



## CONTROLE VETORIAL DIRETO SENSORLESS DE UM MOTOR DE INDUÇÃO TRIFÁSICO UTILIZANDO ESTIMAÇÃO NEURAL DA VELOCIDADE

KLEBBER DE A. OTTOBONI<sup>1</sup>, RICARDO Q. MACHADO<sup>1</sup>, ENES G. MARRA<sup>2</sup>.

1. *Laboratório de Fontes Alternativas e Processamento de Energia - LAFAPE, Departamento de Engenharia Elétrica, Universidade de São Paulo*  
*Av. Trabalhador São-carlense, 400, Arnold Schimidt*  
*São Carlos - São Paulo - Brasil - CEP 13566-590*  
*E-mails: klebberottoboni@usp.br, rquadros@sc.usp.br*
2. *Escola de Engenharia Elétrica, Mecânica e de Computação, Universidade Federal de Goiás*  
*Setor Leste Universitário - Av. Universitária, 1488 - Qd. 86 Bloco A*  
*Goiânia – Goiás – Brasil - CEP 74605-010*  
*E-mail: enes@emc.ufg.br*

**Abstract**— The Three-Phase Induction Motor (TIM), for its constructive and unique operating characteristics, has been the most used electric motor in industrial processes. Its robustness and small costs related to its maintenance meant that gradually the TIM has replaced most of the Direct Current motors (DC). However, a major challenge faced in this process of TIM using expansion is its insertion in applications requiring, for example, control of the speed and of the torque on the motor shaft. In the Direct Vector Control or Control by Direct Field Orientation, the measurement of rotor speed and electromagnetic flux in the machine air gap is required. However, seeking to reduce the cost of installation and maintenance of encoders and sensors, a category called Sensorless uses computational techniques for the estimation of these variables. Many studies have demonstrated the successful results using Artificial Neural Networks (ANN). Based on this, this paper seeks to: implement the TIM Vector Control by Direct Field Orientation in Matlab / Simulink environment; to propose an ANN topology capable of estimate the rotor speed and, later, check the effects on the control system by the replacement of encoders by the proposed ANN.

**Keywords**— Artificial Neural Networks, Estimation, Intelligent Control, Three-Phase Induction Motor.

**Resumo**— O Motor de Indução Trifásico (MIT), por suas características construtivas e de funcionamento singulares, tem sido a máquina elétrica mais utilizada nos processos industriais. Sua robustez e os pequenos custos relacionados à sua manutenção fizeram com que, gradualmente, o MIT substituisse boa parte dos motores de corrente contínua (CC). Porém, um grande desafio enfrentado nesse processo de expansão da utilização do MIT é a sua inserção em aplicações que exijam, por exemplo, o controle da velocidade e do torque no eixo do motor. No controle vetorial definido como Controle Vetorial Direto ou Controle por Orientação Direta de Campo, é necessária a medição da velocidade do rotor e do fluxo no entreferro da máquina. Entretanto, buscando-se reduzir os custos com instalação e manutenção dos encoders e sensores, uma categoria denominada *Sensorless* utiliza técnicas de computação para estimação dessas variáveis. Muitos trabalhos tem demonstrado a obtenção de bons resultados utilizando Redes Neurais Artificiais (RNA). Baseando-se nisso, este trabalho busca: implementar o Controle Vetorial do MIT por Orientação Direta de Campo em ambiente MatLab/Simulink; propor uma topologia de RNA capaz de estimar a velocidade do rotor e verificar os efeitos no sistema de controle da substituição dos sensores e encoders pela RNA proposta.

**Palavras-chave**— Controle Inteligente, Estimação, Motor de Indução Trifásico, Redes Neurais Artificiais.

### 1. Introdução

Apesar do desenvolvimento de técnicas de modelagem das máquinas de indução, com vistas em melhorar a eficiência do seu controle, apresentar avanços significativos desde as primeiras décadas do século XX, apenas recentemente a eletrônica de potência e a microeletrônica tem gerado resultados satisfatórios quanto ao cumprimento das exigências para implementação destas técnicas de maneira economicamente viável (Fitzgerald et al., 2006). Dentre essas técnicas destaca-se o Controle Vetorial, o qual utiliza, principalmente, a representação das variáveis da máquina trifásica nos eixos rotativos qd0 (Chee-Mun Ong , 1998), fornecendo uma considerável simplificação das equações do MIT permitindo uma analogia deste com o Motor CC.

