

## Projeto e Implementação de um Controlador FOPI em uma Bancada de Controle de Nível

Santos, Nikolas M. L.\*. Martins, Paulo R. O.\*\*. Silva Neto, Gustavo C.\*\*. Ayres Junior, Florindo A. C.\*\*\*

\* Universidade Federal do Amazonas  
(e-mail: [nikolasmec@gmail.com](mailto:nikolasmec@gmail.com))

\*\*Universidade Federal do Amazonas, Departamento de Engenharia Mecânica  
(e-mail: [pmartins@ufam.edu.br](mailto:pmartins@ufam.edu.br), [gustavoneto@ufam.edu.br](mailto:gustavoneto@ufam.edu.br))

\*\*\*Universidade Federal do Amazonas, Departamento de Engenharia Elétrica  
(e-mail: [florindoayres@ufam.edu.br](mailto:florindoayres@ufam.edu.br))

**Abstract:** In this work we present the design of a fractional order PI control and a PLC based implementation for an industrial scale level control bench. The system is a first order plus deadtime, with dead zone until it reaches the operating regime, the pump presented saturation power. The controller design used a constrained optimization and took into account the reduction of settling time, null steady state error and robustness specifications. For the implementation, an adaptation of the approximation method of a long memory discrete time PID controller (LDPID) was used. The programming was developed in Ladder logic. Simulations and experimental tests of regulation, tracking and robustness were carried out. The FOPI controller presented the better performance compared to the classic PI both in simulations and in experimental tests.

**Resumo:** Neste trabalho apresentamos o projeto de um controle PI de ordem fracionária e a implementação baseada em CLP para uma bancada de controle de nível em escala industrial. O sistema originalmente apresenta dinâmica lenta de primeira ordem com atraso de tempo, saturação de potência da bomba e zona morta até atingir o regime de operação. O projeto do controlador utilizou uma otimização com restrições e levou em conta a redução do tempo de acomodação, erro nulo em estado estacionário e especificações de robustez. Para a implementação utilizou-se uma adaptação do método de aproximação discreta para um controlador PID de longa memória a tempo discreto (LDPID). A programação foi desenvolvida em lógica *Ladder*. Foram realizadas simulações e testes experimentais de regulação, rastreamento e robustez. O controlador FOPI apresentou melhor desempenho em comparação com PI clássico tanto nas simulações quanto nos testes experimentais.

**Keywords:** FOPI; PLC; Implementation; LDPID.

**Palavras-chaves:** FOPI; CLP; implementação; LDPID.

### 1. INTRODUÇÃO

Os controladores PID são amplamente utilizados na indústria devido a sua praticidade, flexibilidade de aplicação e fácil implementação (Franklin et al., 2013). No entanto, esses controladores podem apresentar redução de desempenho frente a sistemas com não linearidades, incertezas ou distúrbios. Visando alcançar maior robustez da resposta dos sistemas, uma das estratégias utilizadas consiste em se adotar controladores PID de ordem fracionária (Shah and Agashe, 2016).

Nos controladores PID de ordem fracionária (FOPID ou  $PI^{\lambda}D^{\mu}$ ), flexibilizam-se os parâmetros das ações integrativa e derivativa, em suas respectivas ordens, para valores não inteiros. Desse modo a região de abrangência dos parâmetros de ordem do controlador é ampliada como pode ser visto na Figura 1 (Monje et al., 2010).

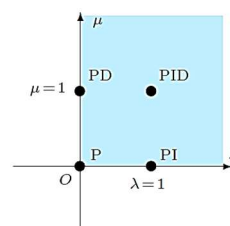


Figura 1 – Abrangência do  $PI^{\lambda}D^{\mu}$ .

A inclusão desses coeficientes ( $\lambda$  e  $\mu$ ) a serem sintonizados permite que o projeto do controlador atinja especificações adicionais em comparação com os PIDs clássicos, porém tornam o processo de sintonia mais desafiador (Shah and Agashe, 2016). Neste artigo, adotou-se o método de sintonia proposto por Monje et al. (2004) que consiste em uma otimização com restrições baseada na resposta em frequência.

Uma vez finalizada a sintonia do controlador, precisa-se aproximar o controlador de ordem fracionária por um controlador de ordem inteira para somente assim prosseguir com a implementação. Faz-se o uso de alguma função de geração (Tustin bilinear por exemplo) e geralmente adota-se uma Expansão dessa função de geração. Dentre os métodos mais comuns, podemos citar a Expansão em Séries de Potência (PSE) ou a Expansão em Frações Contínuas (CFE). Finalmente, o controlador na forma expandida pode ser implementado. Petras (2012) sugere que os CLPs seriam os dispositivos mais indicados para se implementar FOPID para aplicações industriais.

Merrikh-bayat et al. (2014) propôs uma implementação LDPID que consiste numa expansão do tipo PSE da aproximação de Tustin. Oprzedkiewicz et al. (2017) discutiu a implementação de um operador de ordem fracionária em CLP utilizando a aproximação CFE e exemplificou a implementação em uma plataforma CLP SIEMENS SIMATIC S7 1200. Mystkowski and Zolotas (2018), procederam com uma implementação em um CLP S7-300 (SIMATIC, Siemens) utilizando a PSE tanto por Tustin quanto por Euler. O controlador foi validado em uma bancada de controle de nível. O desempenho apresentado com o FOPID foi melhor que com o PID clássico projetado. El-Shafei et al. (2017), implementou um FOPID a um Sistema de Controle Distribuído ABB DCS AC 700F2 de uma bancada de processo de controle de nível Lab-Volt Level Process Station, fazendo uso da aproximação do tipo (LDPID) com programação em código de texto-estruturado e bloco de funções.

O presente trabalho se propôs a implementar um controlador diretamente no CLP Rockwell Micro 820 por meio de programação em linguagem *ladder*. Para tanto, utilizou-se o método de aproximação para um controlador PID de longa memória a tempo discreto (LDPID) proposto por Merrikh-bayat et al. (2015).

