



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CAMPUS FLORIANÓPOLIS  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE AUTOMAÇÃO E SISTEMAS

Caio Henrique Oliveira Cunha

# **Redes ReLU como Modelos Inteiros-mistos Aplicadas à Sintonia de Simuladores de Poços de Petróleo**

Florianópolis

2022



Caio Henrique Oliveira Cunha

# **Redes ReLU como Modelos Inteiros-mistos Aplicadas à Sintonia de Simuladores de Poços de Petróleo**

Dissertação submetida ao Programa de Pós-Graduação em Engenharia de Automação e Sistemas da Universidade Federal de Santa Catarina para a obtenção do título de Mestre em Engenharia de Automação e Sistemas

Orientador: Prof. Eduardo Camponogara, Dr.

Coorientador: Prof. Laio Oriel Seman, Dr.

Florianópolis

2022

Ficha de identificação da obra elaborada pelo autor,  
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Cunha, Caio Henrique Oliveira

Redes ReLU como Modelos Inteiros-mistos Aplicadas à  
Sintonia de Simuladores de Poços de Petróleo / Caio Henrique  
Oliveira Cunha ; orientador, Eduardo Camponogara,  
coorientador, Laio Oriel Seman, 2022.

71 p.

Dissertação (mestrado) - Universidade Federal de Santa  
Catarina, , Programa de Pós-Graduação em , Florianópolis,  
2022.

Inclui referências.

1. . 2. Sistemas de Produção de Petróleo. 3. Inteligência  
Artificial. 4. Redes Relu. 5. Otimização Discreta. I.  
Camponogara, Eduardo. II. Seman, Laio Oriel. III.  
Universidade Federal de Santa Catarina. Programa de Pós  
Graduação em . IV. Título.

Caio Henrique Oliveira Cunha

**Redes ReLU como Modelos Inteiros-mistos Aplicadas à Sintonia de Simuladores de Poços de Petróleo**

O presente trabalho em nível de mestrado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. Argimiro R. Secchi, Dr.  
Universidade Federal do Rio de Janeiro

Prof. Bruno Araújo Cautiero Horta, Dr.  
Universidade do Vale do Itajaí

Prof. Eric Aislan Antonelo, Dr.  
Universidade Federal de Santa Catarina

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de mestre em Engenharia de Automação e Sistemas.

---

Prof. Werner Kraus Junior, Dr.  
Coordenador do Programa

---

Prof. Eduardo Camponogara, Dr.  
Orientador

Florianópolis, 29 de Julho de 2022.



# RESUMO

Nas operações diárias da produção de petróleo são empregadas ferramentas computacionais no auxílio à tomada de decisão, as quais dependem de modelos estáticos representativos de poços, reservatórios e processos. Esses modelos possuem parâmetros-chave que devem ser sintonizados para melhor representação da realidade. Na prática, manter os modelos de poços atualizados é uma tarefa desafiadora, pois estes parâmetros só são revisados quando um poço está sujeito a um teste de separação. Deste modo, os modelos vigentes podem não refletir as condições reais devido à desatualização dos parâmetros-chave, acarretando em consideráveis perdas de produção. Neste contexto, este trabalho apresenta uma metodologia que objetiva encontrar parâmetros-chave atualizados (razão gás-óleo, razão água-óleo e índice de produtividade), de modelos estáticos representativos de poços, encontrados em plataformas reais de petróleo. Para tanto, foram empregadas redes ReLU no aprendizado do comportamento dos poços para diversas condições físicas; a seguir, tais redes foram descritas como modelos inteiros-mistos e empregadas em problemas de otimização visando a sintonia dos parâmetros-chave. Resultados simulados do emprego da metodologia em cenários sintéticos, demonstraram que os parâmetros-chave encontrados foram responsáveis por gerar maiores ganhos de produção de petróleo em comparação ao uso de parâmetros puramente históricos. Além disto, a metodologia proposta auxilia na predição da vazão de petróleo de cada poço individualmente, medições tipicamente não disponíveis quando o poço não se encontra em teste de separação. Por fim, a metodologia apresentada permite a sintonia simultânea de mais parâmetros (maior dimensionalidade) fazendo um uso de menos variáveis binárias, em relação a outras metodologias clássicas de aproximação como, por exemplo, a linearização por partes de funções não-lineares.

**Palavras-chaves:** Sistemas de Produção de Petróleo, Redes Neurais, Redes ReLU, Inteligência Artificial, Otimização Discreta.



# ABSTRACT

In daily operations of oil production, computational tools are used to aid decision making, which depend on static models representing wells, reservoirs, and processes. These models have key parameters that must be tuned for a better representation of reality. In practice, keeping well models up to date is a challenging task, as these parameters are only revised when a well undergoes a separation test. Thus, current models may not reflect real conditions due to outdated key parameters, resulting in considerable production losses. In this context, this work presents a methodology that aims to update key parameters (gas-oil ratio, water-oil ratio, and productivity index) of representative static models of wells found in real oil platforms. Therefore, ReLU networks were used to learn the behavior of wells for different physical conditions; then, such networks were described as mixed-integer models and used in optimization problems aiming at tuning the key parameters. Simulated results of using the methodology in synthetic scenarios showed that the key parameters found were responsible for generating greater gains in oil production compared to the use of purely historical parameters. In addition, the proposed methodology helps to predict the oil flow from each well individually, measurements typically not available when the well is not undergoing a separation test. Finally, the presented methodology allows the simultaneous tuning of more parameters (greater dimensionality) making use of fewer binary variables, in relation to other classical approximation methodologies, such as *piecewise-linearization* of nonlinear functions.

**Key-words:** Oil Production Systems, Neural Networks, ReLU Networks, Artificial Intelligence, Discrete Optimization.



# LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama ilustrativo do processo realizado nesse trabalho para aplicação e validação da metodologia. . . . .	19
Figura 2 – Diagrama ilustrativo da estrutura de um sistema de produção de petróleo que opera com injeção contínua de gás comprimido ( <i>gas-lift</i> ). . . . .	22
Figura 3 – Diagrama simplificado de um processo de elevação artificial de petróleo em poços. . . . .	23
Figura 4 – Curva de Desempenho de Poço (WPC). . . . .	24
Figura 5 – Uma estrutura hierárquica típica da cadeia aprendizado de máquina ( <i>machine learning</i> ). . . . .	29
Figura 6 – Modelo de um único neurônio de uma rede neural artificial. . . . .	30
Figura 7 – Função de Ativação Etapa Binária. . . . .	31
Figura 8 – Função de Ativação Linear. . . . .	32
Figura 9 – Função de Ativação ReLU. . . . .	33
Figura 10 – Arquitetura de uma Rede Neural Alimentada Adiante. . . . .	34
Figura 11 – Arquitetura de uma Rede Neural com Múltiplas Camadas. . . . .	34
Figura 12 – Espaço de busca do Algoritmo <i>Branch-and-Bound</i> . . . . .	39
Figura 13 – Diagrama Ilustrativo da Estrutura de uma Rede Neural ReLU. . . . .	40
Figura 14 – Ilustração da Aproximação Linear Por Partes (PWL) de uma Função Não-linear. . . . .	43
Figura 15 – Ilustração da Representação Linear por Partes da Função Bidimensional $f(x, y) = e^{-y} + e^{-x^2} - x^2 + xy - 2y$ . . . . .	45
Figura 16 – Particionamento do Domínio $D = [-2, 2] \times [-2, 2]$ Segundo a Triangulização <i>Union Jack</i> . . . . .	45
Figura 17 – Diagrama Ilustrativo da Arquitetura das Redes Neurais de Cada um dos Poços. . . . .	53
Figura 18 – Diagrama Ilustrativo das Etapas Desenvolvidas Para Identificação e Validação dos Parâmetros-chave. . . . .	56
Figura 19 – Diagrama Ilustrativo das Etapas Desenvolvidas para o Treinamento da Rede Neural Artificial de Cada Poço. . . . .	57
Figura 20 – Diagrama Ilustrativo das Etapas Desenvolvidas para Realizar a Sintonia dos Parâmetros-chave. . . . .	59
Figura 21 – Parâmetros-chave RGO, BSW e IP de um Poço para os casos <i>Benchmark</i> , Sintonizado e Histórico. . . . .	60
Figura 22 – Diagrama Ilustrativo da Etapa de Maximização da Produção de Petróleo. . . . .	61
Figura 23 – Gráfico da Quantidade de Cenários em Relação às Faixas de Erros (entre Produção Final Ajustada e Histórica em Relação à Produção Final <i>Benchmark</i> ). . . . .	66

Figura 24 – Gráfico da Distribuição do Erro da Produção Final Ajustada e Histórica em Relação à Produção Final <i>Benchmark</i> . . . . .	67
Figura 25 – Gráfico da Quantidade de Cenários em Relação às Faixas de Erros (entre Produção Final Ajustada e Histórica em Relação à Produção Final <i>Benchmark</i> ). . . . .	68
Figura 26 – Gráfico da Distribuição do Erro da Produção Final Ajustada e Histórica em Relação à Produção Final <i>Benchmark</i> . . . . .	68
Figura 27 – Gráfico da Quantidade de Cenários em Relação às Faixas de Erros (entre Produção Final Ajustada e Histórica em Relação à Produção Final <i>Benchmark</i> ). . . . .	69
Figura 28 – Gráfico da Distribuição do Erro da Produção Final Ajustada e Histórica em Relação à Produção Final <i>Benchmark</i> . . . . .	70

# LISTA DE TABELAS

Tabela 1 – Dados do Problema de Dieta Alimentar . . . . .	37
---	----



## LISTA DE ABREVIATURAS E SIGLAS

MILP	Mixed-Integer Linear Programming
MIQP	Mixed-Integer Quadratic Programming
DBO	Derivative-based Optimization
PWL	Piecewise-linear
RELU	Rectified Linear Unit
LP	Linear Programming
MIP	Mixed-Integer Program
PWL	Piecewise Linear
WPC	Well Performance Curve
IA	Inteligência Artificial
EDO	Equação Diferencial Ordinária



# SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>17</b>
1.1	Objetivos	19
1.1.1	Objetivo Geral	19
1.1.2	Objetivos Específicos	20
1.2	Contribuição	20
1.3	Estrutura da Dissertação	20
<b>2</b>	<b>Contextualização do Problema</b>	<b>21</b>
<b>3</b>	<b>Fundamentação Teórica</b>	<b>27</b>
3.1	<i>Surrogate Models</i>	27
3.2	Aprendizagem de Máquina	28
3.2.1	Redes Neurais	29
3.2.2	Tipos de Funções de Ativação	31
3.2.3	Arquiteturas de redes neurais	33
3.2.4	<i>Backpropagation</i>	35
3.3	Programação Matemática	36
3.3.1	<i>Linear Programming</i> (LP)	36
3.3.2	<i>Mixed-Integer Linear Programming</i>	38
3.3.3	Um modelo MILP para uma Rede Neural ReLU	39
3.4	Piecewise Linear Functions	42
3.4.1	Aproximação Unidimensional	42
3.4.2	Aproximação Multidimensional	44
3.4.3	Simulador Marlim	47
<b>4</b>	<b>Metodologia da Solução Proposta</b>	<b>49</b>
4.1	Sistema de Produção de Petróleo	49
4.2	Produção dos Poços de uma Plataforma de Petróleo	49
4.3	Plataforma de Poços de Petróleo	50
4.4	Condições de Operação	51
4.5	Limites para Parâmetros-chave dos Poços	51
4.6	Formulação Matemática do Problema de Ajuste de Parâmetros	52
4.7	Modelo Simplificado dos Poços	52
<b>5</b>	<b>Validação da Metodologia de Sintonia</b>	<b>55</b>
5.1	Detalhamento do Procedimento de Validação	55

5.2	Geração dos dados e treinamento de redes neurais . . . . .	56
5.3	Geração de cenários . . . . .	57
5.4	Sintonia dos Parâmetros-Chave . . . . .	58
5.5	Validação dos parâmetros sintonizados . . . . .	59
5.5.1	Geração de Cenários . . . . .	59
5.5.2	Otimizador da Produção de Óleo . . . . .	61
5.5.3	PWL da Função de Produção de Líquido de Cada Poço . . . . .	62
5.5.4	Otimização da Produção de Líquido de cada Poço . . . . .	63
5.5.5	Projeção dos Sinais de Controle dos Poços . . . . .	64
5.6	Experimentos . . . . .	65
5.6.1	Comparação de Cenários com Variação de 5% . . . . .	65
5.6.2	Comparação de Cenários com Variação de 10% . . . . .	67
5.6.3	Comparação Entre os Cenários Considerando Erros nas Medições . . . . .	69
<b>6</b>	<b>Considerações Finais . . . . .</b>	<b>71</b>
6.1	Sumário . . . . .	71
6.2	Trabalhos Futuros . . . . .	71
	<b>Referências . . . . .</b>	<b>73</b>

# 1 INTRODUÇÃO

Na linha de pesquisa operacional os algoritmos de inteligência artificial (IA), especialmente aqueles pertencentes à classe de aprendizado de máquina, são cada vez mais empregados em sistemas de suporte à decisão. A integração desses algoritmos de aprendizado, orientados a dados, a esses sistemas de suporte só é possível dados avanços científicos, o aumento da capacidade de processamento computacional e a viabilidade de acesso a grandes conjuntos de dados, trazendo um impacto significativo de ganhos nas operações diárias de sistemas industriais (GRIMSTAD; ANDERSSON, 2019).

A disponibilidade de dados é o que torna possível a aplicação do aprendizado de máquina na síntese de modelos representativos que descrevem fenômenos físicos não-lineares, assegurando boas propriedades, como e.g., suavidade e linearidade, para modelos de otimização matemática. Contudo, atualmente, empregar modelos construídos com aprendizado de máquina em uma estrutura de otimização matemática com resolução eficiente é um desafio na programação matemática (GRIMSTAD; ANDERSSON, 2019).

Grandes conjuntos de dados têm sido produzidos por diferentes áreas da sociedade e sobre eles são também aplicadas ferramentas de inteligência artificial, podendo trazer ganhos produtivos (WUESTA et al., 2016). Todavia, para alcançar grandes benefícios com esse imenso volume de dados são necessárias iniciativas com vistas à melhora da qualidade dos dados e para tornar o processo de produção e compreensão dos dados mais eficiente e otimizado, entre outras. (WUESTA et al., 2016).

O setor de petróleo e gás é considerado um dos mais relevantes e estratégicos da indústria mundial, uma vez que alimenta as principais matrizes energéticas, colabora para a formação de divisas através das exportações, gera empregos de alta qualificação e produtividade (IBP, 2019). A rapidez do desenvolvimento das tecnologias de *hardware* e *software* trazem um impacto direto ao setor, o que permite a flexibilização de suas operações diárias através, por exemplo, da coleta de grandes conjuntos de dados na área e do desenvolvimento de aplicações utilizando técnicas avançadas de aprendizado de máquina integradas em estruturas de otimização (CAMPONOGARA et al., 2010).

Na cadeia produtiva do petróleo muitas decisões são tomadas nas operações diárias a fim de maximizar a produção, como a escolha da configuração das válvulas, a seleção de parâmetros-chave, quais poços estarão ativos e a taxa de injeção de gas-lift, entre outras. Atualmente muitas dessas decisões estão fundamentadas em análises de sensibilidade com o uso de ferramentas de simulação e heurísticas. No entanto, mesmo que em alguns casos as análises otimizem a produção de petróleo, elas podem não assegurar um modo de operação que aumente a produtividade diária (KOSMIDIS; PERKINS; PISTIKOPOULOS, 2005). Nessas operações

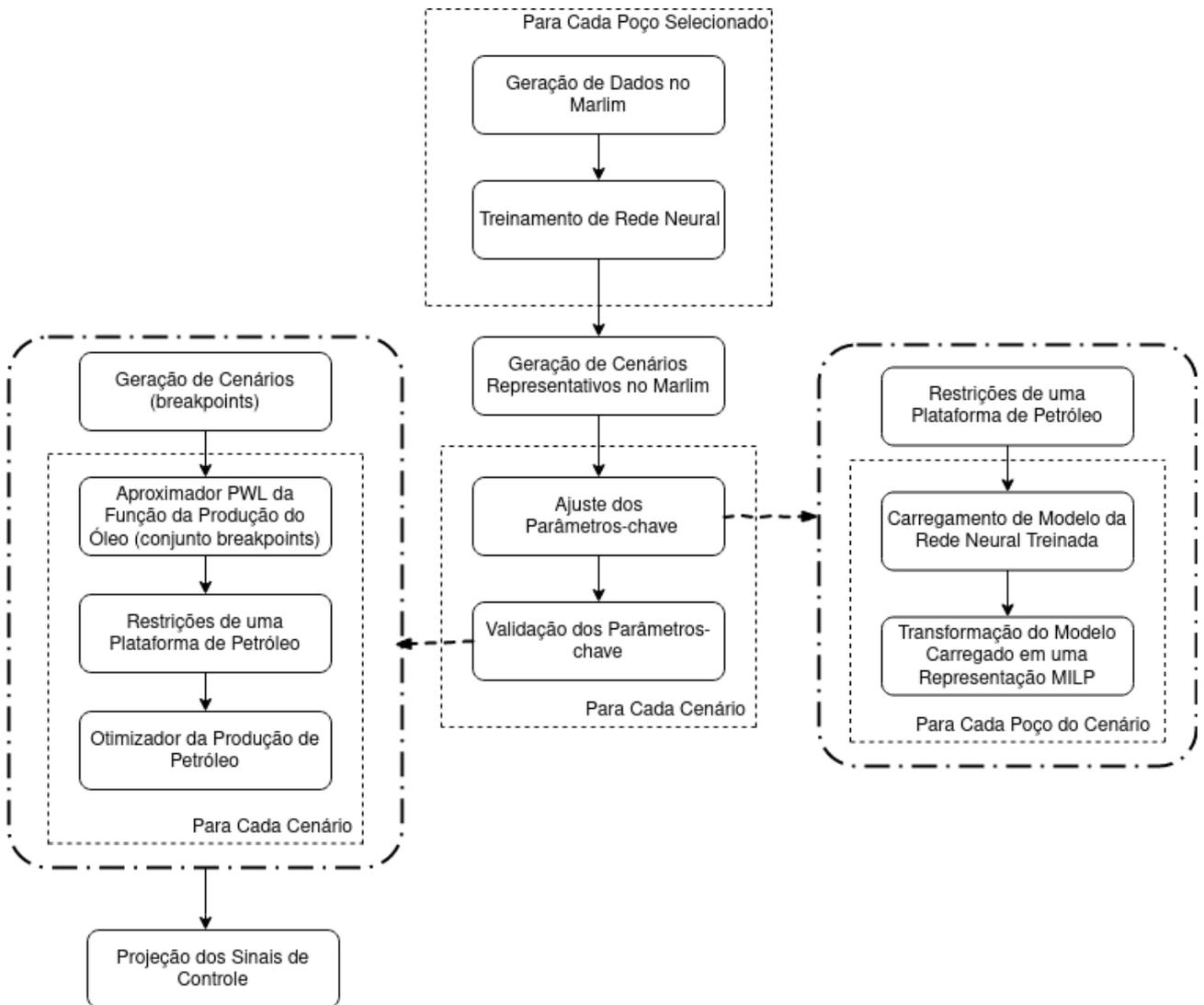
diárias algumas ferramentas computacionais dependentes de modelos estáticos representativos de poços, reservatórios e processos, são de grande utilidade na tomada de decisão pelos engenheiros responsáveis na busca das melhores condições de operação. Esses modelos possuem parâmetros-chave utilizados para sintonizá-los, mantendo-os condizentes com a realidade.

