

Desenvolvimento de um controlador *Fuzzy-PID* aplicado à uma ventoinha

Bryan de Oliveira Ribeiro
Depart. De Engenharia Eletrica
Universidade Tecnologica Federal do
Paraná
Ponta Grossa, Brasil
bryanribeiro92@gmail.com

Guilherme Ricardo Da Silva
Depart. De Engenharia Eletrica
Universidade Tecnologica Federal do
Paraná
Ponta Grossa, Brasil
guilhermes.2017@alunos.utfpr.edu.br

Hugo Valadares Siqueira
Depart. De Engenharia Eletrica
Universidade Tecnologica Federal do
Paraná
Ponta Grossa, Brasil
hugosiqueira@utfpr.edu.br

Fernanda Cristina Corrêa
Depart. De Engenharia Eletrica
Universidade Tecnologica Federal do
Paraná
Ponta Grossa, Brasil
fernandacorrea@utfpr.edu.br

Resumo — Plantas que operam em malha aberta, ou seja, sem sensoriamento e controle, tendem a ser muito sensíveis à perturbações, além de normalmente não apresentarem desempenho transitório adequado em diversas implementações. Assim sendo, este trabalho apresenta os resultados de um estudo de caso sobre a aplicação de estratégias de controle PID e controle híbrido *FUZZY-PID* aplicados à uma ventoinha de computador. O presente trabalho fornece uma análise sobre o comportamento, parametrização, e principais diferenças apresentadas no comportamento dessa planta. A partir dos resultados, nota-se que com um controle adaptativo bem sintonizado é possível obter melhores parâmetros de desempenho, entretanto o desenvolvimento de um projeto como este nem sempre é a melhor solução custo-benefício, em função de sua complexidade de desenvolvimento.

Palavras-chaves — Controle, Ventoinha, Lógica Fuzzy, *Fuzzy-PID*.

I. INTRODUÇÃO

A área de controle e automação se tornou uma poderosa ferramenta no desenvolvimento dos processos industriais, sistemas comerciais e residenciais. A possibilidade de coordenar o comportamento de plantas em tempo real e de forma adaptativa tem possibilitado não somente a manutenção de processos já consolidados no mercado, como também o surgimento de novos, antes impossíveis ou árduos para operação.

Um grande avanço produzido pelas técnicas de controle foi na área de controle e automação industrial. Trabalhar com máquinas como motores de corrente contínua, ou seja, motores CC, de indução, fornos, ventoinhas, é inevitável dentro do segmento industrial. O primeiro desafio é a partida e acionamento desses equipamentos, que exigem picos de energia elétrica, provocando grandes oscilações na rede. A segunda questão é relacionada ao comportamento desses equipamentos na presença de perturbações.

Diante dos desafios apresentados no que se refere à inserção de motores elétricos na indústria, este trabalho propõe a análise de dois controladores, um controlador PID discreto e um controlador *Fuzzy-PID* aplicados à uma ventoinha. De forma geral, os controladores *Fuzzy-PID* têm sido utilizados em diversas aplicações em várias áreas da engenharia, como por exemplo em controle de motores do

tipo BLDC (motor sem escovas de corrente contínua do inglês *brushless direct current*) para o aumento de vida útil [1], em processos químicos de um sistema de biomassa para uma caldeira a vapor [2] e em outras aplicações complexas com funções de transferência de ordem superior não lineares. O controlador *Fuzzy-PID* integra a vantagem de ambas as estruturas de controle e, como consequência uma melhoria no desempenho tanto em termos do transitório quanto ao estacionário [3].

Neste presente estudo, a ventoinha DC é acoplada em uma placa PCB do *Kit PICGenios* para aprendizagem e desenvolvimento, utilizada para a refrigeração dos equipamentos de potência da mesma. Esta ventoinha é alimentada por um sinal PWM (modulação por largura de pulso do inglês *Pulse Width Modulation*) de valor eficaz máximo de 5V que é produzido por um microcontrolador. Desta forma, nas próximas seções serão abordados a modelagem e validação da planta, estratégias de controle implementadas e por fim os resultados obtidos, analisando-se principalmente o comportamento do sistema quando este é submetido à perturbações.

II. OBTENÇÃO DO MODELO MATEMÁTICO DA PLANTA PROPOSTA

A. Definições

Inicialmente, é necessário compreender o comportamento do sistema a ser controlado. Para isso, obtém-se a planta que caracteriza o comportamento do sistema em uma equação matemática.

O *kit* utilizado é semelhante ao mostrado na Fig. 1, sendo acoplado com um microprocessador PIC18F4550. Assim, para a obtenção da planta e para a implementação do controlador, todas as instruções deverão ser executadas em linhas de código embarcada no microprocessador. O programa utilizado para o desenvolvimento do código é o MPLabX e a linguagem base C.

B. Obtenção Da Planta

A estratégia para obter o comportamento do sistema é baseado em interrupções. O PIC18F4550 possui contadores internos que podem ser configurados para dispararem um sinal a qualquer instante desejado. Esse sinal é utilizado para

disparar uma rotina de interrupção, que nada mais é que uma rotina de códigos que interrompe a execução do código principal para ser executada e ao seu fim retorna para o código principal. Esse mecanismo é útil para garantir que os dados obtidos respeitem um intervalo de tempo padrão.



FIG. 1. KIT PICGENIOS.

Para a obtenção de dados configura-se o valor da largura de pulso do PWM no seu limite, ou seja, em 1, e por meio de um sensor a laser instalado entre as pás da ventoinha, cada vez que uma pá sobrepõe o sensor e interrompe o fluxo do laser um contador é incrementado. Assim, toda vez que essa rotina de interrupção for executada, o conteúdo desse contador é registrado em um vetor global e em seguida é zerado. Isso fornece uma curva de $\frac{pás}{T_s}$ onde T_s é o intervalo de tempo da interrupção, que em neste caso está configurado para 100 ms.

