



## **CONTROLADOR FUZZY OTIMIZADO POR PSO APLICADO A UMA MINI PLANTA INDUSTRIAL**

**Victor Santos Matos, victorlico.hp@gmail.com<sup>1</sup>**  
**Lucas Lima Carneiro, llc12.llc@gmail.com<sup>1</sup>**  
**Thiago de Almeida Ushikoshi, thiagoushikoshi@gmail.com<sup>1</sup>**  
**Thiago Pereira das Chagas, tpchagas@uesc.br<sup>1</sup>**

<sup>1</sup>Universidade Estadual de Santa Cruz, Laboratório de Mecatrônica - Campus Soane Nazaré de Andrade km 16 Rodovia Ilhéus-Itabuna, Salobrinho 45662-000 - Ilheus, BA – Brasil.

**Resumo.** *Esse artigo propõe a otimização de um controlador fuzzy para a aplicação em um sistema não linear tipo tanque alimentado por bomba. Por meio de simulações com o modelo dinâmico do sistema e um controlador fuzzy previamente construído, foi aplicado um algoritmo de otimização por enxame de partículas. O controlador otimizado apresentou desempenho robusto, com grande redução no esforço de controle e constância por toda a não linearidade do sistema.*

**Palavras chave:** *Otimização por enxame de partículas. Controlador Fuzzy. Controle de Nível. Não linear. Arduino.*

### **1. INTRODUÇÃO**

Na literatura, a aplicação de sistemas de controle *fuzzy* é extensa e abrange desde controle robótico autônomo e processos não-lineares até controle de processos industriais, incluindo o controle de nível (Ushikoshi, 2018; Zhou et al., 2016; Sabri e Al-mshat, 2015). Sabri e Al-mshat (2015) propôs um controlador *fuzzy* para eliminar o erro em regime permanente de um sistema de nível de água cujo desempenho supera o de um PID, Chabni et al. (2016) desenvolveu um sistema de baixo custo utilizando Arduino e lógica *fuzzy* para controlar uma réplica de planta de processos químicos e Kummar e Reddy (2016) propôs uma lógica *fuzzy* de controle para controle robusto do nível de dois tanques associados.

Ao longo das últimas décadas, os controladores *fuzzy* apresentaram algumas vantagens em relação à métodos clássicos de controle. Dentre essas vantagens, podemos destacar a sua intuitividade – visto que essa nova teoria foi concebida como uma forma de mimetizar o raciocínio humano; sua facilidade de implementação à sistemas complexos com dinâmicas avançadas e de ordem elevada, visto que os controladores *fuzzy* são capazes de desempenhar bem com informações quantitativas escassas e portanto são robustos; e por último, mas não menos importante, a possibilidade de se projetar inteiramente um controlador *fuzzy* sem possuir um modelo matemático da planta a ser operada (Lee, 1990). É válido ressaltar que não há necessidade de um modelo matemático da planta, mas esse pode ser útil para aperfeiçoar o ajuste do controlador através de simulações, por exemplo.

A otimização de um problema é o processo de encontrar a melhor solução para este, sendo necessário representa-lo de forma a mensurar a qualidade da solução (Burke et al., 2005). A otimização por enxame de partículas (PSO) é uma técnica de otimização comum na forma de um algoritmo heurístico baseado na teoria de enxame de partículas. (Eberhart and Kennedy, 1995). Esta técnica tem sido usada para otimização de controladores *Fuzzy* e outros controles aplicados a processos industriais demonstrando (Fang, 2008; Bassi, 2011; Song, 2007). Aqui, a otimização se baseia na função custo proposta com base no conhecimento no sistema.

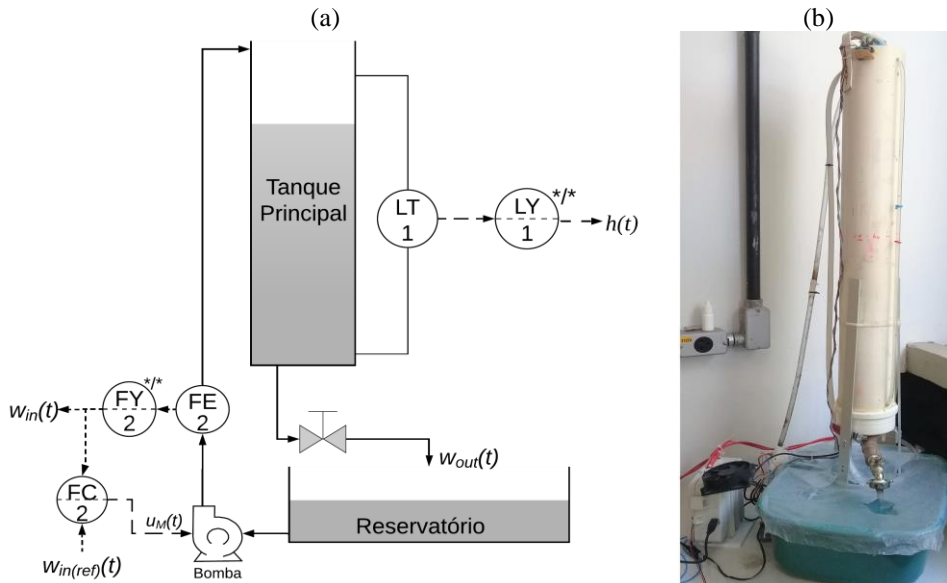
Este artigo é organizado conforme: na seção 2 é apresentada a descrição do sistema; na seção 3 o controlador *fuzzy* é caracterizado; na seção 4 é apresentada uma breve teoria do PSO; na seção 5 é apresentada a metodologia de otimização do controlador; na seção 6 os resultados e discussões levantadas; por último, seção 7 a conclusão das ideias principais.

