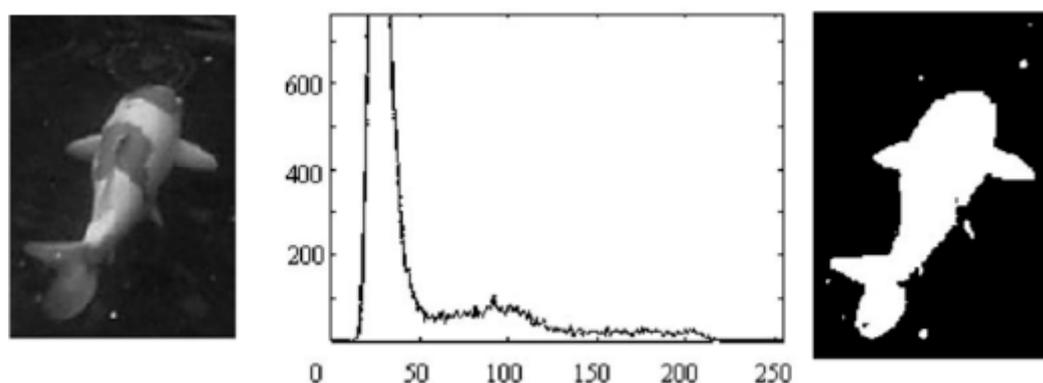


Figura 5-18. Imagem exemplo de Arifin (1974); Imagem real, histograma da imagem e imagem classificada em Arifin (1974).

5.4.1. Objetivos de identificação do algoritmo

O objetivo deste estudo de caso é aplicar o algoritmo BAbyLO-BR na imagem do peixe digitalizada da Figura 5-18 e comprovar se ele é capaz de:

- 1) Reconhecer a formas conforme Arifin (1974), separando em dois comportamentos o fundo preto e o peixe.
- 2) Reconhecer ainda mais detalhes do peixe, entre eles as faixas mais escuras do corpo dele, demonstrando maior acurácia que o algoritmo de Arifin (1974).
- 3) Demonstrar que o algoritmo BAbyLO-BR, de aplicação genérica, consegue separar mais comportamentos do que o algoritmo de aplicação específica desenvolvido em Arifin (1974).

5.4.2. Descrição dos dados de entrada

A imagem da Figura 5-18 foi recuperada do artigo de Arifin (1974) em formato 'pdf'. Pode-se observar que se trata de uma fotografia em branco e preto de um peixe branco sobre fundo preto com franjas cinza.

As informações digitais da imagem não são disponibilizadas em Arifin (1974); unicamente tem-se a própria imagem junto com o histograma da mesma e os resultados obtidos da classificação feita pelo algoritmo desenvolvido nesse artigo. Com isso, para obter as informações digitais da imagem (dados de entrada do algoritmo BAbyLO-BR) utilizou-se a função *imread* da ferramenta Matlab que lê uma imagem e devolve o nível de cinza (0 – 256) de cada pixel que a forma.

Com esta técnica criou-se a matriz de dados de entrada, formada por 92919 filas (número de pixels) e 1 coluna (nível de cinza de cada pixel). Nota-se que neste estudo de caso trabalha-se com dados provenientes de um sistema estático.

5.4.3. Aplicação do algoritmo

O algoritmo foi aplicado sobre a matriz de dados de entrada e a classificação obtida foi segundo mostrado na Tabela 5-8:

Tabela 5-8. Classificação das amostras segundo o nível de cinzas.

ID clusters	Namostras	cinzas Vmin	cinzas Vmax
1	71997	5.5	62.9545
2	16218	61.9545	151.3182
3	3288	150.3182	190.5909
4	1416	189.5909	222.5

Os pixels foram separados em quatro comportamentos diferentes; para visualizar estes resultados em forma de imagem, desenhou-se a imagem em quatro níveis de cinza

representando cada um dos quatro *clusters* formados. A Figura 5-19 mostra a imagem real e as duas classificações realizadas pelos algoritmos BAbyLO-BR e o implementado em Arifin (1974).

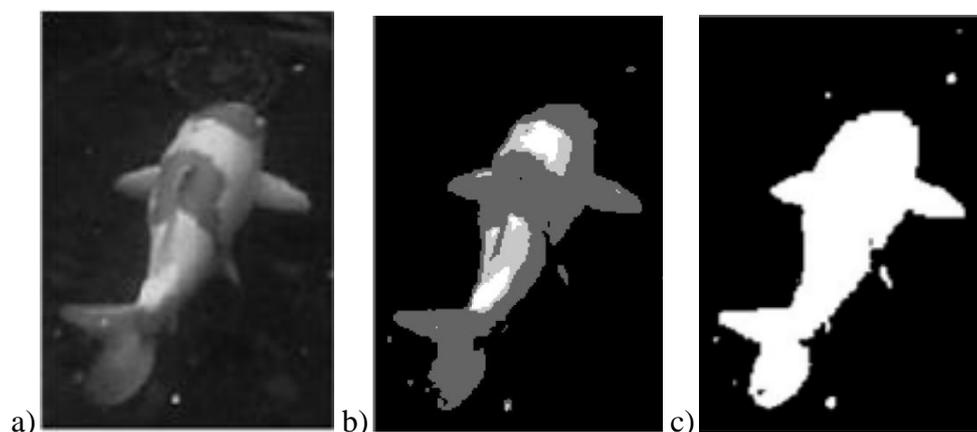


Figura 5-19. Resultados de classificação da imagem do peixe listrado; a) Imagem de entrada do algoritmo, b) Classificação da imagem usando o algoritmo BAbyLO-BR e c) Classificação da imagem realizada em Arifin (1974).

Pode-se observar que os três objetivos desejados foram atingidos; BAbyLO-BR conseguiu, não só separar a forma do peixe do fundo preto conforme em Arifin (1974) mas também diferenciar as franjas mais escuras do peixe. Além disso, existem dois tons de branco no corpo do peixe que o algoritmo reconhece e separa em dois comportamentos diferentes.

Pode-se observar também que a aleta direita do peixe é classificada junto com as franjas de cor cinza escuro, isso se deve à coloração mais escura devido às sombras da imagem.

5.4.4. Utilidade da aplicação do algoritmo

Este estudo de caso permite validar o algoritmo mediante a comparação com a classificação feita por outro algoritmo existente na área. O algoritmo BAbyLO-BR separou os comportamentos conforme esperado, mostrando a generalidade de utilização do mesmo.

5.5. Conclusões

A Tabela 5-9 faz uma recompilação dos quatro estudos de caso abordados, as características de cada um deles, e as simulações realizadas em cada caso. Esta tabela pretende mostrar a abrangência procurada na hora de escolher os estudos de caso, para garantir a maior cobertura possível de casos estudados. Esta cobertura pretende mostrar a generalidade do algoritmo utilizando a seguinte dedução – *Se o algoritmo funciona em um conjunto de estudos de caso variado, cabe esperar que ele terá um comportamento parecido e, portanto, funcionará, em outros estudos de caso onde seja aplicado.* Porém, este trabalho não garante o funcionamento do algoritmo de forma universal.

Tabela 5-9. Classificação de estudos de caso.

ESTUDOS DE CASO	CARACTERÍSTICAS	SIMULAÇÕES
<p>Caso 1) Tubulações hidráulicas.</p>	<ul style="list-style-type: none"> -Sistema dinâmico estudado de forma estática (amostrada). -Número de amostras $n_a = 250$ Número de parâmetros $n_p = 3$ -$p_j \in \mathbb{R}$ -Dados de entrada $A^{250 \times 3}$ -Resultados conhecidos. -Não existe controlador no sistema. -Agrupamentos esperados: normal e falhado(s). 	<ul style="list-style-type: none"> -Simulação de falhas no sistema em diferentes instantes de tempo. -Simulação de falhas nos sensores em diferentes instantes de tempo. -Simulação sem falha (para comprovar que o algoritmo não separa o comportamento normal em dois ou mais <i>clusters</i>).
<p>Caso 2) Falhas na PMM.</p>	<ul style="list-style-type: none"> -Sistema dinâmico estudado de forma estática (amostrada). -Número de amostras $n_a = 24 - 36$ -Número de parâmetros $n_p = 12$ -$p_j \in \mathbb{R}$ -Dados de entrada $A^{36 \times 12}$ -Existem parâmetros inclusos no estudo que podem não estar perturbados pela falha. -A variação máxima entre parâmetros diferentes é muito divergente. Permite testar a capacidade do algoritmo de adaptar o número de janelas do histograma ao range de p_j. -Resultados conhecidos. -Existe controlador. 	<ul style="list-style-type: none"> - Simulação de falhas nos sensores em diferentes instantes de tempo. - Simulação de falhas nos atuadores em diferentes instantes de tempo.

Continua

Tabela 5.9 - Conclusão

ESTUDOS DE CASO	CARACTERÍSTICAS	SIMULAÇÕES
	- Agrupamentos esperados: Normal e falhado(s).	
Caso 3) Identificação de corpos d'água.	<ul style="list-style-type: none"> - Sistema estático. - Número de amostras $n_a = 1048569$ - Número de parâmetros $n_p = 1$ - Dados de entrada $A^{1048569 \times 1}$ - $p_j \in \mathbb{N}$ - Agrupamentos esperados: Normais com características diferentes entre eles. - Abrange uma nova área de conhecimento: algoritmo testado em uma aplicação totalmente diferente. 	-Uma única aplicação (não existem falhas pra simular).
Caso 4) Comparação com outro algoritmo da área.	<ul style="list-style-type: none"> - Sistema estático - Número de amostras $n_a = 92919$ - Número de parâmetros $n_p = 1$ - Dados de entrada $A^{92919 \times 1}$ - $p_j \in \mathbb{N}$ - Agrupamentos esperados: Normais com características diferentes entre eles. - Permite realizar uma validação do algoritmo mediante a comparação com outro algoritmo que utiliza a mesma técnica. 	-Uma única aplicação (não existem falhas pra simular).

O caso 1) é um estudo de caso simples, sem controlador e tridimensional, que pretende validar o algoritmo. Este estudo de caso serviu para comprovar se o algoritmo consegue diferenciar entre os casos normal e falhados e se ele não separa comportamentos iguais (dois agrupamentos diferentes para o mesmo comportamento normal). Este caso também foi utilizado para aprovar as heurísticas escolhidas durante o desenvolvimento. Neste caso foram simuladas falhas aleatórias (com porcentagem de falha entre 0 e 1) para comprovar também a robustez do algoritmo em função da diferença existente entre os valores falhado e não falhado.

O caso 2) foi utilizado para comprovar se as heurísticas são o suficientemente abrangentes (as amostras têm maior número de parâmetros e os valores dos parâmetros são mais díspares), se o algoritmo é o suficientemente robusto para ser aplicado em sistemas multidimensionais e para comprovar se os agrupados criados pelo sistema podem ser utilizados como padrões de comportamento (em várias simulações com dados de entrada diferentes, o algoritmo cria *clusters* similares para comportamentos idênticos).

O caso 3) foi utilizado para comprovar até que ponto o algoritmo é aplicável a análises de agrupamentos que não estejam relacionados com falhas de sistemas. Neste caso pretende-se diferenciar entre comportamentos diferentes (água e não água) sem ser nenhum dos dois um caso falhado. Este estudo de caso é unidimensional, pelo que também é utilizado para comprovar se o algoritmo é funcional tendo um único parâmetro a ser avaliado. Este caso também permitiu comprovar a robustez do algoritmo na hora de encontrar comportamentos não conhecidos pelo analista, graças à independência do conhecimento dos dados de entrada (algoritmo não supervisionado ou sem aprendizado prévio).

O caso 4) foi utilizado para realizar a validação do algoritmo aplicando-o em uma imagem que já tenha sido classificada por outro algoritmo da área (ARIFIN, 1974) e que utiliza a mesma técnica; clusterização particional baseada em histogramas. Neste caso pretende-se comprovar se o BAbyLO-BR, de aplicação genérica, consegue

identificar, no mínimo, os mesmos comportamentos que em Arifin (1974), de aplicação específica em imagens branco e preto. Arifin não consegue diferenciar entre níveis de cinza, fazendo exclusivamente diferença entre branco e preto nos resultados; já o algoritmo BAbyLO-BR consegue também criar vários agrupamentos em função dos níveis de cinza.

Resumindo, o algoritmo respondeu positivamente a todos os estudos de caso em que foi aplicado, sendo a heurística “máscara de suavização” a que mais afeta aos resultados obtidos. Neste trabalho, com o mesmo algoritmo, conseguiu-se classificar com êxito os quatro estudos de caso, validando assim, além da aplicabilidade e a funcionalidade, a generalidade do algoritmo.

6 AVALIAÇÃO DO IMPACTO DE UTILIZAÇÃO DO ALGORITMO NO PROCESSO DE DETECÇÃO DE ANOMALIAS

Para realizar esta avaliação, foi utilizada a experiência adquirida durante a aplicação do algoritmo nos estudos de caso, as utilidades do algoritmo levantadas para cada estudo de caso, incluindo aqueles apresentados no APÊNDICE C - nos quais não foi possível realizar a validação do algoritmo, e as avaliações dos especialistas.

A Tabela 6-1 mostra uma relação das vantagens e desvantagens da utilização do algoritmo:

Tabela 6-1. Vantagens e desvantagens da utilização do algoritmo.

Vantagens	Desvantagens
Resultados obtidos em segundos ou minutos de processamento.	A implementação em <i>Matlab</i> reduz a quantidade de dados de entrada e aumenta o tempo de execução do algoritmo devido à quantidade total de memória disponível para execução do SW.
Possibilidade de realizar múltiplos estudos com diferentes parâmetros e dados até obter os resultados mais interessantes.	Um conjunto de dados de entrada muito grande pode saturar o algoritmo produzindo a obtenção de um único <i>cluster</i> de comportamento.
Capacidade de identificar, num mesmo processamento, erros, comportamentos e limiares de decisão.	Não identifica limiares quando estes não dependem do comportamento dos dados e sim da necessidade do analista de definir um valor limite.
Descobrimto de <i>clusters</i> não esperados segundo o conhecimento prévio dos dados.	Pode acontecer a criação de agrupamentos onde não existe uma classificação natural dos dados.
Aplicação em conjuntos de dados unidimensional ou multidimensional.	Identificação de vários comportamentos diferentes onde existe um único comportamento.
Possibilidade de estudar os dados sem saber os resultados esperados (não é necessário conhecimento a priori dos resultados esperados).	Existência de falsos positivos e falsos negativos que devem ser estudados especificamente pelo analista.

Continua

Tabela 5.9 - Conclusão

Vantagens	Desvantagens
O algoritmo permite estudar sistemas tanto estáticos quanto dinâmicos.	Existe a necessidade de realizar um pré-procesamento de dados de sistemas dinâmicos.
O algoritmo permite realizar o ajuste manual das heurísticas pré-configuradas, caso seja necessário para conseguir uma clusterização.	O algoritmo depende de heurísticas pré-configuradas, especialmente da máscara utilizada para realizar a suavização do histograma.
Sem reajuste manual de limiares, o algoritmo pode ser aplicado rapidamente a cada conjunto específico de dados.	O algoritmo não é útil no reconhecimento de falhas temporais do tipo intermitente, pois ele é de aplicação estática.
O algoritmo BAbyLO-BR, de aplicação genérica, pode substituir algoritmos de aplicação específica, oferecendo inclusive melhores resultados.	O algoritmo é limitado diante de sistemas com ruídos e incertezas, quando o tamanho destes seja comparável à diferença entre comportamentos (confundir ruído com comportamento).

7 CONCLUSÕES E TRABALHOS FUTUROS

7.1. Conclusões

Para realizar o desenvolvimento do algoritmo de análise de agrupamentos BAbyLO-BR, foram aplicadas técnicas de clusterização, matemáticas, estatísticas e de filtragem.

