
Análise da Estabilidade Transitória via Rede
Neural ART-ARTMAP *Fuzzy* Euclidiana
Modificada com Treinamento Continuado

Angela Leite Moreno



UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Campus de Ilha Solteira

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Análise da Estabilidade Transitória via Rede Neural ART-ARTMAP
Fuzzy Euclidiana Modificada com Treinamento Continuado

ANGELA LEITE MORENO

Orientador: Prof. Dr. Carlos Roberto Minussi

Tese apresentada à Faculdade de Engenharia - UNESP - Campus de Ilha Solteira, para obtenção do título de Doutor em Engenharia Elétrica.
Área de Conhecimento: Automação.

Ilha Solteira - SP
Novembro-2010

FICHA CATALOGRÁFICA

Elaborada pela Seção Técnica de Aquisição e Tratamento da Informação
Serviço Técnico de Biblioteca e Documentação da UNESP - Ilha Solteira.

- M843a Moreno, Angela Leite.
Análise da estabilidade transitória via rede neural ART-ARTMAP Fuzzy Euclidiana modificada com treinamento continuado / Angela Leite Moreno. -- Ilha Solteira : [s.n.], 2010.
143 f. : il.
- Tese (doutorado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2010
- Orientador: Carlos Roberto Minussi
Inclui bibliografia
1. ART-ARTMAP. 2. Treinamento continuado. 3. Detecção de intruso. 4. Detecção de anomalia. 5. Detecção de novidade. 6. Teoria da ressonância adaptativa. 7. ARTMAP Fuzzy. 8. ART Euclidiana. 9. Análise de estabilidade transitória.



UNIVERSIDADE ESTADUAL PAULISTA
CAMPUS DE ILHA SOLTEIRA
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA

CERTIFICADO DE APROVAÇÃO

TÍTULO: Análise da Estabilidade Transitória via Rede Neural ART-ARTMAP Fuzzy Euclidiana
Modificada com Treinamento Continuado

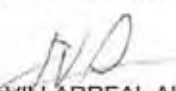
AUTORA: ANGELA LEITE MORENO

ORIENTADOR: Prof. Dr. CARLOS ROBERTO MINUSSI

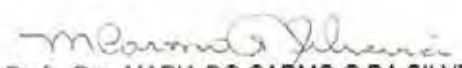
Aprovada como parte das exigências para obtenção do Título de DOUTOR em ENGENHARIA ELÉTRICA, Área: AUTOMAÇÃO, pela Comissão Examinadora:


Prof. Dr. CARLOS ROBERTO MINUSSI

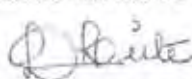
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira


Prof. Dr. FRANCISCO VILLARREAL ALVARADO

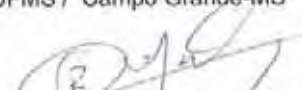
Departamento de Matemática / Faculdade de Engenharia de Ilha Solteira


Profa. Dra. MARIA DO CARMO G DA SILVEIRA

Serviço Técnico de Informática - STI / Faculdade de Engenharia de Ilha Solteira


Profa. Dra. LUCIANA CAMBRAIA LEITE

Departamento de Engenharia Elétrica - DEL / Universidade Federal de Mato Grosso do Sul - UFMS / Campo Grande-MS


Prof. Dr. RICARDO MENEZES SALGADO

Instituto de Ciências Exatas / Universidade Federal de Alfenas - Alfenas/MG

Data da realização: 22 de outubro de 2010.

Ao meu amado esposo,

Roberto,

e à minha querida filha,

Carolina.

Agradecimentos

Em primeiro lugar, agradeço a Deus, que durante todos esses anos esteve sempre comigo, me inspirando em todos os momentos.

Ao meu esposo, Roberto, que durante este tempo me apoiou, me incentivou, me compreendeu e, especialmente, pelo amor que me dedica e pela cumplicidade que temos, com quem compartilho esta conquista.

À minha filha, Carolina, pela paz e alegria que trouxe à minha vida, pela devoção, carinho e amor.

Aos meus pais, pelo carinho e por tudo que fizeram por mim. Às minhas irmãs, Rosinéia e Rosângela, pelo carinho que compartilhamos.

Aos meus amigos Solange, Marleide, Nair, Alcindo e Carlos, uma segunda família para mim, sem o seu apoio e carinho, com certeza eu não conseguiria ter chegado aqui.

Aos meus amigos e professores do curso de graduação em Licenciatura Plena em Matemática pela Universidade Federal de Mato grosso do Sul de Três Lagoas, onde comecei esta jornada, pelo ano que passei com vocês, em especial, aos professores Edson, Romanini, Rosana e Tamarozzi, e, aos amigos, Danielli, Elisabete e Alexandre, pela amizade dedicada.

Aos meus amigos e professores do curso de graduação em Licenciatura Plena em Matemática pela UNESP de Presidente Prudente, em especial os professores Gabriel, José Roberto, Messias e Suetônio que sempre me ajudaram e me apoiaram a continuar os estudos e, os alunos Everaldo e Rodrigo, por serem verdadeiramente amigos. Em especial ao Professor Carlos Rodrigues Biroca, pela sua orientação, amizade e incentivos durante os anos de graduação.

Aos amigos e professores do curso de mestrado em Matemática do ICMC de São Carlos pela disposição em compartilhar o conhecimento. Em especial à Professora Márcia Federson, pela amizade, paciência e dedicação durante minha estada em São Carlos. Aos amigos, pela amizade oferecida durante este período.

Aos professores do curso de Pós-Graduação em Engenharia da Faculdade de Engenharia de Ilha Solteira, pelo conhecimento transmitido, pela cooperação e pela disposição em nos atender sempre com muita atenção.

Sou extremamente grata ao meu orientador, professor Carlos Roberto Minussi que, durante todo o processo de doutoramento, me apoiou e ajudou desde o momento em que traçamos o objetivo até sua completa realização, com sua paciência, sabedoria e compreensão. Por todas as conversas que tivemos.

À professora Mara Lúcia Martins Lopes, por ter me mostrado o caminho, no início de meu percurso.

Aos colegas de pós-graduação, pelas dúvidas e conhecimentos compartilhados durante a primeira fase do curso.

Aos amigos Edson, Elizete, Inocência, Jaime, Marcela, Neide, Osmar e Sirlene, que estiveram em minha companhia nos momentos de diversão.

Aos amigos da Universidade Federal de Alfenas, principalmente do Instituto de Ciências Exatas, pelo apoio e incentivo durante esses dois últimos anos. Não posso deixar de citar: Aldício, Andréa, Denismar, Éric, Érica, Evandro, Fabrício, José Carlos, José Claudinei, José Paulo, Luciana, Luiz e Rejane, sempre tão prestativos e preocupados.

Não poderia deixar de agradecer ao pessoal da Pós-Graduação, em especial, à Marcia e à Onilda, por todas as vezes em que necessitei dos serviços da pós. Também, aos funcionários do Departamento de Engenharia Elétrica, por sua ajuda, às vezes indireta, durante a realização deste trabalho, principalmente, Beto e Deoclécio.

Também agradeço a todos os funcionários da Faculdade de Engenharia de Ilha Solteira, pela colaboração e presteza.

E, por último, mas não menos importante, à CAPES pelo apoio financeiro sem o qual não seria possível realizar este trabalho.

Resumo

Esta pesquisa visa o desenvolvimento de um método para análise da estabilidade transitória de sistemas de energia elétrica multimáquinas, por meio de uma rede neural ART-ARTMAP *Fuzzy* Euclidiana Modificada com Treinamento Continuado. Esta arquitetura apresenta três diferenciais em relação a outras já utilizadas para abordar tal problema: (1) a rede é iniciada com apenas um neurônio ativado e vai se expandindo durante todo o treinamento/análise, (2) possui um módulo de treinamento continuado e (3) possui um módulo de detecção de intruso. No primeiro diferencial, a rede é iniciada com um neurônio e vai se expandindo de acordo com a aquisição de conhecimento, isto faz com que esta se torne muito mais rápida e que o gasto computacional se torne mínimo. Com o módulo de treinamento continuado, a rede neural consegue armazenar novos dados sem a necessidade de realizar o retreinamento. Já o módulo de detecção de intruso faz com que, ao ser apresentada à rede uma configuração “estranha”, a rede execute um treinamento específico para que esta configuração, com um número mínimo de entradas, seja incorporada definitivamente à rede neural. A aplicação para a rede proposta nesta pesquisa, foi a análise de estabilidade transitória, considerando-se o modelo clássico (estabilidade de primeira oscilação), para um sistema composto por 10 máquinas síncronas, 45 barras e 73 linhas de transmissão.

Palavras-chave: ART-ARTMAP. Treinamento Continuado. Detecção de Intruso. Detecção de Anomalia. Detecção de Novidade. Teoria da Ressonância Adaptativa. ARTMAP *Fuzzy*. ART Euclidiana. Análise da Estabilidade Transitória.

Abstract

This doctoral research aims to develop a method to analyze the transient stability of multimachine electric power systems, through a neural network Modified Euclidean Fuzzy ART-ARTMAP with Continuous Training. The architecture presented has three differences in relation to others used to deal with this problem: (1) the network starts with only one neuron activated and expands throughout the training/analysis, (2) has a continuous training module and (3) has an intrusion detection module. The first difference, is the fact that it starts with a neuron and expands according to knowledge acquisition of the network, and causes it to become much faster and the computational expenses becomes minimum. With continuous training module, the neural network can store the new data without the need for the retraining. The intrusion detection module causes, when presented to the network a strange configuration, the network to carry out a specific training for this configuration with a minimum total of inputs so that the configuration is definitely incorporated to the neural network. The application for this network, in this research, was to analyze the transient stability considering the classical model (stability of first oscillation) to a system composed of 10 synchronous machines, 45 buses and 73 transmission lines.

Key words: ART-ARTMAP. Continuous Training. Novelty Detection. Anomaly Detection. Novelty Intrusion. Adaptive Resonance Theory. *Fuzzy* ARTMAP. Euclidean ART. Analysis of Transitory Stability.

Sumário

1	Introdução	16
2	Pesquisa Proposta	20
2.1	Rede Neural Proposta	20
2.2	Justificativas	23
3	Análise da Estabilidade Transitória	24
3.1	Modelo do Sistema	25
3.1.1	Descrição do Modelo do Sistema	26
3.1.2	Modelo do Sistema Referido ao Centro de Ângulos	29
3.2	Margem de Segurança	31
3.3	Análise de Segurança	32
3.3.1	Estados de Operação da Rede elétrica	35
4	Teoria da Ressonância Adaptativa	37
4.1	Rede Neural ART	38
4.1.1	Parâmetros de uma Rede ART	41
4.1.2	Algoritmo de uma Rede Neural ART	42
4.2	Rede Neural ART Euclidiana	46
4.3	Rede Neural ARTMAP <i>Fuzzy</i>	48
4.3.1	Algoritmo da rede neural ARTMAP <i>Fuzzy</i>	49
5	Redes ART e ARTMAP <i>Fuzzy</i> Modificada	56
5.1	Rede Neural ART <i>Fuzzy</i> Modificada	57
5.2	Rede Neural ARTMAP <i>Fuzzy</i> Modificada	62
6	ART Euclidiana Modificada com Treinamento Contínuo	64
6.1	Mecanismo do Treinamento Contínuo	65
6.2	Uma Primeira Aplicação	71

7	Rede Neural Proposta e Aplicação	82
7.1	Problema Proposto	83
7.2	Configuração do Sistema	86
7.3	Análise Experimental	87
8	Conclusão e Sugestão para Trabalhos Futuros	95
8.1	Conclusão	95
8.2	Sugestões para Trabalhos Futuros	97
	Referências	99
A	Redes Neurais Artificiais	105
A.1	Visão Histórica	105
A.2	Algumas Considerações sobre Redes Neurais	108
A.2.1	Por Quê Utilizar Redes Neurais Artificiais?	109
A.3	Estruturas de Redes Neurais	110
A.4	Modelo de Neurônio Biológico e Modelo de Neurônio Artificial	111
A.4.1	Modelo Biológico	111
A.4.2	Neurônio Artificial	112
A.5	Arquitetura da Rede	115
A.6	Algoritmo de Aprendizagem	117
A.7	Aplicações de Redes Neurais	121
B	Lógica <i>Fuzzy</i>	123
B.1	Conjuntos <i>Fuzzy</i>	124
B.2	Estrutura do Sistema Lógico <i>Fuzzy</i>	130
B.3	Variáveis Linguística	130
B.4	Inferência <i>Fuzzy</i>	132
B.4.1	Regras <i>Fuzzy</i>	132
B.4.2	Inferência <i>Fuzzy</i>	133
B.5	Métodos de Defuzzificação	134
B.5.1	Média dos Máximos	134
B.5.2	Centro de Área	136
B.5.3	Método das Alturas	137
C	Dados do Sistema de Energia Elétrica	139
D	Artigos Publicados Relacionados com a Pesquisa de Doutorado	143

Lista de Figuras

2.1	Rede Neural ART-ARTMAP <i>Fuzzy</i> Euclidiana Modificada com Treinamento Continuo.	21
3.1	Estados de Operação.	36
4.1	Mapa de Campo de uma rede ART.	39
4.2	Fluxograma de uma Rede ART Nebulosa.	45
4.3	Fluxograma da Rede ART Euclidiana.	47
4.4	Mapa de Campo de uma Rede ARTMAP <i>Fuzzy</i> .	49
4.5	Fluxograma de uma Rede Neural ARTMAP <i>Fuzzy</i> .	55
5.1	Fluxograma de uma Rede ART <i>Fuzzy</i> Modificada.	62
5.2	Fluxograma de uma Rede Neural ARTMAP <i>Fuzzy</i> Modificada.	63
6.1	Fluxograma da Rede Neural ART Euclidiana Modificada com Treinamento Continuo.	70
6.2	Visualização da Primeira Entrada.	72
6.3	Classificação do Ponto (0, 0; 0, 0) Antes e Depois do Treinamento Continuo.	72
6.4	Classificação do Ponto (0, 5; 0, 3) Antes e Depois do Treinamento Continuo.	73
6.5	Classificação do Ponto (0, 0; 0, 3) Antes e Depois do Treinamento Continuo.	73
6.6	Classificação do Ponto (1, 4; 1, 0) Antes e Depois do Treinamento Continuo.	74
6.7	Classificação do Ponto (0, 9; 0, 8) Antes e Depois do Treinamento Continuo.	74
6.8	Classificação do Ponto (0, 5; 0, 0) Antes e Depois do Treinamento Continuo.	75
6.9	Classificação do Ponto (1, 0; 0, 0) Antes e Depois do Treinamento Continuo.	75
6.10	Classificação do Ponto (−1, 0; 0, 0) Antes e Depois do Treinamento Continuo.	76
6.11	Classificação do Ponto (−1, 0; −1, 0) Antes e Depois do Treinamento Continuo.	76
6.12	Classificação do Ponto (0, 0; −1, 0) Antes e Depois do Treinamento Continuo.	77
6.13	Classificação do Ponto (0, 0; 1, 0) Antes e Depois do Treinamento Continuo.	77

6.14	Classificação do Ponto $(0, 0; 2, 0)$ Antes e Depois do Treinamento Continuado. . .	78
6.15	Classificação do Ponto $(2, 0; 2, 0)$ Antes e Depois do Treinamento Continuado. . .	78
6.16	Classificação do Ponto $(0, 2; 0, 2)$ Antes e Depois do Treinamento Continuado. . .	79
6.17	Classificação do Ponto $(1, 0; 0, 5)$ Antes e Depois do Treinamento Continuado. . .	79
6.18	Classificação do Ponto $(2, 0; 0, 1)$ Antes e Depois do Treinamento Continuado. . .	80
6.19	Classificação do Ponto $(2, 3; 1, 3)$ Antes e Depois do Treinamento Continuado. . .	80
6.20	Classificação do Ponto $(-1, 1; -0, 3)$ Antes e Depois do Treinamento Continuado. . .	81
6.21	Classificação do Ponto $(-1, 5; -1, 1)$ Antes e Depois do Treinamento Continuado. . .	81
7.1	Configuração do Sistema Elétrico da Região Sul do Brasil.	87
A.1	Rede Neural.	111
A.2	Neurônio Biológico.	112
A.3	Neurônio Artificial.	112
A.4	Funções de Ativação e suas Equações.	114
A.5	Neurônio Biológico e Neurônio Artificial: Comparação.	115
A.6	Arquitetura de Redes Neurais Artificiais.	116
A.7	Aprendizado Supervisionado.	118
A.8	Aprendizado Não-Supervisionado.	118
A.9	Aprendizado por Reforço.	119
B.1	Exemplo de Função de Pertinência do Tipo Linear por Partes.	125
B.2	Exemplo de Função de Pertinência do Tipo Gaussiana.	125
B.3	Conjuntos <i>Fuzzy</i> : A e B	126
B.4	Interseção entre Conjuntos <i>Fuzzy</i>	126
B.5	União entre Conjuntos <i>Fuzzy</i>	126
B.6	Exemplo do Complemento de um Conjunto <i>Fuzzy</i>	127
B.7	Lei do Meio Excluído.	127
B.8	Lei da Contradição.	128
B.9	Exemplo de α -Nível.	129
B.10	Sistema Lógico <i>Fuzzy</i>	130
B.11	Exemplo de Variável Linguística.	131
B.12	Exemplo do Método de Defuzzificação MM.	135
B.13	Exemplo de Problemas que o Método de Defuzzificação MM pode Causar.	135
B.14	Exemplo do Método de Defuzzificação CA.	136
B.15	Exemplo do Método de Defuzzificação MA.	137

Lista de Tabelas

6.1	Exemplo Prático da Dinâmica do Treinamento Continuado.	71
7.1	Resultados da Análise da Estabilidade Transitória Obtidos por Simulação.	89
C.1	Dados de barras do sistema sul-brasileiro (10 máquinas).	139
C.2	Dados do sistema de transmissão.	141
C.3	Dados das máquinas síncronas.	142

Introdução

As redes neurais artificiais [24] têm conquistado um grande espaço no contexto da resolução de uma série de problemas em várias áreas do conhecimento humano nas últimas décadas, *e.g.*, engenharias, medicina, economia, biologia, astronomia, agricultura, entre outras especialidades. Este espaço foi conquistado por meio de uma intensa dedicação dos pesquisadores. Vários obstáculos foram superados, desde a proposta do primeiro modelo de neurônio artificial, por McCulloch e Pitts [40], em 1943, passando pelo grande agente desmotivador proclamado por Minsky e Papert [41], em 1969. A partir da proposição do algoritmo retropropagação por Werbos e pelo advento da microeletrônica é que, efetivamente, esta área galgou novos ânimos. Contudo, ainda hoje muitas são as críticas com relação ao uso de redes neurais artificiais. Talvez a maior delas se refere a incapacidade da maioria das arquiteturas neurais de considerar a inclusão de novos conhecimentos à medida que novas informações forem disponibilizadas. Ou seja, uma vez concluído o treinamento, se houver a necessidade de continuar o processo de aprendizado, o treinamento deverá ser reinicializado, podendo, a grosso modo, ser considerado um procedimento bastante limitado.

Em muitos problemas, esta deficiência é crucial, pois o desempenho, no processo de análise, deve ser aprimorado ao longo do tempo. Por exemplo, em sistemas elétricos de potência, muitas são as condições operativas procedentes da mudança da carga durante o dia (curva de carga), da mudança da configuração do sistema, etc. Ressalta-se que a curva de carga varia, também, nos diferentes dias das semanas, dos meses e das estações anuais, etc. Assim, se não for implementado um esquema que possibilite aprimorar a aquisição do conhecimento por parte da rede neural usada, os resultados esperados, certamente, serão pouco úteis.

Visando superar este problema (treinamento não-dinâmico) foi desenvolvido nesta pesquisa um mecanismo para a extração do conhecimento de forma continuada designado treinamento continuado. Para que este mecanismo se adapte ao problema proposto, foi incluído um mecanismo de detecção de intruso, um filtro que indica quando uma configuração não pertence aos padrões já treinados pela rede. Neste caso, a rede neural exige um conjunto de padrões suficientemente dimensionado para sedimentar um conhecimento capaz de fornecer respostas com um nível de precisão adequado em relação a tal configuração. Para a obtenção destes dados é utilizado o programa *Simul*. Nele as simulações são executadas considerando diversos perfis de carga e sementes, estas geradas pseudo-aleatoriamente, para que a rede incorpore o conhecimento necessário para a sedimentação do conhecimento em relação àquela configuração em particular. Após a incorporação da configuração à rede neural, o passo seguinte consistiu na inclusão, na base de dados (conjunto de pesos), de novas informações que se encontram acessíveis. Estas novas informações foram filtradas, ou seja, foi avaliada a relevância de incorporá-las às matrizes de pesos da rede neural, ou descartá-las. Este processo foi realizado com muito cuidado, isto porque, se a rede descartar em demasia, o treinamento permanecerá não-dinâmico (igual ao comumente adotado na literatura) e, se incorporados maciçamente, poder-se-á provocar um sobre-treinamento que é igualmente uma solução nada produtiva. Para resolver este dilema foram incorporados à rede dois tipos de filtros: um mecanismo de detecção de anomalia e um mecanismo de detecção de novidade. O mecanismo de detecção de anomalia impede que a rede incorpore padrões que representem apenas uma anomalia, *i.e.*, que figurem durante o treinamento/análise um número de vezes insuficiente para ser considerado uma informação importante. Já o mecanismo de detecção de novidade é utilizado quando o padrão pertence a uma classe permanente, *i.e.*, faz parte da base de dados. Ele evita que padrões muito próximos sejam retreinados, assim para que um padrão seja adicionado a um *cluster* definitivo é necessário que a distância entre este padrão e o centro do *cluster* seja maior que o índice de novidade. Ainda cabe ressaltar que, a fase de treinamento ocorre independentemente da fase de análise apenas quando o mecanismo de detecção de intruso

for acionado. Nas outras situações, por tratar-se de uma rede com Treinamento Continuado, não há diferenciação entre as fases de treinamento e análise pois ambas ocorrem simultaneamente.

Os mecanismos acima mencionados foram agregados a uma rede neural ART [10]. A nova rede neural gerada com esta pesquisa foi designada rede neural ART-ARTMAP *Fuzzy* Euclidiana Modificada com Treinamento Continuado. O termo “modificado” é empregado para caracterizar o processo simplificado, proposto nesta pesquisa, que busca tornar a execução da rede neural mais rápida, preservando a precisão dos resultados auferidos. Este será, então, o procedimento a ser desenvolvido e apresentado a seguir.

Visando mostrar o desempenho do método proposto (sistema neuro-*fuzzy* euclidiano), apresenta-se uma aplicação considerando-se um sistema elétrico de potência de porte médio, correspondente a uma configuração do sistema Sul-Brasileiro, composto por 10 máquinas síncronas, 45 barras e 77 linhas de transmissão/transformadores.

Este texto está assim organizado:

Capítulo 2: Apresenta-se aqui a rede neural desenvolvida nesta tese, juntamente com algumas informações sobre suas características. Também são apresentadas as justificativas a respeito de como foram escolhidos os mecanismos de detecção de intrusão, detecção de anomalia e detecção de novidade, incorporados à rede, para tratar o problema da análise de estabilidade transitória.

Capítulo 3: Este capítulo é voltado a introduzir o problema a ser abordado pela rede neural: a análise da estabilidade transitória. Nele são apresentados o modelo dinâmico do sistema e o modelo do sistema referido ao centro de ângulos. A seguir é apresentado o conceito de margem de segurança e, finalizando com a apresentação de conceitos voltados a análise de segurança do sistema de energia elétrica.

Capítulo 4: Este é dedicado à Teoria da Ressonância Adaptativa. Primeiramente é dada uma visão geral sobre este tipo de rede para, em seguida, apresentar detalhadamente as redes ART *Fuzzy*, ART Euclidiana e ARTMAP *Fuzzy*, com respectivos algoritmos e fluxogramas detalhados.

Capítulo 5: Neste capítulo é apresentada a primeira contruibuição desta pesquisa, uma rede neural ART *Fuzzy* Modificada e, conseqüentemente, a versão ARTMAP *Fuzzy* Modificada.

Capítulo 6: Neste, apresenta-se a rede neural ART Euclidiana Modificada com Treinamento Continuado que é a principal contribuição desta pesquisa. Foram incorporados a esta rede

módulos de detecção de anomalia e de detecção de novidade para tornar o treinamento mais dinâmico.

Capítulo 7: Apresenta-se, neste capítulo, a rede neural ART-ARTMAP *Fuzzy* Euclidiana Modificada com o Treinamento Continuado já estruturada para a aplicação prevista para este trabalho e, em seguida, a referida aplicação.

Capítulo 8: Dedicado a encerrar esta tese, neste capítulo são apresentados as conclusões deste trabalho e sugestões para trabalhos futuros.

Apêndice A: Neste apêndice são apresentados os principais conceitos de redes neurais, inclusive uma visão histórica de sua evolução.

Apêndice B: Nele são descritos os principais conceitos sobre Teoria de Conjuntos *Fuzzy*, Lógica *Fuzzy*, Inferência *Fuzzy* e Métodos de Defuzzificação.

Apêndice C: São apresentados os dados do sistema teste (versão do sistema elétrico de potência da região sul do Brasil).

Apêndice D: Neste apêndice são apresentados os trabalhos publicados relacionados com esta pesquisa.

Pesquisa Proposta

Neste capítulo é apresentada, resumidamente, a pesquisa proposta bem como as inovações que foram incorporadas à rede neural ART Fuzzy e à rede neural ART Euclidiana, gerando uma nova rede: rede neural ART-ARTMAP Fuzzy Euclidiana Modificada com Treinamento Continuado (Figura 2.1). Isto além do mecanismo de detecção de intruso, uma condição rígida em relação às configurações que são apresentadas a rede visando a detecção de configurações “estranhas”, i.e., que não fazem parte da rede e, realizando o treinamento com um número mínimo de entradas referentes à configuração, “estranha”. Antes deste mecanismo, sempre que fosse apresentada à rede uma nova configuração, era necessário reinicializar o treinamento com todos os dados até o momento.

2.1 Rede Neural Proposta

O primeiro diferencial nesta pesquisa é que, diferentemente do procedimento convencional em que o número de unidades neurais na camada competitiva de cada módulo ART *Fuzzy* é arbitrado

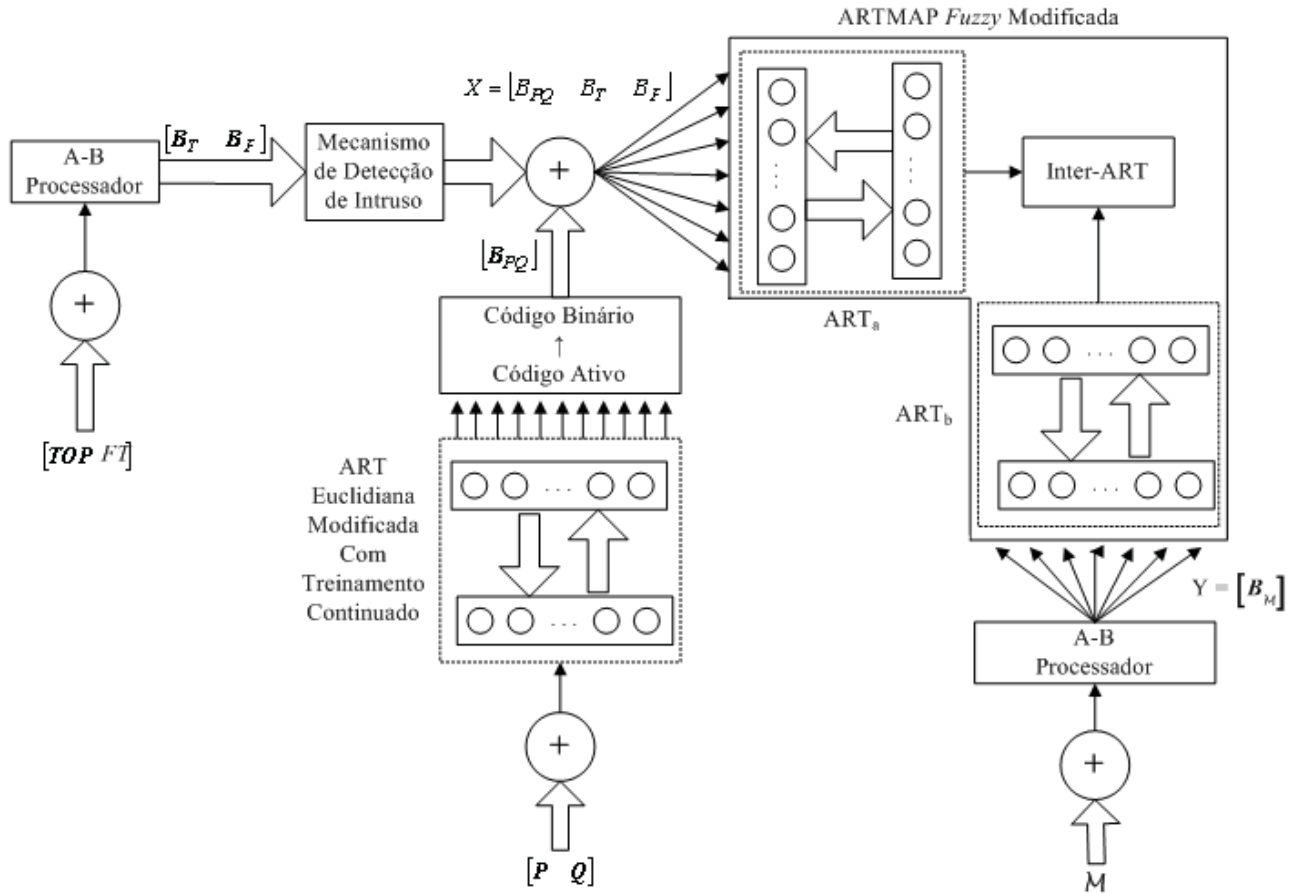


Figura 2.1: Rede Neural ART-ARTMAP *Fuzzy* Euclidiana Modificada com Treinamento Contínuo.

como sendo igual ao número de vetores padrões, tem-se, inicialmente somente 1 neurônio na camada competitiva. Deste modo, toda vez que um novo padrão é processado, constata-se a necessidade de criar um novo *cluster* (agrupamento), um novo neurônio é criado e assim, o número de neurônios vai crescendo à medida em que o conhecimento torna-se mais complexo. Este procedimento é semelhante ao funcionamento do cérebro humano. Usamos não mais do que 5 ou 6% dos cerca de 100 bilhões de neurônios. Toda vez que estimularmos nosso cérebro com um conhecimento recente, um novo neurônio será usado para “codificá-lo” deixando-o de ser inativo. Este procedimento emprega, basicamente, um neurônio dedicado para cada tarefa específica (reconhecimento). O reconhecimento ou não-reconhecimento é ultimado pelo princípio da ressonância, que é algo muito comum no nosso cotidiano, ou seja, por exemplo, a sintonia de uma emissora de rádio, emissor de televisão, o controle remoto, etc. Este dispositivo do neurônio dedicado, a princípio, pode ser interpretado como fragilidade do ponto vista da confiabilidade, *i.e.*, se o neurônio vier a ser danificado, perde-se a informação armazenada. Contudo, no cérebro, a confiabilidade é maximizada via um sistema que verifica a integridade dos neurônios (principalmente durante o

sono). Caso seja constatada alguma “deficiência” em um neurônio em particular, as informações contidas são transferidas para um outro neurônio inativo, passando-o à atividade.

Por conseguinte, cada neurônio da rede neural artificial ARTMAP *Fuzzy* Modificada terá a responsabilidade de organizar e armazenar as informações pertinentes a cada agrupamento.

Já a rede ART Euclidiana Modificada com Treinamento Continuado possui apenas 1 neurônio “temporário” na camada competitiva, isto ocorre devido à detecção de anomalia. Na detecção de anomalia a rede verifica se a informação contida na entrada é importante a ponto de ser incorporada à rede ou trata-se apenas de uma anomalia devendo, portanto, ser descartada. Assim a transformação de um neurônio temporário em um neurônio definitivo só ocorre quando o número de entradas incorporadas a um neurônio temporário ultrapassar um número mínimo pré-determinado. Após o neurônio se tornar definitivo a atualização dos dados nele contidos só ocorrerá se a distância entre a nova entrada e o centro do *cluster* que a representa for superior ao índice de novidade. Uma diferença fundamental desta rede em relação às redes ART tradicionais, onde o treinamento corresponde a uma fase e a análise a outra, é que ambos ocorrem simultaneamente, isto devido à tratar-se de uma rede com treinamento continuado.

É importante ressaltar que, durante todo o processo, quando for identificada uma configuração “estranha” ao conjunto de configurações que já fazem parte da rede, ou seja, quando for identificada uma “intrusão”, realiza-se o treinamento em relação a esta configuração com um número mínimo de dados para que este venha a ser incorporado definitivamente à rede. Este procedimento é designado, nesta pesquisa, de treinamento via detecção de intrusão e é realizado considerando-se um conjunto de vetores padrões envolvendo somente a referida configuração e vários níveis de carga e de geração. A identificação de uma configuração “estranha” é observada calculando-se a distância de *Hamming* entre a configuração sob análise e as demais configurações constantes na base de dados. Se a distância de **Hamming** for diferente de zero, conclui-se que a referida configuração não faz parte da base de conhecimento da rede neural. Como, por definição, a distância entre dois pontos é nula se, e somente se, esses pontos são iguais, independentemente do tipo de distância utilizada, por comodidade, será utilizada a distância euclidiana. Um outro fato importante a se ressaltar, o porquê da escolha do método “detecção de intrusão” ao invés de “detecção de novidade”. Na detecção de novidade existe um índice que controla a distância entre a configuração de entrada e as já pertencentes a rede possuindo, assim, uma tolerância, uma margem positiva; a detecção de intrusão é mais rígida, qualquer configuração diferente, por mais próxima das já existentes, é identificada, aqui a distância deve ser nula.

2.2 Justificativas

Nesta pesquisa, primeiramente foi definido como se trabalharia com os dados da potência elétrica ativa e da potência elétrica reativa, isto porque são dados extensos e, que precisam ser diferenciados quando multiplicados por constante. Essa foi uma das razões mais importantes para se adotar uma rede ART Euclidiana Modificada para classificá-los pois, como não requer a normalização, este tipo de diferenciação acontece. Além disso, não há a necessidade de complementação dos dados. Neste caso, o gasto computacional é consideravelmente menor.

O treinamento continuado foi introduzido apenas na rede ART Euclidiana Modificada em decorrência de que, após a classificação da potência ativa e reativa, a única “novidade” real é com relação às configurações e contingências, que já possuem um módulo de detecção de intrusão no início do treinamento. Além disso, seria necessária outra técnica para fazer o treinamento continuado se esses dados fossem incorporados à rede ART Euclidiana Modificada. Quanto a margem de segurança, o método desenvolvido por [54] já define, de acordo com o número de *bits* a ser utilizado, o número total de classes de saída do módulo ARTb.

Como foi empregado o método introduzido por [54], além da facilidade de se trabalhar com os dados das configurações e das contingências em números binários, a escolha por uma rede neural ARTMAP *Fuzzy* Modificada foi natural. Mesmo porque, com os dados da configuração e contingência binarizados se torna mais fácil sua manipulação e, após a classificação realizada pela rede ART Euclidiana Modificada com Treinamento Continuado, os dados da potência ativa e reativa se tornam códigos ativos, que é um número inteiro positivo, que também são facilmente binarizados.

Análise da Estabilidade Transitória

A estabilidade é a característica do sistema elétrico de potência, ou parte do sistema, que lhe permite desenvolver em seus elementos forças restauradoras iguais ou maiores que as forças perturbadoras, as quais permitem estabelecer um estado de equilíbrio entre os elementos. O limite de estabilidade é o fluxo máximo de potência que pode passar em determinado ponto do sistema, quando todo ele ou parte, a que se refere o limite de estabilidade, está operando de maneira estável [59].

Nos sistemas elétricos de potência, existem limites de estabilidade para regimes permanentes e transitórios. O primeiro é atingido quando o fluxo máximo de potência possível em um determinado ponto do sistema, sem perda de estabilidade, for atingido de forma gradual. Já o segundo, quando o fluxo máximo de potência possível em um determinado ponto, também sem perda de estabilidade, for atingido mas de forma brusca, após uma perturbação do sistema.

Em geral, o limite da estabilidade transitória de um sistema é menor que o limite de estabilidade em regime permanente. Portanto, para que o sistema opere dentro de seus limites de estabilidade, é necessário que seja mantido com margens as quais permitam que a ocorrência

de perturbações causadas por falhas e suas eliminações não levem a perda de estabilidade do sistema.

3.1 Modelo do Sistema

Considere um sistema de potência em ponto de operação estável, isto é, todas as máquinas síncronas possuem a mesma velocidade angular constante, ω_0 (1pu ou velocidade síncrona), e o total das potências elétricas geradas nas máquina é exatamente igual à soma das potências absorvidas nas cargas mais as perdas nas linhas. Suponha que os ângulos das forças eletromotrizes das máquinas sejam δ'_0s .

Suponha que no tempo $t = t_0$ ocorra um grande distúrbio, com a perda de uma linha ou de um curto-circuito em alguma barra ou linha. Em consequência da mudança topológica, o distúrbio causará um desequilíbrio de potência. Surge um excesso ou déficit de energia nas máquinas, que ocasionará a aceleração ou desaceleração de seus rotores. Consequentemente, os ângulos das máquinas serão alterados no tempo, na tentativa de restabelecer o novo equilíbrio de potência.

Se um gerador acelera em relação a um outro, a posição angular do seu rotor em relação ao da máquina mais lenta avançará. Esta diferença angular faz com que parte da carga do gerador mais lento se transfira para o gerador mais rápido, dependendo da relação potência-ângulo e esta função é não-linear. Este fenômeno tende a reduzir a diferença de velocidade entre os dois geradores, consequentemente, a diferença angular também será reduzida. A relação potência-ângulo, após um certo limite, para o aumento na separação angular não é mais acompanhado por uma variação de potência transferida, denominada potência sincronizante. Isto faz com que a separação angular entre as máquinas aumente ainda mais acarretando o fenômeno da instabilidade, ou perda de sincronismo. Em qualquer situação, a estabilidade do sistema dependerá da existência de potências restauradoras suficientes após uma perturbação.

Considerando o sistema dinamicamente estável, se o impacto for pequeno, o sistema pode, por si só encontrar um novo ponto de operação estabelecendo o balanço de potência para uma nova configuração de ângulos δ'_0s . Quando o distúrbio for mais significativo, o sistema torna-se instável, ou seja, não encontra um estado no qual ocorre o equilíbrio de potência. Nessa situação, é necessário atuar no sistema, podendo-se até realizar corte de algumas cargas. Quando isto ocorre, o sistema passa a ter uma nova configuração topológica e se busca saber se o excesso de energia cinética, adquirido durante o defeito pelos rotores das máquinas, pode ser reabsorvido

pelo novo sistema, de forma que a estabilidade de ângulos e velocidades das máquinas sejam mantidas.

Quando um gerador síncrono perde o sincronismo do resto do sistema, seu rotor gira numa velocidade maior (ou menor) do que é necessária para gerar energia na frequência do sistema. O desacoplamento entre a velocidade do campo do estator de uma máquina, que possui uma frequência de rotação imposta pelos outros geradores do sistema, e a velocidade do campo do seu rotor resulta em grandes flutuações na potência de saída desta máquina, nas correntes e nas tensões, fazendo com que o sistema de proteção isole a máquina do resto do sistema. A perda de sincronismo pode ocorrer entre um gerador e o resto do sistema ou entre grupos de geradores. Neste último caso, o sincronismo poderá ser mantido nos sistemas isolados.

