

Identificação de sistemas dinâmicos não lineares usando back-propagation com teacher forcing

V. Leonardo Paucar¹, Marcos J. Rider¹, André L. Morelato², Evandro B. Vuono³

¹UFMA, São Luís, Brasil

²UNICAMP, Campinas, Brasil

³Telesp Celular, São Paulo, Brasil

E-mails: lpaucar@ieee.org, mjrider@ieee.org, morelato@dsee.fee.unicamp.br

Abstract

In this paper it is presented the algorithm and the numerical results for identification of non-linear dynamic systems using multilayer perceptrons artificial neural networks (ANN) trained with back-propagation with teacher forcing (BPTF). There have been analyzed several ANN configurations containing two neurons layers, one hidden and the other one in the output. The proposed artificial neural networks have been applied to double pendulum system and to the identification problem of the third-order model of the induction motor. Results obtained from applications suggest that artificial neural networks using BPTF for non-linear dynamic systems simulation and identification are very useful at least for the first steps after the training time period of the proposed neural networks.

1. Introdução

Durante a última década diversas técnicas de inteligência artificial têm sido aplicadas para resolver problemas relacionados com simulação e controle de sistemas dinâmicos. Uma das técnicas IA que tem recebido maior atenção é redes neurais artificiais (RNA). As redes neurais podem ser classificadas em dois grupos: as redes recorrentes e as não recorrentes. Teoricamente o primeiro representa a possibilidade de mapeamentos não lineares estáticos, enquanto que o segundo grupo tem a capacidade de representar sistemas dinâmicos não lineares realimentados.

A identificação de sistemas dinâmicos é um dos maiores problemas dentro da teoria dos sistemas de controle. Apesar da existência de diversas referências sobre a caracterização de funções não lineares, somente algumas técnicas podem ser aplicadas na identificação de sistemas dinâmicos práticos.

Recentemente, as redes neurais estão sendo usadas na modelagem de sistemas dinâmicos não lineares [1]. Uma das vantagens das RNA é que favorecem a modelagem implícita dos sistemas dinâmicos, isso devido a que a implementação com a tecnologia computacional atual não é muito difícil. [2, 3]

A capacidade que têm as RNA multicamada para aproximar funções não lineares usando o algoritmo de treinamento de retropropagação de erros (BP por *back-propagation error*) faz factível seu uso na identificação, simulação e controle de sistemas dinâmicos. [4]

Neste trabalho são detalhadas as considerações na aplicação do método back-propagation usando o conceito de professor forçado (teacher forcing) o qual permite, via o uso de atrasos convenientes, que as redes neurais multicamada sejam utilizadas para identificar e simular sistemas dinâmicos não lineares. Os resultados obtidos indicam a possibilidade de previsão adequada de alguns estados futuros dos sistemas motivo de aplicação. O sistema dinâmico do pêndulo duplo foi utilizado para mostrar o algoritmo e a utilidade das estimativas feitas. A aplicação do motor de indução de terceira ordem se orienta a uma aplicação mais prática na área da simulação e identificação de parâmetros das máquinas elétricas e os sistemas de potência.

2. Dinâmica do pêndulo duplo

Antes de apresentar o algoritmo back-propagation com teacher forcing, a maneira de exemplo tutorial será descrita a dinâmica do pêndulo duplo (ver Figura 1) modelado no domínio do espaço de estado, desenvolvido usando um conjunto de equações diferenciais de primeira ordem.

As equações diferenciais não lineares são:

$$\dot{x}_1 = x_2 \quad (1)$$

$$\dot{x}_2 = -\frac{g}{L_1} \sin x_1 + \frac{m_2}{m_1} \frac{g}{L_1} \cos x_3 \sin(x_3 - x_1) \quad (2)$$

$$\dot{x}_3 = x_4 \quad (3)$$

$$\dot{x}_4 = -\frac{g}{L_2} \sin x_3 \quad (4)$$

sendo:

$$x_1 = \theta_1 \quad x_2 = \dot{\theta}_1$$

$$x_3 = \theta_2 \quad x_4 = \dot{\theta}_2$$

A integração numérica das equações (1) a (4) pode ser realizada usando por exemplo o método explícito Runge-Kutta de quarta ordem.