Na Fig. 2 é possível observar o comportamento da planta até atingir sua rotação máxima, que neste caso é 40 pás a cada 100 ms, este valor pode ser convertido para uma unidade de mais fácil compreensão como rpm, entretanto isto dificultaria o projeto do controle, assim adotou-se esta unidade como referência.

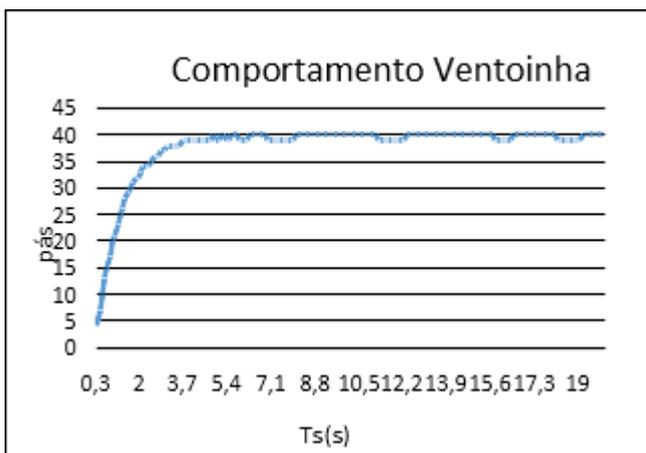


FIG. 2. COMPORTAMENTO DA PLANTA

A partir do comportamento da planta é possível obter a equação matemática que a representa. Por meio de uma breve análise visual, nota-se que esse comportamento é característico de uma planta de primeira ordem que tem

como função de transferência a representação dada na Equação (1) com um K_o (ganho estático) unitário.

$$P(s) = \frac{1}{\tau s + 1} \quad (1)$$

Sendo que τ é a constante de tempo, e quando $t=\tau$ a curva atinge aproximadamente 63% do valor máximo de pás que em nosso caso é 25.5 pás. Dessa forma tem-se que τ é de 1,22 s. Para se obter uma resposta semelhante à real com entradas de tensão eficaz de 0 a 5 V multiplicou-se a planta pelo número máximo de pás que é igual a 40 e em seguida foi dividido pela entrada de tensão máximo que corresponde a 5 V, no qual resultou-se na função de transferência apresentada na Equação (2) com um K_o (ganho estático) igual a 8.

$$P(s) = \frac{8}{1.22s + 1} \quad (2)$$

C. Validação e Comportamento da Planta

Para validar o modelo matemático obtido da planta, é realizada uma implementação no software Matlab e aplicada uma entrada degrau de amplitude 5 ao sistema e assim espera-se obter o mesmo comportamento apresentado na Fig. 2.

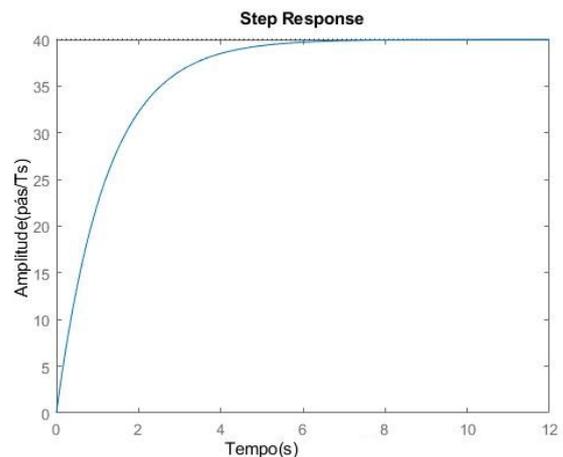


FIG. 3. RESPOSTA AO DEGRAU DO MODELO MATEMÁTICO DA PLANTA.

Comparando-se os comportamentos das Fig 2 e Fig 3, é possível verificar que o modelo matemático obtido se aproxima da curva obtida por experimento, fato que permite a utilização desse modelo para o projeto do controlador.

D. Planta Discreta

A planta real e a atuação do controle na prática não ocorrem de forma contínua e sim de forma discreta, ou seja, em intervalos discretos de tempo. O primeiro passo para se obter o controlador é a representação discreta da planta, a qual representa o comportamento do sistema em uma faixa de amostragem. Facilmente obtêm-se a planta discreta a partir da planta contínua utilizando-se o método de discretização ZOH (Segurador de Ordem Zero do inglês Zero Order Holder). Dessa forma, define-se o tempo de amostragem, o qual é uma variável indispensável, uma vez que o controlador é projetado para atuar nesse período de

amostragem. Com o valor escolhido de 10 ms , têm-se a seguinte função de transferência discreta (Equação 3).

$$Pz(z) = \frac{0.06531}{z - 0.9918} \quad (3)$$

III. ESTRATÉGIAS DE CONTROLE

Para atender o objetivo de desenvolver um controlador *Fuzzy*-PID, primeiramente, um projeto de controle de PID discreto será implementado e na sequência validado, sendo que a lógica *Fuzzy* irá atuar com valores iniciais definidos por esse projeto.

Desta forma, foi desenvolvido um controlador PID a partir da alocação de polos e zeros com o auxílio do *toolbox* *SISOTool* do *Matlab*. Em seguida, é apresentado o desenvolvimento da lógica *Fuzzy* e sua implementação.

A. Controle PID Discreto

Este projeto de PID discreto está pautado no método de alocação de polos e zeros. Será utilizada como ferramenta para o desenvolvimento a *toolbox* do *Matlab*, *SISOTool*, onde é possível importar a planta discreta e visualizar seu comportamento em malha fechada com um controle que pode ser ajustado em tempo real.