Se após a eliminação do defeito o sistema encontrar um ponto de operação estável, este será dito transitoriamente estável. A atuação no sistema de forma a isolar o defeito deve ser feita rapidamente para que se garanta a estabilidade. O tempo máximo em que o isolamento de defeito deve ser feito de tal maneira que o sistema permaneça estável é conhecido como tempo crítico de abertura, t_{cr} . Assim, a eliminação do defeito antes do tempo crítico proporciona um sistema estável e se a eliminação do defeito ocorrer após o tempo crítico, o sistema se torna instável.

3.1.1 Descrição do Modelo do Sistema

Para que se possa estudar a estabilidade de um sistema, é necessário que este seja convenientemente descrito através de equações diferenciais. As equações diferenciais que descrevem o comportamento dinâmico de cada uma das máquinas do sistema podem ser obtidas através de seus balanços de potência. A máquina é acionada por um elemento primário que lhe fornece potência mecânica. Uma grande parte desta potência mecânica é convertida em energia elétrica e transferida à rede, o restante transforma-se em potência de aceleração do rotor da máquina.

Considere assim um sistema composto de ng máquinas síncronas, sendo que P_{T_i} é a potência mecânica do eixo da turbina e P_{G_i} é a potência elétrica gerada. Se estas duas potências forem iguais, desprezadas as perdas, o gerador estará funcionando com velocidade síncrona constante. Caso contrário, se existir uma diferença, esta diferença será usada para mudar a energia cinética ou a velocidade e dominar o conjugado de amortecimento nos enrolamentos de amortecimento.

A equação matemática, de acordo com as referências, pode ser expressa como [13, 35]:

$$P_{T_i} - P_{G_i} = \frac{d}{dt}(W_{cin,i}) + P_{d_i}, \quad (3.1.1)$$

em que:

- P_{d_i} é a potência referente às perdas;
- $W_{cin,i}$ é a energia cinética total do gerador mais turbina, dada em megawatts-segundo (MVs) ou megajoules (MJ), esta é dada pela equação:

$$W_{cin,i} = \left(\frac{f_i}{f^0} \right)^2 W_{cin,i}^0, \quad (3.1.2)$$

com:

- $W_{cin,i}^0$ é a energia cinética da i -ésima máquina síncrona medida na frequência nominal;
- f^0 é a frequência nominal do sistema;
- f_i é a frequência instantânea da i -ésima máquina síncrona.

A frequência f_i pode ser expressa considerando-se as variações em torno da frequência nominal, do seguinte modo [13]:

$$f_i = f^0 + \Delta f_i. \quad (3.1.3)$$

Assim:

$$\begin{aligned} W_{cin,i} &= \left(\frac{f^0 + \Delta f_i}{f^0} \right)^2 W_{cin,i}^0 \\ &= \left\{ 1 + 2 \frac{\Delta f_i}{f^0} + \left(\frac{\Delta f_i}{f^0} \right)^2 \right\} W_{cin,i}^0. \end{aligned} \quad (3.1.4)$$

Como as variações de frequência Δf_i são pequenas, pode-se usar a seguinte aproximação [13]:

$$W_{cin,i} \approx \left(1 + 2 \frac{\Delta f_i}{f^0} \right) W_{cin,i}^0. \quad (3.1.5)$$

A derivada em relação ao tempo da energia cinética total da i -ésima máquina síncrona pode ser expressa por:

$$\frac{d}{dt}(W_{cin,i}) = \left(\frac{2W_{cin,i}^0}{f^0} \right) \frac{d}{dt}\Delta f_i. \quad (3.1.6)$$

e, como:

$$\Delta f_i = \frac{1}{2\pi} \frac{d}{dt}(\Delta \delta_i), \quad (3.1.7)$$

então:

$$\frac{d}{dt}\Delta f_i = \frac{d}{dt} \left\{ \frac{1}{2\pi} \frac{d}{dt}(\Delta \delta_i) \right\} = \frac{1}{2\pi} \frac{d^2}{dt^2}(\Delta \delta_i). \quad (3.1.8)$$

em que:

- δ_i é a posição angular do rotor, medida em *radianos elétricos*, em relação a uma referência que gira de modo síncrono.

Substituindo (3.1.8) em (3.1.6), obtém-se:

$$\frac{d}{dt}(W_{cin,i}) = \left(\frac{W_{cin,i}^0}{\pi f^0} \right) \frac{d^2}{dt^2} \Delta \delta_i. \quad (3.1.9)$$

Deve-se observar que, havendo desvios da velocidade do rotor em relação à velocidade síncrona, haverá a indução de correntes nos enrolamentos de amortecimento do rotor, as quais causam forças que tendem a impedir os movimentos. Esse conjugado tende a crescer com velocidade relativa $\frac{d\delta_i}{dt}$, de tal forma que [13]:

$$P_{d_i} \approx D_i \frac{d\delta_i}{dt}, \quad (3.1.10)$$

onde D_i é um parâmetro positivo da máquina, medido em megawatt por radiano elétrico por segundo, denominado de constante de amortecimento do rotor.

Considerando as equações (3.1.9) e (3.1.10), a equação (3.1.1) pode ser escrita da seguinte forma [13]:

$$P_{T_i} - P_{G_i} = \left(\frac{W_{cin,i}^0}{\pi f^0} \right) \frac{d^2 \delta_i}{dt^2} + D_i \frac{d\delta_i}{dt}. \quad (3.1.11)$$

Colocando a equação (3.1.11) em pu (por unidade), uma vez que é dada em MW, cuja base adotada é a potência nominal da i -ésima máquina síncrona, tem-se [13]:

$$P_{m_i} - P_{e_i} = \frac{H_i}{\pi f^0} \frac{d^2 \delta_i}{dt^2} + D_i \frac{d\delta_i}{dt}, \quad (3.1.12)$$

em que:

- H_i é a constante de inércia em pu definida por:

$$H_i \triangleq \frac{W_{cin,i}^0}{Pr_i}; \quad (3.1.13)$$

- P_{m_i} é a potência mecânica de entrada (pu) dada por:

$$P_{m_i} = \frac{P_{T_i}}{Pr_i}; \quad (3.1.14)$$

- P_{e_i} é a potência elétrica de saída (pu) definida por:

$$P_{e_i} = \frac{P_{G_i}}{Pr_i};$$

- Pr_i é a potência nominal total da i -ésima máquina síncrona.

3.1.2 Modelo do Sistema Referido ao Centro de Ângulos

O comportamento dinâmico da i -ésima máquina pode ser descrito pela seguinte equação diferencial (equação de oscilação da máquina síncrona) [1, 50]:

$$M_i \frac{d^2 \theta_i}{dt^2} - P_i(\theta) = 0, \quad i \in NG, \quad (3.1.15)$$

com:

$$P_i(\theta) = Pm_i - Pe_i - \frac{M_i PCOA}{MT} \quad (3.1.16)$$

e

$$M_i = \frac{2H_i}{\omega_s},$$

em que:

- ω_s é a velocidade síncrona (rad. elét./s) dada por:

$$\omega_s = 2\pi f_0;$$

- θ_i é o ângulo do rotor da i -ésima máquina síncrona referida ao centro de ângulos (CA) (rad. elét.) que é dada por:

$$\theta_i = \delta_i - \delta_0;$$

- δ_i é o ângulo do rotor da i -ésima máquina síncrona medida com relação a uma máquina que gira à velocidade síncrona (rad. elét.);

- $\delta_0 = \sum_{j \in NG} M_j \delta_j;$

- $PCOA$ é a potência acelerante do CA que é dada por:

$$PCOA = \sum_{j \in NG} (Pm_j - Pe_j);$$

- $MT = \sum_{j \in NG} M_j$;
- NG é o conjunto de índices das máquinas síncronas que compõem o sistema:

$$NG = \{1, 2, 3, \dots, ng\};$$

- ng é o número de máquinas síncronas.

A potência elétrica (Pe) usada na equação (3.1.16), considerando-se as simplificações introduzidas no modelo clássico, pode ser calculada de várias formas. Uma delas e a mais comum consiste em realizar o cálculo, a partir da matriz de admitância reduzida às barras internas de geração, como proposto em [1, 50]. Trata-se de um procedimento rápido do cálculo da potência elétrica, principalmente, quando esta se refere à condição de defeito (período de permanência do curto-circuito) e pós-defeito (eliminação do defeito com saída da linha de transmissão), empregando o método proposto na referência [43].

O modelo do sistema referido ao centro de ângulos (Equação (3.1.15)) apresenta as seguintes propriedades:

Propriedade 1. A potência acelerante do centro de ângulo, quando calculada no ponto de equilíbrio estável é nula [42, 50].

Propriedade 2. O somatório do produto da posição angular θ_j pela inércia M_j de cada máquina síncrona do sistema é igual a zero, ou seja [42, 50]:

$$\sum_{j \in NG} M_j \theta_j = 0. \quad (3.1.17)$$

Propriedade 3. O somatório do produto da velocidade angular ω pela inércia de cada máquina síncrona do sistema é igual a zero, ou seja [42, 50]:

$$\sum_{j \in NG} M_j \omega_j = 0. \quad (3.1.18)$$

A Propriedade 1 é estabelecida a partir da formulação do problema do fluxo de potência que é determinado fazendo-se ($P_m - P_e = 0$) para todas as máquinas que compõem o sistema. Isto representa a obtenção de um ponto de operação (equilíbrio). Deste modo, conclui-se que a $PCOA$ é nula quando avaliada em um ponto de operação estável do sistema. A $PCOA$ será diferente de zero sempre que houver deslocamento do estado em torno do ponto de equilíbrio, ou seja, durante as oscilações eletromecânicas.

Da Propriedade 2 conclui-se que os ângulos θ'_i s são linearmente dependentes. O mesmo ocorre com as velocidades ω'_i s (Propriedade 3). Por conseguinte, o conhecimento de $(ng - 2)$ variáveis de estado permite o cálculo das duas variáveis restantes utilizando-se as equações (3.1.17) e (3.1.18).

Outras definições e termos utilizados em problemas de estabilidade de sistemas de energia elétrica podem ser encontrados na referência [28]. Trata-se de um importante texto de referência nesta área do conhecimento.

3.2 Margem de Segurança

A análise da estabilidade transitória de Sistemas de Energia Elétrica pode ser realizada com base no conceito de energia e margem de segurança (método direto) [2, 18, 19, 50]. Considerando-se uma contingência de índice r , tem-se [2, 18, 19, 50]:

$$\mathcal{M}_r = \frac{E_{crit_r} - E_{e_r}}{E_{crit_r}} \quad (3.2.1)$$

em que:

- E_{crit_r} é a energia total crítica do sistema;
- E_{e_r} é a energia total do sistema avaliada no instante de eliminação do defeito (t_e).

A margem de segurança pode ser interpretada como uma “medida de distância” em relação à situação de instabilidade do sistema [20]. A estabilidade do sistema para a r -ésima contingência pode ser avaliada mediante o emprego do conceito da margem de segurança, através do seguinte critério [2, 18, 20, 42, 43]:

- $\mathcal{M}_r < 0$, o sistema é considerado *instável* do ponto de vista de estabilidade transitória;
- $\mathcal{M}_r \geq 0$, o sistema é considerado *estável* do ponto de vista de estabilidade transitória;

Ressalta-se que, quando são consideradas as condutâncias de transferência, a expressão (3.1.16) não se caracteriza como força potencial [42], conseqüentemente, não é possível determinar uma função energia potencial do sistema – representada pela primeira integral do movimento – que se constitui dependente do caminho de integração descaracterizando-a como energia potencial. Porém, por abuso de linguagem, nesta pesquisa o termo energia total do sistema será utilizado sempre que houver referência à função tipo energia. Por conseguinte, a energia total relativa ao sistema (3.1.15) é dada por [2, 50, 18, 42, 20, 43]:

$$E(\theta, \omega) = E_c(\omega) + E_p(\theta) \quad (3.2.2)$$

em que:

- $E_c(\omega)$ é a energia cinética

$$E_c(\omega) = \frac{1}{2} \sum_{i \in NG} M_i \omega_i^2; \quad (3.2.3)$$

- $E_p(\theta)$ é a energia potencial

$$E_p(\theta) = - \sum_{i \in NG} \int_{\theta_i^p}^{\theta_i} P_i(\theta) d\theta_i; \quad (3.2.4)$$

O sobrescrito (p) refere-se ao ponto de equilíbrio pós-defeito e $P_i(\theta)$ é dada pela Equação (3.1.16).

3.3 Análise de Segurança

Os Centros de Supervisão e Controle defrontam-se diariamente com perturbações na rede elétrica. Estas perturbações são de caráter imprevisível e podem ter repercussões graves no nível de estabilidade do sistema e, conseqüentemente, no nível do cliente final. E, como este último torna-se cada vez mais exigente, é necessário que sejam desenvolvidos processos eficientes de análise e de reação a incidentes, que por uma conjunção de fatores não estejam previstos nos critérios de segurança estabelecido. A operação de um sistema de energia elétrica requer que decisões sejam tomadas nos Centros de Supervisão e Controle para garantir o atendimento da demanda com qualidade, de maneira confiável e segura.

Esse sistema, em tempo real, processa sequencialmente dados colhidos da rede elétrica de forma periódica e automática por meio de funções de análise de rede. Considerando que, parte

dos controles desempenhados são realizados no modo preventivo, e que, a análise de contingências representa um esforço computacional muito grande, constata-se aí uma séria restrição de tempo. As funções relacionadas à segurança estática são baseadas em análise de contingências que efetua simulações de uma lista predefinida de prováveis distúrbios: saídas de linhas de transmissão, geradores e compensadores *shunts*. A análise de contingência é normalmente processada utilizando o caso base obtido pelo estimador de estado que é responsável pela filtragem dos dados. O objetivo é manter o sistema elétrico operando em condições de segurança mesmo na eventual ocorrência do mais severo distúrbio.

Um sistema de energia elétrica está operando de forma “segura” quando é capaz de atender à carga sem violar limites operativos preestabelecidos, mesmo se submetido a contingências imprevisíveis. Portanto, o conceito de segurança está intimamente ligado à ideia de sobrevivência após uma perturbação significativa. Chama-se de “análise de segurança” ao conjunto de procedimentos realizados para determinar se, e em que grau, um sistema elétrico de potência é seguro em relação a uma lista de contingências. Se pelo menos uma contingência acarretar violação nos limites operativos, o sistema será considerado inseguro. Somente tem sentido analisar a segurança de sistemas que estejam em “estado normal”, ou seja, toda a carga está atendida e nenhum equipamento está sobrecarregado. Portanto, a primeira tarefa de um processo de análise de segurança é identificar se o sistema está em estado normal ou estado de emergência (monitoramento de segurança). Se o sistema estiver no estado normal, a análise de segurança determina se o sistema está “seguro” ou “inseguro”, com respeito a um conjunto de contingências possíveis. Se estiver “inseguro”, ou seja, se existir pelo menos uma contingência que possa causar violações, devem ser encontradas ações de controle preventivas para levar o sistema a um estado seguro (corretivamente seguro).

Numa rede elétrica é necessário se assegurar condições de estabilidade — quer em regime estacionário quer em regime transitório — sem as quais o serviço essencial que se fornece aos clientes pode ser gravemente prejudicado.

Neste momento, existe um aumento das exigências em termos de qualidade e, como é sabido, podem ocorrer perturbações no sistema elétrico com uma imprevisibilidade tal que ultrapassem largamente todos os critérios de segurança usados pelos Centros de Supervisão e Controle. Essas ocorrências podem ser causadas por diversos fatores, por exemplo [46]:

Atmosféricos: descargas elétricas que afetam “simultaneamente” vários elementos de rede; A combinação nevoeiro mais poluição numa determinada área, ou ainda, chuva, inundação, granizo ou vento forte;

Incêndios: incêndios de enormes proporções que afetam elementos da rede (podem ser as linhas de transmissão ou a confluência de diversos elementos nas proximidades de uma subestação);

Erros Humanos: em termos de manobras, trabalhos de conservação, montagem e ensaios;

Falhas no equipamento: atuação incorreta dos sistemas de proteção ou dos sistemas de comando e controle.

Assim, define-se:

Definição 3.1. (Perturbação (ou distúrbio)) *Define-se perturbação como sendo qualquer ação que venha a provocar a alteração no estado de equilíbrio do sistema de energia elétrica.*

Assim, as ocorrências acima são exemplos de perturbações num sistema de energia elétrica.

Um outro conceito que falta formalizarmos é o de contingências:

Definição 3.2. (Contingências) *Definem-se contingências como sendo as consequências das perturbações.*

Assim, algumas contingências são:

- saída forçada de operação de equipamentos elétricos;
- curto-circuito causado por descargas atmosféricas, ação mecânica, etc.

Observação 3.3. Contingências ou defeitos são de natureza endêmica.

A análise de segurança tem sido classificada como “estática” e “dinâmica”. Na análise de segurança estática as restrições operativas que se deseja monitorar referem-se à sobrecarga em equipamentos (transformadores e linhas de transmissão) e verificação de limites de tensão em barramentos. Supõe-se que o sistema elétrico atinja um novo ponto de operação em regime permanente após ter sido submetido a uma perturbação (desligamento de linha, por exemplo). A ferramenta computacional básica utilizada na análise de segurança estática é um programa de fluxo de carga.

A análise de segurança dinâmica de um sistema de energia elétrica, composto por uma rede interligada que inclui transmissão e geração, é uma tarefa que exige grande esforço computacional. E este tende a crescer a medida que se aumenta a dimensão do sistema em estudo, o número de interconexões, ou quando se deseja uma operação mais precisa. Construir melhores centros de controle representa um grande desafio para a indústria associada a geração, transmissão e distribuição de energia elétrica.

3.3.1 Estados de Operação da Rede elétrica

A análise de segurança verifica a capacidade da rede em resistir às contingências, consistindo basicamente na avaliação do impacto das contingências sobre o estado de operação da rede e na determinação de ações de controle preventivo e/ou corretivo para situações de emergência [46].

Os estados de operação da rede são [60]:

Seguro: Neste estado são obedecidos os três conjuntos de restrições: carga, operação e segurança.

Desta forma toda carga é atendida, não há violações, mesmo que ocorram contingências.

Seguro-Corretivo: Toda carga é atendida. No caso de haver violações causadas por contingências, estas podem ser eliminadas por ações de controle apropriadas, sem ser necessário o corte de carga.

Alerta: Toda carga é atendida, deste modo são obedecidas apenas as restrições de carga e operação. No caso de haver violações causadas por contingências, estas só podem ser eliminadas com o corte de carga.

Emergência-Corretivo: Toda carga é atendida, entretanto os limites de operação são violados, estes podem ser eliminados por ações de controle apropriadas, sem ser necessário o corte de carga.

Emergência-Crise: Tem-se toda carga atendida e os limites de operação violados, entretanto estes só podem ser eliminados com corte de carga.

Restaurativo: Não há violações, mas houve corte de carga. Este estado é atingido quando uma emergência é eliminada por desligamento manual ou automático de partes do sistema, efetuado pelo centro de controle ou por dispositivos locais.

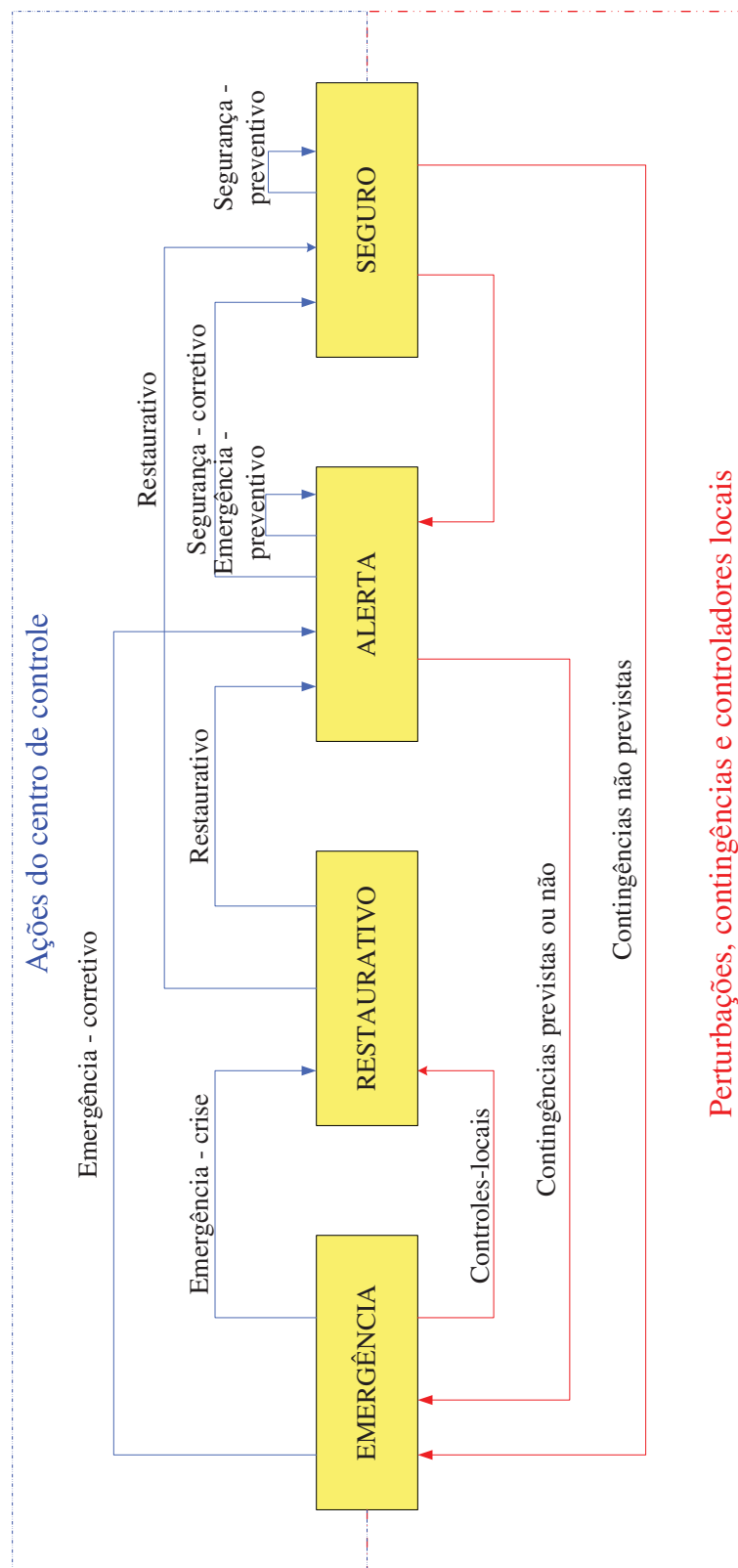


Figura 3.1: Estados de Operação.

Teoria da Ressonância Adaptativa

Uma das mais importantes características das redes neurais artificiais é sua habilidade de generalização, i.e., produzir respostas para padrões de entrada que são similares, mas não idênticos, aos padrões apresentados a rede durante o seu treinamento. Entretanto, o conhecimento a respeito de determinado domínio do mundo real pode se modificar ao longo do tempo, i.e., os padrões de entrada de uma rede podem se modificar para acomodar mudanças ocorridas. Assim, para um número de aplicações, o desempenho da rede pode decair gradativamente à medida que o tempo passa, uma vez que os pesos definidos na fase de treinamento não refletem as novas situações. Então, é necessário um modelo de rede neural incremental, que não precise recomençar seu treinamento do ponto inicial a cada vez que aparecem novos padrões de entrada, e que preserve o conhecimento adquirido antes. Para esta tarefa, apresentam-se as redes ART e ARTMAP que serão descritas neste capítulo.

4.1 Rede Neural ART

Durante décadas, estatísticos aplicaram a Análise Discriminante e Modelos de Regressão na modelagem de padrões nos quais os dados de treinamento disponíveis são rotulados (com entradas e saídas conhecidas) e técnicas de *clustering* quando estes dados não são rotulados. Estes métodos encontram analogias em redes neurais, onde topologias multicamadas com algoritmos de retropropagação do erro são usados quando os dados são rotulados e modelos auto-organizados quando não são rotulados [57]. Essas técnicas de *clustering* têm sido usadas no agrupamento e categorização de dados baseados nas estruturas naturais contidas nestes. O objetivo de um algoritmo de *clustering* apropriado é maximizar o grau de similaridade entre os padrões de um mesmo *cluster*, ao mesmo tempo em que o grau de similaridade entre os diversos *clusters* é minimizado.

Para os problemas relacionados à classificação de padrões sob os quais as informações fornecidas residem apenas nos dados relativos a estes problemas (dados de entrada), sem que exista algum ponto crítico externo que possa orientar os resultados obtidos, seria interessante que um modelo matemático, em particular, um modelo de Inteligência Artificial, elaborasse uma classificação própria baseada na exploração das redundâncias contidas nos dados de entrada disponíveis à rede [62].

Desta forma, é necessária a aplicação de técnicas que não se apoiem em supervisão de erro das saídas como forma de se obter classes agrupadas de acordo com um determinado estado em que se encontram os elementos, tornando possível detectar relacionamentos entre eles. Assim, pode-se gerar matematicamente estruturas de relacionamentos entre vários estados que coexistem sob um mesmo cenário.

Uma arquitetura adequada a este tipo de tarefa deveria identificar padrões através da comparação dos mesmos e, posteriormente, elaborar um critério para agrupamento destes padrões. Existem diversos modelos neurais que atendem a estes requisitos. Contudo, nesta pesquisa é proposto o uso do modelo neural auto-associativo baseado na Teoria da Ressonância Adaptativa [6], ou ART. As redes neurais ART englobam uma variedade ampla de redes neurais baseadas explicitamente na neurofisiologia, e são definidas em relação ao seu algoritmo de treinamento em termos de equações diferenciais detalhadas e críveis como modelos plausíveis dos neurônios biológicos. Na prática, as redes ART são implementadas como soluções analíticas ou aproximações para estas equações diferenciais. A ressonância adaptativa, sobre qual a teoria se refere, ocorre quando padrões de atividade nas camadas de entrada e saída se reforçam em sinergia.

Uma característica fundamental na Teoria de Ressonância Adaptativa, que fundamenta o funcionamento das redes ART, consiste em que se nenhuma classe ressoa com o padrão de entrada apresentado à rede, uma nova classe é criada, de acordo com um parâmetro de vigilância previamente escolhido e que pode ser calibrado. Desta forma, permitem ao usuário controlar o grau de similaridade de padrões associados a uma mesma categoria. Tal característica foi concebida com o intuito de solucionar o dilema ***plasticidade-estabilidade***, denominação utilizada acerca do problema de *trade-off* presente em redes neurais artificiais em se manter capaz de adaptar ou agrupar padrões de entrada indefinidamente (plasticidade) e ao mesmo tempo preservar o conhecimento adquirido, impedindo que os novos padrões se superponham ao conhecimento prévio adquirido pela rede (estabilidade), conforme descrito em [6]. Portanto, esta arquitetura é conduzida por um algoritmo que utiliza um critério racional na alocação de padrões, através de seus vetores de entrada, em direção à geração de agrupamentos baseados na similaridade entre os padrões apresentados.

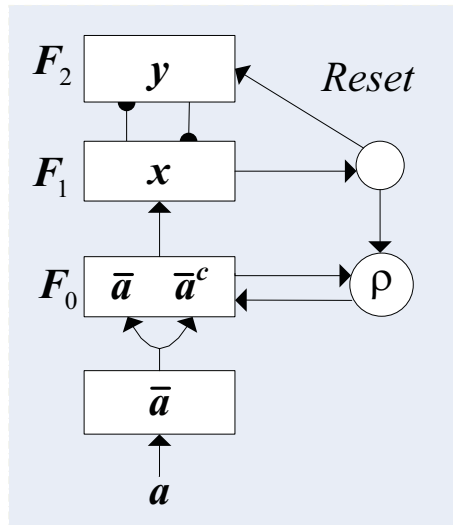


Figura 4.1: Mapa de Campo de uma rede ART.

A arquitetura básica de uma rede ART consiste em três grupos de unidades (Figura 4.1): um campo para processamento de entrada (camada F_1), as unidades *cluster* (camada F_2), e um mecanismo de reinicialização, responsável por controlar o grau de similaridade dos padrões associados a um mesmo *cluster*. A camada F_1 é composta de duas partes: entrada e interface. A porção interface combina sinais da porção entrada e da camada F_2 , para uso na comparação da similaridade entre o sinal de entrada e o vetor de pesos da unidade *cluster* candidata.

A camada F_0 de uma rede ART é destinada à leitura dos vetores de entrada a , ou seja, trata-se de uma camada de entrada. Estes neurônios podem ser considerados como neurônios au-

xiliares que pré-processam as entradas através de uma combinação de operações de normalização e complementação dos dados. Este procedimento faz com que todos os vetores de entrada tenham a mesma magnitude. Esta é a primeira fase do processo de treinamento de uma rede ART qualquer, chamada **etapa de codificação**.

Em seguida, estes valores são enviados a camada F_1 . Neste módulo é realizada a interface entre estes vetores e a própria rede (F_1). São combinados sinais recebidos tanto da leitura dos vetores com os vetores de peso para efeito de comparação da similaridade entre o sinal de entrada com o vetor de pesos do neurônio selecionado como candidato ao aprendizado. Isto torna o modelo ART adequado ao processamento e classificação de dados em categorias baseadas em suas similaridades [6, 7]. Essa é a segunda fase do processo de treinamento de uma rede ART qualquer, chamada **etapa de reconhecimento**.

Para efeito de controle da similaridade dos padrões de entrada alocados em uma mesma categoria existem dois grupos de conexões, cada um com suas próprias ponderações, entre cada neurônio de F_1 e de F_2 . Cada neurônio da camada F_1 se conecta a todos os neurônios da camada F_2 através de conexões ascendentes: a ponderação relativa a cada conexão ascendente relacionada ao i -ésimo neurônio F_1 ligada ao j -ésimo neurônio F_2 será chamado w_{ij} . Por sua vez, cada neurônio da camada F_2 se conecta a todos os neurônios da camada F_1 através de conexões descendentes, logo depois que ocorre a ressonância dos pesos, deste modo, a ponderação relativa a cada conexão descendente relacionada ao j -ésimo neurônio de F_2 ligada ao i -ésimo neurônio F_1 também será chamada w_{ij} , só que neste caso os pesos já estão atualizados.

A camada F_2 é uma camada competitiva. A medida em que o neurônio representativo de cada categoria retorne a maior entrada ponderada de acordo com as conexões ascendentes se torna o candidato ao aprendizado daquele padrão de entrada, e a ativação de todos os outros neurônios de F_2 são zeradas. A seguir, os neurônios de F_1 combinam informações oriundas do processamento ocorrido em F_0 e F_2 . Esta é a terceira fase do treinamento de uma rede ART, denominada de **etapa de comparação**.

A efetivação do aprendizado, e conseqüente incorporação do vetor de entrada ao neurônio candidato escolhido em F_2 , depende da similaridade entre o vetor descendente Y e o vetor de entrada. Esta decisão é tomada pela unidade de *reset*, baseado nos sinais recebidos pelas camadas F_1 e F_2 . Se o neurônio candidato não for efetivamente escolhido como representativo da categoria do vetor de entrada, este é inibido e um novo neurônio de F_2 é selecionado como novo candidato ao aprendizado. Esta é a quarta fase do treinamento de uma rede ART chamada **etapa de busca** e acontece até que um neurônio saia vencedor.

As redes neurais baseadas na topologia ART são utilizadas nas mais diversas aplicações relacionadas à classificação, como reconhecimento de dados em questionários, reconhecimento de alvos militares, reconhecimento de caracteres, processamento de sinais [5, 15], previsão de carga, análise de contingências de sistemas de energia elétrica, análise de estabilidade transitória de sistemas de energia elétrica [16, 36, 38, 39, 54, 55, 56].

Existem redes ART dos mais diversos tipos, utilizando paradigmas de aprendizado tanto supervisionado quanto não-supervisionado. Os modelos mais conhecidos e utilizados são:

ART-1: Utiliza treinamento não-supervisionado. Foram os primeiros modelos desenvolvidos.

Sua principal peculiaridade é a utilização de dados binários como entrada da rede [6];

ART-2: Utiliza treinamento não-supervisionado. Modelo desenvolvido para processar dados contínuos podendo também ser utilizada para entrada de dados binários [7];

ART-3: Utiliza treinamento não-supervisionado. Modelo que utiliza a ação de neurotransmissores na propagação dos dados pela rede [8];

ARTMAP: Utiliza treinamento supervisionado. Composto por duas sub-redes: ART_a , que recebe e processa um determinado padrão de entradas apresentado, e ART_b , que constitui a resposta desejada para o padrão apresentado à rede [10];

Fuzzy ART: Utiliza treinamento não-supervisionado. Este tipo de rede ART processa dados *fuzzy*, *i.e.*, sua arquitetura realiza cálculos baseados na lógica nebulosa [9];

Fuzzy ARTMAP: Utiliza treinamento supervisionado. Esta rede ARTMAP processa dados *fuzzy* [11].

4.1.1 Parâmetros de uma Rede ART

Antes da descrição dos passos de inicialização e do algoritmo de uma rede ART, é apresentada uma lista contendo os parâmetros utilizados no algoritmo de treinamento de uma rede ART [6, 7], bem como seus respectivos papéis no modelo.

M : número de unidade de entrada (camadas F_0 e F_1);

N : número de unidades de *cluster* (camada F_2);

β : taxa de treinamento, sendo que $0 \leq \beta \leq 1$; seu valor indica a taxa de treinamento, isto é, um valor baixo para β indica um treinamento lento, enquanto que $\beta = 1$, leva a um treinamento rápido;

α : taxa de aprendizagem, sendo que $\alpha > 0$; seu valor indica o grau em que os novos padrões pertencentes a cada *cluster* terão na atualização de seus pesos no próximo ciclo de treinamento;

ρ : parâmetro de vigilância, sendo que $0 < \rho < 1$; este parâmetro determina quantos *clusters* serão formados, em uma razão direta ao valor adotado, o que significa que este parâmetro define o grau de variação admitido entre os próprios componentes de cada *cluster*.

No procedimento de atualização dos pesos da rede são possíveis duas abordagens operacionais: treinamento sincronizado e treinamento estável. No treinamento sincronizado a rede efetua um passo de propagação a cada inicialização, e no tratamento estável a rede propaga o padrão até obter uma classificação para o mesmo. Assim, pode-se dizer que se for necessário traçar o caminho do padrão através da rede, lança-se mão do treinamento sincronizado. Caso contrário, utiliza-se o treinamento estável, o qual também requer menos tempo para executar a tarefa. Nesta pesquisa, o interesse está apenas em gerar o *cluster*, deste modo será utilizado apenas o treinamento estável.

Uma outra característica importante das redes neurais da família ART é que os vetores são representados como linhas e não como colunas, como é normalmente utilizado na literatura, isto ocorre por ser um modo mais simples de formulação das equações envolvidas no modelo neural.

4.1.2 Algoritmo de uma Rede Neural ART

Como já foi visto, a rede neural ART *Fuzzy* utiliza os cálculos da Teoria dos Conjuntos Nebulosos [70] em sistemas ART. Mas o que isto significa realmente? Uma rede ART convencional utiliza o operador de intersecção $\cap$ enquanto que a rede ART *Fuzzy* utiliza o operador AND Nebuloso $\wedge$. Esta diferença se faz necessária para que se possa trabalhar com dados análogos e binário ao invés de apenas binários. Para entender melhor o funcionamento de uma rede neural ART, apresenta-se detalhadamente tanto o algoritmo de treinamento da rede ART *Fuzzy*, quanto o fluxograma correspondente.

Algoritmo da Rede Neural ART *Fuzzy*

Passo 1: Leitura dos dados de entrada:

Os dados de entrada são denotados pelo vetor:

$$\mathbf{a} = [a_1 \quad a_2 \quad \dots \quad a_M]$$

M -dimensional. Este vetor é normalizado com o intuito de evitar a proliferação de muitas categorias. Assim:

$$\bar{\mathbf{a}} = \frac{\mathbf{a}}{|\mathbf{a}|}, \quad (4.1.1)$$

sendo: $\bar{\mathbf{a}}$ é o vetor normalizado e:

$$|\mathbf{a}| = \sum_i^M a_i. \quad (4.1.2)$$

Passo 2: Codificação do vetor de entrada:

A codificação de complemento é realizada para preservar a amplitude da informação:

$$\bar{a}_i^c = 1 - \bar{a}_i, \quad (4.1.3)$$

em que: $\bar{\mathbf{a}}^c$ é o vetor complementar do vetor de entrada normalizado. Deste modo, o novo vetor de entrada será um vetor $2M$ -dimensional que será denotado por:

$$\mathbf{I} = [\bar{\mathbf{a}} \quad \bar{\mathbf{a}}^c], \quad (4.1.4)$$

ou ainda:

$$\mathbf{I} = [\bar{a}_1 \quad \bar{a}_2 \quad \dots \quad \bar{a}_M \quad \bar{a}_1^c \quad \bar{a}_2^c \quad \dots \quad \bar{a}_M^c].$$

Observe que:

$$|\mathbf{I}| = \sum_i^M \bar{a}_i + \sum_i^M \bar{a}_i^c = \sum_i^M \bar{a}_i + \sum_i^M (1 - \bar{a}_i) = \sum_i^M \bar{a}_i + M - \sum_i^M \bar{a}_i = M,$$

portanto, todos os vetores de entrada após a normalização e codificação terão o mesmo tamanho: M .

Passo 3: Vetor de Atividade:

O vetor de atividade de F_2 é simbolizado por

$$\mathbf{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_N \end{bmatrix},$$

em que: N é o número de categorias criadas em F_2 . Assim, tem-se

$$y_j = \begin{cases} 1, & \text{se o nó } j \text{ de } F_2 \text{ for ativo,} \\ 0, & \text{caso contrário.} \end{cases} \quad (4.1.5)$$

Passo 4: Parâmetros da Rede:

Os parâmetros utilizados no processamento da rede ART nebulosa são:

1. Parâmetro de escolha: $\alpha > 0$;
2. Taxa de Treinamento: $\beta \in [0, 1]$;
3. Parâmetro de Vigilância: $\rho \in (0, 1)$;

Passo 5: Inicialização dos Pesos:

Inicialmente todos os pesos possuem valor igual a 1, ou seja:

$$w_{j1}(0) = w_{j2}(0) = \cdots = w_{j,2M}(0) = 1, \quad (4.1.6)$$

indicando que não existe nenhuma categoria ativa.

Passo 6: Escolha da Categoria:

Dado o vetor de entrada $\mathbf{I}$ em F_1 , para cada nó j em F_2 , a função de escolha T_j é determinada por

$$T_j(\mathbf{I}) = \frac{|\mathbf{I} \wedge \mathbf{w}_j|}{\alpha + |\mathbf{w}_j|}, \quad (4.1.7)$$

em que $\wedge$ é o operador AND nebuloso, definido por:

$$(\mathbf{I} \wedge \mathbf{w}_j) = \min(I_i, w_{ij}). \quad (4.1.8)$$

A categoria é escolhida como sendo o nó J ativo, ou seja:

$$J = \arg \max \{T_j : j = 1, 2, \dots, N\}. \quad (4.1.9)$$

Usando a equação (4.1.9) podemos encontrar mais de uma categoria ativa, a categoria escolhida será aquela que possuir menor índice.

