



# **Estudo de identificação de um reator de Van de Vusse com Redes Neurais Feedforward**

Gabriela Carvalho Freitas  
Nathalia Braga Antunes Bedor

## **Projeto de Final de Curso**

**Orientador**  
Prof. Bruno Didier Olivier Capron, D.Sc.

Novembro de 2020

# **ESTUDO DE IDENTIFICAÇÃO DE UM REATOR DE VAN DE VUSSE COM REDES NEURONAIS FEEDFORWARD**

*Gabriela Carvalho Freitas*  
*Nathalia Braga Antunes Bedor*

Projeto de Final de Curso submetido ao Corpo Docente da Escola de Química, como parte dos requisitos à obtenção do grau de bacharel em Engenharia Química.

Aprovado por:

---

Marcellus Guedes Fernandes de Moraes, M.Sc.

---

Maurício Bezerra de Souza Júnior, D.Sc.

Orientado por:

---

Bruno Didier Olivier Capron, D.Sc.

Rio de Janeiro, RJ – Brasil  
Novembro de 2020

Bedor, Nathalia Braga Antunes. Freitas, Gabriela Carvalho.

Estudo de identificação de um reator de Van de Vusse com redes neuronais feedforward/Gabriela Carvalho Freitas e Nathalia Braga Antunes Bedor. Rio de Janeiro/EQ,2020.

vii, 44 p.; il.

(Monografia) – Universidade Federal do Rio de Janeiro, Escola de Química, 2020.

Orientador: Bruno Didier Olivier Capron.

1. Reator Van de Vusse. 2. Identificação. 3. Redes neuronais feedforward. 4. Monografia. (Graduação – UFRJ/EQ).
5. Bruno Didier Olivier Capron I. Estudo de identificação de um reator de Van de Vusse com redes neuronais.

## AGRADECIMENTOS

Gostaria de agradecer primeiramente aos meus pais, Lúcia e Hermínio, por sempre me incentivarem e apoiarem durante toda a minha vida. Obrigada por todo carinho, amor e por terem valorizado a educação durante a minha formação.

À toda minha família, em especial, ao meu irmão Matheus, que é meu melhor amigo e sempre contribuiu para meu crescimento, e à sua namorada, Karen, pela cumplicidade e amizade durante todos esses anos. Aos meus avós, Zelia e Antônio, que sempre me receberam de braços abertos com muito carinho e amor. Aos meus primos e tios que estiveram presentes em momentos alegres e difíceis.

Ao meu namorado, Gabriel, por toda paciência, carinho e parceria durante os últimos anos de curso. Obrigada pelos conselhos e apoio acadêmico.

A todos os meus amigos de faculdade, que proporcionaram um ambiente acolhedor e descontraído, forneceram apoio emocional e acadêmico, além de serem fiéis companheiros fora da sala de aula. Em especial gostaria de citar Paola, Helena, Eduardo, Matheus, Mosquito, Psy, Rafael, Lucas, Nathália, João, Carol e André.

Aos meus amigos de longa data Amanda e Ayres, pelo companheirismo, e à Galere que sempre proporciona os dias mais divertidos e memoráveis.

A UFRJ e a Escola de Química, seu corpo docente e equipe administrativa, por fornecerem uma educação de qualidade e toda a infraestrutura que torna isso possível.

Ao meu professor orientador, Bruno, pela confiança, atenção e conhecimentos transmitidos, fundamentais para a elaboração deste projeto. E à minha companheira de TCC, Nathalia pela parceria e dedicação durante o projeto.

A todos que direta ou indiretamente contribuíram e fizeram parte da minha formação, o meu muito obrigada.

**Gabriela Carvalho Freitas**

Gostaria de agradecer e dedicar esta tese as seguintes pessoas:

À minha mãe, Maria das Graças, por todo apoio, carinho e cuidado ao longo da vida. Ao meu pai, Jorge, por me ensinar o valor da educação.

Aos meus irmãos, Priscilla e Arthur, por terem contribuído imensamente na minha formação, tanto pessoal quanto acadêmica.

Aos meus queridos amigos e companheiros de faculdade, Eloi e Pepe, pelas diversas memórias compartilhadas ao longo dos anos, tanto boas e ruins, que fizeram muita diferença durante essa minha jornada.

Aos meus amigos do Kung Fu, especialmente ao Matheus e André, pelos treinos divertidos e jantares do Bandeirão, tornando os últimos anos da faculdade os melhores. A minha primeira Sifu, Bárbara, que apesar de pouco tempo de convívio, me passou lições valiosas que tanto me ajudaram em momentos difíceis.

Ao meu melhor amigo, Celso, por ter me ensinado Python e por ser um grande incentivador.

Ao meu professor orientador, Bruno, e companheira de TCC, Gabriela, pelo empenho dedicado a este projeto de pesquisa.

A todas as pessoas que direta e indiretamente contribuíram para minha formação pessoal e acadêmica.

***Nathalia Braga Antunes Bedor***

Resumo do Projeto Final de Curso apresentado à Escola de Química como parte dos requisitos necessários para obtenção do grau de Bacharel em Engenharia Química.

## **ESTUDO DE IDENTIFICAÇÃO DE UM REATOR DE VAN DE VUSSE COM REDES NEURONAIS FEEDFORWARD**

Gabriela Carvalho Freitas  
Nathalia Braga Antunes Bedor

Novembro, 2020

Orientador: Prof. Bruno Didier Olivier Capron, D. Sc.

A indústria química está repleta de processos que apresentam comportamento dinâmico complexo e não linear, o que dificulta a sua modelagem por métodos tradicionais. Nesse trabalho realizou-se a identificação do reator de Van de Vusse utilizando redes neuronais como modelo. Essa escolha é justificada pela não linearidade e boa capacidade de predição das redes neuronais. Foi analisado primeiramente um caso base, e depois modificaram-se os parâmetros tempo de amostragem, quantidade de dados, quantidade de neurônios na camada escondida, quantidade de camadas escondidas e funções de ativação da camada escondida para realização de uma análise de sensibilidade. Foi avaliado também um caso melhorado com aplicação dos resultados encontrados na análise. Verificou-se um bom ajuste das redes ao comportamento do processo, mostrando-se uma boa alternativa para o campo da modelagem. As alterações realizadas geraram melhorias pequenas, porém os resultados são relevantes para auxiliar a escolha de parâmetros.

## ÍNDICE

Capítulo I – Introdução	1
Capítulo II – Revisão Bibliográfica	3
II.1. Identificação de sistemas	3
II.1.1. Introdução	3
II.1.2. Projeto de um experimento de identificação	5
II.1.2.1. Registro dos dados	6
II.1.2.1.1. Escolha dos sinais de entrada	6
II.1.2.1.2. Escolha do tempo de amostragem	6
II.1.2.2. Determinação do tipo de modelo	8
II.1.2.3. Estimação dos parâmetros	8
II.1.2.4. Validação de modelos	9
II.2. Redes neurais	10
II.2.1. Introdução	10
II.2.2. Modelo de um neurônio	11
II.2.2.1. Tipos de função de ativação	12
II.2.3. Tipos de redes	14
II.2.4. Treinamento de redes	16
II.3. Reator de Van de Vusse	16
II.4. Aplicações	19
Capítulo III – Metodologia	21
III.1. Procedimento de identificação	21
III.1.1. Registro dos dados	22
III.1.1.1 Seleção da região de estudo	22
III.1.1.2 Escolha dos sinais de entrada	23
III.1.1.3. Definição da quantidade de dados	24
III.1.1.4 Definição do tempo de amostragem	24
III.1.1.3.1 Análise com ruído	25
III.1.2. Determinação do tipo de modelo	26
III.1.2.1 Quantidade de neurônios na camada escondida	27
III.1.2.2 Quantidade de camadas escondidas	28
III.1.2.3 Funções de ativação da camada escondida	28
III.1.3. Validação do modelo	28

Capítulo IV – Resultados e Discussão	30
IV.1. Caso base	30
IV.1.1. Definição do tempo de amostragem	30
IV.1.2. Geração dos sinais de entrada	33
IV.1.3. Validação do modelo	35
IV.2. Análise de Sensibilidade	37
IV.2.1. Tempo de amostragem	37
IV.2.1.1. Análise com ruído	38
IV.2.2. Quantidade de dados	39
IV.2.3. Quantidade de neurônios na camada escondida	40
IV.2.4. Quantidade de camadas escondidas	40
IV.2.5. Função de ativação da camada escondida	41
IV.3. Caso melhorado	42
Capítulo V – Conclusão e sugestões para trabalhos futuros	45
Capítulo VI - Referências Bibliográficas	46

# Capítulo I – Introdução

Com base no estudo do cérebro humano, foram desenvolvidos modelos matemáticos que tentam imitar as redes altamente interconectadas de neurônios que compõem esse órgão. Esses modelos são chamados de redes neuronais artificiais ou ANNs (abreviatura para “*Artificial Neural Networks*”). Essas redes possuem características que as tornam bastante populares, como a aprendizagem através de exemplos e capacidade de abstrair características relevantes a partir de dados abundantes (DE SOUZA JR, 2005).

A modelagem de um sistema é o uso de um conjunto de equações matemáticas para representar o comportamento de um processo. O comportamento, neste caso, nada mais é do que a forma como as saídas respondem às entradas do processo. Modelos matemáticos têm sido utilizados em diversas áreas de estudo, com o objetivo de entender e explicar fenômenos observados tanto na natureza quanto em sistemas sociais, biomédicos etc (AGUIRRE, 2004).

No contexto da Engenharia Química, as variáveis que compõem os sistemas estão em constante mudança ao longo do tempo, portanto se faz necessário monitorar e manipular variáveis que afetam diretamente a segurança, qualidade e rendimento dos processos. Nesse sentido, a modelagem pode ser usada para diversos fins como: melhora do entendimento do processo, treinamento da equipe de operação da planta, desenvolvimento de uma estratégia de controle para um novo processo ou otimização da condição de operação do processo (SEBORG, 2011; DE SOUZA JR; TRICA, 2013).

Porém, um desafio encontrado é a grande quantidade de processos que apresentam dinâmica complexa, devido a não linearidade, tempos mortos etc., necessitando de muito tempo e esforço de engenharia para obtenção de modelos teóricos. Como uma alternativa, as ANNs, que podem ser classificadas como modelos empíricos não lineares, podem ser utilizadas para a modelagem utilizando apenas dados históricos da planta. Essa solução pode ser mais genérica e economicamente vantajosa, além de mais fáceis do que a aplicação de métodos tradicionais (DE SOUZA JR, 2005; SEBORG, 2011).

Um exemplo de processo complexo é o reator de Van de Vusse, que consiste em um modelo matemático para uma reação química em um CSTR, que foi introduzido como um *benchmark* para problemas de controle. Este modelo é um sistema não linear de quarta ordem que apresenta dinâmica com zero instável na região de interesse, tornando-o um interessante objeto para estudo de caso.

Com o objetivo de verificar a aplicação de redes neuronais como possibilidade para identificação de sistemas dinâmicos complexos, este trabalho se propõe a realizar e avaliar a identificação do reator de Van de Vusse a partir de redes neuronais *feedforward*. Um objetivo secundário é a avaliação da influência individual de cada parâmetro escolhido, durante o procedimento de identificação, no desempenho do modelo.

Inicialmente apresenta-se uma revisão bibliográfica referente aos assuntos abordados nesse projeto no Capítulo II. Primeiro aborda-se o campo da identificação de sistemas, com uma breve introdução e definição das etapas de um projeto de experimento de identificação. Posteriormente apresentam-se as redes neuronais com suas características, topologia e tipos de redes. Descreve-se então o modelo do processo escolhido como objeto de estudo desse trabalho. Por fim são apresentadas aplicações encontradas na literatura para situar o trabalho.

No Capítulo III apresenta-se a metodologia utilizada para o experimento de identificação realizado e a análise de sensibilidade para cada parâmetro.

No Capítulo IV são aplicados os métodos descritos no Capítulo III e apresentam-se os resultados obtidos para o caso base do experimento e para a análise de sensibilidade. Por fim,

mostra a aplicação das melhorias obtidas durante a análise ao caso base, avaliando um resultado melhorado.

Por fim, o Capítulo V apresenta as conclusões alcançadas bem como sugestões para trabalhos futuros.

# Capítulo II – Revisão Bibliográfica

## II.1. Identificação de sistemas

### II.1.1. Introdução

Segundo Horváth (2002), identificação é o processo de deduzir um modelo matemático de uma parte do mundo predefinida utilizando observações. Um modelo é a descrição formal de um sistema, que por ser uma simplificação é sempre imperfeito. Quando são utilizados apenas dados observáveis de entrada e saída para esta finalidade, denomina-se um modelo caixa preta.

Um sistema pode ser definido, de forma geral, como um objeto onde diferentes variáveis se relacionam gerando sinais observáveis, chamadas de saídas. Sinais que afetam o sistema e podem ser manipulados pelo observador são denominadas entradas; e os demais, que podem ser observáveis ou não, são chamados de distúrbios. Esse conceito é amplamente utilizado em diversas áreas da ciência para a resolução de problemas (LJUNG, 1999a).

Há diversas formas de classificar os modelos. Uma delas é a distinção entre estacionários e dinâmicos. Nos modelos estacionários, as variáveis são representadas como sendo independentes do tempo, ao contrário dos dinâmicos, onde uma ou mais variáveis são dependentes do tempo (DE SOUZA JR; TRICA, 2013). Para Ogunnaike & Ray (1994), os processos químicos podem ser considerados como sendo naturalmente dinâmicos, apresentando um comportamento tempo-dependente.

Outra forma de classificação da modelagem é sua divisão em três grupos: modelagem caixa branca, modelagem caixa preta e modelagem caixa cinza. Na modelagem caixa branca, também denominada fenomenológica ou ainda teórica, é necessário conhecer as leis físicas que descrevem o sistema a ser modelado. Esse tipo de modelagem oferece um profundo entendimento do funcionamento do sistema, porém nem sempre é viável, devido à necessidade de conhecimento e tempo para modelar um sistema partindo do equacionamento dos fenômenos envolvidos.

Quando não se conhecem as equações envolvidas no funcionamento de um sistema, ou seu equacionamento é demasiadamente complexo, utiliza-se a modelagem caixa preta, que necessita de pouco ou nenhum conhecimento prévio do sistema. Essa técnica é também denominada modelagem empírica ou “identificação de sistemas”, que consiste na construção de modelos matemáticos de sistemas dinâmicos baseado em dados observados de entrada e saída. Uma característica importante em relação ao modelo caixa preta é a natureza dos parâmetros que integram o modelo. Na maioria das vezes, os parâmetros determinados não possuem qualquer significado físico, sendo apenas estrutura matemática que descreve uma relação de causa e efeito (LJUNG, 1999a; AGUIRRE, 2004).

Na modelagem caixa cinza ocorre um meio termo entre a modelagem teórica e empírica, também chamada de modelagem híbrida. É geralmente uma combinação de equações de balanço com equações empíricas ou o uso de dados experimentais para o cálculo de parâmetros de um modelo teórico (SEBORG, 2011; DE SOUZA JR; TRICA, 2013).

Ao se construir diferentes modelos para se descrever as mesmas observações de um sistema é necessário se ter princípios orientadores para a escolha do melhor modelo. Para tal, utiliza-se o princípio de parcimônia, que significa que em modelagem é sempre preferível utilizar o modelo mais simples possível, que seja consistente com os dados observáveis.

Algumas considerações simplificadoras devem ser feitas a fim de desenvolver representações aproximadas ao sistema real. A linearidade, segundo Aguirre (2004), é uma suposição frequentemente realizada. Considera-se que o modelo se comporte de maneira

aproximadamente linear, sendo posteriormente verificado seu comportamento em uma faixa estreita de operação.

Para um sistema ser considerado linear, o princípio da superposição deve ser satisfeito. Ou seja, quando o sistema é excitado pelas entradas  $u_1$  e  $u_2$  produz  $y_1$  e  $y_2$ , respectivamente, esse mesmo sistema quando excitado por  $a \cdot u_1 + b \cdot u_2$  deverá produzir a saída  $a \cdot y_1 + b \cdot y_2$ , sendo  $a$  e  $b$  constantes possivelmente complexas, para ser considerado linear. Logo, quando este princípio não é obedecido, o modelo é considerado como sendo não linear.