Tendo como objetivo reduzir a quantidade de sensores e minimizar a influência de variações do ambiente no desempenho dos controladores, Redes Neurais Artificiais tem sido utilizadas, com bons resultados (Dazhi Wang et al. , 2004)(Jing Yang et al. , 2007)(Orlowska-Kowalska et al. , 1997), para a estimação da velocidade do rotor, do fluxo no entreferro e do torque eletromagnético. Essa substituição dos sensores por métodos alternativos de obtenção dos sinais de interesse caracteriza a categoria de sistemas de controle *Sensorless*.

### 2. Definição do Problema

A implementação de sistemas de controle para o MIT exige a utilização de sensores e encoders. Os quais são sensíveis às interferências do meio como:

vibrações, choques mecânicos e mudanças de temperatura. Essa susceptibilidade às interferências do meio os caracteriza como as principais fontes de erros para o sistema de controle e como os dispositivos que mais necessitam de manutenção.

O Controle Vetorial Direto utiliza dois sensores de campo no entreferro da máquina e um tacômetro para medição da velocidade do rotor. Caso haja, por exemplo alguma diferença entre a medição da velocidade e a velocidade real, os controladores podem se tornar instáveis e, além disso, esse erro se propagaria, pois, a velocidade do rotor é utilizada na determinação de variáveis como: correntes de rotor; torque eletromagnético de referência e fluxo concatenado de referência.

Uma alternativa para eliminação da sensibilidade do sistema de controle advinda dos sensores e encoders, é a sua substituição por estimadores baseados em inteligência computacional. Portanto, o objetivo desse trabalho é propor uma topologia de RNA capaz de estimar a velocidade do rotor de um MIT, através de medições das correntes e tensões do estator.

### 3. Metodologia

A metodologia deste trabalho se baseia em simulações computacionais implementadas em ambiente MatLab/Simulink. O MIT utilizado é uma máquina de 50 HP, 460 V, 4 pólos, 60 Hz, apresentado pela referência bibliográfica (Krause et al., 2002). Seus parâmetros estão apresentados na Tabela 1.

Tabela 1. Parâmetros do Motor de Indução Trifásico Utilizado nas Simulações

Parâmetros							
<b>Tensão</b> (V)	460	$r'_r$ ( $\Omega$ )	0,228	$r$ ( $\Omega$ )	0,087	$L_m$ (mH)	34,696
<b>Veloc.</b> (rpm)	1705	$L'_{lr}$ (mH)	0,801	$L_{ls}$ (mH)	0,801	$J$ (kg.m <sup>2</sup> )	1,662

Fonte: Referência (Krause et al., 2002)

A partir da transformação das variáveis do MIT para os eixos de referência  $qd0$ , é obtido um modelo matemático simplificado. Utilizando esse modelo e as devidas considerações, implementa-se o Controle Vetorial Direto. Nessa implementação, a velocidade do rotor é diretamente calculada e fornecida como realimentação para os controladores PI. A realimentação da velocidade do rotor representa a utilização do tacômetro. Para validação do sistema de controle são realizados testes com variação da

velocidade de referência e variação do torque de carga.

Os dados obtidos após a implementação do sistema de controle são utilizados para a elaboração de uma RNA, cujo objetivo é estimar a velocidade do rotor através da medição das tensões e correntes de estator do MIT. Após o treinamento da RNA através dos dados obtidos por simulação, substitui-se a realimentação da velocidade do rotor pela saída dessa rede neural. Para verificação dos resultados obtidos pela RNA, analisa-se o comportamento do sistema de controle durante uma mudança do referencial de velocidade.

