

**UNESP - UNIVERSIDADE ESTADUAL PAULISTA
CÂMPUS DE ILHA SOLTEIRA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

Concurso de Livre-Docência

Texto Sistematizado

Disciplina:

Matemática Aplicada à Engenharia Elétrica

Candidata: Anna Diva Plasencia Lotufo

Ilha Solteira (SP), Maio-2016.

APRESENTAÇÃO

De acordo com a Resolução UNESP-27 (de 15-4-2009), que dispõe sobre norma para a obtenção do título de livre-docência, além de vários requisitos de desempenho, em termos de ensino, pesquisa e de extensão, o candidato deverá apresentar, para o julgamento, o desenvolvimento de pesquisa original (tese), ou texto que sistematize a produção acadêmica visando demonstrar maturidade científica, especialmente após a obtenção do título de doutor. Neste concurso, a opção escolhida é pela apresentação do texto sistematizado. Outro requisito refere-se à apresentação do memorial documentado. Neste caso, a constituição processual adotada é o texto correspondente ao Currículo Lattes-CNPq completo com a devida documentação comprobatória, no qual se encontra listado um grande número de informações sobre o exercício da docência.

Por conseguinte, este relatório contém as atividades de pesquisa desenvolvidas após a obtenção do título de Doutor, com vista à realização do concurso de livre-docência, constituindo, na essência, o texto sistematizado. Acompanha o texto sistematizado, as principais ferramentas que são usadas e citadas no texto.

O título de Doutor, em Engenharia Elétrica, foi obtido em 2004, junto ao PPGEE (Programa de Pós-Graduação em Engenharia Elétrica) da UNESP, Câmpus de Ilha Solteira. Atualmente, o PPGEE possui o conceito 6 de acordo com a avaliação CAPES-MEC. O mestrado em Engenharia Elétrica na UFSC (Universidade Federal de Santa Catarina) em 1982, anteriormente a contratação na UNESP que se deu em 1984. A graduação em Engenharia Elétrica na UFSM (Universidade Federal de Santa Maria), em 1978; graduação em Licenciatura em Matemática na Faculdade de Filosofia Ciências e Letras Imaculada Conceição (atualmente UNIFRA-Centro Universitário Franciscano), Santa Maria (RS), em 1977.

A Candidata tem atuado em pesquisa na área de Engenharia Elétrica com ênfase em SEP (Sistemas Elétricos de Potência). Neste contexto, o maior interesse tem sido dedicado ao desenvolvimento de pesquisa e orientação usando técnicas nominadas inteligentes: redes neurais artificiais, lógica *fuzzy* e sistemas imunológicos artificiais para a resolução de alguns problemas atuais de SEP. Os principais problemas abordados compreendem a estabilidade transitória, previsão de carga, afluências, ventos e de preço da energia, bem como a detecção e a classificação de distúrbios em sistemas de distribuição. São abordados outros temas de caráter geral que, igualmente são resolvidos via técnicas inteligentes. Assim sendo, na sequência, serão descritas as principais atividades desenvolvidas pela Candidata.

**UNESP - UNIVERSIDADE ESTADUAL PAULISTA
CÂMPUS DE ILHA SOLTEIRA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

Concurso de Livre-Docência

SISTEMAS INTELIGENTES E APLICAÇÕES

Candidata: Anna Diva Plasencia Lotufo

Ilha Solteira (SP), Maio-2016.

I ANÁLISE DE ESTABILIDADE

A segurança dos sistemas elétricos é fundamental, tanto para os usuários, assim como para manter o sistema funcionando adequadamente sem falhas. Desta forma, o estudo da análise de estabilidade é a capacidade de manter o sistema operando mesmo quando sujeito a distúrbios. Este foi o tema da tese de doutorado.

A defesa de tese ocorreu em agosto de 2004 com o título *Análise de Sensibilidade por Redes Neurais para estudos da estabilidade transitória de sistemas elétricos de potência*. Nessa pesquisa foi usada a rede neural *perceptron* multicamadas com treinamento via técnica *backpropagation* com algumas modificações no treinamento visando acelerar a convergência: (1) uso de um controlador nebuloso; (2) mudanças na função de ativação como alterando a inclinação e translação. A rede neural foi usada para determinar a sensibilidade a margem de segurança em relação às injeções de potência nodais. Baseado nestes coeficientes de sensibilidade, a partir de um estado inseguro, pode-se determinar mudanças das potências ativas das máquinas síncronas do sistema com vistas a obter um estado seguro. Essa tese gerou um artigo que foi publicado no periódico *Electric Power Systems Research* após a defesa conforme:

LOTUFO, A.D.P., LOPES, M.L.M. e MINUSSI, C.R., Sensitivity analysis by neural networks applied to power systems transient stability, **Electric Power Systems Research**, vol. 77, p. 730–738, 2007.

O programa computacional da tese foi desenvolvido em linguagem de programação Fortran. Assim, foi proposto um projeto de Iniciação Científica à FAPESP (Fundação de Amparo à Pesquisa do estado de São Paulo) para adaptá-lo ao aplicativo MATLAB. Nesse trabalho, foram utilizadas, também, outras variações da rede neural como o uso do “termo momento” para melhorar a convergência e, também, o treinamento por Levenberg-Marquardt. Esta bolsa foi concedida e o trabalho de IC gerou dois artigos em congressos:

Projeto IC

DONDA, Jean Carlos Gomes, **Análise de Sensibilidade por Redes Neurais Aplicada ao Estudo da Estabilidade Transitória de Sistemas Elétricos de Potência**. 2009. *Bolsista da FAPESP. Bolsa concedida em junho de 2006, renovada em junho de 2007, e novamente em maio de 2008. Encerrada em maio 2009.*

Artigos relativos ao projeto

- 1 DONDA, J. C. G., LOTUFO, A. D. P., LOPES, M. L. M., MINUSSI, C. R., Análise de Sensibilidade por Redes Neurais Aplicada ao Estudo da Estabilidade Transitória de Sistemas Elétricos de Potência In: 7th Brazilian Conference on Dynamics Control and Applications, 2008, Presidente Prudente, SP, **Proceedings of the 7th Brazilian Conference on Dynamics Control and Applications**, 2008, p.810 – 816.
- 2 DONDA, J. C. G., LOTUFO, A. D. P., LOPES, M. L. M., MINUSSI, C. R., Análise se Sensibilidade por Redes Neurais Aplicada ao Estudo da Estabilidade Transitória de Sistemas Elétricos de Potência Usando as Toolboxes do MATLAB In: VIII Congresso Brasileiro de Redes Neurais, 2007, Florianópolis, **VIII Congresso Brasileiro de Redes Neurais**. Florianópolis, SC, 2007

Com relação a este assunto da tese de doutorado, orientei outro aluno de IC sem bolsa que realizou algumas alterações no programa com vistas a empregar outra rede neural. A rede neural foi a GRNN (*General Regression Neural Network*), que agora irá avaliar a estabilidade do sistema e realizar o controle preventivo uma vez que a análise de sensibilidade somente é possível com a rede neural com treinamento por backpropagation, ou seja, depende das derivadas da função de ativação.

Marcus Felipe Calori Jorgetto. **Redes Neurais Aplicada ao Estudo da Estabilidade Transitória de Sistemas Elétricos de Potência usando a**

função GRNN do MATLAB. 2010. Iniciação científica (Engenharia Elétrica).

II PREVISÕES

A previsão de carga é fundamental para planejamento e operação de sistemas elétricos para garantir uma operação confiável e segura. Desta forma, vários são os métodos utilizados para esta finalidade. Durante muito tempo prevaleceram os métodos estatísticos, mas com o advento das redes neurais estas suplantaram, com vantagens, os tradicionais métodos estatísticos. Sendo assim, as pesquisas aqui relatadas trazem como ferramenta base as Redes Neurais Artificiais.

Uma previsão de carga confiável também deve muito a uma boa previsão de afluências, que é a quantidade de água que chega a um reservatório. Como a geração elétrica brasileira é basicamente hidrelétrica, a previsão de afluências também é importante no contexto do planejamento e operação do sistema elétrico.

A saber, Redes Neurais Artificiais, são representações matemáticas do cérebro humano a partir de um modelo de neurônio artificial que quando conectados com vários outros neurônios, formando uma rede e treinadas por algoritmos de treinamento, simulam diversas atividades dentre as quais por exemplo efetuar previsões. Os algoritmos de treinamento são baseados principalmente em métodos de otimização não-linear destacando-se o gradiente descendente de primeira ordem (*backpropagation*) e um de segunda ordem baseado nos métodos de Newton (Levenberg-Marquardt).

Portanto, iniciou-se trabalhando com as redes neurais mais básicas e tradicionais destacando-se a *perceptron* multicamadas com treinamento por *backpropagation* (retropropagação) e por Levenberg-Marquardt. Outra rede bastante utilizada também é a RBF (*Radial Basis Function*). A RBF é baseada em

funções de base radial destacando-se a função Gaussiana. Estas redes neurais podem ser programadas em linguagem de programação ou a sua versão para o MATLAB que consta das conhecidas *toolboxes*. Destaca-se também o uso da rede GRNN contida no MATLAB que é uma rede extremamente rápida na convergência (com uma iteração) pois como é baseada em densidade de probabilidade dificilmente encontra mínimos locais o que pode ser problema para a convergência dos treinamentos por backpropagation ou Levenberg-Marquardt.

Lógica Nebulosa é outra ferramenta que explora as incertezas do comportamento do cérebro humano e pode ser utilizada em associação com redes neurais artificiais durante o treinamento para melhorar o desempenho da rede neural. No MATLAB existe uma rede conhecida como ANFIS (*Adaptive Neuro Fuzzy Inference System*) que considera a lógica nebulosa associada a rede neural.

A utilização destas redes neurais citadas acima para previsão de carga e afliências bem como associada com lógica Nebulosa (ANFIS do MATLAB) foi objeto de projetos de Iniciação Científica com financiamento FAPESP e que resultaram também em artigos para congressos da área conforme segue:

1 **Projetos de IC com financiamento FAPESP**

1.1 **NOSE FILHO, Kenji, Estudo e Aprendizado de Redes Neurais Artificiais e Lógica Nebulosa para Aplicação em Previsão de Cargas Elétricas.** 2007. *Bolsista da FAPESP. Concedida em novembro de 2005, renovada em novembro de 2006 e encerrada em outubro de 2007.*

1.2 **MAEDA, Jorge Luiz Yoshito, Previsão de Cargas Elétricas através de uma Rede Neural de Base Radial (RBF) utilizando as toolboxes do MATLAB.** 2008. *Bolsista da FAPESP, Bolsa concedida em junho 2007 e concluída em maio de 2008.*

1.3 **ARAUJO, Klayton Antonio Moreira, Previsão de Cargas Elétricas através de uma Rede Neural Feedforward com treinamento por Levenberg-Marquardt utilizando as toolboxes do MATLAB.** 2008.

Bolsista da FAPESP. Concedida em junho de 2008, renovada em maio de 2009, e novamente em maio 2010, sendo encerrada a pedido em agosto 2010, pois o discente foi participar de intercâmbio pelo acordo BRAFITEC.

1.4 VITTI, Bruno Cesar, **Previsão de Afluências Através de uma Rede Neural de Base Radial (RBF) Utilizando as Toolboxes do MATLAB.** 2009. *Bolsista FAPESP, Concedida em outubro 2009 e encerrada em setembro 2010.*

2 **Artigos completos publicados em eventos relacionados com os projetos acima:**

2.1 MAEDA, J. L. Y., LOTUFO, A. D. P., LOPES, M. L. M., Previsão de Cargas Elétricas Através de Uma Rede Neural de Base Radial (RBF) Utilizando A Função GRNN do MATLAB In: 7th Brazilian Conference on Dynamics, Control and Applications, 2008, Presidente Prudente, SP. **Proceedings of the 7th Brazilian Conference on Dynamics, Control and Applications**, 2008, p.840 - 846.

2.2 NOSE FILHO, K., LOTUFO, A. D. P., LOPES, M. L. M., Utilização de Redes Neurais Artificiais e Redes Neuro Fuzzy para Previsão de Cargas Elétricas In: 7th Brazilian Conference on Dynamics, Control and Applications, 2008, Presidente Prudente, SP., **Proceedings of the 7th Brazilian Conference on Dynamics, Control and Applications**, 2008, p.867 - 872

2.3 ARAUJO, K.A.M., LOTUFO, A. D. P., LOPES, M. L. M., Previsão de cargas elétricas através de uma Rede Neural Feedforward com treinamento por Levenberg-Marquardt utilizando as toolboxes do MATLAB, In: 8th Brazilian Conference on Dynamics, Control and Applications, 2009, Bauru, SP., **Proceedings of 8th Brazilian Conference on Dynamics, Control and Applications**, 2009.

2.4 NOSE FILHO, K., ARAUJO, K.A.M., MAEDA, J. L. Y., LOTUFO, A. D. P.,

Previsão de cargas elétricas utilizando diferentes arquiteturas de redes neurais artificiais com o auxílio das toolboxes do MATLAB, In: 8th Latin American Congress on Electricity Generation and Transmission, 2009, Ubatuba, SP., **Proceedings of 8th Latin American Congress on Electricity Generation and Transmission**, 2009.

2.5 NOSE FILHO, K., MAEDA, J. L. Y., LOTUFO, A. D. P., Comparação de Arquiteturas de Redes Neurais Artificiais Utilizadas Para a Previsão De Cargas Elétricas Utilizando as Toolboxes do MATLAB In: Simpósio Brasileiro de Automação Inteligente- SBAI 2009, 2009, Brasília, DF., **Simpósio Brasileiro de Automação Inteligente**, 2009.

2.6 ARAUJO, K.A.M., VITTI, B. C., LOTUFO, A. D. P., LOPES, M. L. M., Previsão de Afluências Utilizando Redes Neurais Artificiais com o Auxílio do MATLAB, In: XVIII Congresso Brasileiro de Automática, 2010, Bonito, MS., **Anais do XVIII Congresso Brasileiro de Automática**, 2010. p.2585 - 2591.

Com o credenciamento junto ao Programa de Pós-Graduação em Engenharia Elétrica, em 2010, começou a orientação inicialmente de mestrados e continuou-se a explorar a previsão de cargas por redes neurais. Nesta fase da pesquisa sobre previsão de carga era importante também diminuir o erro médio quadrático (MAPE) e, desta forma, novas arquiteturas de redes neurais foram exploradas. Outra experiência foi com o que se chama métodos híbridos seja com mais de uma rede neural ou com uma rede neural e um método estatístico. Aqui cita-se a rede neural baseada na teoria da ressonância adaptativa (ART) mais particularmente a rede neural ARTMAP Nebulosa que é extremamente rápida e sempre converge. Quanto aos métodos estatísticos, utilizou-se o ARIMA (*AutoRegressive Integrated Moving Average*) de Box Jenkins que durante muito tempo foi bastante utilizado para a finalidade de modelagem e previsão de sistemas físicos. Outro método estatístico utilizado foi a seleção de variáveis *stepwise* em que se procurou determinar quais variáveis eram realmente importantes para serem usadas como entradas da rede neural.

As vantagens da rede neural GRNN foram exploradas resultando numa rede GRNN modificada para torná-la ainda mais rápida e aplicada para as previsões. A vantagem de um treinamento rápido é a possibilidade de uso em tempo real.

Outra variação de uma previsão de carga é efetuar a previsão por barra da subestação e não de forma global. A previsão multimodal procura mostrar de maneira mais efetiva a previsão particularizando entre outros fatores o tipo de consumidor em que numa previsão global não é possível. Neste sentido, iniciou-se a orientação de trabalhos para efetuar a previsão multinodal a qual também contou com produção de trabalhos conjuntamente com os membros do grupo de pesquisa.

