

METODOLOGIA PARA VALIDAÇÃO DE SINAL USANDO MODELOS
EMPÍRICOS COM TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL APLICADA A
UM REATOR NUCLEAR

Mauro Vitor de Oliveira

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS
EM ENGENHARIA NUCLEAR.

Aprovada por:

Prof. Roberto Schirru, D.Sc.

Prof. Cláudio Márcio do Nascimento Abreu Pereira, D.Sc.

Prof. Luis Alfredo Vidal de Carvalho, D.Sc.

Dr. Antônio César Ferreira Guimarães, D.Sc.

Dr. Paulo Victor Rodrigues de Carvalho, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

OUTUBRO DE 2005

OLIVEIRA, MAURO VITOR DE

Metodologia para Validação de Sinal
Usando Modelos Empíricos com Técnicas de
Inteligência Artificial Aplicada a um Reator
Nuclear [Rio de Janeiro] 2005

XV, 307 p. 29,7 cm (COPPE/UFRJ, D.Sc.,
Engenharia Nuclear, 2005)

Tese - Universidade Federal do Rio de
Janeiro, COPPE

1. Inteligência Artificial

I. COPPE/UFRJ II. Título (série)

DEDICATÓRIA

Ao meu pai

Washington Victor de Oliveira (In memorian)

À minha mãe

Dorcas Themoteo de Oliveira

À minha esposa

Shirley da Cruz

AGRADECIMENTOS

Ao meu orientador Prof. Roberto Schirru que muito colaborou com idéias, discussões sobre o tema de tese bem como ao apoio e incentivo dados que muito me estimularam para a realização deste trabalho.

Ao amigo José Carlos Soares de Almeida que muito colaborou com discussões durante o desenvolvimento da tese e que teve o carinho e a paciência de ler e fazer um cuidadoso trabalho de revisão do texto.

À todos os amigos que me incentivaram para realização deste trabalho de tese.

À minha esposa pelo estímulo e compreensão.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

METODOLOGIA PARA VALIDAÇÃO DE SINAL USANDO MODELOS
EMPÍRICOS COM TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL APLICADA A
UM REATOR NUCLEAR

Mauro Vitor de Oliveira

Outubro/2005

Orientador: Roberto Schirru

Programa: Engenharia Nuclear

Este trabalho desenvolve uma metodologia para validação de sinal através de modelos empíricos de, com técnicas de inteligência artificial, para realizar a redundância analítica de sinais monitorados em uma planta industrial. A metodologia proposta reúne de forma sistêmica todas as etapas que devem compor um modelo de validação de sinal. Para validar a metodologia, são apresentados, segundo a metodologia, diversos trabalhos realizados em validação de sinal. Além disso, baseado na metodologia proposta é implementado um modelo de validação de sinal. A seleção das etapas de composição deste modelo é feita com base nos resultados obtidos dos ensaios realizados em diferentes técnicas de implementação de cada etapa da metodologia. Ao final do trabalho, o modelo de validação de sinal implementado é ensaiado com sinais monitorados da planta de potência nuclear Angra I, durante uma de suas partidas até atingir 100% de potência. Foram selecionados para validação os sinais da potência nuclear e da temperatura da água na perna quente do circuito 1 do primário do reator. Os resultados obtidos mostraram que o modelo proposto apresenta desempenho eficiente na validação de sinal.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

METHODOLOGY FOR SIGNAL VALIDATION BASED ON EMPIRICALS
MODELS USING ARTIFICIAL INTELLIGENCE TECHNIQUES APPLIED TO
SENSOR MONITORING IN A NUCLEAR POWER PLANT

Mauro Vitor de Oliveira

October/2005

Advisor: Roberto Schirru

Department: Nuclear Engineering

This work describes a methodology for signal validation systems, based on empirical models using artificial intelligence techniques, for analytical redundancy of sensor in an industrial plant. The propose methodology encompass, in a systematic way, all the necessary steps for constructing a signal validation system. Several previously works in signal validation wore set under this methodology for testing its consistence. Based on the methodology, a nuclear power plant signal validation system was implemented. For each module of the signal validation system different implementation techniques were tested and compared to select which one would be applied to. Finally, the role system was tested with signals acquired from Angra I Nuclear Power Plant, from its start to 100% of power. The signals used for testing were the nuclear power and the water hot leg temperature of the reactor loop 1. The obtained results showed that the proposed model has efficient performance in signal validation applications.

ÍNDICE

1. Introdução	1
1.1 Histórico	3
1.2 Métodos de validação de sinal	4
1.3 O canal de instrumentação nuclear	7
1.4 Validação de sinal utilizando técnicas de inteligência artificial	10
1.5 Objetivos/ inovações do trabalho de tese	15
1.6 Contribuições/aplicações do trabalho de tese	16
1.6.1 Metodologia para validação de sinal para modelos empíricos	16
1.6.2 Calibração de instrumentos	16
1.6.3 Aumento da disponibilidade da planta	17
1.6.4 Aumento da confiança do operador	18
1.6.5 Validação dos parâmetros de segurança da planta	18
1.6.6 Sistemas tolerantes à falhas	18
1.7 Seqüência do trabalho de tese	19
2. Processamento de sinal em modelos empíricos	21
2.1 Etapas do processo de validação de sinal	21
2.2 O pré-processamento dos sinais	23
2.2.1 Seleção dos sinais de entrada	23
2.2.1.1 Seleção baseada na correlação física	24
2.2.1.2 Seleção baseada no senso de engenharia	24
2.2.1.3 Seleção baseada na proximidade ao sensor monitorado	24
2.2.1.4 Seleção usando algoritmos genéticos	25

2.2.1.5 Seleção usando sinais redundantes	25
2.2.2 Normalização dos sinais de entrada	25
2.2.3 Filtragem dos sinais de entrada	26
2.2.3.1 Filtragem por média móvel	26
2.2.3.2 Filtragem por média ponderada	27
2.2.3.3 Filtragem por filtro de Kalman	28
2.2.3.4 Filtragem por ondaletas (<i>wavelets</i>)	28
2.2.4 Redução da dimensão do espaço de entrada	30
2.2.4.1 Redução da dimensão do espaço de entrada por PCA	31
2.2.4.2 Redução da dimensão do espaço de entrada por ondaletas	33
2.2.4.3 Redução da dimensão do espaço de entrada por redes neurais	35
2.2.5 Métricas utilizadas	36
2.2.6 Separação em classes	38
2.2.6.1 Métodos de separação em classes simples	39
2.2.6.1.1 Método de separação em classes simples por centróide	40
2.2.6.1.2 Método de separação em classes simples usando redes neurais	41
2.2.6.2 Métodos de separação em classes nebulosas	43
2.2.6.3 Método de separação em classes possibilísticas	44
2.2.6.4 Método de separação em classes nebulosas estendidas	46
2.2.6.5 Otimização do número de classes	49
2.3 O processamento dos sinais	54
2.3.1 Seleção do modelo	54
2.3.2 Seleção do tipo de estrutura	55
2.3.3 Seleção da estrutura	56

2.3.4 Seleção do método de otimização	58
2.4 O pós-processamento dos sinais	58
2.4.1 Métodos de detecção da anomalia de um sensor	58
2.4.1.1 Método do erro (resíduos)	59
2.4.1.2 Método da auto-correlação dos resíduos	59
2.4.1.3 Método da densidade de potência dos resíduos	62
2.4.1.4 Método do teste da razão de probabilidades seqüencial dos resíduos	64
2.4.2 A forma de detecção da anomalia de um sensor	68
2.4.3 A confiança da resposta	69
3. Avaliação das etapas de um modelo de validação de sinal	71
3.1 Seleção dos sinais de entrada para o modelo de validação	71
3.1.1 Seleção dos sinais de entrada para o modelo de validação - 1ª metodologia	72
3.1.2 Seleção dos sinais de entrada para o modelo de validação - 2ª metodologia	76
3.1.3 Seleção dos sinais de entrada para o modelo de validação - aplicação	81
3.1.3.1 Aplicação - 1ª metodologia	82
3.1.3.2 Aplicação - 2ª metodologia	86
3.2 Filtragem dos sinais de entrada	88
3.2.1 Tempo de resposta dos filtros	89
3.2.2 Resposta dos filtros a um sinal tipo rampa	90
3.2.3 Filtragem de sinais de uma planta nuclear	94
3.3 Redução da dimensão do espaço de entrada	94
3.3.1 Redução da dimensão do espaço de entrada usando PCA	96
3.3.2 Falha em um dos sinais de entrada do PCA - caso 1	98
3.3.3 Falha em um dos sinais de entrada do PCA - caso 2	99

3.4 Separação do espaço de entrada em classes	102
3.4.1 Influência das métricas Euclidiana e Mahalanobis	104
3.4.1.1 Conjunto de dados em forma de borboleta	104
3.4.1.2 Conjunto de dados com formas diversas	106
3.4.2 Determinação do número de classes	107
3.4.2.1 Comparação do EFCM com o FCM usando critérios de validação de classes	108
3.4.2.2 Influência da inicialização	110
3.4.2.3 Detecção de linhas	111
3.4.2.4 Conjunto de dados da flor IRIS	112
3.5 Modelo de validação de sinal	114
3.5.1 ANFIS aplicado a uma planta química	116
3.6 Detecção de anomalia de sensor	125
3.6.1 Ensaios do método do erro (resíduos)	126
3.6.2 Ensaios do método de teste da razão de probabilidades seqüencial (SPRT)	128
3.6.3 Ensaios do método da auto-correlação dos resíduos	130
3.6.4 Combinando os métodos de detecção de anomalia	132
3.6.4.1 Decisão por lógica de votação majoritária	132
3.6.4.2 Decisão por lógica de votação tipo OU	133
3.7 Confiança da resposta do sistema	138
3.7.1 Comparação dos algoritmos FCM e PCM	138
3.7.2 Ensaio do modelo de inferência nebulosa	141
3.7.3 Ensaio do modelo de confiança da resposta	142
4. O modelo de validação de sinal implementado	145
4.1 Classificação de alguns modelos de validação segundo a metodologia proposta	145

4.1.1 Aplicação 1 - Modelo de validação de sinal utilizando ANNs	145
4.1.2 Aplicação 2 - Modelo de validação de sinal utilizando ANFIS	147
4.1.3 Aplicação 3 - Modelo de validação de sinal utilizando ANNs	149
4.1.4 Aplicação 4 - Modelo de validação de sinal utilizando ANFIS	152
4.2 O modelo de validação de sinal proposto	154
4.3 Descrição detalhada do modelo implementado	157
4.3.1 Seleção dos sinais de entrada do modelo	157
4.3.2 Normalização dos sinais de entrada do modelo	157
4.3.3 Métrica utilizada	158
4.3.4 Filtragem dos sinais de entrada do modelo	158
4.3.5 Separação em classes dos sinais de entrada do modelo	160
4.3.6 Modelo de validação de sinal	162
4.3.7 Detecção de anomalia	164
4.3.8 Confiança da resposta	167
4.4 Treinamento do sistema de validação de sinal	172
4.4.1 Etapa de pré-processamento	172
4.4.2 Etapa de processamento	173
4.4.2.1 Construindo o ANFIS inicial	174
4.4.2.2 Treinando o ANFIS	175
4.4.3 Etapa de pós-processamento	177
4.5 Operação do sistema de validação de sinal	177
5 Aplicação do modelo de validação de sinal a um reator nuclear	181
5.1 As variáveis monitoradas da planta nuclear de potência	181
5.2 O treinamento do modelo de validação de sinal	189

5.2.1 Treinando o sistema para monitorar a potência nuclear da planta	192
5.2.2 Treinando o sistema para monitorar a temperatura da água na perna quente	196
5.3 A operação do modelo de validação de sinal	200
5.3.1 Monitoração da potência nuclear do reator	200
5.3.1.1 Reator operando sem falha no sinal monitorado - Ensaio 1	200
5.3.1.2 Falhas no sinal monitorado - Operação em estado estacionário	201
5.3.1.2.1 Falha do tipo polarização - Ensaio 2	202
5.3.1.2.2 Falha do tipo desvio positivo - Ensaio 3	203
5.3.1.2.3 Falha do tipo colado - Ensaio 4	205
5.3.1.2.4 Resumo da detecção de falhas pelo sistema - Operação em estado estacionário	206
5.3.1.3 Falhas no sinal monitorado - Operação em subida de potência	207
5.3.1.3.1 Falha do tipo polarização negativa - Ensaio 5	207
5.3.1.3.2 Falha do tipo desvio negativo - Ensaio 6	209
5.3.1.3.3 Falha do tipo colado - Ensaio 7	210
5.3.1.3.4 Resumo da detecção de falhas pelo sistema - Operação em subida de potência	212
5.3.2 Monitoração da temperatura da água na perna quente	213
5.3.2.1 Reator operando sem falha no sinal monitorado - Ensaio 8	213
5.3.3 Monitoração da potência nuclear do reator fora do intervalo de treinamento - Ensaio 9	215
6 Conclusões e comentários	219
6.1 Avaliação dos ensaios das sub-etapas de composição de um modelo	219
6.2 Avaliação dos ensaios do sistema de validação de sinal aplicado a planta nuclear	223
6.2.1 Avaliação do sistema na validação do sinal de potência nuclear	223
6.2.2 Avaliação do sistema na validação da temperatura da água na perna quente	226
6.2.3 Avaliação do sistema na validação da potência nuclear fora do intervalo de treinamento	227

6.2.4 Avaliação dos tempos de resposta do sistema para as falhas inseridas	228
6.3 Contribuições/ inovações do trabalho de tese	229
6.4 Recomendações para trabalhos futuros	230
Referências bibliográficas	231
I. Ondaletas	239
I.1 Transformada por ondaletas contínuas	242
I.1.1 Escalonamento	243
I.1.2 Deslocamento	245
I.1.3 Criando uma transformada por ondaleta contínua	245
I.2 Transformada por ondaletas discretas	247
I.2.1 Filtro de um estágio: aproximação e detalhes	248
I.2.2 Decomposição multi-nível	250
I.3 Reconstrução da ondaleta	252
I.3.1 Filtros de reconstrução	253
I.3.2 Reconstruindo aproximações e detalhes	254
I.3.3 Decomposição e reconstrução multietapas	256
I.4 Tipos de ondaletas	256
I.4.1 Ondaleta Haar	256
I.4.2 Ondaleta Daubechies	257
I.4.3 Ondaleta Biortogonal	257
I.4.4 Ondaleta Coiflets	258
I.4.5 Ondaleta Symlets	259
I.4.6 Outras formas de ondaletas	259
II Análise das Componentes Principais	260

III Técnicas de otimização	269
III.1 Otimização por mínimos quadrados	269
III.1.1 Mínimos quadrados ajustado por uma única variável	270
III.1.2 Mínimos quadrados ajustado por uma reta	272
III.2 Otimização pelo gradiente descendente	274
III.3 Otimização por algoritmos genéticos	276
III.4 Otimização por enxame de partículas	277
IV Lógica nebulosa	281
IV.1 Definições	284
IV.2 Operações	286
IV.3 Representações	288
IV.4 Propriedades	288
IV.5 Relações nebulosas	289
IV.6 Composição de relações nebulosas	291
IV.7 Regras nebulosas multivariáveis	293
V Algoritmos de separação em classes	298
V.1 Algoritmos de separação em classes rígidas	298
V.1.1 O Algoritmo KMEAN de MacQueen	298
V.1.2 O algoritmo por quantização vetorial	299
V.2 Método de separação em classes nebulosas	299
V.2.1 Algoritmo de média-C nebulosa	299
V.3 Método de separação em classes possibilísticas	300
V.3.1 O Algoritmo de média-C possibilístico	300
V.4 Método de separação em classes nebulosas estendidas	301

V.4.1 Método de separação em classes nebulosas estendidas para a métrica Euclidiana .	301
V.4.1.1 Algoritmo de separação em classes nebulosas estendidas para a métrica Euclidiana	302
V.4.2 Método de separação em classes nebulosas estendidas para a métrica Mahalanobis	304
V.4.2.1 Algoritmo de separação em classes nebulosas estendidas para a métrica Mahalanobis	305

CAPÍTULO 1

Introdução

Para operar e controlar uma planta industrial são utilizados sinais de sensores provenientes de várias localizações da planta. O controle “ótimo” e a operação segura de um sistema dinâmico e complexo como uma planta nuclear de potência é dependente da validade dos sinais dos sensores que fornecem ao operador informações sobre o estado do processo. O operador da planta geralmente interpreta os dados baseado na sua experiência e conhecimento do sistema e decide se os dados mostram indícios de falha(s) de instrumento(s) ou se não é necessário efetuar correção nas informações fornecidas pelos instrumentos. Esta decisão pode ser crucial durante uma emergência quando o operador tem que lidar simultaneamente com uma grande quantidade de sinais ou durante uma medição crítica do estado da planta da qual a estabilidade do sistema é fortemente dependente.

Mesmo em operação normal quando o operador deve manter ou levar a planta para o ponto de operação desejado, a monitoração correta dos valores das variáveis é de suma importância para que suas ações baseadas nesses valores, mantenham ou levem a planta para o estado desejado.

Validação de sinal pode ser definida como a detecção, isolamento e caracterização de sinais falhos. A validação de sinal também é referida como a identificação, em tempo real, de falhas no processo de medida e a subsequente produção da melhor estimativa para o valor da variável que está sendo monitorada.

O objetivo da validação de sinal é, então, propiciar ao operador da planta confiança nas medidas do processo que a ele são apresentadas e, deste modo, diminuir sua carga de trabalho cognitivo principalmente durante uma condição transiente ou de emergência.

Os benefícios da validação de sinal são de ordem econômica e de segurança. Uma falha em um sinal de segurança de uma planta pode resultar no desligamento da instalação e, conseqüentemente, perda de receita. Além disso, as ações a serem realizadas pelo operador dependem primariamente das informações fornecidas pelos indicadores da planta, a detecção da falha na instrumentação antes que o operador tome uma decisão baseada numa leitura errônea de um instrumento pode evitar um desligamento não desejado da planta e, conseqüentemente, aumentar sua disponibilidade.

A validação de sinal terá, então, um efeito psicológico no aumento da confiança das ações a serem tomadas pelo operador (NUREG/CR-6127, 1994).

Além disso, após o acidente de *Three Mile Island Unit-2* (TMI-2), a Comissão de Regulamentação Nuclear (NRC - *Nuclear Regulatory Commission*) recomendou (NUREG-0585, 1979) que fossem definidos e adequadamente apresentados, na sala de controle dos reatores nucleares de potência, um conjunto mínimo de parâmetros da instalação que mostrassem a condição de operação da mesma. Esses sistemas (CAIN *et al.*, 1980, NUCLEAR NEWS, 1982) foram chamados de sistemas de apresentação dos parâmetros de segurança (SPDS - *Safety Parameter Display System*). Como esses sistemas utilizam informações provenientes de diversos instrumentos da instalação, a questão da validação de sensor e de sinal tornou-se de extrema relevância na área nuclear.

Neste capítulo é apresentado um pequeno histórico sobre a evolução, os métodos e as aplicações realizadas em validação de sinal nas últimas décadas. Ao final do capítulo são descritos os principais objetivos desta tese que são o desenvolvimento de uma metodologia para construção de sistemas de validação de sinal baseados em modelos

dinâmicos empíricos e a implementação segundo esta metodologia de um sistema de validação de sinal, com aplicação prática a um reator nuclear de potência.

1.1 Histórico

Diversas pesquisas na área de validação de sinal foram realizadas nas últimas décadas. A maioria dos trabalhos originaram-se na indústria aeroespacial (CLARK *et al.*, 1975, CLARK *et al.*, 1976), em aplicações para pilotos automáticos, tráfego aéreo de aviões e naves espaciais. As aplicações iniciais na indústria nuclear são oriundas da transferência desta tecnologia, aplicada a um tipo de dinâmica diferente. Atualmente, as áreas que mais desenvolvem aplicações para validação de sinal são as indústrias aeroespacial e nuclear.

As pesquisas iniciais em validação de sinal se concentraram no método mais simples de validação, que é o uso de sinais redundantes para uma dada variável de processo, para se poder verificar a consistência entre esses sinais. A redundância de sensores é freqüentemente usada para validação de sinal em partes críticas do sistema de controle da planta de modo a garantir o grau de segurança necessário à instalação. A utilização da metodologia de sensor redundante aumenta o custo total do sistema.

De maneira a diminuir o custo causados pela adição de redundância de sensor e, ao mesmo tempo, garantir os requisitos de confiabilidade e precisão das medidas dos sensores foram propostos diversos sistemas baseados em redundância analítica.

Redundância analítica é a estimativa das variáveis de processo através de modelos físicos usando equações de balanço de massa, momento e energia. Esta redundância substitui o uso de sensores redundantes e a lógica de votação majoritária entre esses sensores.

As metodologias de estado da arte para validação de sinal usando redundância analítica variam desde técnicas emergentes baseadas em inteligência artificial (AI - *Artificial Intelligence*) até métodos de espaço-paridade. Os métodos baseados em AI incluem redes neurais artificiais (ANN - *Artificial Neural Network*) (HAYKIN, 1994), sistemas de inferência nebulosa (FIS - *Fuzzy Inference System*) (KOSKO, 1992) e sistemas especialistas (RUSSEL *et al.*, 2003).

1.2 Métodos de validação de sinal

Os métodos disponíveis para realizar o diagnóstico de falha em sensores podem ser divididos em estáticos e dinâmicos. A figura 1.1 apresenta um resumo dos métodos estáticos e dinâmicos utilizados em validação de sinal.

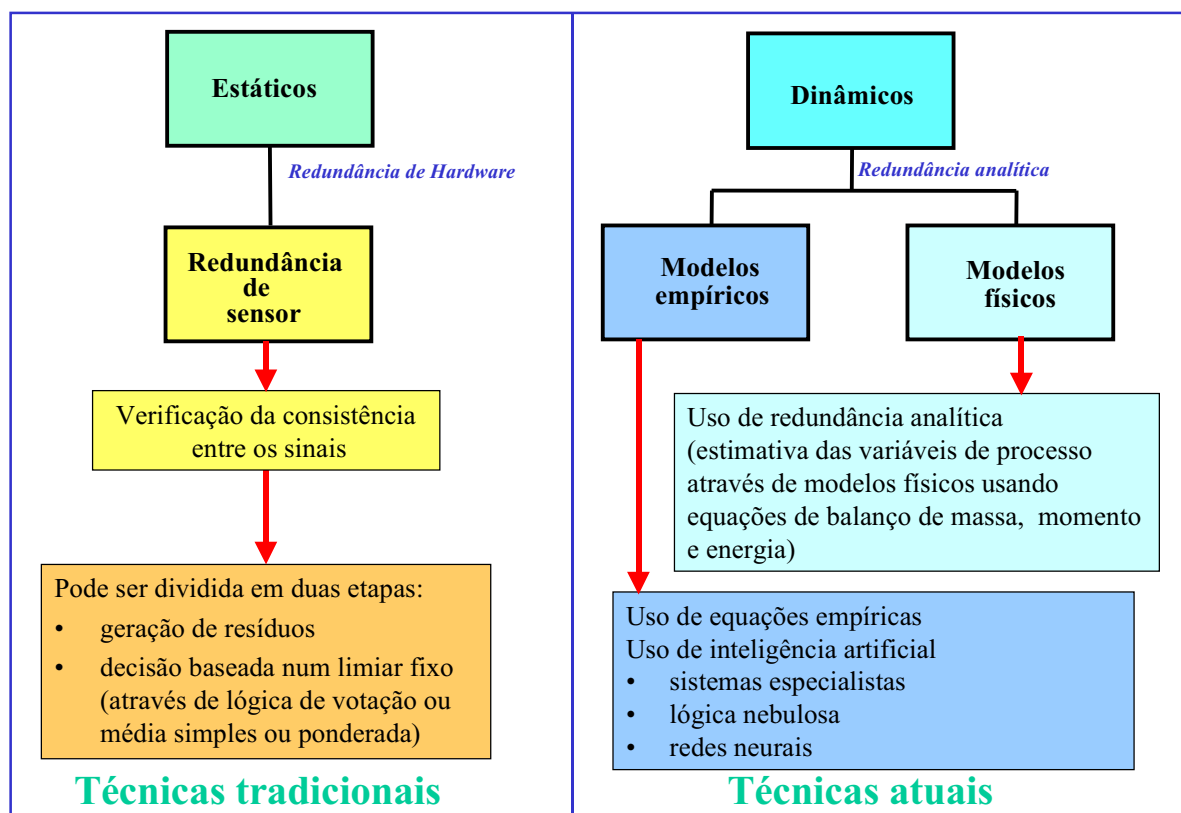


Figura 1.1 - Métodos de validação de sinal.

Os métodos estáticos são aqueles em que, para se determinar o desempenho de um sensor, não são utilizadas correlações com a dinâmica do sistema. Esses métodos incluem a comparação do comportamento entre sensores (redundância de sensor) e a verificação da ultrapassagem de um limite.

Na técnica tradicional de redundância de sensor os sinais dos sensores são comparados para checagem da consistência de suas respostas e através de uma lógica de votação majoritária, ou de uma lógica de média simples, ou ponderada é obtido o sinal validado.

Esta técnica pode ser dividida em duas etapas: a geração de resíduos e a decisão baseada na ultrapassagem de um limiar fixo.

Existem várias técnicas de redundância de sensor entre as quais podemos citar: espaço-paridade (MEIJER *et al.*, 1981), consistência generalizada (UPADHYAYA *et al.*, 1989), reconhecimento de padrões (SACKETT, 1988) e comparação em hipercubos (HOLBERT, 1991).

A técnica espaço-paridade é um dos métodos que pode ser utilizado para geração de resíduos a fim de verificar a consistência entre sinais redundantes e determinar os fatores de ponderação para cada sinal baseados na consistência entre esses sinais.

Assim como o método espaço-paridade, o método de checagem de consistência generalizado (GCC - *Generalized Consistency Checking*) é usado para acompanhar sinais redundantes e identificar inconsistências. O GCC emprega um contador de inconsistência para registrar o número de vezes que o sinal encontra-se inconsistente. Após excluir os sinais com os maiores índices de inconsistências, a cada momento é computada a melhor estimativa do processo como uma média ponderada dos sinais restantes. Este procedimento é apresentado na figura 1.2. Três sinais são comparados em cada instante dois a dois para determinar se suas diferenças caem dentro de um

critério pré-determinado de aceitação. Se o critério de aceitação é satisfeito então os dois sinais são ditos consistentes. Caso contrário, os sinais são inconsistentes e devem ser comparados com outros sinais para se determinar qual deles é inconsistente.

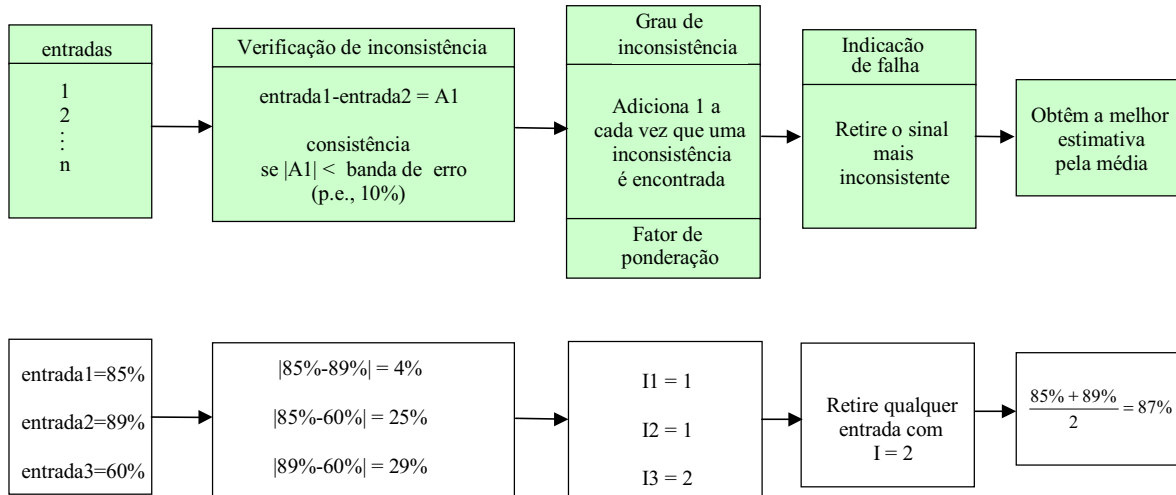


Figura 1.2 - Ilustração da técnica de checagem de consistência generalizado.

Quando não se dispõe de um número suficiente de sinais redundantes para se realizar uma intercomparação, a técnica de redundância por *hardware* é dita inadequada e, então, redundância dinâmica é utilizada para gerar os sinais redundantes.

Os métodos dinâmicos utilizam para validação do sensor a análise do sinal gerado pelo sensor em conjunto com modelos físicos ou empíricos da dinâmica do sistema.

Em sistemas de malhas fechadas como uma planta nuclear de potência, muitos parâmetros do processo são relacionados uns com outros. Neste caso, os modelos dinâmicos físicos obtêm a estimativa da variável de processo a ser validada através do uso de equações analíticas de massa, momento e energia que correlacionam a variável a ser validada com outras variáveis do processo.

Nos modelos dinâmicos empíricos são utilizadas equações empíricas, baseadas na experiência dos especialistas, que descrevem o comportamento do sensor na planta.

Ambas as técnicas, analítica e empírica, se baseiam na aquisição de sinais de um conjunto de sensores, a princípio correlacionados com o sensor a ser validado, que fornecem sinais diferentes do sinal monitorado. Como por exemplo, a obtenção da medida da vazão de um líquido em um tubo utilizando os sinais da temperatura do líquido em diferentes pontos do tubo, fornecidos por termopares. A validação do sinal do sensor é feita com a comparação entre a estimativa da variável do processo e a medida fornecida pela instrumentação.

Atualmente, as pesquisas em validação de sinal têm sido focadas em modelos empíricos, através da utilização de técnicas de inteligência artificial, com aplicações de sistemas especialistas (NING *et al.*, 1992), lógica nebulosa (HOLBERT, 1994) e redes neurais (UPADHYAYA *et al.*, 1992).

1.3 O canal de instrumentação nuclear

Um canal de instrumentação nuclear consiste de um sensor, que fica localizado no campo, um conversor de sinal, um condicionador de sinal e circuitos lógicos e de desligamento (*trip*) que se encontram, na maioria das plantas de potência, nos bastidores de instrumentação da sala de controle ou nos bastidores de instrumentação da sala de distribuição de cabos. A figura 1.3 apresenta os componentes típicos de um canal de instrumentação nuclear relacionado à segurança de uma planta de potência. O sensor pode ser um transmissor de pressão, nível ou vazão, um sensor de temperatura à resistência (RTD - *Resistance Temperature Detector*), para medida da temperatura da perna quente ou fria do refrigerante do reator, um termopar à saída do núcleo, um detector de nêutrons, etc. O equipamento de conversão de sinal pode ser um conversor corrente-tensão, um conversor tensão-corrente, uma ponte resistiva, etc. O equipamento de condicionamento de sinal pode ser um amplificador com filtros atraso/avanço, um

amplificador somador, um isolador, um comparador, etc. O sistema de controle/desligamento (*trip*) pode atuar os sistemas de segurança da planta, provocar o desligamento (*trip*) da planta ou ser utilizado para controle do processo executado pela planta. O sinal de controle/desligamento (*trip*) fará atuar uma válvula, uma bomba, um disjuntor, etc.

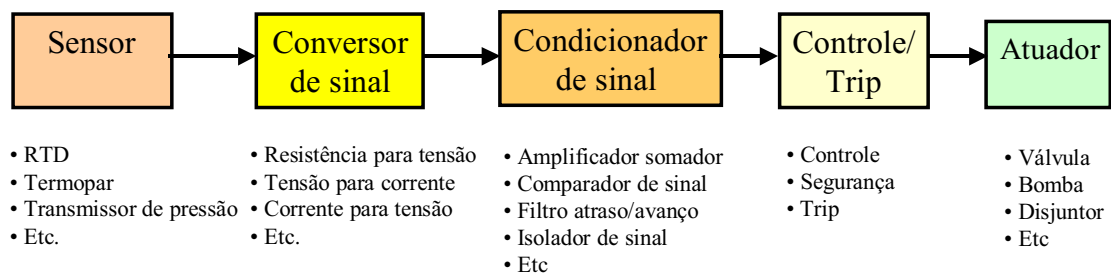


Figura 1.3 - Componentes típicos de um canal de instrumentação nuclear relacionado à segurança.

Na maioria das plantas de potência nucleares os sistemas relacionados à segurança possuem até quatro canais de instrumentação redundantes para cada medição considerada crítica. Neste caso, a consistência dos sinais pode ser verificada através do uso de redundância de sensor associada com lógica de votação majoritária ou de uma lógica de média simples ou ponderada.

A NUREG/CR-6343 (1995) mostra que até um sistema com quatro canais redundantes sem sensor de referência (sensor calibrado periodicamente) tem dificuldade de determinar se um *drift* é gerado pelos sensores ou pelo processo, quando submetidos a um *drift* de modo comum. Todos os sensores estariam variando juntos na mesma taxa resultando em curvas de desvios que somente indicam que os sensores concordam uns com outros, mas não existe informação para detectar qualquer falha de modo comum. A adição de um canal de referência ou a utilização de um dos quatro canais como canal de

referência fornece a capacidade ao sistema de distinguir se é um *drift* do processo ou dos sensores.

Adicionalmente, para sinais que não possuem redundância de *hardware* torna-se impossível a aplicação de redundância estática e, então, faz-se necessário utilizar redundância dinâmica.

A NUREG/CR-6343 conceitua um sistema para validação de sinal que combina os diversos métodos de redundância, a saber:

1. Média dos canais redundantes;
2. Modelagem analítica do processo;
3. Uso de um canal de referência calibrado.

Estas opções podem ser usadas individualmente ou juntas para se obter a melhor estimativa de um parâmetro do processo. A figura 1.4 apresenta o diagrama deste sistema.

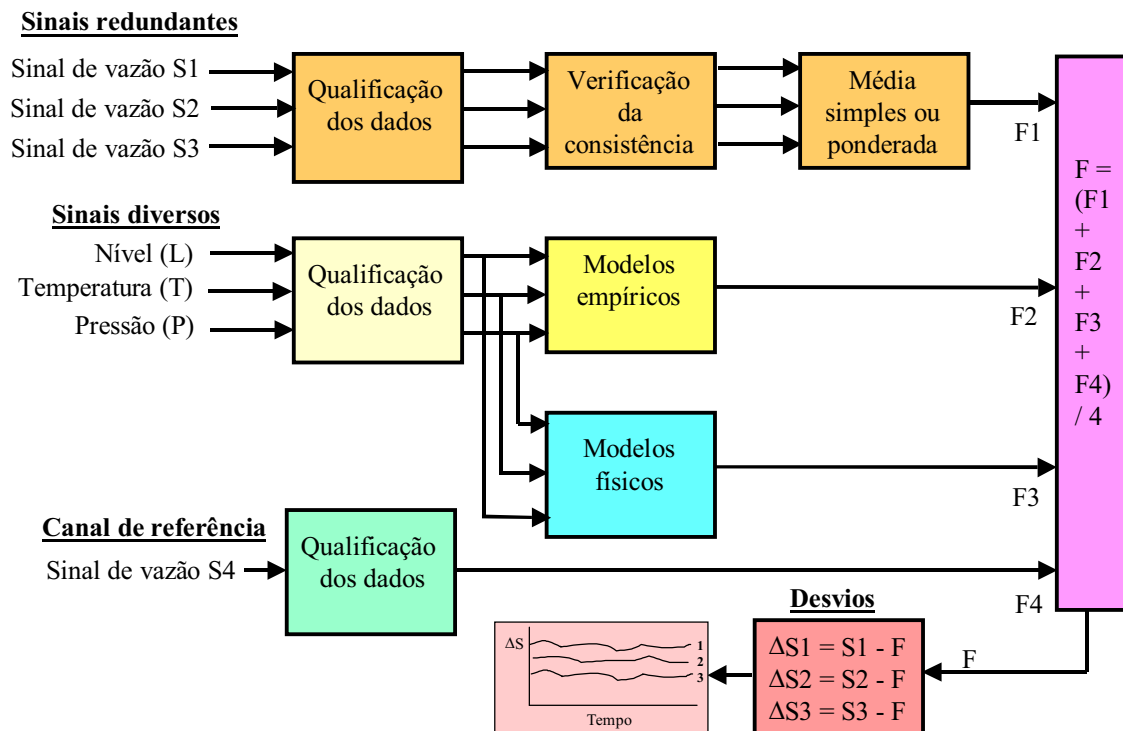


Figura 1.4 - Sistema de validação de sinal proposto pela NUREG/CR-6343.

Caso sejam disponíveis vários canais redundantes, então, uma média simples das leituras redundantes pode fornecer a estimativa do processo necessária para verificar a validade dos canais de instrumentação. Este é o caso de sensores RTDs de medição da temperatura do refrigerante do primário e termopares localizados à saída do núcleo, em reatores à água pressurizados (PWR - *Pressurized Water Reactor*).

Caso o número de canais redundantes seja limitado, a média simples das saídas dos instrumentos podem não fornecer uma estimativa razoável do processo. Deste modo, em adição à média, um canal de referência calibrado ou técnicas analíticas devem ser usadas para verificar a validade dos canais de instrumentação. Este é o caso da maioria dos transmissores de pressão de plantas nucleares.

1.4 Validação de sinal utilizando técnicas de inteligência artificial

A partir de meados do século XX a automação em plantas industriais vem aumentando dia a dia. Com o aumento da automação as ações dos operadores das plantas evoluíram de tarefas simples (manuais) para tarefas mais complexas (cognitivas), conforme mostrado na figura 1.5.

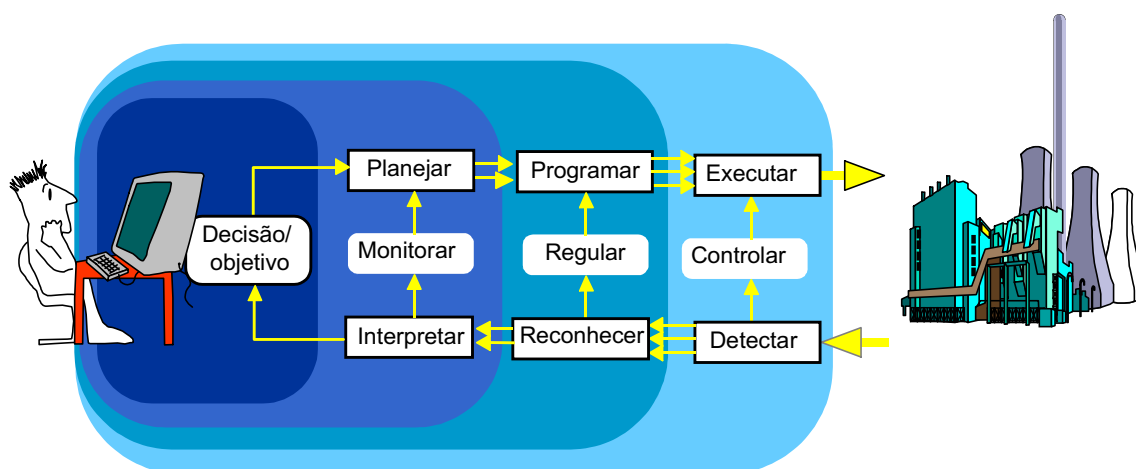


Figura 1.5 - Evolução das ações dos operadores com o aumento da automação.

As tarefas executadas pelos operadores nas primeiras plantas industriais eram somente de controlar o processo. Normalmente, o operador lia as variáveis do processo em medidores, localizados juntos ao processo a ser monitorado, e executava uma ação para controlar a planta. Com o aumento da automação e dos dispositivos para controle e monitoração remotos, cada vez mais o operador foi se afastando do processo monitorado. Atualmente, o operador controla a planta remotamente, recebendo informações sobre o processo monitorado através de sistemas supervisórios e de controle. Este afastamento físico, do operador com o processo monitorado, intensificou ainda mais a importância da validação dos sinais aos quais o operador se utiliza para controlar e regular o processo.

A evolução da tecnologia da informação e da informática permitiu a implementação de sistemas de validação de sinal baseados em redundância analítica para substituir ou corroborar a validação de sinal através de redundância de sensor.

Com o aumento da capacidade de processamento de dados e velocidade de execução de programas dos computadores, principalmente a partir das duas últimas décadas do século 20, houve uma grande evolução na aplicação de técnicas de inteligência artificial para realização de validação de sinal.

Uma das primeiras aplicações de técnicas computacionais em validação de sinal na área nuclear foi realizada no reator regenerador experimental EBR-II (*Experimental Breeder Reactor*), localizado em Idaho, nos Estados Unidos, por SACKETT (1988), para fornecer as estimativas do nível de potência, da temperatura da água à saída do reator, etc., utilizando a técnica de reconhecimento de padrões. Após definir as variáveis de interesse, o próximo passo aplicado foi selecionar os períodos de operação do reator típicos para as condições de interesse da planta. Esses períodos de operação foram

usados para “ensinar” o sistema o que esperar de cada sinal envolvido. Por exemplo, para estimar o nível de potência do reator foram usados 129 sinais para aprender os padrões necessários. Durante a operação do sistema, se quaisquer dos sinais relacionados exibisse um comportamento não usual em relação aos outros, o seu comportamento era registrado e estimado pelo sistema.

HOLBERT *et al.* (1990) aplicaram a técnica de comparação por hipercubos para validação de sinais em um reator à água pressurizada PWR comercial e ao reator regenerador experimental EBR-II.

UPADHYAYA *et al.* (1992) aplicaram redes neurais artificiais (ANNs) ao reator EBR-II com o propósito de validação de sinal. O modelo implementado utilizou duas redes neurais, uma para o circuito primário e outra para o circuito secundário do reator nuclear, para monitoração de algumas variáveis da instalação. Neste modelo foram utilizadas redes neurais auto-associativas, onde as variáveis de entrada e de saída da rede são as mesmas, do tipo *backpropagation*, com três camadas.

HOLBERT *et al.* (1994) aplicaram lógica nebulosa para validação de sinais redundantes. Nesta aplicação, a técnica de lógica nebulosa tinha objetivo de eliminar a fronteira rígida, dos sistemas baseados em ultrapassagem de limiares, entre declarar ou não um sinal falho. Isto é importante desde que as flutuações estatísticas dos sinais do processo levam ocasionalmente à grandes desvios entre as leituras dos sinais, que, deste modo, necessitam de uma larga banda de tolerância para prevenir falsos alarmes, mas que ao mesmo tempo aumentam a probabilidade de perda de alarmes. A utilização de lógica nebulosa na região de transição entre o sinal estar válido ou falho ajuda a prevenção de falhas, isto é, fornece um alerta ao operador de um sinal suspeito antes que ele tenha alcançado o estado falhado, como é requerido, por exemplo, pelo algoritmo espaço-paridade.

FANTONI *et al.* (1996) implementaram um modelo de validação de sinal utilizando ANNs que divide toda a região de operação da planta em diversas sub-regiões, utilizando o método nebuloso de separação em classes, e utiliza uma ANN específica para cada uma dessas sub-regiões. Cada ANN, deste modo, é chamada a operar somente em uma região bem específica da planta. Neste modelo foram utilizadas redes auto-associativas do tipo *backpropagation* com três ou mais camadas. O modelo foi testado com dados gerados por um simulador para um reator do tipo PWR e as redes utilizadas nesta aplicação operavam tanto para as variáveis do primário quanto para as do secundário da reator.

HINES *et al.* (1997) implementaram um modelo de validação de sinal utilizando um sistema de inferência nebulosa adaptativo (ANFIS - *Adaptative Neuro-Fuzzy Inference System*) (FLTbx, 2000). Este sistema foi demonstrado usando dados fornecidos pela planta de potência nuclear de Crystal River unidade 3, localizada no estado da Flórida nos Estados Unidos. As variáveis monitoradas pelo sistema foram o nível do gerador de vapor do *loop A*, a temperatura da perna quente do *loop A* e a vazão de água do *loop A*.

NABESHIMA *et al.* (1998) desenvolveram uma aplicação utilizando uma rede neural auto-associativa para monitorar um conjunto de 12 variáveis importantes da planta nuclear de potência Borselle, localizada na Holanda. Esta aplicação, utilizava uma rede neural adaptativa, isto é, a rede neural era treinada antes e durante sua operação como sistema de validação dos sinais da planta. O treinamento em operação da rede tinha por objetivo capacitá-la a detectar novas regiões de operação da planta que não haviam sido treinadas, como por exemplo, variações nas condições de operações da planta devido a queima do combustível nuclear ao longo do processo de operação da planta.

OLIVEIRA (1999) implementou dois modelos de validação de sinal utilizando ANNs para monitorar um conjunto de 12 variáveis do primário e do secundário da planta nuclear de potência Angra I, localizada no sudeste do Brasil. Um dos modelos utilizava uma única rede neural auto-associativa que era treinada para cobrir todo o espaço de operação da planta nuclear. O outro modelo, que também usava redes neurais auto-associativas, dividia toda a região de operação da planta em diversas sub-regiões, através do método nebuloso de separação em classes e treinava uma ANN específica para cada uma dessas sub-regiões. Deste modo, cada ANN era chamada a operar somente em uma região bem específica da planta. Nestes dois modelos foram utilizadas redes auto-associativas do tipo *backpropagation* com três camadas.

YEN *et al.* (2001) desenvolveram um sistema para validação de sinal baseado em redes neurais autoassociativas. Este modelo emprega um algoritmo de separação em classes baseado em lógica nebulosa que divide o espaço de operação do sistema numa série de sub-espacos e treina redes neurais para operar cada uma em uma dessas regiões do espaço. Neste modelo a separação do espaço em sub-espacos é feita através de um algoritmo que inicialmente supõe todos os dados de treinamento do modelo como pertencentes a uma só classe (modelo com uma única ANN). Através da comparação do sinal de erro, entre o valor estimado pela ANN e o valor medido do sinal monitorado, e um valor de erro máximo desejado para o sistema são adicionadas novas ANNs ao sistema até que esta condição seja satisfeita. Este sistema foi demonstrado com dois conjuntos de dados. O primeiro conjunto de dados consiste de um simulador de falhas de maquinaria *Spectra Quest (SQ Machinery Fault Simulator)* (<http://www.spectraquest.com>). Foram usadas como entrada para o sistema três sensores (acelerômetros) próximos ao sensor monitorado. O segundo conjunto de dados foi uma série temporal de vibração conhecida como conjunto de dados de vibração *Westland*

(CAMERON, 1993). Foram utilizados como entrada para o sistema um arranjo de oito sensores (acelerômetros) fixados em localizações específicas de um helicóptero CH-46 da marinha dos Estados Unidos, ligados, com e sem falhas, à um conjunto de transmissores de potência.

NA *et al.* (2002) desenvolveram um sistema de validação de sinal baseado em um sistema de inferência nebuloso adaptativo que foi utilizado para validar os sinais de nível e pressão do pressurizador e da temperatura da água na perna quente de um dos *loops* do simulador de planta de potência nuclear. Este simulador está localizado na Coréia do Sul e utiliza o código MARS (JEONG, *et al.*, 1999), que é uma versão unificada do COBRA/TF (NUREG/CR-4166, 1985) e RELAP/MOD3 (NUREG/CR-5535, 1995). Nesta aplicação, é utilizada somente uma rede tipo ANFIS para validar cada uma das variáveis em todo o espaço de treinamento. Sendo o treinamento da parte antecedente do sistema ANFIS realizado por algoritmo genético (GA - *Genetic Algorithm*) (GOLDBERG, 1989), enquanto que a parte conseqüente do ANFIS é treinada usando o método dos mínimos quadrados.

1.5 Objetivos/inoações do trabalho de tese

Esta tese tem como principais objetivos/inoações:

- a) Desenvolver uma metodologia para composição de um sistema de validação de sinal por modelos empíricos usando técnicas de inteligência artificial;
- b) Ensaia diferentes técnicas de implementação de cada uma das etapas de composição de um modelo de validação de sinal construído a partir da metodologia proposta;
- c) Desenvolver métodos para seleção de variáveis de entrada para um modelo de validação de sinal a partir do conjunto de variáveis disponíveis da planta;

- d) Desenvolver um sistema de inferência nebulosa adaptativo (ANFIS) com treinamento por enxame de partículas (PSO - *Particle Swarm Optimization*) na sua parte antecedente;
- e) Construir um modelo de validação de sinal baseado na metodologia proposta e ensaiá-lo com dados de uma planta nuclear de potência.

1.6 Contribuições/aplicações do trabalho de tese

A seguir são descritas as principais contribuições/aplicações do trabalho de tese.

1.6.1 Metodologia para validação de sinal para modelos empíricos

Como visto anteriormente, muitos trabalhos têm sido publicados nos últimos anos sobre validação de sinal, porém, nenhum aborda de maneira sistemática todas as etapas do processo de validação de sinal. Este trabalho de tese se propõe apresentar e discutir todas as etapas envolvidas na construção de sistemas de validação de sinal para modelos empíricos usando técnicas de inteligência artificial.

1.6.2 Calibração de instrumentos

A calibração de instrumentos absorve grande parte da força de trabalho da planta para isolar, calibrar e recolocar o sensor em serviço. O esforço da calibração envolve dois passos:

1. Identificação da necessidade de calibração;
2. Calibração caso seja necessário.

O primeiro passo pode ser automatizado pela monitoração *on-line* do sensor. O sistema proposto pode identificar se um instrumento perdeu sua calibração. Isto permite ao pessoal da planta limitar seu esforço de calibração àqueles instrumentos que necessitam de calibração, ao contrário das práticas atuais, que exigem calibrações periódicas na maioria dos instrumentos, conforme mostrado na figura 1.6.

Apesar da monitoração *on-line* para calibração ser aplicável à maioria dos instrumentos, seu principal objetivo é estender o intervalo de calibração para transmissores de pressão, nível e vazão.

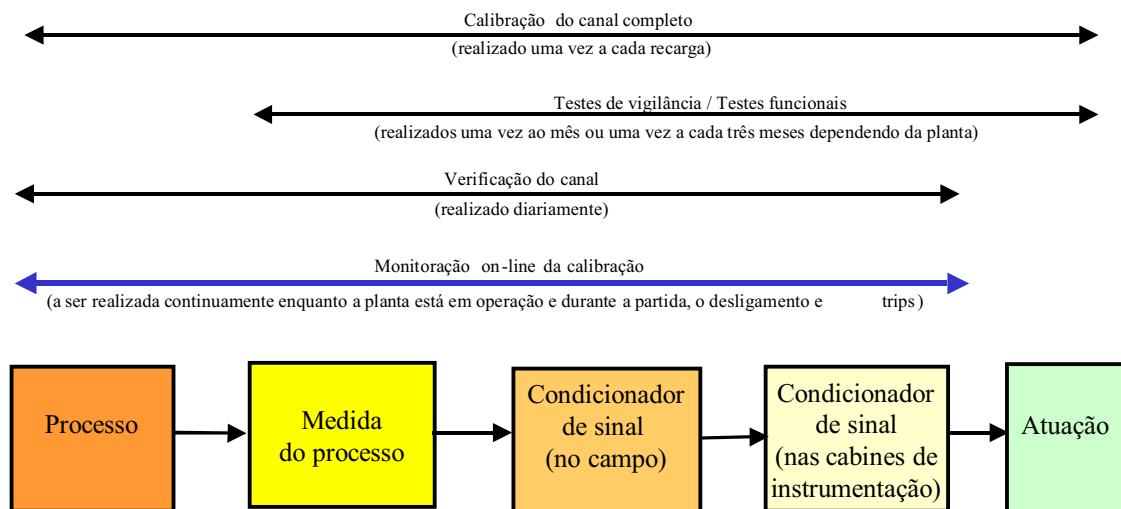


Figura 1.6 - Calibrações realizadas em um canal de instrumentação nuclear.

1.6.3 Aumento da disponibilidade da planta

As ações a serem realizadas pelo operador dependem primariamente das informações fornecidas pelos indicadores da planta. A detecção da falha na instrumentação antes que o operador tome uma decisão baseada numa leitura errônea de um instrumento, que pode resultar no desligamento da planta, irá aumentar sua disponibilidade.

1.6.4 Aumento da confiança do operador

Uma vez que as ações a serem realizadas pelo operador dependem primariamente das informações fornecidas pelos indicadores da planta, a detecção da falha na instrumentação, antes que o operador tome uma decisão baseada numa leitura errônea de um instrumento, terá um efeito psicológico no aumento da confiança do operador nas leituras da planta.

1.6.5 Validação dos parâmetros de segurança da planta

Após o acidente de Three Mile Island Unit-2, a Comissão de Regulamentação Nuclear recomendou que fossem definidos e adequadamente apresentados, na sala de controle dos reatores nucleares de potência, um conjunto mínimo de parâmetros da planta que mostrassem a condição de operação da mesma. Esses sistemas foram chamados de sistemas de apresentação dos parâmetros de segurança. Como esses sistemas utilizam informações provenientes de diversos instrumentos da planta, a questão da validação de sensor e de sinal tornou-se de extrema relevância na área nuclear.

1.6.6 Sistemas tolerantes à falhas

Redundância analítica (modelos empíricos) tem sido usada no desenvolvimento de sistemas tolerantes à falha de tal maneira que a perda de um sinal por uma falha na instrumentação não terá um efeito significativo na operação do processo. O sinal perdido é substituído por um sinal calculado que é obtido de um modelo.

1.7 Seqüência do trabalho de tese

Neste trabalho, inicialmente são apresentadas e discutidas todas as etapas envolvidas na construção de sistemas de validação de sinal para modelos empíricos usando técnicas de inteligência artificial.

A seguir, diferentes técnicas de implementação de cada uma das etapas de composição de um modelo de validação de sinal são ensaiadas com o objetivo de comparação.

Baseado nos ensaios realizados nas etapas de composição de um modelo de validação de sinal é proposto e implementado um sistema de inferência neuro-nebuloso adaptativo para realização de redundância de sensor, com o propósito de validação de sinal.

Para avaliar o desempenho do modelo implementado são utilizados dados reais de um conjunto de variáveis da usina nuclear Angra I, durante uma de suas operações de partida, desde 0% até atingir 100% de potência.

No próximo capítulo, é feita uma descrição das diversas etapas que compõem o processo de validação de sinal. No capítulo 3 são apresentados ensaios realizados nas diversas sub-etapas que compõem cada etapa de um modelo de validação de sinal, com o objetivo de corroborar a escolha do modelo a ser implementado. No capítulo 4, alguns dos modelos citados no capítulo 2 são analisados segundo a metodologia proposta e, com base nesta análise e nos resultados dos ensaios realizados no capítulo 3 em cada etapa que compõe um modelo de validação de sinal, é proposto o modelo implementado na tese. Neste mesmo capítulo são descritas em detalhes as etapas de treinamento e operação do modelo de validação de sinal implementado. O capítulo 5 apresenta a implementação do modelo para a aplicação realizada, onde é feito o treinamento e a operação do sistema de validação de sinal com as amostras do conjunto de dados da

planta. O desempenho do sistema de validação de sinal implementado é avaliado na validação dos sinais de potência nuclear e da temperatura da água na perna quente do circuito 1 do primário da usina nuclear Angra I. Finalmente, o capítulo 6 apresenta as conclusões e comentários sobre o desempenho do modelo e sugestões para trabalhos futuros.

Para auxiliar a compreensão das técnicas de estatística e de inteligência artificial empregadas no decorrer da tese, são apresentados vários anexos: o anexo I apresenta a técnica de filtragem por ondaletas (*wavelets*), o anexo II descreve a análise das componentes principais (PCA - *Principal Components Analysis*), o anexo III apresenta uma descrição resumida das técnicas de otimização por mínimos quadrados, gradiente descendente (GD - *Gradient Descendent*), algoritmos genéticos (GA) e enxame de partículas (PSO), o anexo IV apresenta uma descrição resumida de lógica nebulosa (*fuzzy logic*) e o anexo V apresenta alguns algoritmos de separação em classes rígidas (HCM - *Hard Clustering Means*), nebulosas (FCM - *Fuzzy Clustering Means*), possibilísticas (PCM - *Possibilistic Clustering Means*) e nebulosas estendidas (EFCM - *Extended Fuzzy Clustering Means*).

CAPÍTULO 2

Processamento de sinal em modelos empíricos

Neste capítulo são descritas de maneira geral as diversas etapas que compõem o processo de validação de sinal em modelos empíricos. Algumas das técnicas de inteligência artificial empregadas em cada etapa do processo de validação de sinal, nos modelos discutidos no capítulo 1, são apresentadas com o objetivo de propiciar uma visão geral do estágio de desenvolvimento atual das técnicas de validação de sinal.

2.1 Etapas do processo de validação de sinal

O processo de validação de sinal em modelos empíricos pode ser dividido em três etapas: pré-processamento, processamento propriamente dito e pós-processamento. Cada uma dessas etapas pode ser dividida em sub-etapas, a saber:

- a) Pré-processamento - nesta etapa os sinais que irão compor o sistema de validação de sinal são selecionados, normalizados, filtrados, reduzidos em dimensão e separados em classes segundo uma métrica;
- b) Processamento - nesta etapa define-se o modelo a ser utilizado, o tipo de estrutura do modelo e o método para sua otimização;
- c) Pós-processamento - nesta etapa são definidos os métodos de detecção de falha e de confiança na resposta produzida pelo sistema.

Como já foi citado anteriormente, nos últimos anos foram desenvolvidos diversos sistemas de validação de sinal utilizando modelos empíricos. A tabela 2.1 apresenta um resumo das técnicas de processamento de sinal utilizadas por estes sistemas, classificadas segundo as etapas de processamento de sinal.

A seguir serão detalhadas cada uma das sub-etapas que compõem as três etapas do processo de validação de sinal.

Tabela 2.1
Resumo das técnicas de processamento de sinal utilizadas em modelos empíricos.

Pré-processamento						Processamento				Pós-processamento		
Seleção dos sinais	Normalização dos sinais	Filtragem dos sinais	Redução da dimensão	Métrica	Separação em classes	Modelo	Tipo	Estrutura	Otimização	Detecção da anomalia		Confiança da resposta
										Método	Forma	
-----	Nenhuma	Nenhuma	Nenhuma	Nenhuma	Nenhuma	-----	-----	-----	Nenhuma	-----	-----	Nenhuma
Correlação física	[-1,1]	Digital	PCA	Euclidiana	Centróide (<i>k</i> -means)	Matemático	Auto-associativo	Única	Mínimos quadrados	Erro do sensor (resíduos)	Limiar	FIS
Senso de engenharia	[0,1]	Ondaleta (<i>wavelet</i>)	Ondaleta (<i>wavelet</i>)	Minkowski	AVQ	MLP	Hetero-associativo	Múltipla	Gradiente descendente	Auto-correlação dos resíduos	Lógica de votação	
					Gaussiana	RBF						
Proximidade ao sensor monitorado	[-0.5,0.5]	Filtro Kalman	Rede neural	Mahalanobis	FCM	Sistema especialista			GA	Espectro de potência dos resíduos	Lógica booleana	
Correlação usando GA				Manhattan	PCM	FIS				SPRT dos resíduos	Lógica nebulosa	
Sinais redundantes					FCM Crescente	<i>Neuro-fuzzy</i>				Filtro Kalman dos resíduos		

2.2 O pré-processamento dos sinais

A etapa de pré-processamento de sinal dos modelos empíricos tem por objetivo preparar (pré-processar) os dados de entrada para serem utilizados pelo modelo de validação de sinal. Este pré-processamento deve condicionar os dados de entrada de modo que os mesmos sejam compatíveis com o modelo utilizado. Este condicionamento inclui, por exemplo, a retirada de ruídos espúrios e a normalização dos sinais monitorados.

2.2.1 Seleção dos sinais de entrada

O primeiro passo a ser definido para um sistema de validação de sinal é a seleção dos sinais do processo que serão utilizados como entradas para o modelo. Esta seleção normalmente é feita objetivando-se selecionar as variáveis que possuem correlação com a variável a ser validada.

O número de variáveis a serem aplicadas às entradas de um modelo empírico de validação de sinal deve ser otimizado por várias razões (NA *et al.*, 2005). Primeiro, variáveis de entrada não relevantes podem gerar um modelo de validação instável. Deste modo, é desejável somente usar variáveis de entradas para predição com alto grau de informação (correlação) em relação à variável monitorada. Segundo, caso existam colinearidades entre as variáveis de entrada a capacidade de generalização do modelo pode ser degradada. Deste modo, é desejável remover as variáveis de entrada altamente correlacionadas. Finalmente, quando um modelo tipo caixa-preta, como o modelo empírico por técnicas de inteligência artificial, possui muitas variáveis de entrada, torna-se necessário dispor de um número elevado de observações (amostras contendo pares entradas/saída) para seu treinamento de modo a cobrir completamente o espaço de entrada. O número de observações necessárias para treinamento do modelo cresce

exponencialmente com o número de variáveis de entrada, o que torna a redução da dimensão do espaço de entrada essencial para obtenção de um modelo satisfatório.

Diversas técnicas de seleção de variáveis foram utilizadas em sistemas de validação de sinal. Na tabela 2.1 são listadas as principais técnicas até então utilizadas para este propósito.

2.2.1.1 Seleção baseada na correlação física

A técnica de correlação física entre as variáveis de entrada normalmente está associada à aplicação de um modelo de validação de sinal analítico (matemático). Neste modelo, a variável a ser validada é escrita através de equações que a correlacionam com variáveis do processo monitorado.

2.2.1.2 Seleção baseada no senso de engenharia

A técnica de senso de engenharia busca selecionar as variáveis de entrada para o sistema com base no conhecimento do especialista sobre a dinâmica do processo. Normalmente, são selecionadas variáveis consideradas importantes (correlacionadas) em relação à variável monitorada.

2.2.1.3 Seleção baseada na proximidade ao sensor monitorado

A técnica de proximidade ao sensor monitorado visa buscar variáveis de sensores para entrada do sistema que estejam fisicamente próximos ao sensor monitorado. A idéia por trás deste tipo de seleção de variáveis é supor que sensores fisicamente próximos, mesmo medindo diferentes grandezas, devam ser fortemente correlacionados.

2.2.1.4 Seleção usando algoritmos genéticos

A técnica de correlação usando algoritmos genéticos (GA) se baseia na seleção de variáveis de entrada que estejam fortemente correlacionadas com a variável de saída e pouco correlacionadas umas com as outras, através da utilização de algoritmos genéticos. O GA é utilizado para encontrar um conjunto de variáveis de entrada que minimizam o erro de estimação do sinal de saída do modelo.

2.2.1.5 Seleção usando sinais redundantes

A técnica de sinais redundantes normalmente é utilizada para seleção dos sinais de entrada quando existem sinais redundantes ao sinal a ser monitorado.

2.2.2 Normalização dos sinais de entrada

A normalização das variáveis de entrada a serem aplicadas ao sistema visa garantir que durante a operação do sistema o mesmo não esteja sujeito a valores de variáveis de entrada superiores ou inferiores aos valores que foram utilizados no seu treinamento. Normalmente, os modelos empíricos têm dificuldade de realizar extrapolações provocadas por valores de entrada fora do seu intervalo de treinamento.

A normalização dos sinais é efetuada antes de treinar ou operar o sistema. Deste modo, tanto o processo de treinamento quanto de operação é realizado num hipercubo, que pode ser, por exemplo, $[0,1]^n$, onde n é o número de sinais envolvidos no sistema de validação de sinal. Para este caso a normalização é dada pela seguinte equação

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2.1)$$

onde x , x_{norm} , x_{max} e x_{min} são, respectivamente, o sinal normal, o sinal normalizado, o valor máximo do sinal e o valor mínimo do sinal.

2.2.3 Filtragem dos sinais de entrada

O objetivo de filtrar os sinais utilizados no modelo de validação de sinal é retirar possíveis ruídos captados por estes sinais durante o processo de medida. Esses ruídos podem ser provenientes, por exemplo, de acionamento de bombas, relés, etc., próximas aos cabos de transmissão do sinal do sensor.

A filtragem dos sinais também facilita o processo de treinamento do sistema, uma vez que ela diminui a complexidade dos sinais a serem “aprendidos” pelo sistema.

Como os sinais disponíveis para os sistema de validação de sinal são normalmente fornecidos por computadores da planta, que digitalizam seus sinais analógicos e binários, faz-se necessária, então, a utilização de filtros digitais com o propósito de filtragem dos sinais.

2.2.3.1 Filtragem por média móvel

A técnica de filtragem digital de sinal mais conhecida é a da média móvel, na qual é criada uma janela móvel composta de $n-1$ amostras passadas e a amostra atual. A cada nova amostragem do sinal é calculado o valor médio das n amostras dentro da janela e o valor obtido é considerado como valor da amostra neste instante.

Este filtro digital pode ser representado por

$$\bar{x}_k = \frac{\sum_{i=k}^{k-J} x_i}{J} \quad (2.2)$$

onde \bar{x}_k é o valor médio da variável da amostra atual k obtido pelo filtro, x_i é o valor da variável na amostra i e J é o tamanho da janela móvel.

A escolha do tamanho da janela móvel, J , está diretamente correlacionada com o tempo de resposta a transientes do filtro. Quanto maior for o tamanho da janela maior

será o tempo de resposta do filtro. Por outro lado, quanto menor for o tamanho da janela menor será o tempo de resposta do filtro, porém, neste caso se diminuirmos muito o tamanho da janela o processo de filtragem se reduzirá.

2.2.3.2 Filtragem por média ponderada

A técnica de filtragem digital por média ponderada utiliza uma ponderação entre os valores da amostra atual e das amostras anteriores. Um trabalho proposto (ALLEY *et al.*, 1988) sugeriu a utilização de um filtro digital de média ponderada dada pela seguinte equação

$$\bar{x}_k = \frac{x + (P-1)\bar{x}_{k-1}}{P} \quad (2.3)$$

onde \bar{x}_k é o valor médio da variável da amostra atual k obtido pelo filtro, x é o valor da variável na amostra atual, \bar{x}_{k-1} é o valor médio da variável na amostra anterior $k-1$ e P é o peso de ponderação entre o valor da variável na amostra atual e o valor médio da variável na amostra anterior. O valor do peso P é inicializado em 1 e trocado em potência de 2, a cada amostra nova apresentada, até atingir o seu valor máximo, definido pelo usuário (por exemplo, de 8).

A escolha do peso de ponderação P está diretamente correlacionada com o tempo de resposta a transientes do filtro. Quanto maior for o valor do peso maior será o tempo de resposta do filtro. Por outro lado, quanto menor for o valor do peso menor será o tempo de resposta do filtro, porém, neste caso, se diminuirmos muito o valor do peso de ponderação o processo de filtragem se reduzirá.

2.2.3.3 Filtragem por filtro de Kalman

A técnica de filtragem com filtro Kalman foi desenvolvida por KALMAN (1960), para resolver problemas lineares por filtragem de dados discretos (digitais). O filtro de Kalman é um conjunto de equações matemáticas que fornecem solução computacional (recursiva) de problemas pelo método de mínimos quadrados. A filtragem de Kalman tem vários aspectos positivos: ela suporta estimação do passado, presente e até do estado futuro.

2.2.3.4 Filtragem por ondaletas (*wavelets*)

A técnica de filtragem por ondaleta (*wavelet*) (DAUBECHIES, 1992) é uma evolução na técnica de análise de Fourier. Na análise de Fourier um sinal é transformado do domínio do tempo para o domínio da frequência, isto é, o sinal é representado por suas componentes senoidais de diferentes frequências. A transformada de Fourier tem uma grande deficiência. Na transformação do domínio do tempo para o da frequência a informação temporal é perdida. Quando olhamos a transformada de Fourier de um sinal é impossível dizer quando ocorre um determinado evento.

Se as propriedades do sinal não mudam muito ao longo do tempo, isto é, se o sinal é do tipo estacionário, esta deficiência não é muito importante. Contudo, a maioria dos sinais de interesse possuem numerosas características não estacionária ou transitória: *drifts*, tendências, mudanças abruptas de começo e fim de eventos. Essas características são frequentemente as partes mais importantes do sinal, e a análise de Fourier não é capaz de detectá-las.

A análise por ondaleta é uma técnica de janela com regiões de tamanhos variados. Ela permite usar intervalos de tempo longos (regiões grandes), quando se deseja

informação mais precisa de baixa frequência e intervalos de tempo curtos (regiões pequenas), quando se deseja informação mais precisa de alta frequência.

As ondaletas têm sido aplicadas com sucesso em filtragem de sinal onde se dispõe de uma grande quantidade de dados (elevada quantidade de amostras). Isto significa que a filtragem por ondaletas pode não ter resultados satisfatórios se o conjunto de dados é pequeno, como em aplicações em tempo real. Sabe-se que o erro da filtragem por ondaletas diminui com o aumento da quantidade de dados.

Deste modo, a utilização de ondaletas do tipo digital (DWT - *Digital Wavelet Transform*) em aplicações que envolvam tempo real, naturalmente, irão requerer uma filtragem de um janela móvel de dados. Mas esta aplicação pode levar à problemas porque as DWTs são variantes periodicamente no tempo, ou seja

$$DWT(x+t) \neq DWT(x) \quad (2.4)$$

As ondaletas do tipo SWT (*Stationary Wavelet Transform*) tentam corrigir este problema e são promissoras para aplicações em tempo real de filtragem de sinal. Neste caso, deve-se considerar uma janela de tempo que contenha n amostras, a amostra atual e $n-1$ amostras passadas do sinal a ser filtrado, para que se possa reduzir o erro de estimação da ondaleta.

Este filtro digital pode ser representado por

$$\bar{x}_k = f((x_k, x_{k-1}, x_{k-2}, \dots, x_{k-J}), L) \quad (2.5)$$

onde \bar{x}_k é o valor médio da variável da amostra atual k obtido pelo filtro e f representa a função ondaleta a ser aplicada nas amostras dentro da janela móvel $x_k, x_{k-1}, \dots, x_{k-J}$ com nível de decomposição L e J é o tamanho da janela móvel, dado em potência de 2, ou seja, $J = 1, 2, 4, 8$, etc.

Novamente, a escolha do tamanho da janela móvel, J , está diretamente correlacionada com o tempo de resposta a transientes do filtro. Quanto maior for o tamanho da janela maior será o tempo de resposta do filtro. Por outro lado, quanto menor for o tamanho da janela menor será o tempo de resposta do filtro, porém, neste caso se diminuirmos muito o tamanho da janela o processo de filtragem se reduzirá.

Uma vantagem da técnica de filtragem por ondaletas é a capacidade da recuperação do sinal original a partir do sinal filtrado pelas ondaletas. Esta técnica é conhecida como filtragem com ondaletas em quadratura. O anexo I apresenta um resumo sobre a técnica por ondaletas. Informações mais detalhadas sobre as ondaletas podem ser encontradas em DAUBECHIES (1992) e MISITI *et al.* (2000).

2.2.4 Redução da dimensão do espaço de entrada

A redução da dimensão do espaço das variáveis de entrada visa minimizar a quantidade de informação necessária para o treinamento do sistema. A figura 2.1 apresenta uma visão geral deste processo para uma matriz $m \times n$ representando m medidas feitas de n variáveis, onde n é a dimensão do conjunto de dados.

A redução da dimensão de um conjunto de dados visa mapear a matriz de dados originais em uma matriz de dimensão muito menor $m \times f$ ($f \ll n$), que é capaz de reproduzir a matriz original com um mínimo de distorção, através do processo de demapeamento. A redução de dimensão é útil quando existem correlações entre as variáveis observadas. A matriz reduzida descreve as variáveis latentes extraídas da matriz original. Idealmente, as variáveis latentes f devem possuir todas as variações não aleatórias das variáveis observadas. Os modelos de mapeamento e o demapeamento devem capturar as relações entre as variáveis observadas e as variáveis latentes.

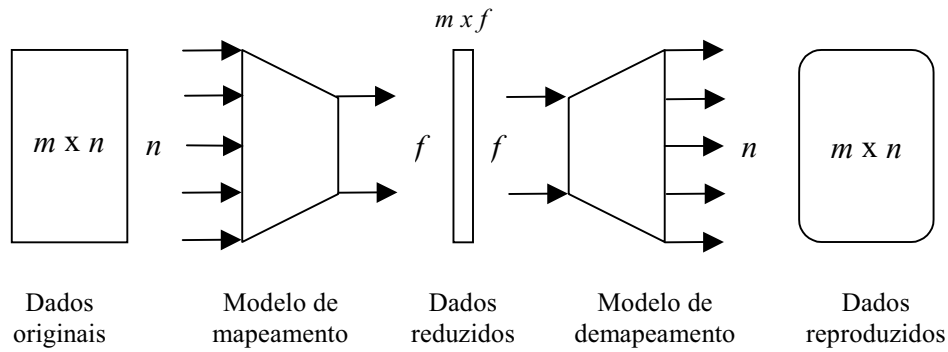


Figura 2.1 - Redução da dimensão de um conjunto de dados.

A redução da dimensionalidade de um conjunto de dados pode ser usada para extrair informações úteis de processos que envolvam uma grande quantidade de variáveis medidas.

Esta redução de dimensão pode ser realizada de diversas maneiras. A seguir são apresentados resumidamente três métodos de redução da dimensão, utilizadas em sistemas de validação de sinal: a redução pela análise das componentes principais, a redução por ondaletas (*wavelets*) e a redução por redes neurais.

2.2.4.1 Redução da dimensão do espaço de entrada por PCA

A análise das componentes principais (PCA) (JACKSON, 1991) é um método de pré-processamento de dados que retira do conjunto de dados as componentes não correlacionadas. O método PCA envolve uma transformação linear do espaço de entrada em um espaço ortogonal que pode ser escolhido para ser de dimensão menor do que o espaço de entrada, com mínima perda de informação.

A decomposição por valor singular (SVD - *Singular Value Decomposition*) é um método computacional eficiente para realizar o PCA. Qualquer matriz A $m \times n$ de grau r pode ser decomposta na seguinte forma

$$A = \vec{u}_1 s_1 \vec{v}_1^T + \vec{u}_2 s_2 \vec{v}_2^T + \dots + \vec{u}_r s_r \vec{v}_r^T \quad (s_1 \geq s_2 \geq \dots \geq s_r > 0) \quad (2.6)$$

onde s_i ($i=1,2,\dots,r$) são escalares positivos em ordem decrescente, \vec{u}_i ($i=1,2,\dots,r$) são vetores ortogonais $m \times 1$ e \vec{v}_i ($i=1,2,\dots,r$) são vetores ortonormais $n \times 1$. Os f primeiros termos da precedente decomposição fornece a melhor aproximação de A com f componentes principais.

O método PCA é realizado pelos seguintes passos:

1. Selecione os dados que serão utilizados no PCA;
2. Subtraia, em cada dimensão dos dados, o valor do dado da média dos valores dos dados na mesma dimensão. Isto produz um conjunto de dados com média zero;
3. Calcule a matriz covariância do conjunto de dados;
4. Calcule os autovetores e os autovalores da matriz covariância;
5. Selecione as componentes a serem utilizadas e construa a matriz de transformação;
6. Obtenha o novo conjunto de dados.

O PCA é uma técnica linear no sentido que ela usa funções lineares para modelar as relações entre as variáveis observadas e as variáveis latentes. As técnicas lineares reduzem a dimensão de um conjunto de dados pela extração das correlações lineares entre as variáveis observadas. Se existirem correlações não lineares entre as variáveis observadas, como ocorre normalmente em um processo nuclear, um método não linear irá descrever os dados com maior precisão e/ou com menos variáveis latentes do que um método linear.

O anexo II apresenta uma aplicação de redução da dimensão de um espaço através da técnica de análise das componentes principais. Informações mais detalhadas dessa técnica pode ser encontrada em JACKSON (1991) e CHEN *et al.* (1999).

2.2.4.2 Redução da dimensão do espaço de entrada por ondaletas

A técnica de redução da dimensão de um conjunto de dados por ondaletas parte das ondaletas mãe e pai para derivar outras funções ondaletas e funções escalas (MISITI *et al.*, 2000). Essas funções constroem uma base ortogonal. Uma função ondaleta $\psi_{b,a}$ e uma função escala $\phi_{b,a}$ são definidas como

$$\psi_{b,a} = \frac{1}{\sqrt{a}} \psi\left(\frac{t-a}{b}\right) \quad (2.7)$$

$$\phi_{b,a} = \frac{1}{\sqrt{a}} \phi\left(\frac{t-a}{b}\right) \quad (2.8)$$

onde ψ é a ondaleta mãe, ϕ é a ondaleta pai, b é o coeficiente de translação e a é o coeficiente escalar.

Se o coeficiente de translação e o coeficiente escalar são dados por (2.9), as ondaletas são discretas e constroem um base ortogonal de acordo com

$$a = 2^{-j}, \quad b = 2^j k \quad (2.9)$$

$$\psi_{j,k} = 2^{j/2} \psi(2^j t - k) \quad (2.10)$$

$$\phi_{j,k} = 2^{j/2} \phi(2^j t - k) \quad (2.11)$$

onde $j, k \in \mathbb{Z}$, \mathbb{Z} é o conjunto dos números inteiros.

Baseado em (2.10) e (2.11) pode-se realizar uma análise multiresolução. Na análise multiresolução, pode-se analisar uma função f em uma dimensão arbitrária, mais especificamente na dimensão diádica, sem perda de informação. Em outras palavras, assume-se f como sendo os coeficientes de aproximação em um espaço de resolução

maior. Então, determina-se os coeficientes de aproximação e detalhe projetando f para espaços de menor resolução como

$$f = \sum_k c_{j,k} \phi_{o,k} + \sum_k \sum_j d_{j,k} \psi_{j,k} \quad (2.12)$$

onde $c_{j,k}$ é o coeficiente de aproximação e $d_{j,k}$ é o coeficiente de detalhe.

O coeficiente de aproximação é definido como (2.13) usando a ondaleta pai e o coeficiente de detalhe é definido como (2.14) usando a ondaleta mãe

$$c_{j,k} = \langle f, \phi_{j,k} \rangle = \int f \cdot \phi_{j,k} \quad (2.13)$$

$$d_{j,k} = \langle f, \psi_{j,k} \rangle = \int f \cdot \psi_{j,k} \quad (2.14)$$

Os coeficientes de aproximação representam a relação entre um espaço maior e um espaço menor, e os coeficientes de detalhe representam a relação entre um espaço maior e o espaço complementar de um espaço menor. Um espaço menor tem uma dimensão reduzida diadicamente quando comparado com um espaço maior de acordo com (2.10) e (2.11).

Pela síntese desses coeficientes, o dado original pode ser reconstruído completamente sem perda de informação. Estes métodos são conhecidos como decomposição e reconstrução por ondaletas.

O anexo I apresenta um resumo sobre a técnica por ondaletas. Informações mais detalhadas sobre as ondaletas pode ser encontrado em DAUBECHIES (1992) e MISITI *et al.* (2000).

2.2.4.3 Redução da dimensão do espaço de entrada por redes neurais

A técnica de redução da dimensão de um conjunto de dados por redes neurais tem sido usada por muitos autores (TAN *et al.*, 1995, WREST *et al.*, 1997). As maiores vantagens da utilização de redes neurais são: (1) elas não precisam saber os mecanismos físicos do sistema a ser mapeado; (2) qualquer sistema linear ou não linear pode ser modelado, e (3) elas são robustas. Em aplicações de redução de dimensionalidade, as redes neurais realizam uma função auto-associativa. Uma função auto-associativa é aquela na qual os sinais dos nós de entrada e de saída da rede são idênticos. Deste modo, a rede neural artificial (ANN) possui uma estrutura simétrica conforme apresentada figura 2.2.

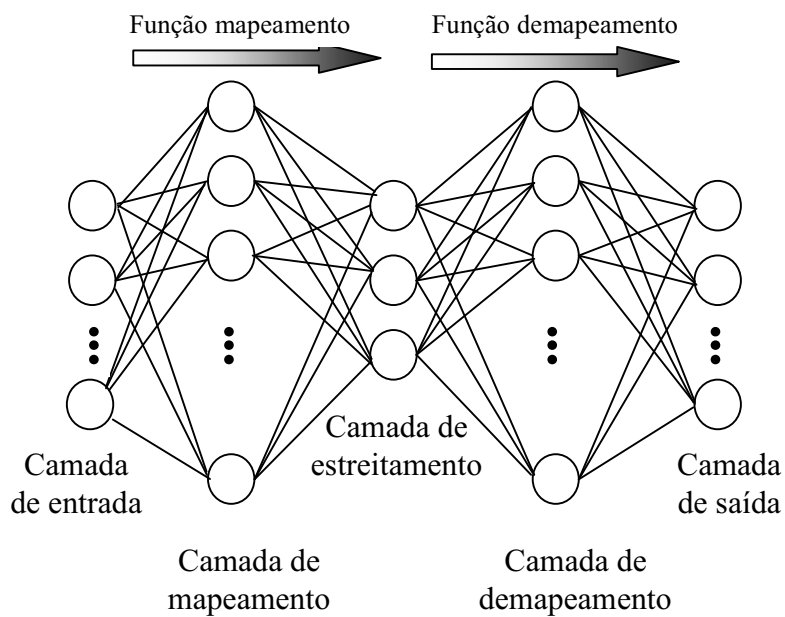


Figura 2.2 - Estrutura de uma ANN.

Geralmente a camada de estreitamento é colocada na rede para que se treine somente as informações intrínsecas dos sinais de entrada. A camada de estreitamento

realiza a mesma função do PCA, onde o seu número de nós é menor do que o número de nós das camadas de entrada e saída.

Informações mais detalhadas sobre a técnica de redução de dimensão utilizando redes neurais pode ser encontrada em KRAMER (1991).

2.2.5 Métricas utilizadas

A escolha da métrica a ser utilizada é fundamental para a realização da separação em classes de um conjunto de dados. Ela deve ser escolhida levando-se em conta a distribuição das amostras dos sinais no espaço de realizações.

A métrica mais utilizada em separação em classes é a distância Euclidiana que é uma função simétrica, isto é, todas as variáveis em um conjunto de dados são consideradas igualmente importantes. Isto na prática não é normalmente verdadeiro, variáveis diferentes são normalmente medidas por instrumentos diferentes, com escalas diferentes, em tempos diferentes e em condições diferentes. Essas medidas não são equivalentes e nesses casos a distância Euclidiana não é apropriada.

Existem diversas métricas que podem ser aplicadas na separação em classes de dados que satisfazem as condições para pertencer ao espaço Euclidiano, como definido a seguir.

Seja \mathcal{R}^n um espaço Euclidiano n -dimensional. Uma função distância em \mathcal{R}^n é uma função $d: \mathcal{R}^n \times \mathcal{R}^n \rightarrow \mathcal{R}$, que para quaisquer $a, b, c \in \mathcal{R}^n$, satisfaz as seguintes condições:

$$\text{a) } d(a,b) \geq 0 ; d(a,b) = 0, \text{ se e somente se } a = b \text{ (condição de contorno);} \quad (2.15)$$

$$\text{b) } d(a,b) = d(b,a) \quad (\text{simetria}); \quad (2.16)$$

$$\text{c) } d(a,b) \leq d(a,c) + d(c,b) \quad (\text{desigualdade triangular}). \quad (2.17)$$

onde dado uma função distância d , o valor $d(a,b)$ é chamado de distância entre a e b .

