

Relatório Unidade 2

.Gustavo Rossi

.Humberto Gama de Carvalho Neto

.Victor Araújo Petch

.Luiz Felipe Santana

.William da Cunha Ribeiro

■ **RESUMO** Este trabalho apresenta quatro diferentes configurações de controladores Fuzzy, incluindo os modelos Mandani e Tagaki-Sugeno-Kang (TSK), com diferentes entradas. Esses controladores são aplicados em um sistema de tanques acoplados Quanser. Através do uso de uma referência constante e uma referência senoidal, foi possível avaliar o desempenho dos modelos propostos, analisando os parâmetros de saída, como *overshoot* e tempo de acomodação. As sintonias dos controladores se dão através de algoritmos genéticos, diversificados ao longo dos experimentos.

Introdução

Apresentado em meados da década de 60 por Lotfi A. Zadeh, o conceito de conjuntos nebulosos introduziu uma nova maneira de interpretar e visualizar problemas no que tange sistemas de controle. Mais adiante todo esse ramo foi ganhando corpo com a introdução de novos conceitos para tomada de decisão partindo de conjuntos nebulosos e principalmente com o desenvolvimento do primeiro controlador *Fuzzy* para sistemas reais.

A partir daí, conforme se percebeu a facilidade de implementação e sua possível aplicação onde o controle clássico tinha bastante impedimento, como em aplicações onde não há modelagem matemática do sistema, as aplicações com controladores *Fuzzy* foram se popularizando cada vez mais. Atualmente a lógica nebulosa é uma tecnologia bastante difundida e extrema-

mente útil no desenvolvimento de sistemas para controle de processos complexos. Sua utilização permite o tratamento, com alto rigor matemático, de informações imprecisas definida por meios de termos linguísticos.

Este trabalho explorará as características de dois tipos de controlador *Fuzzy* em cinco configurações diferentes em uma aplicação de controle a fim de se comprar as melhores implementações.

Modelagem do Problema

Nesse experimento foi utilizado o sistema de tanques acoplados da empresa Quanser projetado em associação com os Professores Karl Åström e Karl Henrik Johansson, que consiste em uma única bomba com dois tanques. Cada tanque é instrumentado com um sensor de pressão para medir o nível da água. A bomba leva a água da bacia inferior até o topo do sistema. Dependendo de como as válvulas de saída são configuradas, a água flui para o tanque superior, tanque inferior ou ambos.

Para essa experiência a configuração dos tanques obedece a imagem abaixo. Nessa montagem a bomba de água alimenta o tanque 1 fluindo para o tanque 2 e o controlador regula o nível de água no tanque 2. Haverá, ainda, o controle do nível de água nos dois tanques.

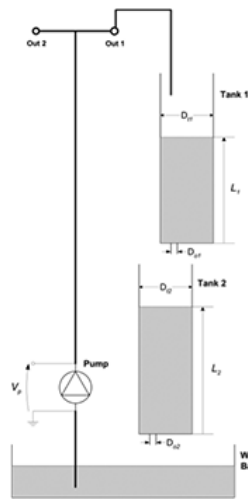


Figura 1. Configuração dos Tanques acoplados
Quanser Fonte: Tech Sim [10]

Fuzzy

Na teoria clássica, um elemento x do universo em discurso pertence ou não pertence a um dado conjunto, enquanto que na teoria dos conjuntos nebulosos esse elemento poderá pertencer parcialmente a um conjunto. Um conjunto nebuloso A é definido por uma função de pertinência μ que varia de zero a um dentro deste universo U - intervalo numérico de todos os possíveis valores que uma variável pode assumir [1].

Existem na literatura diversas formas de função de pertinência, em destaque a trapezoidal e a triangular, nos quais os seus parâmetros são determinados através da experiência que o especialista tem em controlar a planta. As funções de pertinências tem o objetivo de fazer a correspondência de um valor ou uma variável linguística em conjuntos Fuzzy. A variável linguística por sua vez é uma variável cujos valores são nomes de conjuntos Fuzzy, definida em um quintuplo $(x, A(x), U, G, M)$, em que x é o nome da variável; $A(x)$ é o conjunto de termos, isto é, o conjunto de nomes de valores linguísticos de x ; U é o universo do discurso; G é a gramática para gerar os nomes e M é um conjunto de regras semânticas para associar cada x ao seu significado [2].

Um controlador Fuzzy é composto por uma interface de “Fuzzyficação”, uma base de dados e de regras (base de conhecimento) e uma interface

de “deFuzzyficação”, [1]. Na literatura, outras variações aparecem a depender do objetivo do projeto.

A interface de Fuzzyficação” é responsável por mapear os valores reais de entrada para valores de domínio Fuzzy, estes valores mapeados são transformados em variáveis linguísticas definidas pelas funções de pertinência. A base de conhecimento é constituída por uma base de dados e uma base de regras, contendo todo o conhecimento exigido para controlar o sistema. Os dados e regras são representados em uma forma linguística de fácil entendimento, a partir de premissas do tipo: Se <condição> Então <conclusão>. A base de dados armazena as definições das funções de pertinência e a discretização e normalização dos universos de discurso, enquanto que a base de regra é formada por operadores de implicação, compondo as regras que representam o conhecimento representado no sistema Fuzzy.