3 Mestrados concluídos

3.1 Thays Aparecida de Abreu Silva. **Previsão de Cargas Elétricas através de um Modelo Híbrido de Regressão com Redes Neurais Artificiais**. 2012. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

3.2 Cicero Marcelo de Oliveira. **Previsão de Cargas Elétricas Através de uma Rede Neural Híbrida BACK-ART FUZZY**. 2012. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

3.3 Kenji Nose Filho. **Previsão de Carga multinodal utilizando Redes Neurais de Regressão Generalizada**. 2011. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

3.4 José Roberto Campos. **Desenvolvimento de um sistema dinâmico para predição de cargas elétricas por redes neurais através do paradigma de programação orientada a objeto sob linguagem JAVA**. 2010. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

3.5 Marleide Ferreira Alves. **Previsão de Demanda de cargas elétricas por seleção de variáveis stepwise e Redes Neurais Artificiais**. 2013. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

4. Artigos completos publicados em eventos relacionados com os mestrados acima

4.1 ALTRAN, A. B., LOTUFO, A. D. P., LOPES, M. L. M., MINUSSI, C. R., Previsão de Carga Multinodal em Sistemas Elétricos de Potência Usando uma Rede Neural de Base Radial In: 8th Latin American Congress on Electricity, Generation and Transmission, 2009, Ubatuba, SP, **Proceedings of 8th Congress on Electricity, Generations and Transmission**, 2009.

4.2 NOSE FILHO, K., ALTRAN, A. B., MINUSSI, C. R., LOTUFO, A. D. P., Previsão de Carga Multinodal em Sistemas de Energia Elétrica Usando Redes Neurais Artificiais In: XVIII Congresso Brasileiro de Automática, 2010, Bonito, MS, **Anais do XVIII Congresso Brasileiro de Automática**, 2010. p.4418 - 4423.

4.3 CAMPOS, J. R., LOTUFO, A. D. P., MINUSSI, C. R., LOPES, M. L. M., Desenvolvimento de uma Plataforma JAVA na Previsão de Cargas Elétricas utilizando Redes Neurais Artificiais In: Congresso Brasileiro de Sistemas Fuzzy, 2010, Sorocaba, SP, **Proceedings of Congresso Brasileiro de Sistemas Fuzzy**, 2010. p.265 - 272.

4.4 CAMPOS, J. R., LOTUFO, A. D. P., MINUSSI, C. R., LOPES, M. L. M., Implementação se Redes Neurais Artificiais Utilizando a Linguagem de Programação JAVA In: 9th Brazilian Conference on Dynamics, Control and their Applications, 2010, Serra Negra, SP, **Proceedings of 9th Brazilian Conference on Dynamics, Control and their Applications**. Serra Negra, SP, 2010.

4.5 NOSE FILHO, K., LOTUFO, A. D. P., MINUSSI, C. R., Previsão de Potências Ativa e Reativa de Curto Prazo usando Redes Neurais Artificiais de Base Radial In: III Simpósio Brasileiro de Sistemas Elétricos, 2010, Belém, PA, **Proceedings of III Simpósio Brasileiro de Sistemas Elétricos**. Belém, PA, 2010.

4.6 NOSE FILHO, K., LOTUFO, A. D. P., MINUSSI, C. R., Pre-processing data for short-term load forecasting with a general regression neural network and a moving average filter In: IEEE Trondheim Powertech 2011, 2011, Trondheim, Noruega, **Proceedings of Powertech 2011**. Trondheim, Noruega, 2011. p.1 - 7.

4.7 NOSE FILHO, K., LOTUFO, A. D. P., MINUSSI, C. R., Short-term multinodal load forecasting in distribution systems using general regression neural networks In: IEEE Trondheim Powertech 2011, 2011, Trondheim, Noruega, **Proceedings of Trondheim Powertech 2011**. Trondheim, Noruega, 2011. p.1 - 7.

4.8 ABREU, T., ARAUJO, K.A.M., LOPES, M. L. M., LOTUFO, A. D. P., Método Híbrido utilizando os modelos ARIMA e Redes Neurais Artificiais para Previsão de Cargas Elétricas In: XIX Congresso Brasileiro de Automática, 2012, Campina Grande, PB, **XIX Congresso Brasileiro de Automática**, 2012, p.2229 - 2236.

4.9 ABREU, T., PAREDES, M., ARAUJO, K.A.M., LOPES, M. L. M., LOTUFO, A. D. P., Previsão de Cargas Elétricas através de um Modelo Híbrido de Regressão com Redes Neurais Artificiais In: IV Simpósio Brasileiro de Sistemas Elétricos, SBSE 2012, 2012, Goiania, GO, **Proceedings of IV Simpósio Brasileiro de Sistemas Elétricos**, 2012.

4.10 ALVES, M. F., LOTUFO, A. D. P., LOPES, M. L. M., Seleção de Variáveis Stepwise Aplicadas em Redes Neurais Artificiais para Previsão de Demanda de Cargas Elétricas In: XI Simpósio Brasileiro de Automação Inteligente e DINCON 2013, 2013, Fortaleza, CE, **Proceedings of XI Simpósio Brasileiro de Automação**

Inteligente and DINCON 2013. Fortaleza, CE, 2013.

4.11 OLIVEIRA, C. M., LOTUFO, A. D. P., LOPES, M. L. M., SILVEIRA, M. C. G., Previsão de Cargas Elétricas Através de Uma Rede Neural Híbrida Back-ART Fuzzy In: 11 Congresso Brasileiro de Inteligência Computacional, 2013, Porto de Galinhas, PE, **Proceedings of BRICS-CCI & CBIC 2013.** Porto de Galinhas, PE, 2013.

4.12 ABREU, T., SANTOS JUNIOR, C. R., LOPES, M. L. M., LOTUFO, A. D. P., Previsão de Cargas Elétricas através de um modelo de Regressão ARIMA e uma Rede Neural ARTMAP Fuzzy In: 11 Congresso Brasileiro de Inteligência Computacional, 2013, Porto de Galinhas, PE, **Proceedings of BRICS-CCI & CBIC 2013.** Porto de Galinhas, PE, 2013.

4.13 ABREU, T., ALVES, M. F., SANTOS JUNIOR, C. R., LOPES, M. L. M., LOTUFO, A. D. P., Rede Neural STEPWISE-ARTMAP Fuzzy aplicada ao problema de Previsão de Cargas Elétricas In: Congresso Nacional de Matemática Aplicada e Computacional, 2014, Natal, RN, **Proceedings of Congresso Nacional de Matemática Aplicada e Computacional.** SBMAC, 2014.

5 Artigos em periódicos provenientes das pesquisas supracitadas

5.1 NOSE FILHO, K., LOTUFO, A. D. P., MINUSSI, C. R., Short-term Multinodal Load Forecasting Using a Modified General Regression Neural Network. **IEEE Transactions on Power Delivery**, v.26, p.2862 - 2869, 2011.

5.2 ABREU, T., PAREDES, M., ARAUJO, K.A.M., LOPES, M. L. M., LOTUFO, A. D. P., Electrical Load Forecasting using a Hybrid Model with Artificial Neural Networks. **International Journal of Computer and Information Technology**, v.04, p.187 - 193, 2015.

Considerando as ferramentas de redes neurais utilizadas para a

previsão de carga, tentou-se utilizar as mesmas para a previsão de ventos. A geração eólica é uma fonte alternativa de energia, mas o vento, o “combustível” no caso, é de difícil previsão. Neste caso, os erros de trabalhos semelhantes encontrados na literatura são bastante altos, o que não foi diferente da metodologia utilizada que foi uma rede neural associada com o ARIMA de Box e Jenkins. Deste projeto orientou-se um mestrado com um trabalho publicado em evento, conforme abaixo:

Daniela Jardim Rossi. **Previsão da Velocidade dos Ventos por Redes Neurais Artificiais e ARIMA de Box & Jenkins**. 2013. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

ROSSI, D. J., LOTUFO, A. D. P., LOPES, M. L. M., Previsão da Velocidade dos Ventos por Redes Neurais Artificiais e ARIMA de Box & Jenkins In: Congresso de Matemática Aplicada e Computacional Sudeste, 2013, Bauru, SP, **Congresso de Matemática Aplicada e Computacional Sudeste**. Bauru, SP, 2013.

Por meio de um convênio de colaboração entre a UNESP e a UCLM (Universidad de Castilla La Mancha), intermediado pela CAPES, foi previsto intercâmbio entre docentes das duas universidades por um projeto coordenado pelo prof. Rubén Romero da UNESP e prof. Javier Contreras da UCLM. Desta forma, como fazia parte do projeto elaborei um plano de trabalho para desenvolver na UCLM sob a supervisão da Profa. Natalia Alguacil. O assunto escolhido foi a previsão de preços por redes neurais, pois a equipe da UCLM é bastante forte em mercados elétricos. No período passado na UCLM (dezembro 2011-julho 2012) fiz um levantamento bibliográfico sobre o tema e por causa da dificuldade de dados para simulação das previsões, optou-se por simular também estes dados. O método escolhido foi o de um fluxo de carga ótimo DC, diferente do encontrado na literatura, e obteve-se os dados usando o *software* MATPOWER que simula os mercados elétricos. Com os resultados obtidos, efetuou-se a previsão de preços por redes neurais pela orientação de um

mestrado que resultou em dois artigos publicados em eventos. Esta orientação contou com a colaboração da Profa. Natalia Alguacil da UCLM. Dentro do mesmo tema obteve-se três bolsas de IC da FAPESP, tais como:

Mestrado

Patricia Fernanda da Silva Freitas. **Previsão de Preços (LMP) por GRNN (Rede Neural de Regressão Generalizada)**. 2013. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho.

Artigos publicados relativos ao mestrado acima

1 FREITAS, P. F. S., ALGUACIL, N., LOTUFO, A. D. P., SILVEIRA, M. C. G., Generalized Regression Neural Network to Price Forecasting in Electricity Markets In: X Latin American Congress on Electric Power Generation, Transmission and Distribution - CLAGTEE 2013, 2013, Vina del Mar, Chile, **Proceedings of X Latin American Congress on Electric Power Generation, Transmission and Distribution - CLAGTEE 2013**. Vina del Mar, 2013.

2 FREITAS, P. F. S., LOTUFO, A. D. P., ALGUACIL, N., SILVEIRA, M. C. G., Um Método de Previsão de Preço da Carga utilizando Redes Neurais de Regressão Generalizada e Perceptron Multicamadas In: XI Simpósio Brasileiro de Automação Inteligente e DINCON 2013, 2013, Fortaleza, CE, **Proceedings of XI Simpósio Brasileiro de Automação Inteligente and DINCON 2013**. Fortaleza, CE, 2013.

Projetos de IC com financiamento FAPESP

1 MONTEIRO, Pedro Henrique Lehnhardt, **Mercados Elétricos e Redes Neurais com as ferramentas MATLAB e MATPOWER**. 2013. *Bolsista FAPESP, concedida em agosto 2013 e encerrada em janeiro 2014.*

2 MATSUDA, Gilian Sayuri Saccai, **Mercados Elétricos e Redes Neurais MLP com as ferramentas MATLAB e MATPOWER**. 2014. *Bolsa FAPESP, concedida em agosto 2014 e encerrada em agosto 2015.*

3 DE LUCA JUNIOR, Carlos Alberto, **Mercados Elétricos e Redes Neurais GRNN com as ferramentas MATLAB e MATPOWER**. 2014. *Bolsa FAPESP, concedida em agosto 2014 e encerrada em agosto 2015.*

III OUTRAS APLICAÇÕES

As aplicações de rede neurais são as mais amplas possíveis. Dominada a ferramenta, no caso a rede neural, aceitou-se uma orientação que utilizava a rede neural ARTMAP Nebulosa para detecção de *spams* (e-mails indesejados). A comunicação eletrônica é indispensável nos dias de hoje, mas a quantidade de lixo que circula é muito grande. Assim o trabalho é importante para auxiliar a evitar o tráfego de mensagens indesejadas.

Mestrado

Carlos Roberto dos Santos Junior. **Teoria da Ressonância Adaptativa através da linguagem JAVA para detecção e classificação de mensagens indesejadas**. 2013. Dissertação (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho.

Trabalho publicado

SANTOS JUNIOR, C. R., LOTUFO, A. D. P., SILVEIRA, M. C. G., Rede Neural ARTMAP fuzzy para detecção e classificação de e-mails indesejados, In: 11 Congresso Brasileiro de Inteligência Computacional, 2013, Porto de Galinhas, PE, **Proceedings of BRICS-CCI & CBIC 2013**. Porto de Galinhas, PE, 2013.

Em 2012, iniciou-se o credenciamento como orientador, também, de doutorado do PPGEE. Nestes anos o PPGEE mantinha dois DINTER (Programa de Doutorado CAPES interuniversidades), do qual tive a participação como professora, tanto no IFMT como na UNEMAT ministrando a disciplina “Redes Neurais Artificiais”. Do programa DINTER com a UNEMAT foram aceitos dois alunos.

Um deles tinha formação de Engenheiro Civil com mestrado em hidráulica e recursos hídricos. Assim pensou-se em retomar a previsão de aflúências por redes neurais utilizando, agora, a rede neural ARTMAP Nebulosa, que para previsão de cargas e outras aplicações mostrou um desempenho bastante satisfatório. Um dos problemas correntes de efetuar previsões com redes neurais é a dificuldade de obtenção de dados para treinamento e testes da rede neural. Este candidato efetuou um doutorado sanduiche no LNEC (Laboratório Nacional de Engenharia Civil) em Lisboa, Portugal. Com a supervisão da Dra. Maria Teresa Reis, o foco do trabalho mudou um pouco e as previsões foram para agitação marítima e galgamentos em áreas costeiras, em que o laboratório possui um extenso banco de dados com medida *in loco* e também por simulação numérica. A importância da previsão destas variáveis destaca-se pela fragilidade das áreas costeiras e portos às variações dos ventos sobre o mar. Assim, uniu-se a experiência da Dra. Maria Teresa em hidráulica costeira com as redes neurais que era uma novidade nos trabalhos do LNEC. Em junho de 2013, visitei o LNEC (recursos parcialmente financiados pela FAPESP) quando discuti pessoalmente o desenvolvimento do trabalho com a Dra. Maria Teresa. A tese foi defendida em 2013 e diversos trabalhos foram publicados conforme segue, e um artigo foi aceito para publicação em periódico da área:

Doutorado

Francisco Lledo dos Santos. **Redes Neurais Artificiais ARTMAP Fuzzy Aplicadas ao Estudo de Agitação Marítima e Ondas de Lagos**. 2013. Tese (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

Trabalhos publicados em eventos

- 1 SANTOS, F. L., NEVES, D. R., REIS, M. T., FORTES, C. J. E. M., POSEIRO, P., LOTUFO, A. D. P., MACIEL, G. F., Definition of Sines Port Wave Regime using an ARTMAP Artificial Neural Network with Fuzzy Logic, In: 3rd IAHR Europe Congress, 2014, Porto, Portugal, **Proceedings of 3rd IAHR Europe Congress**. Porto, Portugal, 2014. p.1 - 11.

- 2 SANTOS, F. L., REIS, M. T., FORTES, C. J., LOTUFO, A. D. P., MACIEL, G. F., Estudo de Galgamentos Utilizando Redes Neurais Artificiais do Tipo ARTMAP com Lógica Fuzzy In: 8a Jornadas Portugêsas de Engenharia Costeira e Portuária, 2013, Lisboa, Portugal, **Proceedings of 8ª Jornadas Portugêsas de Engenharia Costeira e Portuária**. Lisboa, Portugal, 2013. p.1 – 6.
- 3 SANTOS, F. L., LOTUFO, A. D. P., MACIEL, G. F., REIS, M. T., FORTES, C. J. E. M., Rede ARTMAP Nebulosa Aplicada ao Estudo de Agitação Marítima no Porto de Sines/Portugal In: 11 Congresso Brasileiro de Inteligência Computacional, 2013, Porto de Galinhas, PE, **Proceedings of BRICS-CCI & CBIC 2013**. Porto de Galinhas, PE, 2013.
- 4 SANTOS, F. L., REIS, M. T., FORTES, C. J., LOTUFO, A. D. P., Maciel, G. F., ARTMAP Artificial Neural Networks with Fuzzy Logic. Their Application to Study the Wave Conditions at Sines Port, Portugal In: 6th SCACR – International Short Course/Conference on Applied Coastal Research, 2013, Lisboa, Portugal, **Proceedings of 6th SCACR – International Short Course/Conference on Applied Coastal Research**, 2013. (resumo).