### **2. DESCRIÇÃO DA PLANTA**

A planta industrial miniaturizada, representada em diagrama industrial (*piping and instrumentation diagram* - P&ID), Fig. 1(a), é formada pelo tanque principal, um tanque reservatório, um sistema medidor de nível por pressão diferencial (LT), uma bomba centrífuga, um medidor de vazão por encoder (FE), juntamente com um sistema de aquisição digital de sinais e filtragem (FY e LY) e circuitos de amplificação de potência (FC). Por fim, uma válvula mecânica para ajustar a vazão de saída.

Determina-se a vazão desejada  $w_{in(ref)}(t)$  para FC, que aplica sobre a bomba a tensão  $U_M(t)$ . A bomba conduz o líquido do reservatório para o tanque principal, o qual escoo através da válvula mecânica. Na saída da bomba mede-se sua vazão  $w_{in}(t)$  e sobre o tanque a altura do fluido acumulado  $h(t)$ .

Figura 1. (a) Diagrama P&ID da planta, (b) planta real



O sistema real pode ser visualizado na Fig. 1(b) e é construído com elementos de baixo custo como o tanque principal em tubo de PVC e a placa de prototipagem *Arduino Mega*, juntamente com os circuitos de amplificação e filtragem, como o sistema de aquisição e controle descritos.

Os principais componentes do sistema podem ser representados por modelos matemáticos em função das leis físicas que os governam. A dinâmica da bomba utilizada pode ser representada pelo modelo do circuito de armadura Eq.(1), da dinâmica do rotor Eq.(2) e da dinâmica da tubulação Eq.(3). Onde  $L_M$  é a indutância de armadura,  $R_M$  resistência de armadura,  $I_M(t)$  corrente de armadura,  $\Psi$  constante do motor,  $\omega(t)$  velocidade angular do rotor,  $U_M(t)$  tensão aplicada,  $J_B$  momento de inércia da bomba,  $M_{f0}$  coeficiente de atrito seco,  $\rho$  densidade do fluido,  $h_{th}$  coeficiente de coluna d'água,  $a_f$  característica do tubo,  $h_{nn}$  coeficiente de coluna d'água útil e  $h_{rr}$  atrito viscoso no tubo (Isermann and Münchhof, 2010).

$$\dot{I}_M(t) = \frac{-R_M I_M(t) - \Psi \omega(t) + U_M(t)}{L_M} \quad (1)$$

$$\dot{\omega}(t) = \frac{\Psi I_M(t) - M_{f0} - \rho g h_{th} \omega(t) w_{in}(t)}{J_B} \quad (2)$$

$$\dot{w}_{in}(t) = \frac{(h_{nn} - h_{rr}) w_{in}^2(t)}{a_f} \quad (3)$$

A dinâmica do nível no tanque principal, pode ser determinada pela relação das vazões entrada  $w_{in}(t)$  e saída  $w_{out}(t)$  e pela lei de conservação de massa, onde  $A_1$  representa a área da base do tanque principal.  $w_{out}(t)$  pode ser obtida da pelo princípio de Bernoulli, relacionando a área do orifício da válvula de saída  $A_2$  com a velocidade de escoamento do fluido e a aceleração da gravidade. Resultando na Eq.(4) (Dorf e Bishop, 2009).

$$\dot{h}(t) = \frac{w_{in}(t) - w_{out}(t)}{A_1} = \frac{w_{in}(t) - A_2 \sqrt{2g} \sqrt{h(t)}}{A_1} \quad (4)$$

Esses modelos descritos demonstram, de forma aproximada, os aspectos não lineares dos principais componentes do sistema. Matos e Chagas (2019, no prelo) os descrevem numericamente por técnicas experimentais, resultando na Eq.(5) e (6) que, respectivamente, representam a dinâmica linearizada da bomba - levando em consideração a presença de um controlador (FC) do tipo PI embarcado no próprio *Arduino*; e a dinâmica do nível, escrita com base na Eq.(4) e nos

características construtivas do sistema real.. Descreve-se também que o sistema é dotado de saturações nas principais grandezas,  $w_{in}(t) \in [w_{in,min}, w_{in,max}] = [0,24]cm^3/s$  e  $h(t) \in [h_{min}, h_{max}] = [0,60]cm$ .

$$\frac{W_{in}(s)}{W_{in(ref)}(s)} = \frac{28,16}{s^4 + 8,78s^3 + 20,18s^2 + 40,49s + 28,16} \quad (5)$$

$$\dot{h}(t) = 0,0138w_{in}(t) - 0,0375\sqrt{h(t)} \quad (6)$$

A placa *Arduino* utilizada foi programada para receber os sinais dos sensores e controlar os atuadores, além de estabelecer uma comunicação serial com o computador de forma a enviar para o mesmo as grandezas mensuradas e receber pelo protocolo serial as grandezas de controle para cada atuador. A mesma lógica de comunicação foi aplicada no software que será utilizado para monitorar e controlar o sistema, *Matlab*®.

