

CONTROLE COM ESTRUTURA PID FUZZY APLICADO A PLANTAS INDUSTRIAIS

MARCELO O. DE LIMA, ARTHUR P. S. BRAGA, PAULO P. PRAÇA, WELTON DA SILVA LIMA, RYCHELL MOREIRA SOUSA, LUIS FERNANDO EUFRÁSIO PEIXOTO MOTA

*Departamento de Engenharia Elétrica, Centro de Tecnologia, Universidade Federal do Ceará
Campus do Pici, Caixa Postal 6001 – CEP 60.455-760 Fortaleza, CE, Brasil*

*E-mails: marcelo@ucfacil.com.br, arthurp@dee.ufc.br, paulopp@dee.ufc.br,
welton.lima@fisica.ufc.br, rychell@gmail.com, fernando.epml@gmail.com*

Abstract In this article is proposed the adjustment of the parameters of a inference system structure fuzzy PID and fuzzy PI structure in order to improve the response of the control system. The structures PID fuzzy and PI fuzzy were tested respectively in the simulation of a wind turbine and on the simulation of a DC-DC Buck quadratic. The results were compared with the same systems controlled with a Ziegler-Nichols tuning, and showed improvement on the PID fuzzy and PI fuzzy structure proposed in this paper.

Keywords Fuzzy Control, Adaptive Control, Methods in Control, Feedback systems.

Resumo Neste artigo é proposto o ajuste de parâmetros do sistema de inferência empregado na estrutura PID fuzzy e na estrutura PI fuzzy, de forma a melhorar a resposta do sistema de controle. As estruturas PID fuzzy e PI fuzzy foram testadas, respectivamente, na simulação de uma bancada de testes de um aerogerador e na simulação de um conversor CC-CC Buck quadrático. Os resultados comparativos entre as estruturas fuzzy e os sintonizado por Ziegler-Nichols simulados demonstram os ganhos obtidos com a variação da estrutura dos controladores PID fuzzy e PI fuzzy proposta neste trabalho.

Palavras-chave Controle Fuzzy, Controle Adaptativo, Métodos em controle, Sistemas realimentados.

1 Introdução

Os controladores PID (Proporcional – Integral – Derivativo) são amplamente utilizados no controle de processos com realimentação, havendo trabalhos que comentam que esta estrutura chega a ser utilizada em até 90% dos laços de controle [1]. O principal problema observado no uso destes controladores é a sintonia de seus parâmetros para que sejam produzidas respostas suficientemente rápidas sem ocorrência de sobre-sinal [2], [3]. As metodologias clássicas para a sintonia destes controladores, como, por exemplo, o método de *Ziegler-Nichols* [2], normalmente dependem de levantamento do modelo matemático do processo que pode se revelar impreciso e variável, conforme o ponto de operação. Uma alternativa, está no uso de técnicas que buscam contornar a etapa de levantamento dos modelos dos processos, tendência já observada em trabalhos atuais que tratam da sintonia de controladores PID.

Uma destas técnicas de inteligência computacional, IC, que vem sendo muito empregada na sintonia dos parâmetros do PID é a lógica *fuzzy* [5], [8], [9], [10]. Neste trabalho aplicou-se um procedimento inspirado na estrutura PID *fuzzy* proposta por [3], [4] que permite o ajuste *on-line* dos parâmetros do controlador. Uma nova abordagem foi obtida com a adaptação do sistema de inferência empregado na estrutura PID *fuzzy*, buscando satisfazer um compromisso entre desempenho e complexidade computacional que facilite a implementação desta estrutura de controle em sistemas embarcados.

A adaptação do sistema de inferência *fuzzy* ocorre na escolha da disposição dos conjuntos de

saída do sistema, quando há redução no número de conjuntos, visando obter uma resposta melhor das plantas, sendo objetos deste trabalho um motor CC e um conversor CC-CC. Para realizar esta adaptação no motor CC, utilizou-se o método da Aprendizagem por Reforço (RL – *Reinforcement Learning*) [15], [16] para o ajuste dos conjuntos de saída, enquanto que no conversor CC-CC, utilizou-se os parâmetros de *Ziegler-Nichols* para a escolha dos conjuntos de saída.