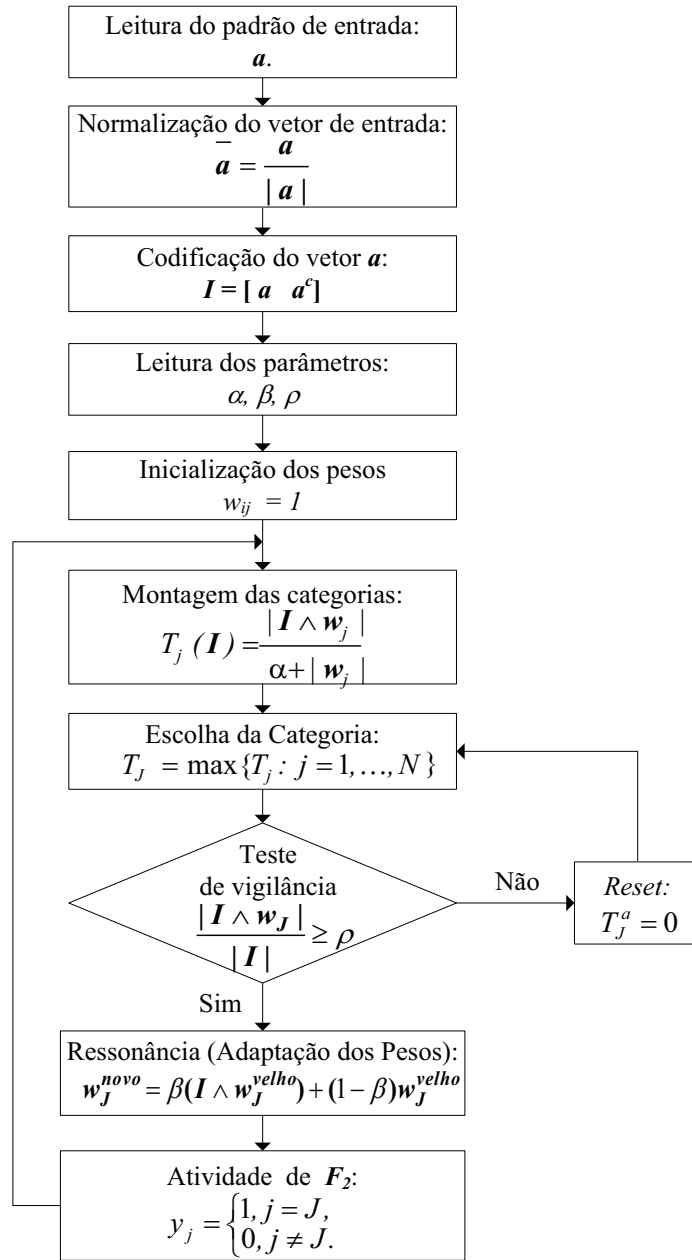


Figura 4.2: Fluxograma de uma Rede ART Nebulosa.

Passo 7: Ressonância ou *Reset*:

A ressonância ocorre se o critério de vigilância (4.1.10) for satisfeito:

$$\frac{|\mathbf{I} \wedge \mathbf{w}_j|}{|\mathbf{I}|} \geq \rho. \quad (4.1.10)$$

Caso o critério definido pela equação (4.1.10) não seja satisfeito ocorre o *reset*. No *reset*, o nó J de F_2 é excluído do processo de busca dado por (4.1.9) através da seguinte equação:

$$T_j = 0. \quad (4.1.11)$$

Então é escolhida uma nova categoria através de (4.1.9) para o processo de ressonância. Este procedimento será realizado até que a rede encontre uma categoria que satisfaça (4.1.10).

Passo 8: Atualização dos Pesos (Treinamento):

Após o vetor de entrada $\mathbf{I}$ ter completado o estado de ressonância, segue o processo de treinamento, no qual ocorre a modificação do vetor peso dado por:

$$\mathbf{w}_J^{novo} = \beta (\mathbf{I} \wedge \mathbf{w}_J^{velho}) + (1 - \beta) \mathbf{w}_J^{velho}, \quad (4.1.12)$$

em que:

- J é a categoria ativa;
- $\mathbf{w}_J^{novo}$ é o vetor peso atualizado;
- $\mathbf{w}_J^{velho}$ é o vetor peso referente à atualização anterior;

O fluxograma relativo ao algoritmo acima está apresentado na Figura 4.2.

4.2 Rede Neural ART Euclidiana

A rede neural ART Euclidiana é uma técnica de classificação que utiliza a distância euclidiana entre os padrões e os centros das classes para decidir qual será a classificação do padrão. Esta técnica troca o operador *fuzzy* encontrado na rede neural ART *Fuzzy* pela distância Euclidiana.

A rede neural ART Euclidiana apresenta algumas particularidades em relação a rede ART tradicional. Podem-se listar as diferenças básicas entre este tipo de rede e a convencional: primeiramente, ela não requer nem a normalização nem a codificação dos vetores de entrada, isto reduz consideravelmente o custo computacional [31], consequentemente, os vetores de entrada diferenciam-se dos vetores que não são múltiplos por constante um do outro obtendo assim uma melhor classificação dos padrões. Além disso o operador AND *fuzzy* é substituído pela distância euclidiana entre os padrões e os centros das classes, com isso, ao invés de hiper-retângulos obtêm-se hiperesferas centradas no centro das classes.

Em relação ao algoritmo da rede ART *Fuzzy*, são poucas as alterações. Estas serão listadas a seguir:

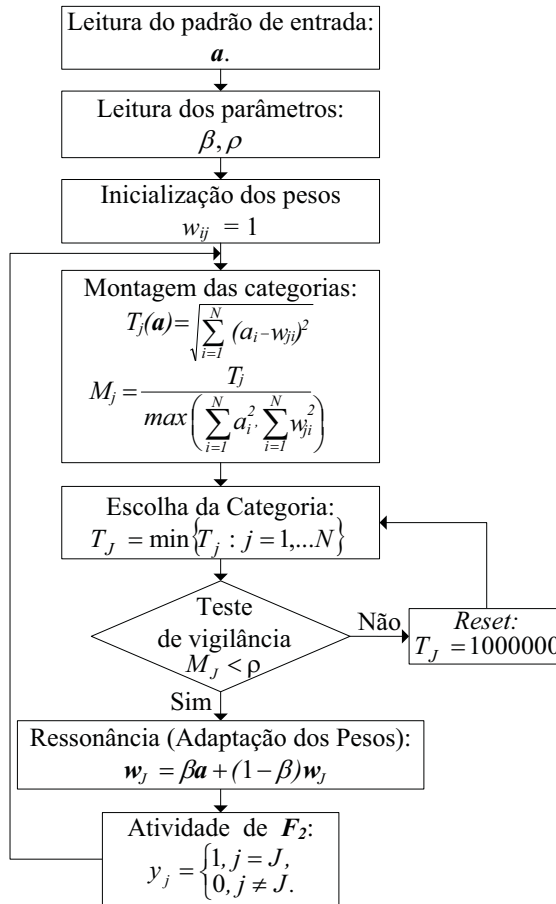


Figura 4.3: Fluxograma da Rede ART Euclidiana.

- As funções de ativação e a função *match* da rede neural ART Euclidiana são dadas por:

$$T_j(a) = \sqrt{\sum_{i=1}^M (a_i - w_{ji})^2}, \quad (4.2.1)$$

$$M_j = \frac{T_j}{\max \left(\sum_{i=1}^M a_i^2, \sum_{i=1}^M w_{ji}^2 \right)}. \quad (4.2.2)$$

- Outra diferença é que ao invés de se escolher a categoria que tem ativação máxima, a categoria escolhida é a que possui ativação mínima e, a ressonância só ocorre se a função *match* da categoria escolhida for menor que o critério de vigilância.

$$J = \arg \min T_j : j = 1, 2, \dots, N, \quad (4.2.3)$$

$$M_j < \rho. \quad (4.2.4)$$

Deste modo, uma baixa vigilância resulta em rigorosa condição de ressonância e um maior número de nós de saída.

- A função de adaptação (vetor de pesos) continua a mesma da arquitetura ART tradicional [31]:

$$\mathbf{w}_J^{novo} = \beta \mathbf{a} + (1 - \beta) \mathbf{w}_J^{velho}, \quad (4.2.5)$$

O funcionamento da rede neural ART Euclidiana pode ser visto através do fluxograma apresentado na Figura 4.3.

4.3 Rede Neural ARTMAP *Fuzzy*

O tipo de arquitetura de uma rede neural ARTMAP *Fuzzy* utiliza o treinamento supervisionado e auto-organizável e é própria para a aproximação de funções não-lineares multidimensionais. Ela é composta de dois módulos ART *Fuzzy*, designados ART_a e ART_b que desempenham as funções de um módulo ART descrito na seção anterior, durante a fase de treinamento (ou aprendizado) as entradas $\mathbf{a}^{(n)}$ são apresentadas ao Módulo ART_a, que deve realizar a montagem dos *clusters* de entrada, cuja saída correspondente $\mathbf{b}^{(n)}$ é apresentada ao módulo ART_b, que realiza a *clustering* dos dados de saída. O módulo inter-ART realiza um mapeamento cujo propósito é estabelecer um “casamento” entre as entradas e saídas, realizando assim a vigilância do sistema, observe a Figura 4.4.

Observe que:

$$|\mathbf{I}| = \sum_i^{M_a} \bar{a}_i + \sum_i^{M_a} \bar{a}_i^c = M_a$$

e

$$|\mathbf{O}| = \sum_i^{M_b} \bar{b}_i + \sum_i^{M_b} \bar{b}_i^c = M_b,$$

portanto, todos os vetores de entrada após a normalização e codificação terão a mesma magnitude: M_a ; do mesmo modo que todos os vetores de saída: M_b .

Definição de Parâmetros: É necessário definir todos os parâmetros que serão utilizados no processamento da rede ARTMAP *Fuzzy*:

1. Parâmetro de escolha: $\alpha > 0$;
2. Taxa de Treinamento: $\beta \in [0, 1]$;
3. Parâmetro de Vigilância do Módulo ART_a: $\rho_a \in (0, 1)$;
4. Parâmetro de Vigilância do Módulo ART_b: $\rho_b \in (0, 1)$;
5. Parâmetro de Vigilância do Módulo ART_{ab}: $\rho_{ab} \in (0, 1)$.

Inicialização dos Pesos: Inicialmente todos os pesos dos três módulos possuem valor igual a 1, ou seja:

$$\begin{aligned} w_{j1}^a(0) &= w_{j2}^a(0) = \dots = w_{j,2M_a}^a(0) = 1, \\ w_{k1}^b(0) &= w_{k2}^b(0) = \dots = w_{k,2M_b}^b(0) = 1, \\ w_{l1}^{ab}(0) &= w_{l2}^{ab}(0) = \dots = w_{lN}^{ab}(0) = 1, \end{aligned} \tag{4.3.4}$$

indicando que não existe nenhuma categoria ativa.

2ª Fase: Treinamento da Rede Neural: Nesta fase é realizado o treinamento da rede neural. É aqui que é feito o *clustering* dos dados de entrada e de saída e também realizado o “casamento” das classes de saída com as da entrada.

Módulo ART_b:

Montagem das Categorias e Escolha da Categoria Vencedora: Dado o vetor de saída $\mathbf{O}$ em F_1^b , para cada nó k em F_2^b , a função de escolha T_k é determinada por:

$$T_k^b = \frac{|\mathbf{O} \wedge \mathbf{w}_k^b|}{\alpha + |\mathbf{w}_k^b|}. \tag{4.3.5}$$

A categoria é escolhida como sendo o nó K ativo, ou seja:

$$K = \arg \max \{T_k^b : k = 1, 2, \dots, N_b\}. \quad (4.3.6)$$

Usando a equação (4.3.6) pode-se encontrar mais de uma categoria ativa, a categoria escolhida será aquela que possuir menor índice.

Ressonância ou *Reset*: A ressonância ocorre se o critério de vigilância (4.3.7) for satisfeito:

$$\frac{|\mathbf{O} \wedge \mathbf{w}_k^b|}{|\mathbf{O}|} \geq \rho_b. \quad (4.3.7)$$

Caso o critério definido pela equação (4.3.7) não seja satisfeito ocorre o *reset*. No *reset*, o nó K de F_2^b é excluído do processo de busca dado por (4.3.6) através da seguinte equação:

$$T_K^b = 0. \quad (4.3.8)$$

Então é escolhida uma nova categoria através de (4.3.6) para o processo de ressonância. Este procedimento será realizado até que a rede encontre uma categoria que satisfaça (4.3.7).

Atualização dos Pesos (Treinamento): Após o vetor de saída $\mathbf{O}$ ter completado o estado de ressonância, segue o processo de treinamento, no qual ocorre a modificação do vetor peso dado por:

$$\mathbf{w}_K^{novo} = \beta (\mathbf{O} \wedge \mathbf{w}_K^{velho}) + (1 - \beta) \mathbf{w}_K^{velho}, \quad (4.3.9)$$

em que:

- K é a categoria ativa;
- $\mathbf{w}_K^{novo}$ é o vetor peso atualizado;
- $\mathbf{w}_K^{velho}$ é o vetor peso referente à atualização anterior;

Atividade da rede ART_b: O vetor de atividade de F_2^b é simbolizado por:

$$\mathbf{y}^b = \begin{bmatrix} y_1^b & y_2^b & \cdots & y_{N_b}^b \end{bmatrix},$$

em que N_b é o número de categorias criadas em F_2^b . Assim, tem-se:

$$y_k^b = \begin{cases} 1, & \text{se o nó } k \text{ de } F_2 \text{ for ativo,} \\ 0, & \text{caso contrário.} \end{cases} \quad (4.3.10)$$

Módulo ART_a :

Escolha da Categoria Módulo ART_a : Dado o vetor de entrada $\mathbf{I}$ em F_1^a , para cada nó j em F_2^a , a função de escolha T_j^a é determinada por:

$$T_j^a = \frac{|\mathbf{I} \wedge \mathbf{w}_j^a|}{\alpha + |\mathbf{w}_j^a|}. \quad (4.3.11)$$

A categoria é escolhida como sendo o nó J ativo, ou seja:

$$J = \arg \max \{T_j^a : j = 1, 2, \dots, N_a\}. \quad (4.3.12)$$

Usando a equação (4.3.12) podemos encontrar mais de uma categoria ativa, a categoria escolhida será aquela que possuir menor índice.

Ressonância ou *Reset*: A ressonância ocorre se o critério de vigilância (4.3.13) for satisfeito:

$$\frac{|\mathbf{I} \wedge \mathbf{w}_J^a|}{|\mathbf{I}|} \geq \rho_a. \quad (4.3.13)$$

Caso o critério definido pela equação (4.3.13) não seja satisfeito ocorre o *reset*. No *reset*, o nó J de F_2^a é excluído do processo de busca dado por (4.3.12) através da seguinte equação:

$$T_J^a = 0. \quad (4.3.14)$$

Então, é escolhida uma nova categoria através de (4.3.12) para o processo de ressonância. Este procedimento será realizado até que a rede encontre uma categoria que satisfaça (4.3.13).

Atividade da rede ART_a : O vetor de atividade de F_2^a é simbolizado por:

$$\mathbf{y}^a = \begin{bmatrix} y_1^a & y_2^a & \cdots & y_{N_a}^a \end{bmatrix},$$

em que N_a é o número de categorias criadas em F_2^a . Assim, temos:

$$y_j^a = \begin{cases} 1, & \text{se o nó } j \text{ de } F_2 \text{ for ativo,} \\ 0, & \text{caso contrário.} \end{cases} \quad (4.3.15)$$

Módulo *Inter-ART*:

Ressonância ou *Reset* A ressonância ocorre se o critério de vigilância (4.3.16) for satisfeito:

$$\frac{|\mathbf{y}^b \wedge \mathbf{w}_l^{ab}|}{|\mathbf{y}^b|} \geq \rho_{ab}. \quad (4.3.16)$$

Caso o critério definido pela equação (4.3.16) não seja satisfeito ocorre o *reset*. No *reset*, o nó J de F_2^a é excluído do processo de busca dado por (4.3.12). Além disso, é realizada uma alteração temporária no parâmetro de vigilância ρ_a dada por:

$$\rho_a = \frac{|\mathbf{I} \wedge \mathbf{w}_j^a|}{|\mathbf{I}|} + \epsilon. \quad (4.3.17)$$

Então é escolhida uma nova categoria através de (4.3.12) para o processo de ressonância. Este procedimento será realizado até que a rede encontre uma categoria que satisfaça (4.3.16).

Atualização dos Pesos do Módulo *Inter-ART* (Treinamento): Como houve um casamento entre os padrões de entrada e saída apresentados a rede, acontece a adaptação dos pesos do módulo *Inter-ART*:

$$\mathbf{w}_{jk}^{ab} = \begin{cases} 1, & j = J, \quad k = K, \\ 0, & j \neq J, \quad k \neq K. \end{cases} \quad (4.3.18)$$

Módulo ART_a :

Atualização dos Pesos do Módulo ART_a (Treinamento): Após o vetor de entrada $\mathbf{I}$ ter completado o estado de ressonância, segue o processo de treinamento, no qual ocorre a

modificação do vetor peso dado por:

$$\mathbf{w}_J^{novo} = \beta (\mathbf{I} \wedge \mathbf{w}_J^{velho}) + (1 - \beta) \mathbf{w}_J^{velho}, \quad (4.3.19)$$

em que:

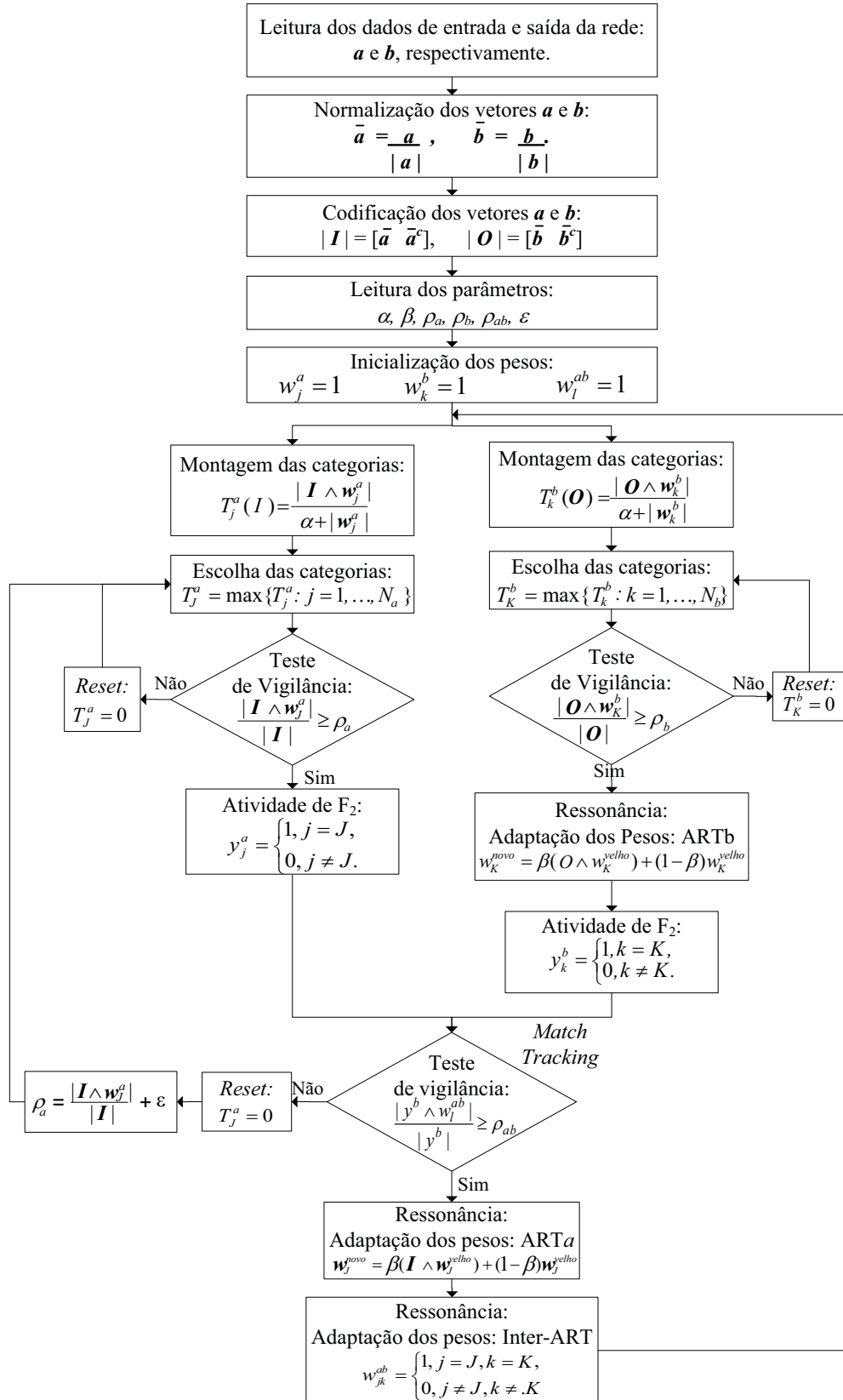
- J é a categoria ativa;
- $\mathbf{w}_J^{novo}$ é o vetor peso atualizado;
- $\mathbf{w}_J^{velho}$ é o vetor peso referente à atualização anterior.

Apresenta-se na Figura 4.5 o fluxograma da rede neural ARTMAP *Fuzzy*.

Observações:

- Se tivermos $\beta = 1$ então teremos o treinamento rápido.
- $\wedge$ é o operador AND nebuloso, definido por:

$$(\mathbf{D} \wedge \mathbf{E}) = \min(d_i, e_i). \quad (4.3.20)$$

Figura 4.5: Fluxograma de uma Rede Neural ARTMAP *Fuzzy*.

Redes ART e ARTMAP Fuzzy Modificada

Em uma rede neural ART tradicional o número de neurônios utilizados na procura do neurônio vencedor é exatamente igual ao número de padrões apresentados à rede, ou seja, ao procurar um neurônio vencedor calcula-se a função de ativação de todos os neurônios já ativados, mas também a de todos os neurônios ainda inativos. Assim, calcula-se a função de ativação para todos os neurônios existentes e, portanto, obtém-se várias funções de ativação iguais pois todos os neurônios inativos têm a mesma função de ativação. Não teria uma forma de fazer isso de um modo diferente? Apresenta-se aqui uma forma mais racional e econômica de se realizar o treinamento, inicia-se com somente um neurônio ativado, com os dados do primeiro padrão apresentado à rede, e, quando for apresentado o segundo padrão, calcula-se a função de ativação apenas para este neurônio, caso o novo padrão seja suficientemente similar ao primeiro, ocorre a ressonância, caso contrário, um novo neurônio é ativado com os dados deste novo padrão e, com isto, quando for apresentado o terceiro próximo padrão tem-se apenas duas funções de ativação para testar. Assim, quando for apresentado à rede o n -ésimo padrão, tem-se p neurônios ativados, com $p \leq n$, para verificar-se a similaridade. Este procedimento é análogo ao que ocorre no

cérebro, ou seja, há uma grande quantidade de neurônios inativos que serão ativados (usados) à medida que novos conhecimentos sejam requeridos. Por exemplo, quando uma pessoa, desconhecida até então, é posta ao alcance do sistema visual, o mecanismo cerebral busca reconhecê-la considerando os neurônios ativos. Como não identifica, neste conjunto de neurônios, a similaridade, novo(s) neurônios (inativos) serão estimulados para “absorver” este novo conhecimento.

5.1 Rede Neural ART *Fuzzy* Modificada

Ao contrário de uma rede neural ART *Fuzzy* tradicional que considera o número de *clusters* igual ao número de padrões a serem apresentados a rede, este tipo de rede começa com apenas um *cluster*, cujos dados representam os dados da primeira entrada, e durante o treinamento vai se expandindo para abrigar novos padrões. Deste modo, o cálculo da função de ativação é executado para um número muito menor de *cluster*, e o teste de vigilância, também. Para que se entenda a dinâmica do funcionamento desta nova rede, apresenta-se a seguir o algoritmo da Rede Neural ART *Fuzzy* Modificada.

Algoritmo da Rede Neural ART *Fuzzy* Modificada

Passo 1: Leitura dos dados de entrada:

Os dados de entrada são denotados pelo vetor:

$$\mathbf{a} = [a_1 \quad a_2 \quad \dots \quad a_M]$$

M -dimensional. Este vetor é normalizado com o intuito de evitar a proliferação de muitas categorias. Assim:

$$\bar{\mathbf{a}} = \frac{\mathbf{a}}{|\mathbf{a}|}, \quad (5.1.1)$$

em que: $\bar{\mathbf{a}}$ é o vetor normalizado e:

$$|\mathbf{a}| = \sum_i^M a_i. \quad (5.1.2)$$

Passo 2: Codificação do vetor de entrada:

A codificação de complemento é realizada para preservar a amplitude da informação:

$$\bar{a}_i^c = 1 - \bar{a}_i, \quad (5.1.3)$$

em que: $\bar{\mathbf{a}}^c$ é o vetor complementar do vetor de entrada normalizado. Deste modo, o novo vetor de entrada será um vetor $2M$ -dimensional que será denotado por:

$$\mathbf{I} = [\bar{\mathbf{a}} \quad \bar{\mathbf{a}}^c], \quad (5.1.4)$$

ou ainda:

$$\mathbf{I} = [\bar{a}_1 \quad \bar{a}_2 \quad \dots \quad \bar{a}_M \quad \bar{a}_1^c \quad \bar{a}_2^c \quad \dots \quad \bar{a}_M^c].$$

Observe que:

$$|\mathbf{I}| = \sum_i^M \bar{a}_i + \sum_i^M \bar{a}_i^c = M,$$

portanto, todos os vetores de entrada após a normalização e codificação terão a mesma magnitude: M .

Passo 3: Vetor de Atividade:

O vetor de atividade de F_2 é simbolizado por

$$\mathbf{y} = \begin{bmatrix} y_1 & y_2 & \dots & y_N \end{bmatrix},$$

em que: N é o número de categorias criadas em F_2 . Assim, temos

$$y_j = \begin{cases} 1, & \text{se o nó } j \text{ de } F_2 \text{ for ativo,} \\ 0, & \text{caso contrário.} \end{cases} \quad (5.1.5)$$

Passo 4: Parâmetros da Rede:

Os parâmetros utilizados no processamento da rede ART *Fuzzy* Modificada são:

1. Parâmetro de escolha: $\alpha > 0$;
2. Taxa de Treinamento: $\beta \in [0, 1]$;

3. Parâmetro de Vigilância: $\rho \in (0, 1)$;

Passo 5: Inicialização dos Pesos:

Diferentemente da rede ART *Fuzzy* tradicional, esta rede começa com apenas um *cluster* que contém os dados do primeiro vetor de entrada, assim tem-se que:

$$w_{11}(0) = I_{11}, \quad w_{12}(0) = I_{12} \quad \cdots \quad w_{jM}(0) = I_{1,2M}, \quad (5.1.6)$$

indicando que existe apenas um *cluster* ativo com os dados do primeiro padrão.

Passo 6: Inicialização do Contador de *clusters*:

$$N = 1 \quad (5.1.7)$$

em que: N indica o número de *clusters* ativos.

Início do Treinamento:

Passo 7: Inicialização do Contador:

Como é necessário verificar se todos os *clusters* foram testados para abrigar a referida entrada e, no caso de nenhum *cluster* já existente comportar a entrada atual, seja criado um *cluster* para esta, foi preciso a inclusão de um contador:

$$Cont = 1 \quad (5.1.8)$$

Toda vez que uma entrada é apresentada a rede, este contador é inicializado.

Passo 8: Cálculo da Função Escolha:

Dado o vetor de entrada $\mathbf{I}$ em F_1 , para cada nó j em F_2 , a função de escolha T_j é determinada por

$$T_j = \frac{|\mathbf{I} \wedge \mathbf{w}_j|}{\alpha + |\mathbf{w}_j|}, \quad (5.1.9)$$

Busca do *Cluster*:

Passo 9: Escolha do *Cluster*:

O *cluster* é escolhido como sendo o nó J ativo, ou seja:

$$J = \arg \max \{T_j : j = 1, 2, \dots, N\}. \quad (5.1.10)$$

Usando a equação (5.1.10) podemos encontrar mais de uma categoria ativa, a categoria escolhida será aquela que possuir menor índice.

Passo 10: Teste de Vigilância:

O critério de vigilância é dado por:

$$\frac{|\mathbf{I} \wedge \mathbf{w}_j|}{|\mathbf{I}|} \geq \rho. \quad (5.1.11)$$

Caso o critério definido pela equação (5.1.11) seja satisfeito vai-se para o Passo 13.

Passo 11: Verificação se todos os *clusters* já foram testados:

A verificação se todos os *clusters* já foram testados é feita através da equação:

$$N < Cont. \quad (5.1.12)$$

Caso o critério definido pela Equação (5.1.12) não seja satisfeito, vai-se ao Passo 12. Caso contrário, o contador é atualizado:

$$Cont^{novo} = Cont^{velho} + 1 \quad (5.1.13)$$

e ocorre o *reset*. No *reset*, o nó J de F_2 é excluído do processo de busca dado por (5.1.10) através da seguinte equação:

$$T_j = 0. \quad (5.1.14)$$

Então retorna-se ao Passo 9.

Passo 12: Criação de um *Cluster*:

Caso o critério definido pela Equação (5.1.12) não seja satisfeito significa que nenhum dos *clusters* já existentes comporta a entrada atual. Deste modo se faz necessário a criação

de um novo *cluster*, assim o contador de *clusters* é atualizado:

$$N^{novo} = N^{velho} + 1 \quad (5.1.15)$$

e

$$J = N.$$

Daí

$$\mathbf{w}_J = \mathbf{I}. \quad (5.1.16)$$

Após a criação do *cluster* vai-se ao Passo 14.

Passo 13: Atualização dos Pesos (Treinamento):

Neste passo ocorre o treinamento, isto é, a modificação do vetor peso dado por:

$$\mathbf{w}_J^{novo} = \beta (\mathbf{I} \wedge \mathbf{w}_J^{velho}) + (1 - \beta) \mathbf{w}_J^{velho}, \quad (5.1.17)$$

em que:

- J é a categoria ativa;
- $\mathbf{w}_J^{novo}$ é o vetor peso atualizado;
- $\mathbf{w}_J^{velho}$ é o vetor peso referente à atualização anterior;

Passo 14: Vetor de Atividade: O vetor de atividade de F_2 é simbolizado por:

$$\mathbf{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_N \end{bmatrix},$$

em que N é o número de *clusters* criados em F_2 . Assim, temos:

$$y_j = \begin{cases} 1, & \text{se o nó } j \text{ de } F_2 \text{ for ativo,} \\ 0, & \text{caso contrário.} \end{cases} \quad (5.1.18)$$

Para mostrar exatamente como isto ocorre apresenta-se o fluxograma da Figura 5.1.

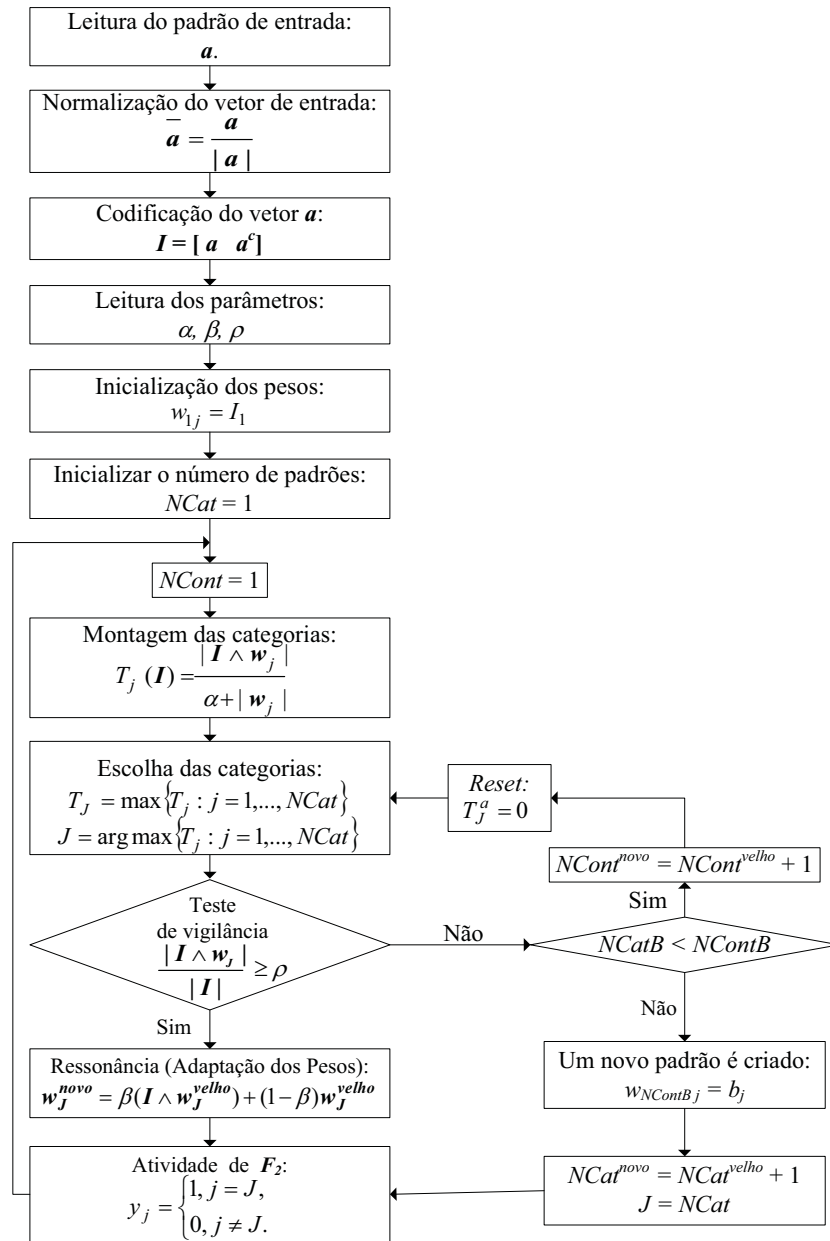
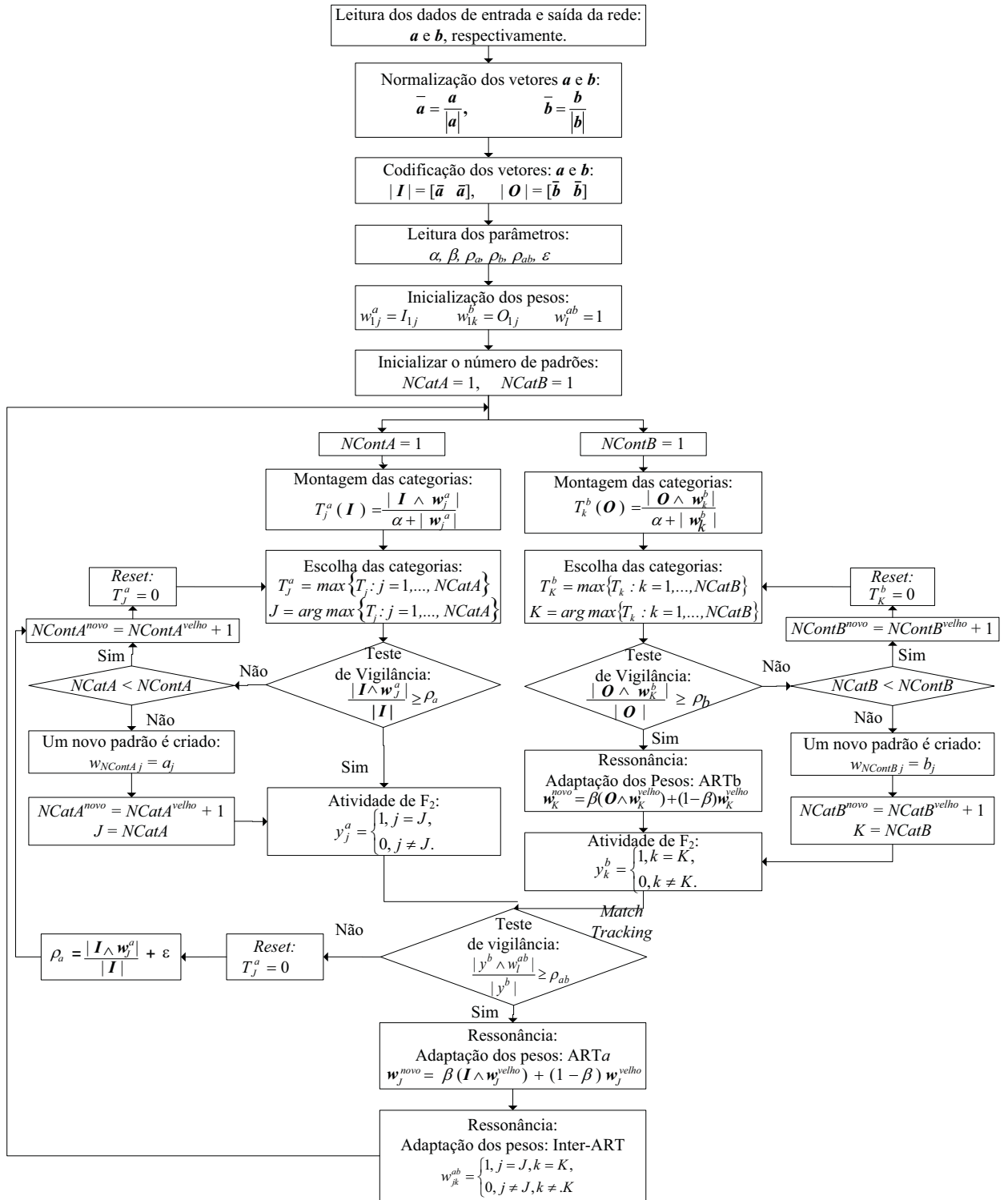


Figura 5.1: Fluxograma de uma Rede ART *Fuzzy* Modificada.

5.2 Rede Neural ARTMAP *Fuzzy* Modificada

Considerando que a rede neural ARTMAP *Fuzzy* Modificada apresenta dois módulos ART *Fuzzy* Modificado no lugar dos dois módulos ART *Fuzzy*, tem-se que este tipo de rede é mais rápida durante no treinamento, isto levando-se em conta as considerações anteriores. Para uma melhor visualização do que acontece neste tipo de rede, apresenta-se na Figura 5.2 seu fluxograma.

Figura 5.2: Fluxograma de uma Rede Neural ARTMAP *Fuzzy* Modificada.

ART Euclidiana Modificada com Treinamento Continuado

O sistema neural desenvolvido nesta pesquisa contempla a rotina de treinamento continuado. Trata-se de um recurso incorporado à rede neural ART Euclidiana que visa tornar o sistema de treinamento e análise mais eficiente, ou seja, a extração do conhecimento é realizada de modo permanente. Esta rotina constitui-se, a princípio, numa efetiva possibilidade do emprego de redes neurais na análise em tempo real, destruindo o conceito, até então, sobre as limitações (instabilidade e não-plasticidade) das redes neurais no trato de problemas reais o que tem conferido pouca confiabilidade de aplicação. Algumas modificações se tornaram necessárias na rede neural ART Euclidiana para que ela pudesse comportar o mecanismo de treinamento continuado. Apresenta-se a seguir todas as alterações, juntamente com o mecanismo de treinamento continuado já incorporado.