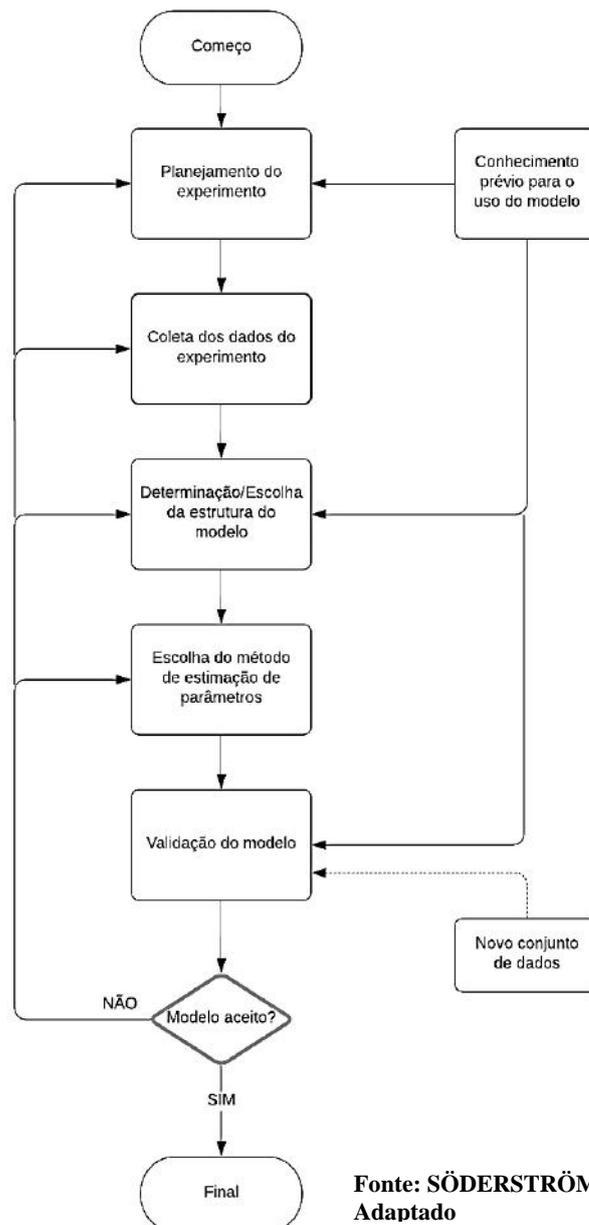
É uma prática comum dentro da engenharia linearizar sistemas que não satisfazem o princípio da superposição devido ao amplo conhecimento disponíveis acerca de sistemas lineares. Entretanto existem duas limitações básicas de tal abordagem. A primeira delas se deve a linearização ser uma aproximação nas vizinhanças do ponto de operação, de forma que ela só é capaz de prever o comportamento local do sistema nas proximidades desse ponto. A segunda, diz respeito a riqueza da dinâmica do sistema não linear frente ao linear. Alguns fenômenos dos sistemas não lineares são apenas manifestados quando há o fator ‘não linearidade’ envolvido, de modo que tais sistemas não podem ser descritos ou preditos por modelos lineares (KHALIL, H.K, 2015).

## II.1.2. Projeto de um experimento de identificação

Geralmente, o processo de identificação segue as seguintes etapas ( SÖDERSTRÖM; PETRE, 1989; LJUNG, 1999; AGUIRRE, 2004) :

1. Registro dos dados;
2. Determinação do tipo de modelo;
3. Estimação dos parâmetros desconhecidos do modelo por meio de um método estatístico;
4. Validação do modelo com os parâmetros estimados;
5. Caso não seja validado, um novo modelo, mais complexo, é determinado e as etapas anteriores são repetidas até que se obtenha um resultado satisfatório.

Na Figura 1 encontra-se o esquema contendo as etapas básicas para a identificação de um sistema dinâmico:



Fonte: SÖDERSTRÖM & PETRE (1989) - Adaptado

Figura 1- Fluxograma do sistema de identificação

A seguir, será apresentada brevemente cada um dos passos necessários para a realização de um projeto de identificação.

#### II.1.2.1. Registro dos dados

Para Aguirre (2004), o estudo da identificação de sistemas possui três aspectos essenciais: o ponto de excitação do sistema, o tipo de sinal para obter os dados que representem adequadamente essa dinâmica, e o modo como tais dados são amostrados.

Para a identificação do modelo que relaciona dinamicamente duas variáveis do sistema, parte-se da hipótese de que as variáveis se correlacionam entre si, de modo a justificar o modelo matemático. Existem algumas ferramentas eficazes que são capazes de apontar se há correlação entre as variáveis ou não. Aguirre cita a correlação cruzada, e, para procedimentos mais elaborados, a decomposição em valores singulares (SVD) e a análise de componentes principais (PCA).

##### II.1.2.1.1. Escolha dos sinais de entrada

Em geral, as entradas utilizadas devem atender ao critério tal que sua função de autocorrelação fosse não-nula apenas para o primeiro atraso do sistema, mesmo quando o método de identificação não utilizar as funções de correlação. Nesse caso, os sinais de entrada teriam comportamento similar aos sinais aleatórios ou pseudoaleatórios.

Para a identificação de sistemas lineares, os sinais binários pseudoaleatórios (PRBS) são frequentemente utilizados. Entretanto, para Leontaritis & Billings (1987a), tais sinais, do ponto de vista teórico, não seriam adequados na identificação de sistemas não lineares. Uma opção melhor nesse caso seria utilizar sinais aleatórios. Já para Aguirre (2004), é possível obter bons resultados com os sinais PRBS para identificar modelos não lineares.

Para a geração de sinais excitantes, costuma-se utilizar um gerador de números aleatórios. Uma boa prática para gerar bons sinais, uma vez definido o tempo de amostragem, é manter constante cada valor escolhido aleatoriamente, em torno de 3 a 5 intervalos de amostragem.

##### II.1.2.1.2. Escolha do tempo de amostragem

A maioria dos sistemas reais são contínuos no tempo, sendo necessário registrar suas variáveis de forma discreta no tempo. Para Aguirre (2004), o tempo de amostragem pode ser definido como o período entre duas amostras  $T_s$ . Existem alguns estudos e recomendações para se obter o melhor tempo de amostragem, isto é, o tempo ideal que nos forneça dados amostrados que preservem as características essenciais do dado original.

Com um  $T_s$  muito maior do que a constante de tempo do sistema, poderá haver pouca informação sobre a dinâmica do sistema nos dados. Já um  $T_s$  muito pequeno, poderia ser pouco informativo por evidenciar o ruído. Assim, o tempo de amostragem ideal é um *trade-off* entre esses dois aspectos. Ljung (1999), através da análise de um exemplo, chega a algumas conclusões: a escolha ideal para o tempo de amostragem está ao redor da constante de tempo do sistema; em muitos casos, é preferível usar um  $T_s$  muito grande do que muito pequeno; e um  $T_s$  muito pequeno pode gerar problemas numéricos.

Segundo o teorema de Shannon (1948), o sinal que não contiver componentes de frequência acima de  $1/2T_s$  pode ser determinado apenas a partir de amostras deste sinal

original separadas pelo tempo de amostragem. O teorema também exige que a frequência de amostragem escolhida seja 2 vezes maior do que a maior frequência de interesse contida nos dados (frequência de Nyquist). Porém, na prática, costuma-se empregar uma frequência de 5 a 10 vezes, ao invés de 2. Tal abordagem, entretanto, pode ocasionar em dados não tão bem amostrados, necessitando assim, do uso de filtros para cortar frequências mais elevadas e evitar o falseamento do sinal.

Aguirre também aponta outras desvantagens ao se utilizar dessa abordagem; a primeira, pela perspectiva numérica, se deve ao risco de a estimação dos parâmetros ser mal condicionada caso o intervalo de amostragem seja muito curto; a segunda, do ponto de vista prático, é consequência de um possível desconhecimento da maior frequência de interesse. Com isso, não seria viável escolher a frequência de amostragem 5 a 10 vezes maior.

Deste modo, o autor propõe uma abordagem para determinar a frequência de amostragem, que tem apresentado bons resultados na prática. Inicialmente, o sinal real do processo  $y^*(t)$  é considerado como registrado em um tempo de amostragem muito pequeno, de modo a representar um sinal superamostrado. Porém, surge o problema de definir a taxa pela qual o sinal  $y^*(t)$  será decimado para que o sinal de trabalho  $y(t)$  seja gerado. Desta forma, almeja-se determinar o número inteiro  $\Delta$  tal que  $y(t) = y^*(\Delta \cdot t)$ .

Para isso, é preciso que o grau de correlação entre observações adjacentes do sinal  $y^*(t)$  seja verificado, de modo a avaliar a redundância entre duas observações consecutivas. As funções de autocovariância linear e não linear, representadas pelas equações (1) e (2), respectivamente, são utilizadas para quantificar os efeitos causados pela sobreamostragem do sinal  $y^*(t)$ :

$$r_{yy,1}(k) = \frac{1}{N} \sum_{i=1}^N (y^*(i) - \bar{y}^*)(y^*(i+k) - \bar{y}^*) \quad (1)$$

$$r_{yy,2}(k) = \frac{1}{N} \sum_{i=1}^N (y^*(i)^2 - \overline{y^*(t)^2})(y^*(i+k)^2 - \overline{y^*(t)^2}) \quad (2)$$

Sendo que  $\overline{y^*(t)^2}$  é a média temporal do sinal medido  $y^*(t)$  elevado ao quadrado. Obtendo  $\tau_{1,y^*}$  e  $\tau_{2,y^*}$  como os primeiros mínimos das funções de autocovariância linear e não linear, respectivamente, pode-se determinar  $\tau_m^*$ , como sendo o mínimo desses valores, conforme mostra a equação (3):

$$\tau_m^* = \min(\tau_{1,y^*}, \tau_{2,y^*}) \quad (3)$$

O valor de  $\Delta$  é escolhido de forma que  $\tau_m^*$  satisfaça a seguinte desigualdade:

$$10 \leq \tau_m \leq 20 \quad (4)$$

$$\text{onde } \tau_m = \frac{\tau_m^*}{\Delta}$$

Substituindo os valores acima, tem-se então:

$$10 \cdot \Delta \leq \tau_m^* \leq 20 \cdot \Delta \quad (5)$$

#### II.1.2.2. Determinação do tipo de modelo

Para Södertröm & Petre (1989), tanto a determinação quanto a validação de um modelo são aspectos importantes no processo de identificação. Se a estrutura do modelo for muito parametrizada (*overfitting*), um custo computacional desnecessário será empregado para encontrar os parâmetros e estimar o modelo. Por outro lado, segundo Aguirre (2004), um modelo pouco parametrizado será um modelo com baixa acurácia e sem a complexidade estrutural necessária para reproduzir a dinâmica do sistema (*underfitting*).

Os autores levantam algumas questões fundamentais para determinação da ordem do modelo:

1. O modelo encontrado é flexível o suficiente para cobrir todo o sistema?
2. O modelo encontrado é muito complexo?
3. Como escolher a estrutura do modelo quando existem um ou mais candidatos?

Um modelo não linear, segundo Aguirre (2004), é mais complexo de definir uma ordem, uma vez que o número de estruturas possíveis cresce muito rapidamente conforme aumenta-se a ordem do modelo. Além disso, termos redundantes em modelos não lineares destroem a dinâmica original do modelo. Por outro lado, seu uso é preferível em situações nas quais o modelo linear não é suficientemente capaz de representar a dinâmica do sistema, de modo que seja necessário o uso de um modelo mais complexo.

#### II.1.2.3. Estimação dos parâmetros

Supondo que as variáveis  $x$  e  $y$  estejam relacionadas de tal forma que  $y = f(x)$ , podemos ter uma função  $f(\cdot)$ , caracterizada pelo vetor de parâmetros  $\theta$ . Com isso, é dito que  $f(\cdot)$  é parametrizada por  $\theta$ , sendo explicitamente expressa por  $y = f(x, \theta)$ . Portanto, a estimação de parâmetros tem o objetivo de estimar  $\theta$  partindo de um conjunto de valores de  $x$  e  $y$ . (AGUIRRE, 2004)

Em problemas de identificação de sistema, tanto  $\theta$  quanto  $\hat{f}(\cdot)$  precisam ser determinados, sendo que geralmente, tais parâmetros estimados não apresentam qualquer significado físico. Isso se deve ao fato de que muitas vezes as funções  $\hat{f}(\cdot)$  são descritas como uma estrutura matemática capazes de descrever uma relação de causa e efeito, e não como equações que descrevem o comportamento de um processo (AGUIRRE, 2004)

Para Ljung (1999), o método de predição de erro é considerado uma abordagem básica para a estimação dos parâmetros e possui três vantagens:

1. aplicabilidade para estrutura de modelos gerais;
2. acurácia assintótica ótima quando o modelo consegue representar o sistema real;
3. Aproximação razoável das propriedades do sistema quando o mesmo não consegue ser representado pela estrutura do modelo.

Outras abordagens são preferíveis quando um sistema linear possui diversas saídas, o que demanda o uso de muito parâmetros. Dentre delas, podemos citar o método do subespaço, que permite a estimação do parâmetro utilizando cálculos numéricos e robusto sem precisar utilizar a busca iterativa. (LJUNG,1999)

Por fim, o autor conclui que não existe “o melhor” método de estimação de parâmetros e recomenda que se teste os métodos disponíveis e valide os resultados de modo a encontrar aquele que seja o mais adequado para o experimento em questão.

#### II.1.2.4. Validação de modelos

A validação de modelos é a etapa final que decide se o produto da identificação de um sistema real é suficientemente bom. Para saber se o modelo, ou o conjunto de modelos, é válido ou não, é necessário saber se o modelo é bom o bastante para os propósitos de sua aplicação e se ele é capaz de descrever o “verdadeiro sistema”. (LJUNG, 1999; AGUIRRE,2004)

Aguirre (2004) conclui que o modelo, sendo um representativo do sistema em alguns aspectos, apenas poderá ser considerado válido caso incorpore tais características do sistema que são essenciais para a sua aplicação. Existe uma gama de ferramentas citadas na literatura que podem ser utilizadas na verificação da qualidade de um modelo.

As medidas de acurácia de predição mais amplamente utilizadas são o erro quadrático médio (MSE), a raiz do erro quadrático médio (RMSE) e o erro absoluto (MAE), representado pelas equações (6), (7) e (8), respectivamente:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (6)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (7)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (8)$$

Aguirre (2004) propõe a utilização do índice de quantificação de parâmetros, RMSE, que pode ser encontrado abaixo:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y(k) - \hat{y}(k))^2}{\sum_{i=1}^n (y(k) - \bar{y})^2}} \quad (9)$$

Sendo que  $\hat{y}(k)$  é a simulação livre do sinal e  $\bar{y}$  é a média do sinal medido  $y(k)$ . Essa métrica compara as predições do modelo com a média temporal do sinal, que seria o preditor padrão. Quanto mais próximo de zero for o valor de RMSE melhor é o desempenho do preditor do modelo.

Para se ter uma visibilidade maior do quão bem o sinal simulado se aproxima do sinal real, podemos calcular um “índice de ajuste” da seguinte forma:

$$\text{Índice de Ajuste (\%)} = (1 - RMSE) \times 100 \quad (10)$$

## II.2. Redes neuronais

### II.2.1. Introdução

Inspirando-se em características de aprendizagem do cérebro humano, que é altamente complexo, não linear e paralelo, foram desenvolvidos modelos matemáticos que contém mecanismos semelhantes, denominados Redes Neuronais Artificiais.

Assim como no cérebro, as redes artificiais possuem neurônios capazes de se “adaptar” e “aprender” com o ambiente. Esses tipos de modelos têm sido foco de atenção devido à ampla possibilidade de aplicação e disponibilidade para uso do conhecimento experimental armazenado (HAYKIN, 2009).

Segundo Haykin (2009), uma rede neuronal pode ser definida como:

“Um processador distribuído altamente paralelo, composto por unidades de processamento simples que possuem a propensão natural para armazenar conhecimento experimental e torná-lo disponível para uso.”

Sua topologia confere algumas capacidades e características úteis à rede. Dentre elas é possível citar:

- Não linearidade. A interconexão entre diferentes neurônios promove uma não linearidade distribuída pela rede;
- Mapeamento de entradas e saídas. Dentro do contexto de Inteligência Artificial, as redes neuronais podem ser utilizadas para aprendizagem supervisionada. Nesse caso, é apresentado à rede uma série de dados de treino, com suas respectivas saídas desejadas, e seus pesos sinápticos são modificados de forma a minimizar a diferença entre a resposta desejada e a atual. Esse treino é repetido diversas vezes, até que determinado critério estatístico seja atingido e a rede alcance um estado permanente.
- Adaptabilidade. Os pesos sinápticos da rede neuronal podem ser facilmente modificados a partir de novos treinamentos, adaptando-se a mudanças nas condições do ambiente. Isso permite sua aplicação em sistemas não-estacionários e em tempo real, como classificação adaptativa, processamento de sinal adaptativo ou controle adaptativo.
- Generalização. Uma vantagem do uso de redes é sua capacidade de generalizar, de forma a fornecer saídas razoáveis para entradas nunca vistas no treinamento, alcançando boas soluções para problemas complexos.