Dessa maneira, esse artigo está dividido como se segue: seção 2: Controladores Fracionários; seção 3: Método de Sintonia; seção 4: Discretização e Implementação, seção 5: Planta de Trabalho; seção 6: Projeto do Controlador; seção 7: Resultados e Discussões, seção 8: conclusão e seção 9: Referências.

## 2. CONTROLADOR FRACIONÁRIO

O cálculo fracionário surgiu como uma ampliação do cálculo tradicional, o qual trabalha com derivadas e integrais de ordem não inteira, as quais podem incluir ordem complexa (Monje et al., 2011; Valerio; Da Costa, 2013). O método de sintonia e implementação utilizados neste artigo englobam apenas o conjunto  $\mathbb{R}$ .

Mediante isso, torna-se necessário ter uma definição a respeito dos operadores que definem a derivada e a integral de ordem não inteira. Para a avaliação da resposta em frequência a definição de Caputo é mais adequada (Monje et al., 2010), a qual é escrita como

$$D_t^\alpha f(t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{f^{(m+1)}(\tau)}{(t-\tau)^\alpha} d\tau, \quad (1)$$

$$D_t^{-\gamma} f(t) = \frac{1}{\Gamma(\gamma)} \int_0^t \frac{f(\tau)}{(t-\tau)^{1-\gamma}} d\tau, \quad \gamma > 0 \quad (2)$$

onde  $D_t^{-\gamma}$  e  $D_t^\alpha$  são os operadores fracionários da integral e da derivada de Caputo, respectivamente;  $\alpha = m + \gamma, m \in \mathbb{Z}; \Gamma(\cdot)$  é a função gama dada por  $\Gamma(p) = (p-1)!$ ;  $t$  é o valor do tempo de integração; e  $m$  é definido de forma que  $m \in \mathbb{N}^+, m-1 < \mu \leq m$ . Aplicando essas definições à equação do PID e usando a transformada de Laplace, o controlador PID pode ser reescrito como (Podlubny, 1994)

$$C(s) = \frac{U(s)}{E(s)} = K_p + K_i s^{-\lambda} + K_d s^\mu \quad (3)$$

onde  $K_p, K_i$  e  $K_d$  são, respectivamente, os ganhos proporcional, integral e derivativo do controlador, e  $\lambda$  e  $\mu$ , com  $(\lambda, \mu) > 0$ , são, os expoentes fracionários de integração e derivação, respectivamente.

## 3. SINTONIA DO $PI^\lambda D^\mu$

Para encontrar os valores dos coeficientes do controlador, utiliza-se o método desenvolvido por Monje et al. (2004). Este método utiliza as seguintes equações:

1) *Margem de fase ( $\varphi_m$ ) e frequência de cruzamento de ganho ( $\omega_{cg}$ ):*

$$|C(j\omega_{cg})G(j\omega_{cg})| = 0 \text{ dB} \quad (4)$$

$$\arg|C(j\omega_{cg})G(j\omega_{cg})| = \varphi_m - \pi \quad (5)$$

2) *Robustez a variação de ganho na planta:* A expressão que garante a resistência a variação de ganho é dada pela (6). Esta restrição faz com que a fase para o sistema em malha aberta apresente uma certa planicidade aos arredores da frequência de cruzamento de ganho, tornando-a quase constante dentro do intervalo de  $\omega_{cg}$ . Isto faz com que o sistema apresente robustez a mudanças de ganho e o sobressinal apresentado é constante no intervalo de ganho (propriedade iso-damping).

$$\left. \frac{d(\arg C(j\omega_{cg})G(j\omega_{cg}))}{d\omega} \right|_{\omega \rightarrow \omega_{cg}} = 0 \quad (6)$$

3) *Função de sensibilidade complementar  $T(j\omega)$ :* é usada para a rejeição de ruídos, que comumente são encontrados em regiões de alta frequência. A fórmula para sua determinação é dada pela (7)

$$|T(j\omega)| = \left| \frac{C(j\omega)G(j\omega)}{1 + C(j\omega)G(j\omega)} \right|_{dB} \leq A \text{ dB}, \quad (7)$$

$$\forall \omega \geq \omega_t \rightarrow |T(j\omega_t)| = A \text{ dB}$$

onde,  $A$  é a rejeição ao ruído desejada para a faixa de frequência compreendida em  $\omega \geq \omega_t$  (região de alta frequência).

4) *Função de sensibilidade  $S(j\omega)$* : esta função age para conferir a planta uma boa rejeição a perturbação, normalmente encontrada nas regiões de baixa frequência. A expressão que define esta rejeição é dada pela (8)

$$|S(j\omega)| = \left| \frac{1}{1 + C(j\omega)G(j\omega)} \right|_{dB} \leq B \text{ dB}, \quad (8)$$

$$\forall \omega \leq \omega_s \rightarrow |S(j\omega_s)| = B \text{ dB}$$

em que  $B$  é a rejeição ao distúrbio desejada para a faixa de frequência  $\omega \leq \omega_s$  (região de baixa frequência).

Os valores são encontrados por meio da minimização da função objetivo escrita a partir da (4), com as restrições de (5)-(8). Utilizou-se o Toolbox do MATLAB denominado FOMCON, desenvolvido por Tepljakov et al. (2011), sendo uma ferramenta prática de projeto e implementação de controladores FOPID. O algoritmo de otimização adotado foi o FMINCON, no modo Interior-Point. A função *fpid\_optim* chama a interface da Figura 2, onde os valores iniciais do controlador podem ser inseridos, assim como as especificações da resposta em frequência almejada e o algoritmo utilizado para encontrar os valores ótimos