O procedimento de inferência por sua vez consiste de verificar o grau de compatibilidade entre os fatos e as cláusulas nas premissas das regras; determinar o grau de compatibilidade total da premissa de cada regra; determinar o valor da conclusão em função do grau de compatibilidade da regra com os dados e a ação e controle constante na conclusão; e por fim agrega os valores obtidos como conclusão nas várias regras, obtendo-se uma ação de controle global [3].

A literatura apresenta diversos modelos de sistemas Fuzzy, classificados, basicamente, como os modelos clássicos, destacando-se o modelo de Mamdani [8], e de interpolação, destacando-se o modelo de Tagaki-Sugeno-Kang (TSK) [5]. Cada modelo Fuzzy apresentará uma abordagem para a representação das premissas e uma representação das ações de controle.

Fuzzy Tagaki-Sugeno-Kang

No modelo TSK todos os termos nebulosos da premissa devem ser funções monotônicas e o consequentes de cada regra é representado por uma função das variáveis de entrada ($f_i(x_1, x_2, \dots, x_n)$) que para cada uma das regras, obtém-se um peso correspondente (ω_i). Nos modelos de sistemas Fuzzy de interpolação, não existe a interface de “DeFuzzyficação”, a saída é obtida de forma direta e para o modelo de TSK

é, comumente, definida por uma média ponderada das saídas geradas por cada uma das regras [9],

$$y = \frac{\sum_{i=1}^k \omega_i f_i(x_1, x_2, \dots, x_n)}{\sum_{i=1}^k \omega_i}, \quad (1)$$

sendo, y a saída do controlador, x_1, x_2, \dots, x_n as entradas, k o número de regras e ω_i o resultado da inferência realizada para a i -ésima regra, com base nos valores gerados pela interface de Fuzzyficação para as entradas x_1, x_2, \dots, x_n , e $f_i(\cdot)$ a i -ésima Função Sugeno de Saída que é definida por,

$$f_i(x_1, \dots, x_n) = c_{0,i} + c_{1,i}x_1 + \dots + c_{n,i}x_n, \quad (2)$$

em que cada c_k é uma constante.

Fuzzy Mandani

Ebrahim Mamdani propôs na década de 70 um método de inferência Fuzzy que se tornou por muitos anos um padrão para a utilização dos conceitos da lógica Fuzzy em processamento de conhecimento. As regras de produção em um modelo de Mamdani possuem relações Fuzzy tanto nos termos antecedentes quanto nos termos consequentes. Diferente do modelo TSK, as saídas das inferências do modelo Mamdani precisam passar por um processo de "DeFuzzyficação". Esse processo muitas vezes é definida analisando a média dos máximos ou o centro de massa da função de pertinência de saída.

Algoritmo Genético

Os algoritmos genéticos compreendem uma família de modelos computacionais inspirados na teoria evolucionária de Charles Darwin. Desenvolvida por John Henry Holland em 1975, esses algoritmos normalmente são usados como otimizadores de funções, agrupando os dados em estruturas semelhantes a cromossomos e realizando operações de seleção, reprodução e mutação semelhante ao que é visto na natureza. Uma implementação de um algoritmo genético começa com uma população aleatória de cromossomos. Essas estruturas geralmente são vetores contendo parâmetros a serem otimizados. Tendo a primeira população criada, geralmente os próximos passos compreendem a avaliação dos indivíduos e a construção da próxima geração.

Este processo se repete até atingir algum valor de erro ou geração ou até o desejo do usuário. Para a avaliação dos cromossomos, utiliza-se funções de custo para calcular o erro gerado por cada indivíduo. A função de custo utilizada depende do problema a ser otimizado e possui bastante impacto na eficiência do algoritmo. Em geral, a construção da próxima geração depende da escolha dos melhores indivíduos para que ocorra a reprodução. Essa escolha é chamada de taxa de elitismo e ela define qual a porcentagem de indivíduos que serão gerados tendo em mente apenas os melhores da geração passada. Para realizar o crossover (termo inglês para cruzamento), é necessário a escolha dos indivíduos que irão reproduzir e qual será o método de mesclagem de cromossomo. Ademais, ainda é possível realizar a mutação do cromossomo de algum indivíduo com base em uma porcentagem, geralmente entre 0 e 10%. Para se ter uma otimização eficiente, é interessante gerar indivíduos diferentes e analisar as funções de custo, reprodução e mutação para que a melhoria dos cromossomos atinja um ótimo local satisfatório ou até um ótimo global.

Sintonia dos controladores

O presente trabalho apresentará cinco propostas de sintonia de controlares Fuzzy (dois controladores TSK e três Mandani) feitos pelo algoritmo genético, em que cada controlador será distinto de seu semelhante em virtude dos tipos e parâmetros utilizados no projeto das funções de pertinência de entrada e de saída, ressaltando que dois dos controlares Mamdani terão 2 entradas (erro e derivada do erro) e o terceiro contará também com o nível do tanque 1 como terceira entrada.

