

Implementação de Controladores Híbridos PID-Fuzzy: Análise de Desempenho em uma Planta de Nível

Gabriel C. S. Almeida* Rafael S. Salles*
Adelson S. Carvalho**

* *Instituto de Sistemas Elétricos e Energia, Universidade Federal de Itajubá, MG, (e-mail: gcaldasalmeida@gmail.com ; sallesrds@gmail.com).*

** *Faculdade de Engenharia de Controle & Automação, Instituto Federal Fluminense, RJ (e-mail: adelsonsc@gmail.com).*

Abstract: This paper presents the implementation and testing of PID controllers with gain scheduling using the fuzzy logic. The tests described were made using the Experimental Level System of Control Laboratory at Instituto Federal Fluminense. Two different approaches were used for PID gain scheduling scheme: the first one was proposed by Campos e Saito (2004) and make use of the benchmark value of the system as input; the second one was proposed by Zhao, Tomizuka e Isaka (1993) and make use of the error signal and its first difference fuzzy system entry. In order to have a full acquaintance of the system were applied techniques of identification and a tuning zone's mapping. From the results obtained, a performance evaluation and a controller's comparison were applied aiming the best one of all tested for the application.

Resumo: Este trabalho apresenta a implementação e os testes de controladores PID com escalonamento de ganhos utilizando lógica fuzzy. Os testes descritos foram realizados na Planta Experimental de Nível do laboratório de Controle do Instituto Federal Fluminense Campus Campos Centro. Para o escalonamento de ganhos do PID foram utilizadas duas abordagens diferentes: a primeira abordagem foi proposta por Campos e Saito (2004) e utiliza a referência do sistema como entrada; a segunda proposta por Zhao, Tomizuka e Isaka (1993) utiliza o erro e a variação do erro como entrada do sistema fuzzy. Para melhor elaboração dos controladores foi realizada uma caracterização do sistema a partir de técnicas de identificação e um mapeamento das regiões de operação do sistema. A partir dos resultados obtidos, uma análise de desempenho dos controladores foi elaborada com o intuito de verificar suas respectivas performances e compará-los a fim de identificar o melhor controlador dentre os testados para a aplicação.

Keywords: Fuzzy Logic; Gain Scheduling; Level Control; MATLAB; PID Controller.

Palavras-chaves: Controlador PID; Controle de Nível; Escalonamento de Ganhos; Lógica Fuzzy; MATLAB.

1. INTRODUÇÃO

A necessidade de controlar processos sempre esteve presente na história da humanidade. Com o passar do tempo, técnicas de controle foram sendo desenvolvidas porém o raciocínio humano ainda continua sendo peça fundamental para a tomada de decisões e resolução de problemas.

Processos repetitivos passaram a ser automatizados com o avanço das tecnologias, exigindo menor interferência humana em seu controle. No entanto, outras tarefas ficaram extremamente complexas para serem controladas desta maneira. Surgiu a necessidade de novas formas de controle e a utilização da Inteligência Artificial (IA) mostrou-se bastante eficaz neste sentido. Uma das técnicas de IA é a aplicação da lógica *fuzzy*.

Os seres humanos são capazes de lidar com processos bastante complexos, baseados em informações imprecisas ou aproximadas (Tanscheit, 2003). Estas informações vagas

são chamadas de termos linguísticos e são com eles que a lógica *fuzzy* trabalha. Esta lógica cria uma análise de valores não somente como verdadeiro ou falso, mas sim uma graduação de pertencimento de um elemento a mais de uma categoria e, com isso, faz com que cada vez mais as estratégias de controle se alinhem aos pensamentos e ações dos seres humanos.

Os controladores baseados em IA podem, em alguns casos, possuir um desempenho melhor quando comparados aos controladores convencionais. Por isso, um estudo mais aprofundado envolvendo controladores inteligentes é necessário para mostrar que seu uso é viável mesmo em sistemas que não se comportam de maneira linear em toda sua faixa de trabalho.

Zhen-Yu Zhao et al. (1993) desenvolveram um sistema de escalonamento de ganhos PID através de lógica *fuzzy* para controle de processos. A abordagem utiliza o erro e a variação do erro para estimar os ganhos. O controlador

demonstrou melhor desempenho quando comparado com outros que utilizavam a sintonia de Ziegler e Nichols e o método de Kitamori. Já a investigação feita por Campos and Saito (2004) abordou um controlador fuzzy para diferentes tipos de cargas para unidade industrial em função das diferentes regiões de operação do sistema. O objetivo proposto foi o de minimizar não linearidades oriundas do processo e evitar perda de desempenho devido a manutenção de uma mesma sintonia do controlador. Esses trabalhos serviram como base para o projeto de sistema de ajuste de controlador aqui proposto.

Erenoglu et al. (2006) implementaram um controlador híbrido de PID e controlador *fuzzy* que, de forma inteligente, identificava a melhor ação de controle durante o processo. Esse esquema de controlador se mostrou mais eficiente do que somente o PID ou o controlador *fuzzy* isolado. O trabalho de (Zulfatman and Rahmat, 2009) propôs um controlador PID auto-ajustado por *fuzzy* com o objetivo de melhorar o desempenho de um atuador eletro-hidráulico. Quando comparado com controladores PID tradicionais, obteve uma significativa melhoria do desempenho.

Em (de Almeida et al., 2016) os autores desenvolveram um controlador *fuzzy* com o objetivo de gerenciar um reservatório de mistura, controlando a temperatura e o nível do fluido utilizado. De acordo com os autores, as estratégias de controle proporcionaram desempenhos satisfatórios. Por sua vez, Paiva et al. (2016) apresentaram um sistema de controle supervisório *fuzzy* para gerir os níveis de qualidade da água e do óleo na separação e no tratamento de petróleo. Os resultados indicaram que o modelo proposto implementou melhorias no processo, reduzindo o tempo de identificação de um possível descontrole nas variáveis e evitando a interferência manual do operador.