O algoritmo desenvolvido neste trabalho utiliza a técnica de clusterização por histogramas, mas difere em dois aspectos dos usos até agora feitos, 1) O algoritmo é aplicável a vários estudos de caso diferentes e de uma ou várias dimensões, 2) O algoritmo não depende do conhecimento prévio do sistema por parte do usuário.

O algoritmo BAbyLO-BR foi validado em quatro estudos de caso. O algoritmo foi aplicado também a mais estudos de caso, onde os resultados esperados são desconhecidos a priori, mas onde as classificações obtidas com o algoritmo são coerentes com os resultados esperados pelos analistas (ver APÊNDICE C -).

O algoritmo de análise de agrupados implementado provê boas técnicas para conseguir separar as amostras segundo os comportamentos naturais delas.

O algoritmo contém heurísticas padrão que permitem a aplicação sem conhecimento prévio do sistema. Porém, a heurística que mais afeta aos resultados é a máscara aplicada para suavização do algoritmo (aplicada em dois dos casos utilizados para validação). Por esse motivo, o algoritmo oferece a possibilidade de ajustar a máscara manualmente. Nota-se que isto está de acordo com a afirmação feita por Dubes (1988) (ver Seção 2.5.1) que constata o mau costume dos algoritmos de clusterização de criar dois *clusters* onde só deveria existir um. Isto é um problema menor, pois a análise posterior do usuário pode verificar esta separação e agrupar (entende-se que separar em dois *clusters* um comportamento causa menor retrabalho do que misturar dois comportamentos em um *cluster*). Poder-se-ia falar que o ajuste desta máscara é o “calcanhar de Aquiles” deste algoritmo.

O algoritmo BAbyLO-BR não depende do conhecimento prévio do sistema (método baseado em dados e não em modelos), o que permite realizar um estudo dos

comportamentos do sistema sem conhecê-lo e sem ter que realizar uma modelagem acurada do mesmo.

O algoritmo não consegue diferenciar entre *clusters* que representam anomalias e *clusters* que representam comportamento normal devido à técnica utilizada para sua implementação (métodos não supervisionados).

O algoritmo pode ser utilizado para criação de padrões de comportamento de forma rápida e sem precisar do conhecimento prévio do sistema ou de fazer um treinamento do algoritmo com dados conhecidos.

O algoritmo tem a debilidade de não poder ser aplicado diretamente a sistemas dinâmicos, precisando realizar um pré-processamento dos dados aplicando estatística conjuntal para estudar o sistema dinâmico de forma estática.

Devido à independência dos dados de entrada, o algoritmo é aplicável a diferentes sistemas tanto da Área de Controle quanto de outras áreas de interesse do INPE (como Sensoriamento Remoto).

7.2. Trabalhos futuros

Propõem-se os seguintes trabalhos futuros:

- 1) Estudar a possibilidade de adaptação do algoritmo para aplicação em SW ou em Sistemas de Controle Baseados em Rede.
- 2) Melhorar o algoritmo mediante a implementação de uma máscara de suavização do histograma adaptativa que, automaticamente, escolha os pesos da máscara a aplicar em função das características do conjunto de amostras de entrada.
- 3) Melhorar o algoritmo mediante a implementação de uma máscara de suavização cujos valores sejam definidos em função nível de ruído aceitável. Note-se que, neste caso, se perde a generalidade do algoritmo, pois precisa-se conhecer o sistema sob estudo específico para implementar o filtro mas, aplicando esta técnica, é esperado ganhar confiabilidade nos *clusters* criados.

- 4) Estudar e validar a aplicação do algoritmo em sistemas com presença de variações nos parâmetros da planta.
- 5) Estender a validação à robustez do algoritmo em sistemas com comportamentos normais, anômalos e falhados (sistemas de dinâmica mais rápida, como lançadores e VANTS).
- 6) Aplicar o algoritmo à identificação de detritos espaciais.
- 7) Estudar e validar a aplicação do algoritmo em sistemas com falhas combinadas.
- 8) Estudar a sensibilidade do algoritmo em relação a modificações nas heurísticas. Para isto, propõe-se primeiro, aplicar variações individuais e controladas nas heurísticas utilizando um conjunto amostral de vários sistemas conhecidos. Depois, propõe-se aplicar variações combinadas nas heurísticas.

REFERÊNCIAS BIBLIOGRÁFICAS

- AGUIAR, D. A.; RUDORFF, B. F. T.; SILVA, W. F.; ADAMI, M.; MELLO, M. P. Remote sensing images in support of environmental protocol: monitoring the sugarcane harvest in São Paulo State, Brazil. **Remote Sensing**. v. 3, n. 12, p. 2682-2703, 2011. doi: <10.3390/rs3122682>.
- ANKERST, M. et al. Optics: Ordering points to identify the clustering structure. **ACM Sigmod Record**. v. 28, n. 2, p. 49-60, June, 1999.
- ARAGÃO, L. E. O. C.; MALHI, Y.; ROMAN-CUESTA, R. M.; SAATCHI, S.; ANDERSON, LIANA O.; SHIMABUKURO, Y. E. Spatial patterns and fire response of recent Amazonian droughts. **Geophysical Research Letters**, v. 34, n. 7, p. Art. No. L07701, Abr. 2007. doi: <10.1029/2006GL028946>..
- ARIFIN, A. Z.; ASANO, A. Image segmentation by histogram thresholding using hierarchical *cluster* analysis. **Pattern Recognition Letters**, v. 27, n. 13, p. 1515-1521, 2006.
- BARBOSA, R. I.; FEARNside, P. M. Incêndios na Amazônia brasileira: estimativa da emissão de gases do efeito estufa pela queima de diferentes ecossistemas de Roraima na passagem do evento “El Niño”(1997/98). **Acta Amazonica**, v. 29, n. 4, p. 513-534, 1999.
- BUDALAKOTI, S.; SRIVASTAVA, A. N.; AKELLA, R. Discovering atypical flights in sequences of discrete flight parameters. IEE AEROSPACE CONFERENCE,2006, Big Sky, MT, USA. **Proceedings...** Big Sky, MT, USA: IEEE, 2006. p. 1-8. DOI: 10.1109/AERO.2006.1656109.
- CASTRO, A. A. M.; PRADO, P. P. L.. Algoritmos para reconhecimento de padrões. **Revista Ciências Exatas**, v. 8, n. 2002, 2001.

CHANDOLA, V. et al. Anomaly detection: A survey. **ACM Computing Surveys** (CSUR), v. 41, n. 3, 15 p., 2009.

CHHIKARA, R. S.; REGISTER, D. T. A numerical classification method for partitioning of a large multidimensional mixed data set. **Technometrics**, v. 21, n. 4, p. 531-537, 1979.

COUTINHO, A. C. **Dinâmica das queimadas no Estado do Mato Grosso e suas relações com as atividades antrópicas e a economia local**. Tese (Doutorado em Ciência Ambiental) - Universidade de São Paulo (USP), São Paulo, SP, 2005.
<http://www.alice.cnptia.embrapa.br/handle/doc/17068>.

GAYARRE, L.; OLIVEIRA e SOUZA, M.; ROCHA, G. An overview of models, methods and tools for verification, validation and accreditation of real time critical software. **SAE International**. 2013-36-0530. 2013. (SAE Technical Paper 2013-36-0530, 2013). doi:10.4271/2013-36-0530.

GAYARRE, L.; ANDERSON, L.; ROCHA, G.; ARAGÃO, L. Novo algoritmo de classificação automática de dados multidimensionais para identificação de comportamentos, limiares de decisão e *outliers* com potencial utilização para dados de sensores remotos. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO, 17., 2015, João Pessoa, PB. **Anais...** São José dos Campos: INPE, 2015.

GOBATO, M. F. **Controles monovariáveis e multivariáveis aplicados a sistemas aeroespaciais fracamente ou fortemente acoplados**. 2006. 388 p. (INPE-14494-TDI/1175). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2006. Disponível em: <<http://urlib.net/sid.inpe.br/MTC-m13@80/2006/06.30.18.04>>. Acesso em: 06 abr. 2015.

GONZALEZ, Rafael C.; WOODS, Richard E. **Processamento de imagens digitais** São Paulo, SP: Blucher, 2000. 509 p. ISBN 85-212-0264-4.

HAIR, J. F. et al. **Análise multivariada de dados**. Porto Alegre: Bookman, 2007.

ISERMANN, R. **Fault-diagnosis systems: an introduction from fault detection to fault tolerance**. Berlin, Germany: Springer, 2006. 475 p. ISBN 978-3-540-24112-6

JAIN, A.K.; DUBES, R.C.; **Algorithms for clustering data**. New Jersey: Prentice-Hall, Inc., 1988. New Jersey. ISBN:0-13-022278-X .

LEAU, Y. B. et al. Software development life cycle AGILE vs traditional approaches. In: INTERNATIONAL CONFERENCE ON INFORMATION AND NETWORK TECHNOLOGY, 2012, Chennai. **Proceedings...** Chennai, 2012. p. 162-167.

LEITE, A. C. **Detecção e diagnóstico de falhas em sensores e atuadores da plataforma multi-missão**. 2007. 374 p. (INPE-15219-TDI/1313). Tese (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2007. Disponível em:<<http://urlib.net/sid.inpe.br/mtc-m17@80/2007/12.07.10.48>>. Acesso em: 24 mar. 2015.

LI, L. **Anomaly detection in airline operations using flight data recorder data**. 2013.147 p. Tese de Doutorado. Massachusetts Institute of Technology, Department of Aeronautics and Astronautics. Massachusetts, 2013.

LOPES, R. V. F.; SILVA, A. R.; SATURNO, M. E.; SANTOS, W. A.; PESSOTA, F. **Multi-Mission Platform Attitude Control and Data Handling (ACDH) Subsystem Specification**, São José dos Campos: INPE, 2001. Relatório Interno.

LUSTOSA, H. D.; SOUZA, M. L. O. Influences of data bus protocols on an aircraft elevator flight control subsystem. In: IEEE/AIAA DIGITAL AVIONICS SYSTEMS CONFERENCE - INTEGRATED MODULAR AVIONICS THE MODERN APPROACH, 27. (DASC)., 2008, St. Paul, Minnesota, EUA. **Proceedings...** IEEE/AIAA, 2008. v. 1, p. 76. CD-ROM.

MANELLI NETO, H. **Estudo de requisitos e especificações para a tolerância a falha simples do sistema de controle de atitude da plataforma multimissão**. 2011. 207 p. (sid.inpe.br/mtc-m19/2011/02.28.13.08-TDI). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos,

2011. Disponível em: <<http://urlib.net/8JMKD3MGP7W/3996485>>. Acesso em: 06 abr. 2015.

MOREIRA, M. L. **Projeto e simulação de um controle discreto para a plataforma multi-missão e sua migração para um sistema operacional de tempo real. 2006. 181 p. (INPE-14202-TDI/1103). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2006. Disponível em: <<http://urlib.net/sid.inpe.br/MTC-m13@80/2006/07.10.13.42>>. Acesso em: 08 jul. 2015.**

MORETTIN, L. G. **Estatística básica: probabilidade** 06. ed. São Paulo, SP: Makron, 1995. 185 p., 305259 cm.

MORETTIN, L. G. **Estatística básica: v.2 – inferência.** São Paulo, SP: Makron, 2000. v2 . 182 p., 305259 cm.

OGATA, K. **Modern control engineering** . Upper Saddle River, NJ: Prentice-Hall, 1997. 637p. ISBN: 0-13-227307-1.

OLIVEIRA, D. A. **Calibração e controle de bobinas de Helmholtz para aplicação espacial.** 2014. 124 p. (sid.inpe.br/mtc-m21b/2014/02.21.12.04-TDI). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2014. Disponível em: <<http://urlib.net/8JMKD3MGP5W34M/3FQ75JS>>. Acesso em: 24 nov. 2014.

PADHY, N. et al. The survey of data mining applications and feature scope. **International Journal of Computer Science, Engineering and Information Technology (IJCEIT)**, v. 2, n. 3, June 2012.

PANG-NING, T. et al. **Introduction to data mining.** Library of Congress, 2006. Cap. 8. ISBN-10: 0321321367 • ISBN-13: 9780321321367.

PINHO, C. M. D.; FEITOSA, F. F.; KUX, H. J. H. Classificação automática de cobertura do solo urbano em imagem Ikonos: comparação entre a abordagem pixel-a-

pixel e orientada a objetos. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO, 12. (SBSR), 2005, Goiânia. **Anais...** São José dos Campos: INPE, 2005. p. 4217-4224. CD-ROM, On-line. ISBN 85-17-00018-8. (INPE-12767-PRE/8057). Disponível em: <<http://urlib.net/ltid.inpe.br/sbsr/2004/11.19.17.58>>. Acesso em: 23 abr. 2015.

PORTER, R.; CANAGARAJAH, N. A Robust Automatic Clustering Scheme for Image Segmentation Using Wavelets. **IEEE Transactions on image processing**, v. 5, n. 4, Apr.l. 1996.

SHI, J. Z. et al. A Desk-top tutorial Demonstration of Model-based Fault Detection and Diagnosis. **International Journal of COMADEM**, v. 10, n. 4, p. 23-31, 2007. ISSN 1363-7681.

SIQUEIRA, J.E.M. **Estudo e melhoria de abordagens para a detecção de falhas em sistemas de controle reconfiguráveis**. São José dos Campos: INPE, 2015. 56 p. (sid.inpe.br/mtc-m21b/2015/06.08.12.07-PUD). Disponível em: <<http://urlib.net/8JMKD3MGP3W34P/3JKT2JH>>. Acesso em: 08 jul. 2015.

SWAGATAM, D.; AJITH, A. Automatic Clustering Using an Improved Differential Evolution Algorithm. **IEE transactions on systems, man, and cybernetics—part a: systems and humans**, v. 38, n. 1, Jan, 2008.

THOMÉ, A. **Processamento de imagens - tratamento da imagem – filtros**. **Rio de Janeiro: UFRJ, Transparências**. Disponível em: <http://equipe.nce.ufrj.br/thome/p_grad/nn_img/transp/c4_filtros.pdf>. Acesso em 17 Nov. 2014.

WEIR, M. D.; HASS, J.; GIORDANO, F. R. **Calculo**: George B. Thomas. Tradução de T. Guimarães 11. ed. São Paulo, SP: Pearson Addison Wesley, 2009. v. 1 . 783 p., il. Título original: Thomas' Calculus. Esta obra tem 2 volumes e exemplares. ISBN 978-85-88639-31-7.

GLOSSÁRIO

Cluster: Cada um dos subconjuntos resultantes da aplicação da técnica de clusterização.

Clusterização hierárquica: Método de clusterização que consiste em transformar uma matriz de dissimilaridade em um conjunto de partições sequenciais. A organização e formação destas partições dependem de como seja interpretada a matriz de dissimilaridade. O resultado deste método é um **dendograma**.

Clusterização particional: O método de clusterização particional consiste em selecionar um critério, avaliar os dados para todas as possíveis partições segundo esse critério e escolher a partição ótima.

Critério de parada: Limiar que define em que momento é considerado que foram atingidos os *clusters* ótimos.

Dendograma: É uma representação em forma de árvore do conjunto sequencial de partições calculadas segundo o método de clusterização hierárquica, onde cada nível representa uma partição e a junção em nodos das linhas representa a classificação de objetos em um mesmo *cluster*.

Histograma: Procedimento que consiste em separar os dados em janelas, de forma a associar a cada janela, um valor correspondente ao número de amostras que se encontram dentro dos limites dela.

Histograma natural: Histograma no qual é considerado que os vales representam separações naturais entre comportamentos dos dados.