Controlador TSK com Funções de Pertinência Triangular e Trapezoidal (TT)

Afim de utilizar o algoritmo genético para a sintonia do controlador Fuzzy TSK, foi construído um indivíduo dotado das seguintes características: parâmetros das funções de pertinência da entrada erro e da derivada do erro; as constantes utilizadas para o cálculo da função Sugeno

de saída e um conjunto de regras que relaciona as entradas e as saídas do controlador.

O passo seguinte foi limitar o espaço de busca trabalhado com base no conhecimento já adquirido de trabalhos anteriores e baseado em limitações físicas da planta. Visto isso, foi fixado que os parâmetros da função da entrada erro estivessem entre -30 e 30 centímetros, da função derivada do erro estivessem entre -1 e 1 centímetros por segundo, e a saída estivesse entre -4 e 4 volts. Todos os valores trabalhados são números reais. Posteriormente, definiu-se também parâmetros próprios do algoritmo genético: número de indivíduos (20), número de geração (100), taxa de mutação (1% por indivíduo), taxa de *crossover* (80%). Definidas as características do controlador Fuzzy e seus limites, a primeira geração de indivíduos foi aleatoriamente gerada. Os parâmetros do controlador foram atualizados e o custo de cada indivíduo por geração foi calculado.

$$Custo_{TT} = IEA + ITEA + IG \quad (3)$$

A função custo utilizada é composta por um somatório entre o IEA, ITEA e o Índice de Goodhart (com coeficientes $\alpha_1 = 2$, $\alpha_2 = 0.75$, $\alpha_3 = 0.25$).

Em seguida, com base nos custos, 2 indivíduos são selecionados para gerar 1 filho. O método de seleção utilizado foi o da roleta, o *crossover* foi feito através do método BLX- α [6] e a mutação usou o método *Creep* [7]. Por fim, é feito um teste de validade em cada indivíduo a fim de garantir que não existe incongruências que o incapacitem de atuar como um controlador Fuzzy. A quantidade de filhos gerados vai depender da taxa de *crossover*, ao atingir a taxa de *crossover* os demais indivíduos da próxima geração serão clonados da geração passada. A seleção dos indivíduos a serem clonados é feito pelo método de elitismo, que simplesmente seleciona os melhores indivíduos da geração passada.

A Fig. 2 apresenta as funções de pertinências criadas para a entrada *erro* e a Fig. 3 apresenta as funções de pertinências criadas para a entrada *dterro*.

As Funções Sugeno de Saída quantificada pela Tal. 1, em que os o parâmetro c_1 multiplica o

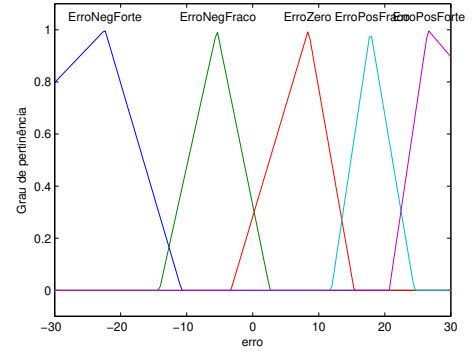


Figura 2. Funções de Pertinência Entrada 1 (TT)

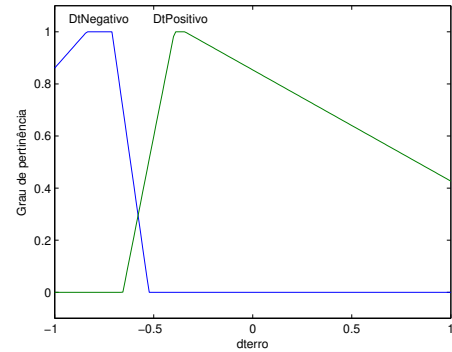


Figura 3. Funções de Pertinência Entrada 2 (TT)

valor da entrada *erro* e o parâmetro c_2 multiplica o valor da entrada *dterro*, enquanto que o coeficiente c_0 é zerado pois não deseja-se obter valores de *offset*.

Tabela 1. Funções Sugeno de Saída (TT)

Funções de Saída	c_2	c_1	c_0
mf1	0.01742	0.4498	0
mf2	0.02097	0.4559	0
mf3	0.02851	0.511	0
mf4	0.0154	0.636	0
mf5	0.02823	0.5944	0
mf6	0.04188	0.4116	0
mf7	0.02011	0.4967	0
mf8	0.03156	0.5929	0
mf9	0.03178	0.4574	0
mf10	0.04356	0.6095	0

Por fim, a Fuzzy Associative Matrix (FAM) conforme demonstrado na Tab. 2.