Para a configuração do controlador utiliza-se a topologia descrita pela Fig. 4, com dois polos fixos em 0 e 1, e um par de zeros complexos ajustáveis. Isso é proveniente da equação do controlador PID discreto conforme Equação (4).

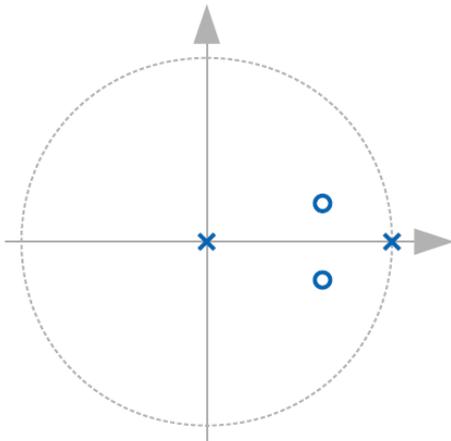


FIG. 4. TOPOLOGIA DO PID.

$$C(z) = \frac{(k_p + k_d + k_i)z^2 - (k_p + 2k_d)z + k_d}{z(z - 1)} \quad (4)$$

Uma boa estratégia ao projetar o controlador PID por esse método é alocar os zeros ajustáveis perto dos polos da planta, neste caso perto do polo real 0.992 . Assim, decidiu-se alocar os zeros em $0,8+j0,2$. A partir da ferramenta *SISOTool* pode-se refinar o controle como mostrado nas Fig. 5 e 6, e por fim obter a função de transferência do controlador discreto (Equação 5).

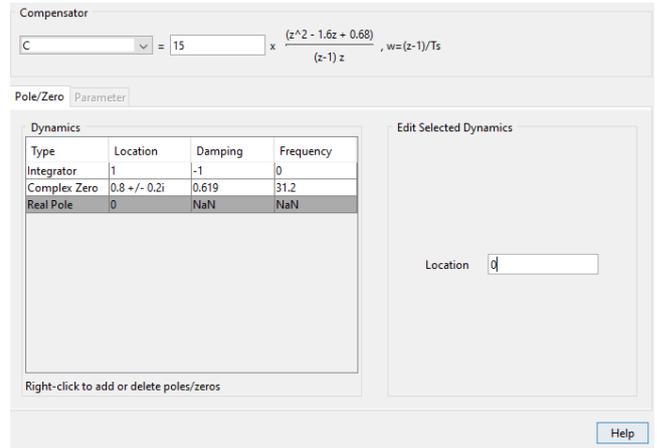


FIG. 5. LOCALIZAÇÃO DOS POLOS E ZEROS DO CONTROLADOR.

A partir da ferramenta *SISOTool* pode-se refinar o controle como mostrado na Fig. 5, e por fim obter a equação do controlador discreto.

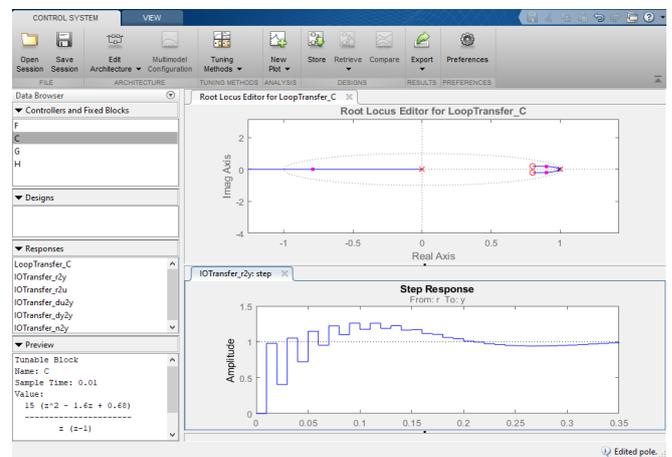


FIG. 6. SISOTOOL.

$$C(z) = \frac{15z^2 - 24z + 10.2}{z(z - 1)} \quad (5)$$

Por último, é avaliado o comportamento do sistema de malha fechada por meio da implementação no *Simulink/Matlab*, conforme pode ser observado na Fig. 7.

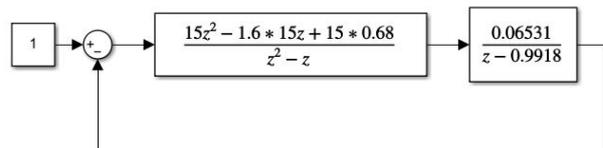


FIG. 7. SISTEMA DE CONTROLE PID DISCRETO

A partir dessa implementação pode-se analisar o comportamento do sistema controlado e verificar se o controlador projetado está operando conforme desejado. A Fig. 8 mostra o sistema com o controlador PID discreto.

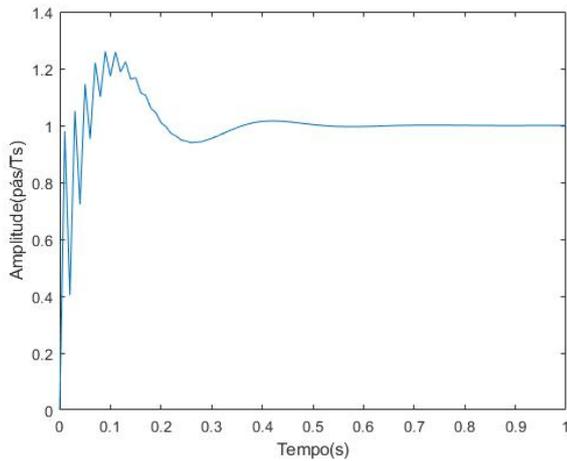


FIG. 8. RESPOSTA DO SISTEMA COM CONTROLE PID DISCRETO.