As funções distância mais utilizadas em separação em classes são apresentadas na tabela 2.2.

Tabela 2.2

Exemplos de funções distância.

Nome	Função
Minkowski	$d_p(a,b) = \left[\sum_{i=1}^S a_i - b_i ^p \right]^{1/p}, p > 0$
Euclidiana	$d_2(a,b) = \left[\sum_{i=1}^S a_i - b_i ^2 \right]^{1/2}$
Hamming	$d_1(a,b) = \sum_{i=1}^S a_i - b_i $
Máximo	$d_\infty(a,b) = \max_{i=1,\dots,S} a_i - b_i $
Mahalanobis	$d_A(a,b) = (a-b)^T A(a-b)$, onde A é uma matriz definida positiva e simétrica

Pode-se notar que a distância Euclidiana e as distâncias d_1 e d_∞ estão incluídas na classe de distância Minkowski. Por outro lado, a distância de Mahalanobis é uma distância completamente diferente. As estruturas encontradas em um conjunto de dados são diferentes para diferentes funções distâncias. Por exemplo, a figura 2.3 apresenta as diferentes formas do círculo unitário para várias funções distâncias da classe Minkowski.

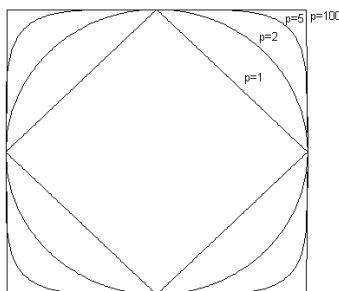


Figura 2.3 - O círculo unitário para várias funções distância da classe Minkowski.

Na distância de Mahalanobis o círculo unitário torna-se uma elipse como mostrado na figura 2.4.

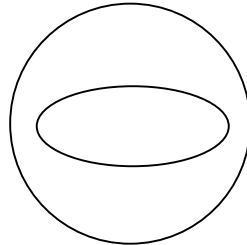


Figura 2.4 - Círculo unitário no espaço Euclidiano e

no espaço Mahalanobis com $A = \begin{bmatrix} 1.5 & 0 \\ 0 & 3 \end{bmatrix}$.

É obvio que usando diferentes funções distância podemos obter resultados muito diferentes na separação em classes de um mesmo conjunto de dados. Deste modo, é necessário selecionar a função distância que mais se adapta a uma dada aplicação.

Os critérios de avaliação de uma separação em classes baseiam-se na idéia de se obter centros de classes cujas funções distância produzam os maiores índices de validade das classes. Dentre os vários critérios para validade das classes utilizados podemos citar: coeficientes da partição, entropia da partição, e compactação e separação.

2.2.6 Separação em classes

A escolha do método de separação em classes é fundamental para o bom desempenho do modelo de validação de sinal.

Separação em classes é um processo que tem por objetivo dividir um conjunto de amostras de entrada em classes que são inerentes a essas amostras. A separação em

classes significa encontrar similaridades nas amostras apresentadas e agrupá-las em diferentes classes segundo estas similaridades.

Os métodos de separação por classes normalmente são divididos em duas famílias de algoritmos: os algoritmos hierárquicos e os algoritmos não hierárquicos.

Os algoritmos hierárquicos são aqueles nos quais se busca uma hierarquia de todas as amostras, a partir da construção de uma árvore de ramos que, em cada iteração, partem da raiz e buscam a fusão das duas classes mais próximas.

Os algoritmos não hierárquicos têm por objetivo classificar as amostras através de uma partição simples em C classes, onde C é o número de classes especificado previamente ou determinado pelo algoritmo.

Como o problema de validação de sinal não necessita encontrar uma hierarquia entre todas as amostras de entrada, mais sim uma partição das amostras em classes, nossa atenção será focada nos algoritmos não hierárquicos.

A seguir são descritos sucintamente três métodos de separação em classes não hierárquicas apresentados na tabela 2.1: a separação simples (através dos métodos do centróide e do AVQ), a nebulosa (FCM) e a possibilística (PCM). Além desses métodos é descrito também o método de separação em classes estendidas (EFCM), proposto nesta tese para ser utilizado no modelo de validação implementado.

2.2.6.1 Métodos de separação em classes simples

No método de separação em classes simples, cada amostra é colocada em somente uma classe. A figura 2.5 apresenta uma representação deste método de separação, onde é feita a separação de N amostras em C classes.

A pertinência μ_{ik} de uma amostra k qualquer em uma classe i é obtida por

$$\mu_{ik} = \begin{cases} 1 & \text{se } d_{ik} < d_{lk}; \quad 1 \leq l \leq C, \quad l \neq i \\ 0 & \text{caso contrario.} \end{cases} \quad (2.18)$$

onde d_{ik} e d_{lk} definem, respectivamente, as distâncias da amostra k ao centróide da classe i e da classe l .

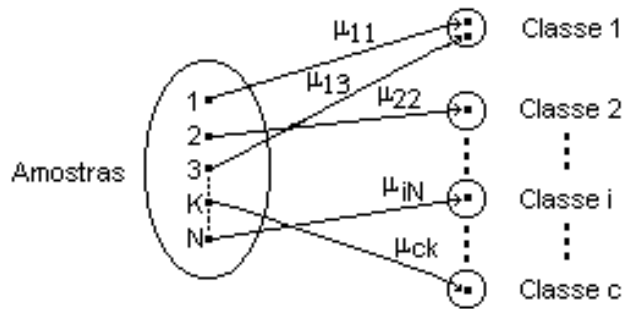


Figura 2.5 - Representação da separação em classes simples.

Os métodos de separação em classes simples exigem, a cada iteração, a alocação total de cada amostra em uma única classe, sendo conhecidos na literatura (KRISHNAPURAM *et al.*, 1993) como métodos rígidos (HCM).

Existem vários métodos não hierárquicos de separação em classes simples entre os quais podemos citar: o método do centróide e métodos realizados por redes neurais.

2.2.6.1.1 Método de separação em classes simples por centróide

O método do centróide pode ser dividido em duas etapas:

1. Etapa de representação: uma classe i é representada pelo seu centróide c_i dado por

$$\mathbf{c}_i = \frac{\sum_{k=1}^N \mu_{ik}^m \mathbf{x}_k}{\sum_{k=1}^N \mu_{ik}^m}; \quad i = 1, \dots, C \quad (2.19)$$

Com $\mathbf{x}_k = (x_{k1}, x_{k2}, \dots, x_{kj})$ (2.20)

onde j é o número de variáveis da amostra, x_{kj} é o valor da j -ésima variável da k -ésima amostra, x_k é a k -ésima amostra e m é o grau de nebulosidade. O valor de m foi feito igual a 2 nesta tese.

2. Etapa de colocação: cada amostra é colocada na classe correspondente ao centróide que esteja mais próximo da amostra.

$$x_k \in c_i \quad \text{se } d_{ik} = \min_{l=1,c} d_{lk} \quad (2.21)$$

O método do centróide alterna iterativamente essas duas etapas até que os centróides converjam para uma configuração estável, ou seja, até que a diferença entre as posições dos centróides em uma iteração com relação a iteração anterior seja menor do que um erro ε estipulado.

2.2.6.1.2 Método de separação em classes simples usando redes neurais

Os métodos de separação em classes simples utilizando redes neurais podem ser realizados por redes neurais do tipo AVQ (*Adaptive Vector Quantization*), RBF (*Radial Base Function*) e Gaussiana, etc.

As redes AVQ, RBF e Gaussiana (KRÖSE, 1993, KOSKO, 1992, HAYKIN, 1994) são redes neurais com lei de aprendizado competitivo (não supervisionado). Uma rede competitiva realiza um processo de separação em classes das amostras de entrada, isto é, as amostras de entrada são divididas em classes separadas tal que as similaridades entre as amostras de entrada na mesma classe são muito maiores do que as similaridades entre entradas em classes diferentes, onde similaridade é medida por uma função distância entre as amostras de entrada.

Por exemplo, para efetuar a separação em classes, escolhemos uma rede por vetor de quantização com lei de aprendizado competitivo não supervisionado (AVQ-UCL - *Adaptive Vector Quantization - Unsupervised Competitive Learning*). A figura 2.6 apresenta a topologia de uma rede com aprendizado AVQ-UCL.

Nesta rede neural, toda a camada i de saída é conectada a todas as entradas j através de pesos w_{ji} . Além disso, todos os neurônios de saída i da rede são conectados a si mesmo por um caminho excitatório, e aos outros neurônios da camada de saída por um caminho inibitório.

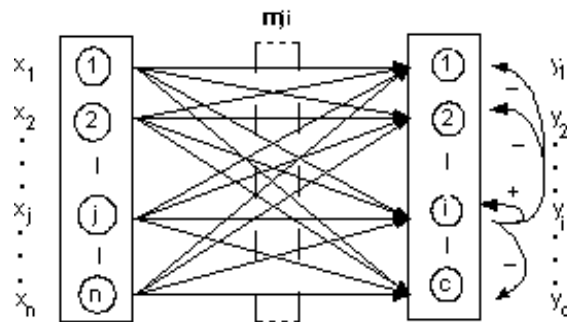


Figura 2.6 - Topologia de uma rede com aprendizado AVQ-UCL.

Cada saída desta rede representa uma determinada classe e o vetor sináptico referente a esta saída representa o centróide desta classe. Do mesmo modo, cada entrada da rede representa uma variável da amostra aplicada. A seleção do neurônio de saída vencedor i , isto é, a classe cuja amostra tem maior afinidade, é feita através da medida da distância da amostra para todos os vetores sinápticos (centróides). O vetor sináptico que estiver mais próximo da amostra de entrada é selecionado, usando como métrica a sua distância à amostra de entrada. A atualização dos pesos para implementar um deslocamento em direção a amostra de entrada é feita pela lei de aprendizado competitivo não supervisionado discreto, sendo dada por:

$$m_i(t+1) = m_i(t) + c_i[x(t) - m_i(t)] \tag{2.22}$$

$$m_j(t+1) = m_j(t) \quad \text{se } j \neq i \tag{2.23}$$

onde C_t é a taxa de aprendizado que diminui ao longo do processo de convergência.

2.2.6.2 Métodos de separação em classes nebulosas

A popularidade da aplicação dos métodos que utilizam conjuntos nebulosos é devida ao fato de que eles são capazes de representar classes ou conceitos vagos através de funções de pertinência definidas sobre o domínio de discussão.

A separação em classes utilizando lógica nebulosa tem mostrado ser vantajosa em comparação com a separação simples, porque ela não exige, a cada iteração, uma alocação total de uma amostra em uma dada classe. A figura 2.7 apresenta a representação da separação em classes nebulosas.

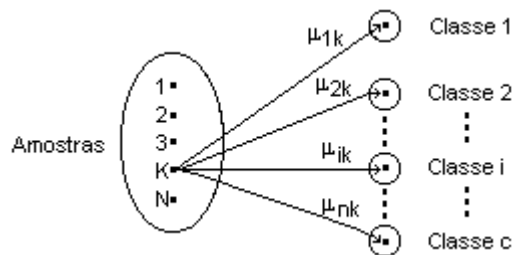


Figura 2.7 - Representação da separação em classes nebulosas e possibilísticas.

Em um classificador nebuloso, a pertinência μ_{ik} devem satisfazer as condições

$$\mu_{ik} \in [0,1] \quad \text{para todo } i \text{ e } k \tag{2.24}$$

$$0 < \sum_{k=1}^N \mu_{ik} < N \quad \text{para todo } i \tag{2.25}$$

$$\sum_{i=1}^C \mu_{ik} = 1 \quad \text{para todo } k \tag{2.26}$$

A maioria das abordagens analíticas de separação em classes utilizando lógica nebulosa são derivadas dos algoritmos de média-C nebulosos (FCM) de Bezdek (ZADEH, 1965). O algoritmo FCM e suas derivações têm sido usados com sucesso em muitas aplicações, tais como, classificação de padrões e segmentação de imagem, no qual o objetivo final é fazer uma decisão rígida (*crisp*). O algoritmo FCM usa a restrição probabilística (2.26), na qual, as pertinências de uma amostra nas classes devem ter por soma o valor 1. Essa restrição vem da generalização da partição- C simples de um conjunto de amostras, e é usada para gerar as equações de atualização iterativa das pertinências para o algoritmo.

Para o classificador nebuloso, a pertinência μ_{ik} de uma amostra k em uma classe i é calculada por

$$\mu_{ik} = \frac{1}{1 + \sum_{l=1}^C \left(\frac{d_{ik}^2}{d_{lk}^2} \right)^{\frac{1}{m-1}}} ; l \neq i \quad (2.27)$$

2.2.6.3 Método de separação em classes possibilísticas

A restrição da pertinência (2.26) usada no algoritmo FCM é feita para evitar a solução trivial, ou seja, todos os valores de pertinência serem iguais a 0. Esta restrição apresenta bons resultados em aplicações onde é apropriado interpretar os valores de pertinência como probabilidades ou graus de compartilhamento. Todavia, como os valores de pertinência gerados por esta restrição são relativos, elas não são apropriadas para aplicações nas quais as mesmas supostamente representam uma “tipicidade”. Entende-se que uma amostra é típica de uma classe quando a mesma pode representar esta classe.

No modelo possibilístico (PCM) (ZADEH, 1978) as funções de pertinência são absolutas (isto é, não relativas) e denotam o grau de pertinência ou “tipicalidade”. Em outras palavras, em tal modelo o valor da pertinência de uma amostra do domínio de discussão em uma classe não depende de seus valores de pertinência nas outras classes definidas sobre o mesmo domínio.

Através de estudos empíricos, ZIMMERMAN *et al.* (1985) mostraram que um bom modelo para as funções de pertinência que modelam conceitos ou classes vagas é dado por

$$\mu(x) = \frac{1}{1 + d_{xx_0}} \quad (2.28)$$

onde d_{xx_0} é a distância de um ponto x no domínio de discussão ao centróide x_0 da classe.

Em outras palavras, nesta formulação o valor da pertinência de um ponto é somente uma função da sua distância ao centróide da classe. O algoritmo FCM e suas derivações não são realmente capazes de gerar tais funções de pertinência, tendo em vista que eles geram pertinências relativas.

Em um classificador possibilístico as pertinências μ_{ik} satisfazem também as condições (2.24) e (2.25), porém, pelo relaxamento, a condição (2.26) torna-se

$$\max_i \mu_{ik} > 0 \quad \text{para todo } k \quad (2.29)$$

Uma função pertinência para o classificador possibilístico foi sugerida por KRISHNAPURAM *et al.* (1993). Nela a pertinência μ_{ik} de uma amostra k em uma classe i é dada por

$$\mu_{ik} = \frac{1}{1 + \left(\frac{d_{ik}^2}{\eta_i} \right)^{m-1}} \quad (2.30)$$

$$\text{onde } \eta_i = \frac{\sum_{K=1}^N \mu_{ik}^m d_{ik}^2}{\sum_{K=1}^N \mu_{ik}^m} \quad (2.31)$$

2.2.6.4 Método de separação em classes nebulosas estendidas

Até agora todos os métodos de separação em classes apresentados consideram os centros das classes (protótipos) como sendo pontos. No método de separação em classes estendidas (EFCM - *Extended Fuzzy Clustering Means*) proposto por KAYMAK *et al.* (2000) os centros das classes de uma partição nebulosa são representados por um volume. A idéia de usar centros das classes (protótipos) com volume em classificação nebulosa foi introduzida por SETNES *et al.* (1998). Os centros das classes (protótipos) com volume estendem os centros das classes (protótipos) de classes de pontos para regiões no espaço a ser separado em classes. A relação dos protótipos com volume com o desempenho do algoritmo de separação em classes foi identificada há muito tempo. Foram desenvolvidas diversas medidas da validade da separação em classes para protótipos com volume (GATH *et al.* 1989, XIE *et al.*, 1991). Outros autores (KELLER *et al.*, 1999) propuseram a adaptação do volume dos protótipos. Recentemente, foi proposto (KRISHNAPURAM *et al.*, 2000) um algoritmo de separação em classes baseado na minimização total do protótipo com volume. Este algoritmo é muito parecido com o proposto por KAYMAK *et al.* (2000).

A principal vantagem de usar protótipos com volume é a redução na sensibilidade resultante do algoritmo de separação em classes à diferentes volumes de classes e a distribuição dos padrões dos dados. Isto torna os algoritmos de separação em classes mais robusto.

Freqüentemente, vários pontos do conjunto de dados perto ao centro de uma classe podem ser considerados como pertencentes completamente a esta classe. Isto é

evidenciado quando existem classes que são bem separadas umas das outras. Neste caso, é interessante estender o núcleo de uma classe de um único ponto para uma região no espaço. Fazendo isto obtemos os protótipos com volume definido a seguir.

Definição: Um protótipo com volume $v \in \mathcal{R}^n$ é um sub-espaco n -dimensional, convexo e compacto de um espaço separado em classes.

Note que de acordo com esta definição o volume do protótipo pode ter forma e tamanho arbitrários. Quando os protótipos originais das classes são pontos é intuitivo selecionar protótipos que estendem uma dada distância em todas as direções. No caso de algoritmos utilizando a métrica Euclidiana, os protótipos das classes com volume \tilde{v}_i são hiperesferas com centros v_i e raio r_i . Similarmente, os protótipos tornam-se hiperelipsóides no caso de algoritmos com métrica Mahalanobis.

O algoritmo de classificação estendida mede a distância das amostras do conjunto de dados para o protótipo com volume. As amostras x_k que caem dentro da hiperesfera, isto é, $d(x_k, v_i) \leq r_i$, são elementos do protótipo com volume \tilde{v}_i e têm por definição o valor de pertinência de 1.0 nesta classe. O tamanho do volume dos protótipos são determinados pelo raio r_i . Conhecendo-se o conjunto de dados este raio pode ser definido pelo usuário (protótipos de tamanho fixos) ou ele pode ser estimado utilizando o conjunto de dados, conforme descrito a seguir.

Uma maneira natural de determinar o raio r_i , $i = 1, \dots, M$, onde M é o número de classes, é relaciona-lo ao tamanho da classe. Isto pode ser conseguido considerando-se a matriz de covariância da classe

$$P_i = \frac{\sum_{k=1}^N \mu_{ik}^m (x_k - v_i)^T}{\sum_{k=1}^N \mu_{ik}^m} \quad (2.32)$$

O determinante $|P_i|$ da matriz de covariância da classe fornece o volume da classe. Por ser P_i uma matriz definida positiva e simétrica ela pode ser decomposta de tal maneira que $P_i = Q_i A_i Q_i^T$, onde Q_i é ortonormal e A_i é diagonal com elementos não-zero $\lambda_{i1}, \dots, \lambda_{in}$. Fazendo os protótipos com volume alcançar uma distância $\sqrt{\lambda_{ij}}$, $j=1, 2, \dots, n$ ao longo de cada autovetor q_{ij} . No caso unidimensional, esta escolha implica na extensão do protótipo da classe para um desvio padrão a partir do centro da classe. No caso multidimensional, o tamanho do raio em cada direção é determinado pela medida das distâncias ao longo das coordenadas transformadas de acordo com

$$\sqrt{\Lambda_i} Q_i^T A_i Q_i \sqrt{\Lambda_i} \quad (2.33)$$

onde $\sqrt{\Lambda_i}$ representa uma matriz na qual os elementos são iguais a raiz quadrada dos elementos de Λ_i e A_i é uma matriz definida positiva e simétrica.

Quando A_i induz uma norma diferente daquela dada pela matriz de covariância serão obtidos n valores diferentes para os raios. Neste caso, um valor único pode ser determinado fazendo uma média dos valores. A forma dos protótipos com volume é a mesma da forma das classes, induzida pela métrica usada.

A medida de distância usada nos algoritmos de classificação estendidos é uma versão modificada das medidas de distâncias originais. Primeiro, a distância d_{ik} é medida da amostra x_k ao centro da classe v_i . Então, a distância medida \tilde{d}_{ik} ao protótipo com volume \tilde{v}_i é determinado levando-se em conta o raio r_i

$$\tilde{d}_{ik} = \max(0, d_{ik} - r_i) \quad (2.34)$$

Devido ao fato que pontos x_k dentro da distância r_i são considerados como pertencente completamente a uma única classe, a influência desse pontos nas demais classes é removida, isto é, esses pontos recebem grau de pertinência zero em outras classes. Isto diminui a tendência de regiões densas atrair outros centros de classes.

2.2.6.5 Otimização do número de classes

Os algoritmos de separação em classes apresentados nos itens 2.2.6.1, 2.2.6.2 e 2.2.6.3 possuem os seguintes pontos em comum:

1. Minimizam implicitamente o erro de partição, isto é, o erro quadrático total das amostras dentro de uma classe dado por

$$J(L, U) = \sum_{i=1}^C \sum_{k=1}^N \mu_{ik}^m d_{ik}^2 \quad (2.35)$$

onde L é uma matriz que contém todos os centróides, ou seja, $L = (c_1, c_2, \dots, c_c)$ e U é uma matriz de dimensão $C \times N$ que contém todas as pertinências μ_{ik} , chamada de matriz de partição- C nebulosa.

2. Partem da condição de se fixar, a priori, a quantidade de classes (C).

Quando se conhece a forma e a distribuição esperada das amostras a serem separadas em classes a última restrição não é importante. Porém, para a maioria das aplicações a quantidade de classes que o espaço deve ser dividido não é conhecida. A determinação do número de classes é uma das maiores discussões na técnica de separação em classes, e existem diversos estudos sobre esta questão.

A seguir são apresentados três critérios, um proposto por SUGENO *et al.* (1993), outro por KRISHNAPURAM *et al.* (1993) e outro por XIE *et al.* (1991), com propósito de determinar a validade de uma separação em classes.

O critério de otimização do número de classes proposto por SUGENO *et al.* (1993) é dado pela equação

$$J(L, U) = \sum_{i=1}^C \sum_{k=1}^N \mu_{ik}^m (d_{ik}^2 - d_{i\bar{x}}^2) \quad (2.36)$$

onde \bar{x} é a média das amostras : $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$.

Pode-se observar pela equação (2.36) que o primeiro termo do lado direito da equação representa a variância das amostras em uma determinada classe e o segundo termo representa a variância da própria classe em relação a média de todas as amostras.

O critério proposto por KRISHNAPURAM *et al.* (1993) parte do conceito de que o simples relaxamento na restrição aplicada em (2.35) produz a solução trivial, isto é, a função objetivo é minimizada fazendo todas as pertinências iguais a zero. Claro que o que se deseja é ter valores de pertinência grande para pontos que representam bem as classes e valores pequenos para os pontos que não as representam bem. O critério de otimização do número de classes proposto por KRISHNAPURAM *et al.* (1993) tenta contornar este problema e é dado por

$$J(L, U) = \sum_{i=1}^C \sum_{k=1}^N \mu_{ik}^m d_{ik}^2 - \sum_{i=1}^C \eta_i \sum_{k=1}^N (1 - \mu_{ik})^m \quad (2.37)$$

onde η_i são números positivos dados pela equação (2.31). O primeiro termo da equação (2.37) faz com que as distâncias entre as amostras e as classes seja a menor possível, enquanto que o segundo termo força a μ_{ik} ser o maior possível, deste modo, evita-se a solução trivial.

O critério proposto por XIE *et al.* (1991) tenta fazer com que a distância entre as classes seja a maior possível e é definido pela seguinte expressão

$$J(L,U) = \frac{1}{N} \frac{\sum_{i=1}^C \sum_{k=1}^N \mu_{ik}^m d_{ik}^2}{\min_{i \neq j} d_{ij}^2} \quad (2.38)$$

onde d_{ij} é a distância entre os centros da i -ésima classe e da j -ésima classe.

A minimização do número de centróides é feita de maneira a $J(L,U)$ alcançar um valor mínimo em relação L , a medida que o número de classes C aumenta. O primeiro ponto de mínimo encontrado, normalmente um mínimo local, é utilizado pelo método. A solução global da minimização de $J(L,U)$ em relação a L não é utilizada porque esta solução levaria a solução trivial, ou seja, implicaria em se ter uma classe para cada amostra. Na verdade, o que se deseja é obter um compromisso entre minimizar o erro quadrático total em relação a L e, ao mesmo tempo, obter um número mínimo de classes.

A minimização em relação às pertinências é feita de maneira a $J(L,U)$ alcançar um valor mínimo em relação a U , a medida que as pertinências são atualizadas. Esta minimização é feita para:

1. Separação em classes simples - automaticamente, ou seja, a pertinência de uma amostra k na classe i é dada pela equação (2.18);
2. Separação em classes nebulosas - ao derivar e igualar a zero a equação (2.36), o ponto de mínimo obtido é dado pela equação (2.27), que foi utilizada para cálculo das pertinências para este método;
3. Separação em classes possibilísticas - ao derivar e igualar a zero a equação (2.37), o ponto de mínimo obtido é dado pela equação (2.30), que foi utilizada para cálculo das pertinências para este método.

Logo, as equações utilizadas para cálculo das pertinências em cada um dos métodos minimizam a função objetivo $J(L,U)$, em relação ao parâmetro U .

Finalmente, descrevemos a seguir o critério proposto por KAYMAK *et al.* (2000) de separação em classes nebulosas usando protótipos com volume (EFCM).

O método inicializa o algoritmo de separação em classes com o número máximo de classes estimadas pelo usuário. Após avaliar as similaridades entre as classes, as classes similares são fundidas. A similaridade entre as classes é calculada com base nas classes nebulosas do espaço das amostras. Se a similaridade entre as classes é maior do que um limiar $\alpha \in [0,1]$, as duas classes mais similares são fundidas em cada iteração do algoritmo.

O índice Jaccard (SETNES *et al.*, 1998) é uma boa medida da igualdade entre conjuntos nebulosos. Em classificação, contudo, o objetivo é obter classes bem separadas no conjunto de dados. Para este propósito, a medida de inclusão é um índice melhor de medida de similaridade entre os conjuntos nebulosos. A figura 2.8 apresenta um conjunto nebuloso A que tem um alto grau de inclusão com o conjunto nebuloso B. De acordo com o índice Jaccard, esses dois conjuntos têm um grau de similaridade pequeno. Para classificação, contudo, o conjunto A pode ser considerado muito similar ao B, tendo em vista que ele pode ser representado em grande parte por B. Esta qualidade é quantificada pela medida inclusão nebulosa. Dado duas classes, $\mu_i(x_k)$ e $\mu_j(x_k)$, que define um ponto em X , a medida de inclusão nebulosa é definida como

$$I_{ij} = \frac{\sum_{k=1}^N \min(\mu_{ik}, \mu_{jk})}{\sum_{k=1}^N \mu_{ik}} \quad (2.39)$$

A medida de inclusão representa a razão da cardinalidade da interseção de dois conjuntos nebulosos dividida pela cardinalidade de um deles.

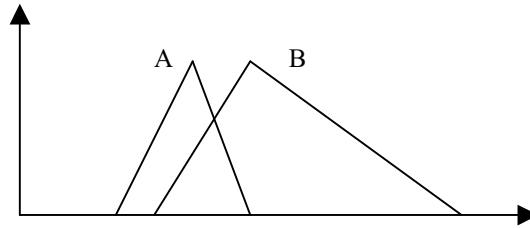


Figura 2.8 - O grau de igualdade entre A e B é pequeno, mas o grau de inclusão de A em B é grande.

A medida de inclusão é assimétrica e pode ser usada para construir uma medida de similaridade simétrica que determina o grau de similaridade $S_{ij} \in [0,1]$, com $S_{ij} = 1$ correspondendo a $\mu_i(x_k)$ totalmente incluído em $\mu_j(x_k)$, ou vice-versa, fazendo

$$S_{ij} = \max(I_{ij}, I_{ji}) \quad (2.40)$$

O limiar $\alpha \in [0,1]$ acima do qual a fusão de classes é realizada depende das características do conjunto de dados (separação entre grupos, densidade das classes, tamanho das classes, etc.), assim como, os parâmetros de classificação, como o grau de nebulosidade m . Em geral, o limiar de fusão é um parâmetro adicional definido pelo usuário no algoritmo de classificação estendida. O grau de similaridade entre duas classes também depende das outras classes na partição. Isto se deve ao fato que a soma das pertinências de uma amostra é restrita a 1.0. Para o caso onde a seleção do limiar é problemática, sugere-se utilizar um limiar adaptativo dependendo do número de classes na partição em qualquer tempo. Foi observado empiricamente que o limiar adaptativo

funciona bem quando o número esperado de classes no conjunto de dados é relativamente pequeno (menor do que 10). Como sugestão propõe-se

$$\alpha^{(l)} = \frac{1}{M^{(l)} - 1} \quad (2.41)$$

como limiar adaptativo. A fusão de classes é realizada quando a variação da similaridade entre duas classes da iteração $(l-1)$ para a iteração (l) é inferior a um limiar ε , e a similaridade é superior ao limiar α . Somente o par de classes mais similar é fundido e o número de classes é diminuído de 1 cada vez que ocorre uma fusão. Em caso de empate, ela é resolvida arbitrariamente. O algoritmo termina quando a troca nos elementos da matriz de partição é menor do que um limiar definido ε (critério de término).

Os algoritmos de separação em classes simples, nebulosa e possibilística são apresentados no anexo V.

2.3 O processamento dos sinais

A etapa de processamento de sinal dos modelos empíricos é o núcleo do sistema de validação de sinal. Como já foi citado, esta etapa pode ser dividida em: seleção do modelo, tipo de estrutura do modelo, quantidade de estruturas e do método de otimização das estruturas do modelo.

2.3.1 Seleção do modelo

Para seleção do modelo a ser utilizado pelo sistema deve-se levar em conta o tipo de aplicação a que se destina o sistema. Dentre os vários modelos disponíveis para realização do sistema de validação de sinal podemos citar: os modelos matemáticos, as redes neurais do tipo multicamadas (MLP - *Multi Layer Perceptron*), as redes neurais do tipo por vetor de quantização (AVQ), as redes neurais do tipo base radial (RBF), os

sistemas especialistas, os sistemas de inferência nebulosa (FIS), os sistemas de inferência nebulosa adaptativo (ANFIS), etc.

Os modelos matemáticos normalmente são construídos por especialistas da planta a ser monitorada que utilizam equações de balanço de massa, momento e energia do processo ou modelos matemáticos empíricos, baseados nas suas experiências.

As técnicas de inteligência artificial através de redes neurais e sistemas de inferência nebulosa adaptativo têm sido recentemente aplicadas na validação de sinal por não necessitarem de especialistas da planta para geração e treinamento do sistema. Esses modelos “aprendem” o comportamento dos sensores a serem modelados através em um conjunto de dados de operação da planta. A lei de aprendizado pode ser supervisionada, para os casos das redes MLP e do ANFIS, e não supervisionada, para o caso das redes AVQ e RBF.

Os sistemas especialistas, como o nome sugere, necessitam de especialistas da planta para realizar a modelagem dos sinais. Esses especialistas utilizam regras de inferência tipo *se-então* para modelar os estados da planta.

Informações mais detalhadas sobre as técnicas de modelagem através de sistemas empíricos podem ser encontradas em HAYKIN (1994), KOSKO (1992) e RICH (1988).

2.3.2 Seleção do tipo de estrutura

Os modelos de validação de sinal podem ser construídos com estruturas auto-associativas ou hetero-associativas.

Modelo com estruturas auto-associativas são aqueles em que as variáveis de entrada e de saída do modelo são as mesmas, ou seja, o número de entradas para o modelo é igual ao número de saídas do modelo. Além disso, o modelo é treinado com pares entrada/saída de mesmo valor.

Modelos com estruturas hetero-associativas são aqueles em que, normalmente, as variáveis de entrada são diferentes das variáveis de saída do modelo, ou seja, o número de entradas para o modelo é diferente do número de saídas do modelo. Normalmente, o número de saídas é menor do que o número de entradas.

As redes neurais se adaptam tanto à estruturas auto-associativas como à estruturas hetero-associativas. Já os sistemas neuro-nebulosos do tipo ANFIS devem ser aplicados a estruturas hetero-associativas com uma única saída. Os sistemas neuro-nebulosos do tipo ANFIS não possuem regras para seu treinamento para aplicações que envolvam mais de uma saída.

2.3.3 Seleção da estrutura

O sistema de validação de sinal pode possuir estrutura de processamento de sinal do tipo: única e múltipla. Entende-se por estrutura única o sistema que possui um único módulo para realizar a redundância dos sinais monitorados. Os modelos com estrutura múltipla utilizam vários módulos para realizar a redundância dos sinais monitorados, cada um operando numa região específica da planta.

É obvio que para os modelos com estruturas múltiplas faz necessário a separação do espaço de operação do modelo em sub-regiões ou classes de modo que cada estrutura seja treinada com as amostras correspondentes às sub-regiões.

A tabela 2.3 apresenta de maneira sucinta algumas vantagens e desvantagens da utilização de um modelo com estrutura única ou com várias estruturas em validação de sinal.

O tempo de treinamento de uma estrutura única para trabalhar em todas as regiões de operação de uma planta é maior do que o tempo gasto para treinar uma estrutura para trabalhar em uma região específica de operação da planta. Entretanto, a soma dos

tempos de treinamento de diversas estruturas, cada uma operando em uma determinada região de operação da planta, pode até ser maior do que o tempo gasto para treinar uma única estrutura para operar em todas as regiões de operação da planta.

Para o modelo com várias estruturas especializadas, durante a fase de operação do modelo, a mudança de classe ao longo do processo de validação leva a troca da estrutura selecionada a operar e, por conseguinte, leva a uma degradação no sinal gerado pelo modelo. Esta degradação é devida ao fato que aumento do número de estruturas especializadas implica num aumento da especialização das estruturas e, conseqüentemente, na redução da amplitude das descontinuidades da resposta de saída do modelo, causada pela mudança da estrutura selecionada. Por outro lado, a redução no número de estruturas especializadas implica na redução do número de descontinuidades, porém, também implica na redução da especialização das estruturas e, conseqüentemente, no aumento da amplitude das descontinuidades da resposta de saída do modelo causada pela mudança da estrutura selecionada.

O problema de extrapolação vem do fato de que após uma estrutura ser treinada, por exemplo, para realizar uma determinada função num intervalo $[a,b]$, receber sinais menores que a ou maiores que b a sua saída irá “saturar” para esses valores.

Tabela 2.3

Vantagens e desvantagens da utilização de única ou várias estruturas.

Estrutura única	Várias estruturas
Treinamento demorado	Treinamento rápido
Menor especialização	Maior especialização
-----	Problema com a mudança da estrutura
	Problema de extrapolação

2.3.4 Seleção do método de otimização

Os modelos baseados em redes neurais e sistemas neuro-nebulosos devem ser treinados com o intuito de “aprender” o mapeamento entrada/saída do conjunto de dados disponível para o seu treinamento. A princípio qualquer algoritmo de otimização pode ser utilizado para treinamento do modelo. Os principais métodos, até então utilizados para treinamento desses modelos são: método dos mínimos quadrados, método de gradiente descendente e algoritmos genéticos. Nesta tese propomos utilizar o método de otimização por enxame de partículas (PSO) para treinar o sistema de validação de sinal proposto.

No anexo III são dados alguns detalhes sobre esses métodos de otimização. A descrição detalhada sobre cada um desses métodos de otimização pode ser encontrada em HAYKIN (1994), GOLDBERG (1989), TRELEA (2003).

2.4 O pós-processamento dos sinais

A etapa de pós-processamento de sinal dos modelos empíricos tem por objetivo fornecer indicação de anomalia de um ou mais sensores da planta e, se for o caso, a indicação da confiança da resposta produzida pelo modelo. Em outras palavras, fornecer ao operador da planta indicação de falha em um sinal monitorado e garantia sobre a confiabilidade nas indicações produzidas pelo sistema.

2.4.1 Métodos de detecção da anomalia de um sensor

Os métodos de detecção de falha em sensor em um sistema de validação de sinal são baseados no sinal de erro do sensor. Este sinal de erro é calculado a cada instante por

$$\varepsilon = y - \hat{y} \quad (2.42)$$

onde ε é o erro calculado da diferença entre o valor instantâneo y produzido pelo sensor e o valor instantâneo estimado \hat{y} pelo sistema de validação de sinal.

Quando o sensor está funcionando corretamente, essa estimativa é virtualmente idêntica à variável monitorada e a diferença é próxima a zero. Se um erro é introduzido na variável monitorada (falha de sensor ou *drift*) uma diferença é encontrada entre o valor produzido pelo sistema e o valor monitorado.

2.4.1.1 Método do erro (resíduos)

O método de detecção de anomalia por resíduos de erro do sensor se baseia na estatística do sinal de erro ε obtida durante a etapa de treinamento do modelo de validação de sinal. Nesta etapa são obtidos os valores de média μ_ε e desvio padrão σ_ε do sinal de erro ε entre o valor monitorado e estimado pelo sistema do sensor. Com base nesta estatística é associado um limiar de erro máximo permitido.

Na etapa de operação do modelo quando este limiar é ultrapassado é gerada uma indicação de falha de sensor.

2.4.1.2 Método da auto-correlação dos resíduos

Auto-correlação é definida como a correlação de uma variável com ela mesma em instantes sucessivos de tempo.

O método de detecção de anomalia por auto-correlação dos resíduos se baseia na correlação entre os sinais de erro ε obtidos em cada instante de operação do sistema. Neste método é definida uma janela de tempo dentro da qual são calculados os coeficientes de auto-correlação entre os resíduos (sinais de erro ε) da seguinte maneira.

Para uma série de dados $\{x^k : k = 1, \dots, K\}$, o n -ésimo coeficiente de auto-covariância é definido por

$$g_n = \sum_{k=n+1}^K (x^k - \bar{x})(x^{k-n} - \bar{x}) / K \quad (2.43)$$

onde \bar{x} é a média das amostras dada por

$$\bar{x} = \left(\sum_{k=1}^K x^k \right) / K \quad (2.44)$$

Então, o n -ésimo coeficiente de auto-correlação é dado por

$$r_n = g_n / g_0 \quad (2.45)$$

No estágio de treinamento, uma janela deslizante é especificada de maneira a calcular r para cada instante que é apresentada uma nova amostra ao sistema. A figura 2.9 apresenta um exemplo dos valores de coeficientes de auto-correlação obtidos em um instante t qualquer para um sinal com uma janela de 50 amostras. O coeficiente de correlação em $t = 0$ é igual a 1, isto é, a amostra é altamente correlacionada com ela mesma. As demais amostras atrasadas têm valores de coeficiente de correlação com a amostra no instante t (na figura, $t = 0$) entre $[-1, 1]$.

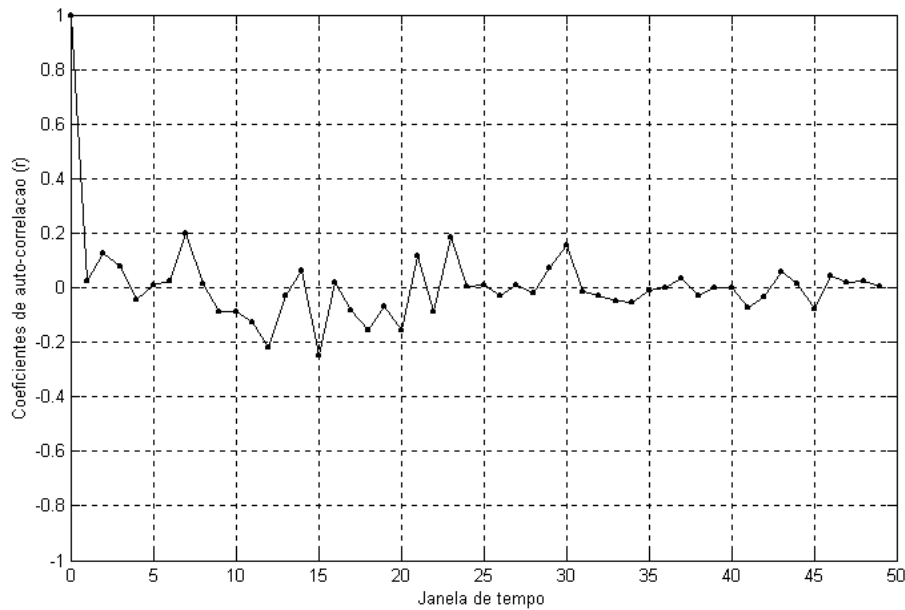


Figura 2.9 - Exemplo de valores de coeficientes de auto-correlação de um sinal com uma janela de 50 amostras.

Ao final do treinamento são obtidos os valores de coeficientes de correlação médios (\bar{r}), de distância média (\bar{d}) entre os vetores r e o vetor \bar{r} , e desvio padrão (σ_d) das distâncias do sinal de auto-correlação dos resíduos ε entre o valor monitorado e estimado pelo sistema do sensor. Com base nesta estatística é associado um limiar de distância máxima permitida, que quando ultrapassado gera a indicação de falha de sensor.

O vetor \bar{r} é obtido através de uma média dos coeficientes de auto-correlação do vetores r de cada janela de tempo, sendo cada uma de suas componentes \bar{r}_n dada por

$$\bar{r}_n = \frac{1}{m} \sum_{i=1}^m r_{ni} \quad (2.46)$$

onde r_{ni} é o valor da n -ésima componente do vetor dos coeficientes de auto-correlação r na i -ésima janela e m é o número de janelas de tempo do conjunto de treinamento.

Uma medida de distância deve ser utilizada de maneira a medir as distâncias d entre objetos ou pontos no espaço de realizações. Existem várias métricas que podem ser usadas como medida de similaridade entre vetores, por exemplo, distância Euclidiana, distância Mahalanobis, distância Minkowski, etc. Sem perda de generalidade, a distância de Manhattan pode ser utilizada para medir a distância entre dois vetores ao longo de eixos ortogonais.

$$d = \|x_1 - y_1\| + \|x_2 - y_2\| + \dots + \|x_n - y_n\| \quad (2.47)$$

onde $x = [x_1, x_2, \dots, x_n]$ e $y = [y_1, y_2, \dots, y_n]$ são os dois vetores a serem medidos.

Assumindo cada componente n do vetor \bar{r}_n e r_n como pertencentes ao eixo ortogonal n da distância de Manhattan, pode-se calcular a distância d entre os vetores \bar{r} e r . Deste modo, são calculadas d_m distâncias entre os m vetores r e o vetor \bar{r} . E, a partir dessas d_m distâncias são obtidas a distância média \bar{d} e o desvio padrão σ_d .

No estágio de operação, a cada nova amostra do sinal apresentada ao modelo é calculado o vetor de auto-correlação dos resíduos de erro do sensor r , dentro da janela deslizante de mesmo tamanho da usada no treinamento. Sendo, então, calculada a distância d entre o vetor r e o vetor \bar{r} , obtido na etapa de treinamento. Se a distância d obtida for maior do que um limiar de distância máxima permitida, é gerado uma indicação de falha de sensor pelo modelo.

2.4.1.3 Método da densidade de potência dos resíduos

O método de detecção de anomalia pela densidade de potência se baseia nos coeficientes de Fourier obtidos dos sinais de erro ε em cada instante de operação do sistema. Neste método é definida uma janela de tempo dentro da qual é calculada os coeficientes de Fourier entre os resíduos (sinais de erro ε) da seguinte maneira.

Uma série de dados $\{x^k : k = 1, \dots, K\}$ pode ser representada pela sua série de Fourier definida por

$$x^k = (1/K) \sum_{n=0}^{k-1} C_n e^{j2\pi nk/K} \quad (2.48)$$

onde C_n são os coeficientes de Fourier e $j = \sqrt{-1}$.

Então, os coeficientes do espectro da densidade de potência podem ser definidos como

$$p_n = \|C_n\|^2 \quad (2.49)$$

No estágio de treinamento, uma janela deslizante é especificada de maneira a calcular p para cada instante que é apresentada uma nova amostra ao sistema. Ao final do treinamento são obtidos os valores de coeficientes de densidade de potência médios (\bar{p}), de distância média (\bar{d}) entre os vetores p e o vetor \bar{p} , e desvio padrão (σ_d) das distâncias do sinal de auto-correlação dos resíduos ε entre o valor monitorado e estimado pelo sistema do sensor. Com base nesta estatística é associado um limiar de distância máxima permitida, que quando ultrapassado gera a indicação de falha de sensor.

O vetor \bar{p} é obtido através de uma média dos coeficientes de densidade de potência do vetores p de cada janela de tempo, sendo cada uma de suas componentes \bar{p}_n dada por

$$\bar{p}_n = \frac{1}{m} \sum_{i=1}^m p_{ni} \quad (2.50)$$

onde p_{ni} é o valor da n -ésima componente do vetor dos coeficientes de densidade de potência p na i -ésima janela e m é o número de janelas de tempo do conjunto de treinamento.

Do mesmo modo que na auto-correlação dos resíduos é utilizada a distância de Manhattan para medir a cada instante a distância d entre os vetores \bar{p} e p . Deste modo, são calculadas d_m distâncias entre os m vetores p e o vetor \bar{p} . E, a partir dessas d_m distâncias são obtidas a distância média \bar{d} e o desvio padrão σ_d .

No estágio de operação, a cada nova amostra do sinal apresentada ao modelo é calculado o vetor de densidade de potência dos resíduos de erro do sensor p , dentro da janela deslizante de mesmo tamanho daquela usada no treinamento. Sendo, então, calculada a distância d entre o vetor p e o vetor \bar{p} , obtido na etapa de treinamento. Se a distância d obtida for maior do que um limiar de distância máxima permitida, é gerado uma indicação de falha de sensor pelo modelo.

2.4.1.4 Método do teste da razão de probabilidades seqüencial dos resíduos

Normalmente, para verificar se um sensor está degradado ou não deve-se computar, a cada nova amostra do sinal monitorado, uma nova média e uma nova variância. Contudo, este procedimento requer muitas amostras para se obter uma média e uma variância significativas. Durante a aquisição das amostras, pode ocorrer uma degradação significativa do processo monitorado. Neste caso, é necessário um método para detectar uma falha usando o grau de degradação e o comportamento contínuo do sensor sem ter que calcular uma nova média e uma nova variância a cada amostra. O teste da razão de probabilidades seqüencial (SPRT - *Sequential Probability Ratio Test*) é um modelo estatístico que foi desenvolvido por Wald em 1945 (WALD, 1947), que satisfaz esses requisitos.

O objetivo da detecção da degradação de sensor é detectar a degradação tão rápida quanto possível com uma pequena probabilidade de fazer uma decisão errônea. Na

aplicação de monitoração de sensor, o SPRT usa o resíduo (diferença entre o valor medido e o valor estimado do sensor - ε). Normalmente, os resíduos dos sinais são aleatoriamente distribuídos, então, eles não são correlacionados e têm uma distribuição normal $P_i(\varepsilon_k, \mu_i, \sigma_i)$ dada por

$$P_i(\varepsilon_k) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left[-\frac{(\varepsilon_k - \mu_i)^2}{2\sigma_i^2}\right] \quad (2.51)$$

onde ε_k é o sinal de resíduo (sinal de erro) no tempo k e μ_i e σ_i são respectivamente a média e o desvio padrão sobre a hipótese i .

A degradação do sensor pode ser descrita em termos da troca na média μ ou a troca na variância σ^2 . O princípio fundamental do SPRT se baseia na razão de duas probabilidades: a probabilidade que o sensor está degradado dividida pela probabilidade que o sensor está normal, sendo dada por

$$\gamma_k = \frac{P_1(\varepsilon_k | H_1)}{P_0(\varepsilon_k | H_0)} \quad (2.52)$$

onde H_1 representa a hipótese que o sensor está degradado e H_0 representa a hipótese que o sensor está normal. Esta razão é atualizada a cada nova amostra.

Se um conjunto de amostras $x_i, i=1,2,\dots,n$, é adquirido com uma função densidade P descrevendo cada amostra do conjunto, a razão de probabilidades total é dada por

$$\gamma_n = \frac{P_1(\varepsilon_1 | H_1).P_1(\varepsilon_2 | H_1).P_1(\varepsilon_3 | H_1) \dots P_1(\varepsilon_n | H_1)}{P_0(\varepsilon_1 | H_0).P_0(\varepsilon_2 | H_0).P_0(\varepsilon_3 | H_0) \dots P_0(\varepsilon_n | H_0)} \quad (2.53)$$

Isto é devido ao fato de que a probabilidade de uma seqüência de medidas x_1, x_2, \dots, x_n é igual ao produto de probabilidades associadas a cada medida, ou seja,

$$P(x_1, x_2, x_3, \dots, x_n) = P(x_1) \cdot P(x_2) \cdot P(x_3) \cdot \dots \cdot P(x_n).$$

Tomando o logaritmo da equação anterior e substituindo as funções de densidade de probabilidade em termos de resíduos, médias e variâncias, o logaritmo da razão de probabilidade (LLR - *Log Likelihood Ratio*), λ_n , pode ser escrito na seguinte forma recorrente:

$$\lambda_n = \lambda_{n-1} + \ln\left(\frac{\sigma_0}{\sigma_1}\right) + \frac{(\varepsilon_n - \mu_0)^2}{2\sigma_0^2} - \frac{(\varepsilon_n - \mu_1)^2}{2\sigma_1^2} \quad (2.54)$$

Esta fórmula é usada para se derivar o algoritmo de *drift* de sensor. Usando a equação anterior, nós podemos identificar dois tipos de degradação de sensor, polarização e ruído. Se somente é verificada a degradação de polarização ($\sigma_1^2 = \sigma_0^2$ e $\mu_0 = 0$), tendo em vista que neste caso os resíduos dos sinais normalmente têm média quase zero, a equação anterior pode ser convertida na seguinte equação

$$\lambda_n = \lambda_{n-1} + \frac{\mu_1}{\sigma_0^2} \left(\varepsilon_n - \frac{\mu_1}{2} \right) \quad (2.55)$$

Se somente é verificada degradação de ruído ($\mu_0 = \mu_1 = 0$), tendo em vista que neste caso os valores de média não mudam ao longo do tempo, a equação pode ser convertida na seguinte equação

$$\lambda_n = \lambda_{n-1} + \ln\left(\frac{\sigma_0}{\sigma_1}\right) + \frac{\varepsilon_n^2}{2} \left(\frac{1}{\sigma_0^2} - \frac{1}{\sigma_1^2} \right) \quad (2.56)$$

Para um sensor normal, o logaritmo da razão de probabilidades deve decrescer e eventualmente alcançar um valor muito menor que zero, chamado de condição de contorno A . Quando o logaritmo da razão de probabilidades alcançar este valor, a decisão feita é que o sensor está normal e, o logaritmo da razão de probabilidade é reinicializada, fazendo-o igual a zero. Para um sensor degradado o logaritmo da razão

de probabilidades irá crescer e eventualmente alcançar um valor muito maior que zero, chamado de condição de contorno B . Quando o logaritmo da razão de probabilidades é igual a B , a decisão feita é que o sensor está degradado. As condições de contorno A e B são escolhidas de tal modo que se tenha uma probabilidade de falso alarme α e uma probabilidade de alarme perdido β dadas por

$$A = \ln\left(\frac{\beta}{1-\alpha}\right) \text{ e } B = \ln\left(\frac{1-\beta}{\alpha}\right) \quad (2.57)$$

Existem dois métodos de implementar o SPRT. O método dos dois lados de Wald, que pode ser usado tanto para polarização como para degradação de ruído, e o método de um lado de CHIEN *et al.* (1976), que pode ser usado para degradação de polarização devido a sua falta de expressão apropriada para o critério de alarme perdido para o caso de degradação por ruído.

Na etapa de treinamento do modelo, são obtidos os valores de média μ_0 e desvio padrão σ_0 do sinal de erro ε entre o valor monitorado e estimado do sensor pelo sistema. Com base nesta estatística são associados dois limiares de erro máximo permitidos. O erro máximo permitido por polarização μ_l e o erro máximo permitido por ruído σ_l .

Na etapa de operação do modelo, a cada instante é calculado o sinal de erro ε e a razão acima é calculada e comparada com os critérios de aceitação para determinar se o sensor está normal (razão menor do que A), falho (razão maior do que B) ou está entre uma coisa ou outra (razão entre A e B). Os critérios de aceitação para o SPRT são selecionados de maneira que as aplicações do método não sejam tão sensíveis de produzir falsos alarmes nem tão insensíveis que resultem na perda de alarmes.

2.4.2 A forma de detecção da anomalia de um sensor

Entende-se por forma de detecção de anomalia de um sensor como sendo a maneira na qual é efetuado o processo decisório sobre o estado de um sensor. Nos métodos de detecção da anomalia de um sensor descritos anteriormente, esta decisão é baseada em um limiar que, quando ultrapassado, gera a informação de falha do sensor.

Os métodos de detecção de anomalia podem ser utilizados sozinhos ou combinados através de uma lógica de votação.

A forma mais comum de lógica de votação para um processo decisório combinando vários métodos de detecção de anomalia é a lógica de votação majoritária. Nesta lógica a indicação que a maioria dos métodos fornecer será assumida como a resposta final do modelo.

Uma outra forma possível de lógica de votação é considerar como resposta final do modelo a indicação de falha produzida por qualquer dos métodos de detecção utilizados. Neste caso, se um método de detecção é deficiente na detecção uma determinada falha e outro não o é, a resposta final do sistema não será comprometida. Por outro lado, qualquer indicação de falso de alarme em um dos sistemas redundantes produzirá uma resposta errônea do sistema.

Uma outra opção para o processo decisório é utilizar uma lógica nebulosa. Neste caso, em vez de se realizar uma lógica binária dos sinais de falha (sinais falhos ou não falhos) produzidos pelos métodos de detecção utiliza-se os valores analógicos produzidos pelos métodos, por exemplo, o sinal de erro do sinal, o sinal de auto-correlação dos resíduos, etc. Esses valores são aplicados às entradas de um sistema de inferência nebulosa que fornece à sua saída a indicação do estado do sinal. Esta forma de processo decisório tem a capacidade de fornecer, por exemplo, indicação de tendência de degradação de um sensor.

2.4.3 A confiança da resposta

Como já foi citado, a etapa de pós-processamento de sinal dos modelos empíricos tem por objetivo fornecer ao operador da planta indicação de falha em um sinal monitorado e garantia sobre a confiabilidade nas indicações produzidas pelo sistema.

Os métodos de detecção de falha em sensor em um sistema de validação de sinal são baseados no sinal de erro ε do sensor, calculado a cada instante pelo sistema.

Deve-se lembrar que ε deve ser considerado como a soma do erro de estimação do modelo (e^{modelo}) e o erro de medida (e^{medida}), ou seja

$$\varepsilon = e^{\text{modelo}} + e^{\text{medida}} \quad (2.58)$$

Se e^{modelo} é minimizado pelo treinamento, ε pode ser considerado como sendo um parâmetro que representa o erro de medida e, conseqüentemente, pode ser usado para validação de sinal. O objetivo do treinamento é minimizar o valor de e^{modelo} para o conjunto de amostras de treinamento. Contudo, se este valor for conhecido somente para o conjunto de amostras de treinamento o comportamento do sistema de validação de sinal é imprevisível quando é apresentada para ele uma amostra diferente das que foram treinadas. Deste modo, o erro associado com o modelo de predição pode variar de uma ordem de grandeza menor a várias ordens de grandeza maior do que o erro de medida. Neste caso, como ε pode tornar-se muito grande, a conclusão do sistema será que o sensor fornece valores errados, que não é um diagnóstico correto.

O módulo de confiança da resposta tem por objetivo fornecer uma indicação ao operador da planta sobre o grau de confiança da estimativa do sinal produzida pelo sistema de validação de sinal. Em outras palavras, ele indica se o sistema de validação de sinal foi treinado para as amostras (sinais monitorados) que são apresentados às suas entradas.

Geralmente, o módulo de confiança da resposta é realizado por um sistema de inferência nebulosa (FIS) que recebe às suas entradas os sinais de resíduos produzidos pelos sistemas de detecção de falhas e um sinal de pertinência das amostras de entrada do sistema ao conjunto de treinamento e fornece à sua saída uma indicação quantitativa sobre a confiança da resposta produzida pelo sistema.

CAPÍTULO 3

Avaliação das etapas de um modelo de validação de sinal

Neste capítulo são apresentados os ensaios realizados nas diversas sub-etapas que compõem as etapas de um modelo de validação de sinal com o objetivo de auxiliar o processo de seleção das metodologias a serem aplicadas no modelo de validação de sinal implementado nesta tese.

3.1 Seleção dos sinais de entrada para o modelo de validação

Dentre as várias técnicas de seleção dos sinais de entrada para modelos de validação de sinal que utilizam redundância dinâmica, citadas no capítulo anterior, as técnicas que buscam obter sinais de entrada com alto grau de correlação entre eles, usando algoritmos de otimização por GA ou PSO, são interessantes, pois não necessitam de um especialista do processo monitorado.

A seguir são apresentadas duas metodologias, usando GA ou PSO, para seleção de variáveis de entrada para um modelo de validação de sinal.

De modo a facilitar o entendimento das técnicas utilizadas nos procedimentos de seleção das variáveis de entrada são apresentados diversos anexos ao final desta tese. O anexo II apresenta um resumo sobre análise das componentes principais, o anexo III sobre técnicas de otimização por mínimos quadrados, gradiente descendente, algoritmos genéticos e otimização por enxame de partículas e o anexo V sobre algoritmos de separação em classes. Informações mais detalhadas sobre os algoritmos genéticos e sobre a otimização por enxame de partículas podem ser encontrados em GOLDBERG (1989) e KENNEDY *et al.* (1995), respectivamente.

3.1.1 Seleção dos sinais de entrada para o modelo de validação - 1ª metodologia

Para se usar GA ou PSO deve-se representar uma solução para um dado problema como um cromossomo ou uma partícula, que podem ser pensados como um ponto no espaço de busca das soluções candidatas. O GA ou o PSO necessitam de uma função de avaliação (*fitness*) que atribui um valor a cada cromossomo ou partícula da população atual. O valor de *fitness* de um cromossomo ou de uma partícula depende de quão bem este cromossomo ou partícula resolve o problema em questão.

Neste caso, a função-*fitness* F que avalia o quanto cada indivíduo (cromossomo ou partícula) satisfaz os critérios para seleção para um determinado conjunto de sinais de entrada para o modelo de validação de sinal leva em conta os seguintes fatores:

- a) A correlação linear entre os sinais de entrada e o sinal de saída;
- b) A correlação linear entre os sinais de entrada.

A função-*fitness* F utilizada é dada por

$$F = 1 - e^{(-\alpha_1 E_1 - \alpha_2 E_2)} \quad \Rightarrow F \in [0,1] \quad (3.1)$$

onde α_1 e α_2 são coeficientes de peso (valores reais maiores que zero) e E_1 e E_2 são dados por

$$E_1 = 1 - \frac{1}{N} \sum_{i=1}^N \text{abs}(S_{io}) \quad \Rightarrow E_1 \in [0,1] \quad (3.2)$$

$$E_2 = \frac{1}{C_N^2} \sum_{i=1}^N \sum_{j=i+1}^N \text{abs}(S_{ij}) \quad \Rightarrow E_2 \in [0,1] \quad (3.3)$$

onde E_1 é o valor de 1 menos o somatório dos valores absolutos das correlações S entre os sinais de entrada i e a saída o do modelo de validação de sinal normalizado pelo número de sinais de entrada N do modelo de validação de sinal e E_2 é o valor do somatório dos valores absolutos das correlações entre os sinais de entrada i e j

normalizado pelo número de combinações C , tomadas duas a duas, desses N sinais de entrada.

Durante o processo de otimização a função-*fitness* F deve ser “minimizada” e, para tal, ela deve maximizar o valor das correlações dos sinais de entrada selecionados com a saída desejada e minimizar o valor das correlações entre os sinais de entrada, ou seja, fazer os termos E_1 e E_2 tenderem para zero.

A representação de um indivíduo de seleção dos sinais de entrada para um modelo de validação de sinal pode ser realizada da seguinte maneira. Considere o indivíduo a ser utilizado como sendo um conjunto de números inteiros cujo número de elementos é igual ao número de sinais de entrada desejados, onde cada número inteiro pode variar de 1 até o número máximo N de sinais de entrada disponíveis. Neste caso, cada variável de entrada disponível corresponde a um número específico. Por simplicidade, a variável de saída é numerada como $N+1$. A Tabela 3.1 apresenta um exemplo de uma aplicação fictícia na qual se deseja selecionar 4 sinais de entrada para um modelo de validação de sinal e são disponíveis 9 sinais como entrada para o modelo.

Tabela 3.1

Exemplo de aplicação do modelo de seleção de sinais de entrada.

	Indivíduo	<i>Fitness</i> (equação 3.1)	Observação
1	2 3 9 5	0,732	
2	1 6 2 8	0,345	
3	7 2 5 3	0,644	
4	4 7 4 1	1,000	Indivíduo não válido
5	3 5 7 9	0,576	

Neste exemplo é utilizada uma população de 5 indivíduos e não é permitida a utilização de uma variável como entrada mais de uma vez (indivíduo número 4 não é

válido e foi penalizado com valor máximo de *fitness*). Se considerássemos esta população como sendo a população ao final do processo de otimização o indivíduo número 2 seria o melhor indivíduo e, neste caso, as variáveis correspondentes aos números 1, 2, 6 e 8 seriam utilizadas como entrada para o modelo de validação de sinal, conforme mostrado na figura 3.1.

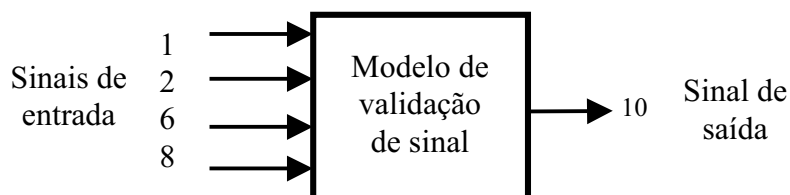


Figura 3.1 - Exemplo de sinais aplicados à entrada de um modelo de validação de sinal.

Na metodologia proposta o usuário deve definir os valores de α_1 e α_2 , as variáveis de entrada disponíveis a serem utilizadas pelo modelo e o erro mínimo de estimação do modelo desejado (critério de parada).

Sem perda de generalidade podemos fazer $\alpha_1 = 1$ em (3.1) e deixar para o usuário a escolha do valor de α_2 .

Por outro lado é intuitivo imaginar que o erro de estimação do modelo de validação de sinal diminua a medida que aumentamos o número de variáveis de entrada. Neste caso, podemos ir aumentando o número de variáveis de entrada para o modelo, selecionadas pela função de otimização dada em (3.1), até que o erro de estimação do mesmo seja satisfatório.

Como critério de satisfação (erro mínimo desejado) sugerimos usar o erro de estimação produzido pelo modelo de validação quando são aplicados às suas entradas os sinais obtidos através do processo de redução da dimensão de entrada, pela obtenção das componentes principais (PCA) das variáveis de entrada disponíveis.

O procedimento do método proposto de seleção das variáveis de entrada para um modelo de validação de sinal do tipo ANFIS é descrito a seguir:

Dado um conjunto de variáveis medidas ao longo do tempo X , retire deste conjunto a variável a ser monitorada e obtenha o conjunto de variáveis de entrada disponíveis.

1. Calcule as componentes principais do conjunto de variáveis de entrada disponível para o modelo de validação de sinal com perda mínima de informação ε_{PCA} definida pelo usuário, conforme descrito no item 3.3;
2. Calcule o número de classes contidas nas componentes principais obtidas pelo método EFCM com métrica Mahalanobis, conforme descrito no item 3.4;
3. Construa e treine um sistema de inferência neuro-nebuloso (ANFIS) com arquitetura similar à descrita no item 3.5, com tantas regras quantas as classes obtidas no passo 2 e com uma entrada para cada variável obtida pelo processo PCA;
4. Obtenha o erro médio quadrático de estimação ε_{crit} do sinal de saída do modelo. Este erro será utilizado como critério de parada no processo de seleção das variáveis de entrada;

Repita para $l = 2, 3, \dots$, onde l é o número de entradas do modelo.

5. Selecione as l variáveis de entrada do modelo através do processo de otimização da função-*fitness* dada por (3.1);
6. Construa e treine um sistema de inferência neuro-nebuloso (ANFIS) com arquitetura similar à descrita no item 3.5, com tantas regras quantas as classes obtidas no passo 2 e com as l variáveis de entrada;
7. Obtenha o erro médio quadrático de estimação ε_{mod} do sinal de saída do modelo.

Até $\varepsilon_{mod} < \varepsilon_{crit}$

Ao final do processo de otimização obteremos um conjunto mínimo de variáveis de entrada para o modelo de validação de sinal que produzem um erro de estimação ε_{mod} da mesma ordem do erro de estimação ε_{crit} produzido pelo método das componentes principais.

3.1.2 Seleção dos sinais de entrada para o modelo de validação - 2^a metodologia

Nesta metodologia, a função-*fitness* F que avalia o quanto cada indivíduo (cromossomo ou partícula) satisfaz os critérios para seleção de um determinado conjunto de sinais de entrada para o modelo de validação de sinal leva em conta os seguintes fatores:

- a) A correlação linear entre os sinais de entrada e o sinal de saída;
- b) A correlação linear entre os sinais de entrada;
- c) O erro médio quadrático de estimação do sinal de saída;
- d) O erro máximo absoluto de estimação do sinal de saída;
- e) O número de sinais de entrada;

Neste caso, a função-*fitness* F utilizada é dada por

$$F = 1 - e^{(-\alpha_1 E_1 - \alpha_2 E_2 - \alpha_3 E_3 - \alpha_4 E_4 - \alpha_5 E_5)} \quad \Rightarrow F \in [0,1] \quad (3.4)$$

onde α_1 , α_2 , α_3 , α_4 e α_5 são coeficientes de peso (valores reais maiores que zero) e E_1 e E_2 são dados por (3.2) e (3.3), respectivamente, e E_3 , E_4 e E_5 são dados por

$$E_3 = \frac{1}{K} \sum_{k=1}^K (y(k) - \hat{y}(k))^2 \quad \Rightarrow E_3 \in [0,1] \quad (3.5)$$

$$E_4 = \max\{|y(k) - \hat{y}(k)|\} \quad \Rightarrow E_4 \in [0,1] \quad (3.6)$$

$$E_5 = \frac{N_{\text{sinais de entrada}}}{N_{\text{sinais total}}} \Rightarrow E_5 \in (0,1] \quad (3.7)$$

onde E_3 é o valor do erro médio quadrático dos erros entre o sinal monitorado $y(k)$ e estimado $\hat{y}(k)$ pelo modelo validação de sinal para as K amostras de dados, E_4 é o valor do erro máximo absoluto para as K amostras de dados e E_5 é o valor da razão entre o número de entradas para o modelo de validação de sinal e o número total de sinais disponíveis.

Durante o processo de otimização a função-*fitness* F deve ser “minimizada” e, para tal, ela deve maximizar o valor das correlações dos sinais de entrada selecionados com a saída desejada e minimizar o valor erro médio quadrático de estimação, o valor do erro máximo absoluto médio de estimação e o número de sinais de entrada, ou seja, fazer os termos E_1, E_2, E_3, E_4 e E_5 tenderem para zero.

Nesta metodologia o usuário deve definir os valores de $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ e α_5 da equação (3.4). A escolha dos valores de $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ e α_5 deve levar em conta as contribuições dos fatores E_1, E_2, E_3, E_4 e E_5 na função-objetivo dada pela equação (3.4), que podem ser divididas em dois grupos. O primeiro grupo é composto pelos fatores E_3 e E_4 que contribuem para aumentar o número de variáveis de entrada selecionadas. É intuitivo imaginar que o erro de estimação do modelo de validação de sinal diminua a medida que aumentamos o número de variáveis de entrada do modelo. Consequentemente, os fatores E_3 (erro médio quadrático) e E_4 (erro máximo absoluto), das equações (3.5) e (3.6), respectivamente, diminuem a função-objetivo a medida que aumentamos o número de variáveis de entrada para o modelo. O segundo grupo é composto pelos fatores E_1 (correlações entre entradas e saída selecionada), E_2 (não correlação entre entradas selecionadas) e E_5 (número de entradas selecionadas), das

equações (3.2), (3.3) e (3.7), respectivamente, que diminuem a função-objetivo a medida que diminuimos o número de variáveis de entrada para o modelo.

A representação de um indivíduo de seleção dos sinais de entrada para um modelo de validação de sinal pode ser realizada da seguinte maneira. Considere o indivíduo a ser utilizado como sendo uma seqüência binária com comprimento igual ao número de sinais disponíveis para entrada do modelo, onde cada sinal disponível corresponde a uma posição específica neste indivíduo. A figura 3.2 apresenta um exemplo de um indivíduo para uma aplicação com 10 sinais disponíveis como entrada do modelo. Posições com valores 1 neste indivíduo significam que as variáveis correspondentes a essas posições são utilizadas como entrada para o modelo. De modo contrário, posições com valores 0 neste indivíduo significam que as variáveis correspondentes a essas posições não são utilizadas como entrada para o modelo.

1	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---

Figura 3.2 - Exemplo de um indivíduo para seleção dos sinais de entrada.

Neste exemplo as variáveis 1, 2, 6 e 8 são utilizadas como entrada para o modelo de validação de sinal, conforme mostrado na figura 3.3.

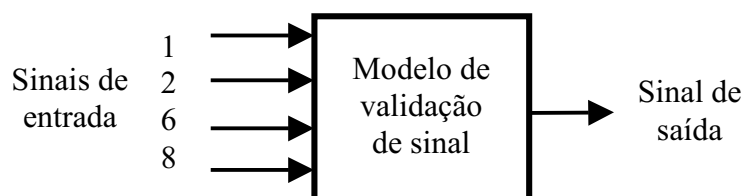


Figura 3.3 - Exemplo de sinais aplicados à entrada de um modelo de validação de sinal.

O diagrama de fluxo para seleção dos sinais de entrada do modelo de validação de sinal usando técnicas de otimização por algoritmos genéticos ou por otimização por enxame de partículas é apresentado na figura 3.4.

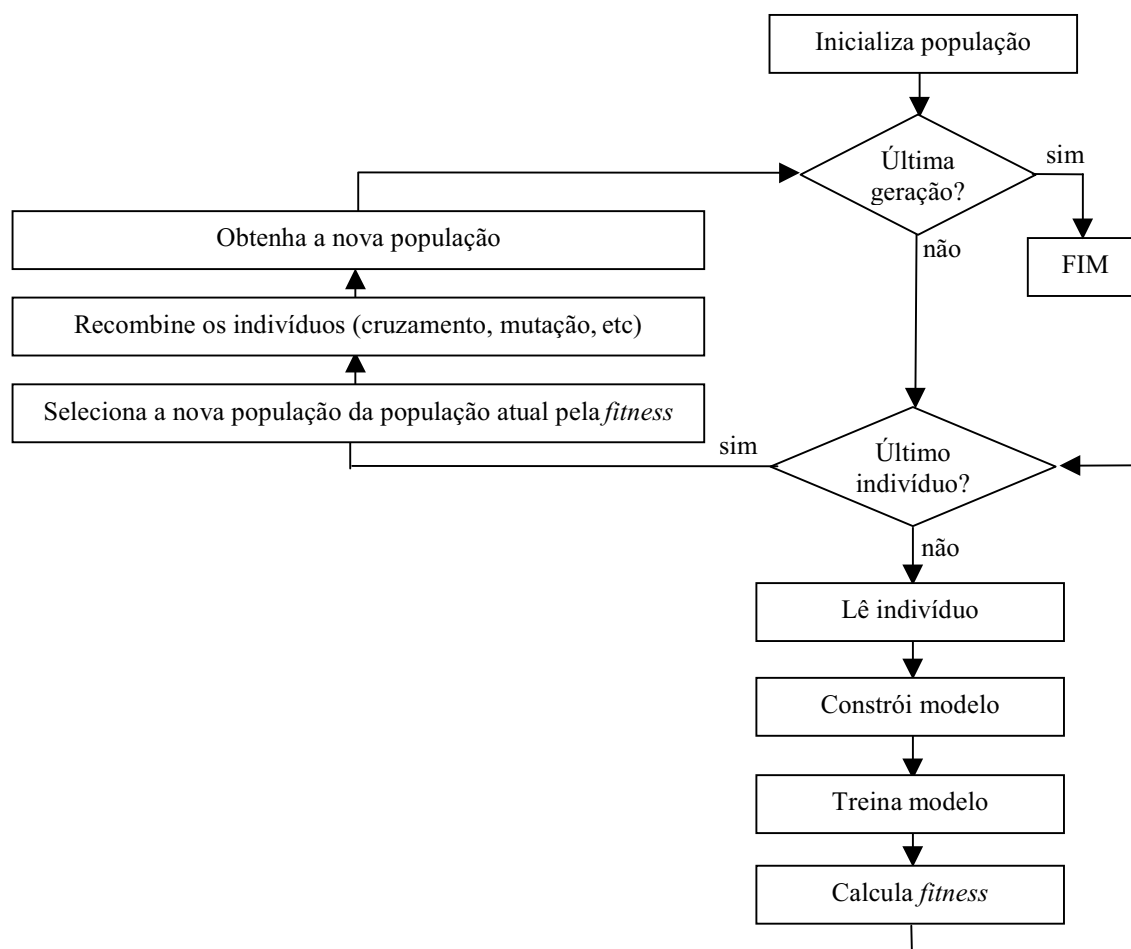


Figura 3.4 - Diagrama de fluxo de otimização da seleção dos sinais de entrada.

O procedimento do método proposto de seleção das variáveis de entrada para o modelo de validação de sinal do tipo ANFIS usando GA é descrito a seguir:

Dado um conjunto de variáveis medidas ao longo do tempo X , retire deste conjunto a variável a ser monitorada e obtenha o conjunto de variáveis de entrada disponíveis.

1. Calcule o número de classes contidas no conjunto de variáveis de entrada pelo método EFCM com métrica Mahalanobis, conforme descrito no item 3.4;
2. Inicialize a população de indivíduos $P(t)$, em $t=0$;

Repita**Repita**

3. Obtenha o valor total das correlações entre os sinais de entrada, selecionados pelo indivíduo da população, e o sinal de saída do modelo de validação de sinal, equação (3.2);
4. Obtenha o valor total das correlações entre os sinais de entrada, selecionados pelo indivíduo da população, equação (3.3);
5. Construa e treine um sistema de inferência neuro-nebuloso (ANFIS) com arquitetura similar à descrita no item 3.5, com tantas regras quantas as classes obtidas no passo 1 e com as variáveis de entrada selecionadas pelo indivíduo da população;
6. Obtenha o erro médio quadrático, equação (3.5), e o erro máximo absoluto, equação (3.6), de estimação do sinal de saída do modelo;
7. Obtenha o valor da razão entre o número de entradas para o modelo de validação de sinal, selecionados pelo indivíduo da população, e o número total de sinais disponíveis, equação (3.7);
8. Calcule o valor da *fitness* do indivíduo da população, equação (3.4);

Até último indivíduo da população

9. $t = t + 1$;
10. Selecione $P(t)$ de $P(t-1)$, baseado na função-*fitness* dos indivíduos, equação (3.4);
11. Recombine as estruturas em $P(t)$;

Até última geração

onde $P(t)$ é a população no tempo t das soluções candidatas da função-objetivo dada pela equação (3.4).

3.1.3 Seleção dos sinais de entrada para o modelo de validação - aplicação

A seguir apresentamos um exemplo de seleção de sinais de entrada para uma planta química (<http://neural.cs.nthu.edu.tw/jang/benchmark/>). Nesta aplicação a planta está produzindo um polímero pela polimerização de alguns monômeros. Tendo em vista que a partida da planta é muito complicada, um operador tem que atuar manualmente no processo durante sua partida. Existem cinco variáveis de entrada que o operador deve observar para controlar a planta, isto é, usar seu raciocínio para selecionar o ponto de ajuste da vazão do monômero. O operador determina o ponto de ajuste para a vazão do monômero e o valor real da vazão do monômero a ser colocado na planta é controlado por um controlador PID (*Proportional Integral Derivative*).

As variáveis de entrada (u_1, u_2, u_3, u_4 e u_5) e de saída (y) do processo são as seguintes:

- u_1 : concentração do monômero;
- u_2 : troca na concentração do monômero;
- u_3 : vazão do monômero;
- u_4, u_5 : temperaturas em locais dentro da planta;
- y : ponto de ajuste da vazão do monômero.

A figura 3.5 apresenta o conjunto de 70 amostras de dados obtidas da operação da planta real.

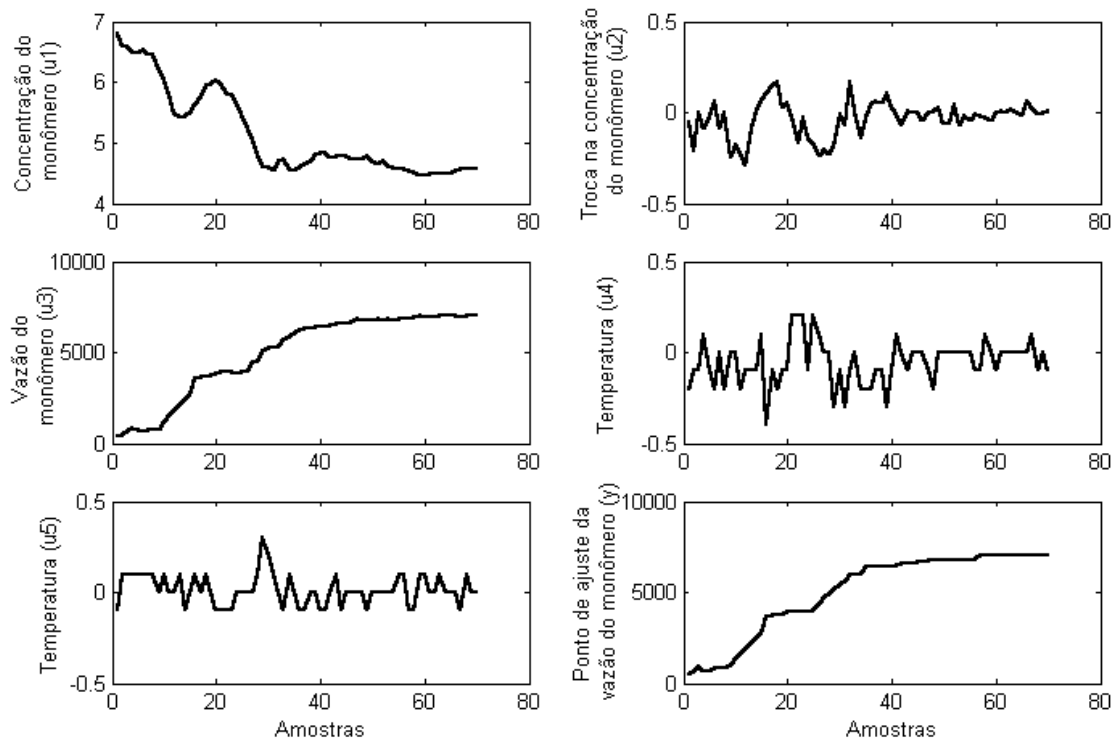


Figura 3.5 - Conjunto de dados da operação da planta química.

SUGENO *et al.* (1993) concluíram que o operador deve somente observar as informações de $u1$, $u2$ e $u3$ para decidir sua ação de controle, isto é, essas três variáveis de entrada modelam “otimamente” a variável controlada y da planta.

Vamos verificar quais os sinais de entrada da planta são selecionados pelas metodologias propostas para um sistema de validação de sinal usando um ANFIS.

3.1.3.1 Aplicação - 1ª metodologia

Primeiramente, os sinais do conjunto de dados da planta química são normalizados entre $[0,1]$. Após a normalização são calculados os coeficientes de correlação linear entre esses sinais. A tabela 3.2 apresenta os coeficientes de correlação linear entre as variáveis de operação da planta química.

Tabela 3.2

Coeficientes de correlação linear para os sinais coletados.

	$u1$	$u2$	$u3$	$u4$	$u5$	y
Concentração ($u1$)	1,0000	-0,1906	-0,9381	-0,0283	0,0837	-0,9461
Troca da concentração ($u2$)	-0,1906	1,0000	0,3404	-0,2173	-0,1795	0,3343
Vazão ($u3$)	-0,9381	0,3404	1,0000	0,1126	-0,2104	0,9984
Temperatura ($u4$)	-0,0283	-0,2173	0,1126	1,0000	-0,3911	0,1012
Temperatura ($u5$)	0,0837	-0,1795	-0,2104	-0,3911	1,0000	-0,2068
Ponto de ajuste da vazão (y)	-0,9461	0,3343	0,9984	0,1012	-0,2068	1,0000

A tabela 3.3 apresenta a contribuição percentual de cada variável obtida pelo método das componentes principais aplicado ao conjunto de variáveis de entrada.

Tabela 3.3

Principais componentes das variáveis $u1$, $u2$, $u3$, $u4$ e $u5$.

Componentes	Contribuição de cada componente (%)
1 ^a	42,8231
2 ^a	27,4167
3 ^a	19,8529
4 ^a	9,1231
5 ^a	0,7842
Total	100

Pode-se notar pela tabela 3.3 que 99,2% de toda a informação das variáveis de entrada ($u1$, $u2$, $u3$, $u4$ e $u5$) estão contidas nas 4 primeiras componentes principais. Neste caso vamos considerar ε_{PCA} igual a 0,01 (99% da informação mantida). A figura 3.6 apresenta o conjunto das 70 amostras obtidas pelo método PCA para as 4 primeiras componentes principais.

Na separação em classes pelo método EFCM com métrica Mahalanobis das 4 componentes principais obtidas pelo PCA são geradas 6 classes. Deste modo, é construído um ANFIS com modelo TSK (Takagi-Sugeno-Kang) de primeira ordem, com topologia considerando o número de regras igual a 6, o número de funções

pertinências por variável de entrada igual a 6 do tipo sino generalizada, treinamento do ANFIS por gradiente descendente para a parte antecedente e mínimos quadráticos para a parte conseqüente e que é treinado por 500 épocas. A figura 3.7 apresenta o diagrama em blocos para treinamento do modelo usando a redução de dimensão por PCA.

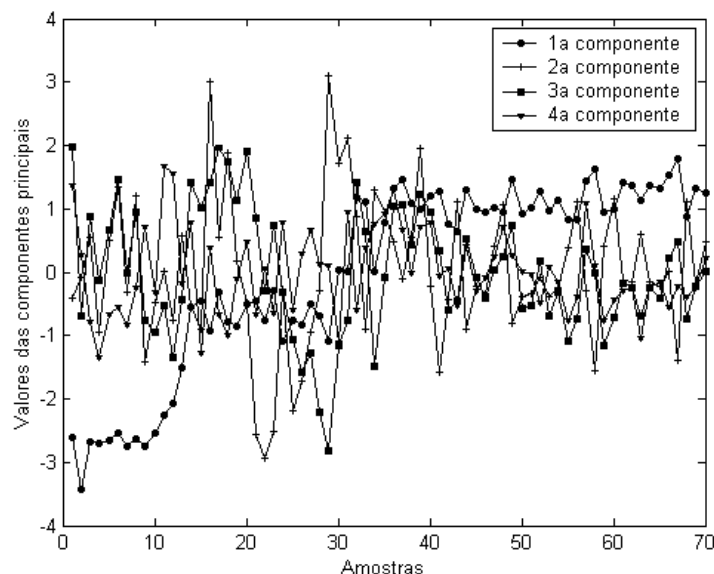


Figura 3.6 - Conjunto das amostras obtidas pelo processo do PCA.

