

# IDENTIFICAÇÃO DE SISTEMAS NÃO LINEARES EM ESPAÇO DE ESTADOS UTILIZANDO REDES NEURAIIS ARTIFICIAIS

## IDENTIFICATION OF NONLINEAR SYSTEMS IN STATE SPACE UTILIZING ARTIFICIAL NEURAL NETWORKS

Rômulo Thiago Silva da Rosa<sup>1</sup>

Paulo Jefferson Dias de Oliveira Evald<sup>2</sup>

Guilherme Brunel Zaffari<sup>3</sup>

Paulo Lilles Jorge Drews-Jr<sup>4</sup>

Silvia Silva da Costa Botelho<sup>5</sup>

Rodrigo Zelir Azzolin<sup>6</sup>

**Resumo:** A identificação de sistemas é método fundamental para o controle e análise de um sistema dinâmico. Assim, o objetivo deste trabalho é apresentar uma metodologia de identificação de sistemas com uma única entrada e uma única saída, utilizando uma rede neural recorrente baseada no modelo de *Wiener*. O processo de identificação é realizado em duas etapas: primeiramente, é definido a ordem do modelo, através dos quocientes de *Lipschitz* e em seguida, a rede neural ajusta seus pesos para modelar as dinâmicas e as não linearidades do sistema dinâmico que está observando. Além disso, a identificação de sistemas não requer conhecimento dos parâmetros do sistema a ser modelado e apresenta os pesos em uma representação no espaço de estados. Para provar a eficácia do método proposto, essa rede neural foi treinada com dados de um sistema de fluídos, obtendo assim um erro quadrático médio de apenas  $19,18 \times 10^{-4}$  com os dados de validação.

**Palavras-chave:** Identificação de Sistemas, Sistemas Não-lineares, Redes Neurais Artificiais, Rede Neural Recorrente baseada no Modelo de *Wiener*, Quocientes de *Lipschitz*.

**Abstract:** System identification process is elementary for control and analysis of a process. This work aims to present a system identification approach for a single-input single-output system utilizing a Wiener-type recurrent neural network. The identification process consists in two steps: firstly, it is to define the model order using Lipschitz quotients and second, the Wiener-type neural network learns the process dynamics and their nonlinearities. This approach presents the neural network weights in space state representation. Moreover, this approach does not require any prior knowledge about modeled system. A proof of effectiveness of

---

<sup>1</sup> Graduando em Engenharia de Automação, Universidade Federal do Rio Grande, Brasil, romulothiago@gmail.com.

<sup>2</sup> Mestrando em Engenharia de Computação, Universidade Federal do Rio Grande, Brasil, paulo.evald@gmail.com.

<sup>3</sup> Mestre em Engenharia de Computação, Universidade Federal do Rio Grande, Brasil, guilhermezaffari@furg.br.

<sup>4</sup> Doutor em Ciência da Computação, Universidade Federal do Rio Grande, Brasil, paulodrews@furg.br.

<sup>5</sup> Doutora em Informática e Telecomunicações, Universidade Federal do Rio Grande, Brasil, silviacb@furg.br.

<sup>6</sup> Doutor em Engenharia Elétrica, Universidade Federal do Rio Grande, Brasil, rodrigoazzolin@gmail.com.

proposed approach is given applying it in a fluid control system, obtaining  $19.18 \times 10^{-4}$  as mean squared error using validation data.

**Keywords:** System Identification, Nonlinear Systems, Artificial Neural Network, Wiener-type Recurrent Neural Network, Lipschitz Quotients.

## 1 INTRODUÇÃO

A obtenção de modelos matemáticos para representar sistemas dinâmicos numericamente pode ser realizada por duas maneiras bem distintas: através de uma modelagem analítica, quando as dinâmicas do sistema são conhecidas e descritas detalhadamente através de suas relações físicas ou através de uma modelagem caixa preta, quando essas não são conhecidas. Para desenvolver um modelo através da modelagem caixa preta, faz-se relação numérica entre as entradas e saídas do sistema dinâmico. Ou seja, aplica-se diversas entradas ao sistema e avalia-se suas respectivas saídas para formular um modelo matemático que descreva a relação entre entrada e saída nesse sistema. Para realizar a modelagem caixa preta são utilizados algoritmos de aprendizado para identificar a relação entre a saída e a entrada do sistema em que o algoritmo está observando. Assim, pode-se afirmar que a modelagem caixa preta é um processo de identificação de sistemas.

A identificação de um sistema é parte fundamental na análise e controle de sistemas dinâmicos (NELLES, 2013). A definição de um modelo matemático é importante para a performance de um controlador, porém, a tarefa de modelar analiticamente um sistema dinâmico nem sempre é trivial: são necessárias técnicas complexas e conhecimento prévio do sistema para obter um modelo que represente um processo da forma mais próxima possível do processo físico. Existem diversas abordagens para identificação de sistemas na literatura, dentre elas: o algoritmo de identificação *Ho-Kalman-Kong* (LIMA; BARROS, 2013), redes neurais com função radial para identificação de sistemas caóticos (GUERRA; SIERAKOWSKI; COELHO, 2005), o método de subespaços (RICCO et al., 2011), dentre várias outras que podem ser achadas na literatura.