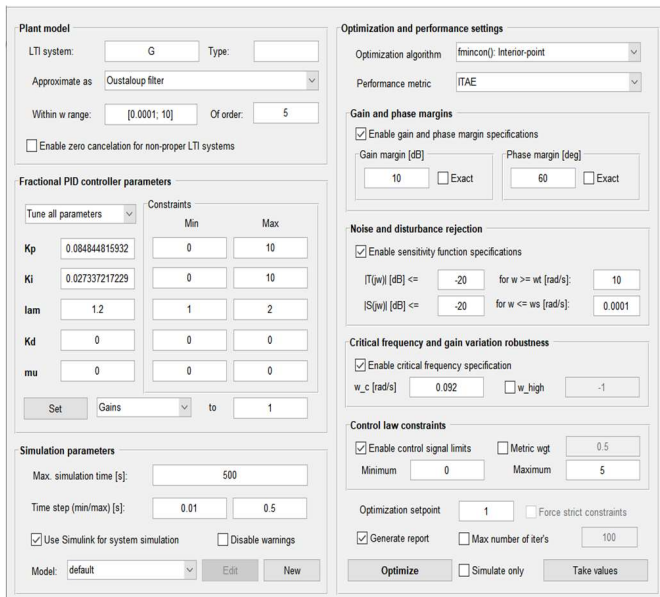


Figura 2 – Interface da função *fpid\_optim* do toolbox FOMCON

#### 4. DISCRETIZAÇÃO e IMPLEMENTAÇÃO

A forma direta da equação do controlador de ordem fracionária pode ser implementada em softwares que solucionem numericamente as funções de ordem fracionária. Porém, é mais viável e prático encontrar uma função de transferência de ordem inteira que aproxime bem dinâmica da expressão de ordem não inteira (Valerio; da Costa, 2013).

Este trabalho utiliza o método desenvolvido pelos autores Merrikh-bayat et al. (2015), a qual aproxima a equação do controlador  $PI^\lambda D^\mu$  para um controlador PID em tempo discreto de longa memória (LDPID), aproximando a expressão diretamente no tempo discreto. Este método é escolhido devido a sua capacidade manter o fenômeno conhecido como *princípio de longa memória*, o qual é inerente aos controladores FOPID, mas que é deteriorado quando se faz a aproximação para uma função de transferência de ordem inteira. Além disso, elimina a necessidade de aplicar filtros passa-baixa em série para implementar a parte derivativa do controlador, diminuindo os efeitos negativos causados pelo atraso no dispositivo que emula o controlador. (Merrikh-bayat et al., 2015). A equação utilizada é

$$C(z) = K_p + K_d \sum_{k=0}^M f_k(\mu)^* z^{-k} + K_i \frac{(1+z^{-1})}{(1-z^{-1})} \sum_{k=0}^M f_k(1-\lambda)^* z^{-k} \quad (9)$$

em que  $K_p = k_p$ ,  $K_d = k_d \alpha^\mu$ ,  $K_i = k_i \alpha^{-\lambda}$  e  $M \in \mathbb{N}^+$  é o número de memória,  $\alpha \triangleq \omega_{cg} / \tan(\omega_{cg}T/2)$  sendo  $\omega_{cg}$  a frequência de cruzamento de ganho em malha aberta e  $T$  o período de amostragem, representado, assim, a passagem dos coeficientes para o domínio discreto diretamente na construção da equação. Também, a função  $f_k(\mu)$  é dada por

$$f_k(\mu) = \frac{1}{k!} * \frac{d^k}{d(z^{-1})^k} \left[ \frac{1-z^{-1}}{1+z^{-1}} \right] \Bigg|_{z^{-1}=0} \quad (10)$$

Considerando que um número de memória  $M = 5$  é uma aproximação suficientemente boa para o controlador fracionário da planta, pode-se escrever a equação de diferenças (El-shafei et al., 2017):

$$C(z) = \frac{w_0 + w_1 z^{-1} + w_2 z^{-2} + w_3 z^{-3} + w_4 z^{-4} + w_5 z^{-5} + w_6 z^{-6}}{1 - z^{-1}} \quad (11)$$

onde

$$w_0 = K_i + K_d + K_p \quad (12)$$

$$w_1 = -K_p - K_i(2(1-\lambda) - 1) - K_d(2\mu + 1) \quad (13)$$

$$w_2 = K_d(2\mu^2 + 2\mu) - K_i(-2(1-\lambda)^2 + 2(1-\lambda)) \quad (14)$$

$$w_3 = -K_i \left( 4 \frac{(1-\lambda)^3}{3} - 2(1-\lambda)^2 + 2 \frac{(1-\lambda)}{3} \right) - K_d \left( \frac{4\mu^3}{3} - 2\mu^2 + 2 \frac{\mu}{3} \right) \quad (15)$$

$$w_4 = K_d \left( 2 \frac{\mu}{3} + \frac{4\mu^3}{3} + \frac{4\mu^2}{3} + \frac{2}{3} \right) - K_i \left( -2 \frac{(1-\lambda)^4}{3} + 4 \frac{(1-\lambda)^3}{3} - 4 \frac{(1-\lambda)^2}{3} + \frac{2}{3} \right) \quad (16)$$

$$w_5 = -K_i \left( 4 \frac{(1-\lambda)^5}{15} - 2 \frac{(1-\lambda)^4}{3} + 4 \frac{(1-\lambda)^3}{3} - 4 \frac{(1-\lambda)^2}{3} - 2 \frac{(1-\lambda)}{5} \right) \quad (17)$$

$$w_6 = K_d \left( \frac{4\mu^5}{15} + \frac{4\mu^3}{3} + \frac{2\mu}{5} \right) - K_i \left( 4 \frac{(1-\lambda)^5}{15} + 4 \frac{(1-\lambda)^3}{3} + 2 \frac{(1-\lambda)}{5} \right) - K_d \left( \frac{4\mu^5}{15} + \frac{2\mu^4}{3} + \frac{4\mu^3}{3} + \frac{4\mu^2}{3} + \frac{2\mu}{5} \right) \quad (18)$$

A (11), para ser aplicada no controlador, é reescrita como

$$u(k) = e(k) \cdot w_0 + e(k-1) \cdot w_1 + e(k-2) \cdot w_2 + e(k-3) \cdot w_3 + e(k-4) \cdot w_4 + e(k-5) \cdot w_5 + e(k-6) \cdot w_6 + u(k-1) \quad (19)$$

em que  $u(k)$  é o sinal de controle atual, calculado com base no erro atual  $e(k)$  e em seus valores passados  $e(k-1) \dots e(k-6)$ , os quais são armazenados na memória do controlador, e do sinal de controle passado  $u(k-1)$  que também é guardada. A (19) é implementada no CLP utilizando a lógica *Ladder*, a qual pode ser vista em dos Santos (2022), onde o bloco denominado *pilha FIFO* funciona como memória para o armazenamento dos valores passados.