As simulações realizadas comparam diferentes estruturas PID *fuzzy* e o PID sintonizado por *Ziegler-Nichols* em um sistema para teste de aero-gerador e em um sistema conversor CC-CC Buck quadrático. Os resultados demonstram os ganhos obtidos com a variação da estrutura PID *fuzzy* proposta neste trabalho.

O restante do artigo está dividido como se segue: na Seção 2, temos a descrição do sistema a ser controlado; na Seção 3, a estrutura PID *fuzzy* é descrita, focando no sistema de inferência; a Seção 4 aborda o algoritmo AR utilizado para ajustar o sistema de inferência; na Seção 5 descreve as simulações realizadas, e os resultados obtidos; a Seção 6 encerra o artigo com as conclusões.

2 Descrição dos Sistemas

2.1. O motor CC

Um dos sistemas a ser controlado é um motor de corrente contínua (CC) de média potência. Motores deste tipo podem ser usados em várias aplicações [6], tais como esteiras e centrífugas. No caso proposto, o motor é acoplado ao eixo de um aero gerador simulando o vento, como ilustra a Figura 1.

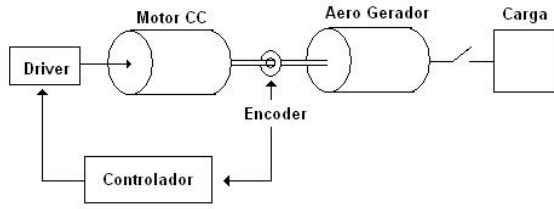


Figura 1. Sistema a ser controlado

No motor CC em questão, a tensão de campo é mantida constante e o controle de velocidade será através da tensão no circuito de armadura [7]. A modelagem do motor CC tratado é dada por:

$$V_i(s) = R_a I_a(s) + sL_a I_a(s) + E_g(s) \quad (1)$$

$$T_w(s) = B\omega(s) + T_j(s) \quad (2)$$

$$T_j(s) = Js\omega(s) \quad (3)$$

$$T_g(s) = T_j(s) + T_w(s) + T_L(s) \quad (4)$$

$$T_g(s) = KfI_a(s) \quad (5)$$

$$E_g(s) = Kf\omega(s) \quad (6)$$

em que:

- R_a é a resistência de armadura;
- L_a é a indutância de armadura;
- $I_a(s)$ é a corrente de armadura;
- $V_i(s)$ é a tensão de armadura;
- $E_g(s)$ é a força contra-eletromotriz induzida no motor;
- J é o momento de inércia equivalente do motor e da carga referido ao eixo do motor;
- K é uma constante de torque do motor;
- f é o fluxo magnético;
- $\omega(s)$ é a velocidade angular;
- B é o coeficiente de atrito viscoso;
- $T_g(s)$ é o torque gerado pelo motor;
- $T_w(s)$ é o chamado conjugado de perdas;
- $T_j(s)$ refere-se à inércia do eixo;
- $T_L(s)$ refere-se à carga;
- $T_j(s)$ é a parcela não linear.

Para se obter os parâmetros do motor, foram realizados ensaios em corrente contínua (CC) e corrente alternada (CA). Os mesmos estão resumidos na Tabela 1.

Através das equações que relacionam as variáveis em questão, podemos encontrar a função de transferência do sistema e o modelo do mesmo. O diagrama de blocos do motor é mostrado na Figura 2.

Tabela 1. Parâmetros do Motor CC

Parâmetro	Valor	Unidade
B	0,13381	$N.m.s/rad$
E_g	218,32	V
I_a	1,95	A
J	0,29077	$N.m.s^2$
L_a	14,95	mH
R_a	0,864	Ω

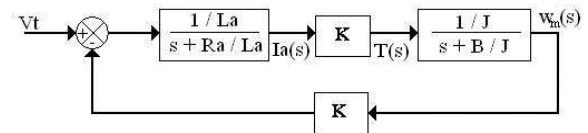


Figura 2. Diagrama de Blocos do Motor

O bloco de entrada da planta diz respeito à parte elétrica do motor, cuja saída representa a corrente de armadura. O bloco de saída diz respeito à parte mecânica do motor, cuja saída representa a velocidade angular. A realimentação é a força contra-eletromotriz, que é proporcional a velocidade angular.