O método de identificação de sistemas orientada à blocos divide o sistema à ser modelado em um subsistema linear e um subsistema não linear. O subsistema linear representa as dinâmicas lineares do sistema, enquanto o subsistema não linear representa as não linearidades desse sistema (GIRI,

2010). Os modelos de *Wiener* (WIENER, 1966) e *Hammerstein* (HAMMERSTEIN, 1930) são modelos orientados à blocos e provaram sua eficácia para modelar sistemas não lineares com alto grau de complexidade. Esses métodos são utilizados para diversos tipos de sistemas, tais como: sistemas caóticos (CHEN; CHEN; OGMEN, 1997), sistemas não-lineares (DEMPSEY; WESTWICK, 2004), sistemas hidráulicos (KNOHL; UNBEHAUEN, 2000), sistemas químicos (AREFI et al., 2008, ZHOU; LI; PAN, 2013), e também na área médica (SLOUN et al, 2015).

Redes neurais artificiais, em sua definição básica, são camadas de unidades de processamento interconectadas por pesos. A função de uma rede neural é aprender, por repetição, uma função matemática que represente a saída em função das suas entradas. Recentes avanços em diversas áreas demonstraram a eficácia de utilizar redes neurais como estimadores de estados (HAARNOJA et al., 2016), localização (KENDALL; GRIMES; CIPOLLA, 2015) e modelagem de sistemas dinâmicos (PUNJANI; ABBEEL, 2015).

A partir da capacidade de uma rede neural e a necessidade de um modelo matemático interpretável, será apresentado neste trabalho uma abordagem de identificação de sistemas do tipo SISO (*Single-Input Single-Output*), utilizando redes neurais recorrentes baseada no modelo de *Wiener*. Neste método, a identificação de sistemas é totalmente automática, sendo possível representar os pesos da rede neural em espaço de estados.

O presente trabalho é organizado da seguinte forma: na seção 2 é apresentado o modelo de *Wiener*, em seguida na seção 3 é apresentada as equações para encontrar a ordem do modelo que representa sistema através dos quocientes de *Lipschitz*. Na seção 4 são introduzidos os conceitos de redes neurais artificiais e adiante na seção 5 é demonstrado o método de obtenção de equação de espaço de estados utilizando redes neurais recorrentes do tipo *Wiener*. Finalmente, na seção 6 é demonstrada uma aplicação do método estudado em um sistema de fluídos e na seção 7 são dadas as considerações finais.

## 2 MODELO DE WIENER

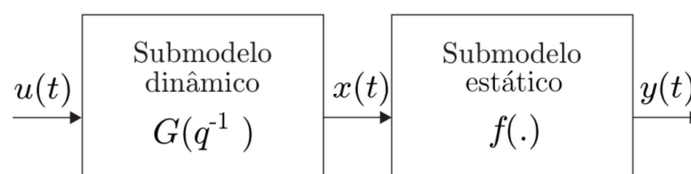
O modelo de *Wiener* é um modelo orientado a blocos (GIRI; BAI, 2010), onde cada bloco é interpretado como um submodelo do sistema dinâmico que está sendo identificado. A construção do modelo de *Wiener* consiste em um bloco linear dinâmico conectado a um bloco não linear estático em cascata. A organização dos blocos no modelo de *Wiener* modela os efeitos da não linearidade dos sensores na saída do sistema (GIRI; BAI, 2010). Por outro lado, a conexão dos blocos em ordem inversa (o bloco não linear estático seguido do bloco linear dinâmico) modela a não linearidade das entradas de controle e é conhecido na literatura como modelo de *Hammerstein*. O modelo de *Wiener* é apresentado na Figura 1 e representado pelas Equações (1) e (2).

$$x(t) = G(q^{-1})u(t) \quad (1)$$

$$y(t) = f(x(t)) \quad (2)$$

onde  $u(t)$ ,  $y(t)$  e  $x(t)$  são a entrada do sistema, a saída do sistema e o estado escondido do modelo de *Wiener*, respectivamente. Já  $G(q^{-1})$ ,  $f(\cdot)$  e  $q^{-1}$  denotam uma função linear dinâmica, uma função não linear estática e o operador de atraso, respectivamente, todos em função do tempo  $t$ .

Figura 1 - Modelo de *Wiener* para sistemas SISO



O submodelo dinâmico pode ser definido como modelo auto regressivo, modelo de *Box-Jenkins*, modelo de *Laguerre*, entre outros (LJUNG, 1999). Já o

submodelo estático pode ser definido como uma função de base radial, estimadores *kernel*, *wavelets*, entre outros (LIU, 2012).

