

## Estimação de estado baseada nos algoritmos LMS em espaço de estado \*

Jadyna Ayres Rodrigues\* Thiago Silva Fernandes\*  
Francisco das Chagas de Souza\*

\* *Laboratório de Sistemas Adaptativos e Processamento de Sinais - LSAPS, Universidade Federal do Maranhão - UFMA, São Luís - MA, (e-mail: jadyna.rodriques@discente.ufma.br; thiago.fernandes@discente.ufma.br; francisco.souza@ufma.br).*

**Abstract:** Based on the state space model, algorithms of type state space LMS (SSLMS - *State Space Least Mean Square*) allow the generation of an estimated state vector, being a possible solution to the state estimation problem. In terms of tracking ability, the SSLMS is superior to the standard LMS (*Least Mean Square*), which is limited due to the assumptions of the linear regression model. By overcoming this limitation, SSLMS shows a significant improvement in tracking performance compared to the standard LMS and its known variants. Considering this principle, we propose in this paper the development of the State Space Zero-Attracting LMS algorithm (SSZA-LMS) built around the state space structure. The proposed algorithm is used to estimate the states of a synchronous motor with a nonlinear model. Numerical experiments show that the proposed algorithm presents superior performance when compared to the SSLMS algorithm.

**Resumo:** Baseado no modelo de espaço de estado, os algoritmos do tipo o LMS em espaço de estado (SSLMS - *State Space Least Mean Square*) permitem a geração de um vetor de estados estimados, sendo uma possível solução para o problema de estimação de estado. Em termos de capacidade de rastreamento, o SSLMS é superior ao algoritmo LMS padrão (*Least Mean Square*), o qual é limitado devido às hipóteses do modelo de regressão linear. Ao superar essa limitação, o SSLMS mostra uma melhoria significativa no desempenho de rastreamento em comparação com o LMS padrão e suas variantes conhecidas. Considerando esse princípio, propõe-se neste trabalho o desenvolvimento de um novo algoritmo adaptativo em espaço de estado, baseado no algoritmo do tipo ZA-LMS (*Zero-Attracting least-mean-square*) construído em torno da estrutura de espaço de estado. O algoritmo proposto é utilizado para a estimação de estado de um motor síncrono com modelo não linear. Experimentos numéricos comprovam que o algoritmo proposto apresenta desempenho superior quando comparado ao algoritmo SSLMS.

**Keywords:** Adaptive filter, SSLMS algorithm, state estimation, state space, synchronous motor.

**Palavras-chave:** Filtro adaptativo, algoritmo SSLMS, estimação de estado, espaço de estado, motor síncrono.

### 1. INTRODUÇÃO

Com um objetivo tão amplo, a teoria da estimação encontra aplicações em diversos campos: comunicações, radar, sonar, navegação, sismologia, engenharia biomédica e engenharia financeira, entre outros (Haykin, 2008).

O problema de estimar uma quantidade desconhecida de interesse é frequentemente encontrado na literatura. Uma dificuldade prática enfrentada na implementação de filtros ótimos é a exigência de informações a priori sobre as estatísticas dos dados. Esses dados podem não estar disponíveis em certos casos. Sabe-se geralmente que

\* O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, da Fundação de Amparo à Pesquisa e ao Desenvolvimento Científico e Tecnológico do Maranhão (FAPEMA) e do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

algoritmos baseados em modelo de estimação (como o filtro Kalman e sua família) têm melhor desempenho que os algoritmos que não se baseiam no modelo em espaço de estados (como a família do LMS), devido a informações adicionais disponíveis na dinâmica do sistema (que pode ser usada para fornecer o modelo de espaço de estado), (Ahmed et al., 2018; Haykin, 2008; Kailath, 1974; Kay, 1993).

Uma versão generalizada do algoritmo LMS (*Least Mean Square*), foi desenvolvido por (Malik, 2004a,b) e proposto com diferentes análises. Esse novo algoritmo, denominado SSLMS (*State Space Least Mean Square*), incorpora um modelo linear de espaço de estado variável no tempo do ambiente subjacente. A versão SS (*State Space*) é derivada como um estimador de estado com base em medições nas quais está presente ou não um ruído de medição. A derivação inclui ambos estimadores de estado atual e previsto (Ahmed et al., 2014). O algoritmo SSLMS (*State Space*

Least Mean Square) e sua variante normalizada SSNLMS (State Space Normalized Least Mean Square), apresentam bons resultados na literatura (Malik and Salman, 2008; Malik and Bhatti, 2004; Malik, 2004a,b).

Neste artigo, propõe-se um novo algoritmo, denominado SSZA-LMS, desenvolvido em torno da estrutura de espaço de estados (SS), com o objetivo de estimar os estados do sistema, experimentos numéricos comprovam que o método proposto apresenta melhor desempenho quando comparado com outros algoritmos já existentes na literatura o SSLMS e SSNLMS.

O modelo do sistema é ilustrado na Figura 1, onde  $\mathbf{u}[k]$  é o vetor de entrada,  $\mathbf{w}[k]$  e  $\mathbf{v}[k]$  são os ruídos de processo e medição respectivamente,  $\mathbf{y}[k]$  é a saída do sistema medido e  $\hat{\mathbf{x}}[k]$  é o estado estimado.