#### 3.1. Controle Vetorial Direto

A aplicação da transformação para os eixos  $qd0$  nas equações de tensão da máquina de indução trifásica resulta nas equações de (1) a (6), onde  $\omega$  e  $\omega_r$  são, respectivamente, a velocidade de rotação do eixo  $qd0$  e a velocidade do rotor.

$$v_{qs} = r_s i_{qs} + \omega \lambda_{ds} + \frac{d\lambda_{qs}}{dt} \quad (1)$$

$$v_{ds} = r_s i_{ds} - \omega \lambda_{qs} + \frac{d\lambda_{ds}}{dt} \quad (2)$$

$$v_{0s} = r_s i_{0s} + \frac{d\lambda_{0s}}{dt} \quad (3)$$

$$v'_{qr} = r'_r i'_{qr} + (\omega - \omega_r) \lambda'_{dr} + \frac{d\lambda'_{qr}}{dt} \quad (4)$$

$$v'_{dr} = r'_r i'_{dr} - (\omega - \omega_r) \lambda'_{qr} + \frac{d\lambda'_{dr}}{dt} \quad (5)$$

$$v'_{0r} = r'_r i'_{0r} + \frac{d\lambda'_{0r}}{dt} \quad (6)$$

Sendo,

$$\lambda_{qs} = L_{ls} i_{qs} + L_M (i_{qs} + i'_{qr}) \quad (7)$$

$$\lambda_{ds} = L_{ls} i_{ds} + L_M (i_{ds} + i'_{dr}) \quad (8)$$

$$\lambda_{0s} = L_{ls} i_{0s} \quad (9)$$

$$\lambda'_{qr} = L'_{lr} i'_{qr} + L_M (i_{qs} + i'_{qr}) \quad (10)$$

$$\lambda'_{dr} = L'_{lr} i'_{dr} + L_M (i_{ds} + i'_{dr}) \quad (11)$$

$$\lambda'_{0r} = L'_{lr} i'_{0r} \quad (12)$$

Utilizando os eixos de referência  $qd0$ , a equação para obtenção do torque eletromagnético resulta em (13).

$$T_e = \left(\frac{3}{2}\right) \left(\frac{P}{2}\right) L_M (i_{qs} i'_{dr} - i_{ds} i'_{qr}) \quad (13)$$

Buscando uma analogia do campo do rotor de uma máquina de indução com o campo de uma máquina de corrente contínua, utiliza-se um eixo de referência  $qd0$  girando à velocidade síncrona, no qual o eixo  $d$  está alinhado com o campo do rotor. Isso garante com que a componente  $q$  do campo gerado pelas correntes do rotor,  $\lambda'_{qr}$ , seja nula. Resultando, a partir de (10), (14).

$$i'_{qr} = -\frac{L_M}{(L'_{lr} + L_M)} i_{qs} \quad (14)$$

A partir de (14) é possível obter uma simplificação para equação do torque eletromagnético (13), de modo que está seja uma função apenas da componente no eixo  $d$  do fluxo do rotor e da componente no eixo  $q$  da corrente do estator (15).

$$T_e = \left(\frac{3}{2}\right) \left(\frac{P}{2}\right) \frac{L_M}{(L'_{lr} + L_M)} i_{qs} \lambda'_{dr} \quad (15)$$

Portanto, caso o fluxo do rotor seja mantido constante, torna-se possível controlar o torque eletromagnético diretamente através do controle da componente  $q$  da corrente do estator.

Para que o fluxo do rotor seja mantido constante é necessário que a componente no eixo  $q$  seja mantida nula e que a componente no eixo  $d$  seja mantida constante, ou seja, as respectivas derivadas em função do tempo devem ser iguais à zero. De (4) e (5), estabelecendo as tensões de rotor nulas (rotor gaiola de esquilo), obtêm-se que  $i'_{dr}$  também deve ser mantida igual à zero e que a diferença entre a velocidade síncrona e velocidade do rotor deve seguir a relação (16).

$$\omega_s - \omega_r = \left(\frac{r'_r}{L'_{lr} + L_M}\right) \left(\frac{i_{qs}}{i_{ds}}\right) \quad (16)$$

Sendo  $\lambda^e_{qm}$  o fluxo medido no eixo  $q$  e  $\lambda^e_{dm}$  o fluxo medido no eixo  $d$ , a partir dos sensores no entreferro, são, então, obtidos o módulo ( $\lambda'_{dr}$ ) e o ângulo ( $\theta$ ) do fluxo concatenado do rotor que gira a velocidade síncrona utilizando (17) e (18) (Krause et al., 2002). O índice sobrescrito “e” é utilizado para explicitar que são grandezas obtidas utilizando os eixos de referência  $qd0$  estacionários.