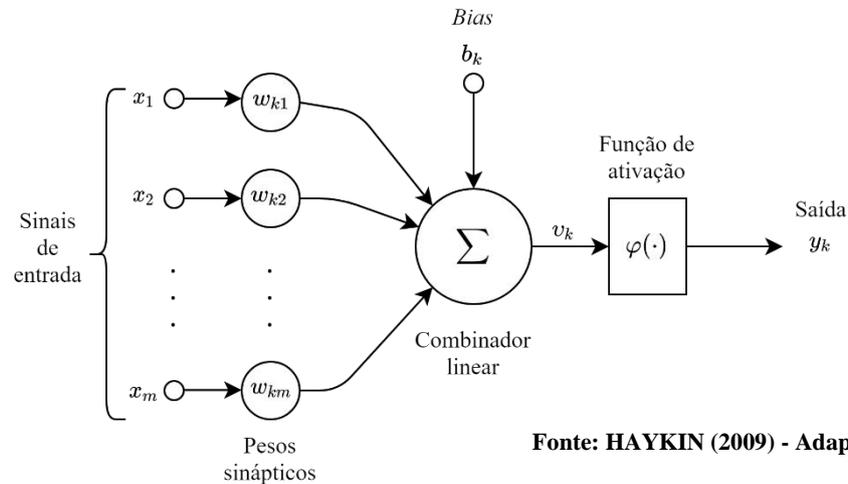
McCulloch and Pitts (1943) foram os primeiros a introduzir a ideia de redes neuronais como máquinas computacionais, criando o primeiro modelo matemático que utiliza uma função degrau limiar para um neurônio. Hebb (1949) postulou a primeira regra para um aprendizado automático organizado. Mais tarde, Rosenblatt (1958) propôs o primeiro método de aprendizagem supervisionada, o perceptron, que é a forma mais simples de uma rede neuronal utilizada para classificar padrões.

O perceptron de Rosenblatt (1958) usa o Modelo de Neurônio de McCulloch-Pitts (1943) e pode ser denominado como uma rede neuronal de camada única. Ele é a forma mais simples de uma rede neuronal, utilizado para a classificação de padrões linearmente separáveis.

Apesar de ter sido alvo de muitas críticas, devido às suas limitações, o perceptron permitiu o posterior desenvolvimento de diversas estruturas de redes mais avançadas (HAYKIN, 2009).

## II.2.2. Modelo de um neurônio

A Figura 2 mostra o modelo de um neurônio, que é a unidade de processamento fundamental de uma rede neuronal.



**Figura 2 - Modelo de um neurônio artificial**

Seus elementos se relacionam da seguinte forma:

A conexão entre os sinais de entrada  $x_1, x_2 \dots x_m$  e o neurônio  $k$  é feita através da multiplicação por um peso sináptico  $w_{k1}, w_{k2} \dots w_{km}$  (Equação 11). Então, um combinador linear realiza o somatório dos sinais de entrada, ponderados pelos pesos, adicionando também um bias ( $b_k$ ) (Equação 12). O valor resultante ( $v_k$ ), denominado campo local induzido, passa por uma função de ativação, responsável por limitar a amplitude do sinal, gerando o sinal de saída  $y_k$  (Equação 13).

O modelo do neurônio  $k$  também pode ser expresso matematicamente através das seguintes equações:

$$u_k = \sum_{j=1}^m w_{kj}x_j \quad (11)$$

$$v_k = u_k + b_k \quad (12)$$

$$y_k = \varphi(v_k) \quad (13)$$

Onde  $u_k$  é o resultado da multiplicação entre o sinal de entrada e os pesos sinápticos;  $v_k$  é o resultado da combinação linear entre  $u_k$  e o bias ( $b_k$ ), e  $\varphi$  é a função de ativação.

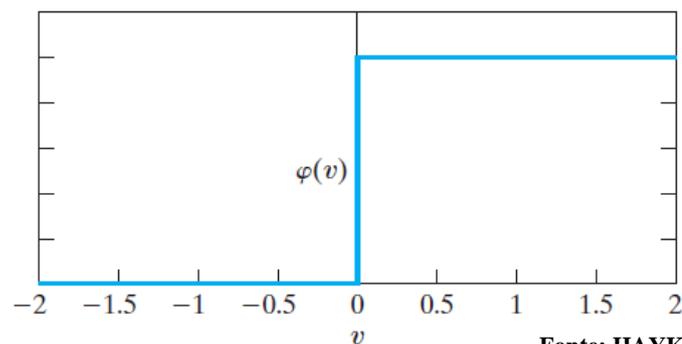
### II.2.2.1. Tipos de função de ativação

A função de ativação é o que define a saída de um neurônio em termos do seu campo local induzido. Apesar da ampla possibilidade de formas, algumas se destacam pela sua aplicação.

#### **Função Limiar**

Trata-se de uma função degrau em que a saída do neurônio assume o valor 1 caso o valor de  $v_k$  seja positivo, e 0 caso seja negativo. Sua equação é definida na Equação 14 e seu formato apresentado pela Figura 3.

$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases} \quad (14)$$



Fonte: HAYKIN (2009)

**Figura 3 - Gráfico da função limiar**

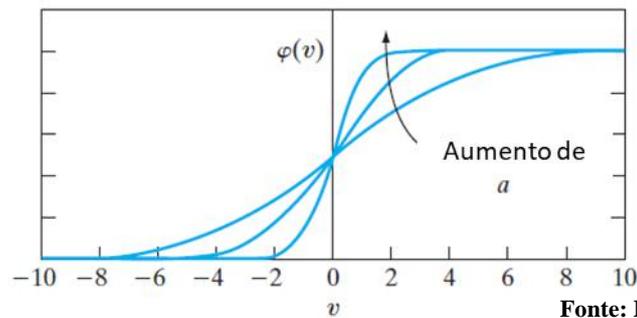
O neurônio que tem como função de ativação a função limiar é também denominado como perceptron, devido ao trabalho de Rosenblatt realizado em 1958.

#### **Função Sigmóide**

Esse tipo de função, que apresenta um formato em “S” é o mais utilizado em redes neuronais. Um exemplo de função Sigmóide é a função logística, definida a partir da seguinte equação:

$$\varphi(v) = \frac{1}{1 + e^{-av}} \quad (15)$$

Onde  $a$  é o parâmetro de inclinação da função. No limite em que o parâmetro  $a$  tende ao infinito, a função se torna a função limiar. Seu formato é mostrado na Figura 4.



Fonte: HAYKIN (2009) - Adaptado  
**Figura 4 - Gráfico da função logística**

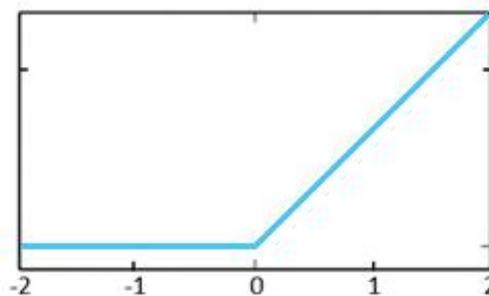
Uma característica importante que difere a função Sigmoide do limiar é sua continuidade, e portanto, sua diferenciabilidade. Um formato também bastante popular dessa função é a tangente hiperbólica, que apresenta o mesmo comportamento, porém limitada entre -1 e 1 ao invés de 0 e 1. Sua equação é descrita abaixo.

$$\varphi(v) = \frac{e^v - e^{-v}}{e^v + e^{-v}} \quad (16)$$

### Função ReLU

Atualmente no campo de redes neurais tem se recomendado o uso da função de ativação *rectified linear unit* ou ReLU, definida pela Equação 16 e com comportamento exibido na Figura 5.

$$\varphi(v) = \max\{0, v\} \quad (17)$$



Fonte: GOODFELLOW, BENGIO, COURVILLE (2016) - Adaptado

**Figura 5 - Gráfico da função ReLU**

Essa função, enquanto realiza uma transformação não linear, mantém muitas propriedades de funções lineares, que são fáceis de otimizar com métodos que usam gradientes e possuem uma boa capacidade de generalização (GOODFELLOW; BENGIO; COURVILLE, 2016).

Antes da introdução das unidades lineares retificadas, a maioria das redes utilizavam a função de ativação Sigmoide logística ou tangente hiperbólica, que são bastante similares. A tangente hiperbólica costuma performar melhor do que a logística pois lembra mais a função identidade, assim seu treinamento se assemelha ao de uma função linear. Um problema das funções sigmoideais é a saturação, pois saturam em um valor grande quando a entrada é muito

positiva e em um valor pequeno quando a entrada é muito negativa, só são sensíveis quando a entrada estiver próxima de 0. Esse comportamento pode dificultar o aprendizado baseado em gradientes e por isso seu uso já não é mais tão encorajado.

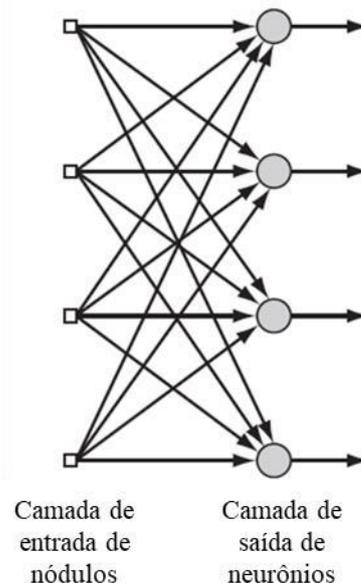
As unidades lineares retificadas foram evitadas por muito tempo devido a seu ponto não-diferenciável. Isso começou a mudar quando Jarret et al (2009) mostrou a importância de usar uma não linearidade retificante para o design da arquitetura de uma rede neuronal. Ela se tornou o padrão para muitos tipos de redes neurais devido a facilidade de treinamento e por alcançar, muitas vezes, resultados melhores, principalmente em redes com muitas camadas (3 ou mais). Glorot et al (2010) mostra que neurônios com retificação linear são melhores modelos de neurônios biológicos e alcançam performance igual ou melhor do que redes com tangente hiperbólica, criando uma representação dispersa com zeros verdadeiros.

### II.2.3. Tipos de redes

A forma como os neurônios estão organizados dentro da rede influencia diretamente no seu aprendizado e performance. Existem, de forma geral, três tipos diferentes de arquiteturas: Redes *Feedforward* de Camada Única, Redes *Feedforward* de Múltiplas Camadas e Redes Recorrentes.

#### **Redes *Feedforward* de Camada Única**

Essa é a estrutura mais básica que uma rede pode apresentar. É composta por uma camada de entrada conectada diretamente à uma camada de saída de neurônios. A camada de entrada não é levada em conta porque não há operações ocorrendo nela, por isso o nome de Camada Única. Sua estrutura é exemplificada na Figura 6.



Fonte: HAYKIN (2009) - Adaptado

Figura 6 – Representação de uma Rede Feedforward de Camada Única

#### **Redes Feedforward de Múltiplas Camadas**

Quando há a adição de camadas escondidas a arquitetura passa a ser denominada de Rede *Feedforward* de Múltiplas Camadas ou *Perceptron* de Múltiplas Camadas (MLP, do nome em inglês). Essas camadas se situam entre a camada de entrada e de saída, e conferem à rede uma ordem mais elevada em relação aos dados de entrada. Isso acontece devido ao maior número de conexões e consequentemente, maior dimensão das interações. A Figura 7 mostra uma rede de múltiplas camadas contendo 10 nódulos de entrada, 4 neurônios escondidos e 2 neurônios de saída.

Outra característica importante desse tipo de rede é se suas camadas são completamente ou parcialmente conectadas. No exemplo da Figura 7, a rede é completamente conectada, pois todos os neurônios ou nódulos de uma camada possui conexão com todos os neurônios da próxima camada.

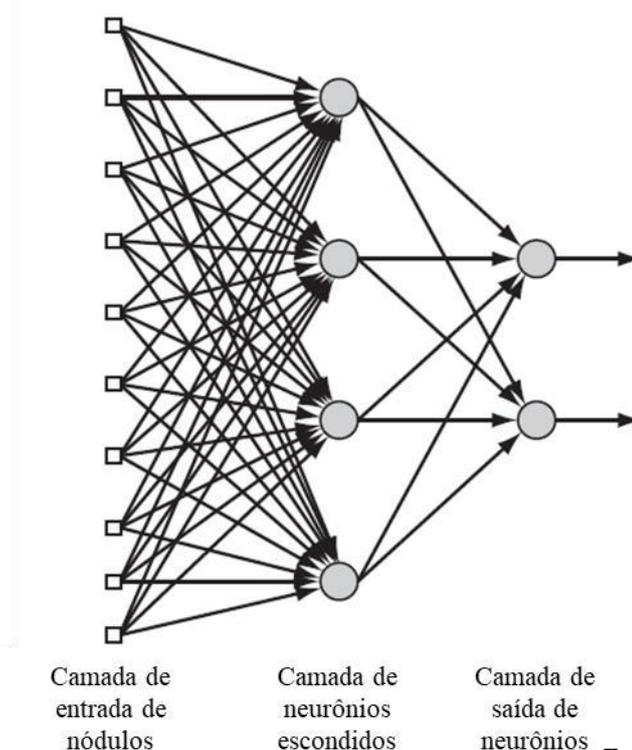
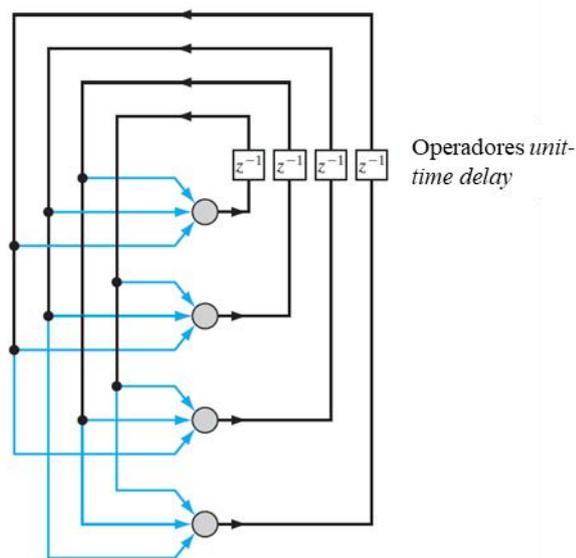


Figura 7 - Representação de uma Rede Feedforward de Múltiplas Camadas

### Redes recorrentes

Uma rede recorrente é caracterizada pela presença de pelo menos um *feedback loop*. Um *feedback loop* é definido pela presença de ramos com elementos de “*unit-time delay*”, que faz com que informações geradas na saída de um neurônio seja realimentado como entrada de outro neurônio. Os neurônios dessa rede podem conter *self-feedback loops* ou não, que se refere a situação em que a saída de um neurônio é realimentada à sua própria entrada. Sua presença tem profundo impacto na capacidade de aprendizagem da rede e na sua performance. Uma possível estrutura para uma rede recorrente é apresentada na Figura 8.



Fonte: HAYKIN (2009) - Adaptado

Figura 8 - Representação de uma Rede Recorrente

#### II.2.4. Treinamento de redes

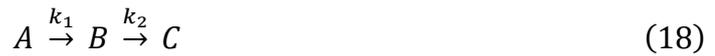
Existem dois métodos diferentes para o treinamento das redes: supervisionado e não supervisionado. No primeiro caso, é fornecido à rede conhecimento sobre o ambiente através de uma série de pares entrada-saída, que representam a resposta desejada. Os parâmetros da rede serão ajustados conforme o vetor de treinamento e o sinal de erro, definido como a diferença entre a resposta atual da rede e a desejada. Isso é realizado iterativamente de forma a minimizar os erros, atingindo um ótimo de forma estatística.

No treinamento não supervisionado não há informação sobre a resposta desejada da rede. Nesse caso, a rede identifica regularidades estatísticas nos dados de entrada e, com isso, cria classes de padrões diferentes presentes nos dados. O método supervisionado é o mais utilizado para redes neurais.

O algoritmo mais utilizado para o treinamento supervisionado de MLP's é a retropropagação ou '*backpropagation*'. Essa técnica foi popularizada através de um artigo de Rumelhart, Hinton e Williams (1986). O termo '*backpropagation*' é usado para descrever o treinamento de uma MLP usando gradiente descendente aplicado à uma função de erro de soma de quadrados. Sendo um algoritmo de treinamento, ele envolve um método iterativo de minimização de uma função de erro através de ajustes nos pesos sinápticos feitos através de duas etapas. Na primeira etapa há a avaliação das derivadas da função de erro em relação aos pesos; na segunda etapa, as derivadas são utilizadas para calcular os ajustes a serem feitos nos pesos. (BISHOP, 1995)

### II.3. Reator de Van de Vusse

O mecanismo de reação apresentado por Van de Vusse se baseia na produção de ciclopentenol (*B*) a partir do ciclopentadieno (*A*) por meio da adição eletrofílica ácido catalisadora de água na solução diluída. São obtidos como subprodutos da reação o dicitlopentadieno (*D*) e o ciclopentanediol (*C*), como pode ser visto no esquema reacional abaixo (VAN DE VUSSE, 1964):

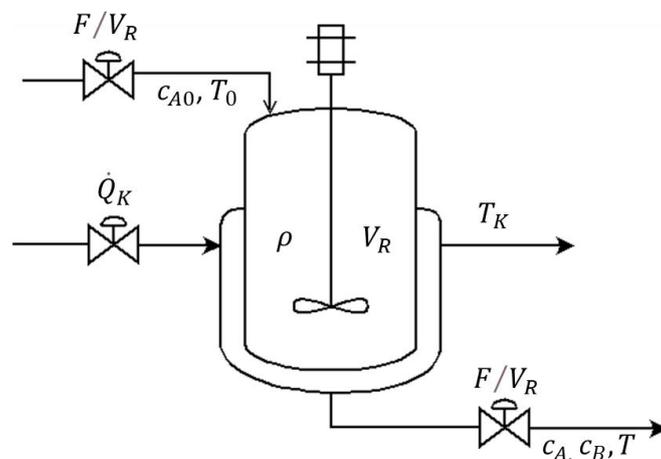


Klatt & Engell (1998) descreve melhor o modelo do reator de tanque agitado contínuo (CSTR) para esta reação. A corrente de entrada do tanque contém apenas A em baixas concentrações ( $c_{A0}$ ), a densidade da solução é constante e considera-se a distribuição do tempo de residência como ideal.

