

SINNET: SOFTWARE TUTORIAL DE IDENTIFICAÇÃO DE SISTEMAS ATRAVÉS DE REDES NEURAS ARTIFICIAIS

Fábio A. Guerra – guerra@tecpa.br

Divisão de Inteligência Artificial (DIA), TECPAR – Instituto de Tecnologia do Paraná
Rua Algacyr Munhoz Mader, 3775, CIC
CEP 81350-010 – Curitiba - Paraná

Leandro dos Santos Coelho – leandro.coelho@pucpr.br

Pontifícia Universidade Católica do Paraná, Grupo Produtônica
Programa de Pós-Graduação em Engenharia de Produção e Sistemas
Laboratório de Automação e Sistemas, Rua Imaculada Conceição, 1155
CEP 80215-901 – Curitiba - Paraná

***Resumo:** Este artigo visa apresentar um ambiente computacional desenvolvido em linguagem de programação C++ para auxílio no ensino dos fundamentos de previsão de séries temporais de sistemas não-lineares com uso de redes neurais artificiais (redes neurais perceptron multicamadas, função de base radial e rede neural baseada em group method of data handling). Os conceitos básicos e exemplos explicativos (passo a passo) dos sistemas caóticos de Chua e Rössler são apresentados, estes facilitam o aprendizado do usuário pela utilização de janelas gráficas, demonstrações e interface amigável.*

***Palavras-chave:** software, redes neurais artificiais, identificação de sistemas, sistemas não-lineares, circuitos caóticos.*

1. INTRODUÇÃO

A análise de séries temporais é uma área de pesquisa relevante em diversos campos do conhecimento, principalmente nas áreas de engenharia e economia. A principal motivação para pesquisas sobre séries temporais é providenciar uma previsão quando o modelo matemático de um fenômeno é complexo, desconhecido ou incompleto. Uma série temporal consiste de medidas ou observações previamente obtidas de um fenômeno que são realizadas sequencialmente sob um intervalo de tempo. Se estas observações consecutivas são dependentes uma da outra então é possível conseguir-se uma previsão ou identificação do sistema.

A concepção de modelos matemáticos para a representação de sistemas complexos de séries temporais é um procedimento relevante e com aplicação prática. Entretanto, em geral, não é uma tarefa simples a construção de modelos matemáticos adequados para os propósitos de engenharia. Nas últimas décadas, diversos de algoritmos para modelagem e identificação de sistemas dinâmicos complexos têm sido propostos na literatura, tais como: métodos freqüenciais, técnicas baseadas em estimativas de modelos Wiener, Hammerstein, bilinear e Volterra, regressão não-linear, *wavelets* e identificação recursiva (LJUNG, 1987; HABER & UNBEHAUEN, 1990). Uma abordagem relevante entre tantas outras para representação matemática de sistemas dinâmicos com comportamento complexo ou caótico é a das redes neurais.

Os modelos matemáticos concebidos baseados em algumas abordagens de redes neurais têm recebido atenção recentemente, tanto da comunidade científica quanto dos profissionais que atuam no meio industrial, por tratarem-se de ferramenta de projeto que oferecem soluções promissoras para problemas de identificação de sistemas dinâmicos complexos (MCLOONE *et al.*, 1998; HOVAKIMYAN *et al.*, 2000; PAN *et al.*, 2001).

Este artigo apresenta os fundamentos de redes neurais, teoria do caos e identificação de sistemas não-lineares. Assim este é organizado da seguinte maneira. Na seção 2, os fundamentos da teoria do caos e identificação de sistemas são apresentados. Na seção 3, os conceitos e potencialidades da utilização de redes neurais artificiais são ressaltados. O software tutorial denominado SSINET é descrito e ilustrado na seção 4. A conclusão deste artigo e as perspectivas de futuros trabalhos são apresentadas na seção 5.

2. FUNDAMENTOS DE TEORIA DO CAOS E IDENTIFICAÇÃO DE SISTEMAS

A área denominada de teoria do caos possibilita o estudo de diversos fenômenos pertinentes aos sistemas dinâmicos (qualquer processo que evolua com o tempo) não-lineares e que apresentam comportamento complexo. A teoria do caos estuda os fenômenos aparentemente imprevisíveis, na busca de padrões escondidos e de leis simples que regem os comportamentos complexos. Porém, este estudo só se tornou efetivamente plausível a partir da década de 1960, quando os computadores começaram a possuir capacidade gráfica e processamento razoáveis, dando aos físicos e matemáticos o poder de descobrir respostas para questões fundamentais da ciência de maneira geral, que antes estavam obscuras.

Muitos dos estudos dos sistemas não-lineares surgiram a partir da teoria do caos que fornece uma explicação muitas vezes adequada (através de fórmulas e equações), a muitos comportamentos presentes na natureza, tais como: fenômenos naturais (populações, turbulências, movimento de fluidos e formação de nuvens), comportamento complexo em circuitos elétricos, comportamento de bolsa de valores e economia, sistemas não-lineares e variantes no tempo, telecomunicações, controle de processos, comportamento dinâmico dos batimentos cardíacos, entre outros (PEI & HEI, 1999; LIAN & LIU, 2000).

