



**WESLLEN FERNANDO ANANIAS**

**USO DE REDES NEURAIAS ARTIFICIAIS PARA CONTROLE  
DO OXIGÊNIO DISSOLVIDO EM BIOPROCESSO AERÓBIO**

**LAVRAS - MG  
2021**

**WESLLEN FENRNADO ANANIAS**

**USO DE REDES NEURAIAS ARTIFICIAIS PARA CONTROLE DO OXIGÊNIO  
DISSOLVIDO EM BIOPROCESSO AERÓBIO**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Curso de Engenharia Química, para a obtenção do título de Bacharel.

Prof. Dr. Gilson Campani Junior  
Orientador

**LAVRAS – MG  
2021**

**WESLLEN FERNANDO ANANIAS**

**USO DE REDES NEURAS ARTIFICIAIS PARA CONTROLE DO OXIGÊNIO  
DISSOLVIDO EM BIOPROCESSO AERÓBIO**

**DISSOLVED OXYGEN CONTROL USING ARTIFICIAL NEURAL NETWORK IN  
AEROBIC BIOPROCESS**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Curso de Engenharia Química, para a obtenção do título de Bacharel.

Aprovada em 20 de maio de 2021.

Prof. Dr. Gilson Campani Junior  
Prof. Dr. Luciano Jacob Correa  
Dr. Thiago José Barbosa Mesquita

Prof. Dr. Gilson Campani Junior  
Orientador

**LAVRAS – MG  
2021**

*Aos meus pais Wilson e Valdirene com todo o meu amor,  
carinho e gratidão.*

*Dedico*

## AGRADECIMENTOS

Agradeço principalmente a Deus por me proporcionar sabedoria, força de vontade e a possibilidade de poder conquistar mais um sonho.

Aos meus pais Valdirene e Wilson, por terem me dado a oportunidade de me tornar quem eu sempre desejei ser, agradeço por serem e fazerem tanto por mim, por acreditarem e apoiarem sempre, por confiarem, se abdicarem e dedicarem, por demonstrarem o mais grandioso sentimento em todos os mínimos gestos.

Ao meu orientador, Gilson Campani Junior, pela orientação, incentivo, confiança e por ser sempre presente. É um orgulho ter desenvolvido esta dissertação sob sua orientação.

A todos os professores da Engenharia Química que dedicaram seu tempo e sua sabedoria em minha formação acadêmica.

À minha melhor amiga, Amanda Castilho e aos meus amigos: Bianca, Fernanda, Laura e Thayná, pelo companheirismo, por terem feito parte desta luta e terem ficado ao meu lado todos os dias.

E em especial aos meus amigos que estiveram comigo durante toda a minha descoberta na vida, Arthur, Fernanda, Isabella e João.

À Universidade Federal de Lavras.

MUITO OBRIGADO!

*“A inteligência é o único meio que possuímos  
para dominar os nossos instintos.”*

*(Sigmund Freud)*

## RESUMO

O oxigênio dissolvido (OD) é uma variável de processo que apresenta dinâmica tipicamente não linear em bioprocessos aeróbios, especialmente naqueles operados em regime transiente. Na literatura, o uso de métodos clássicos de controle como PID (Proporcional-Integral-Derivativo)/PI podem não ser uma alternativa satisfatória para as processos não lineares, pois baseiam-se em modelos linearizados e sistemas em regime estacionário. Portanto, torna-se necessária muitas vezes a utilização de técnicas avançadas, como controladores baseados em redes neurais artificiais (RNA), para o controle de sistemas não lineares, como a regulação do OD. Deste modo, o trabalho propôs o desenvolvimento de um controlador neural para o OD. Esse controlador inovador foi desenvolvido e implementado em simulações do processo aeróbio de produção de uma proteína recombinante pela bactéria *Escherichia coli*. Outros controladores como o PI e o preditivo DMC (*Dynamic Matrix Control*) foram também avaliados e comparados ao controlador RNA, perante aos desafios (a) controle servo, (b) perturbações internas, (c) perturbações externas e (d) ruídos nas medidas. Quanto ao desempenho, o controlador neural teve desempenho satisfatório para o controle do OD em ambas as fases do cultivo celular, crescimento e indução. De forma qualitativa, em nenhum dos desafios propostos a variável controlada divergiu do *setpoint*. Quantitativamente, o desempenho do controlador RNA foi estaticamente próximo ou até mesmo superior ao controlador tradicional PI na fase de crescimento e ao controlador DMC na fase de indução.

**Palavras-chave:** Redes Neurais Artificiais; Oxigênio Dissolvido; Bioprocesso; Controlador

## ABSTRACT

Dissolved oxygen (DO) is a process variable that has typically nonlinear dynamics in aerobic bioprocesses, especially those operated in transient regime. In the literature, the use of classic control methods such as PID (Proportional-Integral-Derivative)/PI may not be a satisfactory alternative for nonlinear processes, as they are based on linearized models and systems in steady state. Therefore, it is necessary to use advanced techniques, such as controllers based on artificial neural networks (ANN), to control nonlinear systems, such as the OD regulation. Thus, the work proposed the development of neural control for DO. This innovative control was developed and implemented in simulations of the aerobic process of producing a recombinant protein by genetically modified *Escherichia coli*. Other controllers such as PI and the predictive DMC (Dynamic Matrix Control) were also evaluated and compared to the RNA controller, considering specific scenarios of (a) step set point tracking, (b) step disturbance rejection, (c) plant-model mismatch and (d) measurement noise. Regarding the performance, the RNA control performed satisfactorily for the control of the DO level, in both phases of cell culture, growth and induction. Qualitatively, the controlled variable converged the *setpoint* for all proposed challenges. Quantitatively, the performance of the RNA controller was statistically close or even superior to the traditional PI controller in the growth phase, and to the DMC control in the induction phase.

**Keywords:** Artificial Neural Network; Dissolved Oxygen; Bioprocess; Controller

## LISTA DE FIGURAS

Figura 1 – Modelo esquemático do controle por retroalimentação em malha fechada.....	10
Figura 2 – Modelo esquemático do controlador preditivo .....	13
Figura 3 – Modelo esquemático de controladores adaptativos.....	14
Figura 4 – Modelo esquemático de controle preditivos adaptativos .....	15
Figura 5 – Modelo de um neurônio artificial.....	17
Figura 6 – Funções de ativação. a) e b) função linear retificada (ReLU) e sua forma gráfica. c) e d) função sigmóide (logística) e sua forma gráfica. e) e f) função tangente hiperbólica e sua forma gráfica. ....	19
Figura 7 – Diagrama blocos para treinamento redes neurais.....	21
Figura 8 – Exemplos de arquiteturas totalmente conectadas. a) Padrão <i>feedforward</i> . b) e c) RNA recursivas .....	23
Figura 9 – Curva de treinamento de rede. a) Curva com sobre-aprendizagem e b) treinamento normal desejado. ....	25
Figura 10 – Controle baseado em modelo de referência. ....	25
Figura 11 – Controle preditivo baseado em RNA .....	26
Figura 12 – Diagrama esquemático do método hierárquico EKF/MHE-DMC.....	28
Figura 13 – Modelo esquemático do bioprocesso controlado .....	30
Figura 14 – Modelo esquemático controle neural .....	32
Figura 15 – Ferramenta <i>nntraintool</i> .....	35
Figura 16 – Fluxograma arquitetura de rede .....	37
Figura 17 - Variáveis de saída para o conjunto de dados referentes à fase de crescimento. ....	39
Figura 18 - Variáveis manipuladas para o conjunto de dados referentes à fase de crescimento com <i>setpoint</i> para OD de 30%. ....	40
Figura 19 – Variáveis de saída para o conjunto de dados fase de indução.....	40
Figura 20 - Variáveis manipuladas para o conjunto de dados referentes à fase de indução com <i>setpoint</i> para OD de 40%. ....	41
Figura 21 – Perfis de <i>setpoint</i> aplicados à fase de crescimento.....	42
Figura 22 – Perfis de <i>setpoint</i> aplicados à fase de indução .....	42
Figura 23 – Critério de validação durante o treino da RNA QAr para a fase crescimento. ....	43
Figura 24 – Rede RNA QAr uma camada oculta para a fase de crescimento .....	44
Figura 25 –Arquitetura da rede RNA QAr com uma camada oculta para a fase de crescimento .....	45

Figura 26 - Rede RNA QAr com duas camadas ocultas para a fase de crescimento .....	46
Figura 27 – Arquitetura final da rede RNA QAr com duas camadas ocultas para a fase de crescimento .....	46
Figura 28 – (a) Variáveis de saída e (b) variáveis manipuladas para o sistema controlado pela RNA QAr durante a fase de crescimento.....	47
Figura 29 – Arquitetura final da rede RNA P para a fase de crescimento .....	48
Figura 30 – Arquitetura final da rede RNA QO2 para a fase de crescimento .....	48
Figura 31 – Variáveis de saída para a simulação do processo sujeito ao cenário modelo para a fase de crescimento .....	49
Figura 32 - Variáveis manipuladas para a simulação do processo sujeito ao cenário modelo para a fase de crescimento .....	49
Figura 33 – Arquitetura final da rede RNA QAr para a fase de indução .....	50
Figura 34 – Arquitetura da rede RNA P para a fase de indução .....	50
Figura 35 – Arquitetura final da rede RNA QO2 para a fase de indução.....	50
Figura 36 - Variáveis de saída para o controle servo modelo na fase de indução.....	51
Figura 37 - Variáveis Manipuladas para a simulação do processo sujeito ao cenário modelo para a fase de indução .....	51
Figura 38 – Desempenho do controlador PI no cenário modelo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) região (i) submetida a uma variação degrau, (c) região (ii) ao aplicar uma perturbação externa na pressão em 4 h seguido de variação em degrau no <i>setpoint</i> e (d) variáveis manipuladas decorrentes da ação do controlador PI.....	53
Figura 39 – Desempenho do controlador PI no cenário alternativo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) regiões (i) e (c) (ii) submetidas a variações degrau no <i>setpoint</i> , e (d) variáveis manipuladas decorrentes da ação do controlador PI.....	54
Figura 40 - Desempenho do controlador RNA no cenário modelo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) região (i) ao aplicar uma perturbação externa na vazão de ar em seguida submetido a uma variação degrau no <i>setpoint</i> , (c) resposta na região (ii) ao aplicar uma variação degrau no <i>setpoint</i> e (d) variáveis manipuladas decorrentes da ação do controlador RNA. (Continua).....	55
Figura 41 - Desempenho do controlador RNA no cenário alternativo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) regiões (i) e (c) (ii) submetidas	

a variações degrau no <i>setpoint</i> e (d) variáveis manipuladas decorrentes da ação do controlador RNA.....	57
Figura 42 - Desempenho do controlador DMC no cenário modelo durante a fase crescimento. Resultados das (a) das variáveis de saída e (b) das variáveis manipuladas decorrentes da simulação.....	59
Figura 43 - Desempenho do controlador DMC no cenário alternativo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) regiões (i) e (c) (ii) indicam a resposta do sistema frente a variações degrau no <i>setpoint</i> e (d) variáveis manipuladas decorrentes da ação do controlador DMC. (continua) .....	59
Figura 44 – Resultados das variáveis de saídas – (a), (b) e (c) – decorrentes da simulação, na fase de crescimento no cenário alternativo, dos controladores PI, RNA e DMC, respectivamente, assim como as variáveis de saída – (d), (e) e (f). .....	61
Figura 45 - Desempenho do controlador PI durante a fase de indução no cenário modelo. Resultados (a) das variáveis de saída, (b) região (i) quando aplicada uma variação degrau no <i>setpoint</i> e (c) variáveis manipuladas. ....	62
Figura 46 - Desempenho do controlador PI durante a fase indução no cenário alternativo. Resultados (a) das variáveis de saída, (b) região (i) quando aplicada uma variação degrau no <i>setpoint</i> e (c) variáveis manipuladas. ....	63
Figura 47 - Desempenho do controlador RNA durante a fase indução no cenário modelo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação degrau no <i>setpoint</i> e (c) variáveis manipuladas. ....	64
Figura 48 - Desempenho do controlador RNA durante a fase indução no cenário alternativo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação de <i>setpoint</i> e (c) variáveis manipuladas. ....	64
Figura 49 – Desempenho do controlador DMC durante a fase de indução no cenário modelo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação de <i>setpoint</i> e (c) variáveis manipuladas. ....	65
Figura 50 - Desempenho do controlador DMC durante a fase de indução no cenário alternativo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação de <i>setpoint</i> e (c) variáveis manipuladas. ....	65
Figura 51 - Resultados das variáveis de saídas – (a), (b) e (c) – decorrentes da simulação, na fase de indução no cenário alternativo, dos controladores PI, RNA e DMC, respectivamente, assim como as variáveis de saída – (d), (e) e (f). .....	67

## LISTA DE TABELAS

Tabela 1 – Descrição e valores dos parâmetros do modelo.....	9
Tabela 2 – Condições iniciais do processo.....	31
Tabela 3 – Configuração de entrada e saída de cada RNA.....	34
Tabela 4 – Valores parâmetros controlador PID.....	38
Tabela 5 – Resultados obtidos.....	45
Tabela 6 – Resultados obtidos.....	46
Tabela 7 – Resultado de erro integral médio ( $ISE \pm ISE$ ) para ensaios em triplicata.....	60
Tabela 8 – Resultado de erro integral médio ( $ISE \pm ISE$ ) para ensaios em triplicata.....	66
Tabela 9 – Síntese dos resultados obtidos a partir dos controladores PI, RNA e DMC para as fases de crescimento e indução.....	68

## SUMÁRIO

<b>1.</b>	<b>INTRODUÇÃO.....</b>	<b>1</b>
1.1	Objetivos.....	3
1.1.1	Objetivo geral.....	3
1.1.2	Objetivos específicos.....	3
<b>2.</b>	<b>REFERENCIAL TEÓRICO .....</b>	<b>4</b>
2.1	Biotecnologia e bioprocesso .....	4
2.1.1	Processo aeróbio: oxigênio .....	5
2.1.2	Modelo matemático do cultivo de <i>Escherichia coli</i> para a produção de PspA.....	6
2.2	Controle de processos .....	9
2.2.1	Controle clássico PI/PID .....	11
2.2.2	Controladores preditivos.....	12
2.2.3	Controles adaptativos .....	14
2.2.4	Inteligência artificial: redes neurais e lógica <i>fuzzy</i> .....	15
2.3	Redes neurais artificiais (RNA) .....	16
2.3.1	Fundamentos teóricos.....	17
2.3.2	Funções de ativação .....	18
2.3.3	Métodos de aprendizagem e treinamento de RNA .....	20
2.3.4	Aprendizado supervisionado por retropropagação de erros.....	21
2.3.5	Arquiteturas .....	23
2.3.6	Conjunto de treinamento, validação e teste .....	24
2.3.7	Aplicação RNA como lei de controle.....	25
2.4	Controle de oxigênio dissolvido .....	27
<b>3.</b>	<b>METODOLOGIA.....</b>	<b>30</b>
3.1	Simulação do processo.....	30
3.2	Aquisição do conjunto de dados .....	30
3.3	Estrutura de controle .....	31
3.4	Estrutura da rede.....	32
3.5	Variáveis de entrada e saída .....	33
3.6	Crítérios de decisão e desempenho.....	34
3.7	Fluxograma para definição da arquitetura da rede .....	36
3.8	Desafios de controle propostos .....	37
3.9	Controladores de referência .....	38
<b>4.</b>	<b>RESULTADOS E DISCUSSÃO.....</b>	<b>39</b>
4.1	Conjunto de dados de treino e validação.....	39
4.2	Conjunto de dados de teste: Cenário modelo e cenário alternativo.....	41

<b>4.3</b>	<b>Definição das arquiteturas das redes neurais .....</b>	<b>43</b>
<b>4.3.1</b>	<b>Teste e validação .....</b>	<b>43</b>
<b>4.3.2</b>	<b>Complexidade da rede vs. desempenho .....</b>	<b>44</b>
<b>4.3.3</b>	<b>Arquiteturas finais.....</b>	<b>48</b>
<b>4.4</b>	<b>Desempenho dos controladores para a fase de crescimento .....</b>	<b>52</b>
<b>4.4.1</b>	<b>Controlador PI.....</b>	<b>52</b>
<b>4.4.2</b>	<b>Controlador RNA .....</b>	<b>55</b>
<b>4.4.3</b>	<b>Controlador DMC .....</b>	<b>58</b>
<b>4.4.4</b>	<b>Comparação entre os controladores PI, RNA e DMC .....</b>	<b>60</b>
<b>4.5</b>	<b>Desempenho dos controladores para a fase de indução.....</b>	<b>62</b>
<b>4.5.1</b>	<b>Controlador PI.....</b>	<b>62</b>
<b>4.5.2</b>	<b>Controlador RNA .....</b>	<b>64</b>
<b>4.5.3</b>	<b>Controlador DMC .....</b>	<b>65</b>
<b>4.5.4</b>	<b>Comparação PI, RNA e DMC .....</b>	<b>66</b>
<b>4.5.5</b>	<b>Síntese dos resultados .....</b>	<b>68</b>
<b>5.</b>	<b>CONCLUSÃO.....</b>	<b>69</b>

## 1. INTRODUÇÃO

A área de bioprocessos está em constante crescimento, proporcionando o surgimento de novas tecnologias, como o uso de microrganismos geneticamente modificados para a produção de bioativos (medicamentos, vacinas, entre outros), ou o desenvolvimento de processos enzimáticos, fermentativos, aplicados em indústrias de alimentos, bebidas e biocombustíveis, sendo listadas pela Embrapa como áreas de investimento em biotecnologia (EMPRAPA, 2021).

Dada essa versatilidade de aplicações, que além de serem objeto de estudo, estima-se que o mercado de biotecnologia avaliado em U\$ 449,06 bilhões em 2019 deverá registrar uma taxa de crescimento anual composta (CARG) de 6,84% para o período de 2020-2026 (“Polaris Markert Research”, 2020), confirmando a importância e relevância dessa área no cenário global.

Todos esses processos e produtos obtidos por meio de bioprocessos tem em comum a aplicação de células, microrganismos e seus componentes, como organelas, enzimas, entre outros, onde fatores como temperatura, pressão, pH, nível de oxigênio dissolvido (OD) afetam diretamente o rendimento do processo (SHULER, 2010).

Tendo em vista a aplicação deste trabalho, baseado na produção de proteínas recombinantes a partir da bactéria *Escherichia coli*, processo esse que ocorre em um reator aeróbio, o oxigênio é um dos substratos chave do metabolismo celular desse microrganismo. Sendo assim, o controle do nível de oxigênio dissolvido (OD) é de extrema importância, pois também é utilizado para monitorar efetivamente todo o processo (CAMPANI et al., 2019).

Na literatura, o uso de métodos clássicos de controle como PID (Proporcional-Integral-Derivativo)/PI podem não ser uma alternativa satisfatória para as processos não-lineares, pois baseiam-se em modelos linearizados e sistemas em regime estacionário (SEBORG et al., 2017). Essas características não são observadas na grande maioria dos bioprocessos, que são geralmente conduzidos em modo batelada e baseados em fenômenos complexos com dinâmica não linear e dependente do tempo (ALEXANDER et al., 2020). O OD é uma variável típica que apresenta dinâmica altamente não linear em bioprocessos aeróbios, especialmente naqueles operados em regime transiente. Portanto, torna-se necessária a utilização de controles avançados como, por exemplo, preditivos baseados em modelos, adaptativos, ou, mais recentemente, controladores baseados em inteligência artificial, como lógica *fuzzy* e redes neurais artificiais (RNA) para controlar o OD nos níveis desejados (CAMPANI et al., 2019; AZWAR et al., 2006).

Neste contexto, a aplicação de técnicas de inteligência artificial no controle de sistemas não lineares tem sido cada vez mais utilizada, dado o caráter das RNA conseguirem aprender e reproduzir o comportamento de funções desconhecidas, sendo aplicadas tanto na modelagem de sistemas não-lineares para reproduzir a dinâmica do sistema, quanto no projeto de controladores (HAYKIN, 2001; BRAGA; CARVALHO; LUDERMIR, 2007).

Outra abordagem na utilização da RNA é treinar a rede para que ela seja capaz de mimetizar o comportamento de um controlador já existente, sendo a saída desejada para a rede correspondente à saída do controlador (MORAIS; et al., 2016). Algumas vantagens nessa metodologia podem ser citadas, por exemplo, substituir controles impraticáveis, reproduzir controle humano, além de as RNA serem capazes de filtrar características indesejáveis dos controladores convencionais, como a sobrelevação, oscilações do sinal (ruídos), entre outros (HAYKIN, 2001).

Pelo fato que as redes neurais podem admitir diversas arquiteturas, não há nenhuma expressão analítica que defina uma arquitetura ótima para cada aplicação (HAYKIN, 2001). Portanto, torna-se um método empírico a definição de cada parâmetro, como número de camadas ocultas, quantidade de neurônios por camada, funções de ativação, sendo preferível aquela que atende ao desempenho desejado com menor complexidade.

Deste modo, no presente trabalho buscou-se desenvolver um controlador avançado para o oxigênio dissolvido baseado em redes neurais, o qual pode ser também adaptado ao controle de outras variáveis em (bio)processos de interesse. Essa nova arquitetura de controle foi aplicada ao processo aeróbio de cultivo de *Escherichia coli* recombinante para a produção de uma proteína de superfície de pneumococos (PspA), que é um componente promissor para o desenvolvimento de novas vacinas contra o *Streptococcus pneumonia* (PERCIANI et al., 2013). As variáveis manipuladas para o controle do OD são vazão de ar, pressão e vazão de oxigênio, conforme descrito por Campani et al. (2019).

## **1.1 Objetivos**

### **1.1.1 Objetivo geral**

Desenvolver um controlador para o oxigênio dissolvido baseado em redes neurais artificiais.

### **1.1.2 Objetivos específicos**

- Treinamento de redes neurais para o controle do oxigênio dissolvido a partir de dados simulados do processo;
- Comparar o desempenho do controlador com diferentes arquiteturas de rede, variando sua complexidade, como número de camadas ocultas e número de neurônios em cada camada;
- Comparar o desempenho da rede neural obtida com controladores clássicos PI (Proporcional-Integral).

## 2. REFERENCIAL TEÓRICO

### 2.1 Biotecnologia e bioprocesso

Os vários conceitos da biotecnologia se referem ao uso de células, microrganismos, nativos ou modificados, ou sistemas bioquímicos em processos de produção de bens ou de prestação de serviços (SMITH-DOERR et al., 1998).

Pode-se listar aplicações nos diversos segmentos industriais, desde produtos para indústria farmacêutica (antibióticos, hormônios, vacinas, vitaminas e proteínas terapêuticas humanas), da indústria de alimentos (aminoácidos, flavorizantes, polissacarídeos e gomas xantana) e da indústria química (etanol, acetona, butanol, glicerol, ácido láctico, ácido cítrico, poliésteres, inseticidas microbianos) (PEREIRA, 2008).

Todas essas transformações ocorrem por meio de processos bioquímicos que ocorrem em biorreatores (ROSSI, 2006; THOMASI, 2010), onde as condições de operação devem ser plenamente controladas para que produzam os metabólitos de interesse (CASSIANO, 2003), visto que os microrganismos e as células devem ser mantidos num ambiente que estimule a produção dos metabólitos desejados ou que reduza metabólitos indesejados.

Entre as condições importantes durante o cultivo, pode-se citar parâmetros físico-químicos como temperatura, composição do meio, pH, velocidades de agitação e aeração, concentração do indutor de síntese de substâncias, período de cultivo, oxigenação e tempo de indução, que influenciam o crescimento das células e a produção de produtos de interesse (BERLEC et al., 2013; GAO, 2007; JESUINO, 2020).

Sendo assim, dada essa versatilidade de aplicações industriais, controlar e monitorar todas essas variáveis é de enorme importância, uma vez que contribui para a diminuição dos custos de produção, aumentando rendimento e mantendo a qualidade dos produtos (VELOSO, 2006).

Na última década, foi observado um aumento expressivo na quantidade de bioprodutos comerciais, especialmente metabólitos secundários e proteínas terapêuticas produzidas com tecnologia de DNA recombinante, tecnologia essa usada pela primeira vez na produção de insulina, em 1982, através de bactérias modificadas, resultando com ganhos de produtividade (PEREIRA, 2008).

Neste contexto, a bactéria *Escherichia coli* é o microrganismo mais utilizado para produzir proteínas recombinantes, não só por ser um dos microrganismos mais conhecidos, mas

também devido à sua capacidade de crescer em altas concentrações celulares, permitindo processos de elevado rendimento em via aeróbia ou anaeróbia (SCHMIDT, 2004).

Verificaram-se, ainda, significativas mudanças na configuração de biorreatores visando à melhoria de seu desempenho e de modo a assegurar operações com maior segurança. O modelo convencional, tipo tanque agitado e aerado, é dominante nas aplicações em bioprocessos (CERRI, 2009). Com relação aos modelos não convencionais, os biorreatores pneumáticos como o de coluna de bolhas e *airlift* são os mais aplicados (THOMASI, 2010). No biorreator *airlift*, a homogeneização e a transferências de massa, energia e quantidade de movimento são realizadas exclusivamente pela injeção de gás no fundo, o que proporciona uma eficiente dispersão gás-líquido pelo próprio gás (CHISTI, 1989). Além disso, devido à ausência de partes móveis, tem-se um equipamento de construção menos complexa, assim como há um menor risco de contaminação decorrente da inexistência do selo mecânico, menor gasto de energia e menor taxa de cisalhamento (HUANG; MCDONALD, 2009).

Portanto, percebe-se que, de modo geral, a biotecnologia transita entre as áreas fundamentais, como a bioquímica, genética, microbiologia e engenharia, a fim de se ter uma maior compreensão sobre os processos bioquímicos e em busca de melhores condições de desempenho. Deste modo, torna-se necessário o uso de outras ferramentas como a modelagem, simulação, o controle e a instrumentação para alcançar resultados eficientes e otimizados (PEREIRA, 2008).

Tendo em vista todos os fatores citados, desde o uso de bactérias recombinantes, escolha do biorreator e utilizando as ferramentas como modelagem, otimização e controle, estudos realizados por Campani (2018) empregaram o cultivo *Escherichia coli* geneticamente modificada para a produção de proteína recombinante, a proteína de superfície A do *Streptococcus pneumoniae* (PspA), em um reator *airlift* pressurizado, sendo esse o modelo utilizado para o objeto de estudo desse trabalho.

### **2.1.1 Processo aeróbio: oxigênio**

Em bioprocessos envolvendo microrganismos aeróbios, como cultivos de *E. coli*, que apresentam alta demanda por oxigênio (CAMPANI, 2018), o oxigênio frequentemente é o fator limitante no sucesso do cultivo (SCHMIDELL, 2001).

O oxigênio é o componente chave no final da cadeia respiratória, uma vez que este possibilita o armazenamento de energia química através da transformação das moléculas de ADP (difosfato de adenosina) em ATP (trifosfato de adenosina), as quais são fundamentais na

sobrevivência e também no surgimento de novas células, no processo de proliferação da biomassa microbiana (SCHMIDELL, 2001).

Devido à baixa solubilidade apresentada em condições normais de cultivo, o oxigênio deve ser alimentado continuamente, uma vez que não é possível suprir toda demanda necessária a uma cultura em uma única vez (PEREIRA, 2008). Na água, a 25°C e com 20,9% de oxigênio presente na corrente gasosa de alimentação, a concentração de saturação é de apenas 8,1 ( $\text{mgO}_2\cdot\text{L}^{-1}$ ), enquanto que utilizando oxigênio puro na corrente de alimentação a concentração é em torno de 40,3 ( $\text{mgO}_2\cdot\text{L}^{-1}$ ) (SCHMIDELL, 2001).

Em soluções aquosas a solubilidade do oxigênio diminui de acordo com a sua composição, em especial de sais dissolvidos. A solubilidade desse gás diminui também com o aumento da temperatura, entretanto, aumenta com aumento da pressão parcial de oxigênio ou com aumento da pressão total (SCHMIDELL, 2001). Algumas vezes emprega-se oxigênio puro ou misturado ao ar, o que resulta em melhorias na transferência de massa da fase gasosa para a líquida, com a conseqüente melhoria no desempenho (PEREIRA, 2008). Campani (2018), por exemplo, manipulou as vazões de ar e oxigênio puro, bem como a pressão do sistema para o controle e suplementação de oxigênio.

De modo geral, a taxa de acúmulo ou esgotamento de oxigênio no meio é dada pela diferença entre o que se fornece e o que é efetivamente utilizado pelas células. O suprimento está relacionado às variáveis de ordem física, tais como aquelas citadas anteriormente (vazão de ar, vazão de oxigênio puro e pressão do sistema) (PEREIRA, 2008), sendo o fenômeno regido pelas teorias clássicas de transferência de massa gás-líquido (SCHMIDELL, 2001). Já a demanda está ligada à fisiologia da célula e diz respeito à quantidade de oxigênio necessária ao cultivo, sendo uma grandeza proporcional à massa de células. (SCHMIDELL, 2001). Portanto, percebe-se que a dinâmica da concentração de oxigênio no meio é de caráter não linear, uma vez que tanto as condições operacionais como as biológicas afetam a sua dinâmica, tornando seu controle um desafio.