Na Figura 2 se apresentam algumas simulações considerando os parâmetros: $m_1 = 0.1$ kg, $m_2 = 0.2$ kg, $L_1 = 0.2$ m, $L_2 = 0.4$ m. Além disso foram consideradas as seguintes condições iniciais: $x_1 = x_3 = 45^\circ$; $x_2 = x_4 = 0$.

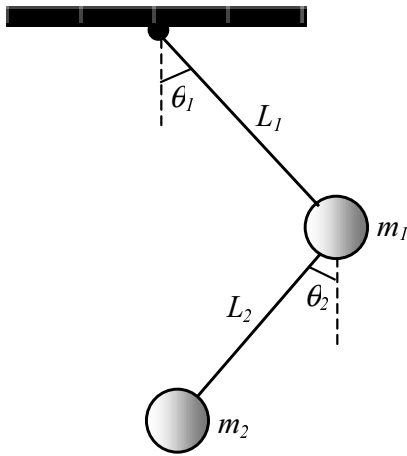


Figura 1: Pêndulo duplo usado como sistema dinâmico não linear nos testes.

As simulações mostradas na Figura 2 indicam a não linearidade do sistema dinâmico. Outras simulações usando diferentes condições iniciais podem alterar o movimento do pêndulo e consequentemente podem ser geradas outras curvas.

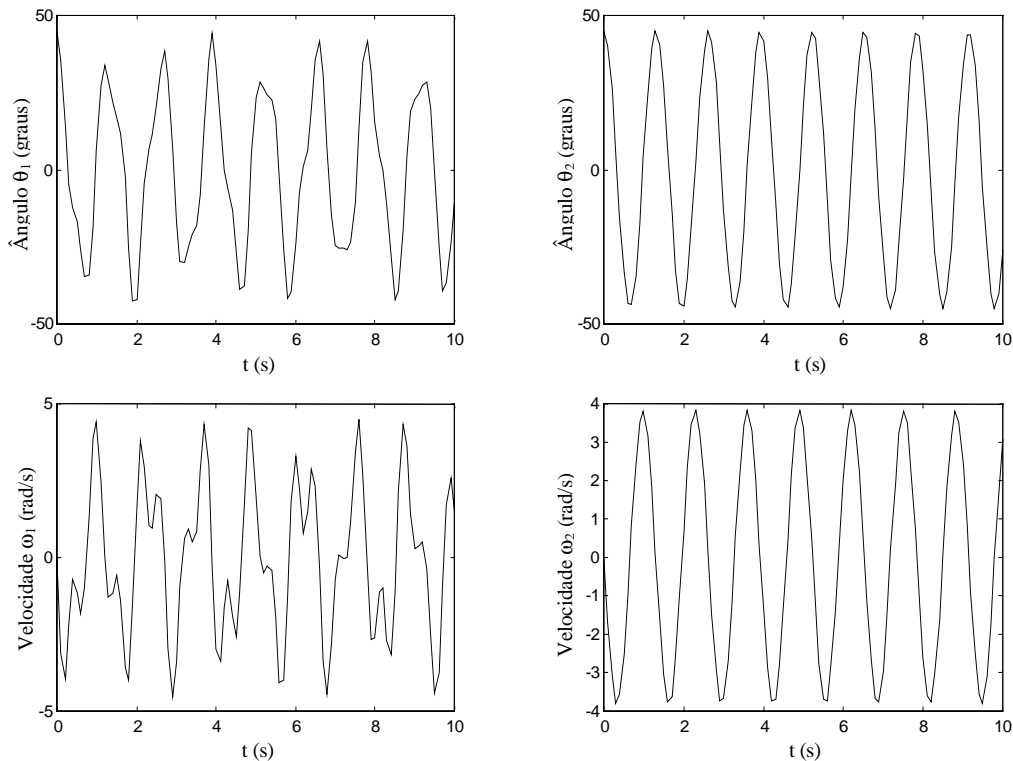


Figura 2: Simulações da dinâmica do pêndulo duplo mostrando as posições angulares θ_1 e θ_2 e as velocidades ω_1 e ω_2 quando as condições iniciais são: $x_1 = x_3 = 45^\circ$, $x_2 = x_4 = 0$.

