



DINCON 2017

CONFERÊNCIA BRASILEIRA DE DINÂMICA, CONTROLE E APLICAÇÕES
30 de outubro a 01 de novembro de 2017 – São José do Rio Preto/SP

Modelagem de um Sistema Dinâmico e Não Linear por meio de Redes Neurais Artificiais MLP e RBF desenvolvidas em MATLAB®.

Eduardo Vilela Pierangeli¹

Graduação em Engenharia de Controle e Automação
Departamento de Engenharia, UFLA, Lavras, MG

Jordann Alessandro Rosa Almeida²

Graduação em Engenharia de Controle e Automação
Departamento de Engenharia, UFLA, Lavras, MG

Wilian Soares Lacerda³

Departamento de Ciência da Computação, UFLA, Lavras, MG

Resumo. Este trabalho aborda a modelagem matemática de um sistema fluídico de controle de vazão, dinâmico e não linear composto por um tanque de processo e um reservatório, possuindo ainda, uma bomba de fluxo constante, uma válvula pneumática e um sensor de vazão. Pretendeu-se obter o modelo desta planta por meio do uso de uma rede neural artificial do tipo MLP (*Multi Layer Perceptron*) e outra do tipo RBF (*Radial Bases Function*), ambas desenvolvidas e treinadas utilizando-se do MATLAB®. Os resultados obtidos demonstram a eficácia da modelagem utilizando as redes neurais artificiais propostas, principalmente no que diz respeito ao mapeamento da dinâmica e não linearidade presentes nesta planta.

Palavras-chave. Modelagem, Sistemas dinâmicos não lineares, Redes Neurais Artificiais.

1 Introdução

Na área de controle de processos, é importante a obtenção de um modelo funcional do sistema em questão. Atualmente existem vários métodos consolidados, mas ainda assim matemáticos vêm desenvolvendo diversos trabalhos voltados para a realização dessa tarefa,

¹ evp.pierangeli@gmail.com, evpierangeli@engautomacao.ufla.br

² jordann.alessandro@gmail.com

³ lacerda@dcc.ufla.br

buscando representar o comportamento real do sistema, com ponderação entre acurácia e simplicidade [1].

Para a escolha da técnica de modelagem a ser empregada, deve-se levar em conta algumas características do sistema, como sua dinâmica, linearidade, variância temporal, entre outros. Sistemas dinâmicos são aqueles em que as saídas não respondem somente ao sinal de entrada aplicado, mas dependem também dos estados anteriores da planta, nesse caso o modelo utiliza equações diferenciais. Em geral sistemas que envolvem grandezas como temperatura, nível e pressão são dinâmicos, justificando o interesse na modelagem desses sistemas [7].

A partir de determinados conjuntos de sinais aplicados nas entradas e colhidos nas saídas de um sistema durante um intervalo de tempo, é possível inferir um modelo matemático do mesmo, relacionando a entrada com a saída [1]. Este modelo pode ser expresso de diversas formas, como uma equação diferencial, função de transferência ou uma rede neural artificial (RNA) [7].

As redes neurais artificiais possuem a capacidade de mapeamento entrada saída, sendo considerados aproximadores universais de função, sejam elas dinâmicas ou estáticas, contínuas ou descontínuas, lineares ou não. O que torna as RNA especialmente funcionais em aplicações de modelagem e controle [2,3,5].

Desta forma, este trabalho se estrutura no desenvolvimento e aplicação de dois diferentes tipos de redes neurais artificiais, objetivando encontrar modelos que representem fielmente o comportamento dinâmico da vazão de um sistema fluídico não linear, como forma de substituição à metodologias de modelagem mais complexas, a fim de facilitar a compreensão do comportamento da planta.

2 Descrição do Problema

Objetivou-se neste trabalho, obter um modelo neural que descreva o comportamento do sistema mostrado na figura 1, relacionando a vazão $y(t)$ com o percentual de abertura da válvula $x(t)$, baseado em dados reais obtidos por meio de ensaios controlados, que foram divididos em conjuntos de treinamento e de teste da rede. Apesar de ser utilizada uma planta didática, ela apresenta características bastante similares às plantas industriais existentes atualmente.

A modelagem de um sistema estático seria de grande simplicidade se realizada por meio de redes neurais artificiais, bastando para isso, uma rede com uma única entrada, uma única saída e com um número adequado de neurônios na(s) camada(s) escondida(s). Entretanto, como o sistema em questão é dinâmico, sua saída não depende apenas da entrada atual, mas também das entradas e saídas anteriores. Desta forma os estados anteriores deverão ser apresentados à rede na forma de entradas.

