

UNIVERSIDADE FEDERAL DE ALAGOAS – UFAL  
CENTRO DE TECNOLOGIA – CTEC  
CURSO DE ENGENHARIA QUÍMICA

GEOVANE DOMINGOS DA SILVA

***MACHINE LEARNING NO CONTROLE PREDITIVO DE PROCESSOS***

Maceió

2020

GEOVANE DOMINGOS DA SILVA

***MACHINE LEARNING NO CONTROLE PREDITIVO DE PROCESSOS***

Trabalho de Conclusão de Curso apresentado  
como requisito para a obtenção do título de  
Bacharel em Engenharia Química.

Orientador: Prof. Dr. Frede de Oliveira  
Carvalho.

Maceió

2020

**Catlogação na fonte**  
**Universidade Federal de Alagoas**  
**Biblioteca Central**  
**Divisão de Tratamento Técnico**

Bibliotecário: Marcelino de Carvalho Freitas Neto – CRB-4 – 1767

S586m Silva, Geovane Domingos da.  
*Machine learning* no controle preditivo de processos / Geovane Domingos da Silva. – 2021.  
48 f. il. : figs. ; tabs. color.

Orientador: Frede de Oliveira Carvalho.  
Monografia (Trabalho de Conclusão de Curso em Engenharia Química).  
Universidade Federal de Alagoas. Centro de Tecnologia. Maceió, 2020.

Bibliografia: f. 46-48.

1. Controle de processos químicos. 2. Controle preditivo. 3. Aprendizado do computador. 4. Python (Linguagem de programação de computador). I. Título.

CDU: 661:004.4'4



UNIVERSIDADE FEDERAL DE ALAGOAS - UFAL  
UNIDADE ACADÊMICA CENTRO DE TECNOLOGIA  
CURSO DE ENGENHARIA QUÍMICA



**GEOVANE DOMINGOS DA SILVA**

***MACHINE LEARNING NO CONTROLE PREDITIVO DE PROCESSOS***

**BANCA EXAMINADORA**

*Frede de Oliveira Carvalho*  
Prof. Dr. Frede de Oliveira Carvalho (Orientador)

*Hugo Valença De Araújo*  
Prof. Dr. Hugo Valença De Araújo

*Rodolfo Junqueira Brandão*  
Prof. Dr. Rodolfo Junqueira Brandão

## **AGRADECIMENTOS**

Inicialmente, gostaria de agradecer a Deus por me dar sabedoria e força para que eu pudesse concluir mais essa etapa da minha vida.

Ao meus pais, Cida e George, pelo amor que sempre me deram e estarem presentes me dando apoio incondicional em todos momentos que eu precisei. Ao meu irmão, George, com quem compartilho grande parte dos meus gostos, por todo o apoio, além da sua ajuda fundamental para o êxito desse trabalho.

Agradeço aos meus amigos e colegas, com quem compartilhei vários momentos. Em especial à minha turma, os Sobrevientes: os Arthurs, Bia, Elisa, Gustavo, Manu e May, pela oportunidade de convívio e cooperação mútua ao longo do curso. A vocês desejo todo o sucesso do mundo.

Por último e não menos importante, gostaria de agradecer ao LABSIA e ao meu orientador Frede, pelo incentivo, paciência e dedicação, estando sempre à disposição sem medir esforços, contribuindo não só para este trabalho, mas para minha vida acadêmica e profissional.

## RESUMO

A popularização do conceito de Indústria 4.0, provocado pelo avanço tecnológico e ampliação do uso de técnicas de Inteligência Artificial, buscando processos cada vez mais eficientes, fez com que o controle de processos evoluísse mais rapidamente na última década. Isso se deu com a implementação de técnicas de controle avançado, como é o caso do controle preditivo com modelo (MPC). Desenvolvido nos anos 80 para o controle de usinas de energia e refinarias de petróleo, o MPC pode agora ser encontrado na indústria química e de alimentos. Esse tipo de estratégia de controle tem como vantagens o fato de poder ser aplicado em sistemas não lineares, que apresentam abordagem mais complicada no controle clássico, e lidar com a presença de restrições operacionais. O MPC necessita de uma implementação computacionalmente mais complexa que as técnicas de controle clássico. Nesse contexto, a linguagem de programação *Python* tem se mostrado promissora em várias aplicações na ciência e indústria, fato que se deve principalmente à sua ampla gama de bibliotecas, que são livres e gratuitas, fazendo com que se torne uma ótima alternativa a *softwares* pagos como o MATLAB®. Com o aumento da automação de processos, graças ao barateamento de equipamentos de monitoramento, tem-se uma maior disponibilidade de dados. Este fato possibilita a abordagem do problema com o uso de modelos internos no controlador MPC baseados em *Machine Learning*. As técnicas de *Machine Learning*, um dos pilares da Indústria 4.0, estudam a construção de algoritmos computacionais que possibilitem o desenvolvimento de modelos a partir da análise de dados fornecidos e, com isso, a generalização da informação aprendida na forma de previsões. Na indústria química, um dos tipos mais comuns de equipamentos é o reator de tanque de mistura (CSTR). Em razão disso, este trabalho abordou a construção de estratégias para a utilização de *Machine Learning* no MPC aplicado no caso de um CSTR com sistema reativo não linear da reação catalítica de hidrogenação do etileno, avaliando também a aplicabilidade do *Python* e suas respectivas bibliotecas específicas para controle e *Machine Learning*, disponibilizadas gratuitamente.

**Palavras-Chave:** Controle de processos; MPC; *Machine Learning*; *Python*.

## ABSTRACT

The popularization of the concept of Industry 4.0, caused by technological advances and the expansion of the use of Artificial Intelligence techniques, seeking increasingly efficient processes, has made process control evolve more rapidly in the last decade. This happened with the implementation of advanced control techniques, as is the case of model predictive control (MPC). Developed in the 1980s to control power plants and oil refineries, MPC can now be found in the chemical and food industries. This type of control strategy has the advantage of being able to be applied in nonlinear systems, which have a more complicated approach in classical control, and to deal with the presence of operational restrictions. The MPC requires a computationally more complex implementation than the classic control techniques. In this context, the Python programming language has shown promise in several applications in science and industry, a fact that is mainly due to its wide range of libraries, which are free and free, making it a great alternative to paid software such as MATLAB®. With the increase in process automation, thanks to cheaper monitoring equipment, there is greater availability of data. This fact makes it possible to approach the problem with the use of internal models in the MPC controller based on Machine Learning. Machine Learning techniques, one of the pillars of Industry 4.0, study the construction of computational algorithms that enable the development of models based on the analysis of the data provided and, with this, the generalization of the information learned in the form of predictions. In the chemical industry, one of the most common types of equipment is the Continuous Stirred Tank Reactor (CSTR). For this reason, this work addressed the construction of strategies for the use of Machine Learning in the MPC applied in the case of a CSTR with a nonlinear reactive system of the catalytic reaction of ethylene hydrogenation, also evaluating the applicability of Python and its respective specific libraries for control and Machine Learning, available for free.

**Keywords:** Process Control; MPC; *Machine Learning*; *Python*.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Esquema típico de um controle feedback .....	6
Figura 2 – Oscilações sustentadas obtidas para $K_{cr}$ .....	8
Figura 3 – Número de aplicações do MPC em alguns tipos de indústria .....	11
Figura 4 – Estrutura do controle MPC .....	12
Figura 5 – Esquema de um neurônio artificial ( <i>perceptron</i> ) .....	14
Figura 6 – Esquema do reator de tanque de mistura .....	16
Figura 7 – Estados estacionários e pontos de operação .....	18
Figura 8 – Respostas da concentração para perturbações degrau na vazão em torno dos dois estados estacionários estudados .....	19
Figura 9 – Respostas obtidas pelas EDOs e pela FT em torno do estado estacionário 1 ...	22
Figura 10 – Respostas obtidas pelas EDOs e pela FT em torno do estado estacionário 2 ..	23
Figura 11 – Respostas da FT no estado estacionário 1 com $K < 0$ (em variável desvio) ...	24
Figura 12 – Respostas da FT no estado estacionário 1 com $K > 0$ (em variável desvio) ...	25
Figura 13 – Respostas da FT no estado estacionário 2 com $K < 0$ (em variável desvio) ...	25
Figura 14 – Respostas da FT no estado estacionário 2 com $K > 0$ (em variável desvio) ...	26
Figura 15 – Lugar das raízes da FT em torno do estado estacionário 1, com $K > 0$ .....	26
Figura 16 – Lugar das raízes da FT em torno do estado estacionário 2, com $K < 0$ .....	27
Figura 17 – Resposta da malha fechada do estado estacionário 2 com $K_{cr}$ (em variável desvio) .....	28
Figura 18 – Perfil de perturbação e resposta em torno do estado estacionário 1 .....	29
Figura 19 – Perfil de perturbação e resposta em torno do estado estacionário 2 .....	29
Figura 20 – Esquemas de entradas e saída utilizadas para o treinamento da rede neural ...	30
Figura 21 – Previsões da MLP para o estado estacionário 1 .....	30
Figura 22 – Previsões da MLP para o estado estacionário 2 .....	31
Figura 23 – Respostas das duas redes treinadas para perturbações típicas de controle .....	31
Figura 24 – Respostas do sistema e o esforço do controlador causados pelo aumento de $K_p$ .....	33
Figura 25 – Sintonia do controlador PI para a FT do estado estacionário 1 (em variável desvio) .....	34
Figura 26 – Sintonia do controlador PID para a FT do estado estacionário 1 (em variável desvio) .....	35
Figura 27 – Sintonia do controlador P para a FT do estado estacionário 2 (em variável desvio) .....	37
Figura 28 – Sintonia do controlador PI para a FT do estado estacionário 2 (em variável desvio) .....	38
Figura 29 – Sintonia do controlador PID para a FT do estado estacionário 2 (em variável desvio) .....	39
Figura 30 – Respostas das sintonias por otimização do controlador PID para a FT do estado estacionário 2 .....	39
Figura 31 – Influência dos pesos na resposta em torno do estado estacionário 1 quando $N_c = 2$ e $N_u = 5$ .....	40
Figura 32 – Influência dos pesos na resposta em torno do estado estacionário 1 quando $N_c = 2$ e $N_u = 7$ .....	41
Figura 33 – Influência dos pesos na resposta em torno do estado estacionário 2 quando $N_c = 2$ e $N_u = 5$ .....	42
Figura 34 – Influência dos pesos na resposta em torno do estado estacionário 2 quando $N_c = 2$ e $N_u = 7$ .....	42

Figura 35 – Respostas para o controlador PI e o MPC na resposta em torno do estado estacionário 1 .....	43
---	----

## LISTA DE TABELAS

Tabela 1 – Ganhos para cada tipo de controlador pelo método de Ziegler-Nichols .....	8
Tabela 2 – Parâmetros do sistema considerados no estudo .....	18
Tabela 3 – Estados estacionários do sistema que foram estudados .....	19
Tabela 4 – Intervalos de busca considerados na sintonia do controlador para a FT do estado estacionário 1 .....	28
Tabela 5 – Intervalos de busca considerados na sintonia do controlador para a FT do estado estacionário 2 .....	28
Tabela 6 – Intervalos de perturbação da vazão $w_1$ e intervalo de tempo utilizado para geração de dados .....	29
Tabela 7 – Métricas do desempenho das redes neurais treinadas .....	31
Tabela 8 – Valores de busca para os parâmetros de sintonia do MPC .....	32
Tabela 9 – Critérios de desempenho calculados para o controlador PI no estado estacionário 1 .....	34
Tabela 10 – Ganhos calculados para o controlador PI no estado estacionário 1 .....	34
Tabela 11 – Critérios de desempenho calculados para o controlador PID no estado estacionário 1 .....	35
Tabela 12 – Ganhos calculados para o controlador PID no estado estacionário 1 .....	36
Tabela 13 – Critérios de desempenho calculados para o controlador P no estado estacionário 2 .....	36
Tabela 14 – Ganhos calculados para o controlador P no estado estacionário 2 .....	36
Tabela 15 – Critérios de desempenho calculados para o controlador PI no estado estacionário 2 .....	37
Tabela 16 – Ganhos calculados para o controlador PI no estado estacionário 2 .....	37
Tabela 17 – Critérios de desempenho calculados para o controlador PID no estado estacionário 2 .....	38
Tabela 18 – Ganhos calculados para o controlador PID no estado estacionário 2 .....	38
Tabela 19 – Influência dos pesos na métricas avaliadas na resposta em torno do estado estacionário 1 quando do $N_c = 2$ e $N_u = 5$ .....	40
Tabela 20 – Influência dos pesos na métricas avaliadas na resposta em torno do estado estacionário 1 quando do $N_c = 2$ e $N_u = 7$ .....	41
Tabela 21 – Métricas para o controlador PI e o MPC na resposta em torno do estado estacionário 1 .....	43

## LISTA DE SIMBOLOS E ABREVIATURAS

### Lista de Abreviaturas

MPC	Controle preditivo com modelo
CSTR	<i>Continuous Stirred Tank Reactor</i> (Reator Agitado de Fluxo Contínuo)
IA	Inteligência Artificial
MLP	<i>Multilayer Perceptron</i> (Perceptrons em Multicamadas)
P	Proporcional (controle)
PI	Proporcional Integral (controle)
PID	Proporcional, integral e derivativo (controle)
ZN	Ziegler-Nichols
RNA	Redes Neurais Artificiais
FT	Função de Transferência
PSO	<i>Particle Swarm Optimization</i> (Otimização por Enxame de Partículas)
IAE	<i>Integral of the absolute value of error</i> (Integral do Erro Absoluto)
ISE	<i>Integral of the squared value of error</i> (Integral do Erro Quadrático)
ITAE	<i>Integral of time-weighted absolute error</i> (Integral do Erro Absoluto Ponderado pelo Tempo)
ITSE	<i>Integral of time-weighted squared error</i> (Integral do Erro Quadrático Ponderado pelo Tempo)
ARX	<i>Auto-Regressive eXogenous model</i>
ARMAX	<i>Auto-Regressive Moving Average eXogenous model</i>
NARX	<i>Nonlinear Auto-Regressive eXogenous model</i>
NARMARX	<i>Nonlinear Auto-Regressive Moving Average eXogenous model</i>
NMPC	<i>Nonlinear Model Predictive Control</i>

SISO                    *Simple Input Simple Output* (Entrada e saída única)

MIMO                   *Multiple Input Multiple Output* (Entradas e saídas múltiplas)

### **Lista de Símbolos**

#### ***Processo***

$C_A$                     Concentração de reagente no reator

$w_1$                     Vazão de alimentação na entrada 1

$h$                       Altura do nível do reator

$V$                       Volume do reator

#### ***Controle Clássico***

$K_p$                     Ganho Proporcional

$\tau_i$                     Tempo Integral

$\tau_d$                     Tempo Derivativo

$K_i$                     Ganho Integral

$K_d$                     Ganho Derivativo

$K_{cr}$                    Ganho Crítico

$P_{cr}$                    Período Crítico

$T_a$                     Tempo de ascensão

#### ***Controle MPC***

$C_a$                     Concentração de reagente no reato

$J$                       Função objetivo ou custo

$N_u$                     Horizonte de controle

$N_2$                     Horizonte de predição

$y$                       Variável de saída

$u$                       Ação do controle

$\delta$                     Peso dos erros

$\lambda$                     Peso do esforço do controlador

#### ***Rede Neural***

$\varphi$                     Função de ativação do neurônio

$x_i$                     Entradas

$y$                       Saídas

$w_i$                     Pesos

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>1</b>
<b>2. OBJETIVOS .....</b>	<b>4</b>
<b>2.1 Objetivo geral.....</b>	<b>4</b>
<b>2.2 Objetivos específicos.....</b>	<b>4</b>
<b>3. REVISÃO BIBLIOGRÁFICA .....</b>	<b>5</b>
<b>3.1 Controle Clássico .....</b>	<b>5</b>
3.1.1 Controle Feedback.....	5
3.1.2 Controlador PID .....	6
3.1.3 Critérios de desempenho .....	7
3.1.4 Técnicas de sintonia.....	8
<b>3.2 Otimização por Enxame de Partículas (PSO).....</b>	<b>9</b>
<b>3.3 Controle Avançado .....</b>	<b>10</b>
3.3.1 Controle Preditivo com modelo (MPC).....	10
3.3.1.1 Modelo de identificação do sistema .....	13
<b>3.4 <i>Machine Learning</i> .....</b>	<b>13</b>
<b>3.5 Python e suas respectivas bibliotecas.....</b>	<b>14</b>
<b>3.6 O processo controlado .....</b>	<b>15</b>
<b>4. METODOLOGIA.....</b>	<b>16</b>
<b>4.1 Modelagem do problema.....</b>	<b>16</b>
4.1.1 Balanço de Massa Global .....	16
4.1.2 Balanço de Massa para o etileno .....	17
<b>4.2 Análise do estado estacionário e escolha do ponto de operação .....</b>	<b>18</b>
<b>4.3 Análise do estado transiente do sistema e sua resposta a perturbações .....</b>	<b>19</b>
<b>4.4 Controle Clássico .....</b>	<b>20</b>

4.4.1	Obtenção da função de transferência.....	20
4.4.2	Estabilidade da malha fechada .....	23
4.4.3	Lugar das raízes .....	24
<b>4.5</b>	<b>Implementação e sintonia do controlador PID .....</b>	<b>27</b>
<b>4.6</b>	<b>Controle Preditivo baseado em modelo (MPC) .....</b>	<b>29</b>
4.6.1	Geração de dados e treinamento da rede .....	29
4.6.2	Implementação e Sintonia do MPC .....	31
<b>5</b>	<b>RESULTADOS E DISCUSSÃO.....</b>	<b>33</b>
<b>5.1</b>	<b>Controle Clássico – Estado estacionário 1.....</b>	<b>33</b>
5.1.1	Análise do controlador P .....	33
5.1.2	Análise do controlador PI.....	33
5.1.3	Análise do controlador PID .....	35
<b>5.2</b>	<b>Controle Clássico – Estado estacionário 2.....</b>	<b>36</b>
5.2.1	Análise do controlador P .....	36
5.2.2	Análise do controlador PI.....	37
5.2.3	Análise do controlador PID .....	38
<b>5.3</b>	<b>MPC – Estado estacionário 1.....</b>	<b>39</b>
<b>5.4</b>	<b>MPC – Estado estacionário 2.....</b>	<b>41</b>
<b>5.5</b>	<b>Comparação entre as estratégias de controle implementadas.....</b>	<b>43</b>
<b>6</b>	<b>CONCLUSÃO.....</b>	<b>44</b>

## 1. INTRODUÇÃO

O controle de processos é uma área essencial da indústria de processos químicos, sendo responsável por assegurar o bom desempenho do processo, de forma a manter as variáveis de processo dentro das faixas desejáveis. Stephanopoulos (1984) destaca que as principais razões para se controlar um processo são: respeitar restrições operacionais, atingir especificações de produção, garantir a segurança da operação, atender regulamentações ambientais, além de fatores econômicos.