$$\lambda'_{dr} = \sqrt{(\lambda^e_{dr})^2 + (\lambda^e_{qr})^2} \quad (17)$$

$$\theta = \arcc(\lambda^e_{qr} - j\lambda^e_{dr}) + \frac{\pi}{2} \quad (18)$$

Onde,

$$\lambda^e_{qr} = \frac{(L'_{lr} + L_M)}{L_M} \lambda^e_{qm} - L'_{lr} i^e_{qs} \quad (19)$$

$$\lambda^e_{dr} = \frac{(L'_{lr} + L_M)}{L_M} \lambda^e_{dm} - L'_{lr} i^e_{ds} \quad (20)$$

### 3.2. Rede Neural Perceptron Multicamadas(PMC)

Como entradas da rede PMC são utilizadas as tensões e correntes nos eixos de referência  $qd0$  estacionários em um momento  $t$  e em um momento  $t_2$  exatamente anterior a  $t$ . A utilização dessas variáveis de entrada em momentos subsequentes de tempo baseia-se na característica dinâmica da relação entre correntes, tensões e velocidade (variável de saída) e na dependência da velocidade com o fluxo. O motor de indução trifásico representado pelos parâmetros da Tabela 1 possui as fases idealmente equilibradas e uma configuração não aterrada, portanto, as componentes “0” podem ser desprezadas, resultando que, a rede PCM proposta possui 8 entradas.

São propostas 5 arquiteturas diferentes com, respectivamente, 10, 15, 20, 25 e 30 neurônios em uma única camada escondida e um neurônio na camada de saída. A função de ativação da camada escondida é a tangente hiperbólica para todos os neurônios e a da camada de saída é a função linear. Para o treinamento da rede neural foram utilizados os dados obtidos através da simulação do Controle Vetorial Direto e o método de treinamento foi o *Levenberg-Marquardt Backpropagation* (Silva et al., 2010) com uma precisão de  $1.10^{-7}$ . A Figura 1 apresenta a velocidade do rotor normalizada utilizada como resposta desejada da saída da rede PMC.

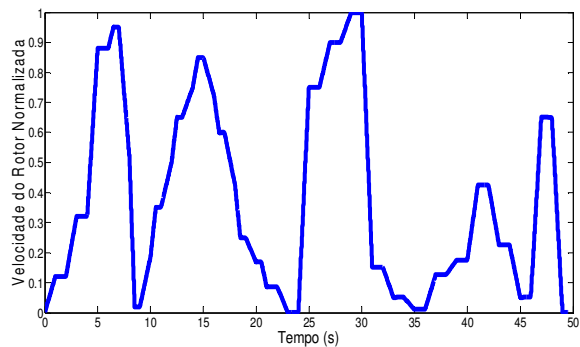


Figura 1. Velocidade do rotor normalizada utilizada para treinamento da rede PMC.

Do conjunto de dados obtidos da simulação, 50% foi utilizado para treinamento e 50% para teste e verificação da capacidade de generalização. Esses

grupos foram divididos de forma aleatória e cada arquitetura foi treinada 3 vezes.

#### 4. Resultados

##### 4.1. Controle Vetorial Direto

A Figura 2 apresenta a velocidade de referência e a velocidade do rotor. Pela grande proximidade entre as curvas, somente na Figura 3, onde é apresentada uma ampliação de um dos patamares, é possível observar como a velocidade do rotor converge rápido à referência e também o quão pequeno é o *overshoot*, aproximadamente 0,3%.

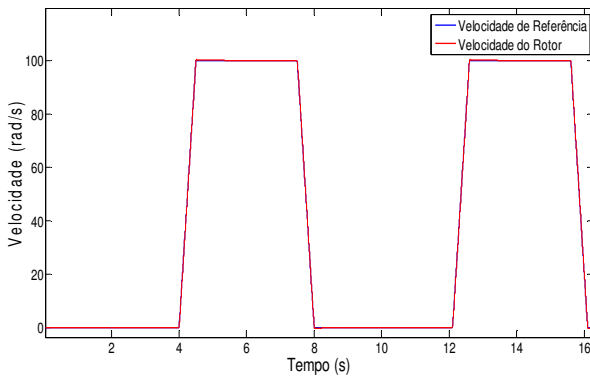


Figura 2. Velocidade de Referência e Velocidade do Rotor – Controle Vetorial Direto