B. Lógica Fuzzy

Finalizado o desenvolvimento e a avaliação do controlador PID discreto, a próxima etapa consiste no desenvolvimento e implementação da lógica *Fuzzy*.

Uma das principais vantagens de se utilizar o controlador Fuzzy é a de que o projetista pode realizar o controle de um sistema com base no comportamento que o mesmo possui, sem a necessidade de obtenção do seu modelo matemático [4]. Porém, para que isso ocorra é necessário que o projetista tenha pleno conhecimento do funcionamento do sistema de modo que consiga ajustar os parâmetros de forma correta fazendo com que o esforço de controle trate de maneira específica cada entrada com o propósito de atingir os objetivos esperados [5].

A conceituação é que a lógica *Fuzzy* receba entradas de erro e sua derivada e retorne valores de ajuste para as constantes K_p , K_d e K_i . Devido a isto, torna-se essencial a elaboração do controle PID discreto, uma vez que as constantes obtidas serão utilizadas como condições iniciais no sistema *Fuzzy*-PID. Assim, a aplicação da lógica *Fuzzy* é acoplada a um controlador PID para ajustar os seus parâmetros automaticamente em um processo on-line, ou seja, se houver modificações na dinâmica da planta, como por exemplo variações de carga, os ganhos do controlador PID são ajustados por meio da lógica *Fuzzy* para se adaptarem a essa mudança em tempo real [3]. Esta união pode tornar o sistema estável mais rapidamente que um controlador PID clássico, resultando em uma diminuição do tempo de acomodação para se atingir o estado estacionário [6].

A estrutura do sistema conforme apresentado na Fig. 9, elabora toda a lógica *Fuzzy* e além disso desenvolve um bloco de controlador PID a partir da equação a diferenças que aceita como parâmetro os valores das constantes K_p , K_d e K_i iniciais e seus valores de ajuste provenientes da saída da lógica *Fuzzy*.

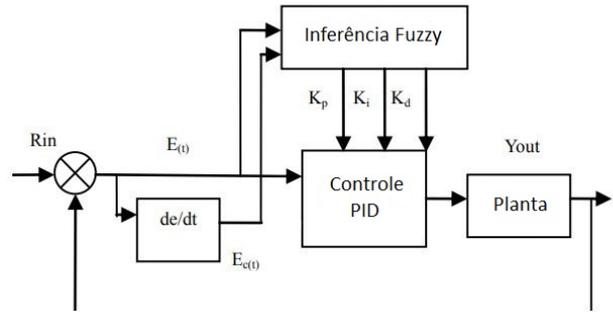


FIG. 9. ESTRUTURA DO SISTEMA DE CONTROLE FUZZY -PID.

Como o sistema de controle é um sistema discreto, deve-se elaborar o bloco de PID discreto, ou seja, a partir da equação a diferenças. Entretanto, as entradas do bloco do Controlador Discreto são os ajustes das constantes de controle, as quais são somadas às constantes iniciais. Sendo assim, é essencial conhecer os valores dessas constantes, o que é feito a partir das Equações (4) e (5) e obtêm-se como resultado $K_p=3,6$, $K_d=10,2$ e $K_i=1,2$. A partir da equação a diferença apresentada na Equação 6, pode-se obter o esquema da saída do controlador conforme mostra a Figura 10.

$$u_k = u_{k-1} + 15e_k - 24e_{k-1} + 10.2e_{k-2} \quad (6)$$

Têm-se então na Fig. 10 o bloco finalizado do PID discreto que aceita ajustes nos parâmetros K_p , K_d e K_i , sendo o próximo passo o desenvolvimento e sintonia da lógica *Fuzzy*.

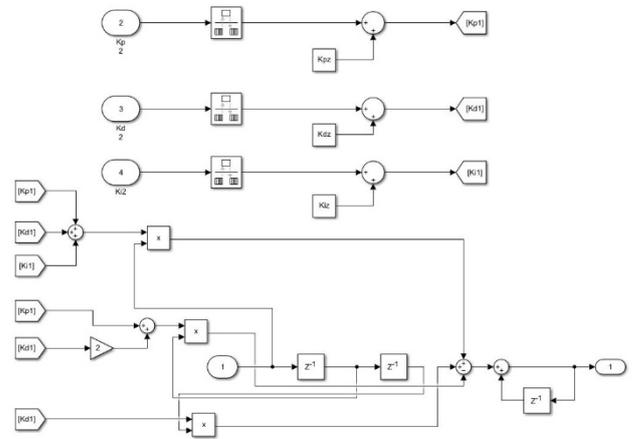


FIG. 10. ESTRUTURA DO BLOCO PID DISCRETO COM PARÂMETROS VARIÁVEIS.

Primeiramente, são estabelecidos os universos de discurso e funções de pertinência das entradas e saídas do sistema *Fuzzy*. Os universos de discursos são estabelecidos da seguinte forma: Erro [-3, 3], Dx_Erro [-10, 10], K_p [-0.5, 0.5], K_d [-0.5, 0.5], K_i [-0.1, 0.1]. Em relação às funções de pertinência dessas variáveis são utilizadas 7 funções, sendo 5 triangulares e 2 gaussianas. Essas funções são definidas da seguinte forma:

- NB – *Negative Big* (Negativo Grande)
- NM – *Negative Medium* (Negativo médio)
- NS – *Negative Small* (Negativo Pequeno)
- Z – Zero
- PS – *Positive Small* (Positivo Pequeno)

- PM – *Positive Medium* (Positivo Médio)
- PB – *Positive Big* (Positivo Grande)

As funções de pertinências utilizadas para a variável de entrada Erro são de acordo com o Quadro 1, e podem ser vistas na Fig 11.