Outra característica importante na justificativa do uso de redes neurais artificiais na modelagem desta planta é sua não linearidade, uma vez que para a obtenção do modelo em função de transferência é necessário que o sistema seja linear ou que seja feita a linearização do sistema em torno de um ponto. Esta abordagem, em geral, apresenta bons resultados apenas para as proximidades do ponto de linearização, diferente do modelo neural que apresenta bons resultados ao descrever comportamentos não lineares.

3 Metodologia

Foi utilizada a bancada da fabricante AMATROL, modelo T5552, e sua representação é ilustrada na figura 1. A base de dados para a modelagem foi obtida por meio de ensaios, nos quais foram aplicados diferentes degraus de percentual de abertura na válvula do sistema e coletadas suas respectivas respostas em vazão.

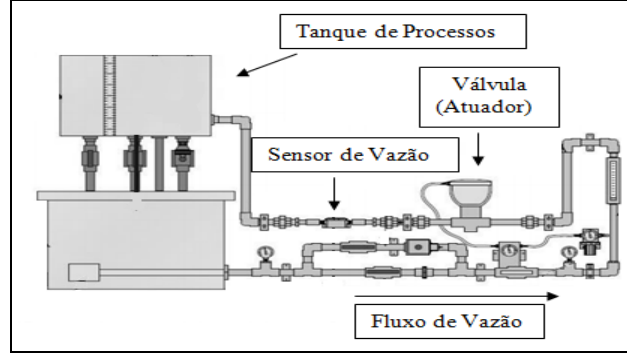


Figura 1: Representação do sistema.

Após a aquisição dos dados, foi feita a normalização dos resultados utilizando a seguinte equação [5]:

$$X_N = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (3.1)$$

Onde X são os dados reais e X_N são os dados normalizados.

Os conjuntos de amostras resultantes dos ensaios possuem a forma do par de Equações 3.2, em que n é o número de amostras do conjunto [6].

$$\begin{cases} x(t) = [x_1 \ x_2 \ x_3 \ \dots \ x_n] \\ y(t) = [y_1 \ y_2 \ y_3 \ \dots \ y_n] \end{cases} \quad (3.2)$$

Porém, como o sistema possui uma dinâmica temporal, ou seja, sua saída depende de seus estados anteriores, é necessário que estes sejam apresentados à RNA como entradas. É preciso ainda determinar o intervalo dos atrasos para os estados, assim, os dados de entradas devem estar na forma do par de equações 3.3 [1,5,6].

$$\text{Entrada} = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_{n-\alpha k} \\ x_{1+k} & x_{2+k} & x_{3+k} & \dots & x_{n-(\alpha-1)k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{1+\alpha k} & x_{2+\alpha k} & x_{3+\alpha k} & \dots & x_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_1 & y_2 & y_3 & \dots & y_{n-\alpha k} \\ y_{1+k} & y_{2+k} & y_{3+k} & \dots & y_{n-(\alpha-1)k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{1+\alpha k} & y_{2+\alpha k} & y_{3+\alpha k} & \dots & y_n \end{bmatrix} \quad (3.3)$$

$$\text{Saída} = [y_1 \ y_2 \ y_3 \ \dots \ y_{n-\alpha k}]$$

Onde k representa o intervalo dos atrasos entre os estados, n é o numero total de amostras e α é o número de amostras em atraso utilizadas.

Após este pré-processamento dos dados, foi dado início ao projeto da rede MLP. A arquitetura proposta para a RNA é ilustrada na figura 2, onde as entradas são as aberturas da válvula e as vazões de entrada no tanque (no instante atual e nos αk instantes anteriores) e a saída é a vazão no próximo instante.

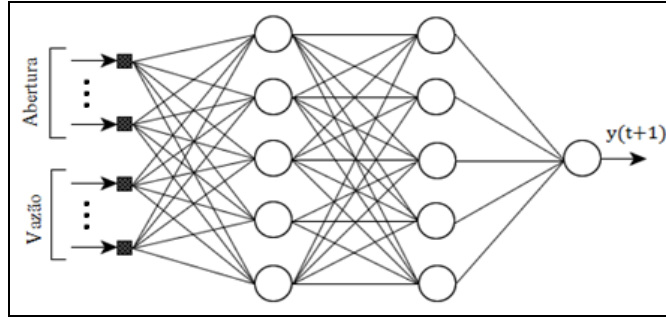


Figura 2: Arquitetura da MLP.