6.1 Mecanismo do Treinamento Continuado

A primeira alteração em relação a rede neural ART Euclidiana tradicional, é que o tipo de rede desenvolvida nesta pesquisa possui *clusters* temporários e *clusters* definitivos. Esta medida se faz necessária para que não sejam incorporados à rede, *clusters* que representem apenas uma anomalia (aparecem raramente).

Os parâmetros são a segunda modificação já que dois novos parâmetros foram incorporados: $NMIN$ e η . O parâmetro $NMIN$ se refere ao número de padrões necessários para que um *cluster* temporário se torne definitivo: a diferença entre os *clusters* temporários e definitivos durante a execução da rede é que todos os temporários são contados e, quando atinge o valor $NMIN$ este padrão é incorporado aos padrões definitivos. Enquanto um padrão pertencer a um *cluster* temporário a ressonância sempre ocorre, ao passo que, quando o *cluster* se torna definitivo, a ressonância só ocorre se distância entre a entrada e o centro do *cluster* for superior ao parâmetro η , evitando assim o sobre-treinamento.

A terceira, é que foram inseridos quatro contadores, sendo um para contagem do número de *clusters* provisórios, outro para os definitivos e o terceiro para controlar a inserção de um novo *cluster* temporário. O último é um contador na forma de vetor onde cada linha é responsável pela contagem do número de padrões cuja representação é dada pelo respectivo *cluster* provisório.

Para melhor compreensão apresenta-se, a seguir, o algoritmo desta rede:

Algoritmo da Rede Neural ART Euclidiana Modificada com Treinamento Continuado

Pré-processamento:

Passo 1: Leitura dos dados da primeira entrada:

Os dados de entrada são denotados pelo vetor:

$$\mathbf{a} = [a_1 \quad a_2 \quad \dots \quad a_M]$$

M -dimensional.

Passo 2: Parâmetros da Rede:

Os parâmetros utilizados no processamento da rede ART Euclidiana Modificada com Treinamento Continuado são:

- Taxa de Treinamento: $\beta \in [0, 1]$;

- Parâmetro de Vigilância: $\rho \in (0, 1)$;
- Índice de Novidade: $\eta > 0$;
- Parâmetro para Permanência: $NMIN \in \mathbb{N}^*$.

Passo 3: Inicialização do *Cluster*:

Ao contrário da rede ART *Fuzzy*, esta rede inicialmente possui apenas um *cluster* que contém o primeiro dado de entrada:

$$w_{11}^T(0) = a_1, \quad w_{12}^T(0) = a_2, \quad \dots \quad w_{1M}^T(0) = a_M, \quad (6.1.1)$$

esse número vai se expandindo à medida em que é realizado o treinamento continuado. Adicionalmente, é inicializado também o contador de número de padrões que esta categoria, enquanto temporária, representa:

$$ContT(1) = 1. \quad (6.1.2)$$

Este contador é importante pois, com ele, verifica-se o número de padrões representado pelo *cluster* temporário. E, assim, consegue-se verificar se as informações nele contidas representam uma informação importante para a rede ou apenas uma anomalia e, assim, deve ser destruída.

Passo 4: Inicialização dos Contadores de Categoria:

$$NCT = 1, \quad NC = 0. \quad (6.1.3)$$

Estes números representam o número de categorias temporárias e o número de categorias definitivas.

Início do Treinamento e Análise Contínua:

Passo 5: Leitura da entrada atual.

Passo 6: Inicialização Contador para Criação de uma Categoria Temporária

$$Cont = 1. \quad (6.1.4)$$

Passo 7: Cálculo da Distância entre a Entrada e o Centro da Categoria já Existente:

Com o mecanismo de treinamento continuado agregado a rede ART Euclidiana, existem categorias temporárias e categorias definitivas. Deste modo há a necessidade de se calcular a distância entre a entrada $\mathbf{a}$ e o centro da categoria tanto da categoria temporária quanto da definitiva. Assim:

$$T_j = \sqrt{\sum_{i=1}^M (a_i - w_{ij})^2}, \quad (6.1.5)$$

e

$$T_j^T = \sqrt{\sum_{i=1}^M (a_i - w_{ij}^T)^2} \quad (6.1.6)$$

Passo 8: Busca da Categoria Vencedora:

A categoria escolhida é a que apresenta menor distância. Nota-se que aqui a busca é feita tanto entre as categorias definitivas quanto nas temporárias, assim precisa-se diferenciar quando uma categoria temporária ganha a disputa de quando uma categoria definitiva a vence. Para isto usa-se um identificador que nos indica se a categoria vencedora é temporária ou definitiva: $TEMP$. Quando o identificador estiver ativado ele possui valor 1, isto é:

$$TEMP = 1. \quad (6.1.7)$$

Passo 9: Cálculo de Similaridade:

Por tratar-se de uma rede neural ART Euclidiana, há a necessidade de se calcular a similaridade entre a entrada e a categoria vencedora, esta similaridade é dada por:

$$M_j = \frac{T_j}{\max \left(\sum_{i=1}^M a_i^2, \sum_{i=1}^M w_{ji}^2 \right)} \quad (6.1.8)$$

ou

$$M_j = \frac{T_j^T}{\max \left(\sum_{i=1}^M a_i^2, \sum_{i=1}^M (w_{ji}^T)^2 \right)}, \quad (6.1.9)$$

pois a categoria vencedora pode ser temporária ou definitiva.

Passo 10: Critério de Vigilância:

Se o critério de vigilância:

$$M_j < \rho \quad (6.1.10)$$

for satisfeito passa-se ao **Passo 12**. Caso contrário, é verificado se o contador para criação de uma categoria temporária já chegou ao número de categorias provisórias mais o número de categorias definitivas. Se isso já aconteceu passa-se ao **Passo 11**. Caso contrário o contador é atualizado:

$$Cont^{novo} = Cont^{velho} + 1$$

e ocorre o *Reset*, isto é, a categoria vencedora tanto se for definitiva quanto se for provisória é excluída do processo de busca. Para isto é utilizado a equação:

$$T_J = 1000000 \quad (6.1.11)$$

se for uma categoria definitiva, ou

$$T_J^T = 1000000 \quad (6.1.12)$$

se for uma categoria temporária. E, retorna-se ao **Passo 8**.

Passo 11: Criação de uma Categoria Temporária:

Quando o contador é igual ao número de categorias temporárias já existentes mais o número de categorias definitivas, significa que nenhum *cluster* já existente é similar o suficiente à categoria para incorporá-la. Assim, é necessária a criação de uma categoria temporária para esse padrão. Primeiramente é necessário indicar a criação de uma categoria, isto é feito através da equação:

$$NCT^{novo} = NCT^{velho} + 1$$

e, então, é criada a categoria temporária:

$$w_j = a \quad (6.1.13)$$

então, retorna-se ao **Passo 5** para a leitura da próxima entrada.

Passo 12: Verificação se a categoria vencedora é temporária ou definitiva:

Caso a categoria vencedora seja definitiva, passa-se ao **Passo 13**. Caso contrário, ocorre a ressonância, ou seja, os dados do referido *cluster* são atualizados:

$$\mathbf{w}_J^{novo} = \beta \mathbf{a} + (1 - \beta) \mathbf{w}_J^{velho}, \quad (6.1.14)$$

e o contador do número de entradas incorporados neste *cluster* temporário é atualizado:

$$ContT_J^{novo} = ContT_J^{velho} + 1, \quad (6.1.15)$$

Caso este número não atinja o número mínimo de entradas para que seja incorporado como um *cluster* definitivo, ou seja, se

$$ContT_J < NMIN$$

então retorna-se ao **Passo 5** para a leitura de um novo dado de entrada. Caso contrário, é criado um *cluster* definitivo:

$$NC^{novo} = NC^{velho} + 1$$

e os dados referentes ao *cluster* temporário são alocado neste novo *cluster*:

$$\mathbf{w}_{NC} = \mathbf{w}_J^T \quad (6.1.16)$$

Em seguida, é feita a atualização do número de *clusters* temporários, dos *clusters* temporários e do contador de *clusters* temporários:

$$NCT^{novo} = NC^{velho} - 1;$$

$$\mathbf{w}_j^T = \mathbf{w}_{j+1}^T, \quad j = J, \dots, NCT; \quad (6.1.17)$$

$$ContT_j = ContT_{j+1}, \quad j = J, \dots, NCT. \quad (6.1.18)$$

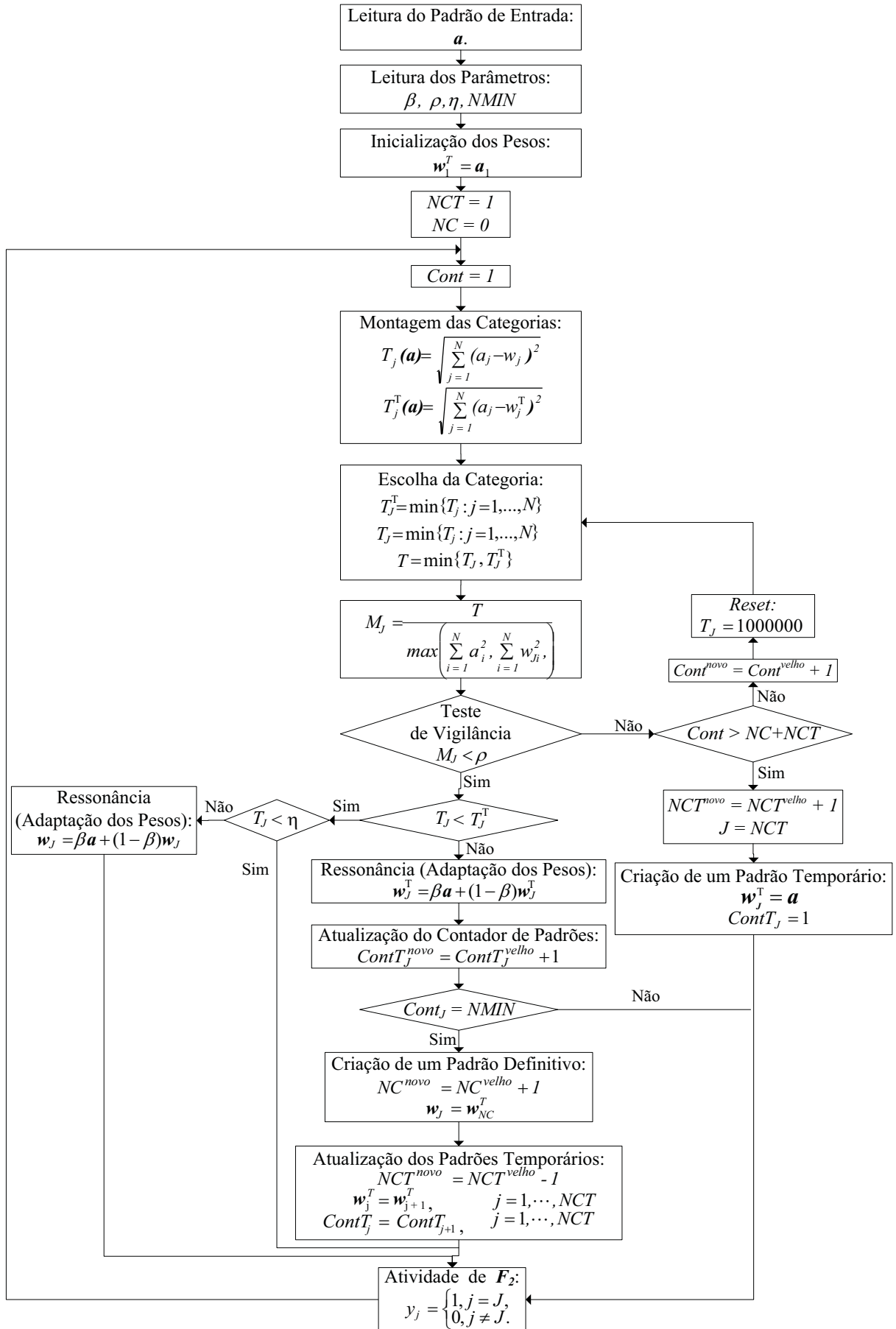


Figura 6.1: Fluxograma da Rede Neural ART Euclidiana Modificada com Treinamento Continuado.

Então, retorna-se ao **Passo 5** para um novo dado de entrada.

Passo 13: Detecção de Anomalia:

Quando uma entrada for classificada em um *clusters* definitivo, é necessário verificar se é apenas uma anomalia ou se há a necessidade de realizar o treinamento com o centro de padrões. Isto é feito através da seguinte equação:

$$T_j < \eta \quad (6.1.19)$$

Se a Equação (6.1.19) for satisfeita, isto significa que não há a necessidade do treinamento, assim retorna-se ao **Passo 5**. Caso contrário, antes é necessário a atualização:

$$\mathbf{w}_J^{novo} = \beta \mathbf{a} + (1 - \beta) \mathbf{w}_J^{velho}. \quad (6.1.20)$$

Apresenta-se na Figura 6.1 o fluxograma relativo a rede neural ART Euclidiana Modificada com Treinamento Continuado.

6.2 Uma Primeira Aplicação

Para ilustrar melhor o funcionamento desta rede considera-se os pontos do plano apresentados na Tabela 6.1, tomados arbitrariamente:

Tabela 6.1: Exemplo Prático da Dinâmica do Treinamento Continuado.

x	y	Previsão Inicial	Previsão Final	x	y	Previsão Inicial	Previsão Final
1,0	1,0	1	1	0,0	-1,0	6	7
0,0	0,0	1	2	0,0	1,0	2	8
0,5	0,3	2	3	0,0	2,0	8	9
0,0	0,3	2	2	2,0	2,0	1	10
1,4	1,0	1	1	0,2	0,2	2	2
0,9	0,8	1	1	1,0	0,5	1	1
0,5	0,0	3	3	2,0	0,1	4	11
1,0	0,0	3	4	2,3	1,3	10	10
-1,0	0,0	2	5	-1,1	-0,3	5	5
-1,0	-1,0	5	6	-1,5	-1,1	6	6

As figuras a seguir representam como o treinamento continuado acontece. Cada cor representa um *cluster*, a primeira figura representa o primeiro *cluster*, já nas demais, a figura da esquerda representa a classificação antes do treinamento continuado e, à direita, representa a classificação após o treinamento continuado.

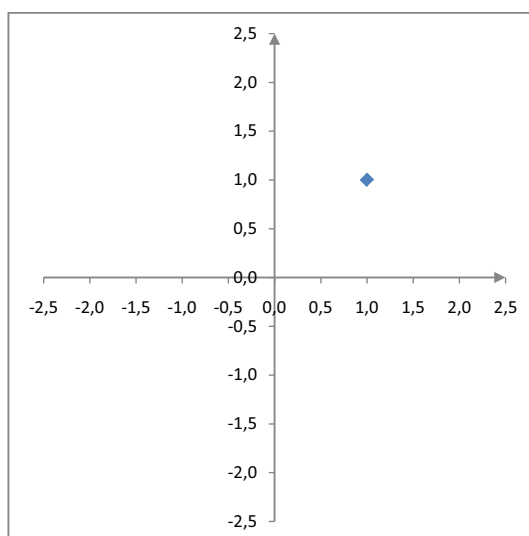


Figura 6.2: Visualização da Primeira Entrada.

A Figura 6.2 representa a classificação do primeiro dado de entrada, $(1, 0; 1, 0)$, este dado automaticamente gera uma categoria.

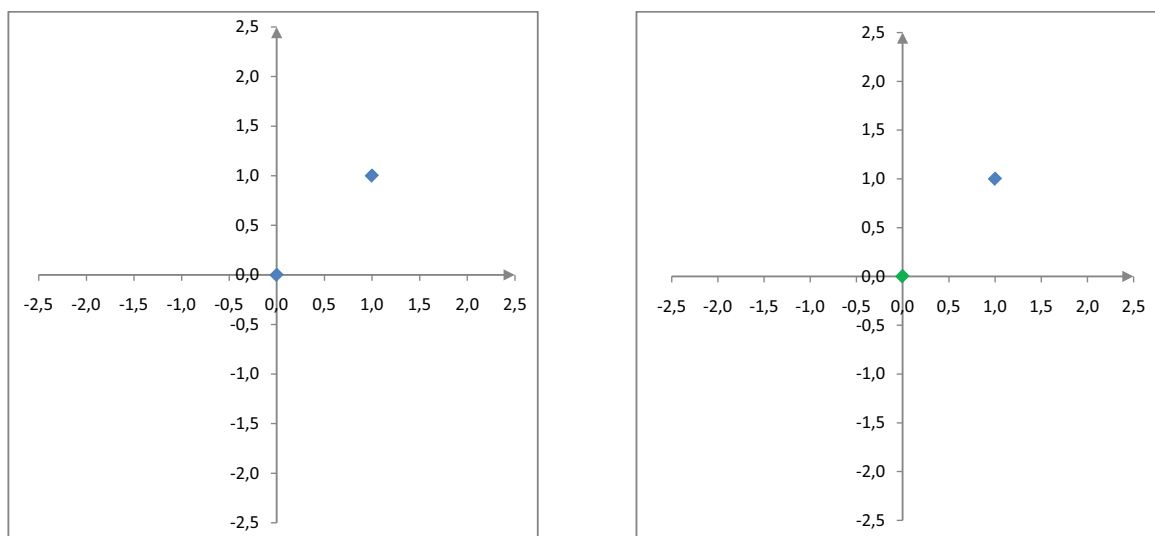


Figura 6.3: Classificação do Ponto $(0, 0; 0, 0)$ Antes e Depois do Treinamento Continuado.

Na Figura 6.3 tem-se a primeira classificação indicada à esquerda, como antes do treinamento continuado existia um único *cluster*, o segundo dado, $(0, 0; 0, 0)$, foi alocado neste *cluster*. Entretanto, após a realização do treinamento continuado, este dado gerou um novo *cluster*.

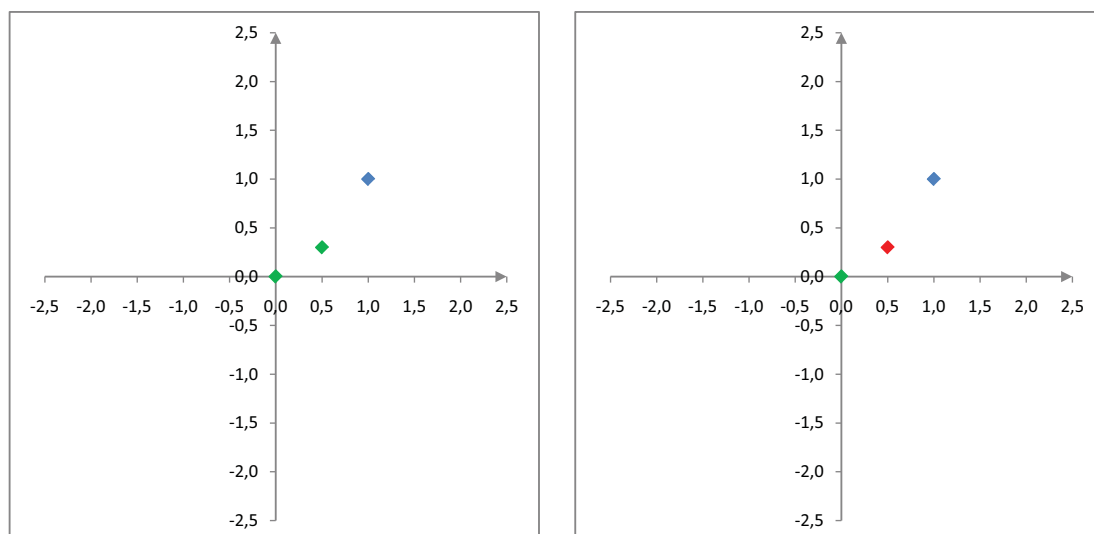


Figura 6.4: Classificação do Ponto $(0, 5; 0, 3)$ Antes e Depois do Treinamento Continuado.

Observe agora a Figura 6.4. Nota-se que, como existem dois *clusters* a rede procura o que é mais similar ao novo dado de entrada e faz uma primeira classificação. Como pode-se observar, este dado, $(0, 5; 0, 3)$ não é suficientemente similar a nenhum dos *clusters* já existentes. Portanto a rede cria um *cluster* para abrigá-lo.

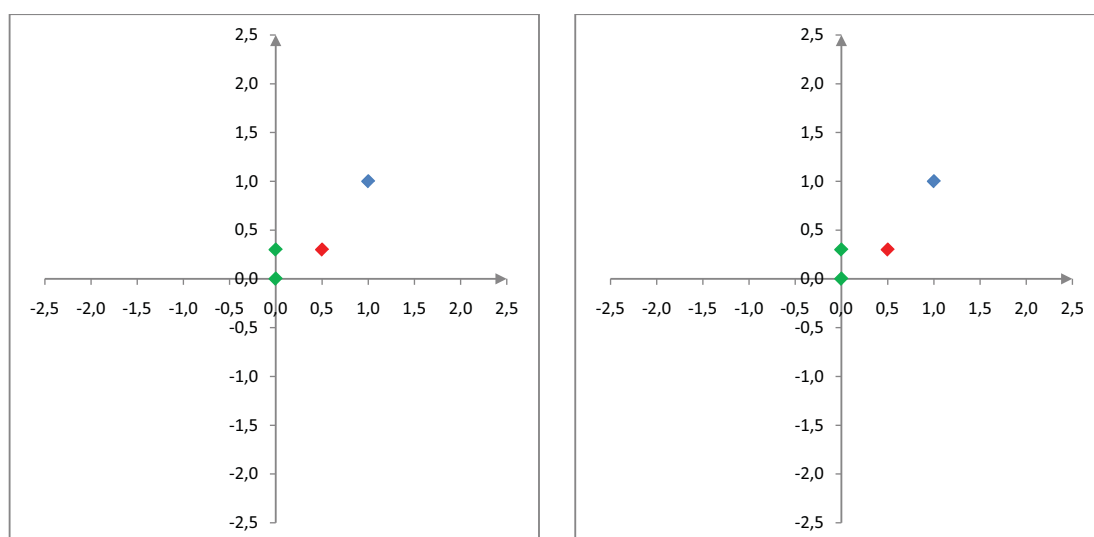


Figura 6.5: Classificação do Ponto $(0, 0; 0, 3)$ Antes e Depois do Treinamento Continuado.

Observe a Figura 6.5, mesmo após o treinamento continuado a classificação continua a mesma, isto ocorre pois o quarto dado, $(0, 0; 0, 3)$, é suficientemente similar a um *cluster* já existente. O mesmo ocorre em relação aos pontos $(1, 4; 1, 0)$, $(0, 9; 0, 8)$ e $(0, 5; 0, 0)$, conforme pode ser observado nas Figuras 6.6, 6.7 e 6.8.

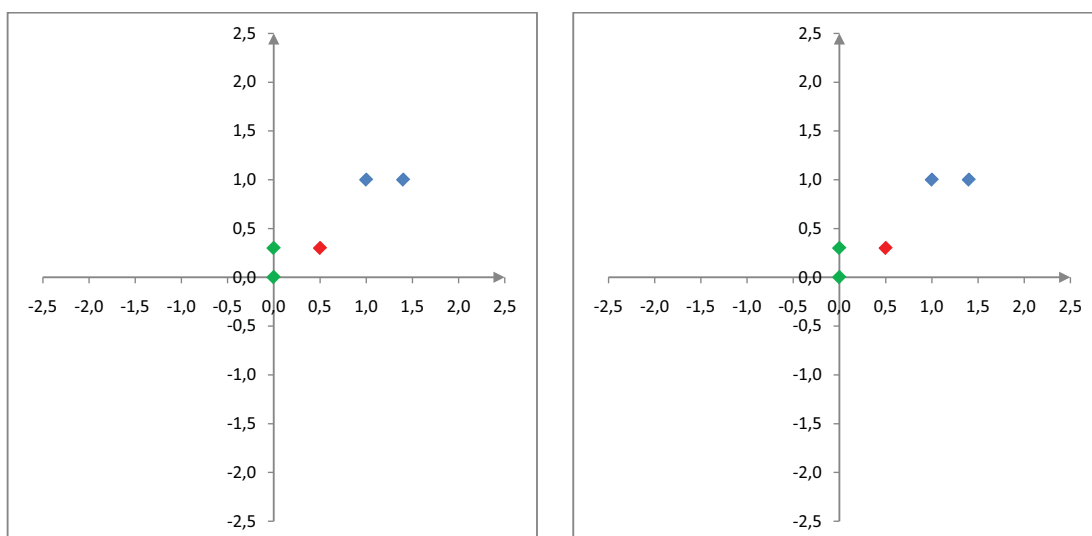


Figura 6.6: Classificação do Ponto $(1, 4; 1, 0)$ Antes e Depois do Treinamento Continuado.

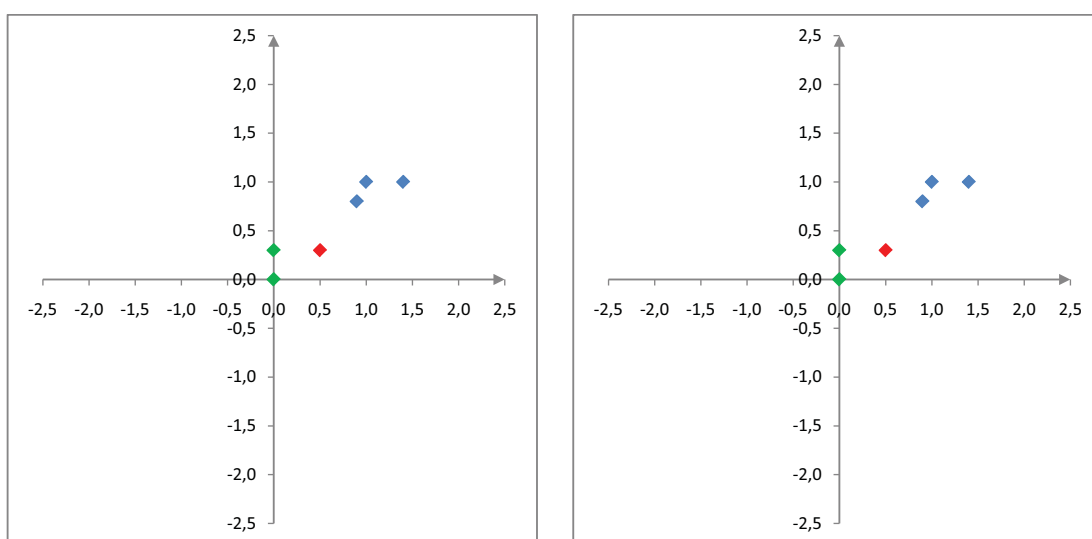


Figura 6.7: Classificação do Ponto $(0, 9; 0, 8)$ Antes e Depois do Treinamento Continuado.

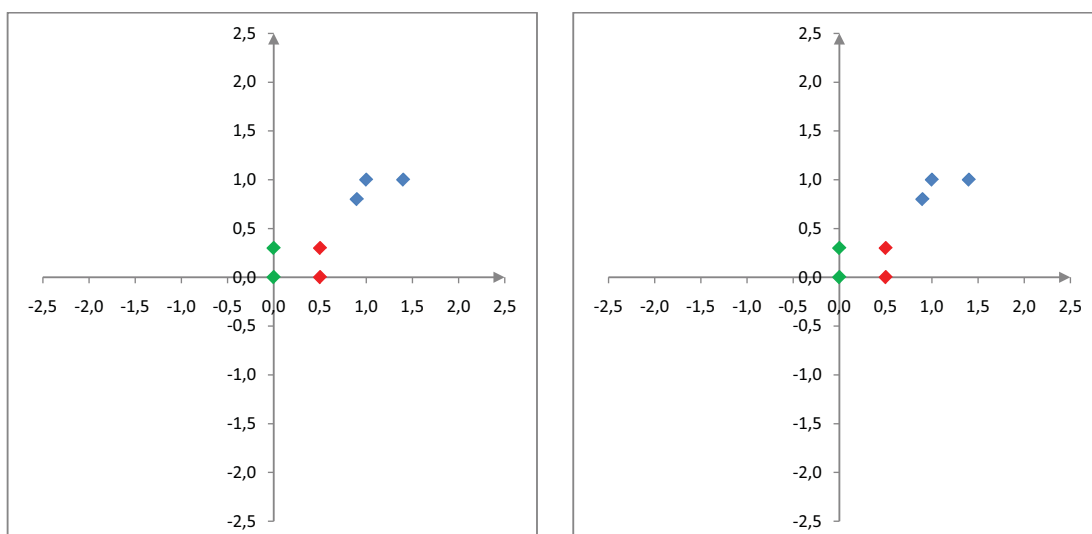


Figura 6.8: Classificação do Ponto $(0, 5; 0, 0)$ Antes e Depois do Treinamento Continuado.

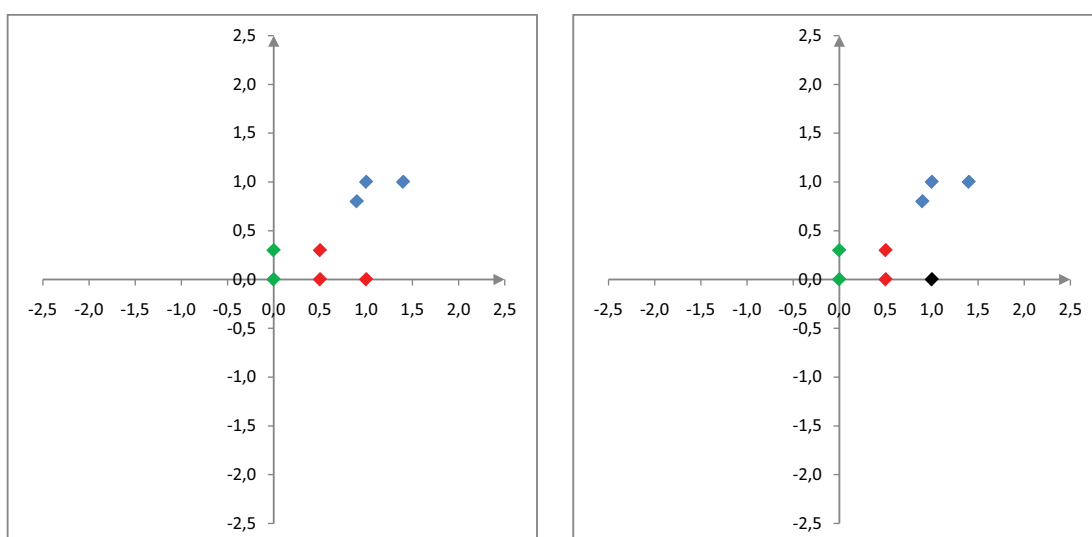


Figura 6.9: Classificação do Ponto $(1, 0; 0, 0)$ Antes e Depois do Treinamento Continuado.

Novamente, tem-se um dado novo para a rede, como anteriormente ele procura o *cluster* que é mais similar ao dado de entrada, faz uma primeira classificação, como não existe uma categoria suficientemente similar ao ponto $(1, 0; 0, 0)$, o treinamento continuado gera, então, um novo *cluster*, observe a Figura 6.9. O mesmo acontece em relação aos pontos: $(-1, 0; 0, 0)$, $(-1, 0; -1, 0)$, $(0, 0; -1, 0)$, $(0, 0; 1, 0)$, $(0, 0; 2, 0)$ e $(2, 0; 2, 0)$. Isto pode ser observado nas Figuras 6.10, 6.11, 6.12, 6.13, 6.14 e 6.15.

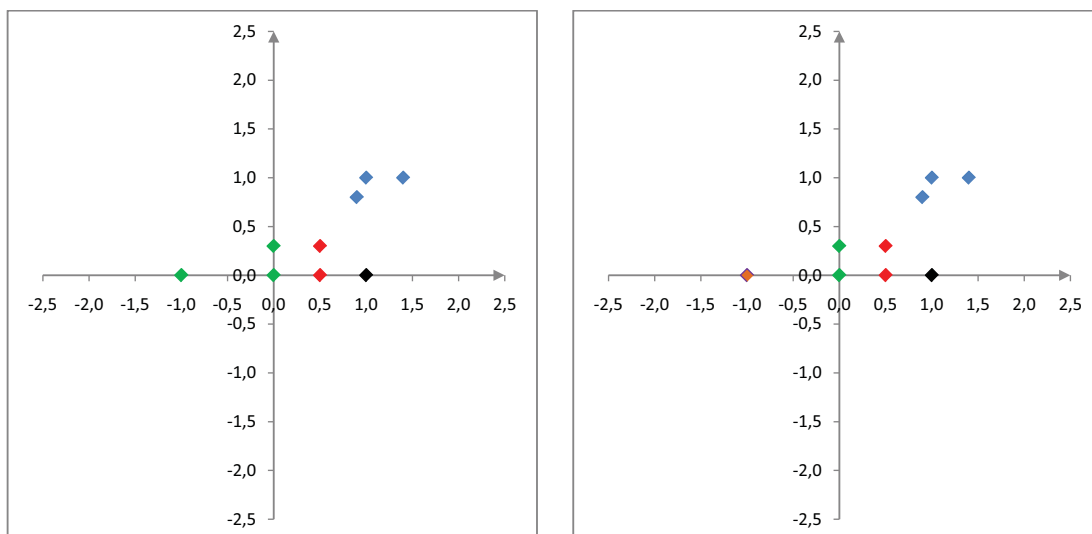


Figura 6.10: Classificação do Ponto $(-1, 0; 0, 0)$ Antes e Depois do Treinamento Continuado.

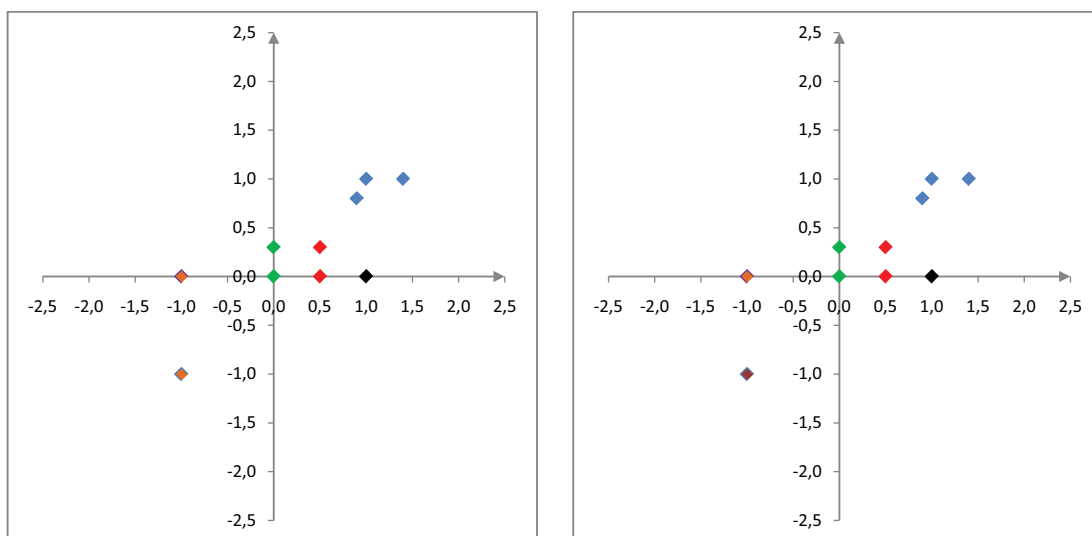


Figura 6.11: Classificação do Ponto $(-1, 0; -1, 0)$ Antes e Depois do Treinamento Continuado.

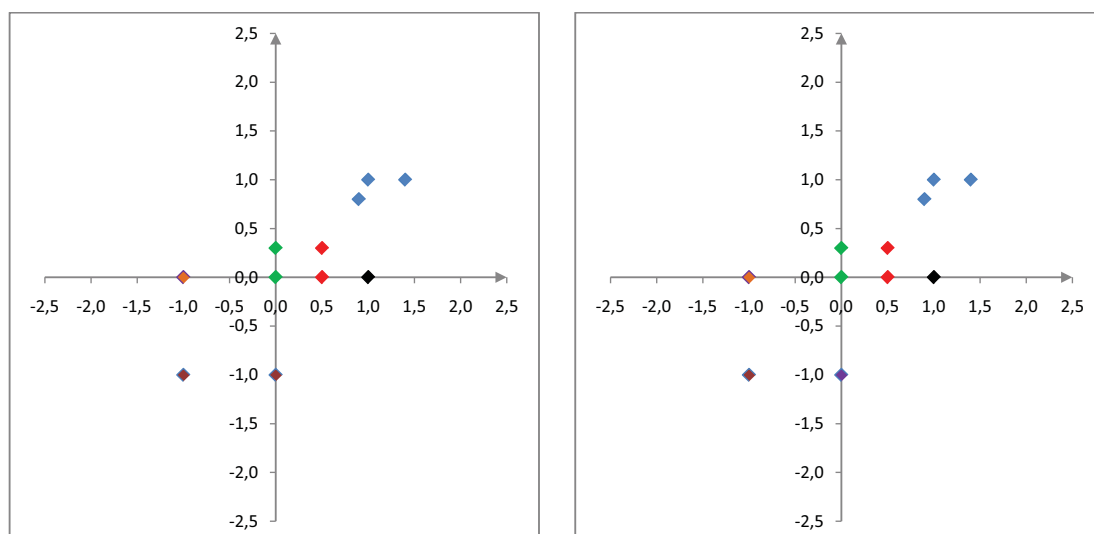


Figura 6.12: Classificação do Ponto $(0, 0; -1, 0)$ Antes e Depois do Treinamento Continuado.

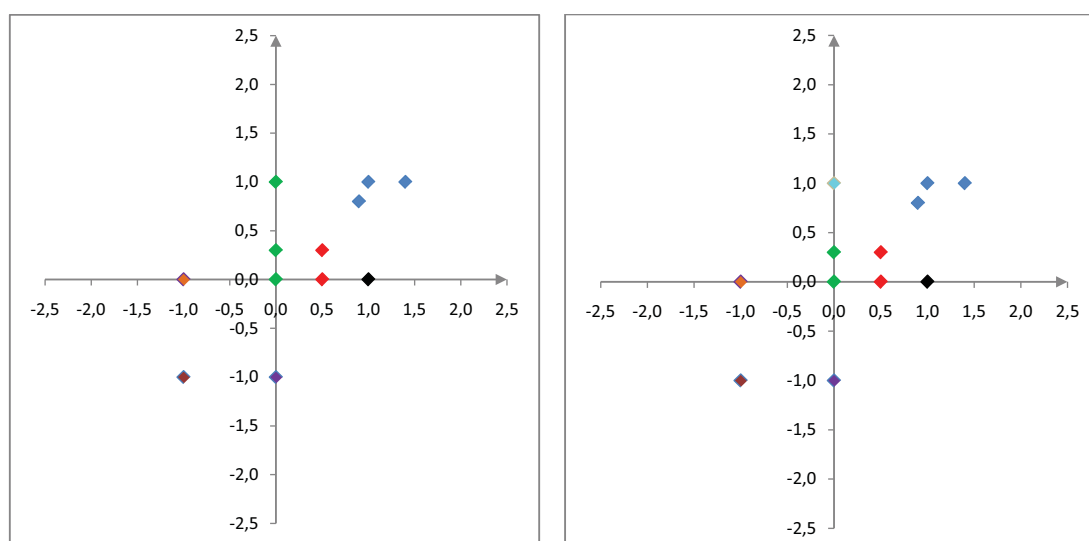


Figura 6.13: Classificação do Ponto $(0, 0; 1, 0)$ Antes e Depois do Treinamento Continuado.