Na indústria química, o reator do tipo tanque de mistura perfeita (CSTR, *Continuous Stirred Tank Reactor*) é um dos tipos mais comuns de reator, sendo empregado em uma ampla gama de processos. O CSTR é um reator do tipo contínuo, o que significa que deve geralmente operar no estado estacionário. Para isso, é imprescindível um sistema de controle eficiente.

O controle atua de forma a minimizar o desvio, ou o erro, entre a variável de interesse, chamada de variável controlada, e um valor ou trajetória de referência (*setpoint*) por meio da manipulação de outras variáveis de processo, que recebem o nome de variáveis manipuladas. Uma das primeiras abordagens e ainda muito utilizada, hoje conhecida como controle clássico, faz uso de um controlador que pode atuar de forma proporcional, integral e derivativa (PID) ao erro. Entretanto, a depender do tipo de sistema, a abordagem do problema utilizando controle clássico se torna ineficiente ou de elevada complexidade (OGATA, 2010; STEPHANOPOULOS, 1984).

Ainda no final do século XX foram desenvolvidas novas técnicas de controle para atender as novas demandas. As denominadas técnicas de controle moderno mais populares são: controle adaptativo, controle preditivo, controle preditivo com modelo (MPC, *Model Predictive Control*), controle robusto e ainda técnicas que empregam inteligência artificial em outras técnicas de controle, conhecido como controle inteligente (VOJTĚŠEK, 2007). Entre essas, o MPC é umas das que mais tem impacto significativo no controle de processos (BRAVO; NORMEY-RICO, 2009).

Desenvolvido desde o final dos anos 80 e sendo destinado inicialmente para o controle em refinarias de petróleo e em usinas elétricas, o controle preditivo com modelo agora pode ser encontrado para o controle na indústria de cimento, torres de secagem, colunas de destilação, plantas de PVC e geradores de vapor (CAMACHO; BORDONS, 2007; QIN; BADGWELL, 2003).

Um dos aspectos fundamentais do MPC é a obtenção de um modelo da planta, que é utilizado para realizar previsões acerca do comportamento futuro da planta. A partir dessas previsões as variáveis são manipuladas de modo que a variável controlada se aproxime do *setpoint*. Isso é feito por meio da minimização de uma função custo, também chamada de função objetivo, que leva a uma solução ótima para uma sequência de ações de controle. Essas características permitem que o MPC possa atuar em sistemas não lineares e permita a inclusão de restrições operacionais do problema durante sua implementação (CAMACHO; BORDONS, 2007).

Na indústria química, o avanço no desenvolvimento de equipamentos para monitoramento e automação de processos, aliado à redução de custos desses mesmos equipamentos, resultou em uma maior quantidade e qualidade de dados de processo disponíveis. Em razão dessa expansão tecnológica, que vem se intensificando cada vez mais, a indústria vem passando por uma nova fase, chamada Indústria 4.0. (VOJTĚŠEK, 2007; FORTUNA et al, 2007)

O conceito de Indústria 4.0, busca explorar os efeitos de um mundo cada vez mais conectado. Isso se dá pela capacidade cada vez maior de processamento, que permite trabalhar com uma grande quantidade de informações disponíveis (*Big Data*) e ampliação do uso de técnicas de Inteligência Artificial (IA), especificamente de *Machine Learning*, para extrair informações do comportamento do processo, fornecendo soluções capazes de tornar os processos cada vez mais eficientes, como é o caso do controle de processos.

No caso do específico do controle preditivo com modelo (MPC), técnicas de *Machine Learning* podem ser empregadas na etapa de identificação do sistema no MPC e, assim, construir modelos de predição capazes de descrever o comportamento do sistema de forma mais fiel. Para isso, se faz uso de técnicas de *Machine Learning* como redes neurais artificiais (RNA), *Support Vector Machine* e Lógica Fuzzy (ZHEJING; DAOYING; YOUXIAN, 2007; SARAIVA et al., 2020).

Computacionalmente, para implementação do controle preditivo com modelo, o MATLAB® e similares oferecem *toolboxes* bastante populares, mas se trata de um ou mais *softwares* pagos. A linguagem de programação *Python* surge, então, como uma possível alternativa, graças a sua extensa biblioteca, que permitiria trabalhar com controle de processos de forma gratuita. O *Python* cada vez mais vem tomando espaço em aplicações na ciência e engenharia, tanto no meio acadêmico quanto industrial. Contudo, embora ofereça bibliotecas

que possam ser utilizadas no controle preditivo com modelo no *Python*, como é o caso do GEKKO e o APMonitor, elas não são implementadas facilmente, pois não contam com *toolboxes* oferecendo uma interface. Além disso, as duas bibliotecas citadas não oferecem suporte para a implementação do MPC utilizando *Machine Learning* (TAKÁCS et al., 2015; BUCHER, 2019).

Reações como a de hidrólise catalítica do etileno em CSTR, são base para diversos estudos para implementação estratégia de sistemas controle, devido a sua forte não linearidade, o que torna complicada a aplicação das estratégias de controle usuais.

Desta forma, neste trabalho foi realizado o estudo e comparação de estratégias de controle clássico e preditivo com modelo aplicados para a reação catalítica de hidrólise do etileno em um CSTR, avaliando a aplicabilidade da linguagem *Python* e suas bibliotecas.

## 2. OBJETIVOS

### 2.1 Objetivo geral

Desenvolver e avaliar estratégias na linguagem de programação *Python* para implementação de um sistema de controle preditivo com modelo (MPC) para um reator CSTR baseado em redes neurais artificiais.

### 2.2 Objetivos específicos

- Aplicar e avaliar ferramentas para modelagem e simulação de reatores CSTR (EDOs) na linguagem de programação *Python*.
- Avaliar a aplicabilidade da linguagem de programação *Python* para utilização de redes neurais artificiais nas construções do modelo para MPC.
- Avaliar a aplicabilidade da linguagem de programação *Python* para implementação do controle clássico de processo *feedback*.
- Avaliar a aplicabilidade da linguagem de programação *Python* para implementação do controle preditivo com modelo.

### 3. REVISÃO BIBLIOGRÁFICA

#### 3.1 Controle Clássico

O controle de processos tem como objetivo manter um determinado processo estável, dentro de condições operacionais previamente determinadas. Para isso um controlador atua de forma a levar a variável controlada ao valor desejado (*setpoint*), de modo que o desvio ou o erro se aproxime de zero. Isso é feito por meio da manipulação de outras variáveis que têm influência sobre do processo (STEPHANOPOULOS, 1984).

Há uma imensa variedade de tipos de sistema, que têm abordagem influenciada pela definição de quais variáveis serão medidas, manipuladas ou controladas; se uma ou mais variáveis são manipuladas e controladas, respectivamente os sistemas *Single Input Single Output* (SISO) e *Multiple Input Multiple Output* (MIMO); e também se a variável medida é aferida antes ou após ocorrer a perturbação no sistema, o primeiro recebe o nome de controle *feedforward* enquanto que o segundo é denominado controle *feedback*. (STEPHANOPOULOS, 1984; LEBLANC; COUGHANOWR, 2009)

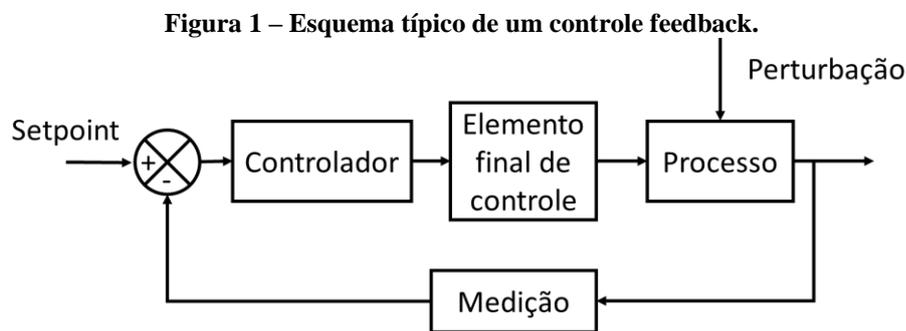
Para uma mesma variável controlada, a ordem de medição e manipulação influencia na qualidade do controle realizado. No controle *feedforward*, as variáveis medidas são as perturbações no processo, e as ações corretivas são tomadas antes que o sistema seja afetado. No controle *feedback* é necessário que exista um desvio na variável controlada para que a variável manipulada seja ajustada pelo controlador. Dessa forma, a variável medida é mensurada após afetar o sistema. No controle *feedforward* é necessário estimar de que forma as perturbações medidas irão afetar o processo. Nem sempre é possível fazer essa estimativa, seja pela complexidade do sistema ou por uma grande quantidade de perturbações que influenciam o sistema, o que faz com que o controle *feedback* seja mais utilizado (LEBLANC; COUGHANOWR, 2009).

##### 3.1.1 Controle Feedback

O controle *feedback* pode ser dividido em dois tipos: o problema do servo, no qual é realizada uma mudança no *setpoint*, e o problema regulatório, em que se controla as perturbações sofridas no processo. O sistema ainda pode ter múltiplas entradas e múltiplas saídas (MIMO, *Multiple Input Multiple Output*) que, diferente dos sistemas com uma única entrada e uma saída (SISO, *Single Input Single Output*), têm abordagem mais complexa (OGATA, 2010; STEPHANOPOULOS, 1984).

A partir dos desvios medidos, conhecidos como erros, são tomadas ações para minimizá-los através da manipulação de variáveis do processo por um controlador. Como mostrado na Figura 1, a configuração típica de controle *feedback* com seus elementos de controle conta com (STEPHANOPOULOS, 1984):

- Controlador: recebe a diferença entre o *setpoint* e o valor medido e envia um sinal para o elemento final de controle;
- Elemento final de controle: recebe o sinal do controlador e atua no processo manipulando alguma variável, abrindo ou fechando uma válvula, por exemplo;
- O processo: sistema sobre o qual o controle atuará;
- Medição: mede a saída do processo, avaliando a variável controlada e enviando para o ponto de soma.



Fonte: Autor (2020)

### 3.1.2 Controlador PID

No controle clássico, o tipo de controlador mais utilizado é o PID (proporcional integral derivativo) leva em consideração a diferença entre a variável e seu valor referência, também conhecido como erro. O controlador é composto por três ações, a saber: a proporcional, que é proporcional ao erro; a integral, que é a integral do erro; e a derivativo, sendo a derivada do erro. (OGATA, 2010)

A equação geral do controlador PID é apresentada na Equação 1, onde  $p(t)$  é a variável controlada,  $\epsilon(t)$  é o erro,  $K_p$  é o ganho proporcional,  $\tau_i$  é o tempo integral e  $\tau_d$  é o tempo derivativo.

$$p(t) = p_0 + K_p \left( \epsilon(t) + \frac{\int \epsilon(t) dt}{\tau_i} + \tau_d \cdot \frac{d\epsilon(t)}{dt} \right) \quad (1)$$

A equação também pode ser escrita em termo de ganhos integral e derivativo,  $K_i = \frac{K_p}{\tau_i}$  e  $K_d = \tau_d K_d$ , respectivamente (Equação 2).

$$p(t) = p_0 + K_p \epsilon(t) + K_i \int \epsilon(t) dt + K_d \frac{d\epsilon(t)}{dt} \quad (2)$$

Os tipos de controladores são dados pelas combinações das ações de controle, assim o controlador pode ser do tipo proporcional (P), proporcional integral (PI), proporcional derivativo (PD) ou ainda a combinação de todos eles, o controlador proporcional integral derivativo (PID). Cada tipo de controlador pode apresenta respostas distintas (COUGHANOWR, Stephanopoulos):

- Controlador P: é o tipo de controlador mais simples só conta com um parâmetro de ajuste, o ganho proporcional  $K_p$ . Ele é rápido, pode reduzir o erro, mas não o elimina, gerando um *offset*;
- Controlador PI: conta com dois parâmetros de ajuste. Além de  $K_p$ , conta com o ganho integral  $K_i$ . A presença da integral leva o erro a zero, eliminando o *offset*, mas, em contra posição, aumenta a instabilidade do sistema, de forma a aumentar comportamento oscilatório do sistema;
- Controlador PID: esse tipo de controlador conta com três parâmetros de ajuste, passando a contar com o ganho derivativo  $K_d$ . Robusto e rápido, reduz a instabilidade do sistema.

### 3.1.3 Critérios de desempenho

Há uma série de critérios que podem ser avaliados na escolha dos parâmetros do controlador. Entre elas, Coughanowr e Leblanc (2009) citam o sobressinal (*overshoot*), que é a porcentagem em que o maior pico excede o valor do estado estacionário; o tempo de ascensão, que é o tempo decorrido até a resposta atingir pela primeira vez seu valor de estado estacionário, e as integrais do erro, que são: a Integral do erro absoluto (IAE, *Integral of the absolute value of error*); a Integral do erro quadrático (ISE, *Integral of the square of the error*); Integral do erro absoluto multiplicado pelo tempo (ITAE, *Integral of time-weighted absolute error*); e Integral do erro quadrático multiplicado pelo tempo (ITSE, *Integral of time-weighted square error*), dadas pelas Equações 3, 4, 5 e 6, respectivamente.

$$IAE = \int |e(t)| dt \quad (3)$$

$$ISE = \int e^2(t) dt \quad (4)$$

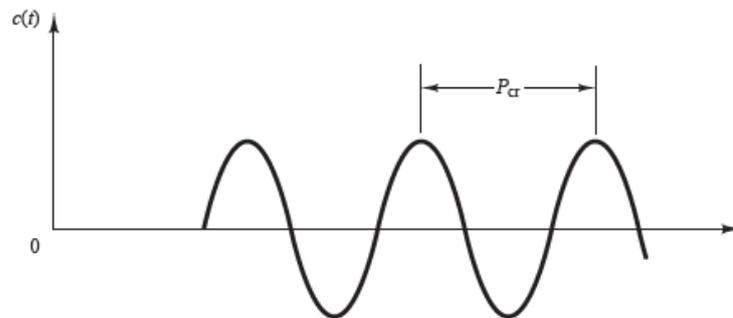
$$ITAE = \int t|e(t)| dt \quad (5)$$

$$ITSE = \int te^2(t) dt \quad (6)$$

### 3.1.4 Técnicas de sintonia

A etapa de escolha dos parâmetros do controlador é chamada de sintonia. Essa etapa é fundamental para que o desempenho adequado do sistema seja alcançado. Há várias técnicas para a sintonia do controlador PID, entre as mais conhecidas pode-se citar: Ziegler-Nichols (ZN) e Cohen-Coon. A técnica de Ziegler-Nichols é baseada na obtenção de um ganho proporcional crítico  $K_{cr}$  em que o sistema apresente oscilações sustentadas (Figura 2). Com o valor de  $K_{cr}$  e o período entre as oscilações  $P_{cr}$  são obtidos os valores dos ganhos para cada tipo de controlado, conforme é mostrado na Tabela 1.

**Figura 2 – Oscilações sustentadas obtidas para  $K_{cr}$**



Fonte: Ogata (2010)

**Tabela 1- Ganhos para cada tipo de controlador pelo método de Ziegler-Nichols.**

Tipo de controlador	$K_p$	$K_i$	$K_d$
P	$0,5K_{cr}$	0	0
PI	$0,45K_{cr}$	$0,54 \frac{K_{cr}}{P_{cr}}$	0
PID	$0,6K_{cr}$	$1,2 \frac{K_{cr}}{P_{cr}}$	$0,075K_{cr}P_{cr}$

Fonte: Adaptado de Ogata (2010)

Existem também as técnicas baseadas na otimização direta dos parâmetros que usam as integrais do erro: ISE, IAE, ITSE e ITAE como função objetivo, que devem ser minimizadas.

Rajinikanth e Latha (2012) utilizaram Otimização por Enxame de Partículas (PSO, *Particle Swarm Optimization*) para a sintonia fina do controlador em aplicado a um controle em um CSTR.

### 3.2 Otimização por Enxame de Partículas (PSO)

A Otimização por Enxame de Partículas (PSO, *Particle Swarm Optimization*) faz parte da família de algoritmos meta-heurísticos baseados no comportamento da natureza, entre outros pode-se citar: Algoritmos Genéticos e Otimização por Colônia de Formigas, que fazem parte da classe dos algoritmos baseados em fenômenos biológicos, assim como o PSO; e Recozimento Simulado, que é baseado na Termodinâmica (VALIN, 2019; MIRJALILI, 2019).

Esses algoritmos se destacam pela capacidade de otimizar problemas complexos, em grande parte não-lineares, de maneira satisfatória exigindo menor custo computacional, podendo, inclusive, lidar com funções multimodais e com a presença ou não de restrições Além de lidarem melhor com a presença de ótimos locais, já que não são baseados em derivadas como a grande parte dos algoritmos de otimização convencionais (ZHANG, 2017).

Proposto inicialmente por Kennedy e Eberhart (1995), o PSO surgiu baseado numa metáfora do comportamento social observado em grupos de animais como cardumes, bando de pássaros e enxames. A partir de estudos sobre o comportamento em grupos de animais, foi identificado que estes animais são capazes de compartilhar informações com o grupo em que estão inseridos, buscando minimizar risco de encontrar predadores e maximizar a sobrevivência do grupo. (ALMEIDA; LEITE, 2019).