3. Back-propagation com teacher forcing

O procedimento teacher forcing através do uso de atrasos ou *delays* tenta introduzir dinâmica no algoritmo de treinamento de redes neurais back-propagation evitando possíveis saturações no processo de realimentação. A arquitetura mostrada na Figura. 3 sugere como realizar o treinamento da rede neural de perceptrons multicamada. O método teacher forcing consiste em ingressar os valores de saída desejadas dentro das unidades a medida que a rede neural é executada; consequentemente, o professor ou teacher vai forçar as unidades de saída terem os estados corretos. Independente do método de treinamento, a minimização do erro quadrático médio, $e^2(n)$, é um esquema de adaptação em malha aberta no sentido que a saída real da rede neural não é realimentada à entrada durante o treinamento. Na literatura sobre controle e processamento de sinais, esse método de treinamento é referenciado como a formulação de equação do erro, enquanto que na literatura sobre redes neurais é conhecido como teacher forcing. [5, 2, 4]

Na Figura 3, a chave A indica a posição para treinamento enquanto a chave B é a opção para usar a rede neural. O esquema de adaptação em malha aberta ou treinamento (posição A), representa a previsão para frente. Uma vez que o treinamento da rede neural tem sido completado, o desempenho de generalização da rede neural é avaliado realizando uma predição recursiva de uma maneira autônoma ou esquema de adaptação em malha fechada (posição B).

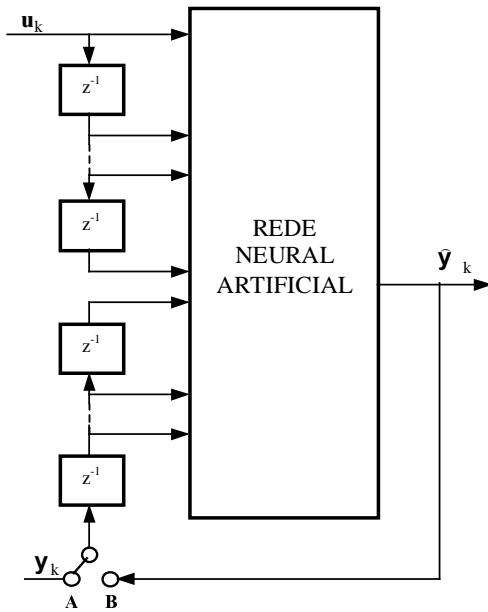


Figura 3: Arquitetura do back-propagation com teacher forcing.

A formulação de um sistema dinâmico não linear no domínio de espaço de estados é a seguinte:

$$x_K = f(x_{K-1}, u_{K-1}) \quad (5)$$

$$y_K = g(x_K) \quad (6)$$

Para reproduzir a evolução do sistema dinâmico não linear é requerido o conjunto de condições iniciais do sistema. Atrasos de tempo são usados para simular a discretização da dinâmica do sistema.

O procedimento teacher forcing mantém a saída da rede muito próximo do caminho desejado e parece acelerar o aprendizado.

A rede neural ficará treinada corretamente somente dentro do intervalo (região de atração) que envolve o conjunto de condições iniciais. Uma previsão de curto prazo é calculada iterativamente ao realimentar-se a seqüência de previsões de um passo para frente, calculadas pela rede, à entrada. Após a inicialização não há entradas externas aplicadas à rede neural. A previsão é considerada como de curto prazo porque normalmente é possível de manter uma identificação confiável durante um número limitado de passos de tempo futuros. O comportamento da rede neural é muito sensível às condições iniciais e a qualidade das previsões degradam rapidamente uma vez que se passem um certo número de passos de tempo futuros.

4. Resultados da identificação com o pêndulo duplo

O pêndulo duplo é o primeiro sistema dinâmico não linear para testar o método back-propagation com teacher forcing. A rede neural proposta foi treinada para reproduzir o movimento do pêndulo dada uma certa condição inicial. Foi desenvolvido um programa em linguagem C++ implementando o BPTT para uma rede de duas camadas escondidas de neurônios. O programa computacional lê a arquitetura da rede, a taxa de aprendizagem, o momentum e número de atrasos de tempo desejados.