### **2.1.2 Modelo matemático do cultivo de *Escherichia coli* para a produção de PspA**

O desenvolvimento de modelos matemáticos, que permitam descrever processos, é essencial, uma vez que, normalmente, modelar um sistema e simular as suas condições de operação demanda menor custo e tempo do que realizar experiências de forma empírica. Além disso, existem razões de ordem prática, de segurança e éticas que tornam inviáveis a experimentação em determinados sistemas reais (JACQUEZ, 1998).

No caso do cultivo de *E. coli* para a produção de proteínas recombinantes, tem-se um conjunto de variáveis essenciais que precisam ser definidas, monitoradas e controladas de modo a atingir objetivos de reprodutibilidade e qualidade do processo. Exemplos de variáveis são concentração de nutrientes, células, produto, temperatura, pH, entre outras.

O modelo matemático utilizado para a simulação do processo neste trabalho foi desenvolvido por Campani (2018). O processo de cultivo para a produção de PspA possui duas fases principais: fase de crescimento celular, que tem por objetivo produzir as células que sintetizarão o produto de interesse (PspA), e fase de indução, durante a qual as células sintetizam a proteína recombinante induzidas pela substância isopropyl- $\beta$ -D-thiogalactopyranoside (IPTG) adicionada ao meio de cultivo. O modelo desenvolvido por Campani (2018) utiliza como premissa as equações de balanço material dos principais componentes do sistema (células  $C_x$ , substrato  $C_s$ , a proteína de interesse  $C_p$  e o nível de oxigênio dissolvido  $OD$ ), apresentadas na Equação 1.

Relações empíricas são usadas para descrever a velocidade específica de produção de célula ( $\mu$ ) (Equação 2). A perda de viabilidade celular presente na fase de indução, que é proporcional à temperatura, é representada pela Equação 3. Por fim, a Equação 4 representa a velocidade específica de síntese da proteína recombinante (PspA) ( $\mu_p$ ), considerando a produção de PspA decorrente da multiplicação celular, e o segundo termo representando o acúmulo intracelular do produto com limite superior dado por  $k_m\mu$ .

As Equações 5 e 6 representam a taxa de transferência de oxigênio (OTR, *Oxygen Transfer Rate*) e a taxa de consumo (OUR, *Oxygen Uptake Rate*), respectivamente, reportadas por Campani et al. (2017). A OTR é proporcional ao coeficiente volumétrico de transferência de oxigênio ( $k_La$ ), coeficiente esse dependente de propriedades físicas das fases gasosa e líquida, como densidade, viscosidade, difusividade e solubilidade oxigênio. Já a OUR é proporcional à taxa específica de consumo de oxigênio ( $q_{o_2}$ ) e à concentração de células ( $C_x$ ). A  $q_{o_2}$  expressa a demanda específica de oxigênio, resultante da relação entre taxa de consumo de oxigênio pela concentração celular somada ao coeficiente de manutenção celular ( $m_{o_2}$ ).

$$\left\{ \begin{array}{l} \frac{dC_x}{dt} = (\mu - \mu_m) \cdot C_x \\ \frac{dC_p}{dt} = \mu_p \cdot C_x \\ \frac{dC_s}{dt} = -\frac{\mu}{Y_{x/s}} C_x \\ \frac{dOD}{dt} = OTR - OUR \end{array} \right. \quad (1)$$

$$\mu = \begin{cases} (a_\mu T + b_\mu) \frac{S}{K_s + S}, & t < t_{ind} \\ (a_\mu T + b_\mu) \frac{S}{K_s + S} \exp \left[ -\beta \frac{C_p}{C_x} (T - T_{ref}) \right], & t \geq t_{ind} \end{cases} \quad (2)$$

$$\mu_m = \begin{cases} 0, & t < t_{ind} \\ a_{\mu m} T + b_{\mu m}, & t \geq t_{ind} \end{cases} \quad (3)$$

$$\mu_p = \begin{cases} 0, & t < t_{ind} \\ \mu \frac{C_p}{C_x} + \frac{1}{T_{px}} \left( k_m \mu - \frac{C_p}{C_x} \right), & t \geq t_{ind} \end{cases} \quad (4)$$

$$OTR = \frac{\overbrace{a_k}^{k_L a}}{1 + \exp \left[ -b_k \left( \frac{Q_{AIR} + Q_{O_2}}{V} - c_k \right) \right]} \theta^{(T-27)} (\varepsilon DO^* - DO) \quad (5)$$

$$OUR = \left[ \frac{\mu}{Y_{x/O_2}} + \overbrace{\left( a_q \cdot T^3 + b_q \right)}^{m_{O_2}(T)} \right] C_x \quad (6)$$

onde T é a temperatura em °C;  $a_\mu$ ,  $b_\mu$ ,  $a_{\mu m}$ ,  $b_{\mu m}$ ,  $k_m$ ,  $T_{px}$ ,  $\beta$  e  $T_{ref}$  são parâmetros empíricos do modelo;  $K_s$  é a constante de Monod;  $Y_{x/s}$  é o coeficiente de rendimento de produção de biomassa;  $t_{ind}$  é o tempo no qual se inicia a fase de indução. Nas Equações 5 e 6, os parâmetros  $a_k$ ,  $b_k$ ,  $c_k$ ,  $\theta$ ,  $a_q$ ,  $b_q$ ,  $\varepsilon$  fator de correção são empíricos;  $DO^*$  é a solubilidade média do oxigênio no meio, proporcional à pressão parcial de oxigênio de acordo lei de Henry;  $DO$  é o oxigênio dissolvido;  $Q_{Air}$  é a vazão de ar;  $Q_{O_2}$  é a vazão de oxigênio;  $V$  é o volume do biorreator;  $Y_{x/O_2}$  coeficiente rendimento de células baseado no oxigênio assimilado;

Os valores dos parâmetros presentes nas Equações 1-6 foram determinados por Campani et al. (2019), os quais são apresentados na Tabela 1 a seguir.

Tabela 1 – Descrição e valores dos parâmetros do modelo

Parâmetro	Descrição	Fase Crescimento	Fase Indução
$Y_{X/S}$	Rendimento biomassa/substrato ( $g_{ms}g_S^{-1}$ )	$0,45 \pm 0,01$	$0,7 \pm 0,2$
$a_\mu$	Constante empírica Equação 2 ( $^{\circ}Ch^{-1}$ )	$0,0256 \pm 0,0003$	$0,17 \pm 0,07$
$b_\mu$	Constante empírica Equação 2 ( $h^{-1}$ )	$-0,359 \pm 0,006$	$-4 \pm 2$
$K_S$	Constante de Monod ( $gL^{-1}$ )	$3,9 \pm 0,3$	$1,2 \pm 0,2$
$\beta$	Constante empírica Equação 2 ( $^{\circ}C^{-1}$ )	-	$2,3 \pm 0,6$
$T_{ref}$	Constante empírica Equação 2 ( $^{\circ}C^{-1}$ )	-	$26,8 \pm 0,8$
$a_{\mu m}$	Constante empírica Equação 3 ( $^{\circ}Ch^{-1}$ )	-	$0,009 \pm 0,004$
$b_{\mu m}$	Constante empírica Equação 3 ( $h^{-1}$ )	-	$-0,1 \pm 0,1$
$T_{px}$	Constante empírica Equação 4 ( $h$ )	-	$7 \pm 4$
$k_m$	Constante empírica Equação 4 ( $g_P g_{mS}^{-1} h$ )	-	$0,2 \pm 0,2$
$a_k$	Constante empírica Equação 5 ( $h^{-1}$ )	$200 \pm 20$	-
$b_k$	Constante empírica Equação 5 ( $min$ )	$1 \pm 0,1$	-
$c_k$	Constante empírica Equação 5 ( $min^{-1}$ )	$2,3 \pm 0,2$	-
$\theta$	Constante empírica Equação 5 ( <i>adimensional</i> )	$1,022 \pm 0,005$	-
$\varepsilon$	Fator de correção Equação 5 ( <i>adimensional</i> )	$0,69$	-
$Y_{X/O_2}$	Rendimento biomassa/oxigênio ( $g_{ms}g_{O_2}^{-1}$ )	$1,43 \pm 0,09$	-
$a_q$	Constante empírica Equação 6 ( $^{\circ}C^{-3}h^{-1}$ )	$(7,6 \pm 0,9) \times 10^{-6}$	-
$b_q$	Constante empírica Equação 6 ( $h^{-1}$ )	$-0,06 \pm 0,03$	-

Fonte: Campani et al. (2019)

## 2.2 Controle de processos

O objetivo primário do controle de processos é manter um processo em uma determinada condição de operação desejada, com segurança e de maneira eficiente, respeitando exigências ambientais e a qualidade do produto (NISE, 2009).

Do ponto de vista de sistemas biológicos, o controle do processo tem como principais objetivos melhorar a estabilidade de operação, aumentando assim a eficiência da produção e o lucro (LIM; LEE, 1990). O controle geralmente está associado a uma ou mais variáveis a partir da manipulação de outras. Segundo Ogata (2002), define-se a variável controlada como a resposta do sistema que deve ser mantida em um valor ou faixa de referência. Já a variável manipulada é a grandeza ou condição que interfere no estado da variável controlada. Por outro

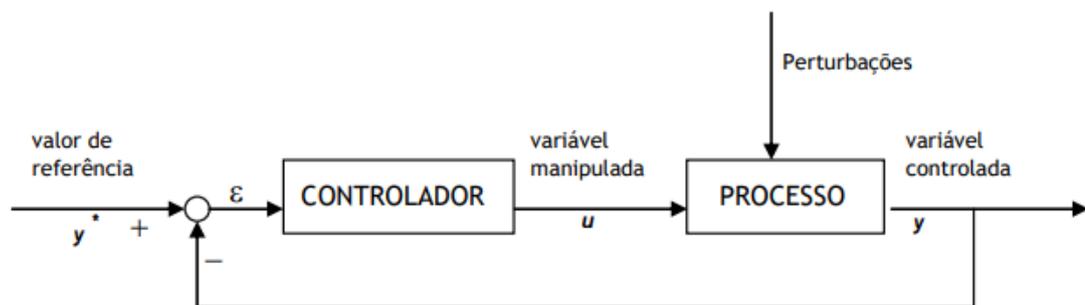
lado, consideram-se os distúrbios (variáveis de perturbação) grandezas ou condições que influenciam no equilíbrio dinâmico de um sistema, podendo ser previstos ou imprevistos durante a etapa de desenvolvimento do controlador (OGATA, 2002).

Em relação às estratégias de controle, pode-se citar o controle em malha aberta, definidas por Nice (2009) como sistemas não capazes de corrigir perturbações e que são comandados simplesmente pela entrada. Enquanto nos sistemas de malha fechada, ou controle *feedback*, a ação de controle é realizada em função do sinal de erro, gerado a partir da diferença entre a referência e a resposta do sistema (VELOSO, 2006).

Em malha fechada, embora não seja necessário um modelo matemático preciso do sistema, a ação de controle apenas ocorre após uma dada perturbação ter ocorrido, levando a desvios inevitáveis na variável controlada. Além disso, no controle em malha fechada são necessárias medidas online das variáveis de estado, o que pode não estar disponível na prática devido ao alto custo dos sensores ou até a indisponibilidade dos mesmos (CAMPANI, 2018).

A Figura 1 representa esquematicamente um controle por retroalimentação em malha fechada.

Figura 1 – Modelo esquemático do controle por retroalimentação em malha fechada



Legenda: O ciclo inicia-se pela medição da variável de saída  $y$ . É gerado um sinal erro,  $\varepsilon$ , subtraindo o valor medido da variável de saída,  $y$ , a um valor de referência desejado,  $y^*$ . O sinal de controle é calculado por aplicação de um dado algoritmo (lei de controle), que decide qual a variação da variável manipulada,  $u$ , de modo a compensar esse desvio e assim reduzir a diferença entre o valor da variável  $y$  e o seu valor de referência.

Fonte: Veloso (2006)

### 2.2.1 Controle clássico PI/PID

A técnica de controle Proporcional-Integral-Derivativo (PID) continua sendo uma estratégia muito popular em aplicações industriais, sendo empregada para a maioria dos problemas de controle enfrentados pela indústria, uma vez que o controlador PID provê suficiente habilidade e confiabilidade comprovadas em campo (MILLER et al., 1999).

No domínio do tempo, a ação de controle ( $u$ ) é dependente do erro entre a variável controla e o *setpoint* ( $e$ ), conforme Equação 7 (SEBORG et al., 2017).

$$u(t) = k_c \left[ e(t) + \frac{1}{\tau_i} \int_0^t e(t) dt + \tau_d \frac{de(t)}{dt} \right] \quad (7)$$

As ações de controle proporcional, integral e derivativo, representados pelos parâmetros  $k_p$ ,  $\tau_i$  e  $\tau_d$ , respectivamente, possibilitam a criação de leis de controle do tipo proporcional (P), o proporcional integral (PI) e o proporcional integral derivativo (PID). Ainda em alguns casos é ainda possível utilizar o controle proporcional derivativo (PD) (BEGA, et al., 2006).

A parcela proporcional ( $k_p$ ) fornece a resposta linear fixa em relação ao sinal de erro em uma malha fechada. Esta ação de controle acarreta um erro residual (*offset*) ao atingir o estado estacionário. Por outro lado, aumentando o ganho proporcional, torna-se a resposta mais rápida, diminuindo o *offset*. Entretanto, pode-se levar o sistema à instabilidade (OLIVEIRA, 1999).

A componente integral ( $\tau_i$ ) resulta em integrar o valor do erro no tempo, anulando o erro residual no estado estacionário. Devido a essa característica, na prática, utiliza-se a combinação proporcional-integral (PI). Por outro lado, esse tipo de configuração pode levar à saturação do controle e é recomendado para sistemas de dinâmica rápida sem atraso (BEGA et al., 2006).

Já com a componente derivativa ( $\tau_d$ ), é aplicada ao sistema uma correção proporcional à velocidade com que o desvio aumenta, resultando em uma antecipação do desvio que ainda não aconteceu. Essa configuração, geralmente conjugada com PI, é sensível a ruídos na medida, uma vez que a derivada dos erros pode ser muito elevada e caótica devido aos ruídos (OLIVEIRA, 1999).

Para obter os melhores valores dos parâmetros de um controlador PID, pode-se utilizar diferentes regras de sintonia. As regras de sintonia propostas por Ziegler-Nichols são bastante difundidas, cujos resultados podem servir de ponto de partida para refinar os parâmetros do

controlador na planta (OGATA, 2002). O método de sintonia online proposto por Ziegler-Nichols se baseia na determinação do ganho crítico ( $K_u$ ) e do período crítico ( $P_u$ ). O ganho crítico é o máximo valor para um controlador com ação proporcional quando se obtém uma oscilação estável. O período obtido para este valor determinado do ganho é denominado período crítico. Este método é conhecido também por método do ganho limite (BEGA et al, 2006).

Uma forma de implementar o controlador PID discreto em meios digitais é através da forma de velocidade (*velocity form*), aproximando os termos integrais e derivativos por diferenças finitas, sendo  $u_k$  e  $u_{k-1}$  as ações do controlador no instante  $t_k$  e no instante anterior  $t_{k-1}$ , e  $e_k, e_{k-1}$  e  $e_{k-2}$  os erros nos respectivos instantes. Nessa abordagem, a variação de saída do controlador ( $\Delta u$ ) é calculada pela Equação 8 (SEBORG et al., 2017):

$$\Delta u = u_k - u_{k-1} = k_c \left[ (e_k - e_{k-1}) + \frac{\Delta t}{\tau_i} e_k + \frac{\tau_d}{\Delta t} (e_k - 2e_{k-1} - e_{k-2}) \right] \quad (8)$$

Apesar do controlador PID ser o mais utilizado devido à sua simplicidade, robustez e versatilidade, ele pode apresentar algumas limitações relacionadas a problemas de sintonia ou não linearidades no sistema a ser controlado, visto que se baseia em modelos linearizados e sistemas em regime estacionário (SEBORG et al., 2017).

Conforme comentado anteriormente, a maioria das variáveis características de bioprocessos apresentam dinâmica não linear, visto que os mesmos são geralmente conduzidos em modo batelada e são baseados em fenômenos complexos (ALEXANDER et al., 2020). Nesses casos podem utilizar-se controladores avançados, como os controladores adaptativos e preditivos (LIM; LEE, 1990).

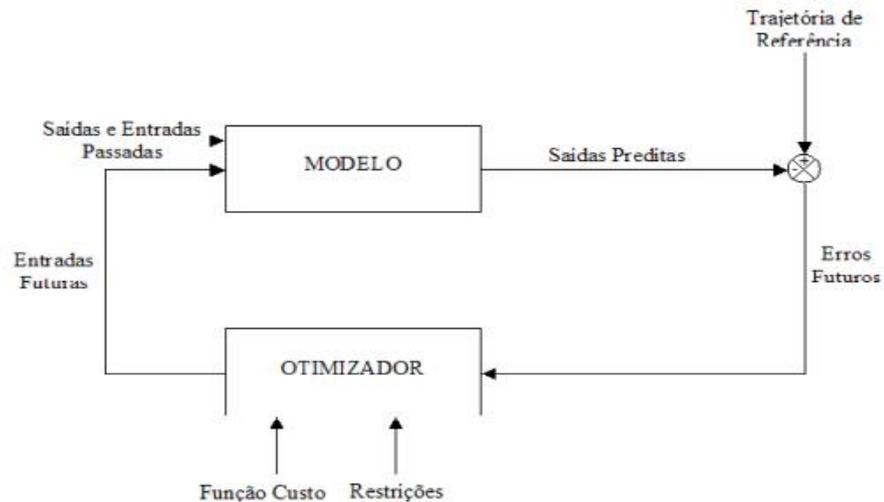
### 2.2.2 Controladores preditivos

O controlador preditivo baseado em modelo (MPC, *Model Predictive Control*) refere-se a uma classe de algoritmos de controle computacionais que utilizam explicitamente um modelo do processo para prever a resposta futura de uma planta (QIN; BADGWELL, 2003). Esses tipos de controladores foram desenvolvidos principalmente devido aos avanços de *hardware* nos computadores digitais, o que tem permitido que esta técnica possa ser implementada em tempo real (MUNIZ, 2004).

De modo geral, no MPC além de utilizar o modelo para prever a saída do processo para instantes de tempos futuros (horizonte de predição), calcula-se uma sequência de ações de

controle que minimizam uma determinada função objetivo. Adota-se então uma estratégia retrocedente, sendo que, a cada passo de tempo, o horizonte é deslocado para o futuro (MUNIZ, 2004). A Figura 2 apresenta um modelo esquemático para o controlador preditivo.

Figura 2 – Modelo esquemático do controlador preditivo



Fonte: Muniz (2004)

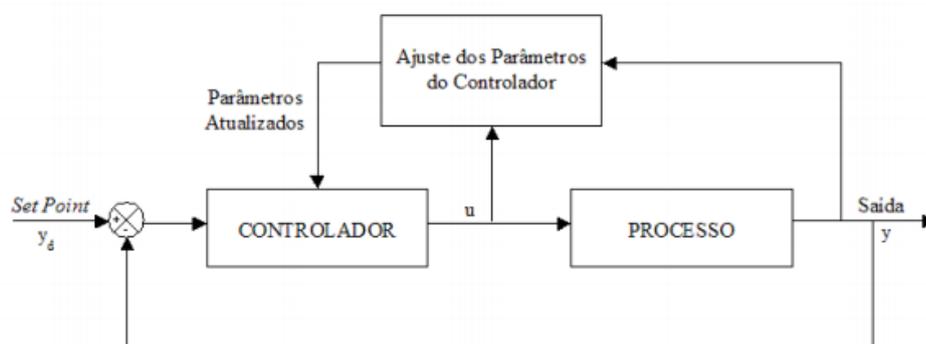
Os vários algoritmos MPC diferem entre si apenas no modelo usado para representar o processo e a também na função objetivo a ser minimizada. Este tipo de controlador é de natureza tão variada que muitos trabalhos têm sido desenvolvidos, tendo ampla aceitação no mundo acadêmico e na indústria (DUTRA, 2003). Entretanto, o grande esforço é em conseguir obter um modelo que represente apropriadamente a dinâmica do processo. Dessa forma, o desempenho deste controlador está diretamente ligado com a representatividade do modelo em relação ao processo real (ASSIS, 2001).

Uma abordagem para o desenvolvimento de controladores preditivos foi proposta, no início da década de 70, utilizando o modelo linear de resposta ao degrau, o qual foi denominado de algoritmo DMC (*Dynamic Matrix Control*). Esta técnica relaciona mudanças na saída do processo a um somatório ponderado de valores passados das mudanças na entrada (CUTLER; RAMAKER, 1979). Usando o modelo da resposta a degrau pode-se escrever as variações preditas para a saída como uma combinação linear das variações das entradas futuras. A matriz que as conectam é chamada de Matriz Dinâmica. Isto resulta em menor esforço computacional, aliado a ações de controle menos agressivas, obtendo-se maior robustez a erros na modelagem (QIN; BADGWELL, 2003).

### 2.2.3 Controles adaptativos

Os controladores adaptativos são sistemas de controle que ajustam de forma automática e online seus parâmetros, de modo a compensar mudanças no processo ou ambiente, fornecendo abordagens sistemáticas e flexíveis para lidar com incertezas, não linearidades e parâmetros de processo variáveis com o tempo. Desse modo, esses algoritmos podem ser utilizados para o controle de processos complexos, onde os mesmos são pouco entendidos ou mudam de forma não previsível (MUNIZ, 2004). A Figura 3 apresenta um modelo esquemático de um controlador adaptativo.

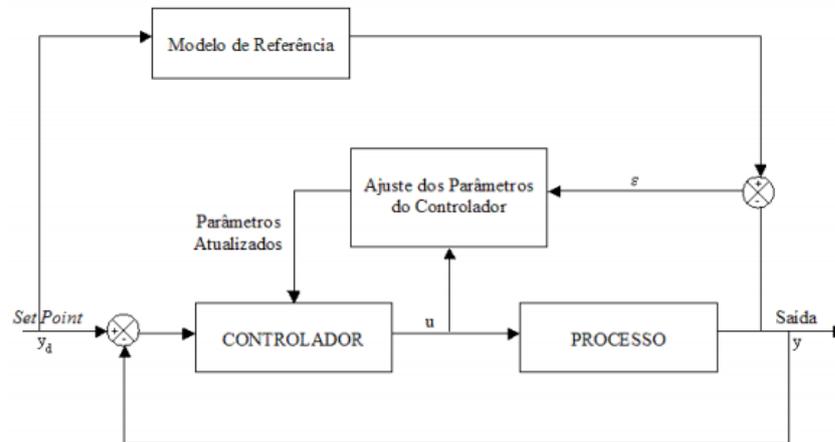
Figura 3 – Modelo esquemático de controladores adaptativos



Fonte: Muniz (2004)

Tendo em vista essa abordagem, existem ainda controladores adaptativos que utilizam um modelo de referência, que usa a entrada e a saída do processo para estimar, recursivamente e de forma online, os parâmetros do modelo do processo, bem como recalculam os parâmetros do controlador. O controlador pode ser então um PID ou preditivo (MPC). A Figura 4 apresenta a estrutura do controlador adaptativo combinado com um preditivo.

Figura 4 – Modelo esquemático de controle preditivos adaptativos



Fonte: Muniz (2004)

No contexto de controle adaptativo, aplicado de forma on-line, percebe-se uma semelhança entre o que se entende por rede neural, uma vez que conceitualmente uma rede neural se ajusta ou "se adapta" a certo conjunto de dados que lhe são apresentados por meio do ajuste de seus pesos, ajuste esse feito principalmente off-line (ASSIS, 2001). Nesta perspectiva, as redes neurais se apresentam como instrumentos com potencial para se realizar a adaptação, conforme destacado por Boskoiv e Narendra (1995), que utilizou redes neurais, associadas a outras técnicas de controle, como subsistemas de identificação adaptativas de sistemas dinâmicos. Dessa forma, a técnica de adaptação tem sido empregada nas estratégias de controle que utilizam redes neurais (BALLINI; ZUBEN, 2000).

#### 2.2.4 Inteligência artificial: redes neurais e lógica *fuzzy*

Em aplicações da engenharia de processos, Baughman e Liu (1995) já destacavam vantagens em desempenho em se utilizar uma classe de controladores que unem redes neurais e lógica nebulosa (*fuzzy*). Também autores como Braga, Carvalho e Ludemir (2007) reforçam as aplicações desses sistemas avançados de controle em processos biológicos.

Entre algumas aplicações, pode-se citar o controle de um processo fermentativo em batelada, com produção de levedura como objetivo e tendo o etanol como produto indesejável devido à inibição. Assim, o objetivo é conduzir o processo fermentativo segundo uma trajetória determinada de crescimento celular, pelo ajuste da temperatura de fermentação (BAUGHMAN, D. R; LIU, Y. A; 1995). Neste caso, a rede neural foi utilizada para prever a concentração de

células nas condições de cultivo e operação, e o controlador *fuzzy* para determinar as modificações na temperatura a fim de alcançar a concentração de células desejada em um tempo futuro.

Uma segunda aplicação relevante encontrada na literatura foi o controle da concentração de glicose em um processo operado em batelada alimentada utilizando *Saccharomyces cerevisiae*. O controle da concentração de glicose é fundamental, pois altas concentrações desta levam a elevadas concentrações de etanol, as quais vez inibem o metabolismo celular (ASSIS, 2001). Por fim, as redes neurais e a lógica *fuzzy* também foram utilizadas no controle da concentração de glicose em um processo de cultivo de alta densidade celular em batelada alimentada de *E. coli* recombinante, por meio da medida indireta de pH. (ASSIS, 2001).

Dessa forma, percebe-se a relevância no emprego de técnicas baseadas em aprendizado de máquina, tais quais as redes neurais, na automação, otimização e monitoramento de processos. Esse fato é ainda mais relevante considerando os bioprocessos, cujos produtos podem ser de alto valor agregado, tais como medicamentos, células animais, insumos para a indústria alimentícia e entre outros (BRAGA; CARVALHO; LUDERMIR, 2007).

### **2.3 Redes neurais artificiais (RNA)**

Uma definição de Rede Neural Artificial (RNA) é apresentada por Haykin (2001), que afirma que a RNA é um processador composto por unidades de processamento simples, que têm capacidade de armazenar conhecimento experimental por meio de um algoritmo de aprendizado, na qual os pesos são ajustados para posterior utilização.

Ainda segundo Haykin (2001), o uso de Redes Neurais oferece vantagens, entre elas:

- Aprendizado: a rede é treinada a partir de exemplos.
- Não linearidade: um neurônio artificial pode ser linear ou não-linear, de modo que sua aplicação é adequada para sistemas físicos não lineares.
- Capacidade de adaptação: apresenta a propriedade de se adaptar à nova informação.
- Capacidade de generalização: uma rede neural possui uma capacidade de generalização do conhecimento adquirido.

Em suma, a rede aprende a partir de um conjunto reduzido de padrões e produz respostas para os dados que não estavam presentes na fase de treinamento. A auto-organização, o mapeamento de entrada-saída, a adaptabilidade e o processamento em tempo real são outras

características presentes em uma rede neural, que fazem dela uma ferramenta hábil na resolução de problemas complexos (HAYKIN, 2001; BRAGA; CARVALHO; LUDERMIR, 2007).

### 2.3.1 Fundamentos teóricos

O termo Rede Neural Artificial (RNA) faz referência ao modelo biológico de um neurônio, dada a semelhança na sua capacidade de aprendizagem.

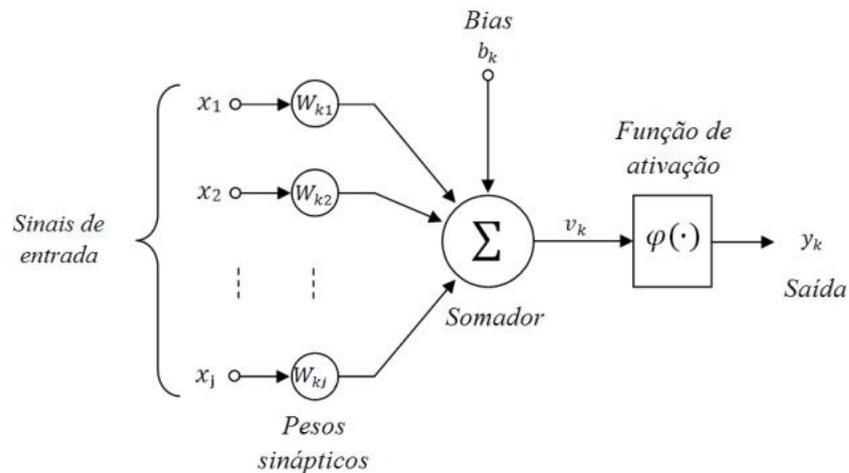
O primeiro modelo matemático de neurônio artificial foi proposto e desenvolvido pelos pesquisadores McCulloch e Pitts (1943). Em 1957 foi criado por Frank Rosenblatt o *Perceptron*, um modelo com saída binária 0 e 1, base atual das RNA, dando origem a vários outros modelos. O modelo matemático está demonstrado pela Equação 9.

$$y(t) = \begin{cases} 1 & \text{se } \sum_{i=1}^n w_i x_i - \theta_i \geq 0 \\ 0 & \text{se } \sum_{i=1}^n w_i x_i - \theta_i < 0 \end{cases} \quad (9)$$

onde  $y(t)$  é a saída do neurônio, na qual  $w_i$  representa o peso da conexão sináptica associado à entrada  $x_i$  e  $\theta_i$  é o limiar de ativação do neurônio.

Um modelo esquemático de um neurônio artificial está apresentado na Figura 5.