### 3 DEFINIÇÃO DA ORDEM DE UM MODELO SISO

A ordem de um modelo é o atraso máximo do sistema, ou seja,  $q^{-1}$ . Além disso, quanto maior for a ordem de um sistema, mais complexas são as relações entre duas dinâmicas internas. Determinar a ordem ótima de um sistema é um fator importante na identificação de sistemas: um sistema modelado com a ordem inferior à real é incapaz de representar as dinâmicas do processo adequadamente e um sistema modelado com a ordem superior à real acarreta um esforço computacional desnecessário.

Em (HE; ASADA, 1993), foi desenvolvido um método para obter a ordem ótima de um sistema SISO através dos quocientes de *Lipschitz*. Para tal, os autores utilizam dados de entrada e saída do sistema analisado.

Considerando um sistema SISO genérico que pode ser representado como (3),

$$y(t) = g(y(t-1), \dots, y(t-p), u(t-1), \dots, u(t-r)) \quad (3)$$

onde  $p$ ,  $r$  e  $g(\cdot)$  são a ordem ótima da saída, referência e uma função não linear contínua e suave, respectivamente.

Reescrevendo a Equação (3) como (4),

$$y(t) = g(\phi_1, \phi_2, \dots, \phi_n) \quad (4)$$

sendo  $n = p + r$ . É definido o quociente de *Lipschitz* conforme Equação (5).

$$q_{ij} = \frac{|y(i) - y(j)|}{|\phi(i) - \phi(j)|}, \quad i \neq j \quad (5)$$

onde  $|\phi(i) - \phi(j)|$  é a distância entre dois pontos no espaço de entrada e  $|y(i) - y(j)|$  é a diferença entre  $g(\phi(i))$  e  $g(\phi(j))$ .

Para dados de entrada com pequenas distâncias entre  $|\phi(i) - \phi(j)|$ , o quociente de *Lipschitz* pode ser reescrito como demonstrado na Equação (6).

$$q_{ij}^{(p+r)} = \frac{|\partial y|}{\sqrt{(\partial \phi_1)^2 + \dots + (\partial \phi_n)^2}} \quad (6)$$

onde  $\partial y = y(i) - y(j)$  e  $\partial \phi_r = \phi_r(i) - \phi_r(j)$  para  $r = 1, \dots, n$ .

Para evitar o efeito de ruídos, o índice representado na Equação (7) é utilizado para determinar a ordem correta.

$$q^{(p+r)} = \left( \sqrt{n} \prod_{s=1}^m q^{(n)}(s) \right)^{\frac{1}{m}} \quad (7)$$

onde  $q^{(p+r)}(s)$  é o  $s$ -ésimo maior quociente de *Lipschitz* entre todos  $q_{ij}^{p+r}$  e o parâmetro  $m \in [0.01L \sim 0.02L]$ , sendo  $L$  o tamanho do conjunto de dados.

Em (HE; ASADA, 1993), foi afirmado que quando há ausência de uma das variáveis de entrada do sistema,  $q^{p+r-1}$  será muito maior que  $q^{p+r}$ . Porém, quando uma variável de entrada do sistema é redundante,  $q^{p+r+1}$  será pouco diferente de  $q^{p+r}$ . Essa afirmação é formalizada na Equação (8).

$$q^{p+r-1} \gg q^{p+r} \approx q^{p+r+1} \quad (8)$$

## 4 REDES NEURAIS ARTIFICIAIS

As rede neural artificial são um modelos computacionais que processam informação através de unidades de processamento, chamadas de neurônios. As unidades de processamento expressam uma saída que é uma função de suas entradas multiplicadas pelos pesos de suas conexões de entrada.

O objetivo de uma rede neural é aprender uma função matemática que mapeia os dados saída em relação aos dados de entrada da rede. Para aprender a função, a rede utiliza algoritmos para aproximar a saída desejada

da saída atual, modificando os pesos das conexões da rede baseado no erro entre a saída desejada e a saída atual (NELLES, 2013).

Em redes neurais multicamadas, as unidades de processamento são agrupadas em camadas e são associadas entre si usando conexões com pesos entre suas camadas anterior e posterior. As camadas presentes numa rede neural multicamada são a camada de entrada, as camadas intermediárias ou camadas escondidas, e a camada de saída (NELLES, 2013). Além disso, em redes neurais multicamadas, todas as variáveis de entrada e de saída são assumidas independentes entre si. Por outro lado, redes neurais recorrentes criam relações temporais entre suas variáveis através de neurônios com conexões de realimentação. As conexões de realimentação permitem a rede aprender relações dinâmicas entre a saída e a entrada da rede.

Na Figura 2 é apresentado uma rede neural multicamada em (a) e uma rede neural recorrente em (b), onde  $w_{cd}^{ab}$  é o peso da conexão entre os neurônios  $c$  e  $d$  das camadas  $a$  e  $b$ . Ambas contém uma camada escondida com dois neurônios, uma entrada e uma saída.