A construção do algoritmo é baseada na utilização de um modelo social simplificado, em que as partículas são possíveis soluções do problema que percorrem o espaço de solução. Por meio da troca de informações entre si, as partículas são influenciadas pelas experiências do enxame e são direcionadas às posições de melhor ajuste, que são calculadas a partir da função objetivo, até eventualmente convergir para a solução (MIRJALILI, 2019; ZHANG, 2017).

A posição de cada indivíduo do enxame é representada por um vetor posição. Para calcular as novas posições de cada partícula dois vetores são ser considerados: o vetor da posição no instante anterior e vetor velocidade de cada partícula, que define a direção e a intensidade do movimento da partícula. O vetor velocidade é expresso em função da velocidade no instante anterior, além das melhores posições encontradas pela própria partícula e pelo enxame, que são multiplicados pelos fatores cognitivo e social, respectivamente. A partir disso,

a nova posição da partícula é calculada pela dos vetores da nova velocidade e da posição atual da partícula (ZHANG, 2017; KENNEDY, EBERHART, 1995).

O movimento do enxame depende dos fatores cognitivo e social de todas as partículas. O componente cognitivo controla a tendência de cada partícula de ser atraída para a sua melhor posição obtida em algum um instante anterior. Já o componente social controla como as partículas são atraídas para a posição encontrada pelo enxame. (MIRJALILI, 2019).

### 3.3 Controle Avançado

Segundo Draeger, Engell e Hanke (1995), desde os anos 90 houve um aumento significativo na qualidade demandada no controle nas indústrias de processo. Isso se deu por um aumento da complexidade das plantas, além de especificações do produto mais elevadas.

Para lidar com particularidades de alguns problemas muito comuns na engenharia química, como a condição de não linearidade do sistema, foram criadas outras abordagens para o controle de processos, conhecidas como controle moderno ou avançado. Entre as técnicas mais conhecidas estão o controle adaptativo, controle preditivo e o controle preditivo com modelo (MPC, *Model Predictive Control*) (VOJTĚŠEK, 2007). Das técnicas de controle citadas, o MPC (do inglês *Model Predictive Control*) tem tido maior destaque, tanto em aplicações industriais como em trabalhos e estudos publicados (BRAVO; NORMEY-RICO, 2009).

Recentemente essa nova área de controle tem sido alvo de vários estudos, por exemplo, Vojtěšek (2007) estudou técnicas de controle adaptativo em um CSTR e PFR para as reações de Van der Vusse, e Saraiva (2017) estudou essa mesma reação para um CSTR utilizando controle preditivo com modelo utilizando MATLAB®/Simulink.

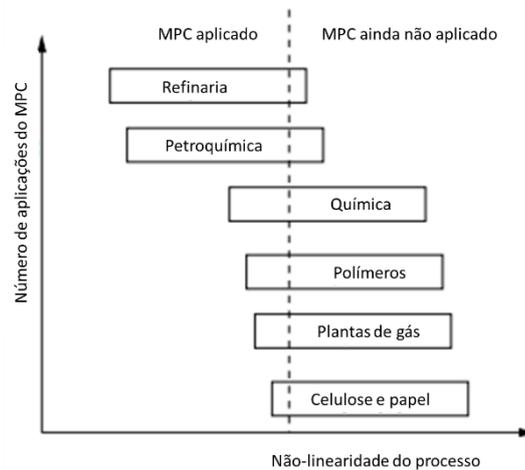
#### 3.3.1 Controle Preditivo com modelo (MPC)

Originalmente desenvolvido nos anos 80 para atender as demandas específicas de controle nas refinarias de petróleo, indústrias petroquímicas e plantas de energia, o MPC passou a ser encontrado em outras indústrias, em grande parte devido ao aumento do poder computacional e barateamento de *hardware*, que possibilitaram a expansão para as indústrias química, de alimentos, automotiva e aeroespacial (QIN; BADGWELL, 2003).

Qin e Badgwell (2003) reportaram que ainda nos anos 90 havia uma tendência no crescimento de áreas e quantidade de aplicações do controle preditivo na indústria, sendo grande parte nas indústrias petroquímicas e refinarias, como mostra a Figura 3. Na época, já

havia estudos visando à implementação na indústria química em geral, destacando-se a indústria de polímeros, celulose e de gases, estas contendo maior não linearidade do sistema.

**Figura 3 – Número de aplicações do MPC em alguns tipos de indústria.**



Fonte: Adaptado de Qin e Badgwell (2000)

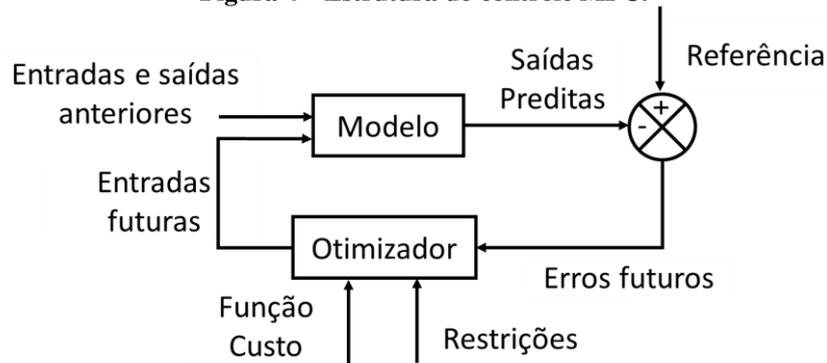
Kumar *et al.* (2018) e Qin e Badgwell (2003) destacaram que há várias razões para se utilizar o MPC, que incluem lidar com sistemas MIMO, com a presença de restrições operacionais e de tempo morto no sistema, além de trabalhar levando em consideração a presença de incertezas no modelo, como: não linearidades do sistema, interações multivariáveis e distúrbios.

O cenário do MPC melhorou significativamente, com um grande aumento do número de aplicações reportadas, melhorias significativas na capacidade técnica, agora contemplando as áreas de refino, indústria química e petroquímica, papel, ar e gases, utilidades, mineração e metalurgia, processamento de alimentos, polímeros e fornos. (QIN; BADGWELL, 2003)

Nos últimos anos percebe-se um acréscimo nas aplicações, dado a grande quantidade de trabalhos desenvolvidos na área, tais como os de Saraiva *et al.* (2020), de Carvalho e Alvarez (2019) e de Sena e Fileti (2019), que estudaram a aplicação do MPC para o processo de produção de ciclopentadieno, no processo de Williams-Otto e controle de pH, respectivamente.

O MPC não é uma estratégia de controle específica, mas um conjunto de métodos que usam um modelo de identificação do sistema para fazer previsões sobre as saídas futuras de um processo (Figura 4). Por meio da minimização de uma função custo, ele atua de modo a selecionar a melhor ação de controle que levará as saídas futuras da variável de interesse tão perto quanto possível da trajetória de referência (*setpoint*) para cada intervalo de tempo (KUMAR *et al.*, 2018; BRAVO; NORMEY-RICO, 2009).

Figura 4 – Estrutura do controle MPC.



Fonte: Adaptado de Camacho e Bordons (2007)

Segundo Camacho e Bordons (2007), a função objetivo leva em conta os erros entre as previsões do modelo e o *setpoint*, além do esforço do controlador, tendo a seguinte forma geral:

$$J = \sum_{j=1}^{N_2} \delta(j) [\hat{y}(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(t+j-1)]^2 \quad (7)$$

Onde  $N_2$  é chamado de horizonte de predição;  $N_u$  é o horizonte de controle;  $\delta$  e  $\lambda$  são os pesos dos erros e do esforço do controlador, respectivamente, ambos são que geralmente assumidos como constantes;  $\hat{y}(t+j|t)$  é a previsão do modelo no instante  $t+j$  dado que o sistema está no instante  $t$ ;  $w(t+j)$  é a referência também no instante  $t+j$ ; e  $\Delta u(t+j-1)$  é o esforço do controlador, isto é, o quanto a variável está sendo manipulada.

Para implementação do MPC é necessário o ajuste dos parâmetros  $N_2$ ,  $N_u$  e dos pesos  $\delta$  e  $\lambda$ , este processo também é conhecido como sintonia do MPC (NERY Jr.,2015). Segundo Camacho e Bordons (2007):

- O horizonte de predição  $N_2$  está relacionado com quantos passos à frente é desejável que a variável controlada alcance a referência;
- O horizonte de predição  $N_u$  é relativo à quantas ações de controle devem ser tomadas para que a variável atinja a referência para o  $N_2$  dado;
- Os pesos  $\delta$  e  $\lambda$  estão vinculados a influência de cada termo da função objetivo e está relacionado diretamente com o tipo de controle alcançado, quando maior for o valor de  $\delta$ , mais rápida será a resposta da variável controlada. Enquanto que o peso  $\lambda$  está relacionado com a suavidade da resposta.

### 3.3.1.1 Modelo de identificação do sistema

Uma etapa determinante no desenvolvimento do MPC é a obtenção do modelo de identificação do sistema, pois é através dele que serão realizadas previsões sobre o comportamento do futuro da planta. O modelo pode ser fenomenológico ou determinístico. No primeiro, o modelo é obtido a partir das leis que governam o sistema (balanço de massa e energia, etc.). O segundo utiliza dados experimentais de entrada e saída para a construção do modelo, que são obtidos pela perturbação da planta e medição da resposta. Este último é útil no caso de sistemas complexos ou que exibem fortes não linearidades, em razão disso, trata-se do método mais utilizado (MALAR; THYAGARAJAN, 2009).

Para um sistema com não-linearidade fraca, uma boa performance pode ser obtida usando um modelo preditivo linear como o ARX (*Auto-Regressive eXogenous model*) e ARMAX (*Auto-Regressive Moving Average eXogenous model*), já para modelos não-linearidade acentuada, um modelo preditivo não-linear com o uso de NARX (*Nonlinear Auto-Regressive eXogenous model*) e NARMARX (*Nonlinear Auto-Regressive Moving Average eXogenous model*) se torna necessário para aproximar a dinâmica do sistema (ZHEJING; DAOYING; YOUXIAN, 2007; VOJTĚŠEK, 2007).

Como a maioria dos processos químicos exibem natureza extremamente não linear, os estudos de NMPC (*Nonlinear Model Predictive Control*) têm despertado maior interesse (ZHEJING; DAOYING; YOUXIAN, 2007; MIAO; WANG, 2002). Esse fato levou à exploração de técnicas de *Machine Learning* como *Support Vector Machine* (SVM), Redes Neurais Artificiais (RNA) e *Fuzzy Model* na identificação de sistemas (SARAIVA *et al.*, 2020).

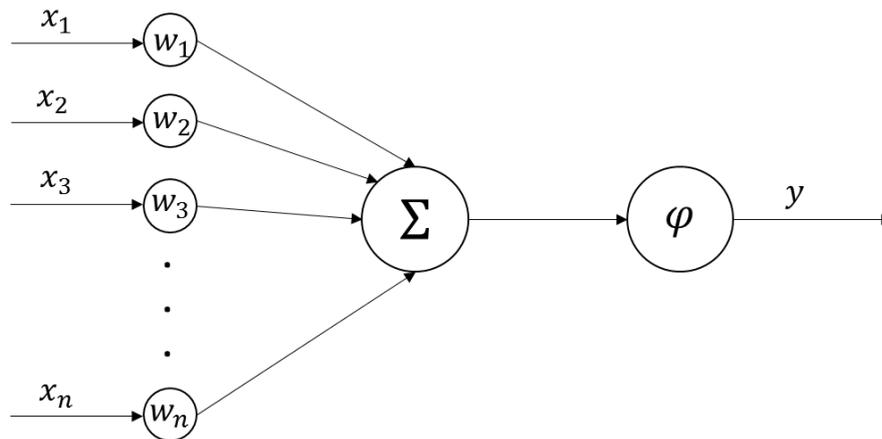
## 3.4 *Machine Learning*

Uma das técnicas mais populares de *Machine Learning* são as redes neurais artificiais (RNA), que consiste em algoritmos baseados no funcionamento do cérebro humano e em como se dá o processo de aprendizado. Para isso, as redes são treinadas por meio de algoritmos de aprendizado para fornecer saídas de acordo com as entradas recebidas. Tudo isso é realizado por meio do ajuste de parâmetros, chamados de pesos e bias, e de uma função de ativação. Juntos eles atuam intermediando a influência do sinal recebido na resposta de cada célula da rede, que é chamada de neurônio artificial ou *perceptron* (HAYKIN, 2001; SARAIVA *et al.*, 2020).

A Figura 5 mostra um esquema de um *perceptron* em que as entradas são representadas pelos  $x_i$ , as saídas pelo  $y$  e os pesos pelos  $w_i$ , enquanto  $\varphi$  se refere à função de ativação, responsável por definir o nível de intensidade da resposta. Expressando isso em notação matemática tem-se:

$$y = \varphi \left( \sum_{i=1}^n x_i w_i \right) \quad (8)$$

Figura 5 – Esquema de um neurônio artificial (*perceptron*).



Fonte: Autor (2020)

Graças a sua não linearidade, adaptabilidade e por promover mapeamento entrada-saída, as redes neurais podem ser consideradas aproximadoras universais de funções, o que as torna perfeitamente adequadas para modelagem de sistemas dinâmicos. É devido a essas características que RNAs são uma ótima alternativa para a construção do modelo utilizado no MPC, como é exposto no trabalho de Saraiva *et al.* (2020) e Assis (2001), que utilizaram com sucesso redes neurais para identificação do processo de Van der Vusse em um CSTR, e em um evaporador de duplo efeito e um CSTR, respectivamente.

### 3.5 Python e suas respectivas bibliotecas

Alguns *softwares*, como o MATLAB®, já oferecem *toolboxes* específicas para a implementação do MPC usando *Machine Learning*, porém se trata de ferramentas pagas. Em contraposição, a linguagem de programação *Python* vem se tornando cada vez mais popular principalmente em aplicações envolvendo *Machine Learning*, graças a um desenvolvimento crescente de novas bibliotecas, além do fato de ser livre e aberta, contanto com uma ampla comunidade. Dessa forma, um dos focos deste trabalho foi também avaliar a existência e

aplicabilidade de bibliotecas gratuitas para a implementação do MPC usando *Machine Learning*.

A biblioteca *Control* apresenta operações essenciais para a análise e *design* de sistemas de controles, já a biblioteca *Pyswarms*, permite trabalhar com PSO. Separadamente, encontra-se bibliotecas específicas para *Machine Learning* em *Python* como o *Scikit-learn*, já que um dos objetivos do trabalho também foi a criação do modelo com *Machine Learning*. Além disso, foram utilizadas outras bibliotecas que fornecem todo o suporte para a implementação computacional. Esse é o caso do *Matplotlib*, que permite plotar gráficos; do *Numpy*, que permite trabalhar com matrizes e vetores; do *Scipy*, que contém rotinas voltadas para o cálculo numérico; e do *Sympy*, que, por sua vez, conta com rotinas para o cálculo simbólico.

### 3.6 O processo controlado

Neste trabalho foi investigado o controle de para um reator do tipo tanque de mistura (CTSR) em que ocorre uma reação com cinética não-linear típica de reações catalíticas em um sistema isotérmico. A reação que será utilizada na implementação das técnicas de controle é a de hidrogenação catalítica do etileno para obtenção de etano (Equações 9 e 10), que teve sua cinética investigada por Sussman e Porrn (1954).



$$r_A = \frac{k_1 C_A}{(1 + k_2 C_A)^2} \quad (10)$$

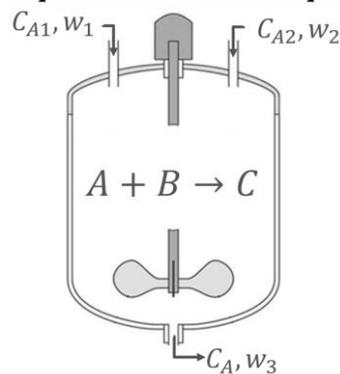
Há uma série de trabalhos que estudaram essa reação em um CSTR. Como o trabalho de Matsuura e Kato (1967), que estudou a estabilidade dessa reação em vários tipos de reatores, o trabalho Li e Biegler (1988), que estudou estratégias de controle para o sistema e a influência causada por restrições nas variáveis, além do trabalho de Brengel e Seider (1989), que formulou um novo algoritmo para o controle preditivo usando múltiplos passos.

## 4. METODOLOGIA

### 4.1 Modelagem do problema

A modelagem do problema foi realizada se baseando no esquema do reator de tanque de mistura (CSTR) estudado nos trabalhos de Li e Biegler (1988) e de Brengel e Seider (1989), mostrado na Figura 6. Os trabalhos fornecem alguns parâmetros do processo, como a concentração e vazão dos reagentes, além da relação da vazão de saída com a altura. Considera-se duas entradas  $w_1$  e  $w_2$  com concentrações  $C_{A1}$  e  $C_{A2}$ , respectivamente com  $C_{A1}$  sendo mais concentrada e  $C_{A2}$  mais diluída. A saída  $w_3$  possui com concentração  $C_A$ .

Figura 6 – Esquema do reator de tanque de mistura



Fonte: Autor (2020)

Para a modelagem do sistema, foram feitas uma série de considerações com o objetivo da simplificação do problema, foram elas:

- Sistema isotérmico, desta forma é possível desconsiderar o balanço de energia;
- Mistura perfeita, de forma que as concentrações dos produtos na corrente de saída sejam iguais as concentrações no seio do reator;
- Densidades e capacidades caloríficas dos componentes constantes ao longo do curso da reação.