Figura 5 – Modelo de um neurônio artificial



Legenda:  $x_1 \dots x_j$  representam sinais de entrada. Toda entrada  $j$  vinculada ao neurônio  $k$  é ponderada por um peso sináptico  $w_{kj}$ . O combinador linear realiza o somatório dos pesos combinado com a entrada, em conjunto com  $b_k$  (*bias*).  $\varphi$  representa a função de ativação responsável por associar o sinal resultante  $v_k$  a  $y_k$ , que é o valor de saída do neurônio.

Fonte: Haykin (2001)

Matematicamente, um neurônio é definido pelas Equações 10 e 11.

$$v_k = \sum_{j=1}^j (w_{kj} \cdot x_j + b_k) \quad (10)$$

$$y_k = \varphi (v_k) \quad (11)$$

onde  $x_j$  são sinais das  $j$  entradas,  $w_{kj}$  são pesos sinápticos da entrada  $j$  e neurônio  $k$ ,  $b_k$  representa o *bias*,  $v_k$  a saída do combinado linear,  $\varphi$  a função de ativação e  $y_k$  a saída do neurônio.

O *bias* é empregado para aumentar ou diminuir a entrada líquida da função de ativação, fornecendo maior liberdade na ponderação dos dados de entrada, além de uma maior capacidade de aproximação da rede. Além disso, esse parâmetro também permite que um neurônio apresente uma saída não nula, caso todos os dados de entrada sejam nulos (SILVA, 2015).

Por fim, a função de ativação  $\varphi$  atua nas saídas das camadas ocultas de uma RNA, introduzindo as não linearidades ao sistema, o que virá representar uma das características de maior destaque (SCHNITMAN, 1998).

### 2.3.2 Funções de ativação

A maioria das funções não lineares poderiam ser aplicadas como funções de ativação. Porém, uma das regras mais comuns para o aprendizado, baseado na retropropagação do erro (backpropagation), é que as funções devem ser diferenciáveis (HAYKIN, 2001). A Figura 6 apresenta algumas das funções de ativação mais utilizadas.

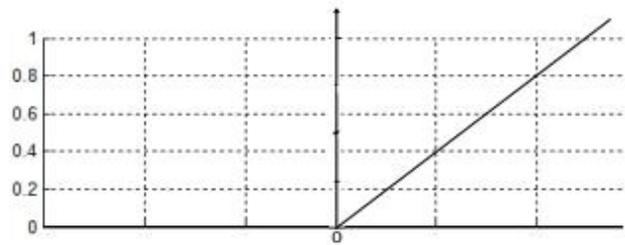
Figura 6 – Funções de ativação. a) e b) função linear retificada (ReLU) e sua forma gráfica. c) e d) função sigmóide (logística) e sua forma gráfica. e) e f) função tangente hiperbólica e sua forma gráfica.

a)

$$\varphi(v) = \text{ReLU}(v) = \max\{0, v\}$$

$$\varphi'(v) = \begin{cases} 1, & \text{se } v > 0 \\ \text{indefinido}, & \text{se } v = 0 \\ 0, & \text{se } v < 0 \end{cases}$$

b)

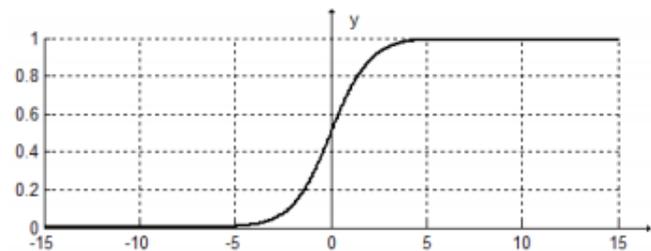


c)

$$\varphi(v) = \sigma(v) = \frac{1}{1 + e^{-v}}$$

$$\varphi'(v) = \sigma(v)(1 - \sigma(v))$$

d)

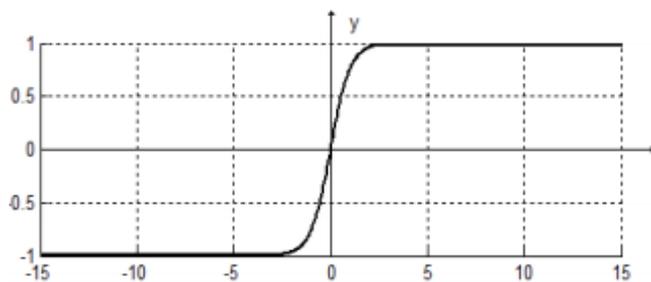


e)

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

$$\varphi'(v) = 1 - \tanh^2(v)$$

f)



Fonte: Do autor (2021)

Embora a derivada da função ReLU não esteja definida em 0, na prática implementa-se sendo 0 ou 1. Por outro lado, as funções sigmóide e tangente hiperbólica são conhecidas por apresentarem suas derivadas de primeira ordem definidas em todos os pontos de seu domínio (SILVA, 2010).

Para os neurônios de saída, deve-se escolher uma função de ativação definida de acordo com a característica dos valores de saída do sistema.

### 2.3.3 Métodos de aprendizagem e treinamento de RNA

Em uma RNA, entende-se como aprendizagem o ajuste dos parâmetros da rede, por meio da qual são ajustados os pesos sinápticos a partir de dados de treinamento. Os principais tipos de aprendizagem são aprendizado supervisionado e aprendizado não supervisionado (FERRARI,2011).

O aprendizado supervisionado consiste na utilização de padrões de entrada e de saída já conhecidos, ou seja, são fornecidos pares de entrada e de saída à rede, de modo que a resposta de saída seja comparada com a resposta desejada. Assim, tem-se o erro, o qual é fruto da diferença entre a resposta desejada e a resposta real (BRAGA; CARVALHO; LUDERMIR, 2007). Sobre esse aspecto, na literatura utiliza-se comumente o critério do erro quadrático médio (MSE). Caso a saída da rede não seja igual ou próxima ao valor desejado, os pesos sinápticos e níveis de *bias* precisam ser ajustados iterativamente, sob a influência do sinal de erro, de forma a minimizar a diferença entre a saída gerada pela rede e a saída desejada. (SILVA, 2015). Na Equação 12 abaixo, tem-se que "e" é a diferença entre a resposta desejada e a resposta real, sendo *n* o número de exemplos e  $\delta$  é o valor final do erro total obtido.

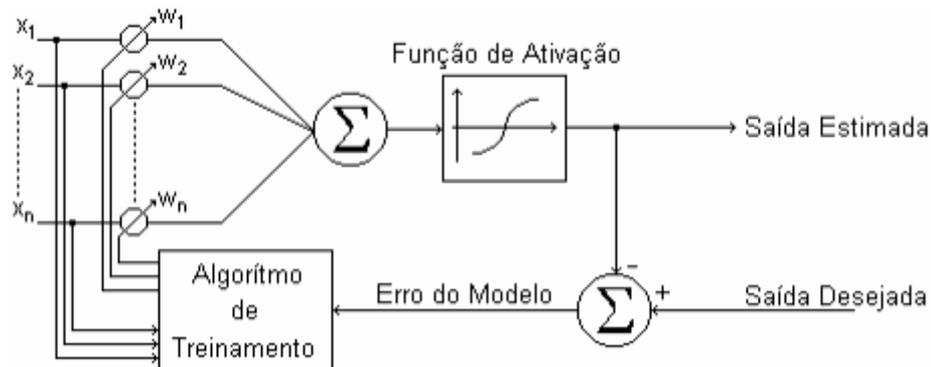
$$\delta = MSE = \frac{1}{n} \sum_1^n e^2 \quad (12)$$

Por sua vez, no aprendizado não supervisionado, a rede estabelece relações de padrões sem o auxílio de dados de saída. Durante esse aprendizado, verifica-se a existência de regularidades nos dados de entrada por meio de suas características estatisticamente mais relevantes (BRAGA; CARVALHO; LUDERMIR, 2007).

O término do método de aprendizagem se dá, basicamente, quando a RNA encontra uma solução para todas as entradas fornecidas. Entretanto, outros parâmetros também podem ser levados em consideração, como, por exemplo, a definição de uma taxa de erro a ser atingida ou o número máximo de ciclos (SILVA, 2015).

Logo, o treinamento de uma RNA é um exercício de otimização numérica e, normalmente, de uma função objetivo não linear. Assim, como não existe um método ideal para a otimização não linear, o mesmo é selecionado com base nas características do problema a ser resolvido (SCHNITMAN, 1998). A Figura 7 apresenta de forma generalizada um dos procedimentos mais comuns para o treinamento dos pesos sinápticos da RNA.

Figura 7 – Diagrama blocos para treinamento redes neurais



Fonte: Schnitman (1998)

No diagrama apresentado na Figura 7, pode-se dividir o treinamento em duas etapas. A primeira etapa, denominada de *forward*, ocorre com a avaliação das entradas da rede, que são propagadas adiante até obter uma saída estimada, após a função de ativação da última camada. Posteriormente esse valor é comparado com o valor da saída desejada a fim de minimizar o erro da resposta atual em relação à saída desejada.

Na segunda etapa, conhecida como *backward*, o erro deverá ser propagado em direção à camada de entrada, cujo principal objetivo é a atualização dos parâmetros livres (pesos sinápticos), sendo o algoritmo de treinamento mais utilizado o de retropropagação do erro (*backpropagation*), proposto por Chauvim e Rumelhart (1995) (SILVA, 2015).

#### 2.3.4 Aprendizado supervisionado por retropropagação de erros

Conforme mencionado na seção 2.3.3, o algoritmo de retropropagação de erros é composto por duas fases, *forward* e *backward*. Na fase *forward*, calcula-se a saída estimada da primeira até a última camada, logo não há alteração dos pesos sinápticos e *bias*. Deste modo, as Equações 10 e 11, apresentadas anteriormente, são utilizadas nessa fase. As Equações 13 e 14 são utilizadas de modo a simplificar a notação dos índices e para um neurônio. Na primeira iteração, deve-se inicializar os pesos utilizando algum critério, desde a escolha de um valor fixo pequeno até mesmo a utilização de algoritmos como o de Nguyen-Widrow (NGUYEN; WIDROW, 1990), que distribui de maneira aleatória uniformemente dentro de um limite determinado.

$$v_k = \sum_{j=1}^j (w_{kj} \cdot x_j + b_k) \rightarrow v = w \cdot x + b \quad (13)$$

$$y_k = \varphi(v_k) \rightarrow y = \varphi(v) \quad (14)$$

Obtendo a saída estimada, pode-se iniciar a fase de *backward*, a partir do cálculo do erro ( $\delta$ ), utilizando, por exemplo, o MSE apresentado na Equação 15, sendo  $y$  a saída estimada e  $\bar{y}$  a saída desejada.

$$\delta = MSE = \sum_1^n \frac{e^2}{n} = \sum_1^n \frac{(y - \bar{y})^2}{n} \quad (15)$$

Em seguida inicia-se a fase de atualização dos pesos, para cada neurônio e cada camada, apresentado nas Equações 16 e 17 (SCHNITMAN, 1998).

$$w_{k+1} = w_k - \alpha \cdot \frac{\partial \delta}{\partial w} = w_k - \alpha \cdot \Delta w \quad (16)$$

$$b_{k+1} = b_k - \alpha \cdot \frac{\partial \delta}{\partial b} = b_k - \alpha \cdot \Delta b \quad (17)$$

sendo  $\alpha$  a taxa de aprendizagem do algoritmo utilizado,  $\Delta w$  e  $\Delta b$  a derivada parcial da função erro por  $w$  e  $b$  respectivamente. Vale lembrar que são derivadas parciais, uma vez que cada neurônio e camada tem seus respectivos pesos.

Utilizando a regra da cadeia para obter o  $\Delta w$ , obtém-se a Equação 18. Observa-se então que a correção dos pesos envolve a utilização da derivada das funções de ativação, no termo  $\frac{\partial y}{\partial v} = \frac{\partial \varphi}{\partial v}$ , sendo esse o motivo a necessidade de utilizar funções de ativação diferenciáveis.

$$\Delta w = \frac{\partial \delta}{\partial w} = \frac{\partial \delta}{\partial y} \cdot \frac{\partial y}{\partial v} \cdot \frac{\partial v}{\partial w} \quad (18)$$

Uma das características mais determinantes para o tempo de convergência do algoritmo *backpropagation* é a taxa de aprendizagem  $\alpha$ . Utilizando baixos valores,  $dw$  torna-se pequeno demais, tornando a convergência do algoritmo muito lenta. Por outro lado, para taxas maiores,

a rede tenderia a minimizar o erro mais rapidamente, entretanto não há garantia da sua convergência (SCHNITMAN, 1998).

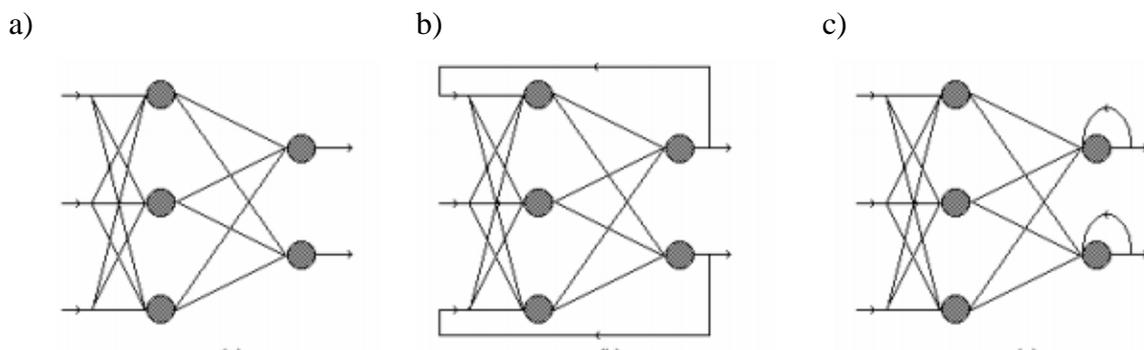
Enquanto o algoritmo padrão *backpropagation* utiliza o gradiente descendente como método de aproximação do mínimo da função erro, o algoritmo de Levenberg-Marquardt (PRESS, et al.,1990) utiliza aproximação pelo método de Newton. Esta técnica é mais rápida para treinamentos de rede de múltiplas camadas (SANTOS, 2008).

### 2.3.5 Arquiteturas

Usualmente, as RNA apresentam a característica de totalmente conectada (*fully-connected*), ou seja, todos os neurônios de determinada camada têm suas saídas conectadas a todos os neurônios da camada seguinte.

Na sua forma padrão, como pode ser observado na Figura 8 (a), define-se a RNA apresentada como uma rede *feedforward*, ou seja, o sinal é propagado entre camadas sempre no sentido progressivo. Outras configurações possíveis são as RNA realimentadas (*feedback*) que possuem camadas em que a saída de um neurônio é entrada para neurônios de camadas anteriores, Figura 8 (b), ou mesmo para o próprio neurônio, Figura 8 (c), as quais são conhecidas como RNA Recursivas (RNAR- *Recursive Artificial Neural Networks*). Existem também RNA com atraso (TDNN - *Time Delay Neural Network*), no qual o vetor de entrada é composto por valores passados da própria entrada e saída da rede.

Figura 8 – Exemplos de arquiteturas totalmente conectadas. a) Padrão *feedforward*. b) e c) RNA recursivas



Fonte: Schitman, (1998).

As arquiteturas de RNA podem admitir diversas camadas, além daquelas de entrada e saída. Essa classe é conhecida como *Perceptrons* multicamadas (*multilayer Perceptron - MLP*),

por admitirem a presença de uma ou mais camadas ocultas (HAYKIN, 2001). Acrescentando-se uma ou mais camadas ocultas, a rede torna-se apta a extrair informações complexas, inclusive de fenômenos subjacentes ao processo (HAYKIN, 2001).

Evidentemente, não faz parte da proposta deste trabalho a análise e compreensão de cada arquitetura de RNA apresentada, uma vez que, de uma forma geral, a aplicação de cada topologia busca soluções adequadas para determinados tipos de problemas. (BRAGA; CARVALHO; LUDERMIR, 2007). Desta forma, procura-se compreender e aplicar os conhecimentos básicos de RNA, dando-se ênfase à sua aplicabilidade em controle de processos dinâmicos não lineares.

### **2.3.6 Conjunto de treinamento, validação e teste**

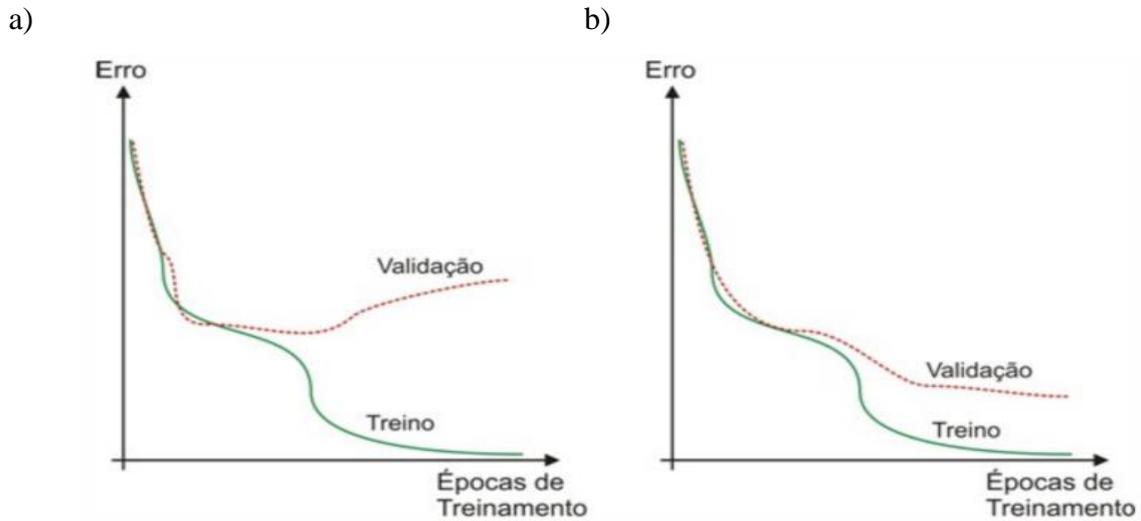
Para o treinamento de uma rede no modo supervisionado, admite-se conhecimento prévio de conjuntos de entrada/saída do sistema a ser modelado pela RNA.

A metodologia usualmente aplicada para aprendizagem das RNA é a divisão do conjunto de amostras em conjuntos de treinamento e validação. O treinamento corresponde à fase de ajuste dos pesos sinápticos, mediante apresentação dos seus pares de entrada/saída. Já no procedimento de validação, avalia-se o desempenho da saída real mediante um sinal de entrada, referenciando-se na saída esperada (BRAGA; CARVALHO; LUDERMIR, 2007). Admitindo-se que os pares de entrada/saída do conjunto de validação jamais foram apresentados à rede na fase de treinamento, a rede é válida se os erros obtidos nesta segunda fase forem inferiores a uma tolerância pré-definida. (BRAGA; CARVALHO; LUDERMIR, 2007). É importante observar que, na fase de validação, não há o ajuste dos pesos sinápticos.

Por outro lado, o conjunto de teste é utilizado para avaliar o desempenho e robustez do modelo com um terceiro conjunto de dados, uma vez que o objetivo é encontrar uma RNA que apresente o melhor desempenho com dados que não foram utilizados no treinamento (BISHOP, 1995).

Tipicamente, as curvas de treinamento da rede se comportam conforme apresentado na Figura 9. Na Figura 9 (a), é possível observar uma curva típica com sobre-aprendizagem (*overfitting*). Esse fenômeno ocorre devido a rede memorizar os padrões de treinamento ao invés de extrair padrões gerais, ocasionando uma especialidade apenas no conjunto de treinamento. Dessa forma, essa rede não obtém bons resultados para os dados de validação. Portanto, pode-se adotar algum critério de parada do treinamento, como, por exemplo, o ponto mínimo da curva de validação (HAYKIN, 2001).

Figura 9 – Curva de treinamento de rede. a) Curva com sobre-aprendizagem e b) treinamento normal desejado.



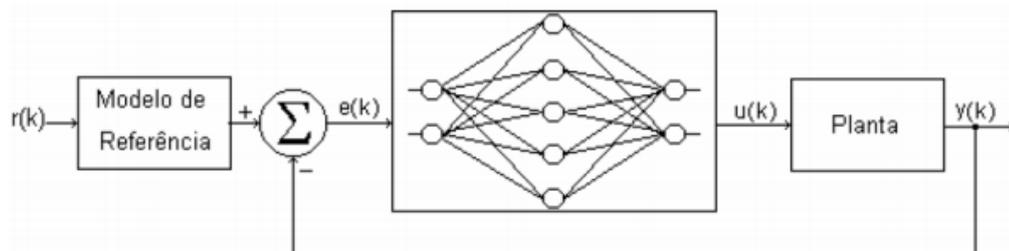
Fonte: Simas (2013)

Por outro lado, quando a complexidade do problema supera a complexidade da rede e esta não é capaz de descrever e representar o domínio do problema, denomina-se *underfitting*. Neste caso, recomenda-se aumentar o número de camadas ocultas (HAYKIN, 2001).

### 2.3.7 Aplicação RNA como lei de controle

Uma forma de atuação de controladores baseados em Rede Neural Artificial é no chamado controle direto, utilizando um modelo de referência. Nesta aplicação, o controlador é responsável por fazer com que a planta se comporte da forma desejada. Na Figura 10 é apresentado um modelo de controle direto baseado em um modelo de referência.

Figura 10 – Controle baseado em modelo de referência.



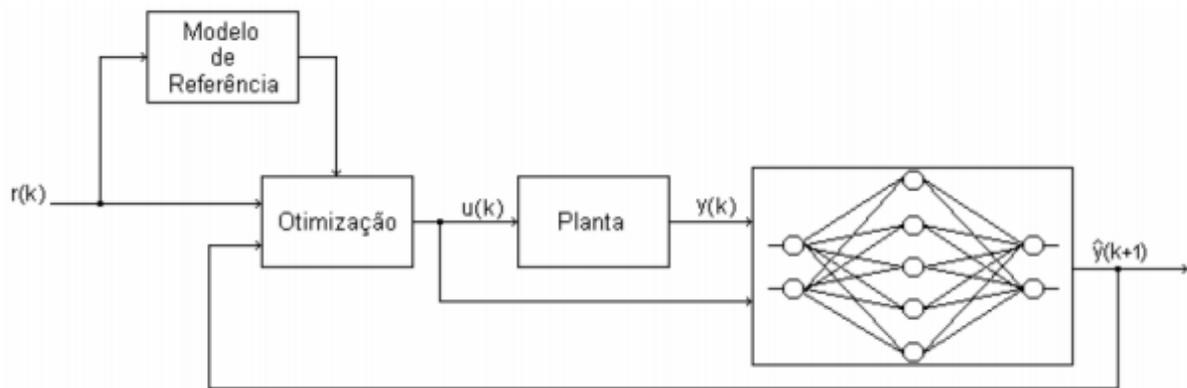
Legenda:  $r(k)$  é o sinal de referência,  $e(k)$  é o erro,  $u(k)$  é o sinal de entrada da planta e  $y(k)$  é o sinal de saída da planta:

Fonte: Schitman, (1998).

Neste contexto, a RNA é treinada com um sinal de controle que corresponda à ação de controle. Os dados podem ser obtidos, por exemplo, diretamente da ação humana, uma vez que utilizar um controle automático baseando-se em técnicas convencionais pode não garantir desempenho satisfatório para alguns processos (SCHITMAN, 1998).

As RNA também podem ser utilizadas com outras técnicas de controle avançando, desde a sua utilização em modelos híbridos com outros controladores (PID, DMC) até a implementação somente da RNA. Como exemplo, pode-se citar o controle preditivo baseado em RNA apresentado na Figura 11.

Figura 11 – Controle preditivo baseado em RNA



Legenda:  $r(k)$  é o sinal de referência,  $e(k)$  é o erro,  $u(k)$  é o sinal de entrada da planta e  $y(k)$  é o sinal de saída da planta:

Fonte: Schitman, (1998).

Na literatura existem diversas aplicações e diferentes configurações utilizando controladores baseados em redes neurais, tanto para a obtenção do modelo dinâmico quanto para a lei de controle.

Kittisupakorn et al. (2009) utilizaram a estrutura de um controle preditivo baseado em uma rede multicamadas *feedforward* para o processo de decapagem de aço, onde é necessário controlar a concentração de ácido clorídrico em um processo não-linear multivariável. Quanto ao treinamento, utilizou-se o algoritmo Levenberg–Marquardt. Nos testes realizados, a rede obtida NN MPC (*neural network model predictive controller*) demonstrou melhor desempenho no controle do sistema em relação ao controlador PI convencional em todos os casos estudados (KITTISUPAKORN et al., 2009).

Behroozi et al. (2013) utilizaram uma neural artificial (RNA) como modelo dinâmico preditivo e controle direto em um sistema de secagem de uva em secador de ar quente, um

sistema dinâmico não-linear. A partir dos dados de entrada, era previsto o conteúdo de umidade um passo à frente. O modelo foi desenvolvido utilizando a ferramenta de redes neurais (*Neural Network Toolbox*) do MATLAB. Foi utilizado o algoritmo de Levenberg–Marquardt durante o treinamento. A rede resultante continha uma camada oculta com 5 neurônios, sendo a função de ativação escolhida a função tangente hiperbólica. A camada de saída, consistindo em uma saída única, utilizava a função linear como ativação. Os resultados demonstraram que a RNA tinha melhor desempenho do que MLR (*multiple linear regression*) (BEHROOZI, et al., 2013).

Yuan Liu et al. (2020) obtiveram uma rede multicamadas *feedforward* para múltiplas entradas e saídas múltiplas (MIMO) em um sistema de controle de temperatura. Utilizou-se um modelo de referência para fornecer o sinal de entrada para o controlador neural. Para verificar o método proposto, simulações foram realizadas em MATLAB/SIMULINK e os resultados obtidos foram comparados quantitativamente com aqueles obtidos a partir dos sistemas de controle PID convencionais, obtendo não apenas uma melhora na resposta transitória do sistema, mas também melhor uniformidade da temperatura (LIU, et al., 2020).

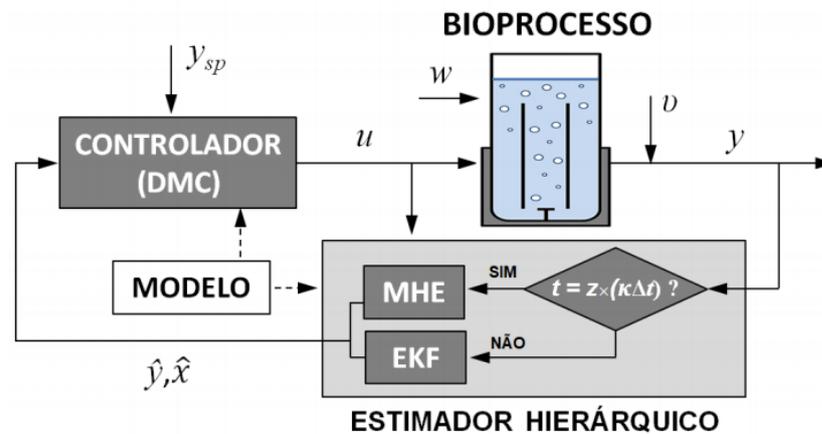
## 2.4 Controle de oxigênio dissolvido

Conforme mencionado, o oxigênio dissolvido apresenta dinâmica não linear em bioprocessos aeróbios. Logo, torna-se necessário a utilização de arquiteturas avançadas de controle, como os controladores preditivos, adaptativos, ou, mais recentemente, controladores baseados em inteligência artificial.

Campani et al. (2019) utilizaram uma estratégia hierárquica EKF/MHE-DMC para o monitoramento e controle do oxigênio dissolvido em um sistema de cultivo em batelada de *E. coli* em biorreator *airlift*. A implementação ocorreu de forma a possibilitar o monitoramento das variáveis de processo e, conseqüentemente, permitir o uso do controlador preditivo linear DMC. Neste sistema de controle, adotou-se a técnica *split-range*, com atuação sequencial nas variáveis vazão de ar ( $Q_{Ar}$ ), pressão no biorreator ( $P$ ) e vazão de oxigênio puro ( $Q_{O_2}$ ) (CAMPANI et al., 2016, 2019). A primeira variável, de acordo com a heurística determinada, é a manipulação da vazão de ar ( $Q_{Ar}$ ), sendo utilizada até atingir o valor máximo permitido de  $20 \text{ Lmin}^{-1}$ . Quando essa variável ultrapassa seu limite estipulado, a próxima variável manipulada é escolhida. No caso, foi feita a pressurização do sistema de 1 bar até 4 bar e, posteriormente, o enriquecimento do gás de entrada com oxigênio puro ( $Q_{O_2}$ ) (CAMPANI, 2018). Vale ressaltar que para aumentar  $Q_{O_2}$ ,  $Q_{Ar}$  deve diminuir, de forma a manter o fluxo de

gás ( $Q_{Ar} + Q_{O_2}$ ) no valor máximo de  $20 \text{ Lmin}^{-1}$ . Essa limitação é necessária devido à validade das equações utilizadas no modelo até essa vazão máxima. Além disso, altas vazões de gás levam à formação intensa de espuma, o que dificulta a transferência de oxigênio no meio (CAMPANI, 2018). Um modelo esquemático do sistema de controle EKF/MHE-DMC está apresentado na Figura 12.