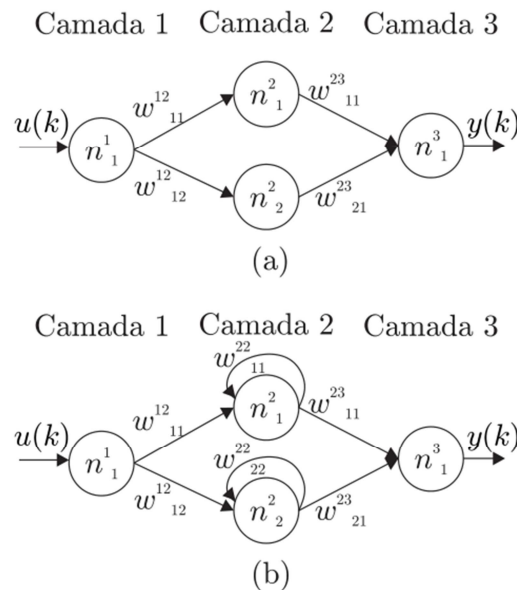
O algoritmo de aprendizagem *backpropagation* é um método de treinamento redes neurais. O algoritmo consiste na atualização dos pesos baseado em um método de otimização. O método de otimização gradiente descendente busca minimizar o erro entre a saída atual e saída desejada. Nas Equações (9) à (11) é apresentado o algoritmo *backpropagation*, onde  $\xi$ ,  $y_d(k)$ ,  $y(k)$  e  $e(k)$  são a taxa de aprendizado, a saída desejada, a saída da rede e a função de erro da rede, respectivamente, todos em tempo discreto  $k$ .

$$e(k) = \frac{1}{2} (y_d(k) - y(k))^2 \quad (9)$$

$$\Delta w_{cd}^{ab}(k) = \xi \frac{\partial e(k)}{\partial w_{cd}^{ab}} \quad (10)$$

$$w_{cd}^{ab}(k) = w_{cd}^{ab}(k-1) + \Delta w_{cd}^{ab}(k) \quad (11)$$

Figura 2 - Redes neurais artificiais: (a) multicamada (b) recorrente

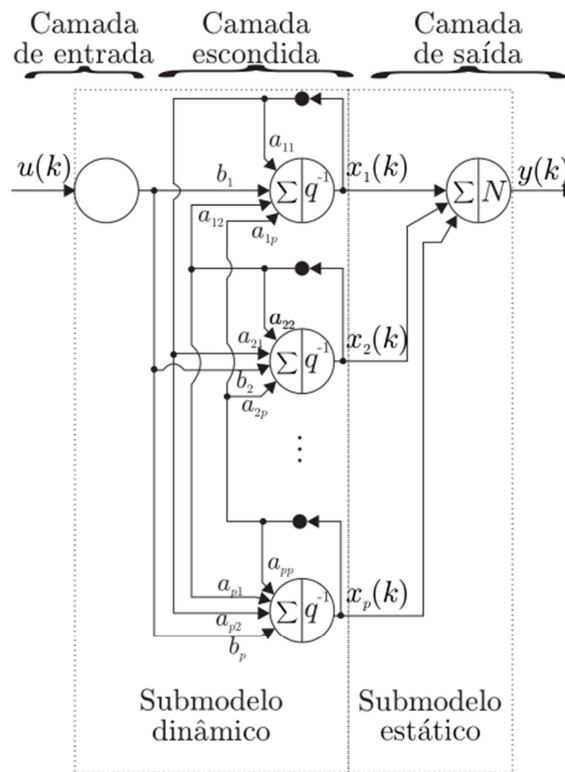


## 5 REDE NEURAL RECORRENTE TIPO WIENER

Em (HSU; WANG, 2009) foi proposto uma rede neural recorrente que integra os dois subsistemas do modelo de *Wiener*. Essa rede neural é composta por três camadas. A camada de entrada transmite os valores de entrada para rede. A camada escondida mapeia o espaço de entrada para o espaço de estados através de conexões recorrentes com operadores de atraso, tal camada representa o submodelo dinâmico do modelo de *Wiener*. Por fim, a camada de saída mapeia as variáveis de estado para as variáveis de saída através de uma transformação não linear, tal camada representa o submodelo estático do modelo de *Wiener*.

Essa rede neural é apresentada na Figura 3 com  $A = [a_{11} \dots a_{pp}]$  sendo os pesos recorrentes da camada escondida,  $B = [b_1 \dots b_p]$  os pesos da conexão entre a camada de entrada e a escondida e  $C = [c_1 \dots c_p]$  os pesos das conexões entre a camada escondida e a camada de saída.