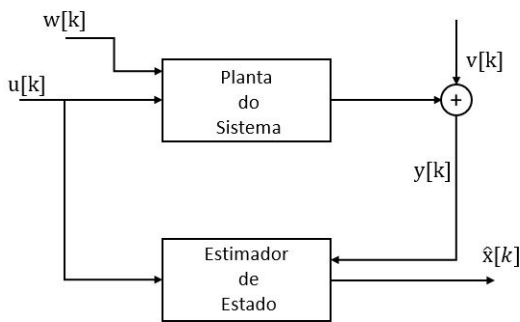


Figura 1. ESTRUTURA DE UM ESTIMADOR DE ESTADO.

O restante deste trabalho está organizado da seguinte forma: na Seção 2, apresenta-se a formulação do problema de estimação de estado e define-se o modelo geral de espaço de estados, considerando o modelo de um motor de ímã permanente bifásico síncrono. Na Seção 3, a revisão dos algoritmos LMS e NLMS em espaço de estados é apresentada. O método proposto SSZA-LMS é apresentado na Seção 4. Resultados da simulação e a discussão da comparação dos algoritmos tratados neste artigo são apresentadas na Seção 5. Na Seção 6, apresentam-se as conclusões.

## 2. FORMULAÇÃO DO PROBLEMA

Nesta seção, inicialmente, define-se o modelo geral em espaço de estados de um sistema linear variável no tempo discreto representado por

$$\mathbf{x}[k+1] = \mathbf{A}[k]\mathbf{x}[k] + \mathbf{B}[k]\mathbf{u}[k] + \mathbf{w}[k], \quad (1)$$

$$\mathbf{y}[k] = \mathbf{C}[k]\mathbf{x}[k] + \mathbf{D}[k]\mathbf{u}[k] + \mathbf{v}[k] \quad (2)$$

onde  $\mathbf{x} \in \mathbb{R}^n$  é o vetor de estado do processo,  $\mathbf{y} \in \mathbb{R}^m$  é o vetor de saída medido de modo que  $m \leq n$ . Nas Equações (1) e (2),  $\mathbf{A}[k]$  é a matriz de transição de estados,  $\mathbf{B}[k]$  é a matriz de entrada,  $\mathbf{u}[k] \in \mathbb{R}^p$  é o vetor de entrada,  $\mathbf{w}[k] \in \mathbb{R}^n$  e  $\mathbf{v}[k] \in \mathbb{R}^m$  são os vetores de ruídos de processo e medição, respectivamente. Em (2),  $\mathbf{C}[k]$  é a matriz de saída, com dimensão  $\dim[\mathbf{C}[k]] = m \times n$  e  $\mathbf{D}[k]$  é a matriz de transmissão direta com  $\dim[\mathbf{D}[k]] = m \times p$ . Presume-se que o sistema discreto por (1) e (2) seja observável.

Um caso especial é o sistema linear não-forçado (i.e., autônomo) variável no tempo, representado como (Ahmed et al., 2016)

$$\mathbf{x}[k+1] = \mathbf{A}[k]\mathbf{x}[k] + \mathbf{w}[k] \quad (3)$$

$$\mathbf{y}[k] = \mathbf{C}[k]\mathbf{x}[k] + \mathbf{v}[k]. \quad (4)$$

A representação do espaço de estado para um sistema de tempo contínuo não linear é

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{w}) \quad (5)$$

$$\mathbf{y} = h(\mathbf{x}, \mathbf{u}, \mathbf{v}) \quad (6)$$

onde  $f$  e  $h$  são funções não lineares e os parâmetros foram definidos anteriormente (Ahmed et al., 2016). O motor síncrono de ímã permanente bifásico pode ser representado como (Simon, 2006; Dhaouadi et al., 1991)

$$\begin{aligned} \dot{i}_a &= \frac{-R}{L}i_a + \frac{\omega\lambda}{L} \text{sen } \theta + \frac{u_a + \Delta u_a}{L} \\ \dot{i}_b &= \frac{-R}{L}i_b - \frac{\omega\lambda}{L} \text{cos } \theta + \frac{u_b + \Delta u_b}{L} \end{aligned} \quad (7)$$

$$\dot{\omega} = \frac{-3\lambda}{2J}i_a \text{sen } \theta + \frac{3\lambda}{2J}i_b \text{cos } \theta - \frac{F\omega}{J} + \Delta\alpha$$

$$\dot{\theta} = \omega$$

$$\mathbf{y} = \begin{bmatrix} I_a \\ I_b \end{bmatrix} + \begin{bmatrix} v_a \\ v_b \end{bmatrix} \quad (8)$$