QUADRO. 1. FUNÇÕES DE PERTINÊNCIA DA VARIÁVEL ERRO

Função	Tipo	Intervalo
NB	Gaussiana	[0.768 -3.021]
NM	Trapezoidal	[-3 -2.1 -0.9]
NS	Trapezoidal	[-2.984 -1.184 0.6159]
Z	Trapezoidal	[-1.8 0 1.8]
PS	Trapezoidal	[-0.6 1.2 3]
PM	Trapezoidal	[0.9 2.1 3]
PB	Gaussiana	[0.732 2.983]

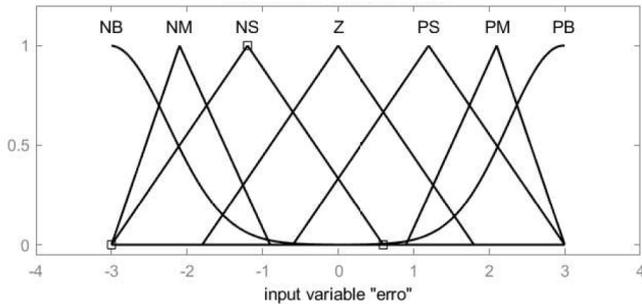


FIG. 11. VARIÁVEL ERRO.

Enquanto a variável de entrada *Variação do Erro* (Dx_Erro) possui as seguintes funções de pertinência conforme Quadro 2 e Fig.12.

QUADRO. 2. FUNÇÕES DE PERTINÊNCIA DA VARIÁVEL VARIACÃO DO ERRO

Função	Tipo	Intervalo
NB	Gaussiana	[2.56 -10.17]
NM	Trapezoidal	[-10 -7 -3]
NS	Trapezoidal	[-10 -4 2]
Z	Trapezoidal	[-5.9 0 6.06]
PS	Trapezoidal	[-2 4 10]
PM	Trapezoidal	[3 7 10]
PB	Gaussiana	[2.44 10.21]

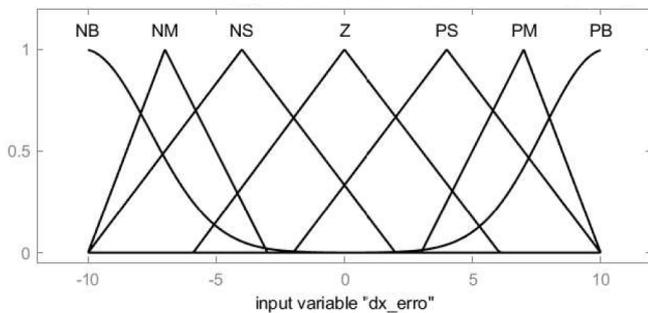


FIG. 12. VARIÁVEL DX_ERRO.

As funções de pertinência da variável de saída K_p são mostradas no Quadro 3 e Fig. 13.

QUADRO. 3. FUNÇÕES DE PERTINÊNCIA DA VARIÁVEL K_p

Função	Tipo	Intervalo
NB	Gaussiana	[0.1292 -0.5]
NM	Trapezoidal	[-0.5 -0.25 0]
NS	Trapezoidal	[-0.3742 -0.125 0.125]
Z	Trapezoidal	[-0.25 0 0.25]
PS	Trapezoidal	[-0.125 0.125 0.375]
PM	Trapezoidal	[0 0.25 0.5]
PB	Gaussiana	[0.1404 0.5]

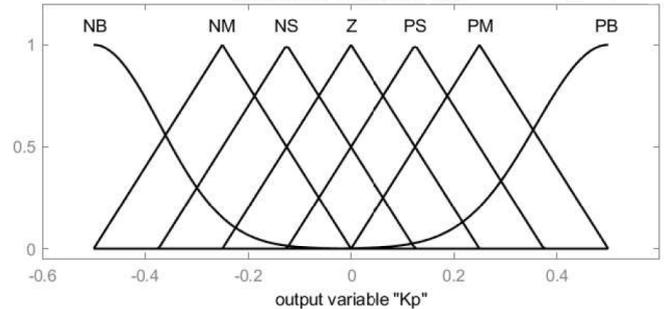


FIG. 13. VARIÁVEL K_p .

Já por meio do Quadro 4 e Fig. 14 têm-se as funções de pertinência da variável de saída K_d .

QUADRO. 4. FUNÇÕES DE PERTINÊNCIA DA VARIÁVEL K_d

Função	Tipo	Intervalo
NB	Gaussiana	[0.1494 -0.5]
NM	Trapezoidal	[-0.5 -0.25 0]
NS	Trapezoidal	[-0.3742 -0.125 0.125]
Z	Trapezoidal	[-0.25 0 0.25]
PS	Trapezoidal	[-0.125 0.125 0.375]
PM	Trapezoidal	[0 0.25 0.5]
PB	Gaussiana	[0.1382 0.5]

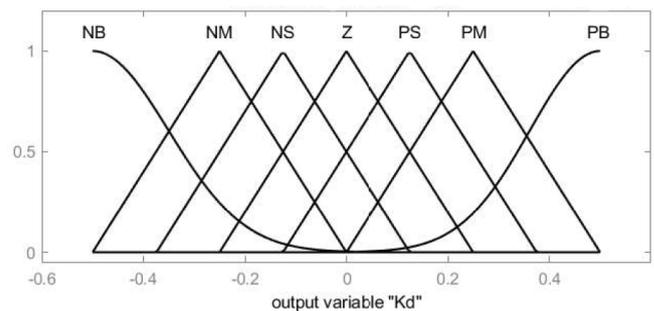


FIG. 14. VARIÁVEL K_d .

Finalmente, o Quadro 5 e a Fig. 15 mostram a divisão em funções de pertinência da variável de saída K_i .