## 5. PLANTA DE TRABALHO

A planta utilizada neste artigo é a bancada de nível mostrada na Figura 3. A água fica no reservatório 1. Quando um setpoint é selecionado no CLP, um sinal de controle (SC) é enviado para o inversor de frequência, controlando a velocidade da bomba. A água do reservatório é bombeada até a região superior do tanque, sua entrada. O tanque é, então, preenchido até o nível selecionado no CLP. Para medir o nível, um sensor ultrassônico Siemens *The Probe* é usado. A função de transferência da planta foi obtida por Alves (2019) e está aqui escrita na (20). A Figura 4 é a resposta ao degrau em malha aberta da planta. Nota-se um tempo muito longo para a acomodação.

$$G(s) = \frac{1.67}{s + 0.0037} e^{-2.51s} \quad (20)$$

A bomba do sistema é a bomba centrífuga CAM-W10 da DANCOR, de 2CV de potência, que funciona em rede trifásica de 220V ou 380V, sendo controlada pelo inversor de frequência WEG CFW08. Seu limite máximo de tensão de trabalho, por razões de segurança, foi selecionado para 5 V. Com isso, cria-se uma saturação de sinal para o controlador, a qual deve ser levada em consideração para o projeto do controlador, assim como a adição de um sistema *anti-windup* que diminua o erro de acumulação causado pelo termo integral (Franklin et al., 2013; Da Silva, 2017). Para controlar a tensão no inversor de frequência, utiliza-se um CLP Rockwell micro

820 modelo LC20-20 QBB, com dois módulos de expansão (duas entradas em corrente e duas saídas em tensão).

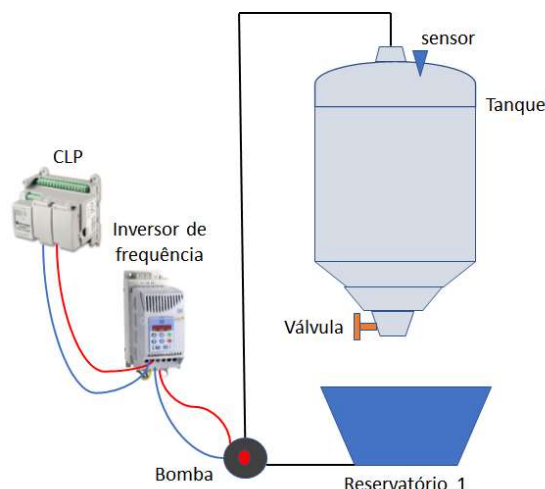


Figura 3 – Esquema de funcionamento da bancada de controle de nível

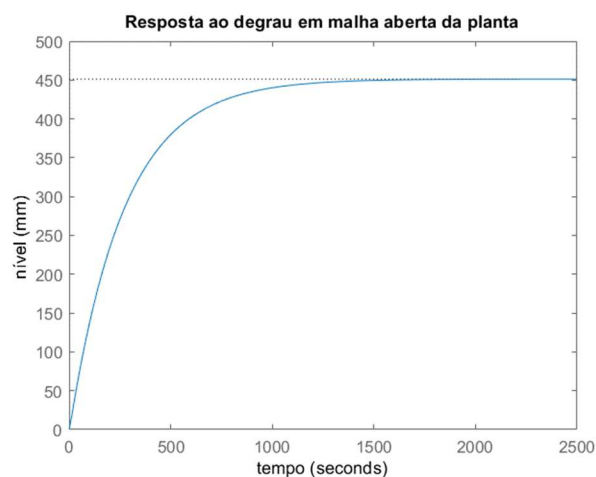


Figura 4 – Resposta ao degrau em malha aberta da planta

## 6. PROJETO DO FOPID

Os dois requisitos principais do projeto são: 1) diminuir significativamente o tempo de acomodação  $T_s$  com 2) erro nulo em regime permanente. Para projetar o controlador fracionário, primeiro foi feito o projeto de um controlador PID de ordem inteira. Alterando interativamente os parâmetros na ferramenta em resposta em frequência, o tipo de controlador que teve o melhor desempenho foi o tipo PI para os valores de frequência de banda  $\omega_{BW} = 0.1581 \text{ rad/s}$  e margem de fase  $\phi_m = 48^\circ$ ,  $K_p = 9.274 \cdot 10^{-2}$  e  $K_i = 3.248 \cdot 10^{-2}$ . De fato, como se trata de uma planta de primeira ordem com atraso relativamente pequeno, mediante ao objetivo de diminuir o tempo de acomodação da planta garantindo erro em regime permanente nulo, um simples PI pode compensar bem o sistema. Porém, esse controlador, durante a implementação,

apresentou dificuldade de rastrear valores de setpoint (SP) abaixo de 250 mm. No entanto, para verificar a capacidade de melhoria de resposta do controlador de ordem fracionária e do método de sintonia, decidiu-se utilizar estes valores como coeficientes iniciais no processo de minimização para encontrar os ganhos do FOPI. Para a comparação, outro controlador PI foi projetado, utilizando uma frequência de banda  $\omega_{BW} = 0.1442 \text{ rad/s}$  e margem de fase  $\phi_m = 60^\circ$ , amortecendo mais a resposta a despeito de um leve aumento no tempo de acomodação. Os coeficientes do controlador foram  $K_p = 8.484 \cdot 10^{-2}$  e  $K_i = 2.734 \cdot 10^{-2}$  na configuração ideal. Sendo a frequência de banda do sistema  $\omega_{BW} \approx 0.273 \text{ rad/s}$ , período de amostragem  $T = 1\text{s}$  corresponde uma frequência de amostragem de aproximadamente 22 vezes a frequência de banda, obedecendo a sugestão de Seborg et al. (2012).