onde  $i_a(t)$  e  $i_b(t)$  são as corrente dos enrolamentos,  $R$  e  $L$  são respectivamente a resistência e indutância do enrolamento do motor,  $\theta(t)$  é a posição angular do rotor,  $\lambda$  é a constante de fluxo do rotor,  $\omega(t)$  é a velocidade angular do rotor,  $F$  é o atrito viscoso do rotor e  $J$  é o momento de inércia do eixo do rotor. As tensões aplicadas aos dois enrolamentos do motor são  $u_a(t) = \text{sen}2\pi t$  e  $u_b(t) = \text{cos}2\pi t$ . Os ruídos  $\Delta u_a$  e  $\Delta u_b$ , representados na Equação (7), são causados pelas incertezas nas entradas de controle  $u_a(t)$  e  $u_b(t)$ ; já  $\Delta\alpha(t)$  é o ruído devido à incerteza no torque da carga e  $v_a(t)$  e  $v_b(t)$  São processos aleatórios gaussianos com média zero e desvio padrão 0.1 A. O vetor de estado do sistema são definidos como

$$\mathbf{x} = [i_a \ i_b \ \omega \ \theta]^T. \quad (9)$$

Com esta definição, obtém-se

$$\begin{aligned} \dot{\mathbf{x}} &= [\dot{x}_1 \ \dot{x}_2 \ \dot{x}_3 \ \dot{x}_4]^T \\ &= f(\mathbf{x}, \mathbf{u}) \\ &= \begin{bmatrix} \frac{-R}{L}x_1 + \frac{x_3\lambda}{L} \text{sen } x_4 + \frac{u_a}{L} \\ \frac{-R}{L}x_2 - \frac{x_3\lambda}{L} \text{cos } x_4 + \frac{u_b}{L} \\ \frac{-3\lambda}{2J}x_1 \text{sen } x_4 + \frac{3\lambda}{2J}x_2 \text{cos } x_4 - \frac{F x_3}{J} \\ x_3 \end{bmatrix} + \begin{bmatrix} \frac{\Delta u_a}{L} \\ \frac{\Delta u_b}{L} \\ \Delta\alpha \\ 0 \end{bmatrix} \end{aligned} \quad (10)$$

O sistema foi discretizado e linearizado para a simulação e implementação dos algoritmos de estimação, são referidos no apêndice.

## 3. REVISANDO OS ALGORITMOS LMS E NLMS EM ESPAÇO DE ESTADOS

Inspirada na estrutura de espaço de estado do filtro de Kalman, a generalização do LMS tem sido obtida na forma

de SSLMS por Malik e Salman em (Malik and Salman, 2008) usando o modelo em espaço de estado do ambiente subjacente e algumas variantes. Provou-se que o algoritmo SSLMS apresenta maior taxa de convergência e desempenho de rastreamento superior em relação às abordagens clássicas existentes, além da capacidade de lidar com sistemas MIMO (*Multiple Input-Multiple Output*) (Qayyum et al., 2019).

A ideia é gerar um vetor de estado estimado  $\hat{\mathbf{x}}[k]$  usando as observações da saída  $\mathbf{y}[1], \mathbf{y}[2], \mathbf{y}[3] \dots \mathbf{y}[k]$ , supondo que as observações  $\mathbf{y}[k]$  são medidas a partir do instante  $k = 1$  e o vetor de estado inicial  $\mathbf{x}[0] = \mathbf{x}_0$  é desconhecido. A hipótese de observabilidade do par  $(\mathbf{A}[k], \mathbf{C}[k])$  permite projetar um estimador de estado. O modelo do sistema dinâmico (caracterizado pelas matrizes do sistema) permite calcular uma predição da estimatição do estado  $\bar{\mathbf{x}}[k]$  no tempo  $k$  (usando observações até no instante  $k - 1$ ) da seguinte maneira (Malik and Salman, 2008):

$$\bar{\mathbf{x}}[k] = \mathbf{A}[k-1]\hat{\mathbf{x}}[k-1], \quad (11)$$

onde  $\hat{\mathbf{x}}[k-1]$  é o estado estimado para o instante  $k-1$  e  $\bar{\mathbf{x}}[k]$  é o estado previsto no instante  $k$ . A atualização do estimador é dada por

$$\hat{\mathbf{x}}[k] = \bar{\mathbf{x}}[k] + \mathbf{K}[k]\varepsilon[k] \quad (12)$$

onde  $\mathbf{K}[k]$  é o ganho do observador (Malik and Salman, 2008). O erro de predição da saída é definido como

$$\varepsilon[k] = \mathbf{y}[k] - \bar{\mathbf{y}}[k] \quad (13)$$

onde  $\mathbf{y}[k]$  é obtido como em (4) e

$$\bar{\mathbf{y}}[k] = \mathbf{C}[k]\bar{\mathbf{x}}[k] \quad (14)$$

é a predição da saída. Define-se o erro de estimação como

$$\mathbf{e}[k] = \mathbf{y}[k] - \hat{\mathbf{y}}[k] \quad (15)$$

onde  $\hat{\mathbf{y}}[k] = \mathbf{C}[k]\hat{\mathbf{x}}[k]$  é a saída estimada. A matriz de ganho  $\mathbf{K}[k]$  é derivada do seguinte modo: Relaciona-se o erro de predição (13) e o erro de estimação (15) da seguinte forma:

$$\mathbf{e}[k] = \mathbf{y}[k] - \hat{\mathbf{y}}[k] = \varepsilon[k] - \mathbf{C}[k]\delta[k] \quad (16)$$

onde

$$\delta[k] = \hat{\mathbf{x}}[k] - \bar{\mathbf{x}}[k]. \quad (17)$$

Assumindo que  $\mathbf{C}[k]$  é posto completo,  $\hat{\mathbf{x}}[k]$  é escolhido de modo que  $\mathbf{e}[k] = \mathbf{0}$ , o que implica o seguinte

$$\varepsilon[k] = \mathbf{C}[k]\delta[k] \quad (18)$$

$\delta[k]$  escolhidos como a solução de norma mínima da equação acima que resulta em

$$\delta[k] = \mathbf{C}^T[k] (\mathbf{C}[k]\mathbf{C}^T[k])^{-1} \varepsilon[k] \quad (19)$$

aqui

$$\mathbf{K}[k] = \mathbf{C}^T[k] (\mathbf{C}[k]\mathbf{C}^T[k])^{-1} \quad (20)$$