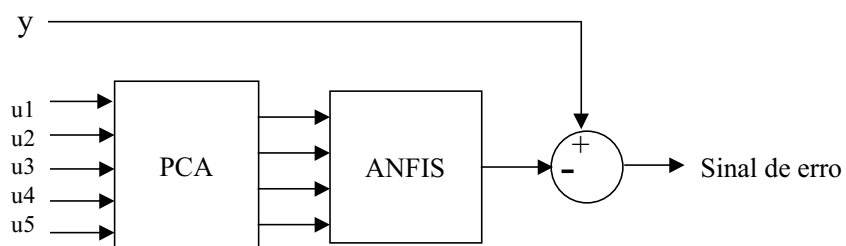


Figura 3.7 - Modelo de treinamento do ANFIS com PCA.

Após o processo de treinamento do ANFIS, considerando todas as amostras como pertencentes ao conjunto de treinamento, é obtido o erro médio quadrático ϵ_{crit} de

Tabela 3.5

Variáveis selecionadas para a planta química considerando $\alpha_1 = 1,0$ e $\alpha_2 = 0,5$.

No de entradas	Variáveis selecionadas	Erro médio quadrático (ε_{mod})
2	$u_1 u_2$	0,044302
3	$u_1 u_2 u_3$	0,004496
4	$u_1 u_2 u_3 u_5$	0,003804
5	$u_1 u_2 u_3 u_4 u_5$	0,002347

Pode-se observar pela tabela 3.5 que o erro médio quadrático ε_{mod} torna-se menor do que ε_{crit} quando são aplicadas 3 variáveis às entradas do modelo. Deste modo, foram selecionadas como entrada para o modelo as mesmas variáveis u_1 , u_2 e u_3 selecionadas anteriormente por Sugeno. Estas variáveis devem ser utilizadas como entradas para o sistema de validação de sinal conforme mostrado na figura 3.8.

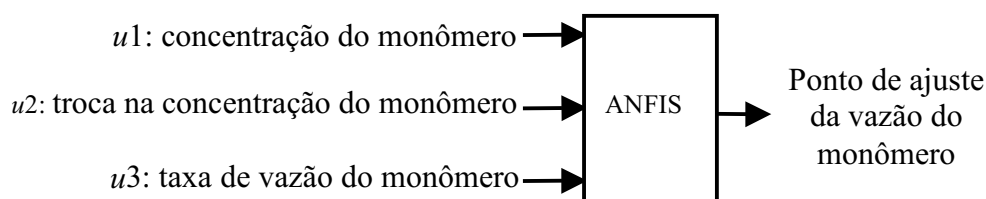


Figura 3.8 - Sinais do processo usados na aplicação da planta química.

3.1.3.2 Aplicação - 2ª metodologia

Primeiramente, os sinais do conjunto de dados da planta química foram normalizados entre $[0,1]$ e foi aplicado o método EFCM com métrica Mahalanobis para obtenção do número de classes das amostras de entrada, considerando todas as variáveis de entrada disponíveis no conjunto de dados. O método EFCM encontrou 3 classes no conjunto total de variáveis disponíveis.

Logo após, foi utilizado GA para obtenção do conjunto mínimo de variáveis de entrada para um sistema de inferência nebulosa adaptativo (ANFIS). A seleção dos sinais de entrada para o ANFIS é feita com base na função-objetivo (3.4). A tabela 3.6

apresenta as variáveis selecionadas pelo método para a aplicação da planta química considerando $\alpha_3 = \alpha_4 = 1$ e α_1, α_2 e α_5 variando entre $[0,1]$. Neste exemplo, foi utilizado GA para seleção das variáveis de entrada com: população de 50 indivíduos; taxa de recombinação de 0,6; taxa de mutação de 0,02 e número de épocas igual a 50.

O modelo de validação de sinal utilizado foi um ANFIS com modelo TSK de primeira ordem, com topologia considerando o número de regras igual a 3, o número de funções pertinência por variável de entrada igual a 3 do tipo sino generalizada, treinamento do ANFIS por gradiente descendente para a parte antecedente e mínimos quadráticos para a parte conseqüente e que é treinado por 500 épocas. A tabela 3.6 também apresenta os erros médios quadráticos de estimação obtidos para as diversas entradas aplicadas às entradas do modelo.

Tabela 3.6

Variáveis selecionadas para a planta química considerando

$\alpha_3 = \alpha_4 = 1,0$ e α_1, α_2 e α_5 variando entre $[0,1]$.

α_1	α_2	α_5	Entradas selecionadas	Erro médio quadrático (ϵ_{mod})
0,000	0,000	0,000	$u1 u2 u3 u4 u5$	0,0062986
0,025	0,025	0,025	$u1 u2 u3$	0,0093122
0,050	0,050	0,050	$u2 u3$	0,0109466
0,075	0,075	0,075	$u2 u3$	0,0109466
0,100	0,100	0,100	$u2 u3$	0,0109466
0,250	0,250	0,250	$u3 u4$	0,0112241
0,500	0,500	0,500	$u3 u4$	0,0112241
0,750	0,750	0,750	$u3 u4$	0,0112241
1,000	1,000	1,000	$u3 u4$	0,0112241

Pelos resultados obtidos podemos verificar que o sinal de saída é fortemente correlacionado com o sinal $u3$, que é selecionado para todos os valores ensaiados de α_1, α_2 e α_5 . As variáveis de entrada $u1 u2 u3$ são selecionadas para $\alpha_1 = \alpha_2 = \alpha_5 = 0,025$.

3.2 Filtragem dos sinais de entrada

Como já foi citado no capítulo 2 existem várias técnicas para filtragem de sinais digitais dentre as quais podemos citar a filtragem por média móvel, vide equação (2.2), a filtragem por média ponderada, vide equação (2.3) e a filtragem por ondaletas, vide equação (2.5).

Nos ensaios realizados nestas três técnicas de filtragem foram utilizadas janelas móveis de 8 e 16 amostras, para as técnicas de média móvel e ondaleta e pesos de ponderação de 8 e 16 amostras, para a técnica de média ponderada.

Para a técnica de filtragem por ondaletas foram utilizadas ondaletas do tipo SWT com as características apresentadas na tabela 3.7.

Tabela 3.7

Características da ondaleta utilizada nos ensaios.

Característica	Descrição
Função	Transformada por ondaleta estacionária (SWT - <i>Stationary Wavelet Transform</i>)
Tipo e forma da ondaleta	Daubechies de ordem 5
Nível de decomposição	$L = \log_2(J)$

A escolha do tamanho da janela móvel J ou do peso de ponderação P está diretamente relacionada com o tempo de resposta para transientes do filtro. Quanto maior for o tamanho da janela ou o valor do peso maior será o tempo de resposta do filtro. Por outro lado, quanto menor for o tamanho da janela ou o valor do peso menor será o tempo de resposta do filtro, porém, neste caso, se diminuirmos muito o tamanho da janela ou o valor do peso o processo de filtragem se reduzirá.

Para diminuir o tempo de resposta dos filtros propõe-se o seguinte procedimento: caso a diferença entre o valor médio da variável correspondente a amostra anterior e o valor da variável da amostra atual seja maior ou menor, em duas amostras consecutivas,

do que duas vezes a raiz quadrada do valor médio da variável correspondente a amostra anterior, ou seja, caso $(x_k - \bar{x}_{k-1}) > 2\sigma$ ou $(x_k - \bar{x}_{k-1}) < -2\sigma$, em duas amostras consecutivas,

onde $\sigma = \sqrt{\bar{x}_{k-1}}$, então:

- a) Para o filtro de média móvel: o valor atual da amostra é assumido como valor médio e a janela J é automaticamente reinicializada em 1, sendo aumentada de 1, a cada nova amostra, até atingir o tamanho de janela J definido pelo usuário;
- b) Para o filtro de média ponderada: o valor atual da amostra é assumido como valor médio e o peso de ponderação P é automaticamente reinicializado em 1, sendo aumentado em potências de 2, a cada nova amostra, até atingir o valor P definido pelo usuário;
- c) Para o filtro de ondaleta: o valor atual da amostra é assumido como valor médio e a janela J é automaticamente reinicializada em 1, sendo aumentada em potências de 2, a cada conjunto de potência de 2 de novas amostras, até atingir o tamanho de janela J definido pelo usuário. O valor de L é feito igual ao logaritmo na base 2 de J .

O anexo I apresenta uma descrição resumida sobre as ondaletas. Informações mais detalhadas da técnica de filtragem por ondaletas ser encontrado em DAUBECHIES (1992) e MISITI *et al.* (2000). Aplicações de filtros digitais de média ponderada podem ser encontradas em ALLEY *et al.* (1988).

A seguir apresentamos alguns resultados obtidos com a filtragem de sinais por essas três técnicas.

3.2.1 Tempo de resposta dos filtros

Tempo de resposta é definido como o tempo que um sistema leva para reagir a uma dada entrada. Para um sinal analógico o tempo de resposta é tempo que o sinal leva

para percorrer entre 10% à 90% da variação total de um sinal tipo degrau aplicado à entrada do sistema. Para um sinal digital o tempo de resposta é o tempo correspondente ao número de amostras que o sinal deve adquirir para percorrer entre 10% à 90% da variação total de um sinal tipo degrau aplicado à entrada do sistema.

A figura 3.9 apresenta os resultados obtidos pelos três métodos de filtragem para um sinal tipo degrau dentro de $\pm 2\sigma$ do sinal medido. Os gráficos mostram resultados obtidos considerando janelas e pesos de 8 e 16. Para os métodos de janela móvel e ondaleta o tempo de resposta corresponde ao intervalo de tempo para aquisição do número de amostras da janela utilizada. Para o método de média ponderada o tempo de resposta corresponde a aproximadamente ao intervalo de tempo para aquisição de $5P$ amostras.

A figura 3.10 apresenta os resultados obtidos pelos três métodos de filtragem para um sinal tipo degrau fora de $\pm 2\sigma$ do sinal medido. Os gráficos mostram resultados obtidos considerando janelas e pesos de 8 e 16. Devido ao procedimento adotado para redução do tempo de resposta, para os três métodos implementados o tempo de resposta corresponde a aproximadamente ao intervalo de tempo entre duas amostras do sinal monitorado.

3.2.2 Resposta dos filtros a um sinal tipo rampa

Um sinal tipo rampa positiva foi aplicado à entrada dos filtros de modo a verificar as suas respostas a esse transiente. A figura 3.11 apresenta os resultados obtidos pelos três métodos de filtragem para este sinal. Os gráficos mostram os resultados obtidos considerando janelas e pesos de 8 e 16. Os filtros do tipo média móvel e ondaletas foram os que obtiveram o menor atraso entre o sinal de entrada e o sinal filtrado.

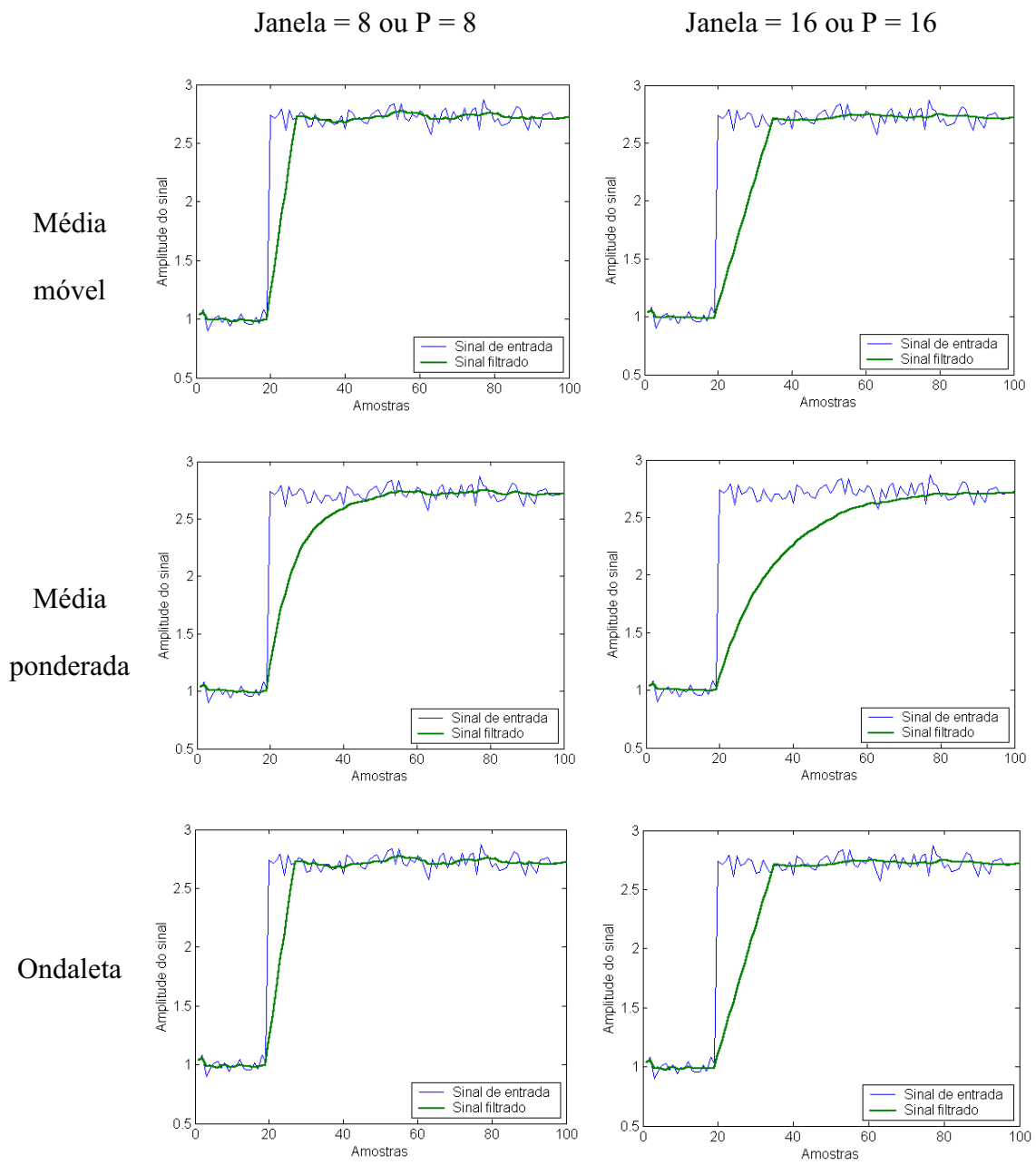


Figura 3.9 - Resultados obtidos pelos três métodos de filtragem para um sinal tipo degrau dentro de $\pm 2\sigma$ do sinal medido.

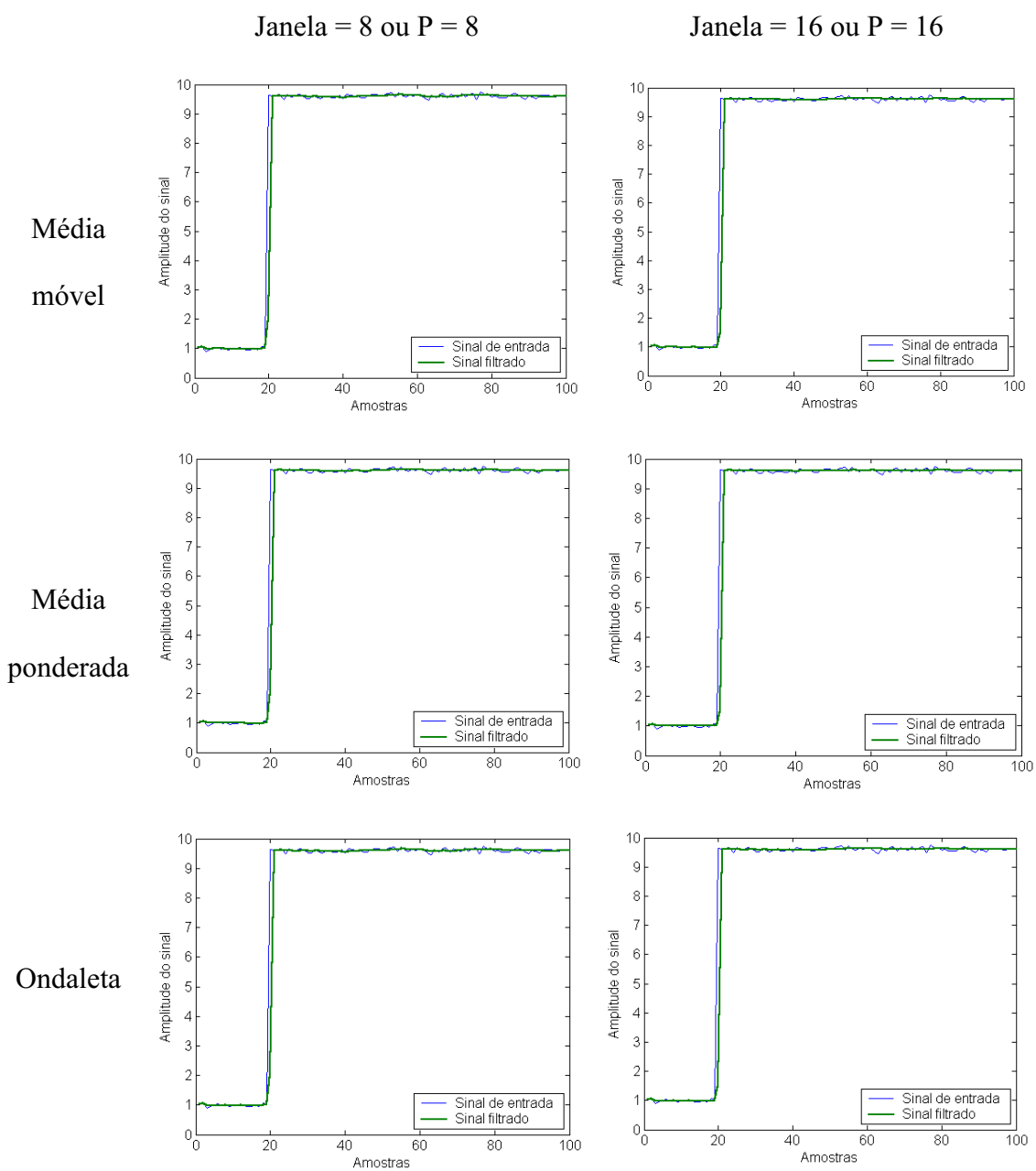


Figura 3.10 - Resultados obtidos pelos três métodos de filtragem para um sinal tipo degrau fora de $\pm 2\sigma$ do sinal medido.

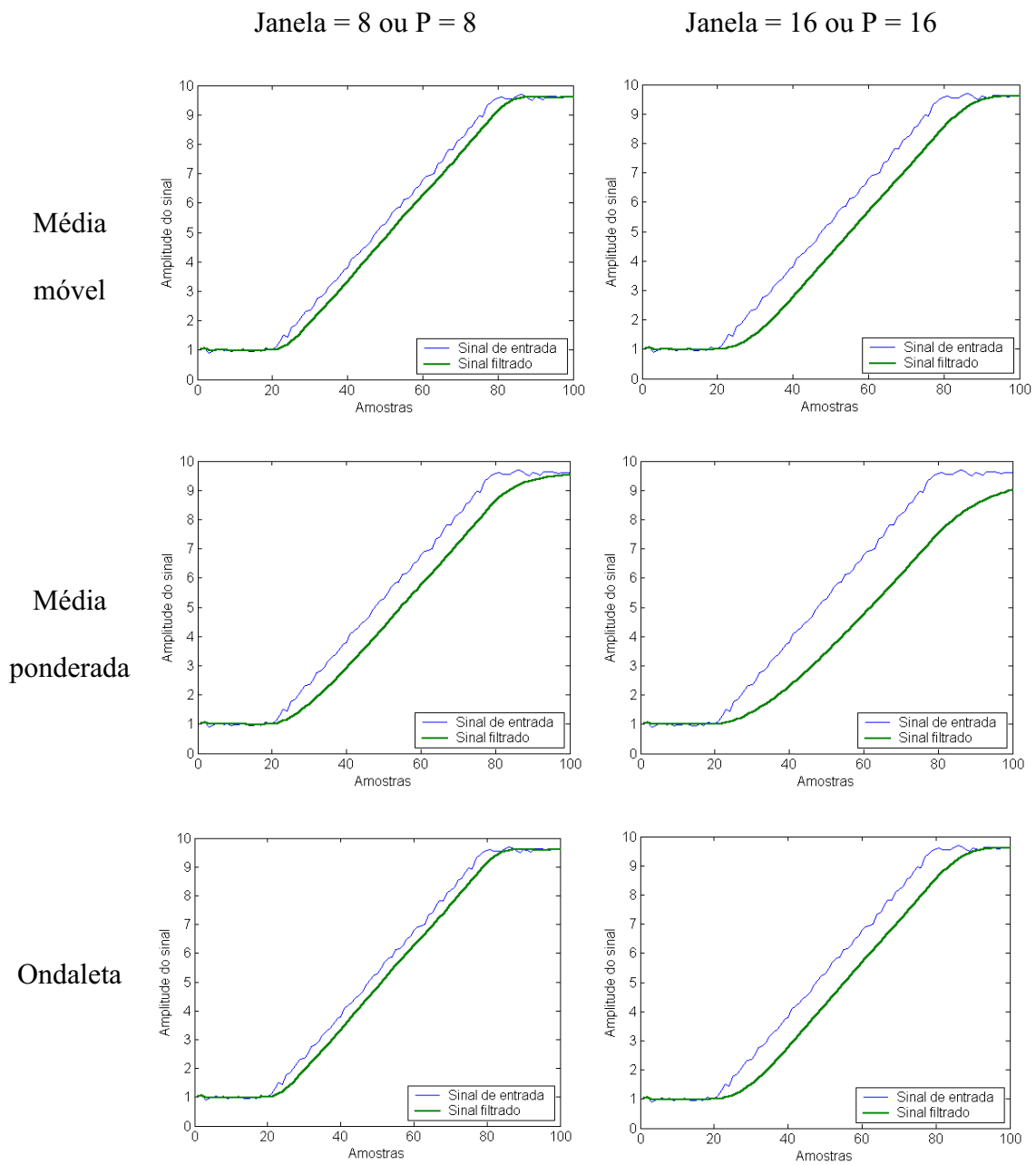


Figura 3.11 - Resultados obtidos pelos três métodos de filtragem para um sinal tipo rampa.

3.2.3 Filtragem de sinais de uma planta nuclear

Nesta aplicação são comparados os resultados da aplicação das técnicas de filtragem digital por média móvel, média ponderada e ondaletas aplicadas a um sinal de uma planta nuclear de potência. O sinal utilizado foi a vazão de água de alimentação de entrada de um dos geradores de vapor da planta nuclear. Os dados da vazão foram adquiridos do computador de processos da planta e correspondem a excursão de potência da planta de 0% à 100%. A figura 3.12 apresenta os resultados obtidos pelas três técnicas de filtragem de sinais. Para facilitar a comparação visual dos resultados obtidos é apresentado, repetido lado a lado, na parte superior da figura 3.12, o sinal de vazão da água de alimentação de entrada do gerador de vapor aplicado às entradas dos filtros.

3.3 Redução da dimensão do espaço de entrada

Como já foi citado no capítulo 2 existem várias técnicas para redução da dimensão do espaço de entrada de um sistema de validação de sinal dentre as quais podemos citar a técnica pela análise das componentes principais (PCA), utilização de ondaletas (*wavelets*) e a utilização de redes neurais auto-associativas.

O anexo II apresenta um resumo sobre a redução da dimensão de um espaço através da técnica de análise das componentes principais. Informações mais detalhadas da técnica de redução da dimensão de espaço de entrada por PCA podem ser encontradas em JACKSON (1991) e CHEN *et al.* (1999). Aplicações de redes neurais auto-associativas na redução da dimensão de espaço de entrada podem ser encontradas em TAN *et al.* (1995) e aplicações de ondaletas podem ser encontradas em HEO *et al.* (2003).

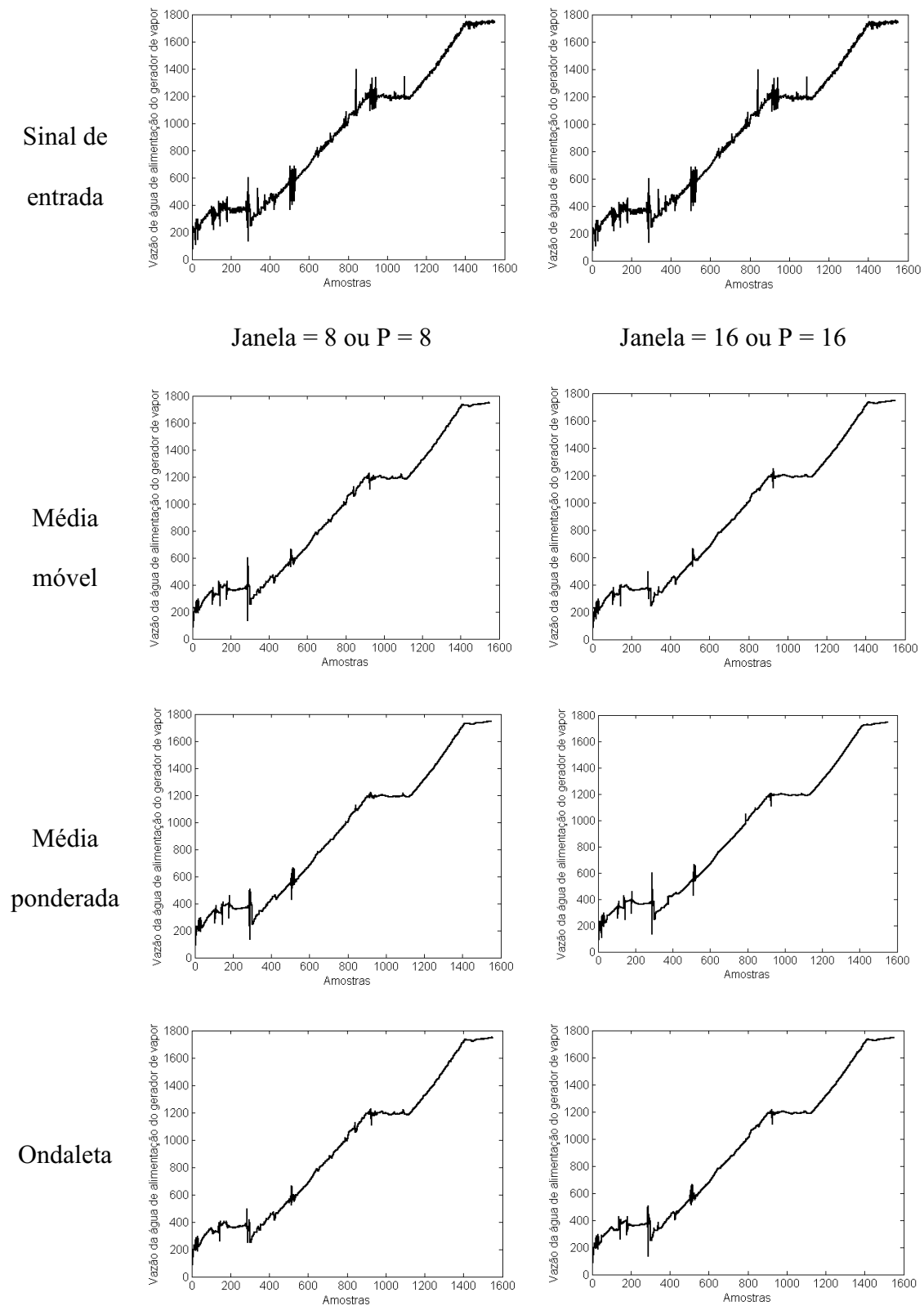


Figura 3.12 - Resultados obtidos pelas três técnicas de filtragem para o sinal de vazão de água de alimentação de uma planta nuclear.

A seguir apresentamos alguns resultados obtidos com a redução da dimensão do espaço de entrada pela análise das componentes principais.

3.3.1 Redução da dimensão do espaço de entrada usando PCA

Nesta aplicação a metodologia de redução da dimensão das variáveis de entrada de um modelo de validação de sinal por PCA foi ensaiada para o conjunto de dados da planta química apresentado no item 3.1.3 desta tese. O objetivo do teste é verificar a degradação dos sinais de entrada causada pela redução de sua dimensão.

Este conjunto de dados é composto por 70 amostras de 6 variáveis de operação da planta química, sendo 5 variáveis disponíveis como entrada do modelo de validação de sinal e uma variável de saída, a ser estimada.

A tabela 3.8 apresenta a contribuição percentual de cada componente obtida pelo método das componentes principais aplicado ao conjunto de variáveis de entrada.

Tabela 3.8

Principais componentes das variáveis u_1 , u_2 , u_3 , u_4 e u_5 .

Componentes	Contribuição de cada componente (%)
1 ^a	42,8231
2 ^a	27,4167
3 ^a	19,8529
4 ^a	9,1231
5 ^a	0,7842
Total	100

Pode-se notar pela tabela 3.8 que 99,2% de toda a informação das variáveis de entrada (u_1 , u_2 , u_3 , u_4 e u_5) estão contidas nas 4 primeiras componentes principais.

A figura 3.13 apresenta um diagrama do processo de redução (compressão) e recuperação (expansão) do conjunto de dados da planta química considerando as cinco

variáveis como entrada para o PCA e ϵ_{PCA} igual a 0,01, ou seja, 99% da informação mantida.

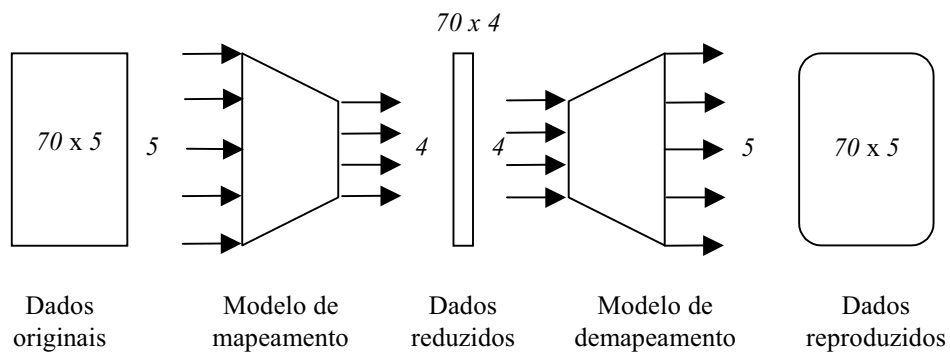


Figura 3.13 - Redução e expansão do conjunto de dados da planta química.

A figura 3.14 apresenta os resultados obtidos após o processo de recuperação (reprodução) do sinal u_1 (concentração do monômero) da planta química. A figura 3.14a mostra que o sinal recuperado está dentro de aproximadamente $\pm 3\%$ do sinal original e a figura 3.14b mostra os valores originais e recuperados após o processo PCA.

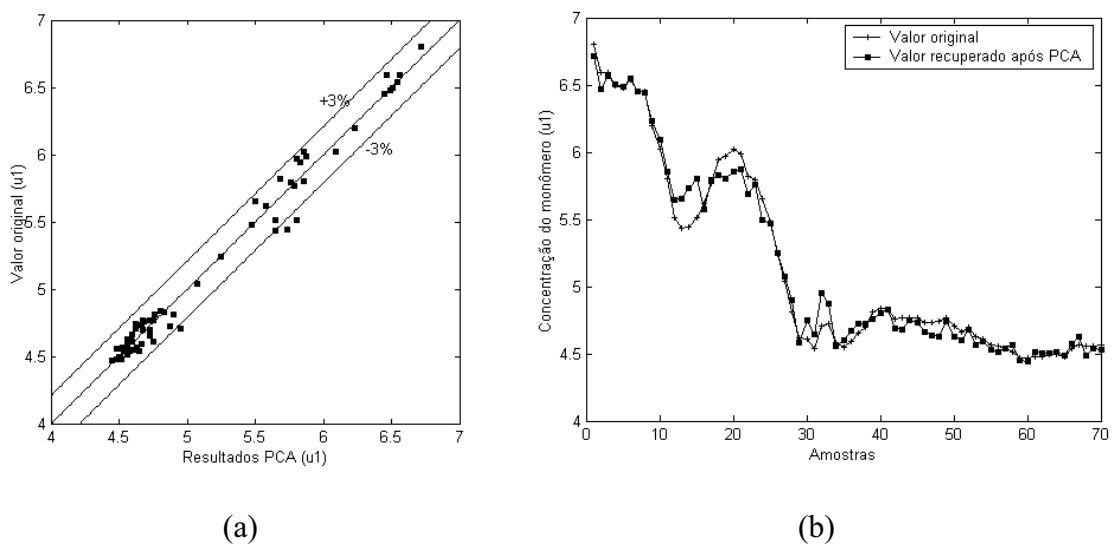


Figura 3.14 - Precisão após aplicação do PCA: (a) Sinal recuperado está dentro de aproximadamente $\pm 3\%$ do sinal original; (b) Valores originais e recuperados.

Pode-se concluir que o PCA após aprender as correlação entre os sinais conseguiu recuperar o sinal de entrada com uma pequena perda de informação.

3.3.2 Falha em um dos sinais de entrada do PCA - caso 1

Para verificar a robustez do PCA para valores dentro do intervalo treinado foi introduzida uma falha em um dos sinais de entrada após o processo de treinamento PCA do exemplo anterior. Suponhamos que o sinal $u1$ do exemplo teve uma falha do tipo “colado” a partir da amostra 20 do conjunto de dados. Isto significa uma degradação no sensor de medida da concentração.

A figura 3.15 apresenta os resultados obtidos após o processo de recuperação (reprodução) do sinal $u1$ (concentração do monômero) da planta química. A figura 3.15a mostra que o sinal recuperado não está mais dentro de aproximadamente $\pm 3\%$ do sinal original e a figura 3.15b mostra que os valores originais e recuperados após o processo PCA estão um pouco diferentes, indicando que foi possível recuperar em parte o sinal original.

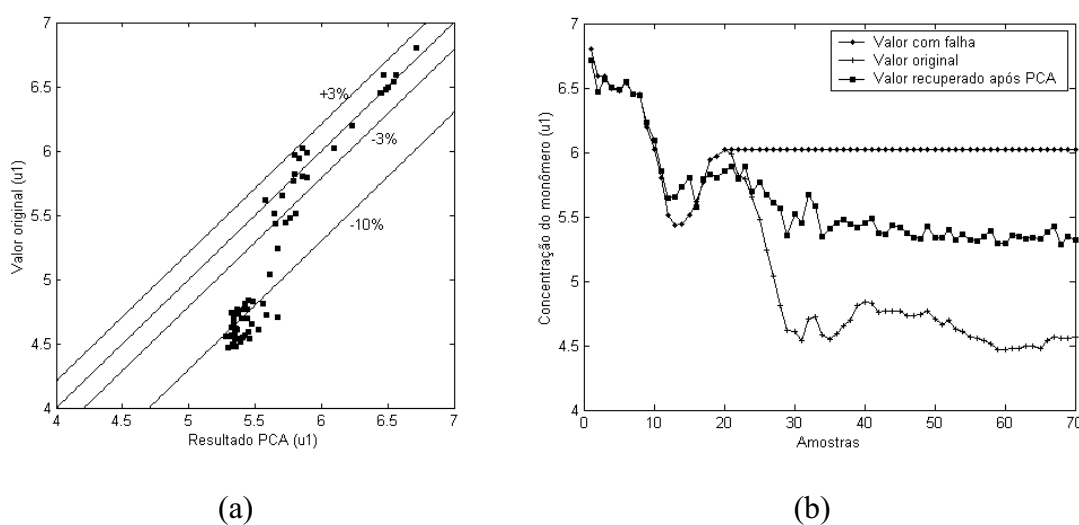


Figura 3.15 - Precisão após aplicação do PCA: (a) Sinal recuperado está fora de $\pm 3\%$ do sinal original após a falha; (b) Valores falhos, originais e recuperados.

O PCA tenta recuperar o sinal original da concentração do monômero (u_1) a partir das informações fornecidas pelas outras variáveis de entrada.

Como o sinal falho está dentro do intervalo usado para o processamento do PCA as componentes principais são pouco modificadas após a falha, conforme apresentado pela figura 3.16. A falha em uma das variáveis de entrada causa deformação maior na primeira componente principal, que contém a maior parte da informação dos sinais de entrada e, deste modo, o sinal não é completamente recuperado.

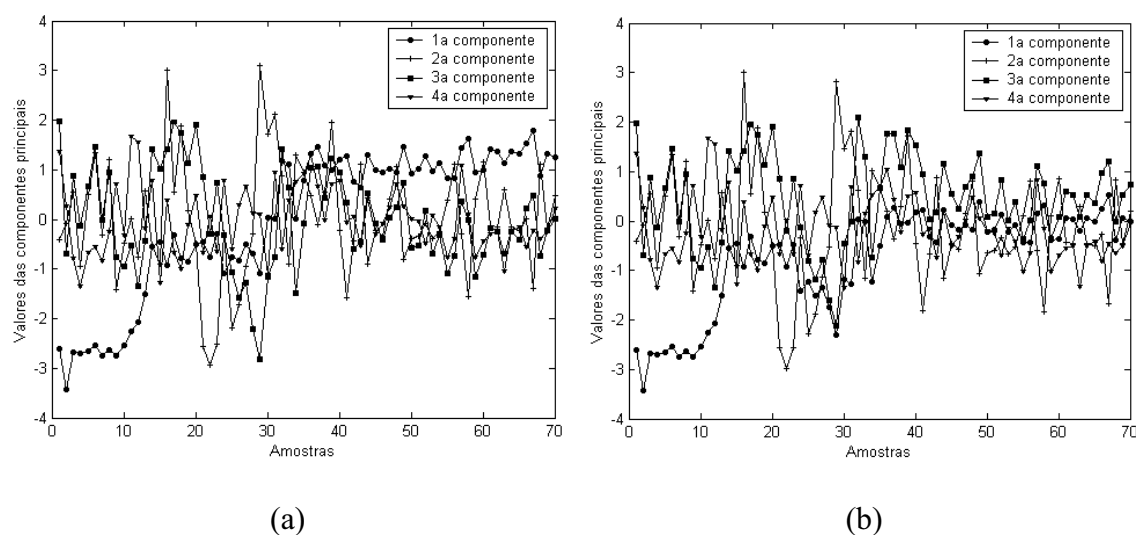


Figura 3.16 - Componentes da transformação PCA: (a) Obtidas pelo mapeamento das entradas originais; (b) Obtidas com a inserção da falha.

3.3.3 Falha em um dos sinais de entrada do PCA - caso 2

Neste caso, vamos supor que são disponíveis como entrada para processamento pelo PCA somente as 30 amostras finais de operação da planta química, ou seja, a planta está operando à plena vazão de monômero.

A tabela 3.9 apresenta a contribuição percentual de cada componente obtida pelo método das componentes principais aplicado ao conjunto de variáveis de entrada.

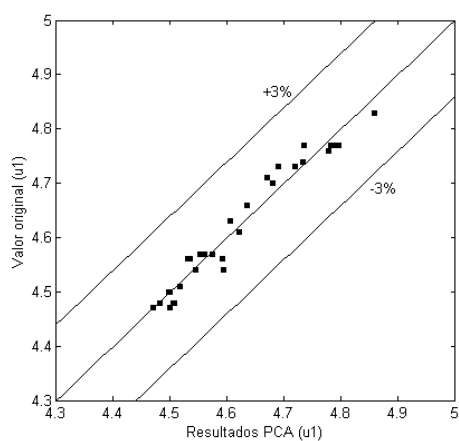
Tabela 3.9

Principais componentes das variáveis u_1 , u_2 , u_3 , u_4 e u_5 .

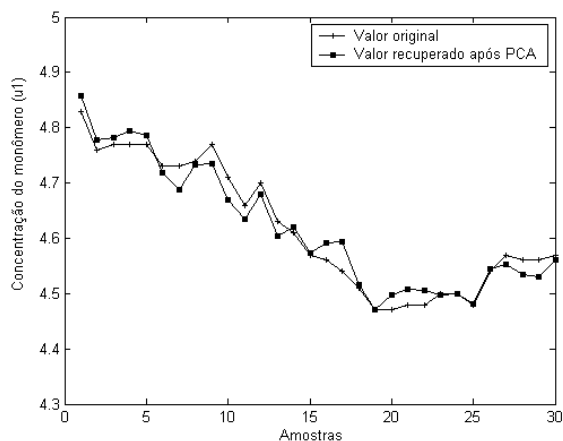
Componentes	Contribuição de cada componente (%)
1 ^a	42,178
2 ^a	26,458
3 ^a	20,819
4 ^a	8,766
5 ^a	1,7792
Total	100

Pode-se notar pela tabela 3.9 que 98,2% de toda a informação das variáveis de entrada (u_1 , u_2 , u_3 , u_4 e u_5) estão contidas nas 4 primeiras componentes principais. Neste caso vamos considerar ε_{PCA} igual a 0,02, ou seja, 98% da informação mantida.

A figura 3.17 apresenta os resultados obtidos após o processo de recuperação (reprodução) do sinal u_1 (concentração do monômero) da planta química. A figura 3.17a mostra que o sinal recuperado está dentro de $\pm 3\%$ do sinal original e a figura 3.17b mostra os valores originais e recuperados após o processo PCA.



(a)



(b)

Figura 3.17 - Precisão após aplicação do PCA: (a) Sinal recuperado está dentro de $\pm 3\%$ do sinal original; (b) Valores originais e recuperados.

Pode-se concluir que o PCA após aprender as correlação entre os sinais conseguiu recuperar o sinal de entrada com uma pequena perda de informação.

Para verificar a robustez do PCA para valores não treinados foi introduzida uma falha em um dos sinais de entrada após o processamento realizado pelo PCA. Suponhamos que o sinal u_1 do exemplo teve uma falha do tipo “mudança brusca” para 80% do seu valor nominal, a partir da amostra 10 do conjunto de dados. Isto significa uma degradação no sensor de medida da concentração para fora do intervalo processado pelo PCA.

A figura 3.18 apresenta os resultados obtidos após o processo de recuperação (reprodução) do sinal u_1 (concentração do monômero) da planta química. A figura 3.18a mostra que o sinal recuperado não está mais dentro de aproximadamente $\pm 3\%$ do sinal original e a figura 3.18b mostra que os valores originais e recuperados após o processo PCA estão bem diferentes, indicando que não foi possível recuperar o sinal original.

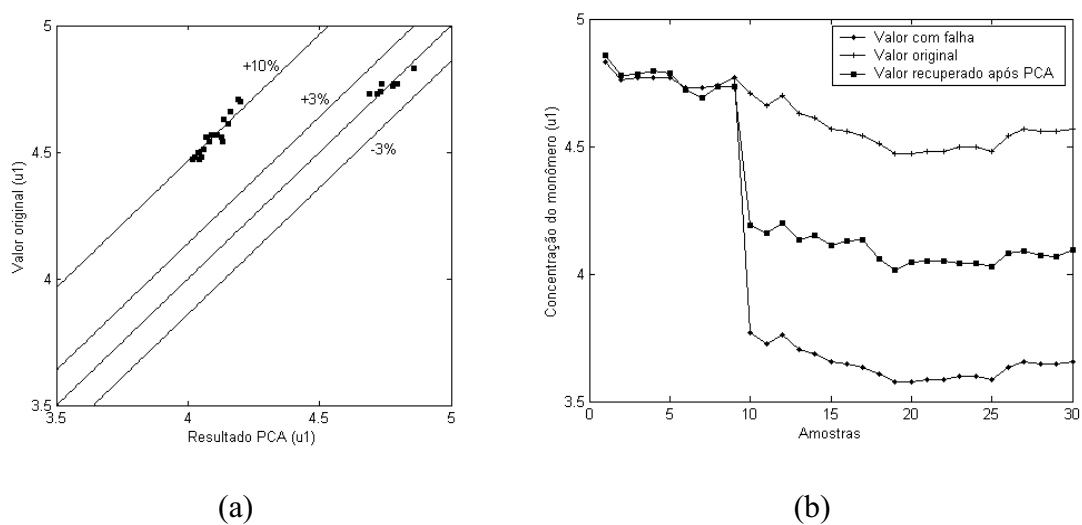


Figura 3.18 - Precisão após aplicação do PCA: (a) Sinal recuperado está fora de $\pm 3\%$ do sinal original após a falha; (b) Valores falhos, originais e recuperados.

O PCA não consegue recuperar o sinal original da concentração do monômero (u_1) a partir das informações fornecidas pelas outras variáveis de entrada. Isto é devido ao fato que o PCA é sensível a valores fora do intervalo das variáveis de entrada usadas no processamento do PCA. Uma variável fora do intervalo utilizado no processamento PCA causa uma deformação em todo o domínio transformado, como pode ser visto na figura 3.19.

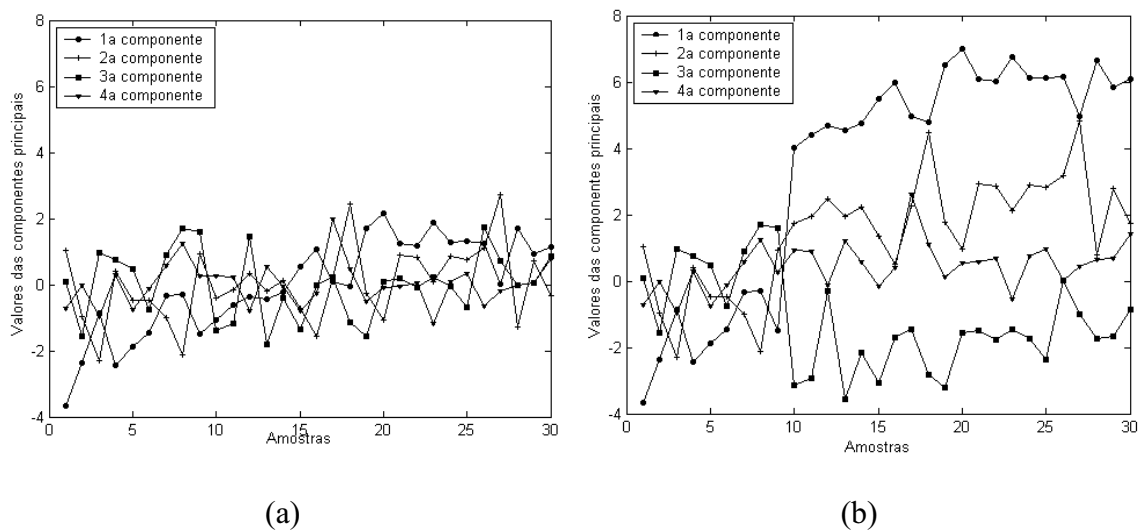


Figura 3.19 - Componentes da transformação PCA: (a) Obtidas pelo mapeamento das entradas originais; (b) Obtidas com a inserção da falha.

3.4 Separação do espaço de entrada em classes

Como foi citado no capítulo 2 os métodos de separação em classes não hierárquicos têm por objetivo classificar as amostras através de uma partição simples em C classes, onde C é o número de classes especificado previamente ou determinado pelo algoritmo.

Como o problema de validação de sinal não necessita encontrar uma hierarquia entre todas as amostras de entrada, mais sim uma partição das amostras em classes, nossa atenção será focada nos algoritmos não hierárquicos.

Os métodos mais conhecidos de separação em classes são: a separação simples, a nebulosa e a possibilística. Aplicações em validação de sinal (OLIVEIRA, 1999) mostraram a superioridade da classificação nebulosa sobre as classificações simples e possibilística.

O algoritmo FCM é descrito no anexo V. O objetivo do algoritmo FCM é encontrar uma partição nebulosa com C classes, onde a função-objetivo dada pela equação (2.35) é minimizada.

De modo a avaliar a validade da partição obtida, após o processo de separação nebulosa, foi usado um critério normalmente utilizado (WINDHAM, 1982) para avaliar partições dado por

$$F_c(U) = \frac{1}{N} \sum_{i=1}^C \sum_{k=1}^N \mu_{ik}^m \quad (3.8)$$

onde U é a matriz de partição nebulosa, N é o número de amostras disponíveis, C é o número de centróides e μ_{ik} é a pertinência da amostra k na classe i .

O valor de $F_c(U)$ é chamado de validade da partição U em C classes nebulosas. Quando é obtida uma partição rígida, que é considerada um resultado ideal, $F_c(U)=1$. Quando é obtida uma partição nebulosa, $F_c(U) < 1$. Em geral, quanto maior $F_c(U)$ melhor é a partição.

A seguir são apresentados os ensaios de separação em classes realizados em diversos conjuntos de dados usando a classificação nebulosa para as métricas Euclidiana e Mahalanobis. Os métodos ensaiados foram o FCM e o EFCM.

Em todos os ensaios realizados o critério de parada para os modelos foi feito igual a 0,001 ($\varepsilon=0,001$) e grau de nebulosidade foi feito igual a 2 ($m=2$).

3.4.1 Influência das métricas Euclidiana e Mahalanobis

Estes ensaios têm por objetivo verificar o efeito das métricas Euclidiana e Mahalanobis no desempenho do método FCM. Para facilitar a compreensão, supõe-se que seja conhecido o número de classes C contidas no conjunto de dados.

3.4.1.1 Conjunto de dados em forma de borboleta

Nesta aplicação são utilizadas amostras de dados que estão distribuídas no espaço Euclidiano numa forma que lembra as asas de uma borboleta. A tabela 3.10 apresenta nas suas duas primeiras colunas o conjunto de dados chamado borboleta que contém 15 amostras bi-dimensionais.

O espaço foi considerado como tendo duas classes ($C=2$).

A figura 3.20 mostra os centróides obtidos das classes (\times), os pontos obtidos na classe C_1 (o) e na classe C_2 (+) e as distâncias médias ponderada pelas pertinências das amostras em cada classe (círculo ou elipse), após o processo de separação em classes. Para a distância Euclidiana, figura 3.20a, os valores dos centróides das classes obtidos foram $C_1=(0,8548; 2,0000)$ e $C_2=(5,1452; 2,0000)$ e o valor obtido da validade da partição foi de $F_2(U)=0,8429$. Para a distância Mahalanobis, figura 3.20b, os centróides das classes obtidos, $C_1=(0,8166; 2,0000)$ e $C_2=(5,1836; 2,0000)$, foram quase iguais aos que foram obtidos com a métrica Euclidiana. Contudo, a validade da partição obtida com a métrica Mahalanobis foi de $F_2(U)=0,8714$. Embora o valor da validade da partição não tenha aumentado muito, porque os graus de pertinências são elevados ao

quadrado na equação de medida de validade, a partição se tornou mais rígida, conforme resultados apresentados na tabela 3.10.

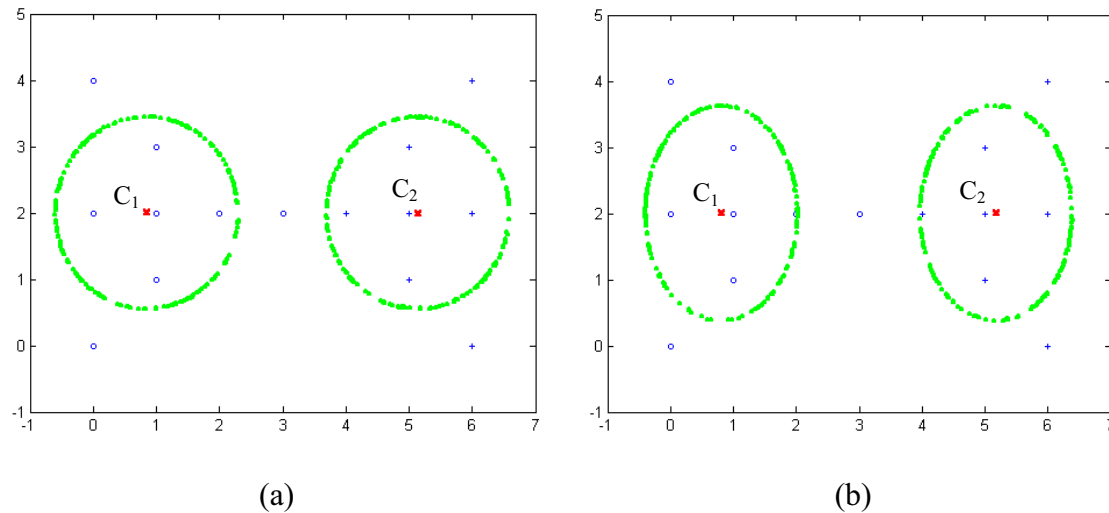


Figura 3.20 - Separação em classes do conjunto de dados tipo borboleta: (a) Com métrica Euclidiana; (b) Com métrica Mahalanobis.

Tabela 3.10

Conjunto de amostras e pertinências nas classes C_1 e C_2 das amostras tipo borboleta.

k	x_k	Métrica Euclidiana		Métrica Mahalanobis	
		C_1	C_2	C_1	C_2
1	(0,0)	0,8656	0,1344	0,9098	0,0902
2	(0,2)	0,9731	0,0269	0,9758	0,0242
3	(0,4)	0,8656	0,1344	0,9098	0,0902
4	(1,1)	0,9468	0,0532	0,9685	0,0315
5	(1,2)	0,9988	0,0012	0,9981	0,0019
6	(1,3)	0,9468	0,0532	0,9685	0,0315
7	(2,2)	0,8829	0,1171	0,8786	0,1214
8	(3,2)	0,5000	0,5000	0,5000	0,5000
9	(4,2)	0,1171	0,8829	0,1215	0,8785
10	(5,1)	0,0532	0,9468	0,0315	0,9685
11	(5,2)	0,0012	0,9988	0,0019	0,9981
12	(5,3)	0,0532	0,9468	0,0315	0,9685
13	(6,0)	0,1344	0,8656	0,0902	0,9098
14	(6,2)	0,0269	0,9731	0,0242	0,9758
15	(6,4)	0,1344	0,8656	0,0902	0,9098

3.4.1.2 Conjunto de dados com formas diversas

Nesta aplicação são utilizadas 30 amostras de dados bi-dimensionais que estão distribuídas no espaço Euclidiano.

O espaço foi considerado como tendo três classes ($C = 3$).

A figura 3.21 mostra os centróides obtidos das classes (\times), os pontos obtidos na classe C_1 (o), na classe C_2 (+) e na classe C_3 (\blacklozenge) e as distâncias médias ponderada pelas pertinências das amostras em cada classe (circulo ou elipse), após o processo de separação em classes, para a distância Euclidiana e para a distância Mahalanobis.

Para métrica Euclidiana foi obtida uma validade da partição de $F_3(U)=0,8756$. Já para a métrica Mahalanobis, a validade da partição obtida foi de $F_3(U)=0,9274$.

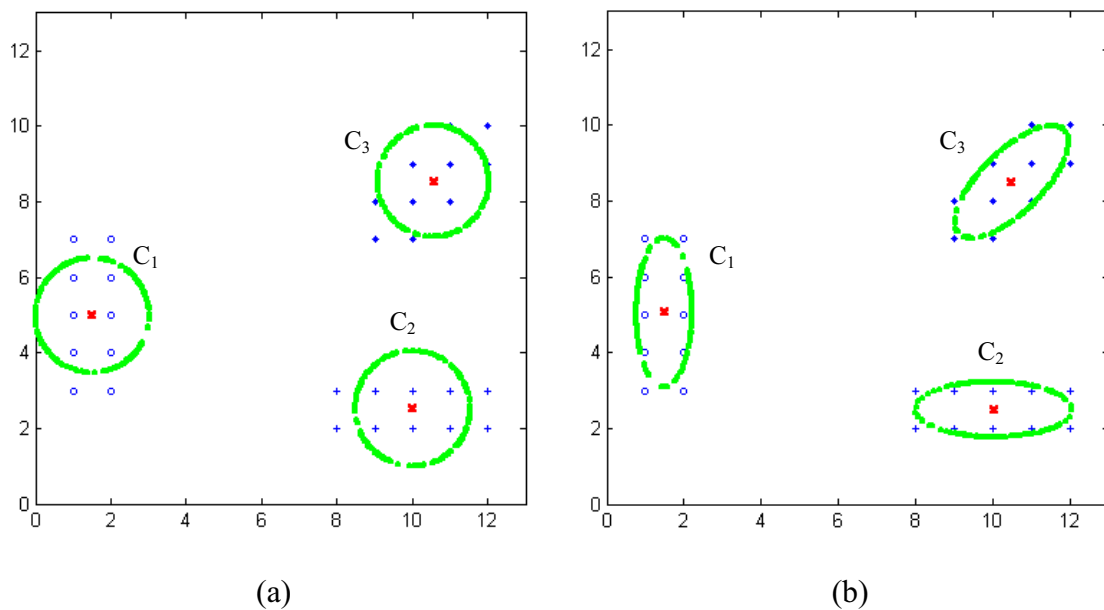


Figura 3.21 - Separação em classes do conjunto com formas diversas: (a) Com métricas Euclidiana; (b) Com métrica Mahalanobis.

A tabela 3.11 apresenta os valores dos centróides reais e obtidos pelos processos de separação em classes. A partição nebulosa usando a métrica Mahalanobis é mais

rígida do que a com métrica Euclidiana. Este fato faz com que diminua a atração dos centróides das classes por amostras que não pertencem a eles e, conseqüentemente, os valores obtidos de centróides pela métrica Mahalanobis se aproximam dos valores reais.

Tabela 3.11

Centróides reais e obtidos no processo de separação em classes.

Classe	Centróide (real)	Centróide (métrica Euclidiana)	Centróide (métrica Mahalanobis)
C ₁	(1,5000; 5,0000)	(1,5138; 4,9980)	(1,4956; 5,0611)
C ₂	(10,0000; 2,5000)	(10,0161; 2,5297)	(10,0329; 2,4983)
C ₃	(10,5000; 8,5000)	(10,5610; 8,5499)	(10,5075; 8,5061)

3.4.2 Determinação do número de classes

Como já foi citado, para a maioria das aplicações a quantidade de classes que o espaço deve ser dividido não é conhecida. A determinação do número de classes é uma das maiores discussões na técnica de separação em classes e existem diversos estudos sobre esta questão.

A seguir são apresentados os ensaios realizados, com propósito de determinar a validade de uma separação em classes, nos quatro critérios descritos no capítulo 2. As partições obtidas foram avaliadas pelas equações (2.35), (2.36) e (2.37) correspondendo, respectivamente, aos critérios de SUGENO *et al.* (1993), KRISHNAPURAM *et al.* (1993) e XIE *et al.* (1991). A minimização do número de classes é feita de maneira a $J(L,U)$ alcançar um valor mínimo em relação a L , a medida que o número de classes C aumenta. O primeiro ponto de mínimo encontrado, normalmente um mínimo local, é adotado pelo método.

Com o propósito de determinar o número de classes dos conjuntos de dados para o algoritmo FCM o número de classes foi variado várias vezes de 2 até 10.

O algoritmo EFCM foi inicializado com 10 classes ($M=10$) e com limiar adaptativo dado pela equação (2.41).

3.4.2.1 Comparação do EFCM com o FCM usando critérios de validação de classes

Nesta aplicação é comparado o desempenho do algoritmo de extensão de classes EFCM com o de separação em classes para o FCM pelos três critérios descritos anteriormente, com a finalidade de descobrir as estruturas escondidas no conjunto de dados. O conjunto de dados utilizado contém quatro grupo de dados bi-dimensionais que foram gerados aleatoriamente com uma distribuição normal em torno de quatro centróides com desvios padrão dados na tabela 3.12. O grupo 1 contém 150 amostras e os outros três grupos contém 50 amostras cada. O objetivo é detectar as classes que refletem as estruturas escondidas nos dados. Tendo em vista que as classes geradas são teoricamente esféricas foram comparados somente os algoritmos FCM e EFCM utilizando a métrica Euclidiana.

Tabela 3.12

Centróides dos grupos de dados (x, y), variância (σ_x^2, σ_y^2) e número de amostras.

Grupo	Grupo de dados		Número de amostras	
	Centróide	Variância	Primeiro caso	Segundo caso
1	(-0,5; -0,4)	(0,04; 0,04)	150	150
2	(0,1; 0,2)	(0,01; 0,01)	50	30
3	(0,5; 0,7)	(0,01; 0,01)	50	30
4	(0,6; -0,3)	(0,04; 0,04)	50	50

A figura 3.22a apresenta os resultados obtidos pelo FCM usando a métrica Euclidiana para o número de amostras do primeiro caso. Neste caso, pode-se observar que os critérios de Sugeno-Yasukawa e Xie-Beni detectaram corretamente o número de classes do conjunto de dados.

O algoritmo EFCM foi aplicado ao conjunto de dados do primeiro caso, com inicialização aleatória usando 10 classes, e detectou quatro grupos no conjunto de dados. A figura 3.22b apresenta os centróides dos grupos (o), a posição final dos protótipos das classes (\times), os pontos obtidos em cada classe (o), (+), (\blacklozenge) e (\ast) e as distâncias médias ponderada pelas pertinências das amostras em cada classe (círculo).

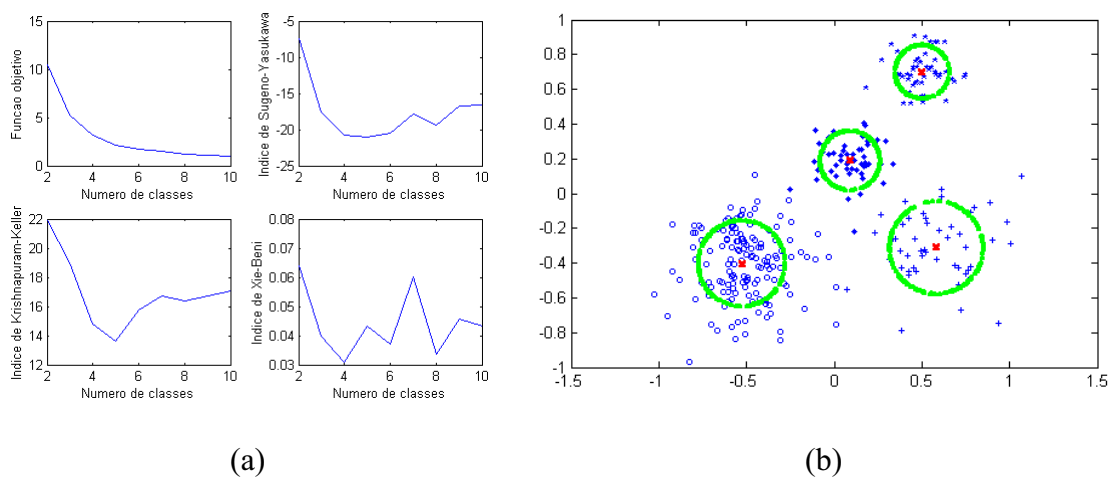


Figura 3.22 - Resultado obtidos para o conjunto de dados: (a) Dois dos três critérios de validação indicam que existem 4 classes para o algoritmo FCM; (b) O algoritmo EFCM detecta corretamente o número de classes.

No segundo caso, o mesmo exemplo foi repetido com o número de dados dos grupos dois e três reduzidos de 50 para 30 amostras. O algoritmo FCM usando os critérios de validação do número de classes falhou em determinar o número correto de estruturas nos dados, conforme mostrado na figura 3.23a. Contudo, o algoritmo EFCM por ser menos sensível à distribuição dos dados devido ao protótipos de classes com volume detectou corretamente os quatro grupos presentes nos dados, conforme mostrado na figura 3.23b.

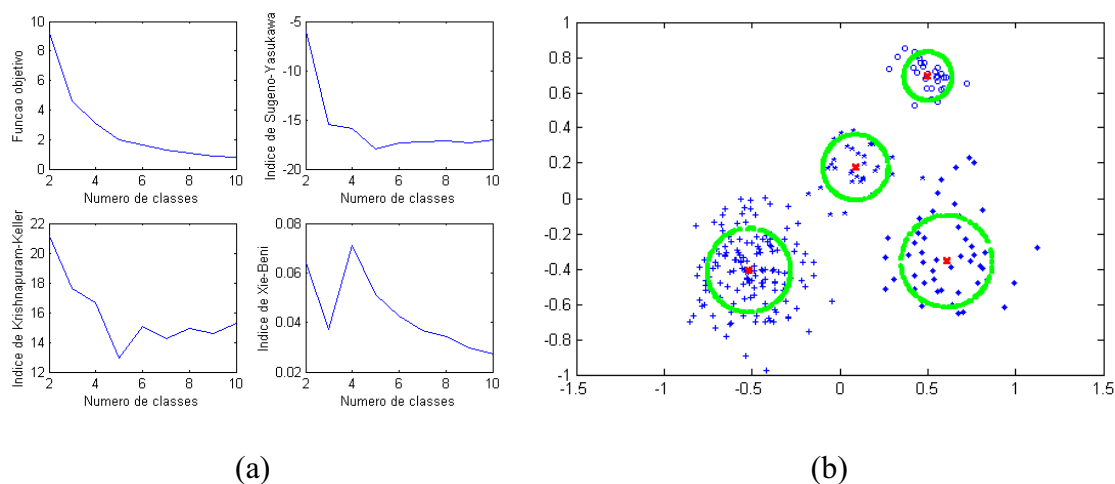


Figura 3.23 - Resultado obtidos para o conjunto de dados: (a) Os critérios de validação não conseguem detectar corretamente o número de classes para o algoritmo FCM; (b) O algoritmo EFCM detecta corretamente o número de classes.

3.4.2.2 Influência da inicialização

Para estudar a influência da inicialização nos algoritmos FCM e EFCM os dados do segundo caso da seção 3.4.2.1 foram separados em classes 1000 vezes com partições inicializadas aleatoriamente. O algoritmo FCM foi ajustado para particionar os dados em quatro classes, enquanto que o algoritmo EFCM foi inicializado com 10 classes. Após cada execução foram registrados os centróides das classes. A tabela 3.13 mostra o valor médio dos centróides das classes e o desvio padrão de cada coordenada do centróide das classes para os 1000 experimentos. Pode-se observar que os centróides das classes encontrados pelo algoritmo EFCM estão mais próximos aos centróides reais do que os encontrados pelo algoritmo FCM. Além disso, os desvios padrão dos centróides são muito menores para o algoritmo EFCM. O algoritmo FCM teve uma maior dificuldade em encontrar o centróide do grupo 2, que parece não ser detectado se a inicialização não for boa. Deste modo, o centróide da classe médio está bem longe de qualquer centróide de classe real, e os desvios padrão das coordenadas do centróide são

muito grandes. O algoritmo EFCM se mostrou muito mais robusto à inicialização da partição. De fato, o nível de discriminação de similaridade α tem um maior impacto no algoritmo do que a inicialização. Isto é esperado tendo em vista que a fusão de muitas ou poucas classes irá mudar significativamente as coordenadas das classes restantes.

Tabela 3.13 - Valor médio dos centróides das classes e o desvio padrão de cada coordenada do centróide das classes encontrados para o FCM e o EFCM.

Grupo	Centróide real	Centróide FCM		Centróide EFCM	
		Média	Desvio padrão	Média	Desvio padrão
1	(-0,5; -0,4)	(-0.54; -0.47)	(0.009; 0.031)	(-0,51; -0,41)	(0,003; 0,005)
2	(0.1; 0,2)	(-0.29; -0.14)	(0.193; 0.147)	(0,09; 0,18)	(0,002; 0,003)
3	(0,5; 0,7)	(0.46; 0.62)	(0.022; 0.029)	(0,50; 0,69)	(< 10 ⁻⁴ ; 0,001)
4	(0,6; -0,3)	(0.59; -0.34)	(0.009; 0.020)	(0,61; -0,36)	(0,004; 0,002)

3.4.2.3 Detecção de linhas

Nesta aplicação é comparado o desempenho do FCM e do EFCM para a métrica Mahalanobis, para descobrir estruturas de várias formas em um conjunto de dados. O conjunto de dados é formado por cinco grupos de dados lineares bi-dimensionais com 100 amostras cada.

A figura 3.24 apresenta os resultados obtidos pelos métodos FCM e EFCM usando a métrica Mahalanobis. Neste caso, pode-se observar pela figura 3.24a que os critérios de validação do número de classes de Sugeno e Krishinapuram detectaram corretamente o número de classes do conjunto de dados. A figura 3.24b apresenta os resultados obtidos pelo FCM e EFCM. Para o FCM o conjunto de dados foi considerado como tendo 5 classes. O EFCM foi aplicado ao conjunto de dados, com inicialização aleatória usando 10 classes, e detectou também os cinco grupos no conjunto de dados.

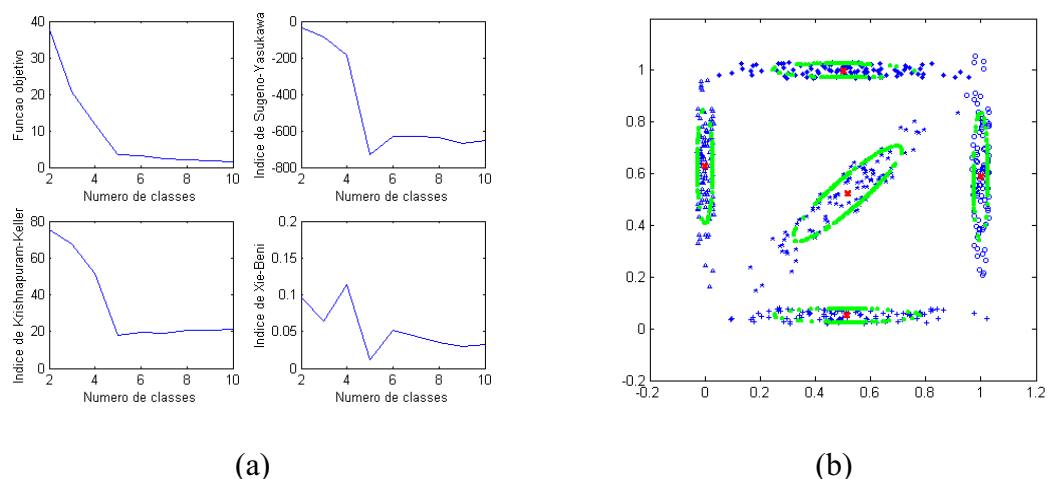


Figura 3.24 - Resultados obtidos usando a métrica Mahalanobis: (a) Dois dos três critérios de validação indicam que existem 5 classes para o algoritmo FCM; (b) Forma das 5 classes obtidas pelos algoritmos FCM e EFCM.

A tabela 3.14 mostra as coordenadas reais dos centróides das classes e as coordenadas dos centróides obtidas pelo FCM e EFCM.

Tabela 3.14 - Coordenadas reais dos centróides das classes e as coordenadas dos centróides obtidas pelo FCM e EFCM.

Grupo	Centróide real	Centróide FCM	Centróide EFCM
1	(0,500; 0,050)	(0,516; 0,051)	(0,513; 0,051)
2	(0,000; 0,600)	(0,001; 0,629)	(0,001; 0,629)
3	(1,000; 0,600)	(1,000; 0,588)	(1,001; 0,595)
4	(0,500; 0,500)	(0,517; 0,521)	(0,515; 0,519)
5	(0,500; 1,000)	(0,503; 0,998)	(0,505; 0,998)

3.4.2.4 Conjunto de dados da flor IRIS

Nesta aplicação é comparado o desempenho do FCM e do EFCM para as métricas Euclidiana e Mahalanobis para descobrir estruturas escondidas no conjunto de dados

IRIS. O conjunto IRIS contém três classes de flores: a IRIS sedosa, a IRIS versicolor e a IRIS virgínica. Para um conjunto de 50 flores em cada classe foi medido o comprimento e a largura da sépala e o comprimento e a largura da pétala. O objetivo é automaticamente detectar classes nos dados que reflitam a estrutura escondida. Tendo em vista que não sabemos a forma das classes vamos aplicar os algoritmos FCM e EFCM para as métricas Euclidiana e Mahalanobis.

Para a validação das classes do algoritmo FCM o número de classes foi variado de 2 até 10.

A figura 3.25 apresenta os resultados obtidos pelo FCM usando as métricas Euclidiana e Mahalanobis. Para o caso do FCM com métrica Euclidiana, figura 3.25a, foram obtidos 3 classes para os critérios de Sugeno-Yasukawa e Krishnapuram-Keller e 2 classes para o critério de Xie-Beni. Para o caso do FCM com métrica Mahalanobis, figura 3.25b, foram obtidos 4 classes para os critérios de Sugeno-Yasukawa e Xie-Beni e 2 classes para o critério de Krishnapuram-Keller.

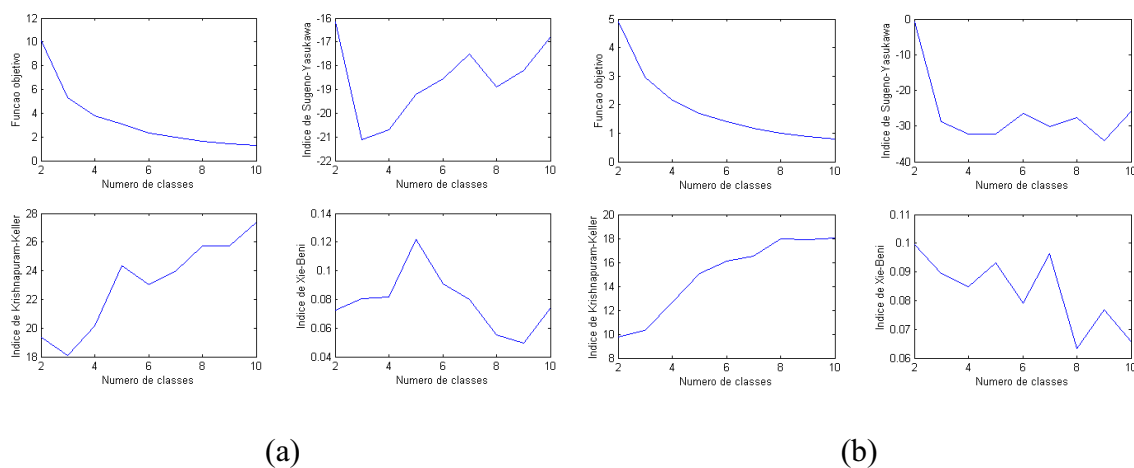


Figura 3.25 – Função-objetivo e número de classes obtidas pelos critérios de Sugeno-Yasukawa, Krishnapuram-Keller e Xie-Beni para o FCM: (a) Com métrica Euclidiana; (b) Com métrica Mahalanobis.

A tabela 3.15 apresenta um resumo dos resultados obtidos pelos métodos FCM e EFCM para o conjunto de dados IRIS.

Tabela 3.15 - Resultados obtidos pelos método FCM e EFCM para o conjunto de dados IRIS.

Método		Distância Euclidiana			Distância Mahalanobis		
		Classes ^a	Erros ^b	$F_c(U)$	Classes ^a	Erros ^b	$F_c(U)$
FCM	Sugeno-Yasukawa	3	18/150	0,7767	4	47/150	0,6185
	Krishnapuram-Keller	3	18/150	0,7767	2	65/150	0,7456
	Xie-Beni	2	53/150	0,8935	4	47/150	0,6185
EFCM		3	18/150	0,8481	3	10/150	0,8764

(a) Número de classes obtidas.

(b) Número de amostras classificadas erradas em relação ao número total de amostras.

Os critérios de determinação do número de classes utilizados para o FCM não conseguiram obter o número correto de classes do conjunto de dados ($C=3$), principalmente para a métrica Mahalanobis. Esta deficiência pode ser devida ao fato de que esses critérios foram criados para aplicações com métrica Euclidiana.

Para a métrica Euclidiana os métodos FCM e EFCM obtiveram 18 classificações errôneas nas 3 classes do conjunto de dados, porém, o método EFCM apresentou um índice maior de eficiência da partição de $F_3(U)=0,8481$.

O método EFCM nesta aplicação teve melhor desempenho do que o FCM, obtendo 10 classificações errôneas para a métrica Mahalanobis, com um índice de eficiência da partição de $F_3(U)=0,8764$.

3.5 Modelo de validação de sinal

A seleção do modelo a ser utilizado pelo sistema de validação de sinal deve levar em conta o tipo de aplicação a que se destina o sistema. As técnicas de inteligência artificial através de redes neurais e sistemas de inferência nebulosa adaptativo têm sido

aplicadas recentemente na validação de sinal por não necessitarem de especialistas da planta para geração e treinamento do sistema. Esses modelos “aprendem” o comportamento dos sensores a serem modelados através de um conjunto de dados de operação da planta.

A aplicação de redes neurais auto-associativas à validação de sinal pode ser encontrada em FANTONI *et al.* (1996) e OLIVEIRA (1999). A seguir é apresentado a aplicação de um sistema de inferência neuro-nebulosa adaptativo (ANFIS) na validação de sinal de uma planta química.

Antes de utilizar um sistema de inferência nebulosa em validação de sinal é necessário efetuar o seu treinamento. O sistema de inferência nebulosa é treinado com pares entrada/saída usando um algoritmo de treinamento híbrido, isto é, os parâmetros da parte antecedente e conseqüente são treinados usando procedimentos completamente diferentes. O processo de treinamento permite ao sistema ajustar seus parâmetros para aprender as relações de entrada/saída contida no conjunto de dados. A princípio qualquer algoritmo de otimização pode ser utilizado para treinamento do modelo. Os principais métodos, até então utilizados para treinamento desses modelos são: método dos mínimos quadrados, método de gradiente descendente e algoritmos genéticos.

A seguir são comparados três diferentes métodos de treinamento de um ANFIS:

1. GD+LSE: algoritmo de retropropagação (*backpropagation*) usando gradiente descendente (GD) para treinar os parâmetros antecedentes e estimação por mínimos quadrados (LSE - *Least-Square Estimation*) para determinar os parâmetros conseqüentes;
2. GA+LSE: algoritmo genético (GA) para treinar os parâmetros antecedentes e estimação por mínimos quadrados (LSE) para determinar os parâmetros conseqüentes;

3. PSO+LSE: otimização por enxame de partículas (PSO) para treinar os parâmetros antecedentes e estimação por mínimos quadrados (LSE) para determinar os parâmetros conseqüentes.

Para os três métodos de treinamento um passo no procedimento de aprendizado é dividido em duas etapas: Na primeira etapa do ciclo de treinamento os padrões de entrada (amostras) são propagados para a saída do FIS, e os parâmetros “ótimos” da parte conseqüente são estimados pelo método dos mínimos quadrados, enquanto que os parâmetros da parte antecedente (funções pertinências) são mantidos fixos. Na segunda etapa os padrões são novamente propagados e, neste momento, é usado retropropagação, algoritmo genético ou otimização por enxame de partículas para modificar os parâmetros antecedentes, enquanto que os parâmetros conseqüentes permanecem fixos. Este procedimento híbrido é, então, repetido iterativamente até que o erro de estimação do FIS seja reduzido a um valor desejado ou até que seja alcançado um número máximo de ciclos de treinamento.

O anexo III desta tese apresenta um resumo sobre as técnicas de otimização por mínimos quadrados, gradiente descendente, algoritmos genéticos e enxame de partículas.

3.5.1 ANFIS aplicado a uma planta química

A seguir apresentamos a aplicação da metodologia de construção e treinamento considerando um único ANFIS realizando o mapeamento das variáveis de entrada na variável de saída de uma planta química (<http://neural.cs.nthu.edu.tw/jang/benchmark/>). Neste exemplo, são avaliados os métodos três de treinamento do ANFIS citados anteriormente.

Nesta aplicação são utilizados quatro sinais monitorados da planta química, que foram selecionados conforme metodologia apresentada no item 3.1 desta tese, a saber:

- u_1 : concentração do monômero;
- u_2 : troca na concentração do monômero;
- u_3 : vazão do monômero;
- y : ponto de ajuste da vazão do monômero.

onde os sinais u_1 , u_2 e u_3 são os sinais a serem aplicados a entrada do ANFIS e o sinal y é o sinal a ser validado.

Nesta aplicação foram usadas 70 amostras das quatro variáveis da operação real da planta. Este conjunto de dados foi usado como conjunto de treinamento do ANFIS. Um conjunto para validação do treinamento, contendo 70 amostras, foi gerado a partir dos dados da planta através da adição de um sinal de ruído gaussiano, com distribuição $N(0;0,01)$, a esses sinais. O conjunto de validação é usado durante o processo de treinamento para monitorar a habilidade do sistema nebuloso de generalizar (mesmo princípio que a validação cruzada do treinamento usada nas redes neurais). Os testes do sistema foram realizados com todo o conjunto de dados, isto é, as 140 amostras (70 originais da planta e 70 dos sinais com a adição do ruído).

O número de regras a serem utilizadas no ANFIS foi feito igual ao número de classes contidas nos dados de operação da planta (as 70 amostras do conjunto de treinamento). Para separar os dados em classes foi usado o algoritmo EFCM com métrica Mahalanobis. Nesta aplicação foi usado o critério de validação do número de classes proposto por KAYMAK *et al.* (2000), conforme definido no item 2.2.6.4 desta tese.

Os sinais da planta foram normalizados antes de se efetuar a separação em classes e o treinamento do ANFIS. Deste modo, a separação em classes e o treinamento são realizados em um hipercubo $[0,1]^n$, onde n é o número de sinais (neste caso $n=4$).

Os valores utilizados dos parâmetros do algoritmo EFCM com métrica Mahalanobis foram os seguintes: número inicial de classes $M^{(0)}=10$; grau de nebulosidade $m=2$; critério de terminação $\varepsilon < 0.001$; e limiar adaptativo α dado por $\alpha^{(l)} = 1/(M^{(l-1)} - 1)$, onde l é a iteração atual do processo de otimização. O processo de separação em classes, após a convergência do algoritmo EFCM com métrica Mahalanobis, encontrou 4 classes no conjunto de amostras das quatro variáveis da planta química.

Para esse conjunto de variáveis foi gerado um ANFIS inicial com 4 regras, como definido pelo algoritmo EFCM. Conseqüentemente, foram geradas 4 funções pertinência do tipo gaussiana para cada entrada, onde o centróide de cada função-pertinência é inicializado com o corresponde centróide do classe nebulosa obtida pelo processo de separação em classes. A estrutura utilizada no ANFIS é descrita no capítulo 4 desta tese.

O ANFIS inicial foi treinado e testado. Como já foi citado, para o conjunto de treinamento foram utilizadas as 70 amostras do conjunto de dados e a capacidade de generalização do ANFIS foi verificada pelas 70 amostras geradas com adição do ruído.

Neste exemplo, a função de otimização que avalia a capacidade de cada indivíduo (no caso o ANFIS obtido em cada passo de treinamento) se ajustar aos objetivos desejados é dada por

$$E = \frac{1}{N} \sum_{k=1}^N (y(k) - \hat{y}(k))^2 \quad (3.9)$$

onde E é o erro médio quadrado, $y(k)$ e $\hat{y}(k)$ são, respectivamente, o sinal medido e o sinal estimado pelo ANFIS e N é o número de amostras.