O FOPID é projetado para ser um FPI. Na interface da figura 2, os valores iniciais para a busca foram os coeficientes do  $K_p$  e  $K_i$  do PI de ordem inteira, sendo o a faixa de procura  $0 < K_p < 10$  e  $0 < K_i < 10$ . O valor do coeficiente fracionário foi de  $\lambda = 1,2$  com faixa de busca de  $1 \leq \lambda \leq 2$ , pois escolher valores entre 1 e 2 já coloca automaticamente a integral do erro, garantindo erro em regime permanente nulo, eliminando o trabalho de reorganizar o coeficiente para ser implementado. Essa reorganização pode ser vista em Monje et al. (2010) e sua realização é simples, mas executá-la pode não ser prático para o operador. Os coeficientes  $K_d = \mu = 0$ . O índice de penalização escolhido é o ITAE, uma vez que este, por ser uma ponderação pelo tempo, é mais sensível às componentes do erro que existem na resposta após um longo intervalo de tempo, ou seja, no que tange a região do regime permanente. O algoritmo selecionado é o FMINCON: Interior Point. A margem de fase do objetivo da minimização é escolhida como  $\phi_m = 60^\circ$ , para garantir um bom amortecimento, margem de ganho  $g_m = 10\text{dB}$ , sendo uma restrição a mais conforme Tepljakov et al. (2011), e a frequência de cruzamento de ganho realocada para aproximadamente  $\omega_{cg} = 0,092 \text{ rad/s}$ . As funções de sensibilidade e sensibilidade complementar foram respectivamente,  $|S(j\omega)| \leq -20 \text{ dB}$ ,  $\forall \omega \geq 10 \text{ rad/s}$ . e  $|T(j\omega)| \leq -20 \text{ dB}$ ,  $\forall \omega \leq 0.0001 \text{ rad/s}$ . A minimização gerou

$$C(s) = 0.094338 + \frac{0.0024668}{s^{1.2459}} \quad (21)$$

Os coeficientes da (22) são inseridos na (6), gerando  $\omega_0 = 9.68 \cdot 10^{-2}$ ,  $w_1 = -9,066 \cdot 10^{-5}$ ,  $w_2 = 1.511 \cdot 10^{-3}$ ,  $w_3 = 1.05 \cdot 10^{-3}$ ,  $w_4 = -1.391 \cdot 10^{-3}$ ,  $w_5 = 1.175 \cdot 10^{-5}$ ,  $w_6 = 2.921 \cdot 10^{-4}$ . A simulação do controlador é realizada no diagrama de blocos da Figura 5. A implementação é feita no CLP.

## 7. SIMULAÇÃO

A simulação do sistema de controle é feita com o diagrama de blocos da figura 5. Uma vez que o controlador é implementado no CLP através da equação de diferenças, optou-se por representá-lo através de um bloco de função no MATLAB cujo algoritmo interno emule a equação de diferenças.

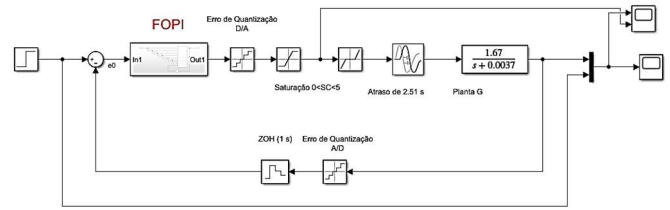


Figura 5 - Diagrama de blocos da simulação FOPI

Também, foi implementado o método *anti-windup* denominado de *signal de controle linear* (da Silva, 2017), cujo princípio de funcionamento se baseia em manter a consistência do sinal de controle que entra na planta G (saturado) com o valor do sinal de controle armazenado  $u(k-1)$ . Isso significa que o valor passado utilizado para o cálculo do valor atual do sinal de controle deve estar na faixa de saturação e, caso seja maior, recebe o valor máximo da saturação  $u(k)_{satMax}$ .

$$\begin{cases} \text{se } u(k) > u(k)_{satMax}, & u(k-1) \leftarrow u(k)_{satMax} \\ \text{se } u(k) < u(k)_{satMin}, & u(k-1) \leftarrow u(k)_{satMin} \\ \text{caso contrário,} & u(k-1) \leftarrow u(k) \end{cases}$$

O número de divisões para entrada digital do controlador com 16 bits é de  $2^{16} = 65536$  bits, tendo uma resolução de  $\frac{5}{65536} = 7.63 \cdot 10^{-2} \text{ mv}$ . Além disso, foi considerado a zona morta para o sinal de controle enviado ao inversor de frequência, a qual corresponde a faixa de potência na qual opera a bomba sem a capacidade de produzir qualquer mudança de nível (PV), ou mesmo manter. Foi observado experimentalmente que para um sinal de controle na faixa de  $0 \text{ V} < SC \leq 2.2 \text{ V}$ , não há potência suficiente na bomba alterar a saída PV.

## 8. RESULTADOS E DISCUSSÕES

### 8.1 Simulação

A simulação foi feita através do SIMULINK do software MATLAB utilizando o diagrama de blocos da figura 5 para o FOPI. Vale ressaltar que a faixa de trabalho do sensor corresponde a parte cilíndrica do tanque, desconsiderando a seção de área variável. Com isso, os dados só são coletados a partir de 235 mm. Por esse motivo, na simulação e nos experimentos, primeiro realizou-se um degrau em 238mm e esperou-se a acomodação nesse nível. Então, a partir desse nível, aplicou-se os degraus desejados. Para a simulação do comportamento do FOPI e do PI, seguindo o procedimento anteriormente descrito, aplicou-se a referência de 500 mm para ambos os compensadores, a simulação resultou nos gráficos da Figura 6 e Figura 7 com os valores de desempenho: tempo de acomodação ( $T_s$ ), tempo de subida ( $T_r$ ), sobressinal e índice ITAE da tabela 1.