Para a representação dos parâmetros cinéticos, constantes e variáveis físicas, além dos componentes do sistema, utilizou-se a seguinte nomenclatura:

<b>Variáveis e Constantes</b>		<b>Índices</b>	
$c$	Concentração	$A$	Ciclopentadieno
$C_p$	Capacidade calorífica	$B$	Ciclopentanol
$F$	Vazão	$K$	Camisa do reator
$k_i$	Razão de coeficiente	$0$	Valor inicial
$t$	Tempo		
$V_R$	Volume de reator		
$T$	Temperatura		
$k_w$	Coeficiente de transferência de calor		
$A_R$	Superfície do reator		
$\dot{Q}_K$	Taxa de refrigeração		
$\Delta H_R$	Entalpia da reação		
$m_K$	Massa do refrigerador		
$\rho$	Densidade		

Em seguida, encontra-se a figura com o esquema do reator para a reação de Van de Vusse:



Fonte: KLATT & ENGELL (1998) - Adaptado

**Figura 9 - Representação esquemática do reator de Van de Vusse.**

As equações de balanço abaixo foram utilizadas para a modelagem do reator levando-se em conta as simplificações citadas anteriormente:

$$\frac{dc_A}{dt} = \frac{F}{V_R}(c_{A0} - c_A) - k_1c_A - k_3c_A^2 \quad (20)$$

$$\frac{dc_B}{dt} = \frac{F}{V_R}c_B + k_1c_A - k_2c_B \quad (21)$$

$$\frac{dT}{dt} = \frac{\dot{V}}{V_R}(T_0 - T) + \frac{k_w A_R}{\rho C_p V_R}(T_K - T) - \frac{k_1c_A\Delta H_R^{AB} + k_2c_B\Delta H_R^{BC} + k_3c_A^2\Delta H_R^{AD}}{\rho C_p} \quad (22)$$

$$\frac{dT_K}{dt} = \frac{1}{m_K C_{pK}}[\dot{Q}_K + k_w A_R(T - T_K)] \quad (23)$$

Os valores dos parâmetros cinéticos da reação assim como as propriedades do reator/camisinha são mostrados nas Tabelas 1 e 2.

**Tabela 1 - Parâmetros cinéticos da reação**

Reação	$k_i = k_{i0}e^{(E_i/T)}$			Entalpia de Reação $\Delta H_R$ (kJ / mol)
	$k_{i0}$	Unidade $k_{i0}$	Energia de Ativação $E_i$ (K)	
$A \xrightarrow{k_1} B$	$1,287 \times 10^{12}$	$h^{-1}$	-9758,3	4,2
$B \xrightarrow{k_2} C$	$1,287 \times 10^{12}$	$h^{-1}$	-9758,3	-11,0
$A + A \xrightarrow{k_3} D$	$9,043 \times 10^9$	$l/(mol \cdot h)$	-8560	-41,85

Fonte: KLATT & ENGELL (1998)

**Tabela 2 - Propriedades físico-químicas e dimensão do reator**

<b>Parâmetro</b>	<b>Valor</b>	<b>Unidade</b>
$\rho$	0,9342	$kg/l$
$C_p$	3,01	$kJ/(kg \cdot K)$
$k_w$	4032	$kJ/(m^2 \cdot h \cdot K)$
$A_R$	0,215	$m^2$
$V_R$	10	$l$
$m_K$	5,0	$kg$
$C_{pK}$	2,0	$kJ/(kg \cdot K)$

Fonte: KLATT & ENGELL (1998)

O reator com cinética de Van de Vusse vem sendo bastante aplicado nos estudos de técnicas de controle desde final do século passado, devido a sua característica não linear. Dentre esses estudos, podemos citar o trabalho de Kuntanapreeda (2012) que desenvolveu um sistema de controle *output-feedback* para regular reatores CSTRs utilizando a reação de Van de Vusse. Os resultados obtidos confirmam que o controlador proposto é capaz de controlar satisfatoriamente o processo apesar das variações dos parâmetros de controle variarem de seus valores nominais. Por fim, os autores concluem que os controladores podem ser utilizados com sucesso para reatores não isotérmicos.

Sheikir e Devera (1982) examinou a cinética de um reator CSTR isotérmico, homogêneo de Van de Vusse partindo dos controles de ondas senoidais e quadradas da concentração de alimentação. O trabalho permitiu ao autor concluir que as duas formas de controle alcançam um ciclo limite único, e que, para ambos, o tempo médio de seletividade dos produtos favorecem a geração de subprodutos, quando comparado ao estado estacionário.

Wu (1999) procurou descrever uma estrutura de controle adaptado não linear composto por linearização aproximada e por procedimentos de *backstepping design* que garante uma ótima qualidade da regulação das saídas quando existem dinâmicas internas instáveis e parâmetros desconhecidos. A eficiência da estrutura foi demonstrada em um reator Van de Vusse isotérmico, apresentando um bom desempenho, dentro do desejado pelo autor.

Já Ho *et al.* (2014), propuseram um controlador preditivo generalizado de dupla adaptação capaz de modelar e sintonizar sincronicamente em tempo real. Os estudos de simulação foram realizados em dois *benchmarks* de problemas de controle de processo, sendo um deles, o reator CSTR de Van de Vusse isotérmico. Os autores concluíram que o esquema desenvolvido apresentou melhorias significativas ao esquema abordado em trabalhos anteriores, além de proporem para estudos futuros o novo esquema capaz de lidar com sistemas multivariáveis.

## II.4. Aplicações

As redes neurais foram empregadas com sucesso em projetos de identificação de modelos de sistemas não-lineares. Dentre essas aplicações, podemos citar o trabalho de Assis (2001), que apresenta como as redes neurais aliado aos métodos de identificação e a teoria de controle são utilizadas para obter um melhor desempenho dos processos químicos. Foram apresentados dois processos, um sistema com dinâmica complexa de uma entrada e uma saída

(SISO) composto por dois reatores tanque conectados em série, onde ocorrem reações exotérmica e com troca térmica; e um sistema com múltiplas entradas e múltiplas saídas (MIMO), composto por um evaporador de duplo efeito.

Todos os resultados confirmaram o potencial uso das redes neuronais, como a NNARX e NNSSIF, principalmente quando aplicadas nas técnicas de controle preditivo.

Também podemos destacar o estudo de Yan-Juan *et al.* (2016), que propõe em seu trabalho uma rede neuronal para a identificação e otimização de um sistema de bomba de calor. Um algoritmo de controle auto operado é proposto, apresentando bons resultados no que se refere à performance do acompanhamento do set-point e à capacidade de rejeição à distúrbios. Ainda, o artigo compara o sistema de controle baseado em redes neuronais com as estratégias convencionais de controle, concluindo a superioridade do primeiro quando se trata de performance do sistema.

Para o reator de Van de Vusse, além dos estudos em controle, existem ainda alguns trabalhos que foram realizados a respeito da identificação do modelo utilizando redes neuronais.

Posser *et al.* (2000) desenvolveram um algoritmo para a geração de uma rede de modelo local (LMN) utilizando a reação de Van de Vusse como exemplo. Este algoritmo permite o desenvolvimento da rede para diferentes níveis de informações que se tem sobre a dinâmica do sistema. São propostas três soluções para contornar o efeito das mudanças nas variáveis pesos, e os autores concluem que a aplicação de uma função de transferência de primeira ordem que elimina essas variáveis ou o atraso delas produzem os melhores resultados. Já o trabalho de Zabirini *et al.* (2013) propõe um sistema de identificação não linear utilizando modelos de rede linear-plus-neuronal paralelo que fornece previsões mais acuradas em um comportamento de processo mesmo em regiões extrapoladas.

Ainda para o reator de Van de Vusse, Silva (2014) busca avaliar o desempenho de um controlador preditivo não linear multivariável baseado em redes neuronais. O resultado foi comparado com o de um controlador com previsões dadas pelo próprio modelo do processo (modelo perfeito). O autor conclui que o controlador baseado em redes neuronais apresenta bom desempenho em regiões com mudança de ganho, apesar de pequenos *offsets* em determinados casos. Já o controlador baseado em predição com o modelo perfeito, apesar de apresentar um desempenho superior, apresentou maior lentidão, de modo a necessitar de tempo de cômputo superior ao tempo de amostragem utilizado.

Partindo do estudo realizado por Silva (2014), o presente trabalho visa melhorar o desempenho das redes neuronais através do método de análise de sensibilidade, que será apresentado no próximo capítulo.

## Capítulo III – Metodologia

### III.1. Procedimento de identificação

Para a avaliação das redes neuronais como modelos paramétricos para a identificação do reator de Van de Vusse será primeiro definido um caso base que servirá como referência para uma análise de sensibilidade. Com isso será feita uma avaliação dos resultados, verificando como as mudanças afetam a performance do modelo e identificando os melhores casos. A partir disso, serão aplicadas modificações no projeto de forma a obter um caso melhorado. O desempenho desse modelo será então comparado com o encontrado para o caso base.

A análise de sensibilidade pode ser entendida como a medida local do efeito que uma dada entrada causa a uma certa saída. Os métodos locais são aqueles que exploram os resultados da variação de apenas um fator, enquanto os métodos globais abrangem a influência de todos os parâmetros simultaneamente. O principal objetivo de realizar esta análise é determinar qual variável precisa ser mais bem determinada e qual aquela que proporciona maior variação nos dados de saída. (SALTELLI, 2004)

Desta forma, a análise de sensibilidade foi feita utilizando o método local aplicado ao caso base. Ou seja, cada parâmetro de um conjunto de fatores escolhido teve o seu valor alterado enquanto os outros se mantiveram constantes. Neste caso, foi realizada a variação dos seguintes parâmetros:

- Tempo de amostragem (com e sem ruído)
- Quantidade de dados
- Quantidade de neurônios na camada escondida
- Quantidade de camadas escondidas
- Funções de ativação da camada escondida

Baseando-se no que foi visto na etapa de revisão bibliográfica sobre esse assunto, foram definidas as etapas para realização deste procedimento de identificação, conforme mostra a Figura 10, que será utilizado para todos os modelos a serem treinados. Para o caso base foi utilizado como principal referência o trabalho realizado por Silva (2014), no qual foram considerados os melhores resultados alcançados para a definição dos parâmetros destacados na Figura 10. Para se obter um caso base já aprimorado, os dois parâmetros restantes, sendo o tipo de sinal utilizado e o tempo de amostragem, foram selecionados de acordo com as propostas encontradas no livro de Aguirre, 2004.

## 1. Registro dos dados

- **Seleção da região de estudo**
- Escolha dos sinais de entrada
- **Definição da quantidade de dados**
- Definição do tempo de amostragem

## 2. Determinação do tipo de modelo

- **Quantidade de neurônios na camada escondida**
- **Quantidade de camadas escondidas**
- **Funções de ativação da camada escondida**

## 3. Validação do modelo

Figura 10: Etapas do procedimento de identificação

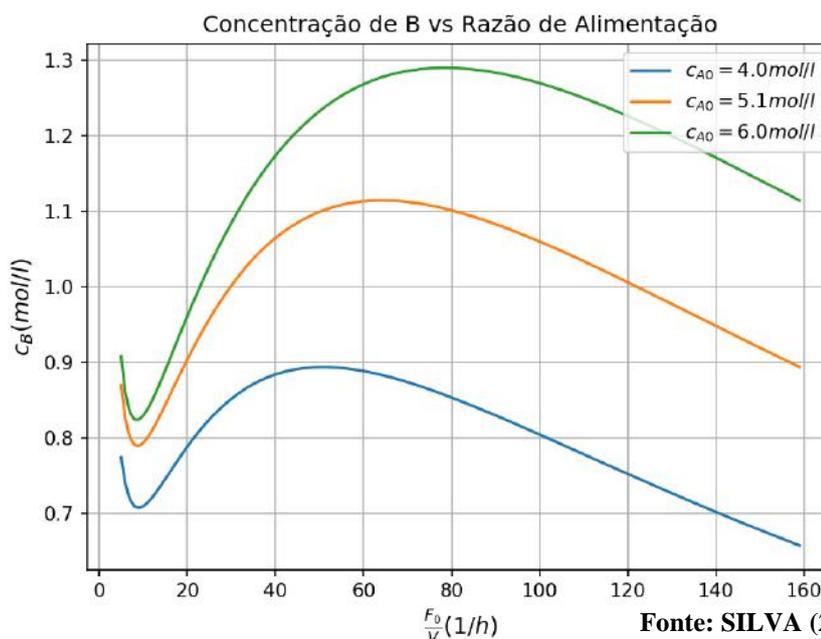
### III.1.1. Registro dos dados

#### III.1.1.1 Seleção da região de estudo

Serão consideradas como variáveis de saída a temperatura do reator ( $T$ ) e a concentração de B na saída ( $c_B$ ). Já as variáveis de entrada consistem na vazão da corrente de entrada do reator ( $F$ ), no calor retirado do reator ( $\dot{Q}_K$ ), na concentração inicial de A ( $c_{A0}$ ) e na temperatura da corrente de entrada ( $T_0$ ).

O experimento de identificação deve ser realizado quando o sistema apresenta modo de operação próximo ao desejado. O produto de interesse nesse caso é o componente B, portanto, quando maior for o  $c_B$  melhor é o resultado do processo.

foi realizado um estudo em estado estacionário para avaliar o comportamento do processo frente a mudanças na razão de alimentação ( $F_0/V_R$ ). Para isso, resolveram-se as equações diferenciais que representam o modelo do reator de Van de Vusse (Equações 20 a 23), através da aplicação do algoritmo *optimize.fsolve* da biblioteca SciPy. Foi fixada a temperatura de entrada ( $T_0$ ) em  $130^\circ\text{C}$ , o calor retirado do reator pela camisa ( $\dot{Q}_K$ ) em  $-4250 \text{ kJ/h}$  e utilizaram-se três valores para a concentração de entrada de A ( $c_{A0}$ ): 4.0, 5.1 e  $6.0 \text{ mol/l}$ . O resultado para a concentração de  $c_B$  é apresentado na Figura 11.



Fonte: SILVA (2014) - Adaptado  
**Figura 11 - Concentração de B no estado estacionário em função da razão de alimentação.**

Observando o gráfico pode-se perceber a característica de inversão de ganho do processo, no qual, para razões de alimentação muito baixas ou altas há a diminuição da concentração de B com o aumento da razão de alimentação, e para razões intermediárias há o aumento da concentração de B. Para o presente trabalho foi adotado o ponto com razão de alimentação  $60 \text{ h}^{-1}$ , com concentração inicial de A em  $5,1 \text{ mol/l}$ .

As variáveis de entrada serão a vazão de entrada do reator  $F_0$ , calor retirado do reator  $\dot{Q}_K$ , concentração de entrada de A  $c_{A0}$  e Temperatura de entrada do reator  $T_0$ . Os limites de cada variável utilizados para gerar os sinais de entrada foram selecionados conforme a Tabela 3.

**Tabela 3 - Definição dos limites de cada variável de entrada para a região de estudo**

Variável	Limite inferior	Limite superior	Unidade
$F_0$	300	900	$l/h$
$\dot{Q}_K$	-4500	-4000	$kJ/h$
$c_{A0}$	4	6	$mol/l$
$T_0$	100	150	$^{\circ}C$

### III.1.1.2 Escolha dos sinais de entrada

O sinal de entrada deve ser de uma classe intitulada persistentemente excitante, assim, o sinal de saída conterá informações suficientes sobre o sistema a ser estudado.

Para os sinais de entrada que serão utilizados como excitação do sistema serão utilizados dois tipos diferentes de sinais: o PRBS (*Pseudorandom Binary Sequence*) que é um sinal periódico que comuta entre dois níveis (Tabela 3), com um tempo mínimo de permanência ( $T_b$ ) em cada nível; e um sinal composto por números aleatórios gerados dentro de dois limites (Tabela 3), com tempo mínimo de permanência ( $T_b$ ) em cada valor, que será intitulado PRNBS (*Pseudorandom Non Binary Sequence*).