Figura 12 – Diagrama esquemático do método hierárquico EKF/MHE-DMC



Legenda:  $y_{sp}$  representa o valor estabelecido para a variável controlada (oxigênio dissolvido),  $u$  representa as variáveis manipuladas ( $Q_{Ar}$ ,  $Q_{O_2}$  e  $P$ ),  $w$  a perturbação no processo,  $v$  o ruído na medida e  $y$  as variáveis de saída ( $C_x$ ,  $C_s$ ,  $C_p$  e  $OD$ ).  $\hat{y}$  e  $\hat{x}$  representam as estimativas.

Fonte: Campani, 2018

Hong-Gui Han e Jun-Fei Qiao (2011) utilizaram um controlador adaptativo baseado em uma rede neural de estrutura dinâmica (ACDSNN) para controlar a concentração de oxigênio dissolvido (OD) em um sistema de tratamento de águas residuais. O controlador ACDSNN proposto utiliza uma estrutura de rede neural *feedforward* variável e sua estrutura é determinada online automaticamente. Deste modo, o controle é adaptado para lidar com as mudanças nas características operacionais (HAN; QIAO, 2011).

Já Mete et al. (2010) desenvolveram uma RNA *feedforward* junto a um modelo preditivo (NN-MPC), aplicado à fermentação utilizando *Saccharomyces cerevisiae* em batelada em um biorreator. No sistema, a variável manipulada é a taxa de fluxo de ar e a rede apresentou resultados satisfatórios junto ao desafio de controle regulatório.

Azwar et al. (2006) compararam diferentes arquiteturas de redes neurais para o controle de OD no processo de lodo ativado através da manipulação pela manipulação da taxa de fluxo de ar. Os controladores foram submetidos a desafios envolvendo controle regulatório, rejeição

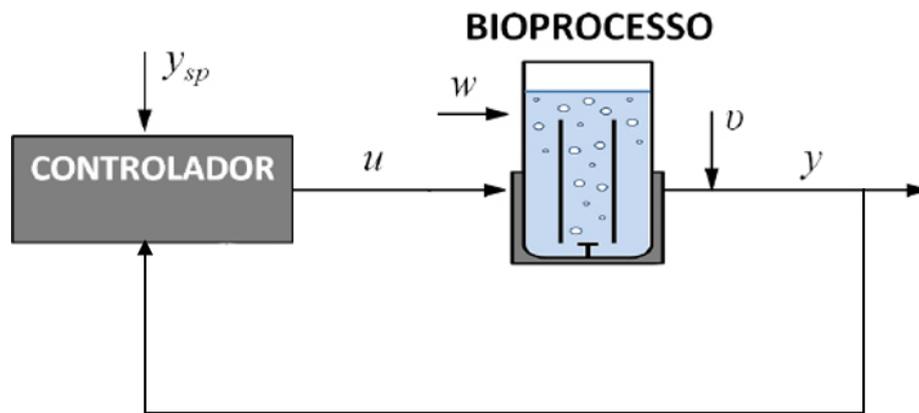
a perturbações e ruídos no sistema, sendo o erro absoluto integral (ISE) utilizado para essa comparação. A proposta com melhor desempenho foi obtida pelo modelo híbrido, empregando um controlador neural em paralelo com um controlador integral proporcional (PI) (AZWAR, et al., 2006). Todas as redes obtidas foram do tipo *forward*, com 14 entradas, 28 neurônios na única camada oculta e uma única saída. A função de ativação utilizada foi a sigmoide, tanto na camada oculta quanto na de saída (AZWAR, et al., 2006).

### 3. METODOLOGIA

#### 3.1 Simulação do processo

Para a obtenção de todos os dados e avaliação do desempenho dos controladores *in silico*, foi realizada a simulação do processo utilizando o modelo matemático desenvolvido por Campani et al. (2019), representado pelas Equações 1-6. O esquema do processo controlado é apresentado na Figura 13. Conforme mencionado, este modelo representa de forma coerente o sistema real nas suas duas fases: crescimento celular e de indução, quando ocorre a síntese a proteína recombinante (PspA). As simulações foram efetuadas através do programa MATLAB™ (2017a).

Figura 13 – Modelo esquemático do bioprocesso controlado



Legenda: No modelo esquemático,  $y_{sp}$  representa o valor estabelecido para a variável controlada (oxigênio dissolvido),  $u$  as variáveis manipuladas ( $Q_{Ar}$ ,  $Q_{O_2}$  e  $P$ ),  $w$  a perturbação no processo,  $v$  o ruído na medida e  $y$  as variáveis de saída ( $C_x$ ,  $C_s$ ,  $C_p$  e  $OD$ ).

Fonte: Do autor (2021)

#### 3.2 Aquisição do conjunto de dados

A partir das condições citadas na Tabela 1, e assumindo as premissas descritas a seguir, conforme proposto por Campani et al. (2019), foram realizadas simulações do processo a fim de se obter o conjunto de dados que foram utilizados no treinamento da rede. Foram obtidos dados operacionais para o nível de oxigênio dissolvido controlado em 20%, 30% e 40% durante toda operação, e adotando um tempo de simulação de 7 h para a fase de crescimento e de 5 h para a fase de indução (CAMPANI et al., 2019).

- a) Temperatura de processo constante à 32 °C;
- b) Intervalo de aquisição de 6 s;
- c) Para o controle preditivo linear DMC (*Dynamic Matrix Control*), adotou-se o horizonte do modelo, de predição e de controle de 30, 40 e 10, respectivamente;
- d) Conhecimento das condições iniciais do processo conforme Tabela 2.

Tabela 2 – Condições iniciais do processo

Valor Inicial	Fase Crescimento	Fase Indução
Concentração Celular ( $C_x$ )	0,7 $gL^{-1}$	9,0 $gL^{-1}$
Concentração Substrato ( $C_S$ )	60 $gL^{-1}$	60 $gL^{-1}$
Vazão de Ar ( $Q_{Ar}$ )	5 $L\ min^{-1}$	20 $L\ min^{-1}$
Pressão manométrica ( $P$ )	0	3.8 bar
Vazão de Oxigênio ( $Q_{O_2}$ )	0	0

Fonte: Campani et al. (2019)

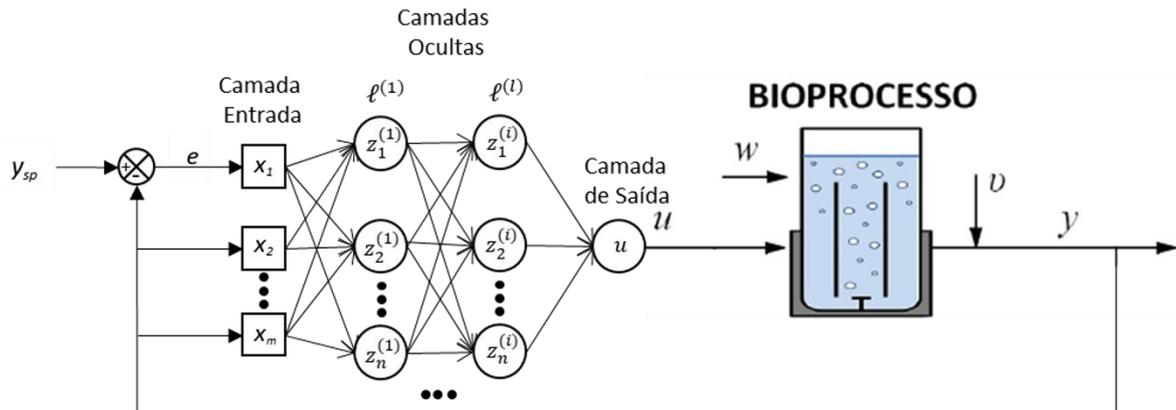
Para o treinamento supervisionado, dividiu-se o conjunto de dados (*dataset*) em 90% para o conjunto de treino e 10% para o conjunto de validação. Já para o conjunto de teste, utilizou-se um outro conjunto de dados, que será apresentado posteriormente.

Vale lembrar que esses dados foram obtidos de forma simulada, onde eventos relacionados a perturbações no processo e ruídos nas medidas, que podem ocorrer na prática da operação do processo, foram também considerados por meio da adição de incertezas na simulação.

### 3.3 Estrutura de controle

De acordo com a análise econômica feita por Campani (2018), a forma mais econômica de controlar o oxigênio dissolvido no processo é empregar um controlador *split-range*, com atuação sequencial e uma de cada vez nas variáveis vazão de ar ( $Q_{Ar}$ ), pressão no biorreator ( $P$ ) e vazão de oxigênio puro ( $Q_{O_2}$ ). Portanto, dada essa estrutura de controle, optou-se pela construção de três redes neurais, uma para cada variável manipulada, ou seja, *RNA*  $Q_{Ar}$ , *RNA*  $P$  e *RNA*  $Q_{O_2}$ , sendo três para cada fase do processo, isto é, para a fase de crescimento e para a fase de indução. Um modelo esquemático genérico desse sistema de controle é apresentado na Figura 14.

Figura 14 – Modelo esquemático controle neural



Legenda: Na representação da rede,  $x_m$  são as  $m$  variáveis de entrada da rede,  $z_n^\ell$  representa o neurônio da camada oculta  $\ell$  e  $n$  os números de neurônios daquela camada,  $y_{sp}$  o valor estabelecido para a variável controlada (oxigênio dissolvido),  $u$  as variáveis manipuladas ( $Q_{Ar}$ ,  $Q_{O_2}$  e  $P$ ),  $w$  a perturbação no processo,  $v$  o ruído na medida e  $y$  as variáveis de saída ( $C_x$ ,  $C_s$ ,  $C_p$  e  $OD$ ).

Fonte: Do Autor (2021)

### 3.4 Estrutura da rede

Todas as redes neurais deste trabalho foram desenvolvidas pela ferramenta de redes neurais *Neural Network Toolbox* presente no MATLAB<sup>TM</sup> (R2017a).

Adotou-se o tipo de aprendizado supervisionado, assim como uma arquitetura totalmente conectada (*fully connected*). Para isso, utilizou-se a função *newff* para criar uma rede do tipo *feedforward*. Para o treinamento da rede foi utilizado o método *backpropagation* e o algoritmo Levenberg-Marquardt. Já para a quantificação da função de otimização foi escolhida a média quadrática dos erros (*MSE*).

Sabe-se que devido aos diversos parâmetros que a construção da rede neural pode adotar, como, por exemplo, a escolha das funções de ativação, número de iterações, épocas, entre outros, teve como premissa adotar os seguintes pontos de partida:

- Número máximo de épocas: 2000;
- Número de iterações: 1500 a 3000;
- Função de ativação camada de saída: Linear (*purelin*);
- Função de ativação das camadas ocultas: Tangente hiperbólica (*tansig*) e Sigmóide (*logsig*).

Outros parâmetros, principalmente aqueles referentes à função de otimização, foi adotado o padrão (*default*) da função implementada no MATLAB, como, por exemplo, o erro desejado, valores máximos e mínimos do gradiente, taxa de aprendizado, taxa de incremento e decremento.

Por outro lado, o número de camadas ocultas assim como o número de neurônios por camada foram objetos de análise para a escolha da melhor arquitetura da rede, tendo como objetivo obter uma rede de menor complexidade que atendesse aos critérios de desempenho desejados. Entende-se por menor complexidade aquela rede com menor número de neurônios e número de camadas ocultas.

### 3.5 Variáveis de entrada e saída

Dada a complexidade da dinâmica do sistema e sendo o treinamento supervisionado escolhido, é necessário fornecer à rede dados de entrada que descrevam todo o processo. Dessa forma, utilizou-se as concentrações de células ( $C_x$ ), substrato ( $C_S$ ) e oxigênio dissolvido (OD), a fim de fornecer à rede dados relacionados à dinâmica do processo controlado.

Sendo o objetivo do sistema de controle manter o nível de oxigênio dissolvido em valores desejados, é importante para a rede o valor de referência ( $sp$ ) e também o erro ( $e$ ), referentes à variável controlada.

Por fim, sendo a estrutura de controle organizada de forma hierárquica, as variáveis predominantes por região de controle tornam-se fundamentais para a rede identificar qual a variável a ser manipulada no controle *split-range*. Logo, para a rede relacionada à pressão do sistema ( $RNA P$ ), incrementa-se uma variável de entrada relacionada à vazão de ar ( $Q_{Ar}$ ). Para a rede que controla a vazão de oxigênio puro ( $RNA Q_{O_2}$ ), incrementa-se a pressão manométrica.

Para a saída da rede, adotou-se o próprio valor absoluto da variável manipulada. Todas as variáveis por rede neural estão apresentadas na Tabela 3, sendo as mesmas variáveis de entrada nas fases de crescimento e indução.

Vale lembrar que todas as entradas das redes foram consideradas mensuráveis de forma online, utilizando sensores específicos e sensíveis a cada variável do processo.

Tabela 3 – Configuração de entrada e saída de cada RNA

Grupo de Variáveis	RNA $Q_{Ar}$	RNA $P$	RNA $Q_{O_2}$
Entradas relacionadas ao processo (g/L)	[ $C_x C_S OD$ ]	[ $C_x C_S OD$ ]	[ $C_x C_S OD$ ]
Entradas relacionadas à variável controlada	[ $e sp$ ]	[ $e sp$ ]	[ $e sp$ ]
Entradas adicionais por região	-	$Q_{Ar}$ (L/min)	$P$ (bar)
Saída	$Q_{Ar}$ (L/min)	$P$ (bar)	$Q_{O_2}$ (L/min)

Legenda:  $C_x$  concentração de células no meio,  $C_S$  concentração de substrato,  $e$  erro variável controlada em relação ao *setpoint*,  $OD$  nível de oxigênio dissolvido no meio,  $sp$  o *setpoint* da variável controlada. RNA  $Q_{Ar}$  refere-se à rede relacionada a vazão de ar, RNA  $P$  refere-se à rede relacionada com a pressão e, por fim, a RNA  $Q_{O_2}$  refere-se à rede relacionada com a vazão de oxigênio.

### 3.6 Critérios de decisão e desempenho

A partir dos dados de entrada e saída da rede determinados, divide-se o conjunto em dados de treino e de validação. O treinamento então corresponde à fase de ajuste dos pesos, mediante apresentação dos seus pares de entrada/saída obtidos. Já o procedimento de validação, mediante um sinal de entrada, apenas avalia o desempenho da saída real referenciando-se na saída esperada, uma vez que o conjunto de validação não foi utilizado para o ajuste dos pesos da rede, utilizados como critério de decisão na parada do treino. Dessa forma, a rede é válida se os erros obtidos nesta segunda fase forem inferiores ao critério de decisão de parada em cada época do treino. O erro é dado pela média absoluta dos erros ao quadrado (MSE), Equação 19, sendo o erro dado como a diferença da variável manipulada predita pela rede e presente nos dados de treino.

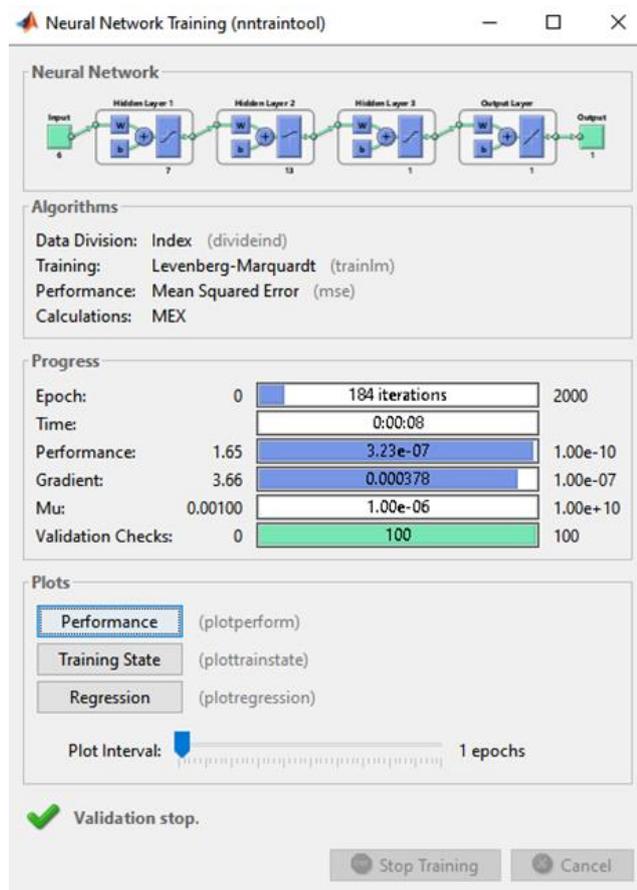
$$MSE = \frac{1}{n} \sum_{i=1}^n e^2 = \frac{1}{n} \sum_{i=1}^n (\hat{u}_i - u_i)^2 \quad (19)$$

sendo  $\hat{u}_i$  o valor da variável manipulada estimada,  $u_i$  o valor de referência da variável manipulada presente nos conjuntos treino. O índice  $i$  refere-se ao valor da variável naquele instante e  $n$  ao número total de dados de entrada fornecidos.

Um número máximo de épocas foi definido empiricamente a fim de evitar o sobreajuste (*overfitting*). O parâmetro “*Validation check*” corresponde ao número de épocas seguidas que o erro não aumentou a partir daquele ponto.

O treinamento das redes pode ser visualizado em tempo real a partir da ferramenta *nntraintool*, na qual todos os critérios apresentados na seção 3.4 podem ser observados, conforme visto na Figura 15.

Figura 15 – Ferramenta *nntraintool*



Legenda: No quadro *Progress* é possível observar os limites inferiores, superiores e valor atual de alguns parâmetros, como número de épocas (*Epoch*), valor do erro MSE (*performance*), valor do gradiente do proveniente do algoritmo Levenberg-Marquardt (*Gradient*), parâmetro de adaptação (*mu*) e o número de épocas seguintes em que o MSE foi menor (*Validation Checks*).

Fonte: Do autor (2021)

O critério de decisão MSE foi utilizado apenas para o treino das redes com diferentes arquiteturas (número de camadas e neurônios), correspondendo ao quanto cada rede consegue repetir o comportamento da variável manipulada.

Já para comparar o desempenho das redes para o controle do nível de oxigênio dissolvido, foi utilizado outro critério, a soma dos quadrados dos erros (SSE), sendo calculado como a soma dos erros entre o *setpoint* no instante  $t$  ( $y_{sp_t}$ ) e o valor obtido pela simulação para o OD ( $\hat{y}_t$ ) (Equação 20).

$$SSE = \sum_{t=0}^t (y_{sp_t} - \hat{y}_t)^2 \quad (20)$$

O coeficiente SSE é útil pois avalia diretamente o desempenho do controlador no processo simulado, por isso foi escolhido como critério de decisão sobre a melhor arquitetura da rede.

### 3.7 Fluxograma para definição da arquitetura da rede

Conforme mencionado anteriormente, a melhor arquitetura para a rede a ser obtida para cada variável manipulada é aquela que possui o menor valor de SSE, com menor número de camadas ocultas e menor número de neurônios por camada.

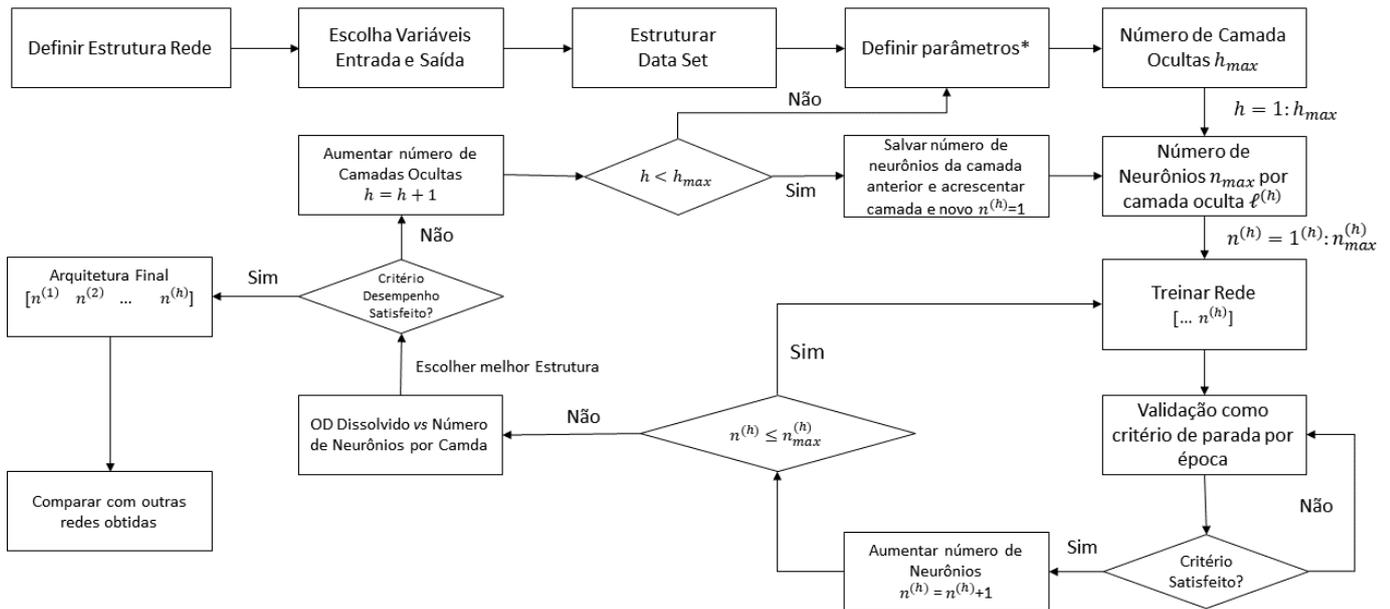
Para isso, variou-se o número de camadas e em cada camada, o número de neurônios, a fim de obter um gráfico entre a complexidade da rede vs. o erro de treino, validação e teste. O menor valor na curva do teste, dado pelo SSE, foi considerado como referente à melhor arquitetura da rede, ou seja, o número de neurônios da camada oculta em análise.

Por fim, após obter as redes, é necessário submetê-las aos desafios a fim obter uma análise quantitativa, por meio do erro integral no desempenho (ISE), Equação 21, e também qualitativa, dada pela observação de erros acentuados ou divergência no controle.

$$ISE = \sum_{t=0}^{t_f} (y_{sp_t} - \hat{y}_t)^2 \Delta t \quad (21)$$

A Figura 16 apresenta um fluxograma resumindo a metodologia apresentada na escolha da melhor arquitetura da rede.

Figura 16 – Fluxograma arquitetura de rede



Legenda: O índice  $h = 1: h_{max}$ , refere-se ao número da camada oculta, sendo  $h_{max}$  o valor máximo desejado para número de camadas ocultas. O índice  $n^{(h)} = 1: n_{max}^{(h)}$ , refere-se ao número de neurônios da camada específica  $h$ , sendo  $n_{max}^{(h)}$ , o maior valor desejado para número de neurônios daquela camada desejada.

Fonte: Do autor (2021)

### 3.8 Desafios de controle propostos

O desempenho do controlador RNA e dos controladores de referência PI e DMC foi avaliado em simulações do processo de cultivo sujeito aos seguintes desafios:

- Controle servo, através da variação em degrau, em diferentes momentos e valores de referência entre 20%, 30% e 40%, no *setpoint* da concentração de OD;
- Perturbações internas, simuladas por meio da adição de incerteza na constante de Monod ( $K_S$ ) do modelo, possuindo distribuição normal com média zero e covariância dada pela incerteza da determinação paramétrica, como demonstrado por Campani et al. (2019);
- Perturbações externas, simuladas como variações não medidas de -10% em  $Q_{Ar}$ ,  $P$  e  $Q_{O_2}$ , durante 60 s, em diferentes momentos. Essas perturbações refletem variações repentinas pressão na linha de entrada de gás, como exemplificado por Campani et al. (2019).
- Ruído nas medidas, assumindo variações aleatórias com distribuição normal, média zero e covariância determinada experimentalmente (Campani et al., 2019).

Além da observação qualitativa dos resultados obtidos para o controle do nível de OD e para a ação controle, determinou-se a integral do erro ao quadrado (ISE), sendo realizado em triplicada para cada mesmo cenário.

Para a avaliação dos controladores, foram adotados dois cenários. O “Cenário Modelo” é aquele de referência, com um perfil de *setpoint* para o qual o controlador PI foi sintonizado, bem como o teste das redes efetuado, conforme descrito na seção 3.6. Já o cenário “Cenário Alternativo”, com um perfil distinto de *setpoint*, é uma condição onde nenhum dos controladores haviam sido testados, o qual foi objeto de análise comparativa entre os sistemas de controle.

### 3.9 Controladores de referência

Conforme mencionado, o desempenho do controlador RNA foi comparado com aqueles obtidos a partir dos controladores de referência DMC e PI. Sendo o controlador DMC proposto por Campani et al. (2019) e o controlador PI sintonizado pelo método Ziegler-Nichols do ganho limite (STEPHANOPOULOS, 1984), ambos sujeitos ao Cenário Modelo. As relações de sintonia originais (PERRY et al., 1973) e adaptadas para o controlador PID estão presentes na Tabela 4.

Tabela 4 – Valores parâmetros controlador PID

<b>Ganho limite Ziegler-Nichols</b>	$K_c$	$\tau_i$	$\tau_d$
Original	0,45 $K_u$	$P_u/1,2$	-
Adaptado ao Cenário Modelo	0,2 $K_u$	$P_u/4$ ou $P_u/8$	-

Foi necessário adaptar as regras originais de sintonia devido à dinâmica não linear e rápida do oxigênio dissolvido no sistema, que contrapõe as premissas de sistemas lineares para o uso de controladores PID. Além disso, foi necessário realizar sintonizações em diferentes intervalos de tempo da simulação, uma vez que única sintonização por meio do método do ganho limite Ziegler-Nichols não foi suficiente para manter estável o controle de oxigênio durante todo o tempo de simulação.

Foi adotado o controlador tipo PI, pois este elimina o erro estacionário (*offset*) da variável controlada em relação ao *setpoint*, ao contrário do controlador P. Além disso, o PI é mais robusto do que o PID a ruídos nas medidas, justificando a sua escolha.

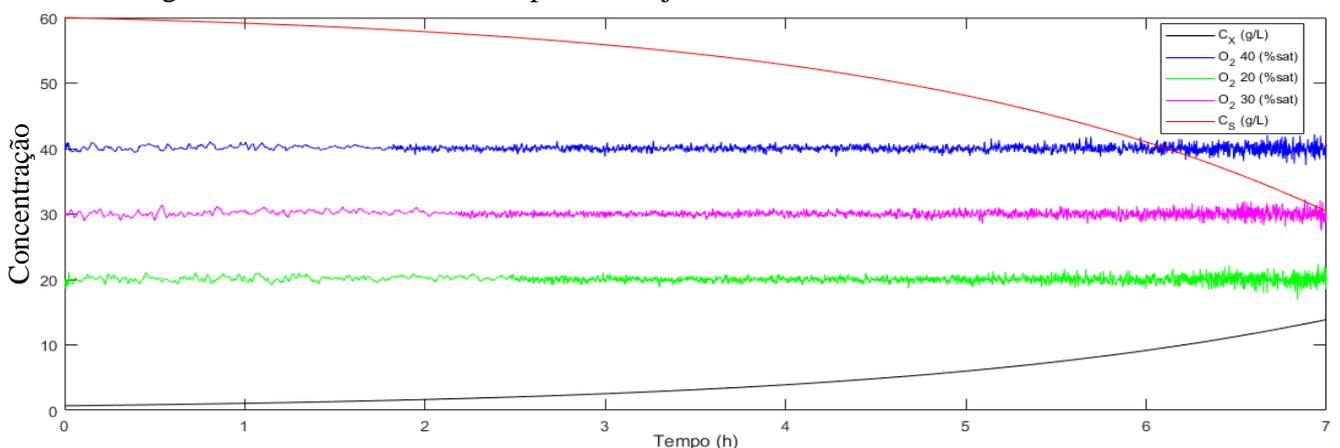
## 4. RESULTADOS E DISCUSSÃO

### 4.1 Conjunto de dados de treino e validação

Os conjuntos de dados para treino e validação foram obtidos conforme mencionado no item 3.2, para os valores de *setpoint* 20%, 30% e 40% de OD. Conforme mencionado, foi utilizado o controlador DMC para obter os dados operacionais para o treinamento supervisionado das redes neurais. O controlador DMC foi escolhido por apresentar o melhor desempenho e robustez (CAMPANI et al., 2019).

As Figuras 17 a 19 apresentam as variáveis de saída referentes à simulação da fase de crescimento do processo. Em razão dos valores de  $C_x$  e  $C_s$  não diferirem para as diferentes operações propostas, são apresentados os dados de operação ao *setpoint* de 30% e na Figura 18 apresentadas as variáveis manipuladas para a mesma operação.