A equação do estimador de (17) e (19) é, portanto, derivada como

$$\hat{\mathbf{x}}[k] = \bar{\mathbf{x}}[k] + \mathbf{C}^T[k] (\mathbf{C}[k]\mathbf{C}^T[k])^{-1} \varepsilon[k] \quad (21)$$

Isso é denominado em (Malik and Salman, 2008) como o algoritmo SSSLMS. No entanto, o estimador acima pode ser instável e não há condições conhecidas de estabilidade e, portanto, a convergência não é garantida. Por esta razão, uma matriz  $\mathbf{G}$  foi introduzida em (Malik and Salman, 2008) para contornar este problema. Em (Malik and Salman, 2008), afirma-se que a escolha dessa matriz  $\mathbf{G}$  depende da natureza do problema, nas simulações  $\mathbf{G}$  é considerada matriz identidade. Em nossa investigação,

vamos considerar o algoritmo SSLMS (Malik and Salman, 2008) definido como

$$\hat{\mathbf{x}}[k] = \bar{\mathbf{x}}[k] + \mu\mathbf{G}\mathbf{C}^T[k]\varepsilon[k], \quad (22)$$

onde

$$\mathbf{K}[k] = \mu\mathbf{G}\mathbf{C}^T[k]. \quad (23)$$

#### 4. MÉTODO PROPOSTO

Nesta seção, apresenta-se o novo algoritmo para a estimatição de estado de sistemas dinâmicos baseado em espaço de estados o SSZA-LMS. A atualização da Equação 24 pode ser considerada como a adaptação com base na descida mais íngreme (Haykin, 2008; Sayed, 2003) e, portanto, pode ser generalizada com a seguinte regra de atualização:

$$\hat{\mathbf{x}}[k] = \bar{\mathbf{x}}[k] - \mu\nabla\mathbf{J}[k], \quad (24)$$

onde  $\mathbf{J}[k]$  é a função de custo a ser minimizada,  $\nabla\mathbf{J}[k]$  é o gradiente. A função de custo do algoritmo SSZA-LMS proposto é

$$\mathbf{J}[k] = \frac{1}{2}\varepsilon^T[k]\varepsilon[k] + \gamma\|\bar{\mathbf{x}}[k]\|_1, \quad (25)$$

onde  $\varepsilon[k]$  é o erro de predição definido como

$$\varepsilon[k] = \mathbf{y}[k] - \bar{\mathbf{y}}[k], \quad (26)$$

sendo

$$\bar{\mathbf{y}}[k] = \mathbf{C}[k]\bar{\mathbf{x}}[k]. \quad (27)$$

Substituindo 26 e 27 em 25 temos:

$$\begin{aligned} \mathbf{J}[k] &= \frac{1}{2}(\mathbf{y}[k] - \bar{\mathbf{y}}[k])^T(\mathbf{y}[k] - \bar{\mathbf{y}}[k]) + \gamma\|\bar{\mathbf{x}}[k]\|_1 \\ &= \frac{1}{2}(\mathbf{C}[k]\mathbf{x}[k] - \mathbf{C}[k]\bar{\mathbf{x}}[k])^T(\mathbf{C}[k]\mathbf{x}[k] - \mathbf{C}[k]\bar{\mathbf{x}}[k]) \\ &\quad + \gamma\|\bar{\mathbf{x}}[k]\|_1. \end{aligned} \quad (28)$$

Agora, derivando  $\mathbf{J}[k]$  em relação a  $\bar{\mathbf{x}}$ , obtém-se:

$$\begin{aligned} \frac{\partial\mathbf{J}[k]}{\partial\bar{\mathbf{x}}[k]} &= \mathbf{C}^T[k]\mathbf{C}[k]\bar{\mathbf{x}}[k] - \mathbf{C}^T[k]\mathbf{C}[k]\mathbf{x}[k] + \gamma\text{sgn}(\bar{\mathbf{x}}[k]) \\ &= \mathbf{C}^T[k](\mathbf{C}[k]\bar{\mathbf{x}}[k] - \mathbf{C}[k]\mathbf{x}[k]) + \gamma\text{sgn}(\bar{\mathbf{x}}[k]) \\ &= \mathbf{C}^T[k](\bar{\mathbf{y}}[k] - \mathbf{y}[k]) + \gamma\text{sgn}(\bar{\mathbf{x}}[k]). \end{aligned} \quad (29)$$