Janela: Cada uma das divisões em que é separado o conjunto de dados para formar um histograma.

Técnica de clusterização: É o processo de classificar objetos em subconjuntos que possuem o mesmo significado em um contexto particular.

Medida de dissimilaridade: Relação existente entre cada par dos diferentes objetos a serem classificados segundo a técnica de clusterização.

Matriz de proximidade: Matriz que armazena as informações de dissimilaridade entre pares de objetos.

Técnicas supervisionadas: Conjunto de técnicas caracterizadas por utilizar o conhecimento prévio do sistema para classificar os dados de saída do mesmo.

Técnicas não supervisionadas: Conjunto de técnicas que utilizam diretamente os dados de saída do sistema para realizar a classificação, sem necessidade de conhecer o comportamento dele.

APÊNDICE A - PROPAGAÇÃO DE ERRO NA MEDIDA DE DISSIMILARIDADE

Este anexo é aplicável ao cálculo da medida de dissimilaridade utilizado na Seção 4.2.

O erro nos parâmetros de saída $[h_1, h_2, h_3]$ do sistema estudado (detecção de anomalias em tubulações hidráulicas, Seção 4.1) foi modelado como um ruído gaussiano $N(0, 0.01)$ [m]. Assim, chamando aos parâmetros reais de saída como $h_{i,real}$, eles podem ser representados como:

$$\begin{cases} h_1 = h_{1,real} \pm 0.01 \\ h_2 = h_{2,real} \pm 0.01 \\ h_3 = h_{3,real} \pm 0.01 \end{cases} \quad (A.1)$$

Quando se realizam operações de soma, diferença, produto e radical, os erros são propagados como segue:

Para soma,

$$\begin{aligned} (h_1 \pm \Delta h_1) + (h_2 \pm \Delta h_2) &= \\ &= (h_1 + h_2) \pm (\Delta h_1 + \Delta h_2) \end{aligned} \quad (A.2)$$

para diferença,

$$\begin{aligned} (h_1 \pm \Delta h_1) - (h_2 \pm \Delta h_2) &= \\ &= (h_1 - h_2) \pm (\Delta h_1 + \Delta h_2) \end{aligned} \quad (A.3)$$

para produto,

$$\begin{aligned} (h_1 \pm \Delta h_1) \times (h_2 \pm \Delta h_2) &= \\ &= (h_1 \times h_2) \pm \left(\frac{\Delta h_1}{h_1} + \frac{\Delta h_2}{h_2} \right) h_1 h_2 \end{aligned} \quad (A.4)$$

E para radical,

$$\sqrt[m]{(h_1 \pm \Delta h_1)} = \quad (A.5)$$

$$= \sqrt[m]{h_1} \pm \frac{\sqrt[m]{h_1} \Delta h_1}{m h_1}$$

Na hora de aplicar o algoritmo de clusterização, é utilizada a distância euclidiana e distância euclidiana quadrada como medida de dissimilaridade (conforme equações (4.4) e (4.5)):

$$\begin{aligned} d_{eucl} &= \sqrt{h_1^2 + h_2^2 + h_3^2} \\ d_{eucl2} &= h_1^2 + h_2^2 + h_3^2 \end{aligned} \quad (\text{A.6})$$

Considerando que um elemento qualquer de valores $[h_1, h_2, h_3]$ foi escolhido como centróide do *cluster* a ser formado, pode-se considerar que a propagação erro no cálculo da distância nesse elemento é:

Para qualquer parâmetro h_i com $i \in \{1,2,3\}$

$$\begin{aligned} h_i^2 &= (h_i \times h_i) \pm \left(\frac{\Delta h_1}{h_1} + \frac{\Delta h_1}{h_i} \right) h_i h_i = \\ &= h_i^2 \pm \frac{2h_i^2 \Delta h_i}{h_i} \\ &= h_i^2 \pm \frac{2\Delta h_i}{h_i} \end{aligned} \quad (\text{A.7})$$

Assim o cálculo do erro acumulado considerando as definições da distância euclidiana e euclidiana quadrada (equação A.6), caso se esteja trabalhando com a distância euclidiana quadrada d_{eucl2} é feito conforme se segue:

$$d_{eucl2} = (h_1^2 + h_2^2 + h_3^2) \pm \left(\frac{2\Delta h_1}{h_1} + \frac{2\Delta h_2}{h_2} + \frac{2\Delta h_3}{h_3} \right) \quad (\text{A.8})$$

e caso se esteja trabalhando com a distância euclidiana d_{eucl}

$$d_{eucl} = \sqrt{h_1^2 + h_2^2 + h_3^2} \pm \left(\frac{\sqrt{h_1^2 + h_2^2 + h_3^2}}{2} \frac{\frac{2\Delta h_1}{h_1} + \frac{2\Delta h_2}{h_2} + \frac{2\Delta h_3}{h_3}}{h_1^2 + h_2^2 + h_3^2} \right) \quad (\text{A.9})$$

APÊNDICE B - CÁLCULO DO LIMIAR DE PARADA DO ALGORITMO BABYLO-BR

O critério de parada do algoritmo BABYLO-BR está definido mediante a aplicação de um conjunto de teorias e técnicas baseadas em dados.

Para compreender porque foram aplicadas essas técnicas deve ser contextualizada a situação e enumeradas as dificuldades:

- 1) Como entrada, tem-se um histograma (unidimensional) com $n_{inicial} = 50$ janelas.
- 2) Como saída, quer-se obter um **histograma natural** onde tenham sido reduzidas as mudanças bruscas de tamanho entre janelas adjacentes.

Para estudar as mudanças nos tamanhos das janelas, utiliza-se o corolário da segunda derivada (vide Seção 2.5.4 para mais informações), de forma que, se existirem valores da segunda derivada maiores do que um **limiar** l_{parada} , considerar-se-á que o histograma está pouco compactado, criando muito ‘ruído’ (muitas variações nos tamanhos das janelas) e precisa ser mais compactado (menos janelas). Um exemplo desta situação seria a Figura B.1 onde se observam variações bruscas no histograma que se desejam evitar.

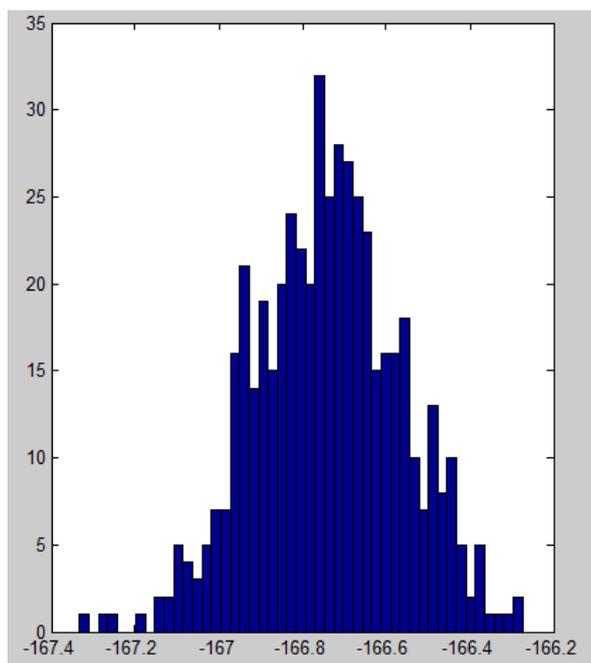


Figura B.1. Exemplo de histograma com variações bruscas onde x representa os valores do parâmetro discretizado e y o número de amostras por janela.

Porém, um limiar específico único não funcionará de forma geral para todos os tipos de sistemas; veja-se isto na Figura B.2. O gráfico B.2a mostra um histograma onde um limiar de 180 não deve provocar a compactação; no entanto, esse mesmo limiar de 180 deveria, sim, provocar a compactação para corrigir o vale inesperado surgido na janela 6 do histograma da Figura B.2b. Esta diferença deve-se principalmente à diferença entre o número de amostras totais que formam o conjunto do gráfico B.2a (aprox.. 250 amostras) e o gráfico B.2b (mais de 1500 amostras). Portanto, **1) o limiar deve depender do número de amostras.**

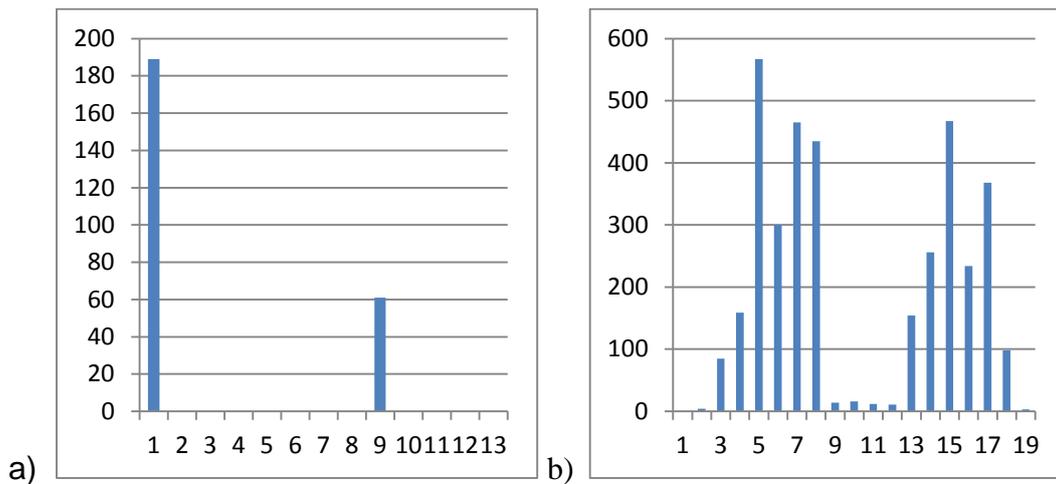


Figura B.2. a) Exemplo 1 de histograma; b) Exemplo 2 de histograma.

A Figura B.2 permite visualizar outra dificuldade: a compactação das amostras dentro do histograma. No gráfico B.2a existem unicamente 2 janelas com amostras, mesmo existindo 13 janelas totais; já no gráfico B.2b existem amostras em quase todas as janelas; mesmo assim, as janelas 9 até 12 contêm poucas amostras em relação às outras. Pode-se deduzir que, quanto mais acumuladas estejam as amostras em poucas janelas, maiores serão as variações da segunda derivada, sem isso significar câmbios bruscos e inesperados no histograma; assim **2) o limiar deve depender então do número de janelas que contém amostras**. Como as janelas com poucas amostras também influem no aumento dos valores da segunda derivada, elas também não serão levadas em conta.

Assim, foi decidido levar em conta as janelas que contivessem um número de amostras maior do que 10% da janela com número de amostras máximo, conforme equação (4.22).

Existe ainda outra dificuldade que deve ser resolvida. No gráfico B.2a, o valor máximo da segunda derivada é muito alto (aproximadamente 180). Sem observar o histograma, pareceria lógico querer compactar para evitar valores tão altos, mas observando o histograma, deduz-se que a compactação não vai diminuir o valor máximo da segunda derivada e, de fato, o histograma vai ser compactado iterativamente até chegar a um histograma de uma única janela. Como isto deve ser evitado, devem-se levar em conta as exceções como a mostrada no gráfico B.2a, onde as amostras estão compactadas em

uma, duas ou três janelas e não é necessário realizar compactação do histograma, pois é esperado que cada uma dessas janelas forme um *cluster* diferente. Estas exceções devem ser identificadas na primeira ou segunda iteração da compactação do histograma e isto pode ser feito definindo um limiar alto na primeira iteração e diminuindo ele conforme avançam as iterações mediante a utilização do fator k_{limiar} . Assim, **3) o limiar deve depender de um fator de porcentagem k_{limiar} .**

Para determinar o valor de k_{limiar} realizou-se um estudo empírico de conjuntos de dados distribuídos em determinado número de janelas como mostrado na Tabela B-1. As primeiras quatro linhas desta tabela correspondem a estudos de caso reais utilizados neste trabalho e os dois últimos casos foram criados por extensão dos estudos de caso reais, para abranger mais exceções. A primeira coluna determina o número de amostras n_a , a segunda coluna determina, uma vez feito o histograma inicial com $n_{inicial} = 50$ janelas, o número de janelas que contêm um número significativo de amostras ($n_{s\ 10\%}$ conforme equação (4.22)), a terceira coluna determina a quantidade de amostras teórica ($n_{a\ teór}$) que $n_{s\ 10\%}$ deveria ter (desconsiderando que existem janelas com baixo número de amostras), já a quarta coluna determina o número de amostras reais da janela com maior número de amostras (h_{max}). A quinta coluna determina o **fator de porcentagem** como a divisão entre o número de amostras teórico e o número de amostras real $\frac{h_{max}}{n_{a\ teór}}$.

Tabela B-1. Estudo empírico do fator de porcentagem para determinação do critério de parada.

n_a	n_s 10%	$n_a \text{ teór} = \frac{n_a}{n_s \text{ 10\%}}$	h_{max}	$\frac{h_{max}}{n_a \text{ teór}}$
250	1	250	204	0.81
250	1	250	215	0.86
250	2	125	189	1.57
250	2	125	197	1.51
250	3	85	136	1.6
250	4	62	100	1.6

O fator de porcentagem só deve ser aplicado em casos onde as **amostras estejam muito compactadas em poucas janelas**, pois nesses casos precisa-se de um limiar de parada com valor mais alto. Assim, o fator de porcentagem aparece multiplicando na definição do limiar de parada.

Agora falta determinar como identificar se as amostras estão ou não compactadas; para isso utiliza-se o conceito de **variança**, explicado na seção 2.5.4. O procedimento será realizado da seguinte forma; calcula-se a variança máxima possível σ_{max}^2 em h'' (i.e. quando todas as amostras estão compactadas em uma janela) e a variança real σ_{real}^2 em h'' .

Assim, para identificar se as amostras estão ou não compactadas, **4) o limiar deve depender da relação entre a variança máxima e a variança real**, de forma que, se a variança real for maior do que a metade da variança máxima, considera-se que as amostras estão muito compactadas e aplica-se k_{limiar} (equação (4.23)) no cálculo do limiar de parada l_{parada} (equação (4.21)).

$$\text{Se } \sigma_{real}^2 * 2 > \sigma_{max}^2 \text{ então é aplicado } k_{limiar} \quad (\text{B.1})$$

Por lógica, se o limiar é atingido (equação (4.20)), o critério de parada não é cumprido e se realiza a compactação do histograma. Observando da Tabela B-1, deduz-se então que o valor mínimo do fator de porcentagem que garante que todas as exceções sejam levadas em conta é $1.6=8/5$. Deve-se lembrar de que essas exceções são localizadas nas primeiras iterações da compactação do histograma. Por este motivo, k_{limiar} deve diminuir a cada iteração até ficar com valor unidade. Como o valor de k_{limiar} é 1.6, essa diminuição é feita subtraindo 0.1 a cada iteração ($i_{compact}$) durante 6 iterações (para chegar à unidade).

São levadas em conta então quatro variáveis para determinar o limiar, enumeradas a seguir:

- 1) Número de amostras n_a .
- 2) Número de janelas que contêm um número de amostras significativo $n_{s\ 10\%}$
- 3) Fator de porcentagem k_{limiar} .
- 4) Critério de variância para aplicação do fator de porcentagem (equação B.1).