As principais condições para treinar o ANFIS usadas por cada método são:

- a) O algoritmo de treinamento GD+LSE usado nesse exemplo é uma função do MATLAB (FLTbx, 2000) que executa uma rotina de treinamento para FIS do tipo Sugeno com as seguintes opções de treinamento: erro de treinamento desejado=0; tamanho do passo inicial=0,01; taxa de decréscimo do passo=0,9; taxa de acréscimo do passo=1.1;
- b) O algoritmo de treinamento GA+LSE é uma função implementada no ambiente do MATLAB onde os parâmetros da parte antecedente do ANFIS são sintonizados por um GA clássico (GEATbx, 2000) com as seguintes opções de treinamento: cruzamento=0,6; mutação=0,02; seleção usando ranqueamento e elitismo (melhor indivíduo);
- c) O algoritmo de treinamento PSO+LSE é uma função implementada no ambiente do MATLAB onde os parâmetros da parte antecedente do ANFIS são sintonizados por um PSO clássico (TRELEA, 2003) com algoritmo do tipo canônico. O fator momento \bar{a} tem todos os elementos iguais a a , onde a é dado por

$$a = \frac{(a_i - a_f)(N - k)}{N} + a_f \quad (3.10)$$

onde N é o número total de iterações, k é o número da iteração atual, a_i é o fator de momento inicial ($a_i=0.7$) e a_f é o fator de momento final ($a_f=0.1$). Do mesmo modo, os coeficientes de atração \bar{b}_1 e \bar{b}_2 têm todos os seus elementos iguais a b ($b=1.5$). Todos os elementos dos coeficientes de posição \bar{c} e velocidade \bar{d} foram feitos iguais a 1.

A figura 3.26 apresenta os resultados médios, para 5 execuções, produzidos pelo ANFIS para os conjuntos de treinamento e teste, para os três métodos de otimização. Os testes foram realizados para um número máximo de 1000 épocas de treinamento.

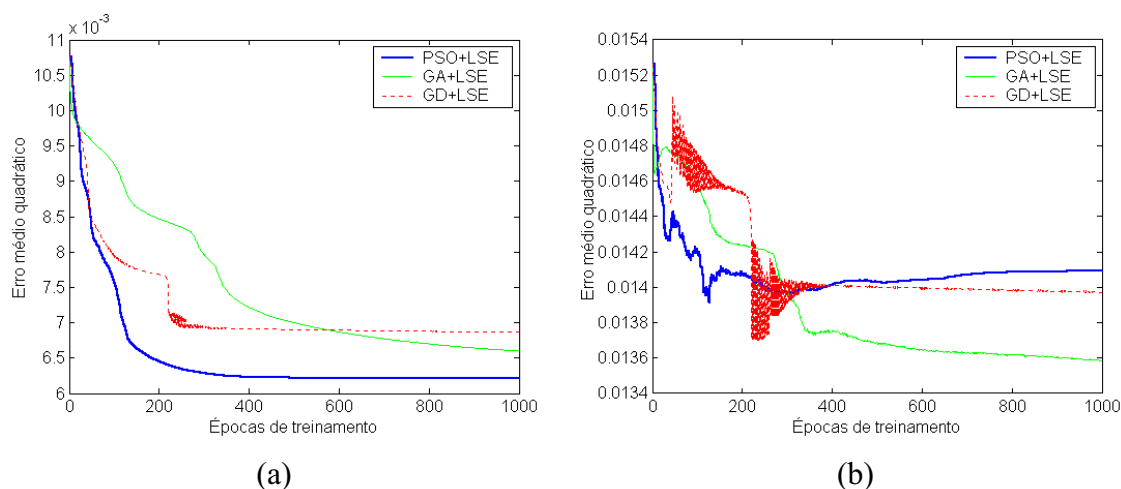


Figura 3.26 - Resultados obtidos do treinamento do ANFIS para a planta química para:

(a) Conjunto de treinamento; (b) Conjunto de teste.

Os resultados obtidos mostram que o método de otimização por PSO obteve um erro médio quadrático de estimação para o conjunto de treinamento menor do que os métodos de otimização por GD e GA. Por outro lado, no conjunto de teste o método de otimização por GA obteve melhor desempenho, indicando uma maior capacidade de generalização para esta aplicação.

Resultados obtidos para aplicações em planta nucleares de potência mostraram (OLIVEIRA, 2005) a superioridade do método de otimização por PSO tanto na diminuição do erro de estimação do ANFIS para o conjunto de treinamento quanto na maior capacidade de generalização para o conjunto de teste.

A figura 3.27 apresenta a estrutura obtida do ANFIS para as variáveis da planta química antes e após ao processo de treinamento pelos três métodos de otimização.

A figura 3.28 apresenta o ponto de ajuste da vazão do monômero estimado e o valor absoluto do erro de estimação antes (ANFIS inicial) e após os processos de otimização pelo algoritmos GD+LSE, GA+LSE e PSO+LSE para as variáveis da planta química e a figura 3.29 apresenta o histograma desses erros de estimação para facilitar sua visualização.

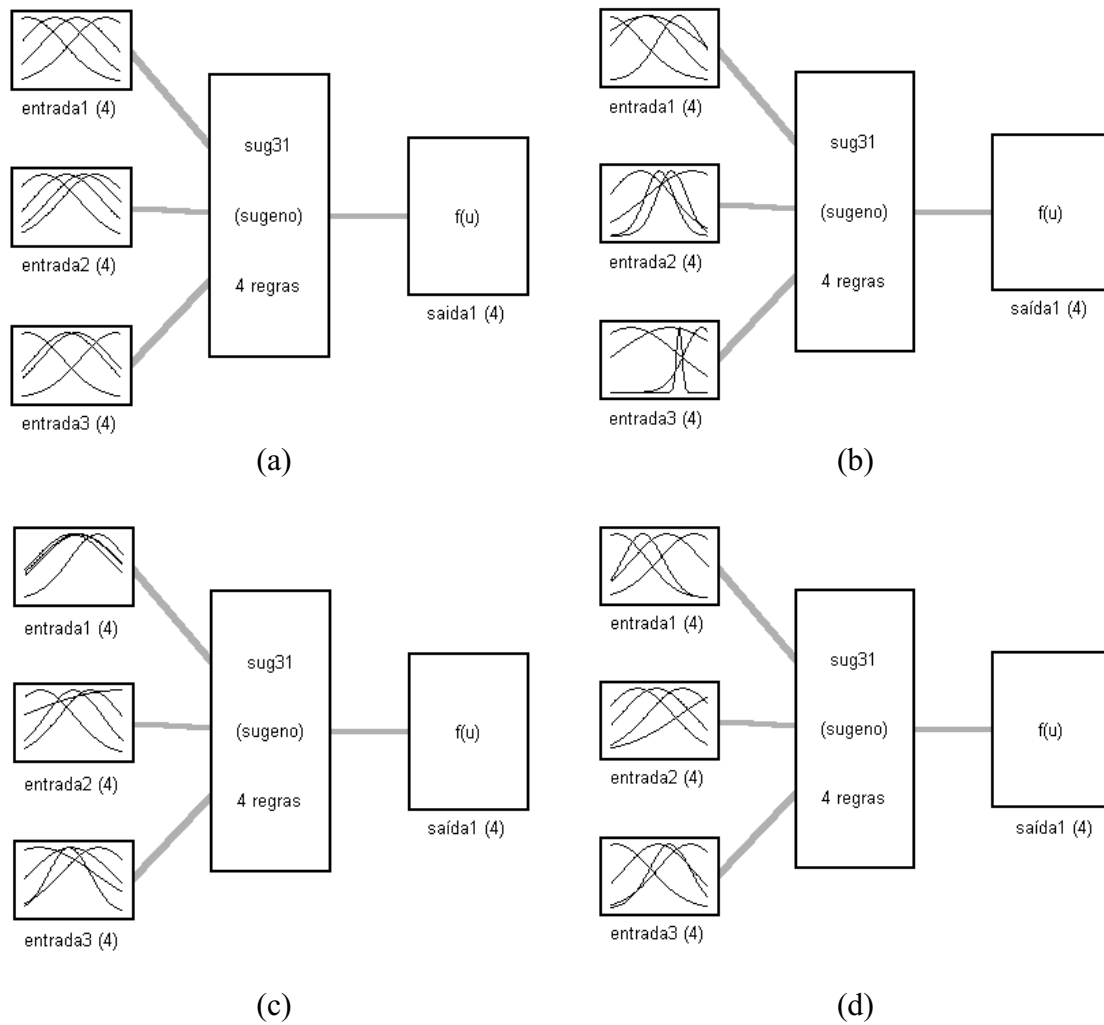


Figura 3.27 - Estrutura do ANFIS para a planta química: (a) Antes da otimização; (b) Após otimização por GD+LSE; (c) Após otimização por GA+LSE; (d) Após otimização por PSO+LSE.

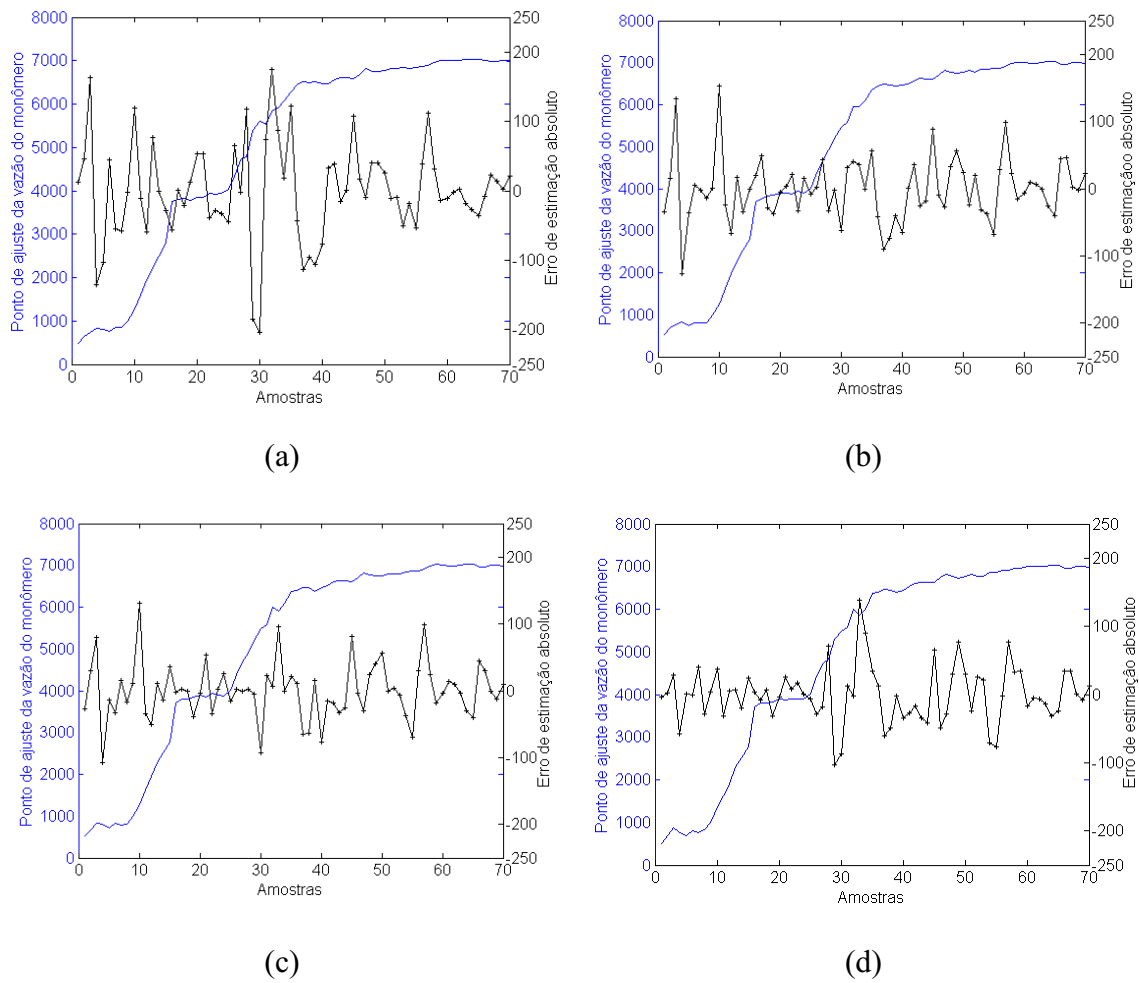


Figura 3.28 - Ponto de ajuste da vazão do monômero estimado pelo ANFIS: (a) Antes da otimização; (b) Após a otimização pelo GD+LSE; (c) Após a otimização pelo GA+LSE; (d) Após a otimização pelo PSO+LSE.

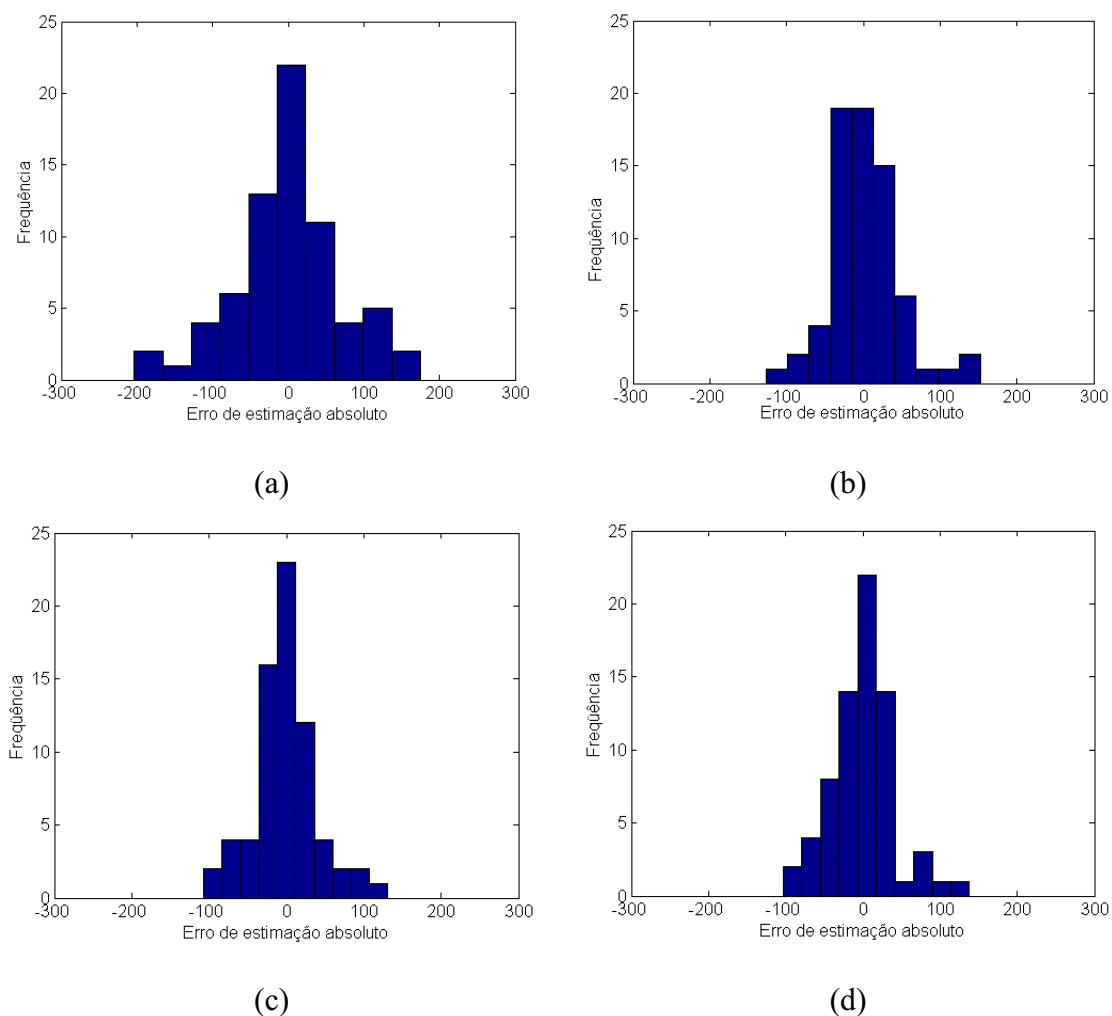


Figura 3.29 - Histograma dos erros de estimação do ANFIS para a planta química: (a) Antes da otimização; (b) Após a otimização pelo GD+LSE; (c) Após a otimização pelo GA+LSE; (d) Após a otimização pelo PSO+LSE.

A tabela 3.16 apresenta os resultados obtidos pelos três métodos de otimização. Para cada método de otimização o erro médio quadrático E (RMS E - *Root Mean Squared E*) mostrado na tabela foi obtido usando o ANFIS treinado correspondente ao erro médio quadrático mínimo obtido para o conjunto de dados de teste. A tabela também apresenta o número de iterações necessárias para alcançar o erro médio quadrático mínimo do conjunto de teste. Cada método de otimização foi executado 5

vezes. Para os métodos GA e PSO os indivíduos e partículas foram inicializados com valores aleatórios no intervalo $[x_{min}, x_{max}]$. O valor x_{min} e x_{max} dos parâmetros de cada função-pertinência (c_{ij} e s_{ij}), correspondente à variável de entrada x_j na i -ésima função-pertinência, são obtidos por

$$x_{c_{ij} \min} = \min(x_j) - span(x_j) \quad \text{e} \quad x_{c_{ij} \max} = \max(x_j) + span(x_j) \quad (3.11)$$

$$x_{s_{ij} \min} = 0.000001/span(x_j) \quad \text{e} \quad x_{s_{ij} \max} = span(x_j) \quad (3.12)$$

onde $\min(x_j)$, $\max(x_j)$ e $span(x_j)$ representam, respectivamente, o valor mínimo, o valor máximo e o valor do intervalo de alcance da variável x_j .

No método PSO foi usada a topologia totalmente conectada (todas as partículas são vizinhas) e as partículas não foram permitidas “voar” para fora da região definida por $[x_{min}, x_{max}]$ e a velocidade não foi restringida.

Tabela 3.16

Resultados obtidos para a planta química.

Método	Conjunto de dados	N^a	I^b	Número de épocas			Número esperado de avaliações ^c	Erro E RMS		
				Min	Max	Med		Min	Max	Med
ANFIS inicial	Treinamento	---	---	---	---	---	---	---	---	0.01076
	Teste									0.01526
	Total									0.01321
GD+LSE	Treinamento	1	1	233	233	233	233	0.00713	0.00713	0.00713
	Teste							0.01370	0.01370	0.01370
	Total							0.01092	0.01092	0.01092
GA+LSE	Treinamento	50	50	343	997	725	1811500	0.00643	0.00727	0.00677
	Teste							0.01331	0.01358	0.01346
	Total							0.01045	0.01090	0.01066
PSO+LSE	Treinamento	500	5	129	1000	445	1112500	0.00629	0.00741	0.00687
	Teste							0.01250	0.01428	0.01378
	Total							0.00990	0.01134	0.01089

^a Número de indivíduos ou partículas.

^b Número de iterações GD, GA ou PSO por época.

^c $N \times I \times$ (número médio de épocas).

3.6 Detecção de anomalia de sensor

Os métodos de detecção de anomalia ou falha em sensor em um sistema de validação de sinal são baseados no sinal de erro entre o valor estimado pelo sistema e medido (desejado) do sensor monitorado. A informação de falha é gerada quando este erro ultrapassa um limiar estipulado.

Foram geradas 1000 amostras aleatórias para o sinal estimado e medido usados nos ensaios de modo a possuírem média 1 e uma distribuição de ruído normal $N(0;0,05)$. A figura 3.30 apresenta os sinais medidos e estimados gerados e o sinal de erro entre esses sinais. A figura 3.31 apresenta o histograma da distribuição do sinal de erro.

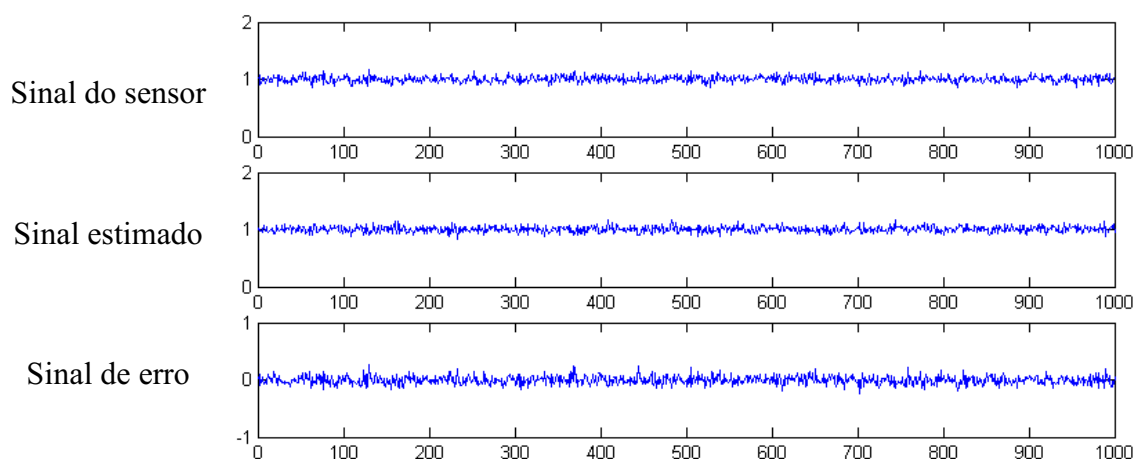


Figura 3.30 - Sinais de entrada medido e estimado e o sinal de erro para ensaio dos modelos de detecção de anomalia.

A seguir são apresentados os resultados de ensaios realizados em três métodos de detecção de anomalia, a saber: o método do erro (resíduos), o método do teste da razão de probabilidades seqüencial dos resíduos e o método da auto-correlação dos resíduos. Em todos os casos o sinal de anomalia foi inserido a partir da amostra 500. A descrição desses métodos encontra-se no capítulo 2 desta tese.

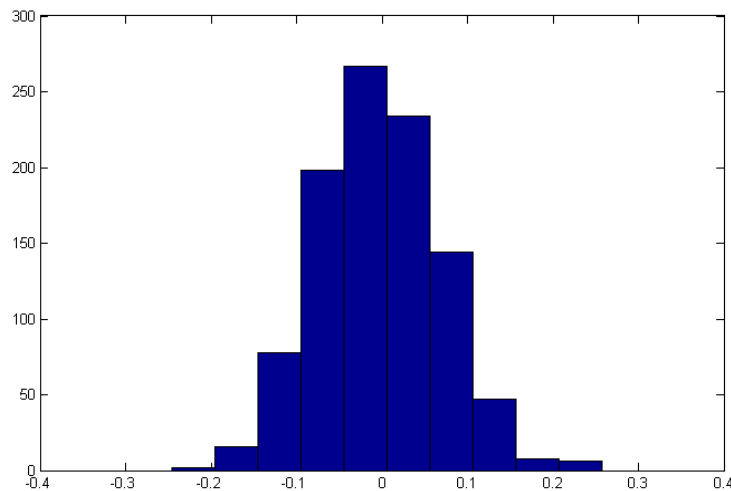


Figura 3.31 - Histograma da distribuição do sinal de erro.

3.6.1 Ensaios do método do erro (resíduos)

De modo a ajudar ao operador da planta na detecção de falhas ou desvios nos sinais monitorados são fornecidas duas bandas de erro. Essas bandas de erro devem ser interpretadas como:

Banda de alerta: É ajustada para ± 2 desvios padrão ($\pm 2\sigma_\varepsilon$) do sinal de erro ε obtido na etapa de treinamento. A ultrapassagem desta banda pelo sinal de erro é considerada como um primeiro alerta, especialmente se a situação persistir.

Banda de alarme: É ajustada para ± 4 desvios padrão ($\pm 4\sigma_\varepsilon$) do sinal de erro ε obtido na etapa de treinamento. A ultrapassagem desta banda pelo sinal de erro é considerada como um alerta definitivo.

A figura 3.32 apresenta os resultados obtidos pelo método do erro (resíduos) para diversos tipos de falha ou anomalia.

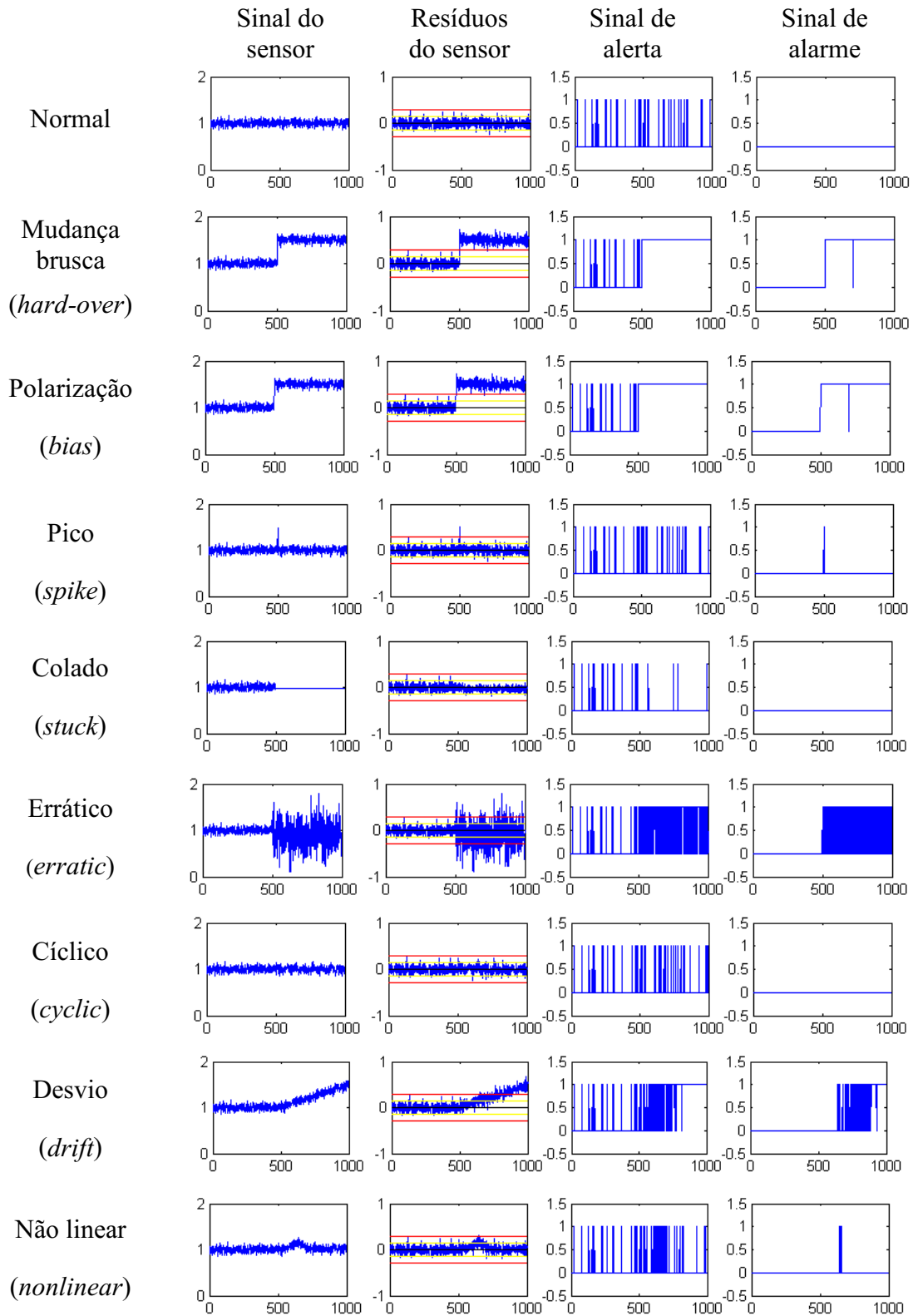


Figura 3.32 - Detecção de falhas pelo método do erro (resíduos).

Os ensaios efetuados evidenciam que o método de erro tem dificuldade de detectar anomalias no sinal de sensor do tipo errática e cíclica e não é capaz de detectar quando o sinal do sensor está colado.

3.6.2 Ensaios do método de teste da razão de probabilidades seqüencial (SPRT)

De modo a ajudar ao operador da planta na detecção de falhas ou desvios nos sinais monitorados são fornecidas duas bandas de erro. Essas bandas de erro devem ser interpretadas como:

Banda de alerta: É ajustada para zero. A ultrapassagem desta banda pela razão de probabilidades é considerada como um primeiro alerta, especialmente se a situação persistir.

Banda de alarme: É ajustada para ser igual ao valor B , calculado em função dos parâmetros α e β , descritos a seguir. A ultrapassagem desta banda pela razão de probabilidades é considerada como um alerta definitivo.

A taxa de alarmes perdidos foi feita igual a 0,001 ($\alpha=0,001$) e a taxa de alarmes falsos igual a 0,001 ($\beta=0,001$). Implicando em um valor de contorno $A=-6,9068$ e $B=6,9068$.

O valor da razão de probabilidades seqüencial dos resíduos é calculado pela equação (2.54) onde os valores da média μ_0 e desvio padrão σ_0 , desta equação, são obtidos a partir do conjunto de dados do treinamento do sistema e os valores μ_1 e σ_1 são feitos iguais a 0,025 da excursão do sinal de treinamento e $3 \sigma_0$, respectivamente.

A figura 3.33 apresenta os resultados obtidos pelo método da razão de probabilidades seqüencial dos resíduos para diversos tipos de falha ou anomalia.

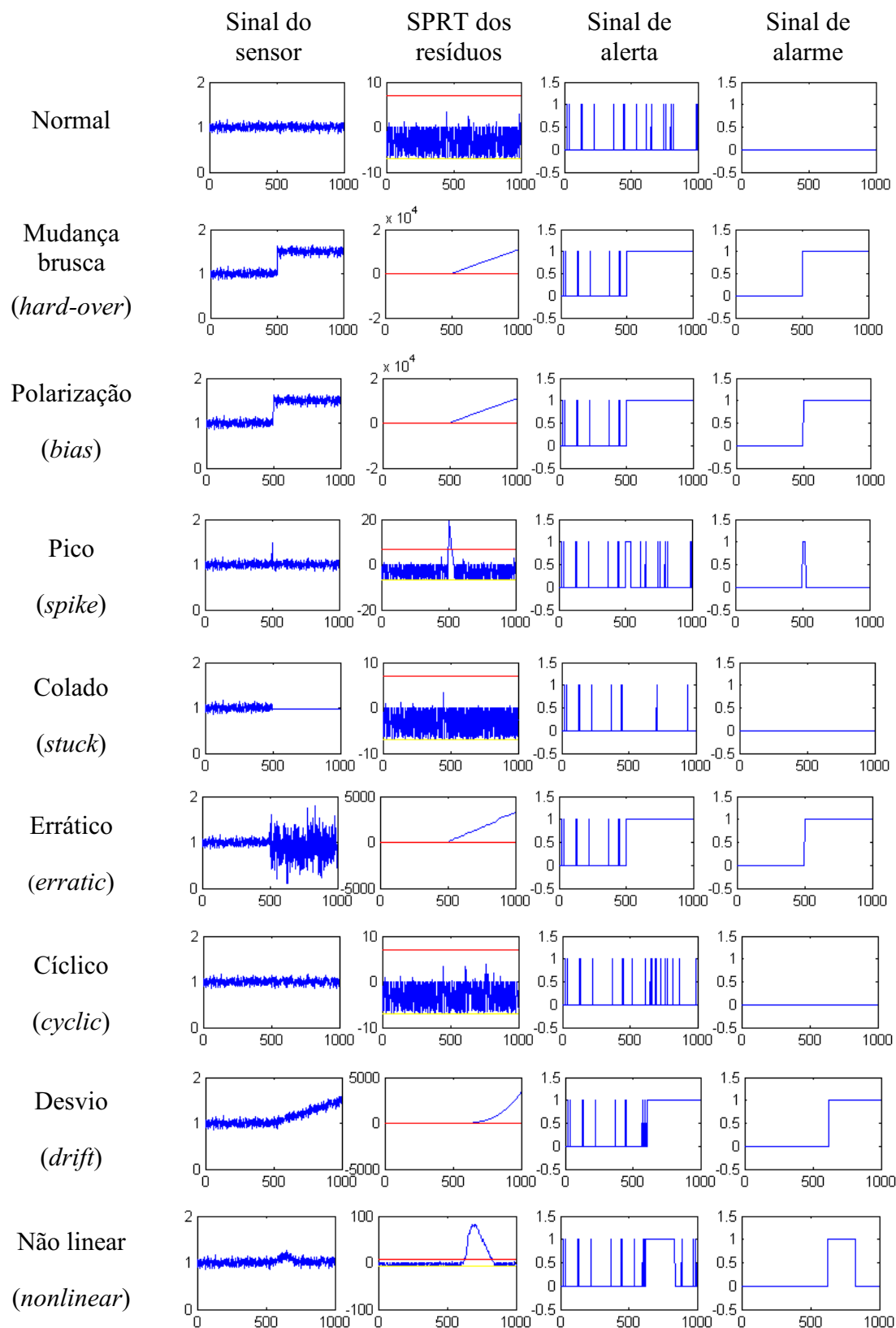


Figura 3.33 - Detecção de falhas pelo método da razão de probabilidades sequencial dos resíduos de erro.

Os ensaios efetuados evidenciam que o método da razão de probabilidades seqüencial dos resíduos de erro tem dificuldade de detectar anomalias no sinal de sensor do tipo cíclica e não é capaz de detectar quando o sinal do sensor está colado.

3.6.3 Ensaios do método da auto-correlação dos resíduos

De modo a ajudar ao operador da planta na detecção de falhas ou desvios nos sinais monitorados são fornecidas duas bandas de erro. Essas bandas de erro devem ser interpretadas como:

Banda de alerta: É ajustada para ± 2 desvios padrão ($\pm 2\sigma_d$) das distâncias do sinal de autocorrelação dos resíduos ε entre o valor monitorado e estimado, obtido na etapa de treinamento. A ultrapassagem desta banda pelo sinal de autocorrelação é considerada como um primeiro alerta, especialmente se a situação persistir.

Banda de alarme: É ajustada para ± 4 desvios padrão ($\pm 4\sigma_d$) das distâncias do sinal de autocorrelação dos resíduos ε entre o valor monitorado e estimado, obtido na etapa de treinamento. A ultrapassagem desta banda pelo sinal de autocorrelação é considerada como um alerta definitivo.

A janela de amostras para cálculo da autocorrelação dos resíduos foi feita igual a 50, ou seja, considera a amostra atual mais 49 amostras passadas.

O valor da autocorrelação dos resíduos é calculado pela equação (2.43) onde os valores da média μ_d e desvio padrão σ_d , desta equação, são obtidos a partir do conjunto de dados do treinamento do sistema.

A figura 3.34 apresenta os resultados obtidos pelo método da auto-correlação dos resíduos para diversos tipos de falha ou anomalia.

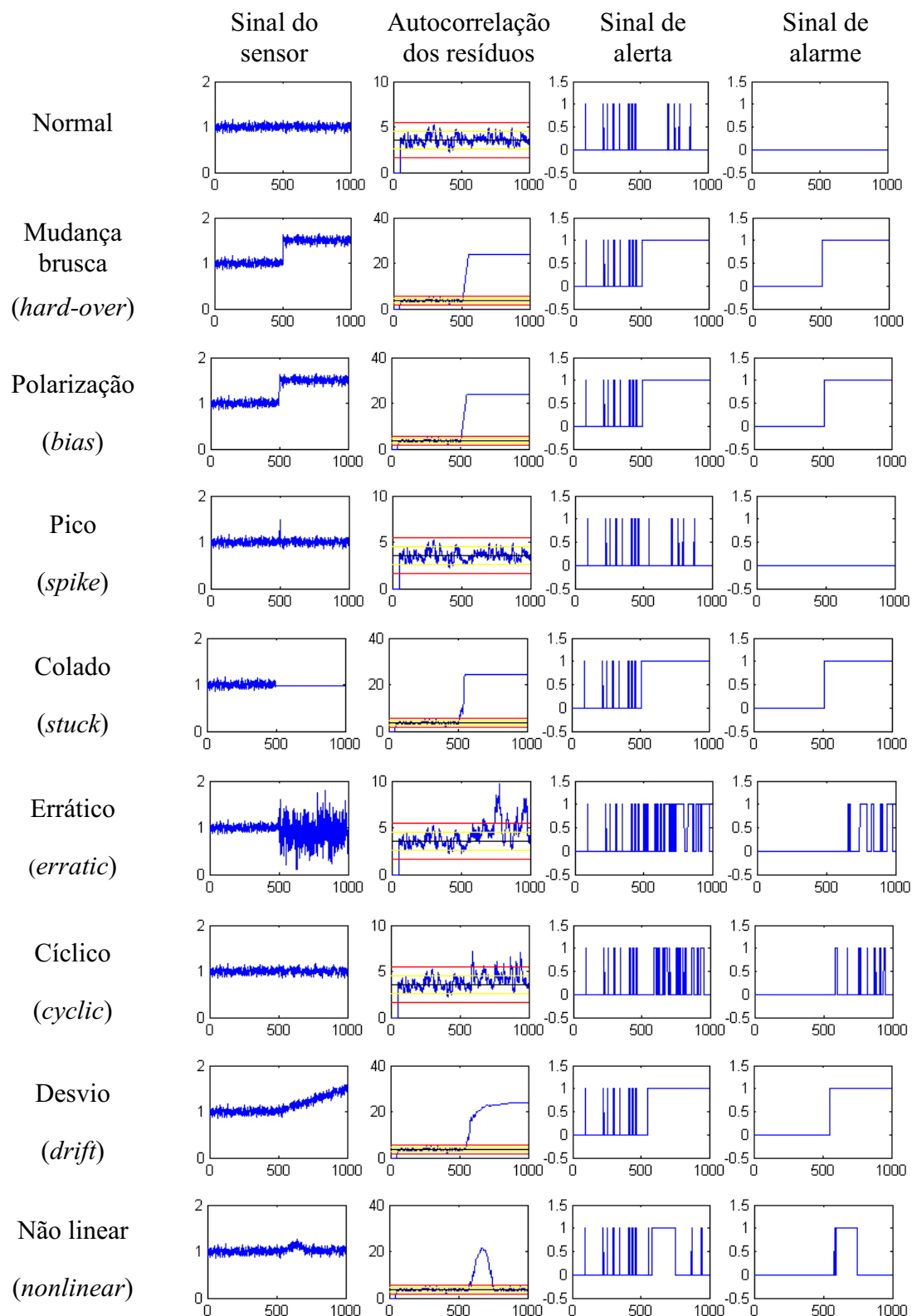


Figura 3.34 - Detecção de falhas pelo método da autocorrelação dos resíduos de erro.

Os ensaios efetuados evidenciam que o método da autocorrelação dos resíduos de erro não é capaz de detectar anomalias no sinal de sensor do tipo pico.

3.6.4 Combinando os métodos de detecção de anomalia

Os ensaios anteriores mostraram que os métodos de detecção de anomalia implementados não são capazes de detectar sozinhos todos os tipos de anomalias ensaiadas. Deste modo, torna-se necessário efetuar uma combinação desses métodos através de uma lógica de votação. A seguir, são apresentados os resultados obtidos nos ensaios para dois tipos de lógica de votação: a lógica de votação majoritária e a lógica de votação tipo OU.

3.6.4.1 Decisão por lógica de votação majoritária

Neste caso, o processo decisório de indicação de anomalia no sensor é obtido através de uma votação onde a indicação que a maioria dos métodos fornecer é assumida como a resposta final do modelo. Como foram implementados 3 métodos de detecção de anomalia teremos uma lógica de votação do tipo 2 em 3, ou seja, uma lógica de votação 2/3. Esta lógica de votação pode ser representada por

$$Alerta = \begin{cases} 1 & \text{se } soma(Alerta_{erro}, Alerta_{spri}, Alerta_{auto}) \geq 2 \\ 0 & \text{caso contrário} \end{cases} \quad (3.13)$$

$$Alarme = \begin{cases} 1 & \text{se } soma(Alarme_{erro}, Alarme_{spri}, Alarme_{auto}) \geq 2 \\ 0 & \text{caso contrário} \end{cases} \quad (3.14)$$

onde *Alerta* é o sinal resultante da lógica de votação majoritária e *soma* efetua a soma aritmética entre os sinais de alerta gerados pelos métodos de erro (*Alerta_{erro}*), de SPRT (*Alerta_{spri}*) e de autocorrelação (*Alerta_{auto}*) e *Alarme* é o sinal resultante da lógica de votação majoritária e *soma* efetua a soma aritmética entre os sinais de alarme gerados

pelos métodos de erro ($Alarme_{erro}$), de SPRT ($Alarme_{sprt}$) e de autocorrelação ($Alarme_{auto}$).

A figura 3.35 e 3.36 apresentam os resultados desta lógica de votação aplicada aos sinais utilizados nos ensaios para os sinais de *Alerta* e *Alarme*, respectivamente. Os ensaios efetuados evidenciam que esta lógica de votação não é capaz de detectar quando o sinal do sensor está colado ou com uma falha do tipo cíclica.

3.6.4.2 Decisão por lógica de votação tipo OU

Neste caso, o processo decisório de indicação de anomalia no sensor é obtido através de uma votação booleana tipo OU, onde a indicação de falha ou anomalia fornecida por um dos métodos é assumida como a resposta final do modelo. Esta lógica é aplicada entre os sinais de alerta e entre os sinais de alarme produzidos por cada método sendo dadas por

$$Alerta = \max(Alerta_{erro}, Alerta_{sprt}, Alerta_{auto}) \quad (3.15)$$

$$Alarme = \max(Alarme_{erro}, Alarme_{sprt}, Alarme_{auto}) \quad (3.16)$$

onde *Alerta* é o sinal resultante do OU (*max*) entre os sinais de alerta pelo método de erro ($Alerta_{erro}$), alerta pelo método SPRT ($Alerta_{sprt}$) e alerta pelo método autocorrelação ($Alerta_{auto}$) e *Alarme* é o sinal resultante do OU (*max*) entre os sinais de alarme pelo método de erro ($Alarme_{erro}$), alarme pelo método SPRT ($Alarme_{sprt}$) e alarme pelo método autocorrelação ($Alarme_{auto}$).

A figura 3.37 e 3.38 apresentam os resultados desta lógica de votação aplicada aos sinais utilizados nos ensaios para os sinais de *Alerta* e *Alarme*, respectivamente. Os

ensaios efetuados evidenciam que todos os tipos de falhas introduzidas foram detectadas por esta lógica de votação.

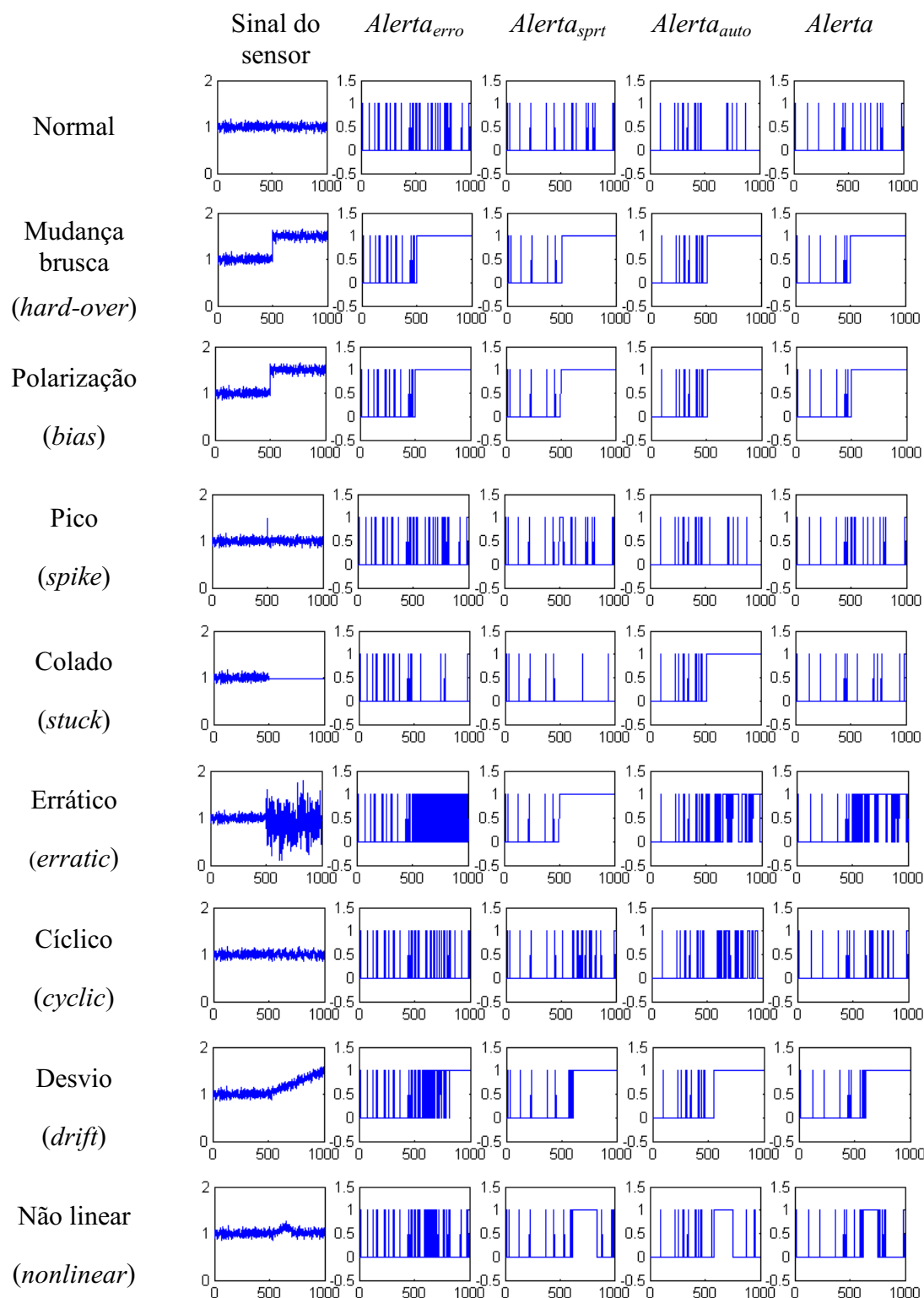


Figura 3.35 - Resultados da lógica de votação majoritária aplicada aos sinais de *Alerta*.

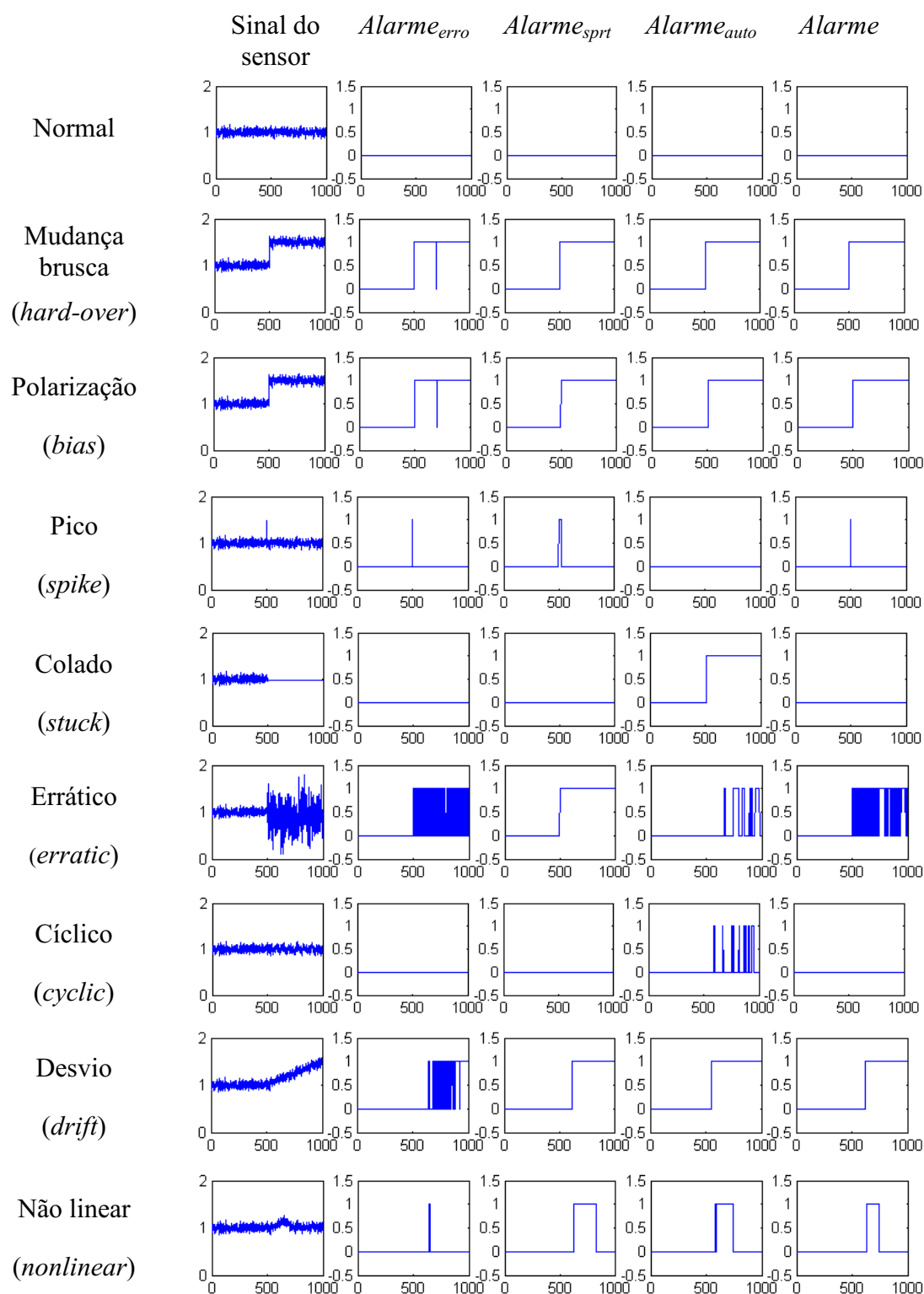


Figura 3.36 - Resultados da lógica de votação majoritária aplicada aos sinais de $Alarme$.

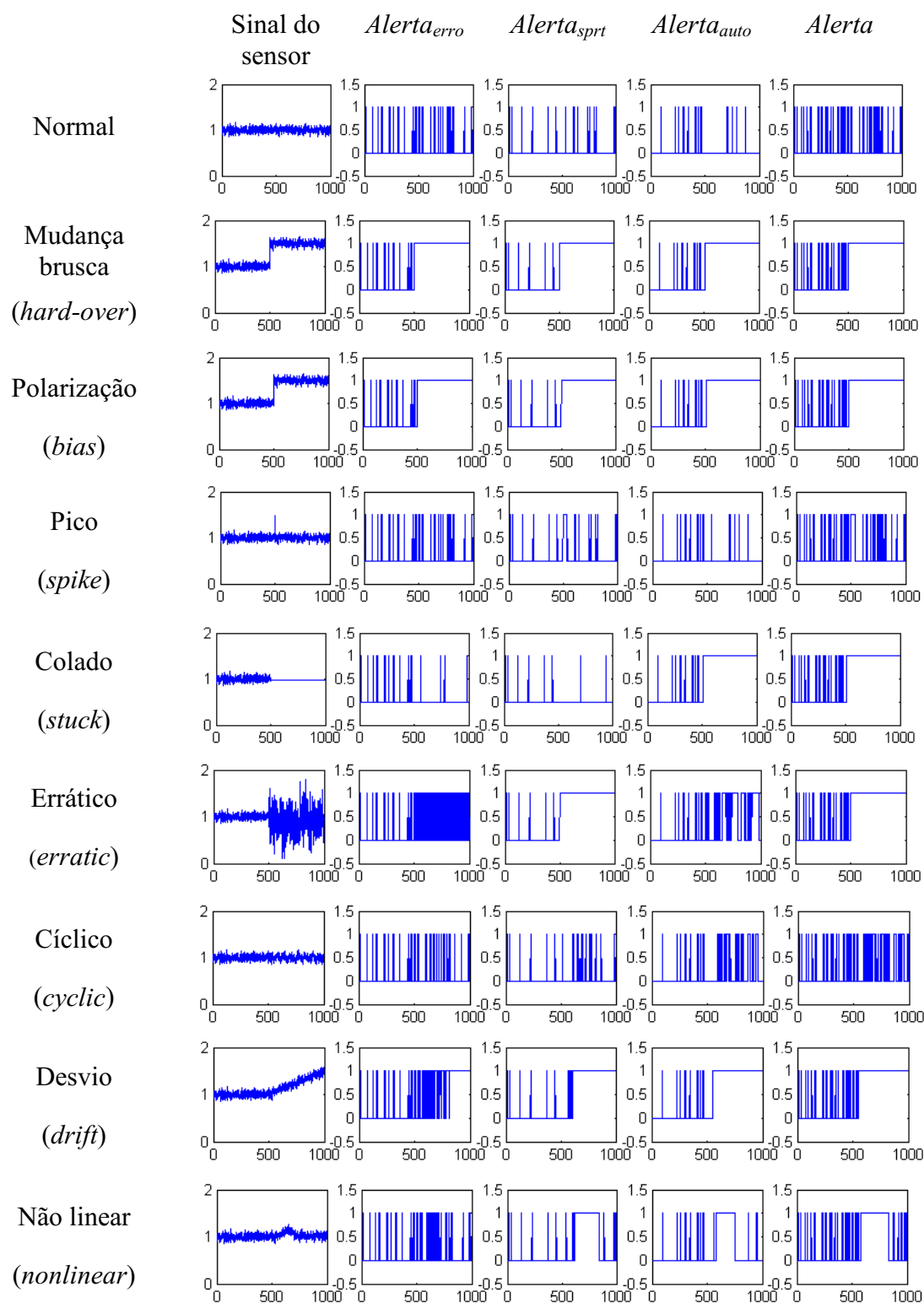


Figura 3.37 - Resultados da lógica de votação tipo OU aplicada aos sinais de *Alerta*.

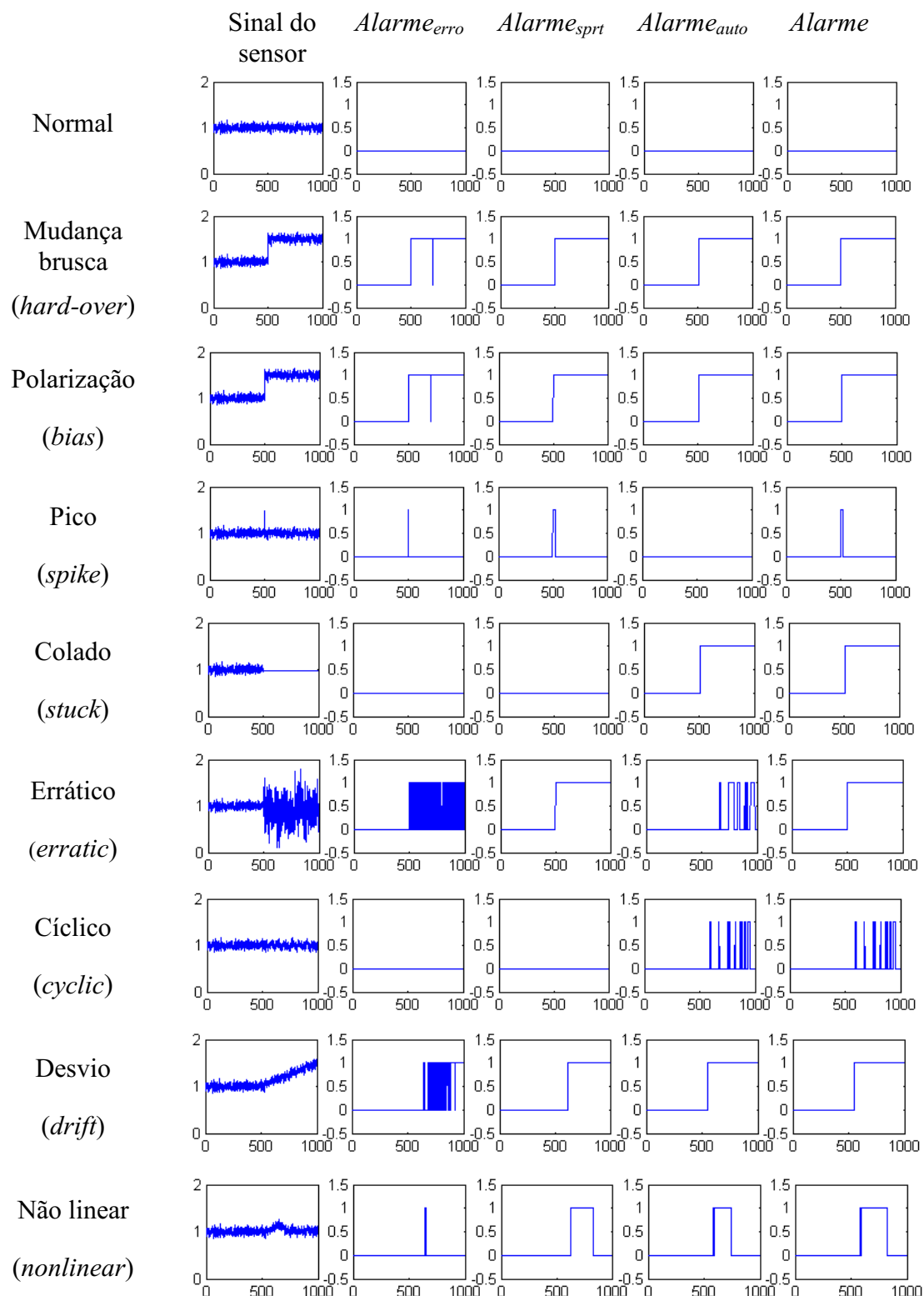


Figura 3.38 - Resultados da lógica de votação tipo OU aplicada aos sinais de $Alarme$.

3.7 Confiança da resposta do sistema

O módulo de confiança da resposta tem por objetivo fornecer ao operador da planta o grau de confiança na estimativa fornecida pelo sistema de validação de sinal do sinal que está sendo monitorado. Em outras palavras, ele indica se o sistema de validação de sinal foi treinado para as amostras (sinais monitorados) que são apresentados às suas entradas.

Como já foi citado, geralmente, o módulo de confiança da resposta é realizado por um sistema nebuloso (FIS) que recebe às suas entradas os sinais de resíduos produzidos pelos sistemas de detecção de falhas e um sinal de pertinência das amostras de entrada do sistema ao conjunto de treinamento e fornece à sua saída uma indicação quantitativa sobre a confiança da resposta produzida pelo sistema.

A seguir são apresentados alguns ensaios realizados no módulo de confiança da resposta. A descrição detalhada do módulo de confiança da resposta implementado nesta tese é apresentada no capítulo 4.

3.7.1 Comparação dos algoritmos FCM e PCM

Neste ensaio são comparados os algoritmos FCM e PCM com o intuito de avaliar o grau de pertinência de amostras a um conjunto de classes. Este grau de pertinência das amostras será usado como um dos sinais de entrada para o módulo de confiança da resposta.

Inicialmente, foram utilizadas 14 amostras bidimensionais que foram classificadas em duas classes ($C=2$) pelos métodos FCM e EFCM, utilizando as métricas Euclidiana e Mahalanobis. A tabela 3.17 apresenta os valores das 14 amostras utilizadas no teste. Os valores esperados (reais) dos centróides (centróides) para as duas classes $C1$ e $C2$ são, respectivamente, $C1=(60;150)$ e $C2=(140;150)$. A tabela 3.17 também apresenta as

pertinências obtidas por cada método para as duas classes. Podemos observar que o método EFCM por considerar centróides das classes volumétricos em vez de pontuais, como no método FCM, atribui a pertinência máxima a algumas amostras, que estão dentro do raio das classes, além das amostras posicionadas muito próximas aos centróides reais das classes.

Pela tabela 3.17 podemos observar que o método EFCM com métrica Mahalanobis foi o que obteve centróides das classes mais próximos aos centróides reais.

Tabela 3.17

Resultados obtidos das pertinências das amostras em cada classe, índice de validade e centróides das classes para separação em classes pelos métodos FCM e EFCM com métricas Euclidiana e Mahalanobis.

Amostra	Método FCM				Método EFCM			
	Euclidiano		Mahalanobis		Euclidiano		Mahalanobis	
	$C11$	$C12$	$C11$	$C12$	$C11$	$C12$	$C11$	$C12$
1 (60; 160)	0,9849	0,0151	0,9217	0,0783	0,9992	0,0008	0,9781	0,0219
2 (140; 160)	0,0151	0,9849	0,0783	0,9217	0,0008	0,9992	0,0219	0,9781
3 (40; 150)	0,9627	0,0373	0,9620	0,0380	0,9838	0,0162	0,9920	0,0080
4 (50; 150)	0,9885	0,0115	0,9881	0,0119	0,9995	0,0005	1,0000	0,0000
5 (60; 150)	1,0000	0,0000	1,0000	0,0000	1,0000	0,0000	1,0000	0,0000
6 (70; 150)	0,9791	0,0209	0,9796	0,0204	0,9988	0,0012	1,0000	0,0000
7 (80; 150)	0,8984	0,1016	0,8993	0,1007	0,9483	0,0517	0,9723	0,0277
8 (120; 150)	0,1016	0,8984	0,1007	0,8993	0,0517	0,9483	0,0277	0,9723
9 (130; 150)	0,0209	0,9791	0,0204	0,9796	0,0012	0,9988	0,0000	1,0000
10 (140; 150)	0,0000	1,0000	0,0000	1,0000	0,0000	1,0000	0,0000	1,0000
11 (150; 150)	0,0115	0,9885	0,0119	0,9881	0,0005	0,9995	0,0000	1,0000
12 (160; 150)	0,0373	0,9627	0,0380	0,9620	0,0162	0,9838	0,0080	0,9920
13 (60; 140)	0,9849	0,0151	0,9217	0,0783	0,9992	0,0008	0,9781	0,0219
14 (140; 140)	0,0151	0,9849	0,0783	0,9217	0,0008	0,9992	0,0219	0,9781
$F_2(U)$	0,9461		0,9134		0,9805		0,9778	
Centróide	C1=(59,74; 150) C2=(140,26; 150)		C1=(59,89; 150) C2=(140,11; 150)		C1=(59,82; 150) C2=(140,17; 150)		C1=(59,91; 150) C2=(140,09; 150)	

Vamos agora classificar duas novas amostras nas classes obtidas pelo método EFCM com métrica Mahalanobis. O motivo de usar o EFCM com métrica Mahalanobis

para esta aplicação é devido ao fato que ele obteve centróides das classes mais próximos aos centróides reais. A figura 3.39 apresenta os centróides das classes obtidos pelo EFCM com métrica Mahalanobis, representados por (\times), a distribuição das 14 amostras na classes $C1$ e $C2$, representadas por (o) e ($+$), respectivamente, a distância média quadrática das amostras nas classes (elipses) e a posição das duas novas amostras a serem classificadas representadas por ($*$).

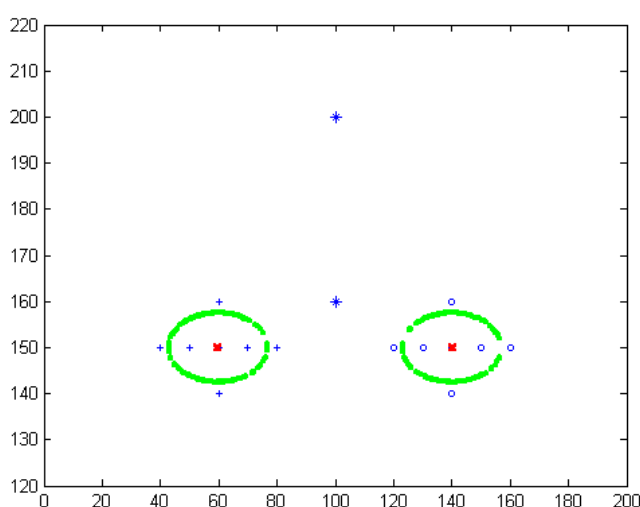


Figura 3.39 - Centróides das classes (\times), amostras das classes $C1$ ($+$) e $C2$ (o), a distância média quadrática das amostras nas classes (elipses) e as duas amostras ($*$) a serem classificadas.

A posição das duas amostras no espaço bi-dimensionais são dadas na tabela 3.18 bem como os valores de suas pertinências nas classes $C1$ e $C2$ para a classificação nebulosa e possibilística. A classificação nebulosa é calculada pela equação (2.27) e a possibilística pela equação (2.30).

Pode-se observar pela tabela 3.18 que foram obtidos valores iguais de pertinências das amostras nas classes $C1$ e $C2$ para os dois métodos. Isto deve-se ao fato que as amostras estão equidistantes dos centróides das classes. Além disso, as pertinências

obtidas pelo método FCM para as duas amostras nas classes $C11$ e $C12$ foram iguais, apesar de que a amostra 2 estar mais distante dos centróides das classes. Em outras palavras, no algoritmo FCM a pertinência de um ponto em uma classe é um número relativo que depende da pertinência do ponto em todas as outras classes e, deste modo, depende indiretamente do número total de classes. Isto é, a pertinência total reflete um compartilhamento do ponto entre as classes. Embora isto não possa ser um problema em algumas aplicações, tal como em classificação de padrões, isto nem sempre é apropriado para muitas aplicações com lógica nebulosa.

Tabela 3.18

Posição e pertinência nebulosa e possibilística das duas amostras.

Amostra	Classificação FCM equação (2.27)		Classificação PCM equação (2.30)	
	$C11$	$C12$	$C11$	$C12$
1 (100;160)	0,5000	0,5000	0,1981	0,1981
2 (100;200)	0,5000	0,5000	0,0239	0,0239
Centróide EFCM	$C1=(59,91;150)$	$C2=(140,09;150)$	$C1=(59,91;150)$	$C2=(140,09;150)$

Por outro lado, no método PCM o valor da pertinência de um ponto em uma classe é somente função da sua distância ao centróide da classe. Os resultados obtidos mostram valores de pertinência menores para a amostra 2 do que para a amostra 1, o que é natural pois a amostra 2 está mais distante dos centróides do que a amostra 1. Esta capacidade deste método será utilizada na etapa de operação dos modelos para obtenção da confiança da resposta do sistema através de um sistema de inferência nebulosa.

3.7.2 Ensaio do modelo de inferência nebulosa

Um modelo de inferência nebulosa (FIS) do tipo Mamdani pode ser dividido em cinco partes: a nebulização das variáveis de entrada, a aplicação dos operadores

nebulosos da parte antecedente, a implicação da parte antecedente para a parte conseqüente, a agregação dos conseqüentes através das regras e a desnebulização.

Antes de utilizar o FIS é necessário selecionar/ajustar os parâmetros de cada parte de sua composição. A nebulização das variáveis de entrada do FIS envolve a seleção do número, da forma (Gaussiana, Sigmóide, etc.) e parâmetros de ajustes das funções pertinência da parte antecedente. A aplicação dos operadores nebulosos envolve a definição das regras e dos operadores nebulosos (E, OU, etc.) aplicados às regras. A implicação da parte antecedente para a parte conseqüente envolve a seleção do número, da forma (Gaussiana, Sigmóide, etc.) e parâmetros de ajustes das funções pertinência da parte conseqüente. A agregação dos conseqüentes envolve a combinação das regras (MIN, MAX, etc.) para se obter uma decisão. Finalmente, a desnebulização envolve a definição do método de agregação (Centróide, bissetor, etc.) para produzir um único sinal de saída do FIS.

A definição de um procedimento para seleção/ajuste dos diversos parâmetros de um sistema de inferência nebulosa do tipo Mamdani para uma aplicação está além do escopo desta tese.

3.7.3 Ensaio do modelo de confiança da resposta

A seguir são apresentados alguns ensaios realizados no modelo de confiança da resposta utilizado no sistema de validação de sinais implementado nesta tese. A descrição detalhada deste modelo é apresentada no capítulo 4.

Foram realizados ensaios considerando as pertinências das amostras de entrada do modelo iguais a 0,8, 0,5 e 0,3. As tabelas 3.19, 3.20 e 3.21 apresentam os resultados obtidos pelo modelo para um sinal de entrada normal, com variação positiva brusca e com variação positiva lenta.

Os sinais de resíduos produzidos pelos sistemas de detecção de falhas foram normalizados antes de serem aplicados às entradas do modelo.

Os resultados obtidos mostram a importância do parâmetro de pertinência da amostra no conjunto de treinamento do modelo. Isto é devido a uma das regras utilizadas no modelo que diz

Se pertinência é grande então confiança é grande.

Por outro lado, quando a pertinência da amostra no conjunto de treinamento for reduzida, os resíduos do erro de estimação tornam-se mandatórios na resposta do modelo de confiança da resposta.

Tabela 3.19

Confiança da resposta para pertinência das amostras de entrada igual a 0,8.

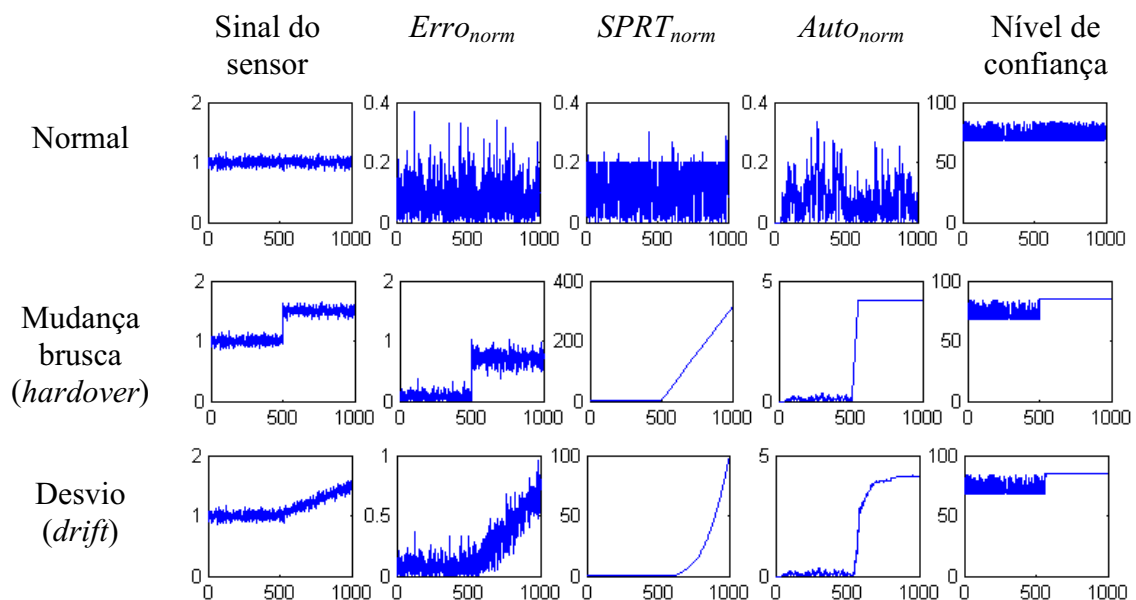


Tabela 3.20

Confiança da resposta para pertinência das amostras de entrada igual a 0,5.

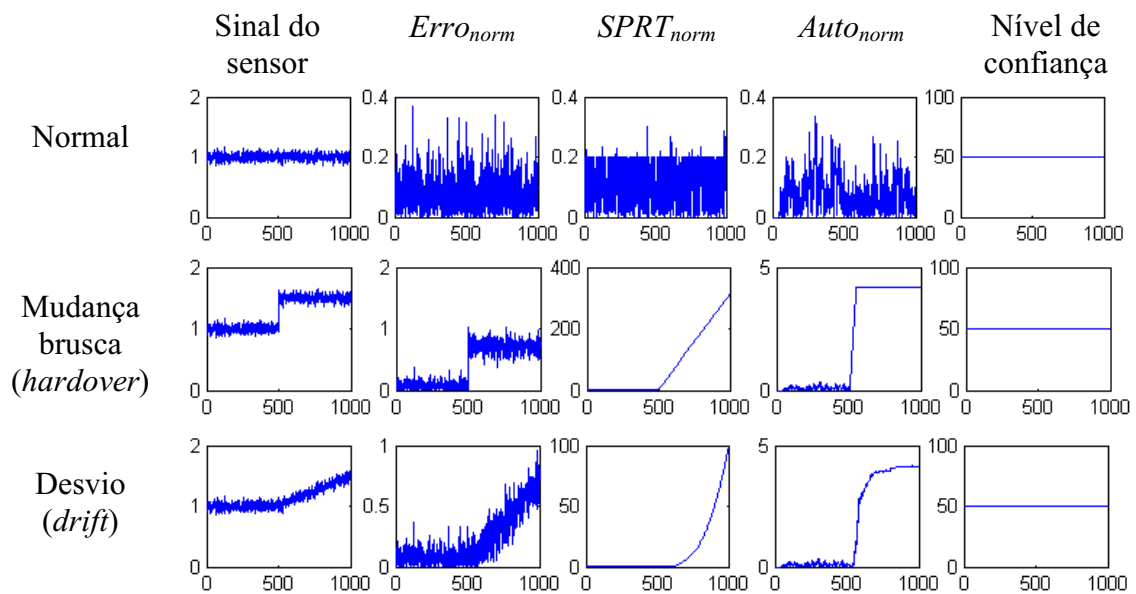
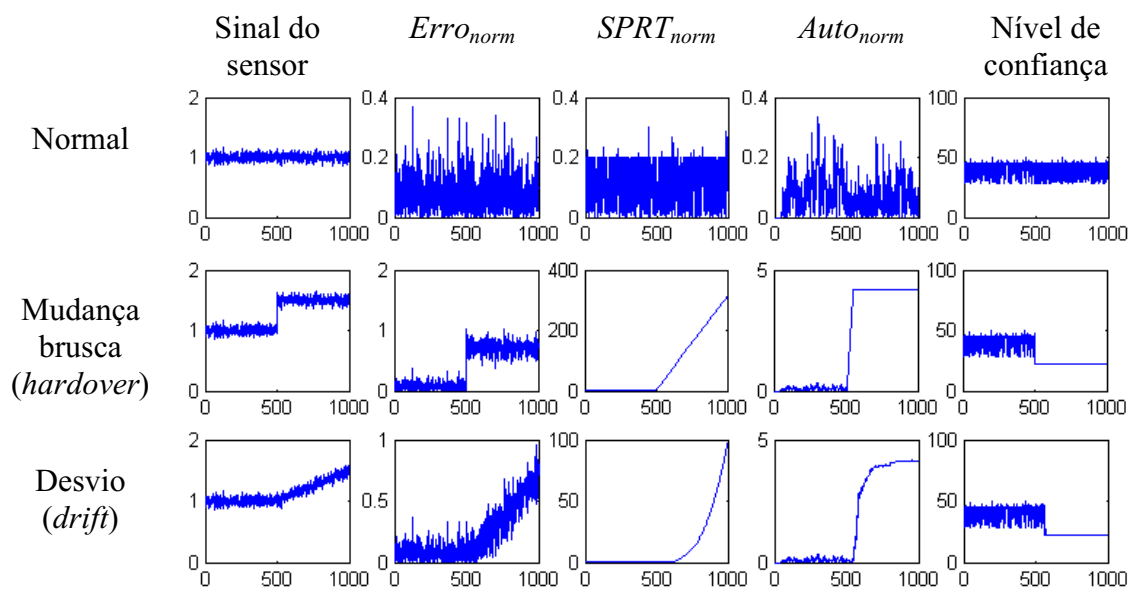


Tabela 3.21

Confiança da resposta para pertinência das amostras de entrada igual a 0,3.



CAPÍTULO 4

O modelo de validação de sinal implementado

Neste capítulo são apresentados e analisados, segundo a metodologia proposta no capítulo 2, alguns dos modelos de validação de sinal discutidos no capítulo 1. Com base nesta análise e nos resultados obtidos nos ensaios realizados nas diversas sub-etapas que compõem um modelo de validação de sinal é proposto o modelo de validação de sinal implementado nesta tese.

4.1 Classificação de alguns modelos de validação segundo a metodologia proposta

A seguir são apresentadas quatro sistemas de validação de sinal implementados anteriormente. Este sistema são classificados segundo a metodologia proposta no capítulo 2.

4.1.1 Aplicação 1 - Modelo de validação de sinal utilizando ANNs

Como já foi citado anteriormente, FANTONI *et al.* (1996) desenvolveram um modelo de validação de sinal utilizando ANNs. As técnicas de processamento de sinal utilizadas na implementação deste modelo são apresentadas na tabela 4.1.

Neste modelo, na etapa de pré-processamento é feita a seleção dos sinais de entrada com base no senso de engenharia, a normalização dos sinais de entrada entre $[-1,1]$, a filtragem dos sinais de entrada por ondaletas e a separação em classes nebulosas (FCM) utilizando como métrica a distância de Mahalanobis, onde toda a região de operação da planta é dividida em diversas sub-regiões.

Na etapa de processamento são utilizadas várias redes neurais multicamadas, do tipo auto-associativas, com três ou mais camadas, onde é treinada uma ANN específica para operar em cada uma dessas sub-regiões. O tipo de treinamento das ANNs utilizado é o de retropropagação do erro otimizado pelo seu gradiente descendente.

Tabela 4.1
Modelo de validação de sinal usando redes neurais (FANTONI *et al.*, 1996).

Pré-processamento						Processamento				Pós-processamento		
Seleção dos sinais	Normalização dos sinais	Filtragem dos sinais	Redução da dimensão	Métrica	Separação em classes	Modelo	Tipo	Estrutura	Otimização	Detecção da anomalia		Confiança da resposta
										Método	Forma	
-----	Nenhuma	Nenhuma	Nenhuma	Nenhuma	Nenhuma	-----	-----	-----	Nenhuma	-----	-----	Nenhuma
Correlação física	[-1,1]	Digital	PCA	Euclidiana	Centróide (<i>k</i> -means)	Matemático	Auto-associativo	Única	Mínimos quadrados	Erro do sensor (resíduos)	Limiar	FIS
Senso de engenharia	[0,1]	Ondaleta (<i>wavelet</i>)	Ondaleta (<i>wavelet</i>)	Minkowski	AVQ	MLP	Hetero-associativo	Múltipla	Gradiente descendente	Auto-correlação dos resíduos	Lógica de votação	
Proximidade ao sensor monitorado	[-0.5,0.5]	Filtro Kalman	Rede neural	Mahalanobis	FCM	Sistema especialista			GA	Espectro de potência dos resíduos	Lógica booleana	
Correlação usando GA				Manhattan	PCM	FIS				SPRT dos resíduos	Lógica nebulosa	
Sinais redundantes					FCM Crescente	<i>Neuro-fuzzy</i>				Filtro Kalman dos resíduos		

Na etapa de pós-processamento do modelo é feita a cada instante a detecção de anomalia de sensor pelo cálculo do resíduo entre o valor estimado pelo modelo e o valor medido das variáveis monitoradas. Se este resíduo ultrapassar um limiar pré-estabelecido é gerado um sinal de falha de sensor. Este modelo também fornece uma indicação de confiança de sua resposta realizado por um sistema de inferência nebulosa. Este sistema recebe a cada instante às suas entradas os sinais de resíduo do erro e o sinal de pertinência possibilística da amostra no conjunto de treinamento do modelo e produz à sua saída uma indicação do grau de confiança da resposta produzida pelo modelo.

O modelo foi testado com dados gerados por um simulador para um reator do tipo PWR e as redes utilizadas nesta aplicação operavam tanto para as variáveis do primário quanto para as do secundário da reator.

4.1.2 Aplicação 2 - Modelo de validação de sinal utilizando ANFIS

HINES *et al.* (1997) desenvolveram um modelo de validação de sinal utilizando um sistema de inferência nebuloso adaptativo (ANFIS). As técnicas de processamento de sinal utilizadas na implementação deste modelo são apresentadas na tabela 4.2.

Neste modelo, na etapa de pré-processamento é feita a seleção dos sinais de entrada é baseada na escolha de variáveis de entrada que estejam fortemente correlacionadas com a variável de saída a ser monitorada, através da utilização de algoritmos genéticos. O GA é utilizado para encontrar um conjunto de variáveis de entrada que minimizam o erro de estimação do sinal de saída do modelo. A normalização dos sinais de entrada é feita entre [0,1].

Tabela 4.2
Modelo de validação de sinal usando inferência neuro-nebulosa (WREST *et al.*, 1997).

Pré-processamento						Processamento				Pós-processamento		
Seleção dos sinais	Normalização dos sinais	Filtragem dos sinais	Redução da dimensão	Métrica	Separação em classes	Modelo	Tipo	Estrutura	Otimização	Detecção da anomalia		Confiança da resposta
										Método	Forma	
-----	Nenhuma	Nenhuma	Nenhuma	Nenhuma	Nenhuma	-----	-----	-----	Nenhuma	-----	-----	Nenhuma
Correlação física	[-1,1]	Digital	PCA	Euclidiana	Centróide (<i>k-means</i>)	Matemático	Auto-associativo	Única	Mínimos quadrados	Erro do sensor (resíduos)	Limiar	FIS
Senso de engenharia	[0,1]	Ondaleta (<i>wavelet</i>)	Ondaleta (<i>wavelet</i>)	Minkowski	AVQ	MLP	Hetero-associativo	Múltipla	Gradiente descendente	Auto-correlação dos resíduos	Lógica de votação	
					RBF	AVQ						
Proximidade ao sensor monitorado	[-0.5,0.5]	Filtro Kalman	Rede neural	Mahalanobis	FCM	Sistema especialista			GA	Espectro de potência dos resíduos	Lógica booleana	
					GAFCM							
Correlação usando GA				Manhattan	PCM	FIS				SPRT dos resíduos	Lógica nebulosa	
Sinais redundantes					FCM Crescente	<i>Neuro-fuzzy</i>				Filtro Kalman dos resíduos		

Na etapa de processamento é utilizado para cada variável monitorada um sistema neuro-nebuloso único, do tipo hetero-associativo, com uma saída correspondente à variável monitorada, que é treinado para operar em todas as regiões de operação da planta. O tipo de treinamento do sistema neuro-nebuloso é o de retropropagação do erro, otimizado pelo seu gradiente descendente para a parte antecedente e de mínimos quadrados para a parte conseqüente do sistema.

Na etapa de pós-processamento do modelo é feita a cada instante a detecção de anomalia de sensor pelo teste da razão de probabilidades seqüencial do resíduo entre o valor estimado pelo modelo e o valor medido das variáveis monitoradas. Se esta razão ultrapassar um limiar pré-estabelecido é gerado um sinal de falha de sensor. Este modelo não fornece qualquer indicação sobre a confiança de sua resposta.

Este sistema foi demonstrado usando dados fornecidos pela planta de potência nuclear de Crystal River unidade 3, localizada no estado da Flórida nos Estados Unidos. As variáveis monitoradas pelo sistema foram o nível do gerador de vapor do *loop* A, a temperatura da perna quente do *loop* A e a vazão de água de alimentação do *loop* A do reator.

4.1.3 Aplicação 3 - Modelo de validação de sinal utilizando ANNs

YEN *et al.* (2001) desenvolveram um sistema para validação de sinal baseado em redes neurais auto-associativas. As técnicas de processamento de sinal utilizadas na implementação deste modelo são apresentadas na tabela 4.3.