#### 4.1.1 Balanço de Massa Global

$$\textit{Entrada} - \textit{Saída} + \textit{Geração} = \textit{Acúmulo} \quad (11)$$

$$w_1 + w_2 - w_3 = \frac{dV}{dt} \quad (12)$$

Considerando que  $w_3 = 0.2\sqrt{h}$  e que a seção transversal tenha área constante igual a 1, ou seja,  $A = 1 \text{ m}^2$

$$w_1 + w_2 - 0.2\sqrt{h} = \frac{dh}{dt} \quad (13)$$

$$\frac{dh}{dt} = w_1 + w_2 - 0.2\sqrt{h} \quad (14)$$

4.1.2 Balanço de Massa para o etileno

$$\mathbf{Entrada} - \mathbf{Saída} + \mathbf{Geração} = \mathbf{Acúmulo} \quad (15)$$

$$w_1 C_{A1} + w_2 C_{A2} - w_3 C_A - rV = \frac{dN_A}{dt} \quad (16)$$

A cinética dada por Sussman e Porm (1954) é

$$r_A = \frac{k_1 C_A}{(1 + k_2 C_A)^2} \quad (17)$$

e sabendo que:

$$C_A = \frac{N_A}{V} \quad (18)$$

$$V = A \cdot h = h \quad (19)$$

$$N_A = h \cdot C_A \quad (20)$$

assim tem-se:

$$w_1 C_{A1} + w_2 C_{A2} - 0.2\sqrt{h} C_A - \frac{k_1 C_A V}{(1 + k_2 C_A)^2} = h \frac{dC_A}{dt} + C_A \frac{dh}{dt} \quad (21)$$

$$\frac{dC_A}{dt} = \frac{w_1(C_{A1} - C_A)}{h} + \frac{w_2(C_{A2} - C_A)}{h} - \frac{k_1 C_A}{(1 + k_2 C_A)^2} \quad (22)$$

E com isso, o sistema estudado pode ser descrito pelo sistema de equação diferenciais ordinárias (EDOs) dado abaixo:

$$\begin{cases} \frac{dh}{dt} = w_1 + w_2 - 0.2\sqrt{h} \\ \frac{dC_A}{dt} = \frac{w_1(C_{A1} - C_A)}{h} + \frac{w_2(C_{A2} - C_A)}{h} - \frac{k_1 C_A}{(1 + k_2 C_A)^2} \end{cases} \quad (23)$$

## 4.2 Análise do estado estacionário e escolha do ponto de operação

Para realizar o estudo do estado estacionário foi feito  $\frac{dC_{As}}{dt} = \frac{dh_s}{dt} = 0$ , já que o sistema não varia com o tempo nessa situação. Aplicando isso no sistema de EDOs dados pela Equação 23, obtêm-se um sistema de equações não lineares, dado pela Equação 24.

$$\begin{cases} w_{1s} + w_{2s} - 0.2\sqrt{h_s} = 0 \\ \frac{w_{1s}(C_{A1} - C_{As})}{h_s} + \frac{w_{2s}(C_{A2} - C_{As})}{h_s} - \frac{k_1 C_{As}}{(1 + k_2 C_{As})^2} = 0 \end{cases} \quad (24)$$

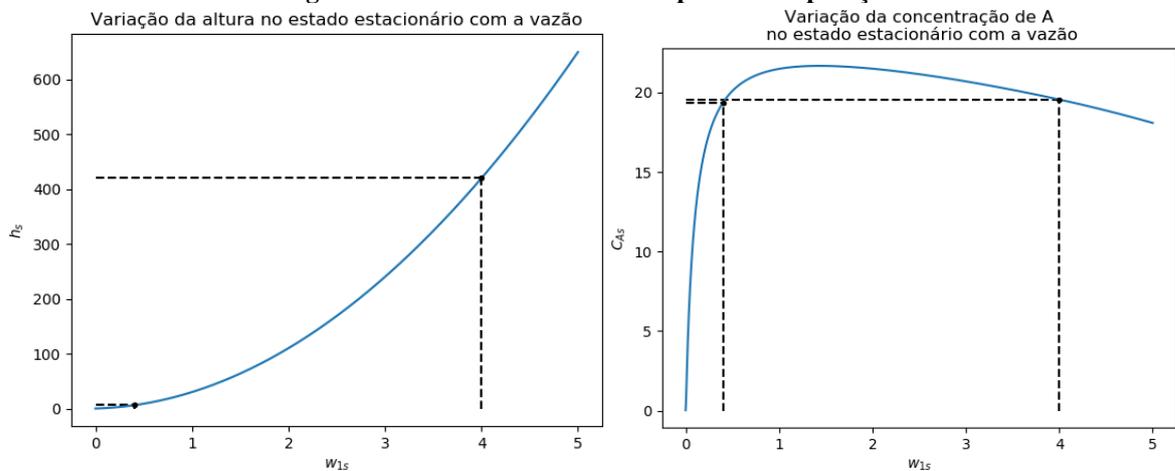
Tomando os parâmetros dados por Brengel e Seider (1989), como mostrados na Tabela 2, e resolvendo o sistema de equações não lineares para valores de  $w_{1s}$  entre 0 e 5 L/s foi possível obter gráficos com valores de  $C_{As}$  e  $h_s$  de acordo com a vazão (Figura 7).

**Tabela 2 – Parâmetros do sistema considerados no estudo.**

Variável	Valor
$k_1$	$1,0 \text{ s}^{-1}$
$k_2$	$1,0 \text{ L/mol}$
$C_{A1}$	$24,9 \text{ mol/L}$
$C_{A2}$	$0,1 \text{ mol/L}$
$w_2$	$0,1 \text{ L/s}$

Fonte: Brengel e Seider (1989)

**Figura 7 – Estados estacionários e pontos de operação.**



Fonte: Autor (2020)

O sistema foi estudado para dois pontos de operação. Como é possível observar na Figura 3, ambos pontos possuem  $C_{AS}$  próximo, mas com as vazões diferentes. Os pontos estudados são dados na Tabela 3.

Embora possuam  $C_{AS}$  próximo, os pontos apresentam um comportamento diferente com relação a uma pequena variação no valor de  $w_{1S}$ . Enquanto no estado estacionário 1, um aumento de  $w_{1S}$  esteja associado a um aumento de  $C_{AS}$ , no estado estacionário 2, o aumento está relacionado a uma diminuição de  $C_{AS}$ , o que fez com que esses pontos fossem escolhidos para serem estudados.

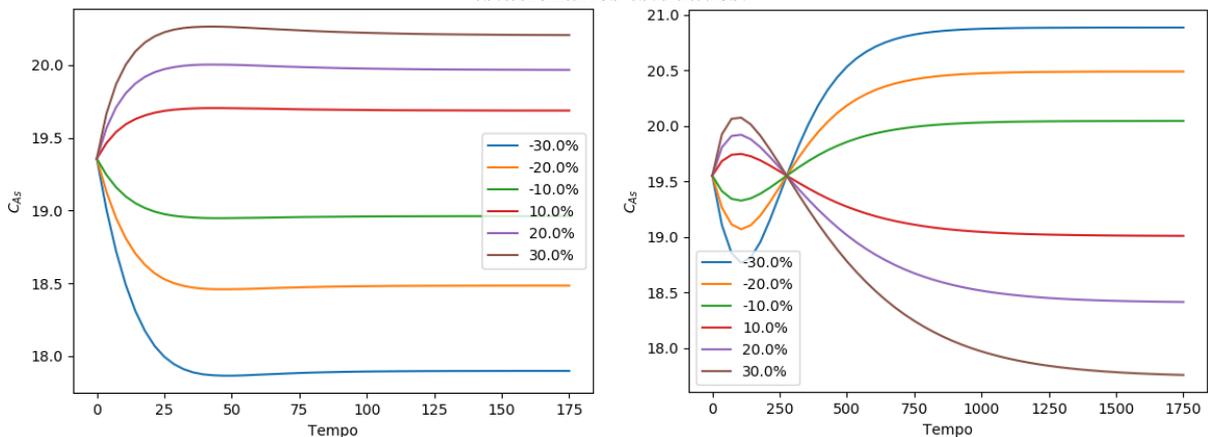
**Tabela 3 – Estados estacionários do sistema que foram estudados.**

Variável	Estado estacionário 1	Estado estacionário 2
$w_{1S}$	0,4	4,0
$h_S$	6,25	420,25
$C_{AS}$	19,3561	19,5500

### 4.3 Análise do estado transiente do sistema e sua resposta a perturbações

Para a análise do estado transiente, o sistema de EDOs foi resolvido usando Runge-Kutta de quarta ordem. Partindo dos dois estados estacionários, ou seja, definindo  $C_A(0) = C_{AS}$  e  $h(0) = h_S$  para  $w_1(0) = w_{1S}$ , foram realizadas perturbações degrau de  $\pm 10\%$ ,  $\pm 20\%$  e  $\pm 30\%$  na vazão  $w_{1S}$ . Como resultado obteve-se os gráficos apresentados na Figura 8, que correspondem aos estados estacionários 1 e 2, respectivamente.

**Figura 8 – Respostas da concentração para perturbações degrau na vazão em torno dos dois estados estacionários estudados.**



Fonte: Autor (2020)

Do gráfico da Figura 8.a, que corresponde a perturbações em torno do estado estacionário 1, pode-se perceber o caráter não linear do processo, além da diferença da magnitude das respostas para perturbações de mesma intensidade, mas de sinal contrário, percebe-se que a diferença entre o espaçamento das respostas não se mantém constante, apesar das perturbações serem igualmente espaçadas. Como era esperado, uma perturbação positiva está associada à um aumento da resposta e vice-versa.

O gráfico para as respostas às perturbações em torno do estado estacionário 2, apresentado na Figura 8.b, mostra um comportamento não linear assim como o primeiro, além de no começo apresentar um comportamento que é o inverso do esperado. Nesse caso, era esperado que uma aumento na vazão provocasse uma diminuição na concentração e vice-versa. Além disso, o sistema apresenta uma resposta mais lenta, levando cerca de dez vezes mais tempo para atingir o estado estacionário se comparado ao primeiro caso.

#### 4.4 Controle Clássico

##### 4.4.1 Obtenção da função de transferência

Para implementar o controle clássico foi necessário obter a função de transferência (FT) que representa o sistema, isso foi feito a partir das EDOs que descrevem o sistema. Para obter a função de transferência é necessário calcular as transformadas de Laplace das equações diferenciais que descrevem o sistema, mas como o sistema apresenta equações diferenciais não lineares, foi necessário linearizar o sistema em torno de um ponto de operação.

Para simplificar os cálculos, as variáveis desvio foram definidas seguinte forma:

$$\begin{cases} x_1 = h - h_s \\ x_2 = C_A - C_{A_s} \\ u = w_1 - w_{1s} \end{cases} \quad (25)$$

e as EDOs foram renomeadas da seguinte maneira:

$$\begin{cases} \frac{dh}{dt} = f_1(h, C_A, w_1) = w_1 + w_2 - 0.2\sqrt{h} \\ \frac{dC_A}{dt} = f_2(h, C_A, w_1) = \frac{w_1(C_{A1} - C_A)}{h} + \frac{w_2(C_{A2} - C_A)}{h} - \frac{k_1 C_A}{(1 + k_2 C_A)^2} \end{cases} \quad (26)$$

A linearização foi feita de acordo com a notação de espaço de estados:

$$\dot{X} = A \cdot X + B \cdot u \quad (27)$$

onde:

$$\dot{X} = \begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \end{bmatrix} \quad (28)$$

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (29)$$

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial h} & \frac{\partial f_1}{\partial C_b} \\ \frac{\partial f_2}{\partial h} & \frac{\partial f_2}{\partial C_b} \end{bmatrix}_{\substack{h=h_s \\ C_A=C_{A_s} \\ w_1=w_{1_s}}} \quad (30)$$

$$B = \begin{bmatrix} \frac{\partial f_1}{\partial w_1} \\ \frac{\partial f_2}{\partial w_1} \end{bmatrix}_{\substack{h=h_s \\ C_A=C_{A_s} \\ w_1=w_{1_s}}} \quad (31)$$

Calculando as derivadas parciais das matrizes A e B, obtêm-se:

$$A = \begin{bmatrix} \frac{-0.1}{\sqrt{h_s}} & 0 \\ -\frac{w_{1s}(24.9 - C_{A_s}) - 0.1C_{A_s} + 0.01}{h_s^2} & \frac{2C_{A_s}}{(C_{A_s} + 1)^3} - \frac{1}{(C_{A_s} + 1)^2} - \left( \frac{w_{1s} + 0.1}{h_s} \right) \end{bmatrix} \quad (32)$$

$$B = \begin{bmatrix} 1 \\ \frac{24.9 - C_{A_s}}{h_s} \end{bmatrix} \quad (33)$$

Para o sistema de controle que foi implementado, se deseja controlar a concentração de A na saída ( $C_A$ ) manipulando a vazão da corrente de alimentação concentrada ( $w_1$ ). Dessa forma foi obtida uma função de transferência para malha aberta que relacione  $X_2(s)$  com  $U(s)$ , que são as variáveis desvio para  $C_A$  e  $w_1$ , respectivamente. Aplicando os parâmetros dados e calculando nas condições dos estados estacionários 1 e 2, obtêm-se, respectivamente:

$$\begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \end{bmatrix} = \begin{bmatrix} -0,04 & 0 \\ -0,0075 & -0,0778 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0,8870 \end{bmatrix} u \quad (34)$$

$$\begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \end{bmatrix} = \begin{bmatrix} -0,0049 & 0 \\ -0,0001 & -0,0076 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0,0127 \end{bmatrix} u \quad (35)$$

Ou ainda, reescrevendo os novos sistemas de EDO lineares:

$$\begin{cases} \frac{dx_1}{dt} = -0,04x_1 + u \\ \frac{dx_2}{dt} = -0,0075x_1 - 0,0778x_2 + 0,8870u \end{cases} \quad (36)$$

$$\begin{cases} \frac{dx_1}{dt} = -0,0049x_1 + u \\ \frac{dx_2}{dt} = -0,0001x_1 - 0,0076x_2 + 0,0127u \end{cases} \quad (37)$$

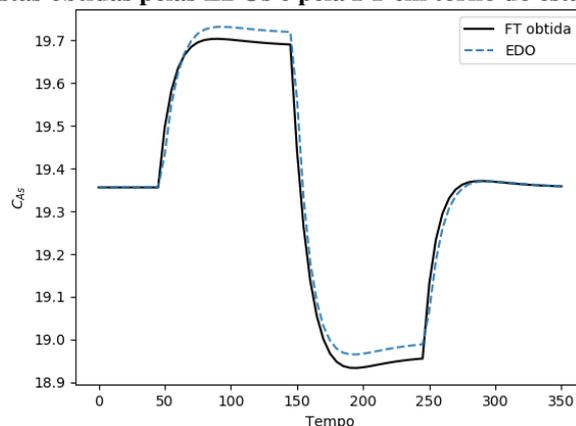
Calculando a Transformada de Laplace e simplificando as equações para obter a função de transferência tendo  $U$  como entrada e  $X_2$  como saída, em que  $G_1(s)$  é a função de transferência para o sistema em torno do ponto de operação 1 (Equação 38) e  $G_2(s)$  é a função de transferência para o sistema em torno do ponto de operação 2 (Equação 39)

$$G_1(s) = \frac{X_2(s)}{U(s)} = \frac{0,8870s + 0,0280}{s^2 + 0,1178s + 0,0031} \quad (38)$$

$$G_2(s) = \frac{X_2(s)}{U(s)} = \frac{0,0127s - 3,777 \cdot 10^{-5}}{s^2 + 0,0125s + 3,724 \cdot 10^{-5}} \quad (39)$$

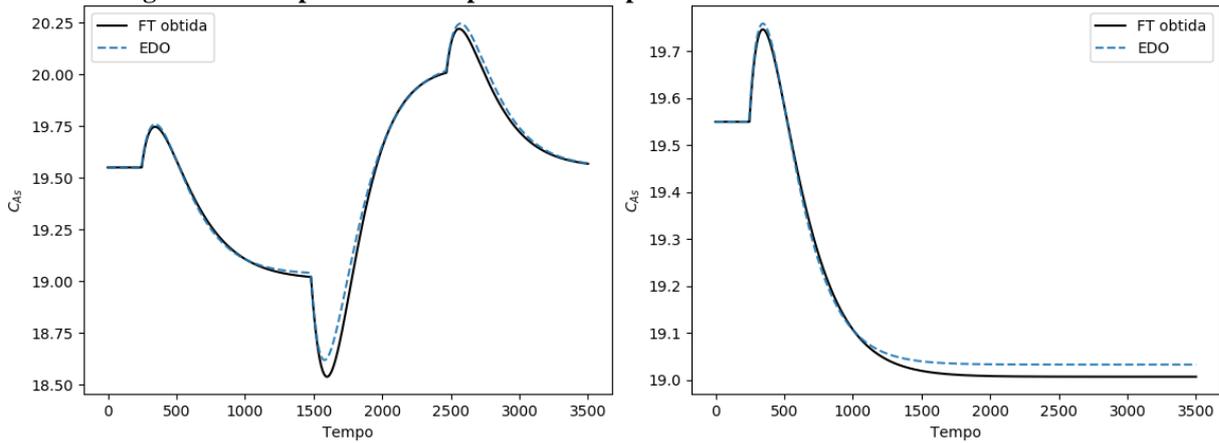
A comparação entre a resposta da função de transferência obtida e do sistema EDOs para o primeiro estado estacionário é mostrada na Figura 9 e para o segundo estado estacionário é mostrada nas Figura 10. Ambas respostas apresentaram comportamento bem próximo ao das EDOs, o que garante que são adequadas para realizar o estudo sobre o controle no sistema.