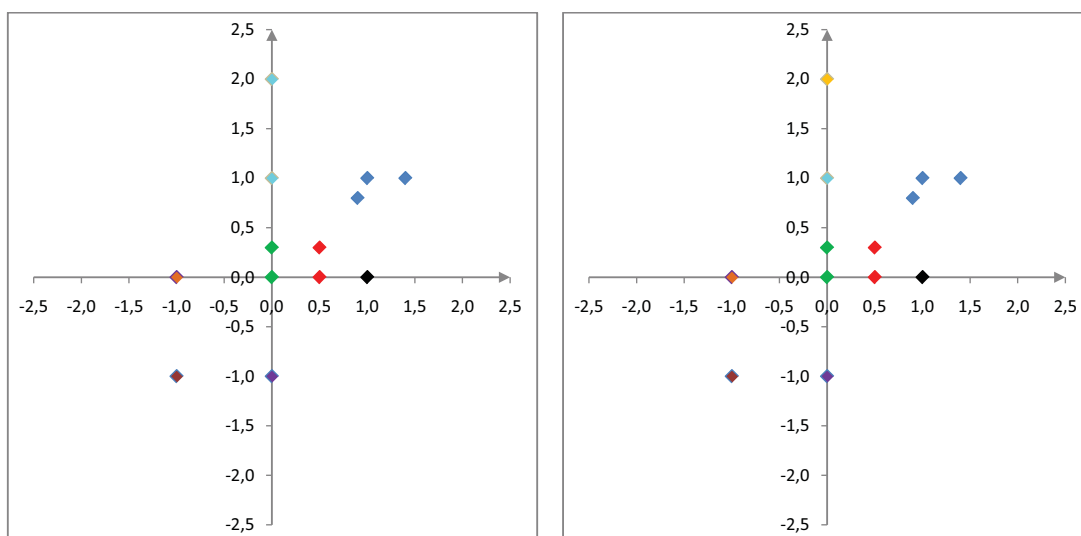


Figura 6.14: Classificação do Ponto $(0, 0; 2, 0)$ Antes e Depois do Treinamento Continuo.

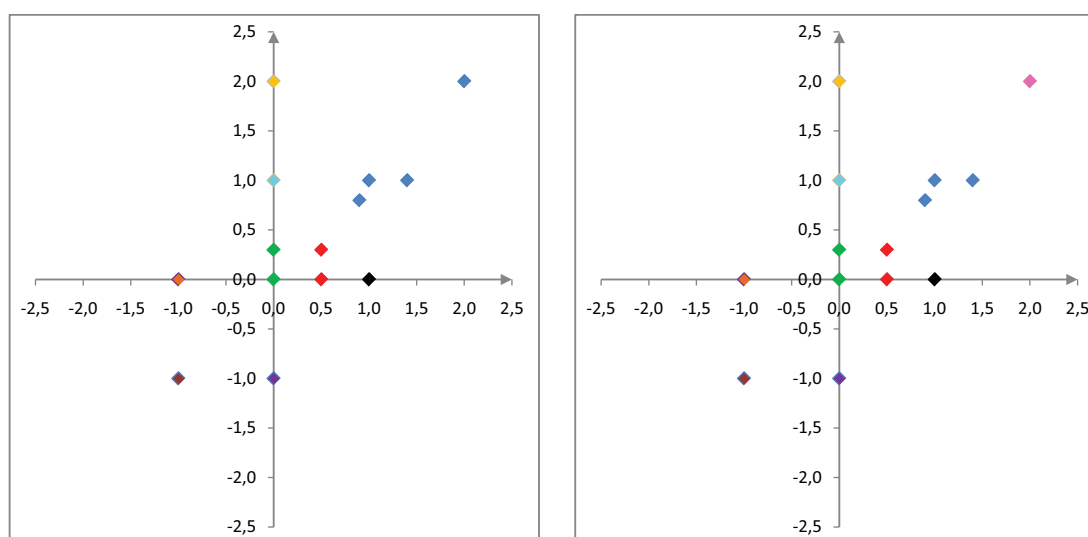


Figura 6.15: Classificação do Ponto $(2, 0; 2, 0)$ Antes e Depois do Treinamento Continuo.

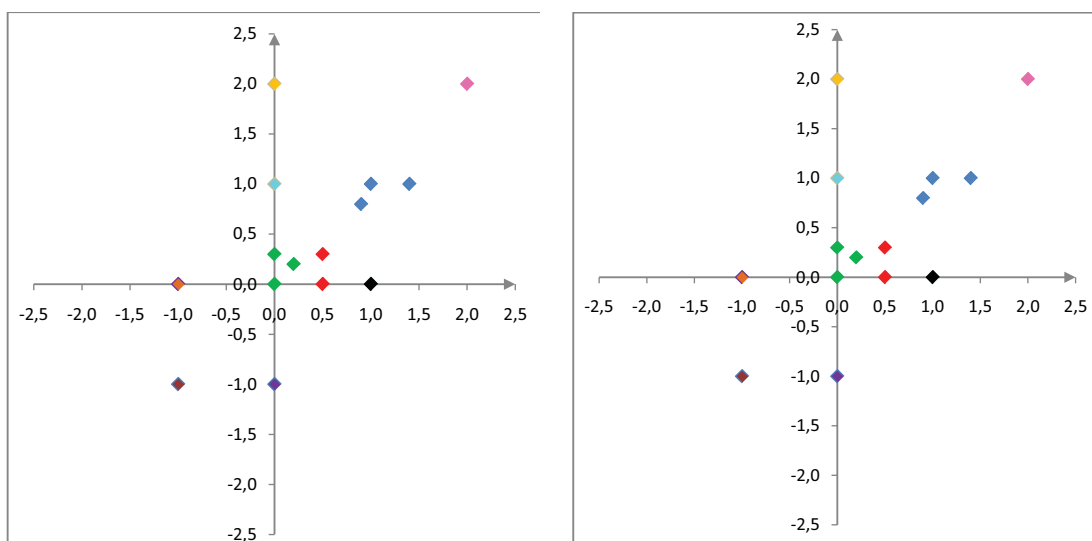


Figura 6.16: Classificação do Ponto $(0, 2; 0, 2)$ Antes e Depois do Treinamento Continuo.

Como pode-se observar na Figura 6.16, o ponto $(0, 2; 0, 2)$ é suficientemente similar a uma classe já existente. Portanto, a classificação antes e depois do treinamento continuado é idêntica. O mesmo ocorre em relação ao ponto $(1, 0; 0, 5)$, Figura 6.17.

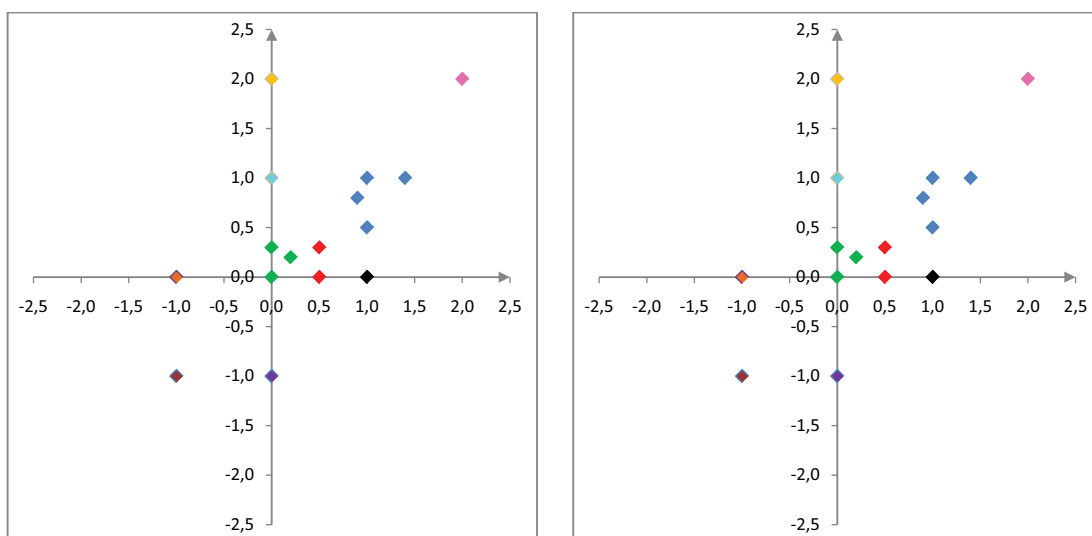


Figura 6.17: Classificação do Ponto $(1, 0; 0, 5)$ Antes e Depois do Treinamento Continuo.

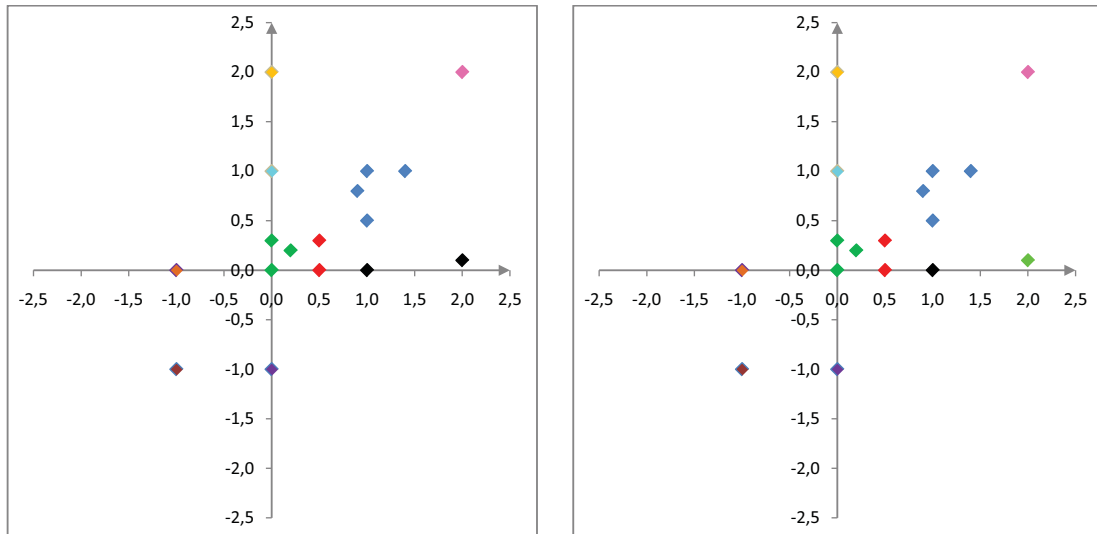


Figura 6.18: Classificação do Ponto $(2, 0; 0, 1)$ Antes e Depois do Treinamento Continuado.

Observando a Figura 6.18, nota-se que ponto $(2, 0; 0, 1)$ não é suficientemente similar a nenhum *cluster* já existente, sendo assim, apesar da classificação inicial, é gerado um *cluster* para comportá-lo.

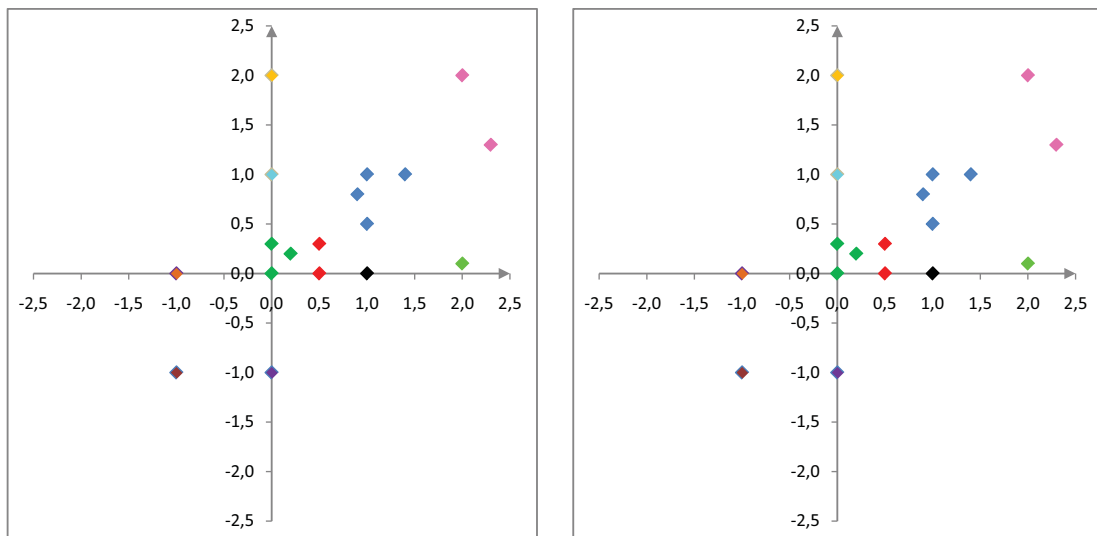


Figura 6.19: Classificação do Ponto $(2, 3; 1, 3)$ Antes e Depois do Treinamento Continuado.

O ponto $(2, 3; 1, 3)$ é suficientemente similar a um *cluster* já existente, Figura 6.19. Deste modo, a classificação antes e depois do treinamento continuado permanece a mesma. Isto ocorre também em relação aos pontos $(-1, 1; -0, 3)$ e $(-1, 5; -1, 1)$, como pode-se observar nas Figuras 6.20 e 6.21.

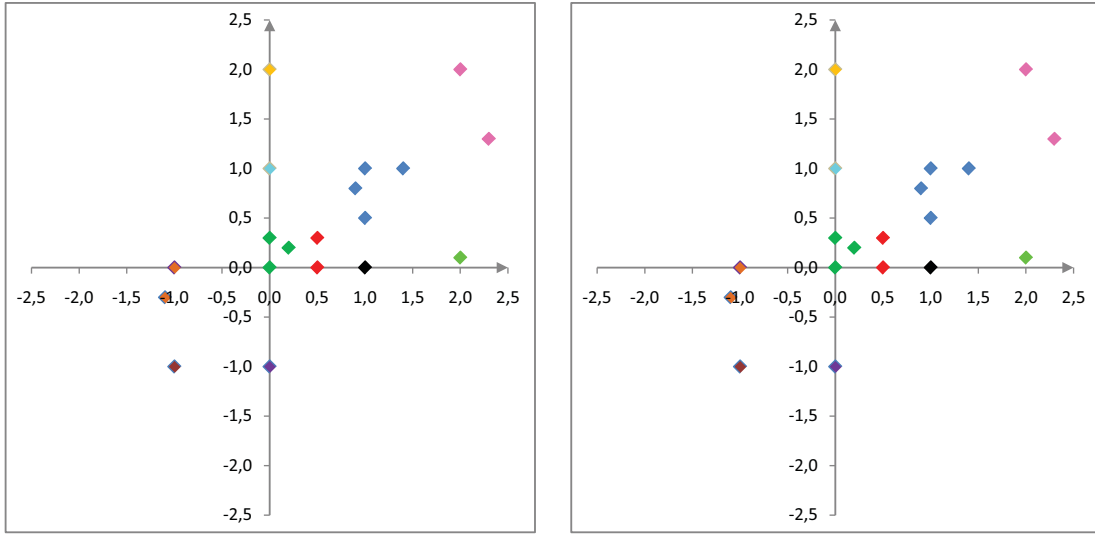


Figura 6.20: Classificação do Ponto $(-1, 1; -0, 3)$ Antes e Depois do Treinamento Continuado.

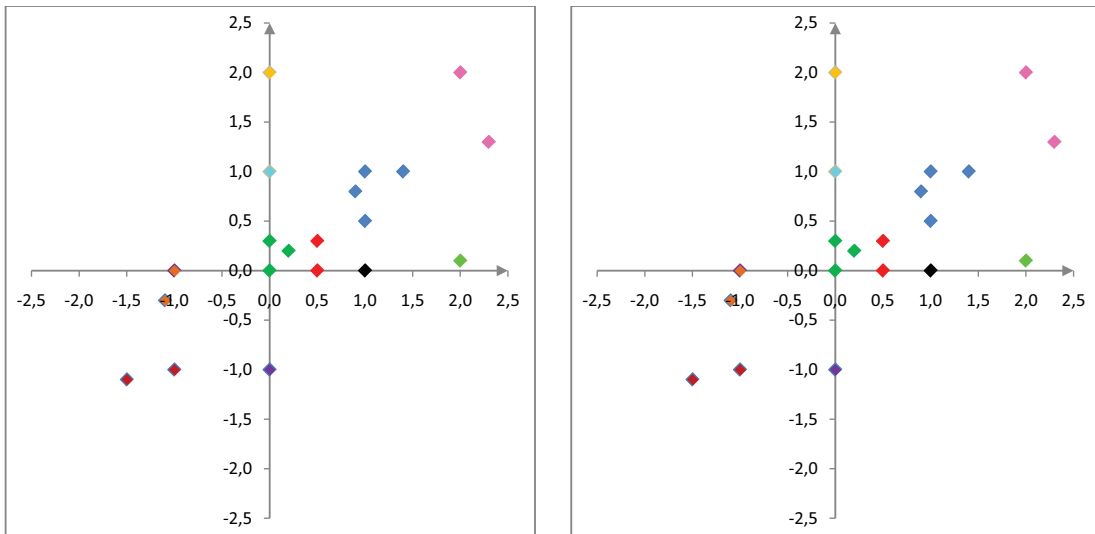


Figura 6.21: Classificação do Ponto $(-1, 5; -1, 1)$ Antes e Depois do Treinamento Continuado.

Nota-se que a rede neural ART Euclidiana Modificada com Treinamento Continuado, após fazer uma classificação inicial, verifica se o dado da entrada é suficientemente similar a algum dos *clusters* já existentes e, no caso de não existir nenhum, cria um *cluster* para abrigá-lo. Durante o processamento na rede neural são criados *clusters* temporários para abrigar os dados que não são similares a nenhum *cluster* já existente. Estes *clusters* temporários só se tornarão definitivos se surgir, durante o treinamento, um número mínimo de padrões similares a ele. Além disso, depois que um *cluster* se torna definitivo o treinamento só ocorre se a distância do novo dado em relação ao centro do *cluster* ultrapassar o valor de η . Caso contrário, ele não é utilizado no processo de adaptação de pesos.

Rede Neural Proposta e Aplicação

A aplicação para a rede neural proposta foi a Análise de Estabilidade Transitória de um sistema multimáquinas. O modelo utilizado foi o Sistema Sul-Brasileiro e os dados obtidos através do programa Simul [43]. O Simul é um aplicativo para análise da estabilidade transitória de sistemas elétricos de potência, considerando-se o modelo clássico. A análise é baseada na margem de segurança (método da energia), como descrito no Capítulo 3. Primeiramente é apresentada a rede neural ART-ARTMAP Fuzzy Euclidiana Modificada com Treinamento Continuoado, juntamente com todas especificações necessárias para o tratamento de tal problema, bem como suas justificativas. Para que a rede neural ART-ARTMAP Fuzzy Euclidiana Modificada com Treinamento Continuoado fosse testada, foi utilizado a análise de estabilidade transitória para o sistema sul-brasileiro. No total foram abordadas 14 configurações diferentes de tal sistema, sendo que, após a primeira configuração de entrada, ao identificar uma nova configuração a rede neural exigia um total de 150 dados iniciais para que as informações em relação a tal configuração se tornasse parte da matriz de pesos definitiva. Ao final do capítulo apresenta-se uma tabela com

algumas simulações comparativas via Simul, utilizando a rede neural proposta antes e depois do treinamento continuado.

7.1 Problema Proposto

Considerando-se o problema da análise transitória de primeira oscilação, cujo modelo das máquinas síncronas refere-se ao modelo clássico que é representado por equações diferenciais não-lineares ordinárias de segunda ordem. Os dados gerados pelo programa *Simul* podem ser escritos da seguinte forma:

$$\mathbf{X}_1^{(k)} = \begin{bmatrix} \mathbf{P}^{(k)} & \mathbf{Q}^{(k)} & \mathbf{TOP}^{(k)} & \mathbf{FT}^{(k)} & \mathcal{M}^{(k)} \end{bmatrix}, \quad (7.1.1)$$

para $k = 1, 2, \dots, NP$, sendo:

- $\mathbf{X}_1^{(k)}$: k -ésimo vetor padrão;
- $\mathbf{P}^{(k)}$: vetor de potência elétrica ativa nodal constituinte do k -ésimo vetor padrão de entrada;
- $\mathbf{Q}^{(k)}$: vetor de potência elétrica reativa nodal constituinte do k -ésimo vetor padrão de entrada;
- $\mathbf{TOP}^{(k)}$: vetor com a configuração associada ao k -ésimo vetor padrão de entrada;
- $\mathbf{FT}^{(k)}$: falta (contingência) associada ao k -ésimo vetor padrão de entrada;
- $\mathcal{M}^{(k)}$: índice de severidade (margem de segurança) associado ao k -ésimo vetor padrão de saída [54];
- NP : número de pares de vetores padrões.

Com o modelo definido pela Equação (7.1.1), pode-se compor a grande maioria das situações operativas dos sistemas elétricos de potência (diversidade de perfis de carga/de geração, bem como a diversidade de configurações do sistema de transmissão).

Os diferentes níveis de geração e de carga (potências elétricas ativas e reativas) são gerados de modo aleatório baseado no procedimento apresentado na referência [54]. Define-se um determinado percentual de acréscimo ou decréscimo (por exemplo, ± 5 , ± 10 , etc.) na carga total do

sistema, cuja distribuição nos nós da rede elétrica é gerada aleatória ou pseudo-aleatoriamente tomando-se diferentes sementes. Assim, para um determinado percentual de acréscimo ou decréscimo no carregamento do sistema, terá distribuições diferentes em cada barra para diferentes sementes. Este procedimento visa usar um espectro de variação da carga e da geração sem a necessidade de depender da previsão de carga, bem como do despacho de geração.

A entrada da rede neural ART Euclidiana Modificada com Treinamento Continuado é o vetor:

$$\mathbf{X}_2^{(k)} = \begin{bmatrix} \mathbf{P}^{(k)} & \mathbf{Q}^{(k)} \end{bmatrix}, \quad (7.1.2)$$

para $k = 1, 2, \dots, NP$. O objetivo desta rede é classificar estes dados em *clusters*, cujos índices serão utilizados como parte do vetor de entrada da rede ARTMAP *Fuzzy* Modificada. Assim, a saída desta rede será o índice do *cluster* que a entrada faz parte. Esta saída, juntamente com os dados referentes a configuração e a contingência, todos binarizados, se tornarão a entrada da rede ARTMAP *Fuzzy* Modificada, cuja entrada será dada por:

$$\mathbf{X}^{(k)} = \begin{bmatrix} \mathbf{B}_{PQ}^{(k)} & \mathbf{B}_T^{(k)} & \mathbf{B}_F^{(k)} \end{bmatrix}, \quad (7.1.3)$$

cujas respectivas saídas serão dadas por:

$$\mathbf{Y}^{(k)} = [\mathbf{B}_{\mathcal{M}}^{(k)}], \quad (7.1.4)$$

para $k = 1, 2, \dots, NP$, sendo:

- $\mathbf{X}^{(k)}$: k -ésimo vetor padrão de entrada da rede neural ARTMAP *Fuzzy* Modificada;
- $\mathbf{B}_{PQ}^{(k)}$: vetor binário representativo ao k -ésimo neurônio da rede neural ART Euclidiana Modificada com Treinamento Continuado;
- $\mathbf{B}_T^{(k)}$: vetor binário representativo da configuração associada ao k -ésimo vetor padrão de entrada;
- $\mathbf{B}_F^{(k)}$: vetor binário representativo da falta (contingência) associada ao k -ésimo vetor padrão de entrada;
- $\mathbf{B}_{\mathcal{M}}^{(k)}$: vetor binário representativo do índice de severidade (margem de segurança) associado ao k -ésimo vetor padrão de saída [20];
- NP : número de pares de vetores padrões.

Nota-se que, no contexto das redes neurais ART, os vetores são representações por linha, como forma de tornar mais fáceis as manipulações numéricas e a visualização das conexões neurais.

A composição do vetor binário $\mathbf{B}_T$ foi projetada para representar todas as configurações da rede elétrica consideradas e, também, outras que efetivamente venham a ser operadas. Tecnicamente, este detalhe foi idealizado tomando-se 14 *bits*, este número de *bits* é o suficiente tanto para representar as configurações básicas como para abrigar as configurações não previstas, isto dentro de um patamar de razoabilidade (abrigar as possíveis necessidades do sistema). Com o uso deste número de *bits* (vetor com 14 componentes) pode-se representar até 16384 arranjos de configurações. Este procedimento é por si só um grande avanço, mesmo se comparado ao procedimento clássico via simulações (há necessidade da resolução de um grande número de cálculo do fluxo de potência, da resolução de um conjunto de equações diferenciais não-lineares, o registro e a interpretação das curvas de oscilação. Por conseguinte, com esta estruturação há considerável redução da base de dados (fornecidos e gerados).

Com relação ao vetor binário $\mathbf{B}_F$ as observações, *mutatis mutandis*, são as mesmas apenas considerando um total de 7 *bits*.

O vetor binário $\mathbf{B}_M$ é formado pela binarização da margem de segurança $\mathcal{M}$. Como para cada vetor de entrada nas condições acima é determinada uma margem de segurança, cujo valor é um número real, houve a necessidade de transformá-la em um número binário, para isto utilizou-se o método proposto em [42]. Esse método consite na determinação de intervalos de margem de segurança em que as contingências se encontram:

$$\vartheta = \frac{1 - \mathcal{M}^A}{1 - \mathcal{M}^B} - 1 \quad (7.1.5)$$

em que:

- é o esforço de deslocamento da margem de segurança;
- $\mathcal{M}^A$ é o valor da margem de segurança inicial do intervalo;
- $\mathcal{M}^B$ é o valor da margem de segurança final do intervalo.

Assim, em termos do conceito de margem de segurança, o intervalo pode ser definido como [54]:

$$IMS_k \triangleq (\mathcal{M}_k^A, \mathcal{M}_k^B) \quad (7.1.6)$$

em que:

- IMS : é o intervalo considerado;
- k : é o índice que indica o k -ésimo intervalo.

Neste trabalho será considerado $\vartheta = 0,35$ e a margem de segurança com valores entre $-\infty$ e 1, com isto utilizaremos apenas 4 *bits* para a saída da rede ARTMAP *Fuzzy* Modificada.

N° do Intervalo	Intervalos da Margem de Segurança (IMS)	Margem de Segurança Binarizada (B_M)
1	$(-\infty, -3]$	0001
2	$(-3; -1,9630]$	0010
3	$(-1,9630; -1,1948]$	0011
4	$(-1,1948; -0,6258]$	0100
5	$(-0,6258; -0,2043]$	0101
6	$(-0,2043; 0,10790]$	0110
7	$(0,10790; 0,33920]$	0111
8	$(0,33920; 0,51050]$	1000
9	$(0,51050; 0,63740]$	1001
10	$(0,63740; 0,73140]$	1010
11	$(0,73140; 0,80110]$	1011
12	$(0,80110; 0,85260]$	1100
13	$(0,85260; 0,89080]$	1101
14	$(0,89080; 0,91910]$	1110
15	$(0,91910, 1)$	1111

7.2 Configuração do Sistema

A principal aplicação da rede neural ART-ARTMAP *Fuzzy* Euclidiana Modificada com Treinamento Continuado refere-se à análise de estabilidade transitória de primeira oscilação (modelo clássico) de sistemas de energia elétrica. O objetivo foi testar sistema neural considerando-se um sistema multimáquinas de grande porte utilizando-se uma configuração do sistema elétrico da

região sul do Brasil [43, 44]. Trata-se de sistema composto por 10 unidades geradoras (máquinas síncronas), 45 barras e 73 linhas de transmissão (Figura 7.1), os dados referentes a tal sistema encontram-se no Apêndice C.

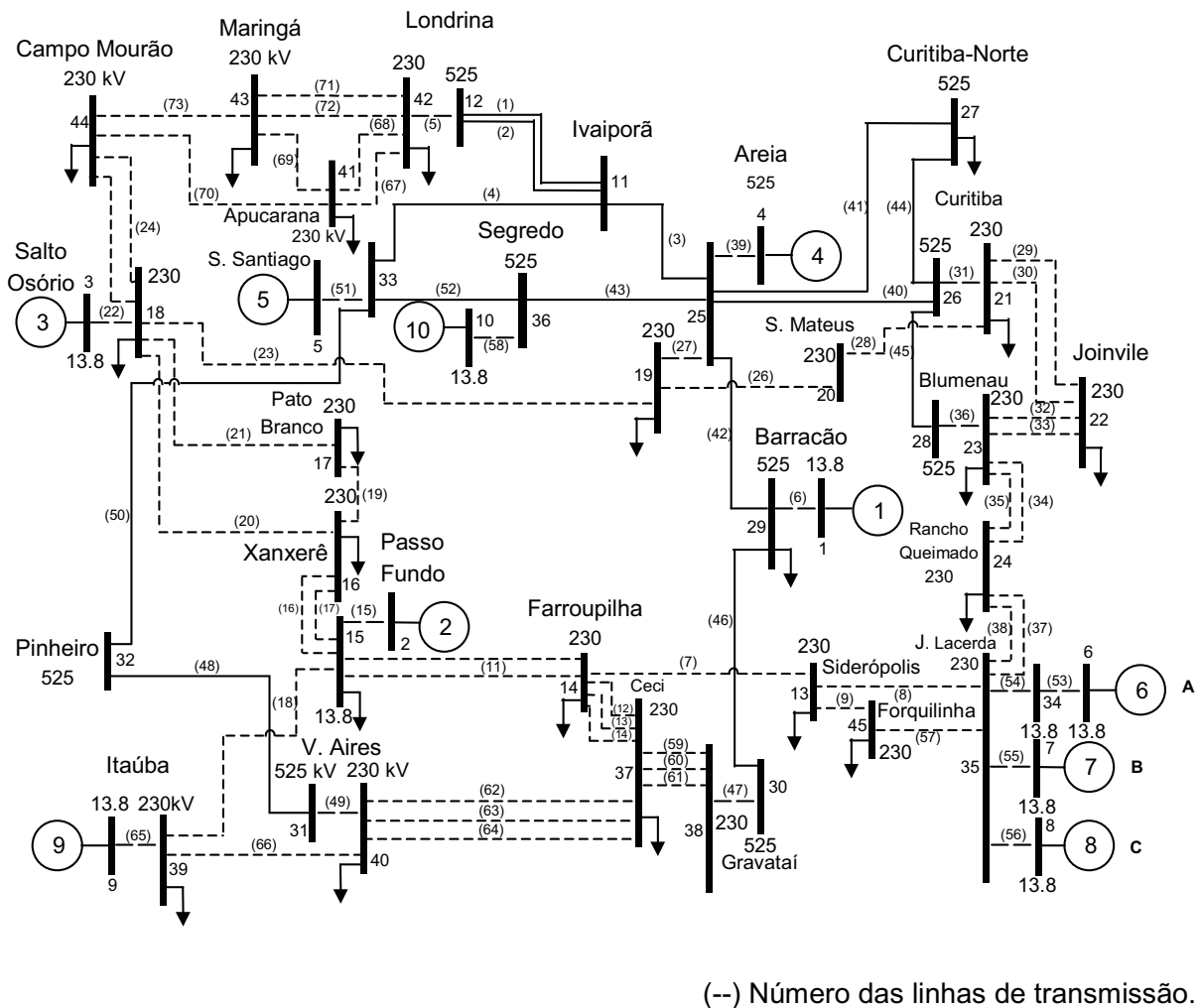


Figura 7.1: Configuração do Sistema Elétrico da Região Sul do Brasil.

7.3 Análise Experimental

Para a obtenção dos dados para o treinamento/análise da rede neural deste trabalho, foi utilizado o programa *Simul* (encontra-se implantado no Laboratório Computacional do Departamento de Engenharia Elétrica – CISA – UNESP). Apesar de durante todo o treinamento já ocorrer a análise, apenas os dados referentes à análise foram considerados para definirmos o índice de acerto e, isto, antes da execução do treinamento continuado pois, após sua execução, a taxa de acerto é de 100%. Este fato já era esperado visto que se trata do treinamento continuado.

Para o treinamento inicial da rede neural utilizou-se um total de 2615 dados, gerados pseudo-aleatoriamente, de perfis de geração/carga como forma de simular uma grande quantidade de situações operativas dos sistemas e, deste modo, a obtenção dos índices de segurança correspondentes. Para a análise foram considerados um total de 718 dados, destes 607 estavam corretos, assim obteve-se um total de quase 85% de acertos. Como a representação da potência ativa e da potência reativa, vetores $\mathbf{P}$ e $\mathbf{Q}$, é muito difícil de visualizar, cada dado possui 45 valores de potência ativa e 45 valores de potência reativa, estes serão representados utilizando a semente de geração e o percentual em relação ao **caso base**. A barra de falta indica a barra sob curto-circuito enquanto que as barras inicial e final, representam a topologia do sistema, isto é, as barras que saíram de operação. A margem de segurança obtida para cada dado apresentado ao sistema foi obtida através do programa *Simul* e, para melhor interpretação do resultado, apresentou-se o intervalo da margem de segurança obtida pelo *Simul*, pela rede neural desenvolvida neste trabalho antes e depois do treinamento continuado.

Para realização da simulação foi utilizada a linguagem FORTRAN 4.0, sistema operacional: Windows XP, processador: Pentium Dual-Core T2310 1,73 GHz, memória: 1GB.

Os parâmetros utilizados na rede neural ART-ARTMAP *Fuzzy* Euclidiana Modificada com Treinamento Continuado, determinados experimentalmente, foram os seguintes:

- Parâmetro de vigilância da rede ART Euclidiana Modificada com Treinamento Continuado: $\rho = 0,01$;
- Parâmetro de vigilância do módulo ART_a , da rede ARTMAP *Fuzzy* Modificada: $\rho_a = 0,90$;
- Parâmetro de vigilância do módulo ART_b , da rede ARTMAP *Fuzzy* Modificada: $\rho_b = 1,00$;
- Parâmetro de vigilância do módulo inter-ART, da rede ART *Fuzzy* Modificada: $\rho_{ab} = 0,90$;
- Taxa de aprendizagem da rede ARTMAP *Fuzzy* Modificada: $\alpha = 0,01$;
- Taxa de treinamento: $\beta = 0,20$;
- Incremento do parâmetro de vigilância do módulo ART_a : $\epsilon = 0,001$;
- Parâmetro para Permanência: $NMIN = 5$;
- Parâmetro para Exclusão: $NMAX = 1500$;
- Índice de Novidade: $\eta = 2,0$;

- Número de Intervalos de Segurança (IMS): 15.

A seguir, apresentam-se os resultados da análise da estabilidade transitória obtidos por simulação (resultados de referência), rede neural ART-ARTMAP *Fuzzy* Euclidiana Modificada com Treinamento Continuado, antes e depois da execução do treinamento continuado:

Tabela 7.1: Resultados da Análise da Estabilidade Transitória Obtidos por Simulação.

Semente	%	Barra de Falta	Barra Inicial	Barra de Final	$\mathcal{M}$	<i>Simul</i>	Previsão Inicial	Previsão Após Treinamento Continuado
335	30,0	15	14	15	-2,4476	2	2	2
59	27,5	15	14	15	0,8338	12	12	12
713	25,0	15	14	15	-0,8887	4	4	4
221	22,5	15	14	15	-0,1751	6	6	6
357	17,5	15	14	15	0,6300	9	9	9
90	15,0	15	14	15	0,1545	7	3	7
500	12,5	15	14	15	0,6605	10	10	10
365	10,0	15	14	15	0,8344	12	12	12
189	7,5	15	14	15	0,8360	12	12	12
20	2,5	15	14	15	0,7983	11	11	11
55	-7,5	15	14	15	0,9372	15	15	15
156	30,0	15	15	16	-0,6220	5	5	5
913	25,0	15	15	16	-3,7302	1	1	1
221	22,5	15	15	16	-0,2207	5	5	5
91	20,0	15	15	16	-0,3022	5	5	5
357	17,5	15	15	16	0,6118	9	9	9
89	12,5	15	15	16	0,4897	8	8	8
48	-7,5	15	15	16	0,8412	12	12	12
23	-10,0	15	15	16	0,9530	15	15	15
215	-12,5	15	15	16	0,8854	13	13	13
323	-15,0	15	15	16	0,9344	15	15	15
131	15,0	15	15	16	0,7596	11	11	11
365	10,0	15	15	16	0,8187	12	12	12
371	5,0	15	15	16	0,7225	10	10	10
59	-10,0	15	15	16	0,8216	12	12	12
193	27,5	16	15	16	0,7120	10	11	10
2	22,5	16	15	16	0,8805	13	13	13
515	17,5	16	16	18	0,6796	10	10	10
350	15,0	16	16	18	0,4356	8	8	8
712	10,0	16	16	18	0,8436	12	12	12
360	12,5	16	16	18	0,7830	11	11	11
15	7,5	16	16	18	0,8618	13	13	13

Tabela 7.3 - Continuação.

Semente	%	Barra de Falta	Barra Inicial	Barra de Final	$\mathcal{M}$	<i>Simul</i>	Previsão Inicial	Previsão Após Treinamento Continuado
39	7,5	16	16	18	0,8127	12	12	12
613	5,0	16	16	18	0,7987	11	11	11
435	2,5	16	16	18	0,8289	12	12	12
8	-5,0	16	16	18	0,8495	12	12	12
888	-7,5	16	16	18	0,7621	11	11	11
23	-10,0	16	16	18	0,8728	13	13	13
117	-12,5	16	16	18	0,8023	12	12	12
188	-15,0	16	16	18	0,7194	10	10	10
620	-20,0	16	16	18	0,8879	13	13	13
67	30,0	17	16	17	0,8376	12	11	12
32	27,5	17	16	17	0,8267	12	12	12
85	22,5	17	16	17	0,8219	12	12	12
60	25,0	17	16	17	0,7908	11	12	11
15	20,0	17	16	17	0,8382	12	12	12
666	17,5	17	16	17	0,7625	11	11	11
90	15,0	17	16	17	0,8670	13	13	13
80	12,5	17	16	17	0,8752	13	13	13
401	10,0	17	16	17	0,8368	12	12	12
331	7,5	17	16	17	0,8585	13	13	13
63	5,0	17	16	17	0,8047	12	12	12
20	2,5	17	16	17	0,8579	13	13	13
607	-2,5	17	16	17	0,8369	12	12	12
351	-5,0	17	16	17	0,8234	12	12	12
177	-7,5	17	16	17	0,8811	13	13	13
17	-10,0	17	16	17	0,8113	12	12	12
139	-12,5	17	16	17	0,8590	13	13	13
547	-15,0	17	16	17	0,9186	14	14	14
361	-17,5	17	16	17	0,7980	11	11	11
617	-20,0	17	16	17	0,6584	10	10	10
67	30,0	17	17	18	0,7870	11	12	11
22	25,0	17	17	18	0,6207	9	9	9
221	22,5	17	17	18	0,3560	8	8	8
662	20,0	17	17	18	0,7740	11	11	11
12	7,5	18	17	18	-0,5579	5	5	5

Tabela 7.3 - Continuação.