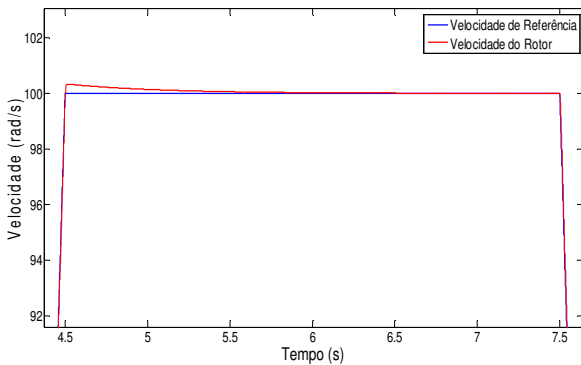


Figura 3. Velocidade de Referência e Velocidade do Rotor – Controle Vetorial Direto (Ampliação)

A Figura 4 apresenta novamente as curvas de velocidade de referência e velocidade do rotor ampliadas, porém, no tempo igual à 8 segundos é adicionada uma carga de 198 N.m ao rotor. Observa-se que há uma boa resposta do sistema de controle projetado, já que, além do pequeno *overshoot*, já observado anteriormente, após a aplicação da carga a velocidade retornou rapidamente ao referencial de 100 rad/s.

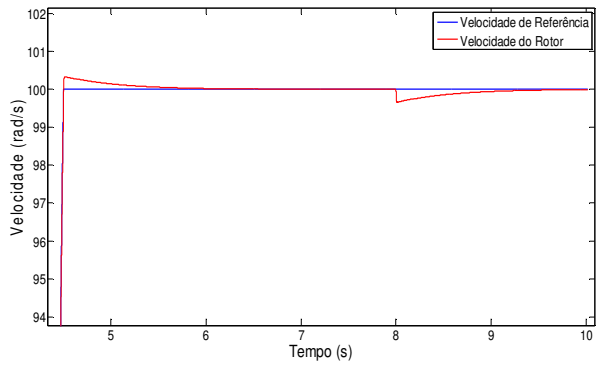


Figura 4. Velocidade de Referência e Velocidade do Rotor – Controle Vetorial Direto com Carga no Rotor de 198 N.m em t = 8 segundos (Ampliação)

##### 4.2. Rede Perceptron Multicamadas

Denominando de T1, T2 e T3, respectivamente, o primeiro, o segundo e o terceiro treinamento realizados para cada uma das arquiteturas propostas, a Tabela 2 apresenta, então, os valores obtidos para o erro quadrático médio de treinamento ( $e_{TRAIN}$ ) e para o erro quadrático médio de teste ( $e_{TESTE}$ ) para a precisão estabelecida e a Tabela 3 apresenta o número de épocas para cada treinamento. Os dados apresentados nessas tabelas são de essencial importância para a definição da melhor arquitetura.

Tabela 2. Erro Quadrático Médio de Treinamento ( $e_{TRAIN}$ ) e Erro Quadrático Médio de Teste ( $e_{TESTE}$ )

Treinamento		Arquitetura (N° de neurônios na camada escondida)				
		10	15	20	25	30
T1	$e_{TRAIN}$	1,45e-4	1,05e-4	2,86e-5	7,20e-5	1,65e-5
	$e_{TESTE}$	5,43e-3	1,20e-4	4,87e-5	1,22e-4	3,52e-5
T2	$e_{TRAIN}$	1,32e-4	3,03e-4	4,03e-5	1,41e-5	1,77e-5
	$e_{TESTE}$	2,24e-4	3,17e-4	5,53e-5	5,06e-5	2,58e-5
T3	$e_{TRAIN}$	6,15e-4	8,48e-5	1,01e-4	4,00e-5	1,58e-4
	$e_{TESTE}$	6,23e-4	1,68e-3	1,13e-4	7,57e-4	1,79e-4

Tabela 3. Número de Épocas para cada Treinamento

Arquitetura (N° de neurônios na camada escondida)	T1	T2	T3
10	114	152	34
15	121	72	163
20	524	407	140
25	112	595	258
30	259	308	76

Os dados da Tabela 3 permitem observar a grande variabilidade do número de épocas. Aumentando o número de neurônios, o número de épocas mostra uma tendência a ser maior, porém, como a matriz de pesos é inicializada aleatoriamente, em alguns treinamentos o processo encontra rapidamente um

ponto de mínimo para a função erro quadrático médio independentemente do número de neurônios. Associando essa análise com os dados da Tabela 2, percebe-se que os piores resultados são encontrados para estes treinamentos que levaram um menor número de épocas.