Trabalho aceito para periódico

SANTOS, F. L., REIS, M. T., FORTES, C. J., LOTUFO, A. D. P., NEVES, D. R., POSEIRO, P., MACIEL, G. F., Performance of a Fuzzy ARTMAP Artificial Neural Network in Characterizing the Wave Regime at the Port of Sines (Portugal). **Journal of Coastal Research**, aceito em julho de 2015 e aguardando publicação prevista para 2016.

O outro aluno aceito para doutorado do DINTER, após extensa pesquisa, sugeriu o tema da tese com uma aplicação de redes neurais. O tema foi reconhecimento de padrões por redes neurais, que é também uma das principais aplicações de redes neurais. Neste caso, o banco de dados foi obtido em um

endereço (*BCI Competition*) de uma competição internacional e de livre acesso sobre implantes cerebrais (para aquisição de sinais) em pacientes com alguma deficiência. Esta tese contou com a colaboração do colega Jozué Vieira Filho na parte de processamento de sinais. A rede neural utilizada foi uma ARTMAP Nebulosa.

Tese

Rhycardo Luiz Monteiro. **Uso de Redes Neurais ARTMAP Nebulosas para a classificação de padrões em sinais ECoG relacionados ao movimento dos dedos.** 2015. Tese (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho.

Artigos publicados

- 5 MONTEIRO, R. L., LOTUFO, A. D. P., VIEIRA FILHO, J., Reconhecendo padrões relacionados com o movimento dos dedos das mãos em sinais ECoG com o uso de Redes Neurais ARTMAP-Fuzzy, In: 11 Congresso Brasileiro de Inteligência Computacional, 2013, Porto de Galinhas, PE, **Proceedings of BRICS-CCI & CBIC 2013**. Porto de Galinhas, PE, 2013.

- 6 MONTEIRO, R. L., LOTUFO, ANNA DIVA P., VIEIRA FILHO, J., Redes Neurais ARTMAP-Fuzzy: Reconhecendo Padrões de Movimento de Dedos em Sinais ECOG, In: XII Simpósio Brasileiro de Automação Inteligente, 2015, Natal, RN, **Proceedings of XII Simpósio Brasileiro de Automação Inteligente**, 2015.

Ainda, em 2012, logo após o retorno do estágio de pós-doutoramento aceitei a orientação de um aluno que ainda não tinha tema definido de tese, mas que estava de saída para um período sanduiche na Universidade de Coimbra, em Portugal. Ficou acertado que ele trabalharia no tema sob supervisão do prof. Jorge Batista da Universidade de Coimbra e eu o orientaria na parte de Redes Neurais. Portanto, o tema foi de reconhecimento de padrões de imagens faciais.

O grupo do Prof. Jorge Batista montou um extenso banco de dados de imagens faciais que podem ser utilizadas para diversas aplicações entre elas a de segurança bancária. A princípio pretendia-se trabalhar com a rede neural ARTMAP Nebulosa. Em junho de 2013 também visitei a Universidade de Coimbra e discuti o trabalho pessoalmente com Prof. Jorge. Quando do retorno do aluno a UNESP e início do programa da rede neural, observou-se que esta não respondia adequadamente à proposta. Sendo assim optou-se por utilizar uma outra técnica inteligente que estava sendo usada pelo grupo, os Sistemas Imunológicos Artificiais (SIA). Os SIA são também baseados no sistema imunológico humano e dentre os diversos algoritmos utilizou-se o Algoritmo de Seleção Negativa que é baseado na discriminação das células entre próprias e não-próprias. Uma vantagem dos SIA em relação as redes neurais é que estes não necessitam de treinamento, diminuindo ainda mais o tempo de processamento. Desta forma, o banco de dados respondeu a contento e a defesa ocorreu em 2015 com um trabalho publicado em evento.

Tese

Jadiel Caparrós da Silva. **Aplicação de Sistemas Imunológicos Artificiais para Biometria Facial: Reconhecimento de Identidade baseado nas características de Padrões Binários**. 2015. Tese (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho.

Trabalho publicado em evento

- 7 SILVA, J. C., LIMA, F. P. A., LOTUFO, A. D. P., Batista, J., Identity Recognition using an Artificial Intelligence based on Artificial Immune System In: 3rd Asia-Pacific Conference on Computer Aided System Engineering (APCASE 2015), 2015, Quito, Equador, **Proceedings of 3rd Asia-Pacific Conference on Computer Aided System Engineering (APCASE)**. Quito, Equador, 2015.

No momento tem-se alguns orientados trabalhando principalmente

com redes neurais em especial a ARTMAP Nebulosa cujos trabalhos serão especificados logo a seguir e também com lógica *fuzzy*. Estes trabalhos tratam de previsões de carga de “baixo para cima”, isto é, a partir de cada residência e suas particularidades como a quantidade de aparelhos eletroeletrônicos existentes e suas especificidades, rotinas diárias dos moradores, temperatura diária entre outros fatores. Nos outros trabalhos que tratavam de previsão de carga esta era feita de forma global ou de “cima para baixo”. A previsão de baixo para cima é uma tendência moderna para adequar as residências em função dos modernos sistemas elétricos inteligentes, os conhecidos *smart grids*. Outra tendência de previsão é efetuar previsões diferenciadas como por exemplo para grandes consumidores, neste caso uma universidade. O banco de dados foi obtido por meio de um projeto da UNIOESTE com medidores inteligentes que foram instalados na universidade e os dados de carga, temperatura entre outros adquiridos a cada 15 minutos. Um dos orientados de doutorado fez o mestrado naquela instituição e já trabalhou com o banco de dados. Para o levantamento da curva de carga de uma residência utilizou-se a lógica *fuzzy* uma vez que é difícil contemplar a diversidade de residências e comportamentos. Sendo assim, a lógica nebulosa é capaz de com poucas informações efetuar o levantamento da curva de carga das residências baseada nas variáveis linguísticas e no comportamento humano.

Teses em andamento

- 1 Marcos Ricardo Muller. **Ferramentas de previsão de carga com informação de *smart grids***. (início 2014). Tese (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho.
- 2 Marleide Ferreira Alves. **Previsão de Cargas Elétricas Não-Residenciais Mistas por Redes Neurais ARTMAP-Fuzzy**. (início 2014). Tese (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho.
- 3 Thays Aparecida de Abreu Silva. **Determinação da carga elétrica residencial baseada em lógica nebulosa**. (início 2012). Tese (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho.

Trabalhos publicados em eventos relativos as teses acima até o momento

- 1 ABREU, T., LOPES, M. L. M., ALVES, U. N. L. T., MINUSSI, C. R., LOTUFO, ANNA D.P., Determinação de Curva de Carga Residencial Baseada num Sistema Fuzzy, In: XII Simpósio Brasileiro de Automação Inteligente, 2015, **Proceedings of XII Simpósio Brasileiro de Automação Inteligente**, 2015.

- 2 ABREU, T., ALVES, U. N. L. T., LOTUFO, ANNA DIVA P., MINUSSI, C. R., LOPES, M. L. M., Residential Electric Load Curve Profile Based on Fuzzy Systems In: 2015 IEEE PES Innovative Smart Grid Technologies Latin America, 2015, Montevideo, **Proceedings of 2015 IEEE PES Innovative Smart Grid Technologies Latin America**, 2015. p.623 - 629.

Para explorar a versatilidade do MATLAB e suas toolboxes (aquisição de última versão por meio de projeto FAPESP), está sob orientação atualmente um aluno de mestrado que irá usar as toolboxes de “Neural Networks” e “Fuzzy Logic” para efetuar previsões de carga dentro do contexto segmentado conforme tendência do “de baixo para cima”. Desta forma, este aluno irá prever a carga de um restaurante universitário cujos dados provém do projeto da UNIOESTE já comentado anteriormente. Pretende-se avaliar o comportamento das redes neurais contidas na *toolbox* principalmente as tradicionais com treinamento por *Backpropagation* e Levenberg-Marquardt, bem como RBF e GRNN.

Dissertação em andamento

Monara Pereira da Rosa Carvalho. **Previsão de carga de um restaurante universitário por redes neurais.** (início 2015).

As vantagens de uma rede neural são várias conforme já comentado com diversas aplicações. Sendo assim, fui procurada por um colega de outra área para desenvolver um trabalho conjunto com aplicação em um banco de dados real. O trabalho foi então proposto a um aluno de doutorado que está desenvolvendo a tese com a colaboração do prof. Alan Rodrigo Panosso. Trata-se de componentes químicos do solo em plantações de cana de açúcar que podem

comprometer o meio ambiente pela liberação de gás carbônico. Um assunto bastante atual tendo em vista a preocupação com os gases do efeito estufa. A rede neural utilizada é a *perceptron* multicamadas com treinamento por *backpropagation* e Levenberg-Marquardt.

Tese em andamento:

Luciana Paro Scarin Freitas. **Estudo da liberação de gás carbônico no ambiente utilizando redes neurais artificiais**, (início 2012). Tese (Engenharia Elétrica)

Como orientou-se um mestrado na área de segurança computacional, este mesmo aluno se candidatou ao doutorado e seguiu a mesma linha de pesquisa. Nesta nova abordagem, o candidato está trabalhando com uma rede neural ARTMAP Nebulosa modificada para o que se chama de treinamento continuado. A vantagem deste treinamento é que se surgir algum novo evento que não consta do banco de dados treinado ele pode ser incorporado sem a necessidade de treinar novamente a rede neural para adquirir o conhecimento. Desta forma, a rede continua sendo rápida e apta para aplicações em tempo real.

Tese em andamento

Carlos Roberto dos Santos Junior. **Deteção e Classificação de Spams por Rede Neural ARTMAP Fuzzy com Treinamento Continuado**. (início 2013). Tese (Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho

Trabalho publicado

SANTOS JUNIOR, C. R., ABREU, T., LOTUFO A.D.P., Rede neural ARTMAP Fuzzy com treinamento continuado para deteção de spams In: 12 Congresso Brasileiro de Inteligência Computacional, 2015, Curitiba, PR, **Proceedings of 12º Congresso Brasileiro de Inteligência Computacional**. Curitiba, PR, 2015.

IV Colaboração com outros membros do grupo de pesquisa

Desde o período do desenvolvimento do doutorado teve-se uma forte colaboração com os outros membros do grupo de pesquisa. Este envolvimento continuou após a defesa sempre procurando colaborar com os demais e vice-versa.

Com a colega Mara Lúcia Martins Lopes desenvolveu-se uma colaboração desde o mestrado desta (quando ainda não havia ingressado no doutorado) e continuou no seu doutorado resultando em diversas publicações. Esta parceria continua até hoje em que a Profa. Mara Lúcia tem uma atuação bastante importante no grupo. Foi por meio desta colaboração que temos trabalhado intensamente na previsão de cargas desde então.

Durante a realização do doutorado, com dois outros colegas (Maria do Carmo Gomes da Silveira e Wagner Peron Ferreira), sob a mesma orientação do Prof. Minussi, trabalhamos conjuntamente em parte de nossas teses que tinham um eixo comum e depois cada um seguiu suas particularidades e aplicações. Desta colaboração mútua publicou-se um artigo em periódico após as defesas que está listado com as demais publicações abaixo.

Assim durante a realização do doutorado das colegas Lilian Yuli Isoda e Sandra Cristina Marchiori procurou-se colaborar na área de estabilidade de sistemas elétricos uma vez que as candidatas têm a formação de Bacharel em Matemática.

Destas colaborações, teve-se participação em diversas publicações tanto em eventos como em periódicos conforme:

Artigos em eventos

1 ISODA, L. Y., LOTUFO, A. D. P., LOPES, M. L. M., MINUSSI, C. R., Desenvolvimento de uma Metodologia Neural Baseada na Teoria da Ressonância Adaptativa para Análise da Estabilidade de Tensão de Sistemas de Energia Elétrica In: 7th Brazilian Conference on Dynamics Control and Applications, 2008, Presidente Prudente, SP, **Proceedings of the 7th Brazilian Conference on Dynamics Control and Applications**, 2008, p.832 - 839.

2 ISODA, L. Y., LOTUFO, A. D. P., LOPES, M. L. M., MINUSSI, C. R., Métodos Matriciais no Estudo da Estabilidade de Tensão de Sistemas de Energia Elétrica In: 68 Seminário Brasileiro de Análise, 2008, São Paulo, SP, **Seminário Brasileiro de Análise**. São Paulo, SP: Instituto de Matemática e Estatística - USP, 2008. v.68. p.1 - 12.

3 ISODA, L. Y., LOTUFO, A. D. P., LOPES, M. L. M., MINUSSI, C. R., Voltage Stability Analysis of Electric Power Systems Using a Fuzzy ARTMAP Neural Network In: IX Congresso Brasileiro de Redes Neurais, 2009, Ouro Preto, MG, **Proceedings of IX Congresso Brasileiro de Redes Neurais**, 2009.

4 MARCHIORI, S. C., MINUSSI, C. R., LOTUFO, A. D. P., Análise da Estabilidade Transitória de Sistemas Elétricos Usando Uma Rede Neural ARTMAP Euclidiana com Treinamento Continuado In: Congresso Brasileiro de Automática- CBA2006, 2006, Salvador, Ba, **Congresso Brasileiro de Automática, CBA2006**, 2006. p.3146 - 3151.

Periódicos

1 ISODA, L. Y., LOTUFO, A. D. P., LOPES, M. L. M., MINUSSI, C. R., Análise de Estabilidade de Tensão em Sistemas Elétricos Usando Uma Rede Neural ARTMAP Fuzzy. TEMA. Tendências em Matemática Aplicada e Computacional, v.9, p.243 - 253, 2008.

2 FERREIRA, W. P., SILVEIRA, M. C. G., LOTUFO, A. D. P., MINUSSI, C. R., Transient stability analysis of electric energy systems via a fuzzy ART/ARTMAP neural network. Electric Power Systems Research. , v.76, p.466 - 475, 2006.

3 MARCHIORI, S. C., SILVEIRA, M. C. G., LOTUFO, A. D. P., MINUSSI, C. R., LOPES, M. L. M., Neural network based on adaptive resonance theory with continuous training for multi-configuration transient stability analysis of electric power systems. Applied Soft Computing, v.11, p.706 - 715, 2011.

4 LOPES, M. L. M., MINUSSI, C. R., LOTUFO, A. D. P., Electric Load Forecasting Using a Fuzzy ART&ARTMAP Neural Network. Applied Soft Computing, v.5, p.235 - 244, 2005.

5 LOPES, M. L. M., LOTUFO, A. D. P., MINUSSI, C. R., Application of the Fuzzy ART&ARTMAP Neural Network to the Electrical Load Forecasting Problem In: Forecasting Models - Methods and Applications ed. iConcept Press Pty Ltd, 2010, p. 179-190. (Capítulo de livro publicado)

V Utilização de Sistemas Imunológicos Artificiais

Uma outra técnica de inteligência computacional em uso pelo grupo de pesquisa é o Sistema Imunológico Artificial que já foi citado anteriormente na tese de Jadiel Caparrós da Silva. Sendo uma novidade no grupo, participa-se como colaboradora nos trabalhos desenvolvidos pelo aluno Fernando Parra dos Anjos Lima uma vez que ainda se estuda as vantagens dos algoritmos que compõem o sistema imunológico artificial. Os temas dos trabalhos são relativos a detecção de distúrbios em sistemas de distribuição de energia elétrica com banco de dados simulados pelo software EMTP, e outro de prevenção de câncer de mama usando-se um banco de dados disponível na web. Dentro desta colaboração foram publicados alguns artigos em eventos e periódicos e participou-se de um congresso com apresentação do artigo contando com recurso FAPESP para tal. Assim, são eles:

Congressos

1 LIMA, F. P. A., LOTUFO, A. D. P., MINUSSI, C. R., Artificial immune systems applied to voltage disturbance diagnosis in distribution electrical systems In: 2013 IEEE Grenoble PowerTech, 2013, Grenoble, France, **2013 IEEE Grenoble Conference**, IEEE, 2013.