QUADRO. 5. FUNÇÕES DE PERTINÊNCIA DA VARIÁVEL Ki

Função	Tipo	Intervalo
NB	Gaussiana	[0.03032 -0.1]
NM	Trapezoidal	[-0.1 -0.06 -0.02]
NS	Trapezoidal	[-0.0695 -0.02952 0.01048]
Z	Trapezoidal	[-0.04 0 0.04]
PS	Trapezoidal	[-0.0073 0.0327 0.0727]
PM	Trapezoidal	[0.02 0.06 0.1]
PB	Gaussiana	[0.03122 0.1]

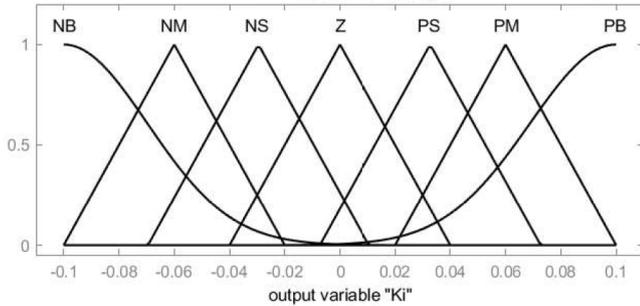


FIG. 15. VARIÁVEL Ki.

Para o mapeamento das regras da lógica *Fuzzy*, três principais regras foram utilizadas, sendo elas:

1. Quando o erro é grande, deve-se ter K_p e K_d grande, isso para que o sistema tenha um bom desempenho em relação ao sobressinal. Neste caso K_i deverá ser zero.
2. Quando o erro e sua derivada forem médios, tem-se um K_p pequeno, K_i médio e K_d grande, para evitar alto sobressinal e melhorar o tempo de acomodação e o erro de estabilização.
3. Quando o erro for pequeno, K_p e K_i deverão ser grandes para garantir boa estabilidade do sistema. Já K_d é dependente da derivada do erro; se esta for grande K_d , deve ser pequeno, se pequeno K_d , deve ser grande.

Com base nessas proposições as seguintes regras foram criadas para a inferência de K_p , K_d e K_i e são mostradas nas Tabelas 1, 2 e 3.

TABELA 1. REGRAS PARA K_p .

E_c	E						
	NB	NM	NS	ZO	PS	PM	PB
NB	PB	PB	PM	PM	PS	ZO	ZO
NM	PB	PB	PM	PS	PS	ZO	NS
NS	PM	PM	PM	PS	ZO	NS	NS
ZO	PM	PM	PS	ZO	NS	NM	NM
PS	PS	PS	ZO	NS	NS	NM	NM
PM	PS	ZO	NS	NM	NM	NM	NB
PB	ZO	ZO	NM	NM	NM	NB	NB

TABELA 2. REGRAS PARA K_d .

E_c	E						
	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NM	NM	NS	ZO	ZO
NM	NB	NB	NM	NS	NS	ZO	ZO
NS	NB	NM	NS	NS	ZO	PS	PS
ZO	NM	NM	NS	ZO	PS	PM	PM
PS	NM	NS	ZO	PS	PS	PM	PB
PM	ZO	ZO	PS	PS	PM	PB	PB
PB	ZO	ZO	PS	PM	PM	PB	PB

TABELA 3. REGRAS PARA K_i .

E_c	E						
	NB	NM	NS	ZO	PS	PM	PB
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NB	NM	NM	NS	ZO
NS	ZO	NS	NM	NM	NS	NS	ZO
ZO	ZO	NS	NS	NS	NS	NS	ZO
PS	ZO						
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PS	PS	PS	PB

Por fim, para um ajuste mais preciso da lógica desenvolvida foram utilizados ganhos de *fuzzificação* e *defuzzificação* (Fig. 16), ou seja, ganhos que modificam levemente as variáveis de entrada e saída da lógica *Fuzzy*. Os ganhos que atenderam melhor o propósito foram $K_{F_erro}=1,7$, $K_{F_d\text{erro}}=1$, $K_{D_Kp}=35$, $K_{D_Kd}=10$ e $K_{D_Ki}=2$.

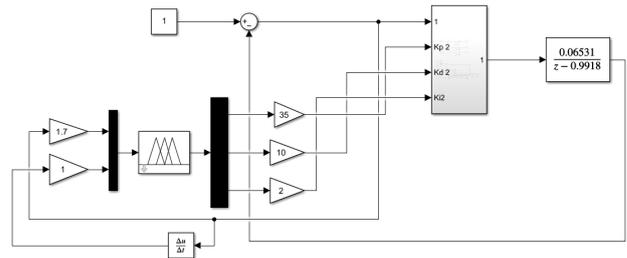


FIG. 16. SISTEMA COM CONTROLADOR PID-FUZZY.

IV. ANÁLISE DOS RESULTADOS

Os resultados provenientes da simulação podem ser observados nas Figuras (17), (18), (19), (20), (21) e (22). A Figura (17) mostra um comparativo entre os controladores PID (azul) e *Fuzzy*-PID (amarelo) quando submetidos por uma entrada degrau unitário. Pode-se verificar que o controlador *Fuzzy*-PID apresenta um melhor desempenho tanto em termos de percentual de sobressinal como tempo de acomodação. Além disso, o sistema com controlador *Fuzzy*-PID não sofre mudança na sua dinâmica quando submetido a um distúrbio.

As Figuras (18) e (19) possibilitam a visualização dos sinais de entrada Erro e Variação do Erro do controlador *Fuzzy*-PID, no qual nota-se que ambos sinais tendem a zero à medida que o tempo tende a infinito, fato este justificado pela atuação do sistema de controle.