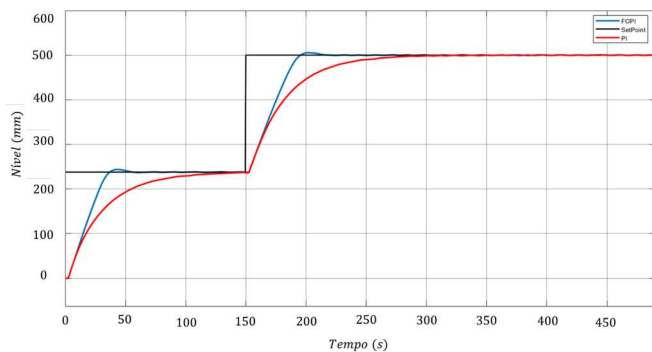


Figura 6 – Simulação da resposta ao degrau de 500 mm do FOPI (traço azul) e PI (traço vermelho)

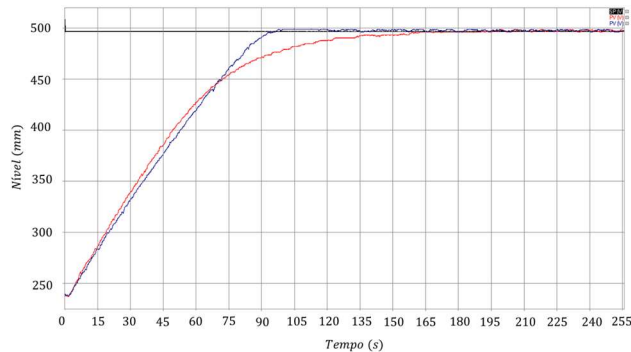


Figura 8 – Resposta ao degrau de 500 mm do sistema com o FOPI (linha azul) e com o PI (linha vermelha)

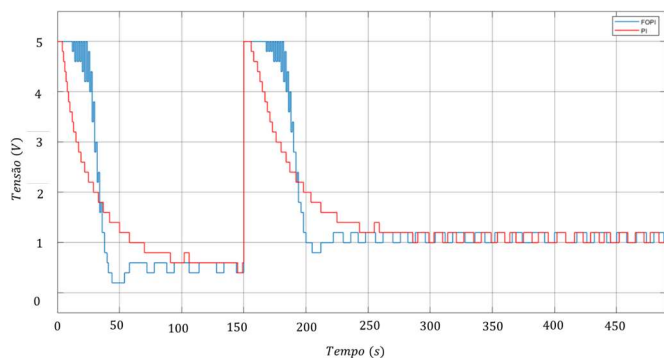


Figura 7 – Sinal de controle da resposta do degrau de 500 mm para FOPI (azul) e do PI (vermelho)

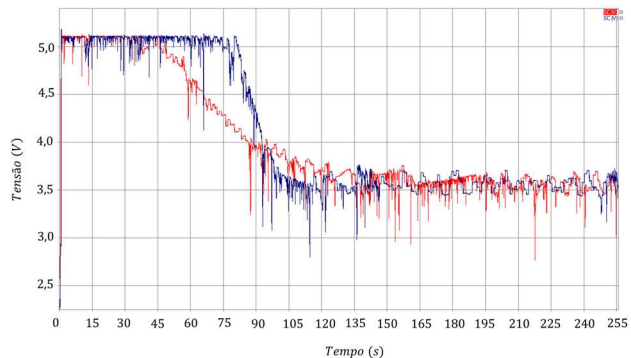


Figura 9 – Sinal de controle do FOPI (linha azul) e do PI (traçado vermelho)

Tabela 1 – Valores de desempenho da simulação

	$T_s$	$T_r$	sobressinal	ITAE
FOPI	81.8395	63.5779	0.4568%	$2.95 \times 10^5$
PI	136.1933	79.4155	0.0%	$4.1813 \times 10^5$

O FOPI apresenta desempenho superior ao PI, com destaque a acomodação mais rápida e um ITAE menor. Entretanto, pode-se observar pela figura 6 que o FOPI exige um esforço de controle maior que o PI.

## 8.2 Experimento

Implementando os coeficientes do PI e do FOPID ao CLP. Quatro experimentos foram feitos para ambos os controladores: Degrau de 500 mm, teste de rastreabilidade de referência e dois testes de perturbação (queda de energia no atuador e queda de energia na alimentação do sensor).

### 8.2.1 Degrau de 500 mm

Aplicando a referência de 500 mm através do comando digital do CLP para o FOPI e o PI, obteve-se os gráficos das Figura 8 e Figura 9, que são do FOPI (traçado azul escuro) e do PI (vermelho).

De fato, o controlador fracionário mostrou desempenho superior ao de ordem inteira. A tabela 2 mostra os valores de desempenho obtidos neste experimento, a qual comprova a superioridade do compensador de ordem não inteira em relação ao PI.

Tabela 2 – Valores de desempenho da implementação

	$T_s$	$T_r$	sobressinal	ITAE
FOPI	86.8318	66.8486	0%	$3.1518 \times 10^5$
PI	157.9805	83.9362	0%	$4.0969 \times 10^5$

### 8.2.2 Teste de rastreabilidade

Para verificar a capacidade de rastreabilidade dos controladores, programou-se no CLP, a partir da referência de 238 mm, os degraus: 300mm, 400 mm, 500 mm, 300 mm, 500mm. O intervalo de tempo entre cada degrau foi de 240 s, com o objetivo de garantir a estabilização do degrau antes de prosseguir para o próximo. Com isso, obteve-se o gráfico das Figuras 10 e 11. Nota-se, novamente, o desempenho superior do FOPI, o qual alcançou as respectivas referências com maior velocidade, conforme o alvo de projeto. Porém, vale evidenciar que o PI controlou a descida da água no tanque, enquanto que o FOPI aparenta não exercer esforço de controle. Entretanto, o gráfico da figura 11 mostra que o controlador aumenta a intensidade de controle conforme o nível (PV) se aproxima da referência de 300 mm, apesar de não influenciar

a descida. Pode-se inferir que esse comportamento é realizado para manter o desempenho da velocidade de acomodação.