Dessa forma, o gradiente da função custo resulta em

$$\nabla\mathbf{J}[k] = -\mathbf{C}^T[k]\varepsilon[k] + \gamma\text{sgn}(\bar{\mathbf{x}}[k]). \quad (30)$$

Das Equações 24 e 30, a equação do estimador é obtida como

$$\hat{\mathbf{x}}[k] = \bar{\mathbf{x}}[k] + \mu\varepsilon\mathbf{C}^T[k] - \rho\text{sgn}(\bar{\mathbf{x}}[k]), \quad (31)$$

onde  $\rho = \mu\gamma$  e  $\text{sgn}(\cdot)$  são os componentes da função sinal definidos como

$$\text{sgn}(g) = \begin{cases} g/|g| & g \neq 0 \\ 0 & g = 0. \end{cases} \quad (32)$$

O SSZA-LMS tem o termo adicional  $-\rho\text{sgn}(\bar{\mathbf{x}}[k])$  que sempre atrai os estado para zero, por isso denominado atrator para zero, cuja força é controlada por  $\rho$  (Chen et al., 2009). O parâmetro  $\mu$  é o tamanho do passo para auxiliar na obtenção rápida da solução. A metodologia proposta é mostrada no Algoritmo 1.

**Algoritmo 1** Método Proposto SSZA-LMS

- 1: **Inicialização:**  $\hat{\mathbf{x}}[0] = \hat{\mathbf{x}}_0$ ,  $0 < \mu < 2$  e  $\rho = 1 \times 10^{-5}$
- 2: **for:**  $k = 1, 2, \dots, N$
- 3: **Medição da saída do sistema:**  $\mathbf{y}[k]$
- 4: **Predição do estado (estimativa a priori)**  
 $\bar{\mathbf{x}}[k] = \mathbf{A}[k-1] + \hat{\mathbf{x}}[k-1]$
- 5: **Cálculo da saída estimada**  
 $\bar{\mathbf{y}}[k] = \mathbf{C}[k]\bar{\mathbf{x}}[k]$
- 6: **Erro de estimação da saída**  
 $\varepsilon[k] = \mathbf{y}[k] - \bar{\mathbf{y}}[k]$
- 7: **Estado estimado (estimativa a posteriori)**  
 $\hat{\mathbf{x}}[k] = \bar{\mathbf{x}}[k] + \mu \mathbf{C}^T[k]\varepsilon[k] - \rho \text{sgn}(\bar{\mathbf{x}}[k])$
- 8: **fim**

5. SIMULAÇÕES

Nesta seção, é apresenta-se o cenário, os experimentos numéricos e gráficos para validar o desempenho do novo algoritmo.

5.1 Cenário da Simulação

Realiza-se a estimação dos parâmetros de estado do motor síncrono na presença de um ruído gaussiano. Uma visão geral da raiz do erro quadrático médio dos estados pode ser consultada na Tabela 2. Na Tabela 1 e 3 encontra-se o erro de estimação individual e o erro global dos estados. Comparamos o desempenho dos algoritmos SSLMS, SSNLMS e SSZA-LMS. O modelo do sistema é discretizado com o período de amostragem  $\Delta t = 0.0002s$  e os algoritmos são simulados com um período de amostragem de  $0.001s$  para simular o efeito do cenário em tempo real, onde os sensores são limitados pelo período de amostragem. O parâmetro  $\rho$  escolhido para o SSZA-LMS foi  $1 \times 10^{-5}$  e passo  $\mu$  para SSLMS, SSNLMS e SSZA-LMS foram escolhidos como 0,2, 0,4 e  $5 \times 10^{-2}$ , respectivamente. A observação foi sujeita a ruído de observação gaussiano de covariância  $\sigma^{-2} = 0.01^2$  e o ruído do processo foi considerado de covariância  $\sigma^{-2} = 0.001^2$ . Os estados iniciais reais do sistema foram considerados como  $\mathbf{x}[0] = [1 \ 3 \ 0.5 \ 2]^T$  e a estimativa inicial para os algoritmos foi escolhida como sendo  $\hat{\mathbf{x}}[0] = [0 \ 0 \ 0 \ 0]$ . Os parâmetros das simulações são referidos em (Simon, 2006).

5.2 Métricas de Avaliação

Para realizar a comparação entre os algoritmos, foram definidas as seguintes funções de desempenho baseadas no erro quadrático médio, erro de estimação de estado, raiz do erro quadrático e raiz do erro quadrático global:

O erro quadrático médio é calculado do seguinte modo:

$$e_{j,k}^2 = \frac{1}{n} \sum_{k=1}^n (\hat{y}_{j,k} - y_{j,k})^2, \quad j = 1, 2. \quad (33)$$

O erro quadrático médio apresenta-se na Figura 4. Os erros de estimação de estado, apresenta-se na Tabela 1 são definidos como:

$$\eta_{i,k} \triangleq (\hat{x}_{i,k} - x_{i,k})^2, \quad i = 1, 2, 3, 4, \quad (34)$$

onde  $\hat{x}$  é o estado estimado de  $x$  e  $i$  indica o estado  $i$ th. Na Tabela 2, apresenta-se a raiz do erro quadrático médio calculada como:

$$\eta = \sqrt{\frac{1}{n} \sum_{i=1}^n \eta_{i,k}}. \quad (35)$$

Na Tabela 3, apresenta-se o erro quadrático global no instante  $k$  calculado como:

$$\eta_k = \frac{1}{4} \sum_{i=1}^4 \eta_{i,k}, \quad i = 1, 2, 3, 4. \quad (36)$$

5.3 Resultado da Simulação

Nas Figuras 2 e 3, pode-se observar a capacidade de rastreamento dos algoritmos, entretanto verifica-se na figura 4, a partir do erro médio quadrático, que o método proposto converge para zero mais rápido em comparação com os outros algoritmos.