**Figura 9 – Respostas obtidas pelas EDOs e pela FT em torno do estado estacionário 1.**



Fonte: Autor (2020)

**Figura 10 – Respostas obtidas pelas EDOs e pela FT em torno do estado estacionário 1.**



Fonte: Autor (2020)

#### 4.4.2 Estabilidade da malha fechada

A análise de estabilidade foi feita a partir do critério de Routh, conforme é mostrado por Coughanowr e Leblanc (2009). Para a equação característica do sistema em malha fechada. A equação é dada por:

$$1 + G(s) = 0 \quad (40)$$

Onde  $G(s) = G_c(s) \cdot G_p(s)$ ,  $G_c(s)$  é a função de transferência do controlador e  $G_p(s)$  é a função de transferência do sistema. Nesse caso, será investigado o caso em que  $G_c(s) = K$ . Assim:

$$1 + KG_c(s) = 0 \quad (41)$$

substituindo, para o primeiro estado estacionário:

$$s^2 + (0,1178 + 0,8870K)s + 0,0031 + 0,0280K = 0 \quad (42)$$

assim a matriz de Routh, toma a seguinte forma:

$$\begin{bmatrix} 1 & 0,0031 + 0,0280K \\ 0,1178 + 0,8870K & 0 \\ 0,0031 + 0,0280K & \end{bmatrix} \quad (43)$$

Para o sistema ser estável, os termos da primeira coluna têm que ser positivos, ou seja,  $0,1178 + 0,8870K > 0$  e  $0,0031 + 0,0280K > 0$ , logo:  $K > -0,1107$ . Substituindo, a função de transferência do segundo estado estacionário, na equação característica, obtém-se:

$$s^2 + (0,0125 + 0,0127K)s + (3,724 - 3,777K) \cdot 10^{-5} = 0 \quad (44)$$

A matriz de Routh, toma a seguinte forma

$$\begin{bmatrix} 1 & (3,724 - 3,777K) \cdot 10^{-5} \\ 0,0125 + 0,0127K & 0 \\ (3,724 - 3,777K) \cdot 10^{-5} & \end{bmatrix} \quad (45)$$

e para o sistema ser estável  $0,0125 + 0,0127K > 0$  e  $3,724 - 3,777K > 0$ , logo:

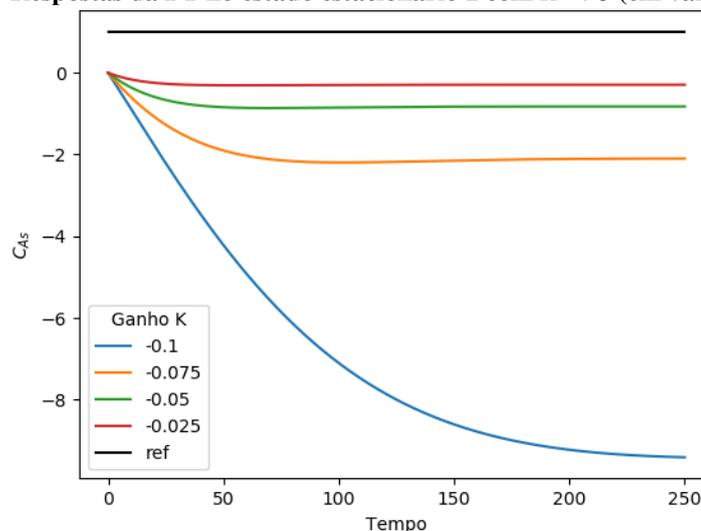
$$-0,9842 < K < 0,9860 \quad (46)$$

#### 4.4.3 Lugar das raízes

Antes de plotar o gráfico do lugar das raízes, é interessante analisar o comportamento da resposta em malha fechada com controlador proporcional à uma perturbação degrau no *setpoint*. Para isso, serão considerados os valores de  $K$  positivos e negativos dentro do intervalo de estabilidade encontrado a partir da análise de Routh.

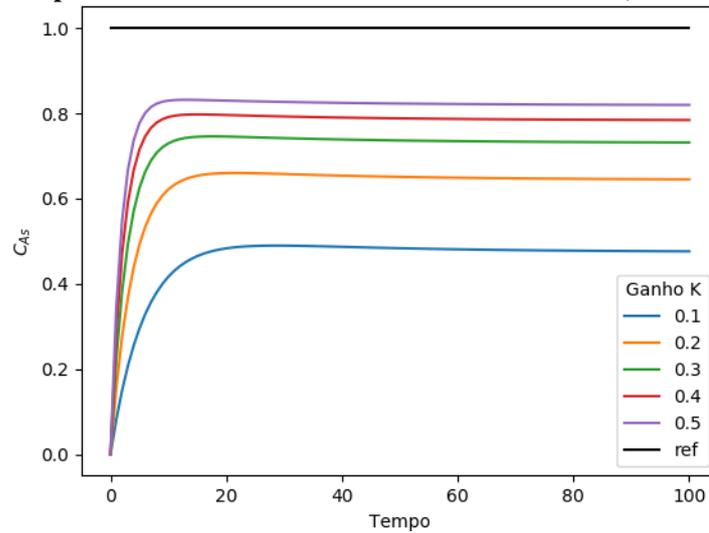
Para o primeiro estado estacionário, as respostas para vários  $K < 0$  pode ser vista na Figura 11. Nesse caso, o sistema se afasta do *setpoint* à medida que  $K$  aumenta em valores absolutos. Já na Figura 12, quando é investigado o caso em que  $K > 0$ , percebe-se que o aumento do  $K$  está associado a diminuição do *off-set*. Em ambos os casos, as respostas obtidas são estáveis, como previsto na análise de Routh.

**Figura 11 – Respostas da FT no estado estacionário 1 com  $K < 0$  (em variável desvio).**



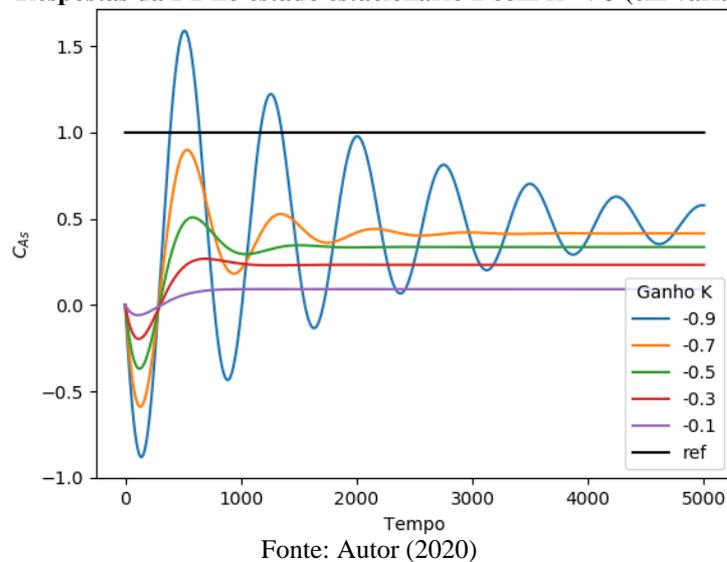
Fonte: Autor (2020)

**Figura 12 – Respostas da FT no estado estacionário 1 com  $K > 0$  (em variável desvio).**

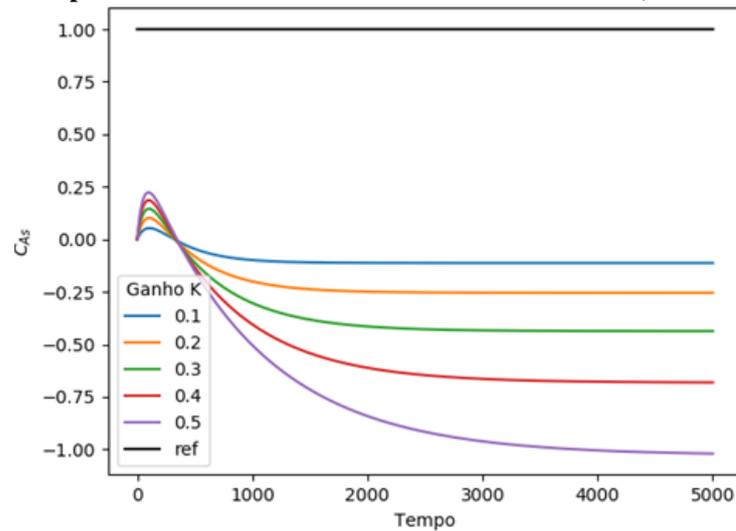


A mesma análise também é feita para o segundo estado estacionário. A Figura 13 mostra o caso em que  $K < 0$ , percebe-se que à medida que  $K$  aumenta em valores absolutos, o sistema passa a ter resposta oscilatória, apresentando uma tendência de se aproximar do *setpoint*. Quando  $K > 0$ , a Figura 14 mostra que o *off-set* aumenta com o aumento de  $K$ . Assim como no caso do primeiro estado estacionário, as respostas observadas são estáveis, já que estão dentro do intervalo de estabilidade obtidos pela análise de Routh.

**Figura 13 – Respostas da FT no estado estacionário 2 com  $K < 0$  (em variável desvio).**



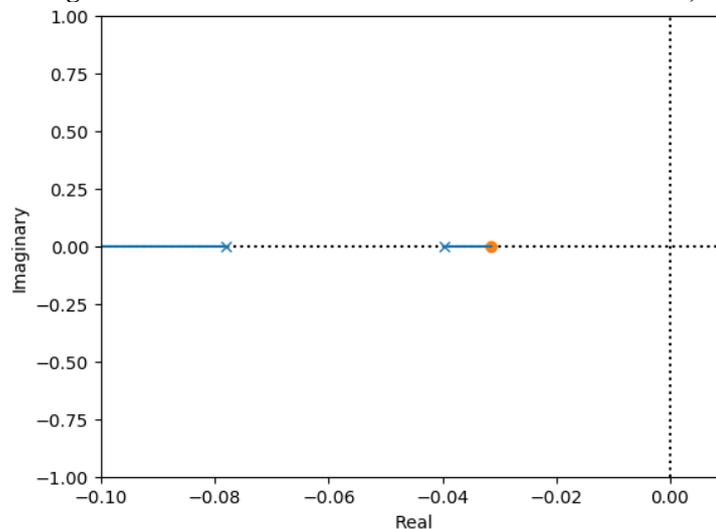
**Figura 14 – Respostas da FT no estado estacionário 2 com  $K > 0$  (em variável desvio).**



Fonte: Autor (2020)

Desse modo, o lugar das raízes para a função de transferência linearizada em torno do estado estacionário 1 (Figura 15), foi obtido para valores positivos de  $K$ . O gráfico mostra que o sistema se mantém sempre estável, já que só está contido no lado esquerdo do plano complexo. Além disso, como se mantém sempre sobre o eixo real, ou seja, não possui parte complexa, a resposta do sistema à uma perturbação degrau não apresenta resposta oscilatória para nenhum ganho proporcional  $K$  positivo. Esse sistema apresenta dois polos, um em  $p_1 = -0,0076$  e outro em  $p_2 = -0,0049$ , além disso, sistema tem um zero em  $z_1 = 0,0030$ .

**Figura 15 – Lugar das raízes da FT em torno do estado estacionário 1, com  $K > 0$ .**



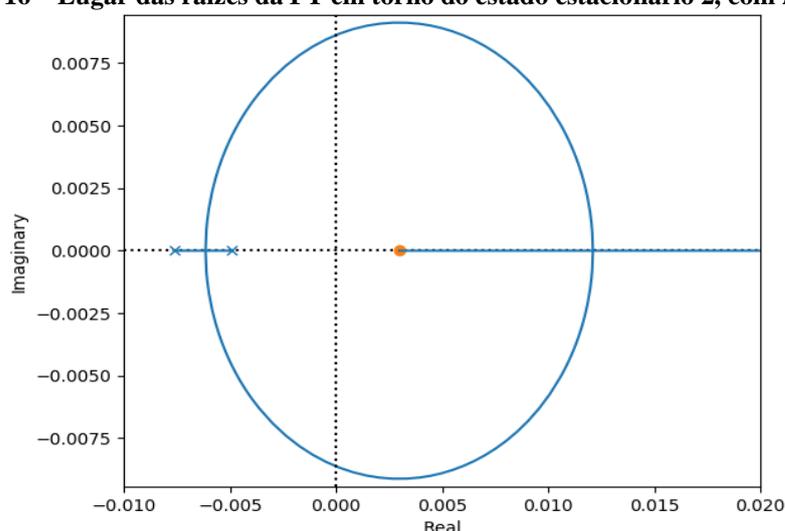
Fonte: Autor (2020)

No caso da função de transferência que foi linearizada em torno do estado estacionário 2, o gráfico do lugar das raízes (Figura 16) foi obtido para valores negativos de  $K$ . O gráfico

mostra que sistema tem tanto uma parte estável quanto uma instável, pois está presente tanto do lado esquerdo quanto direito. O ganho crítico  $K_{cr}$ , que é ganho em que é cruzado o eixo imaginário e o sistema passa de estável para instável, corresponde à  $-0,9842$ , que é o mesmo limite inferior para estabilidade que foi obtido pela análise de Routh.

Para valores pequenos em módulo de  $K$  o sistema apresenta comportamento não-oscilatório, depois passa a ser oscilatório, mas ainda estável, isso pode ser visto na Figura 13. Após ultrapassar  $K_{cr}$ , o sistema ainda continua oscilatório, mas instável. Por fim, deixa de ser oscilatório, continuando instável. O sistema apresenta dois polos:  $p_1 = -0.0076$  e  $p_2 = -0.0049$ ; e um zero:  $z_1 = 0.0030$ .

**Figura 16 – Lugar das raízes da FT em torno do estado estacionário 2, com  $K < 0$ .**



Fonte: Autor (2020)

#### 4.5 Implementação e sintonia do controlador PID

O lugar das raízes da função de transferência em torno do estado estacionário 1 não cruza o eixo imaginário, como visto na Figura 15. Dessa forma, não há um ganho crítico  $K_{cr}$  e, sendo assim, não é possível utilizar o método de Ziegler-Nichols. Dessa forma, foram utilizadas diretamente as técnicas de sintonia baseadas na minimização da IAE, ISE, ITAE e ITSE com PSO. A Tabela 4, mostra quais os intervalos de busca escolhidos para a busca por cada tipo de controle, o intervalo para  $K_p$  foi escolhido com levando em consideração o esforço do controlador que passa ser alto quando  $K_p$  é grande.

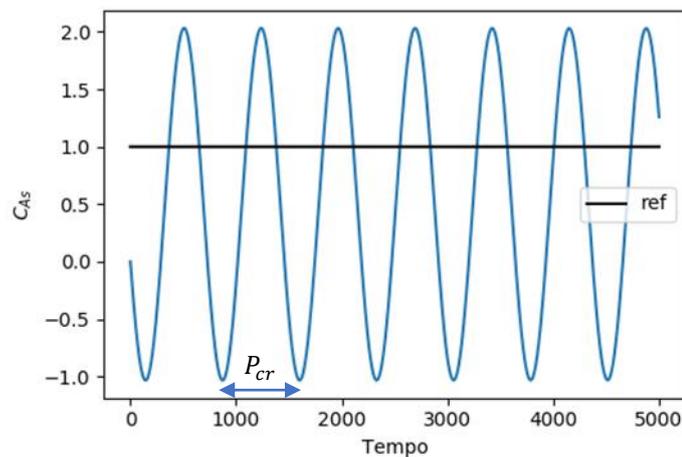
**Tabela 4 – Intervalos de busca utilizados na sintonia do controlador para a FT do estado estacionário 1.**

Controlador	Intervalo para $K_p$	Intervalo para $K_i$	Intervalo para $K_d$
P	[0;2,5]	-	-
PI	[0;2,5]	[-10;10]	-
PID	[0;2,5]	[-10;10]	[-10;10]

O lugar das raízes da função de transferência em torno do estado estacionário 2 cruza o eixo imaginário, como é visto na Figura 16. Sendo assim, também foi utilizado o método de Ziegler-Nichols. A malha fechada apresenta um ganho crítico  $K_{cr}$  igual à -0,9842 e um período crítico  $P_{cr}$  de 728, como mostrado na Figura 17. O sistema também foi sintonizado pela minimização IAE, ISE, ITAE e ITSE. A Tabela 5 mostra os intervalos de busca escolhidos para cada controlador, o intervalo para  $K_p$  foi escolhido com valores próximos ao intervalo de estabilidade para esse obtido pelo método de Routh. Para avaliar a eficiência das sintonias obtidas, além da IAE, ISE, ITAE e ITSE, foram considerados o tempo de ascensão e o sobressinal.

**Tabela 5 – Intervalos de busca utilizados na sintonia do controlador para a FT do estado estacionário 2.**

Controlador	Intervalo para $K_p$	Intervalo para $K_i$	Intervalo para $K_d$
P	[-1;1]	-	-
PI	[-1;1]	[-10;10]	-
PID	[-1;1]	[-10;10]	[-10;-10]

**Figura 17 – Resposta da malha fechada do estado estacionário 2 com  $K_{cr}$  (em variável desvio).**

Fonte: Autor (2020)

## 4.6 Controle Preditivo baseado em modelo (MPC)

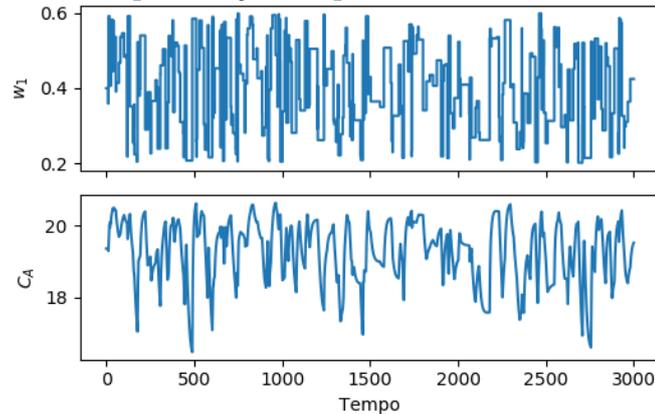
### 4.6.1 Geração de dados e treinamento da rede

Para o treinamento da rede foram gerados de dados de entrada e saída por meio da simulação com as EDOs com perturbações na vazão  $w_1$  em torno dos estados estacionários e obtenção das respostas de  $C_A$  (Figuras). Os intervalos de perturbação e de tempo utilizados são mostrados na Tabela 6, em ambos os casos foram utilizado um tempo de amostragem de 1 segundo. Os perfis de perturbação e resposta são para o estado estacionário 1 e 2 são mostrados nas Figuras 18 e 19, respectivamente.