Manter os modelos de poços atualizados é uma tarefa desafiadora, pois os parâmetros-chave só são revisados quando um poço está sujeito a um teste de separação, o que ocorre apenas em certos períodos de tempo, normalmente em intervalos de um ou mais meses. Deste modo, os modelos vigentes podem não refletir as condições reais durante um período de tempo por desatualização dos parâmetros-chave, podendo acarretar perdas de produção consideráveis. Como medir a vazão de petróleo de cada poço separadamente não é possível, uma vez que os medidores de vazão multifásicos são posicionados após a mistura da produção de vários poços, Seman et al. (2021) propõe uma metodologia que auxilia na previsão da vazão de petróleo de cada poço, separadamente. Nesse contexto, será desenvolvida uma metodologia com o objetivo de sintonizar parâmetros-chave de modelos estáticos de poços encontrados em plataformas de petróleo.

Tomando os trabalhos de Grimstad and Andersson (2019) e Seman et al. (2021) como base, esta dissertação propõe uma metodologia capaz de sintonizar parâmetros-chave, tais como razão gás-óleo (RGO), razão água-óleo (BSW) e índice de produtividade (IP), de cada poço acoplado a uma plataforma de petróleo. Serão empregados os dados disponíveis da produção total de petróleo, gás e água e medições de sensores chaves instalados na plataforma, bem como modelos de simulação de poços. Um modelo de otimização será desenvolvido e resolvido a fim de sintonizar os parâmetros-chave, de forma que o total da produção dos poços se aproxime da produção total da plataforma para as fases de óleo, gás e água, respeitando as restrições críticas do sistema de produção.

Além disso, um modelo matemático voltado à maximização da produção de petróleo será utilizado para validar a produção obtida com os parâmetros-chave encontrados através da metodologia desenvolvida, verificando se a produção obtida está mais próxima da produção de referência (*benchmark*) em relação à produção histórica obtida com os parâmetros identificados no último teste de separação. A Figura 1 ilustra de maneira geral todos os passos realizados para aplicação da metodologia, bem como a sua validação.

Figura 1 – Diagrama ilustrativo do processo realizado nesse trabalho para aplicação e validação da metodologia.



Fonte: O autor

## 1.1 OBJETIVOS

Esta seção formaliza o objetivo geral e os objetivos específicos do trabalho proposto.

### 1.1.1 Objetivo Geral

Desenvolver uma metodologia para sintonizar parâmetros-chave de modelos de simulação de poços de petróleo, encontrados em plataformas de produção, considerando medições locais aos poços e integradas da plataforma, mantendo os modelos atualizados e com isso obtendo ganhos na produção diária de petróleo.

### 1.1.2 Objetivos Específicos

1. Desenvolver *surrogates models* de poços de petróleo utilizando redes neurais artificiais.
2. Desenvolver um modelo de otimização matemática para sintonizar os parâmetros-chave (RGO, BSW e IP) de poços de petróleo utilizando representações MILP de redes neurais artificiais, bem como restrições críticas da plataforma de produção.
3. Avaliar a sintonia dos parâmetros-chave em modelos estáticos, realizada por meio da metodologia proposta, em cenários representativos.

## 1.2 CONTRIBUIÇÃO

Este trabalho propõe uma contribuição de caráter tecnológico, posto que a metodologia desenvolvida é composta por modelos matemáticos, estratégias de otimização e software que permitem sintonizar parâmetros-chave, tais como RGO, BSW e IP de modelos estáticos representativos de poços encontrados em plataformas de petróleo a fim de mantê-los atualizados. Assim, podendo ser aplicada como ferramenta auxiliar para que engenheiros e operadores de plataformas reais encontrem parâmetros-chave atualizados, aumentando a produtividade da plataforma petrolífera.

## 1.3 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação é composta por 5 capítulos, sendo o primeiro a introdução da obra.

Com base no estado da arte das disciplinas envolvidas, o Capítulo 2 apresenta o contexto da atualização dos modelos estáticos de poços no âmbito da engenharia de petróleo, revisando conceitos básicos sobre o petróleo, tais como: sistemas de produção, elevação artificial de fluidos e WPC dos poços.

O Capítulo 3 apresenta uma revisão bibliográfica dos principais temas abordados na dissertação, tais como: *surrogates models*, aprendizagem de máquina, programação matemática, programação linear, programação linear inteira mista e funções lineares por partes (PWL).

O Capítulo 4 expõe a metodologia desenvolvida para sintonizar os parâmetros-chave de modelos estáticos de poços de petróleo, encontrados em campos de petróleo. Esta metodologia considera que os parâmetros-chave dos modelos são obtidos a partir de testes de separação que ocorrem em certos períodos de tempo e podem comprometer as previsões de modelos estáticos dos poços quando os parâmetros ficam desatualizados.

No Capítulo 5 são apresentadas as validações desenvolvidas para avaliar o ganho obtido com a solução proposta pela metodologia exposta. Por fim, o Capítulo 6 apresenta o sumário da dissertação, a síntese dos resultados alcançados e as perspectivas de trabalhos futuros.

## 2 CONTEXTUALIZAÇÃO DO PROBLEMA

Dentre os campos da economia que compõem a riqueza nacional do Brasil o setor de exploração e produção de petróleo é de grande importância, tanto que em 2015 apresentou um crescimento de 4,6 % em relação ao ano anterior (PETROBRAS, 2015). Estima-se que esse setor tenha um crescimento acumulado nacional de 270 % na produção entre 2011 e 2030 (BORBA et al., 2012). Um dos principais fatores para esse aumento foi a descoberta de novas reservas de petróleo provenientes do pré-sal (*offshore*), ambiente no qual a produção de petróleo ocorre em águas profundas e no alto-mar (DIEHL, 2022). No Brasil cerca de 50 % da produção de petróleo é proveniente do pré-sal (*offshore*) (PETROBRAS, 2015). O pré-sal pode ser definido como uma sequência de rochas sedimentares localizadas no fundo do oceano formada há mais de 100 milhões de anos, durante a separação do antigo continente Gondwana originando os continentes Africano e Americano (DIEHL, 2022).

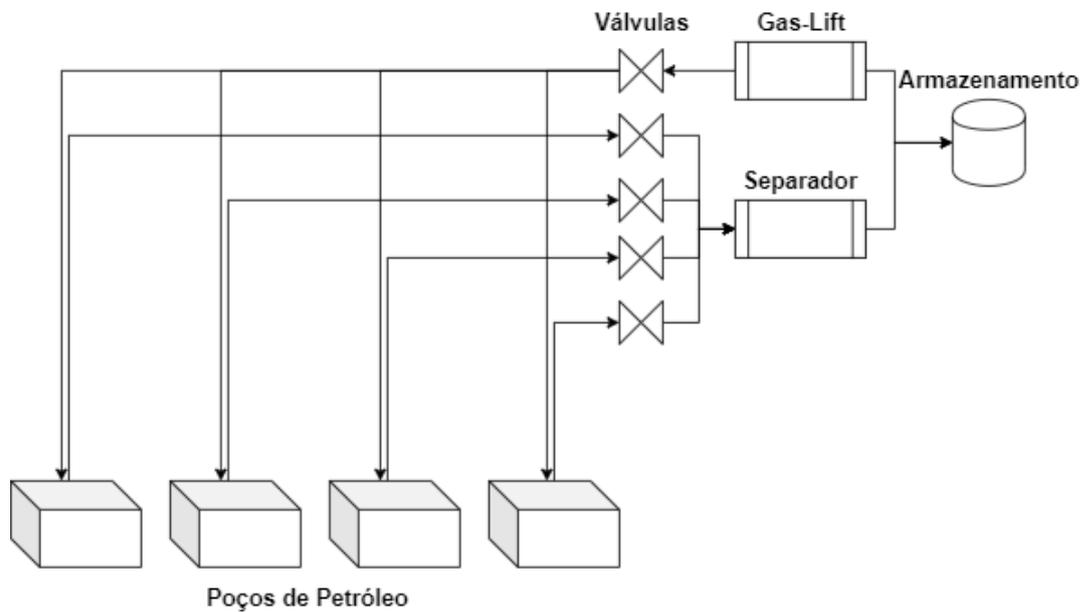
Embora os avanços tecnológicos recentes tenham ampliado as alternativas energéticas disponíveis na sociedade, o petróleo figura como uma das principais fontes de energia do mundo moderno. Ainda que o planeta apresente uma grande quantidade deste recurso ele pode ser exaurido, pois a quantidade disponível é inversamente proporcional a sua produção, por isso é classificado como uma fonte de energia não-renovável. Por consequência, quanto mais este recurso é extraído, mais difícil e onerosa se torna sua extração. Considerando a crescente dificuldade de produção e a competitividade dos mercados, a indústria do petróleo investe cada vez mais em novas tecnologias com o objetivo de otimizar o processo de produção e reduzir os custos operacionais (SILVA, 2012).

A conjuntura criada pelo aumento da demanda por combustíveis fósseis e o mercado petrolífero cada vez mais competitivo cria uma atmosfera favorável ao desenvolvimento de tecnologias avançadas aplicadas ao processo de produção de petróleo. Segundo Redutskiy (2017), é neste contexto que surge o conceito de campos inteligentes, no qual os campos de poços de petróleo passam a gerar dados, viabilizando a criação de tecnologias digitais que otimizam a produção. Atualmente, através do rápido desenvolvimento de hardwares e softwares, foram desenvolvidas tecnologias para medição digital de poços, abrindo uma gama de possibilidades para aplicação de processamento digital de dados em campos de produção. Assim, os campos inteligentes apresentam uma operação cada vez mais integrada e flexível da produção diária de campos de petróleo e gás, minimizando os custos operacionais diários e elevando o fator de recuperação dos reservatórios (CAMPONOGARA et al., 2010).

Um sistema típico de produção encontrado em campos de petróleo e gás é ilustrado conforme a Figura 2.

Tipicamente, os sistemas de produção de petróleo são compostos por poços, *manifolds*

Figura 2 – Diagrama ilustrativo da estrutura de um sistema de produção de petróleo que opera com injeção contínua de gás comprimido (*gas-lift*).



Fonte: (SILVA, 2012)

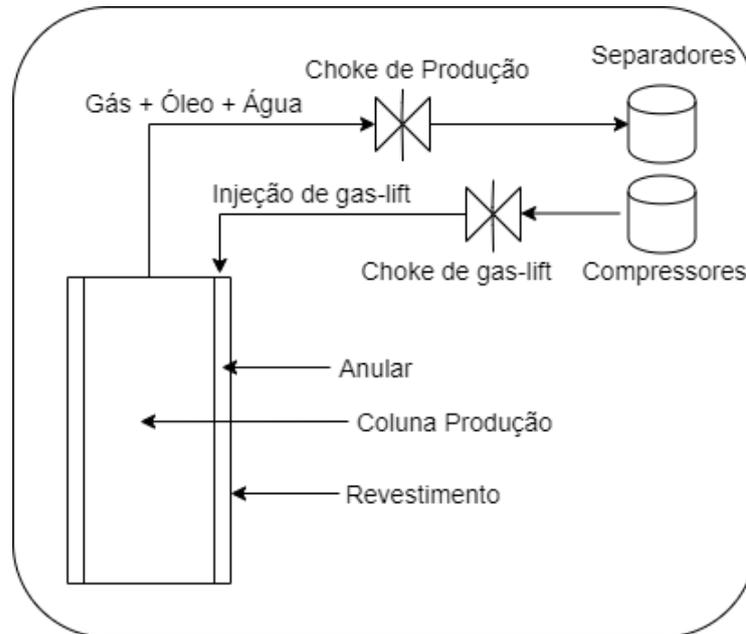
(concentradores), separadores, compressores, tubulações que conectam os componentes e válvulas, entre outros. Segundo Khan et al. (2020) o *gas-lift* é uma das tecnologias mais empregadas para a elevação artificial de fluidos em poços de petróleo. Há uma quantidade ideal de *gas-lift* a ser injetada em cada poço, de forma que o máximo de petróleo seja coletado, o que demonstra a importância de tecnologias para a otimização da produção (SILVA, 2012).

À medida que o petróleo é extraído dos poços, a pressão interna do reservatório decresce, fazendo com que o reservatório não tenha pressão interna suficiente para superar o peso da coluna de óleo na tubulação, em decorrência disso surge a necessidade de elevação artificial para elevar fluidos até à superfície. Há alguns métodos de elevação artificial, tais como injeção de *gas-lift*, bombas elétricas submersas, injeção de água, entre outros. O termo *gas-lift* consiste da injeção de gás pressurizado no fundo da coluna de produção do poço de petróleo, fazendo com que os fluidos, água, óleo e gás, do poço sejam elevados até a superfície (CAMPONOGARA; NAKASHIMA, 2006). Por ser um método robusto e de custo operacional relativamente baixo, a injeção de gás pressurizado (*gas-lift*) é o mais escolhido nas plataformas.

A Figura 3 ilustra o processo simplificado do método de elevação artificial utilizando *gas-lift*, no qual o gás é injetado no espaço anular, formado entre o revestimento e a coluna de produção.

O controle de injeção de gás pressurizado é realizado por uma válvula na superfície, denominada *choke* de injeção de gás. O gás pressurizado do compressor chega ao *anular* do poço e é injetado através da válvula de *gas-lift*, com isso os fluidos, água, óleo e gás, são escoados pela coluna de produção (SILVA, 2012).

Figura 3 – Diagrama simplificado de um processo de elevação artificial de petróleo em poços.



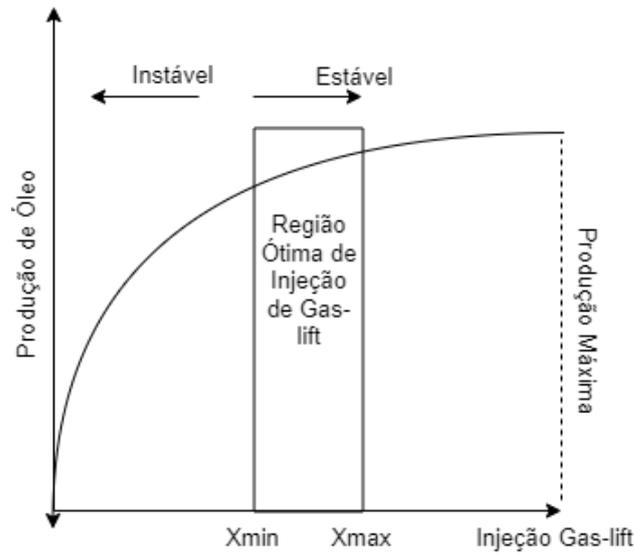
Fonte: (SILVA, 2012).

Uma prática comumente utilizada nas plataformas de petróleo operadas com elevação artificial, sobretudo com injeção de *gas-lift*, é a avaliação das *Well Performance Curves* (WPC). Através desta curva é possível determinar a taxa de injeção de *gas-lift* necessária para obtenção da produção de óleo desejada. Utilizando simulações numéricas e métodos matemáticos apropriados é possível construir as WPCs. Outra forma de determinar a taxa de injeção de *gas-lift* que vem sendo utilizada é por meio dos dados coletados de taxa de *gas-lift* injetado e da produção de óleo em campos inteligentes, possibilitando ajustar uma WPC por meio de uma função polinomial quadrática, bem como, por exemplo, o método dos mínimos quadrados (SAEPUDIN et al., 2007).

A Figura 4 ilustra uma WPC típica de um poço de petróleo. Neste gráfico é possível notar que há uma região considerada ótima para injeção de *gas-lift*, uma vez que existem alguns limites físicos e operacionais, tais como a capacidade de compressão dos compressores, no contexto de um cenário com vários poços produtores e injetores (SILVA, 2012).

Algumas decisões precisam ser tomadas durante as operações diárias dos campos de petróleo para se atingir a produção máxima. Essas decisões dependem de modelos para poços, processos e reservatórios. Dentre essas decisões, cita-se: Quais poços estarão ativos? Qual deve ser a configuração das válvulas? Para qual separador a produção de um poço deve ser alinhada? Qual é taxa de injeção de *gas-lift* ótima de um poço? Essas e outras decisões têm que respeitar as restrições operacionais dos campos, plataforma, tais como: a capacidade de processamento nas instalações da superfície, a disponibilidade de *gas-lift* e a pressão nas tubulações, entre outras (KOSMIDIS; PERKINS; PISTIKOPOULOS, 2005).

Figura 4 – Curva de Desempenho de Poço (WPC).



Fonte: O autor.

A geração de tecnologias de software auxilia diariamente os engenheiros e operadores na tomada de decisão por melhores condições de operação para sistemas de produção de petróleo. O avanço da tecnologia de campos inteligentes é possível graças ao acesso a grandes conjuntos de dados, computadores com maior poder computacional, algoritmos com capacidade de aprendizado, metodologias de otimização em programação matemática, entre outros. Contudo, um desafio encontrado na área está na integração de modelos construídos a partir dos dados disponíveis e inteligência artificial em uma estrutura de otimização completa e na resolução do problema resultante de maneira eficiente (GRIMSTAD; ANDERSSON, 2019).

Alguns trabalhos encontrados na literatura, como Gunnerud and Foss (2010), Hoffmann and Stanko (2017) e Grimstad et al. (2016), criam e empregam modelos que representam condições prevalentes do sistema de produção de petróleo. Entretanto, manter esses modelos atualizados é uma tarefa desafiadora, uma vez que os processos variam no tempo, são de natureza complexa e os principais parâmetros de um modelo de simulação de um sistema de produção de petróleo, tais como RGO, BSW e IP, são revisados apenas quando um poço está sujeito a um teste de separação, que por sua vez é realizado apenas de tempos em tempos. Vale ressaltar que, o emprego de parâmetros desatualizados faz com que o modelo de simulação não reflita as condições atuais do sistema de produção (SEMAN et al., 2021), criando condições para perdas produtivas e econômicas significativas.