Foram desenvolvidos algoritmos em *python* para cada um dos dois sinais, de forma a permitir a sua aplicação para cada variável de entrada a partir da definição dos seus respectivos limites. Um dos parâmetros das funções é o  $T_b$ , no qual uma proposta feita por Aguirre (2004) para seu valor, em termos do tempo de amostragem  $T_s$ , é  $3T_s \leq T_b \leq 5T_s$ . Com isso, foi escolhido utilizar o valor de  $4T_s$  para  $T_b$ .

Assim, para cada incremento de tempo, que será definido pelo tempo de amostragem, os valores das variáveis de entrada eram atualizadas de acordo com o vetor do sinal de entrada gerado. E era realizada a integração das equações diferenciais do modelo (Equações 20 a 23), utilizando o algoritmo *integrate.odeint* da biblioteca SciPy, gerando assim as respostas de  $c_B$  e  $T$ .

### III.1.1.3. Definição da quantidade de dados

Uma amostra pode ser definida como um subconjunto de dados de um conjunto maior de dados. Esse conjunto maior de dados, dentro da estatística, é chamado de população. Uma amostra aleatória, por sua vez, é entendida como um conjunto que contém membros disponíveis da população que foram escolhidos com chances iguais para compor essa amostra. Quando uma amostra não representa de modo satisfatório a população, dizemos que houve um viés de amostragem. (BRUCE, PETER; BRUCE, ANDREW, 2019)

Ao realizar uma estimativa ou um modelo baseando-se em uma amostra, a qualidade dos dados, no geral, se mostra mais importante do que sua quantidade. A plotagem de dados e a inspeção manual são dificultados quando há dados em excesso, além de que, detectar valores faltantes e avaliar *outliers* em dados da ordem de milhões, pode se tornar um processo extremamente caro. A necessidade de uma quantidade massiva de dados ocorre quando os dados são esparsos, de modo que quanto mais se acumularem, melhores são os resultados.

A quantidade de dados utilizada no caso base foi  $N=1000$  pontos (Silva, 2014). Na análise de sensibilidade, a fim de averiguar a influência da quantidade de dados no desempenho do modelo, a rede foi treinada com valores superiores ao do caso base, que é considerado um número pequeno para o treinamento de uma rede. A Tabela 4 apresenta as quantidades testadas.

**Tabela 4 - Quantidade de dados utilizados para a análise de sensibilidade**

Quantidade de dados
10000
15000
50000
100000

### III.1.1.4 Definição do tempo de amostragem

Um período de amostragem muito grande, maior do que as constantes de tempo do sistema, pode gerar dados com pouca informação sobre a dinâmica do sistema. Já um período pequeno pode dificultar a redução de ruídos, tornando o dado menos informativo, além de ocorrer problemas de mal condicionamento numérico (LJUNG, 1999b; MELEIRO, 2002).

Diferentemente das outras seções onde o trabalho de Silva (2014) foi usado para definição do caso base foi considerado, o tempo de amostragem foi obtido pelo método de autocovariância explicado anteriormente na seção II.1.2.1.2.

Nessa abordagem, se faz necessário utilizar um sinal superamostrado do processo  $y^*(t)$ . Segundo Ljung (1999a), um tempo de amostragem ideal está em torno da constante de tempo do sistema. Dessa forma, será utilizado como base a menor constante de tempo do sistema para definição do tempo de amostragem do sinal superamostrado. Para garantir que o tempo de amostragem do sinal superamostrado  $y^*(t)$  ( $T_s^*$ ) seja pequeno o suficiente, o valor encontrado para a menor constante de tempo será dividido por 100.

Para encontrar a menor constante de tempo do sistema, foram realizadas simulações dinâmicas do processo. Para isso foi feita a integração das equações diferenciais do modelo do reator de Van de Vusse (Equações 20 a 23) utilizando o algoritmo *integrate.odeint* da biblioteca SciPy, obtendo as respostas das variáveis  $c_B$  e  $T$  em relação a funções degrau para todas as variáveis de entrada. As amplitudes dos degraus são exibidas na Tabela 5.

**Tabela 5 - Amplitudes dos degraus aplicados às variáveis de entrada**

Variável	Amplitude do degrau	Unidade
$c_{A0}$	-1,1	mol/l
$T_0$	20	°C
$F_0$	100	l/h
$\dot{Q}_K$	250	kJ/h

A determinação da resposta com dinâmica mais rápida foi feita a partir de uma avaliação visual. Em seguida, calcula-se a constante de tempo a partir do método gráfico. Para uma resposta ao degrau na entrada com comportamento aproximado a um modelo de 1° ordem, a resposta alcança 63,2% do seu valor final quando  $t = \tau$ , sendo  $\tau$  a constante de tempo do processo (SEBORG, 2011).

Com o valor de  $T_s^*$  determinado, é gerado um sinal de entrada do tipo PRBS com  $N=10000$ ,  $T_b=4$  e tempo de amostragem  $T_s^*$ . A partir dessa entrada, obtém-se um sinal de resposta do processo, que será avaliado a partir das Equações 1 e 2. Após gerar as curvas, aplicam-se os procedimentos descritos na seção II.1.2.1.2 para encontrar o tempo de amostragem ideal para ser utilizado no sinal de trabalho  $y(t)$  para o caso base.

Na análise de sensibilidade, para avaliar como o desempenho da rede muda com a alteração do tempo de amostragem selecionado, foram realizadas simulações variando-se esse parâmetro em torno do valor encontrado para o caso base

### III.1.1.3.1 Análise com ruído

Ao utilizar dados de processos reais é muito comum que eles contenham ruído, que pode vir de várias fontes como o instrumento de medição, o equipamento elétrico ou do próprio processo. A escolha do período de amostragem é influenciada também pelo ruído. Para uma razão sinal/ruído baixa, um tempo de amostragem muito pequeno pode evidenciar a influência do ruído nas mudanças medidas em detrimento do próprio processo. (SEBORG, 2011).

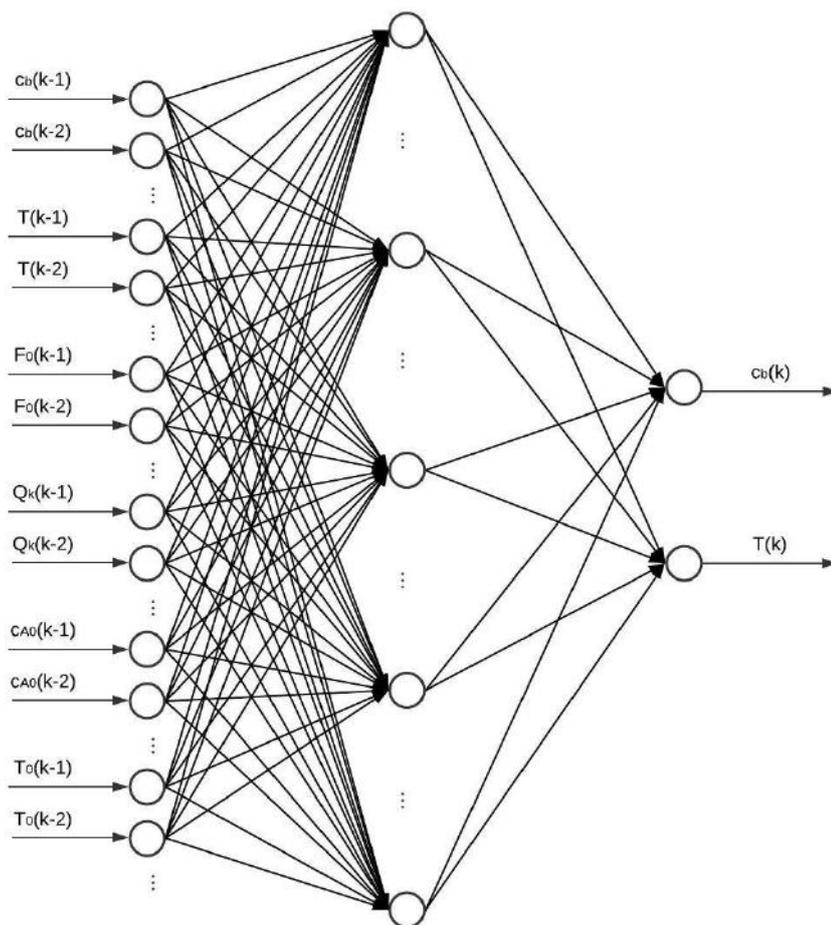
Para avaliar a performance de diferentes tempos de amostragem frente a um dado ruidoso, foram adicionados sinais que simulariam esse ruído às variáveis  $c_B$  e  $T$ . Os sinais foram construídos a partir de um gerador de distribuição pseudoaleatória uniforme, limitado a variações de 5% do valor médio de cada variável.

### III.1.2. Determinação do tipo de modelo

Conforme já mencionado, serão utilizadas redes neurais como modelo do experimento. Como comentando no item II.2.3, pode-se utilizar diferentes arquiteturas de rede neuronal para a identificação. No presente trabalho, optou-se pela rede *feedforward* de múltiplas camadas com dados de entrada atrasados. Tal arquitetura foi escolhida para o trabalho de Silva (2014) considerando as vantagens em se utilizar dados atrasados como entrada para a próxima predição frente ao uso de rede que fornece predições do futuro como saída, como apresentado por De Souza Jr. (1993). Além disso, Silva (2014) mostrou que uma rede com dois dados atrasados alcança um melhor desempenho do que uma rede com apenas um dado atrasado.

Desta forma, a rede foi estruturada de modo que a sua entrada fosse composta por dois dados atrasados das variáveis de entrada do sistema e saída ( $c_B, T, F_0, \dot{Q}_K, c_{A0}, T_0$ ). Já a saída da rede retorna as variáveis preditas  $c_B$  e  $T$  para o instante atual.

Foi utilizado para o treinamento o *keras*, uma biblioteca de *machine learning* da linguagem de programação Python. A Figura 12 abaixo ilustra a estrutura da rede treinada.



Fonte: SILVA (2014) - Adaptado

Figura 12 - Estrutura da rede neuronal utilizada

Para o caso base a rede possui função de ativação linear na camada de entrada e saída, e função hiperbólica na camada escondida. Além dos hiperparâmetros exibidos na Tabela 6.

**Tabela 6 - Hiperparâmetros da rede MLP para o caso base**

Dados atrasados	Função custo	Otimizador	Batch	Épocas	Número de neurônios nas camadas		
					Entrada	Escondida	Saída
2	MSE	Adam	50	150	12	13	2

A função custo utilizada na rede tem o propósito de quantificar o quão ruim é o desempenho do modelo durante o treinamento. Desta forma, é empregado um otimizador com o objetivo de minimizar a função custo. Para este trabalho, o algoritmo Adam foi escolhido devido a sua eficiência computacional e por requerer pouca memória. Já para as quantidades de amostras utilizadas para cada iteração da rede (*Batch*) e de tempo de aprendizado do algoritmo para cada amostra (Época), foram empregados 50 e 150, respectivamente.

Para a análise de sensibilidade foram avaliados os seguintes parâmetros: quantidade de neurônios na camada escondida, quantidade de camadas escondidas e função de ativação da camada escondida. As seções a seguir descrevem as variações impostas a esses parâmetros.

### III.1.2.1 Quantidade de neurônios na camada escondida

Para Zurada (1992), não existe um consenso acerca de um método ideal para se determinar a quantidade de neurônios a ser utilizada na camada escondida de uma rede feedforward. Essa é uma questão ligeiramente difícil de ser resolvida devido à grande complexidade do mapeamento da rede e à natureza não-determinística de diversos processos de treinamento de rede bem sucedidos. Desse modo, a determinação da quantidade ideal de neurônios é feita de modo empírico, e o que existe na literatura são apenas sugestões de abordagens para se chegar a esse número.

Entretanto, deve-se tomar cuidado para que a quantidade de neurônios não seja muito grande, de modo que a rede possa memorizar os dados de treinamento (*overfitting*), e nem muito pequena, de tal forma que a rede gaste tempo em excesso buscando encontrar uma representação ótima (HAYKIN, 2009)

Uma prática comum é dimensionar a quantidade proporcionalmente ao número de neurônios nas camadas de entrada e saída, formando um “funil” com cada vez menos neurônio em cada camada. Em geral, costuma ser mais vantajoso aumentar o número de camadas do que o número de neurônios por camada.

Desta forma, podemos calcular a média aritmética e a média geométrica entre a quantidade de neurônios da camada de entrada e da saída. Além disso, a fim de analisar a influência da quantidade de neurônios da camada escondida no desempenho da rede, foi realizado o teste com números superiores ao da última camada.

A Tabela 7 apresenta as quantidades testadas.

**Tabela 7 - Quantidades de neurônios nas camadas escondidas para a análise de sensibilidade**

<b>Quantidade de neurônios nas camadas escondidas</b>
5
7
20
50
100
200

### III.1.2.2 Quantidade de camadas escondidas

Segundo Géron (2019), apenas uma camada escondida em uma rede MLP pode ser o suficiente para modelar funções mais complexas, desde que contenha quantidade de neurônios suficiente. Entretanto, redes mais profundas, ou seja, composta por mais camadas, apresentam uma eficiência de parâmetros superior aos da rasa. Isso porque elas são capazes de realizar um treinamento muito mais rápido utilizando uma quantidade de neurônios exponencialmente inferior aos das redes superficiais.

Com isso, o autor sugere o uso inicial de uma ou duas camadas escondidas para problemas gerais. Para funções mais complexas, a ideia seria aumentar gradualmente o número de camadas até que o conjunto de treinamento seja sobre ajustado (*overfitting*).

Desta forma, a rede foi treinada com duas e três camadas para averiguar de que forma o aumento gradativo da quantidade de camadas escondidas impacta no desempenho da rede.

### III.1.2.3 Funções de ativação da camada escondida

A escolha da função de ativação a ser utilizada nos neurônios da camada escondida pode ser difícil, e ainda não existe um guia definitivo de princípios teóricos. É uma situação de tentativa e erro, adotando uma hipótese, realizando o treino e avaliando o resultado através da validação. Existem diversos tipos de funções de ativação, neste trabalho serão avaliados três dos mais utilizados: Sigmóide logística, tangente hiperbólica e unidade linear retificada (ReLU).

### III.1.3. Validação do modelo

Para Géron (2019), a melhor maneira de avaliar um modelo é separando seu conjunto de dados em conjunto de treinamento e conjunto de teste, sendo comumente utilizado 80% dos dados para o primeiro conjunto e 20% para o restante. Como os próprios nomes já dizem, os dados de treinamento são empregados para o treino do modelo. Após isso, o modelo é testado com o segundo conjunto e a taxa de erro é estimada. Esse valor irá informar se o modelo responderá bem a dados novos.

Entretanto, vale ressaltar a diferença entre seleção e avaliação do modelo. Segundo Hastier *et al.* (2017), a seleção do modelo é o processo de estimação da performance de diferentes modelos com objetivo de escolher o melhor. Para isso, seria necessário ter um

conjunto de validação. Depois de escolhido o modelo final, utiliza-se o conjunto de teste para estimar o erro de predição.

A escolha da proporção entre as três partes do *dataset* depende de alguns fatores, como a razão sinal/ruído e o tamanho da amostra de treino. Uma divisão comum é utilizar 50% para treinamento, 25% cada para validação e teste. (HASTIE *et al.*, 2017)

Como para o presente trabalho serão utilizados, no caso base, 1000 pontos (uma quantidade considerada pequena), foi escolhida a divisão do conjunto de dados como sendo 70% para treinamento, 15% cada para validação e teste.

Como métrica de avaliação foi utilizado o RMSE, definido pela Equação 9 e Índice de Ajuste, definido pela Equação 10.

# Capítulo IV – Resultados e Discussão

## IV.1. Caso base

### IV.1.1. Definição do tempo de amostragem

Foram gerados os gráficos que mostram as respostas de  $c_B$  e  $T$  para os degraus de entrada em  $c_{A0}$ ,  $T_0$ ,  $F$  e  $\dot{Q}_K$  com as amplitudes exibidas na Tabela 5. Esses gráficos são exibidos nas Figuras 13, 14, 15 e 16, respectivamente.