### 3. CONTROLE FUZZY

Um sistema de controle *fuzzy* convencional pode ser fracionado em quatro estágios, sendo eles a fuzzificação, o mecanismo de inferência *fuzzy*, a base de conhecimento e a defuzzificação. O estágio de fuzzificação é composto por um conjunto de funções de pertinência que classificam as variáveis numéricas de entrada ( $x_1, x_2$ ) em conjuntos *fuzzy*. O mecanismo de inferência *fuzzy*, por sua vez, é responsável por reunir as informações dos estágio periféricos e aplicar as operações *fuzzy* definidas a fim de tomar decisões de controle. A base de conhecimento é um banco de dados que reúne as regras de inferência – do tipo SE-ENTÃO – que são previamente formuladas de acordo com a aplicação. Por fim, uma vez que todos o processo de raciocínio *fuzzy* é realizado no domínio *fuzzy*, é necessária uma interface de conversão que traduza os resultados *fuzzy* para valores do mundo real (Wang, 1997).

No presente trabalho, o controlador *fuzzy* é projetado para realizar o controle de nível de uma planta industrial miniaturizada. O sistema de controle a ser implementado possui o erro  $e(t)$  como entrada, que é a diferença entre o nível de referência  $h_{ref}(t)$  e o nível medido  $h(t)$ . A saída, por sua vez, é uma ação de controle a ser aplicada no único atuador disponível ao sistema: a vazão da bomba de circulação  $w_{in(ref)}(t)$ . Os universos de discurso das variáveis de entrada e saída são, respectivamente,  $-60 \leq e(t) \leq 60 cm$  e  $0 \leq w_{in(ref)}(t) \leq 24cm$ , respeitando-se a saturações do sistema.

Inicialmente, uma abordagem clássica será assumida, onde as funções de pertinência, regras de inferência e demais configurações do controlador serão definidas com base no ajuste manual de um especialista no sistema em em controladores *fuzzy* no processo em questão, sendo responsável por ditar as regras de inferência e, também, definindo o formato das funções de pertinência, de modo que o controlador tenha o melhor desempenho possível.

O principal objetivo desse trabalho é propor uma abordagem alternativa para ajuste do controlador *fuzzy* referido. Essa abordagem visa reduzir o trabalho manualmente exaustivo, e subjetivo do operador, e otimizar o desempenho do sistema proposto sobre diversos aspectos distintos, tais como erro e esforço de controle. Para isso, um algoritmo de otimização será aplicado visando definir, por meio de métodos estocásticos e metaheurísticos, o formato das funções de pertinência previamente definidas. Em seções posteriores a implementação de tal algoritmo será detalhada e um função de otimização será definida.

### 4. OTIMIZAÇÃO PSO

Otimização por enxame de partículas (*particle swarm optimization* - PSO) é um método de otimização estocástico construído de forma a simular comportamento de certos indivíduos em sociedade (Laskari et al., 2002). Ao contrário de métodos evolucionários como algoritmos genéticos, o PSO mimetiza o comportamento de um bando de pássaros procurando por comida e determina o indivíduo mais adequado, o que possui melhor valor na função custo, pela sua posição (Song e Gu, 2004). A ideia é explorar características de comportamento social ao invés de apenas habilidades individuais (Poli et al., 2007).

Neste algoritmo, o bando (*swarm*) é constituído por um número de partículas que possuem determinada velocidade (Song, 2004). As partículas são distribuídas no espaço de busca de um dado problema e cada uma avalia a função custo em sua posição atual. Então, elas ajustam seu “vôo” (direção e velocidade) de acordo com algum conhecimento prévio, como melhores posições em que estiveram, e com o conhecimento de um ou mais indivíduos do bando (Poli et al., 2007; Shy e Emberhart, 1998). Uma perturbação aleatória é adicionada no cálculo da próxima posição de forma semelhante às alterações randômicas ocorridas na natureza (Poli et al., 2007). Assim, o algoritmo é passível de encontrar uma solução ótima através da busca no espaço de soluções do problema (Poli, 2007; Song e Gu, 2004).