2.2. O conversor CC-CC Buck quadrático

Os conversores CC-CC do tipo *Buck* geralmente são empregados em sistemas que não demandam um ganho alto entre a entrada e a saída. Quando este ganho é alto, o ciclo de trabalho do conversor opera em uma faixa muito estreita, comprometendo a eficiência do conversor. Para amenizar este problema, é comum o uso de transformadores de alta frequência, utilizando-se de sua relação de transformação para deixar a saída dentro de um limiar desejado e utilizando o ciclo de trabalho para um ajuste fino.

Os conversores *Buck quadráticos* são outra alternativa para conversores de alto ganho[18]. Tais conversores tem a característica de obter alto ganho entre a entrada e a saída sem a necessidade de uso de transformadores de alta frequência. Tal característica, no então, faz com que haja um trabalho redobrado na escolha do ciclo de trabalho, uma vez que uma pequena variação no ciclo de trabalho acarreta uma grande mudança da relação entrada/saída.

O circuito da Figura 3 ilustra um circuito de um conversor *Buck* quadrático.

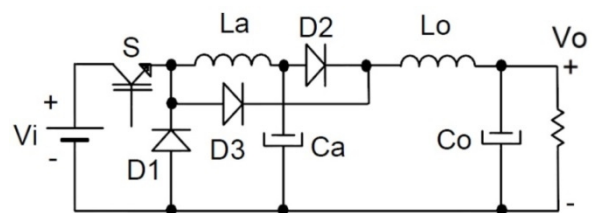


Figura 3. Conversor CC-CC Buck Quadrático

3 O Controlador PID Fuzzy

Os controladores PID convencionais, com parâmetros invariáveis, podem apresentar um comprometimento de seu desempenho em processos que variam seu ponto de operação. A estrutura PID *fuzzy* proposta por [3], [4] é uma alternativa interessante para tratar estes processos por realizar *on-line* o ajuste dos parâmetros do PID. A Figura 4 ilustra a estrutura do controlador PID *fuzzy* [11].

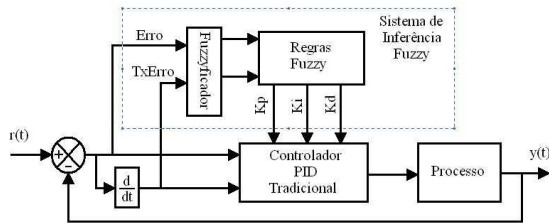


Figura 4. Estrutura do Controlador PID Nebuloso

3.1. O Sistema de Inferência Fuzzy

O Sistema de Inferência *Fuzzy* (SIF) é composto por um conjunto de regras que descrevem o comportamento desejado do controlador. Estas regras seguem o formato [9], [10]:

$$\text{Se} < \text{Condição} > \text{Então} < \text{Conclusão} > \quad (7)$$

Os termos *Condição* e *Conclusão* são descritos por conjuntos *fuzzy* [8] que diferem da teoria clássica de conjuntos por terem a pertinência de um elemento x_i dada por uma função de pertinência, $\mu(x_i)$, tal que:

$$0 \leq \mu(x_i) \leq 1 \quad (8)$$

Com a função de pertinência, os conjuntos *fuzzy* podem representar conceitos. Assim, as regras no formato de (7) mapeiam conceitos relativos às variáveis de entrada com conceitos relativos às variáveis de saída, onde os conjuntos A_i e B_i são

condições, e A'_m é uma conclusão [3], [4]. A Figura 5 ilustra tal conceito.