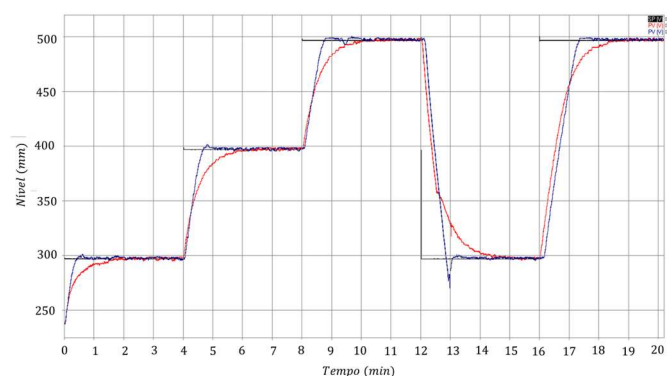


Figura 10 – Degrau de 300 mm, 400 mm, 500 mm, 300 e 500 mm novamente para o FOPI (linha azul) e para o PI (linha vermelha)

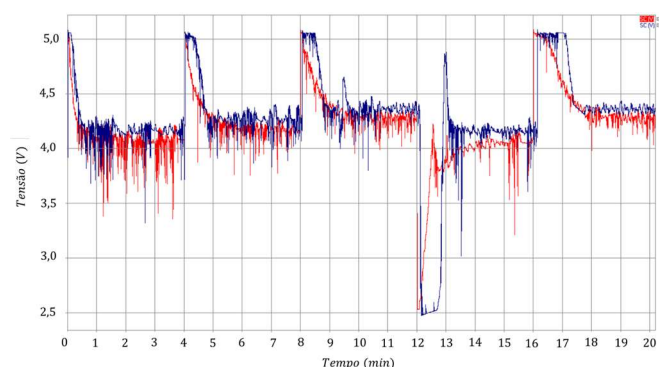


Figura 11 – Sinal de controle para o teste de degraus consecutivos para o FOPI (linha azul) e para o PI (linha vermelha)

### 8.2.3 Teste de queda de energia no atuador

Para simular uma perturbação no sistema de controle, programou-se o CLP para enviar 0 V de sinal para o inversor de frequência enquanto o algoritmo do controlador continua ativo na programação interna do CLP. Essas condições simulam uma queda de energia no atuador. A parada foi programada para acontecer no tempo de 420 s com 10 s de desligamento. O gráfico das Figuras 12 e 13 foram obtidos do experimento. Pelo gráfico da figura 12, observa-se que o FOPI recupera o nível (PV) para o *setpoint* (SP) mais rapidamente que o PI. A figura 13 mostra que as exigências de controle para o FOPI são relativamente mais abruptas que para o PI. Isso se deve a necessidade que o  $PI^\lambda$  possui de obedecer às suas características de projeto, o que inclui um comportamento mais “agressivo”, gerando um maior sobressinal, devido ao expoente fracionário (Monje et al., 2010).

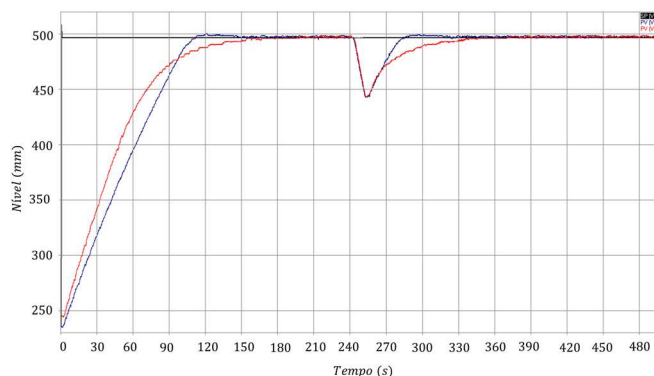


Figura 12 – Degrau de 500 mm com desligamento do inversor usando o FOPI (linha azul) e o PI (linha vermelha)

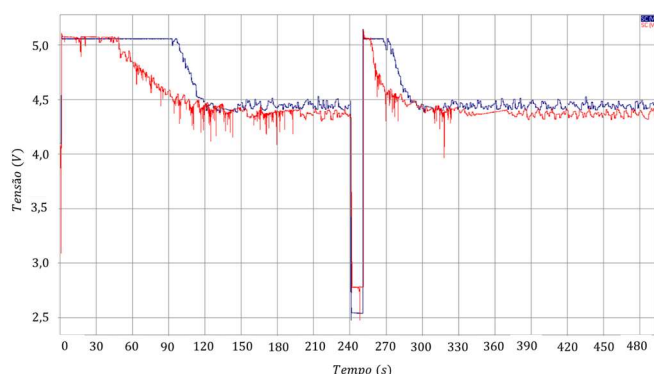


Figura 13 – Sinal de controle do para o FOPI (linha azul) e para o PI (linha vermelha)

### 8.2.4 Teste de queda de energia no sensor

O desligamento do sensor foi simulado através de um envio de sinal 0 para a leitura de nível (PV) na lógica interna do programador por um tempo de 10 s. Esta ação pode ser visualizada nas Figuras 14 e 15. A perda de sinal do sensor é mostrada na queda abrupta da curva PV no instante 240 s. O desligamento do sensor causou, de fato, uma perturbação no sistema de controle, mas pouco influenciou no nível PV. Ainda assim, observa-se o maior poder de controle do FOPI, recuperando o nível PV para o SP de 500 mm mais rapidamente que o PI, conforme pode ser verificado no gráfico da figura 14.