A teoria do caos tenta explicar a aparente desordem de uma forma ordenada, estabelecendo que as coisas não são aleatórias, mas apenas complexas. Muitos eventos aparentemente aleatórios podem ser representados por um método computacional simples que, quando bem projetado, produz resultados complexos. Os valores de cada estágio são realimentados no próximo estágio (*feedback*) e o resultado das iterações, fórmulas e feedback resultam num sistema caótico.

O comportamento dos sistemas caóticos pode apresentar grande sensibilidade em relação às condições iniciais a que estão sujeitos. Apesar de, em termos determinísticos, ser uma tarefa difícil descrever o comportamento de um sistema caótico, em termos probabilísticos esta situação pode ser tratada e alguns paradigmas têm sido apresentados na literatura para esta finalidade (ALLIGOOD *et al.*, 1996).

A área de identificação de sistemas é bem desenvolvida para sistemas lineares. Entretanto um modelo linear é somente útil se o processo físico exibir o comportamento dinâmico similar ao modelo linear operando na região de interesse. O fato é que todos os sistemas físicos são não-lineares a uma extensão, conduzindo a necessidade do desenvolvimento de ferramentas de aproximações de modelos não-lineares e de análise de sistemas não-lineares.

A identificação de processos é uma área relevante em muitos campos do conhecimento, é o procedimento de identificar um modelo de um processo desconhecido, para propósitos de previsão e/ou compreensão do comportamento do processo (LJUNG, 1987). A complexidade inerente de muitos processos reais (não-lineares e variantes no tempo) dificulta a aplicação de técnicas convencionais de identificação. Este aspecto tem motivado o desenvolvimento de técnicas de identificação avançadas, baseadas em inteligência computacional, da qual as redes

neurais, os algoritmos evolutivos e os sistemas nebulosos fazem parte (COELHO & GUERRA, 2002).

A identificação de sistemas é tratada, muitas vezes, como um problema de otimização que envolve algumas medidas para adequação de modelos candidatos a representar um processo real. A escolha de modelos matemáticos e ajuste dos parâmetros adequados são influenciados por diversos fatores, entre os quais: (i) conhecimento a priori do sistema (linearidade, grau de não-linearidade, atraso de transporte); (ii) propriedades do modelo do sistema identificado (complexidade); (iii) escolha da medida de erro a ser minimizado; (iv) presença de ruídos. A identificação de sistemas é um claro exercício que envolve múltiplos e conflitantes objetivos, tipicamente complexidade do modelo, critérios de desempenho e validação que influenciam a seleção das estruturas de modelo matemático mais adequadas. A noção de um “bom modelo” é subjetiva e depende da preferência pessoal, levando a tentativa e erro ser uma regra relevante em identificação de processos.

3. REDES NEURAIS ARTIFICIAIS

Uma rede neural artificial pode ser considerada como uma ferramenta computacional que resultou de pesquisas no campo da inteligência artificial, e procurava entender e modelar o comportamento do cérebro humano. A interação entre neurônios no cérebro humano, na realidade, não é visível, mas é identificada como comportamento inteligente. Com o propósito de desenvolver o mesmo tipo de estrutura para um modelo computacional de comportamento inteligente, neurologistas e pesquisadores da inteligência artificial propuseram uma rede altamente interconectada de nódulos (ou neurônios).

Em síntese, as redes neurais artificiais consistem em elementos de processamento altamente interconectados denominados neurônios, cada qual possuindo um número de entradas e uma saída. A saída de cada neurônio é determinada como uma função não-linear de uma soma ponderada das entradas, embora operações matemáticas mais complexas poderiam ser incluídas. Os neurônios se interconectam através de pesos, os quais são ajustados durante o período de treinamento (HAYKIN, 2000).

As redes neurais artificiais são especificadas pela topologia, características dos neurônios e regras de treinamento. Estas regras de treinamento especificam como o conjunto de pesos iniciais deve ser adaptado para a rede aprender o comportamento de um sistema. Os algoritmos particulares de treinamento (ou aprendizado) de uma rede neural ajustam iterativamente os pesos das conexões entre os neurônios até que os pares desejados de informações de entrada(s) e saída(s) sejam obtidos e as relações de causa e efeito possam ser estabelecidas. Se as condições mudarem de tal modo que o desempenho do modelo não atender aos requisitos de desempenho necessários, pode-se submeter à rede neural artificial a um novo procedimento de treinamento sob estas novas condições de entrada(s) e saída(s) para corrigir seu desempenho. Assim sendo, podem-se apresentar dados a rede neural artificial a uma atualização periódica, resultando num modelo auto-ajustável em linha (LEE & SIM, 1999).

Entre as aplicações usuais de redes neurais artificiais têm-se: reconhecimento e classificação de padrões, *clustering*, previsão de séries temporais, aproximação de funções, predição, otimização, setor militar (processamento de sinais para identificação de alvos e análise de imagens), sistemas especialistas, processamento de sinais (imagens, sensores, voz, caracteres, visão, compressão de dados, filtragem de sinais), telecomunicações, manufatura, monitoramento de processos e robótica (STECK *et al.*, 1996).