Segundo Ushikoshi et al. (2019, no prelo), as partículas são caracterizadas por sua posição  $x_i$ , melhor posição anterior  $p_i$  e velocidade  $v_i$ . Para balancear convergências de algoritmos globais e locais, é usada a modificação de peso inercial (Shi and Eberhart, 1998). Então, as posições das partículas são atualizadas iterativamente através de:

$$v_{i+1} = wv_i + c_1r_1(p_i - x_i) + c_2r_2(p_{ig} - x_i), \quad (7)$$

$$x_{i+1} = x_i + v_{i+1}, \quad (8)$$

onde  $c_1$  e  $c_2$  são constantes positivas,  $r_1$  e  $r_2$  são vetores de números aleatórios uniformemente distribuídos entre  $[0,1]$ ,  $p_{ig}$  é a melhor posição dentre todas as partículas e  $w$  é o peso inercial que pode ser uma constante positiva ou uma função positiva variante no tempo linear ou não linear.

## 5. METODOLOGIA

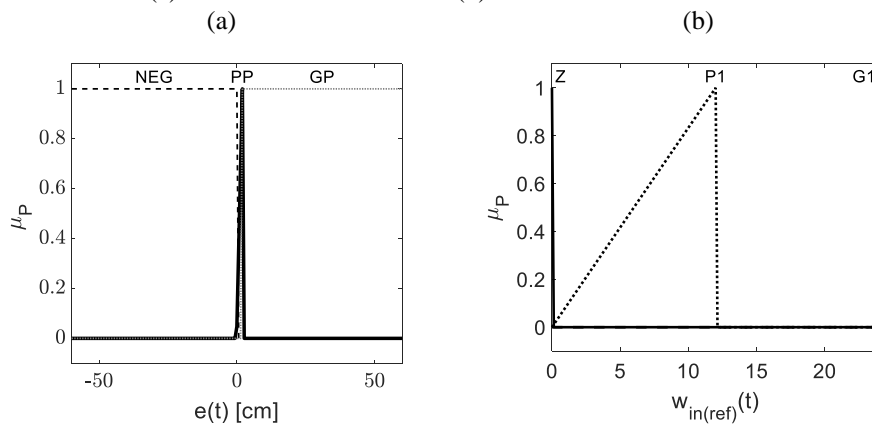
Neste artigo, o algoritmo PSO é utilizado na otimização de um Controlador *Fuzzy* aplicado no controle de nível de um processo industrial (mini planta industrial). O algoritmo é aplicado via software *Matlab* e o funcionamento do conjunto sistema/controlador é simulado em ambiente *Simulink*.

Primeiramente, um controlador foi proposto por um especialista no problema em questão de forma que os ajustes foram feitos manualmente e de forma exaustiva em busca de uma performance melhor. Posteriormente, o controlador proposto inicialmente serviu como base inicial de dados para a execução de um algoritmo de PSO, de modo que o formato das funções de pertinência fosse parcialmente refinado de acordo com uma dada referência. A seguir, as configurações do controlador ajustado manualmente são mostradas.

O controlador ajustado manualmente será referido como FL-C ao longo do texto. Como resultado do ajuste manual feito por um especialista no problema, foi decidido utilizar as operações *fuzzy* t-normas e s-normas convencionais, isto é, *min* e *max*, respectivamente. Além disso, optou-se por utilizar método de implicação *min*, método de agregação *max* e método de defuzzificação centro de gravidade (COG), também conhecido como centróide.

Para a variável de entrada de erro ( $e(t)$ ), foram criadas três funções de pertinência, com significados linguísticos Negativo (NEG), Pequeno Positivo (PP) e Grande Positivo (GP). Já para a variável de saída vazão de referência ( $w_{ref}$ ) foram criadas três funções de pertinência, com valores linguísticos Zero (Z), Pequeno (P1) e Grande (G1). Na Fig. 2 é possível observar o formato final das funções de pertinência de ambas as variáveis de entrada e saída.

Figura 2. Funções de pertinência da entrada erro ( $e(t)$ ) e da saída vazão de referência ( $w_{ref}(t)$ ) para o controlador FL-C. (a) Variável de entrada erro. (b) Variável de saída vazão de referência.



Por se tratar de um controlador SISO (*Single-Input Single-Output*, do inglês) e com poucas funções de pertinência (três na entrada e três na saída), não há necessidade de se desenvolver uma tabela de regras de inferência. O raciocínio por trás das regras de inferência é intuitivo do ponto de vista que a bomba de circulação injeta água no tanque. Sendo assim, as regras são: Se  $e(t)$  é NEG então  $w_{ref}(t)$  é Z; Se  $e(t)$  é PP então  $w_{ref}(t)$  é P1; e por fim, Se  $e(t)$  é GP então  $w_{ref}(t)$  é G1.

Para a simulação, uma referência de nível é criada para abranger dois tipos distintos de comandos para o tanque: aumento de nível de alta e baixa amplitude, respectivamente. Além disso, a referência foi pensada para uma faixa de operação mediana em relação a  $h_{max}$ .