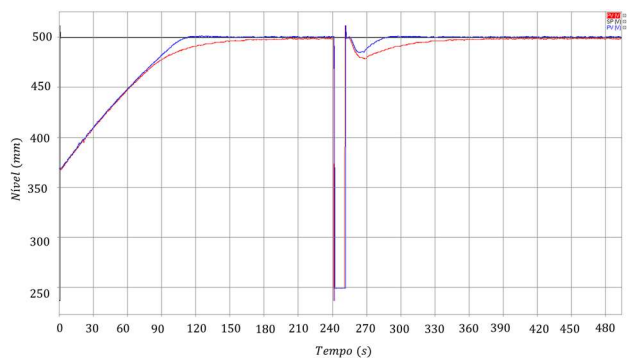


Figura 14 – Degrau de 500 mm com o desligamento do sensor com o FOPI (traçado azul) e usando o PI (linha vermelha)

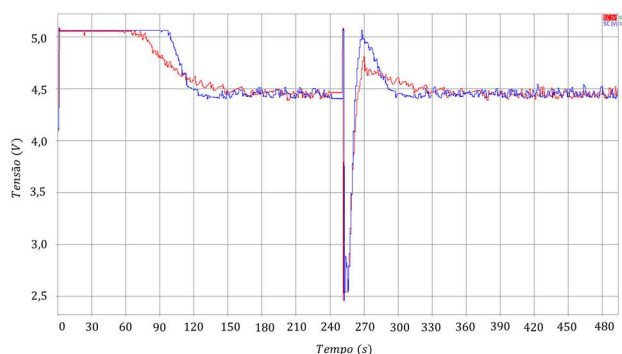


Figura 15 – Sinal de controle para o FOPI (linha azul) e para o PI (linha vermelha)

## 9. CONCLUSÕES

Neste trabalho um controlador tipo  $PI^\lambda$  foi projetado e implementado em uma bancada de nível. Um  $PI^\lambda$  foi usado porque o PI de ordem inteira mostrou ter melhor desempenho que um PID para a bancada de nível. Também, os testes realizados mostraram que, em relação ao clássico PI, o FOPI teve melhor desempenho de controle sobre a planta mediante a circunstâncias de trabalho, ou seja, operação normal ou uma situação eventual (como a queda de energia na rede elétrica com um rápido religamento através de um gerador ou mesmo recuperação da própria rede). A maior robustez em relação a distúrbios na entrada (desligamento do sensor) e na saída (desligamento do atuador – inversor) do controlador fracionário foi observada nesses experimentos, uma vez que a recuperação ocorreu de forma mais rápida que no PI. Com isso, este artigo procura contribuir para fomentar a iniciativa do uso de controladores fracionários na indústria.

## 10. REFERÊNCIAS

Alves, J.B. (2019) “Construção, modelagem e controle de uma bancada de nível.” *Monografia: Universidade Federal do Amazonas*. Amazonas.

El-Shafei, M.A.K., El-Hawwary, M.I. and Emmara, H.M. (2017) “Implementation of Fractional-Order PID Controller in an Industrial Distributed Control System,” in *14th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 713–718.

Franklin, G.F., David, J. and Abbas Emami-Naeini, P. (2013) *Sistemas de Controle para Engenharia*. 6<sup>o</sup> ed. Porto Alegre: Bookman.

Merrikh-Bayat, F., Mirebrahimi, N. and Khalili, M.R. (2015) “Discrete-time fractional-order PID controller: Definition, tuning, digital realization and some applications,” *International Journal of Control, Automation and Systems*, 13(1), pp. 81–90.

Monje, C.A. et al. (2004) “Proposals for Fractional  $PI^\lambda D^\mu$  tuning.” In the *First IFAC Symposium on Fractional Differentiation and its Applications*. p. 369–381.

Monje, C.A. et al. (2010) *Fractional-order Systems and Controls*. 1st edn. London: Springer.

Mystkowski, A. and Zolotas, A. (2018) “PLC-Based Discrete Fractional-Order Control Design for an Industrial-Oriented Water Tank Volume System with Input Delay,” *Springer*, 21(4), pp. 1005–1026.

Oprzedkiewicz, K.; Mitkowski, w.; Gawin, E. 2017. “The PLC Implementation of Fractional-Order Operator Using CFE Approximation”. *Springer International Publishing AG 2017*. pp. 22–33.

Petras. Ivo. 2012. “Tuning and Implementation Methods for Fractional-Order Controllers” *Springer*, 15(2), pp. 282–303.

Podlubny, I. (1994). “*Fractional Order Systems and Fractional-Order Controllers*”. Kosice - Slovakia.

Seborg, D.E. et al. (2012) *Process Dynamics and Control*. 3<sup>o</sup>. Santa Barbara - California: John Wiley & Sons, Inc.

dos Santos, Nikolas M. L. (2022) “Projeto e Implementação de um Controlador PID Fracionário a uma Bancada de Controle de Nível”. *Monografia: Universidade Federal do Amazonas-UFAM*. Manaus (AM).

Shah, P. and Agashe, S. (2016) “Review of fractional PID controller,” *Mechatronics*, 38(1), pp. 29–41.

da Silva, L.R. (2017) “*Controle de sistemas com atraso esaturação: estudo comparativo entre um GPC e um DTC com anti-windup*.” *Dissertação de Mestrado: Universidade Federal de Santa Catarina-UFSC*. Florianópolis (SC).

Tepljakov, A. et al. (2013) “Design and Implementation of Fractional-order PID Controllers for a Fluid Tank System,” *American Control Conference (ACC)*, 17 June

Tepljakov, A., Petlenkov, E. and Belikov, J. (2011) “FOMCON: a MATLAB Toolbox for Fractional-order System Identification and Control,” *International Journal of Microelectronics and Computer Science*, 2(2), pp. 51–62.

Valerio, Duarte. and da Costa, J.S. (2013) “An introduction to fractional control”. *1st edn. London: Institution of Engineering and Technology*.