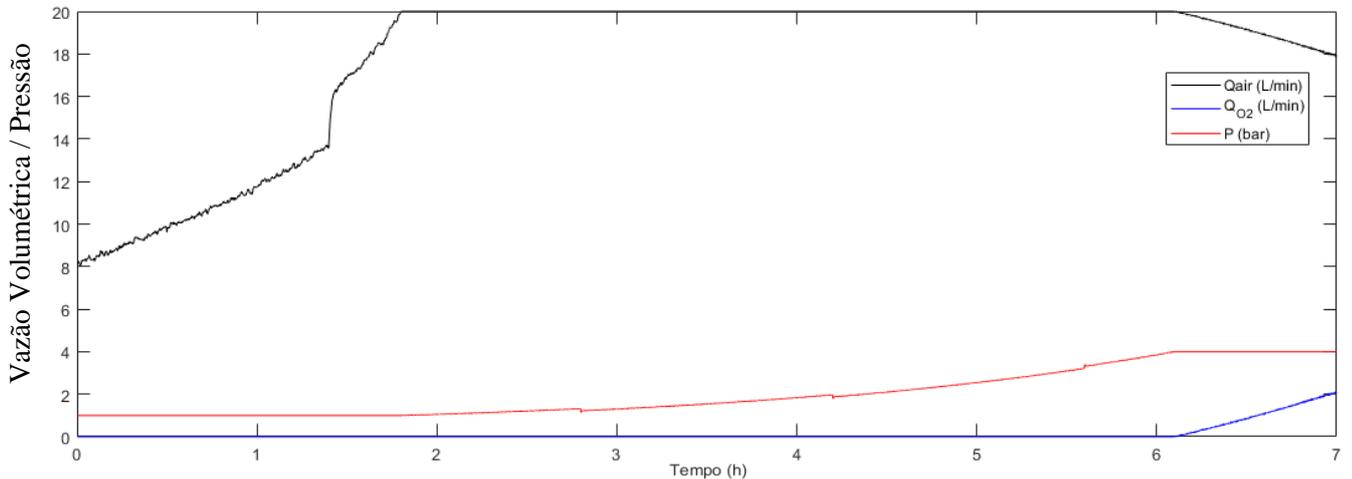
Figura 17 - Variáveis de saída para o conjunto de dados referentes à fase de crescimento.



Legenda: A curva vermelha refere-se ao perfil de concentração de substrato ( $C_s$ ) ao longo do tempo de simulação em horas, com concentração inicial igual a  $60 \text{ gL}^{-1}$ , conforme Tabela 2, e a curva preta, iniciando em  $0,7 \text{ gL}^{-1}$ , representando a concentração de células ( $C_x$ ). As linhas azul, rosa e verde, representam o valor de oxigênio dissolvido (OD) em relação à saturação (%sat) na temperatura de  $32^\circ\text{C}$ , sendo, respectivamente, com *setpoint* fixo de 40%, 20% e 30%.

Fonte: Do autor (2021)

Figura 18 - Variáveis manipuladas para o conjunto de dados referentes à fase de crescimento com *setpoint* para OD de 30%.

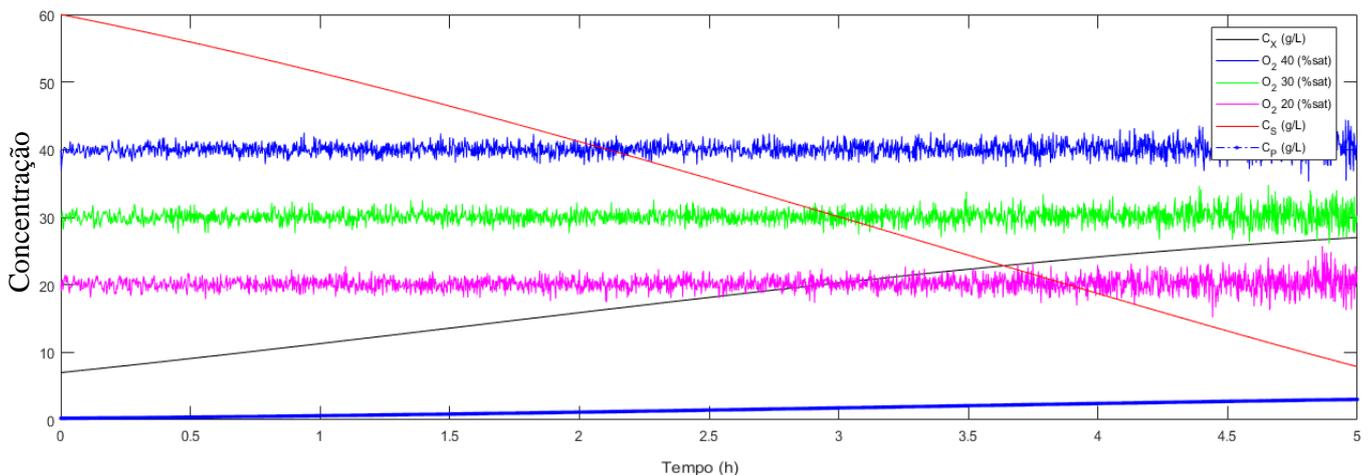


Legenda: A curva vermelha refere-se ao valor absoluto da pressão ( $P$ ) em bar, a curva preta à vazão de ar ( $Q_{Ar}$ ), em  $L \cdot min^{-1}$ , e a azul à vazão de oxigênio puro ( $Q_{O_2}$ ), em  $L \cdot min^{-1}$ , ao longo do tempo, em horas, de simulação.

Fonte: Do autor (2021)

De forma análoga, a Figura 19 – apresenta a saída referente à simulação na fase de indução. Para os valores de  $C_x$ ,  $C_s$  são apresentados os dados de operação ao *setpoint* de 40% e na Figura 20 são apresentadas as variáveis manipuladas para o mesmo *setpoint*.

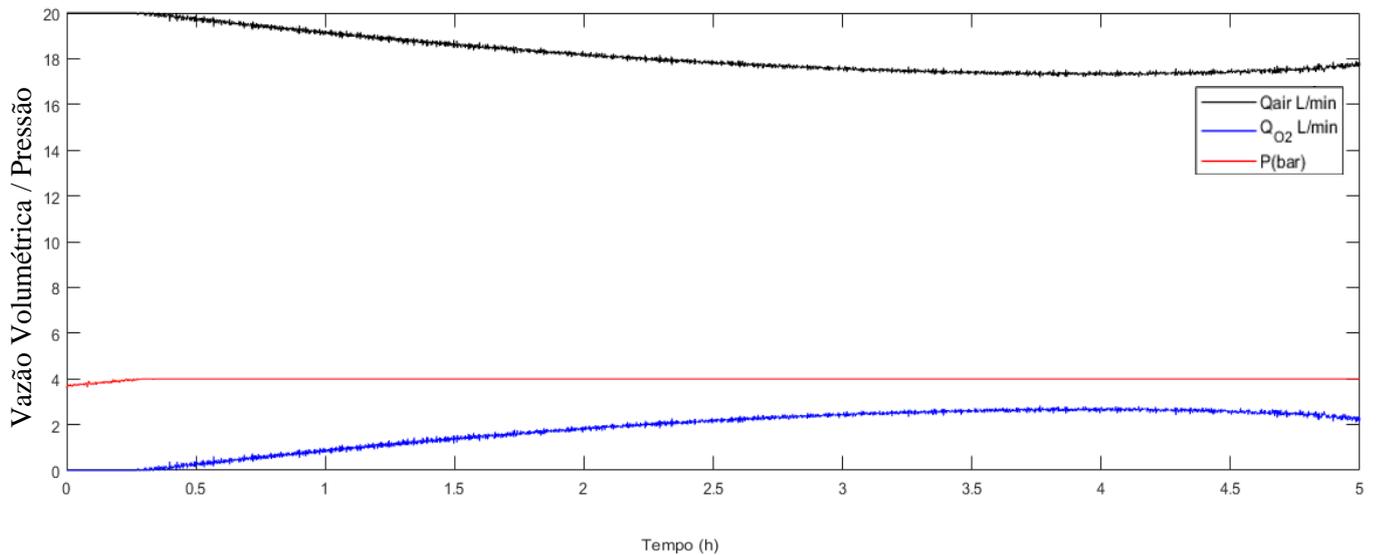
Figura 19 – Variáveis de saída para o conjunto de dados fase de indução



Legenda: A curva vermelha refere-se ao perfil de concentração de substrato ( $C_s$ ) ao longo do tempo de simulação em horas, com concentração inicial igual a  $60 \text{ gL}^{-1}$ , conforme tabela 2, e a curva preta, iniciando em  $9,0 \text{ gL}^{-1}$ , representando a concentração de células ( $C_x$ ); a curva inferior azul refere-se à concentração de proteína ( $C_p$ ), também em  $\text{gL}^{-1}$ . As linhas azul, rosa e verde representam o valor da concentração de OD, em % de saturação, na temperatura de  $32 \text{ }^\circ\text{C}$ , sendo, respectivamente, com *setpoint* fixo em 40%, 20% e 30%.

Fonte: Do autor (2021)

Figura 20 - Variáveis manipuladas para o conjunto de dados referentes à fase de indução com *setpoint* para OD de 40%.



Legenda: A curva vermelha refere-se ao valor absoluto da pressão ( $P$ ) em bar, a curva preta a vazão de ar ( $Q_{Ar}$ ), em  $L\text{min}^{-1}$ , e a azul a vazão de oxigênio puro ( $Q_{O_2}$ ), em  $L\text{min}^{-1}$ , ao longo do tempo, em horas, de simulação.

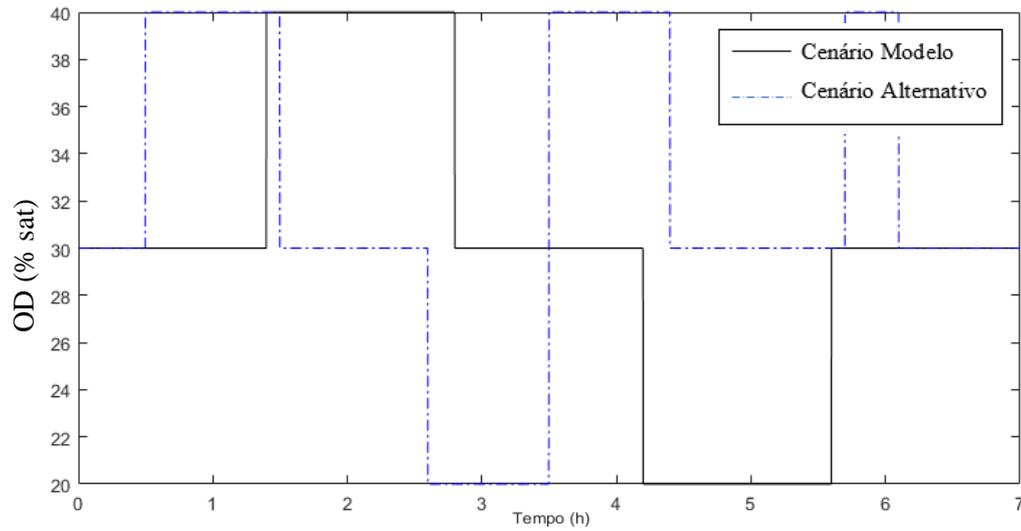
Fonte: Do autor (2021)

Embora os conjuntos de dados tenham sido obtidos através de simulações, percebe-se que estes se comportam de forma similar aos dados reais de processo (CAMPANI et al., 2016), sendo possível observar os ruídos normais presentes na prática.

## 4.2 Conjunto de dados de teste: Cenário modelo e cenário alternativo

A diferença entre os cenários modelo e alternativo está no perfil de *setpoint* imposto ao controle servo do OD. As Figura 21 e Figura 22 apresentam, respectivamente, os perfis de *setpoint* adotados para as fases de crescimento e indução.

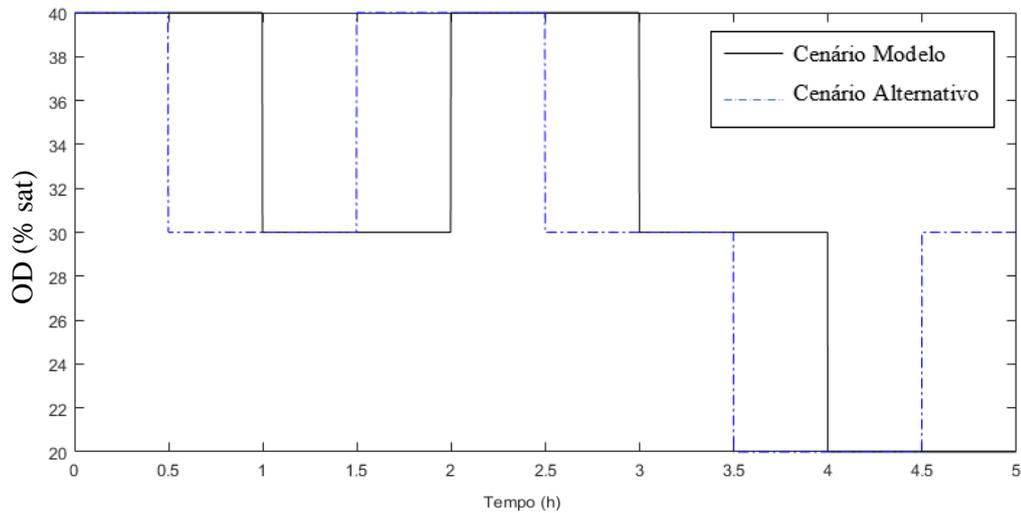
Figura 21 – Perfis de *setpoint* aplicados à fase de crescimento



Legenda: A curva preta refere-se ao valor do *setpoint* para OD, em %sat, no controle servo no cenário modelo e a curva azul ao controle servo alternativo no cenário alternativo.

Fonte: Do autor (2021)

Figura 22 – Perfis de *setpoint* aplicados à fase de indução



Legenda: A curva preta refere-se ao valor do *setpoint* para OD, em %sat, saturado no controle servo no Cenário Modelo e a curva azul ao controle servo no Cenário Alternativo.

Fonte: Do autor (2021)

### 4.3 Definição das arquiteturas das redes neurais

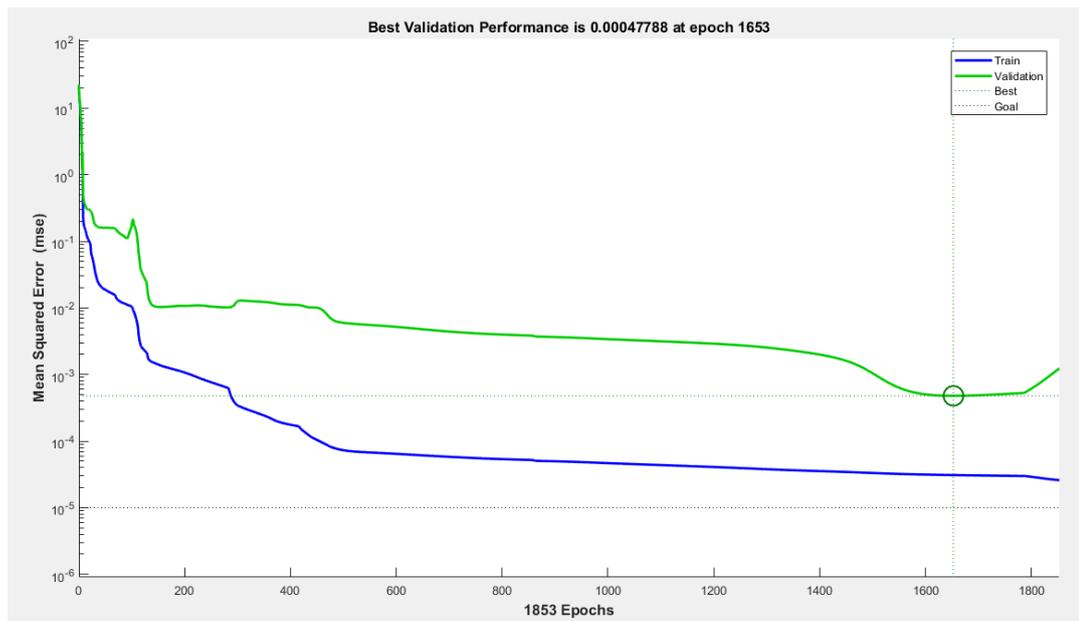
Cada rede foi obtida conforme descrito no fluxograma apresentado na Figura 16. De modo a exemplificar a obtenção de cada arquitetura das RNA, serão demonstrados os resultados da obtenção da rede referente ao controle de OD manipulando a vazão de ar (RNA  $Q_{Ar}$ ) na fase de crescimento. Para as demais redes o procedimento foi semelhante, bem como toda a discussão pertinente.

#### 4.3.1 Teste e validação

Adotando o número de “*Validation check*” igual a 200 épocas, pode-se observar na Figura 23 o comportamento da curva do erro quadrático médio (MSE) para o conjunto de dados de treino e de validação durante o treino da RNA  $Q_{Ar}$  para Fase de Crescimento. Essa rede apresenta a configuração [4,18], sendo 4 neurônios na primeira camada oculta e 18 na segunda camada.

Conforme mencionado no item 3.6, durante cada época do treino, o critério de parada é definido pelo ponto com menor valor de erro na curva, tendo como limite de épocas de treino o parâmetro “*Validation Check*”.

Figura 23 – Critério de validação durante o treino da RNA  $Q_{Ar}$  para a fase crescimento.



Legenda: A curva verde refere-se ao valor de MSE no conjunto de dados na validação. E a curva azul refere-se a MSE para o conjunto de dados no treino.

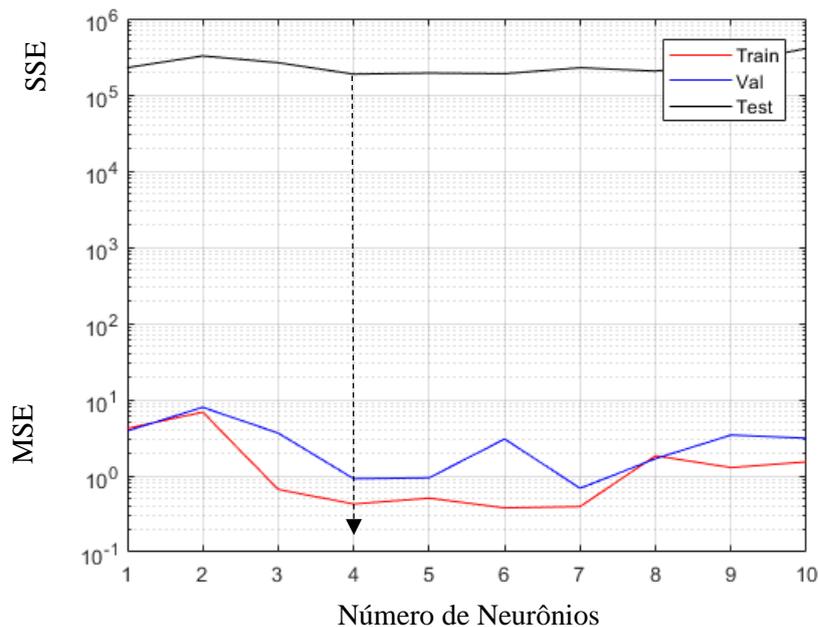
Fonte: Do autor (2021)

O comportamento observado para as curvas é o esperado. O MSE para o conjunto de treino tende a diminuir ao aumentar o número de épocas, uma vez que a rede tende a especializar apenas no conjunto de treino. Enquanto isso, o erro na previsão dos resultados no conjunto validação tende a diminuir até um certo ponto e depois aumentar. Portanto, não é necessário continuar o treino da rede e dessa forma é evitado o sobreajuste do modelo.

### 4.3.2 Complexidade da rede vs. desempenho

Com base no procedimento descrito no item 3.7, variando o número de neurônios com uma camada oculta para a rede RNA  $Q_{Ar}$  para a fase de crescimento, obteve-se o resultado apresentado na Figura 24.

Figura 24 – Rede RNA  $Q_{Ar}$  uma camada oculta para a fase de crescimento



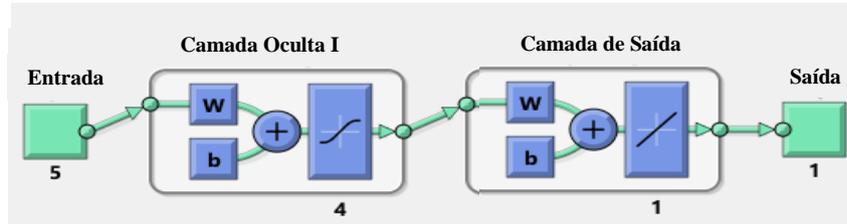
Legenda: A curva preta refere-se ao valor do somatório do erro quadrático (SSE) no conjunto de dados teste, entre o OD e o *setpoint* de referência. A curva azul refere-se ao erro médio quadrático (MSE) para o conjunto de dados de validação e a curva vermelha refere-se ao MSE para o conjunto de dados de treino.

Fonte: Do autor (2021)

Ao analisar a Figura 24, observa-se que com quatro neurônios na primeira camada é obtido o menor SSE, sendo esse então, o número de neurônios definidos para essa camada. A Figura 25 apresenta a arquitetura da primeira camada, sendo a função de ativação adotada

tangente hiperbólica. Os resultados para os erros na condição de mínimo estão apresentados na Tabela 5.

Figura 25 –Arquitetura da rede RNA  $Q_{Ar}$  com uma camada oculta para a fase de crescimento



Fonte: Do autor (2021)

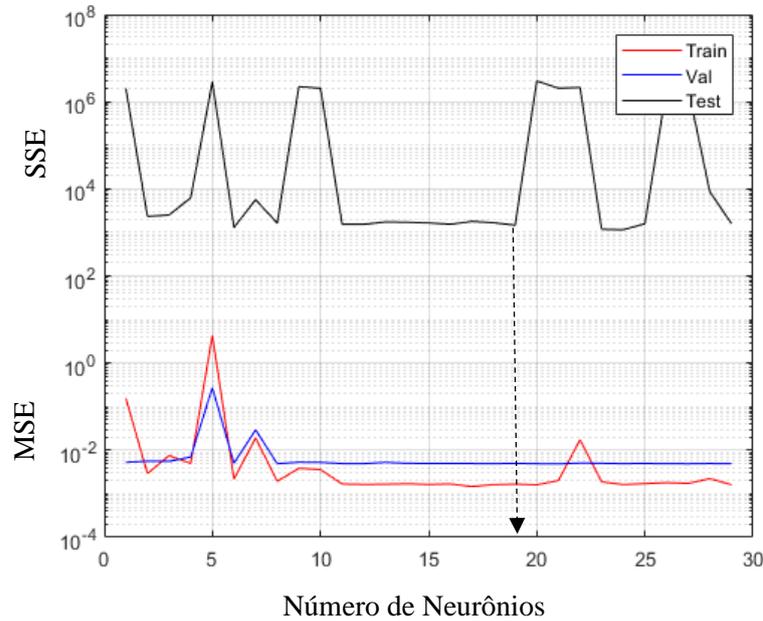
Tabela 5 – Resultados obtidos

	<b><i>RNA Q<sub>Ar</sub></i></b>
Entrada da rede	[ $C_x$ $C_S$ e $OD$ $sp$ ]
Treino (MSE)	0.43
Validação (MSE)	0.92
Teste (SSE)	186600
Saída da rede	$Q_{Ar}$ (L/min)

Legenda:  $C_x$  concentração de células no meio,  $C_S$  concentração de substrato,  $e$  erro da variável controlada em relação ao setpoint,  $OD$  nível de oxigênio dissolvido no meio,  $sp$  o *setpoint* da variável controlada.

Foi observado que com apenas uma camada oculta, a arquitetura ainda é de baixa complexidade para obter um desempenho satisfatório no controle do nível de OD. Portanto, adicionou-se uma segunda camada oculta, mantendo a primeira com 4 neurônios. O resultado para esse teste está apresentado na Figura 26, assim como a arquitetura final obtida na Figura 27 e os resultados para os erros na condição de mínimo estão apresentados na Tabela 6.

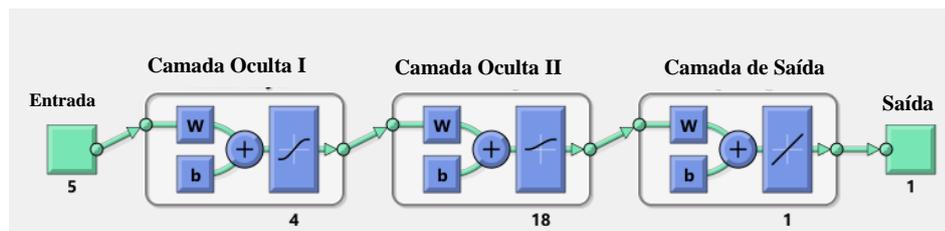
Figura 26 - Rede RNA  $Q_{Ar}$  com duas camadas ocultas para a fase de crescimento



Legenda: A curva preta refere-se ao valor do somatório do erro quadrático (SSE) no conjunto de dados teste, entre o oxigênio dissolvido e o *setpoint* de referência. A curva azul refere-se ao erro médio quadrático (MSE) para o conjunto de dados de validação e a curva vermelha refere-se ao MSE para o conjunto de dados de treino.

Fonte: Do autor (2021)

Figura 27 – Arquitetura final da rede RNA  $Q_{Ar}$  com duas camadas ocultas para a fase de crescimento



Fonte: Do autor (2021)

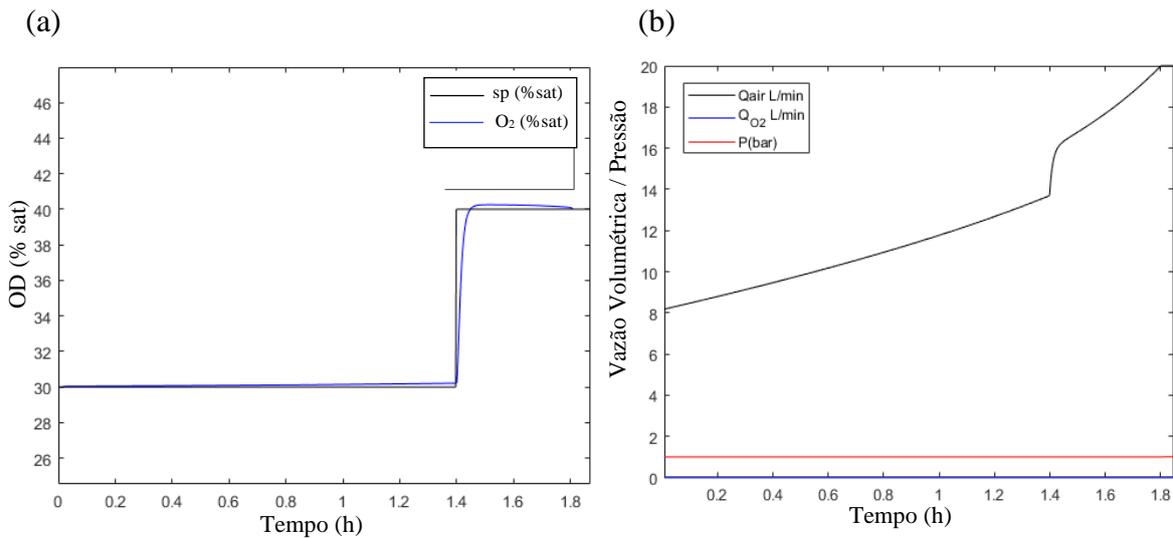
Tabela 6 – Resultados obtidos

	<b><math>RNA Q_{Ar}</math> [4]</b>	<b><math>RNA Q_{Ar}</math> [4 18]</b>
Entrada da rede	$[C_x C_S e OD sp]$	$[C_x C_S e OD sp]$
Treino (MSE)	0.43	0.001627
Validação (MSE)	0.92	0.004917
Teste (SSE)	186600	1669
Saída da rede	$Q_{Ar} (L/min)$	$Q_{Ar} (L/min)$

Legenda:  $C_x$  concentração de células no meio,  $C_S$  concentração de substrato, e erro variável controlada em relação ao *setpoint*,  $OD$  nível de oxigênio dissolvido no meio,  $sp$  o *setpoint* da variável controlada.

Em seguida, foi obtido o gráfico de desempenho do controlador através da simulação do sistema com arquitetura final para a rede RNA  $Q_{Ar}$  [4,18], obtida anteriormente. Segue nas Figura 28 (a) e (b) a dinâmica das variáveis de saída e manipuladas obtidas pela simulação utilizando a rede RNA  $Q_{Ar}$  na fase de crescimento submetidas ao cenário modelo.

Figura 28 – (a) Variáveis de saída e (b) variáveis manipuladas para o sistema controlado pela RNA  $Q_{Ar}$  durante a fase de crescimento.



Legenda – Na figura (a) está representado o valor do *setpoint*, a curva preta, e em azul refere-se ao valor do oxigênio dissolvido, em %sat. A figura (b) representa os resultados das variáveis manipuladas, sendo a curva vermelha referindo ao valor absoluto da pressão manométrica ( $P$ ), em bar, a curva preta a vazão de ar ( $Q_{Ar}$ ) em  $L \cdot min^{-1}$ , e a azul a vazão de oxigênio puro ( $Q_{O_2}$ ), em  $L \cdot min^{-1}$ , ao longo do tempo, em horas, de simulação.

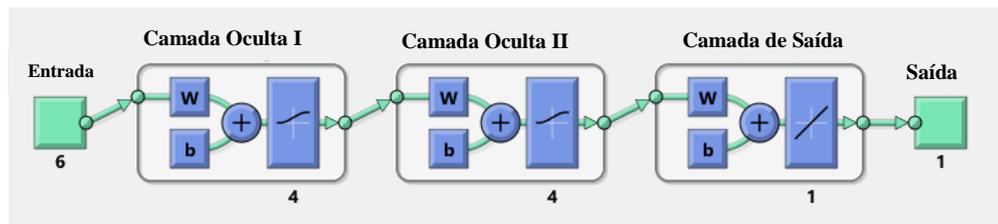
Fonte: Do autor (2021)

Tendo em vista a análise qualitativa no controle da variável controlada, conclui-se que a atual arquitetura para a rede RNA  $Q_{Ar}$  tem desempenho satisfatório perante o cenário modelo.