Controlador TSK com Funções Gaussiana e Trapezoidal (GT)

Com o intuito de realizar a sintonia do controlador Fuzzy TSK utilizando funções Gaussiana e

Tabela 2. Matriz de Associações Fuzzy (TT)

Regras	Derivada do Erro	
	DtNegativo	DtPositivo
Erro Negativo Forte	mf1	mf2
Erro Negativo Fraco	mf3	mf4
Erro Zero	mf5	mf6
Erro Positivo Forte	mf7	mf8
Erro Positivo Fraco	mf9	mf10

Trapezoidal (GT), construí um indivíduo com as seguintes características: funções de pertinência de entrada erro e derivada do erro, contante utilizada no Sugeno e o conjunto de regras para relacionar as entradas com as saídas do controlador.

Conhecendo o sistema, o primeiro passo foi estabelecer limitações ao meu espaço de trabalho. A primeira definição foi que, meu sistema iria trabalhar com números reais. Em seguida foi fixado que os os parâmetros de entrada erro estivessem entre os valores de -30 a 30 cm. Já na função derivada do erro, foi estabelecido de operação entre -1 a 1 cm/s e por fim minha saída trabalhasse entre 0 e 4 volts.

Posteriormente foi definido os parâmetros próprios do algoritmo genético, como: número de indivíduos como 10, número de gerações como 50, abertura da gaussiana como 10, taxa de *crossover* em 80% e elitismo em 20%.

Com os parâmetros do algoritmo definidos e as características do fuzzy, a primeira geração dos indivíduos foi dada de forma aleatória. Nessa etapa também foi realizada a atualização dos parâmetros do meu controlador e o cálculo do custo de cada indivíduo por cada geração.

Para realizar o cálculo do custo utilizei da equação 4, um somatório IEA e o índice de Goodhart, cujo os coeficientes utilizados foram $\alpha_1 = 1.3$, $\alpha_2 = 1$, $\alpha_3 = 1$.

$$Custo_{GT} = IEA + IG \quad (4)$$

Após o cálculo do custo, necessito escolher 2 indivíduos para que seja gerado um filho entre eles. Para isso, utilizo um sistema de roleta. Nesse método, os melhores indivíduos (que apresentam melhor custo) terão maiores chances de serem selecionados para serem os pais, mas não necessariamente os melhores sejam escolhidos, por se tratar de um método probabilístico.

Ao ter os pais selecionado, será realizado o *crossover* entre eles, para a geração dos filhos.

Para o *crossover* é utilizado um método em que o filho irá adquirir uma porcentagem aleatória do pai e outra porcentagem aleatória da mãe. A quantidade de filhos irá depender na taxa de *crossover* utilizada, que no meu caso é 80%. Ao atingir essa taxa, o restante dos filhos serão preenchidos por elitismo, ou seja, com os melhores pais da seleção anterior.

Tabela 3. Funções Sugeno de Saída (GT)

Funções de Saída	c_2	c_1	c_0
mf1	0.0321	0.4842	0
mf2	0.03121	5.37	0
mf3	0.03427	1.993	0
mf4	0.04146	2.212	0
mf5	0.042	1.642	0
mf6	0.0957	0.6386	0
mf7	0.01493	1.425	0
mf8	0.1009	1.263	0
mf9	0.008028	2.592	0
mf10	0.07205	3.774	0

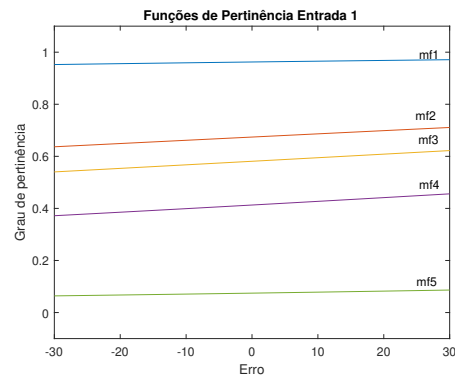


Figura 4. Funções de Pertinência Entrada 1 (GT)

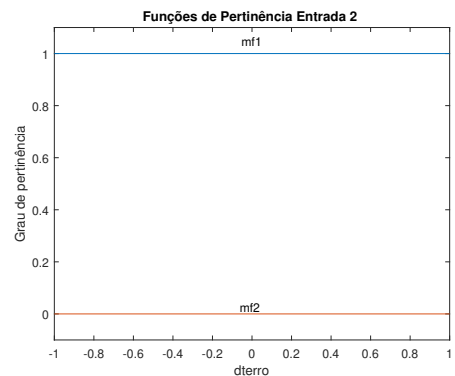


Figura 5. Funções de Pertinência Entrada 2 (GT)

Controlador Mamdani com Funções de pertinência Triangular

Este controlador foi construído com 3 funções de pertinência triangular para cada entrada e saída. Para a sintonia deste controlador os indivíduos foram construídos com as características contendo informações da função de pertinência do erro, da derivada do erro e da saída. Aqui, foram utilizadas funções de pertinência triangulares em virtude da facilidade de implementação do algoritmo que realizará a sintonia.