O algoritmo utilizado foi o *feedforward Backpropagation* Levenberg-Marquardt. O número de neurônios nas camadas ocultas, o número de amostras em atraso e a janela do atraso foram determinados empiricamente [3,7].

Para a RBF foi utilizada a arquitetura mostrada na figura 3. Os centros das funções de base radial, o número de neurônios RBF e o erro de treinamento foram determinados empiricamente por meio de análise de desempenho [4].

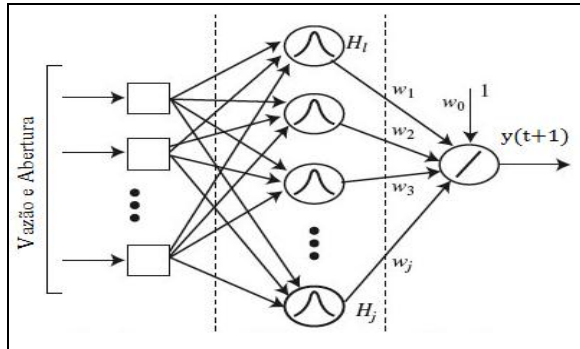


Figura 3: Arquitetura da RBF.

4 Resultados

Para verificar a eficácia da metodologia proposta, foram utilizados dois conjuntos de dados, adquiridos por meio da aplicação de degraus com diferentes amplitudes (variando de 0% a 100% de abertura) na válvula, e coletando as respostas da vazão em tensão (0V a 5V), obtendo-se assim, um conjunto composto por 900 amostras, utilizado para o treinamento das redes e outro, composto por 700 amostras, utilizado para o teste das redes, conforme ilustrado na figura 4.

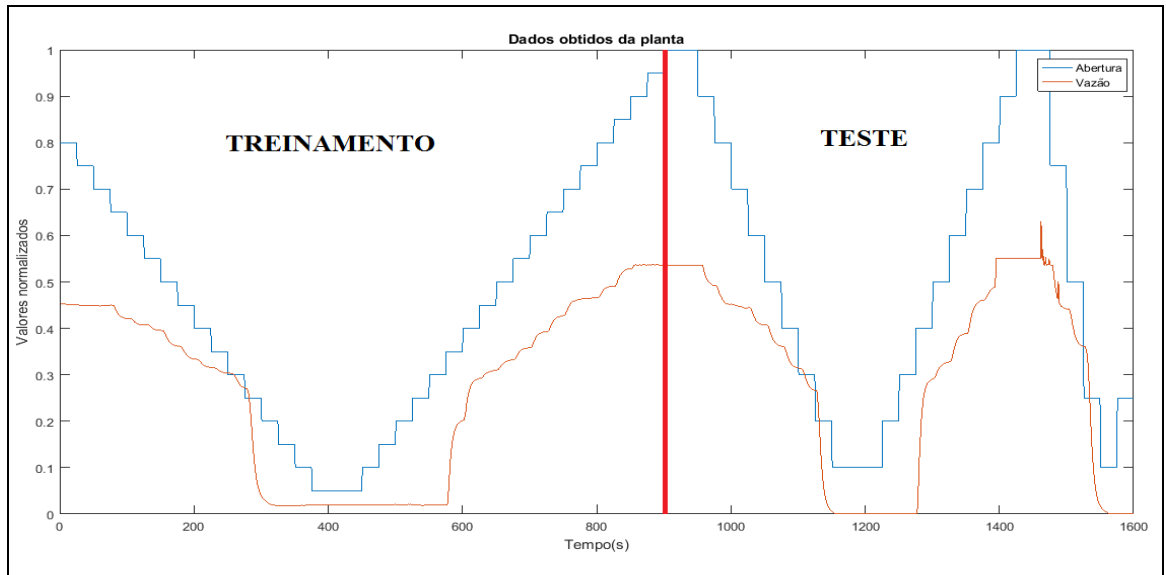


Figura 4: Conjunto dos dados normalizados utilizado.

Uma vez coletados os dados, foi utilizada a *toolbox* “*nntool*” do MATLAB® para realizar o treinamento e validação das redes. A rede MLP recebeu em sua arquitetura 10 entradas (vazão e abertura atuais e nos 24 instantes anteriores espaçados por uma janela de 6 instantes), 20 neurônios na primeira camada escondida, 20 na segunda e 1 na camada de saída. O método de treinamento utilizado foi o Levenberg-Marquardt já implementado no toolbox, os parâmetros utilizados foram os valores padrões do método e as funções de ativação foram tangentes hiperbólicas para as camadas escondidas e linear para a camada de saída.