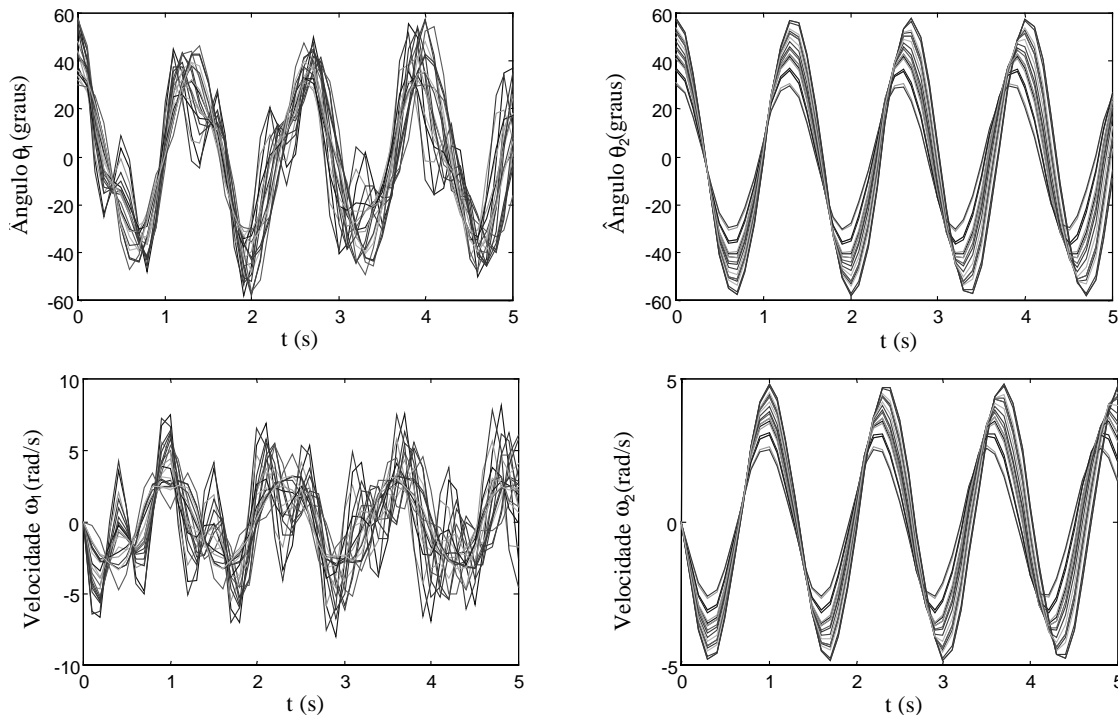


Figura 4: Dados de treinamento do pêndulo duplo.

Para o treinamento foi previsto a realização de super épocas que possam assegurar um treinamento completo para essas condições. Após o treinamento a RNA pode identificar ou reproduzir a dinâmica do sistema não linear a partir de uma condição inicial dentro do intervalo considerado no treino.

Tem sido proposta uma arquitetura de rede neural de duas camadas escondidas com 50 neurônios cada e mais 4 neurônios na camada de saída, 8 delays para cada unidade de saída realimentada à correspondente unidade de entrada (ver Figura 3). Ambas as posições angulares θ_1 e θ_2 , e as velocidades angulares ω_1 e ω_2 , têm sido as correspondentes às unidades de saída.

Durante o processo de treinamento foram usadas 20 condições iniciais aleatórias dentro do intervalo de 30 a 60 graus. Como critério de convergência foi fixado o erro quadrático médio como sendo $1e-6$ para todos os 50 pontos de treinamento considerados dentro do intervalo de tempo predefinido de 5 segundos.

Na Figura 4 são mostrados os dados de treinamento dos estudos de simulação no sistema de teste do pêndulo duplo, levando em conta todos os dados de velocidades e posições angulares para o treinamento. Esses gráficos foram obtidos para 20 condições iniciais aleatórias dentro do intervalo entre 30 e 60 graus. O tempo de simulação total para aplicação da rede neural treinada foi 10 s. Este fato justifica-se devido ao movimento periódico do pêndulo duplo.