2 LIMA, F. P. A., LOPES, M. L. M., LOTUFO, A. D. P., MINUSSI, C. R., Diagnóstico Clínico de Amostras de Câncer de Mama Utilizando Sistemas

Imunológicos Artificiais Com Seleção Negativa In: DINCON - Conferência Brasileira de Dinâmica, Controle e Aplicações, 2013, Fortaleza. Anais SBAI-DINCON, 2013, Fortaleza, CE, **DINCON - Conferência Brasileira de Dinâmica, Controle e Aplicações**. Fortaleza, CE, 2013.

Periódicos

1 LIMA, F. P.A., LOTUFO, A. D. P., MINUSSI, C. R., Wavelet-artificial immune system algorithm applied to voltage disturbance diagnosis in electrical distribution systems. **IET Generation, Transmission & Distribution** (Print), v.9, p.1104 - 1111, 2015.

2 LIMA, F. P. A., LOTUFO, A. D. P., MINUSSI, C. R., LOPES, M. L. M., Artificial Immune Systems with Negative Selection Applied To Clinical Diagnosis of Breast Cancer Samples. **Journal of Computer Engineering & Information Technology**, v.3, p.1 - 7, 2014.

3 LIMA, F. P.A., LOTUFO, A. D.P., MINUSSI, C. R., Disturbance detection for optimal database storage in electrical distribution systems using artificial immune systems with negative selection. **Electric Power Systems Research** (Print), v.109, p.54 - 62, 2014.

4 LIMA, F. P. A.; LOPES, M.L.M.; LOTUFO, A. D.P., MINUSSI, C. R., An Artificial Immune System with Continuous-Learning for Voltage Disturbance Diagnosis in Electrical Distribution System, **Expert Systems With Applications**, pp. 131-142, 2016.

ANEXO

PRINCIPAIS FERRAMENTAS UTILIZADAS

1. REDES NEURAIIS ARTIFICIAIS

1.1 *Perceptron* multicamadas

A *Perceptron* multicamadas é uma RNA (Rede Neural Artificial) das mais utilizadas em diversas aplicações, entre elas para a realização de previsões.

Esta rede neural possui mais de uma camada, geralmente uma escondida, além da entrada e da saída. Na Figura 1 apresenta-se a estrutura de uma MLP (Multi-Layer *Perceptron*).

O treinamento consiste em apresentar um par “entrada-saída” cuja saída será comparada com uma saída desejada e, portanto, vai existir um erro o qual será retropropagado da saída para a entrada visando recalculá-lo (atualização) os pesos dos neurônios até que este erro seja menor que um percentual predefinido (Haykin, 1999).

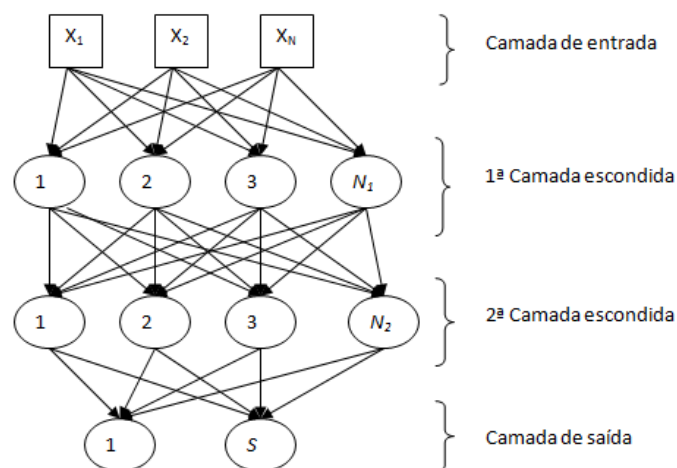


Figura 1 Arquitetura MLP.

1.1.1 Treinamento por retropropagação e retropropagação com momento

O treinamento, via técnica retropropagação (Werbos, 1974), é iniciado pela apresentação de um padrão X à rede, o qual produzirá uma saída Y . Em seguida calcula-se o erro de cada saída (diferença entre o valor desejado e a saída). O próximo passo consiste na determinação do erro propagado no sentido inverso, através da rede associada à derivada parcial do erro quadrático de cada elemento com relação aos pesos e, finalmente, ajustando os pesos de cada elemento. Um novo padrão é apresentado por vez, assim, o processo é repetido, até que ocorra a total convergência ($|\text{erro}| \leq \text{tolerância}$ preestabelecida). Os pesos iniciais são normalmente adotados como números aleatórios. O algoritmo retropropagação consiste na adaptação de pesos com o propósito de minimizar o erro quadrático da rede. A soma do erro quadrático instantâneo de cada neurônio alocado na última camada (saída da rede) é dada por (Widrow & Lehr, 1990):

$$\varepsilon^2 = \sum_{i=1}^{ns} \varepsilon_i^2 \quad (1)$$

sendo:

$$\varepsilon_i = d_i - y_i ;$$

d_i : saída desejada do i -ésimo elemento da última camada da rede;

y_i : saída do i -ésimo elemento da última camada da rede;

ns : número de neurônios da última camada da rede.

Considerando-se o neurônio de índice i da rede neural e utilizando-se o método do gradiente descendente (Widrow & Lehr, 1990), o ajuste de pesos pode ser formulado como (Widrow & Lehr, 1990):

$$V_i (h+1) = V_i (h) + \theta_i (h) \quad (2)$$

sendo:

$$\theta_i(h) = -\gamma [\nabla_i(h)];$$

γ : parâmetro de controle da estabilidade ou taxa de treinamento;

h : representa o índice de iteração;

$\nabla_i(h)$: gradiente do erro quadrático com relação aos pesos do neurônio i avaliado em h ;

V_i : vetor contendo os pesos do neurônio i

$$= [w_{0i} \ w_{1i} \ w_{2i} \ \dots \ w_{ni}]^T.$$

Na equação (2), a direção adotada para minimizar a função objetivo do erro quadrático corresponde à direção contrária ao gradiente. O parâmetro γ determina o comprimento do vetor $[\theta(h)]$.

A função sigmoide é definida pelas equações a seguir, (Widrow & Lehr, 1990), cujas representações gráficas são respectivamente (Figura 2):

$$y_i = \frac{1 - \exp(-\lambda s_i)}{1 + \exp(-\lambda s_i)} \quad (3)$$

ou:

$$y_i = \frac{1}{1 + \exp(\lambda s_i)} \quad (4)$$

sendo:

λ : constante que determina a inclinação da curva y_i .



Figura 2 Função Sigmoide.

Ressalta-se que o espectro da função sigmoide y_i , dada pela equação (3) ou (4), é $[-1, +1]$ ou $[0, +1]$, respectivamente.

O algoritmo retropropagação é abordado na literatura sob várias formas com o propósito de torná-lo mais rápido computacionalmente. Uma formulação bastante interessante refere-se ao algoritmo *retropropagação com momento* Widrow & Lehr, 1990), o qual tem um efeito estabilizador (Haykin, 1999, Widrow & Lehr, 1990).

$$V_{ij}(h+1) = V_{ij}(h) + \Delta V_{ij}(h) \quad (5)$$

em que:

$$\Delta V_{ij}(h) = 2 \gamma (1 - \eta) \delta_i(h) X_j + \eta \Delta V_{ij}(h - 1) \quad (6)$$

sendo:

V_{ij} : peso sináptico de interligação entre os neurônios i e j ;

η : constante de momento variando entre 0 e 1, $0 \leq \eta \leq 1$;

δ_j : derivada do erro quadrático (gradiente);

X : vetor de entrada;

$$\delta_j = \frac{\partial e_j^2}{\partial s_j} \quad (7)$$

sendo:

$$s_j(h) = V_j(h) X_j(h) \quad (8)$$

Os modos de treinamento podem ser de dois tipos, o chamado sequencial e o por lote. O sequencial é aquele em que o ajuste dos pesos é efetuado após cada padrão e o lote com todos os padrões ao mesmo tempo (Haykin, 1999).

Então, efetuando-se o cálculo do gradiente como indicado na equação (2), considerando-se a função sigmoide definida pela equação (3) ou (4) e o termo momento, obtém-se o seguinte esquema de adaptação de pesos (Widrow & Lehr, 1990):

$$V_{ij}(h+1) = V_{ij}(h) + \Delta V_{ij}(h) \quad (9)$$

sendo:

$$\Delta V_{ij}(h) = 2 \gamma (1 - \eta) \beta_j x_i + \eta \Delta V_{ij}(h - 1); \quad (10)$$

V_{ij} : peso correspondente à interligação entre o i -ésimo e j -ésimo neurônio;

γ : taxa de treinamento;

η : constante momento ($0 \leq \eta < 1$).

Se o elemento j encontrar-se na última camada, então:

$$\beta_j = \sigma_j \varepsilon_j \quad (11)$$

em que:

σ_j : derivada da função sigmoide dada pela equação (3) ou (4), respectivamente, com relação a s_j :

$$= 0,5 \lambda (1 - y_j^2) \quad (12)$$

$$= \lambda y_j (1 - y_j). \quad (13)$$

Se o elemento j encontrar-se nas demais camadas, tem-se:

$$\beta_j = \sigma_j \sum_{k \in R(j)} w_{jk} \beta_k \quad (14)$$

sendo:

$R(j)$: conjunto dos índices dos elementos que se encontram na fileira seguinte à fileira do elemento j e que estão interligados ao elemento j .

O parâmetro γ serve como controle de estabilidade do processo iterativo.

Os pesos da rede são inicializados aleatoriamente considerando-se o intervalo $[0,1]$ (Widrow & Lehr, 1990).

O treinamento, via método retropropagação, pode ser efetuado, basicamente, de duas formas:

- *Procedimento 1.* Consiste em ajustar os pesos da rede (considerando-se todas as camadas), fazendo-se com que haja convergência para cada padrão, até que se complete o conjunto de padrões de entrada. O processo deverá ser repetido até a total convergência, *i.e.*, o erro quadrático seja inferior a uma tolerância preestabelecida para os padrões considerados.
- *Procedimento 2.* Este procedimento é idêntico ao primeiro, porém, fazendo-se somente uma iteração (ajuste de pesos) por padrão em cada época.

1.1.2 Controlador Nebuloso

O algoritmo retropropagação é considerado, na literatura especializada, um referencial em termos de precisão. Contudo, a sua convergência é bastante lenta. Deste modo, ajusta-se a taxa de treinamento γ durante o processo de convergência, visando à redução do tempo de execução do treinamento. O ajuste de γ é efetuado via procedimento baseado em um controlador *fuzzy* (Lopes, 2003a).

A ideia básica da metodologia (Lopes, 2003a) consiste na determinação do estado do sistema definido como sendo o erro global εg e a variação do erro global $\Delta\varepsilon g$, cujo objetivo é a obtenção de uma estrutura de controle que consiga que o erro tenda para zero em um número reduzido de iterações, se comparado ao algoritmo retropropagação convencional. O controle é formulado usando os conceitos de lógica nebulosa (Kartalopoulos, 1996). O erro global εg e sua variação $\Delta\varepsilon g$ são os componentes do estado do sistema, e $\Delta\gamma$ é a ação de controle que deve ser exercida no sistema. Inicialmente, define-se o erro global (Lopes, 2003a):

$$\varepsilon g = \sum_{j=1}^{np} \sum_{i=1}^{ns} \varepsilon_i^2 \quad (15)$$

sendo:

εg : erro global da rede neural;

np : número de vetores padrões da rede.

O erro global corresponde ao cálculo de erros de todas as saídas (neurônios de saída), considerando-se todos os vetores padrões da rede. O treinamento deve ser executado utilizando o procedimento 2 (uma iteração por padrão). O erro global é calculado em cada iteração e ajustado o parâmetro γ , por meio de um acréscimo $\Delta\gamma$ determinado via lógica *fuzzy*. Este parâmetro será utilizado para ajustar o conjunto de pesos da rede referente à iteração subsequente. O estado do sistema e a ação de controle são assim definidos (Lopes, 2003a):

$$E^k = [\varepsilon g^k \quad \Delta\varepsilon g^k]^T \quad (16)$$

$$u^k = \Delta\gamma \quad (17)$$

sendo:

k : o índice que indica a iteração corrente.

O processo deverá, então, ser repetido até que o treinamento seja concluído. Trata-se de um procedimento bastante simples cujo sistema de controle requer um esforço adicional bastante reduzido, tendo em vista que o controlador possui duas variáveis de entrada e uma única de saída. O valor u é calculado pelo método da “denebulização” (Kartalopoulos, 1996; Terano *et al.*, 1991).

A denebulização é um processo em que as funções de pertinência são amostradas para encontrar seu grau de pertinência e, então, este grau é usado na equação lógica para definir uma região de saída; a partir disto deduz-se a saída. Dos três métodos de denebulização existentes (Kartalopoulos, 1996) escolheu-se o do centroide que encontra o centro de massa da variável de saída, isto é, da figura definida pelo conjunto de regras sobre o eixo da variável de controle.

Cada variável de estado deverá ser representada entre 3 e 7 conjuntos nebulosos. A variável de controle também deverá ser representada com o mesmo número de conjuntos nebulosos.

A variável εg deverá ser normalizada, considerando-se como fator de escala o primeiro erro global gerado pela rede, ou seja, de índice $k = 0$. Com esta representação, o intervalo de variação de εg deverá estar compreendido entre 0 e +1. Se a heurística de adaptação estiver devidamente sintonizada, a convergência do processo deverá ser exponencial decrescente. A variável $\Delta \varepsilon g$ deverá variar entre -1 e +1. Se o processo de convergência for exponencial decrescente os valores de $\Delta \varepsilon g$ deverão ser sempre negativos.

Neste caso, embora a escala de $\Delta \varepsilon g$ varie entre -1 e +1, deve-se empregar, no conjunto de regras, um ajuste fino entre -1 e 0. No outro intervalo, o ajuste poderá ser mais relaxado. Visando estabelecer prioridade de atuação do regulador no intervalo inicial do treinamento, emprega-se uma função exponencial decrescente para corrigir os acréscimos Δy e evitar instabilidade no processo de convergência.

1.1.3 Treinamento por Levenberg-Marquardt

A rede neural possui a capacidade de aprender baseada na experiência. Para tanto, deve ser treinada, ajustando, assim, os seus pesos. Para tal, necessita-se de um algoritmo de treinamento. Uma alternativa ao treinamento retropropagação tradicional (Werbos, 1974) é o algoritmo de Levenberg-Marquardt, que é um método gradiente de segunda ordem, que se baseia, para a aceleração do treinamento, na determinação das derivadas de segunda ordem do erro quadrático em relação aos pesos, diferindo do algoritmo “retropropagação” (Werbos, 1974) tradicional que considera as derivadas de primeira ordem. Esse algoritmo é considerado o método mais rápido para treinamento de redes, que possui uma quantidade moderada de pesos sinápticos.

O método de Levenberg - Marquardt faz a aproximação para a matriz mostrada em (18), determinada em função da matriz Jacobiana, que contém as primeiras derivadas dos pesos em função dos pesos sinápticos, expressa em (19):

$$H = \frac{\partial^2(E_R(W))}{\partial W^2} \quad (18)$$

em que

$$E_R(W) = \frac{1}{n} \sum_{i=1}^2 (y_i - y_{ei})^2$$
$$J = \frac{\partial e(W)}{\partial W} \quad (19)$$

Em que $e(W)$ é definido conforme a expressão (20):

$$e(W) = \sum_{i=1}^n (y_i - y_{ei}) \quad (20)$$

A determinação da matriz Jacobiana é muito mais simples que a determinação da matriz Hessiana. Como, para uma rede neural, o desempenho de treinamento é expresso em função da soma dos erros quadráticos, a matriz Hessiana pode ser expressa pela expressão (21), (Barbosa, 2005):

$$H = J^T(W) J(W) \quad (21)$$

O método de Newton atualiza os pesos segundo (22):

$$W(k+1) = W(k) - H^{-1} g_k \quad (22)$$

Em que g_k pode ser escrito conforme (23):

$$g_k = 2J^T(W) e(W) \quad (23)$$

O algoritmo de Levenberg-Marquardt procede à atualização dos pesos baseado na expressão do método de Newton (22), realizando as modificações para a determinação da matriz Hessiana, mostrada em (24):

$$W(k+1) = W(k) - [J^T(W)J(W) + \mu_k I]^{-1} J^T(W) e(W) \quad (24)$$

Em que: I é a matriz identidade e μ_k é a constante do método de Levenberg-Marquardt.