Os autores de Moura and Neto (2016) desenvolveram um controlador *fuzzy* para realizar o controle do nível de sólidos em silos e embarcaram esta lógica em CLPs industriais. O comportamento do processo atendeu às expectativas, mantendo o nível mínimo de material no silos e garantindo a produtividade desejada. Já no trabalho de Junior, Ênio Lopes et al. (2018), foi projetado um controlador *fuzzy* multivariável para uma planta de desaguamento de minério de ferro. Para a validação da proposta, dois cenários foram levados em conta e em ambos os controladores foram capazes de estender o tempo de operação sob condições de falhas nos filtros, resultando em um ganho de produtividade do sistema.

No trabalho desenvolvido por Bruno França Coelho et al. (2018) também foi proposto um controlador *fuzzy* para sistemas industriais, porém a variável controlada desta vez foi a vazão dos sólidos em viradores de vagões. Os resultados foram comparados com outros dois controladores (um controlador PI e outro PID sintonizado pelo método Ziegler e Nichols) e os autores concluíram que o controlador baseado em lógica *fuzzy* obteve um melhor desempenho.

Por fim, Carvalho et al. (2020) implementaram um controlador PID híbrido para o controle em uma planta de nível utilizando a referência do sistema como entrada do bloco *fuzzy*. O controlador desenvolvido obteve um melhor desempenho quando comparado ao PID com sintonia fixa.

A planta experimental de nível projetada nesta pesquisa será utilizada no presente artigo, visando complementar este estudo e implementar outros dois tipos de controladores para uma futura comparação entre os métodos.

O artigo proposto tem como objetivo apresentar a análise de desempenho de dois controladores PID escalonados por meio da lógica *fuzzy*, e sua validação realizada em uma planta experimental para controle de nível de reservatório. São utilizadas a abordagem que utiliza o valor de referência do sistema como a entrada Campos and Saito (2004) e a abordagem que por sua vez utiliza o erro e a variação do erro como entradas do sistema fuzzy (Zhen-Yu Zhao et al., 1993), para cada controlador. Uma análise de desempenho é investigada para destacar as vantagens da utilização dos controladores baseados em lógica *fuzzy*. Outra contribuição presente no artigo é a aplicação prática destes controladores em um sistema real, com uma validação mais efetiva quando comparada a sistemas simulados.

O artigo está organizado da seguinte maneira: A seção 1 apresenta a motivação, objetivo e referencial teórico da pesquisa. A seção 2 é dedicada a apresentar a plataforma tecnológica que foi objeto de estudo do trabalho, assim como tratar da proposta dos controladores desenvolvidos. A seção 3 por sua vez mostra as características de cada controlador. A seção 4 tem como objetivo expor os resultados adquiridos e a seção 5 apresenta as conclusões do artigo.

2. PLANTA DE ESTUDO E PROPOSTA DE INVESTIGAÇÃO

2.1 Plataforma Tecnológica

A plataforma utilizada é composta dos softwares MATLAB/Simulink e Labview. O primeiro foi utilizado para o desenvolvimento dos controladores, plotagem e análise dos resultados. Já no Labview, foi realizada a integração do software de controle e a planta experimental de nível com a utilização do DAQ da National Instruments. Também foi desenvolvida uma Interface Homem-Máquina (IHM) para facilitar na observação e manuseio do processo, como mostra a Fig. 1.

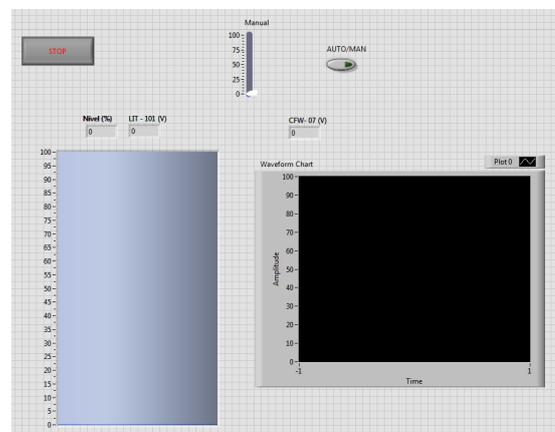


Fig. 1. IHM implementada no software Labview

O DAQ, utilizado como interface de comunicação entre o Matlab e a planta, já possui de forma nativa uma

comunicação com o *software* Labview devido a ambos serem do mesmo fabricante. A comunicação dos *softwares* foi feita por meio do protocolo *Ole for Process Control* (OPC), em uma estrutura cliente-servidor. A partir dessa comunicação, foi possível coletar sinais das variáveis de processo e enviar sinais de comando para o atuador, realizando os testes necessários na planta física.

A planta experimental de nível é montada em uma estrutura metálica composta por 3 bandejas. O tanque de processamento (TQ-01) possui diâmetro externo de 300mm, diâmetro interno de 290mm e altura total de 800mm. O zero de nível corresponde a 249mm e quando está em máxima operação possui um volume de aproximadamente 29 litros. O tanque de armazenamento (TQ-02) é um galão plástico com capacidade de 50 litros. A Fig. 2 apresenta todo este conjunto.



Fig. 2. Planta Experimental de Nível

O Transmissor eletrônico (LIT-100) da MicroCyber modelo NCSPT105S possui alimentação de 11.9 a 42 Vcc, saída de 4 a 20 mA e trabalha na faixa de 0 a 2 psi. Ele fornece a medição do nível de forma indireta por pressão diferencial.

O sistema de aquisição de dados modelo USB-6212 da National Instruments é um dispositivo multifuncional de aquisição de dados. Este dispositivo oferece entradas analógicas, entradas digitais, saídas digitais e dois contadores de 32 bits. Esse dispositivo é ideal para aplicações de teste, controle e projeto, incluindo *data logging* portátil, monitoramento em campo, além das aplicações de ensino e pesquisa desenvolvidas neste artigo.