Nas Tabelas 1,2,3 podemos analisar melhor o desempenho dos algoritmos. O erro de estimação individual dos estados, a raiz do erro quadrático médio e o erro global do algoritmo proposto é menor em comparação com ao SSLMS e SSNLMS e, portanto, esse algoritmo pode muito bem ser implementado e considerado como o novo algoritmo de estimação de estado do sistema.

Tabela 1. ERRO DE ESTIMAÇÃO INDIVIDUAL DOS ESTADOS, EQUAÇÃO (34)

| Estados      | SSLMS           | SSNLMS          | SSZA-LMS         |
|--------------|-----------------|-----------------|------------------|
| Estado $x_1$ | $2.0574e^{-05}$ | $8.3852e^{-05}$ | $2.0967e^{-06}$  |
| Estado $x_2$ | $3.5495e^{-05}$ | $6.1230e^{-05}$ | $3.3808e^{-06}$  |
| Estado $x_3$ | 0.0034          | 0.0058          | $2.6481 e^{-04}$ |
| Estado $x_4$ | 0.0012          | 0.049           | $7.2144e^{-05}$  |

Tabela 2. RAIZ DO ERRO QUADRÁTICO MÉDIO EQUAÇÃO (35)

| Estados e observações | SSLMS  | SSNLMS | SSZA-LMS |
|-----------------------|--------|--------|----------|
| Estado $x_1$          | 0.0057 | 0.0092 | 0.0021   |
| Estado $x_2$          | 0.0077 | 0.0094 | 0.0025   |
| Estado $x_3$          | 0.0546 | 0.0724 | 0.0132   |
| Estado $x_4$          | 0.0377 | 0.0734 | 0.0098   |

Tabela 3. ERRO QUADRÁTICO GLOBAL EQUAÇÃO (36)

| Algoritmo | Erro global     |
|-----------|-----------------|
| SSLMS     | 0.0048          |
| SSNLMS    | 0.0061          |
| SS-ZALMS  | $2.8832e^{-04}$ |

6. CONCLUSÃO

Neste artigo, um novo algoritmo da classe SSLMS foi desenvolvido. O novo algoritmo, denominado SSZA-LMS (*State Space Zero-Attracting LMS*), foi usado para a estimação de estado de um motor síncrono não linear. O desempenho do novo algoritmo foi investigado, comparando-o com os algoritmos SSLMS e o SSNLMS já existentes na literatura. Resultados de simulação mostram que o algoritmo SSZA-LMS supera os algoritmos SSLMS e SSNLMS em termos de capacidade de rastreamento, erro quadrático, raiz do erro quadrático médio de estimação e o erro de estimação individual e global.