**Tabela 6 – Intervalos de perturbação da vazão  $w_1$  e intervalo de tempo utilizado para geração de dados.**

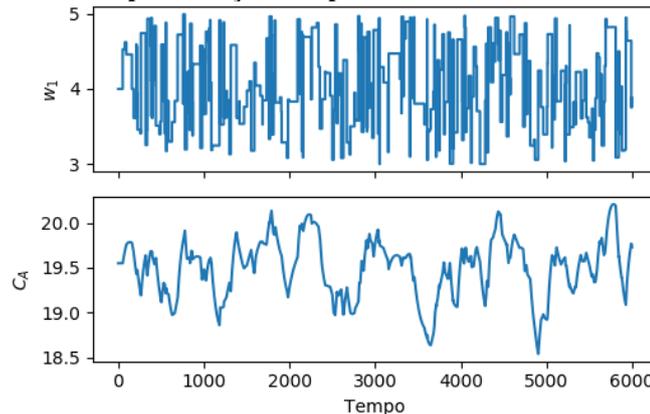
Estado estacionário	Intervalo de perturbação $w_1$ (L/s)	Intervalo de tempo (s)
1	[0.2;0.6]	6000
2	[3.0;4.0]	6000

**Figura 18 – Perfil de perturbação e resposta em torno do estado estacionário 1.**



Fonte: Autor (2020)

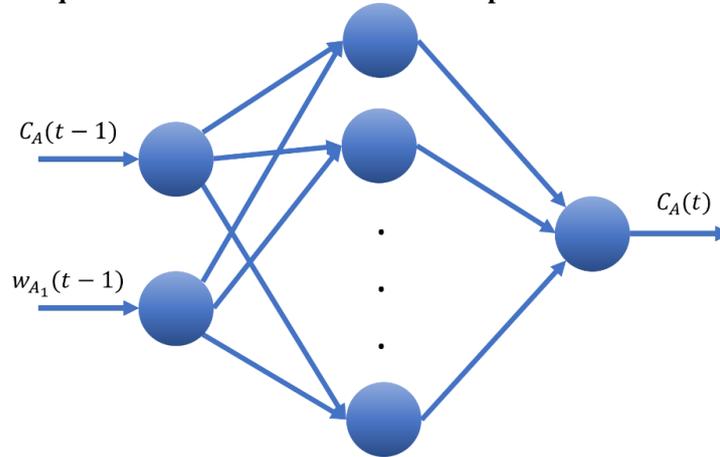
**Figura 19 – Perfil de perturbação e resposta em torno do estado estacionário 2.**



Fonte: Autor (2020)

Para a etapa de identificação do sistema, foram utilizadas as redes neurais MLP (*Multilayer Perceptron*), com apenas uma camada intermediária, usando a concentração de A na saída do reator e a vazão na entrada 1 no instante passado para encontrar a concentração atual de A na saída do reator, como é mostrado na Figura 20

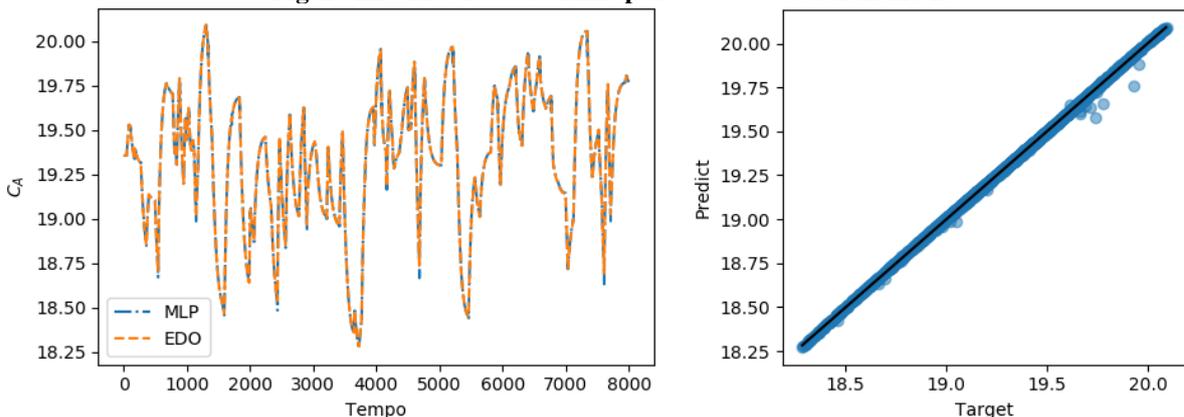
**Figura 20 – Esquemas de entradas e saída utilizadas para o treinamento da rede neural.**



Fonte: Autor (2020)

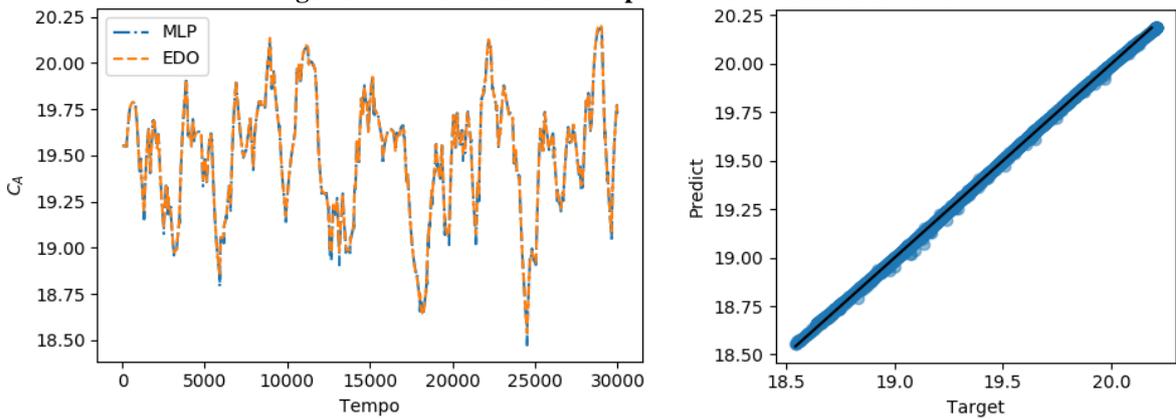
O conjunto de dados gerado foi dividido em 70% para treinamento e 30% para testes, além disso passam por uma normalização. As métricas utilizadas para avaliar a rede foram o coeficiente de determinação  $R^2$  e a raiz quadrada do erro médio quadrático RMSE, a rede foi construída utilizando a biblioteca *scikit-learn*. Os resultados podem ser vistos nas Figuras 21 e 22 que mostram as previsões realizadas pelas redes para o estado estacionário 1 e 2. Além disso, as redes treinadas foram testadas para outras perturbações típicas de controle e comparadas com a resposta gerada pelas EDOs (Figura 23). Em ambos os casos, tanto as métricas quanto os gráficos se mostraram bastante satisfatórios com pode ser visto na Tabela 7.

**Figura 21 – Previsões da MLP para o estado estacionário 1.**



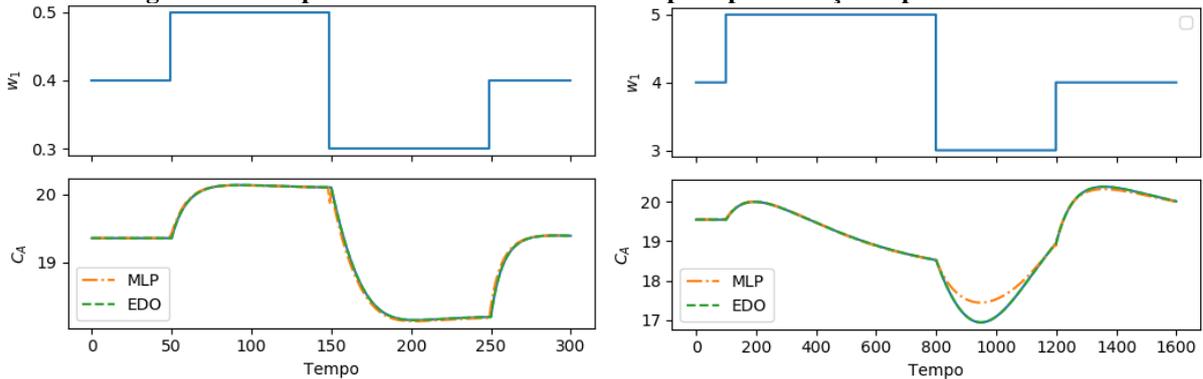
Fonte: Autor (2020)

**Figura 22 – Previsões da MLP para o estado estacionário 2.**



Fonte: Autor (2020)

**Figura 23 – Respostas das duas redes treinadas para perturbações típicas de controle.**



Fonte: Autor (2020)

**Tabela 7 – Métricas do desempenho das redes neurais treinadas.**

Estado estacionário	Dados de Teste		Perturbação Típica	
	$R^2$	RMSE	$R^2$	RMSE
1	0,9997	0,0069	0,9982	0,0309
2	0,9997	0,0056	0,9765	0,1566

#### 4.6.2 Implementação e Sintonia do MPC

Para a implementação do MPC, assim como foi feito no trabalho de Nery (2015), recorreu-se ao PSO para a minimização da função objetivo na busca as ações de controle em cada instante de tempo. Para a sintonia MPC foram utilizados o horizonte de predição  $N_2$ , além dos pesos dos erros ( $\delta$ ) e do esforço do controlador ( $\lambda$ ), a seleção desses parâmetros foi feita de acordo com a Tabela 8. Cenários com variantes para o horizonte de controle  $N_u$  não foram estudados nesse trabalho, que foi mantido fixo em 2, o aumento de  $N_u$  está diretamente

relacionado ao aumento do custo computacional, pois insere no problema de otimização mais variáveis. Para cada  $N_2$  foram testadas combinações dos tipos de pesos.

**Tabela 8 –Valores de busca para os parâmetros de sintonia do MPC.**

Parâmetros	Valores de busca
Horizonte de predição	5 e 7
Peso dos erros	0,1; 1 e 10
Peso dos esforços	0,1; 1 e 10

## 5. RESULTADOS E DISCUSSÃO

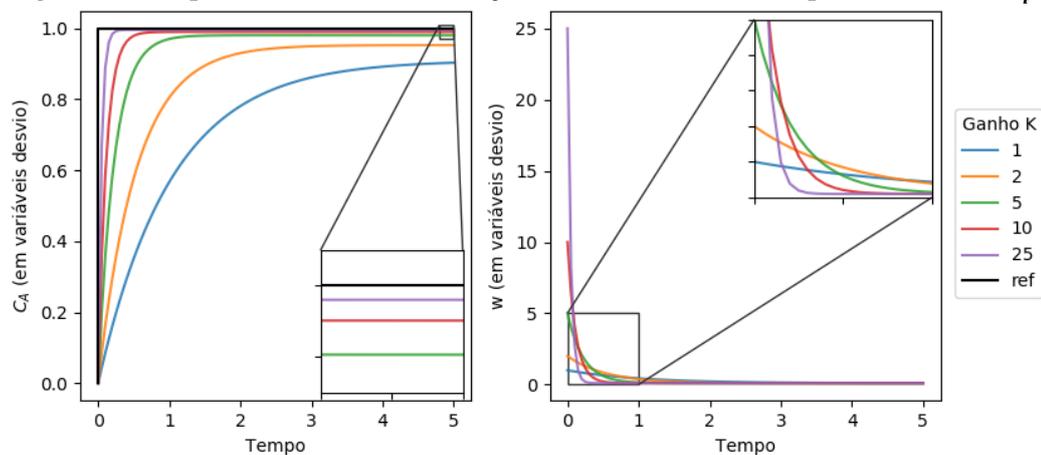
Os resultados serão mostrados começando pelas análises dos controladores clássicos para os estados estacionários 1 e 2 de modo a permitir, a escolha do melhor controlador para cada caso. Logo após, serão realizadas as análises para o MPC, com a análise da influência dos seus parâmetros para o controle. Por fim, serão comparados os controladores MPC e clássico.

### 5.1 Controle Clássico – Estado estacionário 1

#### 5.1.1 Análise do controlador P

Como visto anteriormente, um aumento do valor  $K_p$  está associado a uma diminuição do *off-set*. Nesse caso especificamente, como visto pelo lugar das raízes, esse aumento não está relacionado com uma instabilidade do sistema. Apesar disso, o aumento indiscriminado do valor de  $K_p$  está relacionado a um aumento do esforço do controlador, como é mostrado na Figura 24.

Figura 24 – Respostas do sistema e o esforço do controlador causados pelo aumento de  $K_p$ .



Fonte: Autor (2020)

Dessa forma fica claro que o problema não pode ser otimizado nesse caso, pois o *off-set* e, conseqüentemente, o erro serão minimizados conforme  $K_p$  cresce, tendendo a zero quando  $K_p$  tende a infinito. Em contrapartida, o esforço do controlador também aumenta proporcionalmente.

#### 5.1.2 Análise do controlador PI

Como os critérios IAE, ISE, ITAE e ISTE foram usados na otimização, eles não serão considerados na análise do controlador. Como pode ser visto na Tabela 9 e na Figura 25, as minimizações da ISE e ITAE levaram as respostas a elevados sobressinais, em contraposição

de quando foram minimizadas IAE e ITSE. Isso pode ser visto também pelos ganhos obtidos, dispostos na tabela 4, enquanto que os ganhos proporcionais  $K_p$  se mantiveram próximos, os ganhos integrais obtidos foram elevados justamente nas minimizações da ISE e ITAE, o que justifica o elevado sobressinal e o menor tempo de ascensão ( $T_a$ ). Sendo assim, tanto a sintonia pela minimização do IAE quanto pela minimização do ITSE apresentaram resultados bastante similares e satisfatórios.

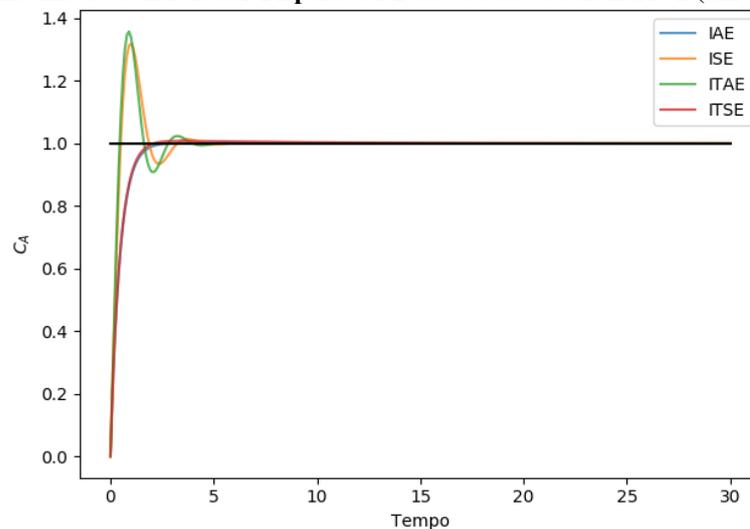
**Tabela 9 – Critérios de desempenho calculados para o controlador PI no estado estacionário 1.**

Técnica utilizada	IAE	ISE	ITAE	ITSE	Sobressinal (%)	Ta (s)
min IAE	<b>4,6547</b>	2,2664	4,6345	0,5128	<b>0,14</b>	2,67
min ISE	5,5705	<b>2,1735</b>	5,2100	0,9158	30,98	<b>0,44</b>
min ITAE	5,6154	<b>2,1735</b>	<b>5,1439</b>	0,9224	35,78	<b>0,44</b>
min ITSE	4,9889	2,2390	7,1031	<b>0,5104</b>	0,82	2,00

**Tabela 10 – Ganhos calculados para o controlador PI no estado estacionário 1.**

Técnica utilizada	$K_p$	$K_I$	$K_D$
min IAE	2,4818	0,2284	-
min ISE	2,4985	7,3910	-
min ITAE	2,4977	9,5108	-
min ITSE	2,4949	0,2817	-

**Figura 25 – Sintonia do controlador PI para a FT do estado estacionário 1 (em variável desvio).**



Fonte: Autor (2020)

### 5.1.3 Análise do controlador PID

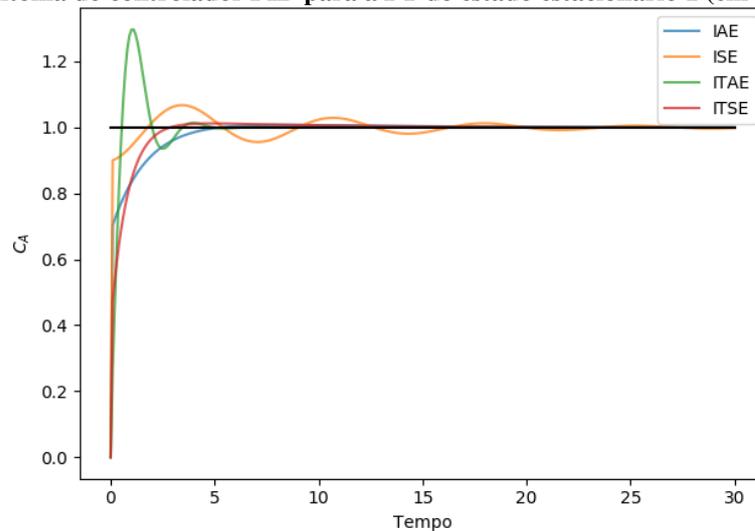
Os resultados para o controlador PID são mostrados na Tabela 11 e na Figura 26. As sintonias baseadas na ITAE e ISE levaram a sobressinais bastante elevados, enquanto que no caso do primeiro obteve-se uma resposta rápida e que após 5 segundos se estabiliza, a segunda teve uma resposta mais lenta e oscilatória. Ao analisar os ganhos calculados, mostrados na Tabela 12, é possível notar que isso se deu por um ganho derivativo  $K_d$  mais alto.

Por outro lado, as sintonias baseadas na IAE e ITSE apresentaram pouco sobressinal, se destacando a segunda por ter o menor sobressinal e ter um tempo de ascensão menor que a primeira. Apesar disso, a performance geral do controlador PID foi pior do que a do PI, o que pode indicar que a presença do ganho derivativo tenha afetado negativamente sistema.