O Inversor de Frequência CFW-07 da WEG consiste em modelo do tipo *Pulse Width Modulation* (PWM) e permite a variação ampla da velocidade de motores de indução trifásico padrão.

Todos estes instrumentos e equipamentos fazem parte da plataforma tecnológica proposta. A seguir, a proposta dos controladores será discutida, assim como as abordagens apresentadas e a caracterização que foi realizada no sistema.

2.2 Proposta de Controlador

O Controlador PID é uma técnica de controle de processos que une as ações proporcional (K_p), integral (K_i) e derivativa (K_d). A ação proporcional tem esse nome pois ela é diretamente proporcional ao erro e atua majoritariamente no regime transitório do sistema. A ação integral é diretamente proporcional à integral do sinal de erro no tempo e tem atuação no regime permanente do sistema. Já ação derivativa é diretamente proporcional a derivada do erro no instante (t) e fornece uma correção antecipada do erro, diminuindo o tempo de resposta e melhorando a estabilidade do sistema. A Fig. 3 representa o diagrama de blocos deste controlador.

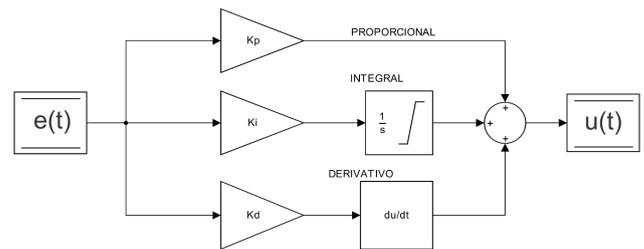


Fig. 3. Diagrama de blocos do Controlador PID.

Por ser considerado um controlador relativamente simples, é bastante provável que uma pessoa com poucos conhecimentos de engenharia e controle seja capaz de utilizá-lo (Tan et al., 2002). Ogata (2010) afirma que “Em particular, quando o modelo matemático da planta não é conhecido e, portanto, métodos de projeto analítico não podem ser utilizados, controles PID se mostram os mais úteis”. Complementa afirmando que tanto controladores PID simples quanto modificados conferem um controle satisfatório na área de controle de processos.

Por todos estes motivos, o Controlador PID foi adotado para a aplicação proposta. Ele terá seus ganhos escalonados por meio da lógica *fuzzy*, que é uma técnica que incorpora o pensamento humano em um sistema de controle. As variáveis linguísticas, funções de pertinência e regras de inferência alheias à lógica permitem esta incorporação.

Foram utilizadas duas abordagens diferentes com escalonamento de ganhos para lógica *fuzzy*. A primeira abordagem foi proposta por Campos and Saito (2004) e utiliza a referência (*set-point*) como parâmetro de entrada do bloco *fuzzy*.

A segunda abordagem foi proposta por Zhen-Yu Zhao et al. (1993) e utiliza o erro e a variação do erro para estimar os ganhos. Este arranjo tem o objetivo de alterar os valores de ganho gerados dentro de uma *range*, assumindo diferentes valores dentro desta amplitude.

No Apêndice A são mostrados com mais detalhes os diagramas de bloco referentes a construção destes controladores no Simulink.

2.3 Caracterização do Sistema

Como estratégia para a construção do controlador PID com escalonamento de ganhos por lógica *fuzzy*, um estudo foi realizado para identificar as melhores sintonias em

diferentes regiões da planta. Dessa forma, as regiões de atuação foram divididas em quatro partes conforme Fig. 4.

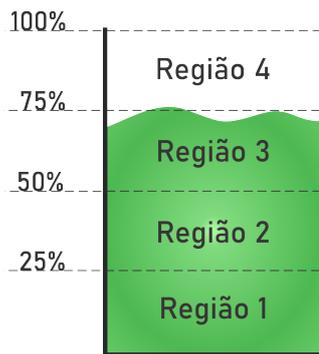


Fig. 4. Regiões de Operação da Planta

Para a sintonia de cada região, foram selecionadas as seguintes especificações de controle: M_p (máximo sobressinal) $\leq 10\%$, t_r (tempo de subida) $\leq 80s$ e t_s (tempo de acomodação) $\leq 90s$. Essas especificações foram determinadas para manter um padrão de desempenho desejável e foram utilizadas apenas para realização deste mapeamento.

A escolha da realização de um mapeamento de sintonias se deu ao fato de que a utilização de um método heurístico de sintonia não atendeu as especificações de desempenho para as diferentes referências possíveis do sistema real. A Tabela 1 apresenta as sintonias para determinados valores de referência encontradas a partir dos testes realizados.

Tabela 1. Relação de ganhos do PID para cada valor de referência.

| Referência (%) | Kp | Ki | Kd |
|----------------|----|------|----|
| 15 | 9 | 0.45 | 6 |
| 25 | 10 | 0.5 | 6 |
| 40 | 11 | 0.6 | 7 |
| 50 | 12 | 0.7 | 7 |
| 65 | 12 | 1.2 | 8 |
| 75 | 13 | 1.7 | 8 |
| 80 | 14 | 2 | 9 |
| 90 | - | - | - |

É importante destacar que o sistema não conseguiu alcançar um nível acima de 82% devido ao grau de abertura de sua válvula manual e por isso não foi realizada sintonia para referências superiores a este valor.

3. CONTROLADORES PID FUZZY

3.1 Controlador I

Nesta seção é apresentada a elaboração do sistema *fuzzy* para escalonar os ganhos do controlador PID de acordo com a referência aplicada ao sistema. Seu funcionamento é similar a um seletor de ganhos que irá comutar diversas sintonias dentro das regiões de operação da planta. Os estudos previamente realizados sobre as regiões de operação e suas sintonias são utilizados como base para o desenvolvimento do sistema baseado em lógica *fuzzy*. Para a elaboração da lógica, foi utilizada uma ferramenta com interface funcional existente no próprio Matlab®, chamada de *Fuzzy Logic Editor*. Na Fig. 5 é apresentada a tela inicial da interface desta ferramenta.