4. SOFTWARE TUTORIAL SINNET

O desenvolvimento do SINNET (*software* tutorial de identificação de sistemas dinâmicos baseado em redes neurais artificiais) tem a intenção de auxiliar estudantes e/ou profissionais de Engenharia em um primeiro contato com a teoria de redes neurais artificiais, identificação de sistemas, teoria do caos e circuitos elétricos com comportamento caótico. O SINNET possui uma forma didática e amigável de apresentação das teorias através de janelas gráficas e com recursos de demonstrações. A primeira interface do SINNET mostra informações dos autores e parceiros envolvidos em seu desenvolvimento, conforme ilustrado na Figura 1.



Figura 1 - Janela de apresentação do SINNET.

Após a tela da Figura 1, o usuário para continuar a utilização do SINNET deve clicar no botão *Continuar* e uma tela de *Opções* será apresentada. A tela do menu de *Opções* possui todas as opções disponíveis no SINNET (versão 1.0), conforme ilustrado na Figura 2.

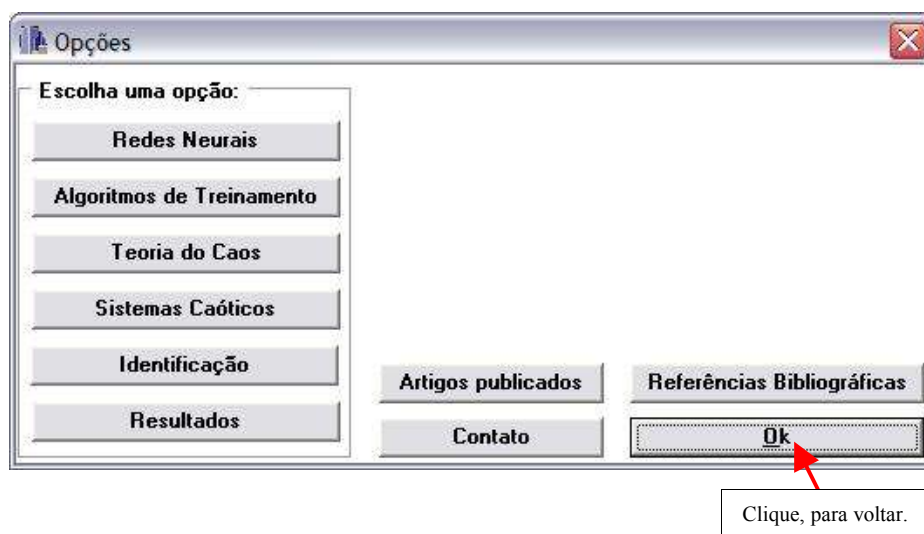


Figura 2 - Janela de opções do *software* tutorial SINNET.

No item “Escolha uma opção”, pode-se encontrar informações sobre:

- **Redes Neurais:** escolhendo a opção “Redes Neurais”, o usuário encontra informações sobre as topologias (estruturas) das redes neurais utilizadas para identificar os sinais no domínio do tempo gerados por sistemas caóticos (osciladores eletrônicos caóticos);
- **Algoritmos de Treinamento:** escolhendo a opção “Algoritmos de Treinamento”, o usuário obtém informações sobre os algoritmos de treinamento (aprendizado) utilizados pelas redes neurais em identificação de sistemas dinâmicos;
- **Teoria do Caos:** escolhendo esta opção “Teoria do Caos”, o usuário encontra uma introdução e aplicações da teoria do caos;
- **Sistemas Caóticos:** escolhendo a opção “Sistemas Caóticos”, o usuário do SINNET encontra informações sobre dois osciladores caóticos. Neste contexto, o SINNET pode também plotar os sinais gerados, visualizar o atratores e os circuitos eletrônicos de onde os sinais caóticos foram obtidos;
- **Identificação:** escolhendo a opção “Identificação”, o usuário encontra uma introdução à teoria de identificação de sistemas dinâmicos não-lineares;
- **Resultados:** escolhendo a opção “Resultados”, o usuário pode visualizar os resultados da identificação dos alguns sistemas dinâmicos com comportamento caótico utilizando redes neurais.

E para finalizar esta interface de opções, esta também possui as seguintes informações:

- **Artigos publicados:** clicando neste botão, o usuário encontra referências de artigos publicados dos autores relacionados ao SINNET;
- **Contato:** clicando neste botão, o usuário obtém informações referentes aos autores do SINNET;
- **Referências Bibliográficas:** clicando neste botão, o usuário acessa as referências bibliográficas utilizadas para o desenvolvimento do SINNET.

Para a demonstração das principais opções existentes no SINNET, inicia-se pela opção “Redes Neurais”, que a partir desta, a seguinte tela é apresentada, como ilustrado na Figura 3.