O parâmetro μ_k funciona como um fator de estabilização do treinamento, ajustando a aproximação de forma a utilizar a rápida convergência do método de Newton e evitando passos muito grandes que possam levar a um erro de convergência, (Barbosa, 2005).

Esse método apresenta convergência em menos iterações, mas requer mais cálculos por iteração por causa do cálculo de matrizes inversas. Apesar do grande esforço computacional, ele segue sendo o algoritmo de treinamento mais rápido para redes neurais baseadas em derivadas, quando se trabalha com um número moderado de parâmetros na rede. Se esse número é elevado, a utilização desse algoritmo não é prática (Ranganathan, 2004).

1.1.4 Modificação da inclinação e translação da função sigmoide

Consiste na realização do ajuste, de forma sistemática, da inclinação (parâmetro λ) e de translação (parâmetro ρ) da função sigmoide como proposto na referência (Stamatis *et al.*, 1999, Lopes, 2003b). É um procedimento em que os parâmetros λ e ρ são adaptados de forma análoga ao que é realizado com relação aos pesos da rede neural, como abordado anteriormente. Neste caso, a regra de adaptação usada é a retropropagação. Assim, não é introduzida nenhuma complicação adicional, exigindo apenas um número maior de parâmetros a serem adaptados. O resultado obtido, contudo é bastante significativo, ou seja, o ajuste da inclinação da função sigmoide permite que se controle o espaço de busca de soluções o que efetivamente representa uma melhor perspectiva de se evitar a paralisia do processo de adaptação. A translação da função sigmoide, também, é uma eficiente alternativa que visa contribuir para que haja maior garantia que o processo adaptativo convirja para um ponto mínimo global. Deve-se ressaltar que o efeito de translação é mais eficiente em comparação ao controle da inclinação sigmoide. Isto decorre pelo fato de que a tendência de melhoria da resposta, via o controle de inclinação, é fazer com que o parâmetro λ seja reduzido para um valor mínimo que é bastante limitado, ou seja, λ não deve ser menor, a princípio, de 0,4 conforme indicação colhida nas várias experiências realizadas. No caso de translação, não há limite imposto. Deve-se ressaltar, ainda, que o algoritmo retropropagação realiza esta sintonia de forma automática.

Via de regra, tem-se observado que quase sempre se consegue realizar o treinamento (completada a convergência) usando os dispositivos ora abordados: controlador nebuloso para o ajuste da taxa de treinamento e a adaptação da inclinação e translação da função sigmoide. Porém, como esta metodologia é determinística não se pode garantir sempre que ocorra a convergência. O que efetivamente pode-se dizer, é que se trata de um procedimento que tem proporcionado bons resultados, destacando-se o caso da previsão de carga de sistemas de energia elétrica por redes neurais:

- (1) Na referência (Lopes et al., 2003a) foi empregado o controlador *fuzzy* da taxa de treinamento, cujo resultado proporcionado foi uma redução de em torno de 25% no tempo de treinamento e a redução do erro da previsão (MAPE (*Mean Absolute Percentage Error*)) de 7 para menos de 3% em comparação ao algoritmo retropropagação convencional (Werbos, 1974).
- (2) Na referência (Lopes et al., 2003b) considerando-se o controlador *fuzzy* e a adaptação de λ e ρ , foram obtidos um erro da previsão de mesma magnitude como proporcionada na referência (Lopes et al., 2003a) e uma redução do tempo de treinamento ainda maior, ou seja, o tempo de treinamento foi da ordem de 10 vezes menor, se comparado ao produzido pelo procedimento convencional.

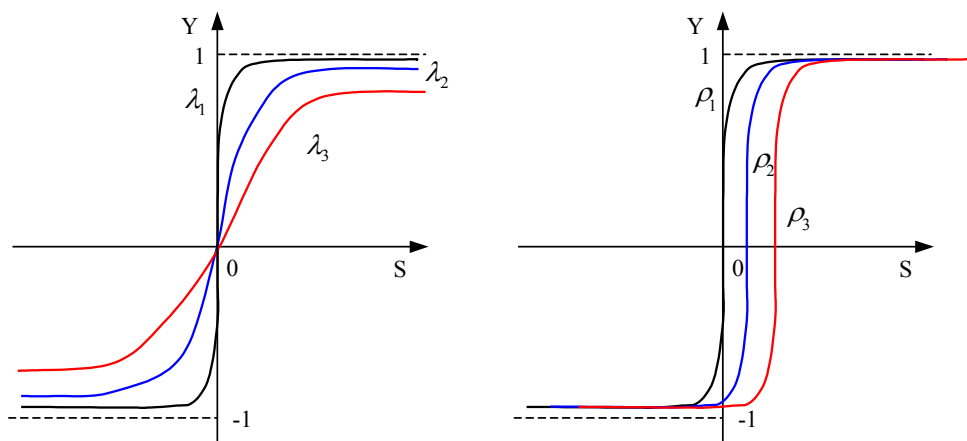


Figura 3. Comportamento da função sigmoide em função do parâmetro λ (inclinação) e ρ (translação).

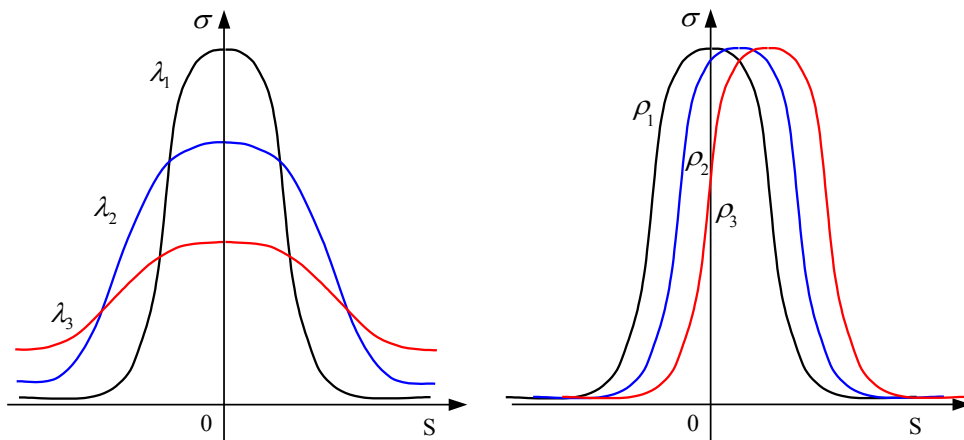


Figura 4. Comportamento da derivada da função sigmoide em relação aos pesos em função do parâmetro λ e ρ (translação).

As Figuras 3 e 4 mostram, respectivamente, os comportamentos da função sigmoide e da função σ (derivada da função sigmoide em relação aos pesos) em função do parâmetro λ : $\lambda_1 > \lambda_2 > \lambda_3 > 0$.

A forma geral das funções pós-sinápticas usadas para ajustar a rede neural é dada por (Stamatis *et al.*, 1999):

$$y_i = f(\lambda, \rho, \mathcal{G}, w) \quad (25)$$

Os parâmetros de inclinação e de translação contêm taxas de aprendizagem correspondentes denotadas por γ^λ e γ^ρ , respectivamente.

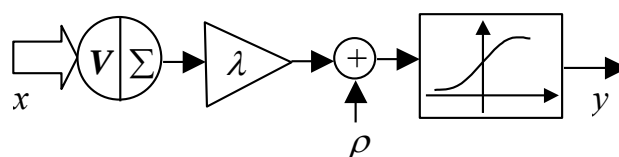


Figura 5. Arquitetura neural para ajustes dos parâmetros de inclinação e translação da função pós-sináptica.

Na Figura 5 ilustra-se a arquitetura neural com ajustes dos parâmetros de inclinação e da translação da função pós-sináptica, neste caso, a função sigmoide (Stamatis *et al.*,1999).

O ajuste do parâmetro de inclinação da função sigmoide é dado por (Lopes *et al.* 2003b):

$$\lambda_i(r+1) = \lambda_i(r) + \theta_i^\lambda(r) \quad (26)$$

em que:

$$\theta_i^\lambda(r) = -\gamma^\lambda [\nabla_i^\lambda(r)];$$

$\nabla_i^\lambda(r)$: gradiente do erro quadrático relacionado a inclinação do neurônio i ;

λ_i : vetor contendo as inclinações do neurônio i ;

$$= [\lambda_{0i} \ \lambda_{1i} \ \lambda_{2i} \ \dots \ \lambda_{mi}]^T.$$

O gradiente ∇_i^λ é representado por (Lopes, 2003b):

$$\nabla_i^\lambda(r) = \frac{\partial \varepsilon_i^2}{\partial \lambda_i(r)} = 2\varepsilon_i \frac{\partial \varepsilon_i}{\partial \lambda_i} = -2\varepsilon_i \frac{\partial y_i}{\partial \lambda_i} = -2\varepsilon_i \sigma_i^\lambda$$

em que:

$\sigma_i^\lambda \triangleq$ derivada da função sigmoide relacionada a λ_i .

A regra geral, que define a adaptação do parâmetro de inclinação da função sigmoide, é dada da seguinte forma:

$$\lambda_i(r+1) = \lambda_i(r) + 2\gamma^\lambda \beta_i^\lambda$$

Regra que define o cálculo de β_i^λ :

$$\text{Última camada} \quad : \quad \beta_i^\lambda = \sigma_i^\lambda \varepsilon_i$$

$$\text{Demais camadas} \quad : \quad \beta_i^\lambda = \sigma_i^\lambda \sum_{j \in Q(i)} w_{ij} \beta_j^\lambda.$$

De forma similar ao procedimento descrito supracitado, o ajuste do parâmetro translação da função sigmoide é indicado por:

$$\rho_i(r+1) = \rho_i(r) + \theta_i^\rho(r) \quad (27)$$

em que:

$$\theta_i^\rho(r) = -\gamma^\rho [\nabla_i^\rho(r)];$$

$\nabla_i^\rho(r)$: gradiente do erro quadrático referente ao deslocamento do neurônio i ;

ρ_i : vetor contendo os deslocamentos do neurônio i ;

$$= [\rho_{0i} \ \rho_{1i} \ \rho_{2i} \ \dots \ \rho_{ni}]^T.$$

E o gradiente ∇_i^ρ é calculado da seguinte maneira:

$$\nabla_i^\rho(r) = \frac{\partial \varepsilon_i^2}{\partial \rho_i(r)} = 2\varepsilon_i \frac{\partial \varepsilon_i}{\partial \rho_i} = -2\varepsilon_i \frac{\partial y_i}{\partial \rho_i} = -2\varepsilon_i \sigma_i^\rho$$

em que:

σ_i^ρ \triangleq derivada da função sigmoide, relacionada a ρ_i .

A regra de adaptação do parâmetro translação da função sigmoide é realizada empregando-se a seguinte equação (Lopes, 2003b):

$$\rho_i(r+1) = \rho_i(r) + 2\gamma^p \beta_i^p$$

Regra que define o cálculo de β_i^p

Última camada : $\beta_i^p = \sigma_i^p \varepsilon_i$

Demais camadas : $\beta_i^p = \sigma_i^p \sum_{j \in Q(i)} w_{ij} \beta_j^p$

A regra de adaptação dos parâmetros de inclinação e de translação da rede é calculada, de forma iterativa, para todo o i -ésimo neurônio.

1.2 REDE NEURAL GRNN

Considerando um conjunto finito de variáveis a GRNN (*General Regression Neural Network*) (Specht et al. 1991) estima uma variável dependente a partir de uma variável independente. Possui treinamento supervisionado, uma estrutura fixa com uma camada de entrada, duas intermediárias e uma de saída. A quantidade de neurônios de cada camada depende da quantidade de padrões de treinamento (Niu et al., 2005).

A variável dependente é y representando a saída e a independente x representando a entrada pelo conjunto finito de variáveis X e Y conforme Figura 6 (Nose Filho 2011, Specht et al., 1991).

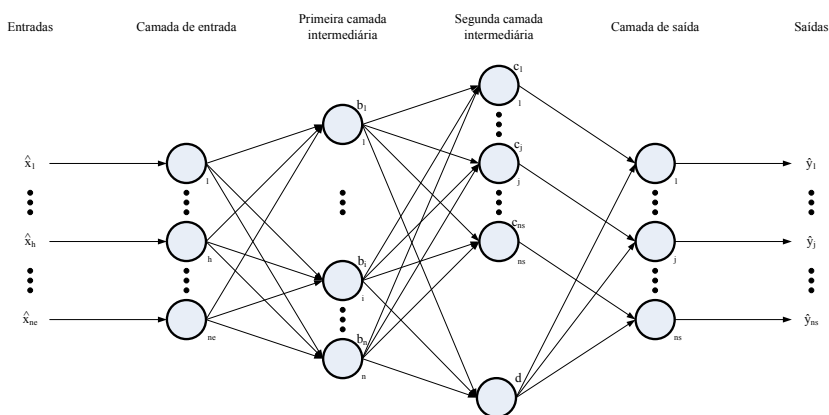


Figura 6 Arquitetura GRNN.

A variável \hat{x} é a entrada desconhecida e \hat{y} é a estimativa pela regressão de \hat{x} considerando os conjuntos X e Y , $f(\hat{x}, Y)$ é a função densidade de probabilidade que corresponde a estimação de Kernel ou janela de Parzen conforme equação (28) (Nose Filho 2011, Specht et al. 1991):

$$E(Y | \hat{x}) = \frac{\int_{-\infty}^{+\infty} y f(\hat{x}, Y) dy}{\int_{-\infty}^{+\infty} f(\hat{x}, Y) dy} \quad (28)$$

As aferições da variável \hat{y} são obtidas por meio de uma variável desconhecida (Specht et al. 1991), conforme equação (29):

$$\hat{y}(\hat{x}) = \frac{\sum_{i=1}^n y_i \exp\left(\frac{-a_i^2}{2\sigma^2}\right)}{\sum_{i=1}^n \exp\left(\frac{-a_i^2}{2\sigma^2}\right)} \quad (29)$$

sendo:

n : número de amostras dos conjuntos X e Y ;

y_i : representa a variável dependente associada com x_i

a_i : representa a i -ésima distância Euclidiana de \hat{x} juntamente com x_i (Specht et al. 1991), conforme equação (30):

$$a_i = (\hat{x} - x_i)^T (\hat{x} - x_i) \quad (30)$$

A função de transferência é do tipo Gaussiana, a qual é determinada por:

$$f(a_i) = \exp\left(\frac{-a_i^2}{2\sigma^2}\right) \quad (31)$$

A variável (σ) representa o *spread*, que define a abertura da função. A toolbox de redes neurais do MATLAB considera a função Gaussiana (*radbas*), (Nose Filho, 2011), representada por:

$$f(a_i) = \exp\left(\frac{-0.8326^2 a_i^2}{\sigma^2}\right) \quad (32)$$

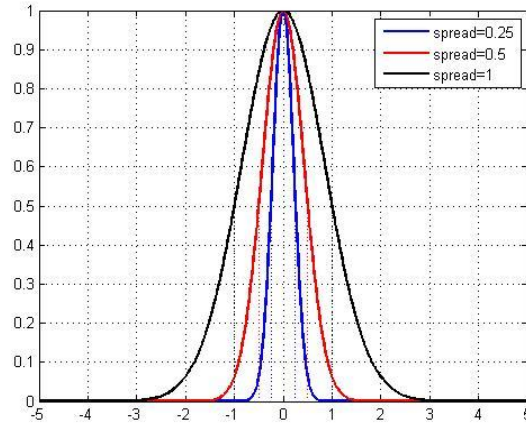


Figura 7- Função *radbas* para diferentes valores do parâmetro *spread*.