Nesta dissertação será desenvolvida uma metodologia capaz de atualizar os parâmetros RGO, BSW e IP associados a um modelo estático de poços encontrados em campos reais de petróleo. Essa estratégia de atualização de modelos é importante para que sejam refletidas as condições reais de operação, permitindo que se estime a produção de cada poço separadamente,

uma vez que nas plataformas de petróleo os medidores de vazão multifásica são frequentemente localizados depois do ponto de mistura da produção dos poços.

A metodologia utilizada para o desenvolvimento da tecnologia de sintonia de parâmetros-chave estima a produção de cada poço pertencente a um separador em comum, empregando os dados disponíveis da produção total dos poços, os dados simulados e modelos de simulação de poços estáticos. Seguindo as estratégias de Chachuat, Srinivasan and Bonvin (2009), Vielma (2015) e Seman et al. (2021), um modelo de otimização é resolvido para sintonizar os parâmetros RGO, BSW e IP, de modo que a produção total da plataforma utilizando os parâmetros estimados, esteja mais próxima da produção de referência, em relação a produção encontrada com parâmetros do último teste de separação.

A sintonia dos parâmetros de modelos é importante para os processos de otimização, uma vez que são gerados ganhos diários na produção de petróleo. Sem um modelo sintonizado corretamente as metodologias de otimização podem gerar soluções subótimas e acarretar em perdas na produção. Ainda que historicamente, do ponto de vista prático, as técnicas de suporte à decisão não rendam altas porcentagens de ganhos, no setor petrolífero por se tratar de uma área de impactos expressivos, essa pequena porcentagem de ganhos se faz significativa tanto do ponto de vista econômico.



## 3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, os principais assuntos relacionados ao projeto são apresentados. Na Seção 3.1 são apresentados os *surrogate models*. Na Seção 3.2 uma breve introdução ao aprendizado de máquina e às redes neurais, a serem utilizadas para construir os *surrogate models*. Na seção 3.3 é feita uma breve introdução a programação matemática, sobretudo a formulação matemática de uma rede neural RELU utilizando um *mixed-integer linear programming* (MILP).

### 3.1 SURROGATE MODELS

Neste capítulo, os *surrogate models* são apresentados em referência ao trabalho de Grimstad and Andersson (2019), trazendo uma visão geral com base em trabalhos relevantes na literatura.

A criação de modelos de funções matemáticas com estrutura desconhecida, bem como redução do tempo da solução, quando funções complexas são substituídas por *surrogate models* com melhores propriedades para otimização, são as principais características que tornam importante o estudo de *surrogate models* na programação matemática (GRIMSTAD; ANDERSSON, 2019).

A construção de modelos complexos, que descrevem fenômenos físicos não-lineares, é possível devido ao acesso a grandes conjuntos de dados e maior poder computacional disponível. Portanto, uma grande quantidade de aplicações de suporte a decisões orientadas a dados, estudados em pesquisa operacional e programação matemática, têm sido desenvolvidas (GRIMSTAD; ANDERSSON, 2019).

As funções que são acessíveis via amostras, são conhecidas como “*black-box*”. Os problemas que são conhecidos como “*grey-box*”, são aqueles que possuem uma mistura de funções “*black-box*” e funções explicitamente conhecidas (funções algébricas, por exemplo). Para a otimização de problemas “*grey-box*” uma abordagem promissora é aproximar e substituir uma função “*black-box*” por um *surrogate model*. Assim, o problema aproximado pode ser resolvido eficientemente por métodos de otimização baseados em derivadas. Essa abordagem é conhecida como DBO (*derivative-based optimization*) (GRIMSTAD; ANDERSSON, 2019).

A análise de processos e otimização poder ser feita por meio de *surrogate models*, que são construídos com funções clássicas de aproximação, tais como: *radial basis function*, *support vector machine*, *tensor-product splines* e *artificial neural networks* (redes neurais). Esses modelos, geralmente, são construídos para serem contínuos e suaves com o intuito de facilitar a otimização de solvers não-lineares (otimização local e global). Os modelos *piecewise-linear* (PWL) pertencem a uma classe importante de *surrogate models*, os quais podem ser diretamente

empregados em formulações MILP. Existem diversos modelos PWL de uma ou mais variáveis, no entanto essas formulações precisam que as amostras sejam interpoladas, o que exige a adição de restrições de interpolação e restrições da estrutura das amostras no problema. Isso faz com que a aplicabilidade do PWL seja limitada para funções de baixa dimensionalidade e para conjuntos de dados relativamente pequenos (GRIMSTAD; ANDERSSON, 2019).

Geralmente métodos de regressão são mais adequados do que métodos de interpolação para a criação de *surrogate models*, a partir de um conjunto de dados (*datasets*) volumoso e ruidoso. Com uma formulação matemática adequada, os modelos de regressão podem ser integrados com um *framework* MILP para otimização. Diversos trabalhos têm sido publicados demonstrando como incorporar *surrogate models*, principalmente modelos gerados com ferramentas de aprendizado de máquina, em modelos de otimização (MARTINEZ et al., 2017), (FAHMI; CREMASCHI, 2012), (MISIC, 2019).

## 3.2 APRENDIZAGEM DE MÁQUINA

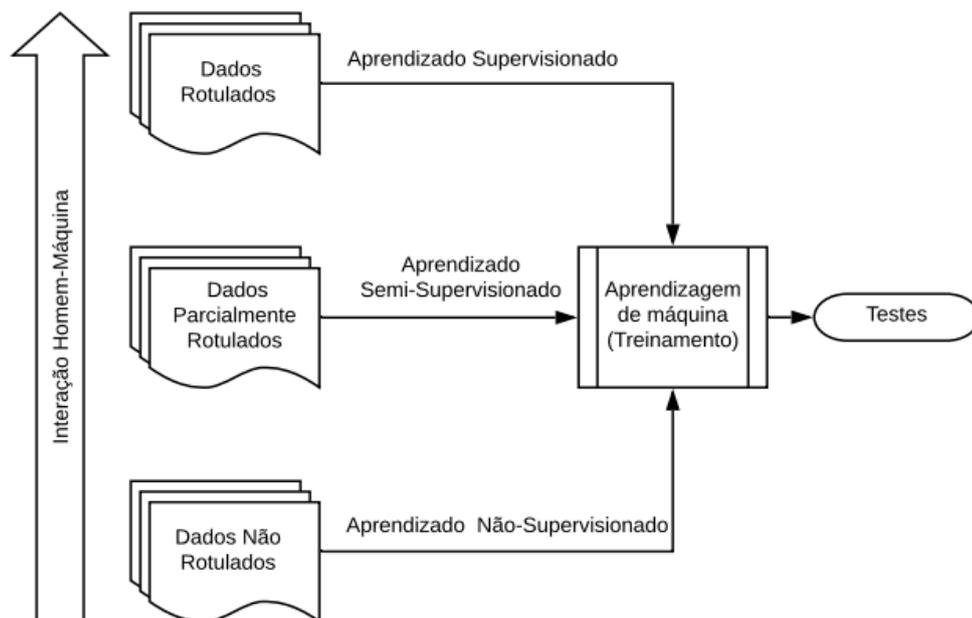
Algoritmos que não precisam ser literalmente programados são conhecidos como “*hard coded*”, como por exemplo, os algoritmos de aprendizado de máquina. Um algoritmo de *machine learning* (aprendizado de máquina) é um processo computacional que utiliza dados de entrada para alcançar uma tarefa desejada, sem que essa tarefa seja completamente programada. Por meio de repetições (experiência) estes algoritmos alteram ou adaptam automaticamente sua arquitetura de forma que possam alcançar o resultado esperado cada vez (repetição) melhor. Portanto, estes algoritmos podem aprender, a partir dos dados de entrada e com isso generalizar para produzir o resultado desejado a partir de novos dados, nunca antes vistos (NAQA; MURPHY, 2015).

Algoritmos computacionais podem adaptar sua própria arquitetura, por meio do processo de treinamento, de algumas maneiras, como por exemplo: Selecionar e ponderar os dados de entrada; Por meio da otimização iterativa alterar parâmetros numéricos variáveis; Determinar distribuições probabilísticas a partir dos dados de entrada; entre outros (NAQA; MURPHY, 2015).

Os algoritmos de aprendizado de máquina podem ser classificados em 3 classes de aprendizado, são elas: *supervised learning*, *unsupervised learning* e *semi-supervised learning*. A Figura 5 (NAQA; MURPHY, 2015) ilustra o quanto cada classe de aprendizado precisa da interação com humanos no processo de aprendizado. A classe de algoritmos *supervised learning* se baseia como um ser humano age, ou seja, a partir de dados entrada e saída, bem como a experiência (repetições), o algoritmo mapeia um conjunto de entradas para uma saída. Por exemplo, em cada amostra de entrada (cor, tamanho, largura) é mapeada uma saída (laranja, uva, pêra). Os algoritmos da classe *unsupervised learning*, desenvolvem seu próprio caminho a partir de dados de treinamento, ou seja, a partir de dados não rotulados. Por exemplo, um algoritmo para tentar lançar um dardo em um alvo. Existe uma variedade de graus de liberdade

no mecanismo que controla o caminho do dardo. Porém, ao invés de tentar programar todas as possibilidades, várias tentativas são realizadas e para cada tentativa, os graus de liberdade são ajustados para que o dardo fique cada vez mais próximo do alvo. Nota-se que esse aprendizado não é supervisionado, uma vez que não há um mapeamento entre entrada e saída. Já os algoritmos *semi-supervised learning* utilizam das duas abordagens, ou seja parte das entradas são mapeadas para uma saída e a outra parte é aprendida a cada interação (NAQA; MURPHY, 2015).

Figura 5 – Uma estrutura hierárquica típica da cadeia aprendizado de máquina (*machine learning*).



Fonte: (NAQA; MURPHY, 2015).

O processo de aprendizagem que busca controlar o aprendizado, acomodando um sistema de feedback é o *reinforcement learning*, que por sua vez é uma classe bem importante de algoritmos de aprendizagem de máquina (NAQA; MURPHY, 2015).

### 3.2.1 Redes Neurais

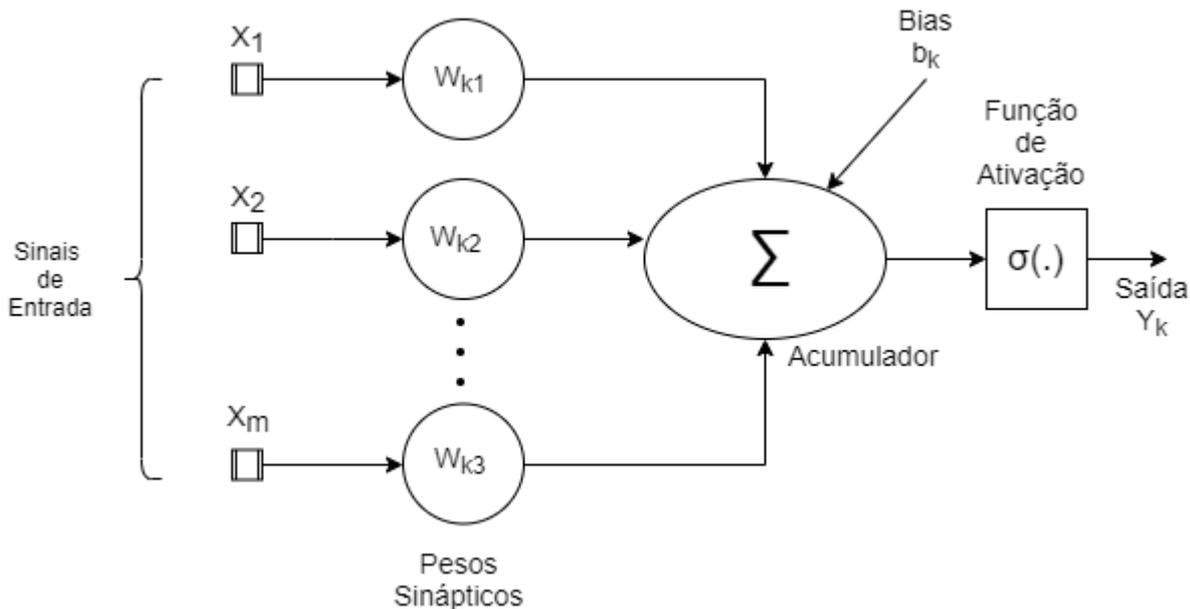
Os algoritmos de redes neurais artificiais pertencem à classe de aprendizado de máquina. Essas redes neurais artificiais são inspiradas nas redes neurais biológicas (cérebro). O cérebro humano é um computador altamente complexo, não-linear e paralelo, que por sua vez tem a capacidade de organizar seus constituintes estruturais, conhecidos por neurônios, de forma a realizar certos processamentos, tais como: reconhecimento de padrões, classificação, percepção, entre outros, muito mais rápido que um computador digital (HAYKIN, 2008).

Uma rede neural pode ser classificada, de maneira geral, como uma máquina projetada para modelar a maneira que o cérebro realiza uma tarefa particular. Uma interligação entre células computacionais simples são denominadas como sinapse. O algoritmo de aprendizagem

de uma rede neural tem como função modificar os pesos sinápticos de forma ordenada para alcançar o aprendizado (HAYKIN, 2008).

Os neurônios são unidades de processamento de informação, que são fundamentais para a operação da rede neural como um todo. O modelo de um único neurônio é apresentado conforme a Figura 6.

Figura 6 – Modelo de um único neurônio de uma rede neural artificial.



Fonte: (HAYKIN, 2008)

Um neurônio recebe algumas conexões (*sinapses*), cada uma caracterizada por um peso ( $W_{kn}$ ). Por exemplo, o modelo do neurônio da Figura 6 recebe os pesos  $W_{k1}, W_{k2}, \dots, W_{km}$ , que por sua vez conecta-se a camada anterior de neurônios com a camada posterior. Um somador (acumulador) é responsável por somar os sinais de entrada, ponderados pelos pesos de cada *sinapse*. Uma função de ativação é utilizada como limitador da saída de um neurônio e há várias funções disponíveis, no qual cada problema possui sua necessidade (HAYKIN, 2008). Podemos descrever um neurônio  $k$  matematicamente através da seguinte equação:

$$U_k = \sum_{j=1} W_{kj} X_j \quad (3.1)$$

$$Y_k = \phi(U_k + B_k) \quad (3.2)$$

onde  $X_1, X_2, X_3, \dots, X_j$  são os valores de entrada;  $W_{k1}, W_{k2}, \dots, W_{kj}$  são os pesos sinápticos do neurônio  $k$ ;  $U_k$  é a saída do somador devido aos sinais de entrada;  $B_k$  é o bias;  $\phi(\cdot)$  é a função de ativação; e  $Y_k$  é a saída do neurônio.

### 3.2.2 Tipos de Funções de Ativação

Um dos principais componentes de uma rede neural são as entradas, que são processadas e transformadas em um resultado conhecido como ativação. Para se obter a ativação de um neurônio, a rede aplica uma função de ativação, que é responsável em escalar as entradas (SHARMA; SHARMA, 2020). Algumas das funções de ativação encontradas na literatura são: Função de etapa binária; Linear; Sigmoid, Tanh, ReLU, Leaky Relu e SoftMax, entre outras. As próximas seções demonstram algumas das funções de ativação.

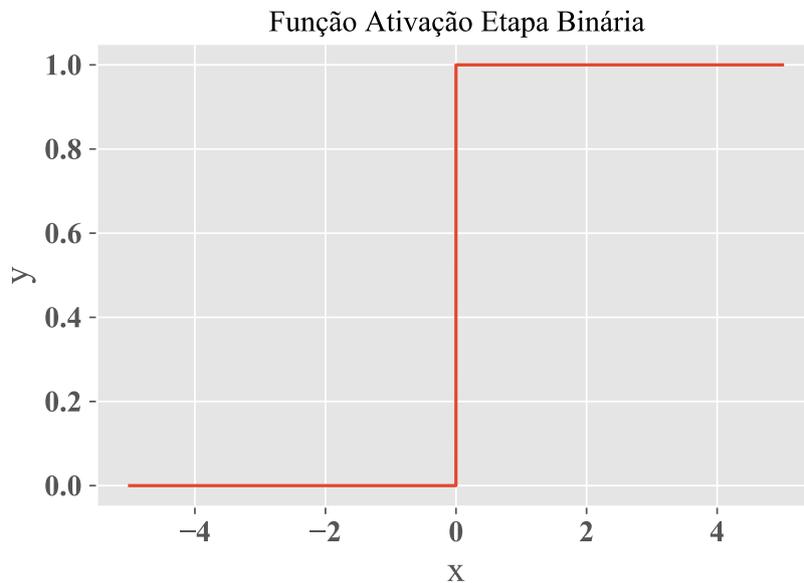
#### Função de Etapa Binária

Essa função pode ser implementada utilizando um *'if-then'* em qualquer linguagem de programação, portanto é considerada uma das mais simples entre as funções de ativação. Geralmente, essa função é criada para o desenvolvimento de um classificador binário (SHARMA; SHARMA, 2020). A sua representação matemática é apresentada conforme a equação abaixo:

$$f(x) = \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases} \quad (3.3)$$

A função de ativação etapa binária é ilustrada conforme a Figura 7.

Figura 7 – Função de Ativação Etapa Binária.



Fonte: O autor.

#### Função Linear

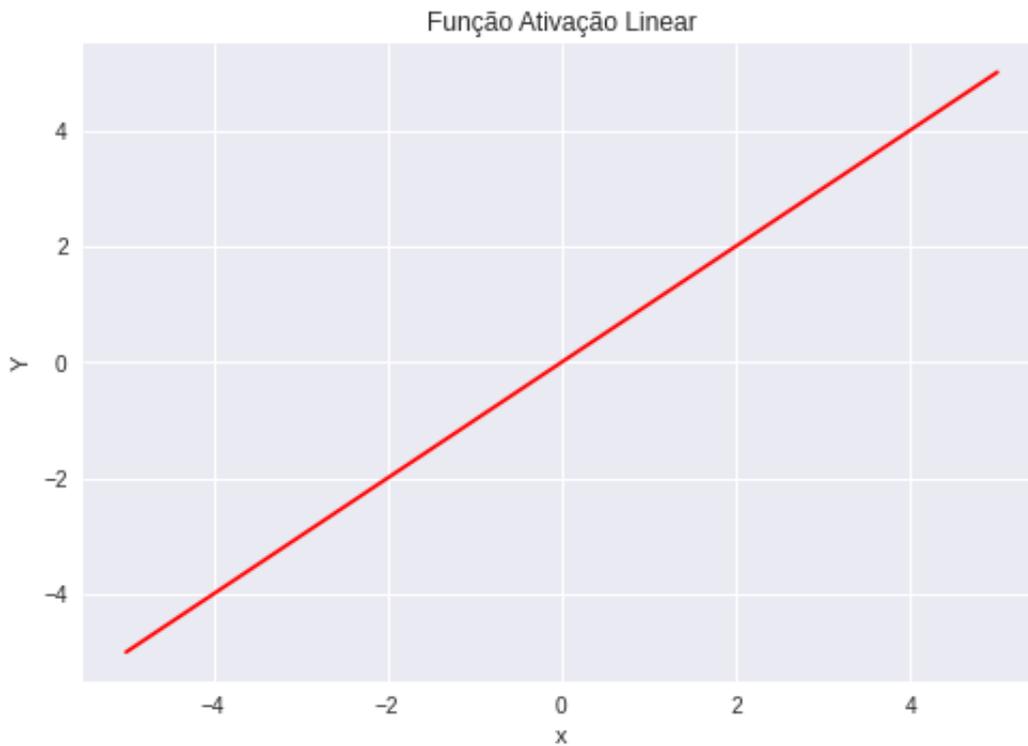
Essa função é diretamente proporcional aos valores de entrada. A principal diferença entre a função anterior é que o gradiente da função não é zero (SHARMA; SHARMA, 2020). A

representação matemática dessa função é apresentada conforme a equação abaixo.

$$f(x) = ax \quad (3.4)$$

A Figura 8 ilustra a função de ativação linear.