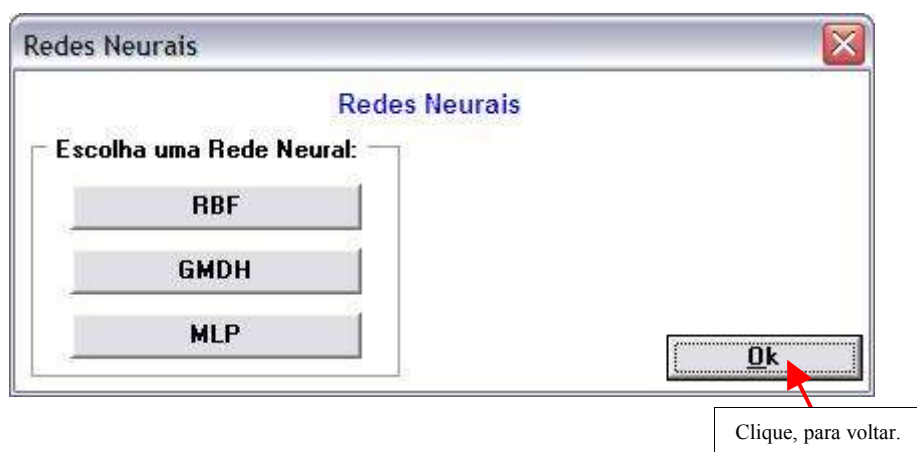


Figura 3 - Janela de redes neurais.

Nesta interface são acessadas informações sobre três tipos de redes neurais, que são as seguintes:

- **RBF:** rede neural de função de base radial (CHEN, 1990);
- **GMDH:** rede neural polinomial baseada em *Group Method of Data Handling* (GMDH) (GAMO *et al.*, 1998; GUERRA & COELHO, 2003);

- **MLP:** rede neural *perceptron* multicamadas (GUERRA & COELHO, 2002; LO, 1998).

Para exemplificar como as informações dessas redes neurais são apresentadas, ilustram-se as informações da rede neural GMDH na Figura 4.

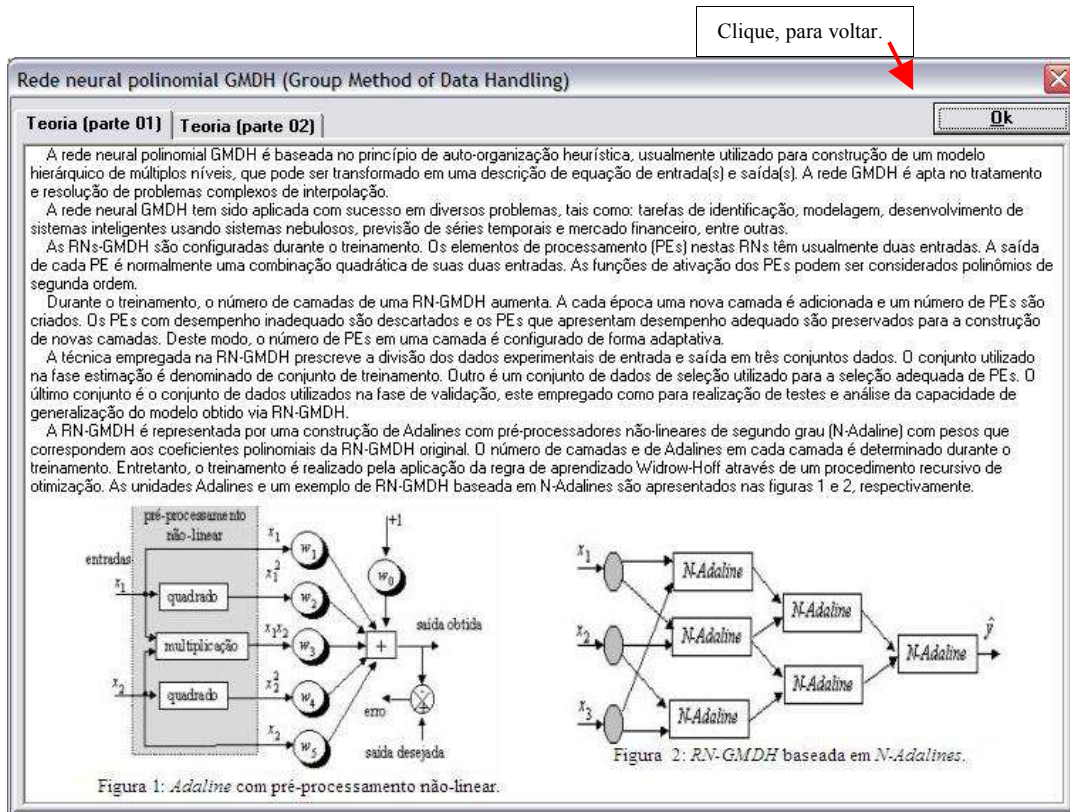


Figura 4 – Janela relacionada a rede neural polinomial GMDH.

Voltando a tela de opções, clique em “Sistemas Caóticos”. Os dois estudos de casos para a identificação que são o circuito de Chua (TÔRRES & AGUIRRE, 1999; TÔRRES & AGUIRRE, 2000; THAMILMARAN, 2000) e o circuito de Rössler (GUERRA & COELHO, 2003; RÖSSLER, 1976) são apresentados na Figura 5.