Assim, o limiar l_{parada} é definido como uma função dependente destas quatro variáveis conforme a equação (4.21) repetida à continuação:

$$l_{parada} = \begin{cases} \frac{n_a}{n_{s\ 10\%}} k_{limiar}, & \text{se } 2\sigma_{real}^2 > \sigma_{max}^2 \\ \frac{n_a}{n_{s\ 10\%}}, & \text{se } 2\sigma_{real}^2 \leq \sigma_{max}^2 \end{cases}$$

e o fator de porcentagem k_{limiar} é calculado conforme a equação (4.23) repetida à continuação:

$$k_{limiar} = \begin{cases} \frac{8}{5} - 0.1 * i_{compact}, & \text{se } i_{compact} < 6 \\ \frac{8}{5}, & \text{se } i_{compact} \geq 6 \end{cases}$$

APÊNDICE C - OUTROS ESTUDOS DE CASO

C.1. Filtro de dados em construção de Bobinas de Helmholtz

Este estudo de caso é um caso real de dados, apresentados por Oliveira (2014), obtidos a partir da leitura do campo magnético durante o processo de calibração de um conjunto de Bobinas de Helmholtz com 3 eixos ortogonais e 1m de comprimento. O experimento consiste em comprovar se a bobina está corretamente calibrada, devolvendo o valor estático esperado durante um tempo $t = 200s$ e é realizado em uma sala onde existem perturbações eletromagnéticas que podem afetar às medidas.

Os dados obtidos em Oliveira (2014) apresentam as saídas do sistema simulado. Cada amostra é caracterizada por três parâmetros que representam os valores do campo $[B_x, B_y, B_z]$. Essa saída será utilizada como entrada no algoritmo BAbyLO-BR.

Neste estudo de caso, trabalha-se com dados provenientes de um sistema estático (um sistema calibrado deve apresentar o mesmo comportamento em diferentes instantes de tempo. Se isto não acontecer, deve-se a uma perturbação que está provocando o desvio na medida ou uma falha do sistema). As amostras ($n_a = 500$) foram tomadas em vários instantes diferentes durante a mesma simulação. Foram realizadas 3 simulações que sofreram diferentes anomalias devido possivelmente às perturbações eletromagnéticas da sala onde foram realizadas as medidas. Estas simulações são:

- Caso 1) Simulação sem perturbação.
- Caso 2) Simulação com perturbação isolada em um instante de tempo e afastada do grupo de amostras de comportamento normal.
- Caso 3) Simulação com falha no sistema que provoca dois comportamentos diferenciados no conjunto de amostras.

C.1.1. Caso 1) Simulação sem perturbação

Esta simulação mostra um comportamento normal das amostras, sem anomalias. A Figura C.1a mostra a distribuição das amostras em função dos parâmetros característicos $[B_x, B_y, B_z]$. Espera-se que o algoritmo reconheça um *cluster* abrangendo

todas as amostras. Pode acontecer que seja criado um *cluster* central com alguns *clusters* ao redor devido ao ruído que provoca desvio de algumas das amostras.

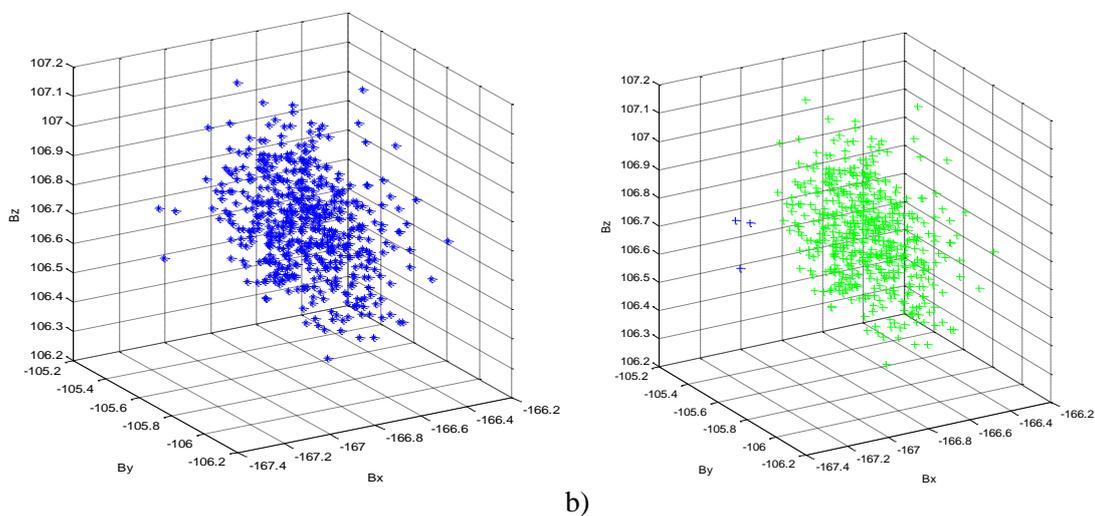


Figura C.1. Representação espacial das amostras de campo magnético em caso de simulação sem perturbação, a) sem classificação e b) após classificação pelo algoritmo.

A classificação surgida uma vez aplicado o algoritmo é a mostrada na Figura C.1b. Como esperado, todas as amostras foram agrupadas como um único comportamento a exceção de 3 amostras que foram classificadas como um comportamento diferente.

Como não existe conhecimento dos resultados esperados, não pode ser avaliado se as amostras em azul são anomalias ou amostras de comportamento normal (falsos positivos), mas em conjunto, pode-se dizer que a classificação conseguida corresponde ao esperado.

C.1.2. Caso 2) Simulação com perturbação em um instante de tempo isolada

Este estudo de caso mostra um comportamento evolutivo das amostras sobre o eixo x e uma anomalia distante. A Figura C.2a mostra a distribuição das n_a amostras em função dos parâmetros característicos $[B_x, B_y, B_z]$.

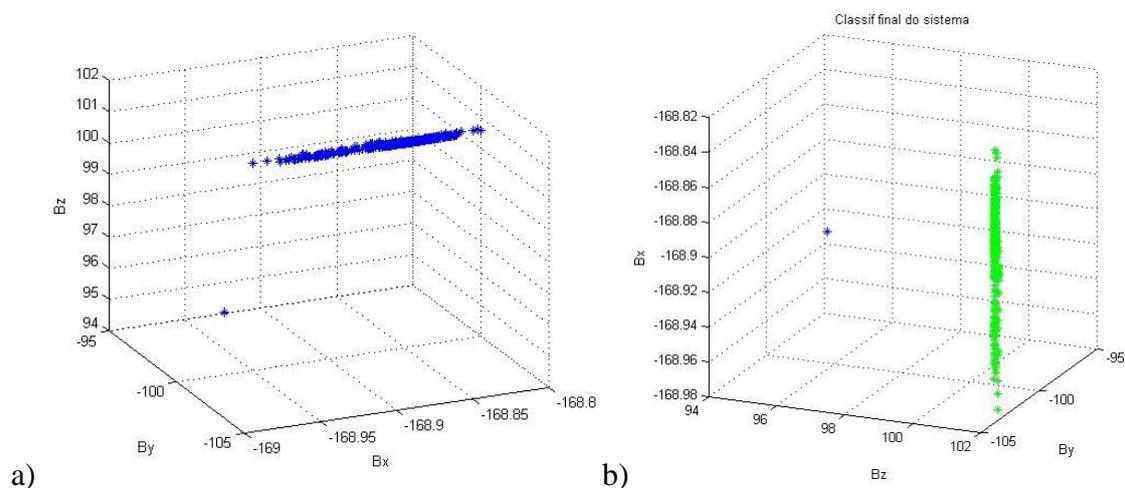


Figura C.2. Representação espacial das amostras de campo magnético em caso de funcionamento com anomalia isolada, a) antes da classificação e b) depois da classificação.

Espera-se que o algoritmo reconheça e separe a anomalia. Quanto à agrupação das amostras alinhadas, espera-se que todas elas sejam agrupadas em um *cluster*.

A Figura C.2b mostra a classificação feita pelo algoritmo. Ele respondeu como era visualmente esperado, separando a amostra afastada em um *cluster* diferente e considerando o resto de amostras como um comportamento único. Com este estudo de caso pode-se conferir que o algoritmo consegue criar *clusters* cuja distribuição não é gaussiana, mas continua em um dos eixos. Pela forma em que foi implementado o algoritmo cabe esperar este comportamento, pois ele vai procurar primeiro os *clusters* unidimensionais (projeções das amostras em cada eixo) para depois fazer a combinação deles.

C.1.3. Caso 3) Conjunto de amostras distribuídas em dois comportamentos diferenciados.

A última simulação contempla uma perturbação continuada. A partir, aproximadamente, do instante 100 s da simulação acontece uma perturbação que desvia os valores de saída medidos nos eixo x e z conforme mostrado na Figura C.3.

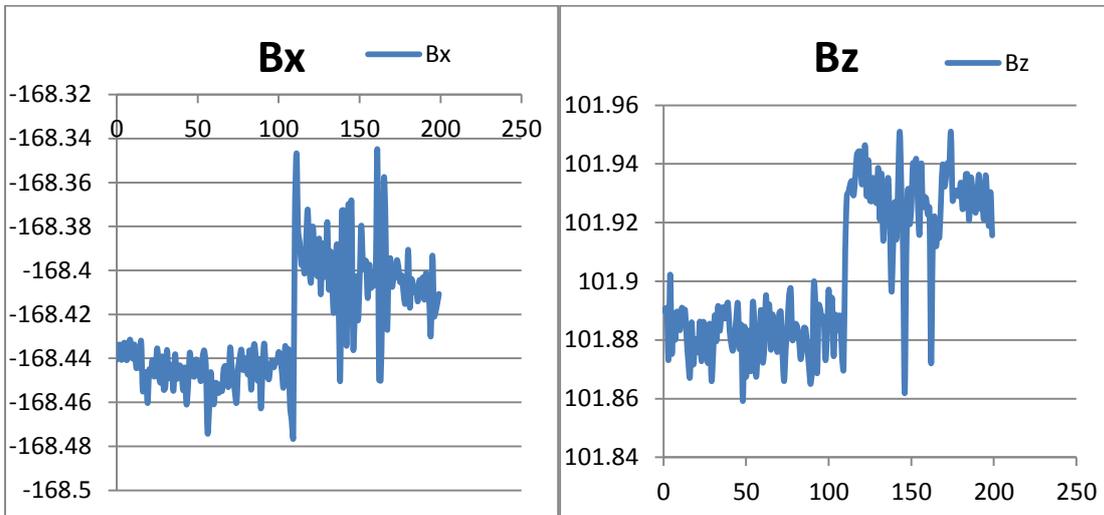


Figura C.3. Evolução temporal dos parâmetros B_x e B_y com perturbação perto de $t=100s$.

A Figura C.4a mostra a distribuição das amostras em função dos parâmetros característicos $[B_x, B_y, B_z]$. Espera-se que o algoritmo reconheça os dois comportamentos e separe os cantos superior direito e inferior esquerdo. As amostras da direita que estão um pouco distantes do comportamento superior poderiam ser agrupadas em outro *cluster*.

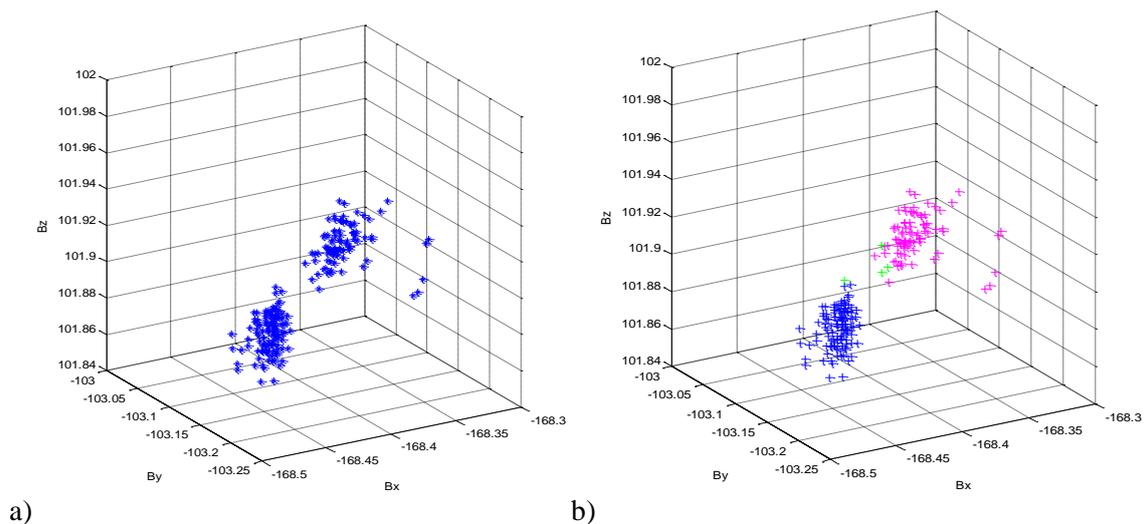


Figura C.4. Representação espacial das amostras de campo magnético em caso de dois comportamentos, a) sem classificação e b) após classificação pelo algoritmo.

A Figura C.4b mostra a classificação feita pelo algoritmo; como esperado, o algoritmo captou os dois comportamentos reconhecidos visualmente e criou dois agrupamentos. Ainda, ele achou outro *cluster* (em verde) que deveria ser estudado pelo analista para avaliar seu comportamento.

C.1.4. Utilidade de aplicação do algoritmo

Como já foi falado, não existe conhecimento a priori da classificação esperada destes dados. Porém, é esta a utilidade que pretende ter este algoritmo; ajudar ao analista na compreensão dos dados quando este não tem conhecimento prévio dos comportamentos esperados. Assim, a classificação realizada pelo algoritmo pode auxiliar na compreensão dos comportamentos do sistema.

Em estudos de caso como este de filtro de dados em construção de Bobinas de Helmholtz, onde o analista é visualmente capaz de reconhecer os comportamentos por estar em um caso tridimensional, não parece útil a aplicação do algoritmo, embora isso não seja certo; a aplicação do algoritmo permitiu obter de forma rápida os valores limítrofes de cada comportamento.

Além disso, a aplicação do algoritmo em casos com até três dimensões permitem comprovar a eficiência do algoritmo para a posterior aplicação em casos com mais de 3 parâmetros, onde o ser humano não tem tanta capacidade de visualizar o resultado utilizando representação em figuras.

C.2. Identificação de padrões espaço-temporais da ocorrência de pixels de calor na superfície brasileira

Os focos de calor utilizados nesse estudo correspondem ao produto MCD14ML derivados do sensor MODIS a bordo das plataformas orbitais Terra e Acqua. Estes dados são estudados pelo grupo de trabalho DSR do INPE.

A série temporal dos dados cedidos está composta por um conjunto de $n_a = 614.500$ amostras com parâmetros apresentados na tabela (Tabela C-1). O conjunto de amostras compreende dados recolhidos pelo sensor entre Janeiro de 2001 a Dezembro de 2010.

Devido à grande quantidade de dados disponíveis, precisa-se realizar um tratamento prévio deles escolhendo unicamente os parâmetros mais representativos. Escolheu-se usar os parâmetros Dia (após tratamento da data), Lat, Lon, aplicando filtros nos parâmetros de confiabilidade (=100%) e na FPR (>1000). Assim, a matriz de dados de entrada do algoritmo está composta por $n_a = 1867$ amostras com $n_p = 3$ parâmetros cada uma, a saber, Dia, Lat e Long.

Tabela C-1. Identificação das informações disponibilizadas no produto MODIS MCD14ML.

Coluna	Nome	Unidade	Descrição
1	YYYYMMDD	-	Ano (YYYY), Mês (MM) e Dia (DD)
2	HHMM	-	Hora (HH) e Minuto (MM)
3	Sat	-	Satélite: Terra (T) ou Aqua (A)
4	Lat	Graus	Latitude no centro do pixel de calor
5	Lon	Graus	Longitude no centro do pixel de calor
6	T 21	Kelvin	Temperatura de brilho do pixel de calor na banda 21
7	T 31	Kelvin	Temperatura de brilho do pixel de calor na banda 31
8	Sample	-	Número da amostra (entre 0-1353)
9	FRP	MW	Força radiativa do fogo (<i>Fire Radiative Power</i>)
10	conf	%	Grau de confiança (entre 0 e 100)

O objetivo principal deste estudo de caso consiste em avaliar a existência de correlação nos dados de queimadas (comportamentos diferentes no espaço-tempo).