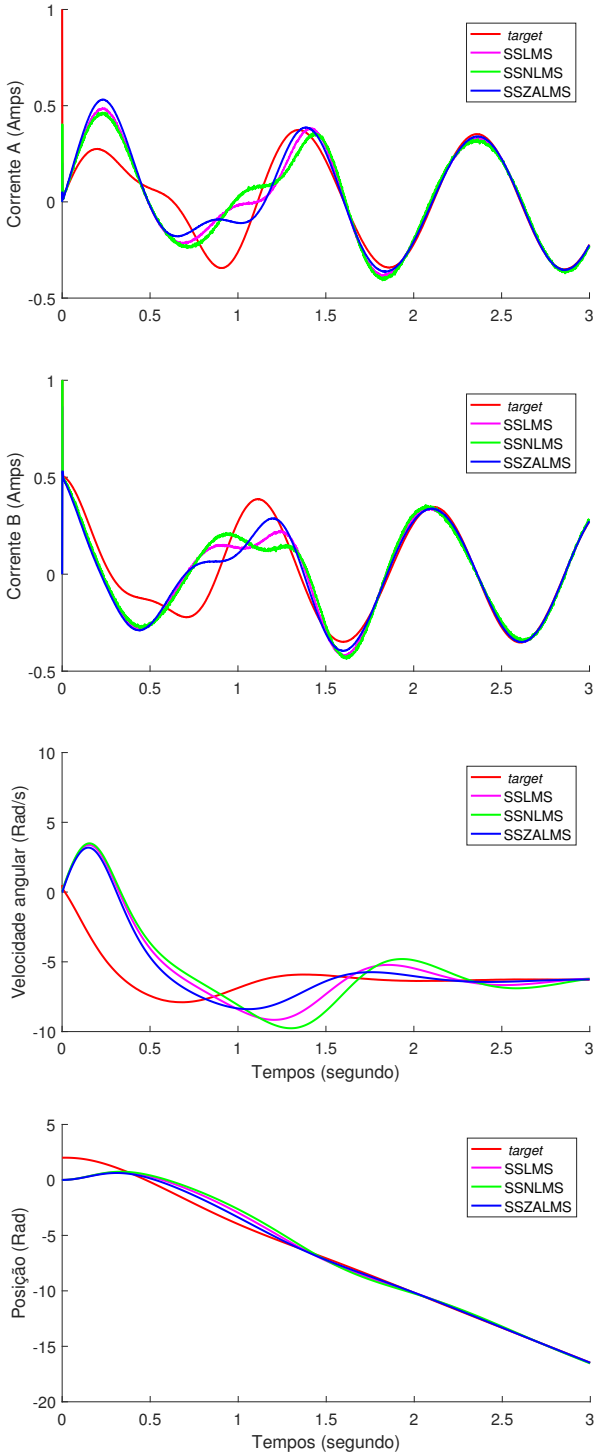


Figura 2. EVOLUÇÃO DOS ESTADOS  $x_1, x_2, x_3$  e  $x_4$ .

### 7. APÊNDICE

Para a simulação do modelo do sistema, é essencial que o sistema seja discretizado. Considerando o sistema do motor síncrono descrito pelo conjunto de equações 7, o sistema é discretizado como descrito em (Simon, 2006)

$$\mathbf{x}[k+1] = f(\mathbf{x}[k], \mathbf{u}[k]) + \mathbf{w}[k]$$

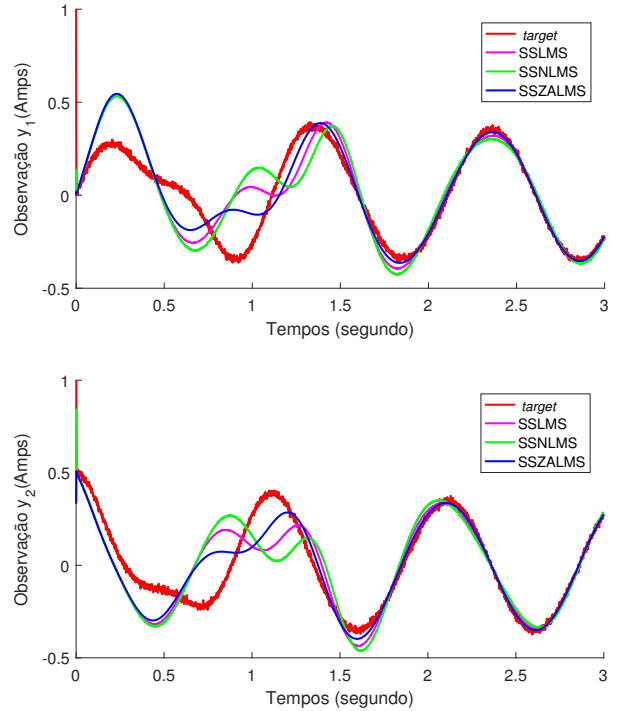


Figura 3. OBSERVAÇÃO  $y_1, y_2$ .

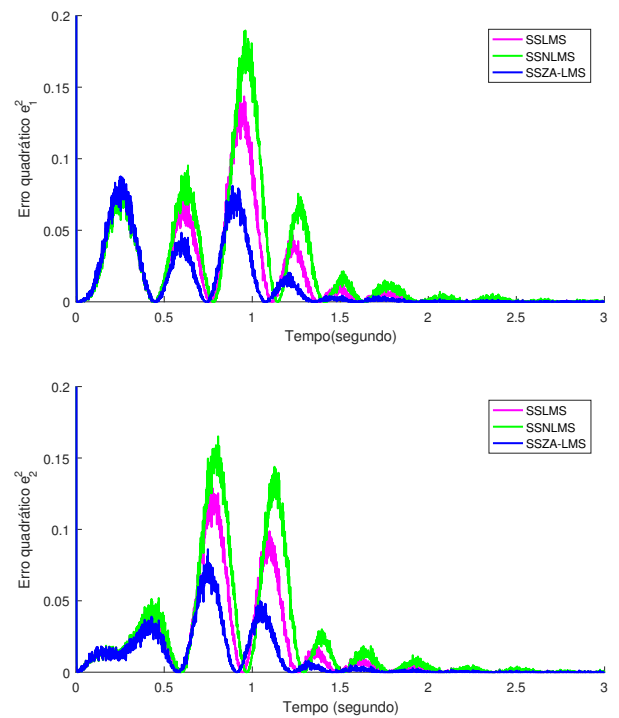


Figura 4. ERRO QUADRÁTICO MÉDIO  $e_1^2, e_2^2$ .