Por outra parte, na Figura 5 são mostrados os erros quadráticos indicados em porcentagem para as posições e velocidades angulares. Esses erros dão uma indicação de que alguns passos futuros, a rede neural foi capaz de prever a dinâmica do pêndulo duplo com uma precisão aceitável.

Em vista que foram considerados 8 atrasos (0.7 s), este período não apresenta erro, depois dele o erro vai aumentando em cada predição.

Simulações estendidas e usando diferente número de atrasos indicam que um maior número de atrasos melhora o comportamento da RNA treinada, para a predição de um maior número de passos futuros. Já o uso de poucos atrasos não melhora a capacidade de predição da rede neural de mais passos futuros. Redes neurais com um maior número de neurônios melhoram o treinamento, porém existem limitações para redes grandes. As características e considerações do algoritmo de treinamento back-propagation padrão são válidas para o método teacher forcing. [2, 6, 7, 8]

5. A dinâmica do motor de indução

O algoritmo BPTF será testado com o sistema dinâmico do motor de indução. Os dados da simulação entrada-saída usados na identificação são obtidos com o modelo de terceira ordem do motor, o qual se obtém a partir da modelo de quinta ordem desprezando os transitórios do estator. A seguir a representação matemática desse modelo de terceira ordem da carga motor de indução [9, 10]:

$$T_o' \frac{dE_q'}{dt} = -\frac{X'}{X} E_q' + \frac{X - X'}{X'} V + T_o' w_s (v - 1) E_d' \quad (7)$$

$$T_o' \frac{dE_d'}{dt} = -\frac{X'}{X} E_d' - T_o' w_s (v - 1) E_q' \quad (8)$$

$$2H \frac{dv}{dt} = \frac{VE_d'}{X'} - Tm \quad (9)$$

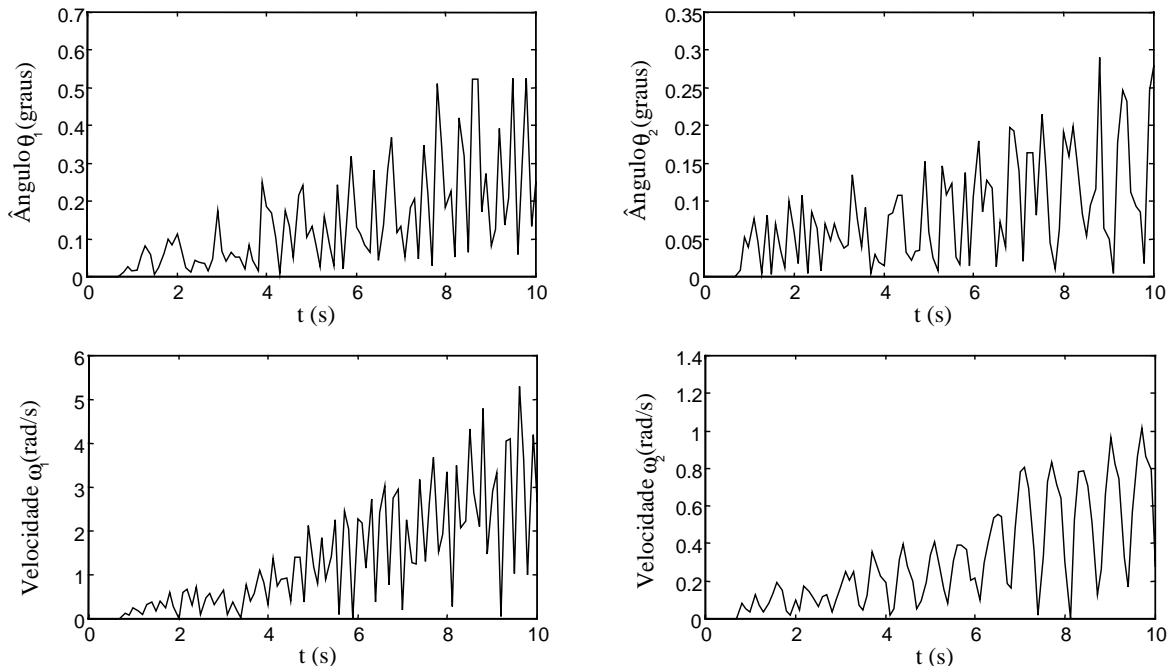


Figura 5: Erro simples do teste da RNA para o pêndulo duplo.