Figura 3 - Rede neural baseada no modelo de *Wiener* para sistemas SISO



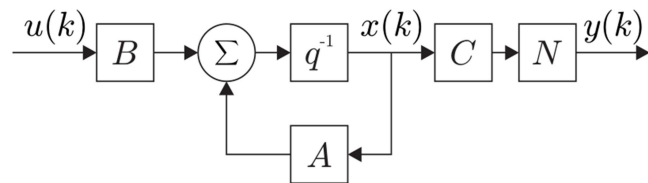
A rede neural recorrente baseada no modelo de *Wiener* é expressa em equações de estados como demonstrado na Equação (12) e na Figura 4.

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= N(Cx(k)) \end{aligned} \quad (12)$$

onde  $x = [x_1, \dots, x_p]^T$  é o vetor de estados,  $u = [u_1, \dots, u_{nu}]^T$  é o vetor de entrada,  $y = [y_1, \dots, y_{ny}]^T$  é o vetor de saída,  $N = [n_1, \dots, n_{ny}]^T$  é o vetor da função não linear, e  $p$ ,  $nu$  e  $ny$  são a quantidade de variáveis de estado, de entradas e saídas, respectivamente. Ainda, a matriz  $A$ , com dimensões  $R^{p \times p}$ , são os pesos das conexões de realimentação da camada recorrente, a matriz  $B$ , com dimensões  $R^{p \times nu}$ , são os pesos das conexões da camada de entrada com a camada escondida e a matriz  $C$ , com dimensões  $R^{p \times ny}$ , são os pesos da camada escondida com a camada de saída.

O presente trabalho trata sistemas SISO, logo  $nu = ny = 1$ .

Figura 4 - Analogia da rede neural da Figura 3 para o espaço de estados



As funções utilizadas na rede neural são lineares para as camadas de entrada e do tipo tangente hiperbólica para a camada escondida escondida, conforme mostrado na Equação (13), para a camada de saída. Devido a saída ser expressa por uma função tangente hiperbólica, os dados precisam ser normalizados para o intervalo  $[-1, 1]$ , cujo intervalo que a função tangente hiperbólica tem capacidade representativa.

$$\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1} \quad (13)$$

Em razão da rede neural ser expressa através das equações do espaço de estados, as características e análise do sistema, como observabilidade, controlabilidade e estabilidade tornam-se simples através dos métodos clássicos da teoria de controle moderno (OGATA; YANG, 1970).

## 6 EXPERIMENTO

A validação da rede neural recorrente baseada no modelo de *Wiener* implementada é organizada da seguinte forma:

1. Apresenta-se o sistema à ser modelado e o conjunto de dados de treinamento e validação;
2. Define-se a ordem do modelo através do quociente de *Lipschitz*;
3. A rede é treinada com a ordem obtida no passo anterior e as características do modelo obtido são analisadas.

## 6.1 Sistema não linear

O sistema SISO, dinâmico e não linear utilizado para treinar a rede neural foi obtido de (WIGREN, 1993). Tal sistema é representado na Equação (14).

$$x(k) = 1,4138x(k-1) - 0,6065x(k-2) + 0,1044u(k-1) + 0,0883u(k-2)$$
$$y(k) = \frac{0,3163x(k)}{\sqrt{0,1 + 0,9(x(k))^2}} \quad (14)$$

com  $u(k)$  representando o sinal de controle pneumático aplicado à uma haste e  $y(k)$  a posição da haste, com  $k$  representando o tempo discreto. Na Equação (14) é representado o equilíbrio dinâmico entre o sinal de controle, uma força de mola e a fricção contra-ativa.

Foram utilizados dois conjuntos de dados de tamanho  $1 \times 10^5$ , um conjunto de dados para treinamento da rede neural (conjunto de treinamento) e outro conjunto de dados para a validação do modelo matemático identificado (conjunto de validação), ambos generalizados no tempo discreto  $k$ , simulados por 1000 segundos e com intervalo de discretização de  $1 \times 10^{-2}$  segundos.

Tal abordagem visa testar a generalização da rede aqui apresentada, visto que os conjuntos de treinamento e validação são distintos. A rede neural será eficaz para aprender a dinâmica de (14) se apresentar baixo erro para um conjunto de dados não apresentado antes. Neste caso, utiliza-se então o conjunto de dados de validação.

## 6.2 Conjunto de dados

Conforme mencionado anteriormente, é necessário que haja um conjunto de dados para treinamento e um conjunto de dados para validação do modelo aprendido pela rede neural artificial.

O conjunto de dados de treinamento é representado pela Equação (15).

$$u(k) = \begin{cases} \text{i.i.d.}[1,-1], & 0 \leq k < 750 \\ \frac{1}{k} \cos(\pi k/30), & 750 \leq k < 1000 \end{cases} \quad (15)$$

sendo i.i.d (*Independent and identically distributed random variables*) as variáveis aleatórias independentes e identicamente distribuídas no intervalo de -1 à 1. Tal escolha visa gerar dados sem padrão para forçar a rede aprender apenas a dinâmica de (14).