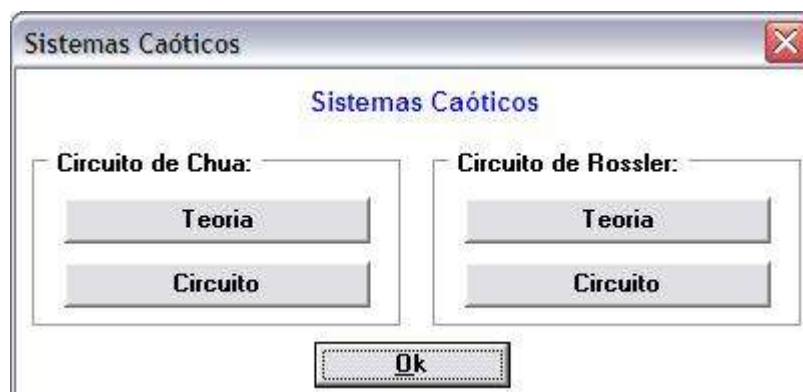


Figura 5 - Janela de sistemas caóticos.

Para exemplificar como as informações de cada oscilador eletrônico caótico são apresentadas, a seguir na Figura 6 são apresentadas algumas informações do circuito de Rössler. Neste caso, o usuário clicando no botão “Circuito” a interface da Figura 6 surge.

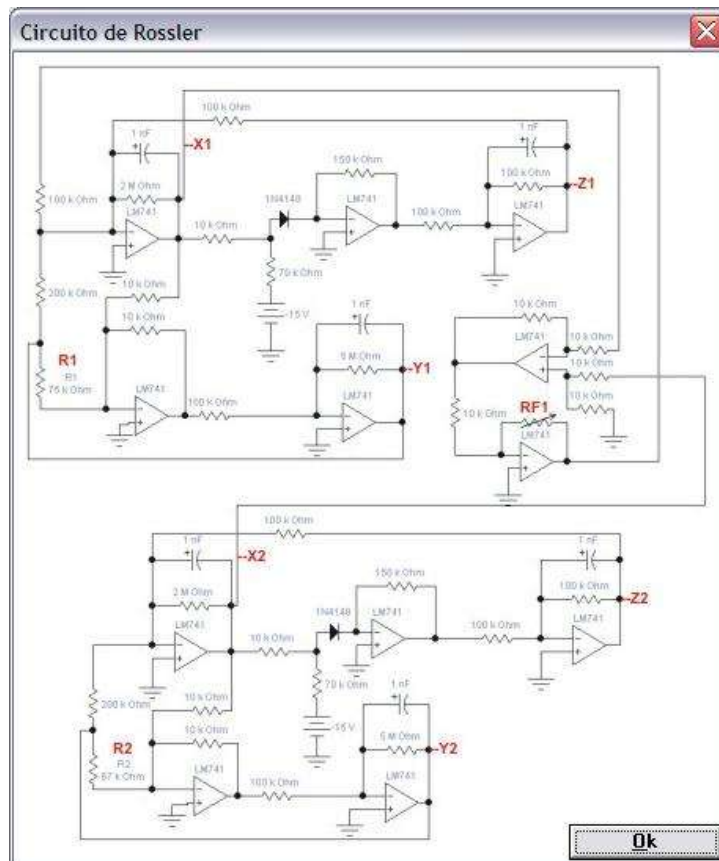


Figura 6 – Janela de apresentação do circuito de Rössler.

Acessando a tela “Circuito de Rössler” observa-se os sinais gerados e também diversas explicações referentes a este circuito. Além disso, a interface da teoria do circuito de Rössler pode ser obtida através do botão de “Teoria” (do item circuito de Rössler), conforme mostrado na Figura 7.