Semente	%	Barra de Falta	Barra Inicial	Barra de Final	$\mathcal{M}$	$Simul$	Previsão Inicial	Previsão Após Treinamento Continuado
503	5,0	18	17	18	-0,7291	4	4	4
629	2,5	18	17	18	-0,5590	5	6	5
767	-5,0	18	17	18	-0,4526	5	5	5
23	-10,0	18	17	18	0,1328	7	7	7
21	27,5	18	18	44	-1,7017	3	3	3
131	15,0	18	18	44	-1,9208	3	3	3
210	17,5	18	18	44	-1,0897	4	4	4
66	12,5	18	18	44	-1,0218	4	4	4
401	10,0	18	18	44	-0,6398	4	4	4
189	7,5	18	18	44	-0,9506	4	4	4
825	5,0	18	18	44	-0,4333	5	5	5
96	2,5	18	18	44	-0,7202	4	4	4
307	-2,5	18	18	44	-0,2761	5	5	5
25	-5,0	18	18	44	-0,6234	5	5	5
888	-7,5	18	18	44	-0,7123	4	4	4
757	-17,5	18	18	44	-0,3787	5	5	5
363	-17,5	18	18	44	-0,3385	5	5	5
97	17,5	17	17	18	0,7525	11	11	11
285	15,0	17	17	18	0,7329	11	11	11
24	10,0	17	17	18	0,7592	11	11	11
331	7,5	17	17	18	0,8323	12	12	12
250	5,0	17	17	18	0,8146	12	12	12
503	5,0	17	17	18	0,7969	11	11	11
629	2,5	17	17	18	0,8224	12	12	12
307	-2,5	17	17	18	0,8466	12	12	12
351	-5,0	17	17	18	0,8052	12	12	12
48	-7,5	17	17	18	0,8903	13	13	13
17	-10,0	17	17	18	0,8231	12	12	12
139	-12,5	17	17	18	0,8735	13	13	13
129	-15,0	17	17	18	0,8294	12	12	12
361	-17,5	17	17	18	0,8024	12	12	12
817	-20,0	17	17	18	0,8050	12	12	12
32	27,5	18	17	18	-0,8433	4	6	4
85	22,5	18	17	18	-0,4182	5	4	5
662	20,0	18	17	18	-0,3678	5	5	5
283	17,5	18	17	18	-0,5601	5	6	5
739	15,0	18	17	18	-1,2389	3	3	3
222	10,0	18	17	18	-0,9979	4	4	4
817	-20,0	18	18	44	-0,3548	5	5	5
789	30,0	33	33	36	-0,0916	6	6	6
193	27,5	33	33	36	0,0426	6	6	6
73	25,0	33	33	36	-0,1017	6	6	6

Tabela 7.3 - Continuação.

Semente	%	Barra de Falta	Barra Inicial	Barra de Final	$\mathcal{M}$	<i>Simul</i>	Previsão Inicial	Previsão Após Treinamento Continuado
385	22,5	33	33	36	0,3307	7	7	7
224	20,0	33	33	36	0,4827	8	8	8
999	17,5	33	33	36	0,5949	9	9	9
605	12,5	33	33	36	0,6899	10	10	10
222	10,0	33	33	36	0,6118	9	6	9
189	7,5	33	33	36	0,6519	10	9	10
135	-2,5	33	33	36	0,7675	11	11	11
25	-5,0	33	33	36	0,7505	11	11	11
338	-7,5	33	33	36	0,7309	10	10	10
23	-10,0	33	33	36	0,7114	10	10	10
129	-12,5	33	33	36	0,8115	12	12	12
991	-15,0	33	33	36	0,7212	10	10	10
817	-20,0	33	33	36	0,5866	9	9	9
32	27,5	35	13	35	-3,6964	1	2	1
291	25,0	35	13	35	-7,1970	1	1	1
98	25,0	35	13	35	-2,1020	2	2	2
915	20,0	35	13	35	-1,2752	3	5	3
666	17,5	35	13	35	-3,0525	1	1	1
421	15,0	35	13	35	-4,6189	1	1	1
80	12,5	35	13	35	-0,7247	4	3	4
120	10,0	35	13	35	-1,0181	4	4	4
309	7,5	35	13	35	0,1744	7	7	7
825	5,0	35	13	35	0,5523	9	4	9
435	2,5	35	13	35	0,7880	11	11	11
933	-2,5	35	13	35	0,8990	14	14	14
203	-5,0	35	13	35	0,9441	15	15	15
888	-7,5	35	13	35	0,9562	15	15	15
342	-10,0	35	13	35	0,6008	9	5	9
409	-12,5	35	13	35	0,9849	15	15	15
991	-15,0	35	13	35	0,5218	9	9	9
817	-20,0	35	13	35	-0,1849	6	5	6
18	30,0	44	18	44	-0,3739	5	5	5
77	27,5	44	18	44	0,7698	11	11	11
913	25,0	44	18	44	0,7998	11	11	11
221	22,5	44	18	44	0,4485	8	8	8
915	20,0	44	18	44	0,7868	11	11	11
701	17,5	44	18	44	0,6877	10	10	10
421	15,0	44	18	44	0,8485	12	12	12
383	12,5	44	18	44	0,8405	12	12	12
51	10,0	44	18	44	0,7881	11	11	11
259	7,5	44	18	44	0,7534	11	11	11
931	5,0	44	18	44	0,7828	11	11	11

Tabela 7.3 - Continuação.

Semente	%	Barra de Falta	Barra Inicial	Barra de Final	$\mathcal{M}$	<i>Simul</i>	Previsão Inicial	Previsão Após Treinamento Continuado
391	-5,0	44	18	44	0,8428	12	12	12
55	-7,5	44	18	44	0,8387	12	11	12
17	-10,0	44	18	44	0,8315	12	12	12
954	-12,5	44	18	44	0,9050	14	14	14
323	-15,0	44	18	44	0,8638	13	13	13
223	-17,5	44	18	44	0,7965	11	11	11
156	30,0	44	41	44	-0,5096	5	5	5
59	27,5	44	41	44	0,7846	11	11	11
713	25,0	44	41	44	0,4538	8	6	8
810	22,5	44	41	44	0,7495	11	11	11
38	20,0	44	41	44	0,7496	11	11	11
83	17,5	44	41	44	0,6873	10	10	10
671	15,0	44	41	44	0,8278	12	12	12
605	12,5	44	41	44	0,8012	12	12	12
24	10,0	44	41	44	0,8740	13	13	13
191	7,5	44	41	44	0,7848	11	11	11
377	5,0	44	41	44	0,8558	13	13	13
152	2,5	44	41	44	0,8961	14	14	14
200	0,0	44	41	44	0,8998	14	14	14
41	-2,5	44	41	44	0,9063	14	14	14
11	-5,0	44	41	44	0,9091	14	14	14
588	-7,5	44	41	44	0,9239	15	15	15
711	-10,0	44	41	44	0,9089	14	14	14
652	-12,5	44	41	44	0,8608	13	13	13
537	-15,0	44	41	44	0,9433	15	15	15
223	-17,5	44	41	44	0,8556	13	13	13
817	-20,0	44	41	44	0,8948	14	14	14
18	30,0	44	43	44	0,0299	6	6	6
590	27,5	44	43	44	-3,7487	1	1	1
322	25,0	44	43	44	0,6819	10	10	10
110	22,5	44	43	44	0,8886	13	13	13
4	20,0	44	43	44	0,8628	13	13	13
357	17,5	44	43	44	0,8182	12	12	12
350	15,0	44	43	44	0,7597	11	11	11
137	12,5	44	43	44	0,8545	13	13	13
120	10,0	44	43	44	0,8976	14	14	14
61	5,0	44	43	44	0,8470	12	12	12
9	2,5	44	43	44	0,8923	14	14	14
25	-5,0	44	43	44	0,8714	13	13	13
177	-7,5	44	43	44	0,9052	14	13	14
6	-10,0	44	43	44	0,9289	15	15	15
954	-12,5	44	43	44	0,9291	15	15	15

Tabela 7.3 - Continuação.

Semente	%	Barra de Falta	Barra Inicial	Barra de Final	$\mathcal{M}$	<i>Simul</i>	Previsão Inicial	Previsão Após Treinamento Continuado
129	-15,0	44	43	44	0,9250	15	15	15
757	-17,5	44	43	44	0,8863	13	13	13
445	-20,0	44	43	44	0,9140	14	14	14
789	30,0	15	15	39	0,6647	10	10	10
59	27,5	15	15	39	0,8087	12	13	12
98	25,0	15	15	39	0,3844	8	9	8
2	22,5	15	15	39	0,6685	10	10	10
378	20,0	15	15	39	0,7043	10	10	10
83	17,5	15	15	39	0,7478	11	11	11
601	15,0	15	15	39	0,7696	11	11	11
204	12,5	15	15	39	0,6613	10	10	10
120	10,0	15	15	39	0,7550	11	11	11
831	7,5	15	15	39	0,6891	10	10	10
503	5,0	15	15	39	0,6657	10	10	10
81	2,5	15	15	39	0,7340	11	11	11
607	-2,5	15	15	39	0,8702	13	13	13
37	-5,0	15	15	39	0,8391	12	11	12
55	-7,5	15	15	39	0,9296	15	15	15
379	-10,0	15	15	39	0,8605	13	13	13
833	-12,5	15	15	39	0,9388	15	14	15
732	-15,0	15	15	39	0,8971	14	14	14
788	-20,0	15	15	39	0,8039	12	12	12

A previsão de cada dado de entrada é realizada em menos de um décimo de segundo e, a previsão depois do treinamento contínuo é obtida após centésimos de segundo da primeira previsão, ou seja, quase que instantaneamente. A medida em que forem acrescentados novos dados a previsão tende a ser cada vez melhor tendo em vista que estes novos dados, caso passem pelo mecanismo de detecção de novidade, serão incorporados a rede. Além disso, ao contrário das redes neurais já utilizadas para a análise da estabilidade transitória, comporta tanto novas topologias quanto novas barras com falta.

Conclusão e Sugestão para Trabalhos Futuros

8.1 Conclusão

A principal contribuição desta pesquisa foi uma nova rede neural ART: a rede ART Euclidiana Modificada com Treinamento Continuado. Esta rede apresenta detecção de novidade, detecção de anomalia, além de não haver diferenciação entre as etapas de treinamento e de análise, pois ambos são executados simultaneamente.

A detecção de anomalia é realizada para que a rede não incorpore à matriz de pesos dados que apareçam durante sua execução menos que um número mínimo de vezes, representando assim apenas uma anomalia e não um dado importante o suficiente para ser incorporado a rede. Assim, se após N_{max} a partir primeira vez que foi apresentado um dado referente aquela categoria se o número de dados similares àquele padrão não atingir o número mínimo de vezes (N_{min}) para

ser incorporado definitivamente à matriz de pesos. Então este dado será rejeitado e, portanto, destruído pela rede.

A detecção de novidade visa o não sobre-treinamento dos padrões definitivos. Assim, só é ativada após o padrão temporário ser reconhecido como realmente importante à rede. Isto é feito utilizando a distância em relação ao centro do *cluster* que representa o dado, se esta for menor que o parâmetro η . Isto significa que o *cluster* já representa bem este dado e, portanto, não é necessário o treinamento em relação a este dado. Agora, se esta distância for maior do que η , há a necessidade realizar o treinamento em relação a este dado, para que esta nova informação seja incorporada à rede.

Agora, para a realização do treinamento continuado, após o primeiro dado de entrada que é automaticamente classificado como um *cluster* temporário e seu contador é inicializado, para cada novo padrão apresentado à rede é realizada uma primeira predição. Neste caso, usa-se o teste de similaridade em relação a todos os *clusters* temporários e definitivos sendo a saída inicial da rede o *cluster* mais similar à entrada. Em seguida é verificado se esta similaridade é o suficiente para ser incorporado à matriz de pesos. Caso não seja, é adicionado à rede um novo *cluster* temporário com os dados relativos a este padrão. Agora, se o padrão for suficientemente similar a um *cluster* já existente, é preciso verificar se este *cluster* é temporário ou definitivo. Caso seja temporário, é executado o treinamento. Este *cluster* passa a incorporar também as informações referentes a este padrão, se com a adição das informações contidas neste padrão ao *cluster* que o representa e, então, é atualizado o número de padrões representado pelo referido *cluster* temporário. Se após a atualização este número atingir N_{min} , este *cluster* se torna definitivo. Agora, se o *cluster* vencedor for definitivo é, então, verificada se a distância entre o centro deste *cluster* e o padrão supera o parâmetro η e, apenas neste caso, é realizado o treinamento em relação a este padrão. Tanto no caso do padrão pertencer a um *cluster* temporário quanto definitivo é realizada a nova previsão, já com os dados referentes ao treinamento do referido dado. É importante ressaltar que, no caso do *cluster* for definitivo e, não ocorrer o treinamento em relação ao dado de entrada, é realizada uma única previsão.

Além desta nova rede, foi incorporado à rede neural ART-ARTMAP Euclidiana com Treinamento Continuado um mecanismo de detecção de intruso para o reconhecimento de novas configurações em relação as já conhecidas pela rede. Este mecanismo impõe a rede condições rígidas para a detecção de uma configuração “estranha” à rede e, após identificada, aciona o programa *Simul* exigindo pelo menos N_0 dados de entrada referentes àquela configuração para que esta seja incorporada à rede.

Cabe ressaltar, ainda, que foi realizada uma modificação na rede neural ARTMAP *Fuzzy* para tornando o processo de busca do *cluster* vencedor mais rápido sem, entretanto, que ocorra a perda de eficiência da mesma.

O problema abordado para testar a rede neural proposta foi a análise da estabilidade transitória cujos dados são referentes a uma configuração do sistema sul-brasileiro (Figura 7.1) composto por 10 máquinas síncronas, 45 barras e 73 linhas de transmissão. Observando os resultados obtidos pode-se concluir que esta rede apresenta um excelente desempenho em relação às outras redes utilizadas para a previsão (confrontado com resultados disponíveis na literatura especializada), primeiramente porque o tempo de processamento é ínfimo e também porque a previsão inicial tem uma boa precisão de acerto mas com a realização do treinamento continuado o índice de acerto se torna 100%.

8.2 Sugestões para Trabalhos Futuros

Os resultados obtidos nesta pesquisa podem ser considerados satisfatórios (do ponto de vista de precisão e rapidez na obtenção de soluções), porém, sugerem-se algumas melhorias e procedimentos que podem ser implementados com o propósito de tornar a análise ainda mais eficiente através da rede ART-ARTMAP *Fuzzy* Euclidiana Modificada com Treinamento Continuado:

1. introduzir novos dispositivos e procedimentos à rede neural desenvolvida, visando a melhoria das soluções:
 - melhorar a função de similaridade do módulo ART Euclidiano com Treinamento Continuado;
 - inserir um mecanismo para que não haja a necessidade do parâmetro de vigilância e, assim, ao invés de se ter um único grau de similaridade, a própria rede define o grau de similaridade para cada classe gerada;
 - incluir novos mecanismos para a geração de classes e a sintonia (*match tracking*) entre a entrada e a saída da rede neural;
 - criar um mecanismo de “esquecimento” para que dados que não sejam acessados por muitíssimo tempo sejam retirados da matriz de pesos definitiva, isto sem comprometer os casos em que há instabilidade no sistema;

2. empregá-la no diagnóstico da estabilidade transitória para modelos mais elaborados do que o modelo clássico, *e.g.*, o modelo completo de Park [1];
3. empregar a rede neural ART Euclidiana com Treinamento Continuado na previsão de cargas elétricas multinodais para aplicações em sistemas de distribuição de energia elétrica;
4. aplicar a rede neural proposta para a resolução de outros problemas do mundo real.

Referências

- [1] ANDERSON, P. M.; FOUAD, A. A. **Power system control and stability**. 2.ed. New York: IEEE Press, 2003. 658p. (IEEE Series on Power Engineering).
- [2] ATHAY, T. et al. Transient energy stability analysis. In: CONFERENCE ON SYSTEM ENGINEERING FOR POWER: EMERGENCY OPERATING STATE CONTROL, 1979, Davos. **Proceedings...** Davos: Department of Energy Publication, 1979. (No. CONF-79004-P1).
- [3] BARTFAI, G. An improved learning algorithm for the fuzzy ARTMAP neural network, In: IEEE ARTIFICIAL NEURAL NETWORKS AND EXPERT SYSTEMS, 1995, New Zealand. **Proceedings...**, New York: IEEE, 1995. p. 34-37.
- [4] BARTFAI, G. Hierarchical clustering with ART neural network. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.2, p.940-944.
- [5] BRAGA, A. P.; LUDERMIR, T. B.; CARVALHO, A. C. P. L. F. **Redes neurais artificiais, teorias e aplicações**. Rio de Janeiro: LTC, 2000. 262 p.
- [6] CARPENTER, G. A.; GROSSBERG, S. A massively parallel architecture for a self-organizing neural pattern recognition machine. **Computer Vision, Graphics, and Image Processing**, [S.l.], v.37, n.1, p.54-115, 1987.
- [7] CARPENTER, G. A.; GROSSBERG, S. ART2: Stable self-organization of pattern recognition codes for analog input patterns. **Applied Optics**, New York, v.26, n.23, p.4919-4930, 1987.
- [8] CARPENTER, G. A.; GROSSBERG, S. ART3: hierarchical search using chemical transmitter in self-organization pattern recognition architectures. **Neural Network**, New York, v.3, n.4, p.129-152, 1990.
- [9] CARPENTER, G. A.; GROSSBERG, S.; ROSEN, D. B. Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system. **Neural Network**, New York, v.4, n.1, p. 759-571, 1991.

- [10] CARPENTER, G. A.; GROSSBERG, S.; REYNOLDS, J. H. ARTMAP: Supervised real-learning and classification of non-stationary data by a self-organizing neural network. **Neural Network**, New York, v.4, n.5, p.565-588, 1991.
- [11] CARPENTER, G. A. et al. Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multidimensional maps. **IEEE Transactions on Neural Networks**, New York, v.3, n.5, p.698-713, 1992.
- [12] CYBENKO, G. **Continuous valued neural networks with two hidden layers are sufficient**. Medford: Department of Computer Science, Tufts University, 1989. (Relatório Técnico).
- [13] ELGERD, O. I. **Introdução à teoria de sistemas de energia elétrica**. São Paulo: MacGraw-Hill, 1978. 606 p.
- [14] ENGLISCH, H.; HIEMSTRA, Y. The correlation as cost function in neural network. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994, v.5, p.3170-3172.
- [15] FAUSETT, L. **Fundamentals of neural networks: architectures, algorithms, and applications**. Englewood Cliffs: Prentice Hall, 1994. 461 p.
- [16] FERREIRA, W. P. et al. Transient stability analysis of electric systems via a fuzzy ART-ARTMAP neural network. **Electric Power Systems Research**, Lausanne, v.76, n.5, p.466-475, 2006.
- [17] FERREIRA, W. P. et al. Análise de estabilidade transitória de sistemas de energia elétrica por rede neural baseada na Teoria da Ressonância Adaptativa. In: LATIN AMERICAN CONGRESS ON ELECTRICITY, GENERATION AND TRANSMISSION, 5., 2003, São Pedro. **Proceedings...** São Paulo: UNESP, 2003. (Artigo B-135)
- [18] FONSECA, L. G. S.; DECKER, I. C. Iterative algorithm for critical energy determination in transient stability of power system. In: SYMPOSIUM PLANNING AND OPERATION IN ELECTRIC ENERGY SYSTEM, 1985, Rio de Janeiro, **Proceedings...** New York: Pergamon Press, 1986. p.483-489.
- [19] FONSECA, L. G. S.; MINUSSI, C. R. Redespacho de geração para controle de segurança dinâmica em sistemas de energia elétrica através da análise de sensibilidade, In: CONGRESSO BRASILEIRO DE AUTOMÁTICA, 8., 1990, Belém. **Anais...** [S.l.: s.n.] 1990. p.735-740,
- [20] FREITAS FILHO, W. **Controle de segurança dinâmica de sistemas de energia elétrica via análise de sensibilidade de redes neurais**. 1996. 83f. Dissertação (Mestrado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Universidade Estadual Paulista "Júlio de Mesquita Filho- UNESP, Ilha Solteira, 1996.
- [21] FU, L. **Neural networks in computer intelligence**. New York: MacGraw Hill, 1994. 460 p.
- [22] GROSSBERG, S. Nonlinear neural networks: principles, mechanisms, and architectures. **Neural Networks**, New York, v.1, n.1, p.17-61, 1988.
- [23] GUPTA, S.; ZIA, R. K. P. Quantum neural networks. **Journal of Computer and System Sciences**, New York, v.63, n.3, p.355-383, 2001.

- [24] HAYKIN, S. **Neural networks: a comprehensive foundation**. New Jersey: Prentice-Hall, 1994. 696 p.
- [25] HEBB, D. O. **The organization of behavior: a neurophysiological theory**, New York: Wiley, 1961. 335 p.
- [26] HEILEMAN, G. L.; GEORGIPOULOS, M.; HWANG, J. A survey of learning results for ART1 networks. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.2, p.1222-1225.
- [27] HOPFIELD, J.J. Neural networks and physical systems with emergent collective computational abilities, **Proceedings of the National Academy of Sciences**, Washington, v.79, n.8, p.2554-2558, 1982.
- [28] IEEE - TASK FORCE, Proposed Terms Definitions for Power System Stability, **IEEE Transactions on Power Apparatus and Systems**, New York, v.101, n.7, p.1894-1898, 1982.
- [29] JAIN, A. K.; MAO, J.; MOHIUDDIN, K. M. Artificial neural networks: a tutorial. **IEEE Computer**, New York, v.29, n.3, p.31-44, 1996.
- [30] KANE, R.; MILGRAM, M. Financial forecasting and rules extraction from trained networks. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.5, p.3190-3195.
- [31] KENAYA, R. Euclidean ART neural networks. WORLD CONGRESS ON ENGINEERING AND COMPUTER SCIENCE, 2008, San Francisco. **Proceedings...** San Francisco: Newswood, 2008. p.1-6.
- [32] KLIR, G.; YUAN, B. **Fuzzy sets and fuzzy logic: theory and applications**. New Jersey: Prentice Hall, 1995. 574 p.
- [33] KOHONEN, T. Self-organized formation of topologically correct feature maps. **Biological Cybernetics**, New York, v.43, n.1, p.59-69, 1982.
- [34] KOHONEN, T. **Self-organization and associative memory**. 3.ed. New York: Springer-Verlag, 1989. 312 p.
- [35] KUNDUR, P. **Power system stability and control**. New York: McGraw-Hill, 1994. 1176 p.
- [36] LOPES, M. L. M. **Desenvolvimento de uma rede neural para previsão de cargas em sistemas de distribuição de energia elétrica**. 2005. 149f. Tese (Doutorado em Engenharia Elétrica)- Departamento de Engenharia Elétrica, Universidade Estadual Paulista "Júlio de Mesquita Filho- UNESP, Ilha Solteira, 2005.
- [37] MAHONEY, B. The limits of a computational model for human thinking. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.4, p.2173.
- [38] MARCHIORI, S. C. **Desenvolvimento de um sistema para análise da estabilidade transitória de sistemas de energia elétrica via redes neurais**. 2006. 110f. Tese (Doutorado em Engenharia Elétrica)- Departamento de Engenharia Elétrica, Universidade Estadual Paulista "Júlio de Mesquita Filho- UNESP, Ilha Solteira, 2006.

- [39] MARCHIORI, S. C.; MINUSSI, C. R.; LOTUFO, A. D. P. Rede neural ARTMAP nebulosa para análise de estabilidade transitória de sistemas de energia elétrica, **Learning and Nonlinear Models**, [S.l.], v.1, n.1, p.49-65, 2003.
- [40] McCULLOCH, W. S.; PITTS, W. A logical calculus of ideas immanent in nervous activity, **Bulletin of Mathematical Biophysics**, New York, v.5, p.115-133, 1943.
- [41] MINSKY, M. L.; PAPERT, S. A. **Perceptrons**: an introduction to computational geometry. Cambridge: MIT Press, 1969. 258 p.
- [42] MINUSSI, C. R. **Controle de segurança dinâmica em sistemas de engenharia elétrica**. 1990. 156f. Tese (Doutorado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Universidade Federal de Santa Cruz, Florianópolis, 1990.
- [43] MINUSSI, C. R.; FREITAS FILHO, W. Sensitivity analysis for transient stability. **IEE Proceedings on Generation, Transmission and Distribution**, London, v.145, n.6, p.669-674, 1998.
- [44] MINUSSI, C. R.; SILVEIRA, M. C. G. Transient stability analysis of electric power systems by neural networks, In: IEEE MIDWEST SYMPOSIUM ON CIRCUITS AND SYSTEMS, 38., 1995, Rio de Janeiro. **Proceedings...** New York: IEEE, 1995. p.1305-1308.
- [45] MITRA, S. Fuzzy inferencing with ART network. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Proceedings...** New York, 1994. v.2, p.1230-1234.
- [46] MONTICELLI, A. J. **Fluxo de carga em redes de engenharia elétrica**. São Paulo: Edgard Blücher, 1983. 164 p.
- [47] MORENO, A. L.; MINUSSI, C. R. Análise da estabilidade transitória de sistemas de energia elétrica usando uma rede neural ART euclidiana ARTMAP com treinamento continuado. In: INTERNATIONAL CONFERENCE OF ENGINEERING AND TECHNOLOGY EDUCATION, 11., 2010, Ilhéus. **Proceedings...** New York: IEEE, 2010. p.622-627.
- [48] MORENO, A. L.; MINUSSI, C. R. Análise da estabilidade transitória de sistemas elétricos usando uma rede neural euclidiana ARTARTMAP. In: SEMINÁRIO BRASILEIRO DE ANÁLISE, 70, 2009. São Paulo. **Anais...** São Paulo: IME-USP, 2009. p.85-97.
- [49] ORTEGA, N. R. S. **Aplicação da teoria de conjuntos fuzzy a problemas da biomedicina**. 2001. 342f. Tese (Doutorado em Ciências) - Instituto de Física, Universidade São Paulo - USP, São Paulo, 2001.
- [50] PAI, M. A. **Power system stability**. Amsterdam: North-Holland, 1981. 251 p.
- [51] PARK, Y. An ART2 trained by two-stage learning on circularly ordered data sequence. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, Orlando, 1994. **Annals...** New York: IEEE, 1994. v.5, p.2928-2933.
- [52] ROSENBLATT, F. The Perceptron: A probabilistic model for information storage and organization in the brain. **Psychological Review**, Washington, v.65, n.6, p.386-408, 1958.
- [53] SARLE, W.S. **Neural network FAQ**. Cary: [s.n.], 2002. Disponível em: <ftp.sas.com/pub/neural/FAQ/txt/zip.>. Acesso em: 05 Jul.2008.

- [54] SILVEIRA, M. C. G. **Análise de estabilidade transitória de sistemas de energia elétrica por redes neurais ARTMAP nebulosas modulares**. 2003. 109f. Tese (Doutorado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Universidade Estadual Paulista "Júlio de Mesquita Filho- UNESP, Ilha Solteira, 2003.
- [55] SILVEIRA, M. C. G. et al. Análise de estabilidade transitória de sistemas de energia elétrica usando uma rede neural ARTMAP nebulosa. In: CONGRESSO BRASILEIRO DE AUTOMÁTICA, 14, 2002, Natal. **Proceedings...** [S.l.:s.n.], 2002. p.3171-3177.
- [56] SILVEIRA, M. C. G.; LOTUFO, A. D. P.; MINUSSI, C. R. Transient stability analysis of electrical power Systems using a neural network based on fuzzy ARTMAP. In: IEEE BOLOGNA POWER TECH 2003 CONFERENCE, 2003, Bologna. **Proceedings...** New York: IEE, 2003. 7p. (To appear).
- [57] SMITH, K. A.; GUPTA, J. N. D. Neural networks in business: techniques and applications for the operations researcher. **Computers Operations Research**, New York, v.27, n.1, p.1023-1044.
- [58] SPECHT, D.F.; ROMSDAHL, H. Experience with adaptive probabilistic neural network and adaptive general regression neural networks. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.2, p.1203-1208.
- [59] STEVENSON JUNIOR, W. D. **Elementos de análise de sistemas de potência**. São Paulo: McGraw-Hill, 1978. 347 p.
- [60] STOTT, B.; ALSAC, O.; MONTICELLI, A. J. Security analysis and optimization. **Proceedings of the IEEE**, New York, v.75, n.12, p.1623-1644, 1987.
- [61] TAKAHASHI, H. et.al. The reliability of neural networks on pattern recognition. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.5, p.3067-3070.
- [62] TAURITZ, D. R. **The adaptive resonance (ART) clearinghouse**. Leiden: [s.n.], 2005. Disponível em: <<http://www.wi.leidenuniv.nl/art/>>. Acesso em: 20 Abr.2007.
- [63] TSAI, W. Y.; TAI, H. M.; REYNOLDS, A. C. An ART2-BP neural net and its application to reservoir engineering. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.5, p.3289-3294.
- [64] VISHWAKARMA, K. P. A neural network to predict multiple economic time series. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.6, p.3674-3679.
- [65] WEIERICH, P.; VON ROSENBERG, M. The use formal measures for the training of linebreak hierarchical Kohonen maps. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1994, Orlando. **Annals...** New York: IEEE, 1994. v.2, p.612-615.
- [66] WERBOS, P. J. **Beyond regression: new tools for prediction and analysis in the behavioral sciences**. 1974. Thesis (Master) - Harvard University, Harvard, 1974.
- [67] WIDROW, B.; HOFF, M. E. **Adaptive switching circuits, IRE WESCON Convention Record**. Los Angeles: Western Electronic Show and Convention, 1960. p.96-104.

-
- [68] WIDROW, B. Generalization and information storage in network of adaline 'neurons'. In: YOVITZ, M. C.; JACOBI, G. T.; GOLDSTEINS, G. D. **Self-Organizing Systems**. Washington: Spartan Books, 1962. p.435-461.
- [69] YEN, J.; LAGARI, R. **Fuzzy logic: intelligence, control, and information**. New Jersey: Prentice Hall, 1999. 548 p.
- [70] ZADEH, L. Fuzzy sets. **Information an Control**, New York, v.8, n.3, p.338-353, 1965.

Redes Neurais Artificiais

Neste apêndice é apresentada uma relação dos fatos históricos que marcaram o surgimento e o desenvolvimento da área de pesquisa: Redes Neurais Artificiais. Em seguida fazem-se algumas considerações a respeito deste tipo de redes e do porquê estudá-las. A partir daí, são apresentados os modelos biológico e artificial de um neurônio e são analisados simultaneamente. Apresentam-se, a seguir, as principais estruturas e paradigmas de aprendizado das redes neurais. Finalmente, mostra-se uma pequena fração de aplicações de alguns tipos de redes neurais.

A.1 Visão Histórica

O marco inicial do nascimento da área de redes neurais artificiais foi a publicação do artigo intitulado “*A Logical Calculus of the Ideas Immanent in Neural Activity*” do psiquiatra e neuroanatomista Warren McCulloch e do matemático Walter Pitts (1943) [40]. McCulloch e Pitts estabelecem, neste trabalho, uma analogia entre o processo de comunicação das células nervosas vivas e o processo de comunicação por transmissão elétrica e propõem a criação de neurônios formais, descrevendo um cálculo lógico que unifica os estudos de neurofisiologia e da

lógica matemática. O neurônio artificial desenvolvido por eles tinha uma única saída, e esta era uma função binária (*threshold*) da soma dos valores de entrada. Em 1947 eles conseguem demonstrar que é possível conectar os neurônios formais e compor uma rede capaz de executar funções complexas.

Em 1949, o psicólogo Donald Hebb dá um passo extremamente importante na história das redes neurais ao elaborar uma teoria baseada no processo de aprendizagem que ocorre no cérebro humano. Em seu livro *The Organization of Behavior: A Neurophysiological Theory* [25], descreveu pela primeira vez o funcionamento quantitativo da sinapse e do processo de treinamento humano. Essas teorias, em especial seu postulado de aprendizagem, são usadas ainda hoje no desenvolvimento de modelos computacionais de sistemas de aprendizagem.

Frank Rosenblatt, do M.I.T. (Massachusetts Institute of Technology), em 1958 publicou o seu livro *Perceptrons* [52], onde apresenta o novo modelo chamado *perceptron*. Esse modelo possui três camadas: a primeira camada recebe os dados de entrada exteriores, possuindo conexões fixas, a segunda camada recebe impulsos da primeira através de conexões cuja eficiência de transmissão (peso) é ajustável, que, por sua vez, envia estímulos de saída para a terceira camada, onde será apresentada a resposta [5].

Com base no modelo de McCulloch e Pitts, em 1960, Widrow e Hoff [67] publicaram um artigo sobre o neurônio artificial chamado *Adaline* (ADaptive LINear Element), cuja diferença com o neurônio de McCulloch e Pitts está na regra de aprendizagem proposta: a regra Delta. Em 1962, Widrow [68] juntamente com seus estudantes propõem uma das primeiras redes neurais em camadas treináveis com múltiplos elementos adaptativos: a rede *Madaline* (*Multiple-ADALINE*).

Durante a década de 1960, grandes pesquisas foram realizadas, principalmente relacionadas aos perceptrons. Nessa época, tudo levava a crer que as redes neurais artificiais poderiam realizar qualquer trabalho, para isto bastaria construir uma rede neural suficientemente grande. Porém, em 1969, Minsky e Papert publicaram um trabalho também intitulado de *Perceptrons*, onde demonstraram matematicamente que o *perceptron* não era capaz de resolver algumas tarefas [41]. Essa publicação veio a desencorajar os pesquisadores e, principalmente, as agências financiadoras a desenvolver novas pesquisas com os *perceptrons*.

Em 1982, John Hopfield publicou um artigo onde propôs um modelo de rede em que as conexões dos neurônios são recorrentes garantindo que a rede obtivesse um estado estável na ausência de entradas externas [27]. Para tanto, Hopfield associou o estado da rede à uma função energia. A obtenção de um estado estável ocorre pela minimização da função energia, até alcançar um estado em que esta seja mínima (estável).

Também em 1982, Kohonen apresentou um modelo em que os neurônios estão conectados com aqueles posicionados mais próximos a eles estabelecendo um método de aprendizagem competitivo chamado de mapas auto-organizáveis (Self-Organizing Maps - SOM) [33]. Este tipo de comportamento baseado na competição entre os neurônios, em que o aprendizado é não-supervisionado, o modelo proposto é capaz de armazenar padrões e posteriormente recuperá-los, independentemente de ruídos. Com estes dois trabalhos, renova-se o interesse de alguns pesquisadores na área de redes neurais.

Somente em 1986 foi que a comunidade científica voltou a se interessar pelas redes neurais artificiais como ferramenta para reconhecimento de padrões com o trabalho de Werbos [66] ao aperfeiçoar a ideia do perceptron criando o algoritmo retropropagação do erro (*error backpropagation*). Mais tarde, Rumelhart et al. notabilizaram o algoritmo retropropagação explorando várias aplicações. O sucesso deste algoritmo estimulou o desenvolvimento de muitas pesquisas em redes neurais artificiais e de uma ampla variedade de modelos cognitivos [24].

Gail Carpenter e Stephen Grossberg, baseados em seu trabalho anterior sobre aprendizagem competitiva, criam o modelo ART (*Adaptive Resonance Theory*) em 1987 [6]. Basicamente, nesta teoria, temos uma camada de reconhecimento de baixo-para-cima (*bottom-up*) e uma camada generativa de cima-para-baixo (*top-down*). Se o padrão de entrada e o padrão realimentado aprendidos coincidirem, então ocorre um processo dinâmico chamado de “ressonância adaptativa”, isto é, ocorre amplificação e prolongamento da atividade neural. Este modelo utiliza um mecanismo de aprendizado não-supervisionado e competitivo criando protótipos (*clusters*) dos padrões aprendidos. Existem redes ART dos mais diversos tipos, utilizando paradigmas de aprendizado supervisionado e não-supervisionado. Entre os mais utilizados estão: ART1 [6, 7, 15], ART2 [7, 15], ARTMAP [6], Fuzzy ART [9].

Atualmente, as áreas de aplicação das redes neurais são várias. No final deste apêndice são citadas algumas aplicações. Na teoria, as redes neurais artificiais podem ser utilizadas em qualquer tipo de problema computacional. Porém, na prática, o que se tem visto é sua aplicação em problemas mais específicos que requerem mapeamentos mais complexos, onde se tem um grande volume de dados disponíveis mas não se consegue descrever regras para o comportamento do sistema. Outro ponto importante das redes neurais artificiais é a capacidade de ser tolerante a ruídos ou erros nos dados.

A.2 Algumas Considerações sobre Redes Neurais

Mas o que é realmente uma rede neural artificial? Tem-se a seguinte definição segundo Haykin [24]:

Definição A.1. *Uma rede neural é um processador maciça e paralelamente distribuído constituído de unidades de processamento simples, que tem a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. Ela se assemelha ao cérebro em dois aspectos:*

- 1. O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.*
- 2. Forças de conexão entre os neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.*

Assim, as redes neurais artificiais são modelos matemáticos que assemelham às estruturas neurais biológicas e que têm capacidade computacional adquirida por meio de aprendizado e generalização. As redes neurais são formadas por unidades de processamento também chamadas de neurônios, que podem estar distribuídas em uma ou em várias camadas. O estado de uma unidade, em um determinado instante, é representado por sua ativação, um valor geralmente no intervalo $[0, 1]$ ou $[-1, 1]$. A primeira camada da rede, chamada de entrada, possui unidades cujas ativações representam valores para os atributos do domínio do problema ao qual a rede está sendo aplicada. Atributos quantitativos são representados por uma unidade. Atributos qualitativos com n possíveis valores são representados por n unidades de entrada. As unidades de camada de saída representam as decisões feitas pela rede. Entre as unidades de entrada e de saída podem existir outras camadas, denominadas de camadas intermediárias. As unidades entre as camadas da rede estão conectadas por meio de pesos, no qual cada peso corresponde a um valor numérico.

Na verdade, as redes neurais artificiais procuram explorar os princípios adotados pelo cérebro, especialmente o paralelismo, em busca de implementar sistemas que possam solucionar problemas da mesma forma que este último. Existe uma grande variedade de arquiteturas e métodos de treinamento de redes neurais, tanto para problemas de aprendizado supervisionado como para não-supervisionado.

O procedimento utilizado para realizar o processo de aprendizagem é chamado de *algoritmo de aprendizagem*, cuja função é modificar os pesos sinápticos da rede de uma forma ordenada para alcançar um objetivo de projeto desejado.

A.2.1 Por Quê Utilizar Redes Neurais Artificiais?

Uma vantagem potencial das redes neurais artificiais é o **processamento altamente paralelo** que ocorre em sua estrutura. Existe outro aspecto porém, o qual coloca as redes neurais como uma classe muito especial de sistemas: a **capacidade de generalização** exibida pelas redes neurais. Por capacidade de generalização entende-se a capacidade de, na fase de diagnóstico, gerar respostas adequadas para entradas que nunca foram apresentadas à rede. Estes dois aspectos fazem com que as redes neurais sejam capazes de solucionar problemas altamente complexos.

A utilização de redes neurais artificiais proporciona algumas características muito úteis aos sistemas que as utilizam [21], a saber:

Não-linearidade: A não-linearidade é uma propriedade inerente ao neurônio, sendo extremamente importante nos casos onde o sinal (ou entrada) analisado é gerado por um dispositivo não-linear, como é a maioria dos sistemas na natureza.

Mapeamento entre a entrada e a saída: Um dos paradigmas de aprendizado mais utilizado é o aprendizado supervisionado, o qual promove a modificação dos pesos das conexões da rede, aplicando um conjunto de entradas com as respectivas saídas. Para cada uma destas entradas escolhida aleatoriamente, é determinada a modificação dos pesos de modo a minimizar a resposta desejada para a respectiva entrada. Desta forma, a rede aprende a partir de exemplos, construindo um mapeamento entre a entrada e a saída.