Os resultados obtidos e expostos na Tabela 2 e na Tabela 3 levam a conclusão de que as melhores arquiteturas são as com 20 e 30 neurônios na camada escondida, ambas com os resultados de treinamento obtidos em T1.

Para uma segunda análise da melhor arquitetura, as figuras 5 e 6 apresentam os respectivos histogramas de erro para as redes com 20 e 30 neurônios obtidos no treinamento T1.

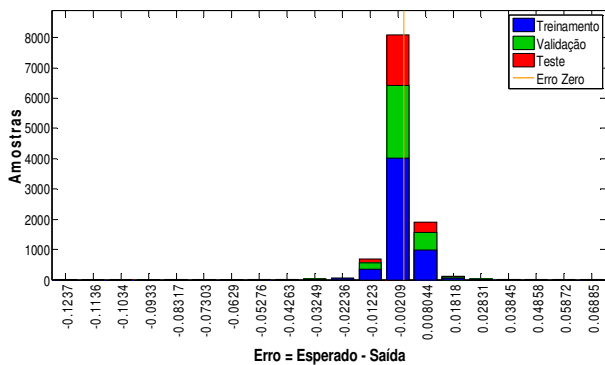


Figura 5. Histograma de Erro - Rede PMC com 20 neurônios na camada escondida e dados de treinamento obtidos em T1

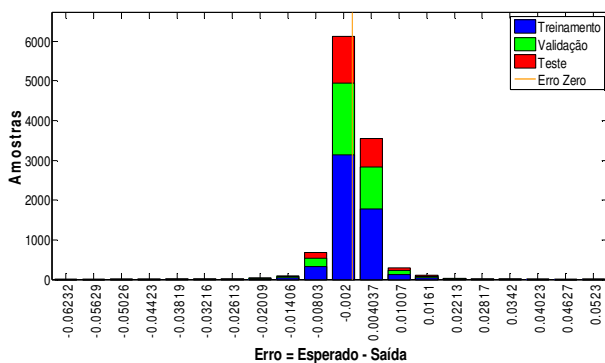


Figura 6. Histograma de Erro - Rede PMC com 30 neurônios na camada escondida e dados de treinamento obtidos em T1

Ambos os histogramas apresentaram uma característica desejável, a concentração na região próxima ao erro zero. Porém, ao compará-los, nota-se que apesar do histograma para a rede PMC com 20 neurônios apresentar uma parcela maior dos dados centralizada no valor de módulo mínimo, essa rede possui distâncias do erro zero aos extremos positivo e negativo consideravelmente maior que as distâncias para a rede PMC com 30 neurônios. Ou seja, para a rede PMC com 20 neurônios algumas amostras apresentam um erro maior que o erro máximo obtido para a rede com 30 neurônios.

Portanto, a melhor rede PMC dentre as arquiteturas propostas é a rede com 30 neurônios na camada intermediária e dados de treinamento obtidos em T1. A Figura 7 apresenta a velocidade do rotor

normalizada utilizada para treinamento e a saída dessa rede neural. A partir dessa figura, confirma-se a grande proximidade da saída da rede com o valor esperado.

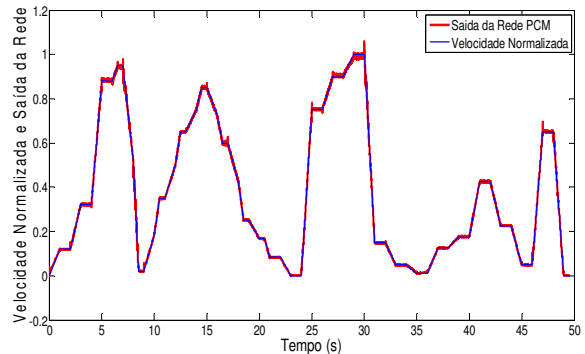


Figura 7. Velocidade do rotor normalizada (azul) e Saída da Rede PMC (vermelho)