Neste estudo de caso trabalha-se com dados provenientes de um sistema estático cujas amostras foram tomadas de vários instantes diferentes (dias diferentes do ano e vários anos).

O algoritmo BAbyLO-BR será aplicado sobre dois conjuntos de dados, 1) um conjunto que representa a área completa de Brasil e 2) um subconjunto do conjunto total que representa a área correspondente a Mato Grosso.

Como no estudo de caso anterior, para este estudo, não existe conhecimento a priori do número de *clusters* esperado nem da classificação esperada. Porém, os resultados obtidos foram avaliados pelos analistas da área DSR do INPE. Estes resultados e avaliações são mostrados a seguir.

C.2.1. Brasil

Estudou-se inicialmente o conjunto de dados completo referente à superfície do Brasil. A Figura C.5 mostra a) o conjunto de amostras de entrada do algoritmo BAbyLO-BR e b) a classificação realizada pelo algoritmo.

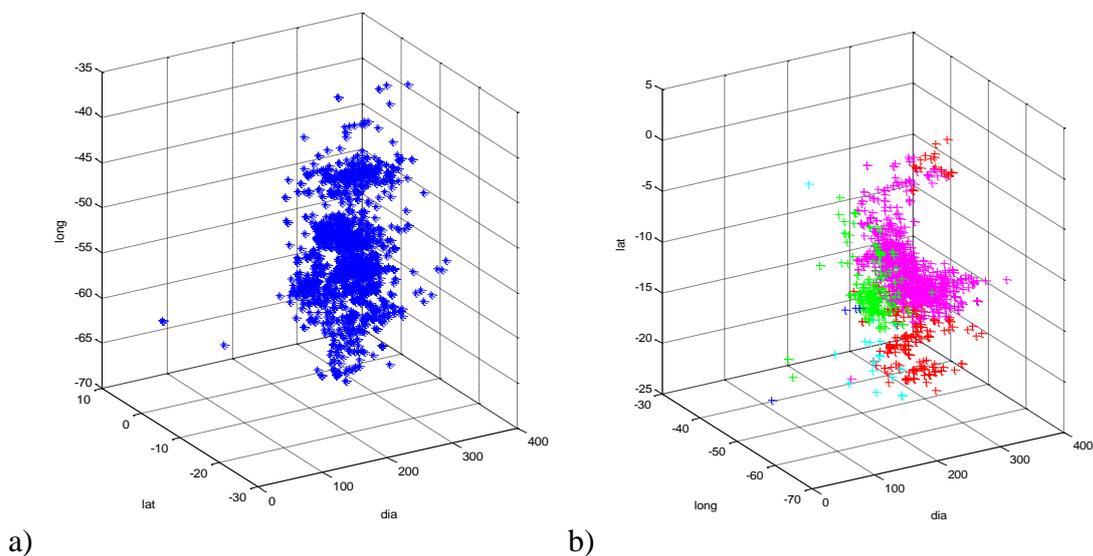


Figura C.5. Focos de calor no Brasil a) sem classificação e b) classificados pelo algoritmo BAbyLO-BR.

Esta classificação foi sobreposta a um mapa de chuva do Brasil para poder avaliá-la, conforme a Figura C.6.

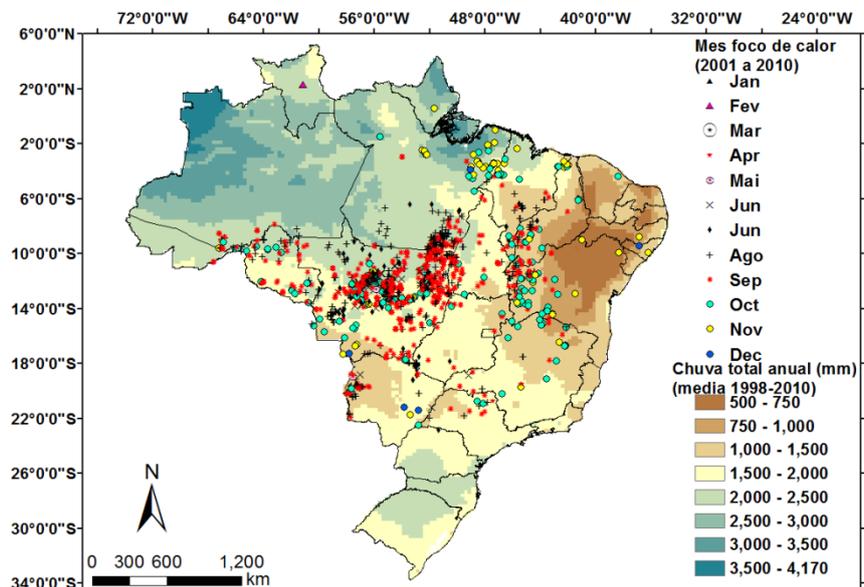


Figura C.6. Distribuição dos focos de calor mês a mês, após a aplicação do algoritmo sobreposto a um mapa de chuva anual acumulada, derivada do TRMM.

Conforme identificado em Gayarre et al. (2015), observa-se que os *clusters* referentes aos meses de Agosto, Setembro e Outubro são os que representam a maioria dos focos observados para o Brasil. O predomínio destes focos ocorre nas regiões em que a precipitação acumulada anual varia entre 1,500 mm a 2,500 mm de chuva e com marcada sazonalidade entre estações de chuva e de seca. Esta região tem predomínio de vegetação do tipo cerrado e de transição entre cerrado e floresta amazônica, destacando-se os Estados do Mato Grosso e Rondônia, que lideraram as taxas de desmatamento na última década. Já os *clusters* de focos observados em Novembro e Dezembro são encontrados permeando as bordas dos *clusters* descritos acima, formando um raio latitudinal compreendendo do leste da Amazônia ao Estado de São Paulo. Finalmente, observa-se o *cluster* referente ao mês de fevereiro, localizado em Roraima, extremo Norte do país. Por esta região encontrar-se no Hemisfério Norte, a estação seca ocorre entre Outubro e Março, e conforme os resultados desta análise apontam para Fevereiro como mês crítico da série histórica de queimadas neste estado. Os padrões observados

na ocorrência de focos de calor está em acordo com estudos descritos para a Amazônia (ARAGÃO ET AL., 2007), Mato Grosso (COUTINHO, 2005), Roraima (BARBOSA ET AL., 1999) e com as queimadas associadas a colheita de cana-de-açúcar observadas nos anos de 2006 a 2011 (AGUIAR ET AL., 2011).

C.2.2. Mato Grosso

Este caso se mostra nesta análise, não pelos resultados obtidos na classificação dos focos de calor na região de Mato Grosso, mais pelo interesse analítico que proporcionou a aplicação do algoritmo.

Por ser um subconjunto de amostras do estudo de caso anterior, $n_a = 177.000$. Igualmente, foram considerados os três parâmetros Dia, Lat e Long, filtrados segundo os valores de confiabilidade = 100% e FPR > 1000.

A Figura C.7a mostra os dados sem classificar. É possível ver que existem valores anômalos; esta anomalia se deve a que eles não pertencem à área de Mato Grosso. Este tipo de anomalias, devidas unicamente à seleção errada dos dados por parte do analista, é um problema reconhecido pelos analistas das áreas. A aplicação do algoritmo permitiu identificar de forma rápida as amostras anômalas (conforme Figura C.7b, *clusters* vermelho e azul) e elas foram retiradas do conjunto de dados no início do estudo.

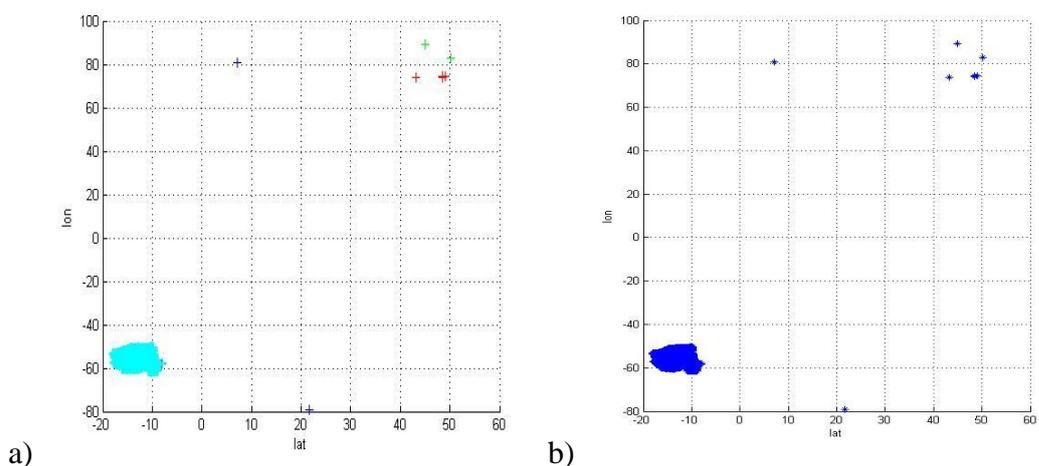


Figura C.7. Distribuição de focos de calor na área de Mato Grosso, a) antes de aplicar o algoritmo e b) depois de aplicar o algoritmo.

C.2.3. Utilidade de aplicação do algoritmo

A aplicação do algoritmo no conjunto de dados de Mato Grosso demonstrou a utilidade de identificar anomalias de forma rápida e direta. Esta utilidade pode ser considerada prescindível quando se trata de conjunto de dados uni/bi/tridimensionais, pois a representação gráfica dos dados permite enxergar os mesmos resultados que o algoritmo oferece, porém a eliminação delas tenha que ser feita de forma manual. O algoritmo identifica as anomalias no conjunto de dados de entrada e elas podem ser eliminadas diretamente.

Quando as amostras estão caracterizadas por mais de 3 parâmetros, a visualização por figuras destas anomalias resulta mais difícil e é comum realizar um estudo detalhado manual parâmetro a parâmetro para identificar as amostras não corretas (conforme afirmado pelos analistas da área). Este estudo leva muito tempo, que a utilização do algoritmo pode poupar.

APÊNDICE D - DINÂMICA DA PMM

Conforme pode ser observado nos trabalhos de (GOBATO, 2006), (LEITE, 2007), (LUSTOSA, 2008), (MANELLI NETO, 2011) e (AMARAL, 2013), a PMM é um sistema físico de dinâmica lenta. Pode-se esperar que suas respostas não sejam abruptas, se comparadas numericamente a respostas de sistemas mais rápidos, como aeronaves e veículos atmosféricos autônomos. Assim, a característica dinâmica do satélite age como uma filtragem “natural” à influência de ruído (como por exemplo, daquele devido à discretização presente na malha de controle) ou de perturbações externas (como torques gravitacionais ou aerodinâmicos de órbita baixa).

A seguir são mostradas características de resposta nos domínios da frequência e do tempo que sustentam esta afirmação.

A partir do trabalho de GOBATO (2006), observando o sistema de equações linearizado e desacoplado para cada um dos três eixos da PMM e levantando-se as funções de transferência entre as entradas de controle (no caso, as tensões de saída dos controladores) e a saída como as taxas angulares, obtêm-se as seguintes funções de transferência:

Para o rolamento do eixo X, e representado na Figura D.1:

$$\frac{V(s)}{\dot{\phi}(s)} = \frac{0.0002029 s + 6.49 \cdot 10^{-21}}{s^2 + 0.05 s} \quad (D.1)$$

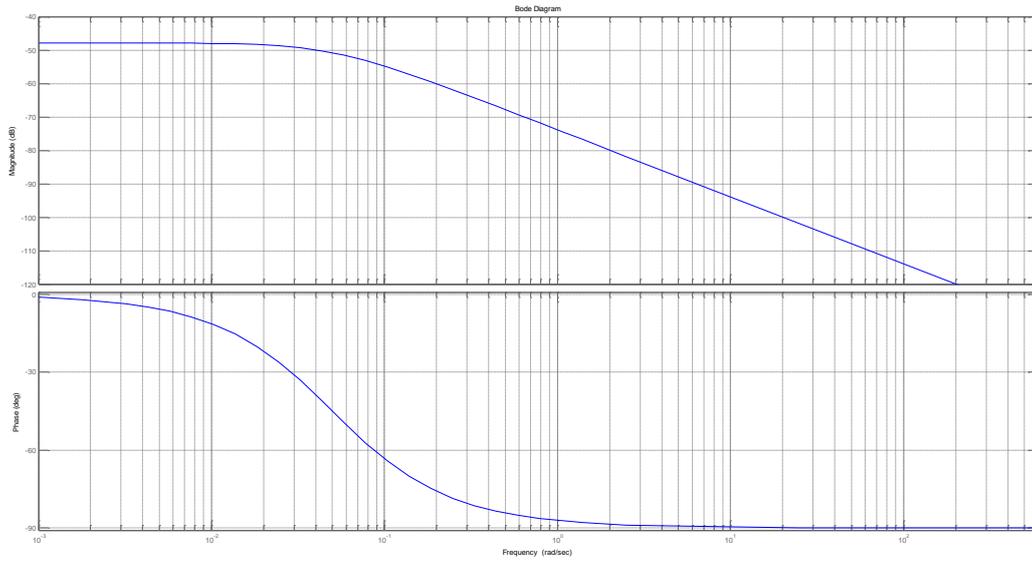


Figura D.1 Resposta em frequência, eixo x, linearizado e desacoplado, malha aberta.

Para o rolamento do eixo Y e representado na Figura D.2:

$$\frac{V(s)}{\hat{\theta}(s)} = \frac{0.0001197s + 1.694 \cdot 10^{-21}}{s^2 + 0.05 s} \quad (D.2)$$

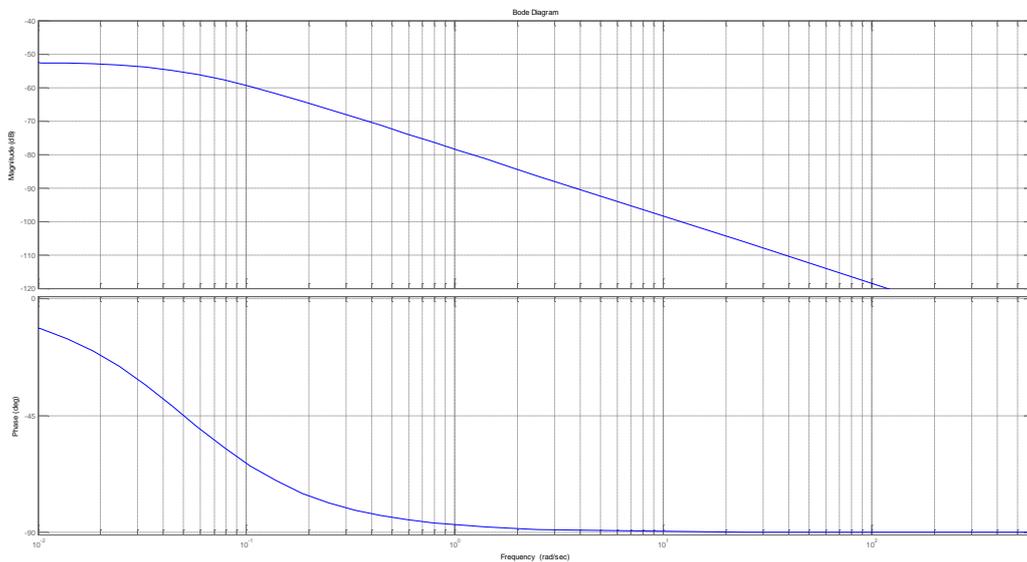


Figura D.2. Resposta em frequência, eixo y, linearizado e desacoplado, malha aberta.