Fig. 5. Tela inicial da interface para edição da lógica *fuzzy* do Controlador I

As Fig. 6, 7, 8, e 9 apresentam as funções de pertinência e universo de discurso das variáveis de entrada e saída do sistema *fuzzy*. Funções de pertinência são funções que descrevem os sistemas dinâmicos através de modelos *fuzzy* e para o Controlador I foram escolhidas funções trapezoidais pois estas conseguiram realizar um melhor mapeamento do comportamento do sistema.

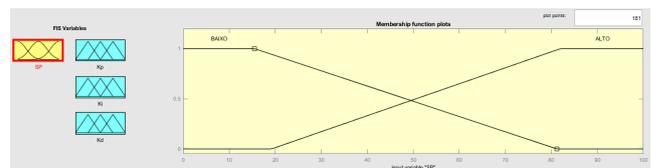


Fig. 6. Funções de pertinência da variável de entrada (*set-point*)

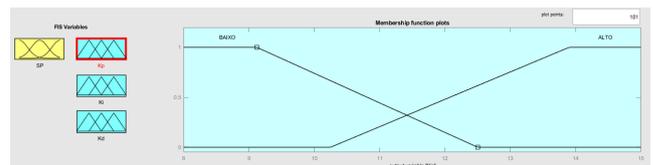


Fig. 7. Funções de pertinência da variável de saída Kp

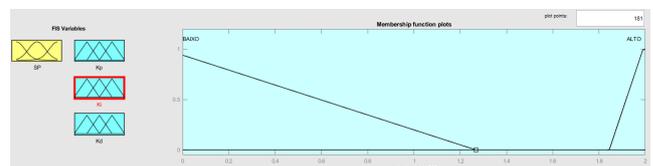


Fig. 8. Funções de pertinência da variável de saída Ki

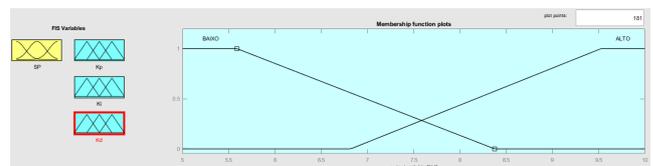


Fig. 9. Funções de pertinência da variável de saída Kd

A Tabela 2 mostra os ranges de cada variável associada ao sistema *fuzzy*.

Tabela 2. Range das variáveis do Sistema Fuzzy para o Controlador II.

| Variável | Range |
|----------------|---------|
| Set-Point (SP) | [0 100] |
| Kp | [8 15] |
| Ki | [0 2] |
| Kd | [5 10] |

Neste sistema, foram utilizadas duas regras linguísticas que possuem o objetivo de realizar a conexão entre as variáveis linguísticas. Estas são descritas a seguir.

- Se "SP" for **baixo** então "Kp" é **baixo**, "Ki" é **baixo** e "Kd" é **baixo**.
- Se "SP" for **alto** então "Kp" é **alto**, "Ki" é **alto** e "Kd" é **alto**.

Como resultado da combinação entre os arranjos das variáveis e as regras *fuzzy*, a ferramenta mostra o comportamento de cada variável de saída em função da variável de entrada em formato de superfície. As Fig. 10, 11 e 12 apresentam as superfícies da lógica elaborada.

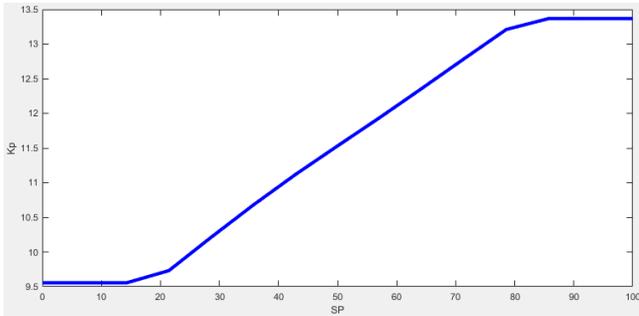


Fig. 10. Superfície Sp x Kp

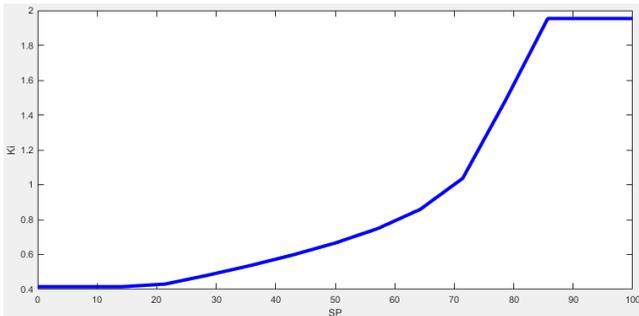


Fig. 11. Superfície Sp x Ki

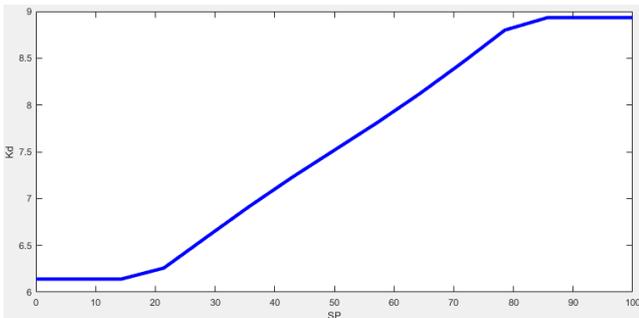


Fig. 12. Superfície Sp x Kd

3.2 Controlador II

Neste tópico será apresentada a elaboração da lógica *fuzzy* que irá escalonar o controlador PID de acordo o erro e a variação do erro do sistema. As variáveis de saída correspondem a Kpp e Kdp, que representam respectivamente Kp e Kd normalizados. Uma outra variável de saída é *alpha*, um fator constante aplicado no cálculo do Ki. As saídas normalizadas são convertidas nos seguintes ranges definidos para Kp e Kd: [3 9] e [7 15], respectivamente. Já Ki, segundo a proposta, é obtido pela equação 1

$$K_i = \frac{K_p^2}{\alpha K_d} \quad (1)$$

Como referido na seção 2.2, as escolhas das variáveis linguísticas assim como as regras *fuzzy* e o cálculo das variáveis são baseadas no modelo proposto por Zhen-Yu Zhao et al. (1993). Foram utilizadas funções de pertinência do tipo gaussiana para as saídas Kpp e Kdp devido ao fato de apresentarem o comportamento mais semelhante ao do modelo proposto.