### 4.3.3 Arquiteturas finais

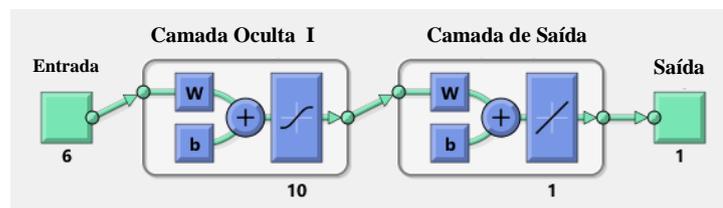
De modo análogo ao apresentado no item 4.3.2, foram obtidas as demais redes, RNA P e RNA  $Q_{O_2}$ , para o controle do OD na fase de crescimento (Figuras 29 e Figura 30, respectivamente).

Figura 29 – Arquitetura final da rede RNA P para a fase de crescimento



Fonte: Do autor (2021)

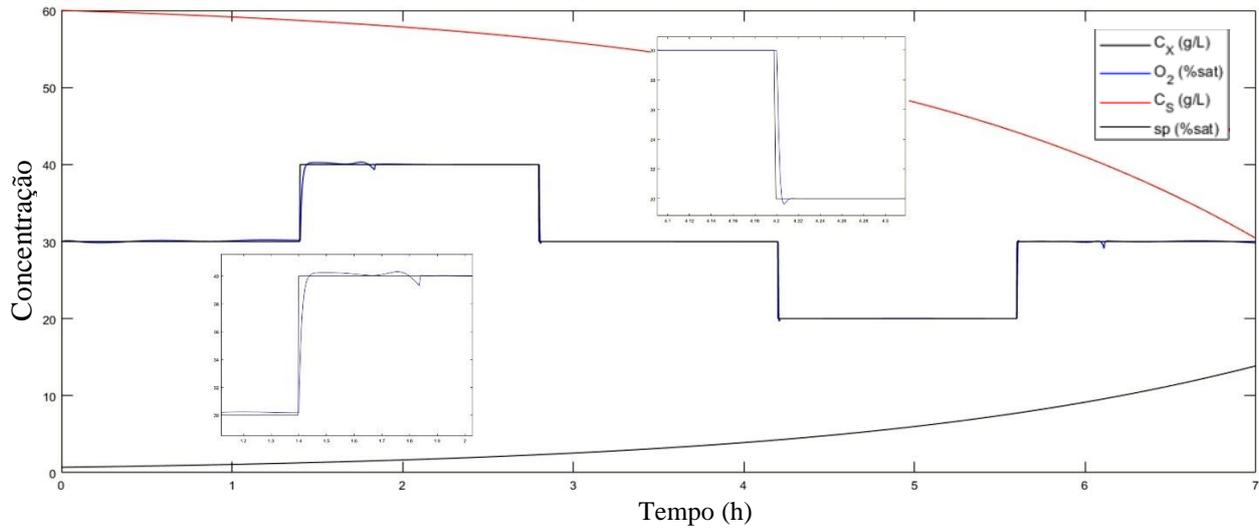
Figura 30 – Arquitetura final da rede RNA  $Q_{O_2}$  para a fase de crescimento



Fonte: Do autor (2021)

Por fim, as Figuras 31 e Figura 32 correspondem, respectivamente, aos valores das variáveis de saída e manipuladas obtidos pela simulação utilizando a estrutura de controle obtida anteriormente, composta pelas três RNA ( $RNA Q_{Ar}$ ,  $RNA P$  e  $RNA Q_{O_2}$ ) na fase de crescimento submetidas ao cenário modelo.

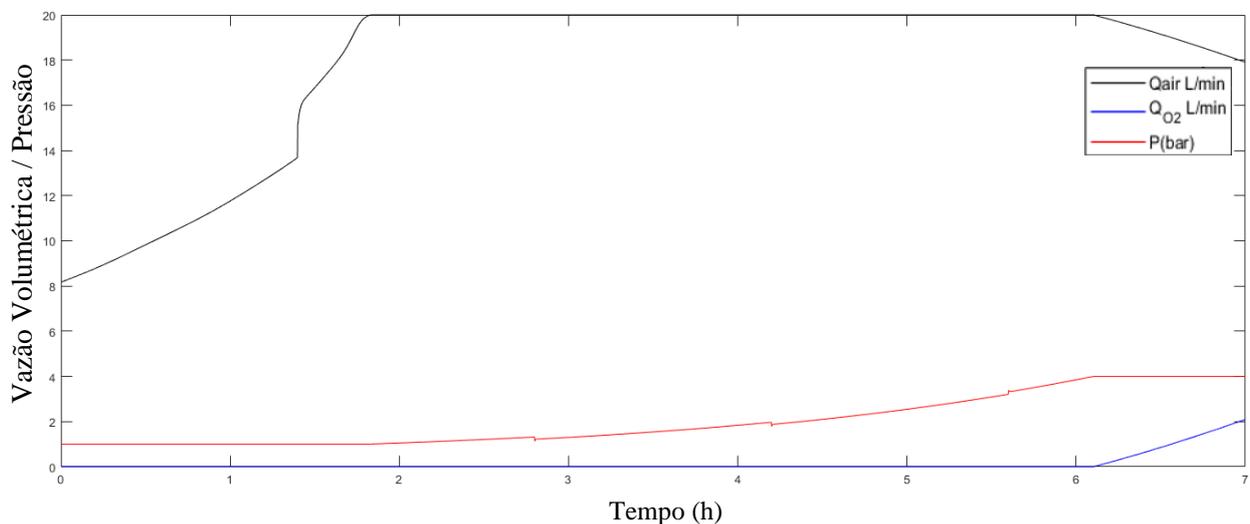
Figura 31 – Variáveis de saída para a simulação do processo sujeito ao cenário modelo para a fase de crescimento



Legenda: A curva vermelha refere-se ao perfil de concentração de substrato ( $C_S$ ), em  $\text{g.L}^{-1}$ , ao longo do tempo de simulação em horas, a curva preta representa a concentração de células ( $C_X$ ), em  $\text{g.L}^{-1}$ . A curva azul representa o valor da concentração de OD, em % de saturação, na temperatura de  $32\text{ }^\circ\text{C}$  e a curva preta refere-se ao *setpoint*, também em %sat.

Fonte: Do autor (2021)

Figura 32 - Variáveis manipuladas para a simulação do processo sujeito ao cenário modelo para a fase de crescimento



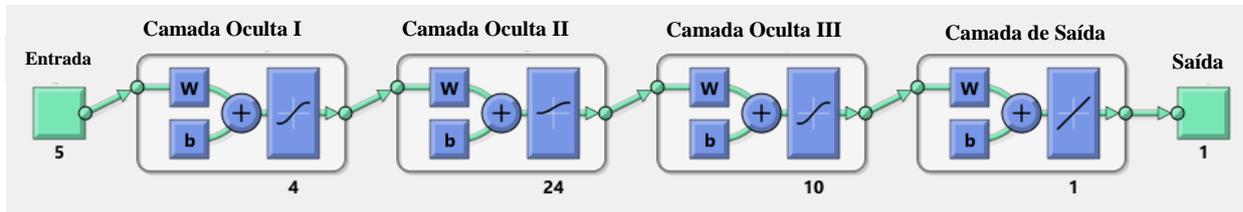
Legenda: A curva vermelha refere-se ao valor absoluto da pressão ( $P$ ) em bar, a curva preta a vazão de ar ( $Q_{Ar}$ ), em  $\text{L.min}^{-1}$ , e a azul a vazão de oxigênio puro ( $Q_{O_2}$ ), em  $\text{L.min}^{-1}$ , ao longo do tempo, em horas, de simulação.

Fonte: Do autor (2021)

Pelo perfil apresentado, pode-se concluir que a atual arquitetura na fase crescimento tem desempenho satisfatório, com os valores de OD acompanhando o *setpoint* durante todo o tempo de simulação, mesmo quando aplicados os degraus nos valores de referência.

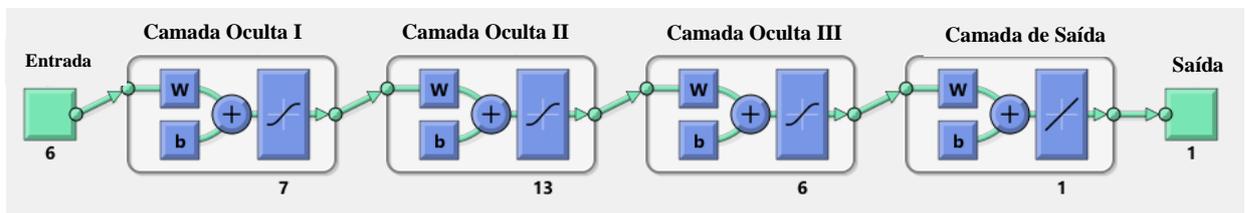
Para a fase de indução, as arquiteturas das redes obtidas RNA  $Q_{Ar}$ , RNA P e RNA  $Q_{O2}$  estão representadas nas Figuras 33, 34 e 35, respectivamente.

Figura 33 – Arquitetura final da rede RNA  $Q_{Ar}$  para a fase de indução



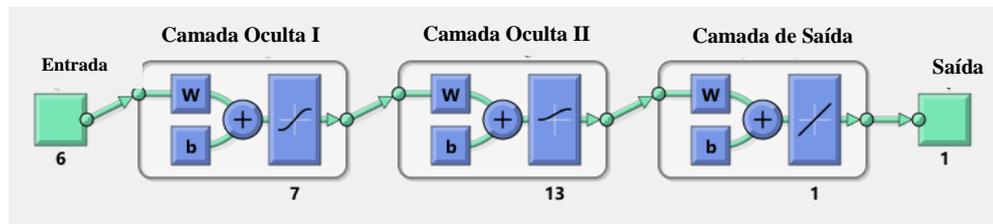
Fonte: Do autor (2021)

Figura 34 – Arquitetura da rede RNA P para a fase de indução



Fonte: Do autor (2021)

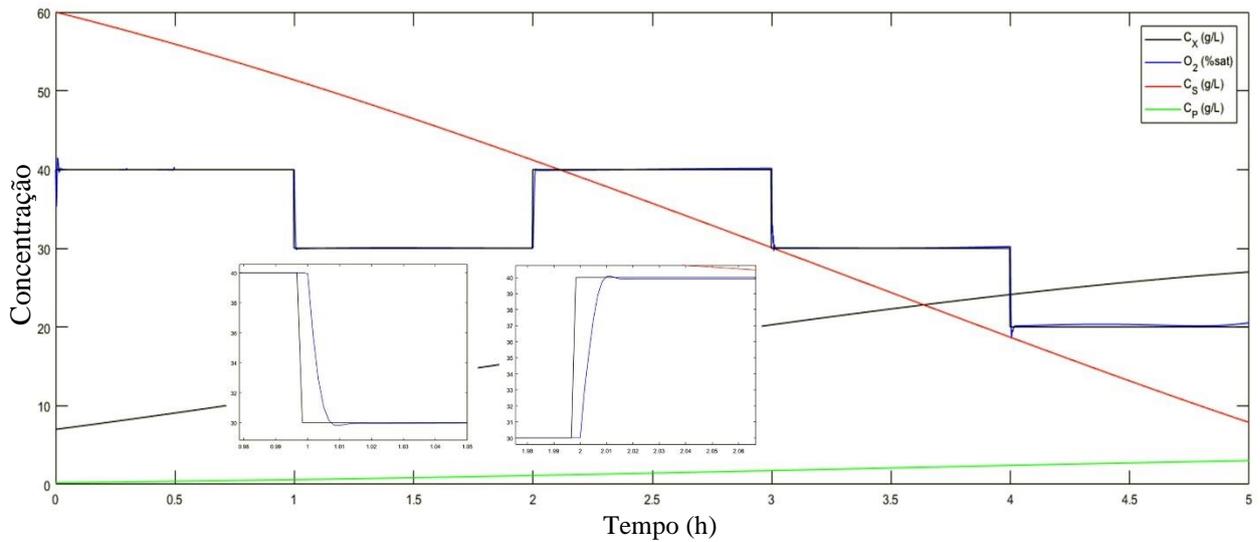
Figura 35 – Arquitetura final da rede RNA  $Q_{O2}$  para a fase de indução



Fonte: Do autor (2021)

Do mesmo modo, as Figuras 36 e 37 correspondem, respectivamente, aos valores das variáveis de saída e manipuladas obtidos pela simulação utilizando a estrutura de controle, conforme demonstrado anteriormente, composta pelas três RNA ( $RNA Q_{Ar}$ ,  $RNA P$  e  $RNA Q_{O2}$ ) na fase de indução submetidas ao cenário modelo. Conforme pode ser observado, a arquitetura apresentada tem desempenho satisfatório.

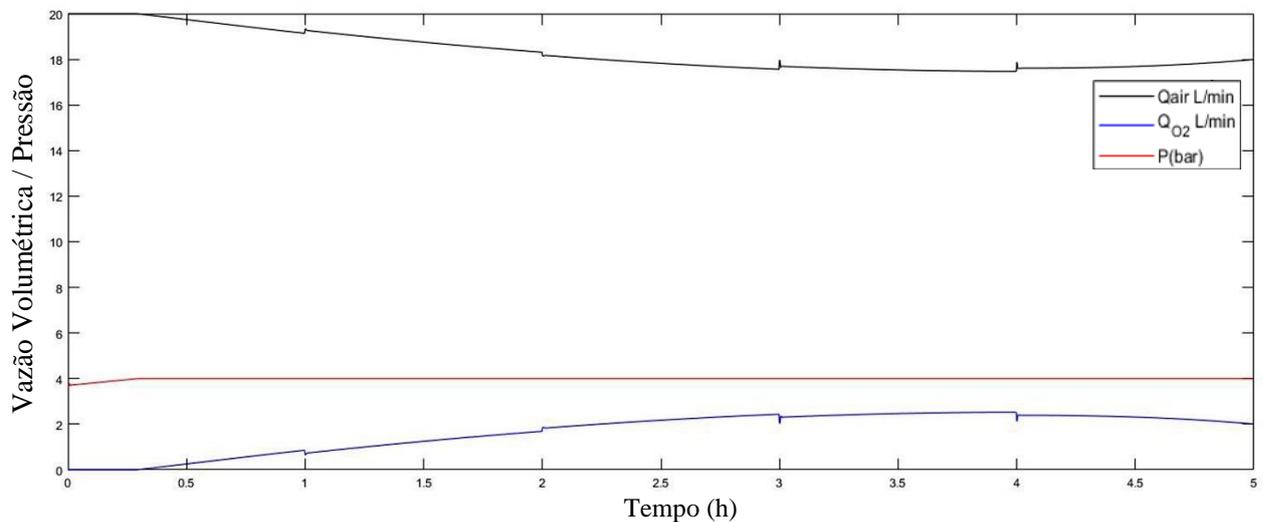
Figura 36 - Variáveis de saída para o controle servo modelo na fase de indução



Legenda: A curva vermelha refere-se ao perfil de concentração de substrato ( $C_S$ ), em  $\text{g.L}^{-1}$ , ao longo do tempo de simulação em horas, a curva preta representa a concentração de células ( $C_X$ ), em  $\text{g.L}^{-1}$ . A curva azul representa o valor da concentração de OD, em % de saturação, na temperatura de  $32^\circ\text{C}$  e a curva preta refere-se ao *setpoint*, também em % sat.

Fonte: Do autor (2021)

Figura 37 - Variáveis Manipuladas para a simulação do processo sujeito ao cenário modelo para a fase de indução



Legenda: A curva vermelha refere-se ao valor absoluto da pressão ( $P$ ) em bar, a curva preta a vazão de ar ( $Q_{Ar}$ ), em  $\text{L.min}^{-1}$ , e a azul a vazão de oxigênio puro ( $Q_{O_2}$ ), em  $\text{L.min}^{-1}$ , ao longo do tempo, em horas, de simulação.

Fonte: Do autor (2021)

#### 4.4 Desempenho dos controladores para a fase de crescimento

A abordagem proposta para o controlador baseado em redes neurais, assim como os de referência, PI e DMC, foram submetidos aos desafios propostos e serão discutidos os resultados obtidos nas seções seguintes. Vale lembrar, conforme apresentado no item 3.8, que os desafios são: (a) controle servo, (b) perturbações internas, (c) perturbações externas e (d) ruídos nas medidas. Todos esses desafios foram aplicados aos cenários modelo e alternativo. Portanto, a diferença entre ambos é no perfil de *setpoint* adotado, conforme pode ser observado no item 4.2.

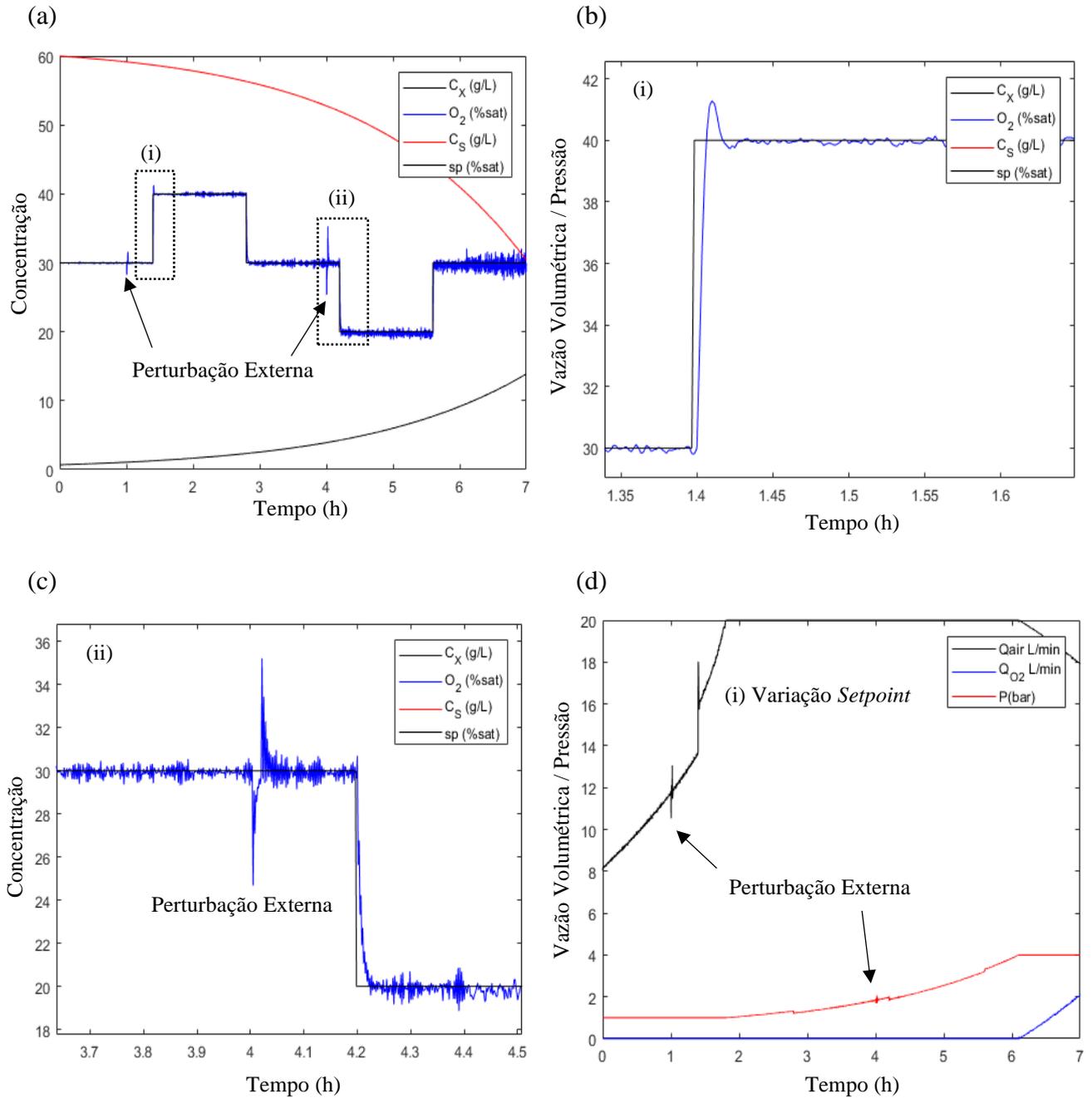
Optou-se por essa abordagem, uma vez que o servo modelo foi usado para teste no controlador RNA e para sintonização do controle PI. Logo foi necessário comparar o desempenho dos controladores frente a um perfil diferente de *setpoint*. A seguir são apresentados os resultados para cada um dos três controladores (PI, RNA e DMC) separadamente.

Para facilitar a comparação dos três controladores (RNA, PI e DMC), foi feito um resumo dos resultados, que são apresentados na última subseção 4.4.4, contemplando uma análise qualitativa e quantitativa através do resultado do erro integral (ISE).

##### 4.4.1 Controlador PI

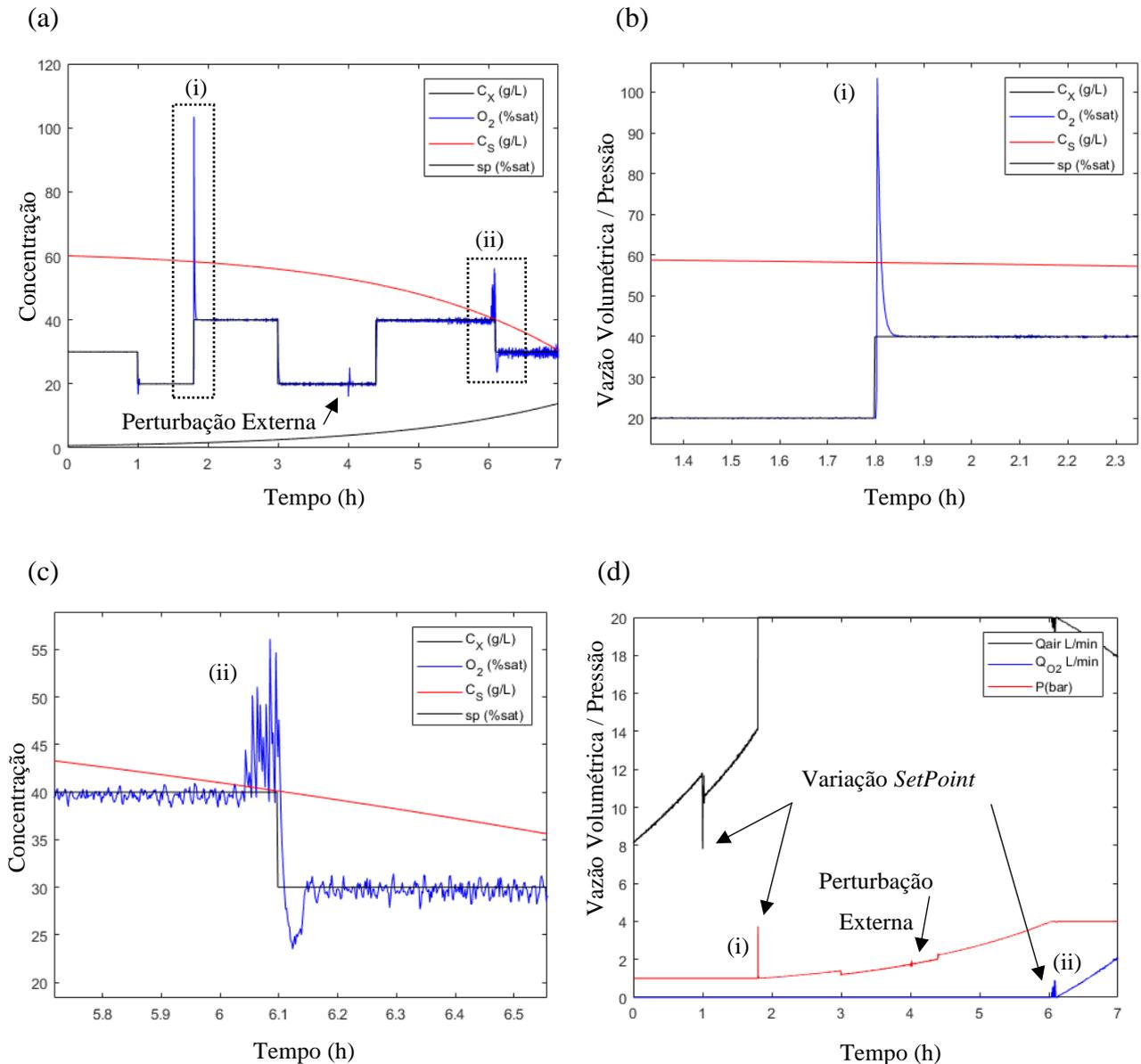
As Figuras 38 e 39 apresentam os resultados obtidos na simulação do controlador PI quando submetido aos cenários modelo e alternativo, respectivamente.

Figura 38 – Desempenho do controlador PI no cenário modelo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) região (i) submetida a uma variação de degrau, (c) região (ii) ao aplicar uma perturbação externa na pressão em 4 h seguido de variação em degrau no *setpoint* e (d) variáveis manipuladas decorrentes da ação do controlador PI.



Fonte: Do autor (2021)

Figura 39 – Desempenho do controlador PI no cenário alternativo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) regiões (i) e (c) (ii) submetidas a variações degrau no *setpoint*, e (d) variáveis manipuladas decorrentes da ação do controlador PI.



Fonte: Do autor (2021)

Ao observar a resposta após uma variação degrau no *setpoint* na região (i), percebe-se uma resposta satisfatória para OD, porém com uma sobrelevação considerável (Figura 38 (b)). O mesmo foi observado para a região (ii). De modo geral, verificou-se que apesar da sintonia ter sido realizada para o cenário modelo, obteve-se um resultado satisfatória no cenário alternativo. Na Figura 39 (c), entretanto, observa-se um desempenho inferior, apresentando oscilações antes mesmo da perturbação, o que não é desejável.

Essas elevadas sobrelevações frente a variações no *setpoint* confirmam a dificuldade em sintonizar o controlador PI para diferentes faixas de operação a fim de obter um desempenho satisfatório. Em relação às perturbações externas e internas, o controlador apresentou boa rejeição a essas variações.

É esperando, conforme observado, uma maior oscilação no final da simulação, uma vez que o desafio interno proposto, baseado na incerteza na constante de Monod ( $K_S$ ), se intensifica quanto menor a concentração de substrato (CAMPANI et al., 2019).

#### 4.4.2 Controlador RNA

As Figuras 40 e 41 apresentam os resultados obtidos na simulação do controlador RNA quando submetido aos cenários modelo e alternativo, respectivamente.

Figura 40 - Desempenho do controlador RNA no cenário modelo durante a fase de crescimento.