**Tabela 11 – Critérios de desempenho calculados para o controlador PID no estado estacionário 1.**

Técnica utilizada	IAE	ISE	ITAE	ITSE	Sobressinal (%)	Ta (s)
min IAE	<b>5,1018</b>	0,7620	13,9744	0,5855	2,42	5,28
min ISE	5,4012	<b>0,2234</b>	41,3927	0,9311	66,18	1,78
min ITAE	5,6348	2,0077	<b>5,7239</b>	0,9248	32,25	<b>0,54</b>
min ITSE	5,1806	1,3860	9,3900	<b>0,5456</b>	<b>1,36</b>	3,23

**Figura 26 – Sintonia do controlador PID para a FT do estado estacionário 1 (em variável desvio).**



Fonte: Autor (2020)

**Tabela 12 – Ganhos calculados para o controlador PID no estado estacionário 1.**

Técnica utilizada	$K_P$	$K_I$	$K_D$
min <i>IAE</i>	2,4548	0,2787	2,4489
min <i>ISE</i>	2,4839	8,4738	9,9823
min <i>ITAE</i>	2,4843	7,0408	0,1001
min <i>ITSE</i>	2,4548	0,2787	0,7586

## 5.2 Controle Clássico – Estado estacionário 2

### 5.2.1 Análise do controlador P

Os resultados dos critérios de desempenho para a sintonia do controlador P são mostradas na Tabela 13, os ganhos calculados estão dispostos na Tabela 14 e as respostas na Figura 27. As respostas das sintonias por otimização apresentaram elevado sobressinal, quando comparados com a sintonia por Ziegler-Nichols, que mesmo assim, já é bastante elevado. Isso demonstra a incapacidade do controlador P ter um desempenho satisfatório. Todas as respostas apresentaram *offsets* que, nesse caso, pela dinâmica do sistema, não podiam ser minimizados pelo aumento do ganho  $K_P$ .

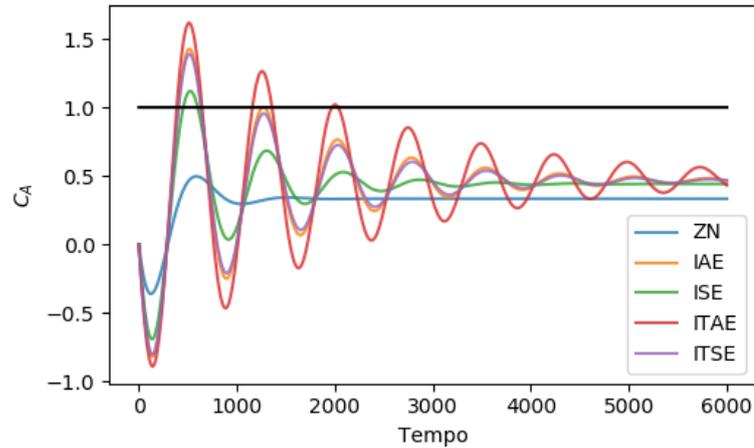
**Tabela 13 – Critérios de desempenho calculados para o controlador P no estado estacionário 2.**

Técnica utilizada	IAE	ISE	ITAE	ITSE	Sobressinal (%)	Ta (s)
Ziegler-Nichols	$4,15 \cdot 10^3$	$2,98 \cdot 10^3$	$1,20 \cdot 10^7$	$8,04 \cdot 10^6$	<b>49,17</b>	418,7
min <i>IAE</i>	<b><math>3,50 \cdot 10^3</math></b>	$2,58 \cdot 10^3$	$9,66 \cdot 10^6$	<b><math>5,54 \cdot 10^6</math></b>	205,85	337,2
min <i>ISE</i>	$3,54 \cdot 10^3$	<b><math>2,45 \cdot 10^3</math></b>	$1,01 \cdot 10^7$	$5,77 \cdot 10^6$	153,32	349,7
min <i>ITAE</i>	$3,57 \cdot 10^3$	$2,92 \cdot 10^3$	<b><math>9,53 \cdot 10^6</math></b>	$5,91 \cdot 10^6$	236,72	<b>331,5</b>
min <i>ITSE</i>	<b><math>3,50 \cdot 10^3</math></b>	$2,55 \cdot 10^3$	$9,70 \cdot 10^6$	<b><math>5,54 \cdot 10^6</math></b>	199,84	338,4

**Tabela 14 – Ganhos calculados para o controlador P no estado estacionário 2.**

Técnica utilizada	$K_P$	$K_I$	$K_D$
Ziegler-Nichols	-0,4921	-	-
min <i>IAE</i>	-0,8622	-	-
min <i>ISE</i>	-0,7770	-	-
min <i>ITAE</i>	-0,9057	-	-
min <i>ITSE</i>	-0,8528	-	-

**Figura 27 – Sintonia do controlador P para a FT do estado estacionário 2 (em variável desvio).**



Fonte: Autor (2020)

### 5.2.2 Análise do controlador PI

Ao observar a Tabela 15 com os critérios de desempenho sintonia do controlador PI e a Figura 28 é possível notar que a sintonia por Ziegler- Nichols teve os piores resultados, embora não tenha apresentado sobressinal, isso resultou em um tempo de ascensão ( $T_a$ ) bastante lento, o que a torna desinteressante. Como a dinâmica do processo é bastante lenta, a sintonia pela minimização do IAE se torna bastante interessante, pois tem uma resposta relativamente rápida e não ser tão oscilatória quanto à por sintonia pela minimização do ITSE.

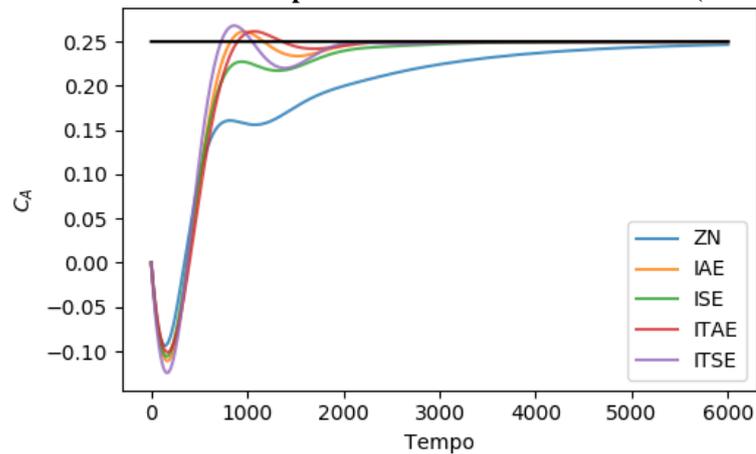
**Tabela 15 – Critérios de desempenho calculados para o controlador PI no estado estacionário 2.**

Técnica utilizada	IAE	ISE	ITAE	ITSE	Sobressinal (%)	$T_a$ (s)
Ziegler-Nichols	$1,38 \cdot 10^3$	$8,60 \cdot 10^2$	$1,73 \cdot 10^6$	$4,34 \cdot 10^5$	<b>0,00</b>	10653
min IAE	<b><math>7,24 \cdot 10^2</math></b>	$7,61 \cdot 10^2$	$2,65 \cdot 10^5$	$1,78 \cdot 10^5$	4,63	826,1
min ISE	$8,22 \cdot 10^2$	<b><math>7,49 \cdot 10^2</math></b>	$4,27 \cdot 10^5$	$1,87 \cdot 10^5$	<b>0,00</b>	4692
min ITAE	$7,31 \cdot 10^2$	$7,63 \cdot 10^2$	<b><math>2,55 \cdot 10^5</math></b>	$1,86 \cdot 10^5$	4,70	897,6
min ITSE	$7,34 \cdot 10^2$	$7,67 \cdot 10^2$	$2,98 \cdot 10^5$	<b><math>1,74 \cdot 10^5</math></b>	7,32	<b>727,4</b>

**Tabela 16 – Ganhos calculados para o controlador PI no estado estacionário 2.**

Técnica utilizada	$K_P$	$K_I$	$K_D$
Ziegler-Nichols	-0,4429	-0,0007	-
min IAE	-0,4423	-0,0014	-
min ISE	-0,4450	-0,0012	-
min ITAE	-0,4020	-0,0014	-
min ITSE	-0,4937	-0,0014	-

Figura 28 – Sintonia do controlador PI para a FT do estado estacionário 2 (em variável desvio).



Fonte: Autor (2020)

### 5.2.3 Análise do controlador PID

Para esse caso, a sintonia por Ziegler-Nichols mostrou um desempenho superior na maioria dos critérios de desempenho como é mostrado na Tabela 17. Apesar disso como é possível ver na Figura 29 apresentou uma resposta extremamente lenta, além de apresentar uma resposta que se moveu na direção contrária ao *setpoint* de forma brusca logo no início da perturbação.

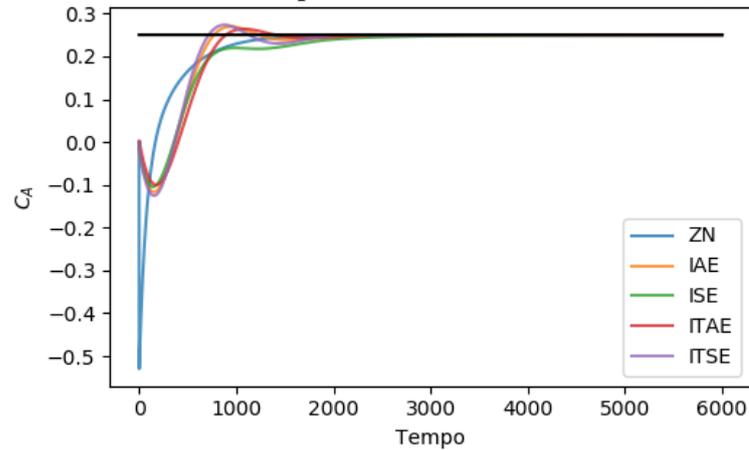
Tabela 17 – Critérios de desempenho calculados para o controlador PID no estado estacionário 2.

Técnica utilizada	IAE	ISE	ITAE	ITSE	Sobressinal (%)	Ta (s)
Ziegler-Nichols	<b><math>6,16 \cdot 10^2</math></b>	<b><math>7,14 \cdot 10^2</math></b>	<b><math>1,93 \cdot 10^5</math></b>	<b><math>8,00 \cdot 10^3</math></b>	<b>0,00</b>	2617
min IAE	<u><math>7,03 \cdot 10^2</math></u>	$7,54 \cdot 10^2$	$2,63 \cdot 10^5$	$1,65 \cdot 10^5$	7,46	770,1
min ISE	$8,21 \cdot 10^2$	<u><math>7,34 \cdot 10^2</math></u>	$4,26 \cdot 10^5$	$1,80 \cdot 10^5$	<b>0,00</b>	4607
min ITAE	$7,40 \cdot 10^2$	$7,69 \cdot 10^2$	<u><math>2,38 \cdot 10^5</math></u>	$1,90 \cdot 10^5$	5,72	893,7
min ITSE	$7,06 \cdot 10^2$	$7,60 \cdot 10^2$	$2,51 \cdot 10^5$	<u><math>1,64 \cdot 10^5</math></u>	8,82	<b>717,0</b>

Tabela 18 – Ganhos calculados para o controlador PID no estado estacionário 2.

Técnica utilizada	$K_P$	$K_I$	$K_D$
Ziegler-Nichols	-0,5905	-0,0016	-53,7373
min IAE	-0,4599	-0,0015	-4,9746
min ISE	-0,4438	-0,0012	-4,7701
min ITAE	-0,3957	-0,0014	1,2803
min ITSE	-0,4907	-0,0015	-4,5393

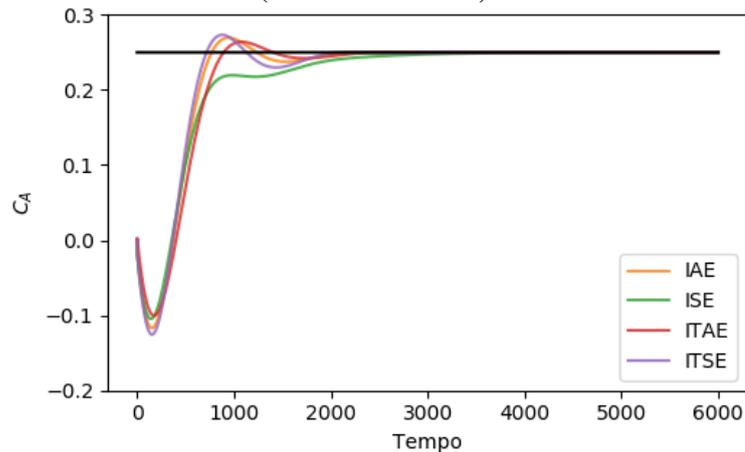
**Figura 29 – Sintonia do controlador PID para a FT do estado estacionário 2 (em variável desvio).**



Fonte: Autor (2020)

Na Figura 30, é mostrada somente as respostas das sintonias por otimização. Nota-se que os resultados foram bastante parecidos com o do controlador PI (Figura 28), mesmo contando com ganhos derivativos significativos, como é mostrado na Tabela 18. Assim como caso do PI, a sintonia que se mostra mais interessante é a obtida pela minimização do IAE, por ter uma resposta rápida e ao mesmo tempo não ser tão oscilatória como a obtida pela minimização ITSE.

**Figura 30 – Resposta das sintonias por otimização do controlador PID para a FT do estado estacionário 2 (em variável desvio).**



Fonte: Autor (2020)

### 5.3 MPC – Estado estacionário 1

Para a análise do MPC no estado estacionário 1, a influência de dos pesos  $\delta$  e  $\lambda$  foram estudados para a resposta do sistema a uma perturbação degrau, simulando uma mudança de *setpoint* partindo do estado estacionário 1. A Tabela 19 e a Figura 31 mostram a primeira análise, que foi feita com  $N_c = 2$  e fixando  $N_u = 5$ . É possível notar que quando  $\lambda = 0,1$  as

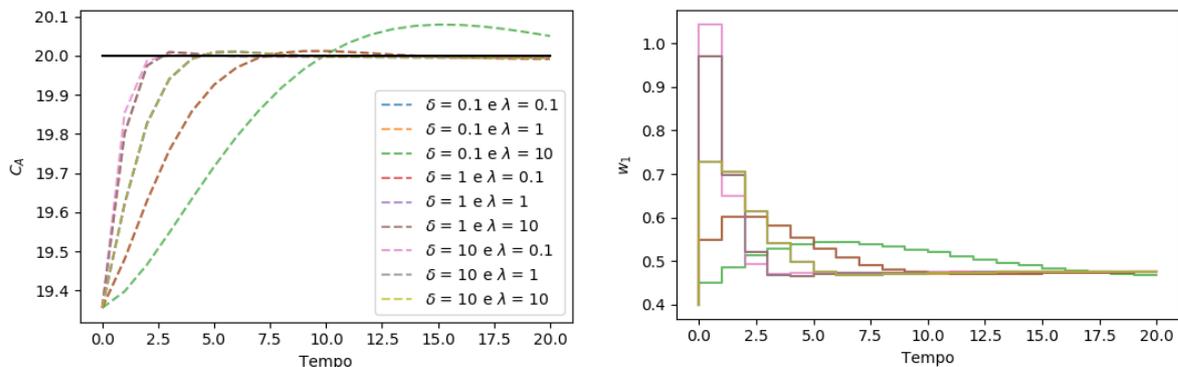
métricas foram melhores para cada um dos valores de  $\delta$ , o que quer dizer que uma menor penalização do esforço do controlador levou a melhores soluções.

Como pode ser visto na figura 31, quando  $\delta = 10$  e  $\lambda = 0,1$  o controlador obteve a resposta mais rápida e com maior esforço do controlador, só que nem sempre isso é o ideal, pois um maior esforço do controlador está relacionado a um maior desgaste dos equipamentos. Fazendo a análise inversa, quando  $\delta = 0,1$  e  $\lambda = 10$ , como era esperado o sistema obteve a resposta mais lenta e com as piores métricas. Um comportamento intermediário ocorreu quando  $\delta = \lambda$ . Como era esperado, nesse caso o gráfico apresenta respostas sobrepostas, além de métricas bastante parecidas.

**Tabela 19 – Influência dos pesos na métricas avaliadas na resposta em torno do estado estacionário 1 quando do  $N_c = 2$  e  $N_u = 5$ .**

$\delta$	$\lambda$	IAE	ISE	ITAE	ITSE
0,1	0,1	<b>0,9895</b>	<b>0,3538</b>	<b>1,8203</b>	<b>0,2488</b>
	1	1,7781	0,6904	4,2486	0,8801
	10	3,6446	1,3814	18,7827	3,6276
1	0,1	<b>0,5439</b>	<b>0,1911</b>	<b>0,8037</b>	<b>0,0547</b>
	1	0,9902	0,3540	1,8259	0,2491
	10	1,7777	0,6890	4,2613	0,8774
10	0,1	<b>0,4449</b>	<b>0,1668</b>	<b>0,5720</b>	<b>0,0295</b>
	1	0,5457	0,1910	0,8215	0,0547
	10	0,9912	0,3540	1,8404	0,2492

**Figura 31 – Influência dos pesos na resposta em torno do estado estacionário 1 quando  $N_c = 2$  e  $N_u = 5$ .**



Fonte: Autor (2020)

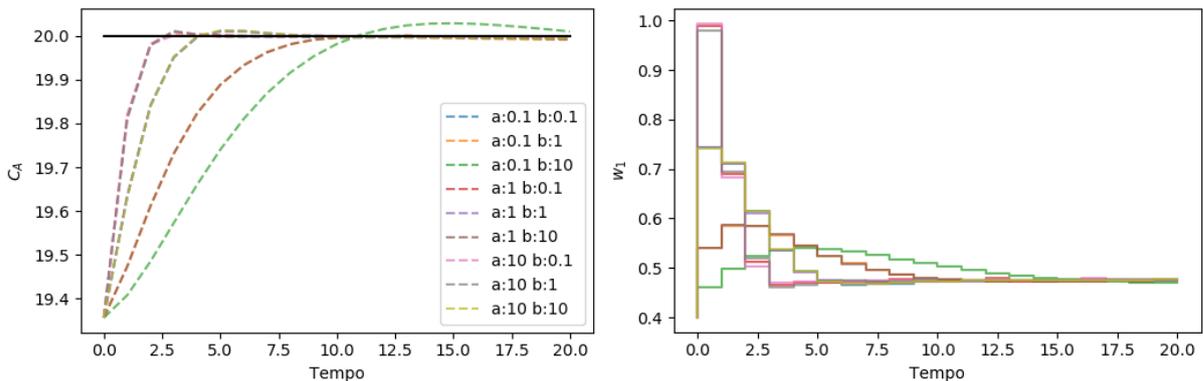
Fazendo uma análise similar ao cenário anterior, quando  $N_c = 2$  e fixou-se  $N_u = 7$ , também se obteve as melhores métricas quando  $\lambda = 0,1$ , para cada um dos valores de  $\delta$ , como

pode ser visto na Tabela 20. Fazendo a comparação entre os dois cenários, nota-se a performance do controlador foi aumentada, como pode ser visto ao comparar as métricas entre si. Ao observar o gráfico, nota-se que o esforço do controlador quando  $\delta = 10$  e  $\lambda = 0,1$  apresentou uma leve redução passando a ser menor que 1,0.