Para o rolamento do eixo Z e representado na Figura D.3:

$$\frac{V(s)}{\dot{\psi}(s)} = \frac{0.0001645 s + 2.265 \cdot 10^{-21}}{s^2 + 0.05 s} \quad (D.3)$$

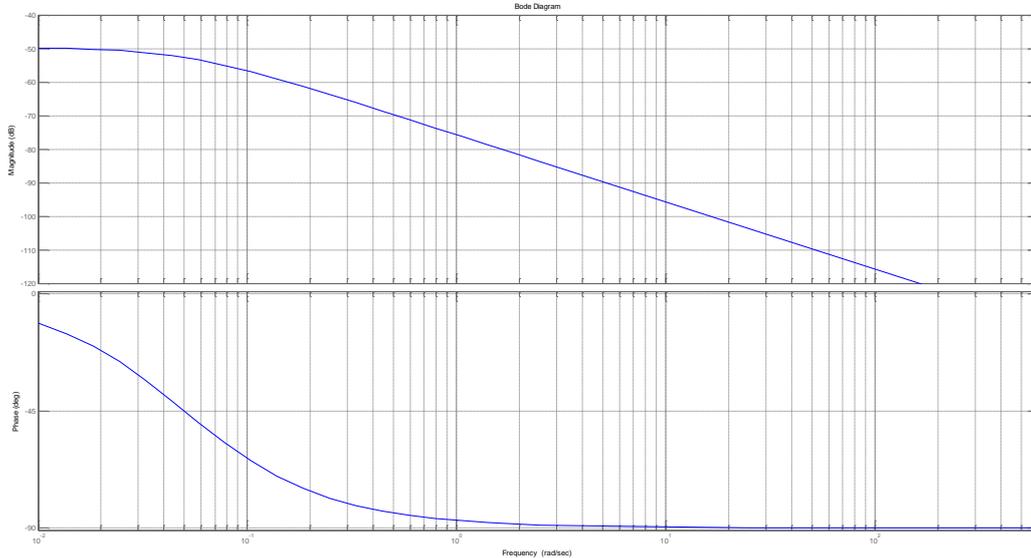


Figura D.3. Resposta em frequência, eixo z, linearizado e desacoplado, malha aberta.

Nitidamente, todas as funções compartilham o mesmo denominador, que tem como características um pólo na origem e outro no semi-plano esquerdo, porém próximo da origem. Conforme ensina Ogata (1997), o pólo próximo sobre a origem proporciona ao sistema uma atenuação em magnitude da ordem de $-20dB/dec$ e o pólo do sistema de primeira ordem, também $-20dB/dec$ (a partir de $\omega \geq \frac{1}{T} = 0.05 rad/s$). Logo, os termos que constam dos denominadores imputam ao sinal de entrada uma atenuação da ordem de $-40dB/dec$.

O termo de primeira ordem no numerador tem ordem de grandeza de 10^{-17} e na prática poderia ser reduzido a um zero posicionado sobre a origem, mas também se posiciona sobre o semi-plano esquerdo. Em qualquer caso, contribui com $20dB/dec$ desde muito próximo à origem no domínio da frequência.

Assim, no gráfico de magnitude para um Diagrama de Bode, entende-se que entre a origem e os zeros, os sinais de entrada não sofrem alteração em sua magnitude. Já a partir de $0.05 rad/s$, passariam a sofrer atenuação de $-20dB/dec$.

Mais detalhes fogem do escopo e da proposta deste trabalho e podem ser encontrados em GOBATO (2006).

APÊNDICE E - COMPORTAMENTOS NORMAL E FALHADOS DA PMM

Este apêndice apresenta graficamente o comportamento normal e a propagação das falhas simuladas na PMM e que são objeto de estudo neste trabalho.

Para cada simulação são apresentadas duas figuras com 6 gráficos cada uma: A primeira figura mostra as medidas tomadas pelos sensores de estrelas nos três eixos (esquerda) e as velocidades indicadas pelos giroscópios também nos três eixos (direita). A segunda figura mostra o sinal de controle (esquerda) e a taxa angular das rodas de reação nos três eixos (direita). Assim, para observar a falha inserida no giroscópio do canal X deve-se estudar o gráfico superior direito da primeira figura e para observar a falha inserida na roda de reação deve-se estudar o gráfico superior esquerdo da segunda figura de cada simulação. O resto de gráficos mostra a propagação da falha nos outros componentes.

A Figura E.1 e Figura E.2 abaixo foram obtidas a partir da simulação de caso em Modo Normal, demonstrando a característica de dinâmica lenta da PMM explicada no APÊNDICE E - .

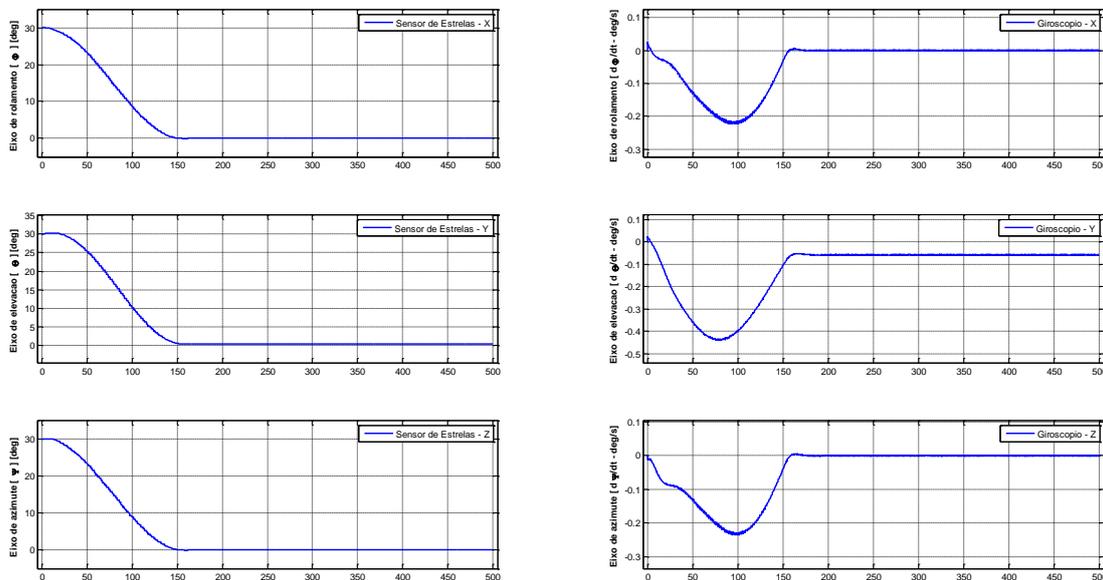


Figura E.1. Ilustração da manobra de *de-tumble* com R-LQR (parâmetros de sensores, valores angulares convertidos para graus, graus/s, graus/s²).

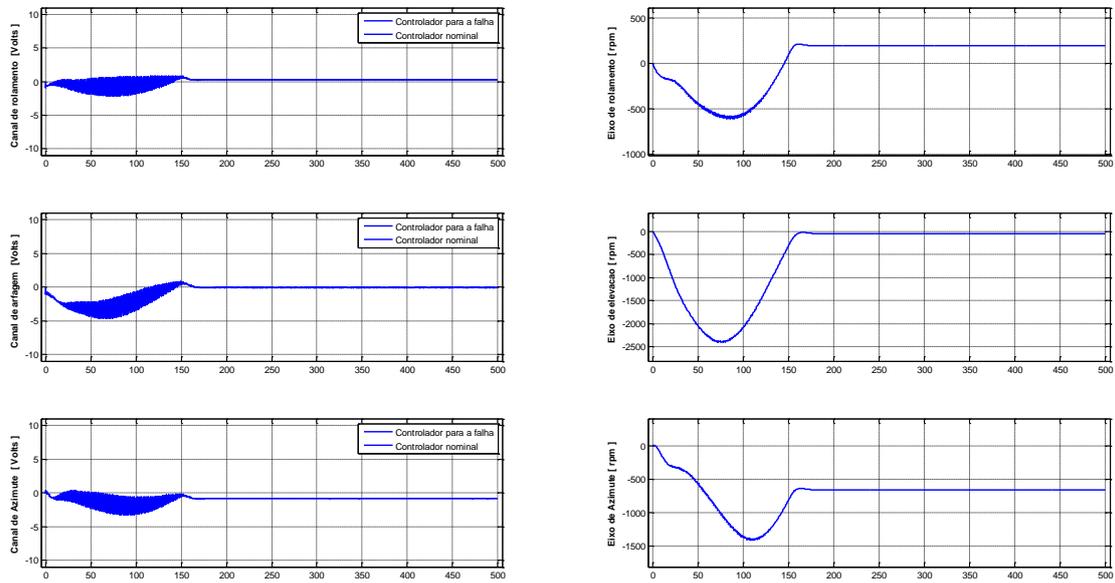


Figura E.2. Ilustração da manobra de *de-tumble* com R-LQR (parâmetros de atuadores).

As Figuras mostradas à continuação mostram a propagação das 4 falhas de sensores simuladas na plataforma PMM e objeto de estudo deste trabalho:

A Figura E.3 e a Figura E.4 mostram a propagação da falha F1 (**Indicação de valor constante, fixado em zero**) no canal do eixo X, injetada em $t = 30$ segundos;

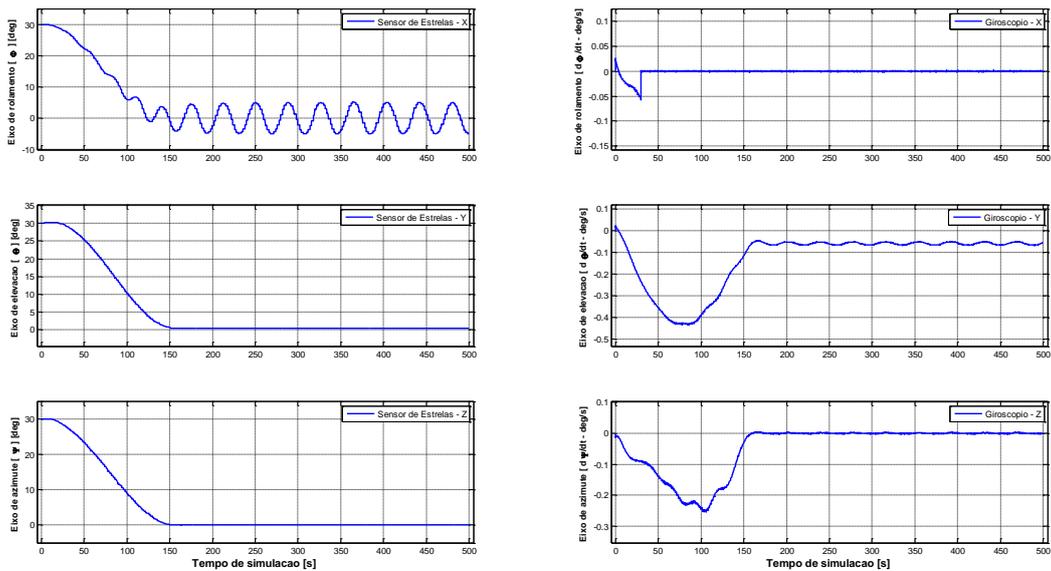


Figura E.3. Propagação da falha F1, eixo X, $t = 30$ segundos.

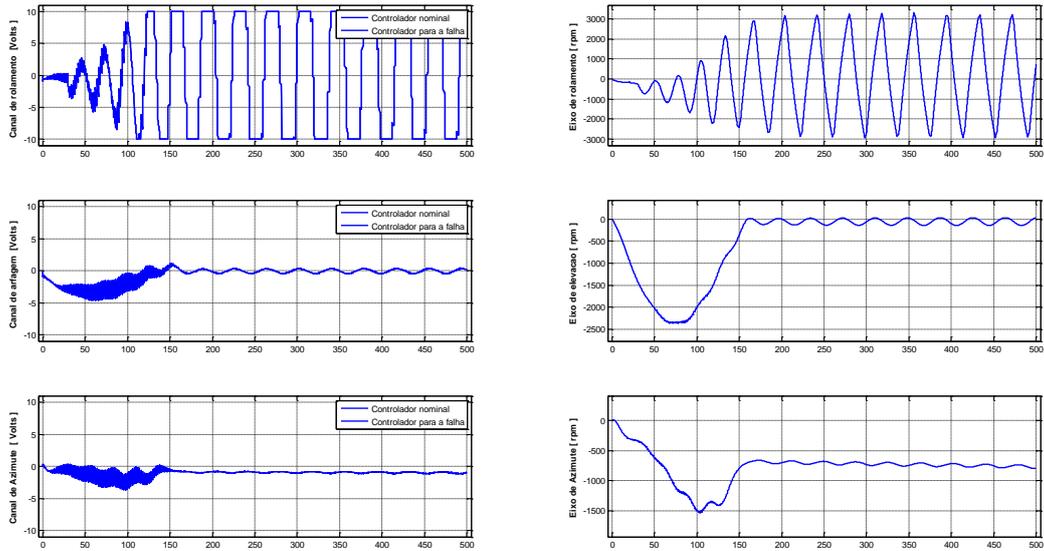


Figura E.4. Propagação da falha F1, eixo X, t = 30 segundos.

A Figura E.5 e a Figura E.6 mostram a propagação da falha F2 (**Indicação de valor constante, congelado no último valor válido**) no canal do eixo X, injetada em t = 30 segundos;

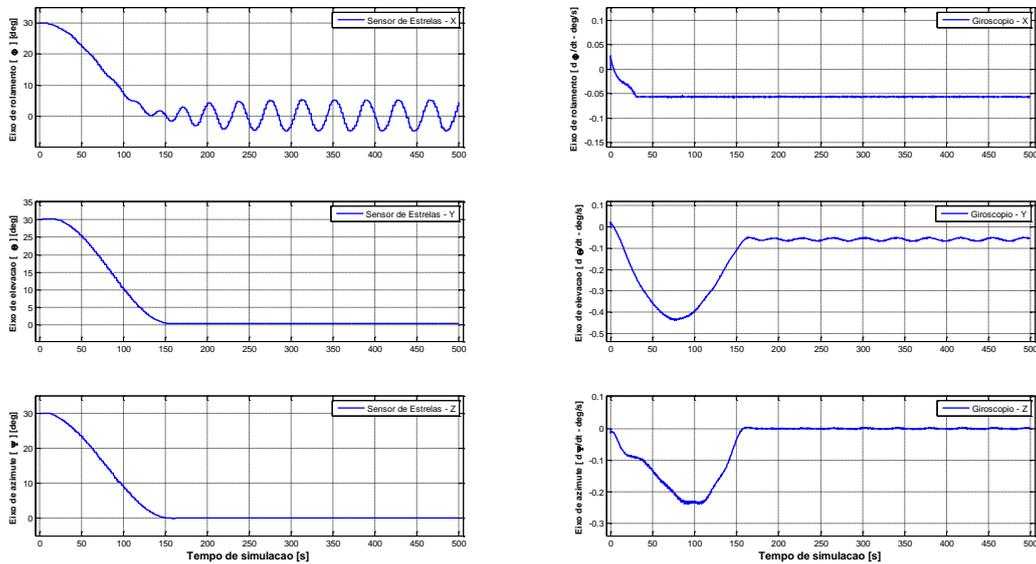


Figura E.5. Propagação da falha F2, eixo X, t = 30 segundos.