Dando sequência ao processo de criação dos controladores, neste tópico será abordado o processo de edição da lógica *fuzzy* para o Controlador II. A Fig. 13 apresenta a tela inicial da interface para edição da lógica deste controlador.



Fig. 13. Tela inicial da interface para edição da lógica *fuzzy* do Controlador II

As Fig. 14, 15, 16, 17, e 18 apresentam as funções de pertinência e universo de discurso de cada variável dentro do editor de lógica *fuzzy* do Matlab®.

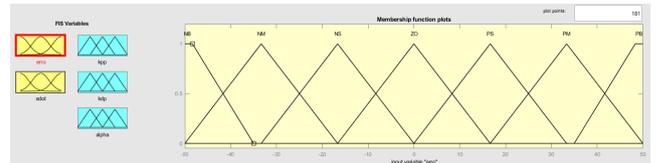


Fig. 14. Funções de pertinência da variável de entrada (erro)

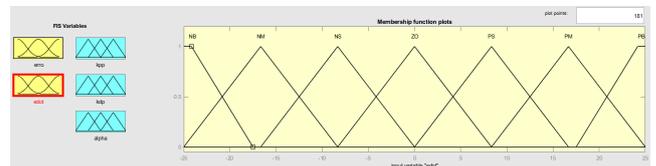


Fig. 15. Funções de pertinência da variável de entrada (variação do erro)

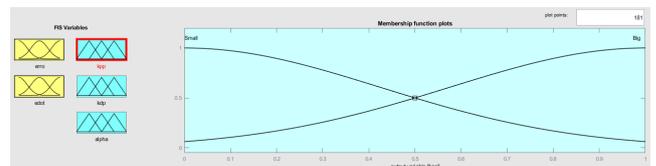


Fig. 16. Funções de pertinência da variável de saída (Kpp)

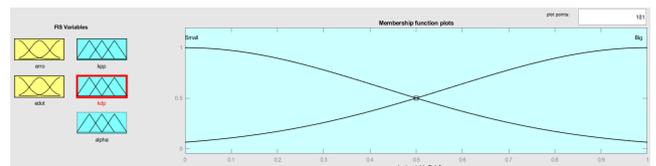


Fig. 17. Funções de pertinência da variável de saída (Kdp)

A Tabela 3 mostra os ranges de cada variável associada ao sistema *fuzzy*.

Neste sistema, segundo a abordagem proposta, quarenta e nove regras linguísticas associam as variáveis de entrada

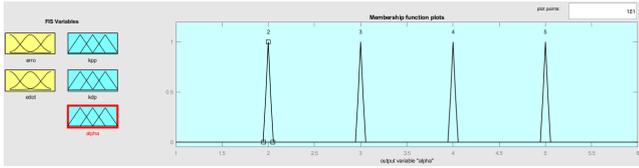


Fig. 18. Funções de pertinência da variável de saída (α)

Tabela 3. Range das variáveis do Sistema Fuzzy para o Controlador II

| Variável | Range |
|-----------------|----------|
| Erro | [-50 50] |
| Varição do Erro | [-25 25] |
| Kpp | [0 1] |
| Kdp | [0 1] |
| Alpha | [1 6] |

com as de saída. As Fig. 19, 20, 21 mostram esse conjunto de regras.

1. If (erro is NB) and (edot is NB) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
2. If (erro is NB) and (edot is NM) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
3. If (erro is NB) and (edot is NS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
4. If (erro is NB) and (edot is ZO) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
5. If (erro is NB) and (edot is PS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
6. If (erro is NB) and (edot is PM) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
7. If (erro is NB) and (edot is PB) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
8. If (erro is NM) and (edot is NB) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
9. If (erro is NM) and (edot is NM) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
10. If (erro is NM) and (edot is NS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
11. If (erro is NM) and (edot is ZO) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
12. If (erro is NM) and (edot is PS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
13. If (erro is NM) and (edot is PM) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
14. If (erro is NM) and (edot is PB) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
15. If (erro is NS) and (edot is NB) then (kpp is Small)(kdp is Big)(alpha is 4) (1)
16. If (erro is NS) and (edot is NM) then (kpp is Small)(kdp is Big)(alpha is 3) (1)

Fig. 19. Primeiro conjunto de Regras Linguísticas do Sistema Fuzzy para o Controlador II

17. If (erro is NS) and (edot is NS) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
18. If (erro is NS) and (edot is ZO) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
19. If (erro is NS) and (edot is PS) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
20. If (erro is NS) and (edot is PM) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
21. If (erro is NS) and (edot is PB) then (kpp is Small)(kdp is Big)(alpha is 4) (1)
22. If (erro is ZO) and (edot is NB) then (kpp is Small)(kdp is Big)(alpha is 5) (1)
23. If (erro is ZO) and (edot is NM) then (kpp is Small)(kdp is Big)(alpha is 4) (1)
24. If (erro is ZO) and (edot is NS) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
25. If (erro is ZO) and (edot is ZO) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
26. If (erro is ZO) and (edot is PS) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
27. If (erro is ZO) and (edot is PM) then (kpp is Small)(kdp is Big)(alpha is 4) (1)
28. If (erro is ZO) and (edot is PB) then (kpp is Small)(kdp is Big)(alpha is 5) (1)
29. If (erro is PS) and (edot is NB) then (kpp is Small)(kdp is Big)(alpha is 4) (1)
30. If (erro is PS) and (edot is NM) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
31. If (erro is PS) and (edot is NS) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
32. If (erro is PS) and (edot is ZO) then (kpp is Big)(kdp is Small)(alpha is 2) (1)