Neste modelo, na etapa de pré-processamento é feita a seleção dos sinais de entrada a serem utilizados com base na proximidade destes ao sinal monitorado. O modelo utiliza a métrica Manhattan como medida de distância da auto-correlação e do

Tabela 4.3
Modelo de validação de sinal usando redes neurais (YEN *et al.*, 2001).

Pré-processamento						Processamento				Pós-processamento		
Seleção dos sinais	Normalização dos sinais	Filtragem dos sinais	Redução da dimensão	Métrica	Separação em classes	Modelo	Tipo	Estrutura	Otimização	Detecção da anomalia		Confiança da resposta
										Método	Forma	
-----	Nenhuma	Nenhuma	Nenhuma	Nenhuma	Nenhuma	-----	-----	-----	Nenhuma	-----	-----	Nenhuma
Correlação física	[-1,1]	Digital	PCA	Euclidiana	Centróide (<i>k</i> -means)	Matemático	Auto-associativo	Única	Mínimos quadrados	Erro do sensor (resíduos)	Limiar	FIS
Senso de engenharia	[0,1]	Ondaleta (<i>wavelet</i>)	Ondaleta (<i>wavelet</i>)	Minkowski	AVQ	MLP	Hetero-associativo	Múltipla	Gradiente descendente	Auto-correlação dos resíduos	Lógica de votação	
					RBF	RBF						
Proximidade ao sensor monitorado	[-0.5,0.5]	Filtro Kalman	Rede neural	Mahalanobis	FCM	Sistema especialista			GA	Espectro de potência dos resíduos	Lógica booleana	
Correlação usando GA				Manhattan	PCM	FIS				SPRT dos resíduos	Lógica nebulosa	
Sinais redundantes					FCM Crescente	<i>Neuro-fuzzy</i>				Filtro Kalman dos resíduos		

espectro de potência dos resíduos do erro de estimação do modelo. Esta métrica é utilizada pelo sistema na etapa de detecção de falha. O modelo emprega um algoritmo de separação em classes baseado em lógica nebulosa que divide o espaço de operação do sistema numa série de sub-regiões e treina redes neurais para operar cada uma em uma dessas sub-regiões do espaço. Neste modelo a separação do espaço em sub-regiões é feita através de um algoritmo que inicialmente supõe todos os dados de treinamento do modelo como pertencentes a uma só classe (modelo com uma única ANN). Através da comparação do sinal de erro, entre o valor estimado pela ANN e o valor medido do sinal monitorado, e um valor de erro máximo desejado para o sistema são adicionadas novas ANNs ao sistema até que esta condição seja satisfeita.

Na etapa de processamento são utilizadas várias redes neurais multicamadas, do tipo hetero-associativas, com três camadas, onde é treinada uma ANN específica para operar em cada uma dessas sub-regiões. O tipo de treinamento das ANNs utilizado é o de retropropagação do erro otimizado pelo seu gradiente descendente.

Na etapa de pós-processamento do modelo, é feita, a cada instante, a detecção de anomalia de sensor através de uma combinação lógica do tipo OU de três métodos de detecção de anomalia: o resíduo de estimação da variável monitorada (error de sensor), a auto-correlação dos resíduos de estimação e o espectro de potência dos resíduos de estimação. Se qualquer dos métodos de detecção de anomalia produzir um sinal de desvio que ultrapasse o seu limiar pré-estabelecido é gerado um sinal de falha de sensor. Este modelo não fornece qualquer indicação sobre a confiança de sua resposta.

Este sistema foi demonstrado com dois conjuntos de dados. O primeiro conjunto de dados consiste de um simulador de falhas de maquinaria *Spectra Quest (SQ Machinery Fault Simulator)*. Foram usadas como entrada para o sistema três sensores (acelerômetros) próximos ao sensor monitorado. O segundo conjunto de dados consiste

de uma série temporal de vibração conhecida como conjunto de dados de vibração *Westland*. Foram utilizados como entrada para o sistema um arranjo de oito sensores (acelerômetros) fixados em localizações específicas de um helicóptero CH-46 da Marinha dos Estados Unidos, ligados, com e sem falhas, à um conjunto de transmissores de potência.

4.1.4 Aplicação 4 - Modelo de validação de sinal utilizando ANFIS

NA *et al.* (2002) desenvolveram um sistema de validação de sinal baseado em um sistema de inferência nebuloso adaptativo (ANFIS). As técnicas de processamento de sinal utilizadas na implementação deste modelo são apresentadas na tabela 4.4.

Neste modelo, na etapa de pré-processamento é feita a seleção dos sinais de entrada baseada na escolha de variáveis de entrada que estejam fortemente correlacionadas com a variável de saída a ser monitorada, através da utilização de algoritmos genéticos. O GA é utilizado para encontrar um conjunto de variáveis de entrada que minimizam o erro de estimação do sinal de saída do modelo. Os sinais de entrada foram normalizados entre $[0,1]$, a filtragem dos sinais de entrada é feita por ondaletas e a redução da dimensão do espaço das variáveis de entrada é realizada por PCA.

Na etapa de processamento é utilizado um sistema neuro-nebuloso único para cada variável monitorada, do tipo hetero-associativo, com uma saída correspondente à variável monitorada, que é treinado para operar em todas as regiões de operação da planta. O tipo de treinamento do sistema neuro-nebuloso é o de minimização do erro de saída, otimizado por algoritmo genético (GA) para a parte antecedente do sistema ANFIS, enquanto que a parte conseqüente do ANFIS é treinada usando o método dos mínimos quadrados.

Tabela 4.4
Modelo de validação de sinal usando inferência neuro-nebulosa (NA *et al.*, 2002).

Pré-processamento						Processamento				Pós-processamento		
Seleção dos sinais	Normalização dos sinais	Filtragem dos sinais	Redução da dimensão	Métrica	Separação em classes	Modelo	Tipo	Estrutura	Otimização	Detecção da anomalia		Confiança da resposta
										Método	Forma	
-----	Nenhuma	Nenhuma	Nenhuma	Nenhuma	Nenhuma	-----	-----	-----	Nenhuma	-----	-----	Nenhuma
Correlação física	[-1,1]	Digital	PCA	Euclidiana	Centróide (<i>k</i> -means)	Matemático	Auto-associativo	Única	Mínimos quadrados	Erro do sensor (resíduos)	Limiar	FIS
Senso de engenharia	[0,1]	Ondaleta (<i>wavelet</i>)	Ondaleta (<i>wavelet</i>)	Minkowski	AVQ	MLP	Hetero-associativo	Múltipla	Gradiente descendente	Auto-correlação dos resíduos	Lógica de votação	
					Gaussiana	RBF						
Proximidade ao sensor monitorado	[-0.5,0.5]	Filtro Kalman	Rede neural	Mahalanobis	FCM	Sistema especialista			GA	Espectro de potência dos resíduos	Lógica booleana	
					GAFCM							
Correlação usando GA				Manhattan	PCM	FIS				SPRT dos resíduos	Lógica nebulosa	
Sinais redundantes					FCM Crescente	<i>Neuro-fuzzy</i>				Filtro Kalman dos resíduos		

Na etapa de pós-processamento do modelo, é feita, a cada instante, a detecção de anomalia de sensor pelo teste da razão de probabilidades seqüencial do resíduo entre o valor estimado pelo modelo e o valor medido das variáveis monitoradas. Se esta razão ultrapassar um limiar pré-estabelecido é gerado um sinal de falha de sensor. Este modelo não fornece qualquer indicação sobre a confiança de sua resposta.

Este sistema foi utilizado para validar os sinais de nível e pressão do pressurizador e da temperatura da água na perna quente de um dos *loops* de um simulador de planta de potência nuclear. Este simulador está localizado na Coréia do Sul e utiliza o código MARS, que é uma versão unificada do COBRA/TF e RELAP/MOD3.

4.2 O modelo de validação de sinal proposto

No item 4.1 foram detalhados quatro modelos de validação de sinal implementados através da metodologia proposta no capítulo 2. Com base nos resultados obtidos nos ensaios realizados no capítulo 3 em cada sub-etapa que compõe um modelo de validação de sinal e na aplicação a que o sistema de validação de sinal se destina, foi definido o modelo implementado nesta tese. Os ensaios realizados teve por objetivo fornecer subsídios para a definição de um sistema de validação de sinal que utilize em cada uma das sub-etapas de sua composição a melhor técnica disponível para implementação. A seguir é descrito sucintamente como foi o processo de definição do sistema de validação de sinal implementado nesta tese.

A aplicação a que se destina o sistema é a validação de sinais da planta nuclear de potência descrita no capítulo 5 desta tese. Nesta aplicação são disponíveis como entrada para o sistema de validação somente 12 sinais que estão distribuídos em diferentes *loops* da planta, não havendo sinais redundantes.

O modelo de validação de sinal implementado nesta tese é apresentado na tabela 4.5.

No modelo implementado, na etapa de pré-processamento é feita a seleção dos sinais de entrada baseada na escolha de variáveis de entrada que estejam fortemente correlacionadas com a variável de saída a ser monitorada, através da utilização de otimização por algoritmos genéticos. O GA é utilizado para encontrar um conjunto de variáveis de entrada que minimizam o erro de estimação do sinal de saída do modelo. Os sinais de entrada foram normalizados entre $[0,1]$, a filtragem dos sinais de entrada é feita por ondaletas e a separação em classes é realizada por conjuntos nebulosos estendidos (EFCM) utilizando como métrica a distância de Mahalanobis, onde toda a região de operação da planta é dividida em diversas sub-regiões.

Na etapa de processamento, para cada variável monitorada, são utilizados vários sistemas neuro-nebulosos (ANFIS), do tipo hetero-associativas, com uma saída correspondente à variável monitorada, onde é treinado um ANFIS específico para operar em cada uma dessas sub-regiões. O tipo de treinamento dos sistemas neuro-nebulosos é o de minimização do erro de saída, otimizado por enxame de partículas (PSO) para a parte antecedente do sistema ANFIS, enquanto que a parte conseqüente do ANFIS é treinada usando o método dos mínimos quadrados.

Na etapa de pós-processamento do modelo é feita, a cada instante, a detecção de anomalia de sensor através de uma combinação lógica do tipo OU de três métodos de detecção de anomalia: o resíduo de estimação da variável monitorada (erro de sensor), a auto-correlação dos resíduos de estimação e o teste da razão de probabilidades seqüencial do resíduo de estimação. Se qualquer dos métodos de detecção de anomalia produzir um sinal de desvio que ultrapasse o seu limiar pré-estabelecido é gerado um sinal de falha de sensor. Este modelo também fornece uma indicação de confiança de

Tabela 4.5
Modelo proposto de validação de sinal.

Pré-processamento						Processamento				Pós-processamento		
Seleção dos sinais	Normalização dos sinais	Filtragem dos sinais	Redução da dimensão	Métrica	Separação em classes	Modelo	Tipo	Estrutura	Otimização	Detecção da anomalia		Confiança da resposta
										Método	Forma	
-----	Nenhuma	Nenhuma	Nenhuma	Nenhuma	Nenhuma	-----	-----	-----	Nenhuma	-----	-----	Nenhuma
Correlação física	[-1,1]	Digital	PCA	Euclidiana	Centróide (<i>k-means</i>)	Matemático	Auto-associativo	Única	Mínimos quadrados	Erro do sensor (resíduos)	Limiar	FIS
Senso de engenharia	[0,1]	Ondaleta (<i>wavelet</i>)	Ondaleta (<i>wavelet</i>)	Minkowski	AVQ	MLP	Hetero-associativo	Múltipla	Gradiente descendente	Auto-correlação dos resíduos	Lógica de votação	
					RBF	AVQ						
Proximidade ao sensor monitorado	[-0.5,0.5]	Filtro Kalman	Rede neural	Mahalanobis	FCM	Sistema especialista			GA	Espectro de potência dos resíduos	Lógica booleana	
					GAFCM							
Correlação usando GA				Manhattan	PCM	FIS			PSO	SPRT dos resíduos	Lógica nebulosa	
Sinais redundantes					FCM Crescente	Neuro-fuzzy				Filtro Kalman dos resíduos		
					EFCM							

sua resposta realizado por um sistema de inferência nebulosa. Este sistema recebe, a cada instante às suas entradas, os sinais de resíduo do erro, da auto-correlação dos resíduos do erro, do teste da razão de probabilidades seqüencial do resíduo do erro e o sinal de pertinência possibilística da amostra no conjunto de treinamento do modelo, e produz à sua saída uma indicação do grau de confiança da resposta produzida pelo modelo.

4.3 Descrição detalhada do modelo implementado

A seguir são detalhados cada um dos blocos que compõem o modelo de validação de sinal implementado.

4.3.1 Seleção dos sinais de entrada do modelo

Diversas técnicas de seleção de variáveis de entrada utilizadas em sistemas de validação de sinal foram descritas no capítulo 2. Dentre essas técnicas as que buscam obter sinais de entrada com alto grau de correlação entre eles, usando algoritmos de otimização por GA ou PSO, são interessantes, pois não necessitam de um especialista do processo monitorado. No modelo de validação implementado nesta tese é utilizada para seleção das variáveis de entrada do modelo a técnica de otimização por GA, descrita no item 3.1.1. O GA é utilizado para encontrar um conjunto de variáveis de entrada que minimizam o erro de estimação do sinal de saída do modelo.

4.3.2 Normalização dos sinais de entrada do modelo

A normalização dos sinais é efetuada antes de treinar ou operar o sistema. Deste modo, tanto o processo de treinamento quanto de operação é realizado num hipercubo $[0,1]^n$, onde n é o número de sinais envolvidos no sistema de validação de sinal. A

normalização utilizada no sistema de validação de sinal implementado é dada pela seguinte equação

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (4.1)$$

onde x , x_{norm} , x_{max} e x_{min} são, respectivamente, o sinal normal, o sinal normalizado, o valor máximo do sinal e o valor mínimo do sinal.

4.3.3 Métrica utilizada

Diversas métricas pertencentes ao espaço Euclidiano foram apresentadas no capítulo 2. Para o sistema de validação de sinal implementado foi escolhida a distância Mahalanobis para realização da separação em classes. A razão para usar esta função distância é que ela não varia quando sujeita a qualquer transformação linear não singular, isto é, os resultados obtidos com esta função distância não dependem das coordenadas utilizadas no sistema.

4.3.4 Filtragem dos sinais de entrada do modelo

Para filtragem dos sinais aplicados ao sistema de validação de sinal foram implementados três métodos: média móvel, média ponderada e ondaletas. Os resultados obtidos nos testes efetuados revelaram desempenho similares para os filtros digitais por média móvel e por ondaletas. O filtro digital por ondaletas foi selecionado para utilização no sistema de validação de sinal implementado.

As ondaletas utilizadas pelo filtro digital são do tipo estacionária (SWT). Estas ondaletas são promissoras para aplicações em tempo real de filtragem de sinal.

Para diminuir o erro de estimação da ondaleta., foi utilizada uma janela de tempo que contém 8 amostras, a amostra atual e 7 amostras passadas do sinal a ser filtrado. Este filtro digital é representado por

$$\bar{x}_k = f((x_k, x_{k-1}, x_{k-2}, \dots, x_{k-J}), L) \quad (4.2)$$

onde \bar{x}_k é o valor médio da variável da amostra atual k obtido pelo filtro e f representa a função ondaleta a ser aplicada nas amostras dentro da janela móvel $x_k, x_{k-1}, \dots, x_{k-J}$ com nível de decomposição L , e J é o tamanho da janela móvel, dado em potência de 2, ou seja, $J=1,2,4$ ou 8.

A tabela 4.6 apresenta as características dos filtros digitais do tipo SWT utilizados pelo sistema de validação de sinal.

Tabela 4.6

Características da ondaleta utilizada no sistema de validação de sinal implementado.

Característica	Descrição
Função	Transformada por ondaleta estacionária (SWT - <i>Stationary Wavelet Transform</i>)
Tipo e forma da ondaleta	Daubechies de ordem 5
Nível de decomposição	$L = \log_2 (J)$

Para diminuir o tempo de resposta dos filtros digitais implementados é utilizado o seguinte procedimento: caso a diferença entre o valor médio da variável correspondente à amostra anterior e o valor da variável da amostra atual seja maior ou menor, em duas amostras consecutivas, do que duas vezes a raiz quadrada do valor médio da variável correspondente à amostra anterior, ou seja, caso $(x_k - \bar{x}_{k-1}) > 2\sigma$ ou $(x_k - \bar{x}_{k-1}) < 2\sigma$, em duas amostras consecutivas, onde $\sigma = \sqrt{\bar{x}_{k-1}}$, então, o valor atual da amostra é assumido como

valor médio e a janela J é automaticamente reinicializada em 1, sendo aumentada em potências de 2, a cada conjunto de potência de 2 de novas amostras, até atingir o tamanho de janela J definido pelo usuário. O valor de L é feito igual ao logaritmo na base 2 de J .

4.3.5 Separação em classes dos sinais de entrada do modelo

O método de separação em classes utilizado no sistema de validação de sinal implementado é o método de separação em classes estendidas (EFCM) proposto por KAYMAK *et al.* (2000), onde os centróides das classes de uma partição nebulosa são representados por um volume. Os centróides das classes (protótipos) com volume estendem os centróides das classes (protótipos) de classes de pontos para regiões no espaço a ser separado em classes.

A principal vantagem de usar protótipos com volume é a redução na sensibilidade resultante do algoritmo de separação em classes a diferentes volumes de classes e a distribuição dos padrões dos dados. Isto torna os algoritmos de separação em classes mais robusto. Frequentemente, vários pontos do conjunto de dados próximos ao centróide de uma classe podem ser considerados como pertencentes completamente a esta classe. Isto é evidenciado quando existem classes que são bem separadas umas das outras. Neste caso, é interessante estender o núcleo de uma classe de um único ponto para uma região no espaço.

Os sistemas dinâmicos, como uma planta nuclear, podem gerar mais dados em certas regiões do espaço de estado do que em outras. Considere, por exemplo, um sistema não linear auto-regressivo dado por

$$\tilde{y}_k = y_k + 0,03e_k \quad (4.3)$$

$$y_{k+1} = \frac{\text{sen}(10\tilde{y}_k)}{10\tilde{y}_k} + 0,03e_k \quad (4.4)$$

onde e_k representa uma distribuição normal de ruído aleatória.

A figura 4.1 apresenta a distribuição dos 1000 pontos no espaço de estado com $y_0 = 0$, onde foi aplicado o algoritmo EFCM com métrica Mahalanobis. O resultado da separação em classes detectou 4 classes e as posições das classes mostram a forma geral da função escondida. Os centróides das classes estão posicionados nas regiões densas de dados. As regiões esparsas são também cobertas pelos alcances das classes. Um sistema de inferência nebulosa adaptativo (ANFIS) pode ser utilizado para modelar os dados em cada uma das classes. Este modelo é objeto do próximo item deste capítulo.

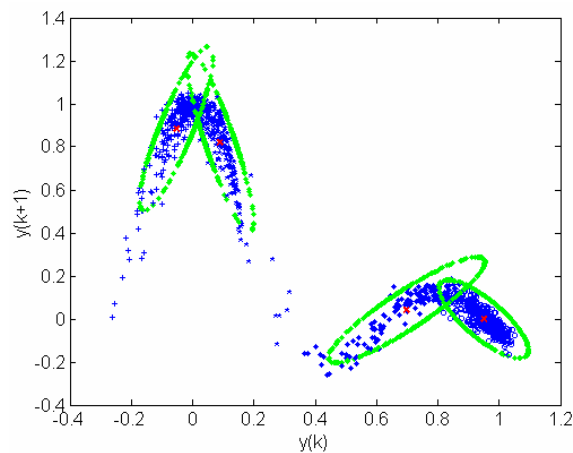


Figura 4.1 - Protótipos das classes obtidas pelo EFCM com métrica Mahalanobis para um sistema dinâmico auto-regressivo.

4.3.6 Modelo de validação de sinal

O módulo mais importante que compõe o modelo de validação de sinal implementado é o sistema de inferência nebulosa adaptativo (ANFIS). O ANFIS é um sistema de inferência nebulosa (FIS) que pode ser treinado para modelar uma coleção de dados entrada/saída. O processo de treinamento permite ao sistema ajustar seus parâmetros para aprender as relações de entrada/saída contida no conjunto de dados.

O sistema de inferência neuro-nebulosa usado nesta tese para validação de sinal é apresentado na figura 4.2. O ANFIS descrito aqui usa o modelo TSK. No modelo TSK, é usada uma única função pertinência à saída do modelo em vez de um conjunto nebuloso, como usado no modelo de inferência nebulosa do tipo Mamdani.

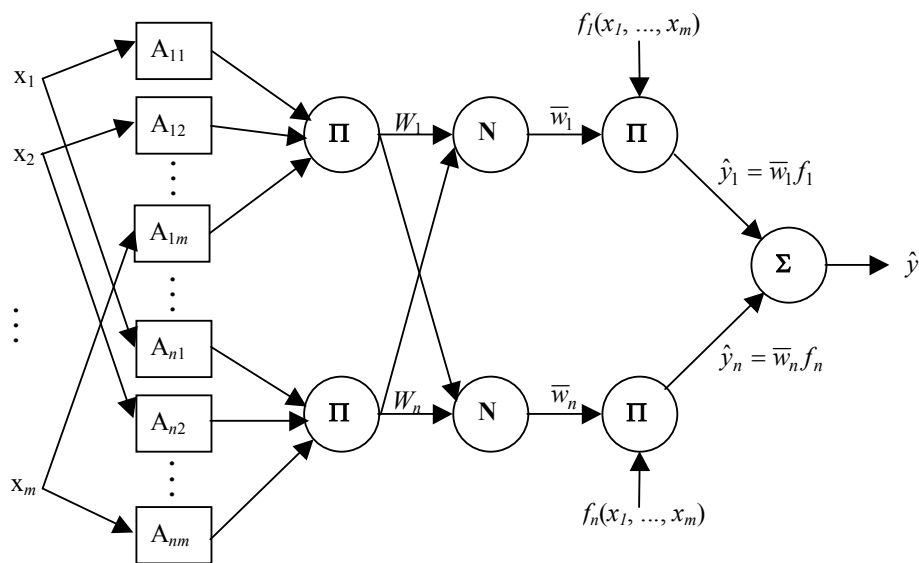


Figura 4.2 - Sistema de inferência neuro-nebuloso.

Uma regra nebulosa típica usando o modelo de primeira ordem TSK é dada por

$$\text{Se } x_1 \text{ é } A_{i1} \text{ E } \dots \text{ E } x_m \text{ é } A_{im} \text{ então } y_i \text{ é } f_i(x_1, \dots, x_m) \tag{4.5}$$

onde x_1, \dots, x_m são as variáveis de entrada para o sistema de inferência neuro-nebuloso (m é o número de variáveis de entrada), A_{i1}, \dots, A_{im} são funções de pertinência da parte antecedente de cada variável de entrada para a i -ésima regra ($i = 1, 2, \dots, n$) e y_i é a saída da i -ésima regra.

A parte consequente f_i das regras do modelo TSK té dada por

$$f_i(x_1, \dots, x_m) = \sum_{j=1}^m q_{ij} x_j + r_i \quad (4.6)$$

onde q_{ij} é o valor do peso da j -ésima entrada na i -ésima regra de saída, r_i é a polarização da i -ésima saída e n é o número de regras.

No modelo implementado é usado para cada variável de entrada a função pertinência do tipo Gaussiana que é dada por

$$A_{ij}(x_j) = e^{-\frac{(x_j - c_{ij})^2}{2s_{ij}^2}} \quad (4.7)$$

onde c_{ij} e s_{ij} são, respectivamente, os valores do centróide e do desvio padrão de cada variável de entrada x_j na i -ésima função pertinência.

A saída de uma i -ésima regra arbitrária, f_i , consiste de um polinômio de primeira ordem de entradas como dado na equação (4.6). A saída de um sistema de inferência nebulosa com n regras é obtida pelo ponderação dos valores das partes consequentes para todas as regras com os correspondentes graus de pertinência. A saída estimada para o sinal de sensor é descrita por

$$\hat{y} = \sum_{i=1}^n \bar{w}_i f_i \quad (4.8)$$

onde

$$\bar{w}_i = \frac{w_i}{\sum_{i=1}^n w_i} \quad (4.9)$$

e

$$w_i = \prod_{j=1}^m A_{ij}(x_i) \quad (4.10)$$

Os valores de entrada para o sistema de inferência nebulosa são x_1, x_2, \dots, x_m , e A_{ij} significa a função pertinência da j -ésima regra. O valor de pertinência para a regra i , w_i , significa o grau de compatibilidade entre as partes antecedentes através de pesos multiplicativos. Se a figura 4.2 for percorrida da esquerda para a direita, os símbolos Π e N significam multiplicação e normalização, que são expressos pelas equações (4.10) e (4.9), respectivamente. O segundo símbolo Π e o símbolo Σ são expressos na equação (4.8). O sinal Σ significa o somatório dos valores de entrada.

Na seção 4.4 o sistema de inferência nebulosa será treinado, significando que suas regras e funções pertinências serão automaticamente sintonizadas.

4.3.7 Detecção de anomalia

A detecção de anomalia ou falha em sensor em um sistema de validação de sinal é baseada no sinal de erro entre o valor estimado pelo sistema e medido pelo sensor monitorado. A informação de falha é gerada quando este erro ultrapassa um limiar estipulado.

Para a detecção de anomalia ou falha de sensor foram estudados três métodos de detecção: o método do erro ou resíduos, o método da auto-correlação dos resíduos e o método do teste da razão de probabilidades seqüencial dos resíduos. Os resultados dos ensaios realizados mostraram que nenhum dos três métodos implementados foi capaz de

identificar sozinho todas as falhas inseridas. Deste modo, optou-se por efetuar uma combinação desses métodos.

A detecção de anomalia do sistema de validação de sinal implementado nesta tese é feita combinando os métodos de detecção pelo erro (resíduos), pelo teste da razão de probabilidades seqüencial dos resíduos (SPRT) e pela auto-correlação dos resíduos, através de uma lógica de votação tipo OU, conforme apresentado na figura 4.3. A lógica de votação é dada por

$$Alarme = \max(Alarme_{erro}, Alarme_{sprt}, Alarme_{auto}) \quad (4.11)$$

onde *Alarme* é o sinal resultante do OU (*max*) entre os sinais de alarme pelo método de erro (*Alarme_{erro}*), alarme pelo método SPRT (*Alarme_{sprt}*) e alarme pelo método auto-correlação (*Alarme_{auto}*).

As seguintes bandas de alarme são utilizadas pelos métodos de detecção de anomalia:

- a) Método de erro: É ajustada para ± 4 desvios padrão ($\pm 4 \sigma_\varepsilon$) do sinal de erro ε obtido na etapa de treinamento. A ultrapassagem desta banda pelo sinal de erro é considerada como um alerta definitivo;
- b) Método SPRT: É ajustada para ser igual ao valor B , calculado em função dos parâmetros α e β , descritos a seguir. A ultrapassagem desta banda pela razão de probabilidades é considerada como um alerta definitivo;

A taxa de alarmes perdidos foi feita igual a 0,001 ($\alpha=0,001$) e a taxa de alarmes falsos igual a 0,001 ($\beta=0,001$). Implicando em um valor de contorno $A=-6,9068$ e $B=6,9068$.

O valor da razão de probabilidades seqüencial dos resíduos é calculado pela equação (2.54) onde os valores da média μ_0 e desvio padrão σ_0 , desta equação, são obtidos a partir do conjunto de dados do treinamento do sistema e os valores μ_1 e σ_1 são feitos iguais a 0,025 da excursão do sinal de treinamento e $3\sigma_0$, respectivamente.

c) Método auto-correlação: É ajustada para ± 4 desvios padrão ($\pm 4\sigma_d$) das distâncias do sinal de auto-correlação dos resíduos ε entre o valor monitorado e estimado, obtido na etapa de treinamento. A ultrapassagem desta banda pelo sinal de auto-correlação é considerada como um alerta definitivo.

O tamanho da janela de amostras para cálculo da auto-correlação dos resíduos é obtido durante o processo de treinamento do modelo de validação de sinal.

O valor da auto-correlação dos resíduos é calculado pela equação (2.43) onde os valores da média μ_d e desvio padrão σ_d , desta equação, são obtidos a partir do conjunto de dados do treinamento do sistema.

A lógica de votação tipo OU considera como resposta final do modelo a indicação de falha produzida por qualquer dos métodos de detecção utilizados. Neste caso, se um método de detecção é deficiente na detecção de uma determinada falha e outro não o é, a resposta final do sistema não será comprometida. Por outro lado, qualquer indicação de falso alarme em um dos sistemas redundantes produzirá uma resposta errônea do sistema.

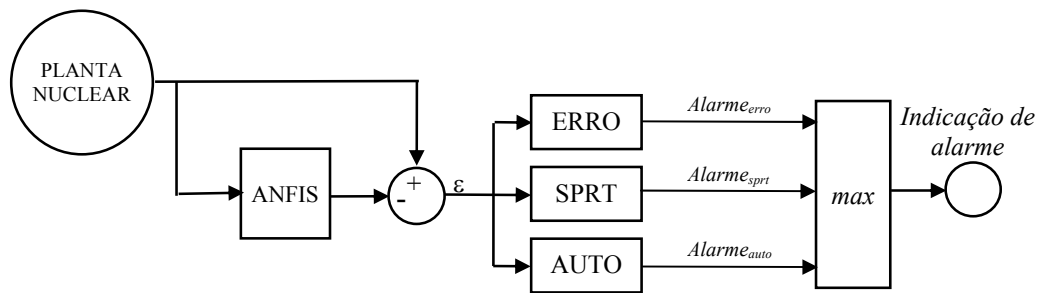


Figura 4.3 - O módulo de detecção de anomalia.

4.3.8 Confiança da resposta

Nesta tese, a função confiança é realizada através de um modelo lingüístico, vide anexo IV, (KOSKO, 1992), (KOSKO, 1997) e (KLIR, 1997), usando um sistema de inferência nebulosa (FIS) do tipo Mamdani. A inferência nebulosa é um processo de formulação de um mapeamento de uma dada entrada em uma saída usando lógica nebulosa. O mapeamento fornece, então, uma base sobre a qual podem ser tomadas decisões ou discernir padrões. O processo de inferência nebulosa envolve: funções pertinências, operadores lógicos nebulosos e regras *se-então*.

O FIS utilizado, figura 4.4, recebe como entradas os sinais de resíduos resultantes da união (*max*) entre os sinais de resíduos produzidos pelos sistemas de detecção de falhas (sinais de erro, sinais de razão da probabilidade seqüencial dos resíduos e sinais de auto-correlação dos resíduos) e os valores da pertinência das amostras de entrada do sistema ao conjunto de treinamento (pertinências possibilísticas das amostras) e fornece à sua saída, para cada amostra de entrada, uma indicação quantitativa sobre o grau da confiança da resposta produzida pelo sistema.

Antes de serem aplicados às entradas do bloco união (*max*), os sinais de entrada são normalizados pelo bloco de normalização N com respeito ao desvio padrão σ_ε , para o sinal

de erro, à condição de contorno B , para o sinal de razão da probabilidade sequencial dos resíduos e ao desvio padrão σ_d , para o sinal de auto-correlação dos resíduos. Essas normalizações são dadas por

$$sinal_{erro} = \frac{sinal\ de\ erro}{10 \times \sigma_\epsilon} \tag{4.12}$$

$$sinal_{spri} = \frac{sinal\ de\ razão\ de\ probabilidade\ sequencial\ dos\ resíduos}{5 \times B} \tag{4.13}$$

$$sinal_{auto} = \frac{sinal\ de\ autocorrelação\ dos\ resíduos}{10 \times \sigma_d} \tag{4.14}$$

onde $sinal_{erro}$ é o sinal de erro normalizado, $sinal_{spri}$ é o sinal de probabilidade sequencial dos resíduos normalizado e $sinal_{auto}$ é o sinal de auto-correlação dos resíduos normalizado.

O sinal de desvio é obtido pela união (max) dos sinais de entrada pela equação

$$Desvio = \max(sinal_{erro}, sinal_{spri}, sinal_{auto}) \tag{4.15}$$

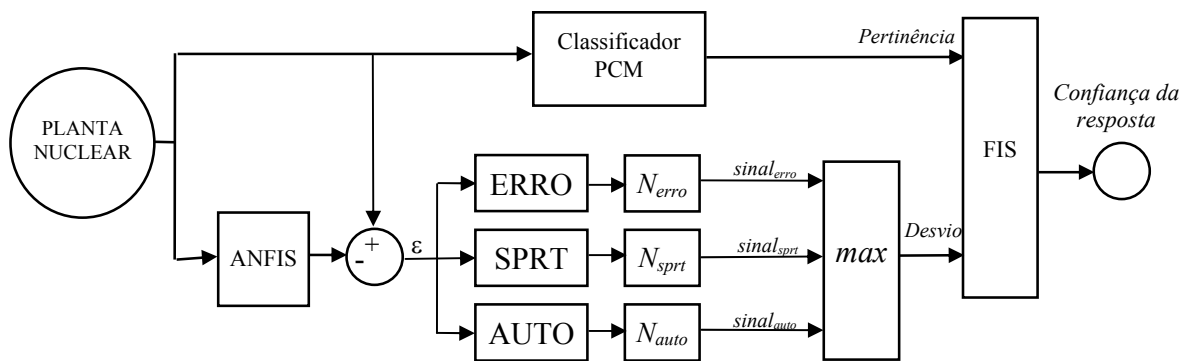


Figura 4.4 - O módulo de confiança da resposta.

O sistema de inferência nebulosa pode ser dividido em cinco partes: nebulização das variáveis de entrada, aplicação dos operadores nebulosos da parte antecedente, implicação da parte antecedente para a parte conseqüente, agregação dos conseqüentes através das

regras e desnebulização. A figura 4.5 mostra uma visão geral da estrutura do FIS usado nesta tese.

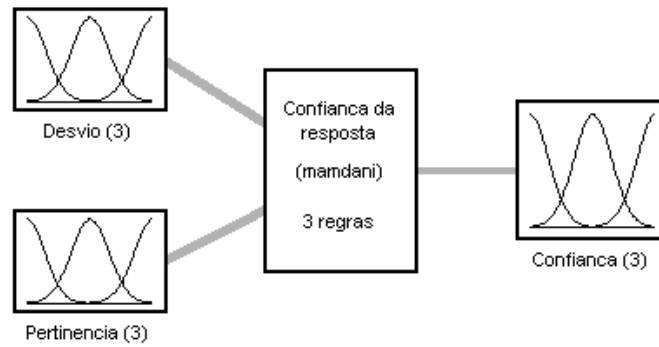


Figura 4.5 - Visão geral da estrutura usada no FIS de confiança da resposta.

A primeira parte é aplicar as entradas e determinar o grau que elas pertencem a cada conjunto nebuloso por meio das funções-pertinência. A figura 4.6 mostra as funções pertinências usadas no FIS do módulo de confiança da resposta.

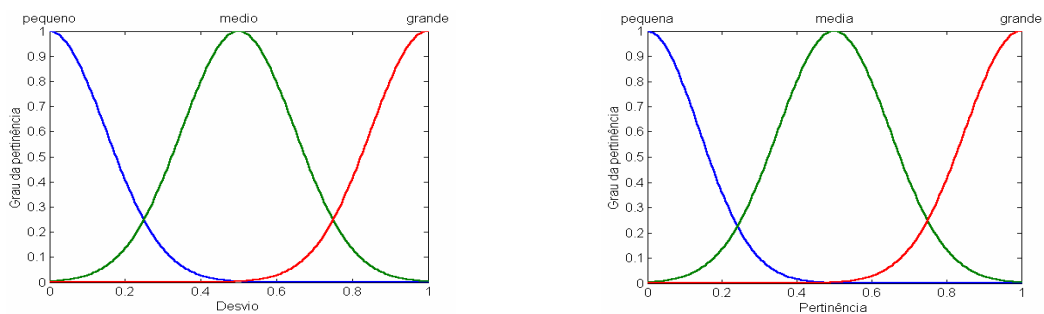


Figura 4.6 - Funções pertinências usadas no FIS do módulo de confiança da resposta.

Uma vez que as entradas foram nebulizadas, sabemos o grau que cada parte do antecedente satisfaz cada regra. Se uma dada regra tem mais que uma parte antecedente, é aplicado o operador nebuloso para obter um número que representa o resultado da parte

antecedente para essa regra. Este número irá, então, ser aplicado à função de saída do sistema nebuloso. Os operadores nebulosos usados nesta tese são o operador mínimo E e o operador máximo OU .

O modelo nebuloso aplica os operadores de implicação tipo Mamdani com as seguintes regras *Se-Então*:

1. **Se** desvio não é pequeno **E** pertinência é pequena **Então** confiança é pequena.
2. **Se** desvio é médio **E** pertinência é média **Então** confiança é média.
3. **Se** pertinência é grande **Então** confiança é grande.

As regras podem ser representadas como uma matriz 3 por 3 conforme mostrado na tabela 4.7.

Tabela 4.7

Regras nebulosas utilizadas.

Pertinência Desvio	Pequena	Média	Grande
Pequeno	Pequena		Grande
Médio		Média	Grande
Grande			Grande

Para cada amostra apresentada ao sistema, as regras nebulosas são disparadas em diferentes graus resultando em três valores de pertinência diferentes para a confiança nos três conjuntos nebulosos: pequeno, médio e grande, conforme mostrado na figura 4.7.

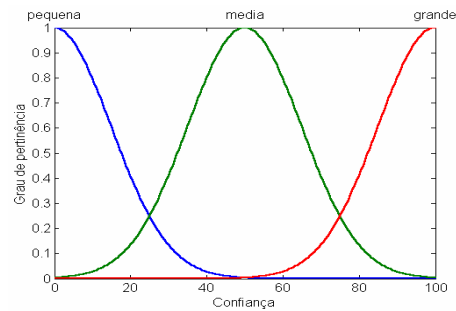


Figura 4.7 - Funções pertinência da parte conseqüente do sistema de confiança da resposta.

Tendo em vista que decisões são baseadas no teste de todas as regras em um sistema de inferência nebulosa, as regras devem ser combinadas de alguma maneira de modo a se obter uma decisão. Agregação é o processo pelo qual os conjuntos nebulosos que representam as saídas de cada regra são combinados em um único conjunto nebuloso. Nesta tese, usamos o método de agregação do tipo máximo.

Finalmente, o processo de denebulização recebe como entrada a saída do conjunto nebuloso obtido pela agregação e produz um único valor como saída (o grau de confiança da resposta do sistema). Nesta tese, usamos o método do maior dos máximos no processo de denebulização. A figura 4.8 apresenta os graus de confiança da resposta do FIS em função dos valores de entrada de desvio e pertinência.

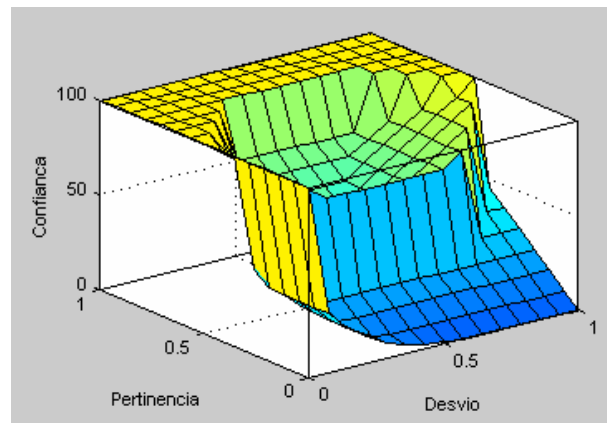


Figura 4.8 - Superfície da confiança da resposta de saída produzida pelo FIS para as entradas de desvio e pertinência.

4.4 Treinamento do sistema de validação de sinal

Nesta etapa todos os parâmetros necessários para operação do sistema de validação de sinal precisam ser obtidos através do conjunto de dados disponíveis para treinamento do sistema. A seguir apresentamos os parâmetros calculados em cada uma das etapas de composição deste sistema.

4.4.1 Etapa de pré-processamento

O primeiro passo do treinamento é efetuar a seleção das variáveis de entrada a serem utilizadas pelo modelo de validação de sinal para estimar a variável a ser monitorada, a partir do conjunto de amostras das variáveis disponíveis da planta para treinamento do sistema.

A seguir deve-se realizar a normalização das amostras das variáveis de entrada e de saída e filtrá-las pelos filtros digitais por ondaletas.

Logo após, deve-se efetuar a separação em classes das amostras dos sinais disponíveis para treinamento dos sistemas de inferência nebulosa adaptativo. A separação é feita de tal maneira que amostras similares sejam colocadas na mesma classe. O critério de similaridade é baseado na distância da amostra ao centróide da classe, representada pelo seu centróide. Neste caso, utilizamos a distância de Mahalanobis como métrica. Como já foi citado, a separação do conjunto de dados de treinamento em classes é efetuada pelo método de separação em classes nebulosas estendida (EFCM). O método EFCM calcula, para cada classe, o seu centróide (equação 2.19) e suas correspondentes amostras.

Nesta etapa, também é feito, para cada classe, o cálculo do parâmetro η_i (equação 2.31) que representa a distância média quadrática das amostras que pertencem a cada classe ao centróide da mesma, ponderada pelas pertinências das amostras.

Os valores de centróides e distâncias média quadrática obtidos serão utilizados, na fase de operação, tanto na etapa de classificação como na etapa de cálculo da confiança da resposta produzida pelo sistema.

4.4.2 Etapa de processamento

Após as amostras estarem separadas em classes, é feito o treinamento dos sistemas de inferência nebulosa adaptativo especializados. São treinados tantos ANFIS quantas forem as classes existentes. Cada ANFIS é treinado com as amostras pertencentes a uma determinada classe. Os ANFIS, após estarem treinados, serão utilizados na etapa de operação do sistema. A descrição detalhada do treinamento de um ANFIS é apresentada a seguir.

4.4.2.1 Construindo o ANFIS inicial

Antes do treinamento, deve ser gerado o ANFIS inicial para cada classe de amostras. A estrutura inicial do ANFIS é gerada, a partir do conjunto de dados contidos na classe, utilizando o método de separação em classes nebulosas estendida (EFCM). O EFCM é usado para gerar um FIS inicial para treinamento ANFIS primeiramente realizando a separação em classes nebulosas do conjunto de amostras da classe. Isto é feito pela extração de um conjunto de regras que modelam o comportamento dos dados. O método de extração de regras primeiramente determina o número de regras a serem usadas no ANFIS como sendo igual ao número de classes determinada pelo algoritmo EFCM do conjunto de dados. O EFCM também é utilizado para determinar as funções-pertinência iniciais da parte antecedente do ANFIS. As equações das regras da parte conseqüente do ANFIS são determinadas através de método de mínimos quadrados aplicado ao conjunto de dados. O ANFIS inicial é uma estrutura que contém um conjunto de regras nebulosas que representam o espaço de realizações.

A arquitetura ANFIS usada, onde o ANFIS gerado não é completamente conectado, conforme mostrado na figura 4.2, evita a explosão do número de regras nebulosas popularmente conhecida como maldição da dimensionalidade. Nesta arquitetura o número de regras gerados é igual ao número de funções-pertinência de cada entrada.

A estrutura do ANFIS usada pode ser resumida do seguinte modo:

- a) Entradas: o ANFIS inicial tem o mesmo número de entradas que o número de variáveis de entrada a serem usadas no seu treinamento;
- b) Saída: o ANFIS inicial tem somente uma saída que corresponde à variável monitorada;

- c) Funções-pertinência de entrada: a função pertinência do tipo gaussiana é usada em todas as entradas. O número de funções-pertinência em cada entrada é igual ao número de regras;
- d) Número de regras: o número de regras usados no ANFIS é igual ao número de classes obtidas no conjunto de dados.

Como já foi citado, para determinar o número de regras do ANFIS inicial foi usado o algoritmo de separação em classes EFCM no conjunto de dados.

4.4.2.2 Treinando o ANFIS

O sistema de inferência neuro-nebulosa é otimizado pela adaptação dos parâmetros antecedentes (parâmetros das funções-pertinência) e parâmetros conseqüentes (os coeficientes dos polinômios da parte conseqüente) de tal modo que uma função objetivo específica seja minimizada. Os métodos de adaptação da maioria dos sistemas de inferência nebulosa se baseiam no algoritmo de retropropagação que é geralmente usado para resolver recursivamente a otimização dos parâmetros. Este algoritmo de otimização convencional é susceptível a ficar preso a um ótimo local. De modo a contornar esta deficiência têm sido usadas técnicas evolucionárias, tal como, algoritmos genéticos. Contudo, essas técnicas necessitam muito tempo computacional se existem muitos parâmetros a serem otimizados. Deste modo, o método dos mínimos quadrados, que é um método de otimização de passagem única, é combinado para otimizar uma parte dos parâmetros. A técnica evolucionária é usada para otimizar os parâmetros antecedentes C_{ij} e S_{ij} e o algoritmo dos mínimos quadrados é usado para resolver os parâmetros conseqüentes q_{ij} and r_i . Nesta tese,

sugerimos usar a técnica de otimização por enxame de partículas para treinar os parâmetros antecedentes do sistema de inferência nebulosa.

Para todos os métodos de treinamento, um passo no procedimento de aprendizado é dividido em duas etapas. Na primeira etapa do ciclo de treinamento os padrões de entrada (amostras) são propagados para a saída do FIS, e os parâmetros ótimos da parte conseqüente são estimados pelo método dos mínimos quadrados, enquanto que os parâmetros da parte antecedente (funções-pertinência) são mantidos fixos. Na segunda etapa os padrões são novamente propagados e, neste momento, é usado retropropagação, algoritmo genético ou otimização por enxame de partículas para modificar os parâmetros antecedentes, enquanto que os parâmetros conseqüentes permanecem fixos. Este procedimento híbrido é, então, repetido iterativamente até que o erro de estimação do FIS seja reduzido a um valor desejado ou até que seja alcançado um número máximo de ciclos de treinamento.

O desempenho de cada um desses métodos de treinamento pode ser observado nos ensaios realizados no item 3.5 do capítulo 3 desta tese. No sistema de validação de sinal implementado nesta tese é utilizado o método de treinamento pela técnica de otimização por enxame de partículas para treinar os parâmetros antecedentes e estimação por mínimos quadrados (LSE) para determinar os parâmetros conseqüentes do ANFIS.

O anexo III apresenta um resumo sobre as técnicas de otimização por mínimos quadrados, gradiente descendente, algoritmos genéticos e enxame de partículas.

4.4.3 Etapa de pós-processamento

Nesta etapa, são calculados, para cada classe, a partir do erros de estimação do sinal monitorado pelo ANFIS, após seu treinamento, os seguintes valores para detecção de anomalia do sensor e cálculo da confiança da resposta do sistema:

- a) Método de erro (resíduos): média μ_ε e desvio padrão σ_ε dos resíduos ε entre o valor monitorado e estimado pelo sistema do sensor para o conjunto de dados de treinamento;
- b) Método da auto-correlação dos resíduos: distância média (\bar{d}) e desvio padrão (σ_d) das distâncias do sinal de auto-correlação dos resíduos ε entre o valor monitorado e estimado pelo sistema do sensor para o conjunto de dados de treinamento;
- c) Método SPRT: média μ_θ e desvio padrão σ_θ dos resíduos ε entre o valor monitorado e estimado do sensor pelo sistema do sensor para o conjunto de dados de treinamento.

Os valores obtidos serão utilizados na etapa de operação do sistema para a detecção de anomalia no sensor monitorado. Para a medida de confiança de resposta do sistema além desses valores é utilizado o valor da distância média quadrática obtida na etapa de separação em classes das amostras de treinamento.

4.5 Operação do sistema de validação de sinal

A figura 4.9 apresenta o diagrama em blocos detalhado do modelo de validação de sinal implementado nesta tese. Nesta figura estão representadas de maneira esquemática todas as etapas e sub-etapas que compõem o modelo implementado.

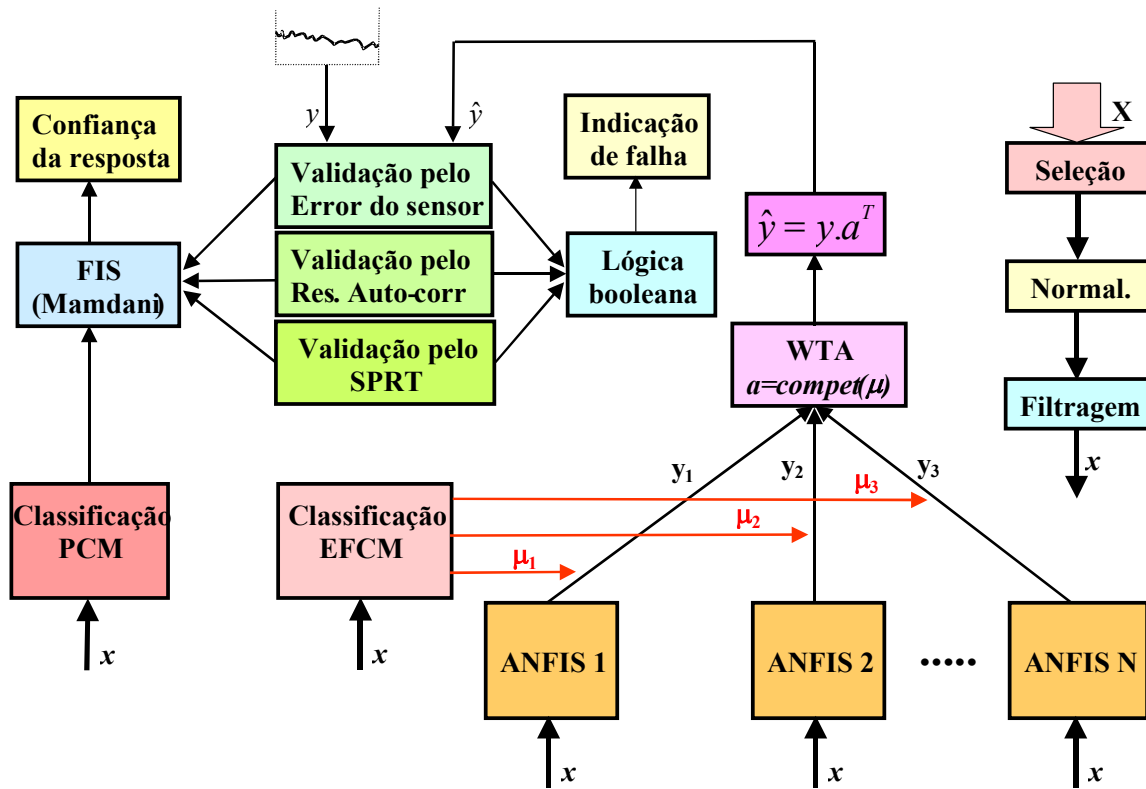


Figura 4.9 - Diagrama em blocos detalhado do modelo de validação de sinal implementado.

Na fase de operação, o sistema recebe, ao longo do tempo, os x sinais de entrada para os ANFIS provenientes da planta monitorada. Esses sinais foram selecionados, durante a fase de treinamento do sistema, a partir do conjunto de variáveis de entrada disponíveis X .

A seguir os sinais de entrada selecionados são normalizados entre $[0,1]$ e filtrados pela técnica de ondaletas.

Logo após, os sinais são enviados ao classificador que faz a classificação da amostra de entrada em uma determinada classe. Esta classificação é feita através da medida da distância da amostra aos centróides das classes existentes. A amostra é associada à classe cujo centróide estiver mais próximo, figura 4.10. O cálculo de proximidade é feito em relação ao parâmetro η_i que foi calculado, na etapa de treinamento, para cada classe.

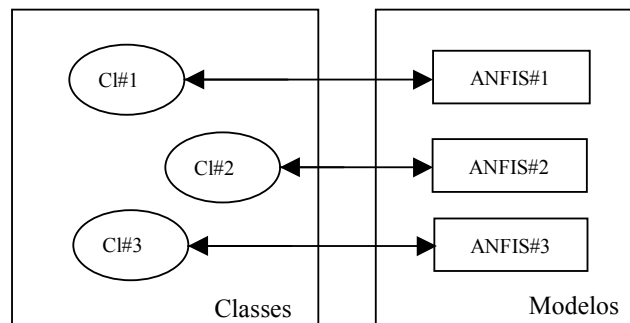


Figura 4.10 - Separação em três classes com os ANFIS associados.

Após a classificação, a amostra é enviada para o ANFIS correspondente à classe que foi selecionada, figura 4.11. O sinal de saída gerado pelo ANFIS selecionado, a partir da amostra de entrada, é enviado à saída do modelo.

Caso haja uma diferença considerável entre o sinal monitorado e o sinal estimado pelo modelo, é gerado pelo sistema uma informação de alarme. E através da informação da confiança da resposta, é possível identificar se é uma falha no sinal monitorado ou se é uma condição de operação da planta para qual o sistema não foi treinado.

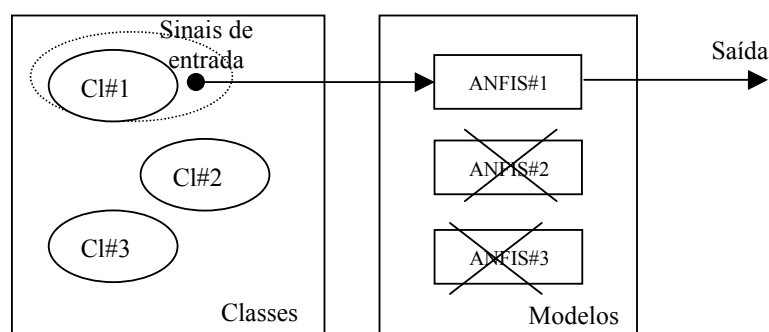


Figura 4.11 - Processo de operação de um modelo com três ANFIS.

A detecção de anomalia de sensor é feita através de uma combinação lógica do tipo OU de três métodos de detecção de anomalia: o erro ou resíduos de estimação da variável monitorada, a auto-correlação dos resíduos de estimação e o teste da razão de probabilidades seqüencial dos resíduos de estimação. Se qualquer dos métodos de detecção de anomalia produzir um sinal de desvio que ultrapasse o seu limiar pré-estabelecido é gerado um sinal de falha de sensor.

A indicação de confiança da resposta é realizada por um sistema de inferência nebulosa. Este sistema recebe a cada instante, às suas entradas, os sinais de resíduo do erro, da auto-correlação dos resíduos do erro, do teste da razão de probabilidades seqüencial do resíduo do erro e o sinal de pertinência possibilística da amostra no conjunto de treinamento do modelo, e produz à sua saída uma indicação do grau de confiança da resposta produzida pelo modelo.

CAPÍTULO 5

Aplicação do modelo de validação de sinal a um reator nuclear

Neste capítulo são descritos e apresentados os resultados dos ensaios realizados no modelo de validação de sinal implementado nesta tese para validar os sinais da potência nuclear e da temperatura da água na perna quente de uma planta nuclear de potência.

5.1 As variáveis monitoradas da planta nuclear de potência

O conjunto de dados utilizados para ensaiar o modelo de validação de sinal proposto nesta tese foi obtido da operação da planta nuclear de potência Angra I. Angra I é uma instalação nuclear para geração de energia elétrica do tipo à água pressurizada (PWR) fabricada pela Westinghouse e está localizada na costa sudeste do Brasil. Esta planta possui dois circuitos de geração de vapor e produz, aproximadamente, 600 MW elétricos, em operação a plena potência.

A figura 5.1 apresenta o diagrama esquemático do circuito 1 de geração de vapor desta usina, com os pontos de monitoração das variáveis utilizadas nos ensaios realizados.

A tabela 5.1 apresenta a lista das 12 variáveis da usina Angra I que foram selecionadas para serem utilizadas pelo modelo de validação de sinal. As variáveis foram selecionadas com base no nosso bom senso (de engenharia). Estas variáveis foram adquiridas através do computador de processos desta planta. O computador de processos da usina Angra I armazena o valor das variáveis de processo à intervalos de 2 segundos entre cada amostragem.

De modo a cobrir toda a excursão de potência da planta e não gerar uma quantidade excessiva de dados para o treinamento do modelo de validação de sinal, foram adquiridas medidas das variáveis da planta de dois em dois minutos, totalizando

1551 amostras. As variáveis foram monitoradas durante uma das partidas da planta nuclear até atingir 100% de potência. As figuras 5.2 a 5.13 apresentam os gráficos dos valores das variáveis adquiridas da planta nuclear.

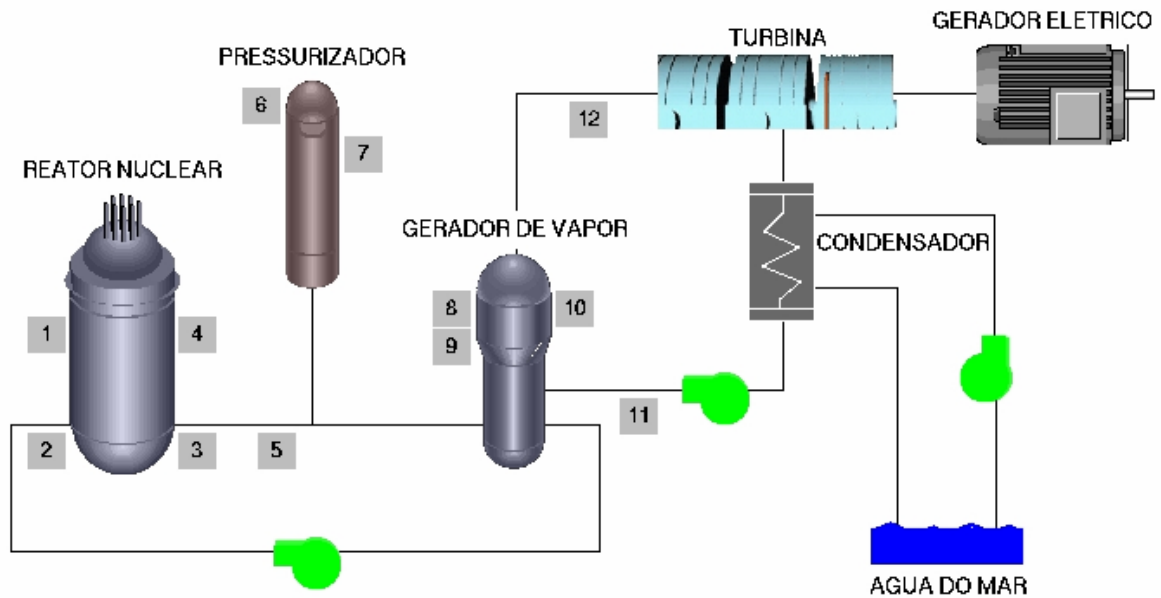


Figura 5.1 Posição dos sensores na usina nuclear Angra I.

Tabela 5.1

Variáveis monitoradas da usina nuclear Angra I

	Variável	Unidade
u1	Potência nuclear	(%)
u2	Temperatura da água na perna fria	(°C)
u3	Temperatura da água na perna quente	(°C)
u4	Temperatura média da água do núcleo	(°C)
u5	Vazão da água do núcleo	(%)
u6	Pressão no pressurizador	(kg/cm ²)g
u7	Nível de água no pressurizador	(%)
u8	Nível de água no gerador de vapor (Faixa estreita)	(%)
u9	Nível de água no gerador de vapor (Faixa larga)	(%)
u10	Pressão no gerador de vapor	(kg/cm ²)g
u11	Vazão da água de alimentação	(kg/s)
u12	Vazão de vapor	(kg/s)

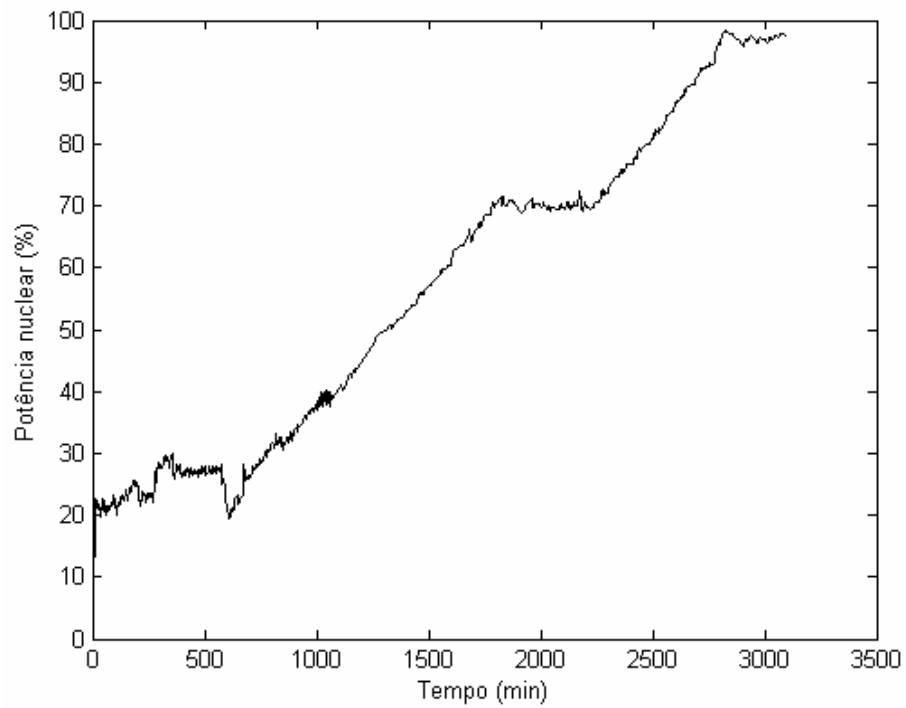


Figura 5.2 - Potência nuclear da usina Angra I.

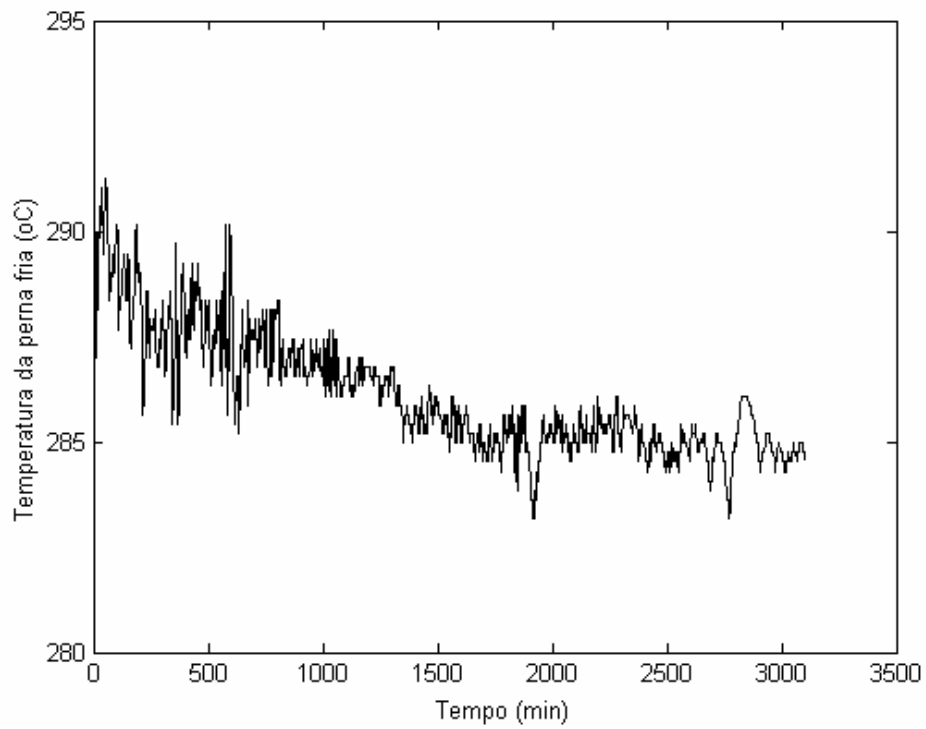


Figura 5.3 - Temperatura da água na perna fria do circuito 1 da usina Angra I.

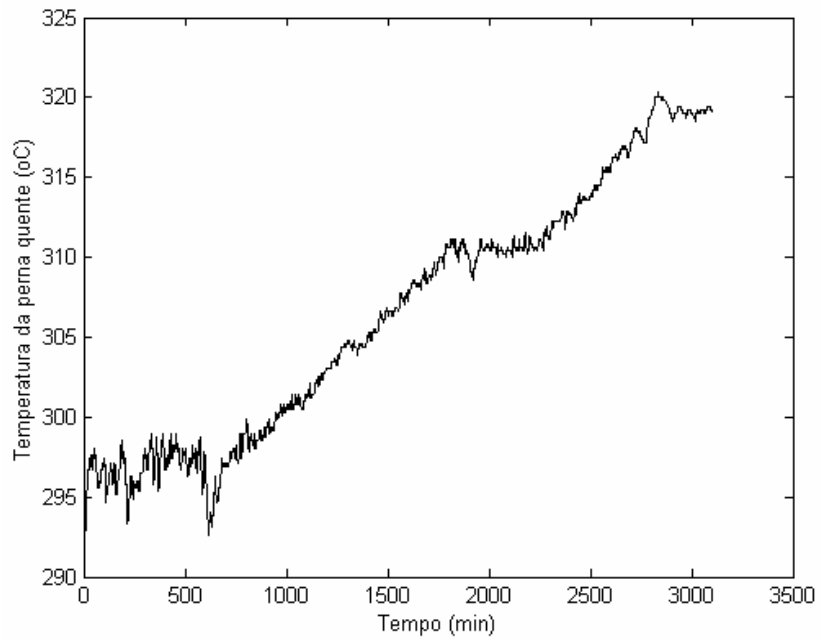


Figura 5.4 - Temperatura da água na perna quente do circuito 1 da usina Angra I.

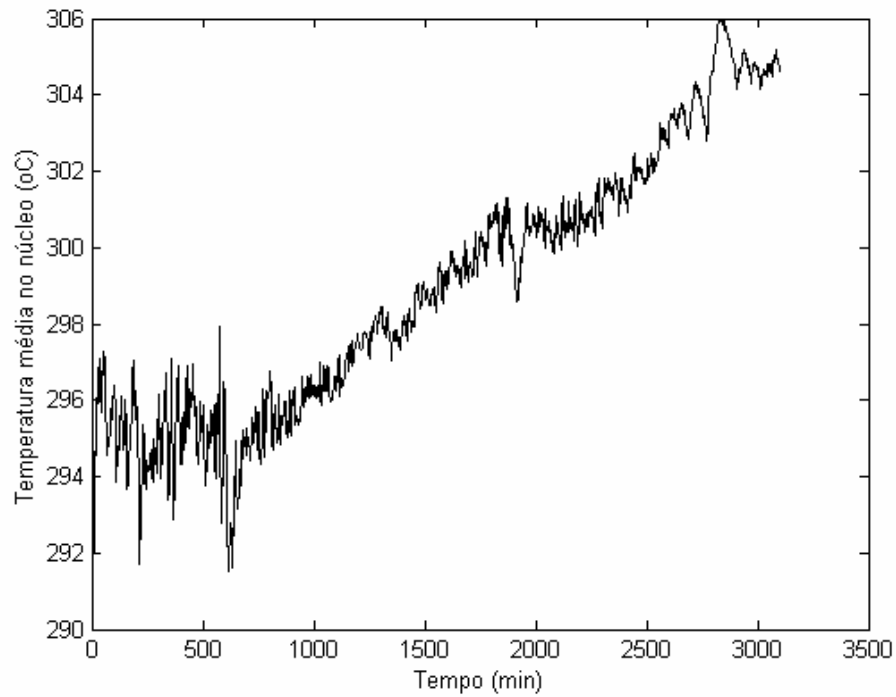


Figura 5.5 - Temperatura média da água no núcleo da usina Angra I.

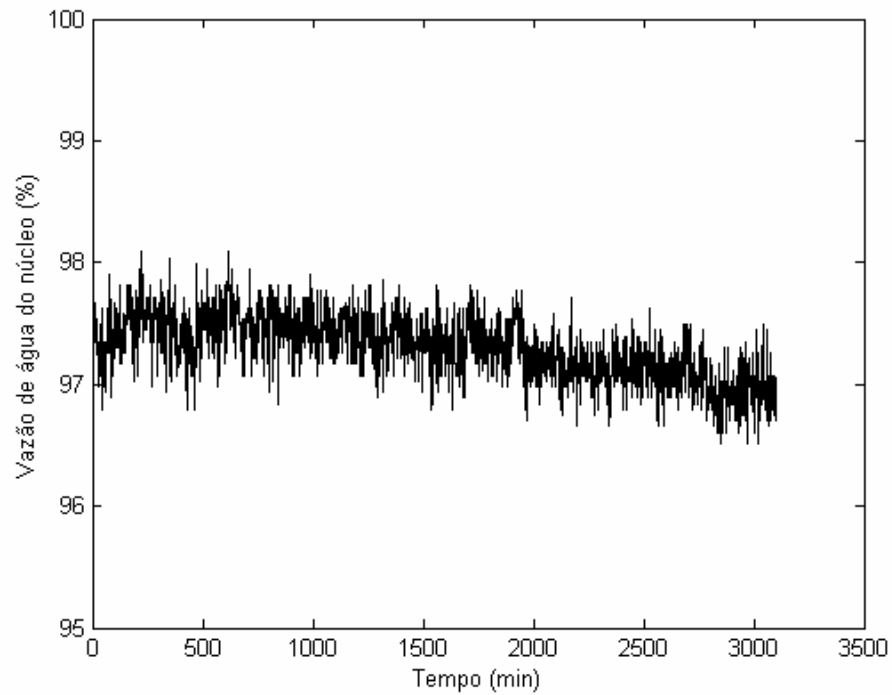


Figura 5.6 - Vazão da água do núcleo da usina Angra I.

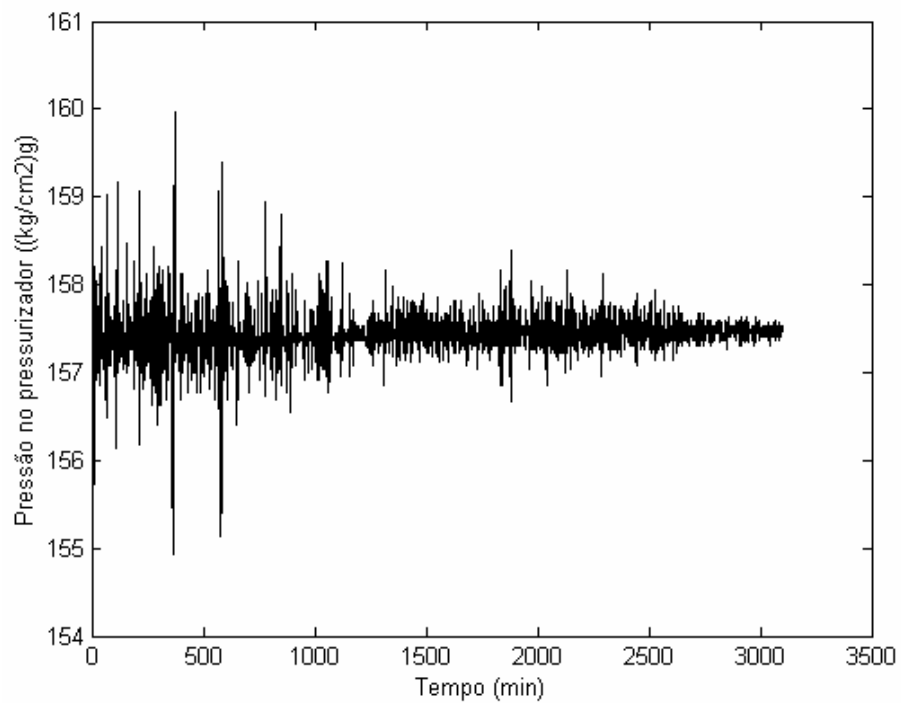


Figura 5.7 - Pressão no pressurizador da usina Angra I.

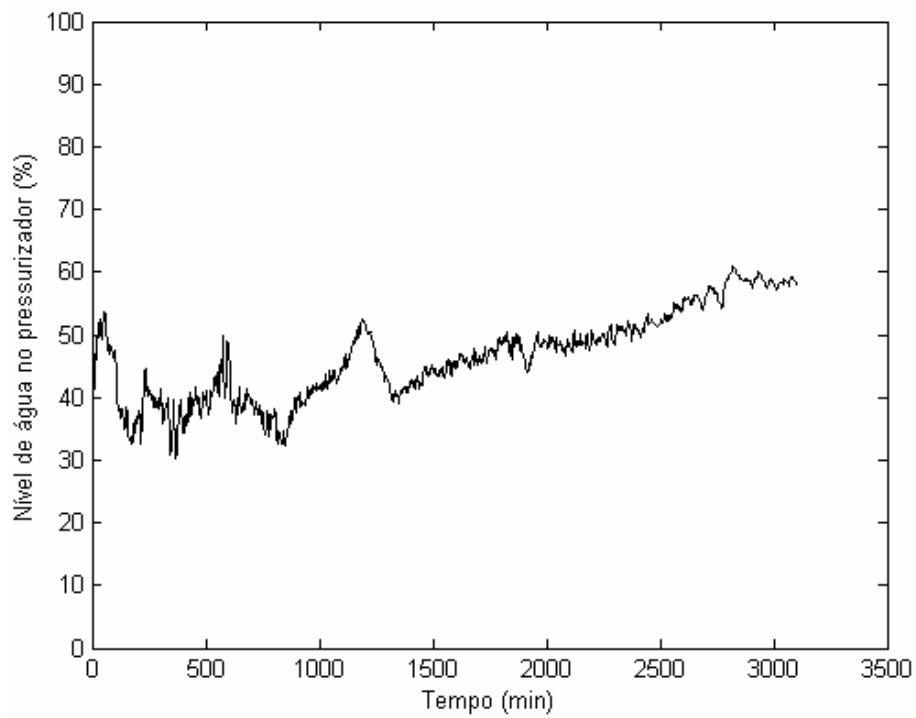


Figura 5.8 - Nível de água no pressurizador da usina Angra I.

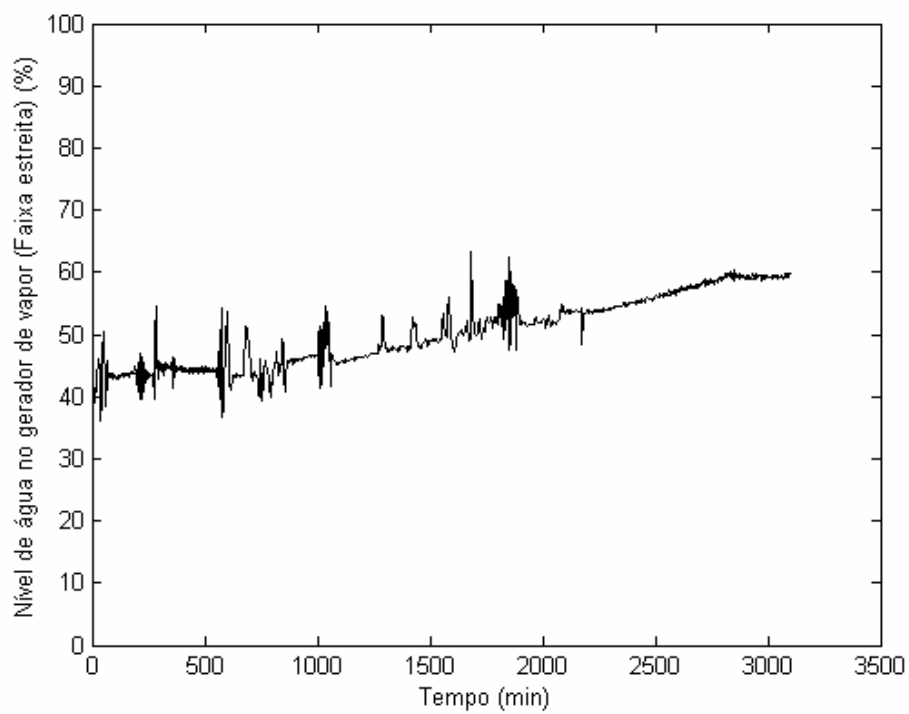


Figura 5.9 - Nível de água no GV (faixa estreita) do circuito 1 da usina Angra I.

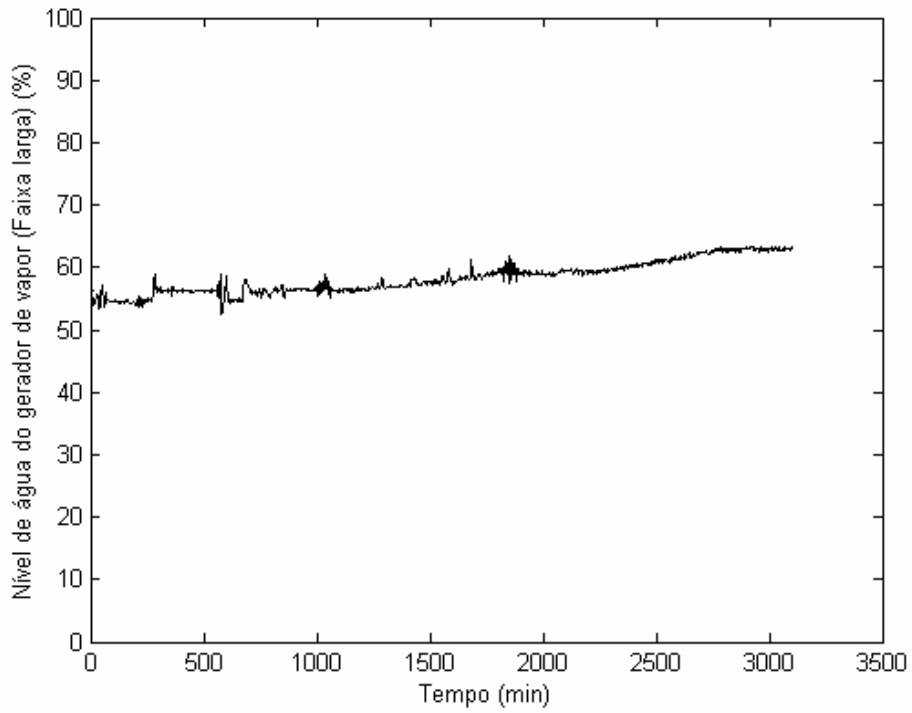


Figura 5.10 - Nível de água no GV (faixa larga) do circuito 1 da usina Angra I.

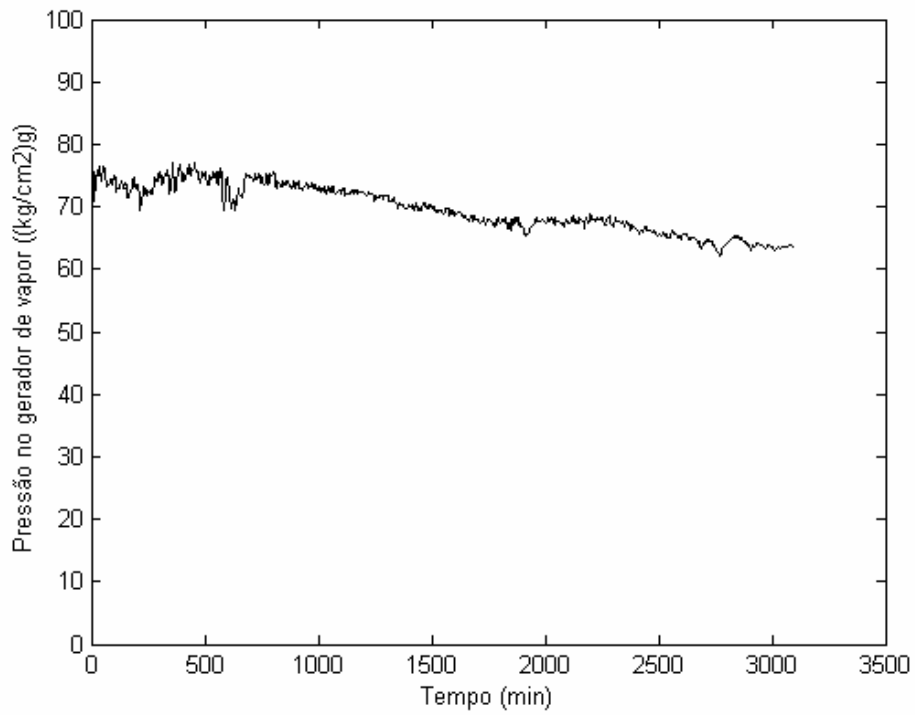


Figura 5.11 - Pressão no GV do circuito 1 da usina Angra I.

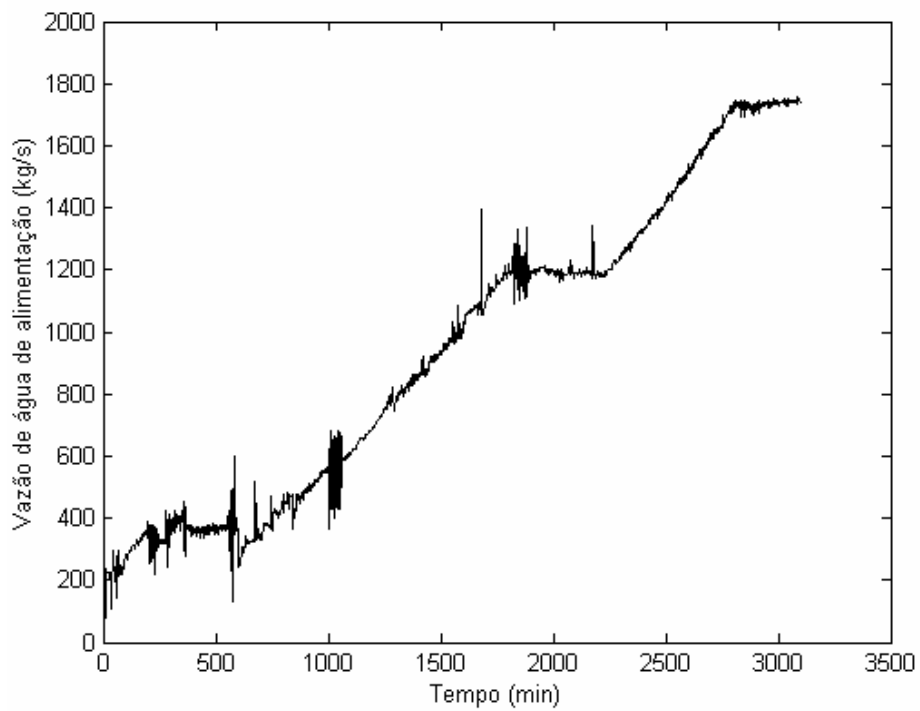


Figura 5.12 - Vazão de água de alimentação do circuito 1 da usina Angra I.