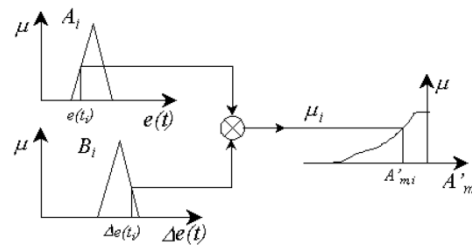


Figura 5. Exemplo da implementação de uma regra *fuzzy*

O procedimento de inferência combina as saídas de todas as regras, e gera a saída do sistema de inferência *fuzzy* [5]. Como as operações no SIF são realizadas sobre conjuntos *fuzzy*, as entradas e as saídas passam, respectivamente, por etapas de *fuzzificação* e *defuzzificação*, para transformar escalares em conjuntos *fuzzy*, e vice-versa [5], [17]. Nos SIFs tratados neste trabalho, as entradas são o erro, $e(t)$, e a variação de erro, $De(t)$, na saída do processo; e as saídas são os parâmetros K_p , K_i e K_d do controlador PID para a planta do motor CC. Para a planta do conversor CC-CC o parâmetro K_d é zerado, obtendo-se assim um controlador PI. Portanto, os parâmetros K_p , K_i são a saída do SIF para a planta do conversor CC-CC.

3.2. Resultados

As seguintes simulações foram realizadas para avaliação comparativa no controle do motor cc do sistema da Figura 1:

- Um PID tradicional sintonizado pelo método clássico de *Ziegler-Nichols*;
- Um o controlador PID *fuzzy* com a estrutura proposta por [3], [4];

A Figura 6 exhibe as respostas em dois pontos de operação (*setpoint* em 100 rad/s e 130 rad/s) do motor CC ilustrado no sistema da figura 1, objetivando mostrar o comportamento dos controladores PID sintonizado pro ZN e o PID *fuzzy*.

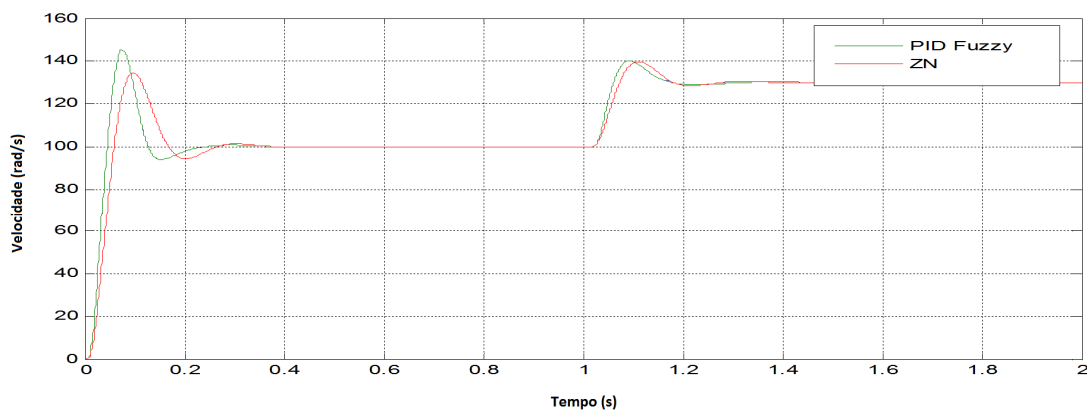


Figura 6. Resposta do motor com o controlador PID sintonizado por ZN e com o controlador PID *fuzzy*

Figuras 7 e 8 exibem, respectivamente, as respostas de corrente e tensão na saída da planta do conversor CC-CC, cuja tensão de entrada de 180V, controladas com um PI e um PI *fuzzy*. Ambas saídas são controladas em 24 Volts. Inicialmente, o conversor é ligado com 50% da carga. No instante

10ms, é adicionada uma carga ao sistema para que o conversor opere em 100% da carga nominal. No instante 33ms a carga que foi adicionada é removida, deixando o conversor novamente operando em 50% da carga nominal.

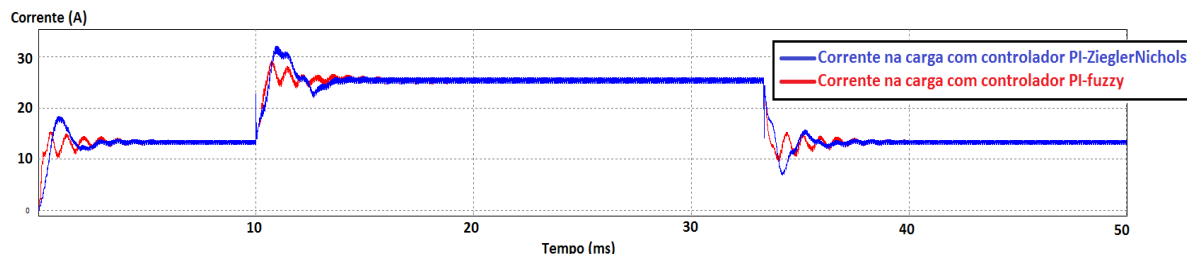


Figura 7. Corrente na saída do conversor CC-CC com os controladores PI e PI *fuzzy*