A primeira entrada estava limitada no intervalo de -26 a 26. A segunda entrada entre -1 e 1. E, por fim, a saída estava contida no intervalo de -0,8 a 0,8. Em seguida, foram definidos os parâmetros necessários para o algoritmo de busca. Neste experimento o algoritmo avaliou para 20 indivíduos durante 40 gerações. Também foi utilizado uma chance de mutação para cada indivíduo de 7%, alterando seu cromossomo em uma posição aleatória.

O custo é definido na equação 5, que segue o mesmo formato do primeiro experimento. Entretanto aqui, foi realizado uma soma ponderada dos termos:

$$Custo_{TT} = 1 \cdot IEA + 3 \cdot ITEA + 1,5 \cdot IG \quad (5)$$

O Índice de Goodhart foi utilizado com coeficientes $\alpha_1 = 0,5$, $\alpha_2 = 0,3$, $\alpha_3 = 0,2$.

Os resultados da sintonia das funções de pertinência podem ser vistos nas figuras abaixo:

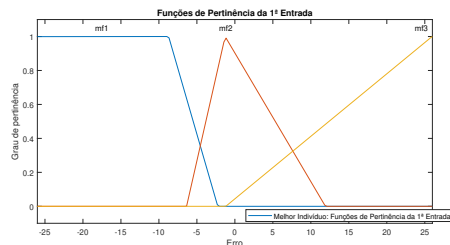


Figura 6. Função de pertinência do erro

Além disso, a adaptabilidade do melhor indivíduo ao longo das últimas 20 gerações é mostrada na Figura 9.

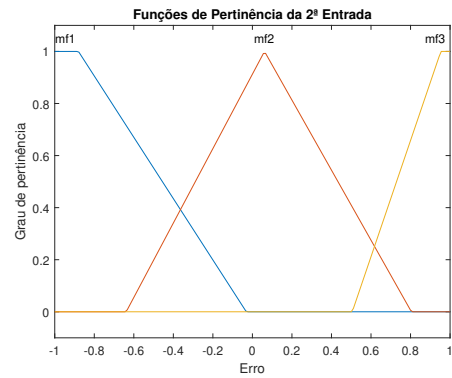


Figura 7. Função de pertinência da derivada do erro

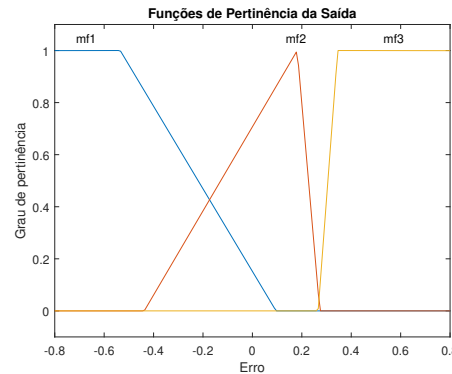


Figura 8. Função de pertinência da saída

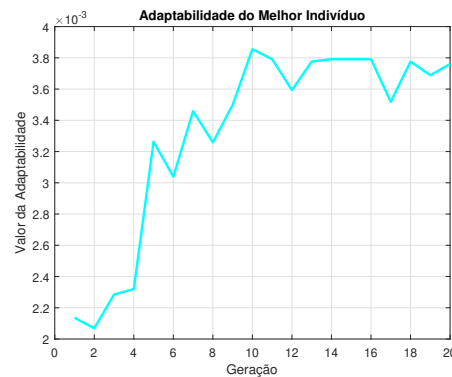


Figura 9. Adaptabilidade do melhor indivíduo

Controlador Mandani de três entradas com funções de pertinência gaussianas

Este controlador foi projetado com três entradas: erro, derivada do erro e nível do tanque 1. Erro e derivada do erro possuem três funções de pertinência gaussianas cada. Nível do tanque 1 possui duas funções de pertinência trapezoidais.

A saída possui cinco funções de pertinência gaussianas. Na execução do algoritmo genético, o melhor resultado foi observado com 20 indivíduos, 100 gerações e uma taxa de mutação de 10%.

A função de custo foi definida como uma combinação entre o IEA (com peso 3), ITEA (com peso 5) e o Índice Goodhart (com peso 1). Dentro do Índice de Goodhart, os seguintes parâmetros foram atribuídos: $\alpha_1 = 0.5$, $\alpha_2 = 0.3$ e $\alpha_3 = 0.2$. O custo médio e a adaptabilidade do melhor indivíduo ao longo do tempo foram:

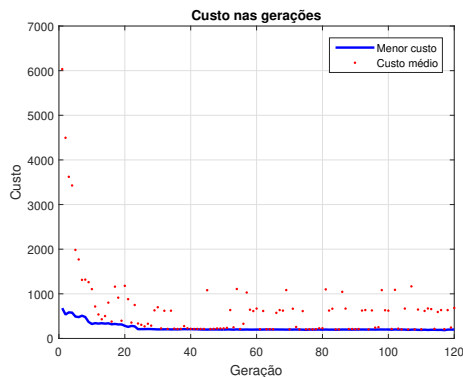


Figura 10. Evolução do custo

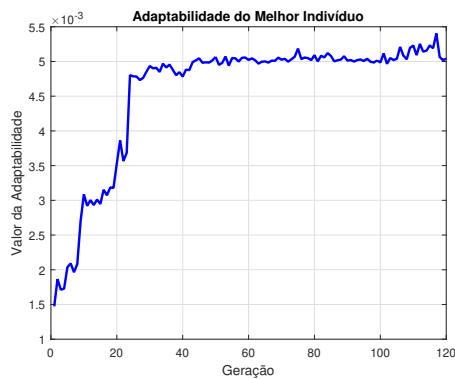


Figura 11. Adaptabilidade do melhor indivíduo

Com todos estes parâmetros definidos, o melhor indivíduo representou as seguintes configurações de funções de pertinência:

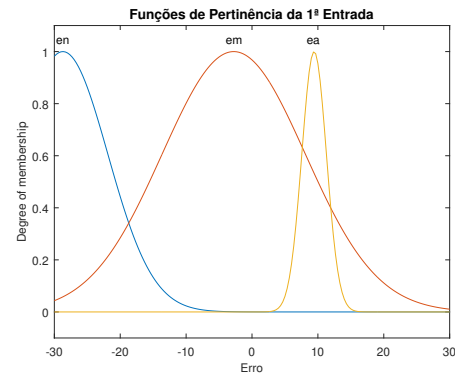


Figura 12. Erro

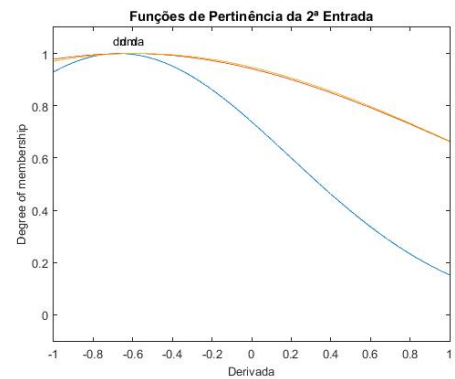


Figura 13. Derivada do erro

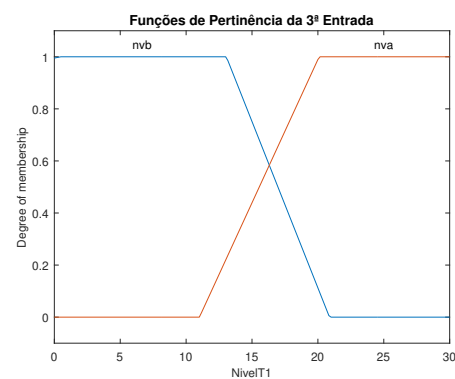


Figura 14. Nível do tanque 1

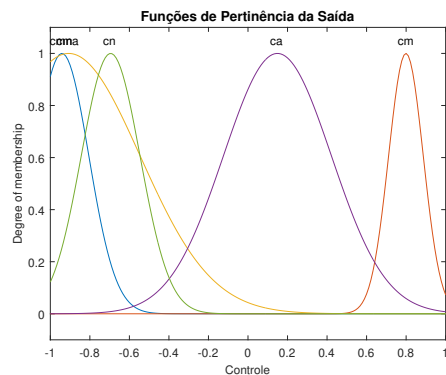


Figura 15. Saída (controle)

A tabela de regras deste Controlador Fuzzy não foi alterada.

Tabela 4. Tabela de regras do controlador

Erro	Derivada do Erro	Nível T1	Saída
en	dn	nva	cmn
en	dn	nvb	cn
en	dm	nvb	cn
en	dm	nva	cmn
en	da	nvb	cn
en	da	nva	cn
em	dn	nva	cn
em	dn	nvb	cm
em	dm	nva	cn
em	dm	nvb	cm
em	da	nvb	cm
em	da	nva	cm
ea	dn	nva	cm
ea	dm	nva	cm
ea	da	nva	cm
ea	dn	nvb	ca
ea	dm	nvb	ca
ea	da	nvb	cma

Resultados

Após sintonizar as funções de pertinência com auxílio do algoritmo genético, um valor de *set point* foi definido para o nível do tanque 2, afim analisar o desempenho de cada controlador sintonizado.

Controlador TSK com Funções de Pertinência Triangular e Trapezoidal (TT)

Utilizando o controlador TSK TT, não foi observado valores significativos de *overshoot* e tempo de acomodação, por outro lado o tempo de subida foi superior a 60 segundos conforme demonstrado na Fig. 16. A Fig. 17 apresenta o sinal de controle, e neste, observa-se um valor de tensão pouco ruidoso e com picos inferiores ao limite do motor DC.