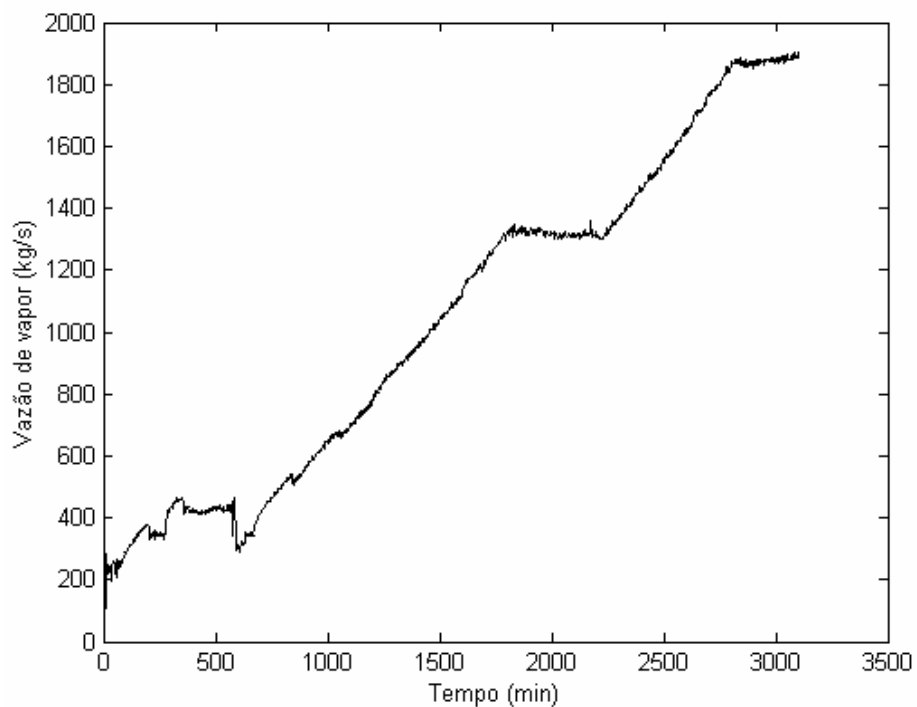


Figura 5.13 - Vazão de vapor do circuito 1 da usina Angra I.

5.2 O treinamento do modelo de validação de sinal

Antes de operar o sistema de validação de sinal faz-se necessário executar o seu treinamento. A seguir são apresentados os resultados obtidos nas diversas etapas de treinamento do sistema de validação de sinal.

A primeira etapa a ser realizada é a normalização das variáveis selecionadas para treinamento do sistema. A figura 5.14 apresenta os gráficos dos 12 sinais disponíveis da planta nuclear após o processo de normalização de cada variável no intervalo [0,1].

A seguir, deve-se efetuar a filtragem dos sinais disponíveis. A figura 5.15 apresenta as mesmas variáveis após o processo de filtragem por ondaletas. A forma e o tipo da ondaleta utilizada nesta aplicação são definidos no item 4.3.4 do capítulo 4.

Logo após, deve ser realizada a seleção das variáveis de entrada para o sistema de validação de sinal. A tabela 5.2 apresenta as variáveis de entrada selecionadas considerando cada uma das variáveis disponíveis da planta como saída do sistema de validação pelo método de seleção por GA utilizado. A otimização por GA foi feita considerando as constantes da equação 3.1 iguais a $\alpha_1=1$ e $\alpha_2=0,5$.

Como foi explicado no capítulo 3, a seleção das variáveis é feita quando o erro de estimação do modelo ε_{mod} é menor do que o critério de erro ε_{crit} .

Vamos agora realizar o treinamento de um sistema de validação de sinal para monitorar a variável potência nuclear (u1) e outro para monitorar a variável temperatura da água na perna quente (u3).

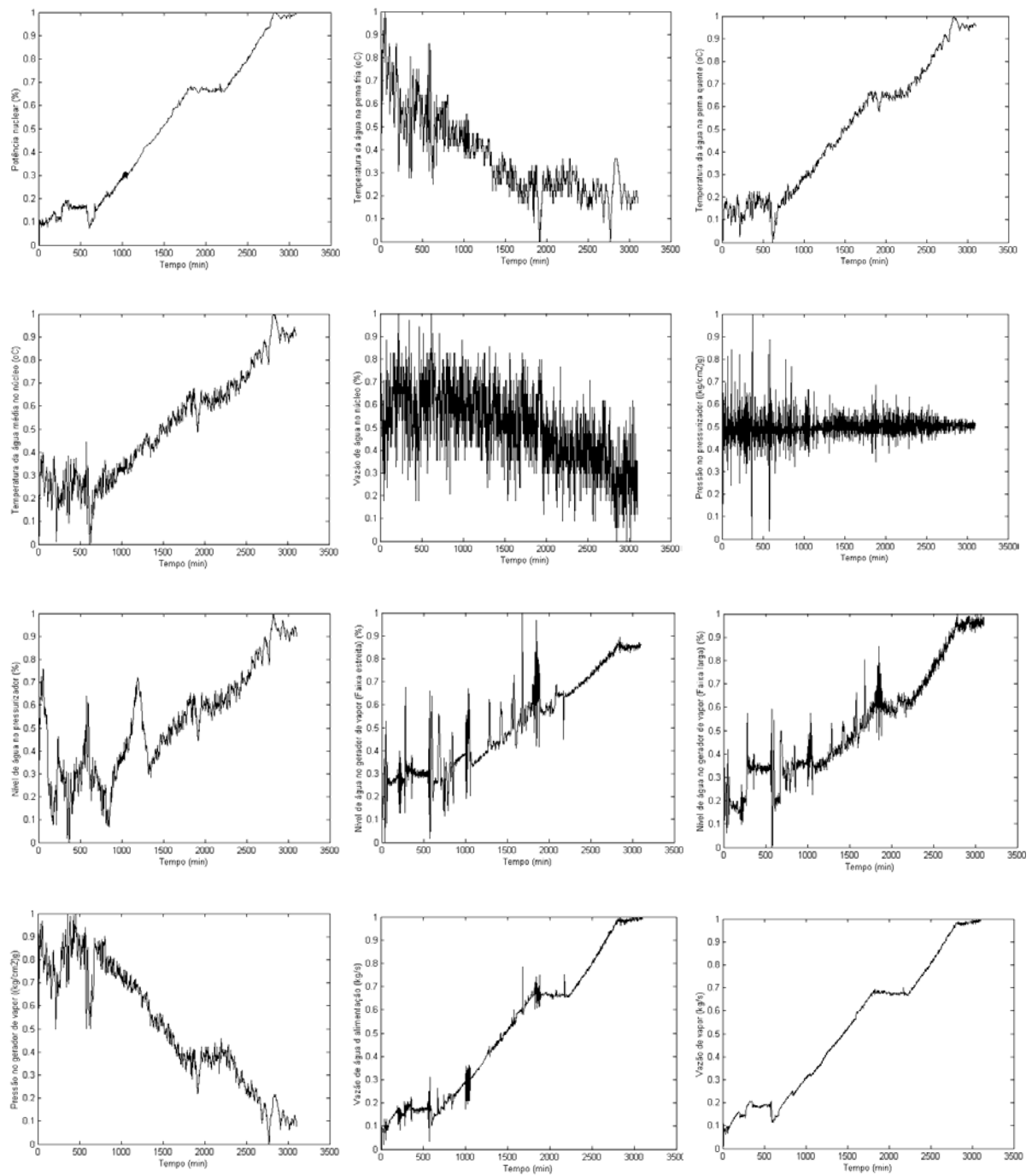


Figura 5.14 - Sinais disponíveis da planta nuclear após o processo de normalização.

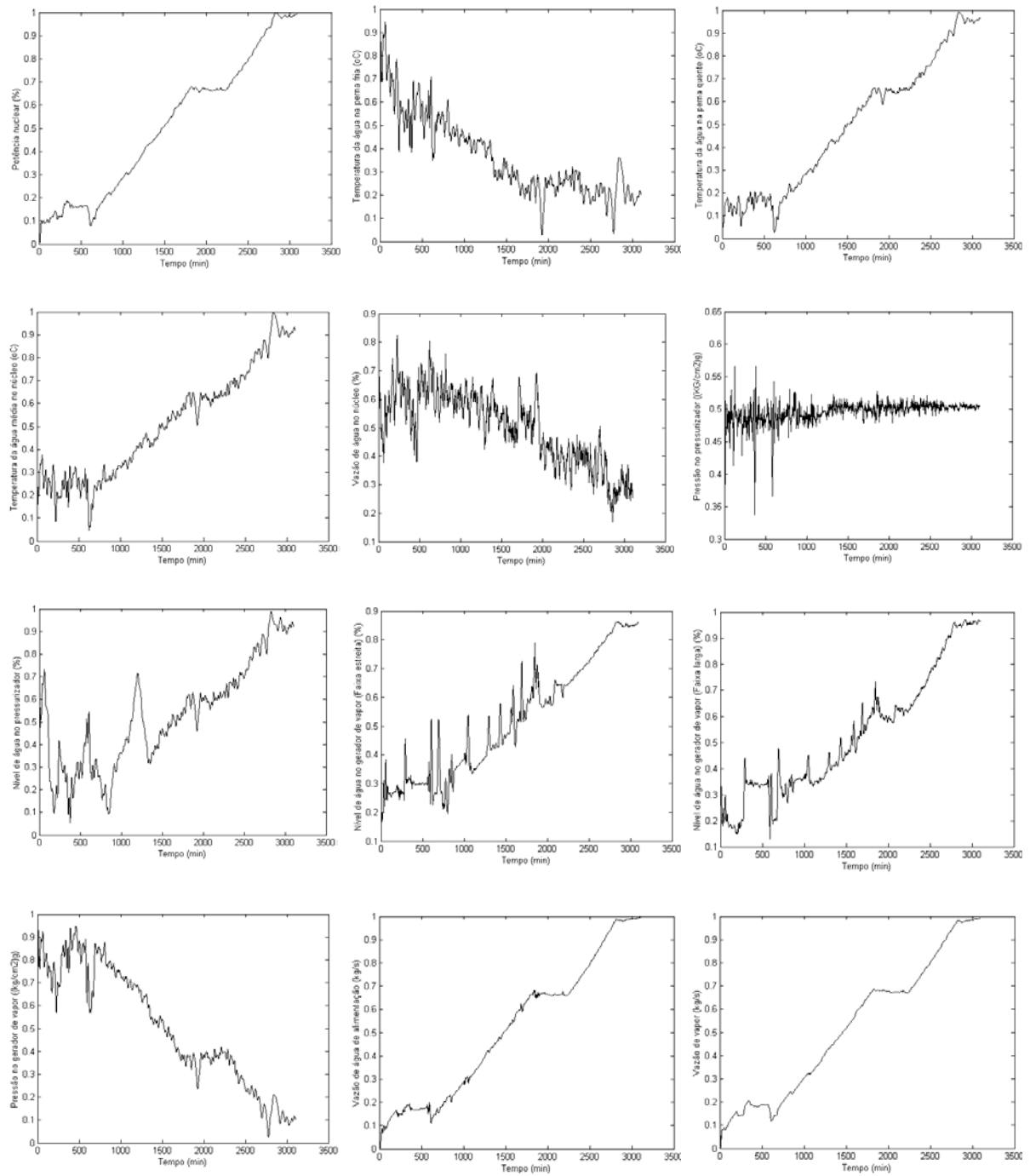


Figura 5.15 - Sinais disponíveis da planta nuclear após a filtragem por ondaletas.

Tabela 5.2

Seleção das variáveis de entrada considerando $\alpha_1=1$ e $\alpha_2=0,5$.

Variável de saída	Critério de erro (ϵ_{crit})	Erro de estimação (ϵ_{mod})	Variáveis de entrada selecionadas
u1	0,013776	0,002989	u2 u4 u5
u2	0,036972	0,033307	u6 u10 u12
u3	0,017476	0,002844	u2 u5 u7 u12
u4	0,020337	0,003350	u2 u3 u5 u7
u5	0,069172	0,066196	u3 u4 u6 u7 u9
u6	0,058825	0,058372	u2 u5 u10
u7	0,087842	0,061977	u4 u5 u9 u10
u8	0,053709	0,032946	u2 u5 u9
u9	0,042564	0,029575	u2 u5 u7 u8
u10	0,032777	0,014956	u2 u7 u8 u11 u12
u11	0,012728	0,005171	u2 u4 u5 u10
u12	0,012944	0,004495	u2 u4 u5

5.2.1 Treinando o sistema para monitorar a potência nuclear da planta

Conforme apresentado na tabela 5.2 para monitorar a variável u1 (potência nuclear) deve-se utilizar as seguintes variáveis de entrada:

u2 - temperatura da água na perna fria;

u4 - temperatura média da água do núcleo;

u5 - vazão de água do núcleo.

Após o treinamento do sistema de validação de sinal para monitorar a potência nuclear a partir das três variáveis de entrada u2, u4 e u5, obtemos o seguinte conjunto de parâmetros, apresentados na tabela 5.3, que serão utilizados na etapa de operação do sistema. Além dos parâmetros listados na tabela 5.3 são obtidos no treinamento outros parâmetros, tais como: o intervalo (*span*) das variáveis de entrada e da variável de saída do sistema, as pertinências das amostras de treinamento nas classes, a distância de Manhattan média \bar{d} das amostras nas classes para cada posição da janela para o método de auto-correlação dos resíduos etc.

Tabela 5.3

Principais parâmetros obtidos no treinamento do sistema para monitorar a potência nuclear.

	Centróide	Matriz de transformação do espaço	Distância média quadrática das amostras	Método ERRO	Método SPRT	Método AUTO	Janela do método AUTO
Classe 1	u2=0,5808 u4= 0,2478 u5= 0,5980	$A_1 = \begin{bmatrix} 0,4728 & -0,1147 & 0,4279 \\ -0,1147 & 2,9416 & 1,5298 \\ 0,4279 & 1,5298 & 2,0290 \end{bmatrix}$	$\eta_1 = 0,0118$	$\mu_\epsilon = 0,0000$ $\sigma_\epsilon = 0,0114$	$\sigma_0 = 0,0114$ $\sigma_1 = 0,0343$ $\mu_0 = 0,0000$ $\mu_1 = 0,0067$	$\mu_d = 6,0864$ $\sigma_d = 1,9290$	48
Classe 2	u2=0,3134 u4=0,5171 u5=0,4960	$A_2 = \begin{bmatrix} 1,8159 & 1,4853 & 0,6190 \\ 1,4853 & 2,6215 & 2,1101 \\ 0,6190 & 2,1101 & 2,4313 \end{bmatrix}$	$\eta_2 = 0,0103$	$\mu_\epsilon = 0,0000$ $\sigma_\epsilon = 0,0070$	$\sigma_0 = 0,0070$ $\sigma_1 = 0,0211$ $\mu_0 = 0,0000$ $\mu_1 = 0,0185$	$\mu_d = 5,2017$ $\sigma_d = 2,7243$	48
Classe 3	u2=0,2161 u4=0,7904 u5=0,3864	$A_3 = \begin{bmatrix} 1,4824 & 0,3285 & 0,5869 \\ 0,3285 & 2,1630 & 2,4407 \\ 0,5869 & 2,4407 & 3,1094 \end{bmatrix}$	$\eta_3 = 0,0127$	$\mu_\epsilon = 0,0000$ $\sigma_\epsilon = 0,0052$	$\sigma_0 = 0,0052$ $\sigma_1 = 0,0157$ $\mu_0 = 0,0000$ $\mu_1 = 0,0095$	$\mu_d = 6,2207$ $\sigma_d = 3,1723$	48

A figura 5.16 apresenta a posição das três classes obtidas pelo treinamento para as variáveis de entrada, onde são representados os centróides das classes por (x) e as distâncias médias quadrática η pelas elipses.

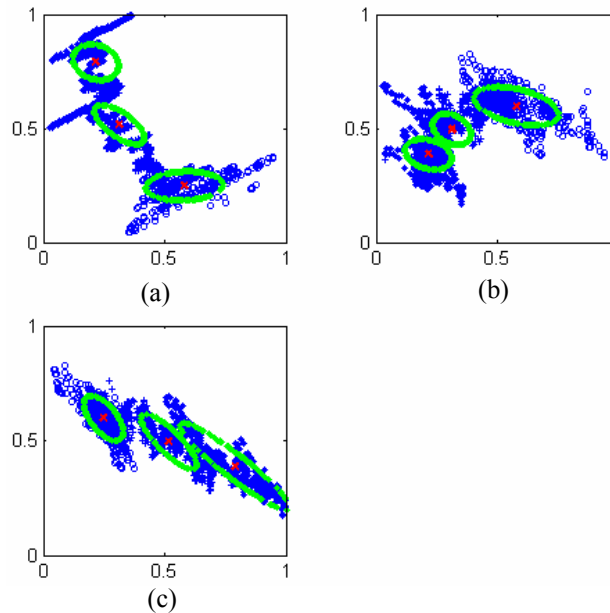


Figura 5.16 - Posição das três classes obtidas pelo treinamento: (a) u2 versus u4; (b) u2 versus u5; (c) u4 versus u5.

A figura 5.17 apresenta os erros de estimação dos três ANFIS para as amostras das classes correspondentes.

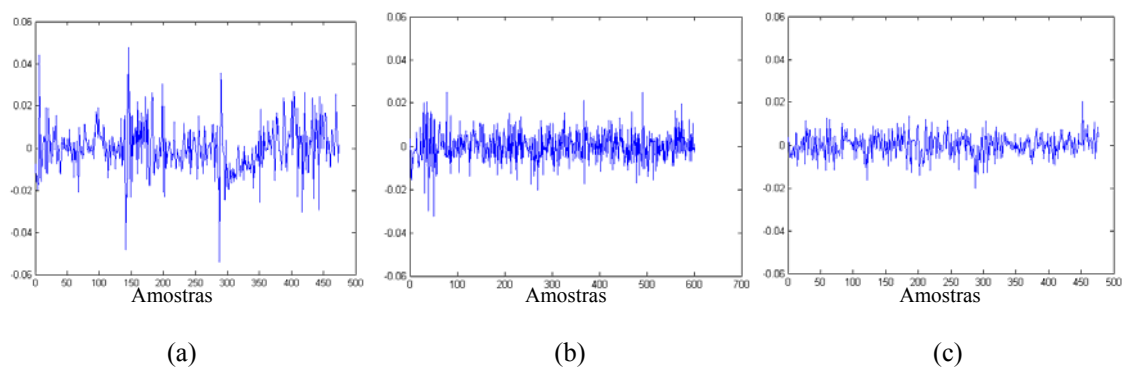


Figura 5.17 - Erros de estimação dos três ANFIS para as amostras: (a) Classe 1; (b) Classe 2; (c) Classe 3.

A figura 5.18 apresenta a arquitetura dos três ANFIS inicial e após o treinamento.

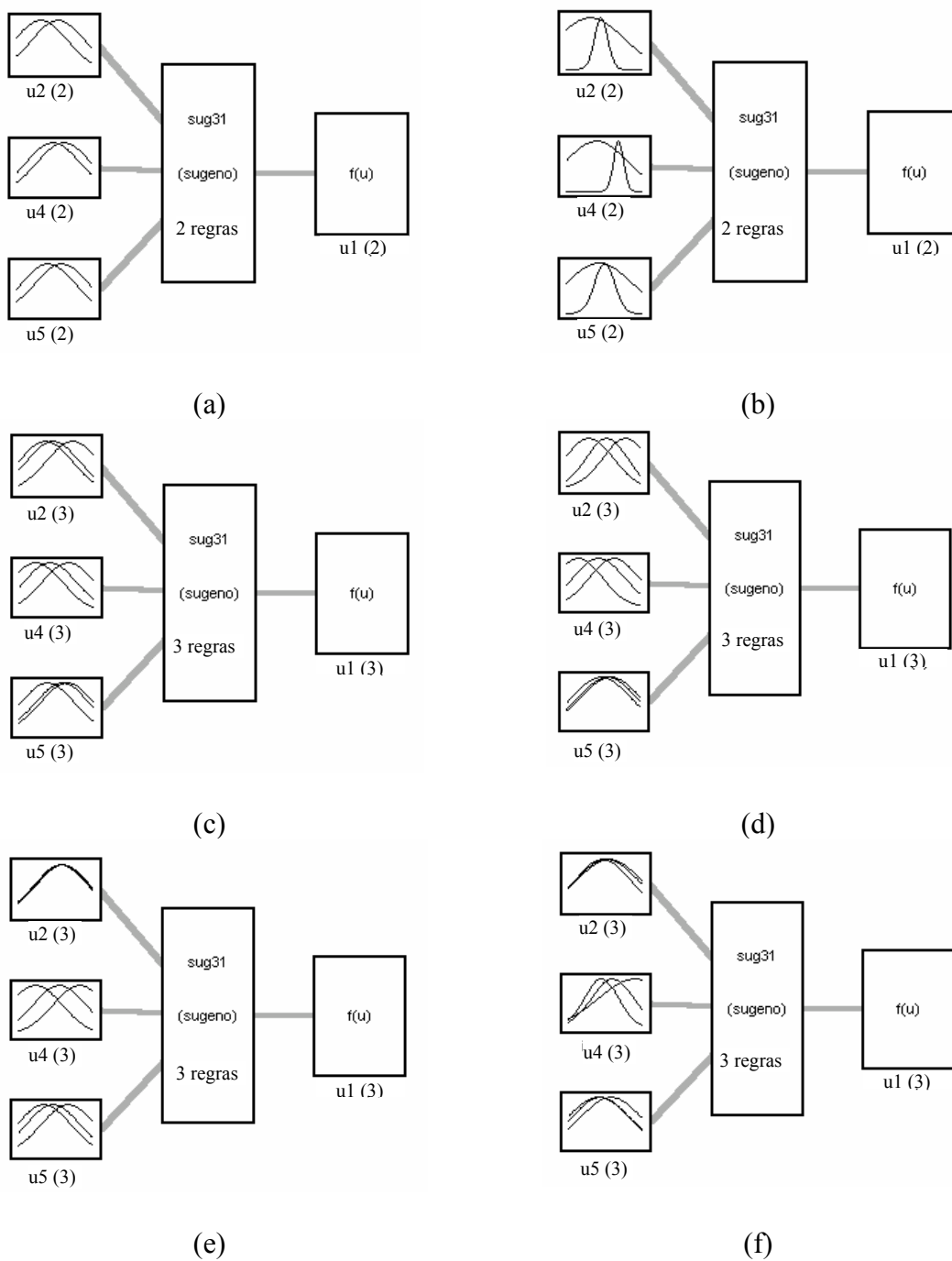


Figura 5.18 - Estruturas iniciais e após o treinamento dos ANFIS: (a) Inicial e (b) final da classe 1; (c) Inicial e (d) final da classe 2; (e) Inicial e (f) final da classe 3.

5.2.2 Treinando o sistema para monitorar a temperatura da água na perna quente

Conforme apresentado na tabela 5.2 para monitorar a variável u_3 - temperatura da água na perna quente deve-se utilizar as seguintes variáveis de entrada:

u_2 - temperatura da água na perna fria;

u_5 - vazão de água do núcleo;

u_7 - nível de água no pressurizador;

u_{12} - vazão de vapor.

Após o treinamento do sistema de validação de sinal para monitorar a temperatura da água na perna quente a partir das quatro variáveis de entrada u_2 , u_5 , u_7 e u_{12} , obtemos o seguinte conjunto de parâmetros, apresentados na tabela 5.4, que serão utilizados na etapa de operação do sistema. Além dos parâmetros listados na tabela 5.4 são obtidos no treinamento outros parâmetros, tais como: o intervalo (*span*) das variáveis de entrada e da variável de saída do sistema, as pertinências das amostras de treinamento nas classes, a distância de Manhattan média \bar{d} das amostras nas classes para cada posição da janela para o método de auto-correlação dos resíduos etc.

A figura 5.19 apresenta a posição das três classes obtidas pelo treinamento para as variáveis de entrada, onde são representados os centroídes das classes por (x) e as distâncias médias quadráticas, η pelas elipses.

A figura 5.20 apresenta os erros de estimação dos três ANFIS para as amostras das classes correspondentes.

A figura 5.21 apresenta a arquitetura dos três ANFIS inicial e após o treinamento.

Tabela 5.4

Principais parâmetros obtidos no treinamento do sistema para monitorar a temperatura da água na perna quente.

	Centróide	Matriz de transformação do espaço	Distância média quadrática das amostras	Método ERRO	Método SPRT	Método AUTO	Janela do método AUTO
Classe 1	u2= 0,2464 u5= 0,4129 u7= 0,6629 u12= 0,7381	$A_1 = \begin{bmatrix} 4,9414 & 0,0745 & -7,9004 & 9,5958 \\ 0,0745 & 0,8654 & 1,1692 & -0,7954 \\ -7,9004 & 1,1692 & 17,035 & -19,488 \\ 9,5958 & -0,7954 & -19,488 & 22,784 \end{bmatrix}$	$\eta_1 = 0,0078$	$\mu_\varepsilon = 0,0000$ $\sigma_\varepsilon = 0,0016$	$\sigma_0 = 0,0016$ $\sigma_1 = 0,0047$ $\mu_0 = 0,0000$ $\mu_1 = 0,0136$	$\mu_d = 3,4083$ $\sigma_d = 1,3341$	25
Classe 2	u2= 0,5862 u5= 0,5926 u7= 0,3080 u12= 0,1871	$A_2 = \begin{bmatrix} 1,4434 & 1,2583 & -0,1722 & 1,7121 \\ 1,2583 & 1,8805 & 0,1107 & 1,8973 \\ -0,1722 & 0,1107 & 0,5328 & 1,0106 \\ 1,7121 & 1,8973 & 1,0106 & 7,0609 \end{bmatrix}$	$\eta_2 = 0,0153$	$\mu_\varepsilon = 0,0000$ $\sigma_\varepsilon = 0,0024$	$\sigma_0 = 0,0024$ $\sigma_1 = 0,0073$ $\mu_0 = 0,0000$ $\mu_1 = 0,0051$	$\mu_d = 3,3288$ $\sigma_d = 1,1303$	25
Classe 3	u2= 0,4293 u5= 0,5867 u7= 0,4373 u12= 0,3532	$A_3 = \begin{bmatrix} 4,4193 & 0,9620 & -0,5191 & 3,3695 \\ 0,9620 & 1,8157 & -0,1763 & 1,2795 \\ -0,5191 & -0,1763 & 0,2807 & -0,6384 \\ 3,3695 & 1,2795 & -0,6384 & 3,6280 \end{bmatrix}$	$\eta_3 = 0,0113$	$\mu_\varepsilon = 0,0000$ $\sigma_\varepsilon = 0,0015$	$\sigma_0 = 0,0015$ $\sigma_1 = 0,0044$ $\mu_0 = 0,0000$ $\mu_1 = 0,0063$	$\mu_d = 2,9416$ $\sigma_d = 1,5449$	25

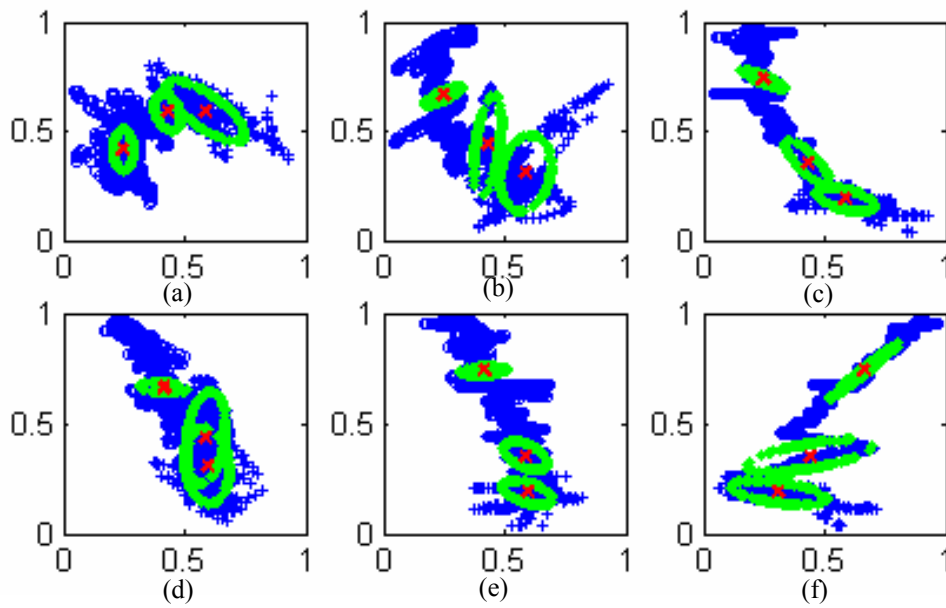


Figura 5.19 - Posição das três classes obtidas pelo treinamento: (a) u_2 versus u_5 ; (b) u_2 versus u_7 ; (c) u_2 versus u_{12} ; (d) u_5 versus u_7 ; (e) u_5 versus u_{12} ; e (f) u_7 versus u_{12} .

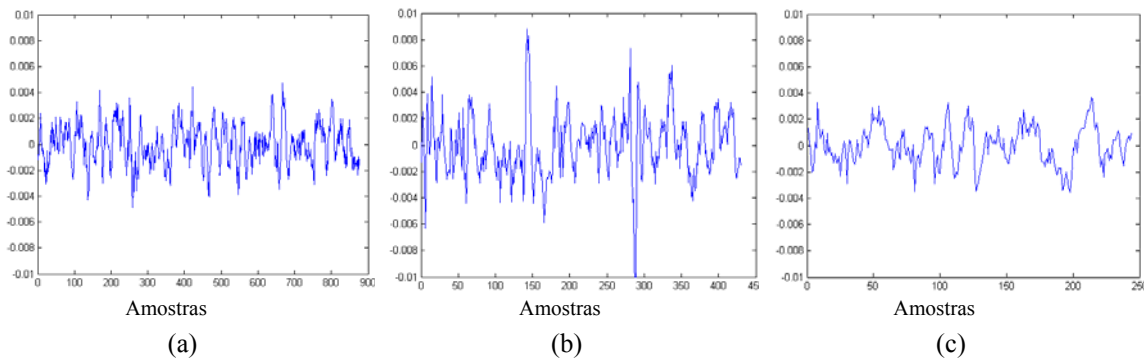


Figura 5.20 - Erros de estimação dos três ANFIS para as amostras: (a) Classe 1; (b) Classe 2; (c) Classe 3.

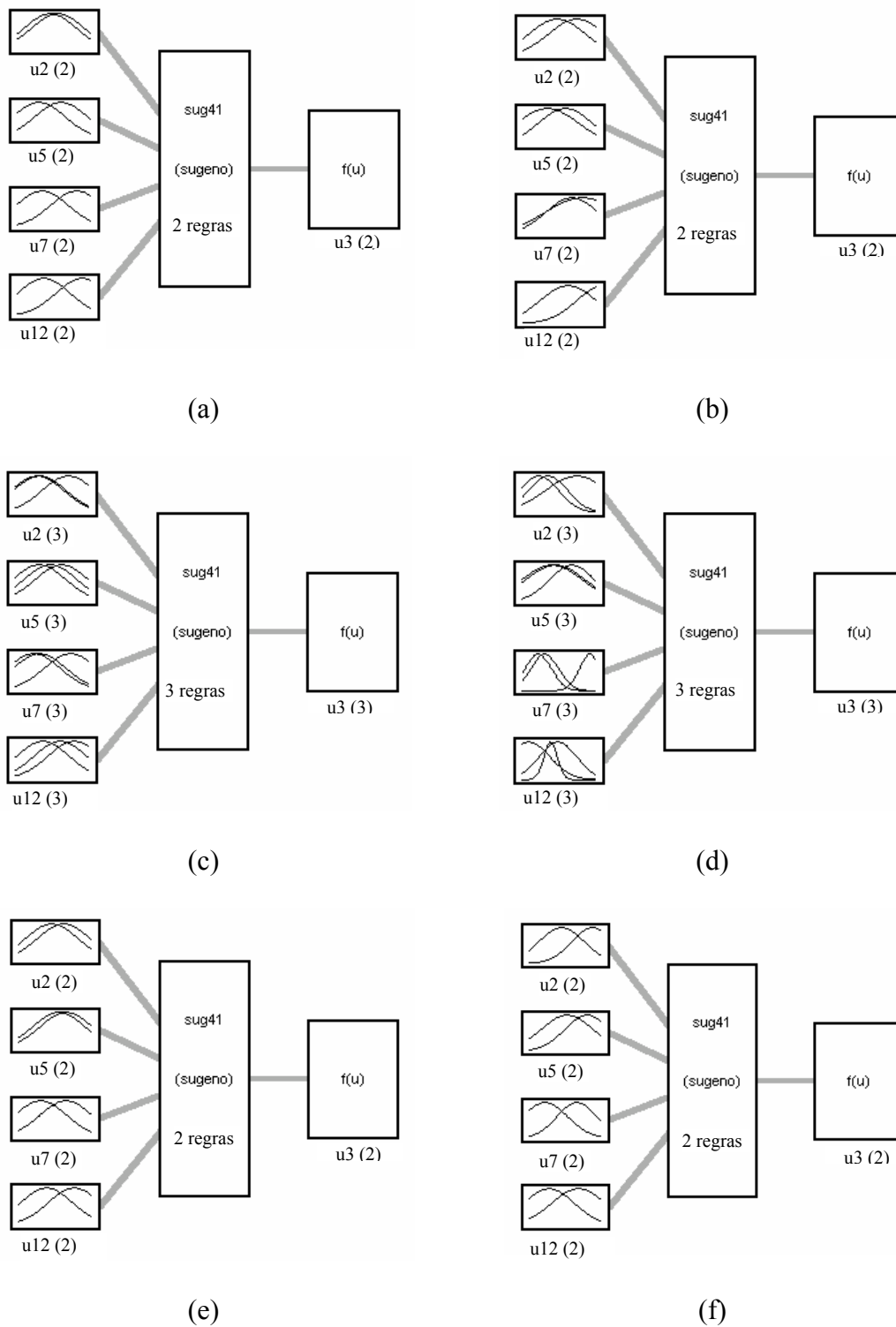


Figura 5.21 - Estruturas iniciais e após o treinamento dos ANFIS: (a) Inicial e (b) final da classe 1; (c) Inicial e (d) final da classe 2; (e) Inicial e (f) final da classe 3.

5.3 A operação do modelo de validação de sinal

A seguir são descritos e apresentados os resultados obtidos nos ensaios realizados nos dois sistemas de validação de sinal treinados para validar os sinais que indicam a potência nuclear e a temperatura da água na perna quente.

5.3.1 Monitoração da potência nuclear do reator

5.3.1.1 Reator operando sem falha no sinal monitorado - Ensaio 1

Neste ensaio o sistema de validação de sinal recebeu como entrada as mesmas amostras utilizadas no seu treinamento. As figuras 5.22, 5.23 e 5.24 apresentam os resultados obtidos no ensaio.

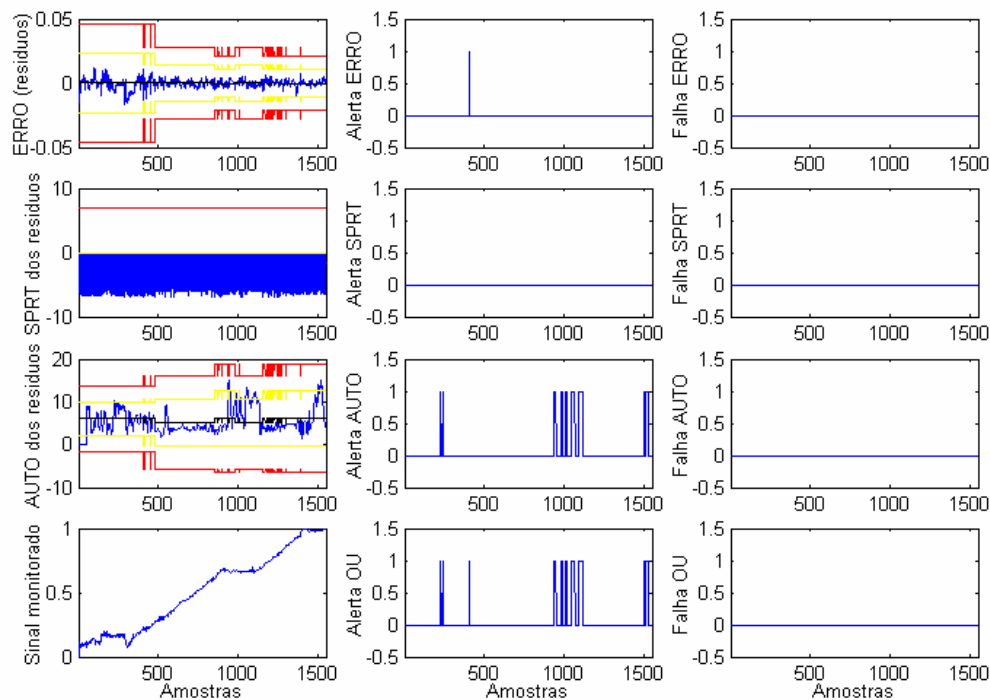


Figura 5.22 - Resultados obtidos pelo sistema de validação de sinal para o sinal monitorado sem falhas.

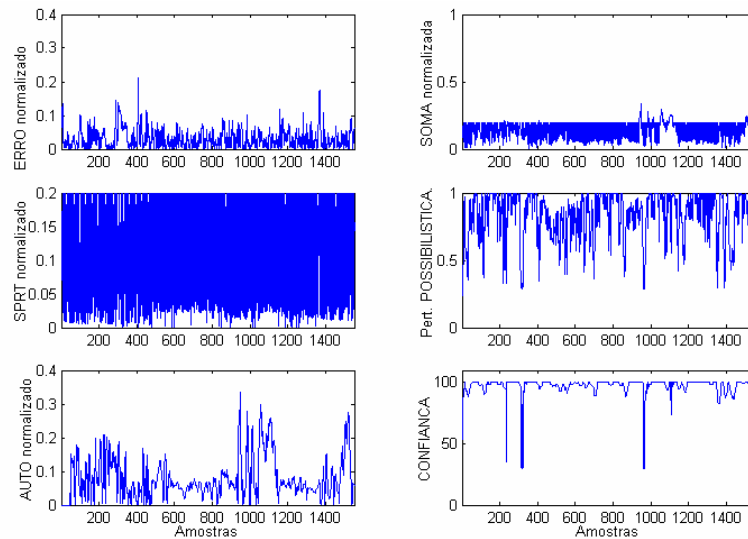


Figura 5.23 - Confiança da resposta obtida pelo sistema de validação de sinal para o sinal da potência nuclear sem falhas.

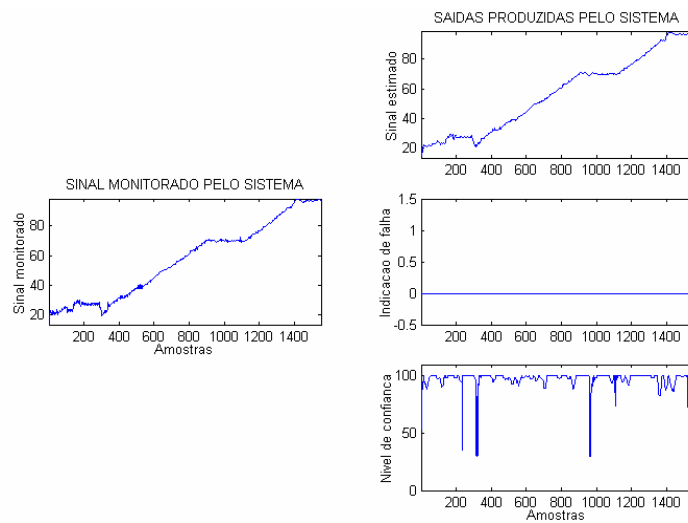


Figura 5.24 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzidos pelo sistema.

5.3.1.2 Falhas no sinal monitorado - Operação em estado estacionário

Nestes ensaios a planta nuclear foi posta a operar com potência nuclear em torno de 70% e foi inserida uma falha no sinal monitorado a partir da amostra 500.

5.3.1.2.1 Falha do tipo polarização positiva - Ensaio 2

As figuras 5.25 a 5.27 apresentam os resultados obtidos pelo sistema de validação de sinal neste ensaio.

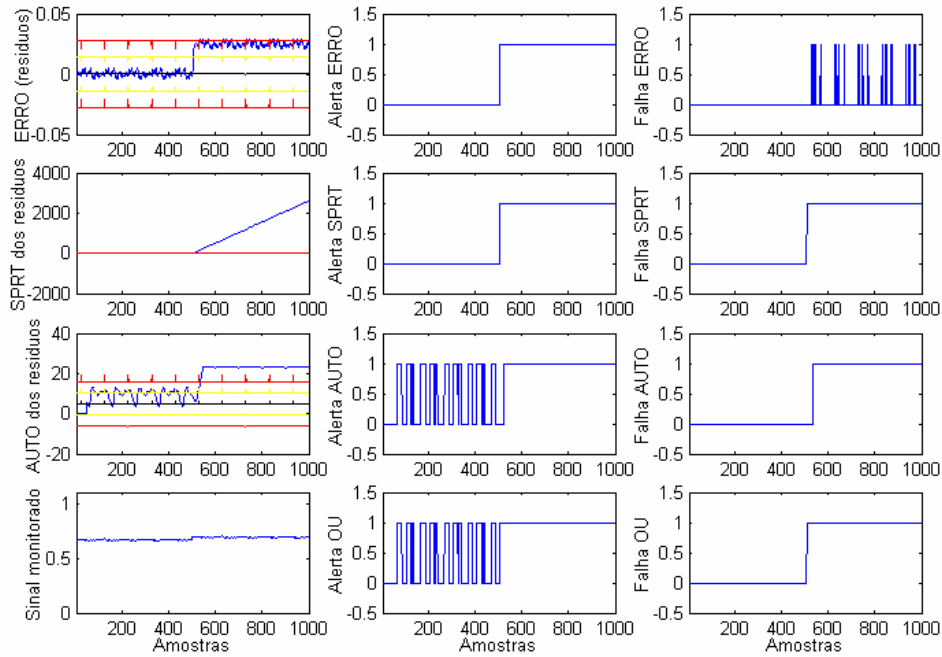


Figura 5.25 - Resultados obtidos para o sinal monitorado com falha do tipo polarização.

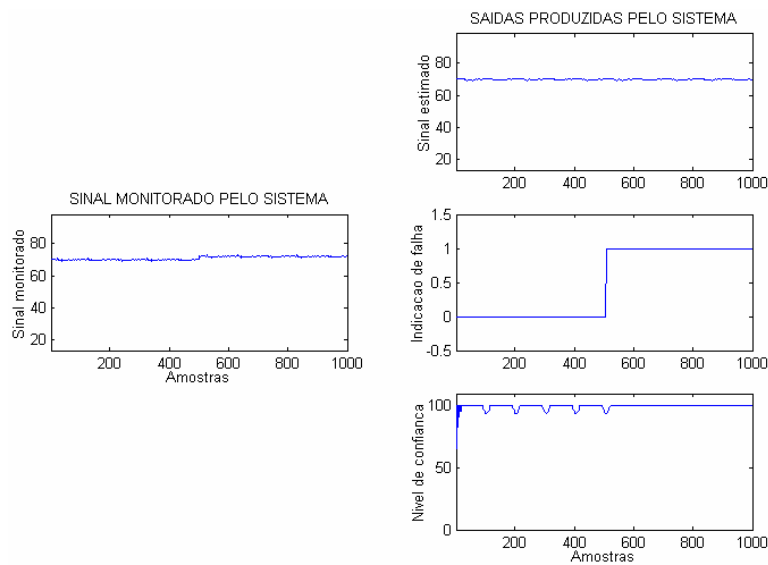


Figura 5.26 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzidos pelo sistema.

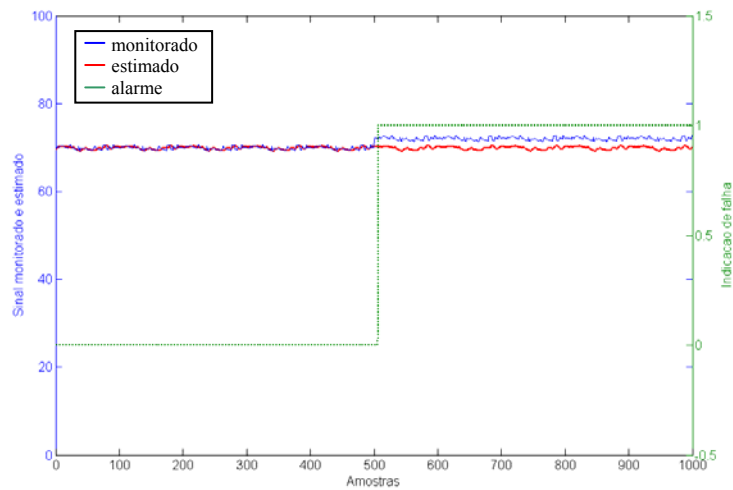


Figura 5.27 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada e de alarme produzidos pelo sistema.

5.3.1.2.2 Falha do tipo desvio positivo - Ensaio 3

As figuras 5.28 a 5.30 apresentam os resultados obtidos pelo sistema de validação de sinal neste ensaio.

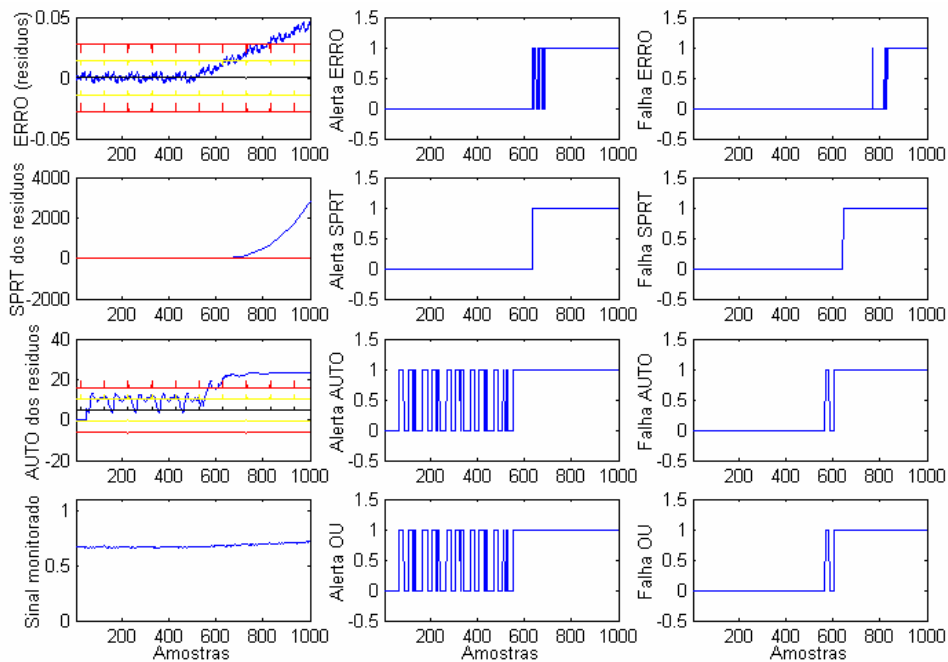


Figura 5.28 - Resultados obtidos para o sinal monitorado com falha do tipo desvio.

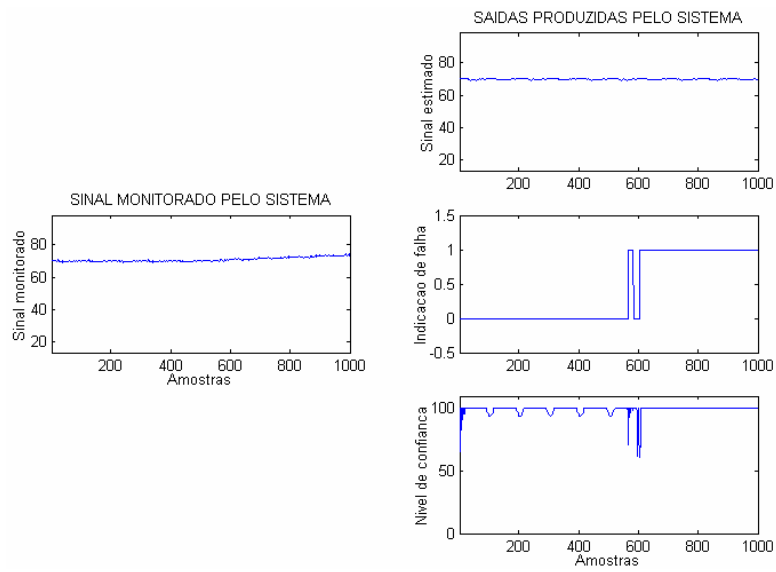


Figura 5.29 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzidos pelo sistema.

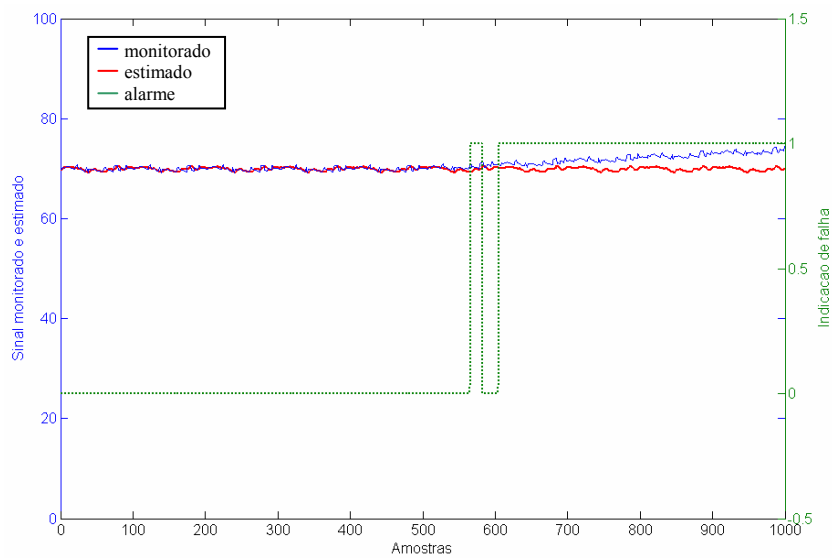


Figura 5.30 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada e de alarme produzidos pelo sistema.

5.3.1.2.3 Falha do tipo colado - Ensaio 4

As figuras 5.31 a 5.33 apresentam os resultados obtidos pelo sistema de validação de sinal neste ensaio.

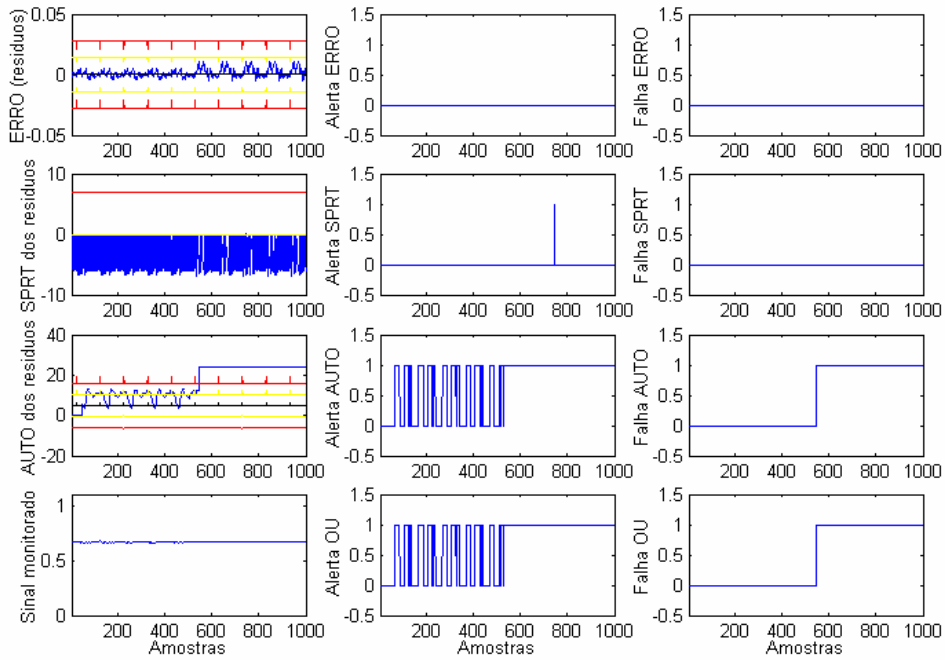


Figura 5.31 - Resultados obtidos para o sinal monitorado com falha do tipo colado.

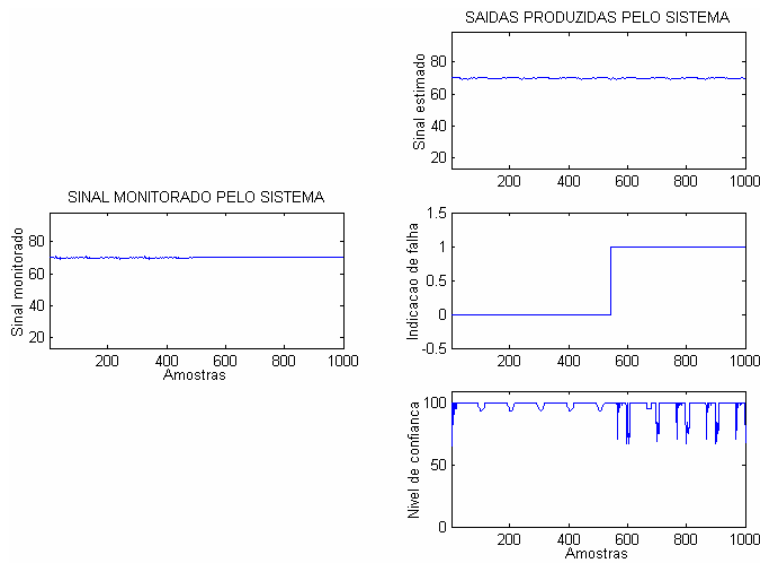


Figura 5.32 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzidos pelo sistema.

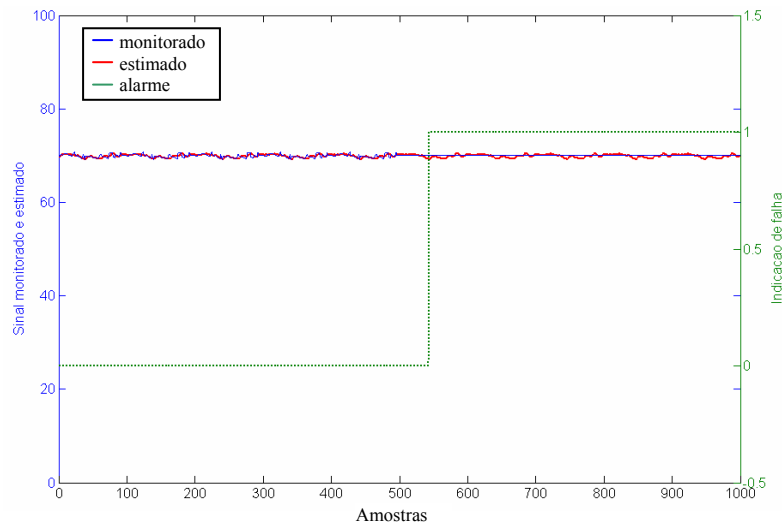


Figura 5.33 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada e de alarme produzidos pelo sistema.

5.3.1.2.4 Resumo da detecção de falhas pelo sistema - Operação em estado estacionário

Outros tipos de falhas foram inseridas no sinal monitorado pelo sistema conforme mostrado na tabela 5.5. Nesta tabela são apresentadas a quantidade de amostras adquiridas pelo sistema de validação de sinal a partir da inserção de uma determinada falha para a produção do sinal de alarme. Como o sistema utiliza uma lógica OU entre os sinais de falha produzidos por cada método de detecção, o número de amostras necessárias para detecção pelo sistema é o menor valor produzido dentre os três métodos. Se considerarmos o tempo entre as amostras adquiridas pelo sistema igual a 1 segundo, os valores apresentados na tabela seriam os tempos em segundos necessários para detecção dos alarmes a partir de seu início.

Tabela 5.5

Quantidade de amostras necessária a partir da introdução da falha para detecção da falha pelo sistema - Operação em estado estacionário.

Falha introduzida	ERRO	SPRT	AUTO	SOMA(OU)
Mudança brusca positiva (<i>positive hard-over</i>)	27	21	40	21
Mudança brusca negativa (<i>negative hard-over</i>)	27	23	41	23
Polarização positiva (<i>positive bias</i>)	27	7	34	7
Polarização negativa (<i>negative bias</i>)	27	8	31	8
Pico positivo (<i>positive spike</i>)	0	1	---	0
Pico negativo (<i>negative spike</i>)	0	1	---	0
Colado (<i>stuck</i>)	---	---	43	43
Errático (<i>erratic</i>)	21	21	---	21
Cíclico (<i>cyclic</i>)	---	---	76	76
Desvio positivo (<i>positive drift</i>)	269	142	66	66
Desvio negativo (<i>negative drift</i>)	286	188	93	93
Não linear (<i>nonlinear</i>)	128	112	73	73

--- Não detectou a falha.

5.3.1.3 Falhas no sinal monitorado - Operação em subida de potência

Nestes ensaios a planta nuclear está subindo sua potência para plena potência e uma falha é inserida quando a potência está em torno de 40%, ou seja, a partir da amostra 500.

5.3.1.3.1 Falha do tipo polarização negativa - Ensaio 5

Neste ensaio o sinal da potência nuclear está sujeito a uma polarização negativa. As figuras 5.34 a 5.36 apresentam os resultados obtidos pelo sistema de validação de sinal neste ensaio.

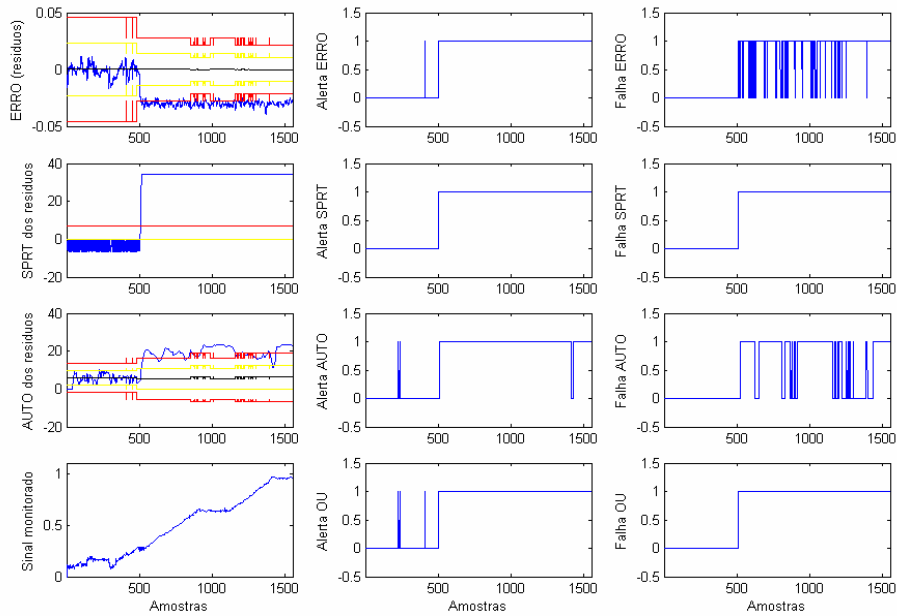


Figura 5.34 - Resultados obtidos para o sinal monitorado com falha do tipo polarização.

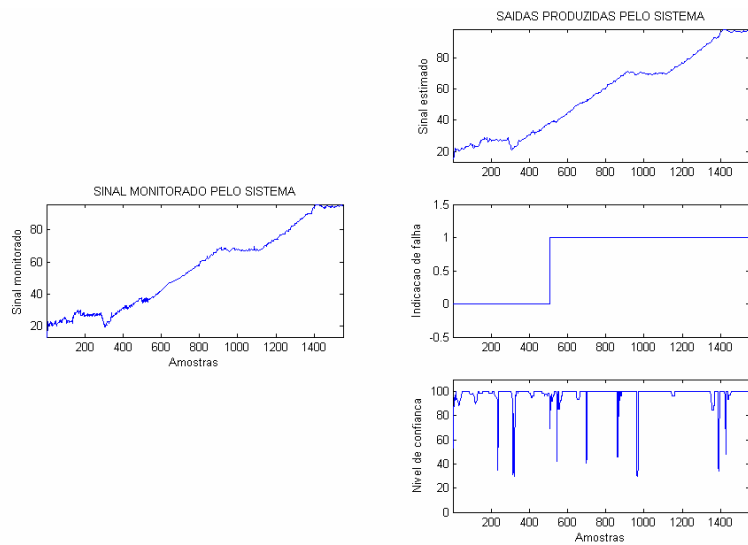


Figura 5.35 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzidos pelo sistema.

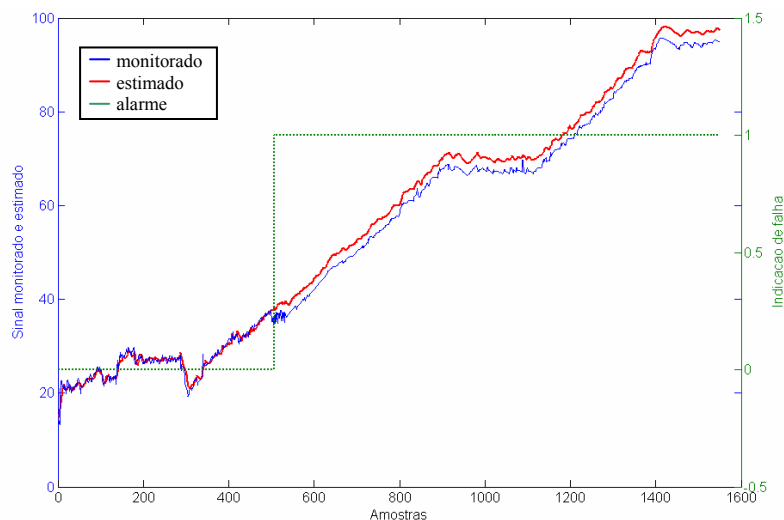


Figura 5.36 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada e de alarme produzidos pelo sistema.

5.3.1.3.2 Falha do tipo desvio negativo - Ensaio 6

Neste ensaio o sinal da potência nuclear está sujeito a um desvio negativo. As figuras 5.37 a 5.39 apresentam os resultados obtidos pelo sistema de validação de sinal este ensaio.

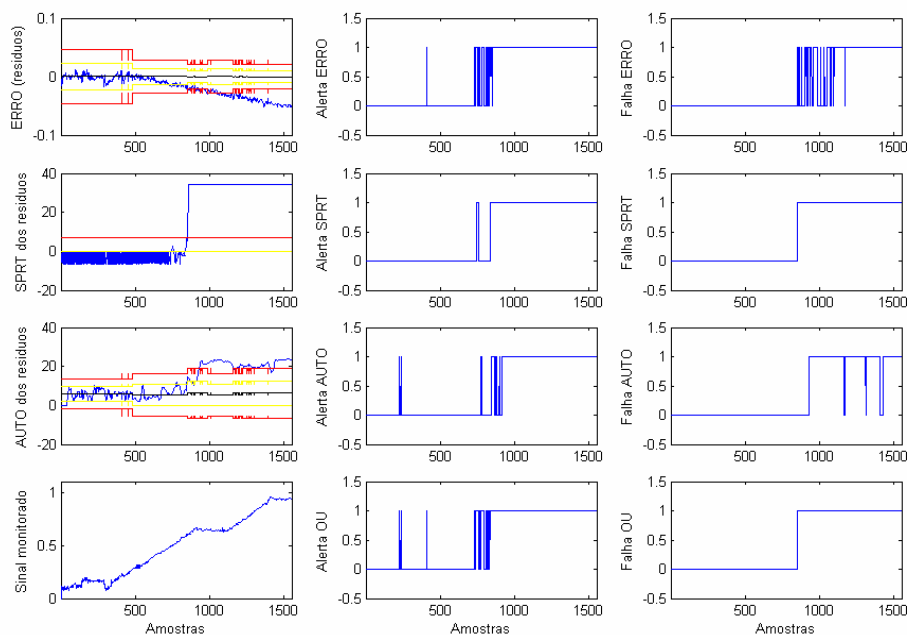


Figura 5.37 - Resultados obtidos para o sinal monitorado com falha do tipo desvio.

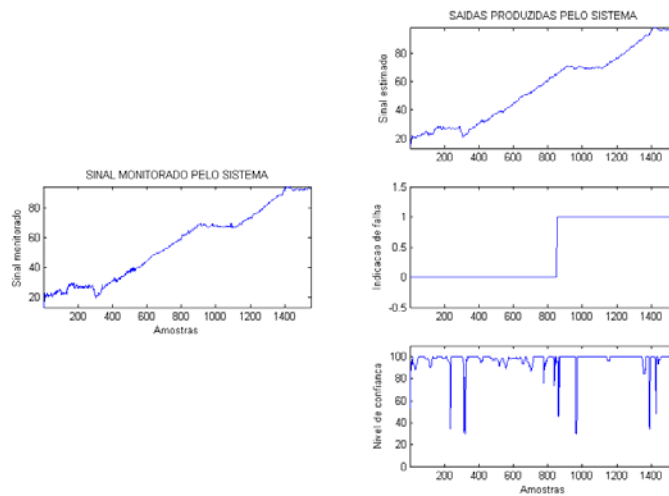


Figura 5.38 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzidos pelo sistema.

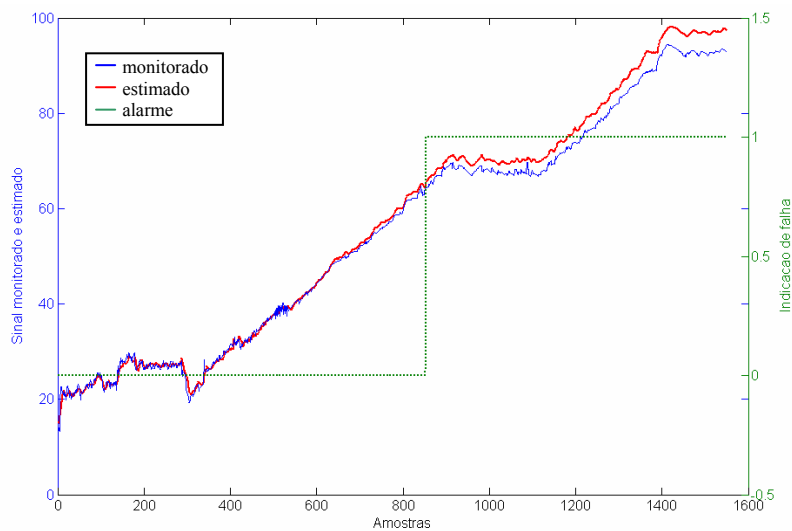


Figura 5.39 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada e de alarme produzidos pelo sistema.

5.3.1.3.3 Falha do tipo colado - Ensaio 7

Neste ensaio o sinal da potência nuclear está sujeito a uma falha do tipo colado. As figuras 5.40 a 5.42 apresentam os resultados obtidos pelo sistema de validação de sinal neste ensaio.

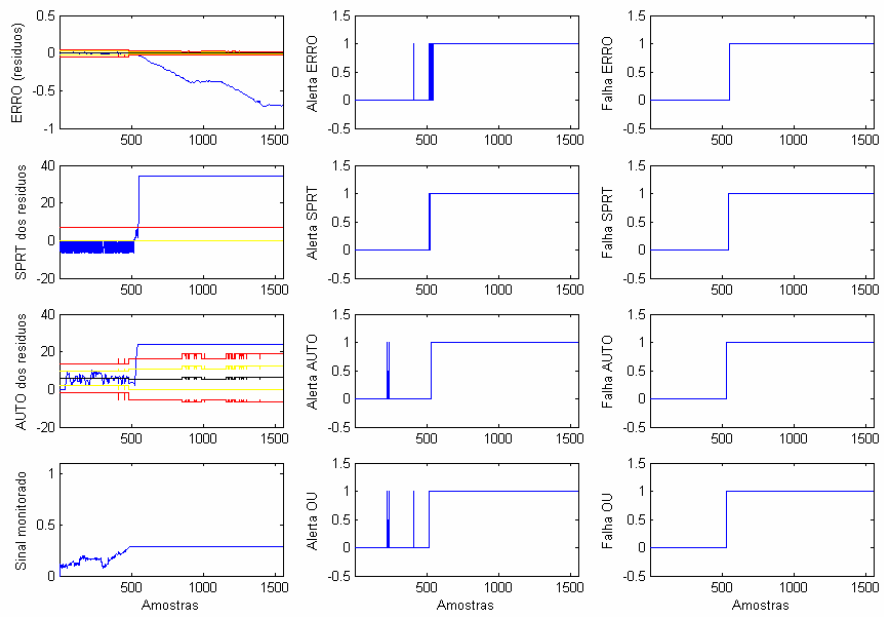


Figura 5.40 - Resultados obtidos para o sinal monitorado com falha do tipo colado.

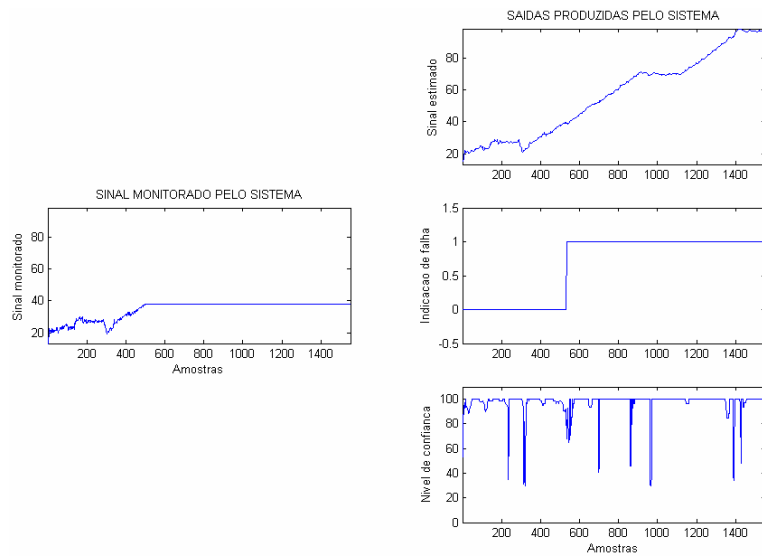


Figura 5.41 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzidos pelo sistema.

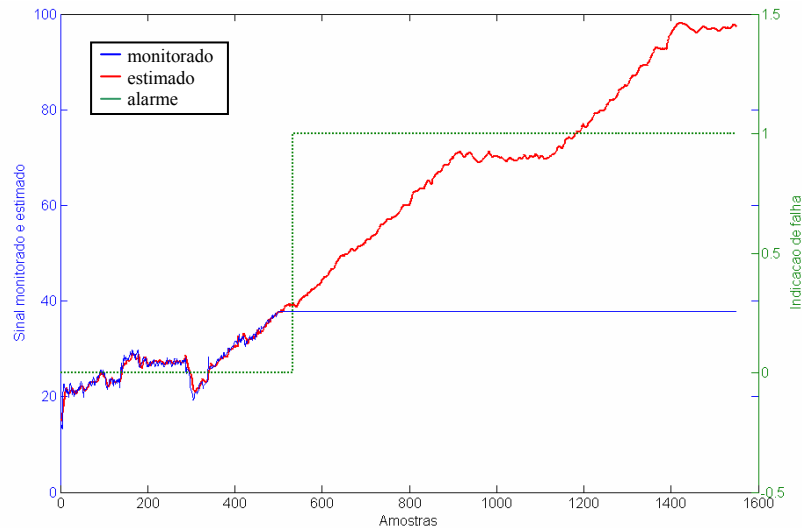


Figura 5.42 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada e de alarme produzidos pelo sistema.

5.3.1.3.4 Resumo da detecção de falhas pelo sistema - Operação em subida de potência

Outros tipos de falhas foram inseridas no sinal monitorado pelo sistema conforme mostrado na tabela 5.6. Nesta tabela são apresentadas a quantidade de amostras adquiridas pelo sistema de validação de sinal a partir da inserção de uma determinada falha para a produção do sinal de alarme. Como o sistema utiliza uma lógica OU entre os sinais de falha produzidos por cada método de detecção, o número de amostras necessárias para detecção pelo sistema é o menor valor produzido dentre os três métodos. Novamente, se considerarmos o tempo entre as amostras adquiridas pelo sistema igual a 1 segundo, os valores apresentados na tabela seriam os tempos em segundos necessários para detecção dos alarmes a partir de seu início.

Tabela 5.6

Quantidade de amostras necessária a partir da introdução da falha para detecção da falha pelo sistema - Operação em subida de potência.

Falha introduzida	ERRO	SPRT	AUTO	SOMA(OU)
Mudança brusca positiva (<i>positive hard-over</i>)	9	9	43	9
Mudança brusca negativa (<i>negative hard-over</i>)	14	13	28	13
Polarização positiva (<i>positive bias</i>)	7	5	41	5
Polarização negativa (<i>negative bias</i>)	10	7	26	7
Pico positivo (<i>positive spike</i>)	0	1	---	0
Pico negativo (<i>negative spike</i>)	0	1	---	0
Colado (<i>stuck</i>)	51	48	34	34
Errático (<i>erratic</i>)	101	48	---	48
Cíclico (<i>cyclic</i>)	21	14	---	14
Desvio positivo (<i>positive drift</i>)	386	269	428	269
Desvio negativo (<i>negative drift</i>)	403	353	430	353
Não linear (<i>nonlinear</i>)	107	78	128	78

--- Não detectou a falha.

5.3.2 Monitoração da temperatura da água na perna quente

5.3.2.1 Reator operando sem falha no sinal monitorado - Ensaio 8

Neste ensaio o sistema de validação de sinal recebeu como entrada as mesmas amostras utilizadas no seu treinamento. As figuras 5.43 a 5.45 apresentam os resultados obtidos no ensaio.

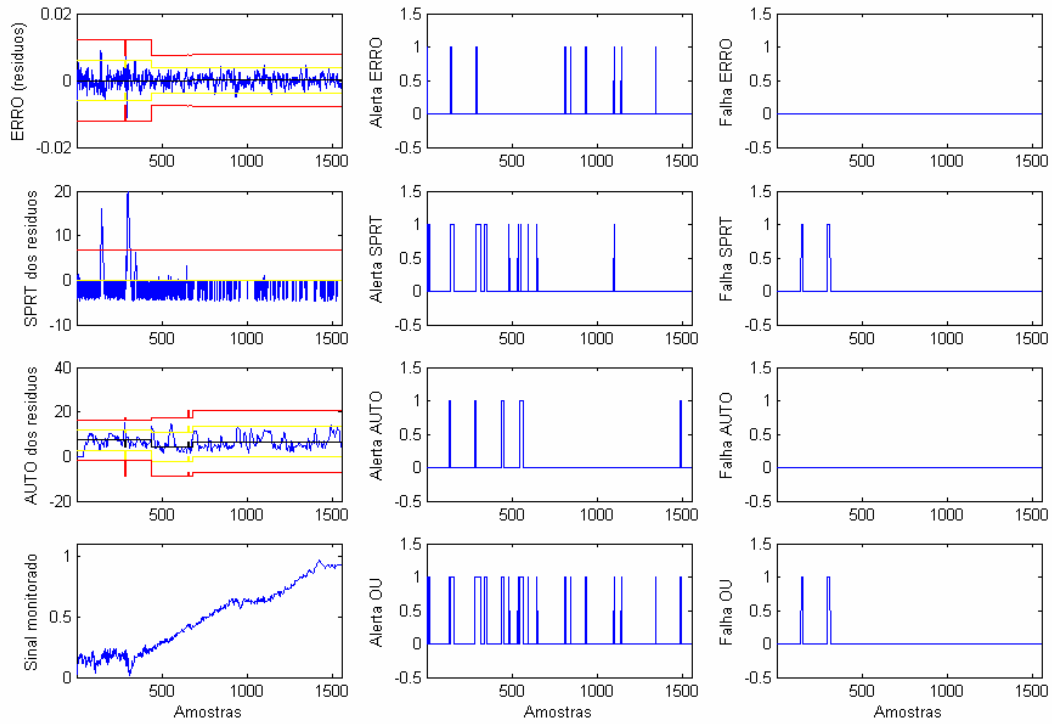


Figura 5.43 - Resultados obtidos pelo sistema de validação de sinal para o sinal monitorado sem falhas.

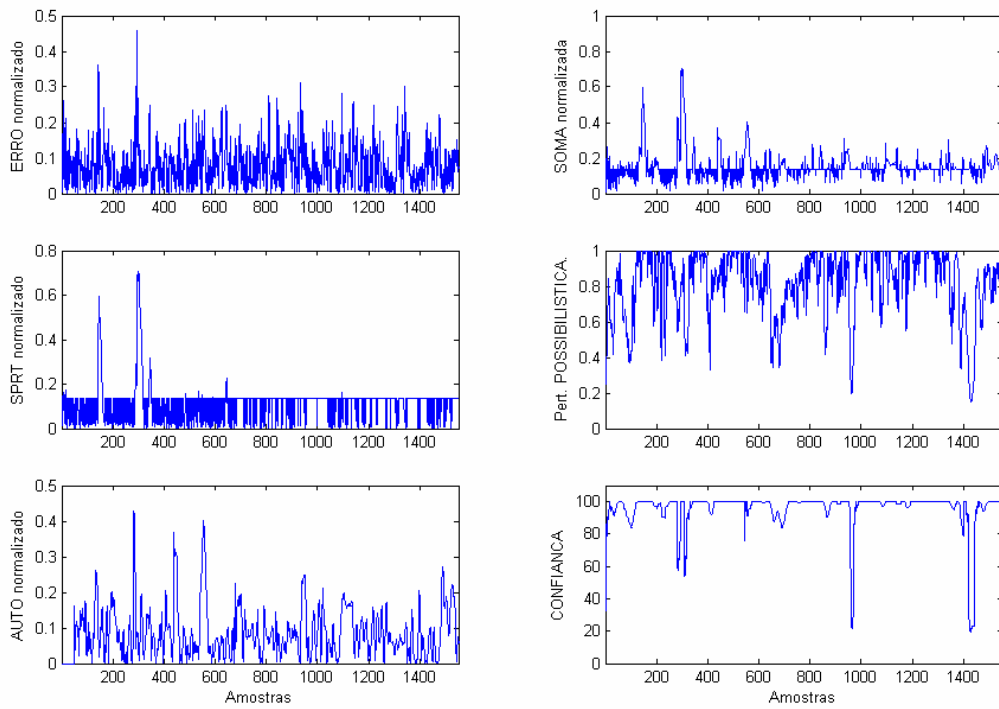


Figura 5.44 - Confiância da resposta obtida pelo sistema de validação de sinal para o sinal da temperatura da água na perna quente sem falhas.

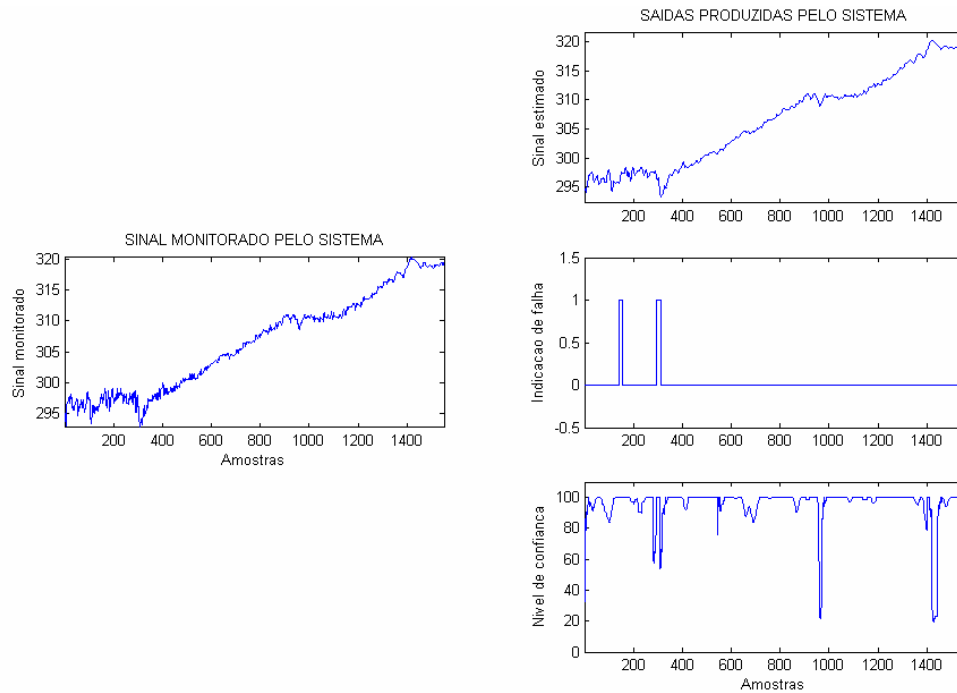


Figura 5.45 - Sinal da temperatura da água na perna quente monitorada e sinais da temperatura da água na perna quente estimada, de alarme e da confiança da resposta produzida pelo sistema.

5.3.3 Monitoração da potência nuclear do reator fora do intervalo de treinamento - Ensaio 9

Neste ensaio o sistema de validação de sinal foi treinado com um conjunto de dados da planta que cobria o intervalo de 40% a 100% da potência nuclear do reator. As figuras 5.46 e 5.47 apresentam os resultados obtidos para este intervalo no ensaio.

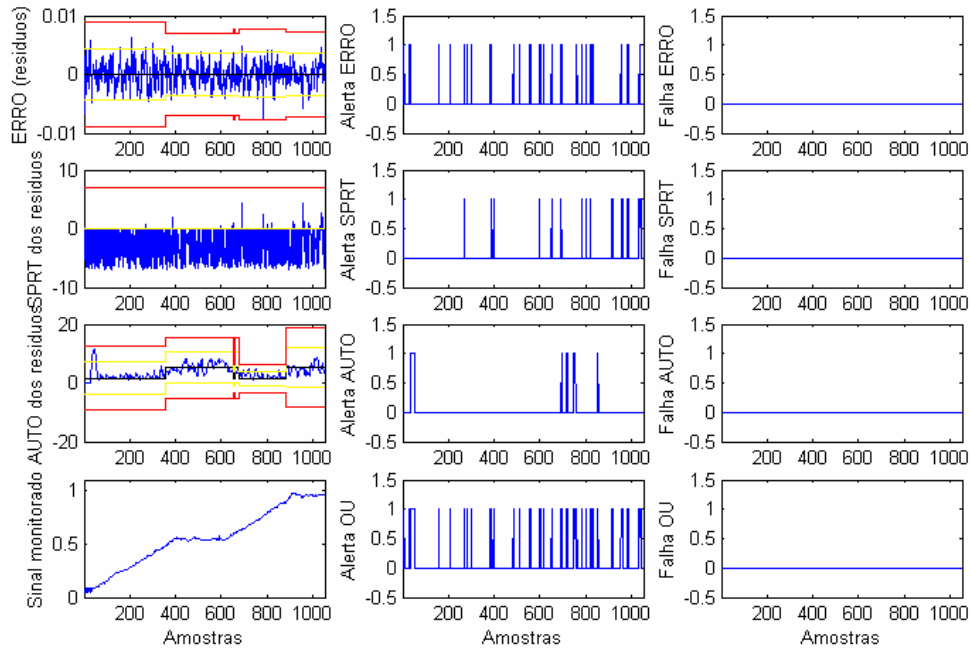


Figura 5.46 - Resultados obtidos para o sinal monitorado dentro do intervalo de treinamento.