O valor do parâmetro *spread* (Figura 7) define a área de vizinhança a ser levada em consideração para a estimação de uma variável. Quanto maior for este valor, maior será a área de vizinhança, e mais padrões serão levados em consideração para a estimação da variável. Quanto menor for este valor, menor será a área de vizinhança e menos padrões serão levados em consideração para a estimação da variável.

A GRNN possui uma estrutura fixa, disposta em quatro camadas, uma camada de entrada, duas camadas intermediárias e uma camada de saída, conforme apresentado na Figura 6.

O número de neurônios de cada camada depende exclusivamente do número de entradas, do número de saídas e do número de amostras do conjunto de dados de treinamento.

O padrão de entrada da rede é representado por um vetor coluna \hat{X} , ne -dimensional e o padrão de saída é representado por um vetor coluna \hat{Y} , ns -dimensional. O conjunto de dados de treinamento é representado por duas

matrizes: (1) uma matriz $X_{n \times n}$ contendo os padrões de entrada e (2) outra matriz $Y_{ns \times n}$ contendo os padrões de saída. O número n de colunas de cada matriz representa a quantidade de amostras do conjunto de dados de treinamento.

Treinamento:

1. Leitura das matrizes X e Y e do parâmetro *spread*.

Diagnóstico/Previsão:

1. Leitura do padrão desconhecido \hat{X} ;
2. Cálculo das distâncias euclidianas, a_i dadas por (33);
3. Cálculo dos níveis de pertinência, b_i dados por (34);
4. Cálculo das saídas dos neurônios do tipo numerador, c_j dadas por (35), e da saída do neurônio do tipo denominador, d dada por (36);
5. Cálculo das saídas estimadas, \hat{y}_j dadas por (37);
6. A saída \hat{Y} é dada por (38).

$$a_i = (\hat{X} - X_i)^T (\hat{X} - X_i), i = 1, \dots, n \quad (33)$$

$$b_i = \exp\left(\frac{-0.8326^2 a_i^2}{\sigma^2}\right), i = 1, \dots, n \quad (34)$$

$$c_j = \sum_{i=1}^n y_{j,i} b_i, j = 1, \dots, ns \quad (35)$$

$$d = \sum_{i=1}^n b_i \quad (36)$$

$$\hat{y}_j = \frac{c_j}{d}, j = 1, \dots, ns \quad (37)$$

$$\hat{Y} = [\hat{y}_1 \dots \hat{y}_j \dots \hat{y}_{ns}]^T \quad (38)$$

1.3 Redes Neuro-Fuzzy

O sistema Neuro-Fuzzy representa a fusão de duas ferramentas já conhecidas: Redes Neurais Artificiais e a Lógica Nebulosa (Fuzzy). Nesse sistema

são agregadas as características de transparência de raciocínio da lógica nebulosa e a capacidade de aprendizado e generalização das Redes Neurais. O sistema *Neuro-Fuzzy* é uma grande ferramenta que está sendo bastante aplicada para a solução de diversas situações problemas, principalmente na área da Engenharia. Nesse contexto, pode-se afirmar que uma rede *Neuro-Fuzzy* pode ser definida como um Sistema *Fuzzy* que é treinado por um algoritmo servido de uma rede Neural.

1.3.1 ANFIS

Uma rede adaptativa baseada em um sistema de inferência Fuzzy baseado em rede adaptativa ANFIS (Jang, 1993) (*Adaptive-Network-Based Fuzzy Inference System*) é um sistema *Neuro-Fuzzy* que identifica parâmetros usando métodos de aprendizado supervisionado. Este tipo de arquitetura serve como base para a construção de um conjunto nebuloso de regras “SE-ENTÃO” com funções de pertinências apropriadas, para a geração de pares de entrada-saída (Jang, 1993).

O desenvolvimento da rede neural ANFIS por Jang, em 1993, foi motivado pela necessidade de um método objetivo para inserir o conhecimento de um especialista dentro de uma base de regras de um sistema de inferência nebuloso, pela necessidade de ajustar as funções de pertinência com a finalidade de minimizar o erro na resposta do sistema de inferência nebuloso. O sistema de inferência nebuloso é organizado na forma de uma rede adaptativa, que possui a capacidade de fazer o mapeamento de entrada e saída baseado no conhecimento de um especialista e em um conjunto de dados formado por pares entrada/saída, além de ser capaz de aproximar qualquer função.

A rede neural ANFIS pode ser vista como a representação em rede de um sistema nebulosos do tipo Sugeno, que possui a capacidade de aprendizado. A arquitetura heterogênea é constituída por um número de camadas de nós, os

quais têm a mesma função para uma dada camada, mas são diferentes de uma para outra.

A rede neural ANFIS pode ser vista, ainda, como uma rede de seis camadas, interligadas por meio de pesos unitários, onde cada camada é responsável por uma operação que resultará em uma saída equivalente à encontrada em uma determinada etapa de um sistema nebuloso do tipo Takagi-Sugeno (Jang, 1993; Jang & Sun, 1995; Jang et al., 1997).

Takagi e Sugeno (Takagi et. al, 1985) foram um dos primeiros pesquisadores que propuseram as estruturas de sistemas de inferência nebulosa, que foram baseadas em um conjunto de padrões de ajuste para a sua definição. Sugeno propõe regras do tipo se x é A e y é B , então, $z = f(x,y)$, sendo x a variável linguística de entrada, y a variável linguística de saída e $f(x,y)$ são funções lineares ou não-lineares das variáveis numéricas de entrada.

Modelo Sugeno de primeira ordem:

$$\text{se } x \text{ é } A_1 \text{ e } y \text{ é } B_1, \text{ então, } z_1 = p_1 x + q_1 y + r_1 \quad (39)$$

$$\text{se } x \text{ é } A_2 \text{ e } y \text{ é } B_2, \text{ então, } z_2 = p_2 x + q_2 y + r_2 \quad (40)$$

Na Figura 8 mostra-se o sistema de inferência nebulosos de Takagi e Sugeno de primeira ordem.

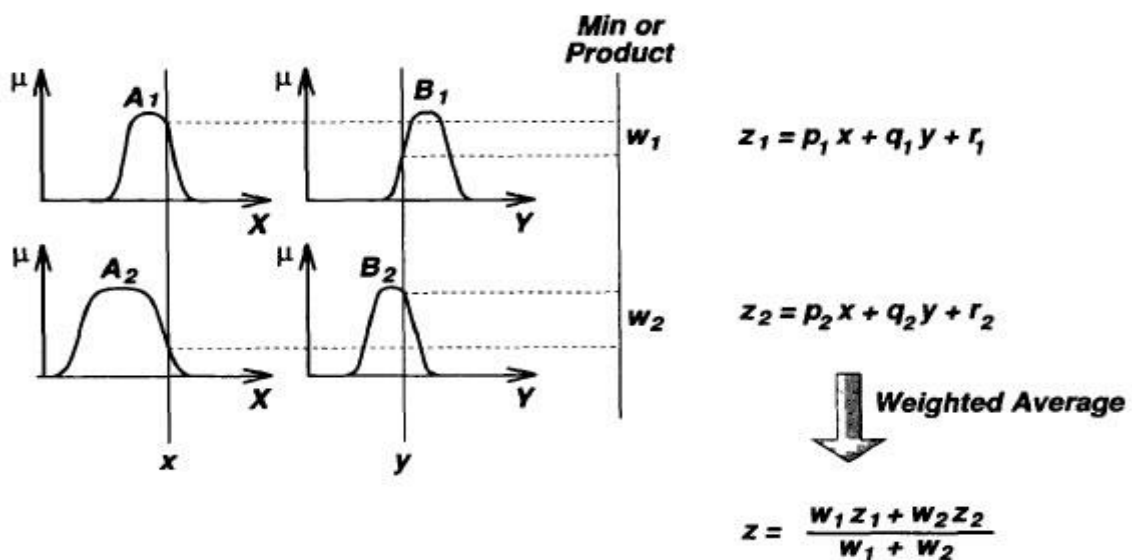


Figura 8 - Sistema de inferência nebulosos de Takagi e Sugeno de primeira ordem.

(Fonte: Jang (1997)).

Os sistemas de inferência nebulosos são muito utilizados para modelar processos complexos, com base em um conjunto regras, do tipo “se... então”, que quando combinadas com métodos de redes neurais artificiais originam um novo método chamado de híbrido. Este método é capaz de vincular a capacidade de aprendizado das redes neurais artificiais com o poder de interpretação linguístico dos sistemas de inferência nebulosos. Dessa maneira, esse sistema híbrido é capaz de resolver a sintonia entre as funções de entrada e de saída, que por sua vez, é um problema que a lógica nebulosa apresenta.

Esta estrutura implementa sistemas do tipo Takagi-Sugeno (Takagi & Sugeno, 1985), com funções lineares ou constantes nos consequentes das regras que formam o sistema, tendo estas regras pesos unitários. Na Figura 9 é apresentada a arquitetura da ANFIS.

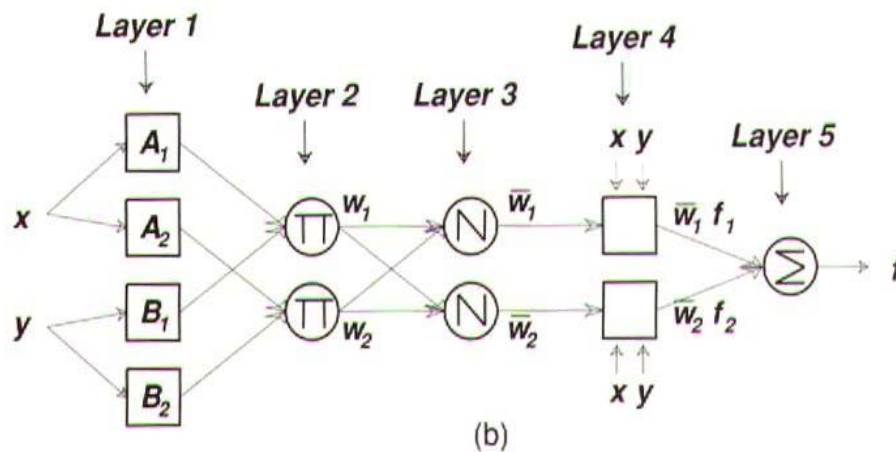


Figura 9 - Estrutura da rede ANFIS. (Fonte: Jang (1993)).

Na primeira camada (camada 1) ocorre a fuzzificação, onde os dados de entrada são codificados. Ela é composta pelo grau de pertinência do padrão de entrada, onde é calculado o grau de pertinência com que as entradas precisas que

satisfazem os termos ($A_1 = \text{Alto}$ e $B_1 = \text{Baixo}$) associados a estes nós. Portanto, os neurônios dessa camada possuem como função de ativação funções de pertinência. Nesta camada os parâmetros dessas funções serão ajustados por um método de otimização ou algoritmo de treinamento escolhido.

Na segunda camada (camada 2) é realizado o cálculo da pertinência do antecedente, ou seja, é realizada a operação de interseção entre as funções de pertinência de entrada, de acordo com o operador escolhido. Nessa camada os neurônios possuem funções de ativação fixas.

Na terceira camada (camada 3) é realizado o cálculo do valor normalizado do grau de ativação da regra. Os neurônios são fixos e cada neurônio está associado a uma regra SE-ENTÃO. A função de ativação dos neurônios é dada pela Equação (41):

$$f(W) = \frac{w_i}{\sum_{j=1}^m w_j} \quad (41)$$

Sendo que w_i é a saída do neurônio da camada anterior e w_j corresponde a soma dos níveis de disparo de todas as regras.

Na quarta camada (camada 4), os neurônios possuem como função de ativação as funções "Sugeno", que são combinações de ordem p dos sinais de entrada. Esta essa camada é responsável pelo cálculo dos valores dos consequentes das regras.

Na quinta camada (camada 5) ocorre a defuzzificação, ou seja, onde é realizado o somatório das saídas da camada anterior, apresentando então a saída da rede neural ANFIS. Nesta camada há um único neurônio.

A rede neural ANFIS apresenta uma grande vantagem em relação às redes neurais artificiais no que se refere em relação a codificação do conhecimento. O conhecimento é codificado em uma estrutura que possui certa aproximação da lógica utilizada por humanos enquanto que nas Redes Neurais Artificiais o conhecimento é codificado em pesos.

Modelagem do Sistema *Fuzzy*

A construção de um modelo nebuloso pode ser realizada em duas etapas que não são totalmente independentes. A primeira etapa deve ser realizada a identificação da estrutura, ou seja, devem ser selecionadas as variáveis de entrada e de saída relevantes, o tipo do sistema de inferência nebuloso (“Sugeno”, por exemplo), o número de termos linguísticos (como exemplo o BAIXO, MÉDIO, ALTO), e a elaboração do conjunto de regras nebulosas SE-ENTÃO. Estas etapas são executadas a partir do conhecimento do especialista.

Após estas etapas, deve ser realizado o ajuste dos parâmetros, ou seja, realizar a escolha de uma função de pertinência (por exemplo: trapezoidal, gaussiana, função seno etc.), definir os parâmetros das funções de pertinência, considerando o conhecimento do especialista e fazer o ajuste dos parâmetros das funções de pertinência para melhorar o desempenho do modelo nebuloso.

Aprendizado

O algoritmo de aprendizado da rede neural ANFIS é um método híbrido supervisionado baseado no método do gradiente e no método de mínimos quadrados.

Na fase “forward” os sinais são propagados até a camada 4 e os parâmetros de relevância são atualizados por quadrados mínimos. Nessa fase, os parâmetros não-lineares são mantidos fixos; os dados de entrada alimentam a rede e são propagados para frente até a que as saídas dos nós da camada 3 sejam obtidas. Em seguida, os parâmetros lineares são identificados por quadrados mínimos, a saída da rede é obtida e o erro é calculado.

Já, na fase “backward”, os sinais de erro são retropropagados e os parâmetros são atualizados. Nessa fase os parâmetros lineares são mantidos fixos. O sinal de erro é propagado na direção da camada de entrada e o gradiente descendente é usado para ajustar os parâmetros não-lineares.

1.4 ARTMAP-*Fuzzy*

A rede neural artificial ARTMAP-*Fuzzy* é baseada na teoria da ressonância adaptativa, e faz parte da família ART (*Adaptive Resonance Theory*) (Carpenter et al., 1992). A aprendizagem é supervisionada, sendo capaz de auto-organizar o reconhecimento estável de categorias em resposta a sequências arbitrárias de padrões de entradas. Esta rede é constituída por duas redes ART, ART_a-*Fuzzy* e ART_b-*Fuzzy*, conectadas por meio de um módulo de memória associativa inter-ART. Cada módulo ART é composto por três camadas: sendo F_0 a camada de entrada, F_1 a camada de comparação, e F_2 a camada de reconhecimento, armazenando as categorias. As camadas F_1 e F_2 são conectadas pelos pesos (w_j^a no módulo ART_a e w_k^b no ART_b). A rede ART_a-*Fuzzy* recebe os padrões de entrada e a ART_b-*Fuzzy* apresenta a resposta desejada para o padrão de entrada que são interligadas pelo módulo de memória associativa Inter-ART, verificando o casamento das entradas e saídas com as categorias existentes (Carpenter e Grosberg, 1987, 1991, 1992).

Os parâmetros da rede neural ARTMAP-*Fuzzy* são: Parâmetro de escolha α ($\alpha > 0$): responsável pela seleção de categorias; Taxa de aprendizagem β ($\beta \in [0,1]$): controla a velocidade de aprendizagem da rede; Parâmetro de vigilância do módulo ART_a, ART_b e inter-ART (ρ_a , ρ_b e $\rho_{ab} \in [0,1]$): parâmetro responsável pelo número de categorias criadas, ou seja, controla a ressonância da rede.