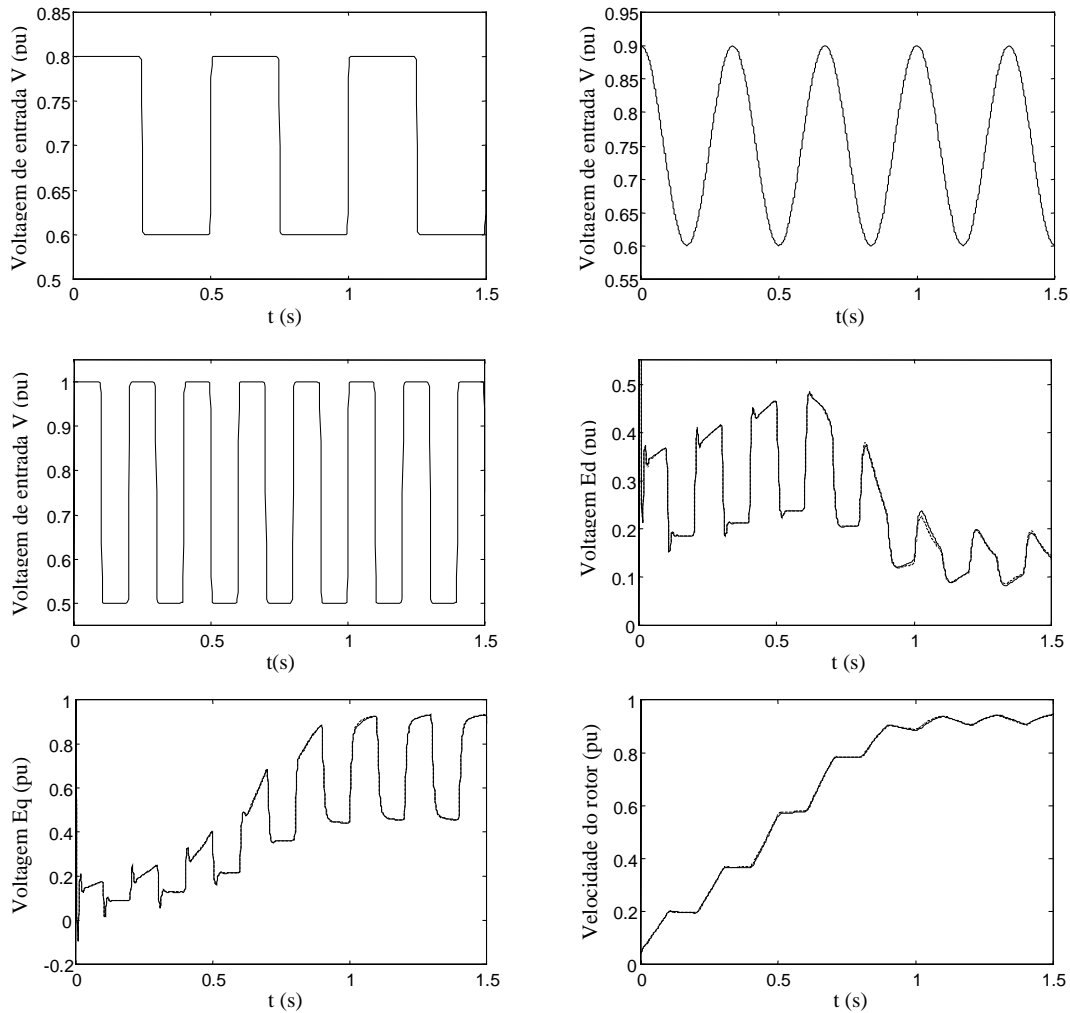


Figura 6: Dados de treinamento para a máquina de indução.