A RBF foi desenvolvida utilizando-se funções gaussianas, foi treinada por meio do algoritmo padrão do “*nntool*” e recebeu como parâmetros um erro quadrático médio (EQM) de treinamento alvo de 10^{-12} e uma variância dos dados de 0.9. Este treinamento resultou em uma rede com 315 neurônios RBF.

As figuras 5 e 6 apresentam o comportamento convergente do EQM de treinamento para as redes MLP e RBF respectivamente, com duração de aproximadamente 30 segundos para a MLP e 44 segundos para a RBF, ambas em um computador com processador *Core i5 2450M* 2.50 GHz, 4 GB de memória RAM.

Para realização dos testes de desempenho foram aplicados os dados separados para teste, até então não apresentados às redes. Os resultados obtidos pelas redes MLP e RBF treinadas, são mostrados na figura 7.

A fim de determinar um desempenho numérico para as redes, foi calculado o valor R^2 , que mensura a semelhança entre os resultados obtidos e esperados, por meio da função “*corr2*” do MATLAB®. Os resultados obtidos foram de 0,999858 para a MLP e 0,9999779 para a RBF, mostrando ligeira superioridade, em termos de grau de correlação, do segundo modelo para o primeiro.

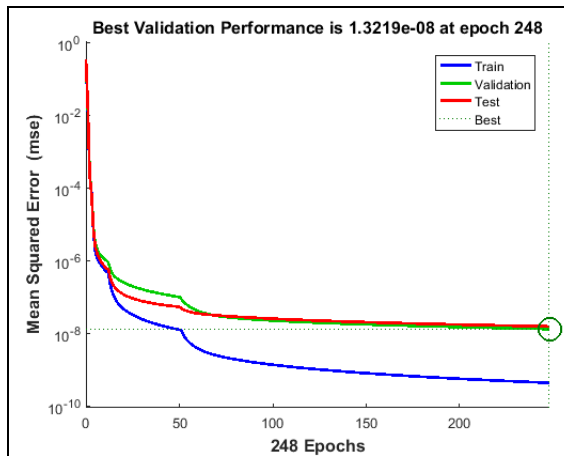


Figura 5: Erro quadrático médio MLP.

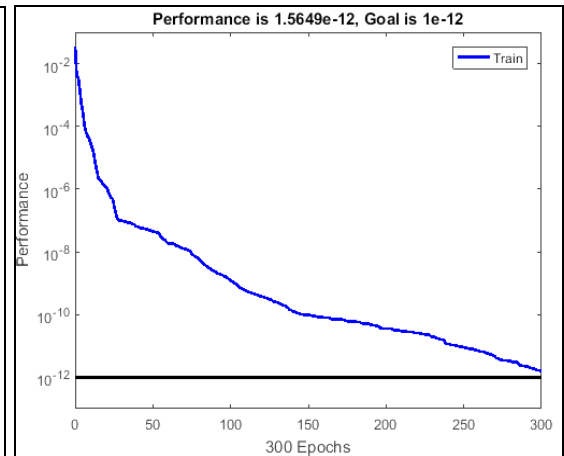


Figura 6: Erro quadrático médio RBF.