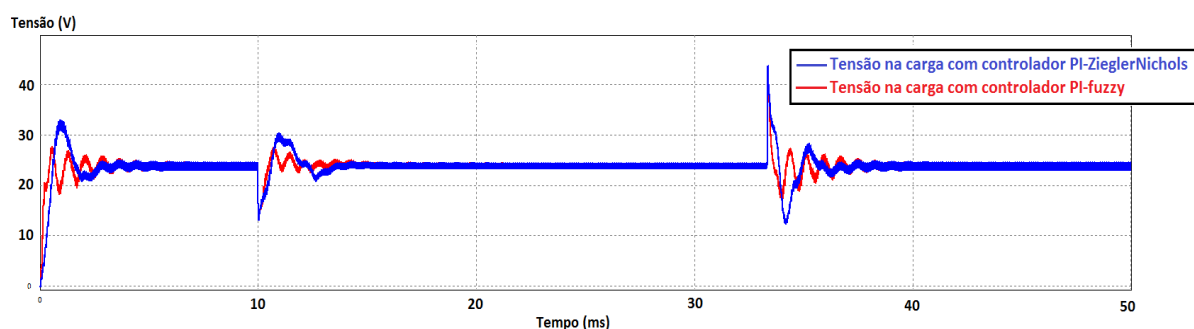


Figura 8. Tensão na saída do conversor CC-CC com os controladores PI e PI *fuzzy*

4 Aprendizado por Reforço

O Aprendizado por Reforço (RL – *Reinforcement Learning*) é uma metodologia muito atraente para solucionar problemas de tomadas de decisão [15], [16].

Em RL, agente e ambiente interagem de forma que cada ação do agente, a partir de um estado do ambiente, recebe uma recompensa (o sinal de reforço) que indica se o agente selecionou uma ação correta ou incorreta [16].

Um exemplo de algoritmo de aprendizado por reforço utilizado é o CARLA (*Continuous Action Reinforcement Learning Automaton*) [14], [16], que possui os seguintes passos:

- 1- Inicializa-se a função de geração de ação com uma distribuição uniforme.
- 2- É gerada uma ação r .
- 3- A resposta do ambiente é calculada (observação da ação ou função custo).
- 4- O sinal de reforço é calculado usando a qualidade da resposta.
- 5- A função de geração de ação é corrigida usando esse sinal de reforço.
- 6- Se o número de iterações não foi atingido, volta para o passo 2. Senão, termina o algoritmo.

Supondo que a variável X pertence ao universo $[X_{\min}, X_{\max}]$, a função de geração de ação é construída da seguinte maneira:

$$f(x,1) = \frac{1}{X_{\max} - X_{\min}} \quad (9)$$

Na n ésima iteração, a ação r é gerada usando a seguinte equação:

$$F(r, n) = \int f(x, n) dx = z(n) \quad (10)$$

Onde $z(n)$ é um número aleatório entre 0 e 1. A ação r então é aplicada, e é calculada a função custo $J(n)$, utilizando-se nesse caso a integral do erro quadrático. $J(n)$ é obtida de acordo com a seguinte equação:

$$J(n) = \int e^2(t) dt \quad (11)$$

Com a função custo $J(n)$, calcula-se o sinal de reforço $B(n)$:

$$B(n) = \min \left(\max \left(0, \frac{J_{med} - J(n)}{J_{med} - J_{min}} \right), 1 \right) \quad (12)$$

onde J_{med} e J_{min} são, respectivamente, os custos médio e mínimo calculados com R iterações aleatórias sem a correção da função de geração de ação.