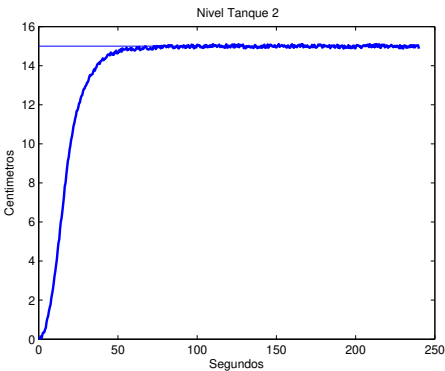


Figura 16. Nível do Tanque 2 (TT)

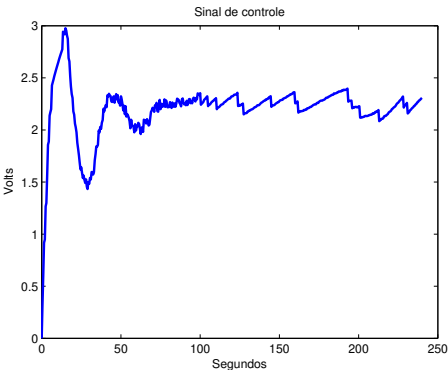


Figura 17. Sinal de Controle Entrada Fixa (TT)

A Fig. 18 apresenta a evolução do melhor indivíduo em cada geração, enquanto que a Fig.

19 apresenta a evolução do custo médio de cada indivíduo por geração.

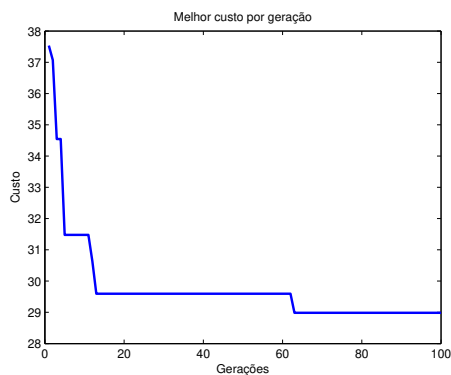


Figura 18. Evolução do melhor custo

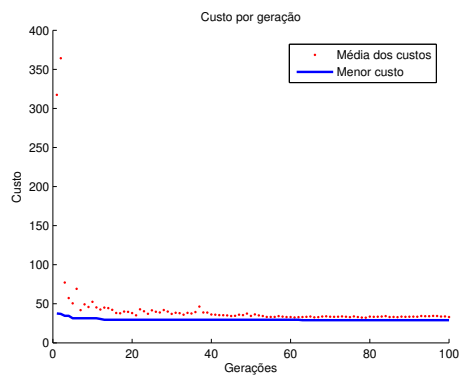


Figura 19. Evolução da média dos custos

Controlador TSK com Funções Gaussiana e Trapezoidal (GT)

Utilizando o controlador TSK GT, não foi observado grandes valores de *overshoot*. Entretanto, podemos observar que o tempo de acomodação foi bem significativo, além de um tempo de subida também elevado, superior a 60 segundos, conforme mostrado na Fig. 20. Já na Fig. 21 apresenta o sinal de controle. No sinal de controle vemos um valor ruidoso no começo, mas que tende a estabilizar com o tempo.

Controlador Mamdani com funções de pertinência triangular

Seguem os resultados da simulação deste controlador com funções de pertinência em formato triangular. Abaixo, a Figura 22 mostra o

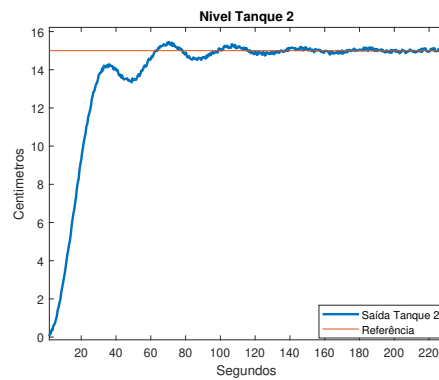


Figura 20. Nível do tanque 2 (GT)

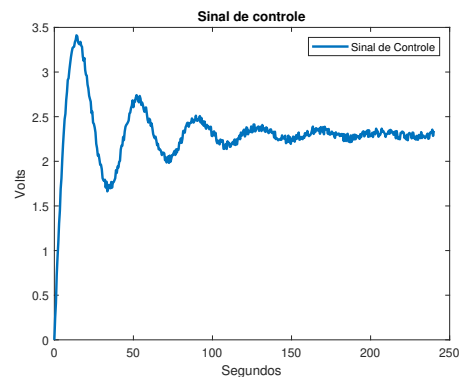


Figura 21. Sinal de Controle (GT)

comportamento do sinal de controle com um comportamento pouco oscilatório. Destaca-se o *overshoot* de quase 5 cm, porém com um tempo de acomodação curto.

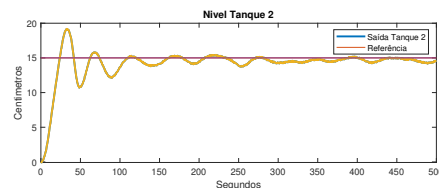


Figura 22. Nível do tanque 2

Controlador Mandani de três entradas com funções de pertinência gaussianas

Para este controlador, o IAE calculado foi 1.2293 e o ITAE 29.3208. Nota-se que não há *overshoot* no objeto de controle (tanque dois). Entretanto, observa-se uma leve característica oscilatória na resposta, mas aparenta não ser considerável.