Figura 8 – Função de Ativação Linear.



Fonte: O autor.

### Função ReLU (*Rectified Linear Unit*)

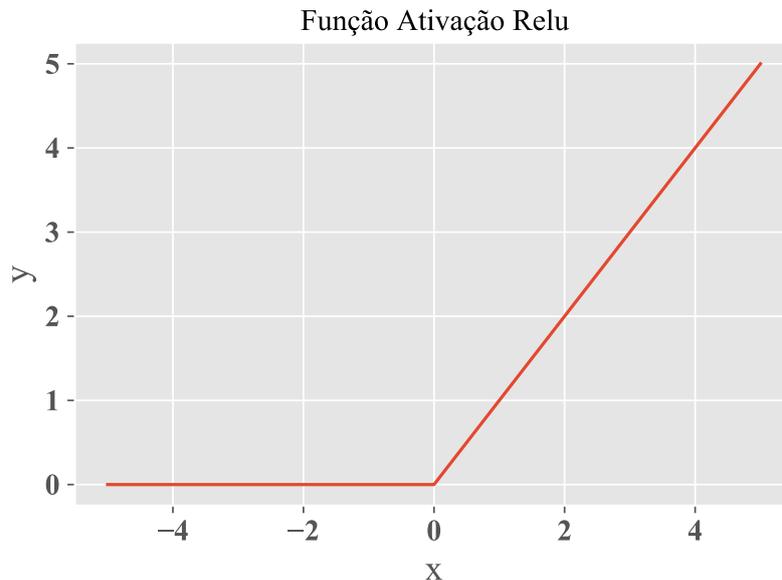
Essa função de ativação é amplamente utilizada em projetos de redes neurais artificiais. Uma das vantagens da utilização da função ReLU é que todos os neurônios não são ativados ao mesmo tempo, o que deixa a rede neural mais eficiente. O neurônio será desativado apenas quando a saída da transformação linear é zero. A equação que rege a função de ativação matematicamente é dada a seguir (SHARMA; SHARMA, 2020)

$$f(x) = \max(0, x) \quad (3.5)$$

A Figura 9 ilustra uma função de ativação ReLU. Como todos os neurônios não são ativados ao mesmo tempo, ou seja são ativados em remessas, a função de ativação ReLU é mais eficiente do que as outras funções.

O valor do gradiente do erro pode ser zero em alguns casos, uma vez que os pesos e bias não são atualizados durante a etapa de "*backpropagation*" no processo de treinamento da rede neural (SHARMA; SHARMA, 2020).

Figura 9 – Função de Ativação ReLU.



Fonte: O autor.

### 3.2.3 Arquiteturas de redes neurais

O algoritmo de aprendizado utilizado para o treinamento da rede neural é um fator determinante para a definição da estrutura dos neurônios da rede. Em geral, podemos encontrar duas arquiteturas de redes neurais fundamentalmente iguais.

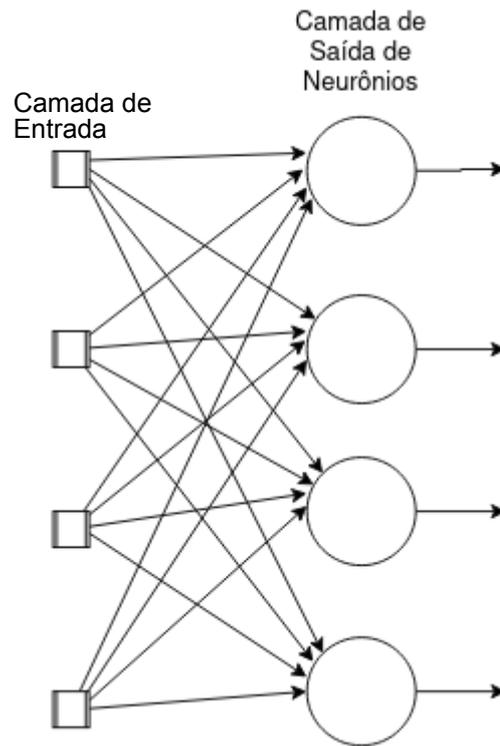
#### Redes Neurais Alimentadas Adiante com Camada Única

Em uma rede neural em camadas, na sua forma mais simples, temos uma camada de entrada de nós (neurônios) que se interliga em uma camada de saída de neurônios. Em outras palavras, esta rede é classificada como alimentada adiante (HAYKIN, 2008). Na Figura 10 é ilustrada uma rede com essa arquitetura, com quatro nós tanto na camada de entrada como na camada saída.

#### Redes Neurais Alimentadas Adiante com Múltiplas Camadas

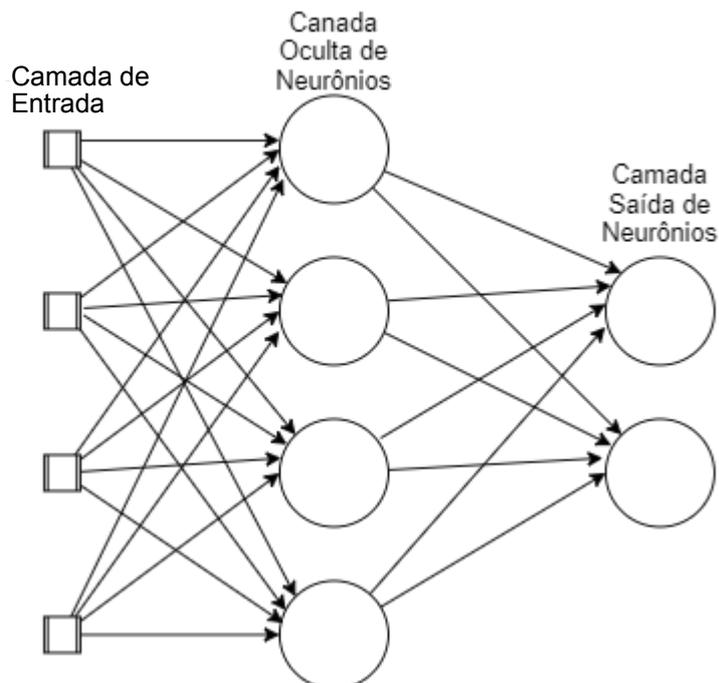
Esse tipo de rede neural difere da anterior pela presença de uma ou mais camadas ocultas. A função dos neurônios das camadas ocultas é intervir entre a camada de entrada externa e a de saída. A cada camada oculta adicionada à rede neural é mais adequada para extrair características de ordem elevada. Essa habilidade é particularmente valiosa quando o tamanho da camada de entrada é grande (HAYKIN, 2008). A Figura 11 ilustra uma rede neural com esse tipo de arquitetura.

Figura 10 – Arquitetura de uma Rede Neural Alimentada Adiante.



Fonte: (HAYKIN, 2008).

Figura 11 – Arquitetura de uma Rede Neural com Múltiplas Camadas.



Fonte: (HAYKIN, 2008).

### 3.2.4 Backpropagation

Em 1970 foi criado o algoritmo *backpropagation*, no qual a sua importância despertou com o lançamento do artigo de David Rumelhart, Geoffrey Hinton e Ronald Williams (RUMELHART; HINTON; WILLIAMS, 1986). Nesse artigo, são apresentadas várias redes neurais em que o algoritmo *backpropagation* é aplicado, bem como é comparada a eficiência dele em relação a outras abordagens de aprendizado. Portanto, com esse algoritmo é possível criar redes neurais capazes de resolver problemas que eram insolucionáveis. Em projetos de redes neurais, esse método *backpropagation* é a peça fundamental para que o aprendizado ocorra (GOODFELLOW; BENGIO; COURVILLE, 2016).

O componente principal de um algoritmo *backpropagation* é a expressão da derivada parcial da função custo de treinamento  $c(w, b)$ , em relação ao peso  $w$  ou o bias  $b$  (GOODFELLOW; BENGIO; COURVILLE, 2016). A expressão é apresentada conforme a equação abaixo.

$$\frac{\partial c}{\partial w} \quad (3.6)$$

Durante o processamento, o algoritmo *backpropagation* calcula a derivada parcial do custo em relação ao peso e ao bias ( $b$ ), com o objetivo de identificar uma direção de minimização do custo (GOODFELLOW; BENGIO; COURVILLE, 2016). A função do custo quadrático ( $c$ ) é apresentada conforme a equação abaixo:

$$c = \frac{1}{2n} \sum_{x \in \mathcal{X}} \|y(x) - a^L(x)\|^2 \quad (3.7)$$

onde  $\mathcal{X} = \{x^1, x^2, \dots, x^n\}$  é o conjunto de amostras de exemplos;  $n$  é o número total de amostras;  $y = y(x)$  é a saída desejada;  $L$  é o número de camadas da rede;  $a^L(x)$  é o vetor da saída da função de ativação quando  $x$  é inserido. O custo da função, representado por  $c$ , pode ser expresso como a média apresentada abaixo:

$$c = \frac{1}{n} \sum_x c_x \quad (3.8)$$

onde o custo de uma única amostra de treinamento é dado por:

$$c_x = \frac{1}{2} \|y - a^L(x)\|^2 \quad (3.9)$$

As equações acima são assumidas devido o algoritmo *backpropagation* computar a derivada parcial  $\frac{\partial c_x}{\partial w}$  e  $\frac{\partial c_x}{\partial b}$  de uma única amostra de exemplo. Então é possível calcular  $\frac{\partial c}{\partial w}$  e  $\frac{\partial c}{\partial b}$  calculando a média de exemplos de treinamento (GOODFELLOW; BENGIO; COURVILLE, 2016).

Além disso, assume-se que o custo pode ser escrito em função da saída de uma rede neural:

$$c_x = \frac{1}{2} \|y - a^L(x)\|^2 = \frac{1}{2} \sum_{j=1}^{n_j^L} [y_j - a_j^L(x)]^2 \quad (3.10)$$

onde  $n_j^l$  é o número de saídas da camada  $l$  e, portanto,  $n_j^L$  corresponde ao número de saídas da última camada da rede neural.

### 3.3 PROGRAMAÇÃO MATEMÁTICA

Os modelos matemáticos empregados para a resolução de problemas de decisão podem ser construídos utilizando-se técnicas da programação matemática. A abordagem utilizada nesta programação é declarativa, ou seja, representa-se um problema a partir de uma formulação matemática composta por restrições e uma função objetivo (CASTRIGNANO et al., 2020). Várias classes de modelos de programação matemática são encontradas na literatura, porém neste trabalho utilizaremos modelos *mixed-integer linear programming* (MILP) e *mixed-integer quadratic programming* (MIQP) que generalizam a classe de *linear programming* (LP) ao admitir variáveis discretas de decisão.

#### 3.3.1 Linear Programming (LP)

Um modelo de otimização matemática em que uma função objetivo é declarada para ser minimizada ou maximizada, sujeita a um conjunto finito de restrições lineares (equações e/ou inequações lineares) é considerado um modelo de programação linear (LP). Existem diversas aplicações na programação matemática em que os modelos LP são utilizados para resolver problemas reais, tais como: agendamento de pousos de companhias aéreas (BEASLEY et al., 2000) e otimização do abastecimento integrado na indústria de petróleo (CATCHPOLE, 2017), entre outros. Esses problemas costumam ter centenas de variáveis e restrições.

A definição de um modelo LP é realizada por meio de funções, equações e inequações lineares. Uma função, equação e inequação linear pode ser representada conforme as equações abaixo (COTTLE; THAPA, 2017).

$$f(x_1, x_2, \dots, x_n) = a_1x_1 + a_2x_2 + \dots + a_nx_n \quad (3.11a)$$

$$b_1x_1 + b_2x_2 + \dots + b_nx_n = b_0 \quad (3.11b)$$

$$d_1x_1 + d_2x_2 + \dots + d_nx_n \leq d_0 \quad (3.11c)$$

onde as variáveis são representadas por  $x_1, x_2, \dots, x_n$  e as constantes por  $a_i, b_i, c_i, i = 0, \dots, n$ . Diferentes exemplos são encontrados na matemática de funções, equações e inequações lineares, desde um simples somatório até a representação de um fenômeno físico complexo envolvendo equações diferenciais ordinárias (EDO) (COTTLE; THAPA, 2017).

As restrições representadas por inequações lineares, de um modelo LP, devem ser respeitadas simultaneamente, no qual tem várias soluções viáveis (*feasible solutions*), portanto utilizando uma equação linear, cria-se um objetivo que é responsável por selecionar a melhor solução viável, ou seja, encontrar o máximo ou mínimo em um conjunto de soluções viáveis.

Uma forma padrão de representar um problema LP é apresentado de acordo com as equações demonstradas abaixo (COTTLE; THAPA, 2017)

$$\text{minimize } c_1x_1 + \cdots + c_nx_n \quad (3.12a)$$

$$\text{subject to } a_{11}x_1 + \cdots + a_{1n}x_n = b_1 \quad (3.12b)$$

$$a_{21}x_1 + \cdots + a_{2n}x_n = b_2 \quad (3.12c)$$

$$\vdots \quad (3.12d)$$

$$a_{m1}x_1 + \cdots + a_{mn}x_n = b_m \quad (3.12e)$$

$$x_j \geq 0, j = 1, \dots, n \quad (3.12f)$$

Nem todos os modelos LP são declarados conforme a forma padrão apresentada acima, podendo apresentar uma mistura de equações e inequações lineares, no entanto é trivial convertê-las para o modelo padrão (COTTLE; THAPA, 2017).

## Modelo Clássico do Problema da Dieta

O problema clássico da dieta na programação linear pode ser definido como: encontrar uma maneira menos dispendiosa de alcançar um determinado conjunto de metas nutricionais diárias, utilizando um conjunto determinado de alimentos. Um exemplo de um problema pequeno que pode ser modelado conforme o LP declarado abaixo:

- Minimizar o custo para alcançar pelo menos 1,0 mg de tiamina, 7,50 mg de fósforo e 10,00 mg de ferro, de três alimentos nomeados como  $a$ ,  $b$  e  $c$ . O custo do alimento  $a$  é 10 centavos *per ounce* (oz), o custo de  $b$  é 8 centavos *per ounce* e o custo da  $c$  é 12 centavos *per ounce*. A Tabela 1 sintetiza o problema anunciado (COTTLE; THAPA, 2017).

Tabela 1 – Dados do Problema de Dieta Alimentar

Elemento Químico	Alimentos		
	$a$ (mg/oz)	$b$ (mg/oz)	$c$ (mg/oz)
Tiamina	0,15	0,10	0,20
Fósforo	0,75	1,70	1,90
Ferro	1,30	1,10	1,40

Considerando que  $x_a$ ,  $x_b$  e  $x_c$  são variáveis não-negativos, com as quantidades dos alimentos  $a$ ,  $b$  e  $c$  respectivamente, o problema da dieta pode ser descrito como um modelo LP na sua forma padrão conforme as equações abaixo (COTTLE; THAPA, 2017).

$$\text{minimize } 10x_a + 8x_b + 12x_c \quad (3.13a)$$

$$\text{subject to : } 0,15x_a + 0,10x_b + 0,20x_c \geq 1,0 \quad (3.13b)$$

$$0,75 x_a + 1,70 x_b + 1,90 x_c \geq 7,50 \quad (3.13c)$$

$$1,30 x_a + 1,10 x_b + 1,40 x_c \geq 10,0 \quad (3.13d)$$

$$x_a, x_b, x_c \geq 0 \quad (3.13e)$$

A solução do problema acima e dos demais problemas encontrados na programação linear podem ser obtidas por meio de algoritmos. Dentre os algoritmos disponíveis na literatura, o algoritmo Simplex de George B. Dantzig se destaca como o algoritmo mais importante encontrado na programação linear (COTTLE; THAPA, 2017).

### 3.3.2 *Mixed-Integer Linear Programming*

Os problemas pertencentes à classe *Mixed-Integer Linear Programming* (MILP) são encontrados em diversas áreas. As principais razões para o sucesso das aplicações dos modelos MILP são os diversos solucionadores (*solvers*) disponíveis, baseados na programação linear e estruturas computacionais de buscas avançadas, e sua modelagem e flexibilidade. Diversos *solvers* eficientes, que incorporam técnicas avançadas da programação linear, são utilizados para solucionar os modelos MILP (VIELMA, 2015).