$$= \mathbf{x}[k] + \begin{bmatrix} \frac{-R}{L} x_1[k] + \frac{x_3[k]\lambda}{L} \sin x_4[k] + \frac{u_a[k]}{L} \\ \frac{-R}{L} x_2[k] - \frac{x_3[k]\lambda}{L} \cos x_4[k] + \frac{u_b[k]}{L} \\ \frac{-3\lambda}{2J} x_1[k] \sin x_4[k] + \frac{3\lambda}{2J} x_2[k] \cos x_4[k] - \frac{F x_3[k]}{J} \\ x_3[k] \end{bmatrix} \Delta t$$

$$+ \begin{bmatrix} \frac{\Delta u_a[k]}{L} \\ \frac{\Delta u_b[k]}{L} \\ \frac{\Delta \alpha}{0} \\ 0 \end{bmatrix} \Delta t + \mathbf{w}[k] \quad (37)$$

$$y[k] = h(x[k]) + v[k] \\ = \begin{bmatrix} x_1[k] \\ x_2[k] \end{bmatrix} + \begin{bmatrix} v_a[k] \\ v_b[k] \end{bmatrix} \quad (38)$$

Para a implementação dos algoritmos o sistema foi linearizado, através a equação do sistema tomando a derivada parcial de  $f(x[k], u[k])$  em relação a  $x[k]$  e  $u[k]$  para obter

$$f'(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \frac{\partial f_1}{\partial x_4} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \frac{\partial f_2}{\partial x_4} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} & \frac{\partial f_3}{\partial x_4} \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_2} & \frac{\partial f_4}{\partial x_3} & \frac{\partial f_4}{\partial x_4} \end{bmatrix} \quad (39)$$

com essa conjuntura, podemos agora encontrar as matrizes derivadas como

$$\mathbf{A}[k] = \frac{\partial f(\hat{x}[k], u[k])}{\partial x} \\ = I + \Delta t \\ \mathbf{x} \begin{bmatrix} -\frac{R}{L} & 0 & \frac{\lambda \text{sen } \hat{x}_4[k]}{L} & \frac{\hat{x}_3[k] \lambda \cos \hat{x}_4[k]}{L} \\ 0 & -\frac{R}{L} & -\frac{\lambda \cos \hat{x}_4[k]}{L} & \frac{\hat{x}_3[k] \lambda \text{sen } \hat{x}_4[k]}{L} \\ -\frac{3\lambda \text{sen } \hat{x}_4}{2J} & \frac{3\lambda \cos \hat{x}_4}{2J} & -\frac{F}{J} & -\frac{3\lambda[\hat{x}_1[k] \cos \hat{x}_4[k] + \hat{x}_2[k] \text{sen } \hat{x}_4[k]]}{2J} \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (40)$$

$$\mathbf{C}[k] = \frac{\partial f(\hat{x}[k])}{\partial x} \\ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (41)$$

## AGRADECIMENTOS

Os autores agradecem à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, a Fundação de Amparo à Pesquisa e ao Desenvolvimento Científico e Tecnológico do Maranhão (FAPEMA) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

## REFERÊNCIAS

Ahmed, A., Moinuddin, M., and Al-Saggaf, U.M. (2014). State space least mean square for state estimation of synchronous motor. In *8th International Conference on Electrical and Computer Engineering*, 461–464. IEEE.

Ahmed, A., Moinuddin, M., and Al-Saggaf, U.M. (2016). State space least mean fourth algorithm for dynamic state estimation in power systems. *Arabian Journal for Science and Engineering*, 41(2), 527–543.

Ahmed, A., Moinuddin, M., and Al-Saggaf, U.M. (2018). q-state space least mean family of algorithms. *Circuits, Systems, and Signal Processing*, 37(2), 729–751.

Chen, Y., Gu, Y., and Hero, A.O. (2009). Sparse lms for system identification. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 3125–3128. IEEE.

Dhaouadi, R., Mohan, N., and Norum, L. (1991). Design and implementation of an extended Kalman filter for the state estimation of a permanent magnet synchronous motor. *IEEE Transactions on Power Electronics*, 6(3), 491–497.

Haykin, S.S. (2008). *Adaptive filter theory*. Pearson Education India.

Kailath, T. (1974). A view of three decades of linear filtering theory. *IEEE Transactions on Information Theory*, 20(2), 146–181.

Kay, S.M. (1993). *Fundamentals of statistical signal processing*. Prentice Hall PTR.

Malik, M.B. and Bhatti, R.A. (2004). Tracking of linear time-varying systems using state-space least mean square. In *IEEE International Symposium on Communications and Information Technology, 2004. ISCIT 2004.*, volume 1, 582–585. IEEE.

Malik, M.B. (2004a). State-space recursive least-squares: Part i. *Signal Processing*, 84(9), 1709–1718.

Malik, M.B. (2004b). State-space recursive least squares: Part ii. *Signal Processing*, 84(9), 1719–1728. doi:https://doi.org/10.1016/j.sigpro.2004.05.021. URL <https://www.sciencedirect.com/science/article/pii/S0165168404001082>. Special Section on New Trends and Findings in Antenna Array Processing for Radar.

Malik, M.B. and Salman, M. (2008). State-space least mean square. *Digital Signal Processing*, 18(3), 334–345.

Qayyum, A., Salman, M., and Malik, M.B. (2019). Receding horizon observer for linear time-varying systems. *Transactions of the Institute of Measurement and Control*, 41(9), 2612–2622.

Sayed, A.H. (2003). *Fundamentals of adaptive filtering*. John Wiley & Sons.

Simon, D. (2006). Using nonlinear Kalman filtering to estimate signals. *Embedded Systems Design*, 19(7), 38.