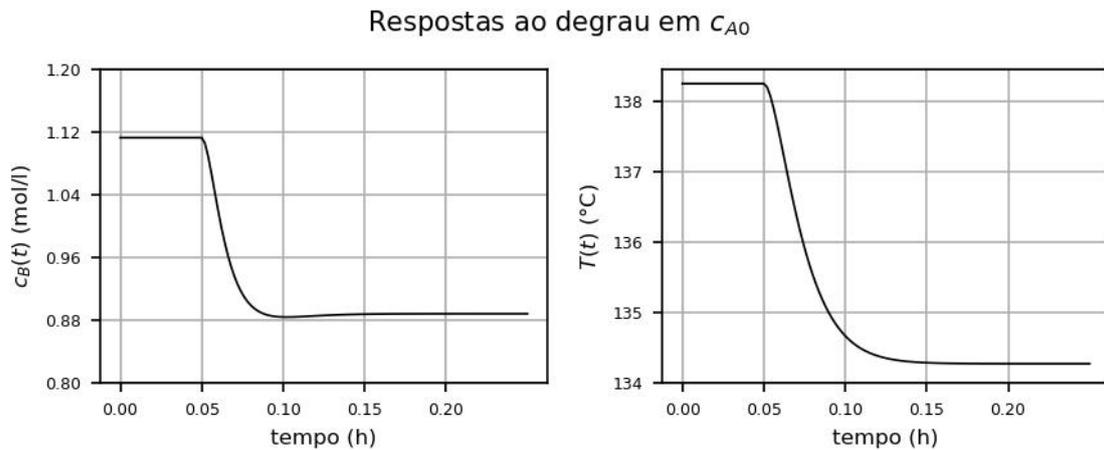


Figura 13 - Respostas de  $c_B$  e  $T$  em relação ao degrau aplicado em  $c_{A0}$ .



Figura 14 - Respostas de  $c_B$  e  $T$  em relação ao degrau aplicado em  $T_0$ .

### Respostas ao degrau em $F$

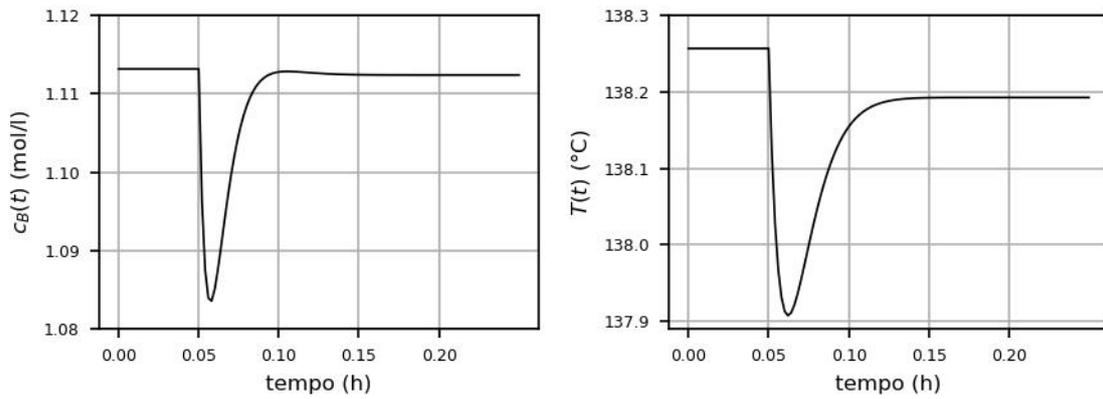


Figura 13 - Respostas de  $c_B$  e  $T$  em relação ao degrau aplicado em  $F$ .

### Respostas ao degrau em $\dot{Q}_K$

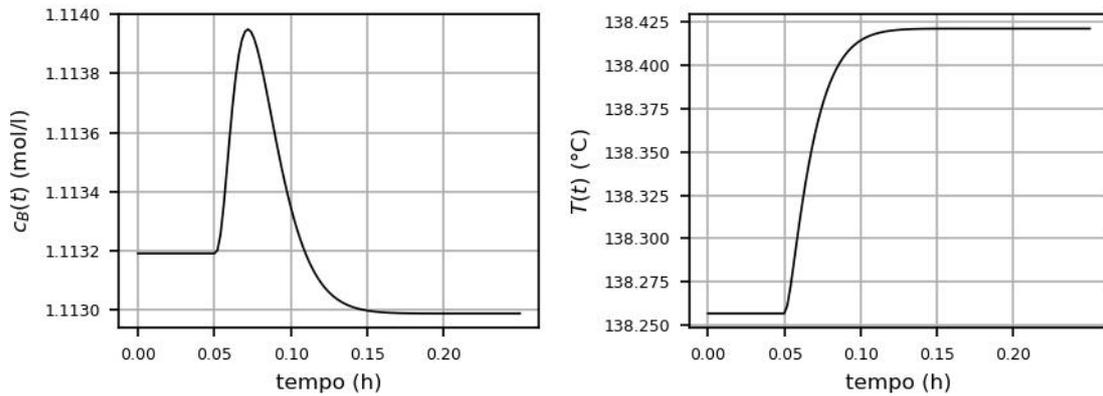
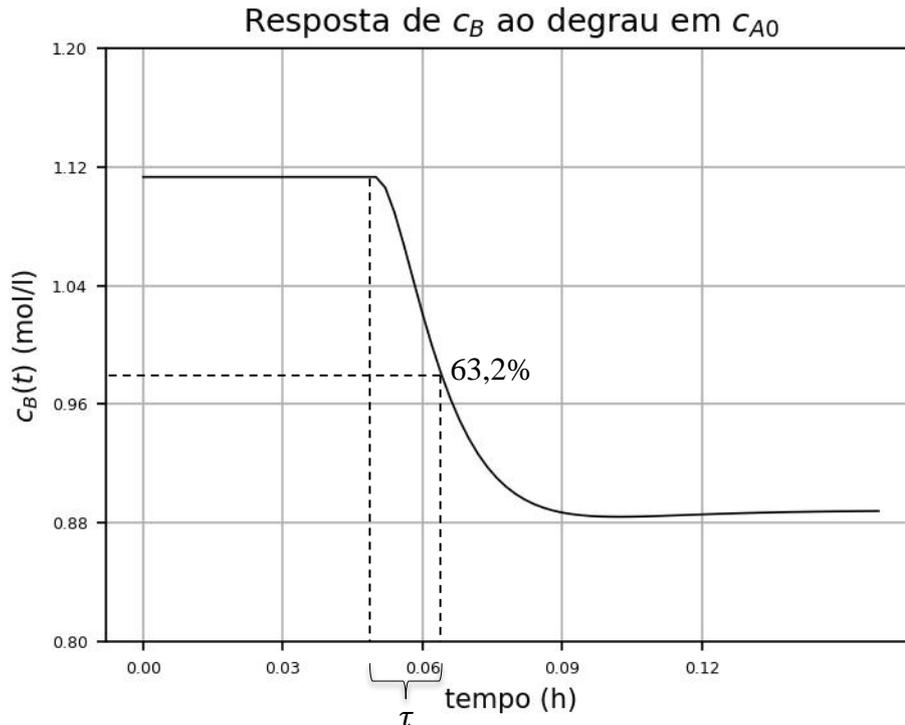


Figura 14 - Respostas de  $c_B$  e  $T$  em relação ao degrau aplicado em  $\dot{Q}_K$ .

A partir de uma análise visual dos gráficos, identificou-se que a resposta mais rápida foi a de  $c_B$  em relação ao degrau em  $c_{A0}$ . Dessa forma calcula-se a constante de tempo através do método gráfico conforme indicado na Figura 17.

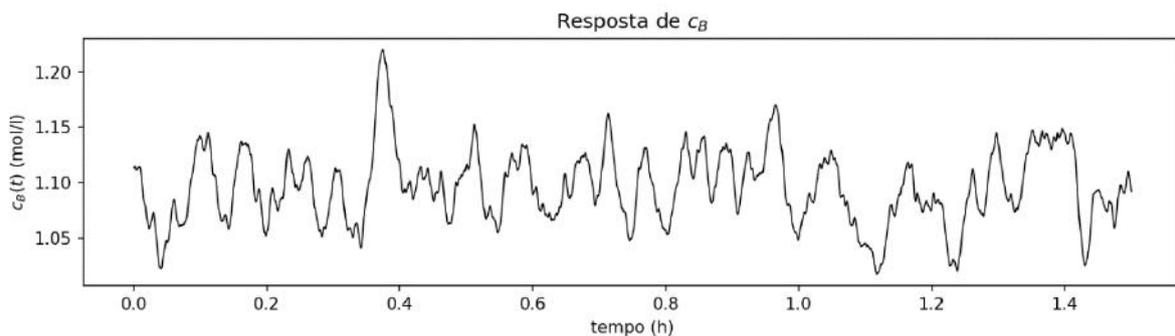


**Figura 17 - Representação do método gráfico para a obtenção da constante de tempo do sistema**

O  $\tau$  obtido foi de 0,015. Com isso calcula-se o tempo de amostragem para o sinal superamostrado:

$$T_s^* = \frac{\tau}{100} = 0,00015$$

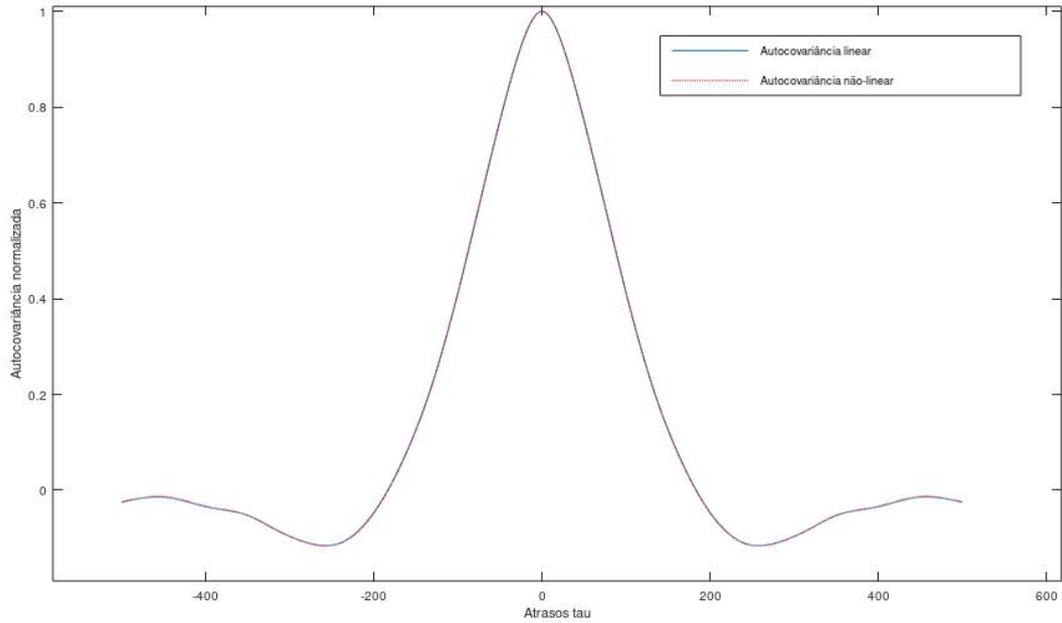
Esse  $T_s^*$  foi utilizado para gerar o sinal de entrada de  $c_{A0}$  como um PRBS, com parâmetros  $N=10000$  e  $T_b=4$ . A Figura 18 apresenta a resposta de  $c_B$  quando esse sinal é aplicado ao sistema.



**Figura 18 - Resposta de  $c_B$  em relação ao sinal de entrada PRBS**

O sinal de resposta foi então submetido às equações de autocovariância linear e não linear (Equações 1 e 2) para verificar a redundância entre duas observações consecutivas.

O gráfico obtido é mostrado na Figura 19.



**Figura 19 - Gráficos da autocovariância linear e não linear para o sinal de  $c_B$ .**

É possível identificar que o valor de  $\tau_m^*$  é aproximadamente 250. Dessa forma tem-se que  $10\Delta \leq 250 \leq 20\Delta$ , ou seja,  $12,5 \leq \Delta \leq 25$ . Escolhendo o valor intermediário de 20 para a taxa de decimação obtemos:

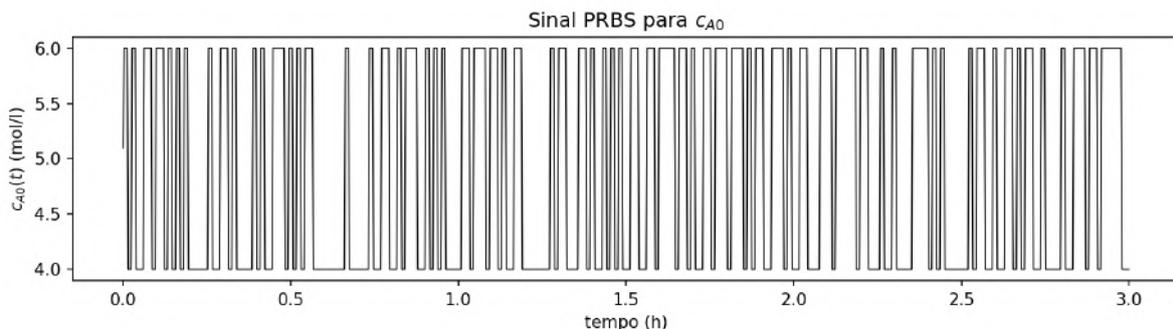
$$T_s = 0,00015 * 20 \approx 0,003$$

Dessa forma, de acordo com esse método, o valor ideal para o tempo de amostragem é  $T_s = 0,003$ .

#### IV.1.2. Geração dos sinais de entrada

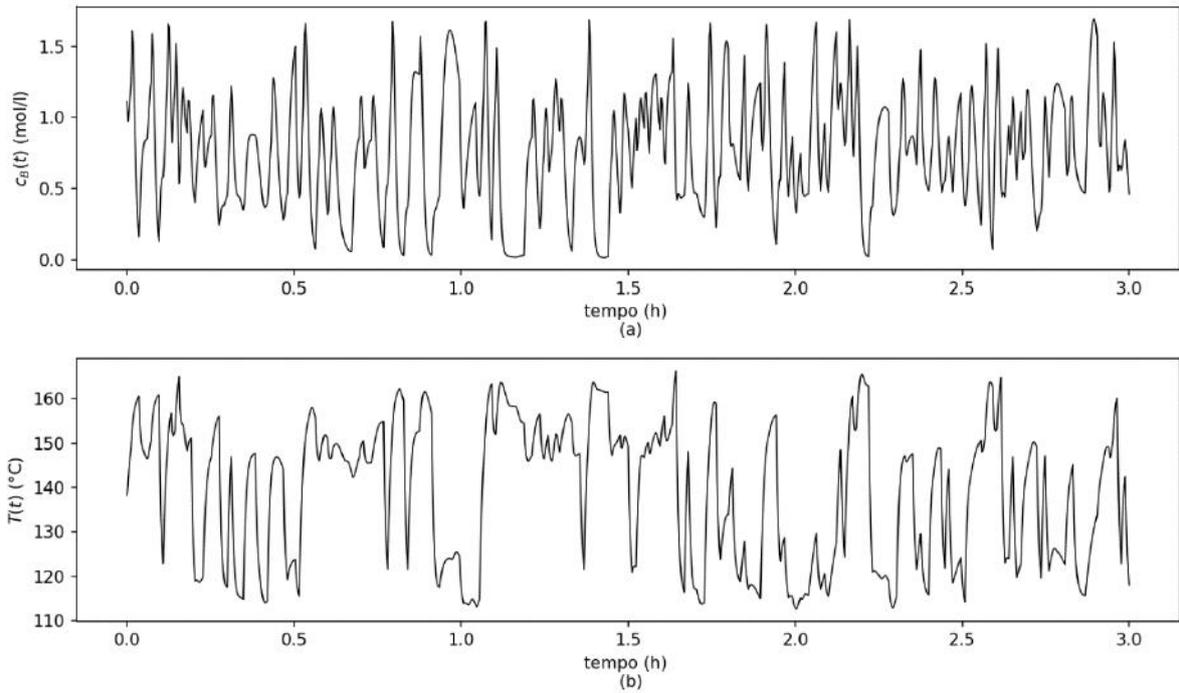
Conforme definido na seção III.1.1.2. foram gerados dois tipos de sinais diferentes para excitar o sistema: PRBS e PRNBS.

A Figura 20 exhibe o sinal PRBS gerado para a variável  $c_{A0}$  de forma a ilustrar o comportamento do sinal. São obtidos comportamentos semelhantes para as demais variáveis.



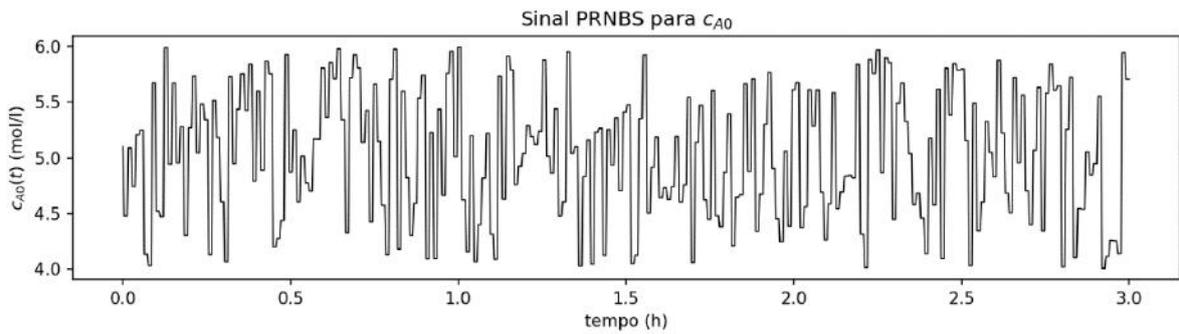
**Figura 20 - Gráfico do sinal PRBS gerado para a variável  $c_{A0}$**

A Figura 21 mostra a resposta do sistema para as variáveis de saída  $c_B$  e  $T$  quando excitado pelo sinal PRBS.