A seguir apresentam-se os passos de funcionamento da rede ARTMAP-*Fuzzy*:

1º) Leitura dos padrões de entrada a e b , os quais são normalizados para evitar a proliferação de categorias conforme equação (42);

$$\bar{a} = \frac{a}{|a|} \quad \text{e} \quad \bar{b} = \frac{b}{|b|} \quad (42)$$

2º) Codificação do vetor de entrada e saída;

$$I = [\bar{a} \bar{a}^c] \quad \text{e} \quad I = [\bar{b} \bar{b}^c] \quad (43)$$

3º) Escolha da categoria ganhadora na camada F_2 em ART_a e ART_b por meio da equação (44):

$$T_j^a(I_a) = \frac{|I_a \wedge w_j^a|}{\alpha + |w_j^a|} \quad \text{e} \quad T_k^b(I_b) = \frac{|I_b \wedge w_k^b|}{\alpha + |w_k^b|} \quad (44)$$

A categoria J com $T_j = \max\{T_j\}$ e índice J é a vencedora em ART_a , e em ART_b , a vencedora é a categoria K com $T_k = \max\{T_k\}$ e índice K mínimo;

4º) Teste de vigilância do módulo ART_b de acordo com (45):

$$\frac{|I_b \wedge w_K^b|}{I_b} \geq \rho_b \quad (45)$$

A equação (45) uma vez satisfeita, deverá acontecer a ressonância e os pesos são atualizados usando-se a equação (46). Do contrário, ocorrerá o *reset*, e o neurônio K é excluído do processo de busca e determina-se um novo neurônio vencedor. Ocorrendo a rejeição de todos os neurônios um novo neurônio é criado pela rede.

$$w_K^{novo} = \beta(I_b \wedge w_K^{velho}) + (1 - \beta)w_K^{velho} \quad (46)$$

5º) Teste de vigilância do módulo ART_a , conforme equação (47);

$$\frac{|I_a \wedge w_j^a|}{I_a} \geq \rho_a \quad (47)$$

Satisfeita a equação (8), ocorre a ressonância para o ART_b e os neurônios vencedores K e J são verificados pelo *match tracking* pela equação (48):

$$\frac{|y_j^b \wedge w_{jK}^{ab}|}{|y_j|} \geq \rho_{ab} \quad (48)$$

Ocorrendo a ressonância no dispositivo *match tracking* o índice J vencedor é ativado e confirmado os índices J e K atualizando os pesos de ART_a e Inter- ART pelas equações (49) e (50), respectivamente.

$$w_j^{novo} = \beta(I_a \wedge w_j^{velho}) + (1 - \beta)w_j^{velho} \quad (49)$$

$$w_{JK}^{ab} = 1$$

$$w_{jk}^{ab} = 0, \quad \text{se } j \neq J ; k \neq K \quad (50)$$

Não ocorrendo a ressonância, a categoria atual é descartada e um acréscimo ε é introduzido na equação (47) (parâmetro de vigilância) para que a categoria que não casou com a categoria de ART_b seja excluída de vez.

$$\rho_a = \frac{|I_a \wedge w_j^a|}{I_a} + \varepsilon \quad (51)$$

Na Figura 10 ilustra-se a arquitetura da rede neural ARTMAP-Fuzzy.

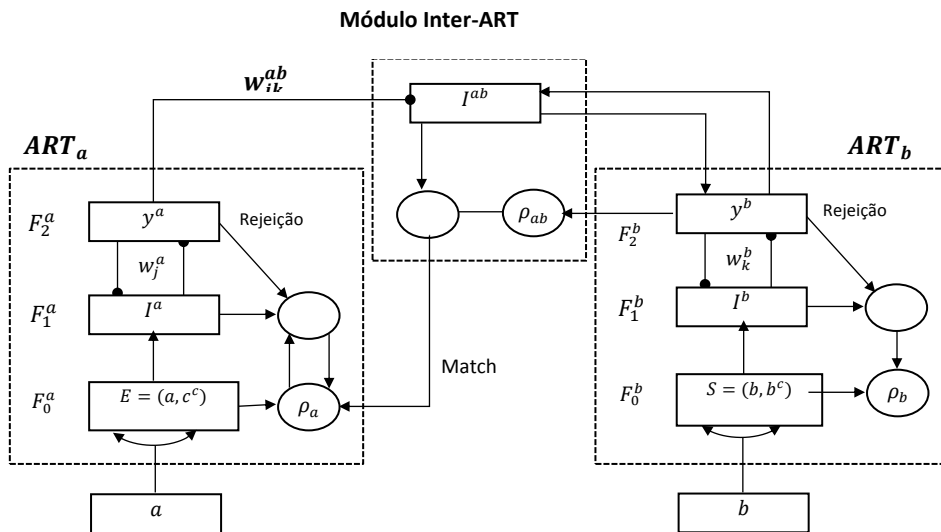


Figura 10: Estrutura da rede neural ARTMAP-*Nebulosa*

2. SISTEMAS IMUNOLÓGICOS ARTIFICIAIS

2.1 Algoritmo de Seleção Negativa

O Algoritmo de Seleção Negativa (ASN) foi proposto pela primeira vez por Forrest et al. (1994) para detecção de anomalias, com aplicações em proteção computacional, baseado na seleção negativa de linfócitos T dentro do timo, sendo uma técnica que se baseia no processo de reconhecimento de padrões exercido pelo SIB (Sistema Imunológico Biológico) elaborado como um modelo computacional. Portanto, este é o processo que representa a discriminação que o organismo realiza com as células, entre próprias e não-próprias. O algoritmo é executado em duas fases, como a seguir (Castro, 2001):

- *Censoriamento*
 - Definir o conjunto de cadeias próprias (S) que se deseja proteger;
 - Gerar cadeias aleatórias e avaliar a afinidade (*match*) entre cada uma delas e as cadeias próprias. Caso a afinidade seja maior que a de um determinado limiar, rejeitar a cadeia. Caso contrário, armazene-a em um conjunto de detectores (R).

No censoriamento do ASN, primeiramente são definidos os detectores próprios, que representam uma condição normal do organismo, sendo conhecidos como cadeias próprias (S), com objetivo de gerar um conjunto de padrões de detectores (R), que tenham a capacidade de reconhecer os padrões não-próprios na fase de monitoramento dos dados. Posteriormente, é feita a leitura dos dados e, então, escolhem-se as cadeias de modo aleatório e verifica-se a afinidade comparando estas cadeias ao conjunto de cadeias próprias (S). O funcionamento desta fase pode ser explicado da seguinte maneira: Se a afinidade é superior a um determinado limiar, rejeita-se a cadeia, senão esta cadeia é aceita no conjunto de detectores (R), e será utilizada para fazer as classificações durante o monitoramento dos dados, sendo assim dotados com a capacidade de detectar

praticamente qualquer elemento não-próprio, uma modificação ou erro nos dados que se deseja monitorar, ou seja, os detectores são análogos às células do tipo T maturadas capazes de reconhecer agentes patogênicos (Lima, 2013).

- *Monitoramento*

- Dado o conjunto de cadeias que se deseja proteger (cadeias protegidas), avaliar a afinidade entre cada uma delas e o conjunto de detectores. Se a afinidade for maior que a de um determinado limiar, então é identificado um elemento não-próprio.

Portanto, nesta fase é realizado um monitoramento nos dados para identificar mudanças no comportamento das amostras. Posteriormente é feita a classificação das mudanças utilizando o conjunto de detectores criados na fase de sensoriamento. Assim, analisando-se as cadeias protegidas (S) e comparando-as com o conjunto de detectores (R), avalia-se a afinidade entre cada uma das cadeias. Por fim, tem-se a resposta para o monitoramento onde se a afinidade entre as cadeias for maior que a um determinado limiar, então, um elemento não-próprio é detectado e classificado (Lima, 2013).

A representação no espaço binário de Hamming, e a regra dos r -bits consecutivos como medida de afinidade é utilizada por Forrest et al. (1994), e ainda em seu trabalho são apresentadas equações estimando a probabilidade de ocorrência de uma ligação em pelo menos r -bits consecutivos entre duas cadeias binárias aleatórias.

A seguir, são apresentados os fluxogramas da fase de sensoriamento e monitoramento do algoritmo de seleção negativa, respectivamente.

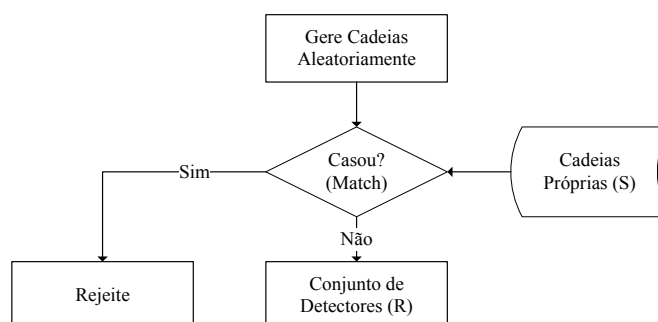


Figura 11 - Fluxograma da fase de sensoriamento do ASN.

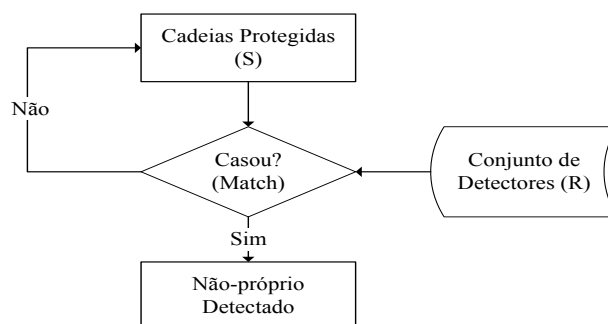


Figura 12 - Fluxograma da fase de monitoramento do ASN.

Note que este algoritmo é capaz de executar tarefas como reconhecimento de padrões armazenando informações sobre o conjunto complementar (não-próprio) ao conjunto dos padrões que se deseja proteger (próprio), e ainda vale lembrar que toda a fase de sensoriamento e de monitoramento são realizadas de modo *off-line* e em tempo real (Castro, 2001; Timmis, 2002).

Concluindo, pode-se levantar mais algumas observações positiva do ASN destacadas anteriormente em vários trabalhos (Dasgupta, 2006; Castro, 2001; Forrest et al., 1997):

- Não necessariamente haverá o crescimento do tamanho do conjunto de detectores com o número de cadeias a serem protegidas (Castro, 2001);
- Para a regra dos r -bits consecutivos, o ASN é ajustável proporcionalmente à probabilidade de detecção de uma anomalia;
- A quantidade de algoritmos de detecção independente aumenta exponencialmente a probabilidade de detecção de anomalias (Castro, 2001; Dasgupta, 2006);
- Pode-se dizer que a detecção é simétrica, pois os padrões que se assemelham ou tenham simetria entre si, são considerados casados, então a detecção depende da simetria entre as cadeias em análise (Castro, 2001; Forrest et al., 1997);
- Há um custo exponencial de geração de detectores em relação ao tamanho do conjunto de cadeias a serem protegidas (próprio), isto porque a geração

aleatória das cadeias do conjunto R leva à geração repetida de diversas cadeias (Castro, 2001).

Critério de Casamento (Combinação) e Afinidade

Será utilizado um critério denominado por “casamento” ou também conhecido como combinação, visando avaliar a afinidade entre as cadeias de antígeno e anticorpo de forma a comprovar que elas são semelhantes. A utilização do método de critério de casamento e afinidade tem como característica positiva que não será necessário transformar o problema em representação binária. Assim, utilizam-se os valores nominais obtendo uma redução do tempo da resposta (Lima, 2013).

O casamento pode ser dividido em perfeito ou parcial, como segue (Lima, 2013):

- Perfeito - É necessário que as duas cadeias que estão sendo analisadas sejam perfeitamente iguais, ou seja, as duas cadeias, devem possuir os mesmos valores em todas as posições.
- Parcial - No casamento parcial as duas cadeias que estão sendo analisadas não necessitam serem iguais, ou seja, não existe a necessidade de que todas as posições dos padrões tenham o mesmo valor. Portanto, apenas uma quantidade de posições entre os padrões deve ter o mesmo valor para se confirmar o casamento. Esta quantidade é a denominada taxa de afinidade e é definida previamente.

A taxa de afinidade foi definida a primeira vez por Bradley e Tyrrell (2002), representando o grau de semelhança necessário para ocorrer o casamento entre as duas cadeias em análise e definida pela seguinte relação:

$$TAf = \left(\frac{An}{At}\right) * 100 \quad (52)$$

sendo:

TAf : taxa de afinidade;

An : número de cadeias normais no problema (cadeias próprias);

At : número total de cadeias no problema (cadeias próprias e não-próprias).

Na equação (52) emprega-se uma relação estatística entre todas as amostras do problema, sendo possível a quantificação de forma correta do valor da taxa de afinidade.

Para esta pesquisa, será utilizado o conceito de casamento parcial, proposto por (Bradley; Tyrrell 2002), para se realizar a verificação da combinação entre os sinais. Assim, não será medida a distância entre o antígeno e o anticorpo para quantificar a afinidade entre os dois padrões e sim utilizar-se-á o conceito apresentado a seguir.

Uma tolerância para mais e para menos é proposta, para a qual é possível aceitar a combinação entre os padrões, dando assim mais dinamismo ao diagnóstico juntamente com um desvio vinculado ao anticorpo (padrão detector). Este desvio irá atuar de forma individual para cada posição i do vetor, permitindo verificar o casamento em cada posição, como apresentado na equação (53).

$$\underline{Ab}_i \leq Ag_i \leq \overline{Ab}_i \quad (53)$$

sendo:

\underline{Ag}_i : valor nominal da posição i do antígeno (padrão em análise);

Ab_i : valor nominal da posição i menos o desvio adotado no anticorpo (padrão detector);

\overline{Ab}_i : valor nominal da posição i mais o desvio adotado no anticorpo (padrão detector).

Portanto, é considerado o casamento para esta posição, caso o valor da posição i do antígeno estiver no intervalo expresso em (53). Desta forma, torna-se possível quantificar a afinidade entre os padrões, analisando posição por posição. Assim, pode-se escrever a expressão a seguir, a qual representa a afinidade total entre os dois padrões em análise (Bradley; Tyrrell, 2002):

$$Af_T = \frac{\sum_{i=1}^L P_c}{L} * 100 \quad (54)$$

sendo:

Af_T : % de afinidade entre os padrões analisados;

L : quantidade total de posições;

P_c : posição casada;

$\sum_{i=1}^L P_c$: somatório (quantidade) das posições casadas.

Observando-se a expressão (54), pode-se dizer que, caso Af_T for maior ou igual a TAf , ocorre o casamento entre os dois padrões, ou combinação, ou seja, os dois são considerados semelhantes. Caso contrário, se Af_T for menor que TAf o anticorpo (detector) não reconhece o antígeno, então não existirá casamento, ou combinação, entre os sinais.

2.2 Algoritmo de Seleção Clonal

O algoritmo de seleção clonal, CLONALG (*Clonal Selection Algorithm*), foi proposto originalmente em (Castro 2000). Diferente do que foi proposto em (Forrest et al., 1994), neste algoritmo são consideradas duas características centrais do princípio de seleção clonal, a maturação e a seleção proporcional à afinidade. Duas versões do algoritmo estão disponíveis na literatura, sendo uma para resolver problemas de aprendizagem de máquina e reconhecimento de padrões e outra para resolver problemas de otimização (Castro, 2001).