Resultados (a) das variáveis de saída, (b) região (i) ao aplicar uma perturbação externa na vazão de ar em seguida submetido a uma variação degrau no *setpoint*, (c) resposta na região (ii) ao aplicar uma variação degrau no *setpoint* e (d) variáveis manipuladas decorrentes da ação do controlador RNA. (Continua)

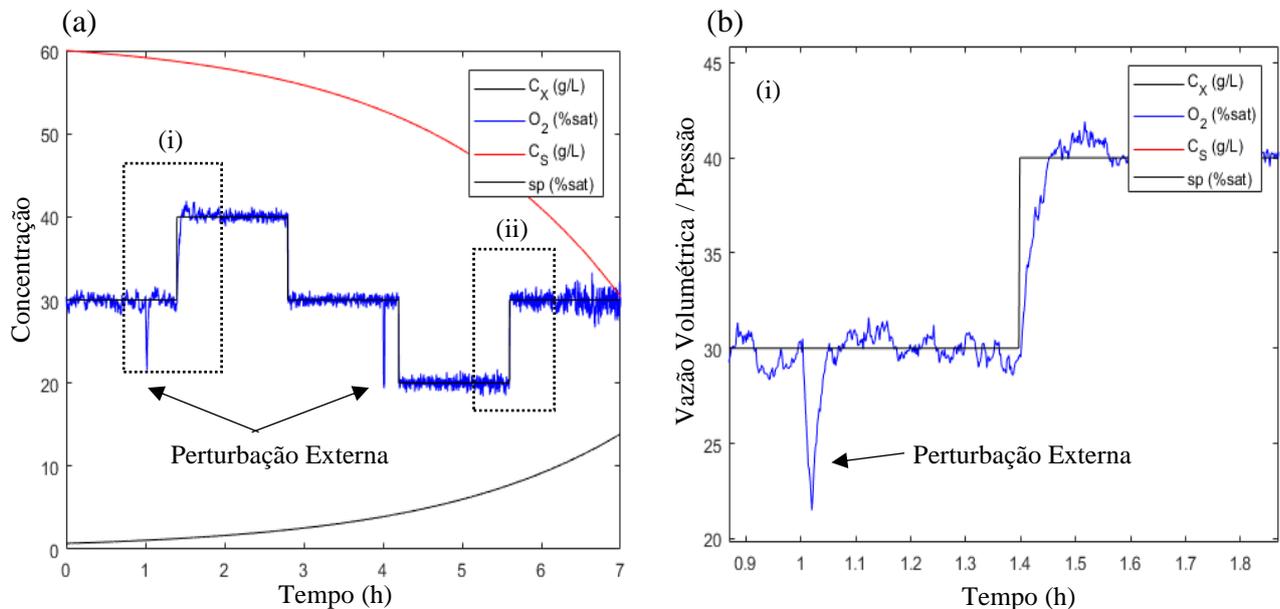
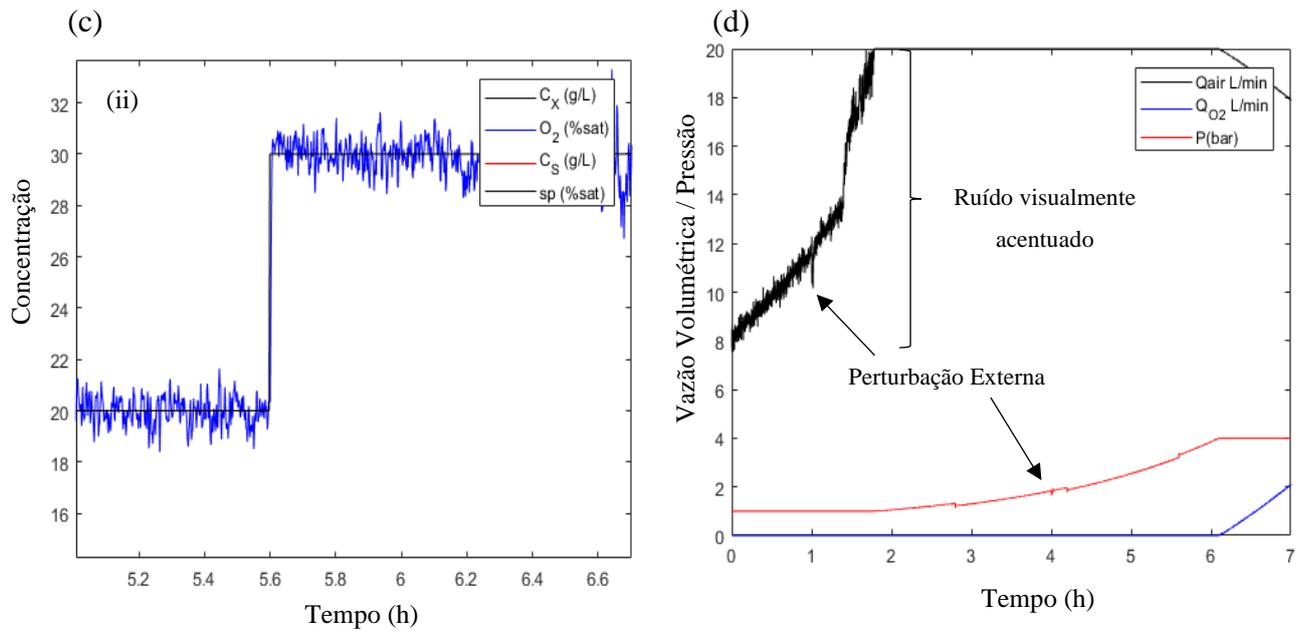


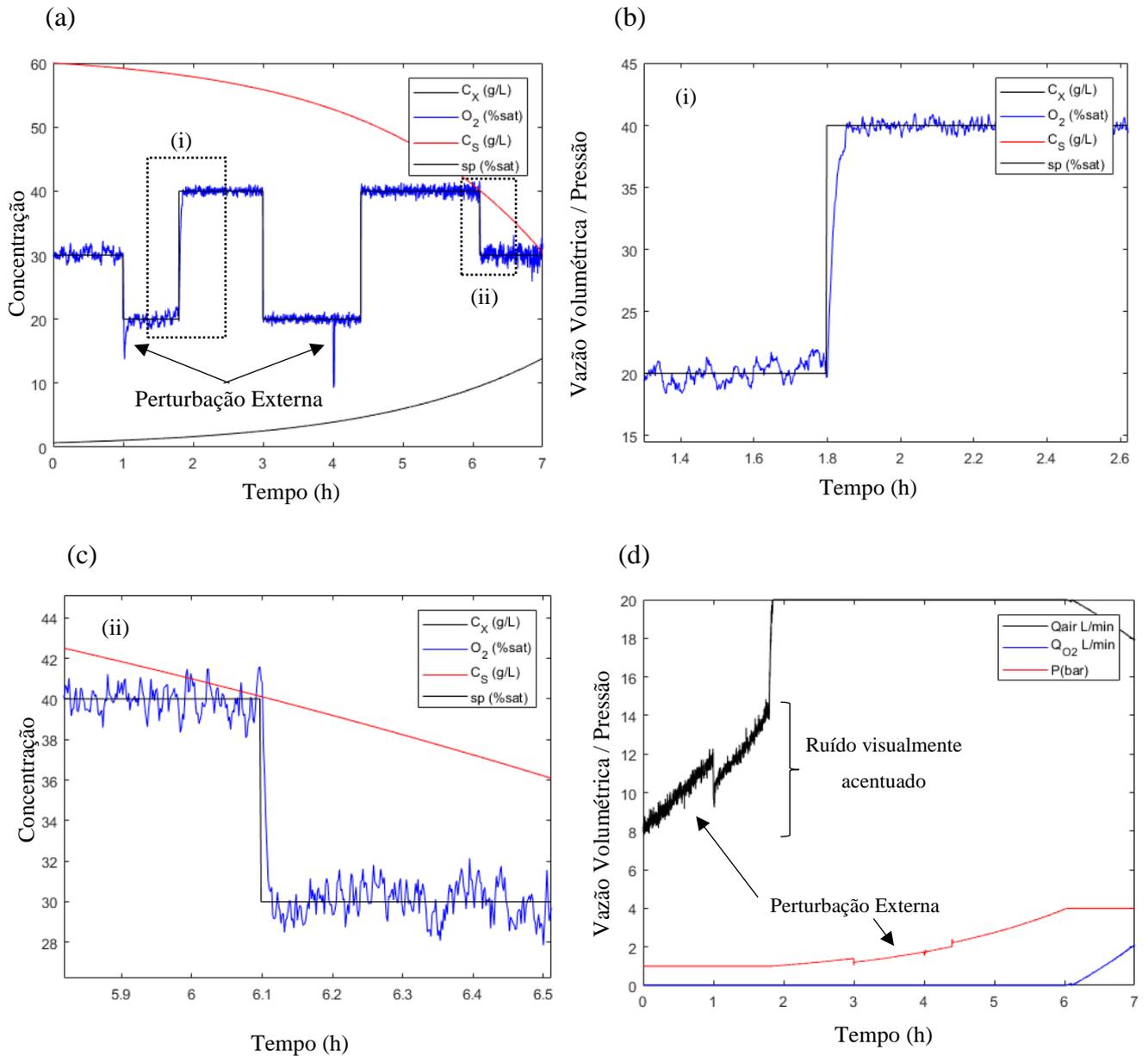
Figura 40 - Desempenho do controlador RNA no cenário modelo durante a fase de crescimento.

Resultados (a) das variáveis de saída, (b) região (i) ao aplicar uma perturbação externa na vazão de ar em seguida submetido a uma variação de grau no *setpoint*, (c) resposta na região (ii) ao aplicar uma variação de grau no *setpoint* e (d) variáveis manipuladas decorrentes da ação do controlador RNA. (Conclusão)



Fonte: Do autor (2021)

Figura 41 - Desempenho do controlador RNA no cenário alternativo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) regiões (i) e (c) (ii) submetidas a variações degrau no *setpoint* e (d) variáveis manipuladas decorrentes da ação do controlador RNA.



Fonte: Do autor (2021)

Analisando a resposta após perturbação degrau, nas regiões (i) e (ii), observa-se que o controlador RNA apresenta nenhuma ou moderada sobrelevação, para ambos cenários modelo e alternativo, como pode ser observado na Figura 40 (b) e (c) e também na Figura 41 (b) e (c).

Em relação às perturbações internas, externas e ao ruído na medida, o controlador RNA é mais sensível a essas variações. Uma vez que além de ser afetado pelo ruído na medida do

oxigênio dissolvido, as medidas das concentrações de substrato e células também são utilizadas pelo controlador e estas também apresentam ruído.

É importante notar a oscilação presente na variável manipulada vazão de ar, observada na Figura 40 (d) e na Figura 41 (d). Essa maior oscilação na vazão de ar do que nas outras variáveis manipuladas pode ser explicada pela menor sensibilidade do OD a essa variável. Com isso, ações de controle mais acentuadas na vazão de ar são necessárias para o controle do OD. Logo, pode ter havido sobreajuste, de modo que a rede não tenha sido capaz de absorver essa variação natural presente nos dados de treinamento.

Por outro lado, em termos operacionais, essa oscilação repentina na variável manipulada, embora possa levar a desgastes ao elemento final de controle, espera-se que a própria dinâmica do atuador filtre tais oscilações, que possuam amplitude máxima de aproximadamente  $0,8 \text{ L min}^{-1}$ .

#### 4.4.3 Controlador DMC

O controlador preditivo linear utilizado foi do tipo DMC (*Dynamic Matrix Control*), desenvolvido por Campani et al. (2019). Uma vez que o controlador já está implementado, não foi necessário utilizar o cenário modelo para ajuste de nenhum parâmetro do controlador. Portanto, é esperado obter um desempenho similar para os cenários modelo e alternativo.

A Figura 42 apresenta os resultados obtidos através da simulação no cenário modelo e a Figura 43 na simulação do cenário alternativo.

Figura 42 - Desempenho do controlador DMC no cenário modelo durante a fase crescimento. Resultados das (a) das variáveis de saída e (b) das variáveis manipuladas decorrentes da simulação.

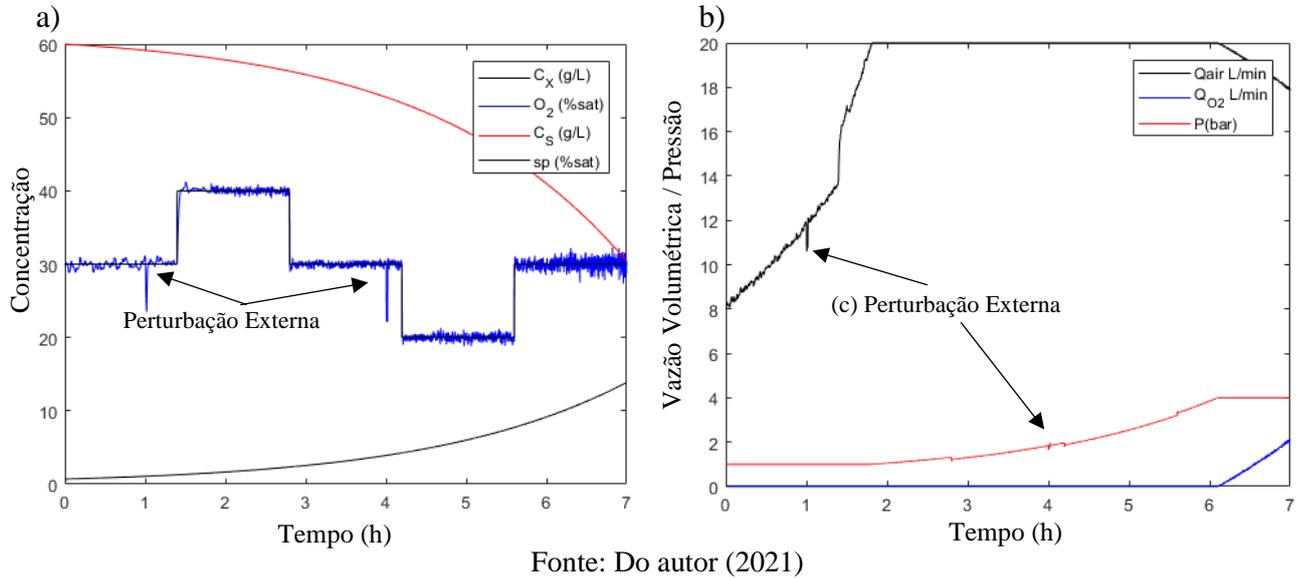


Figura 43 - Desempenho do controlador DMC no cenário alternativo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) regiões (i) e (c) (ii) indicam a resposta do sistema frente a variações degrau no *setpoint* e (d) variáveis manipuladas decorrentes da ação do controlador DMC. (continua)

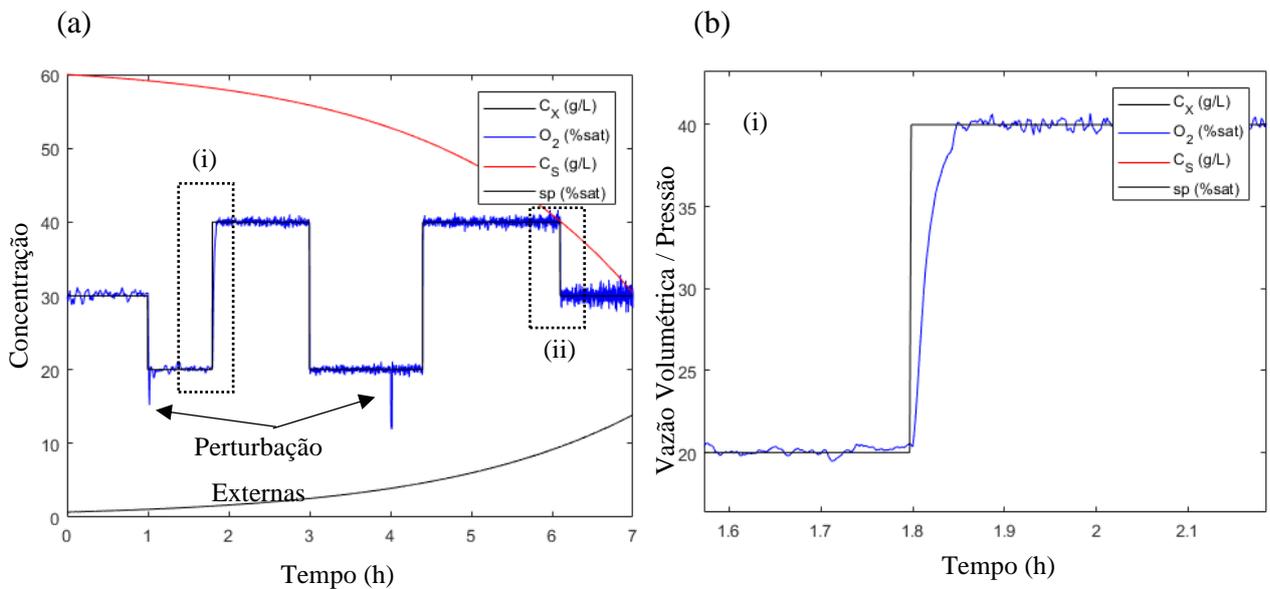
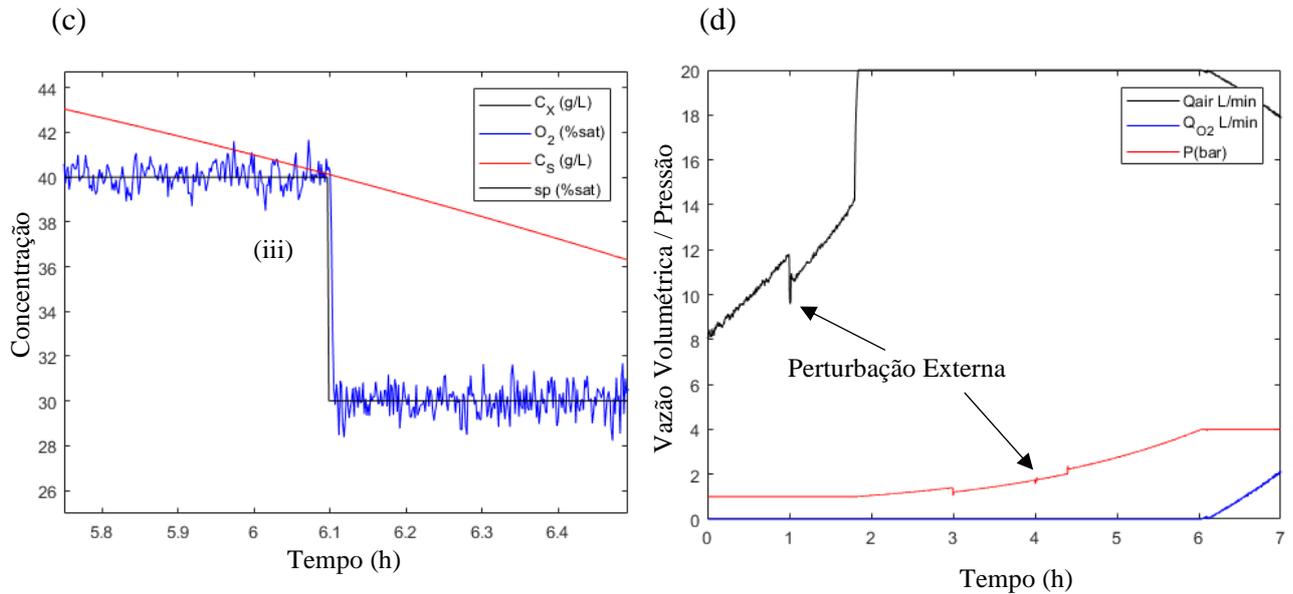


Figura 43 - Desempenho do controlador DMC no cenário alternativo durante a fase de crescimento. Resultados (a) das variáveis de saída, (b) regiões (i) e (c) (ii) indicam a resposta do sistema frente a variações degrau no *setpoint* e (d) variáveis manipuladas decorrentes da ação do controlador DMC. (Conclusão)



Fonte: Do autor (2021)

Conforme mencionado, o controlador DMC apresenta desempenho semelhante entre os cenários modelo e alternativo. Além disso, é possível notar que o DMC é um controlador conservador, alcançando resultados qualitativos satisfatórios em todos os desafios propostos.

#### 4.4.4 Comparação entre os controladores PI, RNA e DMC

Os resultados do erro integral ISE obtidos para as simulações realizadas na fase de crescimento estão apresentados na Tabela 7.

Tabela 7 – Resultado de erro integral médio ( $ISE \pm \overline{ISE}$ ) para ensaios simulados em triplicata

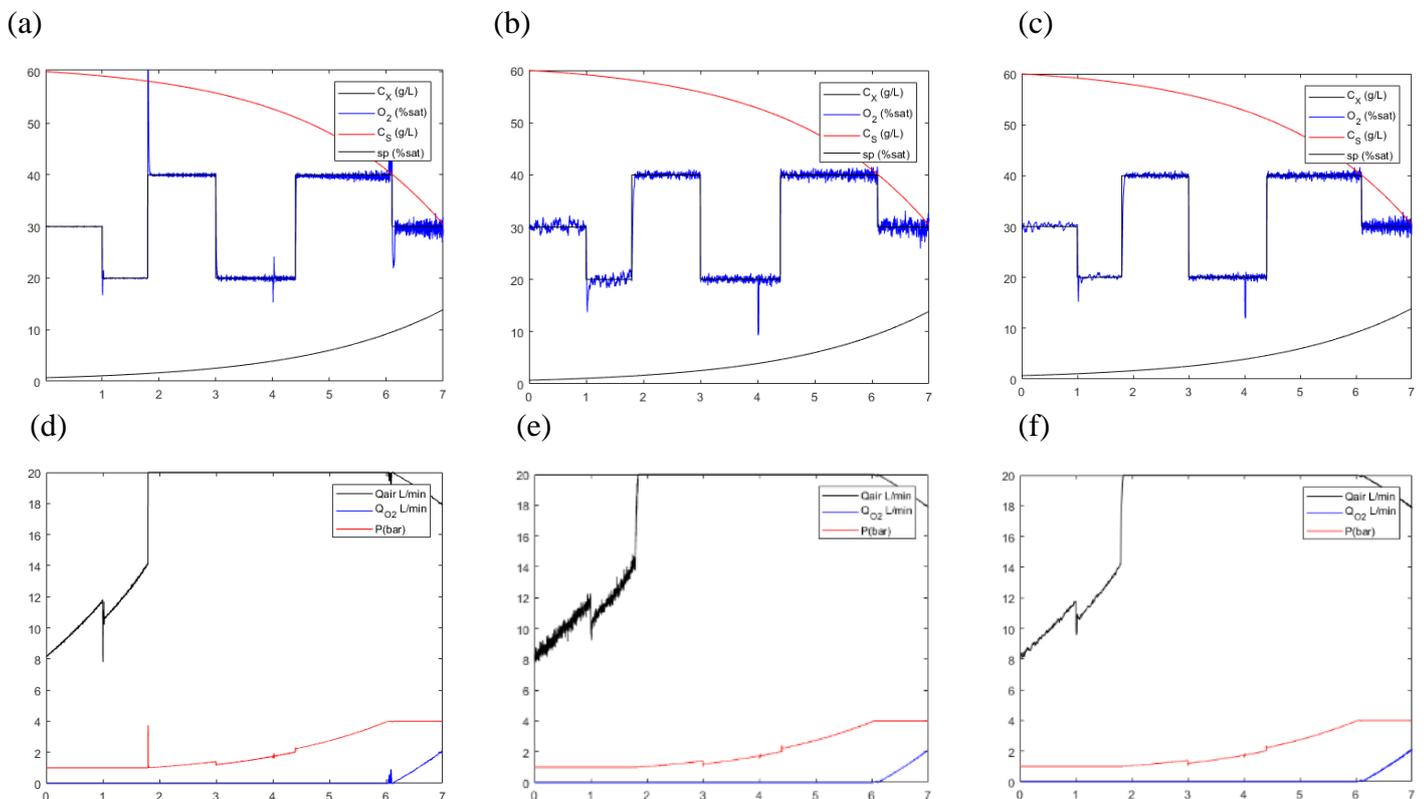
Controlador	Cenário Modelo	Cenário Alternativo
PI	$3,3 \pm 0,1$	$11 \pm 1$
RNA	$4,9 \pm 0,2$	$9,4 \pm 0,2$
DMC	$3,6 \pm 0,1$	$7,1 \pm 0,1$

Tendo em vista os resultados apresentados, a partir do cenário modelo é possível observar como o controlador PI, uma vez que foi sintonizado para aquele cenário, apresenta um desempenho estatisticamente superior aos controladores DMC e RNA. Entretanto, ao empregar o mesmo PI sujeito a outro perfil de *setpoint* (cenário alternativo), este apresenta um desempenho inferior aos demais controladores.

Conforme esperado, o controlador DMC apresenta melhor desempenho global, considerando os cenários modelo e alternativo. Já o RNA demonstra ter desempenho estatisticamente superior ao PI para o cenário alternativo, e com desempenho global satisfatório, demonstrando a capacidade da rede em prever e generalizar ações de controle em diferentes configurações de controle servo.

Ao observar os resultados de forma qualitativa com base na Figura 44, pode-se concluir que o controlador RNA pode ser utilizado para o controle do OD na fase de crescimento, não divergindo em nenhum dos desafios propostos.

Figura 44 – Resultados das variáveis de saídas – (a), (b) e (c) – decorrentes da simulação, na fase de crescimento no cenário alternativo, dos controladores PI, RNA e DMC, respectivamente, assim como as variáveis de saída – (d), (e) e (f).



Fonte: Do autor (2021)

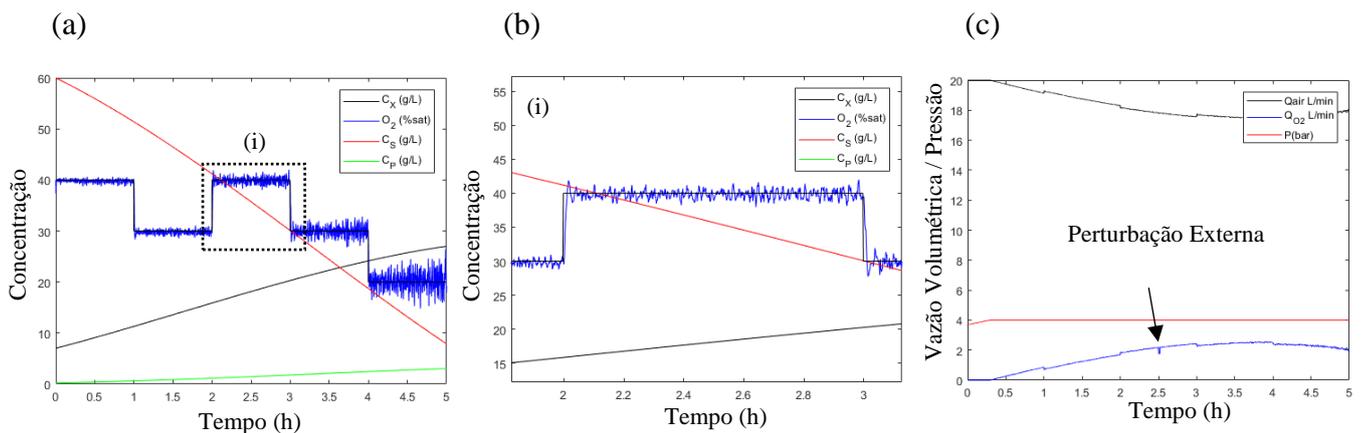
## 4.5 Desempenho dos controladores para a fase de indução

De modo análogo ao apresentado na fase de crescimento, os controladores foram submetidos aos mesmos desafios listados anteriormente: (a) controle servo, (b) perturbações externas, (c) perturbações externas e (d) ruídos nas medidas, para os cenários modelo e alternativo.

### 4.5.1 Controlador PI

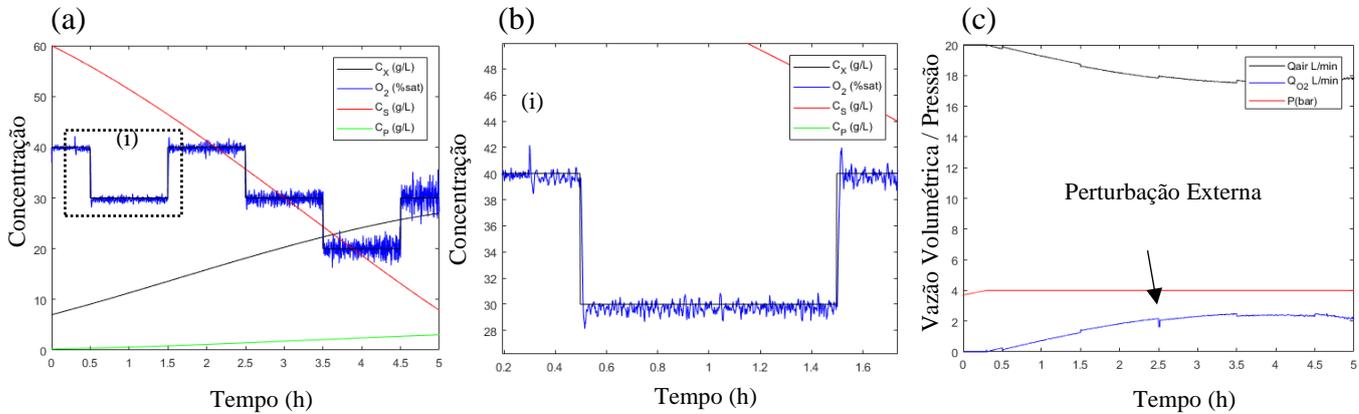
As Figuras 45 e 46 apresentam os resultados do controlador PI nos cenários modelo e alternativo, respectivamente.

Figura 45 - Desempenho do controlador PI durante a fase de indução no cenário modelo. Resultados (a) das variáveis de saída, (b) região (i) quando aplicada uma variação de grau no *setpoint* e (c) variáveis manipuladas.



Fonte: Do autor (2021)

Figura 46 - Desempenho do controlador PI durante a fase indução no cenário alternativo. Resultados (a) das variáveis de saída, (b) região (i) quando aplicada uma variação degrau no *setpoint* e (c) variáveis manipuladas.



Fonte: Do autor (2021)

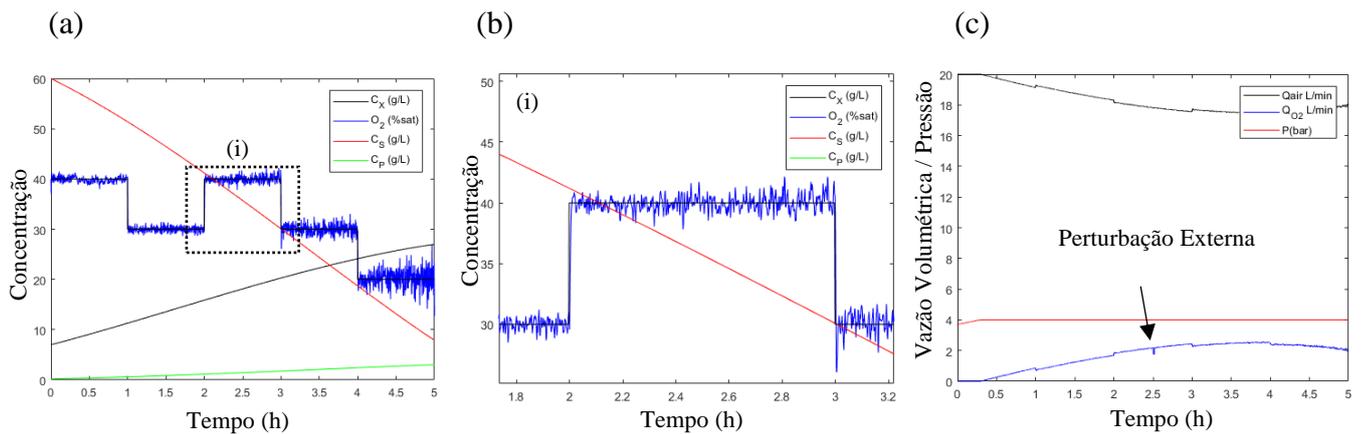
O controlador PI na fase de indução apresentou desempenho satisfatório, com resultados semelhantes para ambos os cenários modelo e alternativo, em decorrência do bom ajuste na sintonização dos parâmetros do controlador PI. Isso pode ser justificado, pois nesta fase a variável manipulada predominante é a vazão de oxigênio, ocorrendo, aproximadamente, de 0,5 h a 5 h. Sabe-se que o nível de oxigênio dissolvido no sistema é bastante sensível a essa variável, sendo necessárias variações mínimas para controlar o OD.