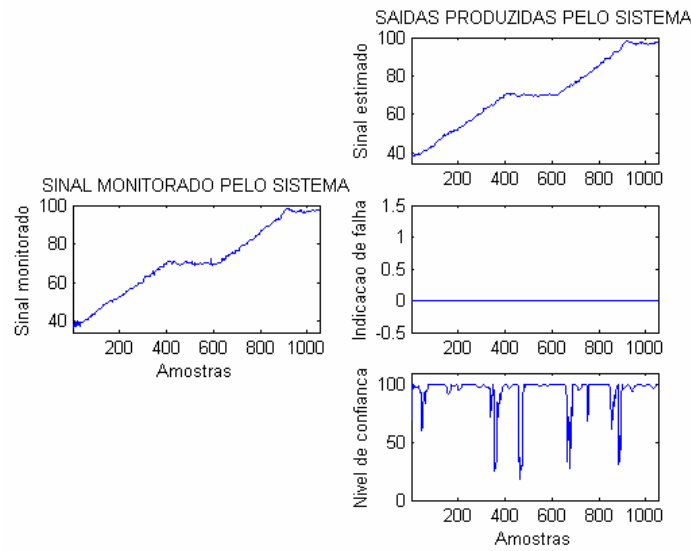


Figura 5.47 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzida pelo sistema.

A seguir o sistema foi posto à operar com todo o conjunto de dados disponível, ou seja, de desde a partida até atingir 100% de potência. As figuras 5.48 a 5.50 apresentam os resultados obtidos pelo sistema de validação de sinal neste ensaio.

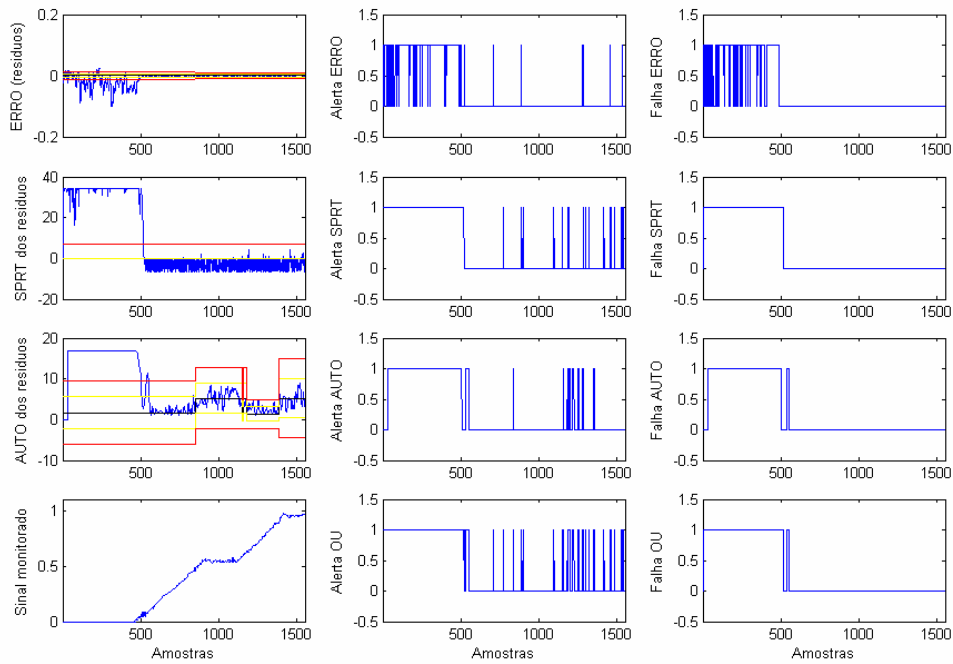


Figura 5.48 - Resultados obtidos para o sinal monitorado fora do intervalo de treinamento.

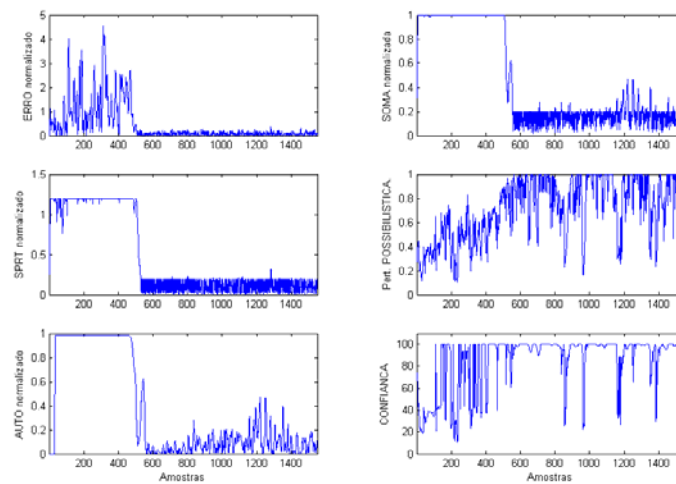


Figura 5.49 - Confiança da resposta obtida pelo sistema de validação de sinal para o sinal da potência nuclear fora do intervalo de treinamento.

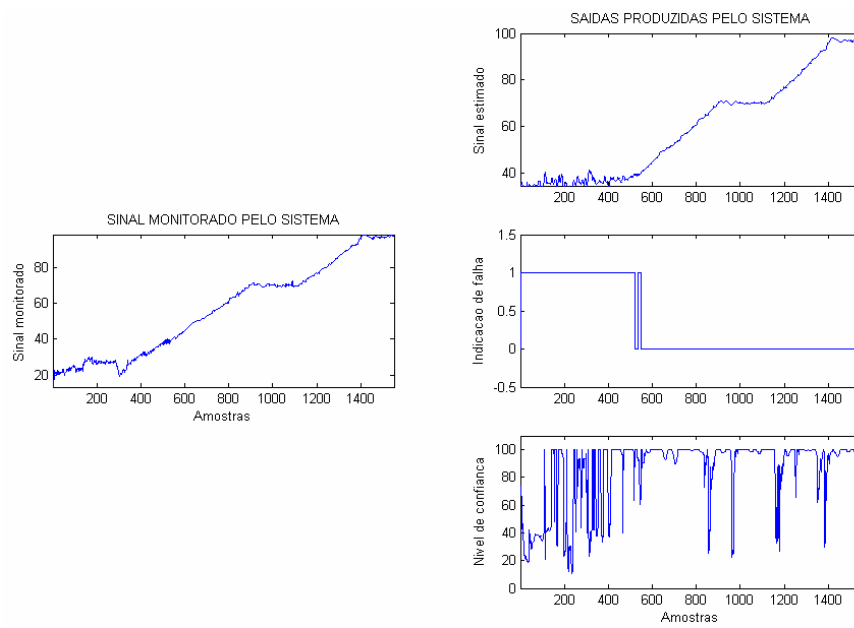


Figura 5.50 - Sinal da potência nuclear monitorada e sinais da potência nuclear estimada, de alarme e da confiança da resposta produzida pelo sistema.

CAPÍTULO 6

Conclusões e comentários

Nesta tese foi apresentada uma metodologia para validação de sinal usando modelos empíricos através de técnicas de inteligência artificial.

Com o intuito de validar a metodologia proposta, foram efetuados ensaios nas diversas etapas que compõem um sistema de validação de sinal. Algumas das técnicas de implementação empregadas em cada etapa do processo de validação de sinal, utilizadas em trabalhos anteriores e propostas nesta tese, foram ensaiadas com o objetivo de selecionar a melhor técnica a ser aplicada em cada etapa. Além disso, foram apresentados e analisados, segundo a metodologia proposta, alguns dos modelos de validação de sinal desenvolvidos anteriormente. Com base nesta análise e nos resultados obtidos nos ensaios realizados nas diversas sub-etapas que compõem um modelo de validação de sinal, foi proposto o modelo de validação de sinal implementado nesta tese.

O modelo implementado foi ensaiado com dados de operação da planta nuclear de potência Angra I. Os resultados obtidos nos ensaios realizados mostraram o bom desempenho do modelo implementado na detecção de falhas e na produção da estimativa dos sinais monitorados.

6.1 Avaliação dos ensaios das sub-etapas de composição de um modelo

Diferentes técnicas de implementação das sub-etapas que compõem um modelo de validação de sinal foram propostas, implementadas e ensaiadas com o objetivo de se definir qual aplicar em cada sub-etapa do modelo.

Para a seleção das variáveis de entrada a serem utilizadas na validação de um determinado sinal da planta foram implementadas duas técnicas baseadas na utilização

de algoritmos genéticos (GAs) para seleção das variáveis de entrada do modelo, utilizando como função-objetivo as correlações entre os sinais disponíveis como entrada e o sinal de saída do modelo a ser validado. Apesar do primeiro método proposto utilizar para esta seleção somente as correlações lineares entre os sinais envolvidos no modelo, ele foi escolhido para utilização no modelo implementado por possuir somente duas constantes (α_1 e α_2) a serem definidas pelo usuário, o que facilita sua utilização.

Para filtragem dos sinais aplicados ao sistema de validação de sinal foram implementados três métodos: média móvel, média ponderada e ondaletas. Para diminuir o tempo de resposta dos três filtros digitais implementados foi desenvolvido um algoritmo de janela ou peso variável. Os resultados obtidos nos testes efetuados revelaram desempenho similares para os filtros digitais por média móvel e por ondaletas. O filtro digital por ondaletas foi selecionado para utilização no sistema de validação de sinal implementado.

Para a sub-etapa de redução da dimensão dos sinais de entrada para o sistema de validação de sinal, foi implementado o método por PCA. Os ensaios realizados neste método mostraram sua deficiência quando são aplicados valores de variáveis às entradas da matriz de transformação (redução) do PCA que estão fora do intervalo utilizado para obtenção da matriz. Por exemplo, uma variável com valores fora do intervalo de treinamento do PCA provoca uma alteração em todas as componentes principais geradas pelo PCA, ou seja, todo o espaço da transformação é deformado e não somente uma das componentes principais. Deste modo, no sistema implementado não é utilizada a redução de dimensão.

A métrica utilizada no modelo implementado foi a métrica Mahalanobis. A seleção da métrica utilizada foi feita com base no fato de que para o modelo implementado deve-se obter classes que representem as diversas regiões de operação da

planta. Normalmente, as regiões de operação de uma planta industrial são superfícies delgadas e a métrica Mahalanobis, que constrói classes em forma de elipses, se adapta melhor a essas superfícies.

Para a separação em classes das amostras de treinamento e de operação do sistema de validação de sinal foram comparados o desempenho dos métodos de separação em classes nebulosas com protótipos pontuais (FCM) e com protótipos com volume (EFCM). Os ensaios realizados mostraram que o método com protótipos com volume apresenta melhor desempenho tanto para a determinação do número de classes em um conjunto de dados quanto para a precisão na determinação dos centros das classes. O método EFCM foi utilizado no sistema de validação de sinal implementado.

Para o processamento dos sinais foram utilizados sistemas de inferência nebulosa adaptativo (ANFIS). Foram ensaiados três métodos de treinamento do ANFIS: o GD+LSE, o GA+LSE e o PSO+LSE. Os resultados obtidos nos ensaios mostraram que apesar das técnicas evolucionárias, tais como GA e PSO, necessitarem de muitas épocas para treinamento quando existem muitos parâmetros a serem otimizados, elas podem evitar o problema dos algoritmos convencionais de treinamento, como o GD, de ficar preso a ótimos locais durante a fase de treinamento do modelo. Para o sistema implementado optou-se por utilizar a otimização por enxame de partículas (PSO) para a parte antecedente do ANFIS e a técnica dos mínimos quadrados (LSE) para a sua parte conseqüente.

Para a detecção de anomalia ou falha de sensor foram implementados três métodos de detecção: o método do erro ou resíduos, o método da auto-correlação dos resíduos e o método do teste da razão de probabilidades seqüencial dos resíduos. Os resultados dos ensaios realizados mostraram que nenhum dos três métodos implementados foi capaz de identificar sozinho todas as falhas inseridas. Deste modo,

optou-se por efetuar uma combinação desses métodos. Foram estudadas duas maneiras de combinar os métodos de detecção de falhas: através de uma lógica de votação majoritária e através de uma lógica tipo OU. Os resultados obtidos nos ensaios mostraram que apesar do aumento da probabilidade de falsos alarmes do sistema de detecção, a lógica tipo OU mostrou-se vantajosa, pois se um método de detecção é deficiente na detecção de uma determinada falha e outro não é, a resposta final do sistema não é comprometida. Para o sistema de validação de sinal implementado optou-se por usar os três métodos de detecção de falhas combinados através de uma lógica de votação do tipo OU.

Finalmente, para se determinar a confiança da resposta produzida pelo sistema de validação de sinal foi utilizado um sistema de inferência nebulosa (FIS) do tipo Mamdani que recebe às suas entradas os sinais de resíduos produzidos pelos três sistemas de detecção de falhas e um sinal de pertinência das amostras de entrada do sistema ao conjunto de treinamento e fornece à sua saída uma indicação quantitativa sobre a confiança da resposta produzida pelo sistema. Os sinais de resíduos dos sistemas de detecção de falhas foram normalizados e combinados, através de uma lógica *max*, de tal forma que somente um sinal resultante é aplicado à entrada do FIS. Para o sinal de pertinência das amostras ao conjunto de dados de treinamento foram comparadas as técnicas de pertinência nebulosa e possibilística.

Lembramos que existem métodos de separação em classes baseados em lógica nebulosa que não fornecem valor de pertinência apropriados para aplicações nos quais as pertinências possam ser interpretadas como graus de possibilidade. Isto é devido ao fato que eles usam uma restrição inerentemente probabilística, que fornece valores relativos. Como resultado, a pertinência de uma amostra em uma classe não depende somente de onde ela está localizada em relação à classe, mas também o quão distante

ela está com respeito a outras classes. Então, esta “lei de conservação da pertinência total” força as pertinências a serem espalhadas nas classes, e a faz ser dependente do número de classes presentes. Embora isso seja desejável em situações onde as pertinências servem como indicação das probabilidades ou graus de compartilhamento, o valor da pertinência resultante não pode distinguir entre bons e maus membros de uma classe.

Por outro lado, a abordagem possibilística parte da premissa que a pertinência de uma amostra em uma classe não tem nada a ver com sua pertinência em outras classes.

Os resultados dos ensaios realizados nas técnicas de pertinência nebulosa e possibilística confirmaram que a técnica possibilística é vantajosa em relação à nebulosa para a aplicação onde se deseja obter o grau de pertinência de uma amostra a uma classe. Deste modo, no modelo de confiança da resposta do sistema foi utilizado a pertinência possibilística das amostras.

6.2 Avaliação dos ensaios do sistema de validação de sinal aplicado a planta nuclear

O modelo implementado foi posto a operar na monitoração dos sinais da potência nuclear e da temperatura da perna quente do circuito 1 da usina Angra I. O modelo implementado pode ser avaliado quanto ao seu desempenho através da análise dos resultados obtidos nos ensaios realizados na validação dos sinais monitorados da operação da usina Angra I.

6.2.1 Avaliação do sistema na validação do sinal de potência nuclear

No ensaio 1, o sistema recebeu como entrada as mesmas amostras que foram utilizadas no seu treinamento para validação do sinal de potência gerada pelo reator nuclear. Pode-se observar pelos resultados obtidos (figura 5.24) que o sistema produziu

à sua saída um sinal de potência nuclear similar ao sinal de potência nuclear monitorada. É interessante ressaltar que a validação de sinal requer “... produção da melhor estimativa para o valor da variável que está sendo monitorada” e, para isto, a correlação entre as variáveis selecionadas com a variável monitorada é de fundamental importância, pois no caso de falha no sinal monitorado o sistema irá produzir a estimativa desse sinal falho através das correlações deste sinal com os sinais que estão aplicados às suas entradas. Estas correlações são aprendidas pelos sistemas de inferência nebulosa adaptativo (ANFIS) na fase de treinamento do sistema.

Além disso, a confiança da resposta no resultado produzido pelo sistema foi alta (em torno de 100%), garantindo que as amostras (pontos de operação da planta) apresentadas à entrada do sistema fazem parte do conjunto de treinamento do sistema.

Finalmente, não houve qualquer indicação de alarme pelo sistema pois o sinal monitorado não continha falhas.

No ensaio 2, a planta nuclear foi posta a operar com potência nuclear em torno de 70% e foi inserida uma falha do tipo polarização positiva no sinal monitorado a partir da amostra 500. O sistema indicou falha (figura 5.26) 7 amostras após a sua inserção (tabela 5.5). Neste caso, o método de identificação por SPRT foi o que apresentou menor tempo de resposta. O sistema também produziu uma estimativa para o valor monitorado e a confiança desta estimativa ficou em torno de 100%.

No ensaio 3, a planta nuclear foi novamente posta a operar com potência nuclear em torno de 70% e foi inserida uma falha do tipo desvio positivo no sinal monitorado a partir da amostra 500. O sistema indicou falha (figura 5.29) 66 amostras após sua inserção (tabela 5.5). Neste caso, o método de identificação por auto-correlação dos resíduos foi o que apresentou menor tempo de resposta. O sistema também produziu

uma estimativa para o valor monitorado e a confiança desta estimativa ficou em torno de 100%.

No ensaio 4, a planta nuclear foi novamente posta a operar com potência nuclear em torno de 70% e foi inserida uma falha do tipo colado no sinal monitorado a partir da amostra 500. O sistema indicou falha (figura 5.32) 43 amostras após sua inserção (tabela 5.5). Neste caso, somente o método por auto-correlação dos resíduos de identificou a falha inserida. Novamente, o sistema também produziu uma estimativa para o valor monitorado e a confiança desta estimativa ficou em torno de 100%.

Novamente, o sistema foi posto a operar com potência nuclear em torno de 70% e foram inseridos outros tipos de falhas. A tabela 5.5 apresenta um resumo da quantidade de amostras necessárias pelo sistema para detecção da falha, a partir de sua introdução. Os resultados obtidos nos ensaios mostraram que os métodos de erro e de razão de probabilidades dos resíduos foram capazes de detectar a falha do tipo colado. Por outro lado, o método de auto-correlação dos resíduos não foi capaz de detectar a falha do tipo pico.

No ensaio 5, a planta nuclear está subindo sua potência para plena potência e uma falha do tipo polarização negativa foi inserida no sinal monitorado, quando a potência está em torno de 40%, ou seja, a partir da amostra 500. O sistema indicou falha (figura 5.35) 7 amostras após sua inserção (tabela 5.6). Neste caso, o método de identificação por SPRT foi o que apresentou menor tempo de resposta. O sistema também produziu uma estimativa para o valor monitorado e a confiança desta estimativa ficou em torno de 100%.

No ensaio 6, novamente a planta nuclear está subindo sua potência para plena potência e uma falha do tipo desvio negativo foi inserida no sinal monitorado, quando a potência está em torno de 40%, ou seja, a partir da amostra 500. O sistema indicou falha

(figura 5.38) 353 amostras após sua inserção (tabela 5.6). Neste caso, o método de identificação por SPRT foi o que apresentou menor tempo de resposta. O sistema também produziu uma estimativa para o valor monitorado e a confiança desta estimativa ficou em torno de 100%.

No ensaio 7, novamente a planta nuclear está subindo sua potência para plena potência e uma falha do tipo colado foi inserida no sinal monitorado quando a potência está em torno de 40%, ou seja, a partir da amostra 500. O sistema indicou falha (figura 5.41) 34 amostras após sua inserção (tabela 5.6). Neste caso, o método de identificação por auto-correlação dos resíduos foi o que apresentou menor tempo de resposta. O sistema também produziu uma estimativa para o valor monitorado e a confiança desta estimativa ficou em torno de 100%.

O sistema foi posto a operar e a planta nuclear está subindo sua potência para plena potência e foram inseridos outros tipos de falhas. A tabela 5.6 apresenta um resumo da quantidade de amostras necessárias pelo sistema para detecção da falha, a partir de sua introdução. Neste caso, os ensaios realizados mostraram que o método de auto-correlação dos resíduos não foi capaz de detectar falhas do tipo pico, errática e cíclica.

6.2.2 Avaliação do sistema na validação da temperatura da água na perna quente

No ensaio 8, o sistema recebeu como entrada as mesmas amostras que foram utilizadas no seu treinamento para validação do sinal de temperatura da água na perna quente do circuito 1 do reator nuclear. Pode-se observar pelos resultados obtidos (figura 5.45) que o sistema produziu à sua saída um sinal de temperatura da água na perna quente do circuito 1 do primário do reator similar ao sinal de temperatura da água monitorada.

Além disso, a confiança da resposta no resultado produzido pelo sistema foi alta (em torno de 100%), garantindo que as amostras (pontos de operação da planta) apresentadas à entrada do sistema fazem parte do conjunto de treinamento do sistema.

Finalmente, observamos que houve duas indicações de alarmes falsos para esta variável monitorada. Isto é devido ao fato de que a lógica de votação adotada no sistema de validação de sinal favorece a indicação de alarmes falsos. Neste caso, o método do SPRT foi o que indicou erroneamente as falhas.

6.2.3 Avaliação do sistema na validação da potência nuclear fora do intervalo de treinamento

No ensaio 9, o sistema de validação de sinal foi treinado para monitorar a potência gerada pelo reator nuclear com um conjunto de dados da planta que cobria o intervalo de 40% a 100% da potência nuclear.

A seguir, o sistema recebeu como entrada as mesmas amostras que foram utilizadas no seu treinamento. Pode-se observar pelos resultados obtidos (figura 5.47) que o sistema produziu à sua saída um sinal estimado da potência nuclear do reator similar ao sinal da potência nuclear monitorada.

Além disso, a confiança da resposta no resultado produzido pelo sistema foi alta (em torno de 100%), garantindo que as amostras (pontos de operação da planta) apresentadas à entrada do sistema fazem parte do conjunto de treinamento do sistema.

Finalmente, não houve qualquer indicação de alarme pelo sistema pois o sinal monitorado não continha falhas.

A seguir o sistema foi posto à operar com todo o conjunto de dados disponível, ou seja, desde a partida até atingir 100% de potência. Os resultados obtidos (figura 5.50) mostram uma grande discrepância entre o sinal monitorado e estimado pelo sistema na faixa que o mesmo não foi treinado.

Para as amostras fora do intervalo de treinamento o sistema gerou uma indicação de falha no sinal monitorado, porém, a indicação de confiança da resposta do sistema para essas amostras é baixa indicando para o operador da planta não confiar nas informações fornecidas pelo sistema para este intervalo.

Numa condição real o operador da planta não saberia que uma condição de operação não foi previamente treinada pelo sistema, porém a indicação de confiança da resposta do sistema fornece ao operador a credibilidade nos valores estimados pelo sistema.

6.2.4 Avaliação dos tempos de resposta do sistema para as falhas inseridas

A tabela 6.1 apresenta um resumo da quantidade de amostras necessárias a partir da introdução de uma falha para sua detecção pelo sistema de validação de sinal. Os valores apresentados na tabela foram obtidos nos ensaios realizados na planta de nuclear na monitoração da potência do reator operando em estado estacionário (70% de potência) e em subida de potência.

Se considerarmos o tempo entre as amostras adquiridas pelo sistema igual a 1 segundo, os valores apresentados na tabela seriam os tempos em segundos necessários para detecção dos alarmes a partir de seu início.

Deve-se ter em mente que uma planta de potência nuclear apresenta tempo de resposta a transientes termo-hidráulicos elevados e, deste modo, os tempos para detecção das diferentes falhas pelo sistema podem ser considerados como satisfatórios.

No caso da falha do tipo desvio os tempos de resposta obtidos foram elevados quando comparados com os tempos de resposta obtidos para as outras falhas, principalmente para o caso da operação em subida de potência. Isto deve-se a própria natureza da falha tipo desvio que se confunde com a variação no processo monitorado.

Tabela 6.1

Quantidade de amostras necessária a partir da introdução da falha para detecção da falha pelo sistema.

Falha introduzida	Operação em estado estacionário	Operação em subida de potência
Mudança brusca positiva (positive hard-over)	21	9
Mudança brusca negativa (negative hard-over)	23	13
Polarização positiva (positive bias)	7	5
Polarização negativa (negative bias)	8	7
Pico positivo (positive spike)	0	0
Pico negativo (negative spike)	0	0
Colado (stuck)	43	34
Errático (erratic)	21	48
Cíclico (cyclic)	76	14
Desvio positivo (positive drift)	66	269
Desvio negativo (negative drift)	93	353
Não linear (nonlinear)	73	78

6.3 Contribuições/ inovações do trabalho de tese

As principais contribuições/ inovações do trabalho de tese são:

- a) O desenvolvimento de uma metodologia para composição de um sistema de validação de sinal por modelos empíricos usando técnicas de inteligência artificial;
- b) O desenvolvimento de metodologias para aplicação de algoritmos genéticos na seleção de variáveis de entrada para um modelo de validação de sinal a partir do conjunto de variáveis disponíveis da planta;
- c) A aplicação de otimização por enxame de partículas (PSO) no treinamento de sistemas de inferência nebulosa adaptativo para otimização da sua parte antecedente;
- d) A construção de um modelo de validação de sinal contendo as inovações introduzidas.

6.4 Recomendações para trabalhos futuros

A seguir são sugeridas alguns pontos de interesse para pesquisa futura nos temas abordados nesta tese.

- Os ensaios realizados no método de redução da dimensão do espaço de entrada por PCA não se mostraram satisfatórios quando são aplicados valores de variáveis às entradas da matriz de transformação (redução) do PCA que estão fora do intervalo utilizado para obtenção da matriz. Como sugestão pode-se utilizar para efetuar esta redução a técnica por ondaletas (*wavelets*).
- Ampliar a quantidade de sinais disponíveis da planta de modo a propiciar uma melhor seleção das variáveis de entrada para o sistema de validação de sinal que sejam fortemente correlacionadas com a variável a ser monitorada e estudar melhor como ajustar as constantes alfas das funções objetivos de seleção das variáveis de entrada para o modelo.
- Estudar outras formas de partição das funções pertinências (grande, média e pequena) das variáveis nebulosas utilizadas no cálculo da confiança da resposta dos modelos.
- O método de detecção de falhas utilizado é uma combinação lógica tipo OU dos três métodos implementados. Esta combinação propicia a produção de eventos do tipo falsos alarmes pelo sistema de validação de sinal. Por outro lado uma lógica tipo majoritária dos três métodos implementados pode levar a perda de alarmes pelo sistema de validação de sinal. Como sugestão propomos estudar um método de combinação dos diferentes métodos de detecção de falhas que minimize mais a produção de falsos alarmes e a perda de alarmes pelo sistema de validação de sinal.

Referências bibliográficas

- ALLEY, G. T., BAUER, M. L., 1988, "Data Processing and Display Algorithms for Portable Instruments", *IEEE Transaction on Nuclear Science*, v. 35, n. 1 (Feb), pp. 559-562.
- ANDERBERG, M.R., 1975, *Cluster Analysis for Applications*. 1ed. Academic press.
- CAIN, D., ZEBROSKI, E., 1980, "The Conceptual Design of a Power Plant Safety Panel", *Nuclear Engineering Instruments*, v. 25, n. 303 (Aug), pp. 40-44.
- CAMERON, B., 1993, "Final Report on CH-46 AFT Transmission Seeded Fault Testing", Westland Research Paper RP907, Westland Helicopters, Ltd.
- CHEN, J., LIU, J., 1999, "Mixture Principal Component Analysis Models for Process Monitoring", *Ind. Eng. Chem. Res.*, v. 38, n. 1478.
- CHIEN, T., ADAMS, M. B., 1976, "A Sequential Failure Detection Technique and Its Application", *IEEE Transaction Autom. Control*, v. 21, pp. 750-757.
- CLARK, R. N., FOSTH, D. C., WALTON, V. M., 1975, "Detecting Instrument Malfunctions in Control Systems", *IEEE Trans. Aerosp. Electron. Syst.*, v. AES-11, n. 4 (Jul), pp. 465-473.
- CLARK, R. N., MASRELIEZ, C. J., BURROWS, J. W., 1976, "A Functionally Redundant Altimeter", *IEEE Trans. Aerosp. Electron. Syst.*, v. AES-12, n. 4 (Jul), pp. 459-463.
- DAUBECHIES, I., 1992, "Ten Lectures on Wavelets", SIAM.
- FANTONI, P. F., FIGEDY, S., RACZ, A., 1998, *Peano, A Toolbox for Real-Time Process Signal Validation and Estimation*. In: Report HWR 515 (Feb), OECD Halden Reactor Project.
- FANTONI, P. F., MAZZOLA, A., 1996, "Multiple-Failure Signal Validation in Nuclear Power Plants Using Artificial Neural Networks", *Nuclear Technology*, v. 113, n. 3 (Mar), pp. 368-374.

- FLTBx, 2000, *Fuzzy Logic Toolbox for Use with MATLAB*, The MathWorks, Inc., User's Guide - version 2.
- GATH, I., GEVA, A. B., 1989, "Unsupervised Optimal Fuzzy Clustering", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, v. 11, n. 7(Jul), pp. 773-781.
- GEATbx, 2000, *Genetic and Evolutionary Algorithm Toolbox for Use with MATLAB*, The MathWorks, Inc., User's Guide - version 3.3.
- GOLDBERG, D. E., 1989, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, Massachusetts, Addison Wesley.
- GUSTAFSON, D., KESSEL, W., 1979, "Fuzzy Clustering with a Fuzzy Covariance Matrix", *In Proc. IEEE CDC*, San Diego, USA, pp. 761-766.
- HARDY, G. H., LITTLEWOOD, J. E., POLUA, G., 1973, *Inequalities*. 2ed. London, Cambridge University Press.
- HAYKIN, S., 1994, *Neural Networks - A Comprehensive Foundation*. 1 ed. New Jersey, Prentice-Hall.
- HAYMAK, U., VAN NAUTA LEMKE, H. R., 1998, "A Sensitivity Analysis Approach to Introducing Weight Factors into Decision Function in Fuzzy Multicriteria Decision Making", *Fuzzy Sets and Systems*, v. 97, n. 2 (Jul), pp. 169-182.
- HEO, G., CHANG, S. H., 2003, "Comparative Study on State Analysis of BOP in NPPs", *IEEE Transaction on Nuclear Science*, v. 50, n. 4 (Aug), pp. 1271-1281.
- HINES, J. W., WREST, D. J., UHRIG, R. E., 1997, "Signal Validation Using an Adaptive Neural Fuzzy Inference System", *Nuclear Technology*, v. 119, n. 8 (Aug), pp. 181-193.
- HOLBERT, K. E., 1991, "Process Hypercube Comparison for Signal Validation", *IEEE Transactions on Nuclear Science*, v. 38, n. 2 (Apr), pp. 803-811.

- HOLBERT, K. E., HEGGER, A. S., ALANG-RASHID, N. K., 1994, “Redundant Sensor Validation by Using Fuzzy Logic”, *Nuclear Science and Engineering*, v. 118, n. 9 (Sep), pp. 54-64.
- HOLBERT, K. E., UPADHYAYA, B. R., 1990, “An Integrated Signal Validation System for Nuclear Power Plants”, *Nuclear Technology*, v. 92, n. 12 (Dec), pp. 411-427.
- HOLLAND, J. H., 1975, “Adaptation in Natural and Artificial Systems”, Ann Arbor: *The University of Michigan Press*.
- <http://neural.cs.nthu.edu.tw/jang/benchmark/>, Human Operation at a Chemical Plant - Data Set, *Working Group on Data Modeling Benchmarks*, IEEE Neural Networks Council Standards Committee.
- <http://www.spectraquest.com>, *Spectra Quest, SQ Machinery Fault Simulator*, SQI, SpectraQuest Inc.
- JACKSON, J. E., 1991, “A User’s Guide to Principal Components”, John Wiley and Sons, Inc., pp. 1-25.
- JEONG, J. J., HA, K. S., CHUNG, B. D., LEE W. J., 1999, “Development of a Multi-Dimensional Thermal-Hydraulic System Code, MARS 1.3.1”, *Annals of Nuclear Energy*, vol. 26.
- KALMAN, R. E., 1960, “A New Approach to Linear Filtering and Prediction Problems”, *Transaction of the ASME, Journal of Basic Engineering* (Mar), pp. 35-45.
- KAYMAK, U., BABUSKA, R., 1995, “Compatible Cluster Merging for Fuzzy Modelling”, In: *Proceedings of the Fourth IEEE International Conference on Fuzzy Systems*, v. 2 (Mar), pp. 897-904, Yokohama, Japan.
- KAYMAK, U., SETNES, M., 2000, “Extended Fuzzy Clustering Algorithms”, *ERIM Report Series Research in Management*, ERS-2000-51-LIS (Nov), ERIM.

- KELLER, A., KLAWONN, F., 1999, "Clustering with Volume Adaptation for Rule Learning". In: *Proceedings of Seventh European Congress on Intelligent Techniques and Soft Computing*, EUFIT'99 (Sep), Aachen, Germany.
- KENNEDY, J., EBERHART, R. C., 1995, "Particle Swarm Optimization", *Proceeding IEEE Conference on Neural Networks*, IV, Piscataway, New Jersey, pp. 1942-1948.
- KLIR, G. T., CLAIR, U. H. S., YUAN, B., 1997, *Fuzzy Set Theory*, 1 ed, New Jersey, Prentice-Hall.
- KOSKO, B., 1992, *Neural Networks and Fuzzy Systems*. 1 ed. New Jersey, Prentice-Hall.
- KOSKO, B., 1997, *Fuzzy Engineering*. 1 ed. New Jersey, Prentice-Hall.
- KOVÁCS, Z. L., 1996, *Redes Neurais Artificiais - Fundamentos e Aplicações*. 2 ed. São Paulo, Edição Acadêmica.
- KRAMER, M. A., 1991, "Nonlinear Principal Component Analysis Using Autoassociative Neural Network", *AICHE Journal*, v. 37, n. 2 (Feb), pp. 233-243.
- KRISHNAPURAM, R., KELLER, J. M., 1993, "A Possibilistic Approach to Clustering", *IEEE Transactions on Fuzzy Systems*, v. 1, n. 2 (May), pp. 98-110.
- KRISHNAPURAM, R., KIM, J., 2000, "Clustering Algorithm Based on Volume Criteria", *IEEE Transaction on Fuzzy Systems*, v. 8, n. 2 (Apr), pp. 228-236.
- KRÖSE, B. J. A., SMAGT, P. P., 1993, *An Introduction to Neural Networks*. 5 ed. The University of Amsterdam.
- MALLAT, S., 1989, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, v. 11, n. 7 (Jul), pp. 674-693.
- MEIJER, C. H., *et al.*, 1981, "On-Line Power Plant Signal Validation Technique Utilizing Parity-Space Representation and Analytic Redundancy", EPRI NP-2110.

- MISITI, M., MISITI, Y., OPPENHEIM, G., POGGI, J.-M., 2000, “Wavelet Toolbox User’s Guide”, MathWorks, version 2 (Sep).
- NA, M. G., OH, S., 2002, “A Neuro-Fuzzy Inference System Combining Wavelet Denoising, Principal Component Analysis, and Sequential Probability Ratio Test for Sensor Monitoring”, *Nuclear Technology*, v. 140, n. 11 (Nov), pp. 178-197.
- NA, M. G., SHI, S. H., JUNG, D. W., 2005, “Design of a Software Sensor for Feedwater Flow Measurement Using a Fuzzy Inference System”, *Nuclear Technology*, v. 150, n. 6 (Jun), pp. 293-302.
- NABESHIMA, K., SUZUDO, T., SUZUKI, K., TÜRKCAN, E., 1998, “Real-time Nuclear Power Plant Monitoring with Neural Network”, *Journal of Nuclear Science and Technology*, v. 35, n. 2 (Feb), pp. 93-100.
- NING, J. N., CHOU, H. P., 1992, “Construction and Evaluation of Fault Detection Network for Signal Validation”, *IEEE Transaction on Nuclear Science*, v. 39, n. 4 (Apr), pp. 943-947.
- NUCLEAR NEWS, 1982, “Computer-Based Display to Aid Operators”, *Nuclear News*, v. 25, n. 13 (Oct), pp. 45-48.
- NUREG/CR-4166, 1985, “Analysis of FLECHT SEASET 163-Rod Blocked Bundle Data Using COBRA-TF”, *U. S. Nuclear Regulatory Commission*, Nuclear Regulatory Research.
- NUREG/CR-5535, 1995, “RELAP/MOD Code Manual, Volume I: Code Structure, System Models, and Solution Methods”, *U. S. Nuclear Regulatory Commission*, Nuclear Regulatory Research, (June).
- NUREG/CR-6127, 1994, “The Effects of Stress on Nuclear Power Plant Operational Decision Making and Training Approaches to Reduce Stress Effects”, *U. S.*

- Nuclear Regulatory Commission*, Nuclear Regulatory Research Washington, DC 20555-0001.
- NUREG/CR-6343, 1995, “On-Line Testing of Calibration of Process Instrumentation Channels in Nuclear Power Plants”, , *U. S. Nuclear Regulatory Commission*, Nuclear Regulatory Research, Phase II Final Report (Nov).
- NUREG-0585, 1979, “TMI-2 Lessons Learned Task Force, Final Report”, *U. S. Nuclear Regulatory Commission*, Nuclear Reactor Regulations Report (Oct), NTIS.
- OLIVEIRA, M. V., 1999, *Modelos de Validação de Sinal Utilizando Técnicas de Inteligência Artificial Aplicados a um Reator Nuclear*. Tese de M.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- OLIVEIRA, M. V., SCHIRRU, R., 2005, “A Neuro-Fuzzy Inference System Tuned by Particle Swarm Optimization Algorithm for Sensor Monitoring”. *INAC 2005 - International Nuclear Atlantic Conference*, Santos, Brasil.
- RICH, E., 1988, *Inteligência Artificial*, 1 ed. São Paulo, McGraw-Hill.
- RUSSEL, S. J., NORVIG, P., 2003, *Artificial Intelligence - A Modern Approach*. 2ed. Prentice Hall Series in Artificial Intelligence.
- SACKETT, J. I., 1988, “Application of AI Technology to Nuclear Plant Operation”, *ASEE Annual Conference*, CONF-8806146-2 (Jun), Portland, Oregon.
- SETNES, M., BABUSKA, R., KAYMAK, U., VAN NAUTA LEMKE, H. R., 1998, “Similarity Measures in Fuzzy Rule Base Simplification”, *IEEE Transactions on Systems, Man and Cybernetics, Part B*, v. 28, n. 3 (Mar), pp. 376-386.
- SETNES, M., KAYMAK, U., 1998, “Extended Fuzzy C-Means with Volume Prototypes and Cluster Merging”. In: *Proceedings of Sixth European Congress on Intelligent Techniques and Soft Computing*, ELITE, v. 2 (Sep), pp. 1360-1364.

- SUGENO, M., YASUKAWA, T., 1993, "A Fuzzy-Logic-Based Approach to Qualitative Modeling", *IEEE Transaction on Fuzzy Systems*, v. 1, n. 1 (Feb), pp. 7-31.
- TAN, S., MAVROVOUNIOTIS, M. L., 1995, "Reducing Data Dimensionality through Optimizing Neural Network Inputs", *AIChE Journal*, v. 41, n. 6 (Jun), pp. 1471-1480.
- TRELEA, I. C., 2003, "The Particle Swarm Optimization Algorithm: convergence analysis and parameter selection", *Information Processing Letters*, v. 85, pp. 317-325.
- UPADHYAYA, B. R., ERYUREK, E., 1992, "Application of Neural Networks for Sensor Validation and Plant Monitoring", *Nuclear Technology*, v. 97, n. 2 (Feb), pp. 170-176.
- UPADHYAYA, B. R., HOLBERT, K. E., KERLIN, T. W., 1989, "Development of an Integrated Signal Validation System and Application to Operating Power Plants", *Proceedings of the 7th Power Plant Dynamics, Control & Testing Symposium*, v. 2 (May), Knoxville, Tennessee.
- WALD, A., 1947, *Sequential Analysis*, New York, John Wiley & Sons.
- WASSERMAN, P. D., 1989, *Neural Computing - Theory and Practice*. 1ed. New York, Van Nostrand Reinhold.
- WINDHAM, M. P., 1982, "Cluster Validity for the Fuzzy C-Means Clustering Algorithm", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, v. 4, (Apr), pp. 357-363.
- WREST, D. J., HINES, J. W., UHRIG, R. E., 1997, "Instrument Surveillance and Calibration Verification through Plant Wide Monitoring using Autoassociative Neural Networks",
- XIE, X. L., BENI, G., 1991, "A Validity Measure for Fuzzy Clustering", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, v. 13, n. 8(Aug), pp. 841-847.
- YEN, G. G., FENG, W., 2001, "Winner Take All Experts Network for Sensor Validation", *ISA Transactions*, Elsevier, v. 40, pp. 99-110.

ZADEH, L. A., 1965, "Fuzzy Sets", *Information and Control*, v. 8, n. 3 (Mar), pp. 338-353.

ZADEH, L. A., 1978, "Fuzzy Sets as a Basis for a Theory of Possibility", *Fuzzy Sets and Systems*, v.1 (Jan), pp. 3-28.

ZIMMERMAN, H. J., ZYSNO, P., 1985, "Quantifying Vagueness in Decision Models", *European J. Operational Res.*, v. 22, pp. 148-158.

ANEXO I

Ondaletas

A técnica de filtragem por ondaleta (*wavelet*) é uma evolução da técnica de análise de Fourier. Na análise de Fourier um sinal é transformado do domínio do tempo para o domínio da frequência, isto é, o sinal é representado por suas componentes senoidais de diferentes frequências. A transformada de Fourier tem uma grande deficiência. Na transformação do domínio do tempo para o da frequência a informação temporal é perdida. Quando olhamos a transformada de Fourier de um sinal é impossível dizer quando ocorre um determinado evento, conforme mostrado na figura I.1.



Figura I.1 - Exemplo da transformada de Fourier.

Se as propriedades do sinal não mudam muito ao longo do tempo, isto é, se o sinal é do tipo estacionário, esta deficiência não é muito importante. Contudo, a maioria dos sinais de interesse possuem numerosas características não estacionária ou transitória: *drifts*, tendências, mudanças abruptas de começo e fim de eventos. Essas características são freqüentemente as partes mais importantes do sinal, e a análise de Fourier não é capaz de detectá-las.

Num esforço para corrigir esta deficiência, DENNIS GABOR (1946) adaptou a transformada de Fourier para analisar somente uma pequena seção do sinal no tempo. Esta técnica é conhecida como janela de sinal. Gabor chamou-a de transformada de Fourier para tempo curto (STFT - *Short Time Fourier Transform*), que mapeia um sinal em uma função de duas dimensões, tempo e frequência, conforme mostrado na figura I.2.



Figura I.2 - Transformada de Fourier para tempo curto.

O STFT representa uma espécie de compromisso, de como ver um sinal, entre a base de tempo e a frequência. Ela fornece alguma informação sobre quando e qual frequência ocorre um evento em um sinal. Contudo, esta informação só é obtida com uma certa limitação de precisão, que é determinada pelo tamanho da janela. O problema é que uma vez escolhido o tamanho da janela de tempo esta janela é a mesma para todas as frequências.

A análise por ondaleta (*wavelet*) representa o próximo passo lógico: uma técnica de janela com regiões de tamanhos variados. A análise por ondaleta permite usar intervalos de tempo longos (regiões grandes), quando se necessita informação de baixa frequência mais precisa e intervalos de tempo curtos (regiões pequenas), quando se deseja informação de alta frequência, conforme mostrado na figura I.3.



Figura I.3 - A transformada por ondaleta.

Uma ondaleta é uma forma de onda de duração limitada que possui valor médio zero. Se compararmos ondaletas com ondas senoidais, que são a base da análise de Fourier, as senóides não têm duração limitada, elas se estendem de menos infinito a mais infinito. Enquanto senóides são suaves e de fácil predição, ondaletas tendem a ser irregulares e assimétricas, conforme mostrado na figura I.4.

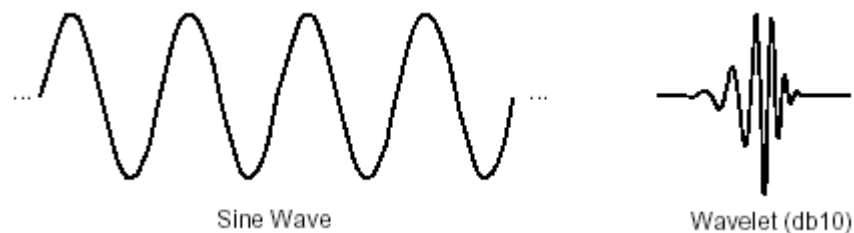


Figura I.4 - Formas de ondas básicas da análise de Fourier e por ondaleta (*wavelet*).

A análise de Fourier consiste em representar um sinal através de ondas senoidais de várias frequências. Similarmente, a análise por ondaleta consiste em dividir um sinal em versões deslocadas e escaladas da ondaleta original (ou mãe).

As ondaletas podem ser aplicadas a sinais de quaisquer dimensões, por exemplo, para aplicações de tratamento de imagem são utilizadas duas dimensões.

Existem basicamente dois tipos de transformadas por ondaleta: a transformada por ondaleta contínua e a transformada por ondaleta discreta.

I.1 Transformada por ondaletas contínuas

Matematicamente, o processo da análise por Fourier é representado pela transformada de Fourier dada por

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt \quad (\text{I.1})$$

que é a soma do sinal $f(t)$ ao longo do tempo multiplicado por uma exponencial complexa. Deve-se lembrar que uma exponencial complexa pode ser quebrada em componentes senoidal real e imaginária.

O resultado da transformada são os coeficientes de Fourier $F(\omega)$, que quando multiplicados por uma senóide de frequência ω , produzem as componentes constituintes do sinal original. Graficamente, o processo se assemelha ao mostrado na figura I.5.

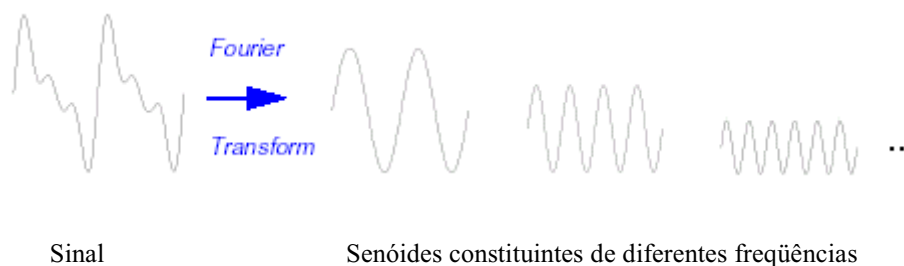


Figura I.5 - Sinal original e transformado por Fourier.

Similarmente, a transformada por ondaleta contínua (CWR - *Continuous Wavelet Transform*) é definida como a soma ao longo do tempo do sinal multiplicado por versões escaladas e deslocadas da função ondaleta Ψ , dada por

$$C(\text{escala}, \text{posição}) = \int_{-\infty}^{\infty} f(t)\Psi(\text{escala}, \text{posição}, t)dt \quad (\text{I.2})$$

Os resultados da CVT são muitos coeficientes de ondaleta C , que são uma função da escala e da posição.

Multiplicando cada coeficiente pela apropriada ondaleta escalada e deslocada produzimos as ondaletas constituintes do sinal original, conforme mostrado na figura I.6.

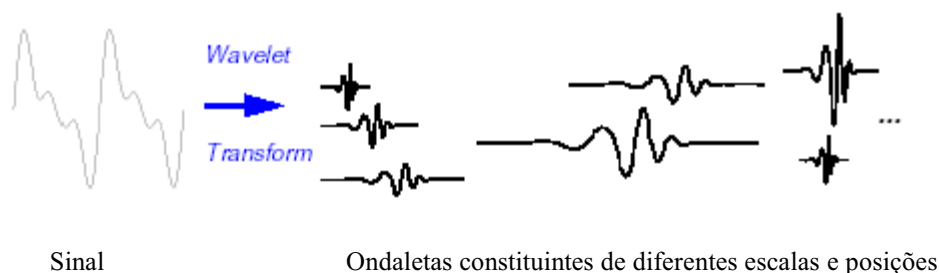


Figura I.6 - Sinal original e transformado por ondaletas.

I.1.1 Escalonamento

Escalonar uma ondaleta significa simplesmente esticá-la (ou comprimi-la) ela. Indo além da descrição coloquial de “esticar”, introduzimos o *fator de escala*, freqüentemente denotado pela letra a . Se falamos, por exemplo, de senóides, o efeito do fator de escalar é fácil de ver, conforme mostrado na figura I.7.

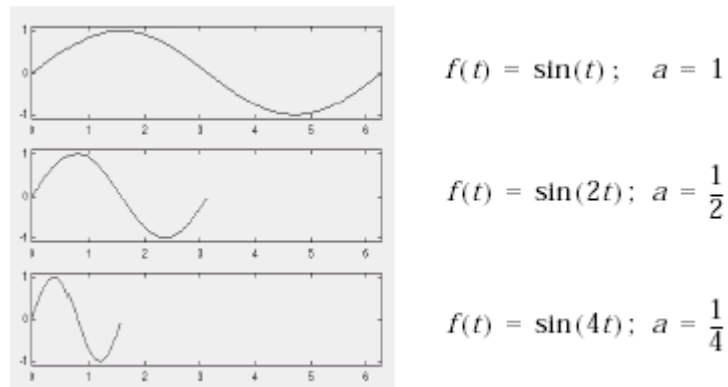


Figura I.7 - Fatores de escala para uma onda sonoidal.

O fator de escala funciona exatamente do mesmo modo com as ondaletas. Quanto menor for o fator de escala mais “comprimida” será a ondaleta, conforme mostrado na figura I.8.

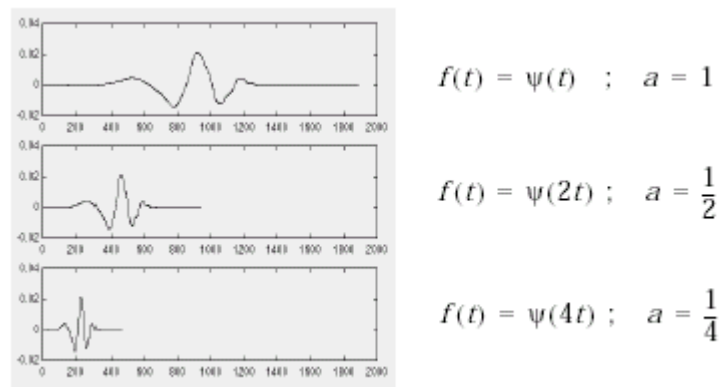


Figura I.8 - Fatores de escala para uma ondaleta.

I.1.2 Deslocamento

Deslocar uma ondaleta significa simplesmente atrasá-la (ou adiantá-la) da sua posição. Matematicamente, atrasar uma função $f(t)$ por k é representada por $f(t-k)$, conforme mostrado na figura I.9.

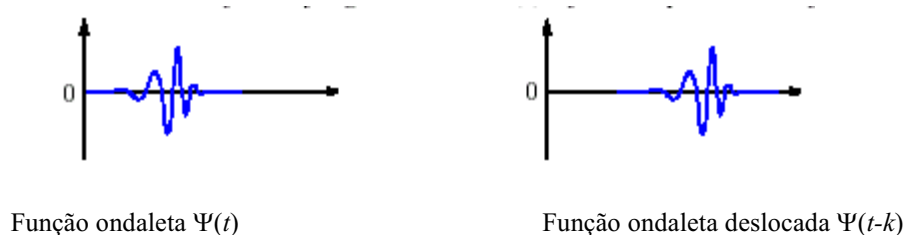


Figura I.9 - Deslocamento para uma ondaleta.

I.1.3 Criando uma transformada por ondaleta contínua

O processo para criar uma ondaleta contínua é composto de cinco passos:

1. Selecione uma ondaleta e compare-a com uma parte no início do sinal original;
2. Calcule o número C , que representa o quanto está correlacionada esta ondaleta com esta seção do sinal. Quanto mais alto for C mais similaridade existe. Mais precisamente, se a energia do sinal e a energia da ondaleta são iguais a um, C pode ser interpretado como um coeficiente de correlação;

Note que o resultado obtido irá depender da forma da ondaleta escolhida, conforme mostrado na figura I.10.

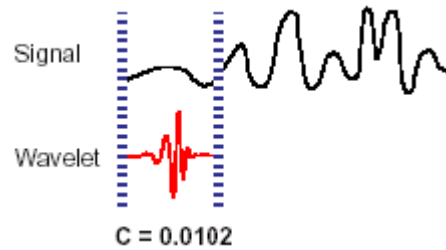


Figura I.10 - Coeficiente C obtido com a aplicação de uma ondaleta em um sinal.

- Desloque a ondaleta para a direita e repita os passos 1 e 2 até que se percorra todo o sinal, conforme mostrado na figura I.11;

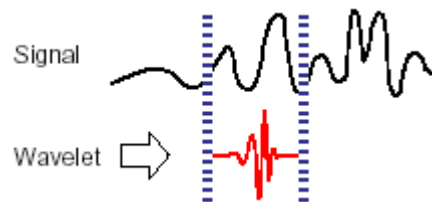


Figura I.11 - Próximo passo na obtenção de C na aplicação de uma ondaleta em um sinal.

- Escalone (“estique”) a ondaleta e repita os passos 1 até 3, conforme mostrado na figura I.12;

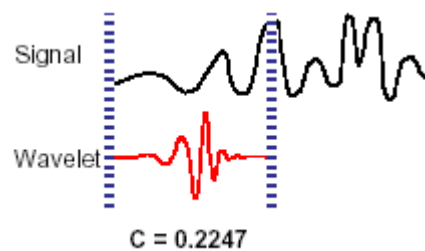


Figura I.12 - Obtenção de C para a aplicação de uma ondaleta “esticada” em um sinal.

- Repita os passos 1 até 4 para todas as escalas.

Quando o processo estiver terminado, teremos os coeficientes gerados para as várias escalas nas diversas seções do sinal. Os coeficientes constituem o resultado de uma regressão do sinal original realizado pelas ondaletas.

A transformada por ondaleta contínua pode operar em cada escala, desde a do sinal original até uma escala máxima que é definida pela necessidade da aplicação. Ela também é contínua em termos dos deslocamentos sobre o domínio da função analisada, conforme mostrado na figura I.13.

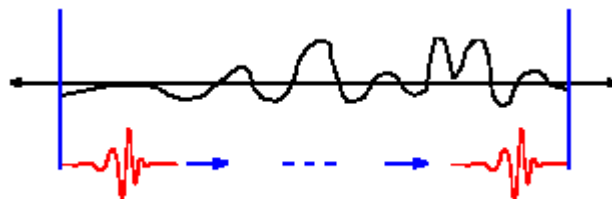


Figura I.13 - Deslocamento de uma ondaleta (*wavelet*) sobre um sinal.

I.2 Transformada por ondaletas discretas

Calcular os coeficientes da ondaleta a cada escala possível é muito trabalhoso e gera uma quantidade enorme de dados inúteis. Se em vez disso escolhermos escalas e posições baseadas na potência de dois - as chamadas escalas e posições diádicas - então nossa análise será mais eficiente e tão precisa quanto usar todas as escalas. Deste modo, obtemos a transformada por ondaleta discreta (DWT - *Discrete Wavelet Transform*).

Uma maneira eficiente de implementar este esquema usando filtros foi desenvolvida por MALLAT (1989). Este algoritmo de filtragem produz uma transformada por ondaletas rápida. Vamos examiná-lo mais detalhadamente a seguir.

I.2.1 Filtro de um estágio: aproximação e detalhes

Para muitos sinais, o conteúdo de frequências baixas é a parte mais importante. É ela que dá ao sinal sua identidade. Por outro lado, o conteúdo de frequências altas fornece as nuances. Considere a voz humana, se removermos as componentes de alta frequência, a voz soa diferente, mas podemos entender o que está sendo dito. Contudo, se removermos suficiente componentes de baixa frequência, escutaremos grunidos.

Na análise por ondaleta, frequentemente falamos em aproximações e detalhes. As aproximações contêm as componentes de escalas grandes, correspondentes as frequências baixas do sinal. Por outro lado, os detalhes contêm as componentes de escalas pequenas, correspondentes as frequências altas do sinal.

O processo de filtragem, ao nível mais básico, assemelha-se conforme mostrado na figura I.14.

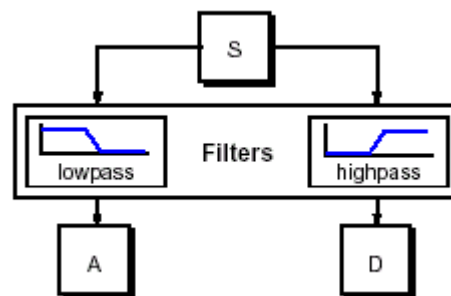


Figura I.14 - Processo de filtragem básico por ondaletas discretas.

O sinal original S é passado através de dois filtros complementares e emerge como dois sinais. Infelizmente, se efetuarmos realmente esta operação em um sinal digital real, obteremos duas vezes mais dados do que o sinal original. Suponha, por exemplo, que o sinal original S contenha 1000 amostras de dados. Então, os sinais resultantes terão 1000 amostras cada, dando um total de 2000 amostras.

Esses sinais A e D são interessantes, mas obtivemos 2000 valores em vez dos 1000 anteriores. Existe uma maneira mais refinada para realizar a decomposição usando ondaletas. Controlando mais cuidadosamente as computações realizadas pelo processo podemos manter somente uma em cada duas das 2000 amostras para manter toda a informação. Isto é conhecido como redução do número de amostras (*downsampling*) de um sinal. Deste modo, produzimos duas seqüências chamadas de cA e cD, conforme mostrado na figura I.15.

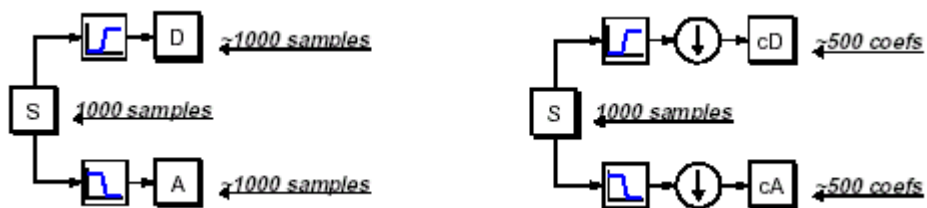


Figura I.15 - Redução do número de amostras de um sinal.

O processo à direita, que inclui a redução do número de amostras, produz os coeficientes DWT.

Para entender melhor este processo vamos realizar a transformada por ondaleta discreta de um estágio em um sinal. Nosso sinal será uma senóide pura com ruído de alta frequência adicionado a ela.

A figura I.16 apresenta o diagrama esquemático desse processo.

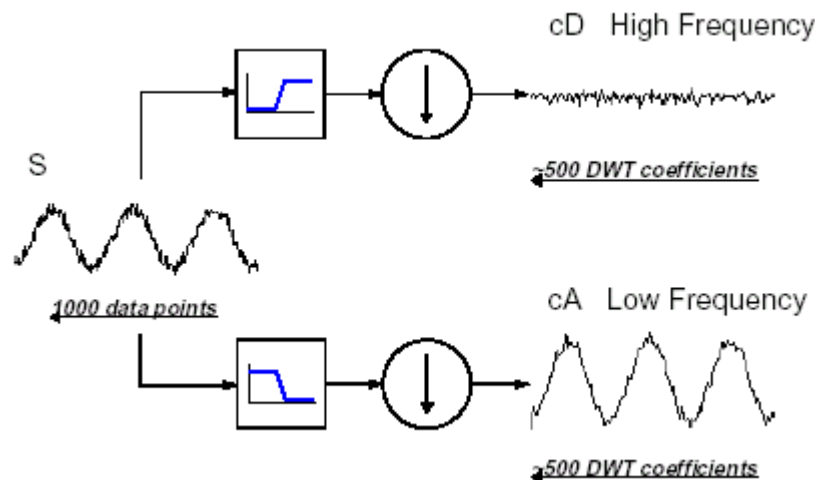


Figura I.16 - Diagrama esquemático do processo de redução do número de amostras.

Podemos observar que os coeficientes de detalhe cD são pequenos e consistem principalmente do ruído de alta frequência, enquanto que os coeficientes de aproximação cA contêm muito menos ruído do que o sinal original. Podemos observar ainda que o tamanho real dos vetores que contêm os coeficientes de aproximação e detalhe são um pouco maiores do que a metade do sinal original. Isto tem a ver com o processo de filtragem, que é implementado pela convolução do sinal com o filtro. O processo de convolução “embota” o sinal, introduzindo várias amostras extras no resultado.

I.2.2 Decomposição multi-nível

O processo de decomposição pode ser iterativo, com sucessivas aproximações sendo executadas de cada vez, de tal maneira que um sinal é dividido em muitas componentes de menor resolução. Isto é chamado de árvore de decomposição da ondaleta, conforme mostrado na figura I.17.

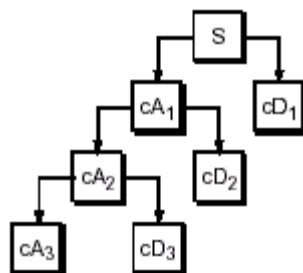


Figura I.17 - Processo sucessivo de decomposição dos coeficientes de aproximação.

Olhar um sinal da árvore de decomposição por uma ondaleta pode fornecer informação valiosa, conforme mostrado na figura I.18.

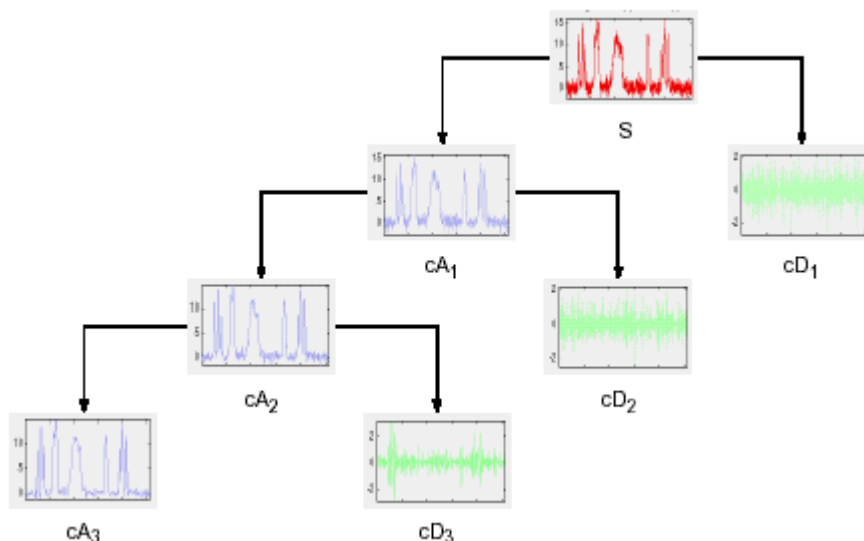


Figura I.18 - Sinais da árvore de decomposição de um sinal.

Tendo em vista que o processo é iterativo, em teoria podemos continuar a redução do número de amostras indefinidamente. Na realidade, o processo de decomposição pode proceder até que se tenha uma única amostra ou pixel. Na prática, selecionamos um número de níveis satisfatórios baseado na natureza do sinal.

I.3 Reconstrução da ondaleta

Vimos como a transformada discreta por ondaleta pode ser usada para analisar, ou decompor, sinais e imagens. Este processo é chamado de decomposição ou análise. O outro lado dessa história é como essas componentes podem ser rearrumadas para se voltar ao sinal original. Este processo é chamado de reconstrução ou síntese. A manipulação matemática que efetua a síntese é chamada de transformada inversa por ondaleta discreta (IDWT - *Inverse Discrete Wavelet Transform*).

Para sintetizar um sinal nós o reconstruímos a partir dos coeficientes da ondaleta, conforme mostrado na figura I.19.

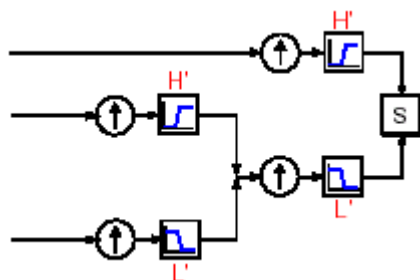


Figura I.19 - Reconstrução de um sinal a partir de seus coeficientes de ondaleta.

Enquanto a análise envolve filtragem e redução do número de amostras, o processo de reconstrução por ondaleta consiste de aumento do número de amostras (*upsampling*) e filtragem. O aumento do número de amostras é um processo de aumentar as componentes de um sinal pela inserção de zeros entre amostras, conforme mostrado na figura I.20.

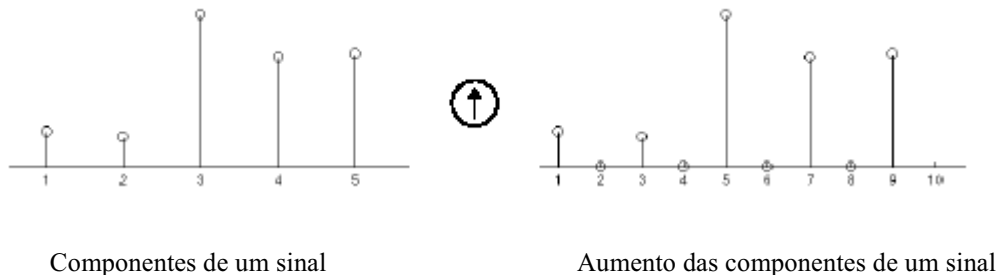


Figura I.20 - Aumento das componentes de um sinal.

I.3.1 Filtros de reconstrução

A escolha do filtro de reconstrução é crucial para se alcançar uma reconstrução perfeita do sinal original. A redução do número de amostras componentes do sinal realizada durante a fase de decomposição introduz uma distorção chamada de *aliasing*. Podemos cancelar o efeito do *aliasing* pela escolha cuidadosa dos filtros para as fases de decomposição e reconstrução de modo que eles sejam relacionados (mas não idênticos).

Os filtros passa baixa e passa alta de decomposição (L e H) juntamente com os seus associados filtros de reconstrução (L' e H') formam um sistema chamado de filtros espelhos em quadratura, conforme mostrado na figura I.21.

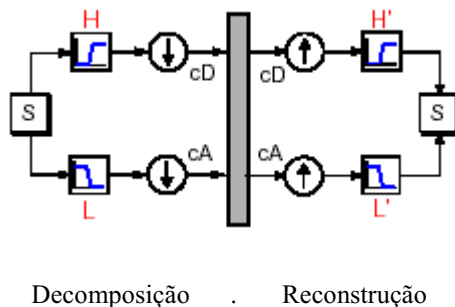


Figura I.21 - Filtros de decomposição e reconstrução em quadratura.

I.3.2 Reconstruindo aproximações e detalhes

Vimos que é possível reconstruir nosso sinal original a partir dos coeficientes de aproximações e detalhes, conforme mostrado na figura I.22.

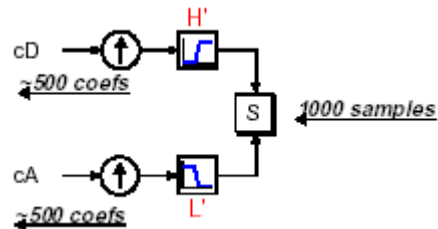


Figura I.22 - Reconstrução a partir dos coeficientes de aproximação e detalhe.

Também é possível reconstruir as aproximações e detalhes a partir de seus vetores de coeficientes. Como exemplo, vamos considerar que desejamos reconstruir o primeiro nível de aproximação $A1$ a partir dos vetores coeficientes $cA1$.

Para tal passamos o vetor de coeficientes $cA1$ através do mesmo processo usado para reconstruir o sinal original. Contudo, em vez de combiná-lo com o detalhe do nível um, $cD1$, nós alimentamos um vetor de coeficientes zero no lugar do vetor de coeficientes de detalhe, conforme mostrado na figura I.23.

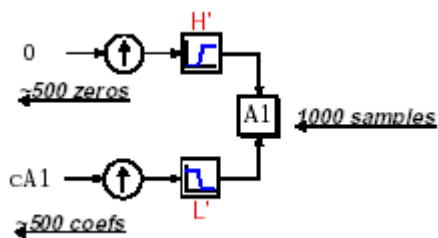


Figura I.23 - Reconstrução a partir dos coeficientes de aproximação e de zeros.

O processo produz uma reconstrução da aproximação $A1$, que tem o mesmo tamanho do sinal original S e que é uma aproximação real dele.

Similarmente, podemos reconstruir o primeiro nível de detalhe, usando um processo análogo, conforme mostrado na figura I.24.

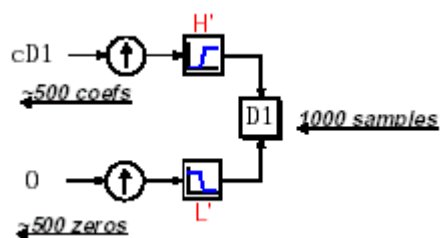


Figura I.24 - Reconstrução a partir dos coeficientes de detalhe e de zeros.

As aproximações e os detalhes reconstruídos são os constituintes verdadeiros do sinal original. De fato, podemos combiná-los do seguinte modo

$$A_1 + D_1 = S \tag{I.3}$$

A figura I.25 apresenta a extensão desta técnica para análise das componentes multinível. Observa-se que relações similares podem ser usadas para reconstruir o sinal original.

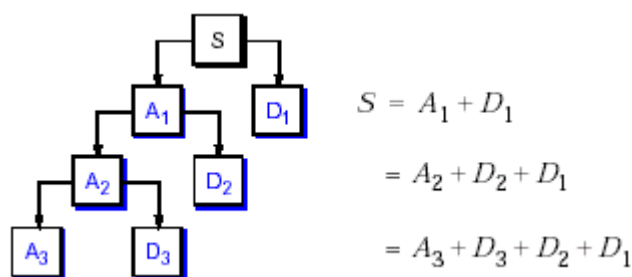


Figura I.25 - Componentes de reconstrução do sinal.

I.3.3 Decomposição e reconstrução multietapas

Um processo multietapas de análise e síntese pode ser representado conforme mostrado na figura I.26.

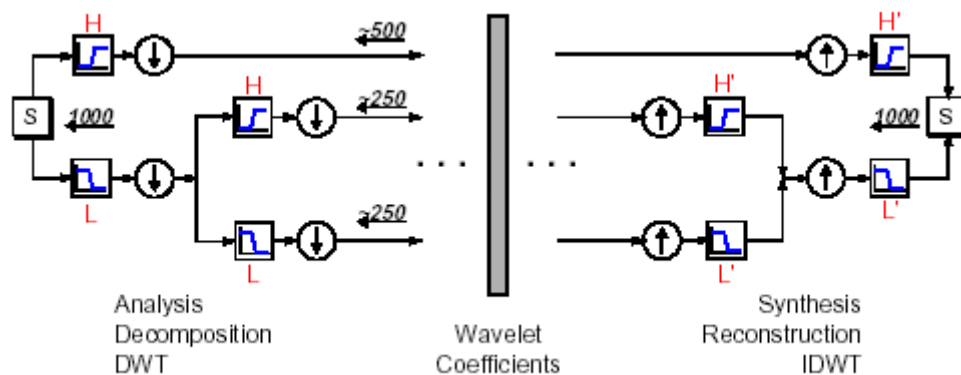


Figura I.26 - Processo de decomposição e reconstrução multietapas.

Este processo envolve dois aspectos: dividir um sinal para obter os coeficientes da ondaleta e rearrumar o sinal a partir dos coeficientes.

I.4 Tipos de ondaletas

Existem vários tipos ou famílias de ondaletas. Algumas delas são apresentadas a seguir.

I.4.1 Ondaleta Haar

Foi a primeira e é a mais simples ondaleta construída. Sua forma lembra uma função degrau, conforme mostrado na figura I.27.

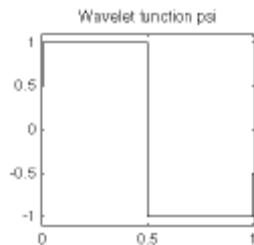


Figura I.27 - Ondaleta tipo Haar.

I.4.2 Ondaleta Daubechies

As ondaletas inventadas por Daubechies tornaram a análise de ondaleta discreta factível. Elas tem a característica de serem ondaletas ortonormais. A figura I.28 apresenta as várias formas das ondaletas Daubechies. A ondaleta db1 é igual a ondaleta Haar.

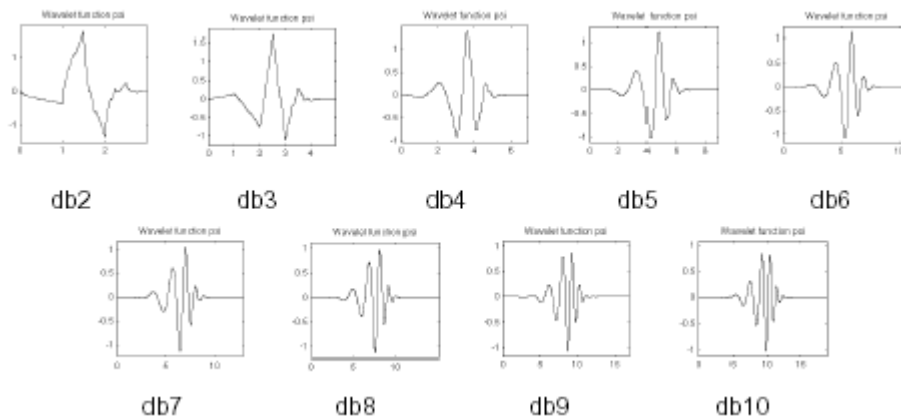


Figura I.28 - Várias formas das ondaletas Daubechies.

I.4.3 Ondaleta Biortogonal

Esta família de ondaletas tem a propriedade de fase linear, que é necessária para a reconstrução de sinais e imagens. A figura I.29 apresenta as várias formas dessas ondaletas.

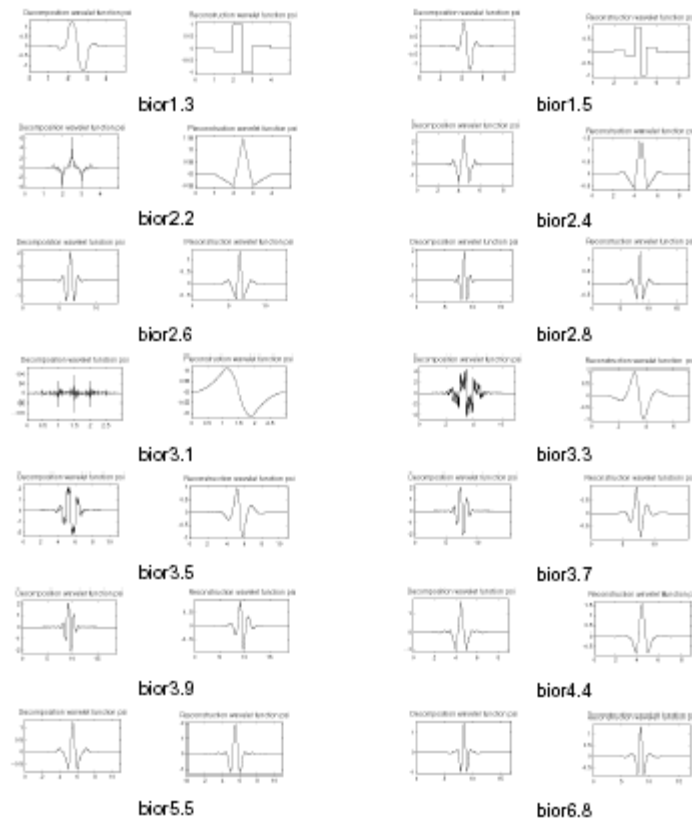


Figura I.29 - Várias formas das ondaletas Biortogonais.

I.4.4 Ondaleta Coiflets

A figura I.30 apresenta as várias formas dessas ondaletas.



Figura I.30 - Várias formas das ondaletas Coiflets.

I.4.5 Ondaleta Symlets

As symlets são ondaletas quase simétricas. A figura I.31 apresenta as várias formas dessas ondaletas.

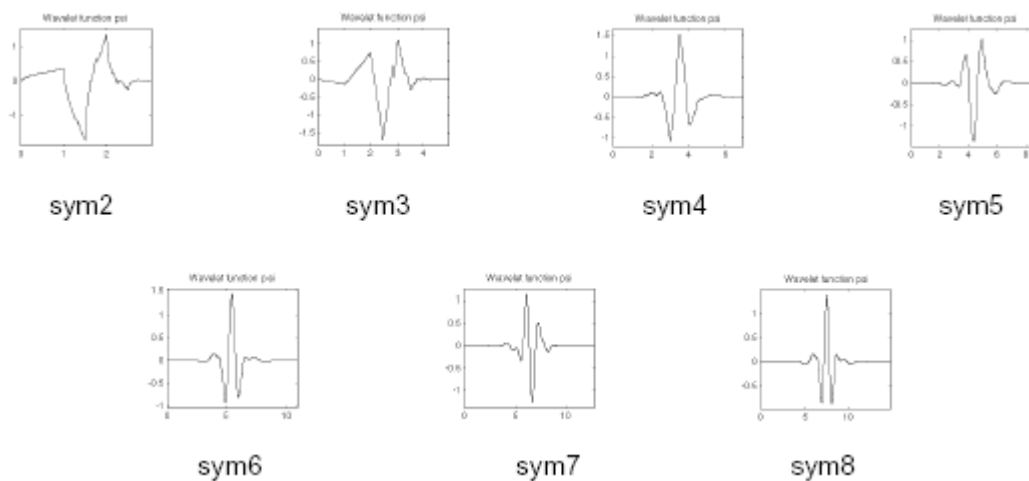


Figura I.31 - Várias formas das ondaletas Symlets.

I.4.6 Outras formas de ondaletas

A figura I.32 apresenta outras formas de ondaletas.

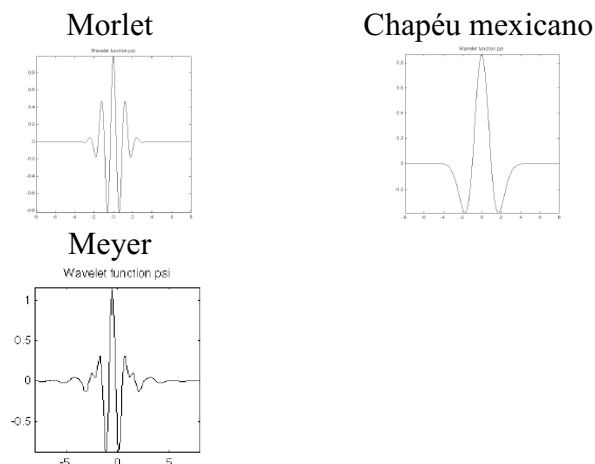


Figura I.32 - Outras formas de ondaletas.

ANEXO II

Análise das Componentes Principais

A análise das componentes principais (PCA - *Principal Components Analysis*) é um método de pré-processamento que retira de um conjunto de dados as componentes não correlacionadas. O método PCA envolve uma transformação linear do espaço de entrada em um espaço ortogonal que pode ser escolhido para ter dimensão menor do que o espaço de entrada, com mínima perda de informação.

O método PCA é realizado pelos seguintes passos:

1. Selecione os dados que serão utilizados no PCA;
2. Subtraia, em cada dimensão dos dados, o valor do dado da média dos valores dos dados na mesma dimensão. Isto produz um conjunto de dados com média zero;
3. Calcule a matriz covariância do conjunto de dados;
4. Calcule os autovetores e os autovalores da matriz covariância;
5. Selecione as componentes a serem utilizadas e construa um vetor de realizações;
6. Obtenha o novo conjunto de dados.

Para ilustrar o processamento realizado pelo PCA apresentamos a seguir uma aplicação deste método no conjunto de dados que foram utilizados na aplicação do sistema de validação de sinal proposto nesta tese.

O conjunto de dados é composto por 1551 amostras de dados de 12 variáveis de uma planta nuclear de potência. A tabela II.1 apresenta a matriz de correlação linear entre as 12 variáveis do processo. O coeficiente de correlação linear entre duas variáveis é obtido pela seguinte expressão

Tabela II.1

Matriz dos coeficientes de correlação linear entre as variáveis da planta nuclear.

	PN	TF	TQ	TM	TV	PP	NP	NE	NL	PG	VA	VP
Potência nuclear (PN)	1.0000	-0.8278	0.9973	0.9790	-0.6114	0.1204	0.8661	0.9485	0.9595	-0.9593	0.9981	0.9996
Temperatura da perna fria (TF)	-0.8278	1.0000	-0.7891	-0.6999	0.3179	-0.1753	-0.5422	-0.7607	-0.7631	0.8681	-0.8319	-0.8370
Temperatura da perna quente (TQ)	0.9973	-0.7891	1.0000	0.9903	-0.6336	0.1075	0.8832	0.9488	0.9593	-0.9436	0.9943	0.9958
Temperatura média do núcleo (TM)	0.9790	-0.6999	0.9903	1.0000	-0.6696	0.1059	0.9091	0.9387	0.9502	-0.9060	0.9744	0.9749
Taxa de vazão do núcleo (TV)	-0.6114	0.3179	-0.6336	-0.6696	1.0000	-0.0340	-0.6267	-0.6057	-0.6157	0.5284	-0.6052	-0.6046
Pressão do pressurizador (PP)	0.1204	-0.1753	0.1075	0.1059	-0.034	1.0000	0.0668	0.1328	0.1206	-0.1220	0.1121	0.1183
Nível do pressurizador (NP)	0.8661	-0.5422	0.8832	0.9091	-0.6267	0.0668	1.0000	0.8455	0.8527	-0.8144	0.8620	0.8584
Nível do gerador de vapor - f.e. (NE)	0.9485	-0.7607	0.9488	0.9387	-0.6057	0.1328	0.8455	1.0000	0.9744	-0.9109	0.9473	0.9476
Nível do gerador de vapor - f.l. (NL)	0.9595	-0.7631	0.9593	0.9502	-0.6157	0.1206	0.8527	0.9744	1.0000	-0.9069	0.9598	0.9594
Pressão do gerador de vapor (PG)	-0.9593	0.8681	-0.9436	-0.9060	0.5284	-0.1220	-0.8144	-0.9109	-0.9069	1.0000	-0.9648	-0.9623
Vazão da água de alimentação (VA)	0.9981	-0.8319	0.9943	0.9744	-0.6052	0.1121	0.8620	0.9473	0.9598	-0.9648	1.0000	0.9982
Vazão de vapor (VP)	0.9996	-0.8370	0.9958	0.9749	-0.6046	0.1183	0.8584	0.9476	0.9594	-0.9623	0.9982	1.0000

$$S(i,j) = \frac{C(i,j)}{\sqrt{C(i,i)C(j,j)}} \quad (\text{II.1})$$

onde $S(i,j)$ é o coeficiente de correlação linear entre as variáveis i e j e $C(i,j)$ é o valor da covariância entre as variáveis i e j .

O coeficiente de correlação varia entre $[-1, 1]$. Valores de correlação próximos à -1 ou 1 significam que existe uma grande correlação linear entre as variáveis. Valores próximos à zero significam que a correlação é fraca entre as variáveis.

A covariância entre duas variáveis X e Y é dada pela seguinte expressão

$$C(X,Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{n-1} \quad (\text{II.2})$$

onde X_i e Y_i são, respectivamente, os valores das variáveis X e Y no instante i , \bar{X} e \bar{Y} são, respectivamente, os valores médios das variáveis X e Y e n é o número de amostras do conjunto de dados.