Fig. 20. Segundo Conjunto de Regras Linguísticas do Sistema Fuzzy para o Controlador II

Como resultado da combinação entre os arranjos das variáveis e as regras fuzzy, a ferramenta mostra o comportamento de cada variável de saída em função da variável de entrada em formato de superfície. As Fig. 22, 23 e 24 apresentam as superfícies da lógica elaborada.

4. RESULTADOS

Depois da elaboração dos dois controladores PID-fuzzy propostos, foram realizados os testes com o objetivo de obter resultados e fazer as devidas comparações que serão

33. If (erro is PS) and (edot is PS) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
34. If (erro is PS) and (edot is PM) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
35. If (erro is PS) and (edot is PB) then (kpp is Small)(kdp is Big)(alpha is 4) (1)
36. If (erro is PM) and (edot is NB) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
37. If (erro is PM) and (edot is NM) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
38. If (erro is PM) and (edot is NS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
39. If (erro is PM) and (edot is ZO) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
40. If (erro is PM) and (edot is PS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
41. If (erro is PM) and (edot is PM) then (kpp is Big)(kdp is Big)(alpha is 3) (1)
42. If (erro is PM) and (edot is PB) then (kpp is Small)(kdp is Big)(alpha is 3) (1)
43. If (erro is PB) and (edot is NB) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
44. If (erro is PB) and (edot is NM) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
45. If (erro is PB) and (edot is NS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
46. If (erro is PB) and (edot is ZO) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
47. If (erro is PB) and (edot is PS) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
48. If (erro is PB) and (edot is PM) then (kpp is Big)(kdp is Small)(alpha is 2) (1)
49. If (erro is PB) and (edot is PB) then (kpp is Big)(kdp is Small)(alpha is 2) (1)

Fig. 21. Terceiro Conjunto Regras Linguísticas do Sistema Fuzzy para o Controlador II

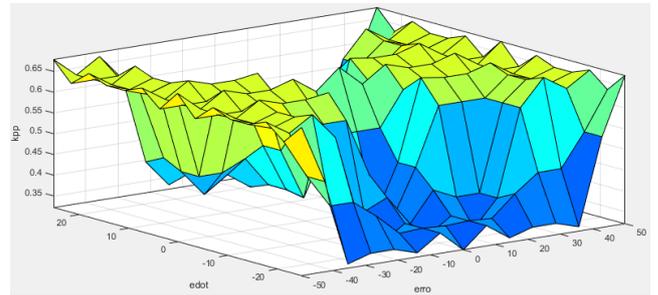


Fig. 22. Superfície Kpp x erro x variação do erro.

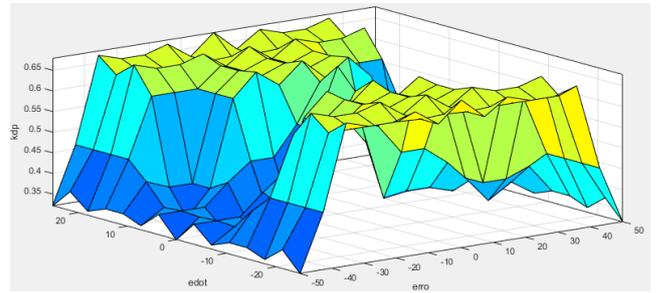


Fig. 23. Superfície Kdp x erro x variação do erro.

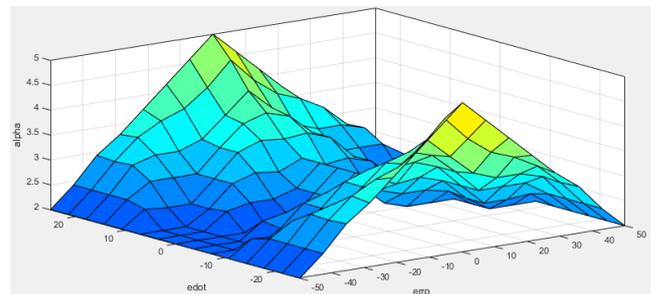


Fig. 24. Superfície Alpha x erro x variação do erro.

mostradas neste capítulo. Estes testes foram realizados com 3 mudanças de *set-point*: 25%, 50% e 75%.

Para esta análise foram escolhidas as seguintes métricas de controle: Mp (Máximo Sobressinal), tr (tempo de subida) e ts (tempo de acomodação ao critério de 5%). Também foi escolhido o MSE (*Mean Square Error*) para medir a eficácia de cada controlador, este é descrito pela equação 2.

$$MSE = \sum_{t=1}^n \frac{(VP(t) - SP(t))^2}{n} \quad (2)$$

Onde MSE é o erro médio quadrático, $VP(t)$ é a variável de processo em função do tempo e $SP(t)$ é a referência em função do tempo. Já n é o número de amostras do sinal.

A Fig. 25 mostra as curvas com desempenho de cada controlador para o teste proposto e a Tabela 4 apresenta as métricas atingidas por cada controlador.