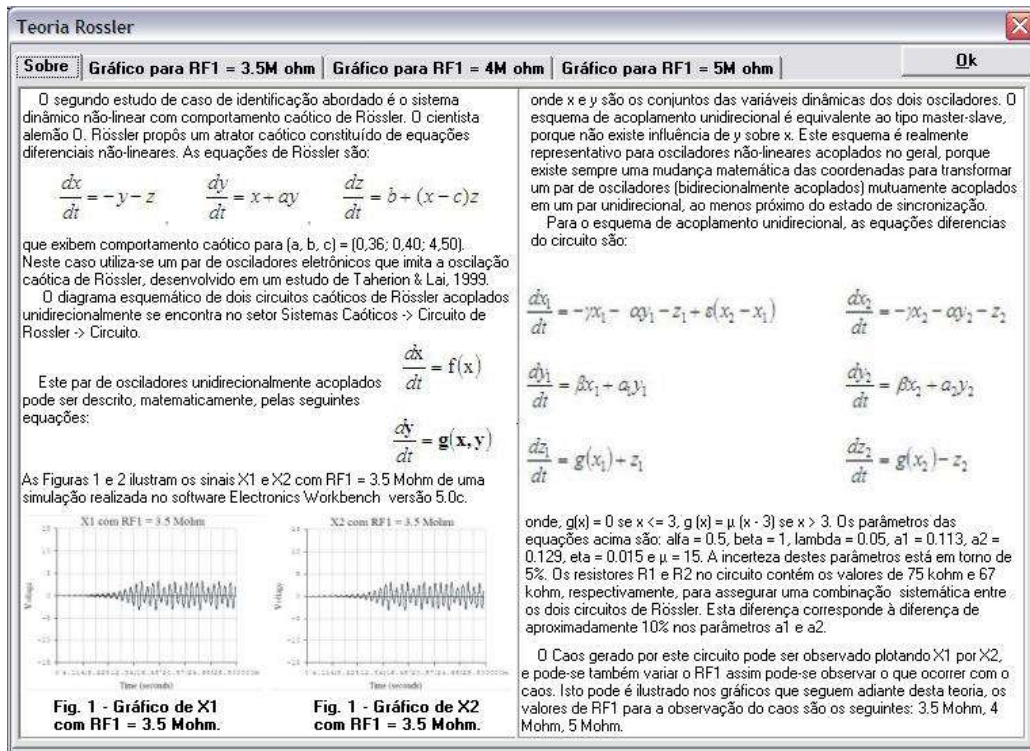


Figura 7 - Teoria do circuito de Rössler.

A partir desta interface é possível a obtenção de informações referentes ao circuito de Rössler, plotar os sinais gerados e também observar cada sinal caótico gerado pelo circuito com diferentes valores de RF1, como ilustram os gráficos da Figura 8.

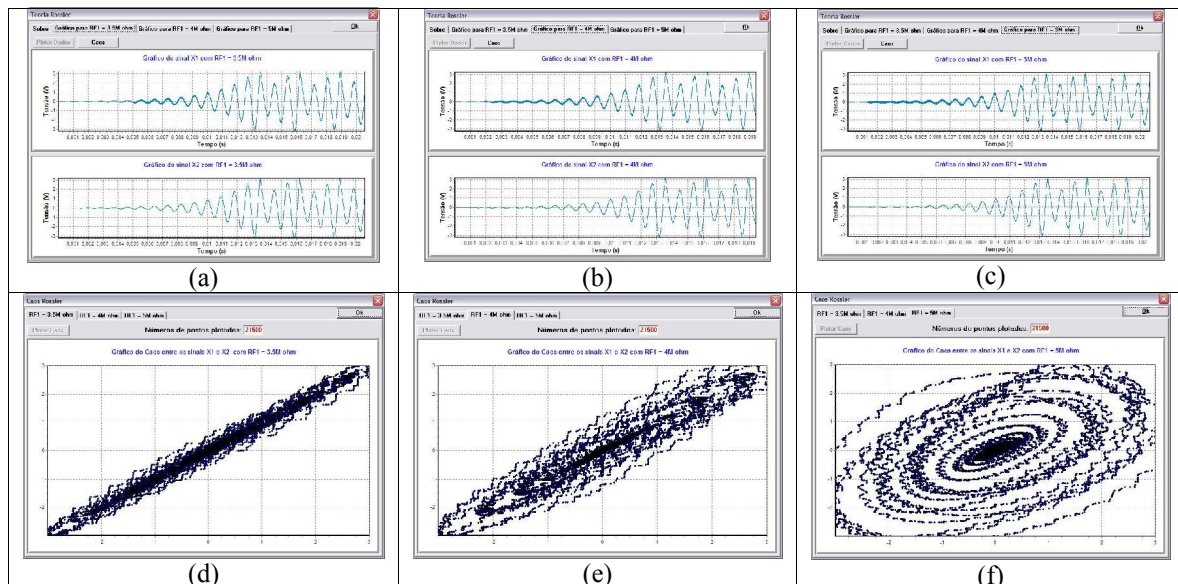


Figura 8 – Sinal gerado com (a) RF1 = 3.5 MΩ, (b) RF1 = 4 MΩ, e (c) RF1 = 5 MΩ. Atrator obtido com: (d) RF1 = 3.5 MΩ, (e) RF1 = 4 MΩ, e (f) RF1 = 5 MΩ.

Retornando a interface de opções e escolhendo a opção “Resultados”, a tela de resultados será apresentada contendo as opções de visualização de resultados como ilustra a Figura 9.

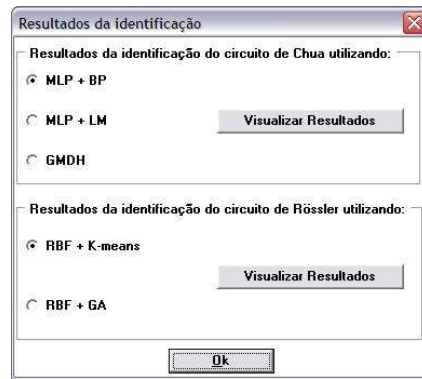


Figura 9 - Tela de resultados.

Para exemplificar os resultados obtidos, da identificação de sistemas não-lineares com comportamento caótico através de redes neurais, as Figuras 10 e 11 ilustram dois resultados obtidos da identificação utilizando redes neurais. A Figura 10 ilustra o resultado da rede neural GMDH e seu erro para o circuito de Chua, enquanto a Figura 11 ilustra o resultado da rede neural RBF utilizando o algoritmo *K-means* e pseudo-inversa para o circuito de Rössler.