### 4.3. Controle Vetorial Direto Sensorless

Na Figura 7 fica clara a presença de pequenas oscilações de alta frequência na saída da rede neural entorno do valor esperado. A inserção direta dessa rede como estimador da velocidade no sistema de controle resulta em problemas de estabilidade, porque, mesmo que essas oscilações sejam muito pequenas em módulo, a diferença entre os sinais do erro real da velocidade do rotor e do erro da velocidade estimada compromete completamente o funcionamento dos controladores PI. Para reduzir os efeitos dessas oscilações, um filtro passa-baixas com constante de tempo igual a 0,0667 é inserido à saída da rede neural.

É essencial notar que no projeto desse filtro é preciso levar em consideração que o estabelecimento de uma constante de tempo muito pequena permite a existência de oscilações, principalmente nos períodos em que a velocidade de referência é constante. Enquanto que, caso seja estabelecida uma constante de tempo muito grande, a saída da rede perderá importantes componentes, principalmente nos momentos em que a velocidade de referência possui um valor alto de derivada, fazendo com que surjam erros na estimação e, conseqüentemente, erros intoleráveis entre a referência e a velocidade do motor. Fica então, explícita a necessidade de uma análise minuciosa do compromisso “precisão x estabilidade/suavidade” durante o projeto desse filtro.

Além disso, durante o exato momento da partida, o estimador (rede neural associada ao filtro) mostra-se incapaz de estimar corretamente a velocidade do rotor. Para reverter essa limitação, a estimação da velocidade é inserida no sistema através de uma chave acionada quatro segundos após a partida. Esse chaveamento possui como consequência a necessidade de que seja definido um valor de saturação para o erro entre a velocidade do rotor e a

velocidade de referência, buscando que essa mudança brusca não exceda um determinado valor e cause saturação da fonte de alimentação, gerando problemas na convergência da velocidade do rotor para a referência.

A Figura 8 apresenta o resultado obtido para a simulação em ambiente MatLab/Simulink do controle da velocidade do motor de indução utilizando a estimação da velocidade através da rede neural proposta e todas as modificações necessárias discutidas.

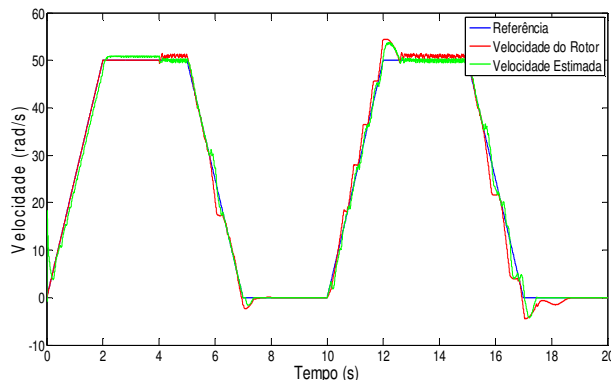


Figura 8. Velocidade de Referência (azul), Velocidade do Rotor (vermelho) e Saída da Rede PMC (verde)

A Figura 9 apresenta uma ampliação de um dos patamares.

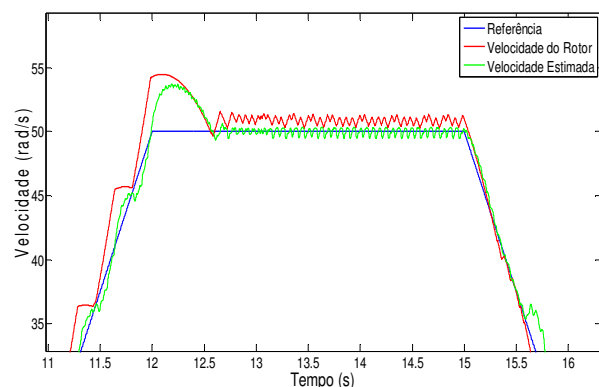


Figura 9. Velocidade de Referência (azul), Velocidade do Rotor (vermelho) e Saída da Rede PMC (verde) – Ampliação

Constata-se que a velocidade do rotor oscila e que essa oscilação possui uma amplitude próxima de 2% da referência. Além disso, desprezando a oscilação, observa-se um erro médio de aproximadamente 1,25% no patamar e um pequeno atraso na subida.