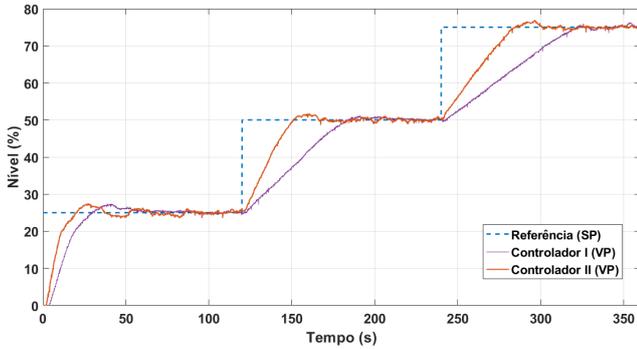


Fig. 25. Desempenho dos Controladores para variações em degrau.

O objetivo da aplicação era encontrar um controlador com adaptação automática e veloz frente às variações de carga e também que possuísse uma taxa de erro relativamente baixa. A partir desta análise, é possível concluir que o Controlador II obteve um melhor desempenho de controle, visto que apresenta uma resposta mais rápida e seu MSE é menor quando comparado com o Controlador I. Mesmo que tenha apresentado Máximo Sobressinal maior em todas as mudanças de patamar, o controlador ainda sim mostrou-se eficaz pois o sistema real utilizado consegue admitir uma taxa de erro aceitável nesta variável específica. A Fig. 26 e a Fig. 27 mostram as variações dos ganhos durante o teste para o Controlador I e o Controlador II, respectivamente.

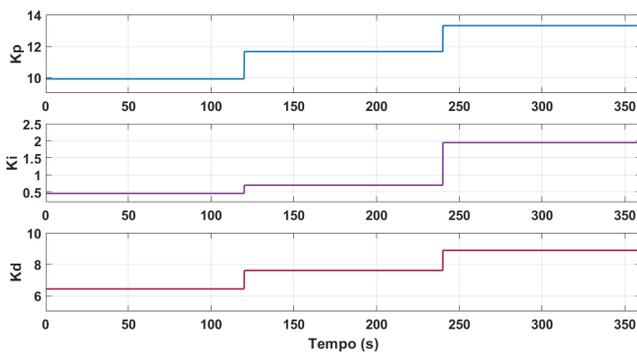


Fig. 26. Variação dos ganhos para o Controlador I.

Os testes demonstram que para o objetivo deste trabalho, variar os ganhos de acordo com o erro e a sua variação, em geral, é melhor que realizar o chaveamento de ganhos a partir da referência

5. CONCLUSÃO

Os resultados deste trabalho demonstraram que utilizar o mapeamento das regiões de operação foi importante

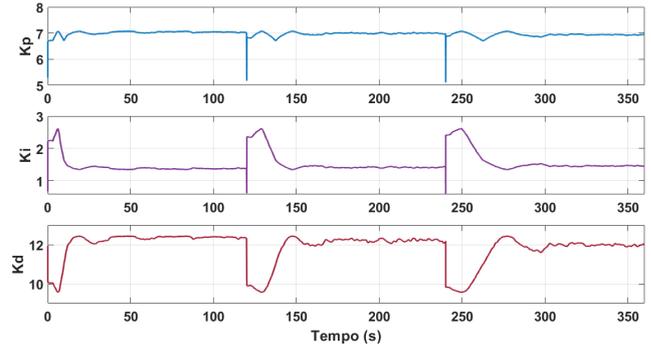


Fig. 27. Variação dos ganhos para o Controlador II.

para identificar a não linearidade inerente ao processo e contribuiu positivamente para o projeto do sistema de escalonamento *fuzzy*.

A abordagem proposta por Campos and Saito (2004), que utiliza a referência como variável de entrada, mostrou-se capaz de aprimorar as capacidades de um PID tradicional, chaveando os ganhos de acordo com a região de atuação. Porém este controlador mostrou-se inferior ao que utiliza a abordagem proposta por Zhen-Yu Zhao et al. (1993). Esta segunda abordagem provou ser um método melhor para lidar com as dinâmicas e não linearidades presentes na planta experimental de nível.

É preciso reconhecer que, de acordo com as características da planta utilizada, obter um controlador com um máximo sobressinal maior era perfeitamente aceitável. Porém, em situações menos maleáveis, esta variável pode ter uma prioridade mais alta e por isso será preciso realizar uma escolha diferente entre os controladores.

Para dar continuidade ao trabalho apresentado, é sugerida a implementação de testes que explorem a reação dos controladores a distúrbios provocados na vazão de saída da planta. Também é interessante que trabalhos futuros envolvendo controladores PID-*fuzzy* utilizem novas abordagens explorando novos tipos de conjuntos *fuzzy* e diferentes arranjos de funções de pertinência.

AGRADECIMENTOS

Os autores agradecem à Coordenação de Aperfeiçoamento Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 pelo suporte financeiro.

REFERÊNCIAS

- Bruno França Coelho, José Pinheiro de Moura, João Viana da Fonseca Neto, and Patrícia Helena Moraes Rêgo (2018). Sintonia online de Controladores PID Sintonizado por Regras Fuzzy em Sistemas Industriais com Aplicação em Processo Operacional de Viradores de Vagões. In *XXII Congresso Brasileiro de Automática*. João Pessoa - PB, Brasil. doi:10.20906/CPS/CBA2018-1134. URL <http://www.swge.inf.br/proceedings/paper/?P=CBA2018-1134>.
- Campos, M.M. and Saito, K. (2004). *Sistemas Inteligentes em Controle e Automação*. Ciência Moderna, Rio de Janeiro, Brasil, 1st edition.
- Carvalho, A.S., de Almeida, G.C.S., and Salles, R.d.S. (2020). Implementação de um Controlador Híbrido

Tabela 4. Resultado das Métricas de Controle

| Controlador | MSE | Mp | | | Tr | | | Ts | | |
|----------------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| | | 0-25% | 25-50% | 50-75% | 0-25% | 25-50% | 50-75% | 0-25% | 25-50% | 50-75% |
| Controlador I | 85.07 | 8.24% | 0% | 0% | 28.69s | 63.10s | 80.90s | 44.85s | 65.20 | 83.10s |
| Controlador II | 47.71 | 11.32% | 3.16% | 2.00% | 20.15s | 30.70s | 46.20s | 36.55s | 37.50s | 51.40s |

Fuzzy-PID: Análise de desempenho em uma planta de nível. *Revista Valore*, 5.

de Almeida, J.P.L.S., Angélico, B.A., and Goedel, A. (2016). Estratégias de Controle Fuzzy Aplicadas a um Processo Multivariável de Temperatura e Nível. In *XXI Congresso Brasileiro de Automática*. Vitória, ES, Brasil.

de Moura, J.P. and Neto, J.V.d.F. (2016). Modelo de um Controle de Nível de Sólidos em Silos Usando Sistemas Fuzzy Aplicados em PLC Industrial. In *XXI Congresso Brasileiro de Automática*. Vitória, ES, Brasil.