Nas equações (7) a (9) v é a velocidade normalizada do rotor ($v = \omega/\omega_s$), ω_s é a velocidade síncrona (377 rad/s). A constante de inércia do rotor é representada por H , T_o' é a constante de tempo transitória a circuito aberto, X , X' são as reatâncias síncrona e transitória.

As variáveis da tensão do rotor nos eixos d-q são representadas por E_d' e E_q' . Tm é o torque mecânico e V é a tensão terminal. Os parâmetros do motor são $X = X' = 2.36$ pu (per unit), $H = 0.7916$ s, $T_o' = 0.1557$ s, $Tm = 1.0$ pu.

No modelo da rede neural para representar a máquina de indução de terceira ordem, a tensão de entrada V , as tensões E_d' , E_q' e a velocidade v do rotor são as entradas no modelo da rede neural, além da tensão aplicada no instante atual. As saídas da rede neural são as tensões rotóricas nos eixos d-q e a velocidade normalizada do rotor.

A estrutura da rede que representa a máquina de indução de terceira ordem é uma rede neural de perceptrons multicamada com 32 entradas, 3 camadas escondidas de 25 neurônios cada, 8 atrasos para cada unidade de saída realimentada à correspondente unidade de entrada (ver Figura 3). Três tensões de excitação V diferentes e plotadas na Figura 6 tiveram de ser

aplicadas ao motor de indução para obter os dados para o treinamento da rede neural.

As respostas da RNA e o modelo de terceira ordem do motor estão incluídos na Figura 6 e Figura 7. Na Figura 6 mostram-se as tensões e a velocidade do rotor para a terceira tensão de excitação V aplicada para o treinamento. A curva sólida representa a resposta do modelo de terceira ordem e a curva restante indica a resposta do modelo da rede neural.

Após o treinamento a RNA representando o motor de indução foi testada pela tensão de excitação V (ver Figura 7). Esse tipo de tensão não foi considerado no processo de treinamento da rede neural artificial. As simulações do modelo de terceira ordem da máquina de indução e da rede neural se mostram na Figura 7.

Analisando os resultados muito próximos obtidos com o modelo de terceira ordem considerando as equações diferenciais, e os resultados do modelo do motor usando redes neurais pode-se assegurar a eficácia da metodologia proposta baseada em RNA com teacher forcing para efeitos de identificação.

Esse modelo proposto do motor de indução usando redes neurais pode capturar a dinâmica dessa máquina elétrica para diversas condições de operação.