Para a construção da função custo, o erro máximo  $e_{max}$  Eq.(9), o erro total  $e_{tot}$  Eq.(10) e a variação do esforço de controle total  $u_{tot}$  Eq.(11) foram calculados e normalizados para assumirem valor unitário para o controlador *fuzzy* original. Estes erros foram escolhidos para o sistema apresentar um menor tempo de subida, menor  $e_i$  quando em regime permanente e menor variação no esforço de controle.

$$e_{max} = \max|e_i| \quad (9)$$

$$e_{tot} = \sum e_i \quad (10)$$

$$u_{tot} = \sqrt{\frac{\sum \Delta u^2}{t_{tot}/T_s}} \quad (11)$$

Somando os erros apresentados anteriormente e ponderando-os para priorizar erro em regime permanente a saída da função custo é dada por:

$$J = 9w_1 e_{max} + 0,5w_2 e_{tot} + 0,5w_3 u_{tot} \quad (12)$$

onde  $w_1 = 1,07 \cdot 10^{-5}$ ,  $w_2 = 0,04$  e  $w_3 = 0,09$  são pesos usados para normalizar os erros.

Os parâmetros de entrada da função custo foram escolhidos como as coordenadas das MFs de entrada ( $e(t)$ ) e saída  $w_{in(ref)}$  do controlador *fuzzy*. A otimização foi realizada obedecendo aos limites do universo de discurso do controlador e o valor 50 foi escolhido para o número máximo de iterações assim como para o número de indivíduos no enxame.

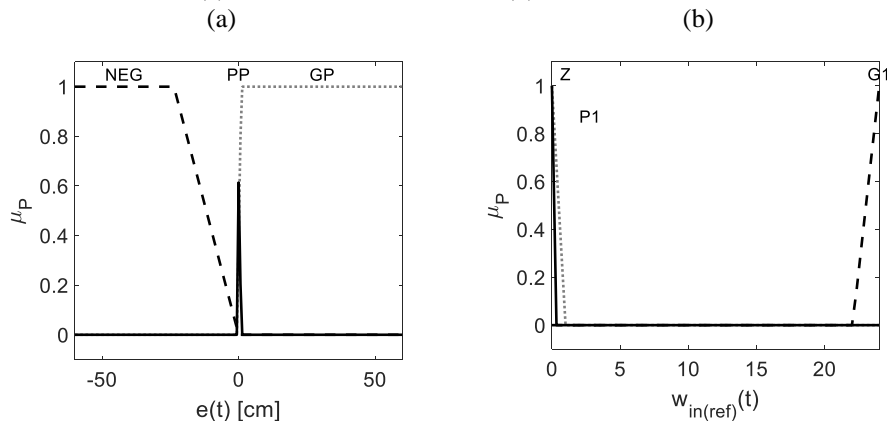
Realizou-se a aplicação dos mesmos teoricamente (em simulação) e na planta real, usando a mesma referência da otimização, para fins de comparação. Além disso aplicou-se uma nova referência, com cada controlador diretamente no sistema para fins de análise de robustez em relação a não linearidade do nível.

## 6. RESULTADOS E DISCUSSÃO

Nesta seção os resultados obtidos no escopo desse trabalho são apresentados. Primeiramente, é apresentada a configuração final do controlador *fuzzy* otimizado. Em seguida, os controladores são aplicados ao sistema e seu desempenho é avaliado sob os termos de uma função custo previamente desenvolvida, tanto em ambiente de simulações computacionais quanto em aplicação prática a um sistema físico construído.

O controlador ajustado via otimização PSO será referido ao longo do texto como PSO-FL-C. As configurações do PSO-FL-C foram mantidas iguais às dos FL-C, sejam as operações *fuzzy*, implicação, agregação, método de defuzzificação, regras de inferência e até mesmo a quantidade de funções de pertinência. No entanto, o formato das funções de pertinência foi alterado pelo algoritmo de otimização PSO e na Fig. 3 é possível observar o formato final dessas funções das variáveis de entrada e saída após a convergência do algoritmo de otimização.

Figura 3. Funções de pertinência da entrada erro ( $e(t)$ ) e da saída vazão de referência ( $w_{ref}(t)$ ) para o controlador PSO-FL-C. (a) Variável de entrada erro. (b) Variável de saída vazão de referência.



Comparando-se a Fig. 3 com a Fig. 2, pode-se observar que a otimização das funções de pertinência do  $e(t)$  gerou uma leve mudança em PP e GP e uma perceptível mudança em NEG, deslocando-a para a esquerda, implicando em um controlador mais sensível a erros ligeiramente positivos – nível levemente abaixo da referência. Em relação as funções de pertinência de  $w_{in(ref)}(t)$ , houve mudanças acentuadas em relação a P1, que foi estreitada e aproximada de Z, deixando-as concentradas em extremos, capazes de gerar um modelo de resposta passiva e outro agressiva.

Como resultado da aplicação simulada no sistema, Fig. 4 e 5, demonstra-se, respectivamente, a dinâmica do nível para ambos os controladores em relação a referência projetada; a vazão determinada por cada controlador e o valor real da mesma aplicada pela bomba. As Fig. 6 e 7 demonstram a aplicação na planta real seguindo a mesma lógica.