**Tabela 20 – Influência dos pesos na métricas avaliadas na resposta em torno do estado estacionário 1 quando do  $N_c = 2$  e  $N_u = 7$ .**

$\delta$	$\lambda$	IAE	ISE	ITAE	ITSE
0,1	0,1	<b>0,9502</b>	<b>0,3378</b>	<b>1,7415</b>	<b>0,2268</b>
	1	1,9696	0,7490	5,0881	1,0668
	10	3,1182	1,2524	11,9811	2,7527
1	0,1	<b>0,5229</b>	<b>0,1852</b>	<b>0,7613</b>	<b>0,0486</b>
	1	0,9489	0,3377	1,7248	0,2265
	10	1,9756	0,7488	5,1691	1,0673
10	0,1	<b>0,4453</b>	<b>0,1663</b>	<b>0,6022</b>	<b>0,0290</b>
	1	0,5246	0,1852	0,7604	0,0486
	10	0,9532	0,3376	1,7828	0,2268

**Figura 32 – Influência dos pesos na resposta em torno do estado estacionário 1 quando  $N_c = 2$  e  $N_u = 7$ .**



Fonte: Autor (2020)

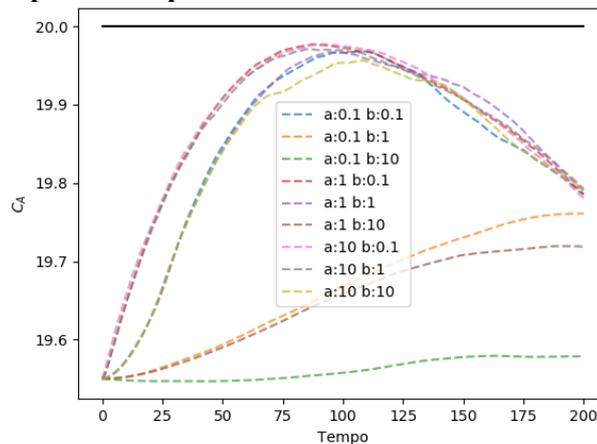
#### 5.4 MPC – Estado estacionário 2

Assim como no estado estacionário 1, para a análise do MPC estudou-se a influência de dos pesos  $\delta$  e  $\lambda$  na resposta a uma perturbação. As Figuras 33 e 34 mostram a resposta do sistema, quando  $N_u = 5$  e  $N_u = 7$ , respectivamente, ambas com  $N_c = 2$ . Ao analisar as duas figuras nota-se que o desempenho do controlador não foi satisfatório para nenhum caso dentro dos dois cenários, ambas exibem um comportamento bastante parecido. Para  $\delta = 0,1$  e  $\lambda = 10$ ;

$\delta = 0,1$  e  $\lambda = 1$ ; e  $\delta = 1$  e  $\lambda = 10$ ; a resposta apresenta um comportamento lento e não se aproxima do *setpoint*. Nos outros casos, a resposta se aproxima rapidamente do *setpoint*, mas após atingir um pico, volta a decrescer sem tocar no *setpoint*.

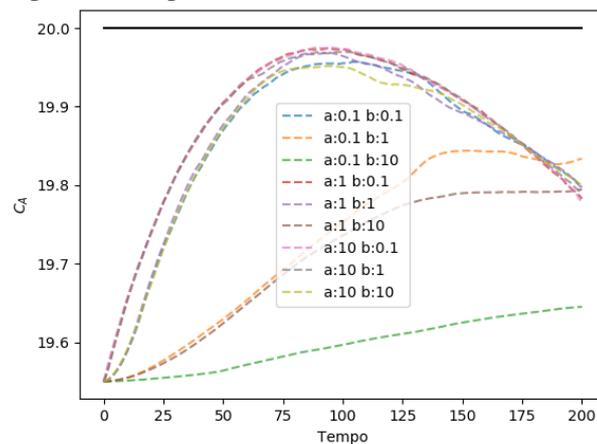
As causas para esse problema podem estar no desempenho das previsões realizadas pela rede, já que o MPC depende fortemente da acurácia com que a rede MLP é capaz de fazer previsões sobre o comportamento futuro da planta. Outros fatores que poderiam ser investigados, mas que fogem do escopo do trabalho, seria a utilização de técnicas de *Deep Learning* como LSTM (*Long-Short Term Memory*) que são destinadas a em identificação em séries temporais como é o caso do problema estudado, já que se deseja realizar previsões sobre o comportamento futuro da planta, baseados nas entradas e saídas anteriores. Além disso poderiam ser investigados a influência variação do tempo de amostragem e horizonte de controle do MPC.

**Figura 33 – Influência dos pesos na resposta em torno do estado estacionário 2 quando  $N_c = 2$  e  $N_u = 5$ .**



Fonte: Autor (2020)

**Figura 34 – Influência dos pesos na resposta em torno do estado estacionário 2 quando  $N_c = 2$  e  $N_u = 7$ .**



Fonte: Autor (2020)

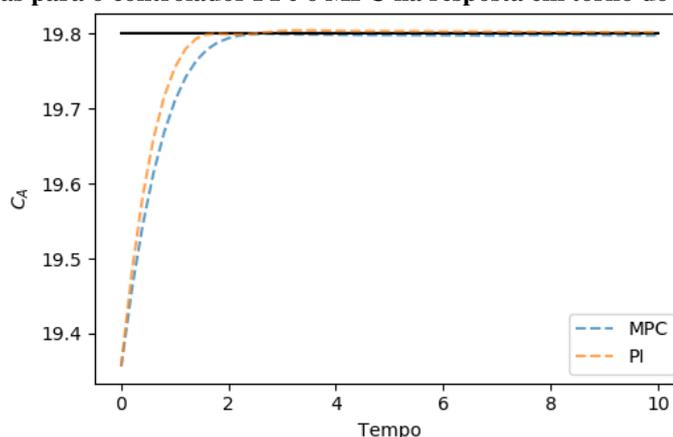
## 5.5 Comparação entre as estratégias de controle implementadas

Para realizar a comparação entre o controle clássico e o MPC, foram escolhidas as melhores sintonias obtidas por cada estratégia. Como não se obteve sucesso na implantação do MPC em torno do estado estacionário 2, a análise só foi realizada para o caso do estado estacionário 1. Conforme foi visto, o melhor do desempenho do controle clássico foi para um controlador PI com sintonia realizada pela minimização do ITSE. Já a para o MPC, a comparação foi feita utilizando um horizonte de controle  $N_c = 2$  e um horizonte de predição  $N_u = 5$ , além de pesos  $\delta = 10$  e  $\lambda = 0,1$ . O resultado dessas comparações está disposto na Tabela 21 e na Figura 35.

**Tabela 21 – Métricas para o controlador PI e o MPC na resposta em torno do estado estacionário 1.**

Técnica utilizada	IAE	ISE	ITAE	ITSE
PI (min IAE)	<b>0,1817</b>	<b>0,0971</b>	<b>0,0160</b>	<b><math>1,681 \cdot 10^{-3}</math></b>
MPC ( $N_c = 2, N_u = 5, \delta = 10$ e $\lambda = 0,1$ )	0,4499	0,1668	0,5720	0,0295

**Figura 35 – Respostas para o controlador PI e o MPC na resposta em torno do estado estacionário 1.**



Fonte: Autor (2020)

A Tabela 21 mostra que todas as métricas do controlador PI foram superiores às do MPC, embora tenham apresentado resultados próximos. No gráfico da Figura 35, é possível ver que a resposta do controlador PI foi mais rápida que a do MPC para alcançar o *setpoint*.

Esses resultados podem ser explicados pelo reduzido espaço de busca, tanto do horizonte de predição quanto os pesos da função objetivo, utilizado na sintonia do MPC. É válido ressaltar também que não foram investigadas as influências do horizonte de controle e tempo de amostragem. Dessa forma, o resultado do MPC pode ser considerado satisfatório dentro do escopo do trabalho.

## 6. CONCLUSÃO

No presente trabalho, foram estudadas as técnicas de controle clássico e de controle preditivo com modelo (MPC), este que faz parte do chamado controle moderno. As técnicas foram aplicadas à reação de hidrólise catalítica do etileno em um reator CSTR. Este sistema foi escolhido por contar com não linearidades típicas de uma reação complexa.

A partir da análise preliminar do estado estacionários, escolheu-se dois pontos de operação para o estudo do problema servo aplicado às técnicas de controle. Devido a multiplicidade de entrada do sistema, esses dois pontos de operação apresentavam comportamento diferentes mesmo com concentrações próximas. No estado estacionário 1, o sistema apresenta comportamento mais trivial, enquanto que no segundo, verificou-se um comportamento mais complexo com uma dinâmica mais lenta. A escolha de duas regiões de operação com comportamento distintos permitiu avaliar o desempenho e a limitações dos dois tipos de controles implementados.

No primeiro estado estacionário, os resultados tanto por controle clássico quanto por MPC conseguiram atingir o *setpoint*. Ao realizar a comparação utilizando as duas técnicas, o controlador PI com sintonia realizada pela minimização do ITSE apresentou desempenho superior ao MPC com horizonte de controle  $N_c = 2$ , horizonte de predição  $N_u = 5$  e pesos  $\delta = 10$  e  $\lambda = 0,1$ . No segundo estado estacionário o desempenho do controle clássico foi bastante lento, mas alcançou o *setpoint*, já o MPC, não conseguiu controlar o processo. Em parte, isso pode ter acontecido devido à necessidade de um refinamento na rede neural que foi utilizada para realizar previsões sobre o comportamento futuro da planta ou até mesmo a investigação de outras técnicas de *Machine Learning* mais poderosas. Além disso, nos dois estados estacionários estudados caberia também uma análise mais aprofundada sobre a influência os parâmetros do MPC.

Foi avaliado também a aplicabilidade da linguagem de programação *Python* e suas bibliotecas para modelagem e simulação de reatores, na criação de redes neurais artificiais para a construção de modelos para MPC, além da implementação das duas técnicas de controle citadas. O Python se mostrou uma poderosa ferramenta, com potencial para atender as demandas dentro do controle de processos.

Deste modo, o presente trabalho atingiu os objetivos propostos, sendo possível aplicar possível consolidar dos conhecimentos adquiridos durante a graduação nas áreas de modelagem, simulação e controle de processos. Além disso, foram obtidos novos

conhecimentos nas áreas de controle preditivo com modelo e *Machine Learning*, que não são muito explorados durante a graduação.

## REFERÊNCIAS BIBLIOGRÁFICAS

ALMEIDA, B. S. G.; LEITE, V. C. Particle Swarm Optimization: a powerful technique for solving engineering problems. In: **Swarm Intelligence: recent advances, new perspectives and applications**. London: IntechOpen, 2019.

ASSIS, A. J. **Identificação e controle de processos não lineares utilizando redes neurais artificiais**. 2001. 195p. Tese (Doutorado em Engenharia Química) - Universidade Estadual de Campinas, Faculdade de Engenharia Química, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/267577>>. Acesso em: 29 out. 2020.

BRAVO, C. O. A.; NORMEY-RICO, J. E. Controle de plantas não lineares utilizando controle preditivo linear baseado em modelos locais. **Sba Controle & Automação**, v. 20, n. 4, p. 465-481, 2009.

BRENGEL, D. D.; SEIDER, W. D. Multistep nonlinear predictive controller. **Industrial & engineering chemistry research**, v. 28, n. 12, p. 1812-1822, 1989.

BUCHER, R. **Python for control purposes**. Scuola Universitaria Professionale della Svizzera Italiana, 2019.

CAMACHO, E. F.; BORDONS, C. **Model Predictive Control**. 2<sup>nd</sup> ed. London: Springer-Verlag, 2007.

CARVALHO, R. F.; ALVAREZ, L. A. Sintonia ótima de um controlador MPC aplicado à planta de Williams-Otto. In: CONGRESSO BRASILEIRO DE INSTRUMENTAÇÃO, SISTEMAS E AUTOMAÇÃO, 2019, Campinas. **Anais eletrônicos...** Campinas, Galoá, 2019.

COUGHANOWR, D. R.; LEBLANC, S. E. **Process Systems Analysis and Control**. 3<sup>rd</sup> ed. New York: McGraw-Hill, 2009.

DRAEGER, A; ENGELL, S; RANKE, H. Model predictive control using neural networks. **IEEE Control Systems Magazine**, v. 15, n. 5, p. 61-66, 1995.

FORTUNA, L. et al. **Soft sensors for monitoring and control of industrial processes**. London: Springer-Verlag, 2007.

HAYKIN, S. **Redes neurais: princípios e prática**. 2. ed. Porto Alegre: Bookman, 2001.

KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: ICNN'95 – INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1995, Perth. **Proceedings...**, p. 1942-1948.

KUMAR, S. S. P. et al. A deep learning architecture for predictive control. **IFAC-PapersOnLine**, v. 51, n. 18, p. 512-517, 2018.

LI, W. C.; BIEGLER, L. T. Process control strategies for constrained nonlinear systems. **Industrial & engineering chemistry research**, v. 27, n. 8, p. 1421-1433, 1988.

- MALAR, R. M. S.; THYAGARAJAN, T. Modelling of continuous stirred tank reactor using artificial intelligence techniques. **International Journal of Simulation Modelling**, v. 8, n. 3, p. 145-155, 2009.
- MATSUURA, T.; KATO, M. Concentration stability of the isothermal reactor. **Chemical Engineering Science**, v. 22, n. 2, p. 171-183, 1967.
- MIAO, Q; WANG, S. Nonlinear model predictive control based on support vector regression. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING AND CYBERNETICS. **Proceedings...** Beijing: IEEE, 2002. p. 1657-1661.
- MIRJALILI, S. **Evolutionary Algorithms and Neural Networks: theory and applications**. Cham, Switzerland: Springer International Publishing, 2019.
- NERY Jr.,G.A. **Sintonia ótima de controladores MPC considerando incertezas de modelagem**. 2015. 92p. Dissertação (Mestrado em Engenharia Industrial) – Universidade Federal da Bahia, Escola Politécnica, Salvador, BA. Disponível em: <<http://repositorio.ufba.br/ri/handle/ri/19395>>. Acesso em: 3 nov. 2020.
- OGATA, K. **Engenharia de controle moderno**. 5. ed. São Paulo: Pearson Prentice Hall, 2010.
- QIN, S. J.; BADGWELL, T. A. A survey of industrial model predictive control technology. **Control engineering practice**, v. 11, n. 7, p. 733-764, 2003.
- QIN, S. J.; BADGWELL, T. A. An overview of nonlinear model predictive control applications. In: **Nonlinear model predictive control**. Birkhäuser, Basel, p. 369-392, 2000.
- RAJINIKANTH, V.; LATHA, K. Ajuste e reajuste do controlador PID para sistemas instáveis usando algoritmo evolutivo. **ISRN Chemical Engineering**, v. 2012, 2012.
- SARAIVA, S. V. **Estudo de um Controlador Preditivo Baseado em Sistemas Inteligentes**. 2017. 67p. Trabalho de conclusão de curso (Graduação em Engenharia Química) – Universidade Federal de Alagoas, Centro de Tecnologia, Maceió, AL.
- SARAIVA, S. V.; SILVA, F. V.; CARVALHO, F. O. Comparative Analysis of Machine Learning Models for Predictive Control of the Cyclopentadine Production Process. **Chemical Product and Process Modeling**, v. 1, n. ahead-of-print, 2020.
- SENA, H.; FILETI, A. M. F. Controlador MPC Neural Com Adaptação Online do Modelo Aplicado ao Processo de Controle de pH. In: CONGRESSO BRASILEIRO DE INSTRUMENTAÇÃO, SISTEMAS E AUTOMAÇÃO, 2019, Campinas. **Anais eletrônicos...** Campinas, Galoá, 2019. Disponível em: <<https://proceedings.science/cobisa-2019/papers/controlador-mpc-neural-com-adaptacao-online-do-modelo-aplicado-ao-processo-de-controle-de-ph>>. Acesso em: 02 dez. 2020.
- STEPHANOPOULOS, G. **Chemical process control: an introduction to theory and practice**. Upper Saddle River, NJ: Prentice-Hall, 1984.
- SUSSMAN, M. V.; POTTER, C. Kinetics of propylene and ethylene hydrogenation. **Industrial & Engineering Chemistry**, v. 46, n. 3, p. 457-465, 1954.

TAKÁCS, B. et al. Export of explicit model predictive control to python. In: 2015 20TH INTERNATIONAL CONFERENCE ON PROCESS CONTROL (PC). **Proceedings...** Štrbské Pleso: IEEE, 2015. p. 78-83.

VALIM, I. C. et al. Use of Genetic Algorithm and Particle Swarm Optimization in the Estimation of Kinetic Parameters of Green Coconut Biomass. **Chem. Eng. Trans.**, v. 74, p. 121-126, 2019.

VOJTĚŠEK, Jiří. **Chemical Reactors: Modern Control Methods**. 2007. 167p. Doctoral Thesis - Tomas Bata University in Zlín, Faculty of Applied Informatics, Zlín.

ZHANG, Y. et al. Multi-Objective Particle Swarm Optimization Approach for Cost-Based Feature Selection in Classification. **Trans. Comput. Biol. Bioinform.**, v. 14, n. 1, p. 64-75, 2017.

ZHEJING, B. A. O.; DAOYING, P. I.; YOUXIAN, S. U. N. Nonlinear model predictive control based on support vector machine with multi-kernel. **Chinese Journal of Chemical Engineering**, v. 15, n. 5, p. 691-697, 2007.