Um modelo MILP na sua forma padrão é representado por restrições, que são expressas com base em equações ou inequações lineares, e uma função objetivo representada por uma equação linear a ser minimizada ou maximizada. Um conjunto de variáveis reais e inteiras, definidas de acordo com o problema, também fazem parte do modelo. Algebricamente um modelo MILP pode ser representado conforme as equações abaixo (WOLSEY, 2008).

$$\text{minimize } cx + hy \quad (3.14a)$$

$$\text{subject to : } Ax + Gy \geq b \quad (3.14b)$$

$$x \in \mathbb{R}_+^p \quad (3.14c)$$

$$y \in \mathbb{Z}_+^n \quad (3.14d)$$

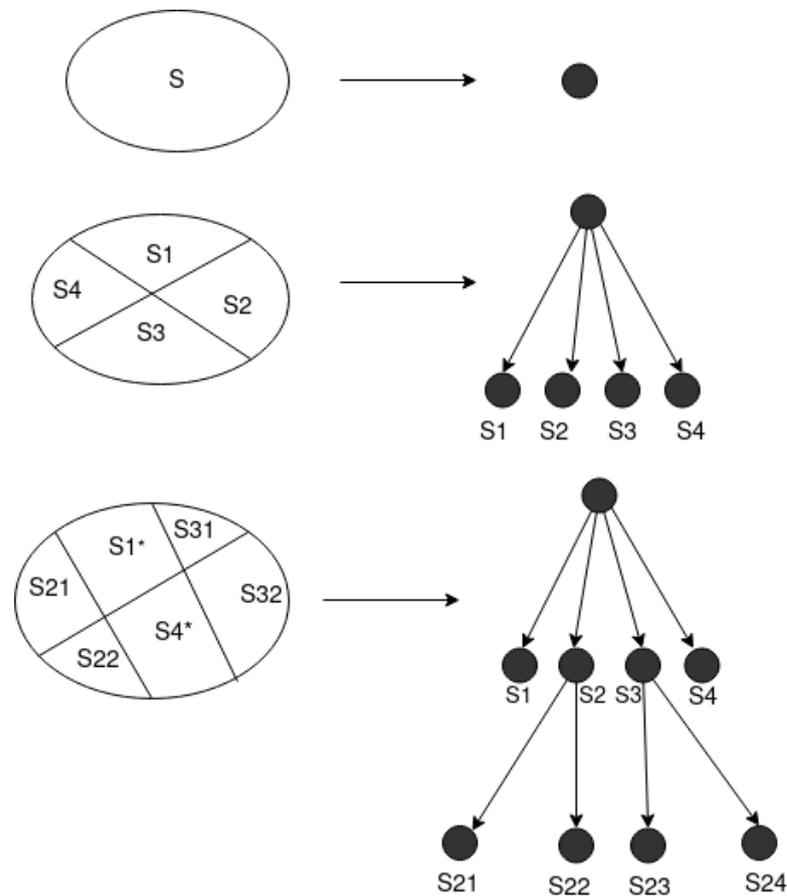
em que o conjunto  $\mathbb{R}_+^p$  representa o espaço vetores  $p$ -dimensionais reais não-negativos; o conjunto  $\mathbb{Z}_+^n$  representa o espaço vetores  $n$ -dimensionais inteiros não-negativos; o vetor  $x$  representa as  $p$  variáveis contínuas; o vetor  $y$  representa as  $n$  variáveis inteiras;  $A \in \mathbb{R}^{m \times p}$  e  $G \in \mathbb{R}^{m \times n}$  são matrizes que definem as restrições;  $b$  é um vetor na forma  $m$ -dimensional que faz parte das restrições;  $c$  e  $h$  são vetores de dimensão  $p$  e  $n$ , respectivamente, empregados na função objetivo (WOLSEY, 2008).

A função objetivo é representada pela equação linear (3.14a), as restrições são representadas pelas equações (3.14b) e os conjuntos são definidos pelas equações (3.14c) e (3.14d). Nota-se uma diferença do modelo MILP apresentado acima em relação ao modelo LP apresentado na Seção 3.3.1, uma vez que no modelo MILP é adicionado a variável  $y$  que só assume valores inteiros. As principais classes de algoritmos utilizados para solucionar os modelos MILP são: *Branch and Bound* e *Branch and Cut*.

## Algoritmo *Branch-and-Bound*

O paradigma *Branch-and-Bound* está presente em algoritmos utilizados para resolução de problemas discretos da otimização matemática, muitos deles de complexidade *NP-Hard*, especialmente aqueles que pertencem à classe MILP. O princípio desse paradigma é dividir o problema original em subproblemas menores, ganhando eficiência na resolução. Inicialmente, existe apenas um subproblema, nomeado como espaço de solução completa, e a solução ótima pode ser definida como tendo valor  $+\infty$ . Os sub-problemas são representados por *nodes* em uma *search tree* gerada dinamicamente, a qual é inicializada com apenas um *node*, nomeado de *root*. A cada iteração, o algoritmo *Branch-and-Bound* seleciona um nó aberto para processamento, calcula limites e gerar novos *nodes* a partir dos limites calculados. A Figura 12 ilustra a geração de *search trees* a partir dos subproblemas (CLAUSEN, 1999).

Figura 12 – Espaço de busca do Algoritmo *Branch-and-Bound*.



Fonte: (CLAUSEN, 1999).

### 3.3.3 Um modelo MILP para uma Rede Neural ReLU

Considerando uma rede neural ReLU com  $K + 1$  camadas de neurônios, numerados de 0 até  $K$ . A camada de entrada,  $k = 0$ , normalmente não é considerada como uma camada,

enquanto que a última camada,  $k = K$ , é a camada de saída. Cada camada  $k \in \{0, \dots, K\}$  possui  $n_k$  neurônios numerados de 1 até  $n_k$ . Para simplificar, considera-se as redes neurais com camadas completamente conectadas com parâmetros  $\theta = (W^k, b^k)$  para  $k \in \{1, \dots, K\}$  onde  $W^k \in R^{n_k \times n_{k-1}}$  e  $b^k \in R^{n_k}$ . Portanto, uma rede neural pode ser declarada como  $f_\theta : R^{n_0} \rightarrow R^{n_K}$  onde  $k, n_0$  e  $n_k$  ( $k = 1, \dots, K$ ) estão embutidos no conjunto de parâmetros  $\theta := \{\theta^0, \theta^1, \dots, \theta^K\}$  (GRIMSTAD; ANDERSSON, 2019).

O vetor  $x^k \in R^{n_k}$  é a saída da camada  $k$  e  $x_j^k$  é a saída do  $j$ -ésimo neurônio (nó) para  $j = \{1, \dots, n_k\}$ . Baseando nessa notação,  $x_j^0$  é o  $j$ -ésimo valor de entrada, enquanto  $x_j^K$  é o  $j$ -ésimo valor de saída de uma rede neural. Portanto, para cada camada oculta  $k \in \{1, \dots, K-1\}$ , o vetor de saída  $x^k$  é calculado conforme (GRIMSTAD; ANDERSSON, 2019):

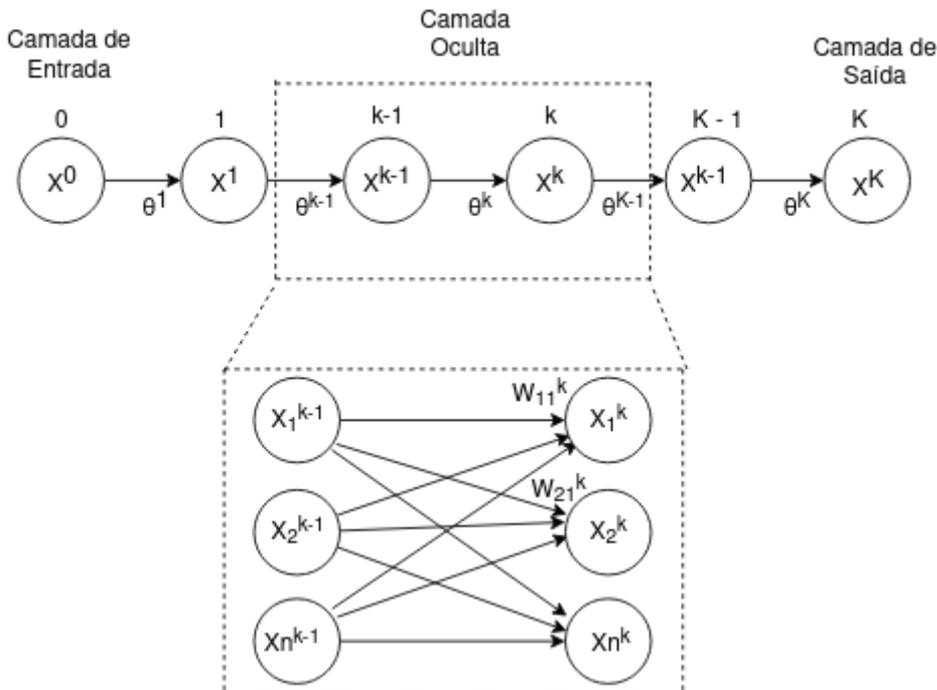
$$x^k = \sigma(W^k \cdot x^{k-1} + b^k) \quad (3.15)$$

onde  $\sigma(y) := \max\{0, y\}$  é a função de ativação ReLU para um vetor real  $y$ . A saída de uma rede neural completamente conectada,  $x^K \in R^{n_K}$  é dada pela equação abaixo:

$$x^K = W^K \cdot x^{K-1} + b^K \quad (3.16)$$

A Figura abaixo ilustra uma estrutura em blocos de uma rede neural ReLU.

Figura 13 – Diagrama Ilustrativo da Estrutura de uma Rede Neural ReLU.



Fonte: (GRIMSTAD; ANDERSSON, 2019).

Nota-se que a saída da camada  $k$  é calculada usando a entrada  $x^{k-1}$  da camada anterior e o parâmetro  $\theta^k = (W^k, b^k)$ . Existem diversas formas para modelar o operador ReLU, porém

consideramos a equação linear, baseado em (GRIMSTAD; ANDERSSON, 2019), apresentada conforme a equação abaixo.

$$w^T \cdot y + b = x - s \quad (3.17)$$

onde  $w$  é o vetor de pesos da conexão, o  $y$  é a saída,  $b$  é um escalar com o *bias*, o  $x$  é a entrada e  $s$  é uma variável não negativa.

Nota-se que a saída da rede ReLU é separada em uma parte positiva  $x > 0$  e uma parte negativa  $s \geq 0$ , portanto a saída da rede ReLU pode ser obtida colocando uma das variáveis ( $x$  ou  $s$ ) em zero (GRIMSTAD; ANDERSSON, 2019). Assumindo que podemos computar valores finitos  $L$  e  $U$  de modo que:

$$L \leq w^T \cdot y + b \leq U \quad (3.18)$$

onde  $L \leq 0$  e  $U \geq 0$  são respectivamente o limite inferior e superior. Assim, podemos modelar a lógica da função ReLU utilizando restrições tipo Big-M, adicionando uma variável binária  $z$  para cada neurônio, conforme as equações apresentadas abaixo.

$$0 \leq x \leq Uz \quad (3.19a)$$

$$s \leq -L(1 - z) \quad (3.19b)$$

$$z \in \{0, 1\} \quad (3.19c)$$

Note que  $z = 1$  se  $w^T \cdot y + b \geq 0$ , implicando em  $0 \leq x \leq U$  e  $s = 0$ . Por outro lado, se  $z = 0$ , então  $w^T \cdot y + b \leq 0$ , o que implica em  $x = 0$  e  $0 \leq s \leq -L$ .

Assim, uma rede neural artificial ReLU pode ser expressa por meio de uma formulação MILP, conforme as equações abaixo, considerando uma variável binária para cada neurônio  $(j, k)$  da rede neural.

- **Camada de Entrada:**

$$L^0 \leq x^0 \leq U^0 \quad (3.20)$$

- **Camada Oculta (*Hidden Layer*):** Para as equações (3.21a) e (3.21b), considera-se  $k \in \{1, \dots, K - 1\}$ ,

$$W^k \cdot x^{k-1} + b^k = x^k - s^k \quad (3.21a)$$

$$x^k, s^k \geq 0 \quad (3.21b)$$

Para as equações (3.22a), (3.22b) e (3.22c) considera-se  $k \in \{1, \dots, K - 1\}$  e  $j \in \{1, \dots, n_k\}$ :

$$z_j^k \in \{0, 1\} \quad (3.22a)$$

$$x_j^k \leq U_j^k z_j^k \quad (3.22b)$$

$$s_j^k \leq -L_j^k (1 - z_j^k) \quad (3.22c)$$

- **Camada de Saída:** O sinais de saída da rede ReLU são calculados conforme segue:

$$W^K \cdot x^{K-1} + b^K = x^K \quad (3.23a)$$

$$L^K \leq x^K \leq U^K \quad (3.23b)$$

A saída  $x^K$  de uma rede neural ReLU é a mesma para a saída da formulação acima, considerando a mesma entrada  $x^0$ . Portanto, a formulação acima modela uma rede neural ReLU (GRIMSTAD; ANDERSSON, 2019).

## 3.4 PIECEWISE LINEAR FUNCTIONS

A modelagem de não-linearidades presentes em funções objetivo, ou em restrições, contidas em um modelo de otimização matemática é uma ferramenta importante para a criação de tecnologias que empregam programação linear. As funções lineares por partes (*Piecewise-Linear Functions*, PWL) vêm sendo utilizadas para aproximar comportamentos não-lineares de funções, deixando-as com melhores propriedades (suavidade e linearidade) para otimização. Quando se cria um modelo de uma função não-linear empregando modelos PWL, o problema pode ser resolvido utilizando-se um modelo MILP introduzindo variáveis binárias e restrições adicionais (PADBERG, 2000).

### 3.4.1 Aproximação Unidimensional

Seja  $\phi(x_1, \dots, x_n)$  uma função separável qualquer não-linear do  $\mathbb{R}^n$  para  $\mathbb{R}$ . O termo separável significa que a função pode ser declarada como (PADBERG, 2000):

$$\phi(x_1, \dots, x_n) = \sum_{j=1}^n \phi_j(x_j) \quad (3.24)$$

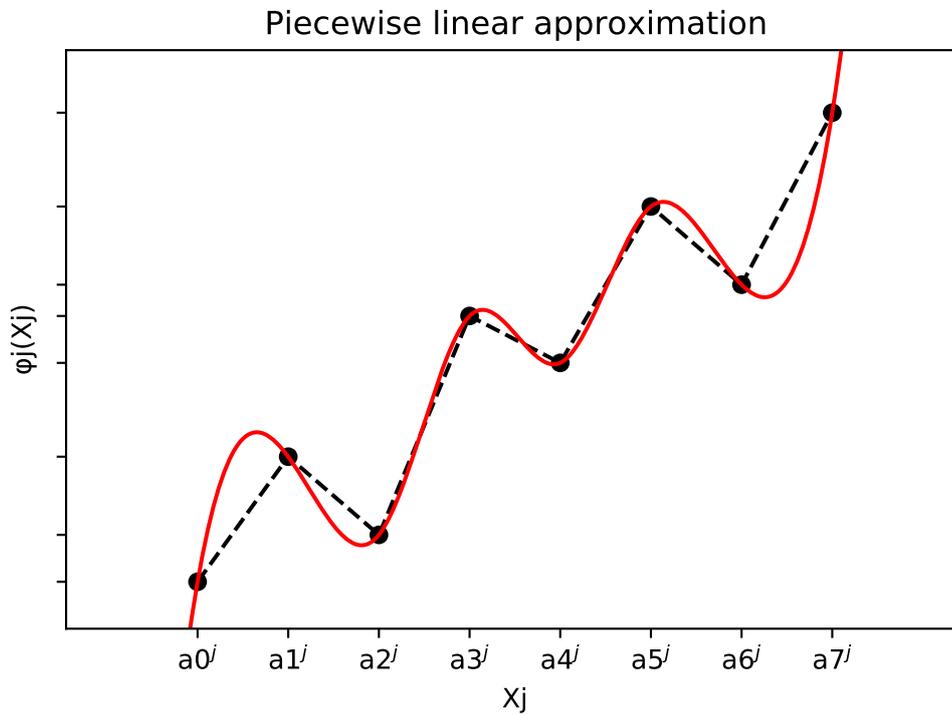
onde cada termo  $\phi_j(x_j)$  mapeia um valor de  $\mathbb{R}$  para  $\mathbb{R}$ . Para cada variável  $x_j$ , tal que  $j \in \{1, \dots, n\}$ , o seu domínio é definido como um intervalo finito  $[a_0^j, a_u^j]$  e assim sobre o intervalo é aproximado cada termo  $\phi(x_j)$  por uma função PWL representada por  $\hat{\phi}_j(x_j)$ . Para cada intervalo  $[a_0^j, a_u^j]$ ,  $j \in \{1, \dots, n\}$ , deve ser definido um particionamento  $a_0^j \leq a_1^j \leq a_2^j \leq \dots \leq a_{k_j}^j = a_u^j$ . A Figura 14 abaixo ilustra este processo.

Tomando um  $k_j$  grande o suficiente para que a distância entre dois pontos quaisquer consecutivos de um particionamento seja pequeno o suficiente, é possível aproximar  $\phi_j(x_j)$  utilizando as funções PWL. Para cada ponto  $a_l^j$  do particionamento, é calculado o valor correspondente da função (PADBERG, 2000),

$$b_l^j = \phi_j(a_l^j) \quad \text{for } 0 \leq l \leq k_j \quad (3.25)$$

com  $j \in \{1, \dots, n\}$ . Na programação matemática encontram-se alguns modelos MILPs que representam uma função PWL, tais como: *Convex Combination*, *Incremental* e *SOS2*, entre outros.

Figura 14 – Ilustração da Aproximação Linear Por Partes (PWL) de uma Função Não-linear.



Fonte: (PADBERG, 2000).

### Modelo *Convex Combination* (CC)

Considerando os pontos  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$  de uma representação gráfica, o modelo CC pode ser representado conforme as equações abaixo (VIELMA; AHMED; NEMHAUSER, 2009)

$$x = \sum_{i=0}^n \lambda_i x_i \quad (3.26a)$$

$$y = \sum_{i=0}^n \lambda_i y_i \quad (3.26b)$$

$$1 = \sum_{i=0}^n \lambda_i \quad \lambda_i \geq 0, i = 0, \dots, n \quad (3.26c)$$

$$1 = \sum_{i=1}^n z_i \quad z_i \in \{0, 1\}, i = 1, \dots, n \quad (3.26d)$$

$$\lambda_0 \leq z_1 \quad (3.26e)$$

$$\lambda_i \leq z_i + z_{i+1}, i = 1, \dots, n - 1 \quad (3.26f)$$

$$\lambda_n \leq z_n \quad (3.26g)$$

Para  $z_i$  ser igual a 1 é necessário que  $x \in [x_{i-1}, x_i]$ .

## Modelo Incremental

Considerando os pontos  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$  de uma representação gráfica, o modelo incremental pode ser representado conforme as equações abaixo (VIELMA; AHMED; NEMHAUSER, 2009):

$$x = x_0 + \sum_{i=1}^n \sigma_i \quad (3.27a)$$

$$y = y_0 + \sum_{i=1}^n \frac{(y_i - y_{i-1})}{(x_i - x_{i-1})} \sigma_i \quad (3.27b)$$

$$\sigma_1 \leq (x_1 - x_0) \quad (3.27c)$$

$$\sigma_i \leq (x_i - x_{i-1})z_{i-1}, \quad i = 2, \dots, n \quad (3.27d)$$

$$\sigma_i \geq (x_i - x_{i-1})z_i, \quad i = 1, \dots, n - 1 \quad (3.27e)$$

$$\sigma_n \geq 0 \quad (3.27f)$$

$$z_i \in \{0, 1\}, \quad i = 1, \dots, n - 1 \quad (3.27g)$$

Considerando o sistema de restrições, se  $z_j = 1$  então obrigatoriamente  $z_i = 1$  para  $i = j - 1, \dots, 1$ . Além disso, se  $z_i$  for igual a 1 então  $\sigma_i = (x_i - x_{i-1})$ .