Por outro lado, percebe-se que a partir de 3,0 h, o valor da concentração de substrato é inferior a  $30,14 \text{ g.L}^{-1}$ , o equivalente a 7 h da fase de crescimento. Portanto, percebe-se que o sistema tende a oscilar devido ao maior impacto das perturbações internas (simulada como variações aleatórias em  $K_s$ ) no sistema na condição de baixos valores de concentração de substrato.

#### 4.5.2 Controlador RNA

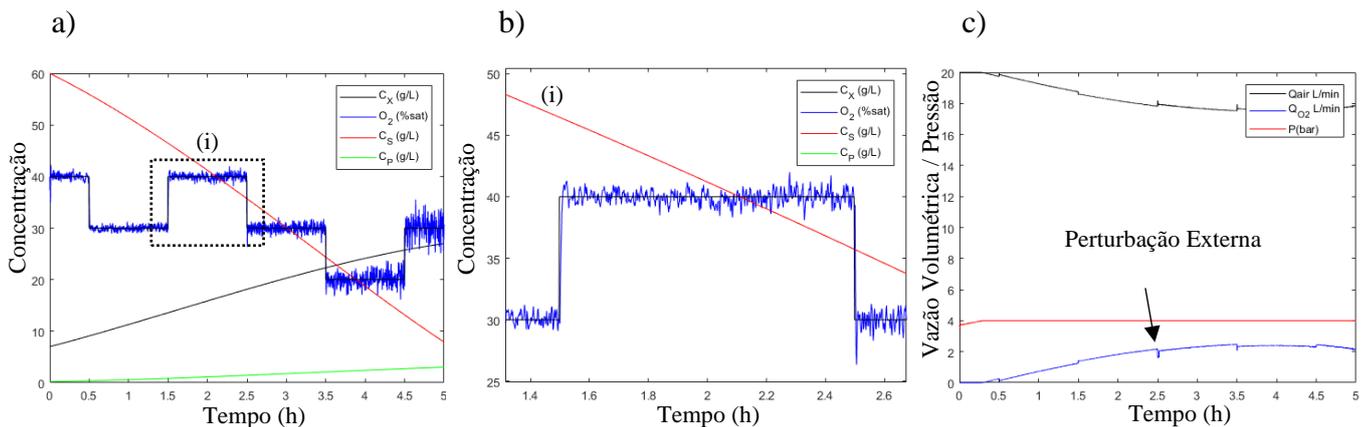
As Figuras 47 e 48 apresentam os resultados do controlador RNA nos cenários modelo e alternativo, respectivamente.

Figura 47 - Desempenho do controlador RNA durante a fase indução no cenário modelo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação degrau no *setpoint* e (c) variáveis manipuladas.



Fonte: Do autor (2021)

Figura 48 - Desempenho do controlador RNA durante a fase indução no cenário alternativo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação de *setpoint* e (c) variáveis manipuladas.



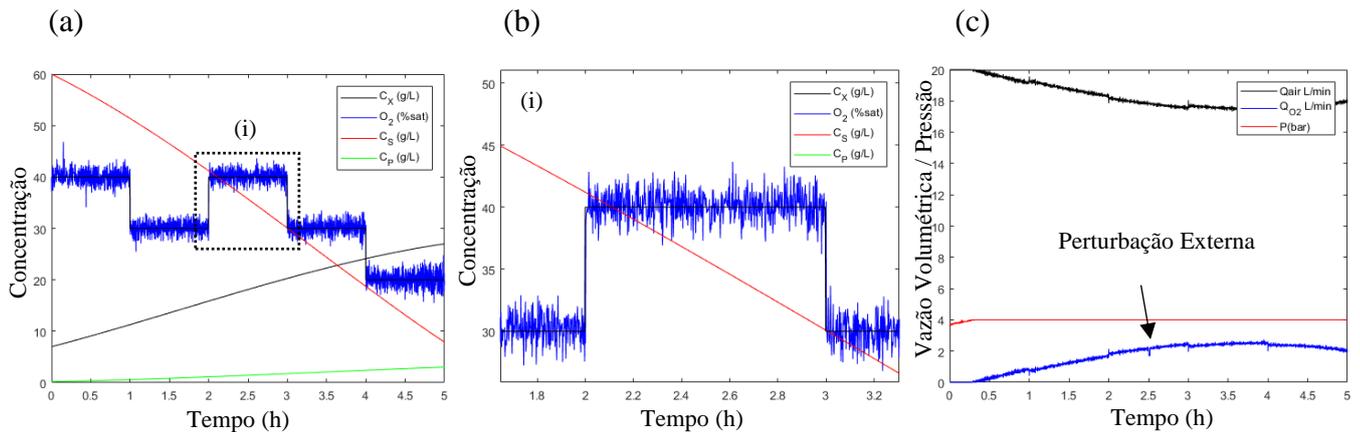
Fonte: Do autor (2021)

Conforme observado para o controlador PI, o controlador RNA também obtém desempenho satisfatório para ambos cenários, modelo e alternativo.

### 4.5.3 Controlador DMC

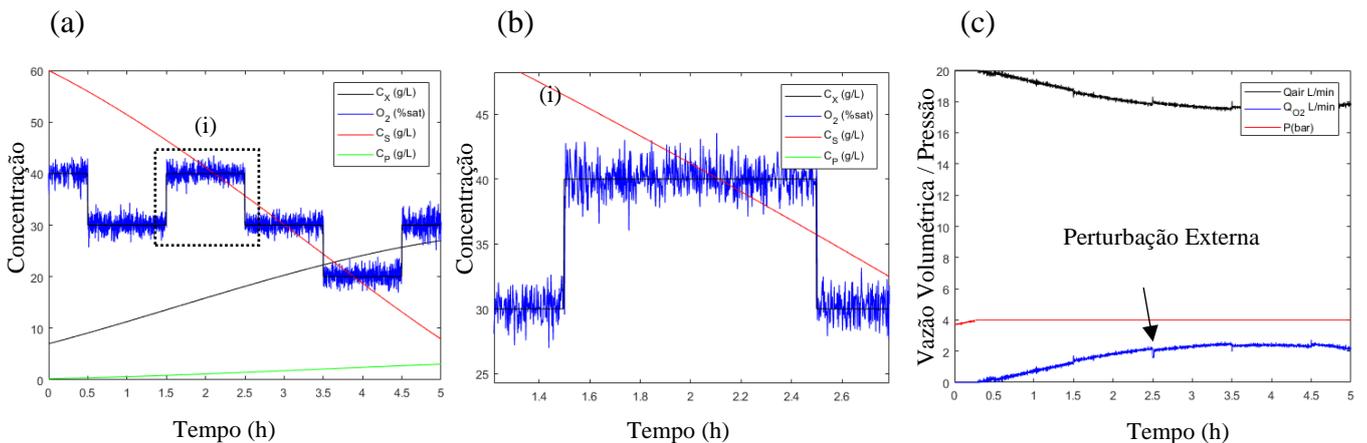
As Figuras 49 e 50 apresentam os resultados do controlador DMC nos cenários modelo e alternativo, respectivamente.

Figura 49 – Desempenho do controlador DMC durante a fase de indução no cenário modelo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação de *setpoint* e (c) variáveis manipuladas.



Fonte: Do autor (2021)

Figura 50 - Desempenho do controlador DMC durante a fase de indução no cenário alternativo. Resposta (a) das variáveis de saída, (b) região (i) submetida à variação de *setpoint* e (c) variáveis manipuladas.



Fonte: Do autor (2021)

Embora o desempenho do controlador DMC seja semelhante para ambos os cenários, modelo e alternativo, observa-se uma maior oscilação na variável controlada. Este fato pode ser

explicado devido ser um controlador preditivo, onde torna-se necessário utilizar medidas de todas as variáveis do processo (concentração de células, substrato e proteína) para a previsão das ações de controle. Portanto, os ruídos das medidas são propagados para as ações de controle, afetando, conseqüentemente, a variável controlada OD.

Logo, para melhorar o desempenho desse controlador, torna-se necessário a utilização de filtros, desde filtros de média móvel, até filtros avançados, como o filtro Kalman, conforme proposto por Campani et al. (2019). A utilização de filtros melhora o desempenho não somente para o controlador DMC, mas para todos os controladores, devido filtrar justamente o desafio do ruído presente nas medidas das variáveis do processo.

#### 4.5.4 Comparação PI, RNA e DMC

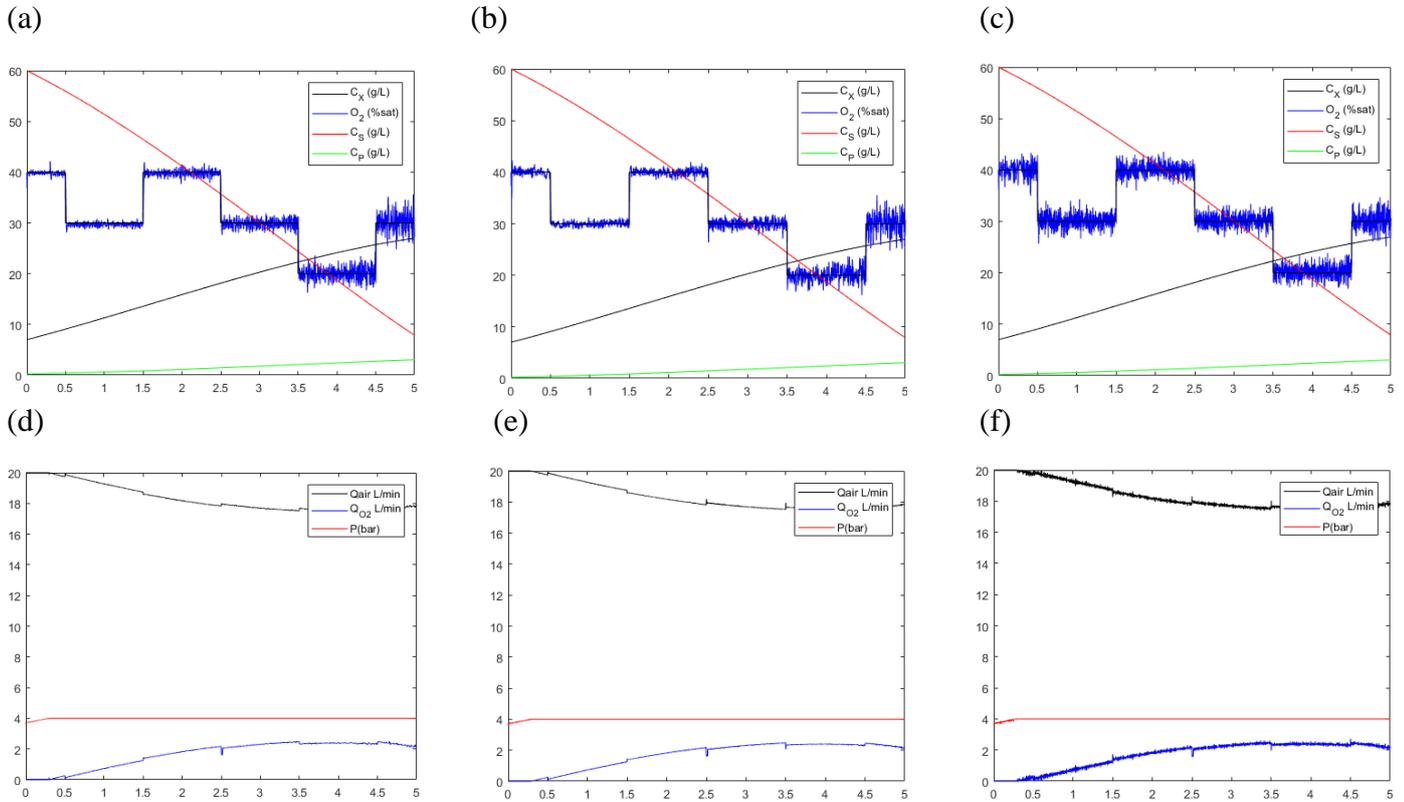
Os resultados de erro integral (ISE) para as simulações realizadas na fase de indução empregando os diferentes controladores avaliados estão apresentados na Tabela 8.

Tabela 8 – Resultado de erro integral médio ( $ISE \pm \overline{ISE}$ ) para ensaios simulados em triplicata

Controlador	Cenário Modelo	Cenário Alternativo
PI	6,5 $\pm$ 0,5	8,3 $\pm$ 0,2
RNA	6,6 $\pm$ 0,1	7,9 $\pm$ 0,2
DMC	9,6 $\pm$ 0,3	10,0 $\pm$ 0,2

Embora o esperado para o controlador DMC era apresentar desempenho superior aos demais controladores, isso não ocorreu. Conforme mencionado, no cenário proposto o DMC utiliza medidas de todas as variáveis do processo (concentração de células, substrato, oxigênio e proteínas), as quais possuem ruídos, impactando negativamente no desempenho desse controlador. O controlador neural, por exemplo, utiliza as três primeiras medidas, enquanto o controlador PI utiliza apenas medidas da variável controlada OD para a determinação da ação de controle. Portanto, é observado na Figura 51 (c) uma maior oscilação da resposta para o controlador DMC.

Figura 51 - Resultados das variáveis de saídas – (a), (b) e (c) – decorrentes da simulação, na fase de indução no cenário alternativo, dos controladores PI, RNA e DMC, respectivamente, assim como as variáveis de saída – (d), (e) e (f).



Fonte: Do autor (2021)

Deste modo, ao observar os resultados de forma qualitativa na Figura 51, pode-se concluir que o controlador RNA apresenta desempenho satisfatório superior aos demais controladores, PI e DMC, na fase de indução. Portanto, o controlador neural apresenta grande potencial de utilização para o controle do OD, não divergindo em nenhum dos desafios propostos *in silico*.

#### 4.5.5 Síntese dos resultados

Em síntese, perante os desafios apresentados, é apresentado na Tabela 9 um resumo das análises obtidas dos resultados apresentados anteriormente nas seções 4.4.1-4.4.3 e 4.5.1-4.5.3.

Tabela 9 – Síntese dos resultados obtidos a partir dos controladores PI, RNA e DMC para as fases de crescimento e indução

<b>Desafios</b>	<b>PI</b>	<b>RNA</b>	<b>DMC</b>
<b>Controle Servo</b>	Resposta apresenta sobrelevação significativa.	Resultado satisfatório.	Resultado satisfatório.
<b>Controle Regulatório: Perturbações Externas</b>	Controlador com melhor rejeição à perturbação na variável manipulada.	Rejeição adequada à perturbação na variável manipulada.	Rejeição adequada à perturbação na variável manipulada.
<b>Controle Regulatório: Perturbações Internas</b>	Moderada ou pouca oscilação.	Moderada ou pouca oscilação.	Moderada ou pouca oscilação.
<b>Ruído na medida</b>	Pouco sensível devido usar apenas medidas de OD.	Sensibilidade média, pois usa medidas de Cx, Cs e OD.	Bastante sensível, principalmente na fase de indução, uma vez que se baseia em medidas de todas as variáveis (Cx, Cs, OD e Cp).
<b><math>(ISE \pm \overline{ISE})</math> Fase de Crescimento</b>	11,0 $\pm$ 1	9,4 $\pm$ 0,2	7,1 $\pm$ 0,1
<b><math>(ISE \pm \overline{ISE})</math> Fase de Indução</b>	8,3 $\pm$ 0,2	7,9 $\pm$ 0,2	10,0 $\pm$ 0,2

## 5. CONCLUSÃO

Este trabalho propôs o desenvolvimento de um controlador avançado do oxigênio dissolvido (OD) para bioprocessos aeróbios baseado em redes neurais (RNA). Esse controlador inovador foi desenvolvido e implementado em simulações do processo aeróbio de produção de uma proteína recombinante pela bactéria *Escherichia coli*. Outros controladores como o PI (Proporcional-Integral) e o preditivo DMC (*Dynamic Matrix Control*) foram também avaliados e comparados ao controlador RNA.

O controlador neural desenvolvido neste trabalho teve desempenho satisfatório para o controle do nível do OD, em ambas as fases do cultivo celular, crescimento e indução. De forma qualitativa, em nenhum dos desafios propostos a variável controlada divergiu do *setpoint*. Quantitativamente, o desempenho do controlador RNA foi estatisticamente próximo ou até mesmo com desempenho superior ao controlador tradicional PI na fase de crescimento, e ao controlador DMC na fase de indução. A RNA possui a vantagem de não precisar ser adaptada ao longo do processo, ao contrário do controlador PI, que foi sintonizado para diferentes intervalos de tempo do cultivo.

Outro benefício do controlador RNA é a sua maior facilidade de implementação em relação ao DMC, uma vez que independe do modelo do processo. No entanto, a RNA possui como desvantagem a sua aplicabilidade limitada às condições operacionais incluídas no conjunto de dados de treinamento, sem possibilidade de extrapolações.

Para melhorar o desempenho dos controladores, com relação a possíveis oscilações provocadas por ruídos na medida, pode-se utilizar filtros nas medidas para a implementação em um sistema real, seja filtros básicos, como de média móvel, até aqueles mais sofisticados, como os filtros de Kalman e horizonte móvel.

Como melhorias com relação ao treinamento supervisionado das redes, pode-se utilizar um conjunto de dados mais diversos, que inclua diferentes formas de operação, aumentando a abrangência dos dados de aprendizado e, conseqüentemente, ampliando a aplicabilidade do controlador RNA. Além disso, pode-se também utilizar normalização dos dados para evitar sobreajuste, situação essa que pode diminuir a oscilação da ação de controle na variável  $Qar$ .

## REFERÊNCIAS

ALEXANDER R, CAMPANI G, DINH S; et al. **Challenges and Opportunities on Nonlinear State Estimation of Chemical and Biochemical Processes**. Processes; 8(11):1462. <https://doi.org/10.3390/pr8111462,2020>

ASSIS, A. J. **Identificação e Controle de Processos não Lineares Utilizando Redes Neurais Artificiais**. Tese de Doutorado- UNICAMP. Campinas, 2001

AZWARM. A; HUSSAIN, K. B; RAMACHADRAN; **The study of neural network-based controller for controlling dissolved oxygen concentration in a sequencing batch reactor**. Bioprocess Biosyst Eng 28: 251–265 DOI 10.1007/s00449-005-0031-2. 2006

BALLINI, R., ZUBEN, F. J. von. Book reviews: **Application of neural networks to adaptive control of nonlinear systems**. Automatica, 2000.

BAUGHMAN, D. R., LIU, Y. A. **Neural networks in bioprocessing and Chemical Engineering**. Academic Press, 1995.

BEGA, Egídio A.; FINKEL, Vitor S.; KOCH, Ricardo; **Instrumentação industrial, Instituto Brasileiro de Petróleo e Gás**. Rio de Janeiro, 2006.

BEHROOZI, K. N; TAVAKOLI, T.; GHASSEMIAN H.; et al. **Applied machine vision and artificial neural network for modeling and controlling of the grape drying process**. Computers and Electronics in Agriculture, 98, 205–213. doi:10.1016/j.compag.2013.08.010. 2013

BERLEC, A.; STRUKELJ, B. **Current state and recent advances in biopharmaceutical production in Escherichia coli, yeasts and mammalian cells**. Journal of Industrial Microbiology & Biotechnology. v. 40, n. 3, p. 257-274, 2013.

BISHOP, C. M. **Neural network for pattern recognition**. Oxford University. Oxford, 1995.

BOSKOVIC, J. D., NARENDRA, K. S. **Comparison of linear, nonlinear and neural-network -based adaptive controllers for a class of fed-batch fermentation processes**. Automatica. 1995.

BRAGA, A. P.; CARVALHO, A. C. P. L. F.; LUDEMIR, T. B. **Redes neurais artificiais: teoria e aplicações**. 2. ed. Rio de Janeiro: LTC, 2007.

CAMPANI, G, et al. **A hierarchical state estimation and control framework for monitoring and dissolved oxygen regulation in bioprocesses**. *Bioprocess and Biosystems Engineering*; <https://doi.org/10.1007/s00449-019-02143-4>. 2019

CAMPANI, G. **Otimização do Controle do Oxigênio Dissolvido em Biorreator Airlift Pressurizado: Aplicação em Cultivos de Escherichia Coli Recombinante**. Tese de doutorado, UFScar, 2018.

CAMPANI, G; GONÇALVES, G; ZANGIROLAMI, TC; et al. **Recombinant Escherichia coli cultivation in a pressurized airlift bioreactor: assessment of the influence of temperature on oxygen transfer and uptake rates**. *Bioprocess Biosyst Eng* 40:1621–1633. <https://doi.org/10.1007/s00449-017-1818-7>. 2017.

CAMPANI, G.; SANTOS MP; SILVA, GG; et al. **Recombinant protein production by engineered Escherichia coli in a pressurized airlift bioreactor: a techno-economic analysis**. *Chem Eng Process Process Intensif* 103:63–69. <https://doi.org/10.1016/j.cep.2015.10.020>. 2016

CASSIANO, L, C, M, F. **Avaliação do Desempenho de Biorreator “Airlift” para produção de ácido clavulânico por *Streptomyces clavuligerus***. São Carlos: UFSCAR, 2003.

CERRI, M, O. **Hidrodinâmica e transferência de oxigênio em três biorreatores airlift de circulação interna geometricamente semelhantes**. São Carlos: UFSCAR, 2009

CHAUVIM, Y; RUMELHART, D.E.; **Backpropagation theory: architectures and applications**. Laurence Erlbaum Associates Inc. 1995.

CHISTI, M, Y. **Airlift Bioreactors**, Elsevier, New York: 1989.

CUTLER, C.R., RAMAKER, B.L. - **Dynamic Matrix Control – a Computer Control Algorithm**, AICHE National Meeting, Houston, TX, 1979.

DUTRA, C. B. S. **Controle Preditivo Multiobjetivo para Processos com Atraso**. Tese Doutorado. Florianópolis, 2003.

EMPRAPA. **Emprapa e Bioeconomia**. Disponível em <<https://www.embrapa.br/tema-bioeconomia/sobre-o-tema>> Acesso: 10/03/2021.

FERRARI, A. C. K. **Controlador PID sintonizado por Redes Neurais Artificiais**. Trabalho de Conclusão de Curso. Universidade Federal do Paraná, Curitiba, 2010. Disponível em: <http://www.eletrica.ufpr.br/ufpr2/tccs/190.pdf>. Acesso em: 08/11/2020.

GAO, H.; GU, W. **Optimization of Polysaccharide and Ergosterol Production from Agaricus brasiliensis by Fermentation Process**. Biochemical Engineering Journal. v. 33, p. 202–210, 2007.

HAN, H. G; QIAO, J.F. **Adaptive dissolved oxygen control based on dynamic structure neural network**. Applied Soft Computing, doi:10.1016/j.asoc.2011.02.014. 2011

HAYKIN, S. **Redes Neurais - Princípios e Prática**. Bookman, 2001.

HUANG, T.-K.; MCDONALD, K. **Bioreactor engineering for recombinant protein production in plant cell suspension cultures**. Biochemical Engineering Journal, v. 45, n. 3, p. 168–184, ago. 2009.

JACQUEZ, J. A. **Design of experiments**. Journal of the Franklin Institute-Engineering and Applied Mathematics, 335B (2), 259-279. 1998

JESUINO, A. C. B. M; **Diferentes estratégias de cultivo de Escherichia coli recombinante para produção de proteína verde fluorescente (Green Fluorescent Protein - GFP)**. Tese de Doutorado. Araraquara, 2020.

KITTISUPAKORN, P., THITIVASOOK, P., HUSSIAN, M. A., et al.. **Neural network based model predictive control for a steel pickling process**. Journal of Process Control, doi:10.1016/j.jprocont.2008.09.003. 2009

LIM, H.C; LEE, K.-S. **Control of Bioreactor Systems**. Karl Schügerl (ed), **Biotechnology - Measuring, Modelling and Control**. VCH, Weinheim. 1990.

LIU Y.; XU S.; HASHIMOTO S.; et al; **A Reference-Model-Based Neural Network Control Method for Multi-Input Multi-Output Temperature Control System**; doi:10.3390/pr8010050; 2020.

MCCULLOCH, W.S; PITTS, W.H.: **A logical calculus of ideas immanent in nervous activity**. Bull Math Biophys. 1943.

METE, T.; OZKAN, G.; HAPOGLU, H.; et al.; **Control of dissolved oxygen concentration using neural network in a batch bioreactor**. Computer Applications in Engineering Education. doi:10.1002/cae.20430. 2010.

MILLER, M.R.; KWOK, K.E.; SHAH, L.S; et al. **Predictive PID**. ISA Transactions, Vol. 38. 1999.

MORAIS, I; ALMEIDA, L. et al. **Redes Neurais Artificiais para Controle de uma Planta de Nível**. ABRICOM, 2016. Disponível em <[http://www.abricom.org.br/wp-content/uploads/2016/08/CBRN2005\\_187.pdf](http://www.abricom.org.br/wp-content/uploads/2016/08/CBRN2005_187.pdf)> Acesso em: 08/11/2020.

MUNIZ, L. A. R. **Controle Preditivo Adaptativo Aplicado a um Reator de Pirólise Operando em Regime Semibatelada**. Tese Doutorado. Florianópolis, 2004.

NGUYEN, D; WIDROW, B. **Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights**. Proceedings of the International Joint Conference on Neural Networks, 3:21–26, 1990

NICE. S. **Engenharia de sistemas de controle**. 5. ed. Rio de Janeiro: LTC. 2009.

OGATA, K. **Modern Control Engineering**. 4th, A'eizh. 2002.

OLIVEIRA, A. L. LIMA, **Instrumentação – Fundamentos de Controle de Processo**. SENAI - Serviço Nacional de Aprendizagem Industrial. Espírito Santo. 1999.

PERCIANI, CT; BARAZZONE, GC; GOULARD C; et al. **Conjugation of polysaccharide 6B from Streptococcus pneumoniae with pneumococcal surface protein A: PspA conformation and its effect on the immune response**. Clin Vaccine Immunol 20:858–866. <https://doi.org/10.1128/CVI.00754-12>, 2013.

PEREIRA, J N. **Tecnologia de bioprocessos**. Rio de Janeiro: Escola de Química/UFRJ, ISBN 978-85-903967-2-7. 2008.

PERRY, R. H.; CHILTON, C. H; **Chemical Engineers' Handbook**, 5th. edt., New York, McGraw-Hill Book Co., 1973.

POLARIS MARKET RESEARCH, **Biotechnology Market Size, Share, Trends & Analysis Report 2020-2026**. Disponível em <<https://www.polarismarketresearch.com/industry-analysis/biotechnology-market>>. Acesso em 10/03/2021.

PRESS, W.H.; TEUKOLKY, S.A.; VETTERLING, W.T.; et al. **Numerical recipes in fortran**. Cambridge University. 1990.

QIN, S.J., BADGWELL, T.A. – **A Survey of Industrial Model Predictive Control Technology**. Control Engineering Practice, v 11. 2003.

ROSSI, M, J. **Tecnologia para produção de inoculantes de fungos ectomicorrizicos utilizando cultivo submerso em biorreator AirLift**. Florianópolis: UFSC, 2006.

SANTOS, A. F. **Aplicação de RNA no controle eficiente do motor de indução trifásico**. Dissertação Mestrado, UFPE, 2008.

SCHMIDELL, W. **Agitação e Aeração em Biorreatores**. São Paulo: Biotecnologia Industrial, Engenharia Bioquímica, v. 2, c.14, Ed. Edgard Blücher , 2001

SCHMIDT, F.R. Recombinant **expression systems in the pharmaceutical industry**. **Applied Microbiology and Biotechnology**, 65, 363-372, 2004.

SCHNITMAN, L. **Controladores Preditivos baseados em Redes Neurais Artificiais**. Dissertação de Mestrado. Universidade Federal da Bahia, Salvador, 1998. Disponível em: <<http://www.ppgee.eng.ufba.br/teses/3fa84f4664183cab6c0c9022d582c762.pdf>> Acesso em: 08/11/2020.

SEBORG, E. D. et al. **Process Dynamics and Control**. 3. ed. Hoboken, NJ: John Wiley & Sons Inc, 2017.

SHULER, K. **Bioprocess Engineering Basic Concepts**. 2nd Edition Paperback – January 1, 2010

SILVA, C. A. A. **Contribuição para o estudo do embarque de uma rede neural artificial em field programmable gate array**. Natal, RN, 2010.

SILVA, C. A. A. **Implementação de uma matriz de neurônios dinamicamente reconfigurável para descrição de topologias de redes neurais artificiais multilayer perceptrons**. Tese de Doutorado. Natal, 2015.

SIMAS, E. **Notas de Aula Disciplina Introdução às Redes Neurais Artificiais**. UFBA, Bahia 2013.

SMITH-DOERR, L; OWEN-SMITH, J., KENNETH, W. K., et al. **Networks and Knowledge Production: Collaboration and Patenting in Biotechnology**. Forthcoming in **Corporate Social Capital**, eds. R.T.A.J. Leenders & S. Gabbay. Addison Wesley, Boston, USA, 1998.

STEPHANOPOULOS, G. **Chemical process control: An introduction to theory and practice**, Prentice-Hall International Inc. New Jersey, 1984.

THOMASI, S, S. **Avaliação de parâmetros de desempenho de três modelos de biorreatores pneumáticos de bancada**. São Carlos: UFSCAR, 2010.

VELOSO, A. C. A.; **Optimização de Estratégias de alimentação para a identificação de Parâmetros de um modelo de *E.coli*. Utilização do modelo em monitorização e controle**. Universidade Minho, 2006.