O conjunto de dados de validação é constituído da função de validação obtida de (JUANG; LIN, 1999) e expressa na Equação (16). Esse conjunto de dados seve para avaliar se o modelo aprendido pela rede neural com o conjunto de treinamentos é fiel ao processo físico. Isto, o modelo é identificado com os dados de um conjunto de treinamento, realizando sua aprendizagem iterativamente até que o modelo apresente um comportamento dinâmico muito similar ao comportamento observado no conjunto de treinamento. A partir do momento em que esse modelo consegue representar de forma adequada o comportamento do sistema dinâmico inerentes ao conjunto de treinamento, então um novo conjunto de dados é simulado com novos comportamentos. Esse novo conjunto de dados é o conjunto de validação, pois se o modelo representar as novas dinâmicas adequadamente sem a necessidade de um novo conjunto de treinamento, admite-se então que o modelo matemático do sistema dinâmico está devidamente validado.

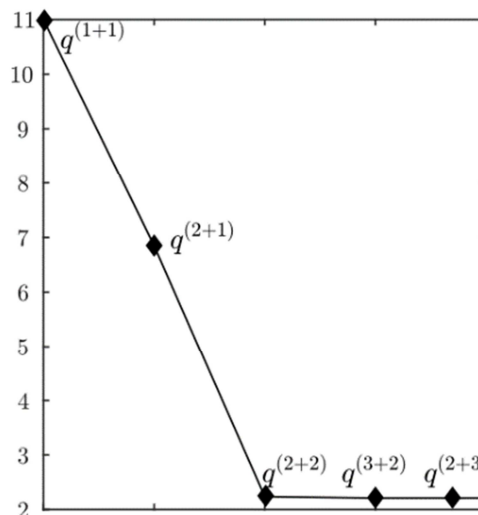
$$u(k) = \begin{cases} \text{sen}(\pi k/25), & 0 \leq k < 250 \\ 1, & 250 \leq k < 500 \\ -1, & 500 \leq k < 750 \\ 0,3\text{sen}(\pi k/25) + \\ 0,1\text{sen}(\pi k/32) + \\ 0,6\text{sen}(\pi k/10), & 750 \leq k < 1000 \end{cases} \quad (16)$$

### 6. 3 Definição da ordem do modelo

Para obtenção dos quocientes de *Lipschitz* são utilizados os primeiros 500 valores da dupla  $\{u,y\}$ . Tal escolha foi realizada, pois foi verificado que a utilização de mais valores da dupla  $\{u,y\}$  não influenciavam na ordem ótima do sistema, porém elevavam o tempo de determinação da ordem ótima do modelo.

Os quocientes calculados, conforme apresentados na Figura 5, são:  $q^{1+1} = 10,99$ ,  $q^{2+1} = 6,85$ ,  $q^{2+2} = 2,23$ ,  $q^{3+2} = 2,21$  e  $q^{2+3} = 2,21$ . Como  $6,85 \gg 2,23 \approx 2,21$ , logo a ordem ótima é  $p = r = 2$ .

**Figura 5 - Quocientes de *Lipschitz* para o modelo proposto**



#### 6. 4 Obtenção do modelo

A rede neural tipo *Wiener* foi implementada no programa *TensorFlow* (ABADI et al., 2016) e o conjunto de dados utilizado para treinar a rede é o conjunto de dados de treinamento mostrado na Equação (15). Além disso, foi utilizado o algoritmo de otimização do gradiente descendente, com uma taxa de aprendizado de  $1 \times 10^{-4}$ .

A rede neural foi treinada em um ciclo do conjunto de treinamento, ou seja, foi observado que uma única iteração ao longo de todo o conjunto de

treinamento foi suficiente para que a rede aprendesse as dinâmicas do sistema adequadamente.

O sistema aprendido é apresentado nas Equações (17) e (18), já no espaço de estados, em tempo discreto.

$$x(k+1) = \begin{bmatrix} -0,48 & 0,30 \\ 0,04 & -0,93 \end{bmatrix} x(k) + \begin{bmatrix} 0,64 \\ 0,19 \end{bmatrix} u(k) \quad (17)$$

$$y(k) = N[0,46 \quad 0,45] \quad (18)$$

O modelo aprendido pela rede neural recorrente tipo *Wiener* apresentou erro quadrático médio de  $19,18 \times 10^{-4}$  nos dados de validação em relação a saída do sistema real, apresentada na Equação (14). Demonstrando assim que a rede neural é capaz de aprender de forma a generalizar um conjunto de dados desconhecido à rede.

Como pode-se observar, os autovalores do modelo estão posicionados no semiplano esquerdo do plano complexo, em -0,4548 e -0,9552, logo o modelo aprendido representa um sistema estável.