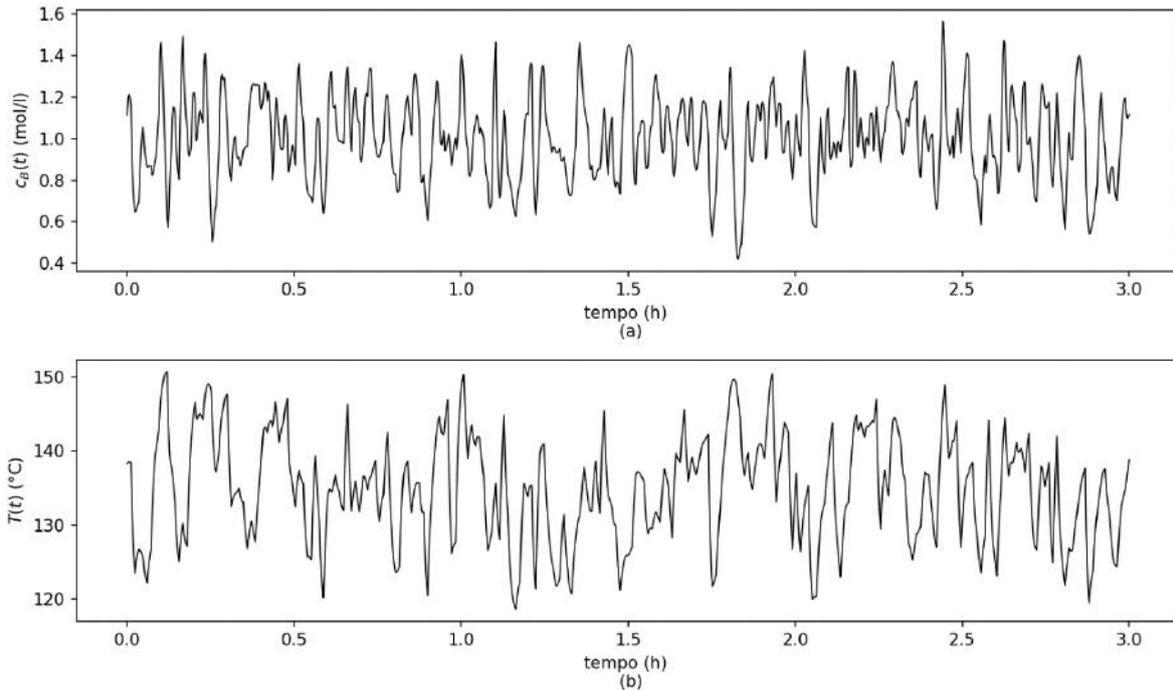
**Figura 21 - Resposta do sistema ao sinal PRBS. (a) Resposta de  $c_B$  e (b) resposta de  $T$ .**

Da mesma forma, a Figura 22 exibe o sinal PRNBS gerado para a variável  $c_{A0}$  de forma a ilustrar o comportamento do sinal. São obtidos comportamentos semelhantes para as demais variáveis.



**Figura 22 - Gráfico do sinal PRNBS gerado para a variável  $c_{A0}$ .**

A Figura 23 mostra a resposta do sistema para as variáveis de saída  $c_B$  e  $T$  quando excitado pelo sinal PRNBS.



**Figura 23 - Resposta do sistema ao sinal PRNBS. (a) Resposta de  $c_B$  e (b) resposta de  $T$ .**

#### IV.1.3. Validação do modelo

Aplicando as métricas definidas na seção III.1.3 ao conjunto de dados de validação do modelo obteve-se como resultado para o caso base os valores exibidos na Tabela 8.

**Tabela 8 - Resultados para métricas de avaliação do modelo para o caso base**

Sinal de entrada	$c_B$		$T$	
	RMSE	Índice de Ajuste	RMSE	Índice de Ajuste
<b>PRBS</b>	0,196	80,4	0,285	71,4
<b>PRNBS</b>	0,193	80,7	0,267	73,3

As Figuras 24 e 25 mostram gráficos comparando a resposta da predição dos modelos com o valor real das variáveis  $c_B$  e  $T$  para os experimentos com sinal de entrada PRBS e PRNBS, respectivamente.

Predição do modelo com entrada PRBS

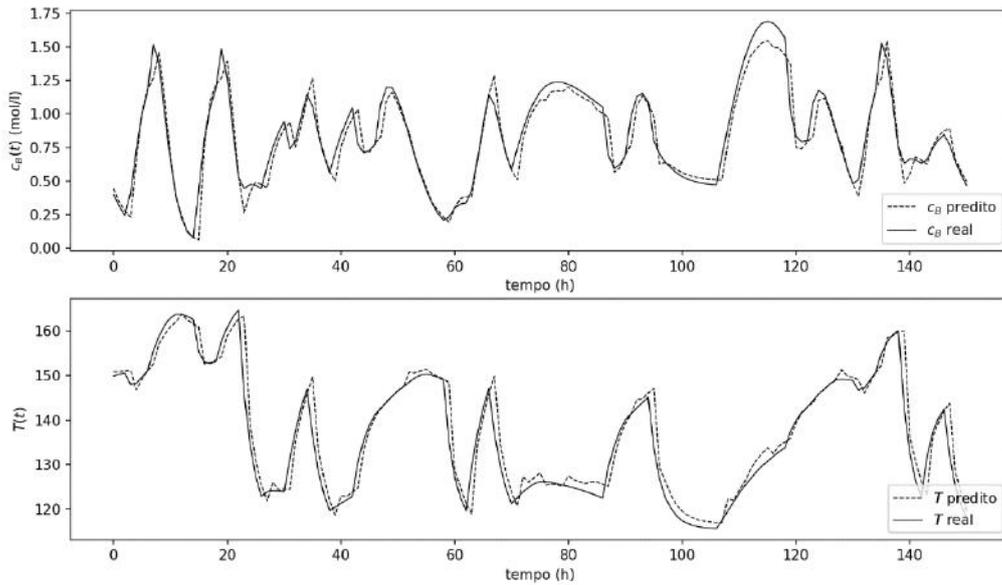


Figura 24 - Resultado da predição do modelo em relação ao valor real das variáveis para o sinal PRBS (caso base)

Predição do modelo com entrada PRNBS

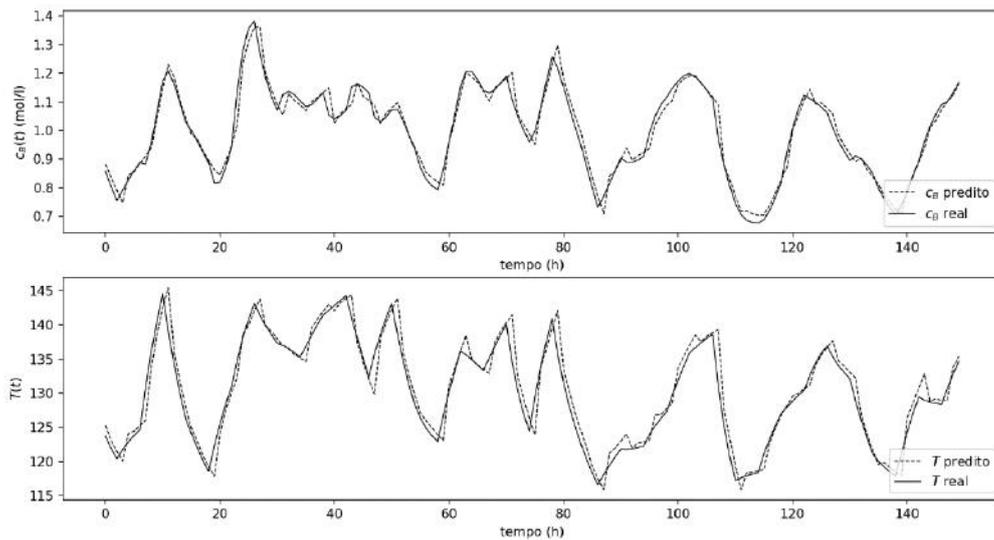


Figura 25 - Resultado da predição do modelo em relação ao valor real das variáveis para o sinal PRNBS (caso base).

Observa-se que os modelos tiveram um bom ajuste aos dados de forma geral, conseguindo prever os comportamentos de subida e descida, porém com atrasos em alguns momentos. O desempenho das redes para os dois sinais de entrada é bastante similar, sendo o resultado para o sinal PRNBS discretamente melhor. Isso pode ocorrer devido à natureza mais excitante do sinal PRNBS em relação ao PRBS.

## IV.2. Análise de Sensibilidade

### IV.2.1. Tempo de amostragem

Para a análise de sensibilidade foram selecionados os valores exibidos na Tabela 9 para tempos de amostragem. Dentre esses valores, destacam-se 0,00015 e 0,008. O valor de 0,00015 foi o encontrado para a menor constante de tempo do processo. Já o valor de 0,008 foi utilizado como tempo de amostragem no trabalho realizado por Silva (2014), sendo possível avaliar o ganho obtido ao se aplicar o método da autocovariância para definição desse parâmetro.

**Tabela 9 - Tempos de amostragem utilizados para a análise de sensibilidade**

<b>Tempo de amostragem (h)</b>
0,00015
0,001
0,008
0,01

Foram realizados os experimentos e os resultados para os sinais de entrada PRBS e PRNBS são exibidos nas Tabelas 10 e 11.

**Tabela 4 - Resultados do modelo para diferentes tempos de amostragem utilizando o sinal PRBS**

<b>PRBS</b>				
<b>Tempo de amostragem (h)</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>0,00015</b>	0,162	83,7	0,221	77,9
<b>0,001</b>	0,223	77,6	0,234	76,5
<b>0,003 (caso base)</b>	0,196	80,4	0,285	71,4
<b>0,008</b>	0,320	68,0	0,395	60,5
<b>0,01</b>	0,422	57,8	0,421	57,9

**Tabela 5 - Resultados do modelo para diferentes tempos de amostragem utilizando o sinal PRNBS.**

<b>PRNBS</b>				
<b>Tempo de amostragem (h)</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>0,00015</b>	0,132	86,7	0,119	88,1
<b>0,001</b>	0,172	82,7	0,187	81,2
<b>0,003 (caso base)</b>	0,193	80,7	0,267	73,3
<b>0,008</b>	0,251	74,8	0,372	62,8
<b>0,01</b>	0,302	69,8	0,406	59,4

Os valores obtidos para o RMSE de  $c_B$  e  $T$  indicam que um tempo de amostragem menor gera resultados melhores para o treinamento da rede. Observa-se uma grande diferença no desempenho da rede, mostrando a grande influência exercida por esse parâmetro no resultado.

Nesse caso é possível perceber que o caso base gerou um ganho para o modelo em relação ao valor de 0,008 utilizado por Silva (2014). Também o valor encontrado para a constante de tempo do sistema 0,00015 alcançou uma performance melhor do que o encontrado pelo método da autocovariância (caso base). Porém, é importante ressaltar, que valores muito pequenos para o tempo de amostragem tendem a evidenciar o ruído presente nos dados, que não foi considerado na geração desses resultados por utilizarem o modelo matemático do processo. Na próxima seção essa questão será abordada.

#### IV.2.1.1. Análise com ruído

Utilizando o método descrito na seção III.1.1.3.1 para adição de ruídos nos dados, foram gerados os resultados exibidos nas Tabelas 12 e 13.

**Tabela 6 - Resultados do modelo para diferentes tempos de amostragem utilizando o sinal PRBS com ruído.**

<b>PRBS</b>				
<b>Tempo de amostragem (h)</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>0,00015</b>	0,756	24,3	1,097	-9,75
<b>0,001</b>	0,338	66,2	0,579	42,0
<b>0,003 (caso base)</b>	0,293	70,7	0,414	58,6
<b>0,008</b>	0,380	62,0	0,433	56,6
<b>0,01</b>	0,452	54,8	0,451	54,8

**Tabela 7 - Resultados do modelo para diferentes tempos de amostragem utilizando o sinal PRNBS com ruído.**

<b>PRNBS</b>				
<b>Tempo de amostragem (h)</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>0,00015</b>	0,957	4,3	1,019	-1,9
<b>0,001</b>	0,366	63,4	0,782	21,8
<b>0,003 (caso base)</b>	0,301	69,9	0,561	43,8
<b>0,008</b>	0,314	68,5	0,494	50,5
<b>0,01</b>	0,366	63,3	0,514	48,5

Para o sinal PRBS o resultado indica que um tempo de amostragem intermediário, nesse caso 0,003, obteve um melhor desempenho. No caso do PRNBS, o tempo de 0,003 é melhor para a variável  $c_B$ , mas perde para tempos maiores para a variável  $T$ . Em ambos os

sinais é possível verificar um desempenho muito pior para o menor valor de período de amostragem 0,00015, o que corrobora com a hipótese do aumento da influência do ruído nessa situação.

É importante ressaltar o bom resultado alcançado pelo tempo de amostragem obtido através do método da autocovariância proposto por Aguirre (2004), validando a aplicação desse método.

#### IV.2.2. Quantidade de dados

Os resultados da análise de sensibilidade do modelo frente as alterações da quantidade de dados podem ser observadas na Tabela 14 e 15:

**Tabela 8 - Resultados do modelo para diferentes quantidades de dados utilizando o sinal PRBS.**

<b>PRBS</b>				
<b>Quantidade de dados</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>1000 (caso base)</b>	0,196	80,4	0,285	71,4
<b>10000</b>	0,167	83,2	0,185	81,5
<b>15000</b>	0,162	83,7	0,166	83,4
<b>50000</b>	0,161	83,9	0,175	82,5
<b>100000</b>	0,158	84,7	0,179	82,1

**Tabela 9 - Resultados do modelo para diferentes quantidades de dados utilizando o sinal PRNBS.**

<b>PRNBS</b>				
<b>Quantidade de dados</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>1000 (caso base)</b>	0,193	80,7	0,267	73,3
<b>10000</b>	0,202	79,8	0,190	80,9
<b>15000</b>	0,191	80,9	0,174	82,6
<b>50000</b>	0,177	82,3	0,179	82,1
<b>100000</b>	0,177	82,3	0,180	82,0

A partir dos resultados obtidos, é possível concluir que, no geral, quanto mais informações fornecemos a rede, isto é, quanto maior o volume de dados, menor é o RMSE e, consequentemente, maior é o índice de ajuste. Tal comportamento pode ser observado para os dois tipos de sinais, PRBS e PRNBS.

Entretanto, vale ressaltar que a partir de uma certa quantidade de dados, o aumento progressivo no volume não ocasiona um aumento significativo do índice de ajuste. Isso pode ser constatado para as variáveis  $c_B$ , a partir 10000 (PRBS) e 50000 pontos (PRNBS), e  $T$  com 15000 pontos (PRBS e PRNBS). Desta forma, é nítida a constatação de que um volume muito grande de dados, além de não trazer uma melhoria significativa nas métricas de avaliação de modelo, traz um custo computacional maior, de modo que o seu uso não seja vantajoso para o treinamento da rede.

Com base nisso, o cenário que apresentou o melhor resultado, utilizando a menor quantidade de dados possível dentro do conjunto de possibilidades apresentado, foi aquele utilizando 15000 dados (PRBS) e 50000 (PRNBS).

#### IV.2.3. Quantidade de neurônios na camada escondida

Utilizando diversas quantidades de neurônios na camada escondida, foi possível obter os seguintes resultados:

**Tabela 10 - Resultados do modelo para diferentes quantidades de neurônios na camada escondida utilizando o sinal PRBS.**

<b>PRBS</b>				
<b>Nº de neurônios na camada escondida</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>5</b>	0,198	80,2	0,301	69,9
<b>7</b>	0,200	80,0	0,292	70,8
<b>13 (caso base)</b>	0,196	80,4	0,285	71,4
<b>20</b>	0,196	80,4	0,289	71,1
<b>50</b>	0,202	79,8	0,279	72,1
<b>100</b>	0,200	80,0	0,288	71,2
<b>200</b>	0,203	79,7	0,275	72,5

**Tabela 11 - Resultados do modelo para diferentes quantidades de neurônios na camada escondida utilizando o sinal PRNBS.**

<b>PRNBS</b>				
<b>Nº de neurônios na camada escondida</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>5</b>	0,185	81,5	0,299	70,1
<b>7</b>	0,202	79,8	0,288	71,2
<b>13 (caso base)</b>	0,193	80,7	0,267	73,3
<b>20</b>	0,210	79,0	0,296	70,4
<b>50</b>	0,205	79,5	0,268	73,1
<b>100</b>	0,208	79,2	0,281	71,9
<b>200</b>	0,193	80,7	0,268	73,2

Nota-se que houve pouca variação nos índices conforme a adição e diminuição de neurônios na camada escondida, o que nos leva a concluir que a rede neuronal não é sensível a mudança no número de neurônios. É possível também concluir que para ambos os sinais, o caso base obteve os melhores resultados. Para Baughman & Liu (1995), a escolha da quantidade ótima de camadas e neurônios por camada da rede neuronal deve se basear na estrutura que fornecer o menor número com os melhores índices de desempenho.