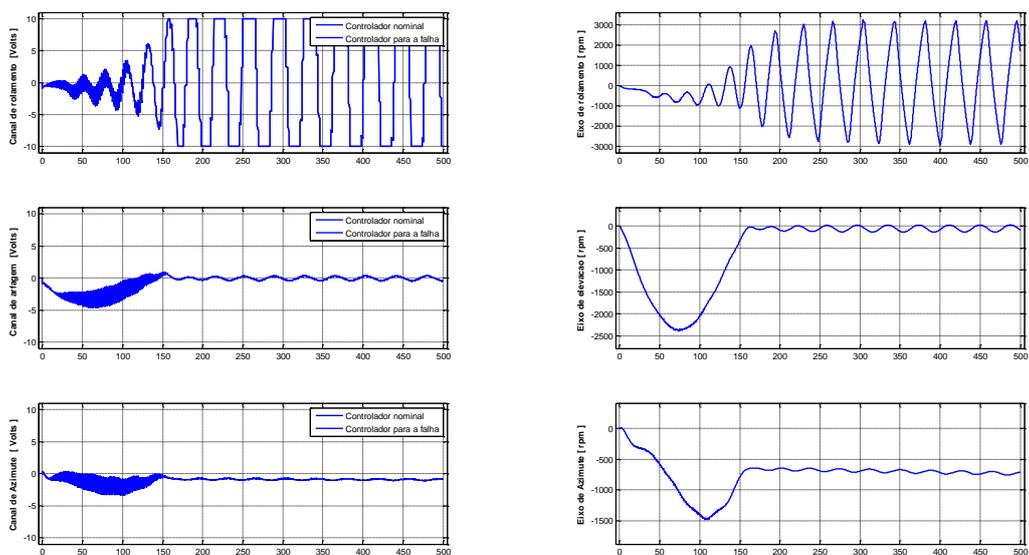


Figura E.6. Propagação da falha F2, eixo X, $t = 30$ segundos.

A Figura E.7 e a Figura E.8 mostram a propagação da falha F3 (**Indicação de valor sujeito a deriva de *offset***) no canal do eixo X, injetada em $t = 30$ segundos;

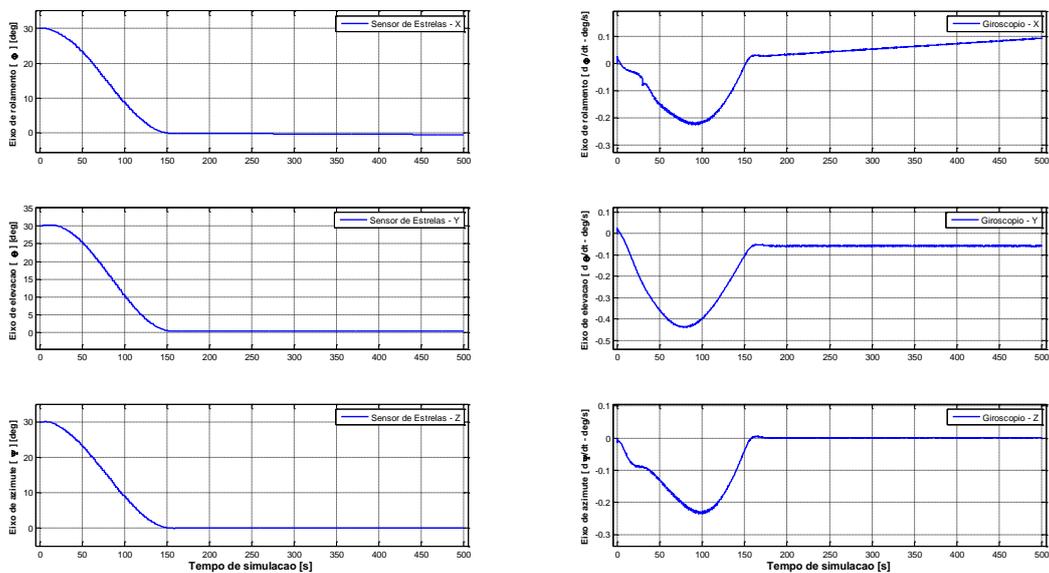


Figura E.7. Propagação da falha F3, eixo X, $t = 30$ segundos.

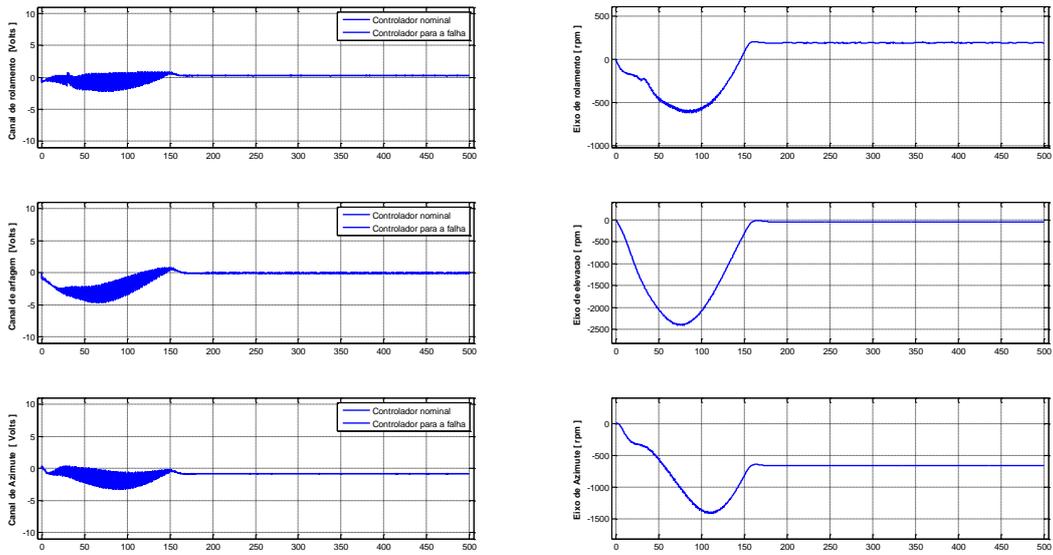


Figura E.8. Propagação da falha F3, eixo X, t = 30 segundos.

A Figura E.9 e a Figura E.10 mostram a propagação da falha F4 (**Indicação de valor sujeito a deriva de fator de escala**) no canal do eixo X, injetada em t = 30 segundos;

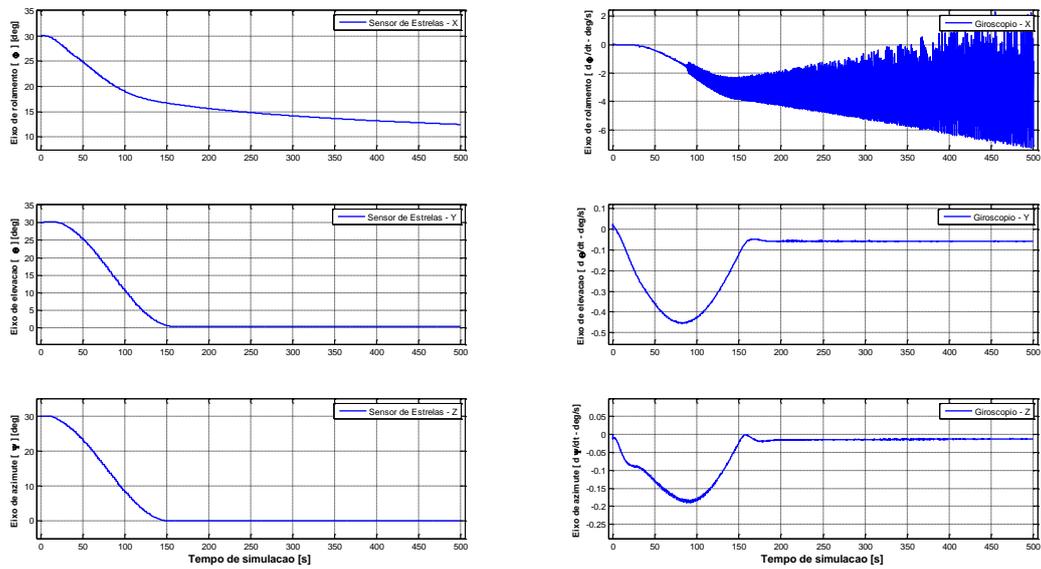


Figura E.9. Propagação da falha F4, eixo X, t = 30 segundos.

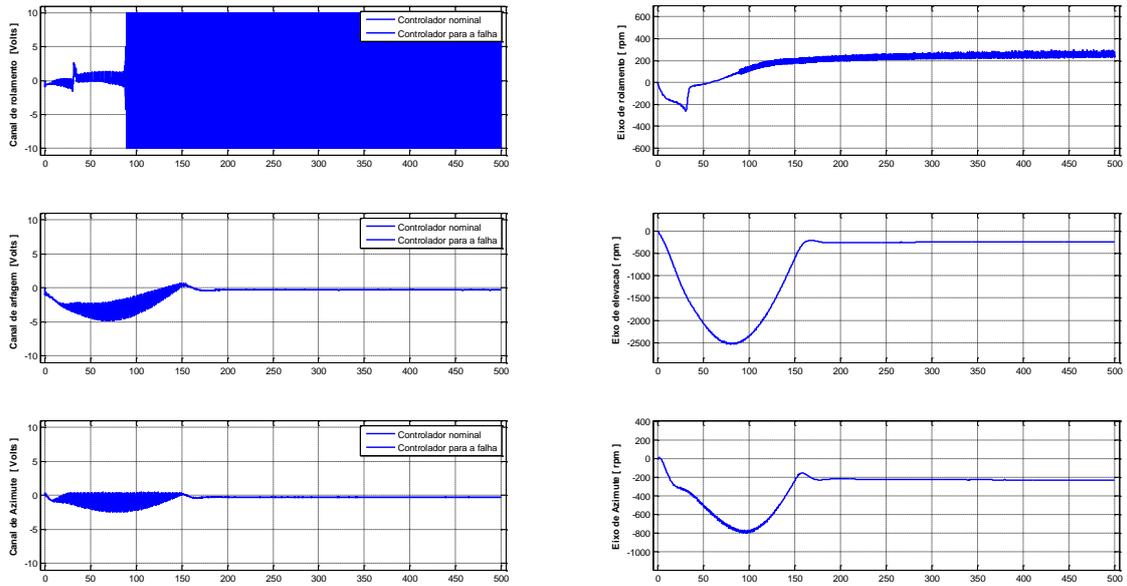


Figura E.10. Propagação da falha F4, eixo X, $t = 30$ segundos.

Da mesma forma, as figuras mostradas a continuação mostram a propagação das 4 falhas de atuadores simuladas na plataforma PMM e objeto de estudo deste trabalho

A Figura E.11 e a Figura E.12 mostram a propagação da falha T1 (**Saturação do comando de potência**) na roda de reação do eixo X, em $t = 30$ segundos.

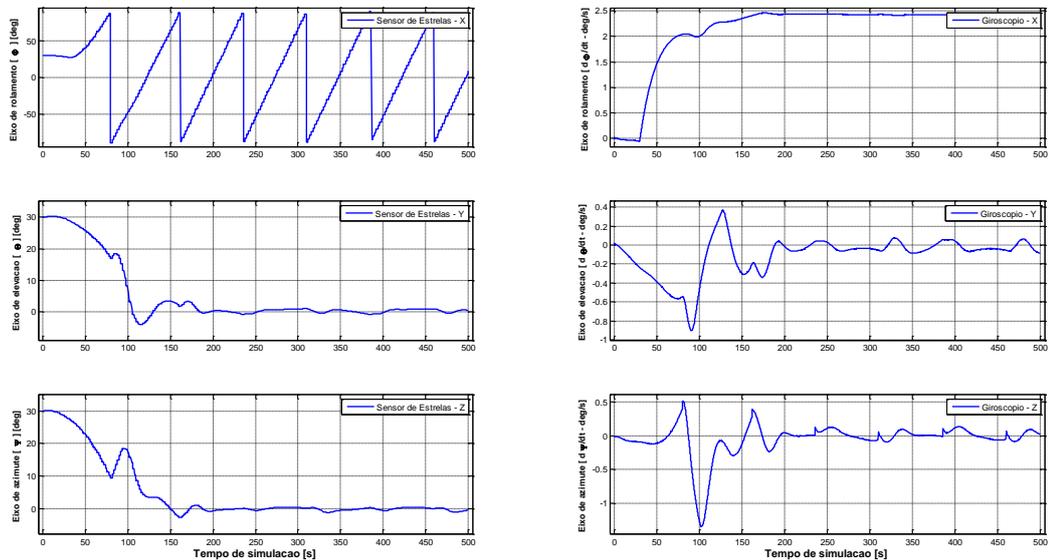


Figura E.11. Propagação da falha T1, eixo X, $t = 30$ segundos.

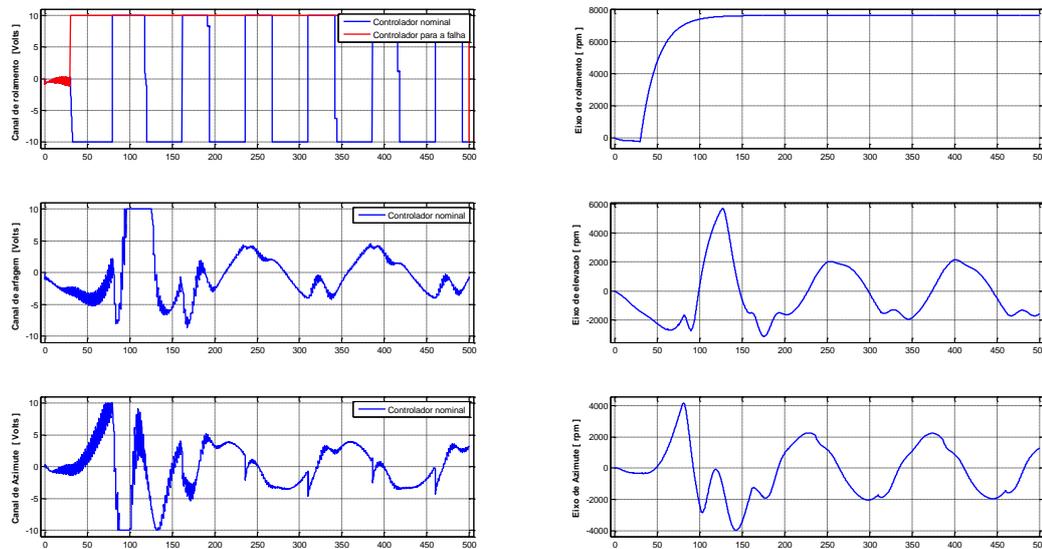


Figura E.12. Propagação da falha T1, eixo X, t = 30 segundos.

A Figura E.13 e a Figura E.14 mostram a propagação da falha T2 (**Falência da roda de reação**) na roda de reação do eixo X, em t = 30 segundos.