Em ambas as aplicações, os controladores otimizados demonstraram superioridade em relação ao de ajuste manual, em relação ao nível Fig. 4 e 6, em simulação e prática o controlador otimizado teve erro em regime reduzido em relação ao de ajuste manual, que além de apresentar grande erro, apresentou também característica oscilatória, que pode ser entendida por meio das Fig. 5 e 7, nas quais pode-se ver esforços de controle de característica chaveada, fazendo a bomba trabalhar em seus extremos.

O controlador otimizado apresentou redução drástica no esforço de controle sobre o sistema, sendo que na prática, Fig. 7, é possível visualizar um leve ruído nesse sinal gerado, porém este é pouco visualizado na dinâmica de nível,

indicando a capacidade do controlador de suavização da dinâmica do sistema. Esses resultados apontam para a eficácia e robustez do controlador *fuzzy* sobre essa aplicação não linear, uma vez que a dinâmica na aplicação prática foi muito similar a simulada, ainda que a otimização tenha sido feita com modelos aproximados do sistema.

Figura 4. Nível do tanque para controladores com ajuste manual e otimizado por simulação

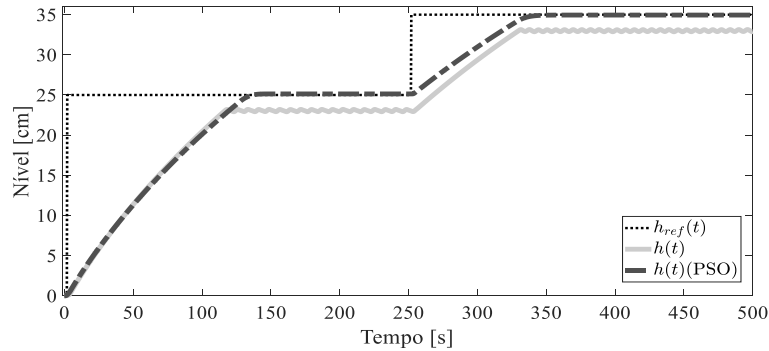


Figura 5. Esforço de controle e vazão da bomba para ambos controladores por simulação

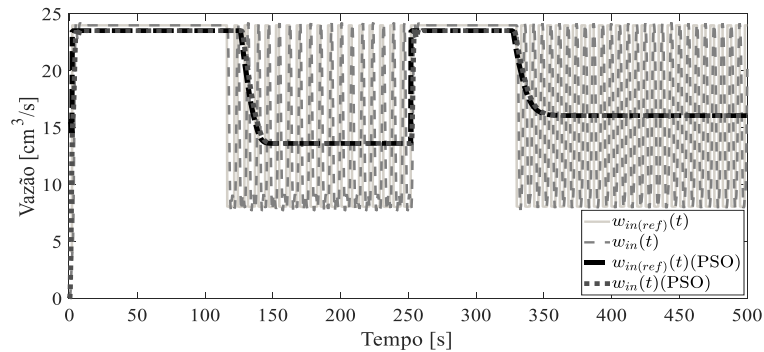


Figura 6. Nível do tanque para controladores com ajuste manual e otimizado aplicados à planta

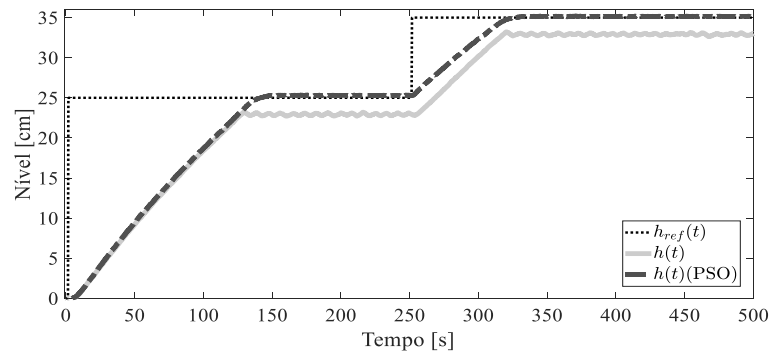
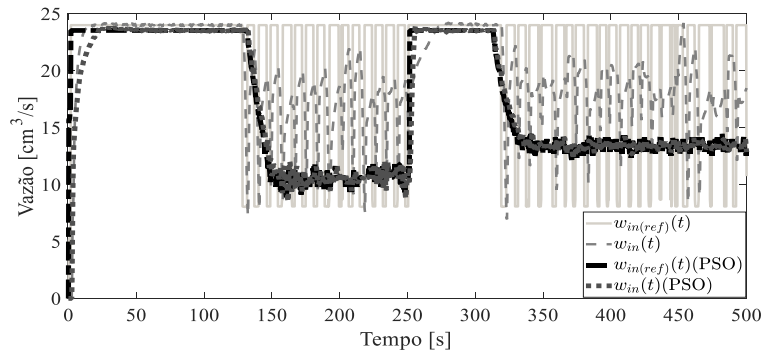


Figura 7. Esforço de controle e vazão da bomba para ambos controladores aplicados à planta



A Tabela 1 mostra o custo e seus componentes de cada experimento avaliado. Pode-se inferir que o processo de otimização se mostra muito eficiente na redução do custo total, seja na simulação como na prática. O principal componente minimizado foi  $u_{tot}$ , confirmando a já indicada redução do esforço de controle. Por essa variável é possível

perceber que na aplicação prática o sistema tem dinâmica com considerável diferença ao que foi estimado por seus modelos Eq.(5) e (6), reafirmando a presença de erros de modelagem e interferências externas. Uma observação a ser feita é que os valores de  $u_{tot}$  calculados para o sistema real são aproximados por derivação numérica em relação ao período de amostragem da planta, o que intensifica erros de estimação.