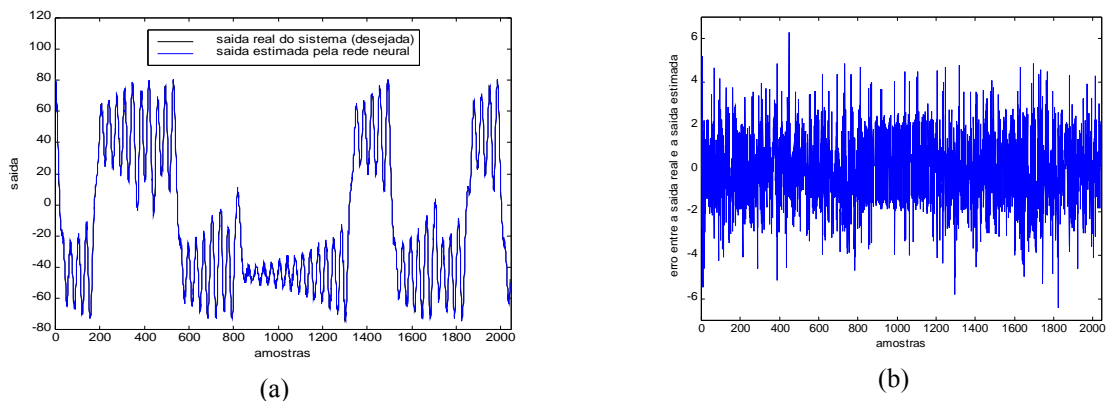


Figura 10 - (a) Resultado da identificação e (b) erro de cada amostra na identificação do circuito de Chua usando uma rede neural GMDH.

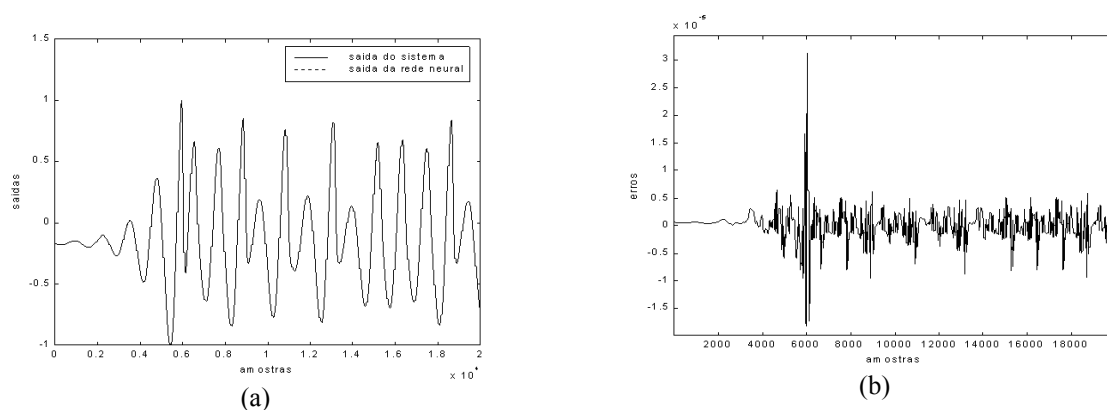


Figura 11 - (a) Resultado da identificação e (b) erro de cada amostra na identificação do circuito de Rössler usando uma rede neural RBF.

5. DISCUSSÃO E CONCLUSÃO

A motivação para o desenvolvimento de um *software* tutorial de identificação de sistemas não-lineares (SINNET) é promover o primeiro contato de graduandos de cursos de Engenharia com a área de redes neurais e identificação de sistemas. Numa visão geral do sistema, o

SINNET traz um conjunto introdutório de informações sobre: redes neurais, algoritmos de treinamentos, sistemas não-lineares com comportamento caótico (circuitos eletrônicos), teoria do caos e identificação de sistemas não-lineares.

O objetivo do SINNET é encontrar graduandos interessados em pesquisa. Para isto, este contém dois exemplos de casos, os osciladores eletrônicos caóticos de Chua e de Rössler. Inicialmente os graduandos interessados podem iniciar com a implementação (montagem) física destes circuitos para que um interesse inicial não seja puramente teórico, que é um dos motivos que afastam possíveis iniciantes em pesquisa. A partir deste primeiro passo, o aluno pode iniciar sua pesquisa em redes neurais para identificação dos sistemas não-lineares.

Outros temas deverão ser abordados após em versões futuras do SINNET, por exemplo, a teoria e aplicação de sistemas nebulosos (*fuzzy*) e algoritmos genéticos em identificação não-linear.