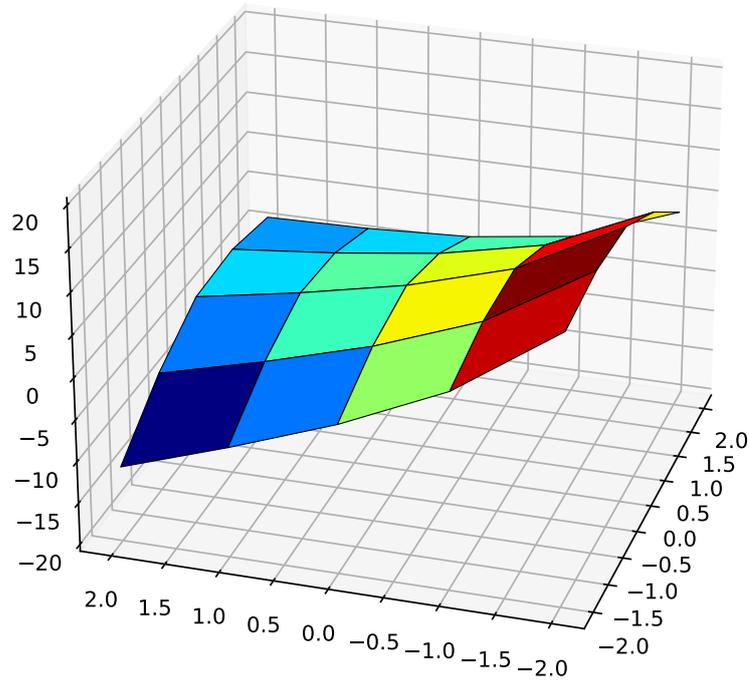
### 3.4.2 Aproximação Multidimensional

Quando uma função separável não-linear tem mais de uma dimensão, ou seja, a função é multidimensional, a aproximação da função também pode ser obtida com um modelo PWL. Para o caso de uma função bidimensional, por exemplo  $f(x, y) = e^{-y} + e^{-x^2} - x^2 + xy - 2y$  conforme ilustrada na Figura 15, um modelo PWL pode ser gerado para realizar a aproximação (SILVA; CAMPONOVARA, 2014)

O domínio da função  $f(x, y)$  acima é dado por  $D = [-2, 2] \times [-2, 2]$  e pode ser representado como um conjunto de polítopos. Um exemplo é a triangulação J1 (*Union Jack*), conforme ilustra a Figura 16, que particiona o domínio  $D$  no conjunto de polítopos  $P = \{P_1, P_2, \dots, P_{32}\}$ . O conjunto de vértices dos polítopos  $p \in P$  são  $V(P_1) = \{(-2, -2), (-2, -1), (-1, -1)\}$ ,  $V(P_2) = \{(-2, -1), (-1, -1), (-2, 0)\}$ ,  $\dots$ ,  $V(P_{32}) = \{(2, -1), (1, -1), (2, -2)\}$  (SILVA; CAMPONOVARA, 2014).

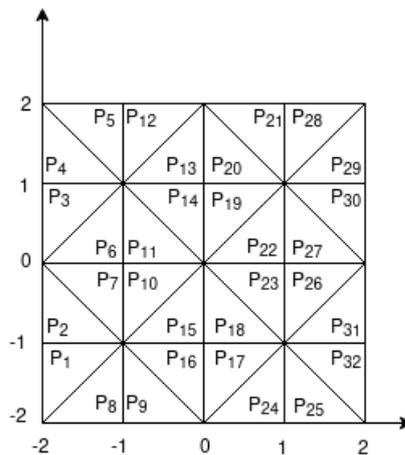
A triangulação *Union Jack* produz um particionamento de domínio que é compatível com todos os modelos PWL, exceto o modelo SOS2. A formalização e o detalhamento da triangulação *Union Jack* podem ser encontrados em (SILVA, 2012). Na literatura existem algumas maneiras diferentes de representar as funções PWL por meio dos modelos MILPs disponíveis, incluindo: combinação convexa (*Convex Combination*) dos vértices de um único polítopo ( $P_n$ ), conjunto de equações de hiperplano e desigualdades lineares, incrementalmente a partir de um conjunto de polítopos ordenados (simples) e vértices, e o modelo SOS2, entre outros (SILVA; CAMPONOVARA, 2014).

Figura 15 – Ilustração da Representação Linear or Partes da Função Bidimensional  $f(x, y) = e^{-y} + e^{-x^2} - x^2 + xy - 2y$ .



Fonte: O autor.

Figura 16 – Particionamento do Domínio  $D = [-2, 2] \times [-2, 2]$  Segundo a Triangularização Union Jack.



Fonte: (SILVA; CAMPONOGARA, 2014).

### Modelo Convex Combination (CC)

Seja  $P$  um conjunto de poliedros que definem o domínio,  $V$  o conjunto de *breakpoints*, e  $V(p)$  o conjunto de vértices de um poliedro  $p \in P$ . Cada vértice  $v \in V$  tem um peso associado declarado como  $\lambda_v \in [0, 1]$ . Considere um ponto  $x \in D$  dentro do domínio da função  $f(\cdot)$ . Se  $x \in p$ , para  $p \in P$ , então a variável binária  $y_p \in \{0, 1\}$  assume o valor 1, caso contrário  $y_p$

assume o valor 0. Portanto, dadas estas definições e declaração de variáveis, pode-se formular um modelo CC (SILVA; CODAS; CAMPONOGARA, 2012) conforme segue:

$$f = \sum_{v \in V} f(v) \lambda_v \quad (3.28a)$$

$$x = \sum_{v \in V} v \lambda_v \quad (3.28b)$$

$$1 = \sum_{v \in V} \lambda_v \quad (3.28c)$$

$$1 = \sum_{p \in P} y_p \quad (3.28d)$$

$$\lambda_v \leq \sum_{p \in P(v)} y_p \quad (3.28e)$$

$$\lambda_v \geq 0, \quad v \in V \quad (3.28f)$$

$$y_p \in \{0, 1\}, \quad p \in P \quad (3.28g)$$

em que o conjunto de poliedros que contém um *breakpoint*  $v$  é definido como  $P(v) = \{p \in P : v \in V(p)\}$ .

## Modelo SOS2

Sejam  $X = \{x_1, x_2, \dots, x_n\}$  e  $Y = \{y_1, y_2, \dots, y_n\}$  os conjuntos de *breakpoints* para a função bidimensional  $f$  com domínio  $(x, y) \in [x_1, x_n] \times [y_1, y_n]$ . O conjunto de poliedros de uma partição regular é definido como  $P = \{[x_i, x_{i+1}] \times [y_j, y_{j+1}] : i = 1, \dots, n-1 \text{ e } j = 1, \dots, n-1\}$ , onde  $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$  define um poliedro. Com base nestas definições, pode-se formular um modelo SOS2 (SILVA; CAMPONOGARA, 2014) conforme segue:

$$f = \sum_{x \in X} \sum_{y \in Y} f(x, y) \lambda_{x,y} \quad (3.29a)$$

$$x = \sum_{x \in X} \sum_{y \in Y} x \lambda_{x,y} \quad (3.29b)$$

$$y = \sum_{x \in X} \sum_{y \in Y} y \lambda_{x,y} \quad (3.29c)$$

$$1 = \sum_{x \in X} \sum_{y \in Y} \lambda_{x,y} \quad (3.29d)$$

$$\lambda_{x,y} \geq 0, \quad \forall x \in X \text{ e } \forall y \in Y \quad (3.29e)$$

$$n_x = \sum_{y \in Y} \lambda_{x,y}, \quad \forall x \in X \quad (3.29f)$$

$$n_y = \sum_{x \in X} \lambda_{x,y}, \quad \forall y \in Y \quad (3.29g)$$

$$\{n_x\}_{x \in X} \text{ é um conjunto SOS2} \quad (3.29h)$$

$$\{n_y\}_{y \in Y} \text{ é um conjunto SOS2} \quad (3.29i)$$

Na otimização discreta um conjunto ordenado especial (SOS) é um conjunto ordenado de variáveis, usado como uma forma adicional de especificar condições de integralidade em um modelo de otimização. Há dois tipos de conjuntos ordenados de variáveis, SOS1 e SOS2, para esse trabalho, utiliza-se o SOS2, que são um conjunto ordenado de variáveis não negativas, das quais no máximo duas podem ser diferentes de zero, e se duas forem diferentes de zero, elas devem ser consecutivas em sua ordenação (BEALE; FORREST, 1976).

### 3.4.3 Simulador Marlim

A função de produção de líquido é tipicamente disponível em simuladores que modelam a estrutura física do poço, a interface poço-reservatório e o método de elevação artificial, como o gas-lift. O simulador Marlim é um destes simuladores e pode representar condições estáticas de poços de petróleo, permitindo a geração de dados relacionados aos poços simulados. Basicamente, o Marlim realiza uma análise de sensibilidade a partir de parâmetros de entrada (taxa de gas-lift e pressão na cabeça do poço, por exemplo) (SEMAN et al., 2021).



## 4 METODOLOGIA DA SOLUÇÃO PROPOSTA

Com base no trabalho de Seman et al. (2021), algumas formulações foram desenvolvidas com o objetivo de identificar parâmetros-chave de modelos computacionais (simulador) de poços que compõem uma plataforma de petróleo específica, operando em estado estacionário. Esses modelos utilizam dados coletados a partir de medições de campo em condições operacionais definidas pelo operador. Tais condições são, por exemplo, a taxa de injeção de *gas-lift* e a pressão na cabeça de cada poço. Os parâmetros de interesse para este trabalho são RGO, BSW e IP, que são revisados periodicamente quando a produção do respectivo poço é direcionada a um separador de teste. Portanto, os parâmetros podem ficar desatualizados entre um teste de separação e outro, uma vez que os parâmetros não são estimados em tempo real. Nas próximas seções serão apresentadas as formulações matemáticas desenvolvidas para estimar os parâmetros citados.

### 4.1 SISTEMA DE PRODUÇÃO DE PETRÓLEO

Considerando uma plataforma de petróleo com vários poços representados pela variável  $n \in \mathcal{N}$ , para cada poço  $n$  a metodologia desenvolvida busca identificar os parâmetros  $rgo^n$ ,  $bsw^n$  e  $ip^n$  sintonizados às condições vigentes, porém que fiquem próximos dos parâmetros  $rgo$ ,  $bsw$  e  $ip$  obtidos no último teste de separação, os quais por conveniência são nomeados neste trabalho como  $rgo_{hist}^n$ ,  $bsw_{hist}^n$  e  $ip_{hist}^n$ . Para reduzir a diferença (erro) entre os parâmetros preditos com os parâmetros históricos, adota-se a função objetivo apresentada conforme a equação abaixo.

$$f = \min \sum_{n \in \mathcal{N}} \sigma_{rgo}^n \cdot (rgo^n - rgo_{hist}^n)^2 + \sum_{n \in \mathcal{N}} \sigma_{bsw}^n \cdot (bsw^n - bsw_{hist}^n)^2 + \sum_{n \in \mathcal{N}} \sigma_{ip}^n \cdot (ip^n - ip_{hist}^n)^2 \quad (4.1)$$

onde  $\mathcal{N} = \{1, \dots, N\}$  é o conjunto dos poços de uma plataforma e  $\sigma_{rgo}^n$ ,  $\sigma_{bsw}^n$  e  $\sigma_{ip}^n$  são pesos positivos para os erros. Os pesos da equação são para os operadores da plataforma, que conhecem bem a plataforma, atribuir ponderações conforme a incerteza relativa aos valores históricos, por exemplo, atribuindo pesos maiores para  $bsw$  e menores para  $rgo$  quando há maior incerteza na medição da fração de água em relação à medição da razão gás-óleo, ou vice-versa.

### 4.2 PRODUÇÃO DOS POÇOS DE UMA PLATAFORMA DE PETRÓLEO

Os poços acoplados a plataformas de produção de petróleo considerados neste trabalho operam com elevação artificial por injeção contínua de gás comprimido (*gas-lift*), caracterizados

por uma produção de líquido induzida pela pressão da cabeça do poço ( $whp$ ) e a vazão de injeção de *gas-lift* ( $q_{gl}$ ). Conforme apresentado no Capítulo 2, basicamente aplica-se *gas-lift* no fundo do poço forçando a vazão multifásica (petróleo, água e gás) até a superfície devido à diferença de pressão. As equações apresentadas abaixo modelam a produção de petróleo individual de cada poço  $n \in \mathcal{N}$ .

$$q_w^n + q_o^n = q_l^n(q_{gl}^n, whp^n, rgo^n, bsw^n, ip^n) \quad (4.2a)$$

$$q_g^n = rgo^n \cdot q_o^n \quad (4.2b)$$

$$q_w^n \cdot (1 - bsw^n) = bsw^n \cdot q_o^n \quad (4.2c)$$

onde  $q_o^n$ ,  $q_g^n$  e  $q_w^n$  são as vazões das fases de petróleo, gás e água de cada poço, respectivamente. A função de produção de líquido total do poço  $n$  é representada por  $q_l^n(\cdot)$ . A restrição representada pela equação (4.2a) define a produção de líquido produzido por cada poço encontrado em um reservatório. As restrições caracterizadas pelas equações (4.2b) e (4.2c) fornecem as vazões de gás e água produzidas por cada poço da plataforma, com base na produção de líquido e os parâmetros  $rgo^n$  e  $bsw^n$ .

### 4.3 PLATAFORMA DE POÇOS DE PETRÓLEO

Por convenção neste trabalho consideramos um sistema único de processamento de fluidos, ou seja um único separador. Portanto, as equações abaixo em um dado instante de tempo  $t \in \mathcal{T}$  representam o total produzido em cada fase por poços conectados a um separador da plataforma.

$$q_{oil-prod}^t = \sum_{n \in \mathcal{N}} q_o^{n,t} \quad (4.3a)$$

$$q_{gas-prod}^t = \sum_{n \in \mathcal{N}} q_g^{n,t} \quad (4.3b)$$

$$q_{water-prod}^t = \sum_{n \in \mathcal{N}} q_w^{n,t} \quad (4.3c)$$

$$q_{gas-lift}^t = \sum_{n \in \mathcal{N}} q_{gl}^{n,t} \quad (4.3d)$$

A produção de gás total (representada pela variável  $q_{gas-prod}$ ), normalmente, não é medida nas plataformas, sendo calculada por meio do gás exportado, o gás queimado no *flare* e o consumo na turbina. A equação abaixo representa o cálculo do gás produzido pela plataforma em um dado instante de tempo  $t \in \mathcal{T}$ .

$$q_{gas-exp}^t = q_{gas-prod}^t - q_{flare}^t - q_{turbine}^t \quad (4.4)$$

Algumas restrições da plataforma estão relacionadas às limitações encontradas no processo de produção como, por exemplo, a capacidade de tratamento de óleo e gás. As equações

abaixo correspondem às restrições utilizadas neste trabalho:

$$q_{gas-prod}^t + q_{gas-lift}^t - q_{flare}^t \leq q_{gtc}^t \quad (4.5a)$$

$$q_{gas-exp}^t \leq q_{gas-exp}^{max} \quad (4.5b)$$

$$q_{flare}^{min} \leq q_{flare}^t \leq q_{flare}^{max} \quad (4.5c)$$

$$q_{oil-prod}^t + q_{water-prod}^t \leq q_l^{max} \quad (4.5d)$$

$$q_{water-prod}^t \leq q_w^{max} \quad (4.5e)$$

$$q_{oil-prod}^t \leq q_{oil}^{max} \quad (4.5f)$$

$$q_{gas-prod}^t \leq q_{gas}^{max} \quad (4.5g)$$

## 4.4 CONDIÇÕES DE OPERAÇÃO

Para cada instante de tempo  $t \in \mathcal{T}$ , as restrições apresentadas abaixo foram utilizadas para assegurar que a pressão na cabeça dos poços ( $whp$ ) e a vazão de injeção de *gas-lift* ( $q_{gl}$ ) fossem fixadas nos valores das medições realizadas no instante  $t$ . Logo, para todo  $n \in \mathcal{N}$ , temos que:

$$whp^{n,t} = whp_{st}^{n,t} \quad (4.6a)$$

$$q_{gl}^{n,t} = q_{gl,st}^{n,t} \quad (4.6b)$$

onde  $whp_{st}^{n,t}$  e  $q_{gl,st}^{n,t}$  são respectivamente os valores de pressão na cabeça e a vazão de injeção de *gas-lift* medidos no tempo  $t$ , que caracterizam as condições operacionais em regime permanente dos poços.

## 4.5 LIMITES PARA PARÂMETROS-CHAVE DOS POÇOS

As restrições a seguir estabelecem um intervalo de valores que os parâmetros RGO, BSW e IP identificados podem assumir:

$$rgo_{min}^n \leq rgo^n \leq rgo_{max}^n \quad (4.7a)$$

$$bsw_{min}^n \leq bsw^n \leq bsw_{max}^n \quad (4.7b)$$

$$ip_{min}^n \leq ip^n \leq ip_{max}^n \quad (4.7c)$$

Uma tolerância ( $\epsilon$ ) pequena é permitida para os valores preditos da produção de petróleo, gás e água da plataforma em relação aos valores medidos. Portanto, algumas restrições são utilizadas para garantir essa tolerância, considerando o regime estacionário em um tempo  $t \in \mathcal{T}$ , conforme as equações abaixo:

$$q_{oil-prod}^{t,st} \cdot (1 - \epsilon_{oil}) \leq q_{oil-prod}^t \leq q_{oil-prod}^{t,st} \cdot (1 + \epsilon_{oil}) \quad (4.8a)$$

$$q_{gas-prod}^{t,st} \cdot (1 - \epsilon_{gas}) \leq q_{gas-prod}^t \leq q_{gas-prod}^{t,st} \cdot (1 + \epsilon_{gas}) \quad (4.8b)$$

$$q_{water-prod}^{t,st} \cdot (1 - \epsilon_{water}) \leq q_{water-prod}^t \leq q_{water-prod}^{t,st} \cdot (1 + \epsilon_{water}) \quad (4.8c)$$

em que  $q_{oil-prod}^{t,st}$ ,  $q_{gas-prod}^{t,st}$  e  $q_{water-prod}^{t,st}$  são respectivamente as medições de produção total de petróleo, gás e água da plataforma durante o período de regime estacionário no tempo  $t \in \mathcal{T}$ .

Nota-se que há uma tolerância para cada uma das fases, representadas por  $\epsilon \in \{\epsilon_{oil}, \epsilon_{gas}, \epsilon_{water}\}$ , permitindo ao operador da plataforma definir limites mais apertados para o petróleo, enquanto que a água e o gás sejam mais livres na sintonia. Portanto, as restrições (4.8a), (4.8b) e (4.8c) são impostas para apertar os limites das fases com base no conhecimento do operador.