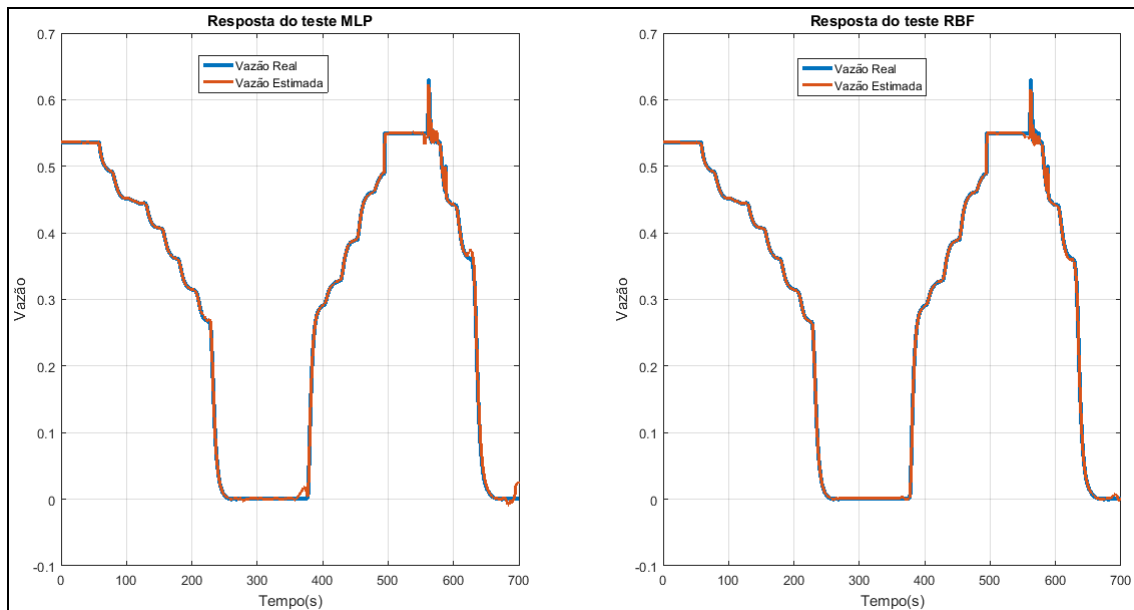


Figura 7: Respostas aos dados de teste das duas redes.

5 Conclusão

Com base nos resultados mostrados, a utilização de RNAs dos tipos MLP e RBF para a modelagem de um sistema dinâmico e não linear apresentou resultados bastante satisfatórios. Apesar da estimação de dados feita pela rede poder acumular erros ao longo do tempo, a validação realizada foi feita em um ensaio de 11,67 minutos, sem apresentar erros significativos, obtendo em ambos os casos, um R^2 muito próximo de 1, demonstrando confiabilidade dos modelos [7].

Apesar do desempenho ligeiramente superior da RBF em relação à MLP, o uso de ambas as técnicas pode ser considerado, pois o tempo de treinamento da MLP mostrou-se um pouco inferior ao da RBF. Desta forma, não é possível inferir sobre qual delas é a mais indicada para a solução do problema em questão.

Além disso, a simplicidade de implementação e uso das RNA para modelagem matemática, em alguns casos, viabiliza sua utilização em substituição à outras técnicas de modelagem, como função de transferência por exemplo [5]. Assim, alterando-se apenas alguns parâmetros, é possível que se utilize essas arquiteturas de RNA para modelagem e previsão de estados de diversos sistemas [6], tornando-as especialmente úteis para projeto de controladores que dispensam o uso da função de transferência, como é o caso da técnica de Ziegler-Nichols [5,7].

Agradecimentos

Os autores agradecem à Universidade Federal de Lavras, à Pró-reitoria de Pesquisa, ao Laboratório de Controle do Departamento de Engenharia e a FAPEMIG pelo apoio no desenvolvimento e publicação deste trabalho.

Referências

- [1] L. A. Aguirre , Introdução à Identificação de Sistemas. 1. ed. Belo Horizonte: UFMG, 2007.
- [2] A. P. Braga, A. P. Leon e T. B. Ludemir, Redes Neurais Artificiais: Teoria e Aplicações. 2. ed. Belo Horizonte: LTC, 2007.
- [3] B. M. T. Cargas, Controle De Nível De Um Sistema De Tanques Acoplados Utilizando Redes Neurais Artificiais, UFOP, 2013.
- [4] L. S. Coelho, Modelagem de Comportamento Dinâmico de Um Sistema Térmico Baseada em Rede Neural com Treinamento Usando Filtro de Kalman Evolutivo, Braz. Soc. of Mechanical Sciences and Engineering, Curitiba, 2006.
- [5] I. M. Costa, L. L. de Almeida, S. N. D. Lisboa, F. M. U. de Araujo, Redes Neurais Artificiais para Controle de uma Planta de Nível, VII Congresso Brasileiro de Redes Neurais, Campos do Jordão, 2005.
- [6] I. N. da Silva, D. H. Spatti e R. A. Flauzino, Redes Neurais Artificiais para engenharia e ciências aplicadas. 1. ed. São Paulo: Artliber, 2010.
- [7] D. C. Viana e W. S. Lacerda, Modelagem de um sistema dinâmico utilizando rede neural artificial implementada em SCILAB, Conferência Brasileira de Dinâmica, Controle e Aplicações, 2015. Natal, 2015.