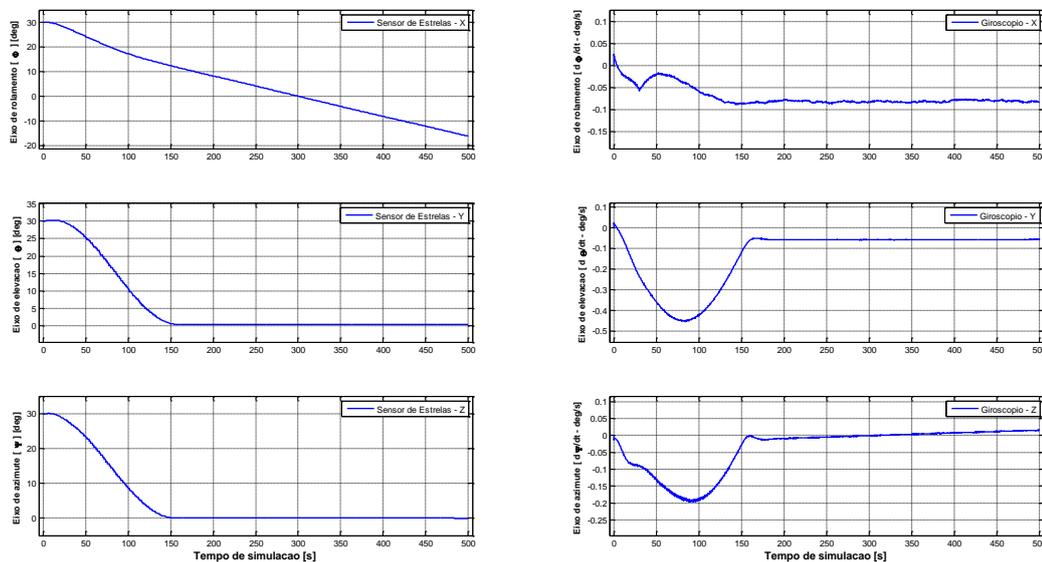


Figura E.13. Propagação da falha T2, eixo X, t = 30 segundos

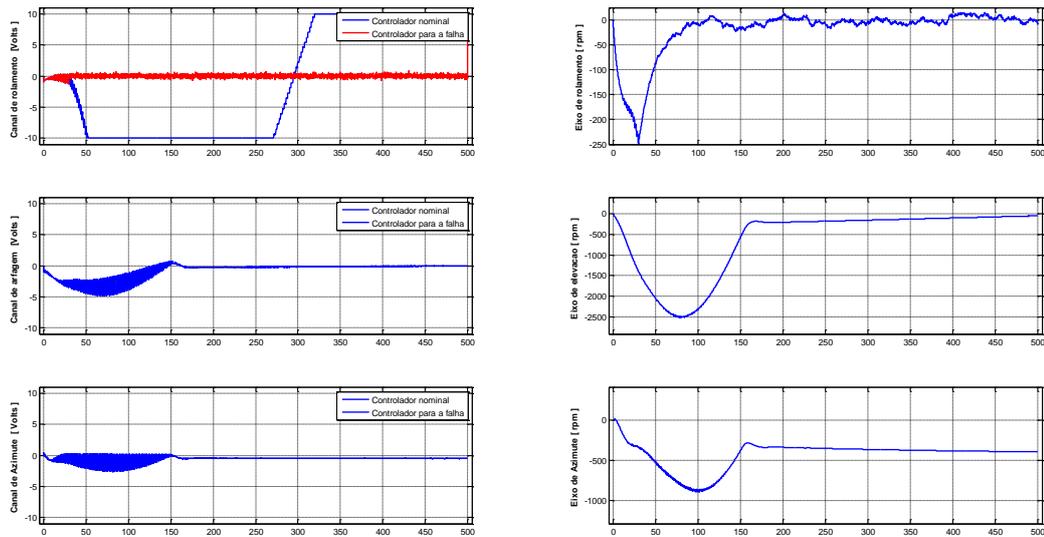


Figura E.14. Propagação da falha T2, eixo X, t = 30 segundos.

A Figura E.15 e a Figura E.16 mostram a propagação da falha T3 (**Atrito excessivo nos mancais das rodas**) na roda de reação do eixo X, em t = 30 segundos.

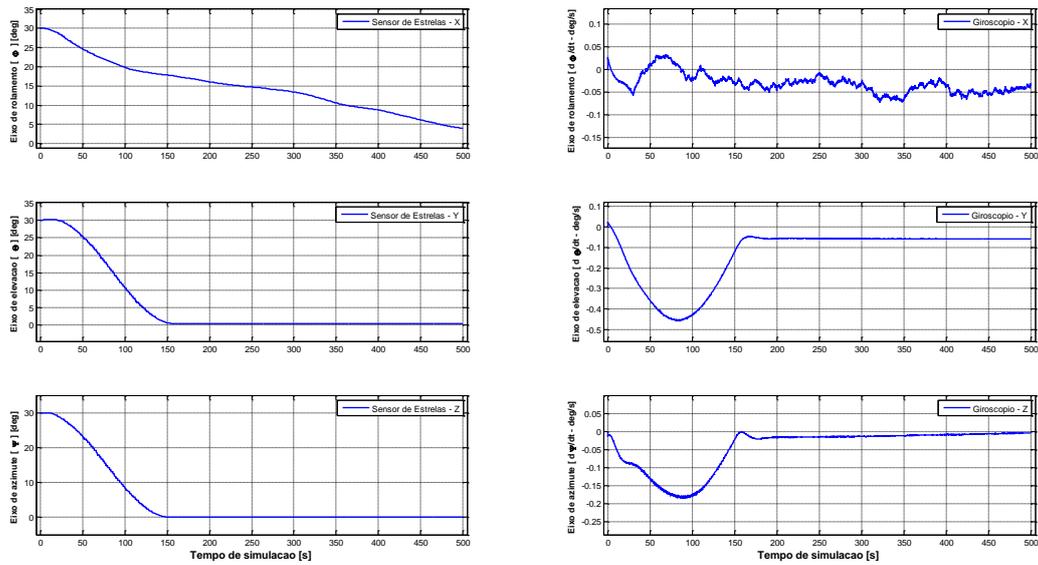


Figura E.15. Propagação da falha T3, eixo X, t = 30 segundos.

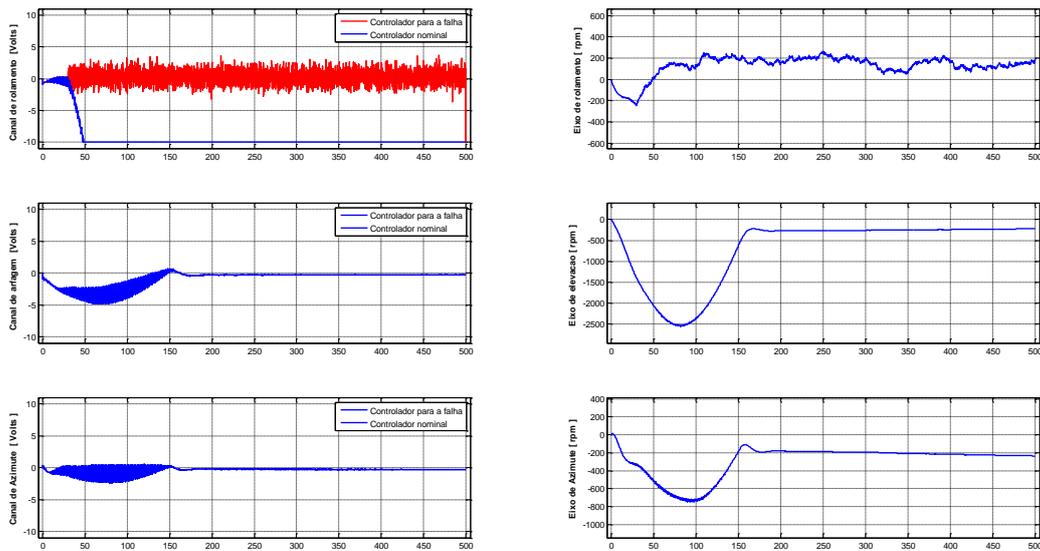


Figura E.16. Propagação da falha T3, eixo X, t = 30 segundos.

A Figura E.17 e a Figura E.18 mostram a propagação da falha T4 na roda de reação (Disparo de comando em regime permanente) do eixo X, em t = 30 segundos (para comparação com os demais casos de falha acima) e, particularmente, a Figura E.19 e Figura E.20 mostram a propagação desta falha na roda de reação do eixo X, em t = 250 segundos (em acordo com o postulado da definição da falha T4).

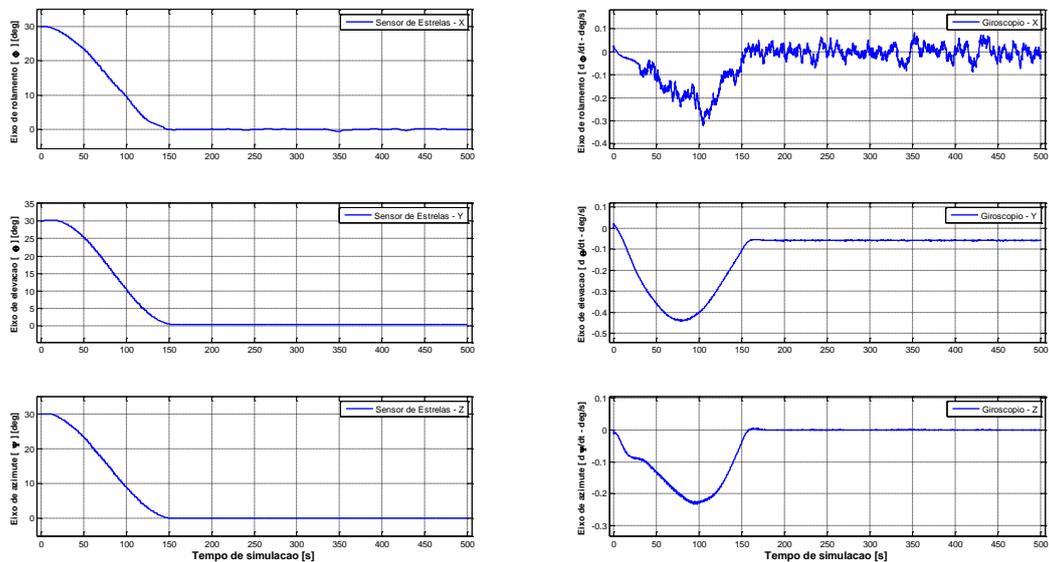


Figura E.17. Propagação da falha T4 eixo X, t = 30 segundos.

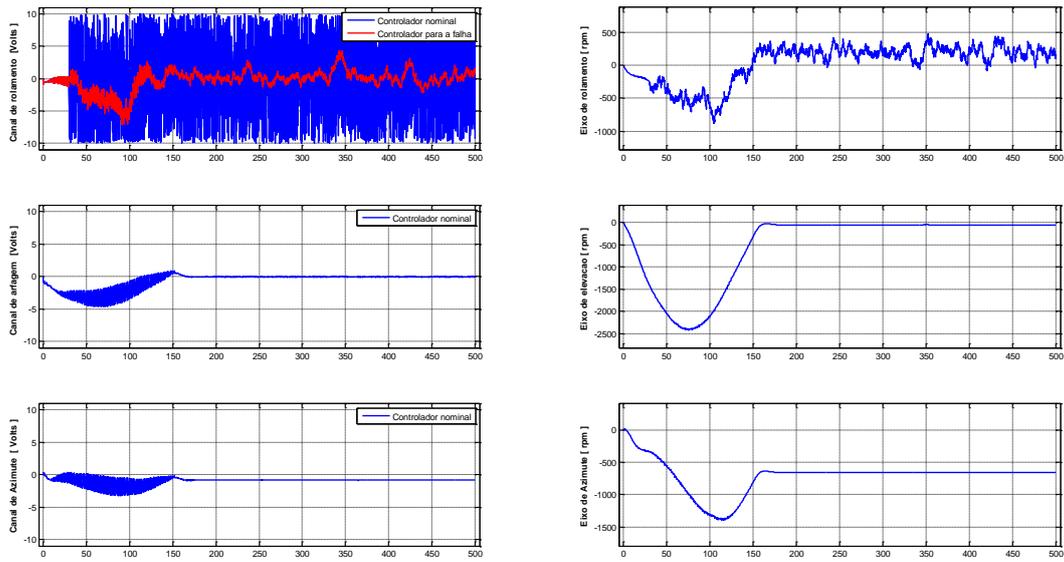


Figura E.18. Propagação da falha T4, eixo X, t = 30 segundos.

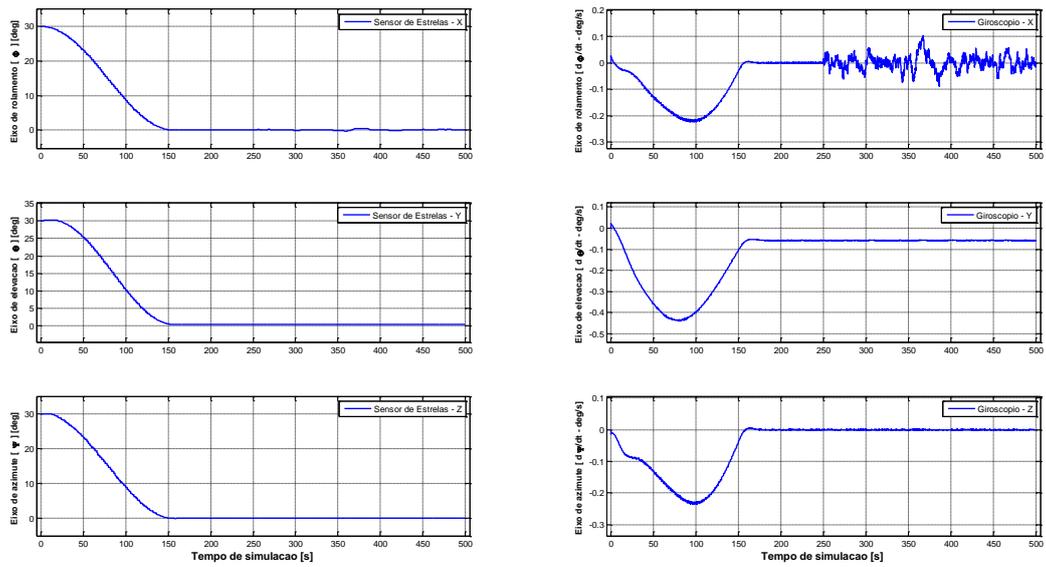


Figura E.19. Propagação da falha T4, eixo X, t = 250 segundos

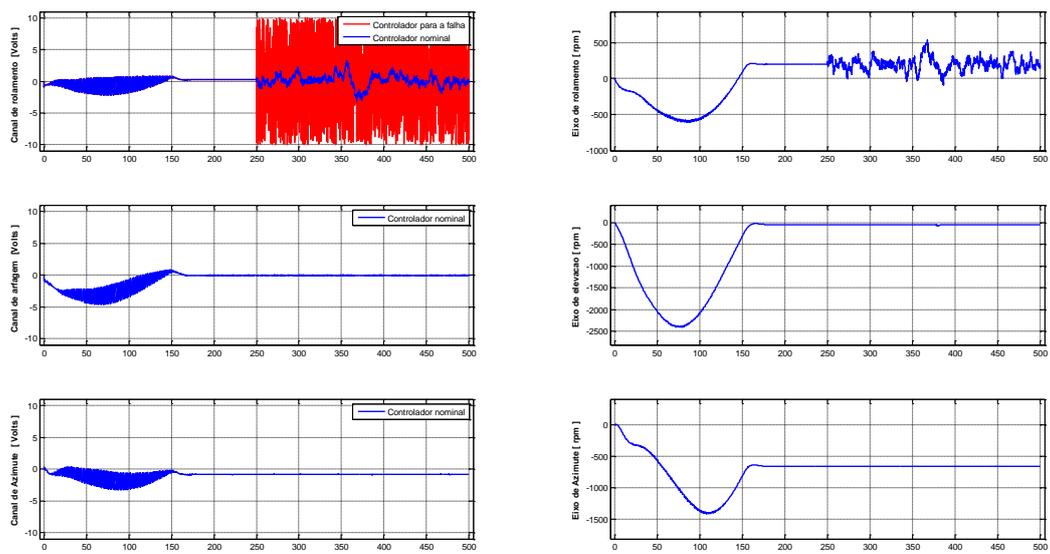


Figura E.20. Propagação da falha T4, eixo X, t = 250 segundos.