## 4.6 FORMULAÇÃO MATEMÁTICA DO PROBLEMA DE AJUSTE DE PARÂMETROS

A sintonia dos parâmetros-chave dos poços pode ser formalizado com as seguintes equações:

$$f = \min \sum_{n \in \mathcal{N}} \sigma_{rgo}^n \cdot (rgo^n - rgo_{hist}^n)^2 + \sum_{n \in \mathcal{N}} \sigma_{bsw}^n \cdot (bsw^n - bsw_{hist}^n)^2 + \sum_{n \in \mathcal{N}} \sigma_{ip}^n \cdot (ip^n - ip_{hist}^n)^2 \quad (4.9a)$$

$$\text{s.t. : (4.2) - (4.8)} \quad (4.9b)$$

onde os vetores com variáveis de decisão do problema são representados por:  $rgo = (rgo^n : n \in \mathcal{N})$ ,  $bsw = (bsw^n : n \in \mathcal{N})$  e  $ip = (ip^n : n \in \mathcal{N})$ . A vazão de líquido  $q_l^n(\cdot)$  não é uma função explicitamente dada e que possa ser otimizada. No entanto ela está disponível em simuladores que modelam a estrutura física de um poço, o método de elevação artificial com *gas-lift* e a interface poço-reservatório. Uma das dificuldades de emprego de tais simuladores em algoritmos de otimização é o custo computacional para simular múltiplos poços, o grande número de licenças necessárias e a não diferenciabilidade, entre outras. Para contornar estas dificuldades, Seman et al. (2021) utilizaram a regressão polinomial da função de produção de líquido  $q_l^n(\cdot)$ , técnica que tem obtido sucesso em alguns projetos de otimização da produção de curto prazo. Neste trabalho, utiliza-se uma rede neural artificial para representar (modelar) a função de produção de líquido dos poços, uma vez a quantidade de variáveis binárias utilizadas no modelo MILP da rede neural está diretamente relacionada à quantidade de neurônios da rede, independentemente da quantidade de variáveis (dimensionalidade) do problema. Por outro lado, em uma representação linear por partes, a quantidade de variáveis binárias está diretamente relacionada à dimensão do problema.

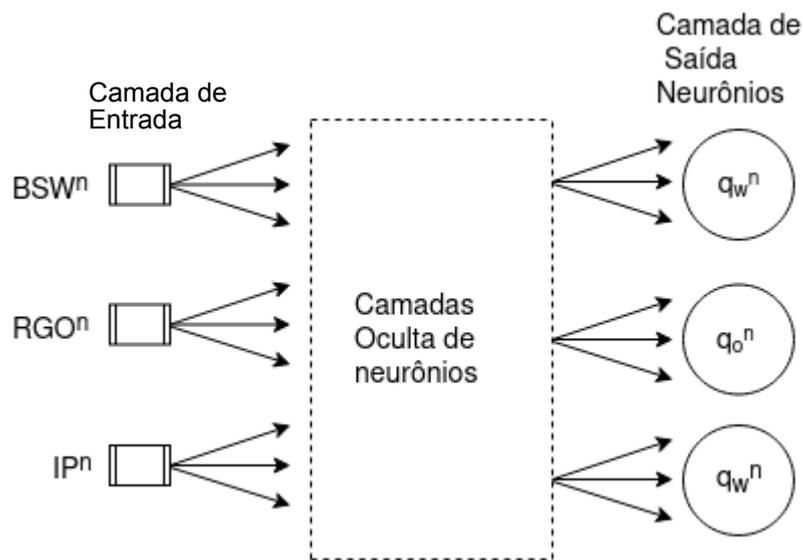
## 4.7 MODELO SIMPLIFICADO DOS POÇOS

Como a função  $q_l^n(\cdot)$  não é explicitamente dada e os simuladores dessa função podem ser onerosos. Como alternativa, Seman et al. (2021) utilizam uma regressão polinomial para

aproximar o comportamento dos poços. Nesta dissertação foram realizados diversos testes com redes neurais artificiais, considerando diferentes arquiteturas e utilizando dados obtidos com o simulador MARLIM da Petrobras, com o intuito de aproximar a função de produção de líquido  $q_l^n(\cdot)$  de cada poço  $n$ . Para integrar as redes neurais treinadas (uma para cada poço) no modelo de otimização apresentado em (4.9) realiza-se a conversão apresentada na Seção 3.3.3, em que uma rede neural é convertida em um modelo MILP.

A partir de dados gerados pelo simulador MARLIM, foram utilizados dados de  $BSW^n$ ,  $RGO^n$ ,  $IP^n$  como entrada da rede, enquanto a saída corresponde à vazão de água ( $q_w^n$ ), vazão de óleo ( $q_o^n$ ) e vazão de gás ( $q_g^n$ ) produzidas. A Figura 17 apresenta o esboço de uma rede neural desenvolvida para um único poço, respeitada pelos demais poços.

Figura 17 – Diagrama Ilustrativo da Arquitetura das Redes Neurais de Cada um dos Poços.



Fonte: O autor.

Nota-se que as restrições (4.2b) e (4.2c) adicionam não-linearidades ao problema, uma vez que há termos bilineares nas restrições, por exemplo  $rgo^n \cdot q_o^n$  e  $bsw^n \cdot q_o^n$ . No entanto, utilizando redes neurais artificiais é possível encontrar as vazões  $q_w^n$ ,  $q_o^n$  e  $q_g^n$  e com isso eliminar a bilinearidade do problema de identificação de parâmetros-chave de poços.



## 5 VALIDAÇÃO DA METODOLOGIA DE SINTONIA

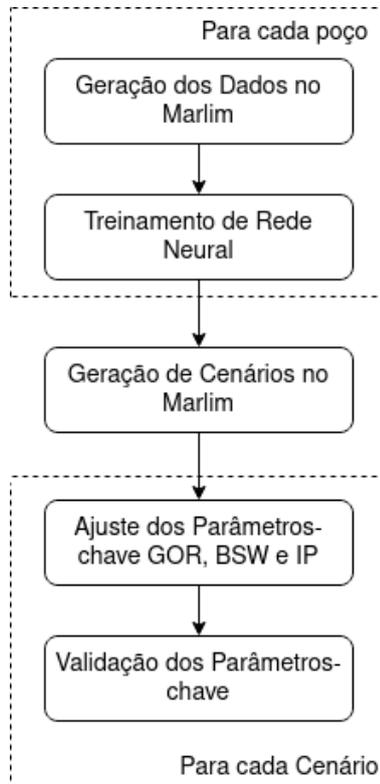
Neste capítulo é apresentada a validação da metodologia proposta para identificação de parâmetros-chave empregando redes neurais, com base no trabalho de Seman et al. (2021), com o objetivo de validar para cada cenário os parâmetros  $RGO_{adj}^n$ ,  $BSW_{adj}^n$  e  $IP_{adj}^n$  selecionados pela metodologia. Essa validação consiste em verificar em cada cenário se os parâmetros sintonizados estão mais próximos dos parâmetros de referência (“*benchmark*”), quando comparados aos parâmetros encontrados no último teste de separação ( $RGO_{hist}$ ,  $BSW_{hist}$  e  $IP_{hist}$ ). Além disso, em cada cenário compara-se a produção final de petróleo utilizando os parâmetros sintonizados, históricos e *benchmark*, por meio de um modelo de maximização da produção de petróleo, formalizado em programação matemática. Nas próximas seções serão apresentadas as etapas desenvolvidas neste trabalho para identificação e validação dos parâmetros sintonizados.

### 5.1 DETALHAMENTO DO PROCEDIMENTO DE VALIDAÇÃO

A Figura 18 apresenta em blocos as etapas desenvolvidas neste trabalho, com o objetivo de sintonizar os parâmetros  $RGO$ ,  $BSW$  e  $IP$  de poços acoplados em uma plataforma de petróleo da Petrobras (simulados), bem como validar se com a sintonia dos parâmetros obtêm-se ganhos significativos na produção de petróleo.

Considerando a sequência *top-down*, o primeiro bloco da Figura 18 representa um *script* Python que foi desenvolvido com a função de exportar dados do simulador de poços MARLIM, desenvolvido pela Petrobras. O segundo bloco, representa um *script* Python desenvolvido com a função de acessar e normalizar os dados exportados no bloco 1, criar e treinar uma rede neural para cada poço, armazenar os modelos das redes treinadas e os parâmetros (média e desvio padrão, entre outros) de cada poço. O terceiro bloco representa a utilização do simulador MARLIM para a geração cenários, em que a partir dos valores de parâmetros-chave históricos, gera-se valores de parâmetros-chave de referência (*benchmark*), variando em até 5% os valores dos parâmetros-chave histórico. Para cada cenário gerado, o quarto bloco representa um *script* Python desenvolvido para sintonizar os parâmetros  $RGO$ ,  $BSW$  e  $IP$  de cada poço selecionado, resolvendo o problema definido nas equações (4.9). O último bloco representa um *script* Python que tem como função validar se com os parâmetros sintonizados obtêm-se ganhos significativos. Nas próximas seções será apresentado em mais detalhes cada um dos blocos dispostos na Figura 18.

Figura 18 – Diagrama Ilustrativo das Etapas Desenvolvidas Para Identificação e Validação dos Parâmetros-chave.



Fonte: O autor.

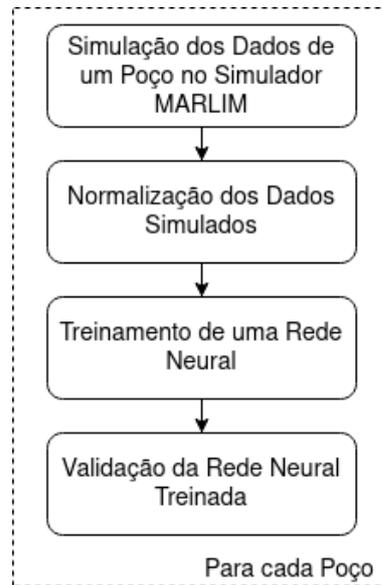
## 5.2 GERAÇÃO DOS DADOS E TREINAMENTO DE REDES NEURAS

Com o simulador de poços de petróleo MARLIM obtém-se dados dos poços para realizar o treinamento das redes neurais artificiais. O simulador recebe como entrada um conjunto de poços, bem como parâmetros de controle fixos ( $q_{gl}^n$  e  $whp^n$ ) e o mesmo retorna um conjunto de dados (*dataset*) com os respectivos valores de  $RGO$ ,  $BSW$ ,  $IP$ ,  $q_o$ ,  $q_w$  e  $q_g$  de cada poço selecionado.

Um *surrogate model* é desenvolvido para cada poço, treinando uma rede neural artificial com os dados exportados no simulador. As rede neurais são treinadas para modelar a função de produção de líquido  $q_l(\cdot)$  de cada poço, utilizando os dados de parâmetros-chave  $RGO_n$ ,  $BSW_n$  e  $IP_n$  como entrada e as vazões  $q_o$ ,  $q_w$  e  $q_g$  como saída. A Figura 19 ilustra em mais detalhes os blocos 1 e 2 da Figura 18.

Considerando uma única condição operacional, a partir da arquitetura em blocos apresentada na Figura 19 é desenvolvido um *script* Python, com a função de gerar dados com o simulador de poços de petróleo MARLIM, organizar e normalizar os dados simulados, treinar uma rede neural artificial RELU para cada poço, validar e salvar os modelos das redes treinadas.

Figura 19 – Diagrama Ilustrativo das Etapas Desenvolvidas para o Treinamento da Rede Neural Artificial de Cada Poço.



Fonte: O autor.

A equação abaixo apresenta a fórmula utilizada para normalizar os dados simulados.

$$x_{bsw\_norm} = (x_{bsw}^n - media_{bsw}^n) / std_{bsw}^n \quad (5.1)$$

onde a variável  $x_{bsw}^n$  representa a coluna (vetor)  $bsw$  do *dataset* do poço  $n$ , a variável  $media_{bsw}^n$  representa a média total do vetor  $x_{bsw}^n$ , e o  $std_{bsw}^n$  representa o desvio padrão do  $x_{bsw}^n$ . Para as outras entradas, como  $RGO^n$  e  $IP^n$  a mesma função de normalização é utilizada, bem como as saídas  $q_o^n$ ,  $q_w^n$ ,  $q_g^n$ .

Com os dados simulados e normalizados diversos testes foram realizados no treinamento das redes neurais dos poços, alterando a arquitetura e a quantidade de neurônios das redes. A arquitetura de rede neural que apresentou melhores resultados para todos os poços selecionados foram às redes neurais compostas por 4 camadas de 10 neurônios cada. As funções do erro do treinamento das redes neurais foram geradas para a validação das redes.

## 5.3 GERAÇÃO DE CENÁRIOS

A representação física dos poços, a interface do poço com o reservatório e os fluidos produzidos de cada poço são modelados no simulador MARLIM da Petrobras. Esse simulador é utilizado rotineiramente nas plataformas de petróleo para análises de vazões e sensibilidade dos poços operados com *gas-lift* (SEMAN et al., 2021). Portanto, para realizar a sintonia dos parâmetros-chave de poços foram gerados 100 cenários, em que o número de poços variava entre 4 a 10 poços por cenário. O retorno do simulador para cada cenário é um *dataset* com

parâmetros-chave históricos e “*benchmark*” de cada poço, bem como a produção total de petróleo processada por um único separador.

Algumas características dos poços foram mantidas inalteradas na geração dos cenários, tais como a taxa de *gas-lift* ( $q_{gl}$ ) e a pressão na cabeça ( $whp$ ), enquanto que os valores dos parâmetros históricos  $RGO_{hist}$ ,  $BSW_{hist}$  e  $IP_{hist}$  foram variados em até 5%, para a geração de parâmetros de referência (*benchmark*). Uma vez que os valores de  $q_{gl}$  e  $whp$  de cada poço são definidos como entrada, o simulador retorna um *dataset* com dados sobre os poços que fazem parte do cenário. Esse *dataset* retornado é composto por:  $BSW^n$ ,  $RGO^n$ ,  $IP^n$ ; a vazão de *gas-lift* ( $q_{gl}^n$ ); a pressão na cabeça ( $whp^n$ );  $BSW_{hist}^n$ ,  $RGO_{hist}^n$ ,  $IP_{hist}^n$ ; vazão total de água ( $qw_{total}$ ), óleo ( $qo_{total}$ ) e gás ( $qg_{total}$ ) de todos os poços do cenário. O conjunto de poços que fazem parte do cenário são referenciados neste trabalho como plataforma de petróleo.

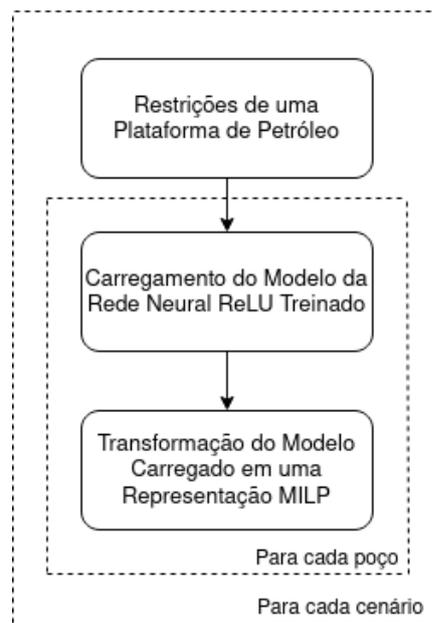
## 5.4 SINTONIA DOS PARÂMETROS-CHAVE

A sintonia dos parâmetros-chave  $RGO$ ,  $BSW$  e  $IP$  de cada poço selecionado foram realizados conforme a metodologia apresentada na Seção 4, segundo a qual cria-se um modelo matemático MIQP definido em (4.9), com algumas restrições que convertem o modelo de uma rede neural artificial RELU em um modelo MILP, conforme explicado por Grimstad and Andersson (2019). Além disso, algumas restrições operacionais da plataforma de produção de petróleo são consideradas no modelo, conforme demonstrado por Seman et al. (2021).

Baseado na arquitetura apresentada na Figura 20, um *script* Python foi desenvolvido com a função de realizar a sintonia dos parâmetros-chave  $RGO$ ,  $BSW$  e  $IP$  de cada poço selecionado. Nesse *script* cria-se um modelo matemático com as restrições da plataforma (conforme as equações da Seção 4), importa os modelos treinados das redes neurais RELU de cada poço, transforma cada modelo importado em um modelo MILP (restrições e limites da Seção 3.3.3) e encontra o ótimo da função objetivo apresentada na equação (4.1), resolvendo o problema (4.9). Esse processo é executado para cada cenário, correspondendo a diferentes condições operacionais, simulando diferentes situações características da plataforma. Após encontrar o  $RGO_{adj}^n$ ,  $BSW_{adj}^n$  e  $IP_{adj}^n$  de cada cenário, os mesmos são salvos em memória permanente.

Em cada cenário, gera-se dados referentes aos parâmetros-chave históricos e *benchmark* de cada um dos poços selecionados, que compõem uma plataforma de petróleo, conforme explicado na Seção 5.3. Os parâmetros-chave encontrados pela metodologia são nomeados neste trabalho como sintonizados (“predito”) e uma das formas de validá-los, é comparando-os com os parâmetros-chave históricos, em relação aos parâmetros-chave (*benchmark*). Como uma primeira validação, considerando um único cenário gerado sintoniza-se os parâmetros  $RGO^n$ ,  $BSW^n$  e  $IP^n$  e juntamente com os parâmetros históricos e *benchmark* gera-se um gráfico conforme apresentado na Figura 21.

Figura 20 – Diagrama Ilustrativo das Etapas Desenvolvidas para Realizar a Sintonia dos Parâmetros-chave.



Fonte: O autor.

Nota-se que a validação acima não é suficiente, pois mesmo que os parâmetros-chave sintonizados estejam mais próximos do *benchmark*, em relação ao histórico, a produção de petróleo do conjunto de poços selecionados é desconhecida. Portanto, na próxima seção será apresentada uma maneira consistente de validar a produção final de petróleo de cada um dos poços selecionados, utilizando os parâmetros-chave sintonizados, histórico e *benchmark*.