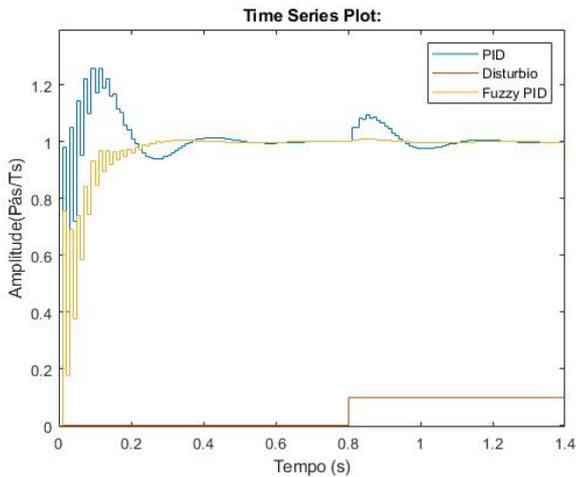


FIG. 17. COMPORTAMENTO DO SISTEMA DE MALHA FECHADA PARA UMA ENTRADA DEGRAU COM PRESENÇA DE DISTÚRBIO.

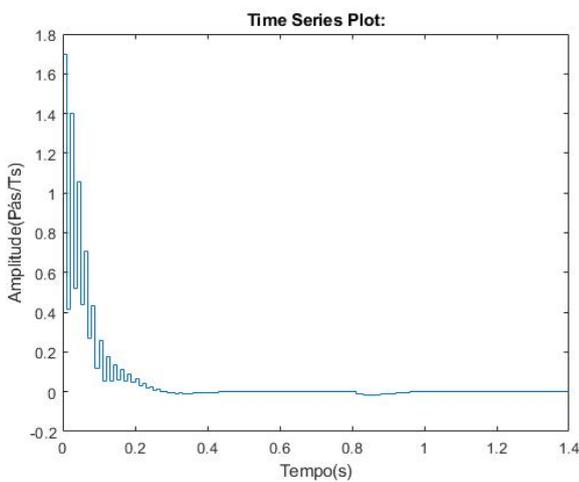


FIG. 18. COMPORTAMENTO DA ENTRADA ERRO DA LÓGICA FUZZY.

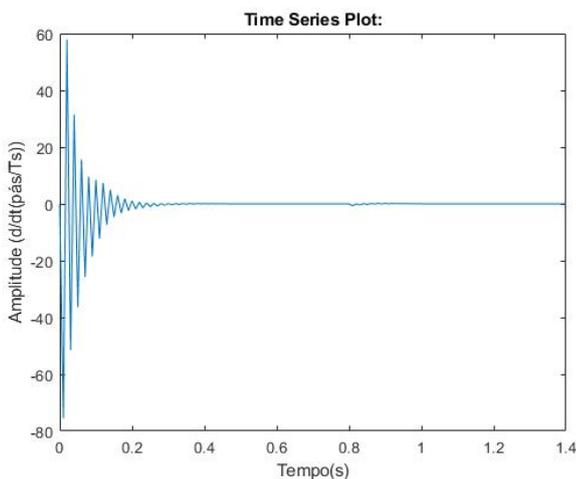


FIG. 19. COMPORTAMENTO DA ENTRADA DERIVADA DO ERRO DA LÓGICA FUZZY.

As Figuras (20), (21) e (22) apresentam o comportamento dos ganhos K_p , K_d e K_i , respectivamente, no qual estes ganhos são calculados a partir da lógica fuzzy. Pode-se perceber que o K_p possui uma grande magnitude de oscilação, sendo o ganho responsável pelo tempo de resposta do sistema. O ganho K_d possui uma menor magnitude de oscilação, tornando-se nulo quando o sistema

entra em regime estacionário, fato este justificado pela ausência de oscilação na resposta do sistema controlado. Enquanto o K_i é o ganho que apresenta uma menor magnitude de oscilação, sendo o responsável pela correção do erro estacionário.

Na Tabela 4 é possível identificar o comportamento dinâmico de cada controle quantitativamente. É notório como o controlador *Fuzzy* – PID, quando bem sintonizado pode apresentar resultado superior de desempenho; neste caso, reduzindo em relação ao controle PID quase 50% do tempo de acomodação e levando o sobressinal do sistema a zero. Para a presença de um distúrbio de 10% do valor de estabilização nota-se uma melhoria de quase 90% do sobressinal no sistema com controle *Fuzzy* – PID em relação ao sistema com controle PID, assim como redução do tempo de estabilização para 1% em aproximadamente 78,5% em relação ao PID convencional.

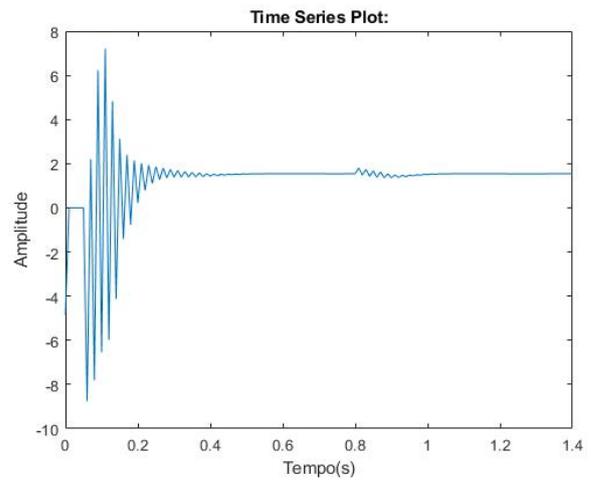


FIG. 20. COMPORTAMENTO DA ENTRADA K_p DO PID.