## 5. Conclusões

O controle vetorial em si, como esperado, apresentou uma resposta excelente, mesmo quando o motor foi submetido a mudanças do torque de carga. Entretanto, a inserção da rede PMC como estimador da velocidade no sistema de controle causou alguns efeitos indesejados. A rede proposta apresentou pequenas oscilações de alta frequência em torno do

valor real. Quando essa rede é inserida no sistema de controle vetorial, substituindo a realimentação, essas oscilações causam interferências consideráveis no controle, mesmo após a inserção de um filtro passa-baixas na sua saída. Para permitir a estabilidade do funcionamento dos controladores PI, são inseridos blocos de saturação para o erro.

O resultado final se mostrou satisfatório. Nos patamares da velocidade de referência ainda há uma oscilação de amplitude igual a 2% da mesma e um erro médio de 1,25% mas, por exemplo, para a velocidade de 50 rad/s isso representa a velocidade média de aproximadamente 50,6 rad/s. Erro que pode ser tolerado de acordo com a aplicação e que se torna irrelevante perante as vantagens de substituir um sensor mecânico por uma ferramenta computacional simples, de rápido processamento e baixo custo.

## Referências Bibliográficas

- Fitzgerald, A. E. ; Kingsley, C.; Umans, S. D., Máquinas Elétricas. Bookman, 2006;
- Krause, P. C.; Wasynczuk, O.; Sudhoff, S. D., Analysis Of Electric Machinery And Drive Systems – IEEE PRESS, 2002;
- Chee-Mun Ong, Dynamic Simulation of Electric Machinery: Using MatLab/Simulink. Prentice Hall PTR, 1998;
- Dazhi Wang; Renyuan Tang; Hui Jin; Jie Yang, Sensorless-speed control strategy of induction motor based on artificial neural networks. Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on , vol.5, no., pp.4467,4471 Vol.5, 15-19 June 2004;
- Jing Yang; Ligu Wang; Dianguo Xu; Bing Xue, Sensorless Speed Estimation for Line-connected Induction Motor Based on Recurrent Multilayer Neural Network. Automation and Logistics, 2007 IEEE International Conference on , vol., no., pp.2013,2018, 18-21 Aug. 2007;
- Orlowska-Kowalska, T.; Kowalski, C.T., Neural network application for flux and speed estimation in the sensorless induction motor drive. Industrial Electronics, 1997. ISIE '97., Proceedings of the IEEE International Symposium on , vol., no., pp.1253,1258 vol.3, 7-11 Jul 1997;
- Michalewicz, Z.; Fogel, D. B., How to Solve It: Modern Heuristics. Springer, Second Edition, 2004, Germany;
- Silva, I. N.; Spatti, D. H.; Flauzino, R. A., Redes Neurais Artificiais: Para Engenharia e Ciências Aplicadas, São Paulo, Artliber, 2010;
- Park, R. H., Two-Reaction Theory of Synchronous Machines – Generalized Method of Analysis. Part I, AIEE Transactions, Vol. 48, July 1929, pp. 716-727;
- Stanley, H. C., An Analysis of the Induction Motor. AIEE Transactions, Vol. 57 (Supplement), 1938, pp. 751-755;
- Kron, G., Equivalent Circuits of Electric Machinery, John Wiley and Sons, New York, 1951;
- Brereton, D. S.; Lewis, D. G.; Young, C. G., Representation of Induction Motor Loads During Power System Stability Studies. AIEE Transactions, Vol. 76, August 1957, pp. 451-461;
- Krause, P. C.; Thomas, C. H., Simulation of Symmetrical Induction Machinery. IEEE Transactions on Power Apparatus and Systems, Vol. 84, November 1965, pp. 1038-1053;
- Hui Deng; Jing Yang; Dianguo Xu, Sensorless Speed Estimation for Induction Motor based on Neural Network. Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on , vol.1, no., pp.2891,2895, 0-0 0;
- Liping Fan; Yi Liu, Neural network based speed identification for speed-sensorless induction motor drives. Control and Decision Conference (CCDC), 2010 Chinese , vol., no., pp.3093,3097, 26-28 May 2010