REFERÊNCIAS BIBLIOGRÁFICAS

- ALLIGOOD, K. T.; SAUER, T. D.; and YORKE, J. A. **Chaos: an introduction to dynamical systems**. Springer, 1996.
- CHEN, S.; BILLINGS, S. A.; COWAN, C. F.; GRANT, P. M. Practical identification of NARMAX models using radial basis functions. **International Journal of Control**, v. 52, n. 6, p. 1327-1350, 1990.
- COELHO, L. S.; GUERRA, F. Rede neural de base radial aplicada à identificação de um processo experimental multivariável. **XIV Congresso Brasileiro de Automática**, Natal, RN, v. 1, p. 1-6, 2002.
- GAMO, S. O.; RACHID, A.; and OULADSINE, M. GMDH method applied to nonlinear identification of a turbocharged diesel engine. **Proceedings of American Control Conference**, Philadelphia, Pennsylvania, USA, p. 962-963, 1998.
- GUERRA, F.; COELHO, L. S. Identificação de sistema dinâmico caótico usando rede neural perceptron multicamadas. **1º Simpósio Sul-Brasileiro de Matemática e Informática**, Curitiba, PR, v. 1, p.104-104, 2002.
- GUERRA, F.; COELHO, L. S. Identificação do sistema caótico de Chua usando uma rede neural polinomial. **VI Congresso Brasileiro de Redes Neurais**, São Paulo, v. 1, p. 271-276, 2003.
- GUERRA, F.; COELHO, L. S. Usando uma abordagem de redes neurais na identificação do sistema dinâmico caótico de Rössler. **XXXV Simpósio Brasileiro de Pesquisa Operacional**, Natal, RN, v. 1, p. 2547-2547, 2003.
- HABER, R.; UNBEHAUEN, H. Structure identification of nonlinear dynamic systems – a survey on input/output approaches, **Automatica**, v. 26, n. 4, p. 651-677, 1990.
- HAYKIN, S. **Redes neurais: princípios e aplicações**, Bookman, Porto Alegre, RS, 2000.
- HOVAKIMYAN, N.; LEE; H.; CALISE, A. On approximate NN realization of an unknown dynamic system from its input-output history, **Proceedings of the American Control Conference, Philadelphia**, Pennsylvania, USA, v. 2, p. 919-923, 2000.
- LEE, D.-W.; SIM, K.-B. Evolving chaotic neural systems for time series prediction, **Proceedings of the Congress on Evolutionary Computation**, Washington, DC, v. 1, p. 310-316, 1999.
- LIAN, K. W.; LIU, P. Synchronization with message embedded for generalized Lorenz chaotic circuits and its error analysis. **IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications**, v. 47, n. 9, p. 1418-1430, 2000.
- LJUNG, L. **System identification: theory for the user**. Prentice-Hall, New York, 1987.
- LO, J. T. Multilayer perceptrons and radial basis functions are universal robust approximators, **International Joint Conference on Neural Networks**, Anchorage, AK, v. 2, p. 1311-1314, 1998.

- MCLOONE, S., BROWN, M. D., IRWIN, G., LIGHTBODY, A. A hybrid linear/nonlinear training algorithm for feedforward neural networks, **IEEE Transactions on Neural Networks**, v. 9, n. 4, p. 669-684, 1998.
- PAN, Y.; SUNG, S. W.; LEE, J. H. Data-based construction of feedback-corrected nonlinear prediction model using feedback neural networks, **Control Engineering Practice**, v. 9, p. 859-867, 2001.
- PEI, L. Y.; HE, Z. Identification of dynamical noise levels in chaotic systems and application to cardiac dynamics analysis. **International Joint Conference on Neural Networks, IJCNN**, v. 5, p. 3680-3684, 1999.
- RÖSSLER, O. E. An equation for continuous chaos, **Physical Letters**, v. 35A, p. 397-398, 1976.
- STECK, J. E.; ROKHSAZ, K.; SHUE, S. P. Linear and neural network feedback for flight control decoupling. **IEEE Control Systems**, v. 16, n. 4, 1996, p. 22-30.
- THAMILMARAN, M.; LAKSHMANAN; MURALI, K. Rich variety of bifurcations and chaos in a variant of Murali-Lakshmanan Chua circuit. **International Journal of Bifurcation and Chaos**, v. 10, n. 7, p. 1781-1785, 2000.
- TÔRRES, L. A. B.; AGUIRRE, L. A. Extended chaos control method applied to Chua's circuit. **Electronics Letters**, v. 35, n. 10, p. 768-770, 1999.
- TÔRRES, L. A. B.; AGUIRRE, L. A. Inductorless Chua's circuit. **Electronics Letters**, v. 36, n. 23, p. 1915-1916, 2000.

SINNET: TUTORIAL SOFTWARE OF SYSTEMS IDENTIFICATION BASED ON ARTIFICIAL NEURAL NETWORKS

Abstract: *This paper presents a computational environment developed in C++ language for aid in learning of basic concepts of nonlinear time series using artificial neural networks (multilayer perceptron, radial basis function neural network, and group method of data handling networks). The fundamentals and didactic examples (step by step) of chaotic systems of Chua and Rössler are presented. These examples motivate the user learning by using of graphical windows, demonstrations and user-friendly interface.*

Key-words: *software, artificial neural networks, systems identification, nonlinear systems, chaotic circuits.*