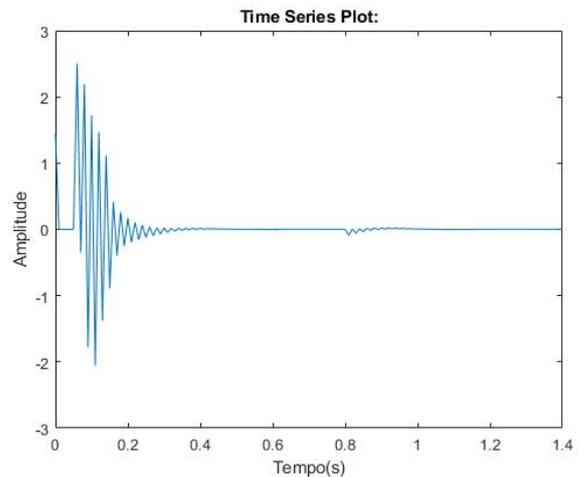


FIG. 21. COMPORTAMENTO DA ENTRADA K_d DO PID.

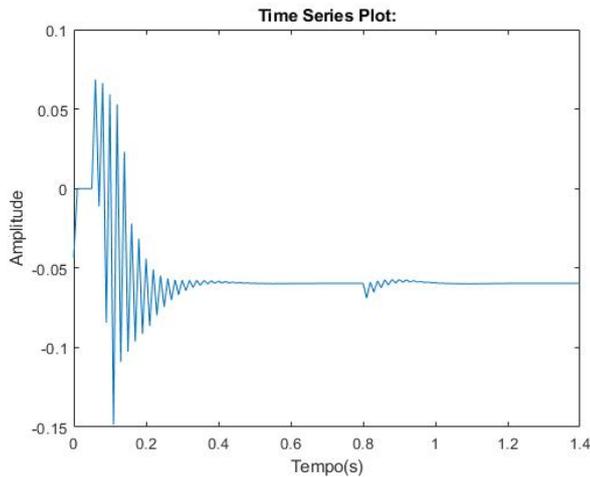


FIG. 22. COMPORTAMENTO DA ENTRADA KI DO PID

V. CONCLUSÕES

Neste trabalho, realizou-se a comparação entre a resposta de um sistema de uma ventoinha, onde foram aplicadas estratégias de controle PID pura e PID-Fuzzy. Para a elaboração dos resultados foi usado o *software* de simulações Simulink o qual faz parte do *software* de matemática *Matlab*.

TABELA 4. COMPARATIVO DOS COMPORTAMENTOS DO SISTEMA

	Sem controle	PID	Fuzzy - PID
Tempo de Subida (s)	2.68	0.008166 2	0.08694
Tempo de acomodação (s)	5.62	0.46693	0.24515
Sobressinal (%)	0	25.973	0.5998
Pico (Pás/Ts)	39.999	1.2597	1.006
Tempo de pico (s)	12.866	0.09	0.33

Entretanto, esse melhor comportamento do sistema de controle Fuzzy PID é diretamente proporcional a sua complexidade. Como acompanhado neste estudo, um controle PID é facilmente desenvolvido e implementado, contando com diversas opções de projeto. Já quando implementada a lógica Fuzzy, o trabalho se torna mais árduo, isso levando em conta que não é possível demonstrar neste trabalho o tempo e serviço empregado para o desenvolvimento de todos os parâmetros da lógica Fuzzy e sua sintonização, que na maior parte das vezes, quando não utilizado nenhuma forma de otimização, é por tentativa e erro.

Um exemplo da complexidade da sintonia desse tipo de controle é possível perceber ao analisarmos a entrada de erro da lógica Fuzzy, Fig 18. Nota-se que a magnitude do sinal foi de 0 a 1,4 comparando com a magnitude estabelecida da variável erro [-3, 3] pode-se verificar que ela não é totalmente aproveitada. Isso implica em algumas regras nunca serem ativadas, neste caso não houve muita interferência. No entanto, se o controlador Fuzzy for melhor sintonizado é possível obter uma melhoria de unidades de ms no tempo de acomodação assim como uma melhora na

ordem de 10^{-1} no sobressinal, ou seja, algo insignificante. Entretanto, nem todos os sistemas se comportarão dessa maneira, algo desse tipo pode levar o sistema a instabilidade total se não corrigido.

Para trabalhos futuros pode-se estudar a implementação de um sistema de otimização para reduzir o tempo gasto com a sintonia de todos os parâmetros da lógica Fuzzy, para evitar a tentativa e erro, e então obter uma melhor sintonização e otimização na estratégia de controle. O algoritmo genérico (GA) e o algoritmo de inteligência de enxame (PSO) são ferramentas de otimização dos parâmetros do controlador Fuzzy que podem ser aplicadas para que reduzam o tempo e trabalho no desenvolvimento da lógica.

REFERÊNCIAS

- [1] M. Lopez, P. Ponce, L. Soriano, A. Molina, and J. Rivas, "A novel fuzzy pso controller for increasing the lifetime in power electronics stage for brushless dc drives," IEEE Access, vol. 7, pp. 47 841–47, abr 2019.
- [2] J. Jin, H. Huang, J. Sun, and Y. Pang, "Study on fuzzy self-adaptive pid control system of biomass boiler drum water," Journal of Sustainable Bioenergy Systems, vol. 03, pp. 93–98, 01 2013.
- [3] R. Goswami and D. Joshi, "Performance review of fuzzy logic based controllers employed in brushless dc motor," Procedia computer science, vol. 132, pp. 623–631, 2018.
- [4] G. Feng, G. Lu, D. Sun, and S. Zhou, "A model reference adaptive control algorithm for fuzzy dynamic systems," in Proceedings of the 4th World Congress on Intelligent Control and Automation (Cat. No. 2EX527), vol. 4. IEEE, 2002, pp. 3242–3246.
- [5] Simões, Marcelo Godoy, and Ian S. Shaw. Controle e modelagem fuzzy. Editora Blucher, 2007.
- [6] V. Geetha and S. Thangavel, "Performance analysis of direct torque controlled bldc motor using fuzzy logic," International Journal of Power Electronics and Drive Systems, vol. 7, no. 1, p. 144, 2016