O algoritmo CLONALG para problemas de reconhecimento de padrões e aprendizagem de máquina pode ser descrito conforme os passos a seguir (Castro, 2001; Castro e Timmis, 2002):

- Passo I: Inicialização: gere uma população ($Ab = Ab_{\{M\}} + Ab_{\{R\}}$) aleatoriamente com N linfócitos para cada antígeno (Ag_i). N é dado por $M+R$;
- Passo II: Avaliação da afinidade: cada antígeno (Ag_i) é apresentado a todos os linfócitos da população (Ab) em um processo de avaliação da afinidade. Um vetor de afinidade f é determinado;
- Passo III: Seleção: os n linfócitos com maior valor de afinidade f em relação a (Ag_i) são selecionados para compor uma subpopulação ($Ab_{\{n\}}$);
- Passo IV: Clonagem: os n linfócitos selecionados irão se proliferar (clonagem) proporcionalmente às suas afinidades ao antígeno (Ag_i), gerando uma população C de clones. Quanto maior a afinidade f , maior o número de clones de cada um dos n linfócitos selecionados;
- Passo V: Hipermutação: em seguida, a população C de clones é submetida ao processo de maturação de afinidade gerando uma nova população C^* , onde cada linfócito irá sofrer uma mutação com uma taxa inversamente proporcional a sua afinidade f ;
- Passo VI: Avaliação da afinidade: determine a afinidade f^* entre o conjunto C^* de clones mutados e o antígeno (Ag_i);
- Passo VII: Re-seleção: da população madura C^* , re-selecione os n melhores linfócitos maturados compondo a subpopulação ($Ab_{\{n\}}$). A partir dessa subpopulação, escolha os melhores linfócitos para entrarem no conjunto de memória ($Ab_{\{M\}}$). Um linfócito entra em no conjunto de memória quando apresenta altas taxas de afinidade, podendo substituir algum linfócito de memória;
- Passo VIII: Metadinâmica: substitua d anticorpos de ($Ab_{\{R\}}$) por ($Ab_{\{d\}}$) novos indivíduos, induzindo diversidade no repertório. Os anticorpos com menores afinidades são escolhidos para serem substituídos.
- Passo IX: Repita os passos de II a VIII até satisfazer o critério de parada.

Ao final do processo iterativo o conjunto de memória ($Ab_{\{M\}}$) possui M linfócitos com altas taxas de afinidade em relação ao antígeno (Ag_i). Este conjunto de memória pode ser utilizado pelo ASN para detectar e classificar o antígeno aprendido no processo de seleção clonal.

Vale ressaltar que para problemas de aprendizado, o passo VIII (metadinâmica) não é executado. Desta forma, o parâmetro $d = 0$.

A quantidade N_c de clones gerada no Passo IV, para cada linfócito i ; é dada pela equação (55) (Castro, 2001):

$$N_c^i = \text{round} \left(\frac{\beta N}{i} \right) \quad (55)$$

em que: β é um fator multiplicativo entre $[0,1]$; N é a quantidade total de linfócitos da população Ab , e $\text{round}(\cdot)$ é o operador de arredondamento para o inteiro mais próximo.

A taxa de mutação (α), de cada clone, é definida pela equação (56) (Castro, 2001):

$$\alpha = \exp(-\rho f^n) \quad (56)$$

em que: ρ é um parâmetro de controle de amortecimento da função exponencial e f^n é o valor normalizado da afinidade f , que pode ser calculado conforme apresentado na equação (57):

$$f^n = \frac{f}{f_{\max}} \quad (57)$$

Desta forma, cada clone sofre um processo de mutação dado por (de França, 2005):

$$m = \text{round}(\alpha * N(0,1)) \quad (58)$$

sendo: m a quantidade de mutações que cada clone sofrerá, $\text{round}(\cdot)$ é o operador de arredondamento para o inteiro mais próximo, a é a taxa de mutação e $N(0,1)$ é uma variável randômica gaussiana de média zero e desvio padrão $\sigma = 1$.

3. LÓGICA FUZZY

Lógica *Fuzzy* é uma forma matemática para representação de definições vagas. Conjuntos *Fuzzy* são generalizações da teoria de conjunto convencional. Contém objetos que contemplam imprecisão no referido conjunto. O grau de pertinência é definido por um valor da função de pertinência a qual possui valores compreendidos entre 0 e +1. Deste modo, a seguir são apresentados os principais conceitos sobre lógica nebulosa que são utilizados nesta pesquisa.

Definição 1. Considere uma coleção de objetos Z . Então, um conjunto nebuloso A em Z é definido como sendo o conjunto de pares ordenados:

$$A = \{(z, \mu_A(z)) \mid z \in Z\} \quad (59)$$

sendo:

$\mu_A(z)$: valor da função de pertinência do conjunto nebuloso A correspondente ao elemento z .

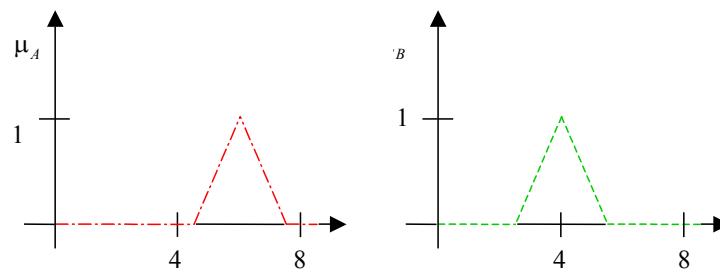


Figura 13 Conjuntos *Fuzzy*.

Operações semelhantes a *AND*, *OR* e *NOT* são alguns dos mais importantes operadores de conjuntos nebulosos. Supondo-se que A e B são dois conjuntos nebulosos com funções de pertinências designadas por $\mu_A(z)$ e $\mu_B(z)$, respectivamente, então, tem-se (Kartalopoulos, 1996, Terano *et al.*, 1991):

- a) *Operador AND ou interseção de dois conjuntos.* A função de pertinência da interseção destes dois conjuntos nebulosos ($C = A \cap B$) é definida por:

$$\mu_C(z) = \min \{ \mu_A(z), \mu_B(z) \}, z \in Z \quad (60)$$

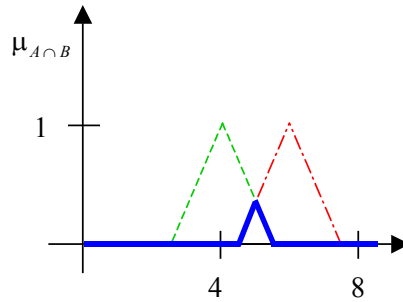


Figura 14. Operador AND.

- b) *Operador OR ou união entre dois conjuntos.* A função de pertinência da união destes conjuntos nebulosos ($D = A \cup B$) é definida por:

$$\mu_D(z) = \max \{ \mu_A(z), \mu_B(z) \}, z \in Z \quad (61)$$

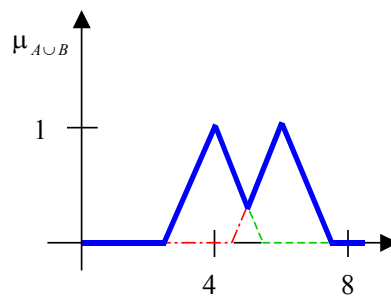


Figura 15. Operador OR.

- c) *Operador NOT ou o complemento de um conjunto nebuloso.* A função de pertinência do complemento de A , A' é definida por:

$$\mu_{A'}(z) = 1 - \mu_A(z), z \in Z. \quad (62)$$

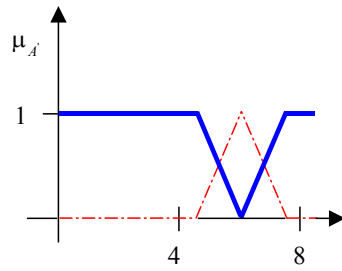


Figura 16. Operador NOT.

- d) *Relação Nebulosa*. A função nebulosa R de A em B pode ser considerada como um grafo *fuzzy*. É caracterizada pela função de pertinência $\mu_R(z,y)$, a qual satisfaz a seguinte regra de composição:

$$\mu_B(z) = \max_{z \in Z} \{ \min [\mu_R(z,y), \mu_A(z)] \} \quad (63)$$

Na Figura 43 mostram-se as características da função de pertinência, na qual a altura ou magnitude é normalizada em 1. O grau de pertinência é determinado pela projeção vertical do parâmetro de entrada do eixo horizontal no limite mais alto da função de pertinência, possuindo valores compreendidos entre 0 e 1, (Lopes, 2003a).

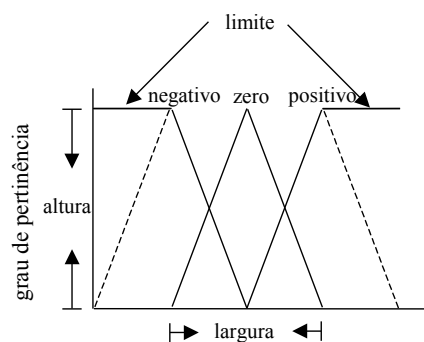


Figura 17. Funções de Pertinência.

O controle nebuloso é um mecanismo constituído, basicamente, de três partes: *nebulização* que converte variáveis reais em variáveis linguísticas; *inferência* que consiste na manipulação de base de regras utilizando declarações *if-then* e, ainda, operações nebulosas, como definidas anteriormente (equações

(1)–(5)) e *denebulização* que converte o resultado obtido (variáveis linguísticas) em variáveis reais.

As funções de pertinência nebulosas (*fuzzy*) podem ter diferentes formas, tais como triangular, trapezoidal e gaussiana, de acordo com a preferência / experiência do projetista.

O método mais comum de *denebulização* é o método de centro de área (centroide) (Kartalopoulos, 1996; Terano *et al.*) que encontra o centro da gravidade da solução dos conjuntos nebulosos. Para um conjunto nebuloso discreto tem-se (Kartalopoulos, 1996; Terano *et al.*, 1991):

$$u = \frac{\sum_{i=1}^n \mu_i \delta_i}{\sum_{i=1}^n \mu_i} \quad (64)$$

sendo:

δ_i : valor do conjunto que possui um valor de pertinência μ_i ;

n : número de regras nebulosas.

O valor u calculado pela equação (64) corresponde à projeção do centro de inércia da figura definida pelo conjunto de regras sobre o eixo da variável de controle.

Cada variável de estado geralmente é representada entre 3 e 7 conjuntos nebulosos.

REFERÊNCIAS

Barbosa, A. H.; Freitas, M. S. R.; Neves, F.A. (2005), Confiabilidade Estrutural Utilizando o Método de Monte Carlo e Redes Neurais, Revista Escola de Minas, v.58, no.3, Jul/Set 2005, pp. 247-255

- Bradley, D.W.; Tyrrell, A.M. (2002), Immunotronics - Novel Finite-State-Machine Architectures with Built-In Self-Test Using Self-Nonsel Self Differentiation. *IEEE Transactions on Evolutionary Computation*, v. 6, pp. 227-238.
- Carpenter, G.A. e Grossberg, S. (1987), A Massively Parallel Architecture For a Self-Organizing Neural Pattern Recognition Machine, *Computer Vision, Graphics, and Image Processing*, v. 37, pp. 54-115.
- Carpenter, G.A.; Grossberg, S. e Reynolds, J.H. (1991), ARTMAP: Supervised Real-Learning and Classification of Non-Stationary Data By A Self-Organizing Neural Network, *Neural Network*, v. 4, n. 5, pp. 565-588.
- Carpenter, G.A.; Grossberg, S. et al. (1992), Fuzzv ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. *IEEE Transactions on Neural Networks*, v. 3, n. 5.
- de Castro, L. N.; Von Zuben, F. J. (2000), The Clonal Selection Algorithm With Engineering Applications. In: *Proceedings of GECCO, Workshop on Artificial Immune Systems and Their Applications, 2000, Las Vegas*. pp. 36-39.
- de Castro, L. N. (2001), *Engenharia Imunológica: Desenvolvimento e Aplicação de Ferramentas Computacionais Inspiradas em Sistemas Imunológicos Artificiais*. Tese de Doutorado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, Brasil, 2001.
- de Castro, L. N. and Timmis, J. (2002), *Artificial Immune Systems: A New Computational Intelligence Approach*, Springer. 1st edition.
- Dasgupta, D. (1998), *Artificial Immune Systems and Their Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- Forrest, S., A. Perelson, A. L. and Cherukuri, R. (1994), Self-Nonsel Discrimination in a Computer, *Proc. do IEEE Symposium on Research in Security and Privacy*, pp. 202-212.

- de França, F. O., Von Zuben, F. J., de Castro, L. N. (2005), An Artificial Immune Network for Multimodal Function Optimization on Dynamic Environments. In: Proc. GECCO, Washington, DC, USA, pp. 289-296.
- Haykin, S. (1999), Neural Networks: A Comprehensive Foundation, *Prentice-Hall*, Upper Saddle River, New Jersey, USA, 1999.
- Jang, J.R.S. (1993), ANFIS: Adaptive-Network-Based Fuzzy Inference System, *IEEE Transactions on Systems, Man and Cybernetics*, pp. 665-685.
- Jang, J.R.S.; Sun, C. T. (1995), Neuro-Fuzzy Modeling and Control. Proceedings of the IEEE, March, pp. 378 - 406.
- Jang, J.-S. R., Sun, C. T.; Mizutani, E. (1997), *Neuro-Fuzzy and Soft Computing*, Prentice Hall, New York-USA.
- Kartalopoulos, S. V. (1996), *Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications*, *IEEE Press*, Piscataway, NJ, USA.
- Lima, F.P.A. (2013), *Analysis of Voltage Disturbances in Distribution Electrical Systems Based on Artificial Immune Systems*, Dissertation, University Estadual Paulista - UNESP, Ilha Solteira, March-2013, 169 p.
- Lopes, M. L. M.; Minussi, C. R.; Lotufo, A. D .P. (2003a), Electric Load Forecasting Formulation by a Fast Neural Network, *Engineering Intelligent Systems*, pp. 51-57, v. 11, n. 1, March 2003.
- Lopes, M. L. M.; Minussi, C. R. and Lotufo, A. PP. (2003b), A Fast Electric Load Forecasting Using Adaptive Neural Networks, *Proceedings of the 2003 IEEE Bologna Power-Tech*, Italy, v. 1, pp. 23-26, 2003.
- Niu, D. X.; Wang, H. Q.; Gu, Z. H. (2005), Short-term Load Forecasting Using General Regression Neural Network. In: *International Conference on Machine Learning and Cybernetics*, 2005, Guangzhou. Proceedings New York: IEEE, 2005. v. 7, pp. 4076-4082.
- Nose-Filho, K. (2011), *Previsão de Carga Multinodal Utilizando Redes Neurais de Regressão Generalizada*. 2011. 81f. Dissertação (Mestrado) – Faculdade de Engenharia, Universidade Estadual Paulista, Ilha Solteira, 2011.

- Ranganathan, A. (2004), The Levenberg- Marquardt Algorithm, [S.l: s.n.], 8 Jun. 2004, 5pp.
- Simpson, P.K. (1989), Artificial Neural Systems: Foundations, Paradigms, Applications, and Implementations. New York, *Pergamon Press*, 1989.
- Specht, D. F. (1991), A generalized regression neural network. *IEEE Transactions on Neural Networks*, New York, v. 2, pp. 568-576.
- Stamatis, N.; Parthimos, D. and Griffith, T. M. (1999), Forecasting Chaotic Cardiovascular Times Series With an Adaptive Slope Multilayer Perceptron Neural Network, *IEEE Transactions on Biomedical Engineering*, v. 46, n. 12, pp. 1441-1453.
- Takagi, T.; Sugeno, M. (1985), Fuzzy Identification of Systems and its Applications to Modeling and Control. *IEEE Transactions on Systems, Man and Cybernetics*, v.15, n.1, pp. 116-132.
- Terano, T.; Asai, K.; Sugeno, M. (1991), *Fuzzy Systems Theory and Its Applications*, Academic Press, USA. 1991.
- Werbos, P. J. (1974), *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. 1974. 453 f. Thesis (Master), Harvard University, Harvard, 1974.
- Widrow, B.; Lehr, M. A. (1990), 30 Years of Adaptive Neural Networks: Perceptron, madaline, and backpropagation. *Proceedings of the IEEE*, Winston, v. 78, n. 9, pp. 1415-1442.