A gaussiana centrada na ação r é a função usada para a correção da função de geração de ação. Representa-se da seguinte maneira:

$$H(x, r) = \frac{g_h}{X_{max} - X_{min}} \times \dots \times \exp \left(-\frac{1}{2} \times \frac{(x - r)^2}{(g_w (X_{max} - X_{min}))^2} \right) \quad (13)$$

Finalmente, para chegar ao passo 6 do algoritmo, a função de geração de ação é corrigida da seguinte maneira:

$$f(x, n+1) = \alpha [f(x, n) + B(n) \cdot H(x, r)] \quad (14)$$

sendo α uma constante que faz com que a área sob a curva da $f(x, n+1)$ seja sempre 1.

5 Controlador PID Fuzzy Ajustável

Neste trabalho, buscou-se melhorar a resposta do controlador PID *fuzzy* ao se ajustar os conjuntos CRISP de saída do sistema de inferência *fuzzy*. Para a planta do motor CC, tal ajuste foi realizado através do algoritmo CARLA. Para o conversor CC-CC, o ajuste dos conjuntos CRISP foi feito simplesmente somando 50% do valor de *Ziegler-Nichols* para o conjunto grande e subtraindo 50% do valor de *Ziegler-Nichols* para o conjunto pequeno.

O algoritmo CARLA gera, a cada iteração, uma ação. Esta ação é a escolha de valores para os três parâmetros do controlador PID (K_p , K_i , K_d). Quando essa ação é gerada, o sistema é observado, e uma recompensa para tal ação é obtida. Na sequência, ocorre a atualização das funções de densidade de probabilidade. É importante frisar que, da primeira iteração até a iteração R, não há atualização das funções densidade de probabilidade, ou seja, estas têm derivada zero em todo o seu universo, significando que a probabilidade de ocorrência de qualquer valor, dentro dos possíveis, é idêntica [14], [16]. Este procedimento é adotado para a obtenção do vetor J inicial (vetor de recompensas). Assim, quando o sistema é observado, é gerada a recompensa (11) (função $J(n)$) e de posse desta, juntamente com (12) e (13), a função de densidade de probabilidade passa a ser atualizada (14). Caso a recompensa seja positiva

em relação às R últimas obtidas, a probabilidade desta ação ser tomada será maior, caso contrário, será menor.

5.1 Configuração do Sistema Fuzzy

O sistema fuzzy é composto de:

- 2 entradas: erro e taxa de erro;
- 3 saídas: K_p , K_i e K_d (K_d = zero para o conversor);
- Sistema tipo Mamdani com 25 regras para o motor;
- Defuzzificação através do método dos conjuntos CRISP.

5.2 Funções de Pertinência

Nas Figuras 9 e 10 tem-se, respectivamente, as funções de pertinência das variáveis de entrada, que são o erro e a variação do erro, e saída, que são os ganhos de cada parcela do controlador PID.

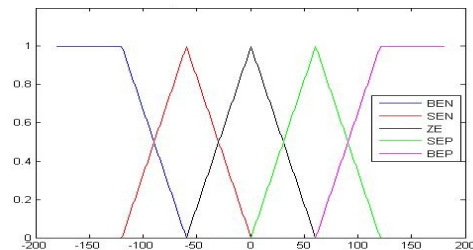


Figura 9 - Funções de Pertinência das Variáveis de Entrada

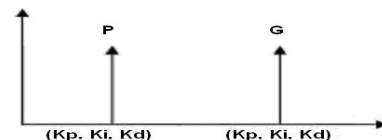


Figura 10 - Funções de Pertinência das Variáveis de Saída

5.3 Regras Fuzzy

O conjunto de regras para o ajuste dos parâmetros K_p , K_i e K_d para o controlador PID da planta do motor, estão resumidas nas Tabelas 2, 3 e 4, respectivamente. Para a planta do conversor CC-CC, é utilizado o centro das tabelas 2 e 3, uma vez que o sistema para esta planta é de 9 regras. A tabela 4 não é utilizada para esta planta, pois o controlador é PI.