Erenoglu, I., Eksin, I., Yesil, E., and Guzelkaya, M. (2006). An Intelligent Hybrid Fuzzy Pid Controller. In *ECMS 2006 Proceedings edited by: W. Borutzky, A. Orsoni, R. Zobel*, 62–66. ECMS, Bonn, Germany. doi:10.7148/2006-0062. URL <http://www.scs-europe.net/dlib/2006/2006-0062.htm>.

Junior, Ênio Lopes, Ferreira, Antônio Eustáquio, Moreira, Vinicius Silva, and Euzébio, Thiago Antonio Melo (2018). Projeto de um Controlador Fuzzy Multivariável para uma Planta de Desaguamento de Minério de Ferro. In *XXII Congresso Brasileiro de Automática*. João Pessoa - PB, Brasil. doi:10.20906/CPS/CBA2018-1283. URL <http://www.swge.inf.br/proceedings/paper/?P=CBA2018-1283>.

Ogata, K. (2010). *Engenharia de Controle Moderno*. Pearson Prentice Hall, São Paulo, 5 edition.

Paiva, V.W.M.M.C., Moratori, P.B., de Campos, M.C.M.M., Bertini, L., and Copetti, A. (2016). Controle Supervisório Fuzzy Aplicado ao Processo de Separação e Tratamento de Óleo e Água em uma Plataforma de Petróleo. In *XXI Congresso Brasileiro de Automática*. Vitória, ES, Brasil.

Tan, K., Huang, S., and Ferdous, R. (2002). Robust self-tuning PID controller for nonlinear systems. *Journal of Process Control*, 12(7), 753–761. doi:10.1016/S0959-1524(02)00005-7. URL <https://linkinghub.elsevier.com/retrieve/pii/S0959152402000057>.

Tanscheit, R. (2003). Sistemas Fuzzy. In *VI Simpósio Brasileiro de Automação Inteligente*. Bauru, SP.

Zhen-Yu Zhao, Tomizuka, M., and Isaka, S. (1993). Fuzzy gain scheduling of PID controllers. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(5), 1392–1398. doi:10.1109/21.260670. URL <http://ieeexplore.ieee.org/document/260670/>.

Zulfatman and Rahmat, M.F. (2009). Application of Self-Tuning Fuzzy PID Controller on Industrial Hydraulic Actuator Using System Identification Approach. *International Journal on Smart Sensing and Intelligent Systems*, 2(2), 246–261. doi:10.21307/ijssis-2017-349. URL https://www.exeley.com/in_jour_smart_sensing_and_intelligent_systems/doi/10.21307/ijssis-2017-349.

Apêndice A. DIAGRAMAS DE BLOCO

Neste apêndice contém as figuras com os diagramas de blocos utilizados no Simulink. A Figura A.1 mostra o

diagrama de blocos da malha de controle no Simulink para o método do Controlador I.

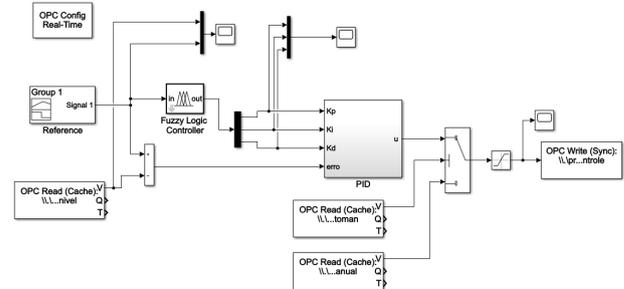


Fig. A.1. Diagrama de Blocos para o Método do Controlador I

A Figura A.2 mostra o diagrama de blocos da malha de controle no Simulink para o método do Controlador II.

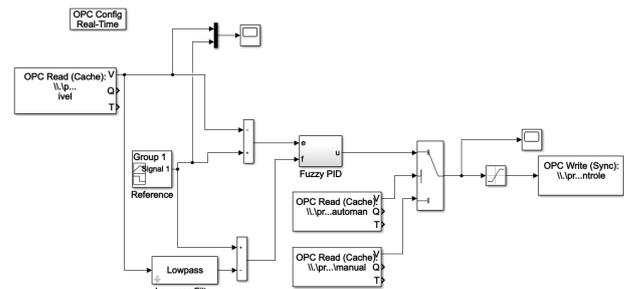


Fig. A.2. Diagrama de Blocos para o Método do Controlador II

Por fim, a Fig. A.3 mostra o esquema dentro do subsistema Fuzzy PID na malha de controle do Controlador II.

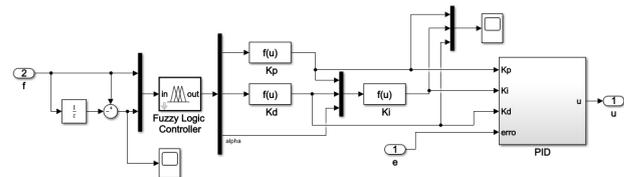


Fig. A.3. Diagrama de Blocos para o Método do Controlador II - Dentro do Subsistema "Fuzzy PID"