#### IV.2.4. Quantidade de camadas escondidas

As Tabelas 18 e 19 apresentam os resultados utilizando diferentes quantidades de camadas escondidas:

**Tabela 12 - Resultados do modelo para diferentes quantidades de camadas escondidas utilizando o sinal PRBS.**

<b>PRBS</b>					
Nº de camadas escondidas	Nº de neurônios por camada	$C_B$		$T$	
		RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>1 (caso base)</b>	13	0,196	80,4	0,285	71,4
<b>2</b>	13-7	0,187	81,3	0,287	71,3
<b>3</b>	13-10-5	0,206	79,4	0,291	70,9

**Tabela 13 - Resultados do modelo para diferentes quantidades de camadas escondidas utilizando o sinal PRNBS.**

<b>PRNBS</b>					
Nº de camadas escondidas	Nº de neurônios por camada	$C_B$		$T$	
		RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>1 (caso base)</b>	13	0,193	80,7	0,267	73,3
<b>2</b>	13-7	0,187	81,3	0,290	71,0
<b>3</b>	13-10-5	0,198	80,2	0,298	70,2

Observando as métricas, é fácil perceber que não houve uma alteração significativa no desempenho da rede conforme o aumento do número de camadas. Apesar da pouca variação, fica evidente que os maiores índices de ajustes foram alcançados utilizando uma e duas camadas para os dois tipos de sinais.

#### IV.2.5. Função de ativação da camada escondida

Foram realizados experimentos alternando as funções de ativação da camada escondida entre Tanh, ReLU e Sigmóide. Os resultados são exibidos nas Tabelas 20 e 21.

**Tabela 14 - Resultados do modelo para diferentes funções de ativação da camada escondida utilizando o sinal PRBS.**

<b>PRBS</b>				
Função de ativação	$C_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>Tanh</b>	0,196	80,4	0,285	71,4
<b>ReLU</b>	0,214	78,5	0,261	73,9
<b>Sigmóide</b>	0,235	76,4	0,294	70,6

**Tabela 15 - Resultados do modelo para diferentes funções de ativação da camada escondida utilizando o sinal PRNBS.**

<b>PRNBS</b>				
<b>Função de ativação</b>	$c_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>Tanh</b>	0,193	80,7	0,267	73,3
<b>ReLU</b>	0,203	79,7	0,269	73,1
<b>Sigmóide</b>	0,236	76,3	0,270	73,0

De forma geral, há pouca variação na performance das redes em relação à mudança na função de ativação, indicando que não é um fator de grande influência no modelo. Porém, avaliando as mudanças, observa-se que para o sinal PRBS a melhor performance foi obtida com as funções Tanh e ReLU e para o sinal PRNBS, com a Tanh, porém com resultados próximos ao obtido com a ReLU. A pior performance para ambos os sinais foi obtida utilizando-se a função Sigmóide. Esses resultados condizem em parte com o que foi visto na literatura, sendo as funções Tanh e ReLU obtendo melhor desempenho do que a Sigmóide.

### IV.3. Caso melhorado

Com as avaliações dos resultados obtidos na seção IV.2., foram escolhidos os parâmetros que alcançaram melhores desempenhos para os dois tipos de sinais, exibidos na Tabela 22, para configurar um caso melhorado. Nas situações onde mais de um parâmetro obteve o melhor resultado, foi utilizado o somatório do índice de ajuste das duas variáveis, selecionando o maior.

No caso do tempo de amostragem, a seleção foi baseada no resultado gerado através da análise com ruído, por esse representar melhor um sistema real.

**Tabela 16 - Parâmetros selecionados para o caso melhorado dos dois tipos de sinais.**

<b>Parâmetro</b>	<b>Tipo de sinal</b>	
	PRBS	PRNBS
<b>Tempo de amostragem</b>	0,003	0,003
<b>Quantidade de dados</b>	15000	50000
<b>Quantidade de neurônios na camada escondida</b>	13	13
<b>Quantidade de camadas escondidas</b>	2	1
<b>Função de ativação da camada escondida</b>	ReLU	Tanh

Aplicando-se os parâmetros nos experimentos, foram realizadas as simulações, que geraram os resultados exibidos na Tabela 23. Os gráficos de comparação do sinal real e o predito pelos modelos do caso base e melhorado para o sinal PRBS é exibido na Figura 26 e para o sinal PRNBS na Figura 27.

Tabela 17 - Resultados para métricas de avaliação dos modelos

Sinal de entrada	$C_B$		$T$	
	RMSE	Índice de Ajuste (%)	RMSE	Índice de Ajuste (%)
<b>PRBS (caso base)</b>	0,196	80,4	0,285	71,4
<b>PRBS (caso melhorado)</b>	0,151	84,8	0,163	83,7
<b>PRNBS (caso base)</b>	0,193	80,7	0,267	73,3
<b>PRNBS (caso melhorado)</b>	0,184	81,5	0,180	82,0

Predição do modelo com entrada PRBS

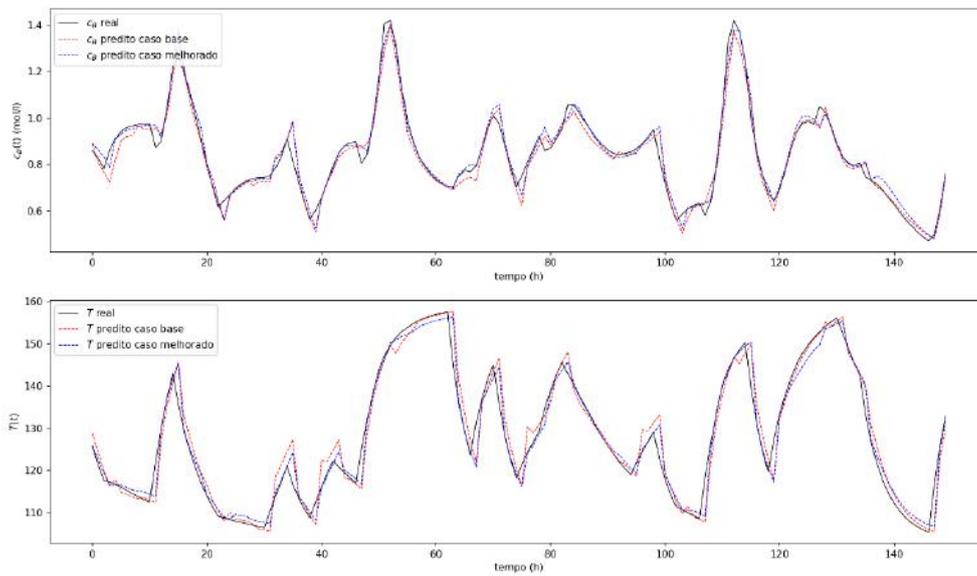
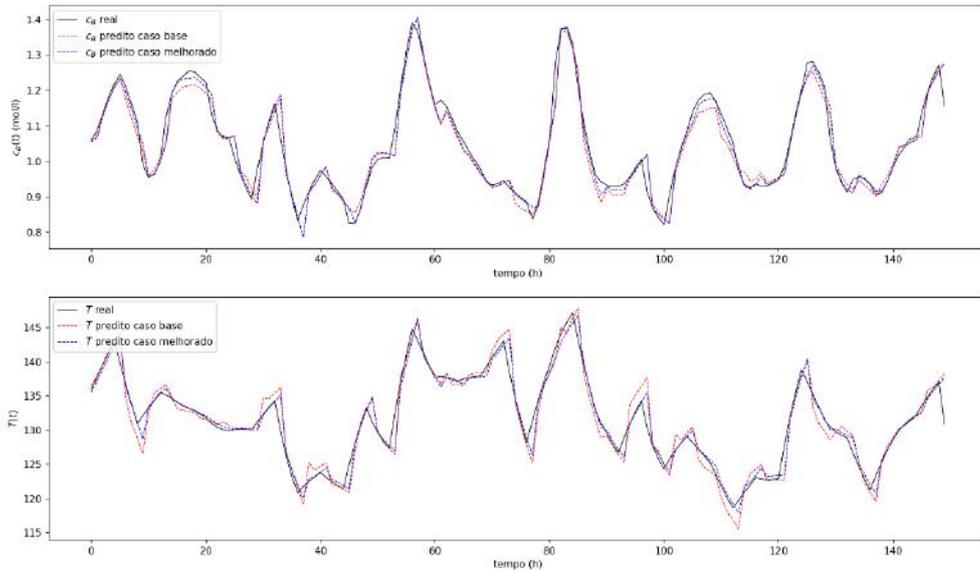


Figura 26 - Resultado da predição do modelo em relação ao valor real das variáveis para o sinal PRBS para o caso base e melhorado.

Predição do modelo com entrada PRNBS



**Figura 27 - Resultado da predição do modelo em relação ao valor real das variáveis para o sinal PRNBS para o caso base e melhorado.**

Observando os resultados percebe-se que houve uma melhora para os dois modelos em relação ao caso base. As Figuras 26 e 27 mostram um ajuste dos dados um pouco melhor para o caso melhorado, com valores mais próximos em situações de pico. Com os dados da Tabela 23, calcula-se que para o sinal PRBS houve uma melhora de 5,5% em relação à  $c_B$  e 17,2% em relação à  $T$ . Já para o sinal PRNBS, a melhora foi de 1% para  $c_B$  e 11,8% para  $T$ . Nota-se que o sinal PRBS obteve ganhos mais significativos do que o PRNBS, talvez por ter sofrido maiores modificações. Em relação às variáveis, houve uma maior sensibilidade aos ajustes no caso de  $T$ , com uma melhora mais expressiva do que para  $c_B$ , que apresentou pouca melhora.

## Capítulo V – Conclusão e sugestões para trabalhos futuros

O presente trabalho teve como objetivo estudar a identificação do reator de Van de Vusse, um processo complexo e com características não lineares, utilizando o recurso das redes neuronais *feedforward* para tal. Para isso, foi avaliado primeiro um caso base, para então realizar uma análise de sensibilidade dos parâmetros do experimento, revelando o impacto que eles causam na performance do modelo. Finalmente, com os resultados da análise foram propostas melhorias para os modelos configurando um caso melhorado.

A relevância do trabalho se dá ao fornecer, enquanto se avalia um caso pertinente à indústria de processos, uma avaliação quantitativa da influência das escolhas de parâmetros durante um experimento de identificação por redes neuronais. Muitos desses parâmetros não possuem na literatura, métodos ou boas práticas sobre sua seleção.

Com os resultados, pode-se dizer que as redes neuronais foram capazes de identificar o sistema, alcançando índices de ajuste em torno de 80%. Sendo uma opção atraente em casos onde o custo de levantamento do modelo é muito alto. Porém sua utilização depende do valor de incerteza permitido pela aplicação.

Quanto aos parâmetros do experimento, é possível perceber que há grande sensibilidade do modelo em relação a escolha do tempo de amostragem e quantidade de dados utilizada. Uma boa maneira de definir o tempo de amostragem é aplicando o método da autocovariância de Aguirre (2004), que revelou ser promissor para o uso em sistemas reais. Em relação à quantidade de dados, verificou-se que quanto maior a quantidade melhor é o desempenho, porém isso tende a atingir um platô para valores muito grandes.

Os parâmetros referentes à rede neuronal avaliados, sendo estes a quantidade de neurônios, número de camadas escondidas e função de ativação da camada escondida não geraram grandes alterações na performance do modelo ao serem modificados.

Ao se aplicar as modificações nas redes, houve pouca melhora, além de ser mais pronunciada para uma das variáveis de saída do processo. Dessa forma, uma sugestão para trabalhos futuros seria utilizar redes individuais para cada variável de forma a otimizar esses resultados.

Outra sugestão seria explorar mais a inclusão de *unit-time delay* no modelo, testando diferentes quantidades de dados atrasados ou até modelos de redes recorrentes, comparando seu resultado com os obtidos.

## Capítulo VI - Referências Bibliográficas

AGUIRRE, L. A. **Introdução à Identificação de Sistemas. Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais.** 2ª edição ed. Belo Horizonte: Editora UFMG, 2004.

BISHOP, C. M. **Neural Networks for Pattern Recognition.** Birmingham, UK: Clarendon Press, Oxford, 1995.

BRUCE, PETER; BRUCE, ANDREW. **Estatística Prática para Cientistas de Dados: 50 Conceitos Essenciais.** 1. ed. Rio de Janeiro: Altas Books, 2019.

BAUGHMAN, D. R.; LIU, Y. A. **Neural Networks in Bioprocessing and Chemical Engineering.** London, UK: Elsevier, 1995.

DE SOUZA JR, M. B. **Redes Neurais Artificiais na Engenharia Química, Apostila da disciplina de Redes Neurais,** 2005.

DE SOUZA JR, M.; TRICA, D. **Introdução a Modelagem e Dinâmica para Controle de Processos.** Rio de Janeiro, Rio de Janeiro: Publit, 2013.

GÉRON, AURÉLIEN. **Mãos à Obra Aprendizado de Máquina com Scikit-Learn & TensorFlow: Conceitos, Ferramentas e Técnicas para a Construção de Sistemas Inteligentes.** Rio de Janeiro: Altas Books, 2019.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning.** [s.l.] MIT Press, 2016.

HAYKIN, S. S. **Neural networks and learning machines.** 3rd ed ed. New York: Prentice Hall, 2009.

HO, Y. K. Generalized Predictive Control Algorithm with Real-Time Simultaneous Modeling and Tuning. **Industrial & Engineering Chemistry Research**, v. 53, p. 9411–9426, 2014.

KHALIL, H.K. **Nonlinear Control.** 1ª ed. Essex: Pearson Education Limited, 2015.

KLATT, K.-U.; ENGELL, S. Gain-scheduling trajectory control of a continuous stirred tank reactor. **Computers chem. Engng**, v. 22, p. 491–502, 1998.

KUNTANAPREEDA, S.; MARUSAK, P. M. Nonlinear extended output feedback control for CSTRs with van de Vusse reaction. **Computers & Chemical Engineering**, v. 41, p. 10–23, jun. 2012.

LJUNG, L. **System Identification\_ Theory for the User.** New Jersey, USA: Prentice Hall, 1999.

MELEIRO, L. A. DA C. **Projeto e Aplicação de Controladores baseados em Modelos Lineares, Neurais e Nebulosos.** Doutorado em Engenharia Química—Campinas, São Paulo: Universidade Estadual de Campinas, 2002.

OGUNNAIKE, B. A.; RAY, W. H. **Process dynamics, modeling, and control.** New York: Oxford University Press, 1994.

POSSER, M. S.; TRIERWEILER, J. O.; SECCHI, A. R. A Novel Algorithm to Local Model Network Generation. **IFAC Proceedings Volumes**, v. 33, n. 10, p. 383–388, jun. 2000.

SALTELLI, A. (ED.). **Sensitivity analysis in practice: a guide to assessing scientific models.** Hoboken, NJ: Wiley, 2004.

SEBORG, D. E. **Process dynamics and control.** 3rd ed ed. Hoboken, N.J: John Wiley & Sons, Inc, 2011.

SHANNON, C. E. A Mathematical Theory of Communication. **The Bell System Technical Journal**, v. 27, p. 55, 1948.

SILVA, G. DE C. **Estudo de um Controlador Preditivo Não Linear Multivariável Baseado em Redes Neurais.** Mestrado em Ciências—[s.l.] UFRJ, 2014.

SKEIRIK, R. D. Square-wave and sine-wave control of feed concentration in an isothermal CSTR with Van de Vusse kinetics. **Chemical Engineering Science**, v. 37, n. 4, p. 1015–1029, 1982.

VAN DE VUSSE, J.G. Plug-flow type reactor versus tank reactor. **Chemical Engineering Science**, v. 19, n. 12, p. 994–996, 1964.

WU, W. Adaptive nonlinear control of nonminimum-phase processes. **Chemical Engineering Science**, n. 54, p. 3815–3829, 18 jan. 1999.

YAN-JUAN, L. et al. **Identification and self-tuning control of heat pump system based on neural network**. 2016 Chinese Control and Decision Conference (CCDC). **Anais...** In: 2016 CHINESE CONTROL AND DECISION CONFERENCE (CCDC). Yinchuan, China: IEEE, maio 2016 Disponível em: <<http://ieeexplore.ieee.org/document/7532200/>>. Acesso em: 12 set. 2020

ZABIRI, H. et al. Integrated OBF-NN models with enhanced extrapolation capability for nonlinear systems. **Journal of Process Control**, v. 23, n. 10, p. 1562–1566, nov. 2013.