Tabela 2 – Regras de Ajuste do Parâmetro K_p

K_p	Erro				
	TxErro	BN	SN	EZ	SP
BN	G	G	G	G	G
SN	G	G	P	G	G
EZ	G	P	P	P	G
SP	G	G	P	G	G
BP	G	G	G	G	G

Tabela 3 - Regras de Ajuste do Parâmetro Ki

Ki	Erro				
	TxErro	BN	SN	EZ	SP
BN	P	P	G	P	P
SN	P	P	G	P	P
EZ	P	G	G	G	P
SP	P	P	G	P	P
BP	P	P	G	P	P

Tabela 4 - Regras de Ajuste do Parâmetro Kd

Kd	Erro				
	TxErro	BN	SN	EZ	SP
BN	P	P	P	P	P
SN	G	P	P	P	G
EZ	G	P	P	P	G
SP	G	P	P	P	G
BP	P	P	P	P	P

No caso da otimização da saída dos conjuntos CRISP no controlador PID nebuloso, o algoritmo CARLA é utilizado de forma a escolher seis valores que correspondem às posições dos conjuntos CRISP de saída (Figura10).

Para a otimização do controlador PID *fuzzy* foram realizadas trinta mil iterações *off-line*. Na

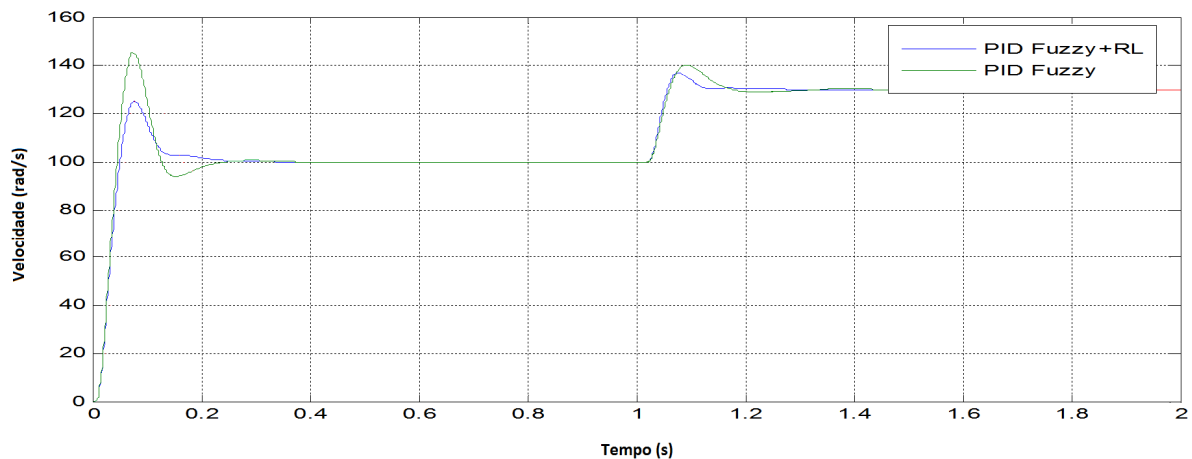


Figura 11 – Resposta do motor com o controlador PID *fuzzy* e com o PID *fuzzy* ajustado pelo CARLA.

6 Conclusão

Neste artigo, métodos para a sintonia de um controlador PID aplicado a um motor CC foram simulados. Estas simulações comparam duas diferentes estruturas PID *fuzzy* e um PID sintonizado por *Ziegler-Nichols*. As estruturas PID *fuzzy* obtiveram respostas melhores que o método clássico de ajuste de PID, *Ziegler-Nichols*. Isto ocorre por se realizar a atualização dos parâmetros do controlador em tempo-real utilizando lógica *fuzzy*.

Os resultados obtidos com a planta da figura 1 demonstram ganhos obtidos com a variação da estrutura PID *fuzzy* proposta neste trabalho na redução do *overshoot*, e na redução do tempo de estabilização da resposta do processo. Resultados práticos do controlador PID *fuzzy* ajustado pelo CARLA aplicado ao conversor da figura 3 serão levantadas como próxima etapa deste trabalho.

simulação computacional foi utilizado o software MATLAB®, mais especificamente o Simulink [12]. As seguintes simulações foram realizadas para avaliação comparativa no controle do motor cc do sistema da Figura 1:

- Um o controlador PID *fuzzy* com a estrutura proposta por [3], [4];
- Um controlador PID *fuzzy* com a estrutura proposta por [3], [4], tendo seus conjuntos de saída ajustados por aprendizagem por reforço.