Tabela 1. Custo e seus componentes para cada experimento realizado

Experimento	$e_{max}$	$e_{tot}$	$u_{tot}$	$J$
Teórico não otimizado	$2,4967 \cdot 10^1$	$9,4133 \cdot 10^4$	$12,480 \cdot 10^0$	$10,182 \cdot 10^0$
Teórico otimizado	$2,4877 \cdot 10^1$	$8,4296 \cdot 10^4$	$1,1010 \cdot 10^0$	$8,7213 \cdot 10^0$
Prático não otimizado	$2,5070 \cdot 10^1$	$11,4840 \cdot 10^4$	$597,87 \cdot 10^0$	$38,142 \cdot 10^0$
Prático otimizado	$2,5120 \cdot 10^1$	$10,0030 \cdot 10^4$	$98,3980 \cdot 10^0$	$14,565 \cdot 10^0$

As Fig. 8 e 9 demonstram a aplicação da referência proposta para teste a robustez dos controladores. Essa referência foi projetada com alguns tipos de entrada e percorrendo todo o tanque de forma a permitir a verificar a capacidade do controlador de atuar fora do ponto de operação a qual foi otimizado. Observando-se a resposta dos controladores para essa referência, a versão otimizada é, claramente, superior à versão manual, uma vez que, a primeira responde prontamente seguir a referência por quase toda a aplicação, enquanto a segunda além de apresentar alto erro em regime – que cresce nas regiões mais altas (portanto, de maior não linearidade), apresenta oscilação visível. Pela observação dos esforços de controle e vazões reais, acentua-se a capacidade da versão otimizada, que indicando a robustez dada ao controlador pelo ajuste otimizado por PSO.

Figura 8. Medida do nível do tanque para teste de robustez

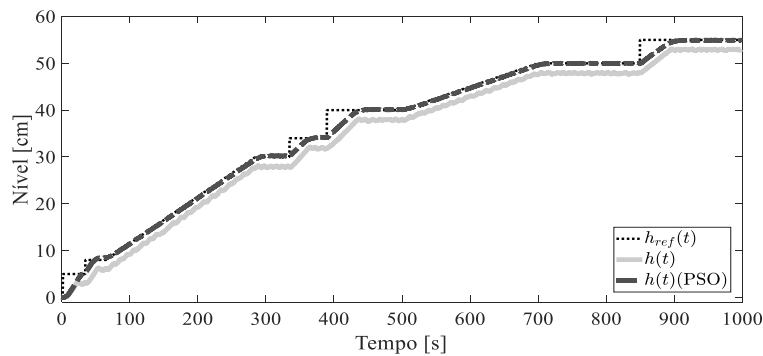
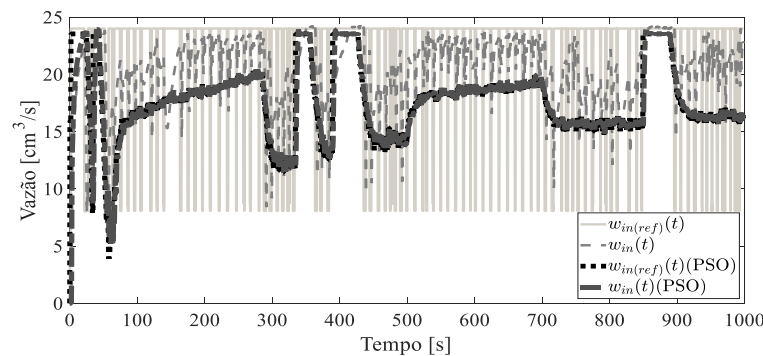


Figura 9. Esforço de controle e vazão da bomba para teste de robustez



## 7. CONCLUSÃO

A planta miniaturizada aqui descrita corresponde a um elemento comum em processos industriais, sistema tanque-bomba. Sua não linearidade indica o uso de controladores mais robustos e capazes de não depender diretamente de um modelo perfeito do sistema. Os controladores *fuzzy* aqui descritos demonstram capacidade de lidar com esse tipo de contexto. Chamando-se atenção para a versão otimizada, que responde de forma completa para todo o universo do sistema.

A superioridade do ajuste otimizado é confirmada pela capacidade dual de resposta do controlador: agressiva e, também, suave, que de forma ótima, foi capaz de impor esforços de controle mínimos, porém efetivos, mesmo que otimizado com uso de modelos do sistema pouco acurados.