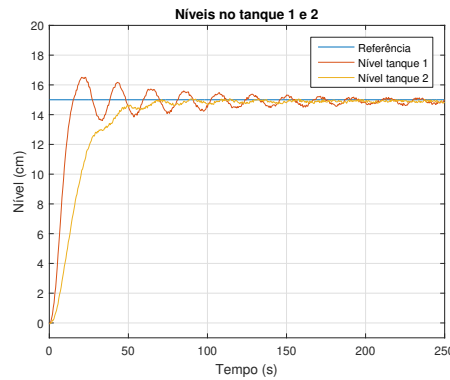


Figura 23. Evolução dos níveis dos tanques

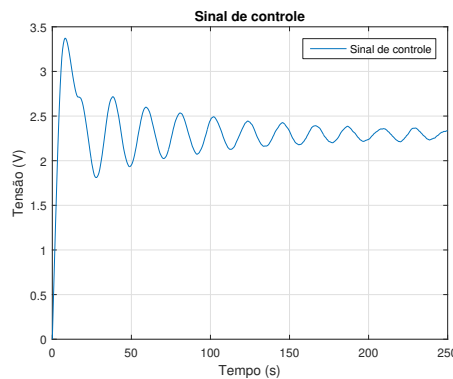


Figura 24. Sinal de controlar

Conclusão

Este trabalho explorou o uso de controladores Fuzzy em um sistema de tanques da Quanser a fim de controlar o nível de fluido do tanque 2. É proposta a sintonia de controladores Fuzzy do tipo TSK e Mamdani com diferentes tipos de funções de pertinência, sendo capaz de realizar o controle de nível de fluido para diferentes *setpoints* (valores fixos e oscilatórios).

Os resultados apresentados pelo controlador Fuzzy TSK (TT), para o *setpoint* fixo, demonstram-se eficientes quando desejado uma resposta que não apresente *overshoot* e que tenha um alto tempo de subida. A ausência de ruídos significativos e com uma faixa de atuação dentro dos limites de tensão do motor DC, o sinal de controle demonstra-se eficiente por não trazer danos significativos a integridade física da planta.

A utilização de funções triangulares para a construção do controlador Mamdani de 2 entradas demonstrou uma resposta rápida porém relativa-

mente oscilatória, se comparado ao Mamdani de 3 entradas. O *overshoot* também se apresenta elevado apenas no primeiro Mamdani apresentado. É interessante notar que para as funções de pertinência da saída, a estratégia da construção das funções foi a mesma das entradas, com aberturas das funções triangulares livres para serem alteradas pelo algoritmo genético.

O controlador Fuzzy Mamdani de três entradas com funções de pertinência gaussianas apresentou-se, para referência constante, eficiente tanto quanto ao baixo *overshoot* quanto ao tempo de acomodação para os critérios de 5% e 2%. A oscilação, por não ter alta significância, tende a não interferir na integridade física da planta.

Por fim, vale ressaltar a importância da sintonia dos parâmetros relevantes para o algoritmo genético a fim de realizar a busca ótima dos parâmetros dos controladores. A função de custo e os métodos escolhidos para a construção da nova geração de indivíduos têm um grande impacto na eficiência do algoritmo. É interessante testar outras estratégias, principalmente ao definir a construção das funções de pertinência por parte do algoritmo, para melhor estudar a diferença entre sintonias manuais e a utilização de métodos de busca.

Referências

1. Zadeh, Lotfi A and others, "Architecture of the IBM System/360", vol. 8, no. 2, pp. 87–101, 1964. (journal)
2. Bauer, Peter and Nouak, Stephan and Winkler, Roman, "A brief course in fuzzy logic and fuzzy control", vol. 10, no. 2, pp. 2005, 1996. (journal)
3. Simoes, Marcelo Godoy and Shaw, Ian S, "Controle e modelagem fuzzy", 2007. (journal)
4. Mamdani, Ebrahim H and Assilian, Sedrak, "An experiment in linguistic synthesis with a fuzzy logic controller", vol. 7, no. 1, pp. 1–13, 1975. (journal)
5. Sugeno, Michio and Kang, GT, "Structure identification of fuzzy model", vol. 28, no. 1, pp. 15–33, 1988. (journal)
6. Zhang, Hong and Ishikawa, Masumi, "Performance improvement of hybrid real-coded genetic algorithm with local search and its applications", vol. 1, no. 1, pp. 1171–1176, 2005. (book)
7. Corcoran, Arthur L and Sen, Sandip, "Using real-valued genetic algorithms to evolve rule sets for classification", pp. 120–124, 1994. (IEEE)
8. Mamdani, Ebrahim H, "Application of fuzzy algorithms for

- control of simple dynamic plant", vol. 121, no. 12, pp. 1585–1588, 1974. (journal)
9. Driankov, Dimiter and Hellendoorn, Hans and Reinfrank, Michael, "Stability of fuzzy control systems", vol. 1, no. 1, pp. 245–292, 1993. (journal)
10. TechSim, <https://techsim.com.br/site/images/Produtos-Quanser/CoupledTankB.png>, 07 03 2021.