Adaptatibilidade: As redes neurais artificiais têm a capacidade natural de adaptar seus parâmetros, em resposta a mudanças no ambiente a sua volta, através do retreinamento. Contudo, se o ambiente é do tipo não-estacionário, a rede neural artificial pode ser construída de forma a adaptar os parâmetros de suas conexões em tempo real, em resposta às mudanças no ambiente. Um cuidado porém deve ser observado quanto a esta adaptatibilidade, uma vez que pode levar a perda de robustez da rede, ou seja, a rede neural artificial pode passar a responder a estímulos (ou entradas) às quais não deveria responder,

como por exemplo, ruídos. Este problema é conhecido como o dilema da **estabilidade-plasticidade**.

Tratamento de informação contextual: Dependendo da arquitetura adotada na fase de treinamento, a saída da rede é afetada pela atividade de todos os neurônios, e a correção dos parâmetros pode atingir potencialmente todas as conexões. Por exemplo, no caso dos pesos, este efeito faz com que a informação armazenada na rede seja influenciada por todo o conjunto de dados apresentado, considerando não apenas cada padrão apresentado mas também a correlação entre os mesmos, produzindo uma forma de informação contextual.

Tolerância a falhas: Em vários modelos de rede, a saída recebe contribuições de todos os neurônios. Caso haja falha em um neurônio ou em um conjunto de conexões, a saída será apenas degradada. As redes neurais artificiais tendem a não apresentar falhas catastróficas com a perda de alguns neurônios, ou seja, é bastante tolerante a falhas.

Robustez: As redes neurais artificiais permitem processar ruído ou informação incompleta de forma eficiente, assim são capazes de manter o seu desempenho quando há desativação de algumas das suas conexões.

A.3 Estruturas de Redes Neurais

Apresenta-se a seguir, o modelo de neurônio mais conhecido na literatura especializada, ou seja, o neurônio de McCulloch & Pitts. Ressalta-se que centenas de tipos de neurônios têm sido identificados, cada um dos quais distingue-se dos demais pela forma do corpo celular. Essas diferenças morfológicas exibem especificações funcionais importantes. A identificação das funções dos vários tipos de neurônios representa um dos tópicos mais importantes dos estudos referentes à compreensão do cérebro humano. Os resultados destes estudos poderão orientar o desenvolvimento de redes neurais artificiais mais eficientes, principalmente com relação à capacidade e velocidade de aprendizado.

A.4 Modelo de Neurônio Biológico e Modelo de Neurônio Artificial

A.4.1 Modelo Biológico

O sistema nervoso humano é formado por células nervosas denominadas neurônios. Estima-se que o cérebro humano possua cerca de 10^{12} neurônios no córtex, conectados num sistema altamente complexo de 10^{14} a 10^{15} conexões ou sinapses.

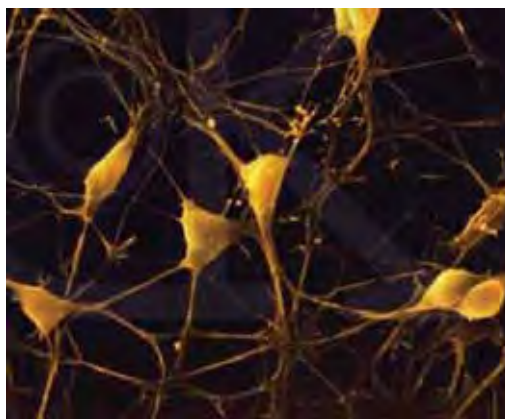


Figura A.1: Rede Neural.

Os neurônios biológicos atuam em conjunto na realização de suas funções, através da arquitetura paralela composta por bilhões de células realizando um processo concomitante. Esta formatação em particular torna o sistema nervoso capaz de realizar determinadas tarefas com resultados melhores do que os obtidos por um processamento computacional em série, por exemplo, modelagem e identificação de padrões e processamento de imagens e sinais [5]. Mas como este processamento ocorre no cérebro? Como é fisicamente cada uma dessas unidades de processamento denominadas neurônios?

Cada neurônio é formado por corpo celular (sôma no idioma grego), dendritos e axônios, e possui uma estrutura simples que permite três funções básicas: entrada, processamento e saída de sinais. Os dendritos têm volumes muitas vezes maiores do que o próprio corpo celular e formam uma árvore dendrital. Além disso, eles são os responsáveis pelo recebimento das informações ou impulsos nervosos de outros neurônios e de conduzi-los das extremidades para o corpo celular. Desta forma, eles são as conexões através das quais os sinais de entrada chegam aos neurônios. O corpo celular ou núcleo é o processador do neurônio, ou seja, ele processa os sinais recebidos pelos dendritos e, se o valor resultante estiver acima de um certo limite, o neurônio se excita e

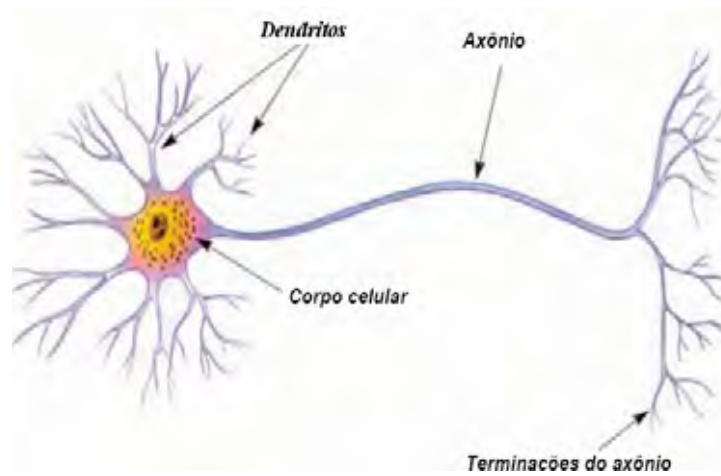


Figura A.2: Neurônio Biológico.

libera novos impulsos, que são transmitidos para a extremidade da célula através dos axônios, caso contrário, ele fica inibido. Os axônios servem como canal de saída do neurônio, eles estão ligados aos dendritos de outros neurônios através das sinapses, um axônio pode medir cerca de 0,1 milímetro podendo chegar a 1 metro. As sinapses não são ligações físicas, mas sim químicas e temporárias, e se dão através de neurotransmissores. A Figura A.2 é uma representação bastante simplificada de um neurônio biológico.

A.4.2 Neurônio Artificial

Os neurônios artificiais foram desenvolvidos baseados no funcionamento dos neurônios biológicos. Vários modelos foram propostos na literatura. A seguir apresenta-se o modelo de McCulloch Pitts, que é o mais empregado em problemas de reconhecimento de padrão.

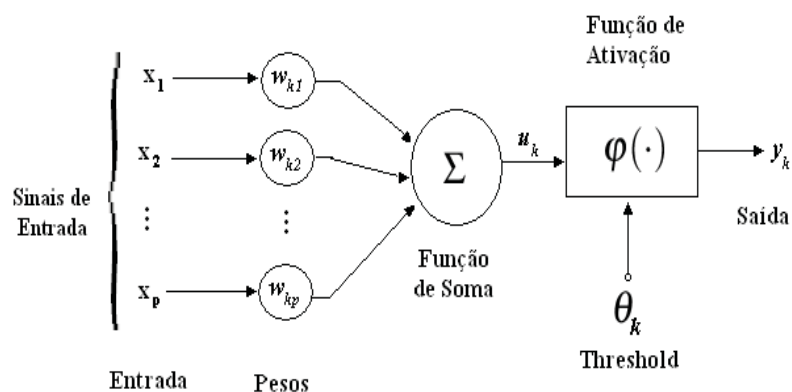


Figura A.3: Neurônio Artificial.

Neurônio de McCulloch-Pitts

O modelo de McCulloch-Pitts foi desenvolvido em 1943, o qual descreve um neurônio cuja atividade u_k é a soma de p entradas ponderadas por seus respectivos pesos. Esta atividade alimenta uma função não-linear $\varphi(\cdot)$ que produz um sinal que será enviado aos demais neurônios. As não-linearidades mais empregadas são: relé, lógica *threshold* e sigmoide. Apresenta-se na Figura A.3 um neurônio artificial que utiliza a não-linearidade lógica *threshold*.

Assim, a descrição deste neurônio artificial baseia-se em p terminais de entrada $x_1, x_2, \dots, x_p$, onde as características dos padrões são apresentadas à rede. Esses terminais de entrada são acoplados aos pesos $w_{k1}, w_{k2}, \dots, w_{kp}$, cujos valores podem ser positivos ou negativos. Tais valores estão relacionados com os tipos de sinapses utilizados pela rede. Se for utilizada uma sinapse inibitória, as trocas de informações entre os neurônios serão dificultadas ou impedidas, já se ocorrer uma sinapse excitatória, as passagens de informação entre os neurônios serão permitidas.

Depois de efetuada a multiplicação entre os terminais de entrada e os pesos, $x_j w_{kj}$, o corpo celular realiza a soma ponderada dos valores recebidos $x_j w_{kj}$ e decide se o neurônio deve ou não disparar. É aplicada então uma função de ativação, que ativa ou não a saída do neurônio, dependendo do valor da soma ponderada de suas entradas. A saída do neurônio poderá ser linear ou não-linear de acordo com a função de ativação utilizada. As funções de ativação mais usadas são as funções rampa, degrau, degrau bipolar, sigmoide, tangente hiperbólica e base radial, entre outras (Figura A.5). O neurônio terá sua saída ativada quando:

$$\sum_{i=0}^p x_i w_{ki} \geq \lambda \quad (\text{A.4.1})$$

em que:

- p é o número de neurônios da rede;
- x_i é a entrada da rede, $i = 1, 2, \dots, p$;
- w_{ki} é o peso associado à entrada x_i ;
- λ é o limiar de excitação (*threshold*) do neurônio.

O neurônio de McCulloch e Pitts apresenta apenas uma saída y_k , onde o resultado final é concluído e apresentado [24]. A Figura A.5 apresenta o neurônio artificial idealizado por McCulloch e Pitts. Os neurônios artificiais procuram simular o funcionamento dos neurônios biológicos e, para que isso aconteça, utilizam-se pesos sinápticos. Isso faz com que as conexões dos neurônios artificiais se comportem como conexões dos neurônios biológicos.

Função	Gráfico	Equação
Degrau Bipolar		$\begin{aligned} \text{Se } x > 0 &\rightarrow y = 1 \\ \text{Se } x = 0 &\rightarrow y = 0 \\ \text{Se } x < 0 &\rightarrow y = -1 \end{aligned}$
Linear		$y = x$
Tangente Hiperbólica		$y = \frac{1 - e^{-\beta x}}{1 + e^{\beta x}}$
Logística (Sigmoide)		$y = \frac{1}{1 + e^{-\beta x}}$
Degrau		$\begin{aligned} \text{Se } x \geq 0 &\rightarrow y = 1 \\ \text{Se } x < 0 &\rightarrow y = 0 \end{aligned}$
Rampa		$\begin{aligned} \text{Se } x > 1 &\rightarrow y = 1 \\ \text{Se } -1 \leq x \leq 1 &\rightarrow y = x \\ \text{Se } x < -1 &\rightarrow y = -1 \end{aligned}$
Base Radial		$y = e^{-x}$

Figura A.4: Funções de Ativação e suas Equações.

Resumindo, as redes neurais artificiais têm se desenvolvido inspirada na cognição humana ou na biologia neural, com base nas seguintes hipóteses genéricas:

- O processamento de informações ocorre em unidades simples de processamento denominadas neurônios;
- São transmitidos sinais entre os neurônios através de conexões (sinapses) existentes entre eles;
- Cada conexão tem associada a si uma ponderação, a qual, em uma rede neural típica, processa o sinal transmitido; o conjunto destas ponderações pode ser visto como armazenamento na rede, contido em uma memória distribuída;

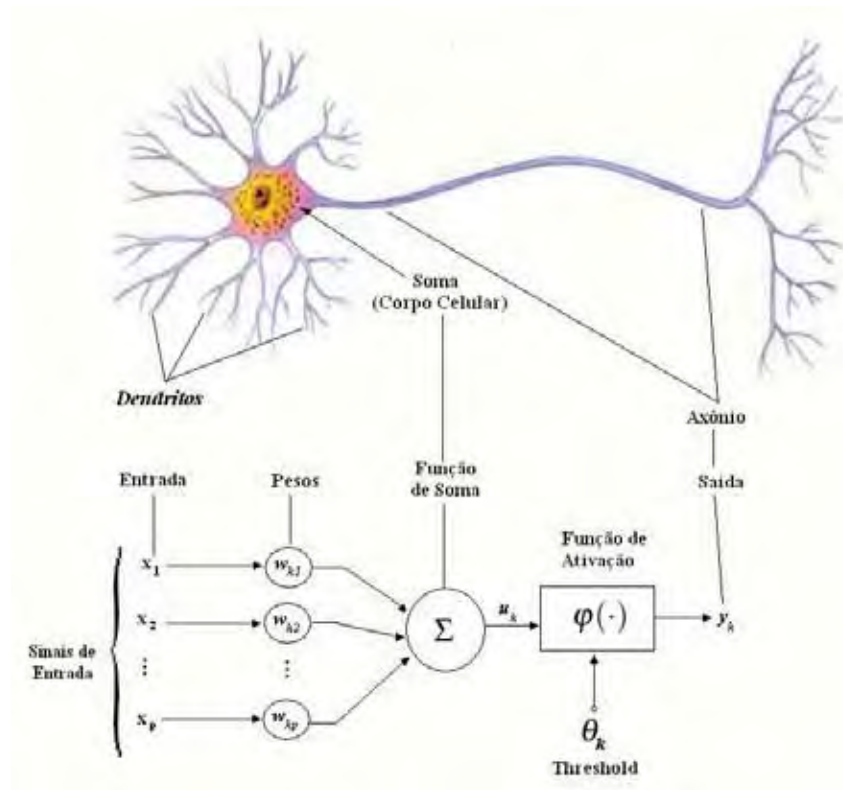


Figura A.5: Neurônio Biológico e Neurônio Artificial: Comparação.

- Neurônios que desempenham função semelhante durante o processamento da informação recebida são dispostos em uma mesma camada; assim, camadas sucessivas são criadas na rede neural com o intuito de se realizar, em cada uma delas, uma fase distinta do processamento (rede multicamadas).
- A cada neurônio se aplica uma função de ativação (geralmente não-linear) que computa seu sinal de saída a partir dos dados recebidos.

A.5 Arquitetura da Rede

A forma como os neurônios são organizados é denominada de arquitetura da rede neural. Os neurônios artificiais podem ser organizados e conectados de várias formas, resultando em diferentes arquiteturas neurais, com características e aplicações bem distintas [22]. A arquitetura afeta o desempenho da rede, assim como as aplicações para as quais ela é desejada e sua estrutura está intimamente ligada ao algoritmo de aprendizado utilizado para o treinamento. Algumas redes permitem que as conexões caminhem tanto no sentido entrada-saída quanto saída-entrada. Outras permitem que os neurônios da mesma camada estejam conectados. Ainda há as que permitem que o neurônio envie sinais de volta para ele mesmo.

Dentre as arquiteturas, pode-se citar a *Multilayer Perceptron*, que é formada de uma camada de entrada, uma ou mais intermediárias e uma camada de saída, este é o tipo de arquitetura mais encontrado. Há também a rede de Kohonen, que é formada por uma camada de entrada e uma única camada de saída, onde cada neurônio está conectado a todos os seus vizinhos. Na rede de Hopfield, não há neurônios de entrada ou saída. A entrada de um neurônio é a saída dos demais neurônios e sua saída vai ser a entrada dos demais neurônios da rede. A rede ART é formada por camada de entrada e saída, além de controladores de fluxo de sinais. Para este estudo foi escolhida a rede ARTMAP, que veremos com cuidado mais adiante.

Uma forma de classificar as arquiteturas das redes neurais é como recorrente e não-recorrente. Apresentaremos esses dois tipos de arquiteturas:

- **Redes Neurais Artificiais Não-Recorrentes:** Redes não-recorrentes, também conhecidas como redes *feedforward*, são redes que não possuem laços de realimentação de suas saídas para suas entradas. Sua estrutura pode ser formada por uma camada única ou por múltiplas camadas, também conhecidas como redes multicamadas. As redes neurais multicamadas não-recorrentes contém um conjunto de entradas, uma ou mais camadas escondidas e uma camada de saída. A camada de saída contém os neurônios que fornecem a saída da rede. As camadas que não possuem ligações diretas com a entrada nem com a saída são denominadas camadas ocultas. No caso de redes não-recorrentes como ilustra a Figura A.6, não existem conexões ligando um neurônio de uma camada a outro de uma camada anterior, nem a um neurônio da mesma camada, as conexões são feitas somente para neurônios da camada seguinte.

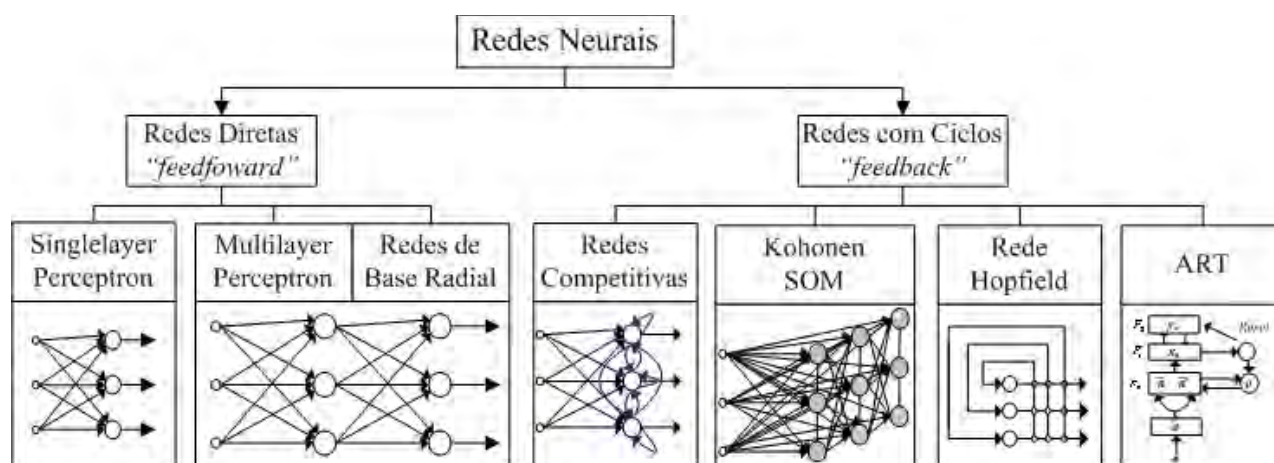


Figura A.6: Arquitetura de Redes Neurais Artificiais.

- **Redes Neurais Artificiais Recorrentes:** Redes neurais recorrentes são redes que contém laços de realimentação das saídas para as entradas, sendo suas saídas determinadas pelas entradas atuais e pelas saídas anteriores. As redes recorrentes, quando organizadas em camadas, possuem interligações entre neurônios da mesma camada e entre camadas não consecutivas, gerando interconexões bem mais complexas que as redes neurais não-recorrentes [29].

A.6 Algoritmo de Aprendizagem

A propriedade mais interessante das redes neurais, herdada do sistema nervoso humano, é sua habilidade de aprender com o ambiente e melhorar seu desempenho através deste aprendizado. As redes são capazes de adquirir conhecimento a partir de experiências com o ambiente que as envolve na tentativa de solucionar problemas. A partir destas experiências, as sinapses são alteradas de modo a adaptar-se a inúmeras situações. As sucessivas alterações das sinapses e consequentemente a adaptação das redes neurais caracterizam a aprendizagem tornando capazes de generalização e reprodução de raciocínio aproximado [5, 24].

Geralmente uma rede produz resultados cada vez melhores à medida que existe uma maior quantidade de informações (dados) inseridos e disponíveis em seu processo de aprendizado. O conjunto de regras e procedimentos numéricos adotados para a obtenção de estabilidade nos parâmetros da rede neural, através de um ajuste contínuo dos pesos das conexões que definem os procedimentos para a solução de um problema de aprendizado, é chamado de algoritmo de treinamento. Os algoritmos de treinamento mais empregados são os seguintes:

- **Supervisionado:** Este método consiste na apresentação de um conjunto de padrões para a rede neural, cada qual formado por um par entrada/saída. Para cada exemplo, a rede neural recebe a entrada e deve aprender a gerar a saída correspondente. No caso da rede não gerar a saída desejada, é encontrada a diferença entre a saída obtida e a saída desejada (erro). A partir desta diferença, os pesos sinápticos dos neurônios são atualizados para que, nas próximas iterações (ou seja, próximos instantes em que um par de entrada/saída for apresentado à rede) a saída obtida se aproxime ao máximo da saída desejada. Portanto, não há a necessidade de que a rede neural descubra quais saídas devem ser apresentadas como resposta às entradas. As saídas corretas são estabelecidas pelos padrões de treinamento. Um exemplo dessa arquitetura é o *Backpropagation*.

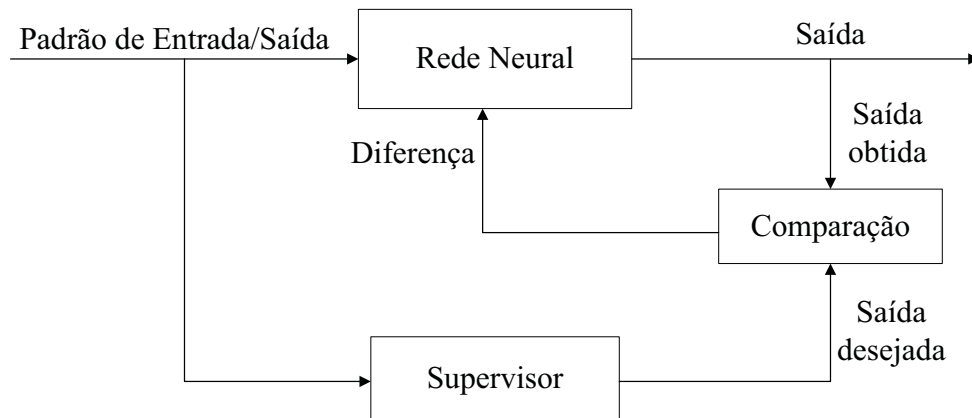


Figura A.7: Aprendizado Supervisionado.

- **Não-Supervisionado:** Neste tipo de método, a aprendizagem se processa sem que as saídas corretas sejam fornecidas. Ao contrário do método supervisionado, somente o padrão de entrada é fornecido a rede. Este método capacita as redes neurais extrair correlações presentes nos dados de entrada, ou seja, permite a formação de representações internas para codificar características da entrada e criar novas classes automaticamente. Após a aprendizagem, a rede neural torna-se capaz de classificar suas entradas de acordo com as relações aprendidas. A Figura A.8 ilustra o processamento neural do aprendizado não-supervisionado.

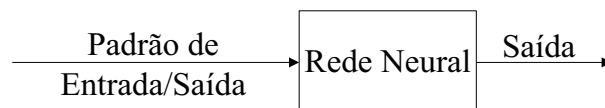


Figura A.8: Aprendizado Não-Supervisionado.

- **Aprendizado por Reforço:** Neste tipo de aprendizado, um agente é aquele que, a partir da interação com o ambiente em que atua, aprende de maneira autônoma uma política ótima de atuação, por experimentação direta, sem ser ensinado por meio de exemplos fornecidos por um supervisor, como ocorre com o aprendizado supervisionado. Um agente itera com o ambiente em intervalos de tempo discreto em um ciclo de percepção (Figura A.9) e observa (a cada intervalo do ciclo) o estado atual s e escolhe a ação a para realizar, alterando o estado atual do ambiente. Ao executar a ação, o agente recebe um sinal de reforço escalar $r_{s,a}$ indicando o quão desejável é o estado resultante. Tal sinal de reforço corresponde a uma penalização ou recompensa como consequência da ação realizada pelo agente. O comportamento deve-se restringir a escolha de ações que tendem a aumentar, ao longo do tempo, a soma dos valores do sinal de reforço. Desta forma, o agente torna a

rede apta a realizar uma determinada tarefa ao longo das iterações através da sistemática tentativa-e-erro, a qual pode ser realizada por uma grande variedade de algoritmos.

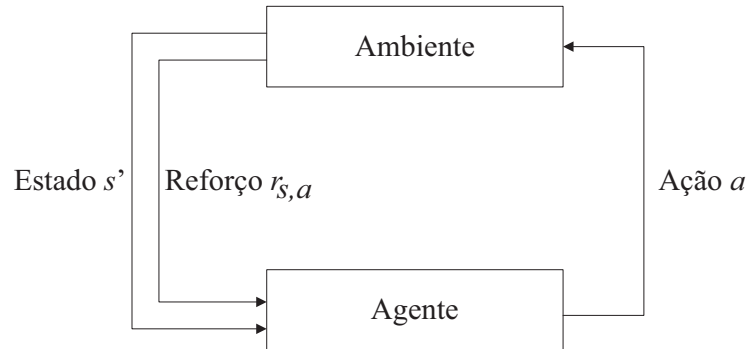


Figura A.9: Aprendizado por Reforço.

- **Híbrido:** Combina treinamento supervisionado e treinamento não-supervisionado. Parte de seus pesos são ajustados através do treinamento supervisionado enquanto outros são obtidos através de treinamentos não-supervisionados.

A seguir listam-se as topologias neurais mais utilizadas e conhecidas, conforme levantamento disponível em [53], subdivididas de acordo com o paradigma de aprendizado adotado, o que demonstra a imensa diversidade de topologias existentes que realizam processamento neural:

1. *Algoritmo Supervisionado:*

- 1.1. ***Feedforward:*** *Linear*, *Hebbiano* (Hebb, 1949; Fausett, 1994), *Perceptron* (Rosenblatt, 1958, Minsky & Papert, 1969; Minsky & Papert, 1988; Fausett, 1994); *Adaline* (Widrow & Hoff, 1960; Fausett, 1994), *Higher Order* (Bishop, 1995), *Functional Link* (Pao, 1989), *MLP - Multilayer Perceptron* (Bishop, 1995; Reed & Marks, 1999; Fausett, 1994), *Back-propagation* (Rumelhart, Hinton & Williams, 1986), *Cascade Correlation* (Fahlman & Lebiere, 1990; Fausett, 1994), *Quickprop* (Fahlman, 1989), *RPROP* (Riedmiller & Braun, 1993), *Redes RBF* (Bishop, 1995, Moody & Darken, 1989; Orr, 1996); *OLS - Orthogonal Least Squares* (Chen, Cowan & Grant, 1991), *CMAC - Cerebellar Model Articulation Controller* (Albus, 1975; Brown & Harris, 1994);
- 1.2. ***Somente Classificadoras:*** *LVQ - Learning Vector Quantization* (Kohonen, 1988; Fausett, 1994), *PNN - Probabilistic Neural Network* (Specht, 1990; Masters, 1993, Hand, 1982; Fausett, 1994);

- 1.3. **Somente Regressão:** *GRNN - General Regression Neural Network* (Specht, 1991, Nadaraya, 1964, Watson, 1964), *Feedback* (Hertz, Krogh & Palmer, 1991; Medsker & Jain, 2000), *BAM - Bidirectional Associative Memory* (Kosko, 1992; Fausett, 1994), *Boltzman Machine* (Arkley et al., 1985; Fausett, 1994);
- 1.4. **Séries de Tempo Recorrentes:** *Backpropagation through time* (Werbos, 1990), *Elman* (Elman, 1990), *FIR - Finite Impulse Response* (Wan, 1990), *Jordan* (Jordan, 1986), *Real-time recurrent network* (Williams & Zipser, 1989), *Recurrent backpropagation* (Pineda, 1989; Fausett, 1994), *TDNN - Time Delay NN* (Lang, Waibel & Hinton, 1990);
- 1.5. **Competitivas:** *ARTMAP* (Carpenter, Grossberg & Reynolds, 1991), *Fuzzy ARTMAP* (Carpenter et al., 1992; Kasuba, 1993), *Gaussian ARTMAP* (Williams, 1995), *Counterpropagation* (Hecht-Nielsen, 1987, Hecht-Nielsen, 1988; Hecht-Nielsen, 1990, Fausett, 1994); *Neocognitron* (Fukushima, Miyake & Ito, 1982; Fukushima, 1988, Fausett, 1994).

2. Algoritmo Não-Supervisionado:

- 2.1. **Competitivo:** *Vector Quantization* (Grossberg, 1976; Kohonen, 1984), *Conscience* (Desieno, 1988), *Self-Organizing Maps* (Kohonen, 1995; Fausett, 1994), *Local Linear* (Mulier & Cherkassky, 1995);
- 2.2. **Adaptive Resonance Theory:** *ART-1* (Carpenter & Grossberg, 1987a; Moore, 1988; Fausett, 1994), *ART-2* (Carpenter & Grossberg, 1987b; Fausett, 1994); *ART-2A* (Carpenter, Grossberg & Rosen, 1991a); *ART-3* (Carpenter & Grossberg, 1990), *Fuzzy ART* (Carpenter, Grossberg & Rosen, 1991b), *DCL - Differential Competitive Learning* (Kosko, 1992);
- 2.3. **Dimension Reduction:** *Dimension Reduction* (Diamantaras & Kung, 1996), *Hebbiano* (Hebb, 1949; Fausett, 1994); *Oja* (Oja, 1989), *Sanger* (Sanger, 1989), *Hebbiano Differential* (Kosko, 1992);
- 2.4. **Autoassociação:** *Linear autoassociator* (Anderson et al., 1977; Fausett, 1994), *BSB - Brain State in a Box* (Anderson et al., 1977; Fausett, 1994), *Hopfield* (Hopfield, 1982; Fausett, 1994).

A.7 Aplicações de Redes Neurais

Apesar da grande frustração nas décadas de 60 e 80 devido a incapacidade das topologias empregadas à época (um único neurônio ou vários neurônios dispostos em uma única camada) em realizar classificação de padrões não-linearmente separáveis, o sucesso da proposição de uma arquitetura multicamada com ajuste dos pesos sinápticos, via retropropagação do erro obtido, deram um enorme impulso à aplicabilidade das redes neurais artificiais [24]. Especificamente no caso de arquiteturas multicamadas, estudos comprovam que duas camadas intermediárias são suficientes para aproximar qualquer função matemática [12], o que indica serem estes modelos classificadores universais de funções.

Contudo existem várias arquiteturas neurais propostas, e cada uma delas gera resultados melhores para determinados tipos de problemas, o que faz a experimentação destes modelos, em relação ao desempenho de certas tarefas, um procedimento necessário e interessante na medida que são identificados os melhores usos para cada uma das arquiteturas existentes. Esta plasticidade operacional torna os modelos de redes neurais passíveis de realizarem as mais diversas aplicações, o que as torna amplamente difundidas em diversas áreas do conhecimento, desde aquelas ligadas à pesquisa científica e tecnológica quanto aos negócios. Sendo assim, o campo de atuação de um pesquisador através do estudo de redes neurais acaba se tornando multidisciplinar [15], em virtude de que as aplicações em que são empregadas são de cunho geral e englobam a conceituação simultânea de diversas áreas do conhecimento [5]. Assim, de acordo com Kohonen [34], a grande contribuição dos modelos neurais se dá nas áreas de representação do conhecimento e reconhecimento de padrões.

A opção por modelos de rede artificiais na solução de tarefas diversas se dá geralmente por conta da complexidade inerente à mesma; sistemas excessivamente complexos e/ou compostos por conjuntos de dados vastos, complexos e inter-relacionados são mais satisfatoriamente resolvidos com o uso desta metodologia [5, 23]. A razão principal repousa no fato de que, quanto maior a quantidade de informações disponíveis, maior a dificuldade na modelagem do sistema representado por esta informação [5].

Com o objetivo de ilustrar a diversidade de aplicações nas quais redes neurais artificiais têm sido utilizadas, abaixo listam-se alguns dos usos em que modelos de redes neurais artificiais são largamente empregados, de acordo com as áreas de concentração nas quais foram apresentados estudos durante o IEEE International Conference On Neural Networks, realizada em 1994 em Washington, nos Estados Unidos.

- Análise de movimento [45];
- Aprendizado e reconhecimento de padrões [4];
- Ciência cognitiva [37];
- Controladores inteligentes [65];
- Economia e negócios [64];
- Estimação e identificação [30];
- Inteligência Computacional [61];
- Neurobiologia [26];
- Previsão de séries de tempo [14];
- Processamento e transmissão de sinais [58];
- Sistema de energia [63];
- Sistemas híbridos [51].

Todos estes usos possíveis para modelos de redes neurais se devem a sua adaptabilidade às tarefas mais distintas e às diversas arquiteturas propostas, que empiricamente são testadas e têm seus resultados comparados entre si e entre outros modelos utilizáveis para a mesma finalidade. Este é um trabalho enorme e que ainda tem muito a retribuir à medida em que mais e mais sistemas complexos são estudados sob o ponto de vista do processamento neural.

Para a realização deste trabalho, como em qualquer um realizado com redes neurais, em primeiro lugar deve-se decidir sobre o paradigma de aprendizagem a ser empregado e, em seguida, definir a topologia neural. No caso da escolha recair sobre um paradigma de aprendizado não-supervisionado, em vista principalmente da falta de conhecimento padronizado a cerca do sistema a ser estudado, existe uma classe de redes neurais bastante semelhantes às estruturas neurobiológicas conhecidas que apresentam resultados expressivos no desempenho de tarefas cujo conteúdo de informação está exclusivamente contido nos dados de entrada do sistema. Estas classes neurais possuem como característica mais marcante a capacidade de auto-organização de seus neurônios, tanto no aspecto de classificação neural como no aspecto visual da representação destas estruturas, e são conhecidas como redes neurais auto-organizadas.

Lógica Fuzzy

Em 1964 o professor Lofti A. Zadeh, Departamento de Engenharia Elétrica e Ciências da Computação da Universidade da Califórnia-Berkeley, apresentou a Teoria dos Conjuntos *Fuzzy* para representar os problemas de classificação de conjuntos com os quais trabalhava, pois a representação clássica não satisfazia tais conjuntos porque não possuíam fronteiras bem definidas, *i.e.*, a transição entre os conjuntos é suave e não abrupta. Como em diversas situações, a relação de pertinência no problema trabalhado por ele não era bem definida e, portanto, não se podia dizer se o elemento pertencia ou não ao conjunto dado. O que Zadeh propôs foi flexibilizar a pertinência de elementos aos conjuntos criando a ideia de grau de pertinência. Dessa forma, um elemento poderia pertencer parcialmente a um conjunto dado. Esta sua idéia foi publicada em 1965 [70], sendo este o marco de nascimento da *Teoria dos Conjuntos Fuzzy*.

O termo *fuzzy* significa incerto, impreciso, subjetivo, nebuloso, difuso, e se refere ao fato de muitas vezes não se conhecer completamente o sistema que está sendo analisado. Assim, Zadeh propôs considerar uma função de pertinência que fornecesse o **grau de pertinência**.

B.1 Conjuntos Fuzzy

Antes de apresentar as características dos conjuntos *Fuzzy*, mostram-se algumas operações e características dos conjuntos clássicos.

A teoria de conjuntos clássicos está baseada na função característica clássica, dada por:

$$\mu_A(x) = \begin{cases} 1, & x \in A; \\ 0, & x \notin A, \end{cases}, \quad x \in X,$$

em que X é o conjunto de dados, $A \subset X$. Assim a função μ_A realiza um mapeamento do conjunto universo no conjunto $\{0, 1\}$, portanto ela discrimina entre os elementos de X quais são os que pertencem a A e quais os que não pertencem a A , dividindo o conjunto X em duas partes com fronteira bem definida, neste caso, os conjuntos são ditos “crisp”. As operações básicas dos conjuntos clássicos são:

- a união:

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)); \quad (\text{B.1.1})$$

- a interseção:

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)); \quad (\text{B.1.2})$$

- o complemento:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x). \quad (\text{B.1.3})$$

Mas, quer-se trabalhar com conjuntos cujas fronteiras não estão bem definidas e, desta forma, não se pode utilizar a teoria dos conjuntos clássicos para lidar com este tipo de problema, tendo visto que as transições de uma classe para outra acontecem de forma suave. Assim, para se obter os conjuntos *fuzzy* e suas operações, generaliza-se a função característica da lógica clássica para o intervalo $[0, 1]$, ou seja:

$$\mu_A : U \rightarrow [0, 1],$$

o que implica que se consideram valores contínuos de pertinência e não mais apenas: *pertence* ou *não pertence*. Assim, o elemento x pertencerá ao subconjunto A com grau de pertinência que é um valor do intervalo $[0, 1]$.

Definição B.1. Um conjunto *fuzzy* A em um conjunto X é um conjunto de pares ordenados de um elemento genérico $x \in X$ e seu grau de pertinência $\mu_A(x)$ da forma:

$$A = \{(x, \mu_A(x)) | x \in X\}.$$

Deste modo, um conjunto *fuzzy* é caracterizado por uma função pertinência e o grau de pertinência pode ser caracterizado como uma medida que expressa a possibilidade de que um dado elemento seja membro de um conjunto *fuzzy*.

Quanto à representação de um conjunto *fuzzy* tem-se que se ele for discreto pode-se simplesmente enumerar seus elementos juntamente com o grau de pertinência na forma:

$$A = \sum_i \mu_A(x_i) / x_i,$$

em que o somatório se refere a operação união (disjunção) e a notação $\mu_A(x_i) / x_i$ se refere ao elemento x_i que pertence ao conjunto *fuzzy* A com grau de pertinência $\mu_A(x_i)$. Por simplicidade, apenas lista-se no conjunto A aqueles elementos que têm grau de pertinência diferente de zero.

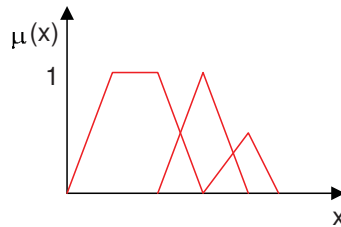


Figura B.1: Exemplo de Função de Pertinência do Tipo Linear por Partes.

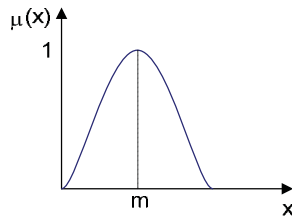


Figura B.2: Exemplo de Função de Pertinência do Tipo Gaussiana.

Entretanto, quando os conjuntos *fuzzy* são contínuos sua representação é a própria função de pertinência. As formas para as funções de pertinência são totalmente arbitrárias, todavia, as mais utilizadas são:

- linear por partes (triangular ou trapezoidal), Figura B.1;
- quadrática;

- gaussiana, Figura B.2.

Devido a própria simplicidade, as funções lineares por partes são as mais populares, isto além do fato do custo computacional exigido por outros tipos de funções não refletem, em geral, uma melhoria significativa na qualidade dos valores de saída dos sistema [69].

As três operações básicas definidas através das equações (B.1.1), (B.1.2) e (B.1.3) são idênticas para a teoria de conjuntos *fuzzy*. A diferença está apenas na utilização da função característica. Como na lógica clássica, os operadores de intersecção e união correspondem respectivamente aos operadores lógicos de conjunção (**E**) e disjunção (**OU**).