A capacidade de manter um erro em regime baixo, ainda que para entradas rampa, indica possível correlação com controle de dupla ação integral, mesmo que não conscientemente incluso na formulação da função custo.

Todos os fatores culminam para a confirmação da robustez agregada ao controle com o ajuste por PSO.

Trabalhos futuros podem incluir a utilização do PSO como otimizador do modelo do sistema e de ajustes de outros controladores comumente associados ao contexto de controle de processos industriais.

## 8. REFERÊNCIAS

- Burke, E.K., Kendall, G., et al. (2005). *Search Methodologies*. Springer.
- Chabni, F., Taleb, R., Benbouali, A., & Bouthiba, M. A. (2016). "The application of fuzzy control in water tank level using Arduino". In *International Journal of Advanced Computer Science and Applications (IJACSA)*, 7(4), 261-265.
- Bassi, S. J., Mishra, M. K., & Omizegba, E. E. (2011). "Automatic tuning of proportional-integral-derivative (PID) controller using particle swarm optimization (PSO) algorithm". In *International Journal of Artificial Intelligence & Applications*, 2(4), 25.
- Dorf, R. C., & Bishop, R. H. (2009). *Sistemas de controle modernos*, 11a edição. Rio de Janeiro, LTC.
- Fang, G., Kwok, N. M., & Ha, Q. (2008, December). "Automatic fuzzy membership function tuning using the particle swarm optimization". In: *2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application (Vol. 2, pp. 324-328)*. IEEE.
- Isermann, R., & Münchhof, M. (2010). *Identification of dynamic systems: an introduction with applications*. Springer Science & Business Media.
- Kumar, K. A., & Reddy, G. P. (2016). "A Robust Fuzzy Logic Control of Two Tanks Liquid Level Process". In *International Journal of Advanced Engineering, Management and Science*, 2(5).
- Laskari, E. C., Parsopoulos, K. E., & Vrahatis, M. N. (2002, May). "Particle swarm optimization for integer programming". In: *Proceedings of the 2002 Congress on Evolutionary Computation*. CEC'02 (Cat. No. 02TH8600) (Vol. 2, pp. 1582-1587). IEEE.
- Lee, C. C. (1990). "Fuzzy logic in control systems: fuzzy logic controller". In *II. IEEE Transactions on systems, man, and cybernetics*, 20(2), 419-435.
- Matos, V. S. e Chagas, T. P. "A Miniaturized Industrial Plant for Educational Purpose in Industrial Control". In: *14º SBAI - Simpósio Brasileiro de Automação Inteligente*, 2019, Ouro Preto. 14º Simpósio Brasileiro de Automação Inteligente, 2019. No prelo.
- Poli, R., Kennedy, J., & Blackwell, T. (2007). *Particle swarm optimization*. *Swarm intelligence*, 1(1), 33-57.
- Sabri, L. A., & Al-mshat, H. A. (2015). "Implementation of Fuzzy and PID Controller to Water Level System using LabView". In *International Journal of Computer Applications*, 116(11).
- Shi, Y., & Eberhart, R. (1998, May). "A modified particle swarm optimizer". In: *1998 IEEE international conference on evolutionary computation proceedings*. IEEE world congress on computational intelligence (Cat. No. 98TH8360)(pp. 69-73). IEEE.
- Song, M. P., & Gu, G. C. (2004, August). "Research on particle swarm optimization: a review". In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 04EX826) (Vol. 4, pp. 2236-2241)*. IEEE.
- Song, Y., Chen, Z., & Yuan, Z. (2007). "New chaotic PSO-based neural network predictive control for nonlinear process". In *IEEE transactions on neural networks*, 18(2), 595-601.
- Ushikoshi, T. A., Carneiro, L. L., Coutinho, P. H. e Chagas, T. P. "Fuzzy Maneuvering Controller Optimization using an Optimum Differential-Drive Mobile Robot Model". In: *14º SBAI - Simpósio Brasileiro de Automação Inteligente*, 2019, Ouro Preto. 14º Simpósio Brasileiro de Automação Inteligente, 2019. No prelo.
- Ushikoshi, T. D. A., Peixoto, K. P., Souto, F. H., Thiago, P., & Schnitman, L. (2018, July). "Fuzzy Maneuvering Controller applied to a Dynamic Model of a Differential Drive Mobile Robot". In *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE) (pp. 1-8)*. IEEE.
- Wang, L. X.. (1997). *A course in fuzzy systems and control (Vol. 2)*. Upper Saddle River, NJ: Prentice Hall PTR.
- Zhou, Q., Li, H., Wu, C., Wang, L., & Ahn, C. K. (2016). "Adaptive fuzzy control of nonlinear systems with unmodeled dynamics and input saturation using small-gain approach". In *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(8), 1979-1989.

## 9. RESPONSABILIDADE PELAS INFORMAÇÕES

Os autores são os únicos responsáveis pelas informações incluídas neste trabalho.