A tabela II.2 apresenta a quantidade de informação de cada componente obtida pelo processo de geração das componentes principais para cada variável do processo considerando as outras 11 variáveis como variáveis de entrada. Podemos observar que quase 99,9% de toda informação do conjunto das 11 componentes principais está contida em 8 componentes.

Se considerássemos como satisfatório fazer uma redução na dimensão das variáveis de entrada do sistema que mantivesse 98% da informação inicial, obteríamos, pela tabela II.2, um número de 4 componentes principais para serem utilizadas no processo de compressão. Neste caso, o processo de redução da dimensão por componentes principais é representado na figura II.1.

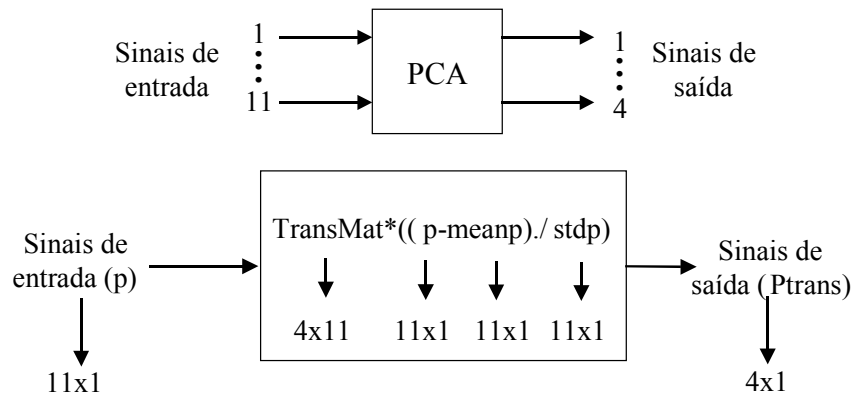


Figura II.1 - Exemplo de processo de redução de dimensão por componentes principais.

Na parte superior da figura os sinais estão representados de forma esquemática e na parte inferior os sinais estão representados de forma vetorial, onde p é o vetor que contém a amostra dos sinais de entrada para o método, **TransMat** é a matriz de transformação obtida pelo método de componentes principais, **meanp** é o vetor que contém as médias de cada sinal e **stdp** é o vetor que contém os desvios-padrão dos sinais de entrada.

Por outro lado, para recuperarmos o sinal original a partir das componentes principais de um conjunto de variáveis devemos realizar o processo inverso de redução de dimensão. Este processo é conhecido como expansão do espaço e para realizá-lo devemos efetuar a transformação inversa do processo de componentes principais, conforme mostrado na figura II.2.

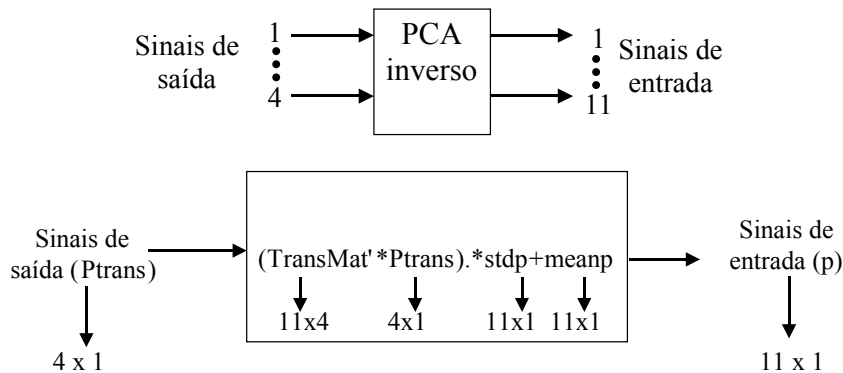


Figura II.2 - Exemplo do processo de recuperação dos sinais de entrada.

Novamente, na parte superior da figura os sinais estão representados de forma esquemática e na parte inferior os sinais estão representados de forma vetorial, onde **TransMat'** é a transposta da matriz de transformação obtida pelo método de componentes principais.

A figura II.3 apresenta a comparação dos valores das variáveis de entrada originais e recuperadas após o processo de compressão do PCA para as 11 variáveis da planta nuclear de potência (sem a variável potência nuclear), considerando a preservação de 98% da informação nos dados de entrada, ou seja, redução para 4 componentes principais.

A figura II.4 apresenta a mesma comparação considerando a preservação de 99,9% da informação nos dados de entrada, ou seja, redução para 8 componentes principais. Para este caso, só apresentamos na figura os gráficos dos resultados obtidos pelo PCA versus o valor original de cada variável.

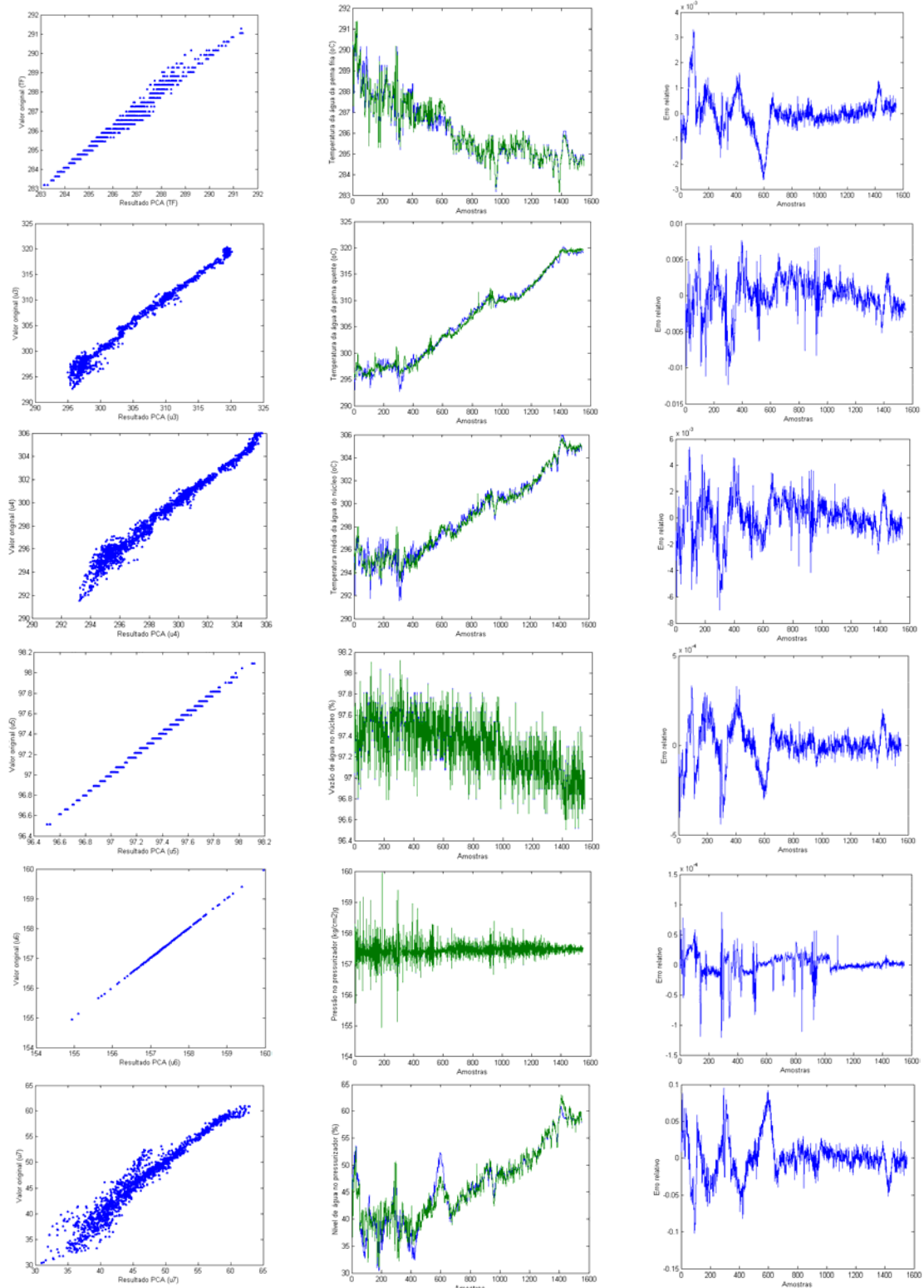


Figura II.3 - Comparação dos valores das variáveis de entrada originais e recuperadas após o processo de compressão do PCA para as 11 variáveis da planta nuclear.

(Continuação)

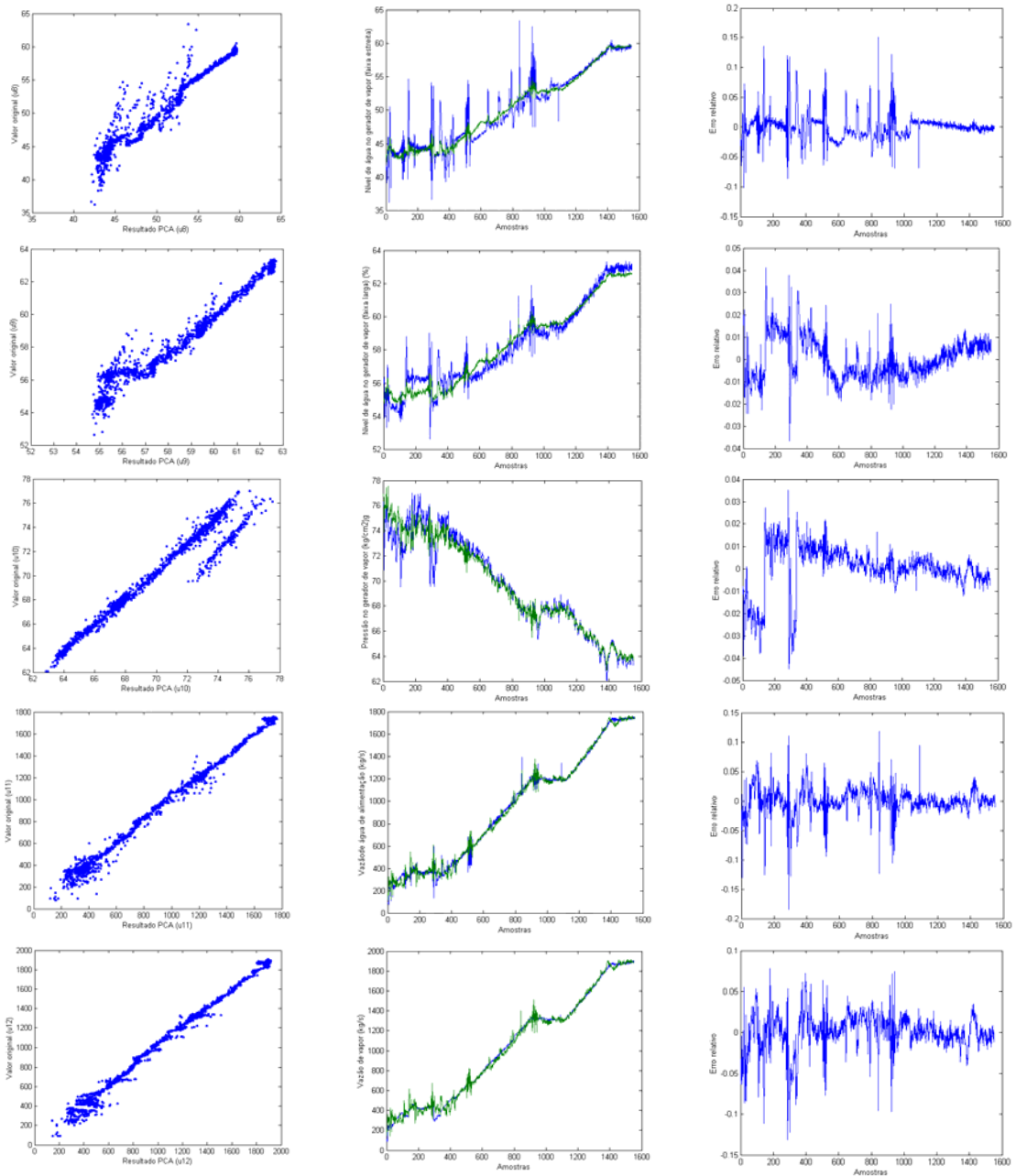


Figura II.3 - Comparação dos valores das variáveis de entrada originais e recuperadas após o processo de compressão do PCA para as 11 variáveis da planta nuclear.

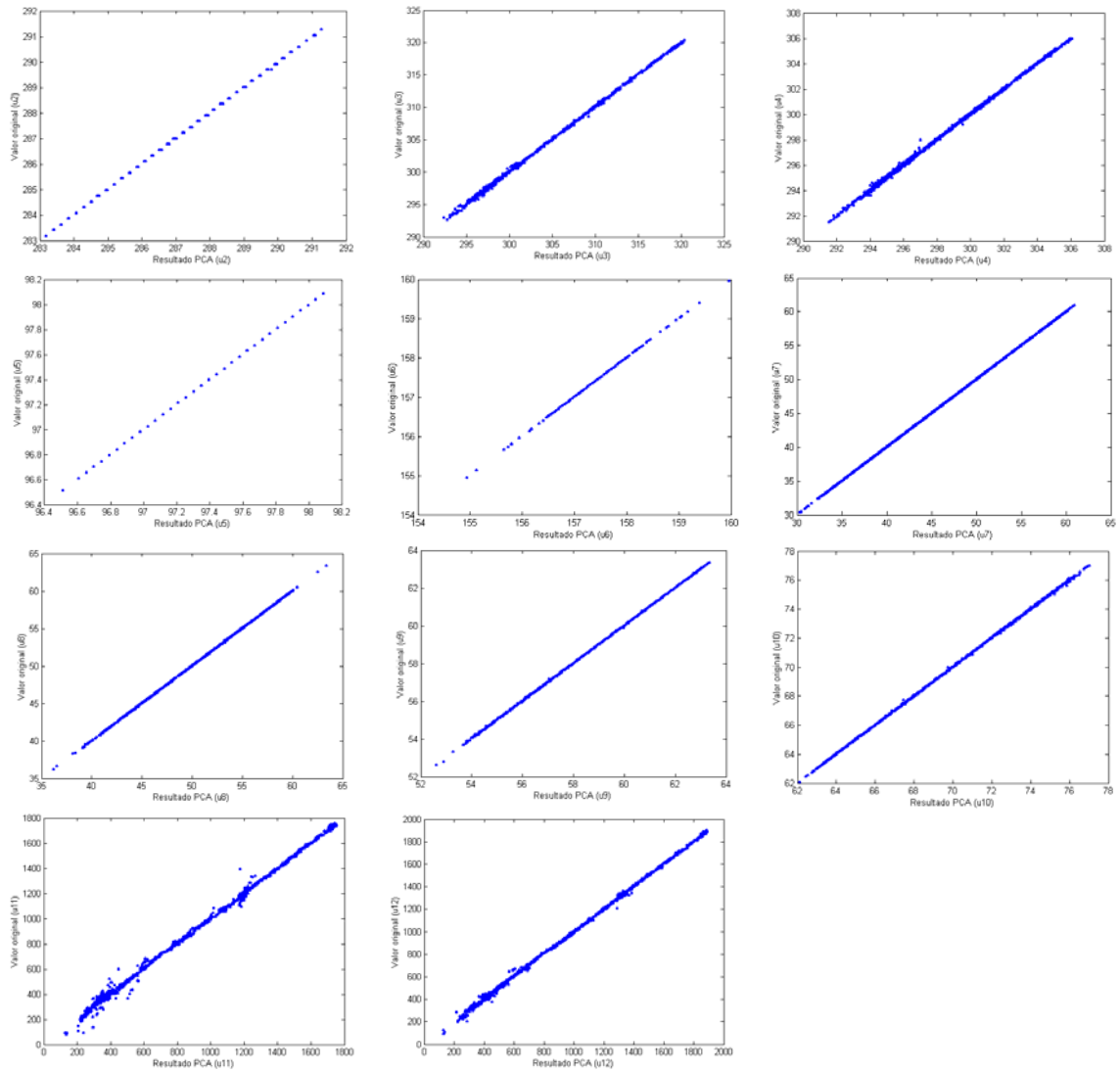


Figura II.4 - Comparação dos valores das variáveis de entrada originais e recuperadas após o processo de compressão do PCA para as 11 variáveis da planta nuclear.

ANEXO III

Técnicas de otimização

A seguir são descritas resumidamente as técnicas e os algoritmos de otimização por mínimos quadrados, gradiente descendente, algoritmos genéticos e enxame de partículas.

III.1 Otimização por mínimos quadrados

Em muitos experimentos buscamos uma equação que relacione uma variável X com uma variável Y . Experimentalmente medimos Y para diferentes valores de X , desenhamos os resultados das medições e, então, tentamos traçar uma curva através desses pontos. Este processo é chamado de regressão de Y em X e pode ser usado para prever o valor de Y para um dado valor de X . A seguir é dado o formalismo matemático usado para encontrar a melhor curva que representa os dados.

O critério dos mínimos quadrados para ajustar uma curva é baseado na idéia que nós queremos minimizar a diferença entre os dados e a curva de predição. Esta diferença é encontrada comparando cada ponto do conjunto de dados Y_i com seu valor predição \hat{Y}_i . Não podemos simplesmente adicionar estas diferenças, tendo em vista que os erros positivos e negativos se cancelaram. O que se faz na prática é adicionar os quadrados dessas diferenças e minimizar a soma final.

A equação geral do método dos mínimos quadrados é dada por

$$S \equiv \sum_{i=1}^N w_i (Y_i - \hat{Y}_i)^2 \quad (\text{III.1})$$

onde w_i é o peso do i -ésimo resíduo. Por exemplo, coeficientes de peso diferentes podem ser utilizados se as medições são feitas com confiabilidades diferentes. Por simplicidade consideraremos $w_i = 1$ nos exemplos a seguir.

O método dos mínimos quadrados é um conjunto de regras para minimizar S . Uma função Qui-quadrada é definida por

$$\chi^2 \equiv \frac{S}{\sigma_s^2} \quad (\text{III.2})$$

onde σ_s é o desvio padrão das amostras observadas.

Se considerarmos σ_s constante, então, χ^2 é mínimo quando S for mínimo. Deste modo, o método dos mínimos quadrados minimiza S e χ^2 . O método dos mínimos quadrados também minimiza o desvio padrão externo, isto é, a incerteza nos parâmetros de ajuste.

III.1.1 Mínimos quadrados ajustado por uma única variável

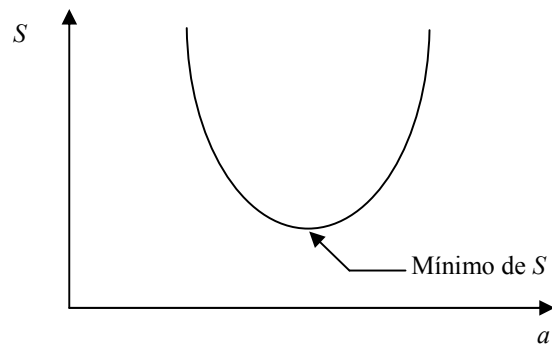
Vamos avaliar o método dos mínimos quadrados considerando somente uma única variável a sendo variada. Neste caso a incerteza da variável a é dada por σ_a .

O valor mínimo de S será encontrado variando-se a e verificando S para cada valor de a . Este processo é apresentado na figura III.1.

Nesse caso teremos

$$S \equiv \sum_{i=1}^N w_i (x_i - a)^2 \quad (\text{III.3})$$

onde consideraremos w_i como sendo igual a 1.

Figura III.1 - Variação de S com a .

Para calcular o valor de a que minimiza S será necessário calcular $dS/da = 0$, a partir de III.2, dado por

$$\frac{dS}{da} = \frac{d}{da} \left(\sum_{i=1}^N (x_i - a)^2 \right) = -2 \sum_{i=1}^N (x_i - a) = 0 \quad (\text{no mínimo}) \quad (\text{III.4})$$

Resolvendo a equação obtemos

$$\sum_{i=1}^N (x_i - a) = \sum_{i=1}^N x_i - Na = 0 \quad \therefore \quad a = \frac{1}{N} \sum_{i=1}^N x_i = \bar{x} \quad (\text{III.5})$$

Então, o valor médio quadrático de a é \bar{x} (valor médio de x) e $a = \bar{x}$ fornece o menor valor possível da soma dos quadrados dos resíduos (χ^2) e do desvio padrão externo.

O desvio padrão das observações σ_s é computado como

$$\sigma_s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - a)^2} \quad (\text{III.6})$$

E o desvio padrão da média como

$$\sigma_a = \frac{\sigma_s}{\sqrt{N}} \quad (\text{III.7})$$

III.1.2 Mínimos quadrados ajustado por uma reta

Considere um conjunto de dados com N valores de pares ordenados (x,y) e que somente a coordenada Y está sujeita a incertezas, conforme mostrado na figura III.2.

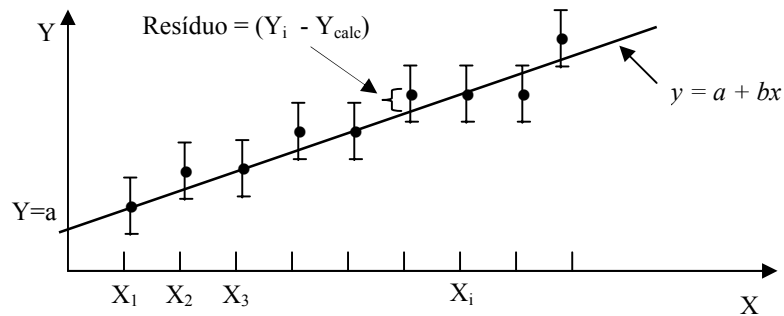


Figura III.2 - Conjunto de dados, resíduos e equação de ajuste.

Considere que a equação para ajustar esses pontos é $y = a + bx$. Seja y_i a ordenada observada no i -ésimo ponto, então, o resíduo desse ponto é dado por

$$\text{resíduo} = y_{\text{observado}} - y_{\text{calculado}} = y_i - (a + bx_i) \quad (\text{III.8})$$

onde a e b são obtidos da curva de ajuste usando o método dos mínimos quadrados.

Este resíduo é um resíduo em y porque a incerteza nos nossos dados foi assumida estar toda em y . Então, a soma a ser minimizada é dada por

$$S \equiv \sum_{i=1}^N w_i (y_i - a - bx_i)^2 \quad (\text{III.9})$$

onde w_i é o peso, isto é, a importância dada a y_i .

Para esse exemplo consideraremos que todos os pesos w_i são iguais a unidade.

Se encontrarmos os valores de a e b que minimizam S , então, teremos o resultado dos mínimos quadrados para a melhor linha que descreve os dados.

Para encontrar o mínimo S nós diferenciamos S com respeito as parâmetros desconhecidos, neste caso a e b , e fazemos essas derivadas parciais iguais a zero.

$$\frac{\partial S}{\partial a} = -2 \sum_{i=1}^N (y_i - a - bx_i) \equiv 0 \quad (\text{III.10})$$

$$\frac{\partial S}{\partial b} = -2 \sum_{i=1}^N x_i (y_i - a - bx_i) \equiv 0 \quad (\text{III.11})$$

Resolvendo o conjunto de equações anteriores obtemos

$$a = \bar{y} - b\bar{x} \quad \text{e} \quad b = \frac{\sum_{i=1}^N x_i y_i - N\bar{x} \cdot \bar{y}}{\sum_{i=1}^N x_i^2 - N\bar{x}^2} = \frac{\sum_{i=1}^N x_i y_i - N\bar{x} \cdot \bar{y}}{m} \quad (\text{III.12})$$

$$\text{Onde } \bar{x} \equiv \frac{1}{N} \sum_{i=1}^N x_i, \quad \bar{y} \equiv \frac{1}{N} \sum_{i=1}^N y_i \quad \text{e} \quad m = \sum_{i=1}^N x_i^2 - N\bar{x}^2 \quad (\text{III.13})$$

Usando esses valores de a e b e aplicando operações algébricas obtemos o valor de χ^2 mínimo dado por

$$\chi^2 = \frac{S}{\sigma_y^2} = \frac{1}{\sigma_y^2} \left(\sum_{i=1}^N y_i^2 - N\bar{y}^2 - b^2 \cdot m \right) \quad (\text{III.14})$$

onde a incerteza da coordenada y do ponto σ_y é assumida ser a mesma em todos os pontos.

Pode-se mostrar que o desvio padrão em a e b são dados por

$$\sigma_a = \sigma_y \left(\frac{1}{N} + \frac{\bar{x}^2}{m} \right) \quad (\text{III.15})$$

$$\sigma_b = \sigma_y \left(\frac{1}{m} \right)^{\frac{1}{2}} \quad (\text{III.16})$$

O número de graus de liberdade é a diferença entre o número de pontos de dados (x_i, y_i) e o número de parâmetros a ser determinado. No caso da equação de uma reta existem dois parâmetros, as constantes a e b . O valor mínimo χ^2 é a soma dos quadrados dos desvios dos pontos de medida a partir da melhor reta de ajuste dividido pelo quadrado da incerteza estimada σ_y^2 .

III.2 Otimização pelo gradiente descendente

Gradiente descendente (GD) é uma técnica de otimização que basicamente move o vetor direção (ao longo do qual o ajuste dos parâmetros é feito) na direção negativa do vetor gradiente da função a ser otimizada. A técnica de gradiente descendente faz uma busca no espaço de discussão de modo a alcançar o valor mínimo da superfície de erro da função.

O algoritmo gradiente descendente (HAYKIN, 1994) é o método básico usado para treinar redes neurais retropropagadas.

Basicamente, o processo de treinamento por retropropagação envolve três estágios: aplicação às entradas dos padrões de treinamento, cálculo e retropropagação dos erros de saída associados, e ajuste dos parâmetros variáveis de modo a minimizar o erro de saída.

Uma estrutura tipo rede similar à uma rede neural pode ser usada para interpretar o mapeamento entrada/saída de um sistema de inferência neuro-nebulosa, que mapea

entradas através das funções pertinências e parâmetros associados e, então, através de funções pertinências de saída e parâmetros associados em saídas.

Os parâmetros associados com as funções-pertinência serão modificados através do processo de aprendizado. A computação desses parâmetros (ou seus ajustes) é facilitado pelo vetor gradiente, que fornece uma medida de quão bem o sistema de inferência nebulosa está modelando os pares entrada/saída para um dado valor dos parâmetros. Uma vez que o vetor gradiente é obtido, qualquer das diversas rotinas de otimização pode ser aplicada de modo a ajustar os parâmetros de maneira a reduzir a medida de erro. Neste caso, a medida de erro é dada pelo erro médio quadrático, isto é, o valor médio da raiz quadrada da soma dos quadrados da diferença entre os valores estimados e medidos da variável de saída do modelo.

Uma peculiaridade da superfície de erro que implica na performance do algoritmo de retropropagação é a presença de mínimos locais, isto é, vales isolados, em adição ao mínimo global. Tendo em vista que a retropropagação é basicamente uma técnica de subida de montanha, ela corre o risco de ficar presa a um mínimo local, onde pequenas mudanças nos valores dos parâmetros de otimização aumenta a função custo. Mas em algum lugar no espaço de otimização dos parâmetros existe um outro conjunto de valores dos parâmetros que a função-custo é menor do que o mínimo local em que o sistema de inferência neuro-nebulosa está preso.

Informações mais detalhadas sobre otimização por gradiente descendente podem ser encontrados em HAYKIN (1994).

III.3 Otimização por algoritmos genéticos

Os algoritmos genéticos (GAs) são uma classe de algoritmos de busca probabilística que emulam operadores genéticos naturais, e são capazes de localizar uma solução ótima ou próxima da ótima em uma larga escala de problemas. A técnica GA foi desenvolvida por HOLLAND (1975), inspirada no uso de conceitos tirados da genética e da teoria da evolução.

Um esquema geral do procedimento do GA é dado por

começo

$t = 0;$

inicializa $P(t)$

avalia as estruturas em $P(t)$

enquanto a condição de término não é satisfeita ***faça***

começo

$t = t + 1$

selecione $P(t)$ de $P(t-1)$

recombine as estruturas em $P(t)$

avalie as estruturas em $P(t)$

fim

fim

fim

onde $P(t)$ é a população das soluções candidatas da função objetivo no tempo t .

Para ilustrar a idéia básica do GA, considere $y=x^2$ uma função não negativa que desejamos maximizar sobre o conjunto dos inteiros entre 0 e 31, inclusive. Em vez de usar y diretamente vamos codificar os valores de y como uma seqüência binária. Para este exemplo, vamos codificar de acordo com a representação na base 2 usando 5 *bits* de tal modo que 00101 corresponda a 5. Primeiro começamos com uma geração de seqüências P_0 .

Depois vamos buscar reforçar um tipo de “sobrevivente por maior valor da função-objetivo” permitindo às seqüências binárias realizar “reproduções” a uma taxa proporcional à seus valores de função objetivo. Suponha que 00101 e 11010 sejam membros de P_0 que foram selecionados para reprodução. Esses indivíduos correspondem a valores de função objetivo de 5 e 26, respectivamente. Então um número aleatório entre 1 e 4 é gerado para decidir onde cortar a seqüência binária para o “cruzamento” genético. Por exemplo, se o número aleatório selecionado para este par específico é 2, então, o cruzamento ocorre entre o segundo e o terceiro *bits* dando os indivíduos 00010 e 11101. Nós permitimos ocorrer “mutações” (complementar o valor de um *bit*), com uma pequena probabilidade,. Por exemplo, o primeiro bit do primeiro indivíduo é trocado de 0 para 1 dando os indivíduos 10010 e 11101, que correspondem a valores de função objetivo de 18 e 29, respectivamente. O exemplo mostra um aumento no valor da função-objetivo de ambos os indivíduos. A idéia é repetir este processo altamente estruturado, ainda que aleatório, de busca pela criação de sucessivas gerações até uma população ótima, ou próximo a ótima, seja alcançada.

Informações mais detalhadas sobre os algoritmos genéticos podem ser encontrados em GOLDBERG (1989).

III.4 Otimização por enxame de partículas

A otimização por enxame de partículas (PSO - *Particle Swarm Optimization*) é uma técnica de computação evolucionária paralela desenvolvida por KENNEDY e EBERHART (1995), inspirada na metáfora do comportamento social de bandos de pássaros ou cardume de peixes.

O PSO compartilha muitas similaridades com técnicas de computação evolucionária tais como algoritmos genéticos (GA - *Genetic Algorithm*). Contudo, diferentemente do GA, o PSO não possui operadores evolucionários tais como cruzamento e mutação.

O algoritmo PSO é inicializado com uma população de soluções candidatas aleatória, chamada de partículas. A cada partícula é designada uma velocidade inicial aleatória e ela é movida iterativamente através do espaço de soluções do problema. Ela é atraída para as localizações de melhor função objetivo (*fitness*) alcançada, até então, por ela mesma (parte local do algoritmo) e para a localização da melhor função objetivo alcançada, até então, por toda a população (parte global do algoritmo).

O algoritmo padrão PSO pode ser descrito na notação vetorial por

$$\vec{v}_{k+1} = \vec{a} \otimes \vec{v}_k + \vec{b}_1 \otimes \vec{r}_1 \otimes (\vec{p}_1 - \vec{x}_k) + \vec{b}_2 \otimes \vec{r}_2 \otimes (\vec{p}_2 - \vec{x}_k) \quad (\text{III.17})$$

$$\vec{x}_{k+1} = \vec{c} \otimes \vec{x}_k + \vec{d} \otimes \vec{v}_{k+1} \quad (\text{III.18})$$

onde o símbolo \otimes significa multiplicação vetorial elemento por elemento. Na iteração k , o vetor velocidade \vec{v}_k é atualizado baseado no seu valor atual afetado por um fator de momento \vec{a} e por um termo que atrai a partícula para as melhores posições alcançadas anteriormente: a sua própria melhor posição anterior \vec{p}_1 e a melhor posição global \vec{p}_2 em todo o enxame. A força de atração é dada pelos coeficientes \vec{b}_1 e \vec{b}_2 . A posição da partícula \vec{x}_k é atualizada usando seu valor atual e a nova velocidade computada \vec{v}_{k+1} , afetada pelos coeficientes \vec{c} e \vec{d} , respectivamente. Variações aleatórias são introduzidas através dos vetores de números aleatórios \vec{r}_1 e \vec{r}_2 . Eles são normalmente selecionados como números aleatórios no intervalo $[0,1]$.

O pseudo código do procedimento executado pelo PSO é dado por

começo

$k=0$;

inicializa partículas \bar{x}_k e velocidades \bar{v}_k ;

enquanto condição de término não é satisfeita

para cada partícula

calcule o valor da função objetivo;

se o valor da função objetivo atual é melhor do que o melhor valor até o presente

faça a posição atual da partícula ser o novo \bar{p}_1 ;

fim

escolha a melhor posição de todas as partículas até o presente como \bar{p}_2 ;

$k = k+1$;

para cada partícula

calcule a velocidade da partícula de acordo com a equação (III.17);

atualize a posição da partícula de acordo com a equação (III.18);

fim

fim

fim

O algoritmo de otimização por enxame de partículas pode ser representado em uma forma de diagrama de fluxos. A figura III.3 apresenta de modo simplificado esta representação.

A condição de término pode ser o número de iterações, a convergência do enxame ou o alcance de um valor de *fitness* desejado.

Informações mais detalhadas sobre a otimização por enxame de partículas podem ser encontrados em KENNEDY *et al.* (1995).

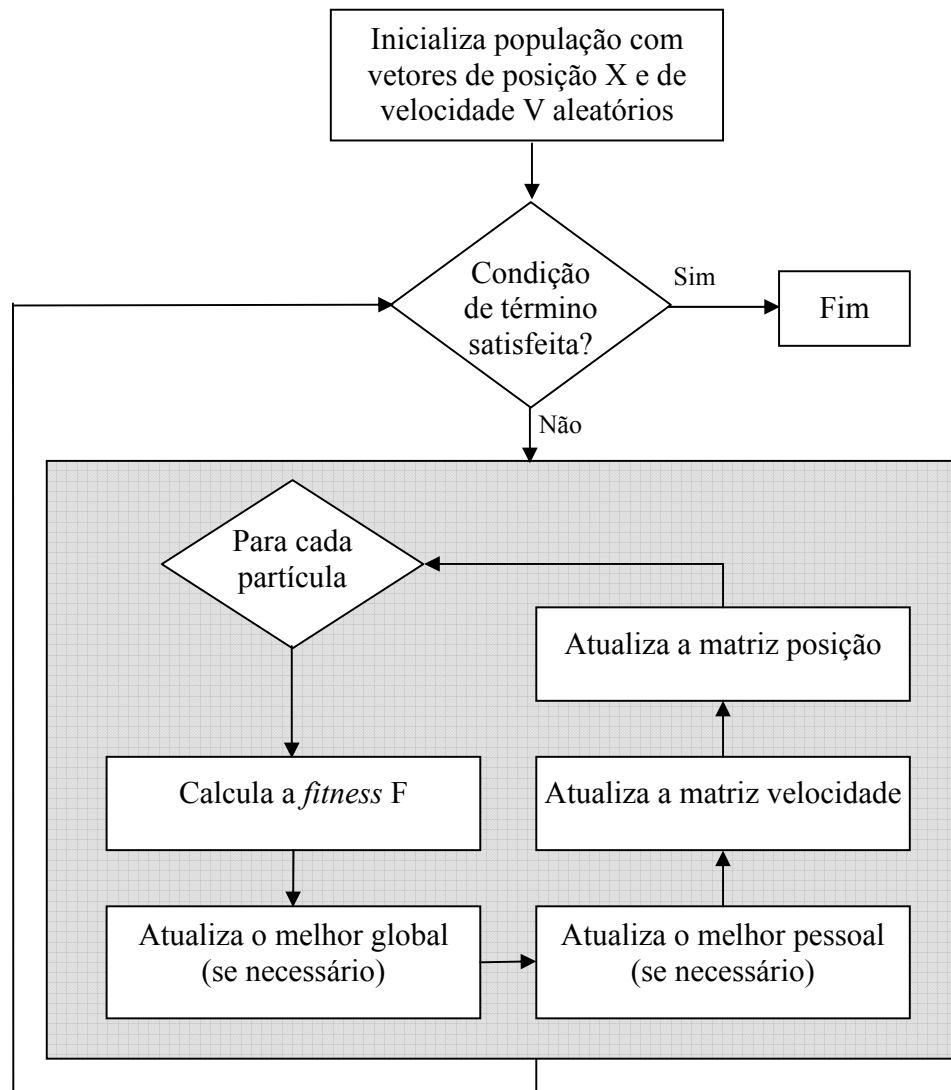


Figura III.3 - Gráfico de fluxos do algoritmo de otimização por enxame de partículas.

ANEXO IV

Lógica nebulosa

Conjuntos nebulosos são uma generalização da teoria dos conjuntos convencional, introduzidos por Zadeh em 1965, como um meio matemático para representar conceitos vagos utilizados na linguagem diária. A idéia básica dos conjuntos nebulosos é fácil de entender. Suponha que um aprendiz a motorista se aproxime de um semáforo vermelho e que o instrutor deva avisá-lo para pisar no freio do automóvel. O que ele deve dizer, “Comece a pisar no freio a 200 metros do semáforo” ou ele deve dizer, “Pise no freio quando estiver se aproximando”. Obviamente é a última frase; a instrução formal é precisa demais para ser utilizada. Isto ilustra que a precisão em certos casos pode ser quase inútil, enquanto a utilização de conceitos vagos pode ser melhor interpretada e gerar ações. A linguagem utilizada no dia a dia é um exemplo de como a “nebulosidade” é usada e propagada. Crianças rapidamente aprendem como interpretar e implementar instruções nebulosas (“Vá para cama em torno das 10”). Todos nós assimilamos e usamos (e agimos com base em) dados vagos, regras vagas e informação imprecisa, exatamente como nós somos capazes de tomar decisões sobre situações que parecem ser governadas pela sorte (ou pelo azar). Logo, modelos computacionais de sistemas reais deveriam ser capazes de reconhecer, representar, manipular, interpretar, e usar (agir com base) em ambas as incertezas, nebulosa e estatística.

Interpretações nebulosas de estruturas de dados são um modo muito natural e intuitivamente plausível para formular e resolver vários problemas. Conjuntos convencionais (*crisp*) contêm objetos que satisfazem propriedades precisas requeridas por pertinências. O conjunto H dos números de 6 a 8 é *crisp*; nós escrevemos $H = \{r \in \mathfrak{R} \mid 6 \leq r \leq 8\}$. Equivalentemente, H é descrito por sua função pertinência (FP) (ou função característica), $m_H : \mathfrak{R} \mapsto \{0,1\}$, definida como

$$m_H(r) = \begin{cases} 1; & 6 \leq r \leq 8; \\ 0, & \text{caso contrário.} \end{cases} \tag{IV.1}$$

O conjunto *crisp*, H, e o gráfico de m_H estão mostrados na metade esquerda da figura IV.1. Para qualquer número real r, ou ele está em H ou não está, uma vez que m_H mapeia todos os números reais $r \in \mathfrak{R}$ nos dois pontos (0,1). Os conjuntos *crisp* correspondem a uma lógica de dois valores: é ou não é, ligado ou desligado, branco ou preto, 1 ou 0. Na teoria dos conjuntos convencional, valores de m_H são chamados valores verdade com referência à questão, “r está em H?”. A resposta é sim, se e somente se, $m_H(r) = 1$; caso contrário, é não.

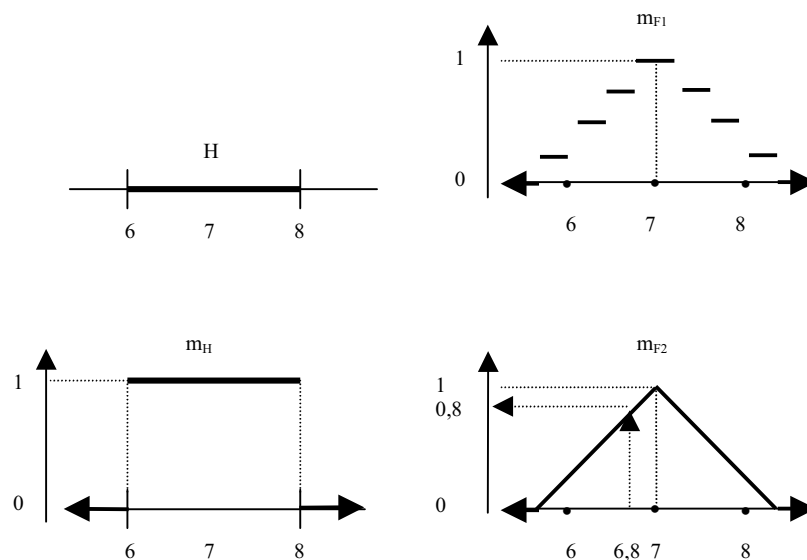


Figura IV.1 - Funções pertinências para subconjuntos rígidos e nebuloso de \mathfrak{R} .

Considere agora o conjunto F de números reais que estão próximos a 7. Desde que a propriedade “próximo a 7” é vaga, não há uma única função pertinência para F. Ao invés, é preciso decidir o que m_F deveria ser, baseado nas aplicações e propriedades desejadas para F. Propriedades que parecem ser plausíveis para F incluem: (i) normalidade ($m_F(7) = 1$); (i,i) monotonicidade (quanto mais perto r estiver de 7,

mais próximo $m_F(r)$ deve estar de 1 e, reciprocamente, quanto mais afastado r estiver de 7 mais próximo $m_F(r)$ deve estar de 0), e (iii) simetria (números igualmente distantes à esquerda e à direita de 7 devem ter pertinências iguais). Dadas estas restrições intuitivas, cada uma das funções mostradas na metade direita da figura IV.1 pode ser uma representação útil de F . No gráfico de barras horizontais m_{F1} é discreta, enquanto no gráfico triangular m_{F2} é contínua, mas não suave. Pode-se construir facilmente uma FP para F de tal forma que qualquer número tenha alguma pertinência positiva em F , mas não devemos esperar números “muito afastados de 7”, 20.000.987 por exemplo, para exagerar. Uma das maiores diferenças entre conjuntos clássicos (*crisp*) e nebulosos é que os primeiros sempre têm FPs únicas, enquanto qualquer conjunto nebuloso tem um número infinito de FPs que podem representá-lo. Isto é, ao mesmo tempo, uma fraqueza e uma força; a unicidade é sacrificada, mas isto dá, ao mesmo tempo, um ganho em termos de flexibilidade, habilitando modelos nebulosos a serem “ajustados” para a máxima utilidade em uma dada situação.

Na teoria dos conjuntos convencional, conjuntos de objetos reais, tais como os números em H , são equivalentes a, e isomorficamente descrito por, uma única função pertinência como m_H . Entretanto, na teoria dos conjuntos nebulosos não há conjuntos equivalentes de “objetos reais” correspondendo a m_F . Conjuntos nebulosos são sempre (e somente) funções, de um “universo de objetos”, digamos X , no intervalo $[0,1]$.

Como definido anteriormente, qualquer função $m: X \mapsto [0,1]$ é um conjunto nebuloso. Enquanto isto é verdade num senso matemático formal, muitas funções que se qualificam nesta base não podem ser adequadamente interpretadas como realizações de um conjunto nebuloso conceitual. Em outras palavras, funções que mapeam X no intervalo unitário podem ser conjuntos nebulosos, mas se tornam conjuntos nebulosos

quando, e somente quando, eles condizem com alguma descrição semântica intuitivamente plausível de propriedades imprecisas de objetos em X .

Adicionalmente, é importante distinguir nebulosidade de probabilidade. Conjuntos nebulosos não são uma forma inteligente de disfarçar modelos estatísticos. Estes dois tipos de modelo (nebuloso e estatístico) possuem, filosoficamente, diferentes espécies de informação: pertinências nebulosas representam similaridades de objetos para propriedades definidas imprecisamente, e probabilidades transmitem informação sobre frequências relativas. De outra forma, a lógica nebulosa é um meio de representar, manipular e utilizar dados e informações que possuem incertezas não estatísticas.

IV.1 Definições

Universo de discurso (X): é um conjunto clássico completo de elementos individuais capaz de ser referido ou quantificado.

Conjunto nebuloso (A): no universo de discurso X é um conjunto de pares ordenados.

$$A \equiv \{ (x, \mu_A(x) \mid x \in X) \} \quad (\text{IV.2})$$

onde $\mu_A(x)$ é chamada função pertinência de A , e é uma medida da pertinência de x ao conjunto A , ou seja, quanto x pertence a A

Normalmente, a função pertinência $\mu_A(x)$ mapeia o universo de discurso X no intervalo $[0,1]$:

$$\mu_A(x): X \mapsto [0,1] \quad (\text{IV.3})$$

Singleton: Na equação definida (V.2) os pares $(x, \mu_A(x))$ são chamados singletons.

Conjunto suporte: O conjunto suporte $\text{Supp}(A)$ de um conjunto nebuloso A é um conjunto rígido contendo aqueles elementos X cujos graus de pertinência são positivos:

$$\text{Supp}(A) \equiv \{ x \in X \mid \mu_A(x) > 0 \} \quad (\text{IV.4})$$

Diagrama de Zadeh: é uma representação gráfica das funções pertinência de um conjunto nebuloso A .

Por exemplo, um conjunto nebuloso A dos números próximos a 5 pode ser subjetivamente definido por meio da seguinte função pertinência $\mu_A(x)$

$$\mu_A(x) = \frac{1}{1 + 0.2(x-5)^2} \quad (\text{IV.5})$$

Conjunto nebuloso normal: Um conjunto nebuloso é chamado normal se existe pelo menos um elemento x_0 no domínio de discurso X onde sua função pertinência $\mu_A(x_0)$ é igual a um, isto é,

$$\mu_A(x_0) = 1 \quad (\text{IV.6})$$

Por exemplo, o conjunto nebuloso definido pela equação (IV.5) é um conjunto nebuloso normal.

Cortes alfa: Um corte alfa é um conjunto rígido consistindo de elementos do conjunto nebuloso A cujas pertinências são maiores ou igual a α . Os cortes alfa também são chamados conjuntos de nível e são denotados por

$$A_\alpha \equiv \{ x \in X \mid \mu_A(x) \geq \alpha \} \quad (\text{IV.7})$$

IV.2 Operações

As operações dos conjuntos nebulosos são realizadas através das suas funções pertinência. Existem operações relacionadas a um conjunto nebuloso (por exemplo, dilatação, intensificação de contraste, etc.) e operações que afetam dois ou mais conjuntos nebulosos (interseção, união, etc.). Desde que um conjunto nebuloso é definido com a ajuda de sua função-pertinência, é suficiente descrever como as operações afetam as funções pertinência. Após estas operações, o conjunto nebuloso resultante pode ser obtido usando uma de suas representações.

Há mais que um modo de definir as operações acima. Nós seguiremos as definições mais comuns onde a interseção e união são definidas via operações de mínimo (min) e máximo (max). Uma forma geral pode ser encontrada usando normas triangulares (t-normas) e conormas triangulares (t-conormas), entretanto, não há necessidade de coisas tão sofisticadas para este trabalho.

União: a função pertinência μ_U da união $U = A \cup B$ de dois conjuntos nebulosos A e B é definida como:

$$\mu_{A \cup B} \equiv \mu_A \vee \mu_B \equiv \max(\mu_A, \mu_B) \quad (\text{IV.8})$$

Interseção: a função pertinência μ_I da interseção $I = A \cap B$ de dois conjuntos nebulosos A e B é definida como:

$$\mu_{A \cap B} \equiv \mu_A \wedge \mu_B \equiv \min(\mu_A, \mu_B) \quad (\text{IV.9})$$

As definições acima podem ser estendidas a mais de dois conjuntos nebulosos.

Complemento: a função pertinência do complemento \bar{A} do conjunto nebuloso A é

$$\mu_{\bar{A}} \equiv 1 - \mu_A \quad (\text{IV.10})$$

Há aplicações onde é desejável controlar a incerteza (nebulosidade) de um conjunto nebuloso modificando o contraste entre os graus de pertinência alto e baixo. A operação apropriada é denominada contraste de intensificação.

Contraste de intensificação: de um conjunto nebuloso A é um conjunto nebuloso INT(A) com função pertinência dada por

$$\mu_{\text{INT}(A)} = \begin{cases} 2[\mu_A(x)]^2; & 0 \leq \mu_A(x) \leq 0,5 \\ 1-2[1-\mu_A(x)]^2; & 0,5 \leq \mu_A(x) \leq 1,0 \end{cases} \quad (\text{IV.11})$$

Esta operação pode ser repetidamente aplicada ao conjunto nebuloso A, entretanto, ao fim de certo número de repetições pode resultar num conjunto aproximadamente *crisp*.

Os seguintes modificadores são úteis quando são usadas variáveis lingüísticas.

Contração: a contração CON(A) de um conjunto nebuloso A é equivalente a elevar ao quadrado sua função pertinência.

$$\mu_{\text{CON}(A)} \equiv [\mu_A(x)]^2 \quad (\text{IV.12})$$

A operação inversa é a dilatação.

Dilatação de um conjunto nebuloso A é um novo conjunto nebuloso DIL(A) com uma função pertinência dada pela raiz quadrada da função pertinência do conjunto original A.

$$\mu_{\text{DIL}(A)} \equiv \sqrt{\mu_A(x)} \quad (\text{IV.13})$$

Até aqui, as operações sobre conjuntos nebulosos foram definidas através das funções-pertinência dos conjuntos resultantes. Isto pode realmente ser feito porque a função pertinência define univocamente o conjunto nebuloso por si mesma. Por outro

lado, dar a forma analítica da função pertinência não é o único modo de definir um conjunto nebuloso.

IV.3 Representações

A representação mais comum de um conjunto nebuloso é feita por meio da sua função-pertinência. Conforme já foi visto anteriormente, o diagrama de Zadeh é também uma representação possível de um conjunto nebuloso A através do gráfico de sua função pertinência $\mu_A(x)$.

Outra representação é possível quando temos um universo de discurso discreto X . Em tal caso, o conjunto nebuloso A pode ser representado pela união de seus *singletons*:

$$A = \max \{ (x_1 ; 0,8) ; (x_2 ; 0,3) ; (x_3 ; 0,5) \} \quad (\text{IV.14})$$

Ou como é mais freqüentemente escrito na literatura, $A = 0,8 / x_1 + 0,3 / x_2 + 0,5 / x_3$, onde o símbolo $+$ significa apenas uma união entre os *singletons*.

Genericamente,

$$A = \sum_{x_i \in X} (\mu_A(x_i) / x_i) \quad (\text{IV.15})$$

onde Σ indica união.

IV.4 Propriedades

As propriedades listadas abaixo se aplicam tanto aos conjuntos *crisp* quanto aos conjuntos nebulosos. Assim, sejam A e B conjuntos nebulosos com funções pertinência μ_A e μ_B , respectivamente.

a) Lei de dupla negação

$$\neg(\bar{A}) = A \quad (\text{IV.16})$$

b) Comutatividade

$$A \cap B = B \cap A \quad \text{e} \quad A \cup B = B \cup A \quad (\text{IV.17})$$

c) Associatividade

$$(A \cap B) \cap C = A \cap (B \cap C) \quad \text{e} \quad (A \cup B) \cup C = A \cup (B \cup C) \quad (\text{IV.18})$$

d) Leis de De Morgan

$$\neg(A \cap B) = \overline{B} \cup \overline{A} \quad \text{e} \quad \neg(A \cup B) = \overline{B} \cap \overline{A} \quad (\text{IV.19})$$

Em contrapartida, a Lei da Contradição e a Lei do Meio Excluído não se aplicam aos conjuntos nebulosos. Assim, tem-se que

$$A \cap \overline{A} \neq \phi \quad \text{e} \quad A \cup \overline{A} \neq X \quad (\text{IV.20})$$

onde ϕ é o conjunto vazio e X é o universo de discurso.

Considerando que um conjunto nebuloso pode ser expresso por uma função pertinência, as propriedades mencionadas podem, equivalentemente, ser expressas por funções pertinência. Como exemplo, a primeira Lei de De Morgan pode ser expressa como:

$$1 - [\mu_A(x) \wedge \mu_B(x)] = \mu_{\overline{A}}(x) \vee \mu_{\overline{B}}(x) \quad (\text{IV.21})$$

IV.5 Relações nebulosas

Qualquer associação entre elementos de diferentes conjuntos nebulosos é uma relação nebulosa. Funções são tipos especiais de relações nebulosas. Relações nebulosas podem ser interpretadas como um mapeamento onde os elementos do conjunto A são mapeados (transformados) em elementos do conjunto B , de uma forma ordenada. Se $a \in A$ e $b \in B$, dizemos que uma relação de A em B é um conjunto de pares ordenados

$(a, b) \in A \times B$, representada por $R = \{(a,b) \mid a \in A \text{ e } b \in B\}$ ou aRb . O fato de ser verdade aRb não implica necessariamente que seja verdade bRa . A ordem em uma relação é um atributo muito importante.

Enquanto relações clássicas descrevem somente a presença ou ausência de associação entre os elementos de dois ou mais conjuntos, relações nebulosas são capazes de capturar a força da associação. Em geral, relações nebulosas são conjuntos nebulosos definidos a partir de produtos cartesianos dos universos de discurso X, Y, Z , etc. como $X \times X$, $X \times Y$, $X \times Y \times Z$ ou qualquer outra combinação.

Como ilustração, consideremos relações binárias definidas sobre o produto cartesiano $X \times Y$. Uma vez que uma relação nebulosa é um conjunto nebuloso, ela pode ser representada pelos seus *singletons*. Neste caso, os *singletons* são pares ordenados (x, y) com a função pertinência $\mu_R(x, y)$ que medem o grau da associação existente entre os elementos x e y .

$$R = \{(x, y), \mu_R(x, y)\} \quad (\text{IV.22})$$

A relação R pode também ser representada, conforme mostrado anteriormente para o caso de conjuntos discretos, como união de seus *singletons*.

$$R = \sum_{(x_i, y_i) \in X \times Y} \mu_R(x_i, y_i) / (x_i, y_i) \quad (\text{IV.23})$$

Ou ainda como uma matriz de pertinência que é uma forma tabular dos valores da função pertinência.

$$R = \begin{bmatrix} \mu_R(x_1, y_1) & \mu_R(x_1, y_2) & \cdots & \mu_R(x_1, y_n) \\ \mu_R(x_2, y_1) & \mu_R(x_2, y_2) & \cdots & \mu_R(x_2, y_n) \\ \vdots & \vdots & \vdots & \vdots \\ \mu_R(x_m, y_1) & \mu_R(x_m, y_2) & \cdots & \mu_R(x_m, y_n) \end{bmatrix} \quad (IV.24)$$

O benefício de usar a matriz pertinência é a ilustração fácil das relações entre os elementos.

IV.6 Composição de relações nebulosas

A composição de duas relações, denotada por $R=P \circ Q$ consiste de pares (x,z) do produto cartesiano $X \times Z$ que pode ser compostos por meio de duas relações dadas por: P formada por pares $(x,y) \in X \times Y$; e Q formada por pares $(y,z) \in Y \times Z$.

Quando P e Q são relações nebulosas, cada composição de x e z via relações e um elemento particular $y \in Y$ é uma questão de grau. Este grau depende dos graus de pertinência $P(x,y)$ e $Q(y,z)$ e é determinado pelo menor destes dois graus de pertinência. Isto é, o grau de pertinência de uma cadeia (x,y,z) é determinada pelo grau das mais fracas das duas ligações, (x,y) e (y,z) , nesta cadeia. Também, entre as cadeias que conectam x a z , o maior grau de pertinência deverá ser o grau de pertinência que caracteriza a relação de x para z . Assim, a composição de relações nebulosas P e Q é definida para cada $(x,z) \in X \times Z$ pela expressão

$$R(x,z) = (P \circ Q)(x,z) = \max_{y \in Y} \min[P(x,y), Q(y,z)] \quad (IV.25)$$

A composição de relações nebulosas é mais fácil de visualizar em termos de mapeamento relacional nebuloso. Assim, sejam os conjuntos:

$$X = \{ a, b, c \} \quad (IV.26)$$

$$Y = \{ 1, 2, 3, 4 \} \tag{IV.27}$$

$$Z = \{ A, B, C \} \tag{IV.28}$$

E considere as relações nebulosas P, Q e $R = P \circ Q$ definidas em $X \times Y$, $Y \times Z$, $X \times Z$ pelos mapas relacionais definidos na figura IV.2 abaixo.

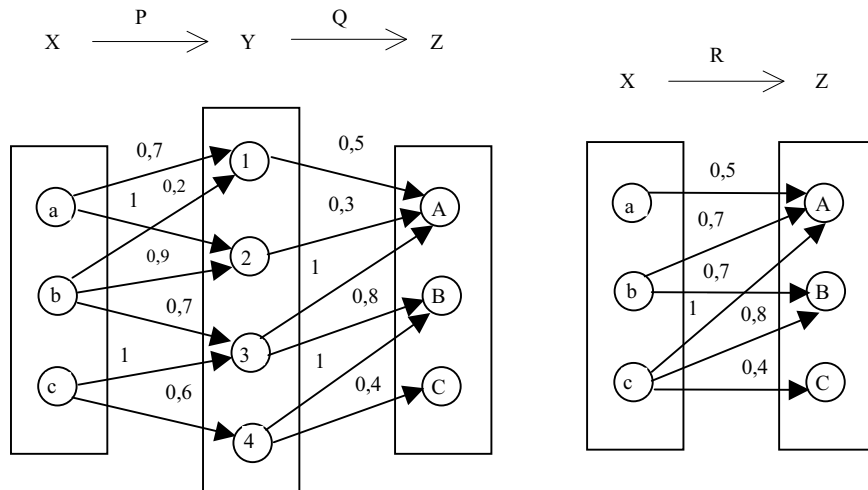


Figura IV.2 - Composição das relações nebulosas binárias P e Q.

Teremos:

$$\begin{aligned} (PoQ)(b, a) &= \max \{ \min [P(b,1), Q(1,A)]; \min [P(b,2), Q(2,A)]; \min [P(b,3), Q(3,A)] \} \\ &= \max \{ \min [0,2 ; 0,5]; \min [0,9 ; 0,3]; \min [0,7 ; 1] \} \\ &= \max \{ 0,2 ; 0,3 ; 0,7 \} = 0,7 \end{aligned} \tag{IV.29}$$

Composições de relações nebulosas são convenientemente realizadas em termos de suas representações matriciais. Sejam

$$\mathbf{P} = [p_{ij}] ; \mathbf{Q} = [q_{ik}] \text{ e } \mathbf{R} = [r_{ik}] \tag{IV.30}$$

as representações matriciais das relações nebulosas para quais $P \circ Q = R$. Então, podemos escrever

$$[r_{ik}] = [p_{ij}] \circ [q_{jk}] \quad (\text{IV.31})$$

$$\text{onde } r_{ik} = \max_j \min(p_{ij}; q_{jk}) \quad (\text{IV.32})$$

IV.7 Regras nebulosas multivariáveis

Normalmente as regras nebulosas *se-então* só possuem uma única variável nebulosa nas partes antecedente e conseqüente. Contudo, existem regras nebulosas que contêm, normalmente na parte antecedente, mais do que uma variável nebulosa.

O exemplo dado a seguir serve para esclarecer como utilizar estes tipos de regras.

Assuma que um controlador baseado em lógica nebulosa atue em cinco regiões que caracterizam o tempo t de atuação desse controlador expresso em linguagem natural como: Muito Pequeno (MP), Pequeno (P), Médio (M), Grande (G) e Muito Grande (MG). Assuma também que $t \in [0, t_{\max}]$. A figura IV.3 apresenta uma definição razoável desses números nebulosos representando essas expressões lingüísticas.

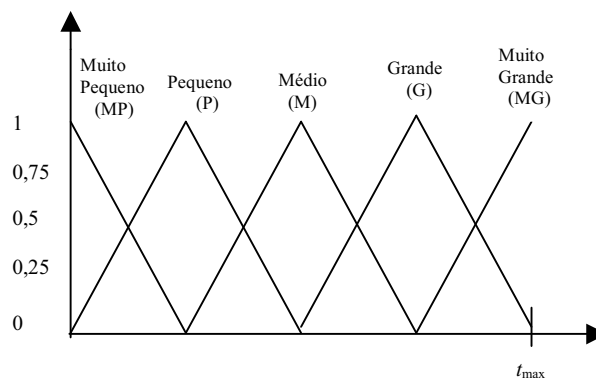


Figura IV.3 - Caracterização do tempo de atuação do controlador.

Suponha que para expressar o conhecimento sobre este processo utilizamos proposições nebulosas condicionais da forma

$$A = \square \quad \text{e} \quad B = \square \quad \text{então} \quad C = \square$$

Suponha também que são utilizadas três variáveis lingüísticas para descrever este processo.

Para cada proposição os valores das três variáveis lingüísticas são colocadas nas caixas vagas A e B. Como as variáveis A e B possuem três estados cada, o número total de pares possíveis desses estados são nove. Para cada um desses estados de pares ordenados, nós temos que determinar (usando qualquer conhecimento disponível) um estado apropriado da variável C. Isto resulta em nove distintas proposições nebulosas condicionais da forma mostrada acima. Essas proposições são usualmente chamadas de regras de inferência nebulosas ou regras nebulosas *se-então*. Exemplos de três dessas regras são,

$$\text{Se } A = P_A \text{ e } B = P_B \text{ então } C = MP_C$$

$$\text{Se } A = M_A \text{ e } B = M_B \text{ então } C = M_C$$

$$\text{Se } A = G_A \text{ e } B = P_B \text{ então } C = G_C$$

onde, os estados das variáveis lingüísticas podem, por exemplo, sere dados por

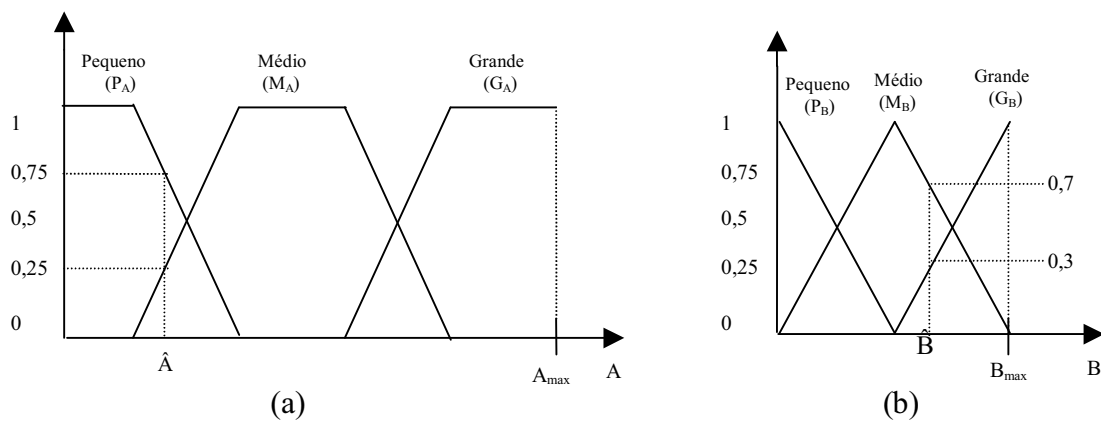


Figura IV.4 - Números nebulosos representando 3 níveis: (a) Variável nebulosa A;
(b) Variável nebulosa B.

Em cada regra, os estados de A e B são chamados antecedentes e o estado de C é chamado conseqüente.

Uma maneira conveniente de definir todas as regras necessárias é a matriz mostrada na tabela IV.1. As linhas da matriz correspondem aos estados das variáveis A, as colunas correspondem aos estados da variável B e os dados na matriz são estados da variável C. Observe que as regras de inferência nebulosa definidas por esta matriz estão em harmonia com o nosso senso comum.

Tabela IV.1

Regras de inferência nebulosa

		B		
		t	P_B	M_B
A	P_A	MP_C	P_C	M_C
	M_A	P_C	M_C	G_C
	G_A	M_C	G_C	MG_C

As nove regras de inferência nebulosas representam o conhecimento que temos do processo. Para valores dados das variáveis A e B podemos determinar o valor da variável C executando os seguintes passos:

Passo 1: Quando os valores específicos da medida das variáveis de entrada A e B, chamadas de \hat{A} e \hat{B} , são recebidos, suas pertinências são determinadas com os correspondentes antecedentes de todas as regras de inferência. Por exemplo, o valor medido \hat{A} , mostrado na figura IV.4a, é compatível com $A = P_A, M_A, G_A$ nos graus de 0,75; 0,25 e 0; respectivamente. Similarmente, os valores medidos \hat{B} , mostrados na figura IV.4b, é compatível com $B = P_B, M_B, G_B$ nos graus de 0; 0,7; 0,3;

respectivamente. Somente as regras para o qual são positivos os valores medidos de ambos antecedentes são utilizados na determinação do valor da variável do controlador. Essas regras são usualmente referidas como regras que disparam. No nosso exemplo, na tabela IV.1, as quatro regras que disparam estão identificadas por negritos.

Passo 2: Uma inferência é feita por cada regra que dispara. Para entender como isto é feito, é essencial descobrir qual das regras atende a aproximar a função f , que é virtualmente impossível de determinar exatamente pelos números nebulosos apropriados. No nosso exemplo, a função tem a forma

$$C = f(A, B) \quad (IV.33)$$

A aproximação de uma função pelos números nebulosos implicitamente envolve o princípio da extensão. Dada uma regra de inferência nebulosa particular com dois antecedentes, suas combinações é um conjunto nebuloso em \mathfrak{R}^2 (um conjunto nebuloso de duas dimensões). Se os antecedentes são independentes, como no nosso exemplo, sua combinação é definida como uma interseção de suas extensões cilíndricas. Os dois conjuntos nebulosos de duas dimensões obtidas desta maneira é então mapeado na consequência da regra. Qualquer antecedente que possua grau de compatibilidade com uma dada medida que seja menor do que 1 é truncada neste grau. A interseção das extensões cilíndricas desses antecedentes truncados é então truncado no grau mínimo de pertinência (assumindo interseção nebulosa padrão) e este truncamento é herdado pela parte consequente da regra pelo princípio da extensão.

Passo 3: Dadas as conclusões obtidas pelas regras de inferência nebulosas individuais, nós obtemos a conclusão geral fazendo a união de todas as conclusões individuais. No nosso exemplo, a conclusão geral do conjunto nebuloso $C_{\hat{A}, \hat{B}}$, cujas funções pertinência é definida para cada $x \in [0, t_{\max}]$, é dado pela fórmula

$$C_{\hat{A},\hat{B}}(t) = \max\{\min[P_A(\hat{A}), M_B(\hat{B}), P_C(t)], \min[P_A(\hat{A}), G_B(\hat{B}), M_C(t)], \min[M_A(\hat{A}), M_B(\hat{B}), M_C(t)], \min[M_A(\hat{A}), G_B(\hat{B}), G_C(t)]\} \quad (\text{IV.34})$$

Passo 4: Este é o último passo que é chamado de denebulização. O seu propósito é converter o conjunto nebuloso representando a conclusão geral obtida no passo 3 em um número real, que em algum senso, melhor representa o conjunto nebuloso. Apesar de existirem vários métodos de denebulização, o método mais comum é determinar o valor para o qual a área sob o gráfico da função pertinência é igualmente dividida. Este método é chamado de método de denebulização do centro de gravidade. Em geral, dado um conjunto nebuloso A , definido no intervalo $[a_1, a_2]$, o centro de gravidade da denebulização a , de A , é definido pela fórmula

$$a = \frac{\int_{a_1}^{a_2} xA(x)dx}{\int_{a_1}^{a_2} A(x)dx} \quad (\text{IV.35})$$

Aplicando esta fórmula ao nosso exemplo, obtemos

$$a = \frac{\int_0^{t_{\max}} xC_{\hat{A},\hat{B}}(t)dt}{\int_0^{t_{\max}} C_{\hat{A},\hat{B}}(t)dt} \quad (\text{IV.36})$$

ANEXO V

Algoritmos de separação em classes

V.1 Algoritmos de separação em classes rígidas

V.1.1 O Algoritmo KMEAN de MacQueen

O algoritmo Kmean de MacQueen efetua a separação de N amostras em C classes usando os seguintes passos:

1. Primeiro as C primeiras amostras do conjunto de dados são feitas como centróides que representam as C classes que se deseja dividir o conjunto de amostras;
2. Cada uma das $N-C$ amostras restantes são colocadas na classe com o centróide mais próximo. Após cada colocação, o centróide da classe vencedora é recalculado;
3. Após todas as amostras terem sido colocadas nas classes, os centróides das classes existentes são feitos como pontos fixos e novamente é feita uma recolocação de todo o conjunto de dados, onde cada amostra é colocada na classe com centróide mais próximo;

Para melhorar a convergência foi adicionado ao algoritmo o seguinte passo:

4. Os centróides são recalculados e cada amostra é recolocada na classe com centróide mais próximo até que nenhuma amostra troque mais de classe.

V.1.2 O algoritmo por quantização vetorial

O algoritmo de separação em classes AVQ-UCL pode ser realizado do seguinte modo:

1. Inicialmente são inicializados o número de classes desejadas C e os vetores sinápticos;

$$\mathbf{m}_i(0) = \mathbf{x}(i); i = 1, \dots, C \quad (\text{V.1})$$

2. Para cada amostra aleatória $\mathbf{x}(t)$ apresentada é calculado o vetor sináptico mais próximo

$$\mathbf{m}_i(t);$$

$$\| \mathbf{m}_i(t) - \mathbf{x}(t) \| = \min_i \| \mathbf{m}_i(t) - \mathbf{x}(t) \| \quad (\text{V.2})$$

onde $\| \mathbf{x} \|^2 = \mathbf{x}_1^2 + \dots + \mathbf{x}_n^2$ é o quadrado da norma Euclidiana de \mathbf{x} .

3. O vetor sináptico vencedor (mais próximo) $\mathbf{m}_i(t)$ é atualizado pela lei de aprendizado.

$$m_i(t+1) = m_i(t) + c_t [x(t) - m_i(t)] \quad (\text{V.3})$$

$$m_j(t+1) = m_j(t) \quad \text{se } j \neq i \quad (\text{V.4})$$

V.2 Método de separação em classes nebulosas

V.2.1 Algoritmo de média-C nebulosa

Dado um conjunto de amostras vetoriais \mathbf{x}_k , $1 < k < N$ e, se classificarmos essas amostras em um certo número C de classes nebulosas, então, uma dessas classes nebulosas é caracterizada por uma pertinência μ_{ik} , que mostra o grau de pertinência da k -ésima amostra na i -ésima classe. Para o cálculo de μ_{ik} assume-se que a equação (2.26) seja verdade para todas as k amostras. Se definirmos uma matriz U contendo todos os μ_{ik} , o problema fica resumido a encontrar C e determinar U .

O algoritmo de média- C nebulosa pode, então, ser realizado do seguinte modo:

1. Inicialmente são inicializados o número de classes desejadas C , o valor de m ($1 < m < \infty$) e a matriz de partição- C nebulosa U ;
2. Os centróides c_i são calculados usando a equação (2.19) e a matriz U é atualizada pela equação (2.27);
3. O passo 2 é repetido até que a diferença entre os valores de U calculados na iteração atual e na iteração anterior sejam menores do que um erro ε estipulado.

Neste algoritmo m é um parâmetro ajustável, que indica o grau de nebulosidade aplicado, que por simplicidade para as equações foi feito igual a 2.

V.3 Método de separação em classes possibilísticas

V.3.1 O Algoritmo de média- C possibilístico

O algoritmo de média- C possibilístico pode, então, ser realizado do seguinte modo:

1. Inicialmente são inicializados o número de classes desejadas C , o valor de m ($1 < m < \infty$) e a matriz de partição- C nebulosa U ;
2. O valor de η_i é calculado pela equação (2.31);
3. A matriz U é atualizada pela equação (2.30) até que a diferença entre os valores de U calculados na iteração atual e na iteração anterior sejam menores do que um erro ε estipulado.

V.4 Método de separação em classes nebulosas estendidas

V.4.1 Método de separação em classes nebulosas estendidas para a métrica Euclidiana

O algoritmo de separação em classes nebulosas estendidas (EFCM - *Extended Fuzzy C-Means*) com limiar adaptativo proposto por KRISHNAPURAM *et al.* (2000) considerando como métrica a distância Euclidiana é apresentado a seguir. Neste caso, a matriz norma para o algoritmo FCM é a matriz identidade.

Usando $\sqrt{\Lambda_i} Q_i^T I Q_i \sqrt{\Lambda_i}$ como o tamanho dos protótipos das classes obtemos

$$R_i = \sqrt{\Lambda_i} Q_i^T I Q_i \sqrt{\Lambda_i} = \Lambda_i \quad (\text{V.5})$$

Conseqüentemente, são obtidos valores diferentes para o raio dependendo da direção de seleção. Em geral, um valor entre o valor mínimo e o valor máximo dos elementos da diagonal de Λ_i pode ser usado como raio. A seleção do raio médio corresponde, então, a uma operação de média. O operador de média generalizado

$$D_i(S) = \frac{1}{n} \left\{ \sum_{j=1}^n \lambda_{ij}^S \right\}^{1/S}, \quad S \in \mathfrak{R} \quad (\text{V.6})$$

pode ser usado para este propósito (HARDY *et al.*, 1973). A escolha de valores diferentes de S em (V.6), que controla a polarização da agregação do tamanho de λ_{ij} , leva à obtenção de operadores diferentes de média. Para $S \rightarrow -\infty$, (V.6) se reduz ao operador mínimo e, deste modo, o protótipo com volume se torna a hiperesfera mais larga que pode ser construída dentro da classe com volume (hiperelipsóide) como mostrado na Figura V.1. Para $S \rightarrow \infty$, o operador máximo é obtido e, deste modo, o protótipo com volume torna-se a menor hiperesfera que pode ser construída dentro da classe com volume (hiperelipsóide). É sabido (HAYMAK *et al.*, 1998) que uma agregação sem polarização é obtida para medidas

em uma métrica espacial com $S \rightarrow 0$. O operador de média (V.6), então, se reduz a uma média geométrica e, deste modo, o raio do protótipo é dado por

$$r_i = \sqrt[n]{\prod_{j=1}^n \lambda_{ij}^{1/n}} = \sqrt{|P_i|^{1/n}} \quad (\text{V.7})$$

Assim sendo, esta seleção para o raio leva a protótipos esféricos que preservam o volume das classes.

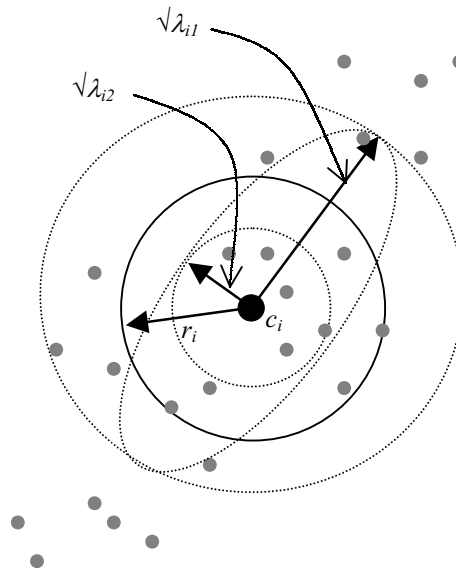


Figura V.1 - Classe com volume e o raio do EFCM para um exemplo bidimensional.

V.4.1.1 Algoritmo de separação em classes nebulosas estendidas para a métrica Euclidiana

Dado um conjunto de dados X , escolha um número inicial de classes $1 < M^{(0)} < C$, o grau de nebulização $m > 1$ e o critério de terminação $\varepsilon > 0$. Inicialize $U^{(0)}$ (por exemplo, aleatoriamente) e faça $S_{i^*j^*}^{(0)} = 1$, $\beta^{(0)} = 1$.

Repita para $l = 1, 2, \dots$

1. Calcule os protótipos das classes pontuais dados por

$$c_i^{(l)} = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m x_k}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.8})$$

2. Calcule os raios dos protótipos das classes a partir da covariância nebulosa

$$P_i = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m (x_k - c_i^{(l)})(x_k - c_i^{(l)})^T}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.9})$$

$$r_i = \beta^{(l-1)} \sqrt{P_i^{1/n} / M^{(l-1)}}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.10})$$

3. Calcule as distâncias para os protótipos das classes com volume

$$d_{ik} = \max\left(0, \sqrt{(x_k - c_i^{(l)})(x_k - c_i^{(l)})} - r_i\right), \quad \text{com } 1 \leq i \leq M^{(l-1)} \quad \text{e } 1 \leq k \leq C \quad (\text{V.11})$$

4. Atualize a matriz de partição

Para $1 \leq k \leq C$, faça $\phi_k = \{i \mid d_{ik} = 0\}$

Se $\phi_k = 0$,

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^{C^{(l-1)}} (d_{ik} / d_{jk})^{2/(m-1)}}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.12})$$

Caso contrário,

$$\mu_{ik}^{(l)} = \begin{cases} 0 & \text{se } d_{ik} > 0 \\ 1/|\phi_k| & \text{se } d_{ik} = 0 \end{cases} \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.13})$$

5. Selecione o par de classes mais similares

$$S_{ij}^{(l)} = \frac{\sum_{k=1}^N \min(\mu_{ik}^{(l)}, \mu_{ij}^{(l)})}{\sum_{k=1}^N \mu_{ik}^{(l)}}, \quad 1 \leq i, j \leq M^{(l-1)} \quad (\text{V.14})$$

$$(i^*, j^*) = \arg \max_{(i,j)} (S_{ij}^{(l)}), \quad i \neq j \quad (\text{V.15})$$

6. Junte as classes mais similares

$$\text{Se } \left| S_{i^* j^*}^{(l)} - S_{i^* j^*}^{(l-1)} \right| < \varepsilon \quad (\text{V.16})$$

$$\text{Faça } \alpha^{(l)} = 1/(M^{(l-1)} - 1) \quad (\text{V.17})$$

$$\text{Se } S_{i^* j^*}^{(l)} > \alpha^{(l)} \quad (\text{V.18})$$

$$\mu_{i^* k}^{(l)} = (\mu_{i^* k}^{(l-1)} + \mu_{j^* k}^{(l-1)}), \quad 1 \leq k \leq C \quad (\text{V.19})$$

remova a linha j^ de U*

$$M^{(l)} = M^{(l-1)} - 1 \quad (\text{V.20})$$

Caso contrário aumente o volume do protótipo

$$\beta^{(l)} = \min(M^{(l-1)}, \beta^{(l-1)} + 1) \quad (\text{V.21})$$

$$\text{Até } \| U^{(l)} - U^{(l-1)} < \varepsilon$$

V.4.2 Método de separação em classes nebulosas estendidas para a métrica Mahalanobis

O algoritmo de separação em classes nebulosas estendidas com limiar adaptativo proposto por KRISHNAPURAM *et al.* (2000) considerando como métrica a distância Mahalanobis é apresentado a seguir. GUSTAFSON *et al.* (1979) propuseram restringir o determinante da matriz norma em 1, isto é, $|A_i| = 1.0$. Então, a matriz norma é dada por

$$A_i = |P_i|^{1/n} P^{-1} \quad (\text{V.22})$$

Usando $\sqrt{\Lambda_i} Q_i^T A_i Q_i \sqrt{\Lambda_i}$, o tamanho do protótipo da classe é calculado como

$$R_i = \sqrt{\Lambda_i} Q_i^T |P_i|^{1/n} Q_i \Lambda_i^{-1} Q_i^T Q_i \sqrt{\Lambda_i} = |P_i|^{1/n} I \quad (\text{V.23})$$

Deste modo, o raio do protótipo com volume é determinado a partir da classe com volume como

$$r_i = \sqrt{|P_i|^{1/n}} \quad (\text{V.24})$$

V.4.2.1 Algoritmo de separação em classes nebulosas estendidas para a métrica Mahalanobis

Dado um conjunto de dados X , escolha um número inicial de classes $1 < M^{(0)} < C$, o grau de nebulização $m > 1$ e o critério de terminação $\varepsilon > 0$. Inicialize $U^{(0)}$ (por exemplo, aleatoriamente) e faça $S_{i^*j^*}^{(0)} = 1$, $\beta^{(0)} = 1$.

Repita para $l = 1, 2, \dots$

1. Calcule os protótipos das classes pontuais dados por

$$c_i^{(l)} = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m x_k}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.25})$$

2. Calcule os raios dos protótipos das classes a partir da covariância nebulosa

$$P_i = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m (x_k - c_i^{(l)})(x_k - c_i^{(l)})^T}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.26})$$

$$r_i = \beta^{(l-1)} \sqrt{|P_i|^{1/n} / M^{(l-1)}}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.27})$$

3. Calcule as distâncias para os protótipos das classes com volume

$$d_{ik} = \max\left(0, \sqrt{(|P_i|^{1/n})(x_k - c_i^{(l)})(x_k - c_i^{(l)}) - r_i}\right), \text{ com } 1 \leq i \leq M^{(l-1)} \text{ e } 1 \leq k \leq C \quad (\text{V.28})$$

4. Atualize a matriz de partição

Para $1 \leq k \leq C$, faça $\phi_k = \{i \mid d_{ik} = 0\}$

Se $\phi_k = 0$,

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^{C^{(l-1)}} (d_{ik} / d_{jk})^{2/(m-1)}}, \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.29})$$

Caso contrário,

$$\mu_{ik}^{(l)} = \begin{cases} 0 & \text{se } d_{ik} > 0 \\ 1/|\phi_k| & \text{se } d_{ik} = 0 \end{cases} \quad 1 \leq i \leq M^{(l-1)} \quad (\text{V.30})$$

5. Selecione o par de classes mais similares

$$S_{ij}^{(l)} = \frac{\sum_{k=1}^N \min(\mu_{ik}^{(l)}, \mu_{jk}^{(l)})}{\sum_{k=1}^N \mu_{ik}^{(l)}}, \quad 1 \leq i, j \leq M^{(l-1)} \quad (\text{V.31})$$

$$(i^*, j^*) = \arg \max_{(i,j)} (S_{ij}^{(l)}), \quad i \neq j \quad (\text{V.32})$$

6. Junte as classes mais similares

$$\text{Se } \left| S_{i^*j^*}^{(l)} - S_{i^*j^*}^{(l-1)} \right| < \varepsilon \quad (\text{V.33})$$

$$\text{Faça } \alpha^{(l)} = 1/(M^{(l-1)} - 1) \quad (\text{V.34})$$

$$\text{Se } S_{i^*j^*}^{(l)} > \alpha^{(l)} \quad (\text{V.35})$$

$$\mu_{i^*k}^{(l)} = (\mu_{i^*k}^{(l-1)} + \mu_{j^*k}^{(l-1)}), \quad 1 \leq k \leq C$$

remova a linha j^ de U*

(V.36)

$$M^{(l)} = M^{(l-1)} - 1$$
(V.37)

Caso contrário aumente o volume do protótipo

$$\beta^{(l)} = \min(M^{(l-1)}, \beta^{(l-1)} + 1)$$
(V.38)

Até $\|U^{(l)} - U^{(l-1)}\| < \varepsilon$