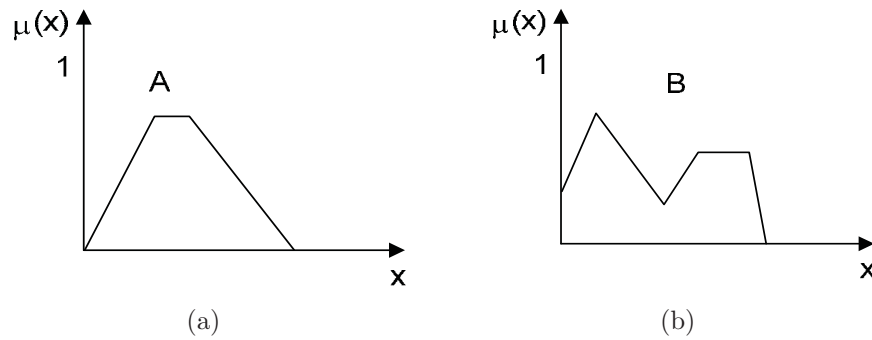


Figura B.3: Conjuntos *Fuzzy*: A e B .

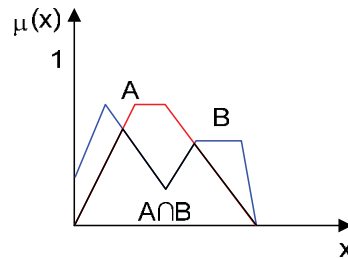


Figura B.4: Intersecção entre Conjuntos *Fuzzy*.

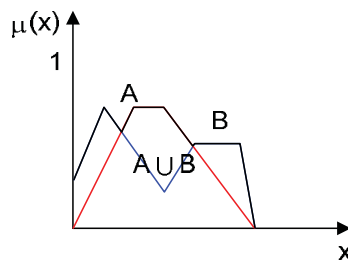


Figura B.5: União entre Conjuntos *Fuzzy*.

Existem muitas formalizações matemáticas para esses operadores e a relação dual entre eles se mantém, ou seja, a escolha de um operador conjunção "define" qual será o operador disjunção e vice-versa. O par de operadores mais amplamente utilizados nas técnicas *fuzzy* são o operador

min (mínimo) para a conjunção e o operador **max** (máximo) para a disjunção *fuzzy* (Figuras B.3, B.4, B.5 e B.6).

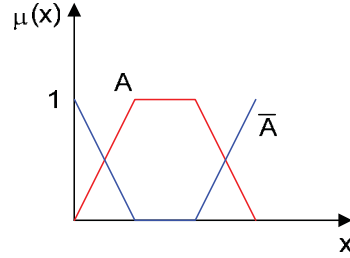


Figura B.6: Exemplo do Complemento de um Conjunto *Fuzzy*.

Uma das consequências mais interessantes da definição de conjuntos *fuzzy*, em contraste com os conjuntos clássicos, é a Lei do Meio Excluído e da Lei da Contradição. Enquanto que nos conjuntos clássicos tem-se:

$$A \cup \bar{A} = X \qquad A \cap \bar{A} = \emptyset,$$

pode-se ver, através das Figuras B.7 e fig:Contra, que:

$$A \cup \bar{A} \neq X \qquad A \cap \bar{A} \neq \emptyset.$$

Isso ocorre devido a flexibilidade da função característica, pois existem incertezas não estatísticas e imprecisões no processo.

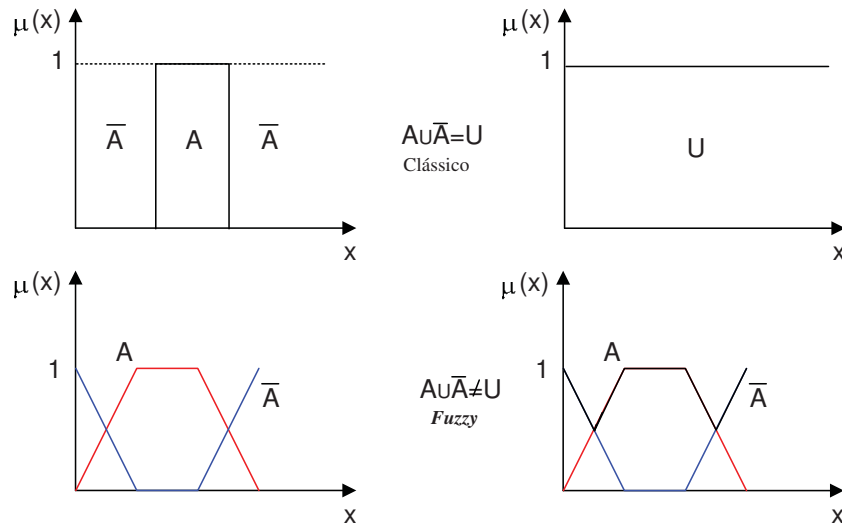


Figura B.7: Lei do Meio Excluído.

Outra consequência da definição está em contraste com a teoria das probabilidades, visto que os valores de pertinência de um mesmo elemento a nos diversos conjuntos *fuzzy* não precisam somar 1.

Existem dois outros conceitos importantes na teoria de conjuntos *fuzzy*:

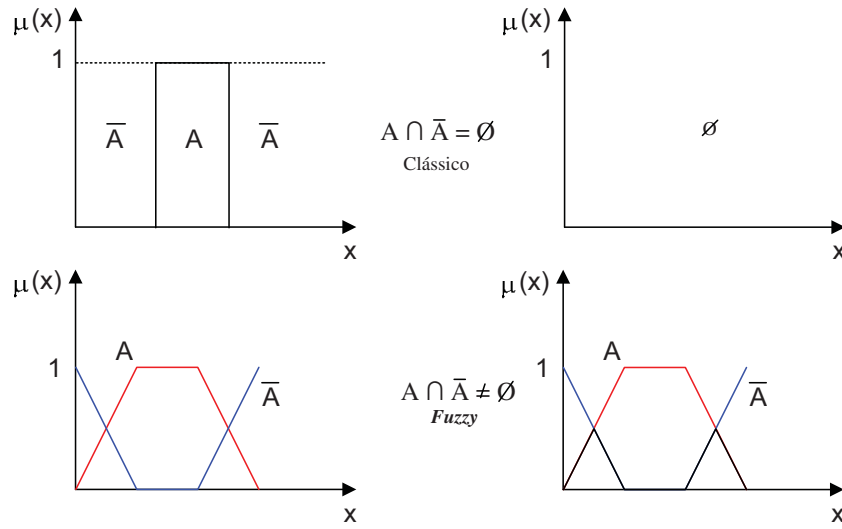


Figura B.8: Lei da Contradição.

Definição B.2. Suponha que $S(A)$ seja o conjunto clássico de todos os elementos $x \in X$ cuja função de pertinência tenha valor diferente de zero. Então, $S(A)$ será denominado o suporte do conjunto fuzzy A :

$$S(A) = \text{supp}(A) = \{x \in X | \mu_A(x) > 0\} = \text{supp}(A).$$

Definição B.3. O conjunto elementos que pertençam a um conjunto fuzzy A , pelo menos com grau α , será chamado conjunto α -nível ou conjunto alpha-corte:

$$A_\alpha = \{x \in X | \mu_A(x) > \alpha\}.$$

A importância desta definição está no fato de fornecer outra maneira de considerar um conjunto *fuzzy*. Uma vez que a função de pertinência determina completamente um conjunto *fuzzy*, e seus valores pertencem ao intervalo $[0, 1]$, então um conjunto *fuzzy* pode ser descrito pela união de todos os conjunto α -níveis:

$$A = \bigcup_{\alpha} A_\alpha, \quad \alpha \in [0, 1].$$

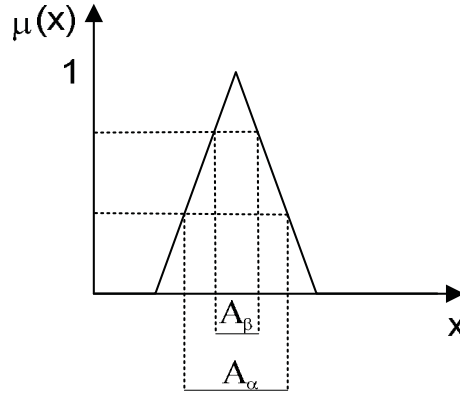


Figura B.9: Exemplo de α -Nível.

A importância dos conjuntos α -níveis vem do fato deles serem conjuntos clássicos e, sendo assim, muito do formalismo matemático da teoria de conjuntos *fuzzy* pode ser desenvolvido no espaço dos conjuntos clássicos, aproveitando inclusive resultados já desenvolvidos na teoria clássica (teoremas, lemas, etc.) [49]. Ainda deve-se ressaltar a importância dos conceitos de cardinalidade e de altura desses conjuntos. A cardinalidade de um conjunto é o número total de elementos no conjunto. Uma vez que os elementos podem pertencer parcialmente a um conjunto *fuzzy*, uma generalização natural da noção clássica de cardinalidade consiste em pesar cada elemento através do grau de pertinência. Sendo assim, define-se:

Definição B.4. A cardinalidade de um conjunto *fuzzy* é definida por:

$$Card(A) = \sum_{x_i} \mu_A(x_i), \quad (\text{B.1.4})$$

em que: A é um conjunto *fuzzy* e os x_i são elementos do conjunto universo X .

O conceito de cardinalidade torna-se importante quando se está interessados no tamanho dos conjuntos, como acontece em sistemas de informação e análise de banco de dados *fuzzy*, e quando se precisa definir fatores de normalização para os conjuntos *fuzzy*, como acontece em alguns processos de defuzzificação [69].

Definição B.5. A altura de um conjunto *fuzzy* é o maior valor de pertinência da sua função de pertinência:

$$hgt(A) = \max_{x_i} \mu_A(x_i). \quad (\text{B.1.5})$$

Os conjuntos fuzzy com altura igual a 1 são chamados normais e aqueles cuja altura é inferior a 1 são chamados de não-normais.

B.2 Estrutura do Sistema Lógico Fuzzy

A estrutura de todo sistema lógico *fuzzy* está baseada em três operações:

Fuzzificação: que transforma os dados iniciais em variáveis linguísticas;

Inferência *Fuzzy*: cujo objetivo é relacionar, através de regras preestabelecidas, as possíveis variáveis entre si obtendo resultados linguísticos;

Defuzzificação: que transforma os resultados linguísticos em dados finais.

A Figura B.10 mostra como isto acontece.

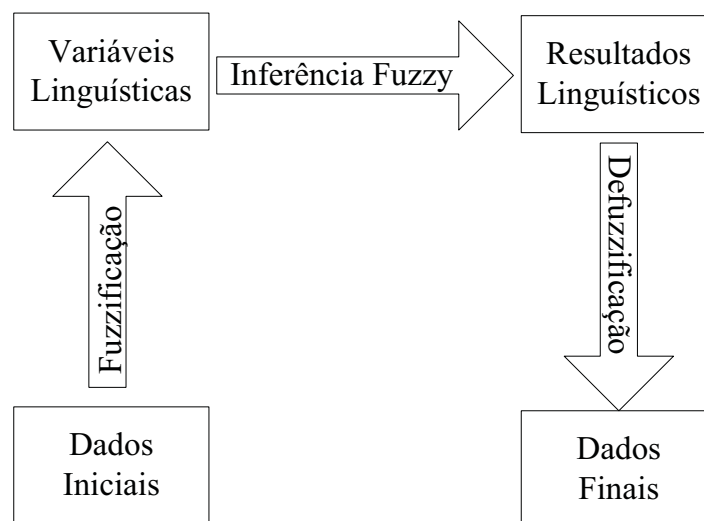


Figura B.10: Sistema Lógico *Fuzzy*.

A seguir são apresentadas as etapas e uma descrição das variáveis citadas acima.

B.3 Variáveis Linguística

Começa-se esta seção definindo variável linguística: Uma variável linguística *fuzzy* é uma variável cujo valor é expresso qualitativamente por um termo linguístico (que fornece um conceito à variável) e quantitativamente por uma função de pertinência.

Definição B.6. A tripla (V, X, T) define uma variável linguística, em que:

- V é o nome da variável linguística, por exemplo: febre, temperatura, pressão, etc.;
- X é o domínio (Universo) de valores de V sobre o qual o significado do termo linguístico é determinado, por exemplo: a febre pode estar entre 35 e 40°;
- T é o conjunto de termos linguísticos de V , por exemplo: elevado, baixo, pouco, extenso, etc..

A Figura B.11 mostra um exemplo de variável linguística.

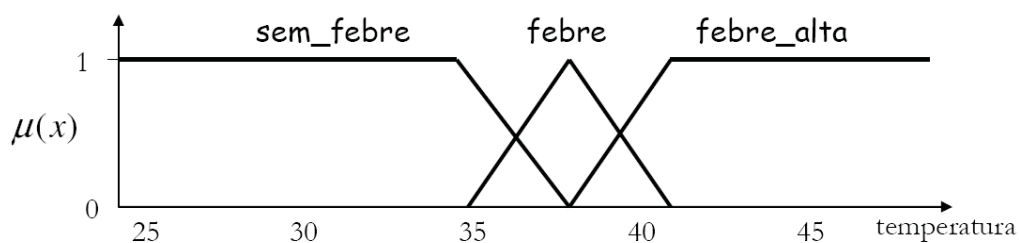


Figura B.11: Exemplo de Variável Linguística.

Note que:

- V é a temperatura;
- X é o conjunto de temperaturas válidas;
- T assume os valores: sem febre, febre, febre alta.

Lembre-se que na formulação clássica, os termos linguísticos atribuídos para Febre são: *presente* e *ausente*.

É importante notar que a variável linguística é expressa em termos de uma variável básica, que denota a sua medida. No exemplo acima Febre é medida em graus centígrados (temperatura). Esta medida pode ser quantitativa, no caso em que é possível o uso de aparelhos de medida, ou pode ser expressa de forma qualitativa, como no caso do uso de cruzes em exames clínicos.

Os termos linguísticos são usados para expressar conceitos e conhecimentos na comunicação humana, e em muitas áreas eles são a forma mais importante (quando não a única) de quantificar os dados/informações. O uso de termos linguísticos é frequente no nosso cotidiano, diz-se que "o dia está muito quente", "A água está meio fria", "Ele é muito alto e magro", etc.. Todos estes termos possuem um significado e transmitem informação.

As variáveis linguísticas são expressas dentro de um certo domínio de valores. Em geral é o especialista que define esse domínio e realiza sua partição *fuzzy*. As variáveis numéricas expressam o seu valor e utilidade e são amplamente empregada nas ciências exatas (engenharia, física, matemática, etc). Porém, as variáveis simbólicas têm conquistado cada vez maior importância devido ao desenvolvimento das áreas de inteligência artificial e processos de decisão. A capacidade de combinar variáveis simbólicas (linguísticas) e numéricas é uma das principais razões do sucesso das aplicações da lógica *fuzzy* em sistemas inteligentes, tanto na engenharia quanto em muitas outras áreas que lidam com domínios contínuos.

B.4 Inferência Fuzzy

Antes de tratarmos de inferência *fuzzy*, vejamos o que são as *Regras Fuzzy*:

B.4.1 Regras Fuzzy

As regras *fuzzy* são estruturas vastamente utilizadas em várias abordagens da teoria *fuzzy*. Elas podem ser entendidas de diversas maneiras. Conceitualmente, as regras *fuzzy* descrevem situações específicas que podem ser submetidas a análise de um painel de especialistas, cuja inferência nos conduz a algum resultado desejado. A inferência baseada em regras *fuzzy* pode também ser compreendida como um funcional que mapeia um conjunto de entradas do sistema para um conjunto de saídas (como em um esquema de interpolação).

A regra *fuzzy* é uma unidade capaz de capturar algum conhecimento específico, e um conjunto de regras é capaz de descrever um sistema em suas várias possibilidades. Cada regra *fuzzy*, da mesma forma que uma afirmação clássica, é composta por uma parte antecedente (a parte **Se**) e uma parte consequente (a parte **Então**), resultando em uma estrutura do tipo:

Se antecedentes **então** consequentes.

Os antecedentes descrevem uma condição enquanto a parte consequente descreve uma conclusão ou uma ação que pode ser esboçada quando as condições se verificam. A diferença entre os antecedentes de uma regra *fuzzy* e uma regra clássica é que os primeiros descrevem uma condição elástica, ou seja, uma condição que pode ser parcialmente satisfeita, enquanto os últimos descrevem uma condição rígida, ou seja, a regra não funcionará se os antecedentes não forem completamente satisfeitos.

Os antecedentes definem uma região *fuzzy* no espaço das variáveis de entrada do sistema e os consequentes, por sua vez descrevem uma região no espaço das variáveis de saída do sistema, qual seja a sua conclusão/ação. Sendo assim, a construção dos antecedentes muitas vezes resulta em um trabalho de classificação, enquanto a elaboração dos consequentes exige um conhecimento, ainda que empírico, sobre a dinâmica do sistema. Pode-se esperar então que a elaboração dos consequentes de uma regra seja mais complexa do que a dos antecedentes.

Uma vez construído o conjunto de regras *fuzzy*, necessita-se de uma “máquina de inferência” para extrair dela a resposta final. Existem vários métodos de inferência possíveis e a escolha por um deles depende do sistema que está sendo analisado. No entanto, a inferência mais comum, e amplamente utilizada no controle de sistemas, é o *Método de Mamdani*.

As regras são processadas em paralelo, ou seja, todas as regras (circunstâncias) são consideradas ao mesmo tempo, e ao final obtém-se uma resposta que pode ser tanto um valor numérico clássico, quanto um conjunto *fuzzy* ou um funcional, a depender do tipo de consequente utilizado. Às vezes é necessário que a saída do sistema seja um número, o que é muito comum em controladores *fuzzy*, pois o sistema precisa ser realimentado. Nestes casos, se a saída do sistema for um conjunto *fuzzy*, então se faz necessário um processo de defuzzificação para se obter um número apropriado.

B.4.2 Inferência Fuzzy

Nos modelos linguísticos as equações diferenciais, que normalmente caracterizam os sistemas dinâmicos, são substituídas por um conjunto de regras cuja construção abrange predicados vagos e, muitas vezes, conhecimentos heurísticos. Estes modelos são estruturados sobre uma base de conhecimentos e seu poder de decisão depende da elaboração das regras e dos mecanismos de raciocínio *fuzzy*.

Na teoria de sistemas, os modelos são classificados de acordo com algumas características que o conjunto de equações diferenciais, que é quem estabelece as relações entre as entradas e saídas do sistema, apresentam; como por exemplo, linearidade, não-linearidade, dinâmica, estática, etc. Nos modelos *fuzzy* quem assume este papel são as famílias de regras. Assim, os diferentes modelos resultam de distintas interpretações dos conhecimentos contidos nas regras e dos mecanismos de raciocínio adotados.

Existem diversos tipos de inferência para relacionar as entradas e saídas do conjunto de regras. Eles diferem pelo tipo de operador utilizado e pelos tipos de antecedentes e consequentes. As

regras podem, por exemplo, ser constituídas por uma única entrada e um única saída. Esse é o caso dos sistema "Single-Input/Single-Output" (SISO). Por outro lado, o sistema pode ser caracterizado por regras com múltiplos valores de entrada e múltiplos valores de saída, o que consiste em uma generalização do caso anterior (MIMO). Os modelos linguísticos *fuzzy* para sistemas dinâmicos são também muito interessantes e amplamente aplicados. Nesses modelos a dinâmica do sistema é obtida através de um conjunto de regras iterativas, onde a evolução temporal é obtida pela iteração da saída do passo de tempo anterior como entrada do passo posterior.

O método de inferência determina a forma operacional do modelo linguístico. Ele é um mapeamento que define uma transformação do valor *fuzzy* de entrada em um valor de saída. Existem diversos métodos de inferência. Deve-se, portanto, escolher aquele que melhor se adapta ao sistema que estamos modelando. Um dos métodos mais divulgados, pela sua simplicidade e por se adaptar muito bem aos controladores *fuzzy*, é o *Método de Mamdani* [49].

Os modelos linguísticos *fuzzy* fornecem uma saída para o sistema que é também um conjunto *fuzzy*. Todavia, muitas vezes está-se interessado em obter um número como resposta, ou seja, desejamos obter um resultado clássico a partir da saída *fuzzy* obtida. Este objetivo pode ser alcançado através dos métodos de defuzzificação.

B.5 Métodos de Defuzzificação

A defuzzificação é o procedimento que permite interpretar a distribuição de possibilidades da saída de um modelo linguístico *fuzzy* de forma quantitativa, ou seja, ela nos fornece um valor numérico representativo que captura o significado essencial dessa distribuição de possibilidades. Existem muitas técnicas de defuzzificação e entre as mais utilizadas estão:

- Média dos Máximos;
- Centro de Área;
- Método das Alturas.

B.5.1 Média dos Máximos

O método de defuzzificação da Média dos Máximos (MM) calcula a média de todos os valores de saída que tenham os maiores graus de possibilidade. Suponha-se que " y é A " seja uma

conclusão *fuzzy* que deva ser defuzzificada. O método de defuzzificação MM pode ser expresso da seguinte forma [32, 69]:

$$y_0 = \frac{\sum_{y^* \in P} y^*}{|P|}, \quad (\text{B.5.1})$$

em que P é o conjunto de todos os valores de saída com máximos grau de possibilidade em A , ou seja:

$$P = \left\{ y^* | \mu_A(y^*) = \sup_y \mu_A(y) \right\}, \quad (\text{B.5.2})$$

e $|P|$ é a cardinalidade do conjunto P , ou seja, o número de elementos de P . A Figura B.12 representa um exemplo desse método. Note que, se P for um intervalo, então a técnica MM de defuzzificação fornece o ponto médio desse intervalo.

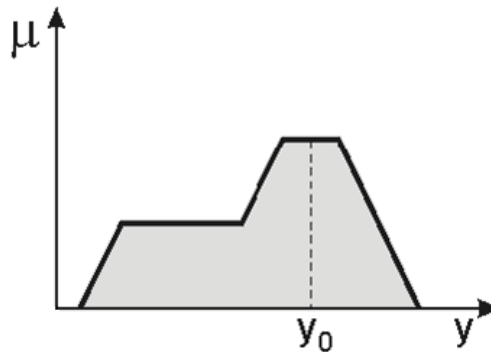


Figura B.12: Exemplo do Método de Defuzzificação MM.

A principal limitação do método de defuzzificação MM é que não considera a forma total do conjunto *fuzzy* de saída. Sendo assim, duas distribuições de possibilidades que apresentem formas diferentes com o mesmo conjunto de valores de grau de pertinência máximo quando defuzzificados com esta técnica proporcionarão o mesmo valor clássico, o que vai contra nossa intuição. A Figura B.13 mostra um exemplo onde isso ocorre.

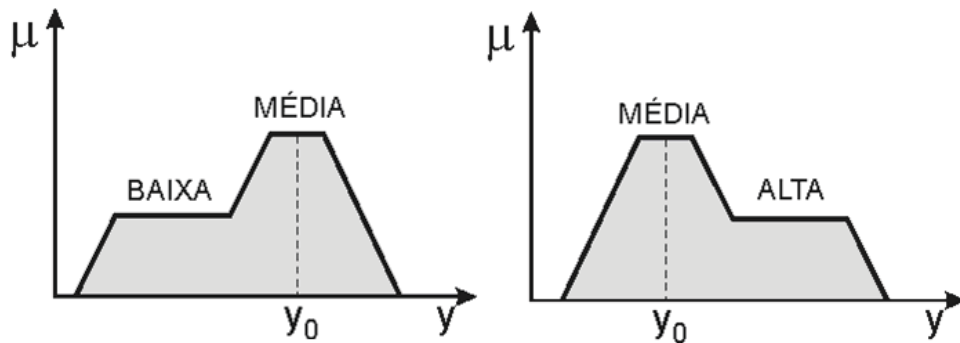


Figura B.13: Exemplo de Problemas que o Método de Defuzzificação MM pode Causar.

B.5.2 Centro de Área

O Método do Centro de Área (CA), também conhecido como Método do Centro de Gravidade ou Método do Centróide, é a técnica de defuzzificação mais utilizada. Diferentemente do MM, a técnica do Centro de Área considera toda a distribuição de possibilidade de saída do modelo para calcular o valor clássico. O procedimento é similar ao usado para calcular o centro de gravidade em física, se considerarmos a função de pertinência $\mu_A(x)$ como a densidade de massa de x . Por outro lado, o CA pode ser compreendido como uma média ponderada, onde $\mu_A(x)$ funciona como o peso do valor x . Se x for discreto, então a defuzzificação da conclusão *fuzzy* A será dada por:

$$y_0 = \frac{\sum_x x \cdot \mu_A(x)}{\sum_x \mu_A(x)}. \quad (\text{B.5.3})$$

Da mesma forma, se x for contínuo, então:

$$y_0 = \frac{\int x \cdot \mu_A(x) dx}{\int \mu_A(x) dx}. \quad (\text{B.5.4})$$

A Figura B.14 exemplifica o método de defuzzificação CA. A principal desvantagem desse método é o seu custo computacional, principalmente no caso em que x é contínuo.

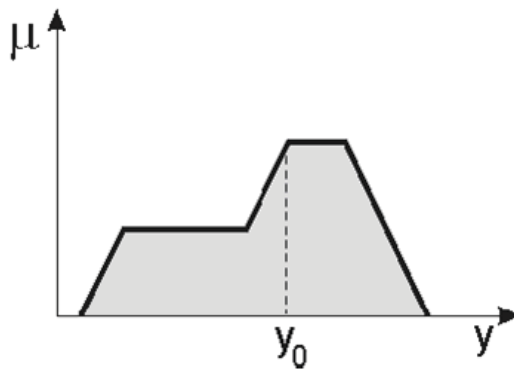


Figura B.14: Exemplo do Método de Defuzzificação CA.

B.5.3 Método das Alturas

O método de defuzzificação das Alturas (MA) pode ser visto como uma aproximação do método do Centro de Área. Ele é utilizado em dois passos. Primeiro, converte-se as funções de pertinência C_i , os consequentes das regras, em consequentes clássicos $y = c_i$, onde c_i é o centro de gravidade de C_i . A defuzzificação do centróide é então aplicada para as regras com os consequentes clássicos através da expressão [32, 69]:

$$y_0 = \frac{\sum_{i=1}^m c_i \omega_i}{\sum_{i=1}^m \omega_i}. \quad (\text{B.5.5})$$

em que ω_i é grau de ativação da i -ésima regra e m o número de regras *fuzzy* do modelo. Uma vez definidas as regras os valores c_i estarão determinados e se manterão fixos durante toda a inferência, o que simplifica muito o cálculo do valor clássico representativo, reduzindo o custo computacional. A Figura B.15 ilustra o exemplo desse método de defuzzificação.

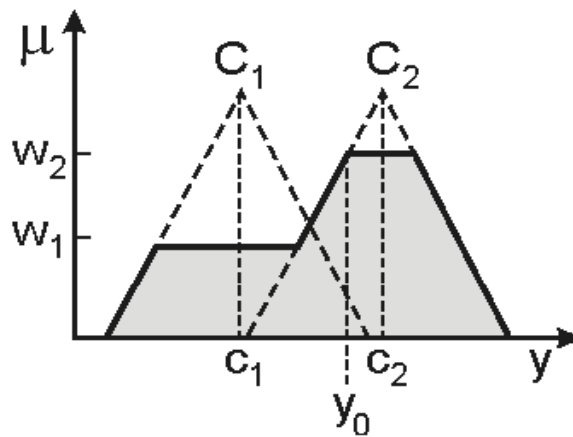


Figura B.15: Exemplo do Método de Defuzzificação MA.

Algumas considerações se fazem necessárias:

Média dos Máximos: “Solução mais plausível” pois desconsidera o formato das funções de pertinência da saída. Trata-se de um **método descontínuo**. Usado para o reconhecimento de padrões (identificar objetos pela classificação do sinal de um sensor; suporte à decisões (qualitativas: detecção de fraude em cartões de crédito e avaliação de credito).

Centro de Área: Desvantagens: esforço computacional e a sobreposição de áreas para uma mesma saída *fuzzy* não é corretamente contabilizada. Trata-se de um **método contínuo**.

Método das Alturas: Trata-se de um **método contínuo**. Usa apenas os máximos das funções de pertinências. Aplicado em malha fechada (a propriedade da continuidade é importante); suporte à decisão (decisões quantitativas: alocação de recursos ou priorização de projetos).

Dados do Sistema de Energia Elétrica

A seguir apresentam-se os dados [42] referentes a uma configuração baseada no sistema sul-brasileiro (Figura 33) composto por 10 máquinas síncronas, 45 barras e 73 linhas de transmissão (Tabelas C.1, C.2, C.3). Os dados referem-se a uma base de 100 MVA. Este sistema será a base de análise desta pesquisa, sem que se descarte o uso de outros sistemas.

Tabela C.1: Dados de barras do sistema sul-brasileiro (10 máquinas).

Identificação		Tensão		Carga		Geração	
Nº	Nome	Módulo	Ângulo	Ativa (pu)	Reativa (pu)	Ativa (pu)	Reativa (pu)
1	Barracão 13,8	1,020	-11,430	0,000	0,000	6,500	1,007
2	P. Fundo 13,8	1,040	-17,100	0,000	0,000	2,150	1,496
3	S. Osório 13,8	1,020	5,080	0,000	0,000	10,500	2,450
4	F. Areia 13,8	1,022	-6,380	0,000	0,000	11,100	3,377
5	S. Santiago 13,8	1,018	3,290	0,000	0,000	13,250	1,651
6	J. Lacerda A 13,8	1,030	-29,920	0,000	0,000	0,900	0,465
7	J. Lacerda B 13,8	1,030	-28,380	0,000	0,000	1,200	0,548
8	J. Lacerda C 13,8	1,030	-27,020	0,000	0,000	2,410	0,905
9	Itaúba 13,8	1,000	-16,150	0,000	0,000	4,900	1,365
10	Segredo 13,8	1,020	0,000	0,000	0,000	13,640	2,051
11	Ivaiporã 525	1,007	-11,180	0,000	-2,130	0,000	0,000

Tabela C.1 - Continuação.

Identificação		Tensão		Carga		Geração	
N°	Nome	Módulo	Ângulo	Ativa (pu)	Reativa (pu)	Ativa (pu)	Reativa (pu)
12	Londrina 525	0,995	-13,510	0,000	0,000	0,000	0,000
13	Siderópolis 230	0,969	-37,470	1,770	0,680	0,000	0,000
14	Farroupilha 230	0,978	-37,240	1,910	0,420	0,000	0,000
15	P. Fundo 230	0,978	-22,680	1,710	0,185	0,000	0,000
16	Xanxerê 230	0,946	-17,990	1,260	0,470	0,000	0,000
17	Pato Branco 230	0,954	-10,870	0,460	0,147	0,000	0,000
18	S. Osório 230	0,999	-1,660	2,810	0,565	0,000	0,000
19	Areia 230	0,978	-15,420	2,790	0,607	0,000	0,000
20	S. Mateus 230	0,997	-26,600	1,300	-0,706	0,000	0,000
21	Curitiba 230	0,991	-26,760	4,270	-1,250	0,000	0,000
22	Joinville 230	0,952	-33,640	3,100	0,410	0,000	0,000
23	Blumenau 230	0,967	-32,190	4,240	0,094	0,000	0,000
24	R. Queimado 230	0,963	-34,310	1,170	0,531	0,000	0,000
25	Areia 525	1,003	-10,550	0,000	0,000	0,000	0,000
26	Curitiba 525	0,986	-24,330	0,000	-1,447	0,000	0,000
27	Cur. Norte 525	0,988	-22,750	3,680	-0,756	0,000	0,000
28	Blumenau 525	0,968	-30,100	0,000	0,000	0,000	0,000
29	Barracão 525	1,010	-16,350	1,740	-1,138	0,000	0,000
30	Gravataí 525	1,019	-33,630	0,000	-1,596	0,000	0,000
31	V. Aires 525	1,027	-26,790	0,000	-1,603	0,000	0,000
32	Pinheiro 525	1,018	-13,680	0,000	-1,615	0,000	0,000
33	S. Santiago 525	1,010	-5,150	0,000	0,000	0,000	0,000
34	J. Lacerda 138	0,994	-34,310	1,260	0,398	0,000	0,000
35	J. Lacerda 230	0,996	-33,090	0,000	0,000	0,000	0,000
36	Segredo 525	1,010	-5,160	0,000	0,000	0,000	0,000
37	Ceci 230	1,012	-37,550	8,130	1,100	0,000	0,000
38	Gravataí 230	1,026	-36,940	6,120	-4,550	0,000	0,000
39	Itaúba 230	0,975	-22,960	4,040	1,350	0,000	0,000
40	V. Aires 230	1,029	-29,400	3,930	-1,110	0,000	0,000
41	Apucarana 230	0,973	-18,930	2,620	0,132	0,000	0,000
42	Londrina 230	0,989	-15,600	2,290	0,830	0,000	0,000
43	Maringá 230	0,983	-18,900	1,840	-0,398	0,000	0,000
44	C. Mourão 230	0,981	-15,340	1,390	-0,463	0,000	0,000
45	Forquilha 230	0,977	-37,360	0,901	-0,447	0,000	0,000

Tabela C.2: Dados do sistema de transmissão.

N°	Barra de Origem	Barra de Destino	Impedância	
			$R(\%)$	$X(\%)$
1	Ivaiporã 525	Londrina 525	0,07	1,45
2	Ivaiporã 525	Londrina 525	0,07	1,45
3	Ivaiporã 525	Areia 525	0,18	2,27
4	Ivaiporã 525	S. Santiago 525	0,14	2,04
5	Londrina 525	Londrina 230	0,00	0,63
6	Barracão 13,8	Barracão 525	0,00	1,36
7	Siderópolis 230	Farroupilha 230	3,86	19,85
8	Siderópolis 230	J. Lacerda 230	0,96	4,91
9	Siderópolis 230	Forquilha 230	0,33	1,67
10	Farroupilha 230	P. Fundo 230	4,63	23,78
11	Farroupilha 230	P. Fundo 230	4,63	23,78
12	Farroupilha 230	Ceci 230	1,77	9,10
13	Farroupilha 230	Ceci 230	1,77	9,10
14	Farroupilha 230	Ceci 230	1,77	9,10
15	P. Fundo 13,8	P. Fundo 230	0,00	4,60
16	P. Fundo 230	Xanxerê 230	1,63	8,35
17	P. Fundo 230	Xanxerê 230	1,63	8,35
18	P. Fundo 230	Itaúba 230	2,50	15,48
19	Xanxerê 230	P. Branco 230	1,63	8,35
20	Xanxerê 230	S. Osório 230	3,16	16,21
21	P. Branco 230	S. Osório 230	1,53	8,61
22	S. Osório 13,8	S. Osório 230	0,00	1,14
23	S. Osório 230	Areia 230	3,06	15,23
24	S. Osório 230	C. Mourão 230	3,44	17,60
25	S. Osório 230	C. Mourão 230	3,44	17,60
26	Areia 230	S. Mateus 230	2,45	12,56
27	Areia 230	Areia 525	0,00	3,00
28	S. Mateus 230	Curitiba 230	0,88	4,15
29	Curitiba 230	Joinville 230	1,82	9,35
30	Curitiba 230	Joinville 230	1,82	9,35
31	Curitiba 230	Curitiba 525	0,00	0,62
32	Joinville 230	Blumenau 230	1,54	7,76
33	Joinville 230	Blumenau 230	1,54	7,76
34	Blumenau 230	R. Queimado 230	2,16	11,05
35	Blumenau 230	R. Queimado 230	2,16	11,05
37	R. Queimado 230	J. Lacerda 230	1,80	9,20
38	R. Queimado 230	J. Lacerda 230	1,80	9,20
39	F. Areia 13,8	Areia 525	0,00	0,67
40	Areia 525	Curitiba 525	0,19	2,80
41	Areia 525	Cur. Norte 425	0,19	2,74
42	Areia 525	Barracão 525	0,14	1,95
43	Areia 525	Segredo 525	0,05	0,70
44	Curitiba 525	Cur. Norte 525	0,05	0,69

Tabela C.2 - Continuação.

N°	Barra de Origem	Barra de Destino	Impedância	
			$R(\%)$	$X(\%)$
45	Curitiba 525	Blumenau 525	0,12	1,75
46	Barracão 525	Gravataí 525	0,21	3,09
47	Gravataí 525	Gravataí 230	0,00	0,62
48	V. Aires 525	Pinheiro 525	0,22	3,00
49	V. Aires 525	V. Aires 230	0,00	0,62
50	Pinheiro 525	S. Santiago 525	0,14	1,95
51	S. Santiago 13,8	S. Santiago 525	0,00	1,14
52	S. Santiago 525	Segredo 525	0,05	0,70
53	J. Lacerda A 13,8	J. Lacerda 13,8	0,00	8,71
54	J. Lacerda 13,8	J. Lacerda 230	0,00	5,90
55	J. Lacerda B 13,8	J. Lacerda 230	0,00	7,01
56	J. Lacerda C 13,8	J. Lacerda 230	0,00	4,50
57	J. Lacerda 230	Forquilha 230	1,29	6,57
58	Segredo 13,8	Segredo 525	0,00	0,68
59	Ceci 230	Gravataí 230	0,22	1,11
60	Ceci 230	Gravataí 230	0,22	1,11
61	Ceci 230	Gravataí 230	0,22	1,11
62	Ceci 230	V. Aires 230	2,07	9,33
63	Ceci 230	V. Aires 230	2,07	9,33
64	Ceci 230	V. Aires 230	2,07	9,33
65	Itaúba 13,8	Itaúba 230	0,00	2,36
66	Itaúba 230	V. Aires 230	2,02	11,29
67	Apucarana 230	Londrina 230	1,25	6,40
68	Apucarana 230	Londrina 230	0,89	4,61
69	Apucarana 230	Maringá 230	1,10	11,84
70	Apucarana 230	C. Mourão 230	2,29	11,74
71	Londrina 230	Maringá 230	1,72	8,84
72	Londrina 230	Maringá 230	1,72	8,84
73	Maringá 230	C. Mourão 230	1,81	9,29

Tabela C.3: Dados das máquinas síncronas.

Número da Máquina Síncrona	$X'd(\%)$	$H(s)$	Potência
			Mecânica (pu)
1	3,67	30,44	6,50
2	10,37	10,56	2,15
3	2,84	38,34	10,50
4	2,41	62,25	11,10
5	2,43	55,24	13,25
6	13,52	4,37	0,90
7	15,34	6,80	1,20
8	8,00	12,50	2,41
9	4,32	20,24	4,90
10	2,16	62,46	Referência

Artigos Publicados Relacionados com a Pesquisa de Doutorado

Neste apêndice são apresentados os artigos que foram publicados relacionados com a pesquisa de doutorado:

1. MORENO, A. L.; MINUSSI, C. R. Análise da estabilidade transitória de sistemas de energia elétrica usando uma rede neural ART euclidiana ARTMAP com treinamento continuado. In: INTERNATIONAL CONFERENCE OF ENGINEERING AND TECHNOLOGY EDUCATION, 11., 2010, Ilhéus. *Proceedings...* New York: IEEE, 2010. p.622-627.
2. MORENO, A. L.; MINUSSI, C. R. Análise da estabilidade transitória de sistemas elétricos usando uma rede neural euclidiana ARTARTMAP. In: SEMINÁRIO BRASILEIRO DE ANÁLISE, 70, 2009. São Paulo. *Anais...* São Paulo: IME-USP, 2009. p.85-97.
3. Finalização de um artigo a ser submetido a veículo científico internacional *peer review* da área de Engenharia Elétrica ou de áreas afins.