A matriz de observabilidade  $O$  e a matriz controlabilidade  $M$  são definidas em (19) e (20), respectivamente.

$$O = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 0,46 & 0,45 \\ -0,28 & -0,28 \end{bmatrix} \quad (19)$$

$$M = \begin{bmatrix} B & AB \end{bmatrix} = \begin{bmatrix} 0,62 & -0,25 \\ 0,19 & -0,15 \end{bmatrix} \quad (20)$$

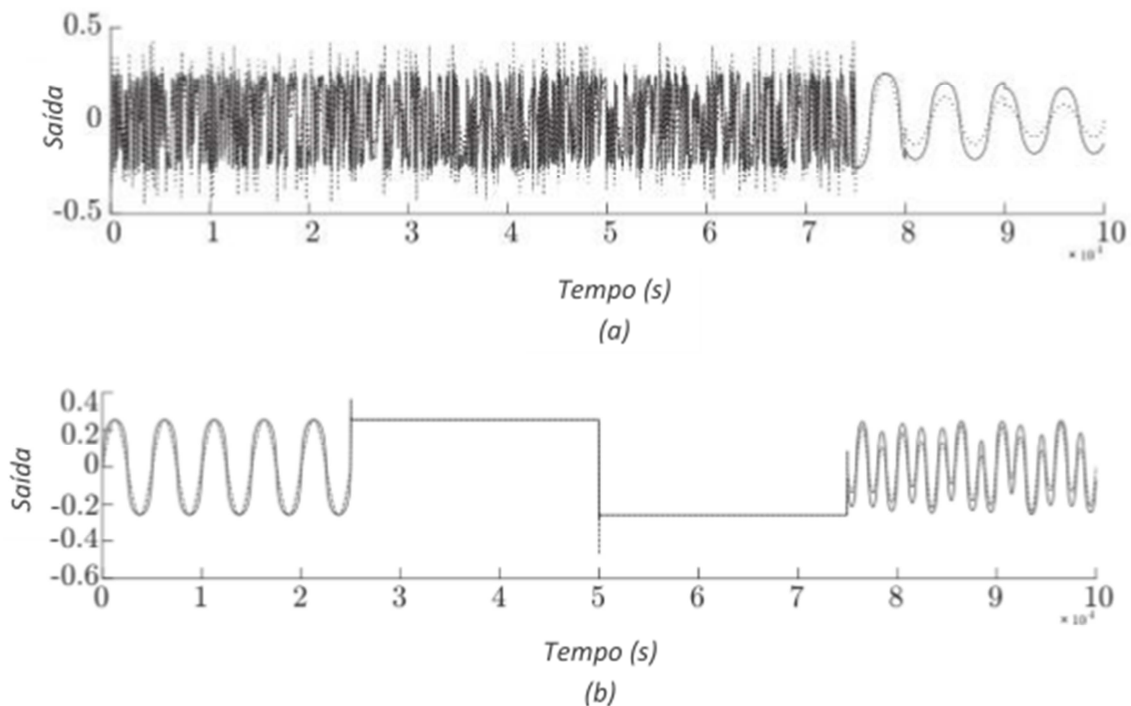
onde o posto( $M$ ) = posto( $N$ ) = 2, logo, o sistema é completamente controlável e observável.

Na Figura 6 são apresentadas as curvas do modelo obtido e as curvas com o conjunto de dados de treinamento em (a) e de validação em (b),

respectivamente. Com a curva cinza contínua representado a saída real e a curva preta tracejada a saída do modelo obtido.

É possível observar que o modelo obtido através da rede neural aqui apresentada se aproxima do modelo real.

**Figura 6 - Resultados da identificação do sistema, sendo a saída real expressa pela curva contínua cinza e a saída obtida do modelo aprendido expressa em preto tracejado (a) resultado obtido com o conjunto de dados de treinamento e (b) resultado obtido com o conjunto de dados de validação.**



### 3 CONSIDERAÇÕES FINAIS

Neste trabalho foi demonstrado um método de identificação de sistemas não lineares em espaço de estados totalmente automático. A utilização de redes neurais junto à um algoritmo de determinação da ordem do modelo torna o método apresentado aplicável à sistemas sem conhecimento prévio do processo a ser modelado. Ainda assim, a representação em espaço de estados permite a análise do modelo do sistema aprendido.

O método apresentado neste trabalho foi aplicado em um sistema que representa a posição de uma haste por sinais pneumáticos. Foi obtido um modelo estável, totalmente controlável e observável, e com erro quadrático médio de  $19,18 \times 10^{-4}$  em relação a saída real do sistema. Provando assim que o método é eficaz para identificar um sistema não linear sem conhecimento prévio do processo a ser modelado. Para trabalhos futuros, será modelado a planta de um robô de soldagem linear (MOR et al, 2016) utilizando o método proposto neste trabalho.

## REFERÊNCIAS

ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., CORRADO, G. S., DAVIS, A., DEAN, J., DEVIN, M. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. **arXiv preprint arXiv:1603.04467**, 2016.