5.4 Resultados

A Figura 11 exibe as respostas do motor CC com os controladores PID *fuzzy* e o controlador PID *fuzzy* ajustado pelo CARLA.

Analisando a figura 11, observa-se que há uma melhoria ao se adotar o ajuste dos conjuntos CRISP de saída do sistema de inferência visto há um menor *overshoot* e um menor tempo de estabilização em comparação ao PID *fuzzy* sem ajuste.

Referências Bibliográficas

- [1] Åström, K. J., Hägglund, T. (2001). The Future of PID Control, *Control Engineering Practice*, v. 9, p. 1163-1175.
- [2] Åström, K. J.; Hägglund, T. (1995). *PID Controllers: Theory, Design and Tuning*, Instrument Society of America.
- [3] Almeida, O. M., Coelho, A. A. R.; (2001). Controlador PID com Escalonamento Nebuloso dos Ganhos: Auto-Sintonia, Análise e Implementação, *V Simpósio Brasileiro de Automação Inteligente – V SBAI*.

- [4] Almeida, O. M., Coelho, A. A. R.; (2002). A Fuzzy Logic Method for Autotuning a PID Controller: *SISO* and *MIMO* Systems, *15th IFAC World Congress on Automatic Control*.
- [5] Jantzen, J. 2007. *Foundations of fuzzy control*. John Wiley & Sons Ltd.
- [6] Ogata, K. (1993). *Engenharia de Controle Moderno*. PHB, 2ª edição, 1993.
- [7] Coelho, A. A. R. e Coelho, L. S. (2004). *Identificação de Sistemas Dinâmicos Lineares*. Editora da UFSC, Florianópolis - SC.
- [8] Zadeh, L. A. (1965). Fuzzy sets. *Fuzzy Sets, Information and Control*, 8:338 353.
- [9] Yuan, M. e Zhou, H. (2001). Fuzzy pid control implementation in ff function block. *IEEE*, page 388 393.
- [10] Sandri, S. e Correa, C. (1999). Lógica nebulosa. *V Escola de Redes Neurais, Promoção: Conselho Nacional de Redes Neurais*, pág. c073 c090, Julho 1999.
- [11] Bezerra, O. A. L. R. L. e Lima, S. (2004). A mimo fuzzy logic autotuning pid controller: Method and application. *WSC9*, October 2004.
- [12] Hines, J. W.; Tsoukalas, L. H. e Uhrig, R. E. (1997). *MATLAB Supplement to Fuzzy and Neural Approaches in Engineering*, John Wiley & Sons.
- [13] Kabudian, J.; Meybodi, M.R.; Homayounpour, M.M., "Applying continuous action reinforcement learning automata (CARLA) to global training of hidden Markov models," *Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on*, vol.2, no., pp. 638-642 Vol.2, 5-7 April 2004.
- [14] Howell, M.N.; Gordon, T.J.; Best, M.C. (2000). "The application of continuous action reinforcement learning automata to adaptive PID tuning," *Learning Systems for Control (Ref. No. 2000/069)*, *IEE Seminar*, vol., no., pp.2/1-2/4.
- [15] Sutton, R.S. e Barto, A. (1998). *Introduction to Reinforcement Learning*. MIT Press / Bradford Books, Cambridge, MA.
- [16] Kaelbling, L. P., Littman, M. L. e Moore, A. W. (1996). Reinforcement learning: A survey, *Journal of Artificial Intelligence Research*, V. 4, pp: 237-285.
- [17] Shaw, I. S. e Simões, M. G. (1999). *Controle e modelagem fuzzy*. Ed. Edgard Blücher Ltda.
- [18] BARRETO, L. H. S. C. ; COELHO, E. A. A. ; FARIAS, V. J. ; de OLIVEIRA, J. C. ; de FREITAS, L. C ; Vieira, J. B. Jr . A Quasi-Resonant Quadratic Boost Converter Using a Single Resonant Network. *IEEE Transactions on Industrial Electronics*, v. 52, n.2, p. 552-557, 2005.