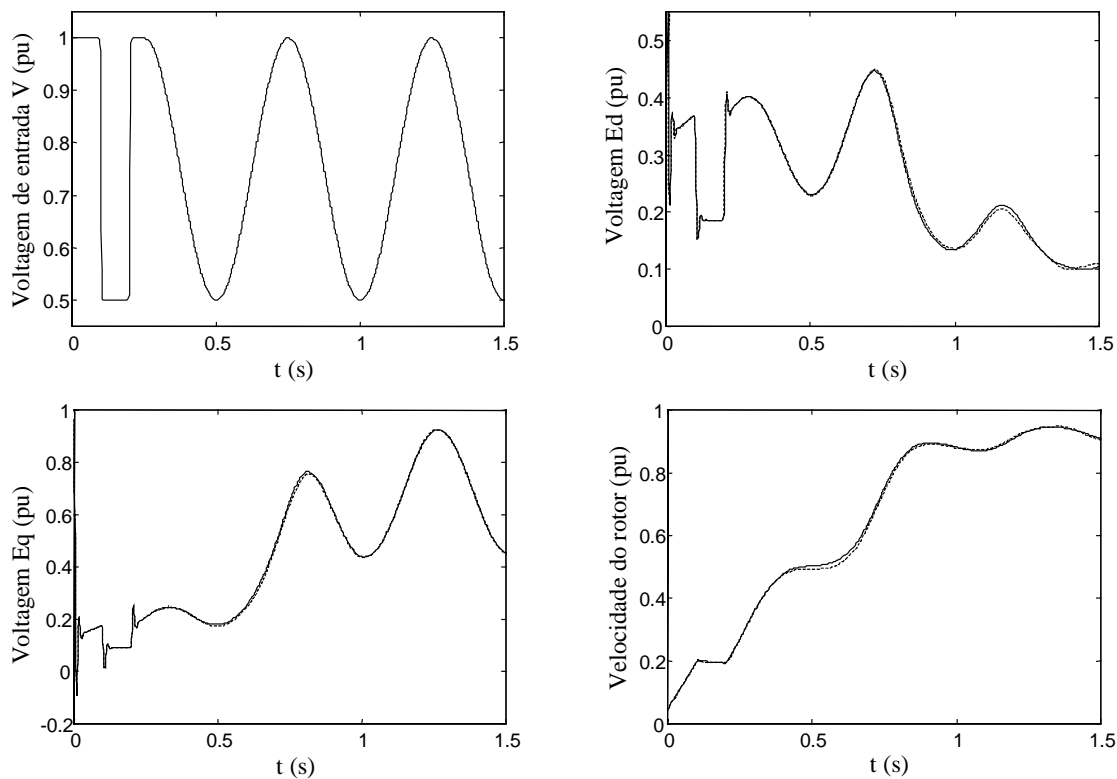


Figura 7: Resultados da máquina de indução considerando a identificação com RNA e com as equações diferenciais

6. Conclusões

O treinamento das redes neurais artificiais com backpropagation adotando o procedimento teacher forcing permite identificar a dinâmica de sistemas dinâmicos não lineares.

Os resultados dos testes realizados usando o sistema dinâmico do pêndulo duplo e da carga dinâmica do tipo motor de indução de terceira ordem, revelam que a identificação e a simulação usando o método BPTF podem ser muito eficientes principalmente durante os primeiros passos da identificação e após o período de tempo usado durante o treinamento da RNA.

Esses resultados indicam as vantagens do teacher forcing o qual deveria ser usado nas diversas áreas de controle e identificação dentro da engenharia elétrica.

As redes neurais do tipo backpropagation e com teacher forcing foram treinadas em forma melhorada com o método de segunda ordem Levenberg-Marquardt com o qual se consegue um treinamento com menos épocas com tempos de processamento bem menores que as versões do backpropagation e com uma maior precisão nos resultados.

7. Agradecimentos

Os autores agradecem às instituições brasileiras: CAPES (Coordenação de Aperfeiçoamento Superior) e CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), pelo suporte.

Referências

- [1] Chu S.R., Shoureshi R. and Tenório M., "Neural Networks for System Identification", IEEE Control Systems Magazine, pp. 31-35, April 1990.
- [2] Narendra K.S. and Parthasarathy K., "Identification and Control of Dynamical Systems Using Neural Networks", IEEE Transactions on Neural Networks, Vol.1, No.1, pp. 4-27, March 1990.
- [3] Fausett L., "Fundamental of Neural Networks. Architectures, Algorithms and Applications", Prentice-Hall, 1994.
- [4] Haykin S., "Neural Networks: a comprehensive foundation", 2nd.ed., Prentice Hall, 1998.
- [5] Hertz J.A., Krogh A. and Palmer R.G., "Introduction to the Theory of Neural Computation", Addison-Wesley Pub.Co., USA, 1991.
- [6] Pearlmutter B.A., "Dynamic Recurrent Neural Networks", Carnegie Mellon University, USA, 1990.
- [7] El-Sharkawi M. and Niebur D., "Artificial Neural Networks with Applications to Power Systems", IEEE PES special publication 96 TP 112-0, 1996.
- [8] Page G.F., Gomm J.B. and Williams D., "Application of Neural Networks to Modeling and Control", London: Chapman and Hall Limited, UK, 1993.
- [9] Bostanci M., Koplowitz J. and Taylor C.W., "Identification of Power System Load Dynamics Using Artificial Neural Networks", IEEE Transactions on Power Systems, Vol.12, No.4, pp.1468-1473, Nov. 1997.
- [10] Lesieutre B.C., Sauer P.W. and Pai M.A., "Development and Comparative Study of Induction Machine Based Dynamic P,Q, Load Models", IEEE Transactions on Power Systems, Vol.10, No.1, pp.182-191, Feb. 1995.