AREFI, M. M., MONTAZERI, A., POSHTAN, J.; JAHED-MOTLAGH, M. Wiener-neural identification and predictive control of a more realistic plug-flow tubular reactor. **Chemical Engineering Journal**, v. 138, n. 1, p. 274-282, 2008.

CHEN, G.; CHEN, Y.; OGMEN, H. Identifying chaotic systems via a Wiener-type cascade model. **IEEE Control Systems**, v. 17, n. 5, p. 29-36, 1997.

DEMPSEY, E. J.; WESTWICK, D. T. Identification of Hammerstein models with cubic spline nonlinearities. **IEEE Transactions on Biomedical Engineering**, v. 51, n. 2, p. 237-245, 2004.

GIRI, F. **Block-oriented nonlinear system identification**. London: Springer, 2010.

GUERRA, F.; SIERAKOWSKI, Cezar A.; COELHO, L. S. Rede neural função de base radial usando otimização por aproximação estocástica em identificação de sistemas não-linear com comportamento caótico. **VII Simpósio Brasileiro de Automação Inteligente, São Luís, MA**, v. 1, p. 1-5, 2005.

HAARNOJA, T., AJAY, A., LEVINE, S.; ABBEEL, P. L. Backprop KF: Learning discriminative deterministic state estimators. In: **Advances in Neural Information Processing Systems**. 2016. p. 4376-4384.

HAMMERSTEIN, A. Nichtlineare integralgleichungen nebst anwendungen. **Acta mathematica**, v. 54, n. 1, p. 117-176, 1930.

HE, X.; ASADA, H.. A new method for identifying orders of input-output models for nonlinear dynamic systems. In: **American Control Conference, 1993**. IEEE, 1993. p. 2520-2523.

HSU, Y.; WANG, J. A Wiener-type recurrent neural network and its control strategy for nonlinear dynamic applications. **Journal of process control**, v. 19, n. 6, p. 942-953, 2009.

JUANG, C.; LIN, C. A recurrent self-organizing neural fuzzy inference network. **IEEE Transactions on Neural Networks**, v. 10, n. 4, p. 828-845, 1999.

KENDALL, A.; GRIMES, M.; CIPOLLA, R. Posenet: A convolutional network for real-time 6-dof camera relocalization. In: **Proceedings of the IEEE international conference on computer vision**. 2015. p. 2938-2946.

KNOHL, T.; UNBEHAUEN, H. Adaptive position control of electrohydraulic servo systems using ANN. **Mechatronics**, v. 10, n. 1, p. 127-143, 2000.

LIMA, R.; BARROS, P. Identificação de sistemas por meio de realizações baseadas na resposta a um pulso duplo. **XI Simpósio Brasileiro de Automação Inteligente SBAI**, 2013.

MÓR, J. L., EVALD, P. J. D. O., STEFFENS, C. R., LEONARDO, B. Q., BOTELHO, S. S. C., AZZOLIN, R. Z. (2016). Uma contribuição para o controle de velocidade de um robô soldador, **XXI Congresso Brasileiro de Automática CBA**, 2016.

NELLES, O. **Nonlinear system identification: from classical approaches to neural networks and fuzzy models**. Springer Science & Business Media, 2013.

OGATA, K.; YANG, Y. Modern control engineering. 1970.

PUNJANI, A.; ABBEEL, P. Deep learning helicopter dynamics models. In: **Robotics and Automation (ICRA), 2015 IEEE International Conference on**. IEEE, 2015. p. 3223-3230.

RICCO, R. A., VERLY, A., TEIXEIRA, B. O. S., AGUIRRE, L. A.. Identificação usando técnicas de subespaços de uma planta de bombeamento de Água - um estudo de caso. **X Simpósio Brasileiro de Automação Inteligente SBAI**, 2011.

SLOUN, R. J. G., DEMI, L., POSTEMA, A. W., DE LA ROSETTE, J. J., WIJKSTRA, H.; MISCHI, M. Imaging of the dispersion coefficient of ultrasound contrast agents by wiener system identification for prostate cancer localization. In: **Ultrasonics Symposium (IUS), 2015 IEEE International**. IEEE, 2015. p. 1-4.

WIENER, N. **Nonlinear problems in random theory. Nonlinear Problems in Random Theory, by Norbert Wiener.** ISBN 0-262-73012-X. Cambridge, Massachusetts, USA: The MIT Press, 1966.

WIGREN, T. Recursive prediction error identification using the nonlinear Wiener model. **Automatica**, v. 29, n. 4, p. 1011-1025, 1993.

ZHOU, L.; LI, X.; PAN, F. Gradient-based iterative identification for MISO Wiener nonlinear systems: application to a glutamate fermentation process. **Applied Mathematics Letters**, v. 26, n. 8, p. 886-892, 2013.

Enviado em: 01 ago. 2017

Aceito em: 10 nov. 2017

Editor responsável: Mateus das Neves Gomes