## 5.5 VALIDAÇÃO DOS PARÂMETROS SINTONIZADOS

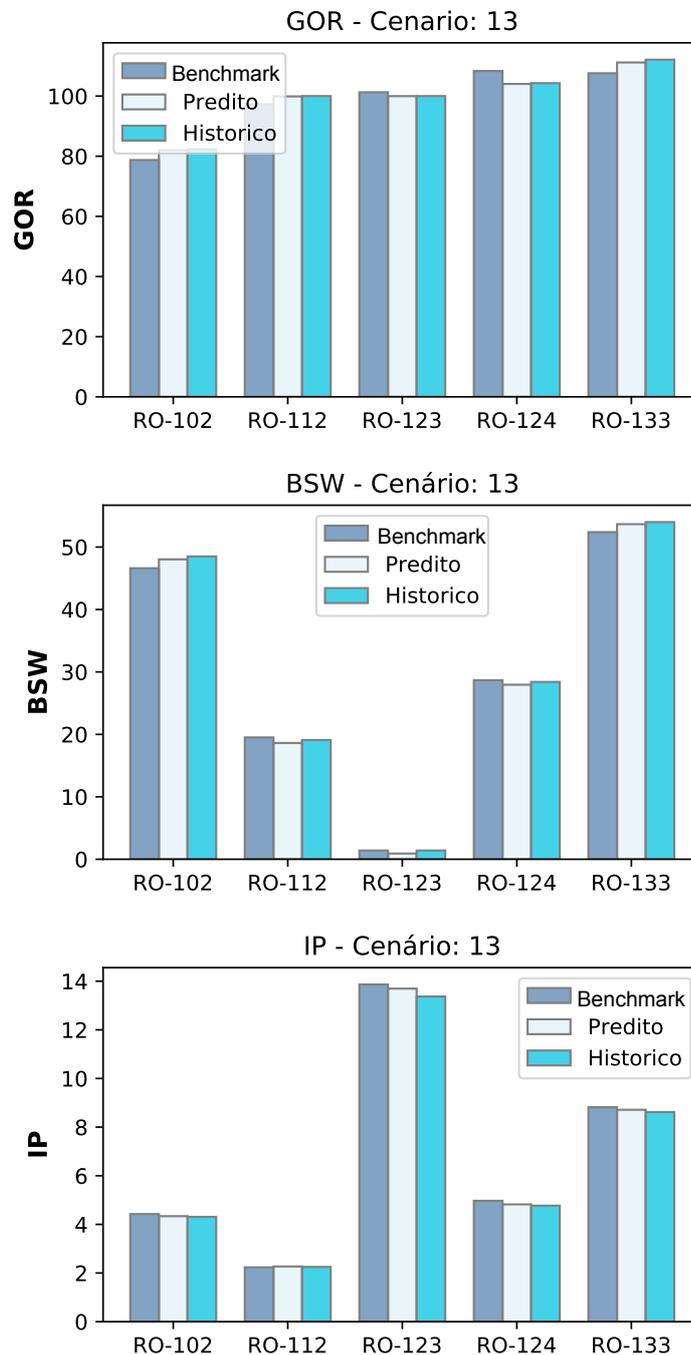
Com os parâmetros-chave encontrados pela metodologia, nomeados de  $RGO_{adj}^n$ ,  $BSW_{adj}^n$  e  $IP_{adj}^n$ , obtém-se separadamente a produção final de petróleo, utilizando os parâmetros sintonizados, históricos e *benchmark*. Para isso, baseado na arquitetura em blocos apresentada na Figura 22 um *script* Python é desenvolvido.

Nas próximas seções serão apresentadas cada um dos blocos da metodologia apresentada na Figura 22, que tem como objetivo encontrar o máximo de petróleo de cada poço em diferentes cenários, baseado em um modelo clássico de produção de petróleo apresentado nas equações (5.2a)-(5.3c).

### 5.5.1 Geração de Cenários

Os cenários são gerados no simulador de poços de petróleo MARLIM, no qual os parâmetros-chave sintonizado, histórico e *benchmark* são inseridos no simulador e o mesmo

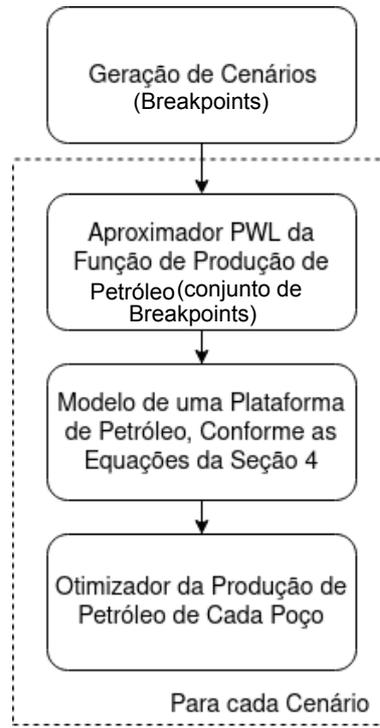
Figura 21 – Parâmetros-chave RGO, BSW e IP de um Poço para os casos *Benchmark*, Sintonizado e Histórico.



Fonte: O autor.

retorna, separadamente, os valores de controle  $q_{gl}^n$  (vazão de injeção de *gas-lift*),  $whp^n$  (pressão na cabeça do poço) e  $q_{oil}^n$  (óleo produzido). Os dados retornados pelo simulador são separados e organizados para os 3 casos (sintonizado, histórico e *benchmark*).

Figura 22 – Diagrama Ilustrativo da Etapa de Maximização da Produção de Petróleo.



Fonte: O autor.

### 5.5.2 Otimizador da Produção de Óleo

Com os parâmetros de controle encontrados em cada cenário, conforme explicado na seção 5.5.1, declara-se um modelo matemático MILP para cada caso (sintonizado, histórico e *benchmark*) pertencente ao cenário. Esses modelos contêm as restrições da operação de uma plataforma de petróleo, conforme as equações apresentadas na Seção 4, bem como conjuntos SOS2 empregados na aproximação linear por partes da função da vazão de produção. A função de líquido é representada por  $q_l^n(q_{gl-adj}^n, whp_{adj}^n, bsw_{adj}^n, rgo_{adj}^n, ip_{adj}^n)$ , para o caso em que os sinais de controle ( $q_{gl-adj}^n$  e  $whp_{adj}^n$ ) e os parâmetros-chaves ( $rgo_{adj}^n$ ,  $bsw_{adj}^n$  e  $ip_{adj}^n$ ) pertencem ao caso sintonizado. Uma vez definidas as restrições, o objetivo desse modelo é maximizar a produção de petróleo total ( $q_{oil-prod}$ ) da plataforma (poços contidos no cenário). Algebricamente, para o caso sintonizado o modelo pode ser representado conforme as equações abaixo, considerando um instante de tempo  $t \in T$ :

$$\max q_{oil-prod} \quad (5.2a)$$

$$\text{s.t.} : q_o^n + q_w^n = q_l(q_{gl-adj}^n, whp_{adj}^n, rgo_{adj}^n, bsw_{adj}^n, ip_{adj}^n) \quad (5.2b)$$

$$q_g^n = rgo_{adj}^n \cdot q_o^n \quad (5.2c)$$

$$q_w^n \cdot (1 - bsw_{adj}^n) = bsw_{adj}^n \cdot q_o^n \quad (5.2d)$$

$$q_{oil-prod} = \sum_{n \in \mathcal{N}} q_o^n \quad (5.2e)$$

$$q_{water-prod} = \sum_{n \in \mathcal{N}} q_w^n \quad (5.2f)$$

$$q_{gas-lift} = \sum_{n \in \mathcal{N}} q_{gl}^n \quad (5.2g)$$

$$q_{gas-prod} = \sum_{n \in \mathcal{N}} q_g^n \quad (5.2h)$$

$$q_{gas-exp} = q_{gas-prod} - q_{flare} - q_{turbine} \quad (5.2i)$$

$$q_{gas-prod} + q_{gas-lift} - q_{flare} \leq q_{gtc} \quad (5.2j)$$

$$q_{gas-exp} \leq q_{gas-exp}^{max} \quad (5.2k)$$

$$q_{flare}^{min} \leq q_{flare} \leq q_{flare}^{max} \quad (5.2l)$$

$$q_{oil-prod} + q_{water-prod} \leq q_l^{max} \quad (5.2m)$$

$$q_{water-prod} \leq q_w^{max} \quad (5.2n)$$

$$q_{oil-prod} \leq q_{oil}^{max} \quad (5.2o)$$

$$q_{gas-prod} \leq q_{gas}^{max} \quad (5.2p)$$

Nota-se no modelo acima que os sinais de controle ( $whp_{adj}^n$  e  $q_{gl-adj}^n$ ) não são fixos, como no modelo da sintonia de parâmetros-chave (restrições (4.6a) e (4.6b)), e tornando-se variáveis de decisões do modelo. As restrições apresentadas conforme as equações (4.7a), (4.7b) e (4.7c) são transformadas conforme as equações apresentadas abaixo, fixando os parâmetros  $RGO^n$ ,  $BSW^n$  e  $IP^n$  com os valores sintonizados.

$$rgo_{min}^n = rgo^n = rgo_{max}^n = rgo_{adj}^n \quad (5.3a)$$

$$bsw_{min}^n = bsw^n = bsw_{max}^n = bsw_{adj}^n \quad (5.3b)$$

$$ip_{min}^n = ip^n = ip_{max}^n = ip_{adj}^n \quad (5.3c)$$

As restrições (4.8a), (4.8b) e (4.8c) são removidas desse modelo, uma vez que os valores das vazões dependem dos valores ótimos. O modelo composto pelas equações (5.2a) - (??) declarado foi solucionado também para os casos históricos e *benchmark*, alterando as variáveis dos sinais de controle e parâmetros-chave.

### 5.5.3 PWL da Função de Produção de Líquido de Cada Poço

Ao aproximar a função da produção de líquido de cada um dos poços com funções lineares por partes, pode-se encontrar o ótimo global dessa função, utilizando métodos eficientes como *branch-and-bound*. Com os cenários gerados, conforme explicado na Seção 5.5.1, em cada um dos casos tem-se o  $q_{gl}$  e  $whp^n$  associado a uma produção de petróleo  $q_{oil}^n$ . Os conjuntos  $K_{gl} = \{1, \dots, k_{gl}\}$  e  $J_{whp} = \{1, \dots, j_{whp}\}$  contém os índices da vazão de injeção de gas-lift ( $q_{gl}^n$ ) e a pressão na cabeça ( $whp^n$ ), respectivamente. Considerando que  $Q_{gl}^n = \{q_{gl}^{n,k} : k \in K_{gl}\}$  e  $W_{whp}^n = \{whp^{n,j} : j \in J_{whp}\}$  sejam os conjuntos de dados gerados para  $q_{gl}^n$  e  $whp^n$ , pode-se

declarar a função de produção de líquido ( $q_l^n$ ) do caso sintonizado conforme propõem Seman et al. (2021):

$$Q_{liq}^n = \left\{ q_l^{n,k,j} : q_l^n(q_{gl-adj}^{n,k}, whp_{adj}^{n,j}, bsw_{adj}^n, rgo_{adj}^n, ip_{adj}^n), k \in K_{gl}, j \in J_{whp} \right\}, \quad (5.4)$$

onde obtém-se a produção de líquido dos poços variando a os sinais de controle ( $q_{gl-adj}$  e  $q_{gl-adj}$ ). Para os demais casos, utiliza-se os parâmetros *benchmark* e histórico.

Seja  $f : X \rightarrow \mathbb{R}$  uma função que mapeia do domínio  $X \subset \mathbb{R}$  para números reais. Seja ainda  $X' \subset X$  um conjunto de pontos ordenados  $x \in X'$ , tal que o valor  $f(x)$  é conhecido. Os valores da função  $f(x)$ , em que os elementos  $x \in X'$  são os *breakpoints*, são utilizados para aproximar a função  $f$ , gerando uma função linear por partes  $f'$  (SEMAN et al., 2021). Baseado no modelo SOS2 apresentado em 3.4.2, para cada poço  $n$ , a aproximação PWL é dada por:

$$q_{gl}^n = \sum_{k \in K_{gl}} \sum_{j \in J_{whp}} \theta_{k,j}^n \cdot q_{gl}^{n,k} \quad (5.5a)$$

$$whp^n = \sum_{k \in K_{gl}} \sum_{j \in J_{whp}} \theta_{k,j}^n \cdot whp^{n,k} \quad (5.5b)$$

$$q_l^n = \sum_{k \in K_{gl}} \sum_{j \in J_{whp}} \theta_{k,j}^n \cdot q_l^{n,k,j} \quad (5.5c)$$

$$1 = \sum_{k \in K_{gl}} \sum_{j \in J_{whp}} \theta_{k,j}^n \quad (5.5d)$$

$$\theta_{k,j}^n \geq 0, k \in K_{gl}, j \in J_{whp} \quad (5.5e)$$

Um conjunto de variáveis odernadas satisfaz propriedade SOS2 se, no máximo duas variáveis são não nulas, e se duas variáveis forem não nulas, elas são consecutivas na ordem, por exemplo  $\theta_k$  e  $\theta_{k+1}$ . Algumas restrições do tipo SOS2, são necessárias para garantir que a combinação combinação convexa induzida pelas variáveis de ponderação  $\theta_{k,j}^n$  induzem uma linearização por partes. Estas restrições são:

$$\eta_k^n = \sum_{j \in J_{whp}} \theta_{k,j}^n, k \in K_{gl} \quad (5.6a)$$

$$\eta_j^n = \sum_{k \in K_{gl}} \theta_{k,j}^n, j \in J_{whp} \quad (5.6b)$$

$$\{\eta_k^n\}_{k \in K_{gl}} \text{ is SOS2} \quad (5.6c)$$

$$\{\eta_j^n\}_{j \in J_{whp}} \text{ is SOS2} \quad (5.6d)$$

onde  $\eta_k^n$  e  $\eta_j^n$  são variáveis SOS2.

#### 5.5.4 Otimização da Produção de Líquido de cada Poço

Com a implementação PWL, baseada em conjuntos SOS2, apresentada na Seção 5.5.3, declarado para linearização por partes a função  $q_l(\cdot)$  e com o modelo da produção de petróleo,

apresentado na Seção 5.5.2, possibilita representar a função  $q_l(\cdot)$  em um modelo de otimização, de forma que o mesmo seja integrado no problema de otimização da produção de petróleo. O modelo de otimização da produção pode ser expresso matematicamente conforme apresentado nas equações abaixo.

$$\max q_{oil-prod} \quad (5.7a)$$

$$s.t. : q_{oil-prod} = \sum_{n \in \mathcal{N}} q_o^n \quad (5.7b)$$

$$q_{water-prod} = \sum_{n \in \mathcal{N}} q_w^n \quad (5.7c)$$

$$q_{gas-lift} = \sum_{n \in \mathcal{N}} q_{gl}^n \quad (5.7d)$$

$$q_{gas-prod} = \sum_{n \in \mathcal{N}} q_g^n \quad (5.7e)$$

$$q_{gas-exp} = q_{gas-prod} - q_{flare} - q_{turbine} \quad (5.7f)$$

$$q_{gas-prod} + q_{gas-lift} - q_{flare} \leq q_{gtc} \quad (5.7g)$$

$$q_{gas-exp} \leq q_{gas-exp}^{max} \quad (5.7h)$$

$$q_{flare}^{min} \leq q_{flare} \leq q_{flare}^{max} \quad (5.7i)$$

$$q_{oil-prod} + q_{water-prod} \leq q_l^{max} \quad (5.7j)$$

$$q_{water-prod} \leq q_w^{max} \quad (5.7k)$$

$$q_{oil-prod} \leq q_{oil}^{max} \quad (5.7l)$$

$$q_{gas-prod} \leq q_{gas}^{max} \quad (5.7m)$$

$$\text{Eqs. (5.5a) – (5.6d)} \quad (5.7n)$$

### 5.5.5 Projeção dos Sinais de Controle dos Poços

Os sinais de controle computados pela metodologia, que maximizam a produção de petróleo por meio da solução do problema dado em (5.7), podem ser aplicados diretamente em uma plataforma de produção de petróleo real, desde que sejam factíveis. No entanto, os modelos dos poços, dos casos sintonizado e histórico, desenvolvidos na geração de cenários (Seção 5.5.1) podem ser imprecisos, exigindo que alguma estratégia seja utilizada para avaliar os ganhos da metodologia proposta. A abordagem utilizada consiste em encontrar os sinais de controle factíveis para  $q_{gl}^n$  e  $whp^n$  que mais se aproximam dos sinais de controle encontrados pela metodologia ( $q_{gl-adj}^n$  e  $whp_{adj}^n$ ). Para isso, resolve-se o seguinte problema de projeção (SEMAN et al., 2021), considerando um único  $t \in T$ :

$$\min_{(q_{gl}, whp)} \sum_{n \in \mathcal{N}} [(q_{gl}^n - q_{gl, caso}^n)^2 + (whp^n - whp_{caso}^n)^2] \quad (5.8a)$$

$$s.t. : q_{oil-prod} = \sum_{n \in \mathcal{N}} q_o^n \quad (5.8b)$$

$$q_{water-prod} = \sum_{n \in \mathcal{N}} q_w^n \quad (5.8c)$$

$$q_{gas-lift} = \sum_{n \in \mathcal{N}} q_{gl}^n \quad (5.8d)$$

$$q_{gas-prod} = \sum_{n \in \mathcal{N}} q_g^n \quad (5.8e)$$

$$q_{gas-exp} = q_{gas-prod} - q_{flare} - q_{turbine} \quad (5.8f)$$

$$q_{gas-prod} + q_{gas-lift} - q_{flare} \leq q_{gtc} \quad (5.8g)$$

$$q_{gas-exp} \leq q_{gas-exp}^{max} \quad (5.8h)$$

$$q_{flare}^{min} \leq q_{flare} \leq q_{flare}^{max} \quad (5.8i)$$

$$q_{oil-prod} + q_{water-prod} \leq q_l^{max} \quad (5.8j)$$

$$q_{water-prod} \leq q_w^{max} \quad (5.8k)$$

$$q_{oil-prod} \leq q_{oil}^{max} \quad (5.8l)$$

$$q_{gas-prod} \leq q_{gas}^{max} \quad (5.8m)$$

$$\text{Eqs. (5.5a) – (5.6d)} \quad (5.8n)$$

onde o conjunto  $n \in N$  representa o conjunto de poços do cenário e o  $q_{gl, caso}^n$  e  $whp_{caso}^n$  representam os valores encontrados em cada cenário de gas-lift e pressão na cabeça para os casos históricos e sintonizados. Nota-se que o  $RGO^n$ ,  $BSW^n$  e  $IP^n$  referem-se apenas aos parâmetros-chave do caso *benchmark*, logo:

$$RGO^n = RGO_{benchmark}^n \quad (5.9a)$$

$$BSW^n = BSW_{benchmark}^n \quad (5.9b)$$

$$IP^n = IP_{benchmark}^n \quad (5.9c)$$

O modelo apresentado acima pode ser definido como um MIQP, no qual podem ser solucionados por meio de métodos da programação matemática.

## 5.6 EXPERIMENTOS

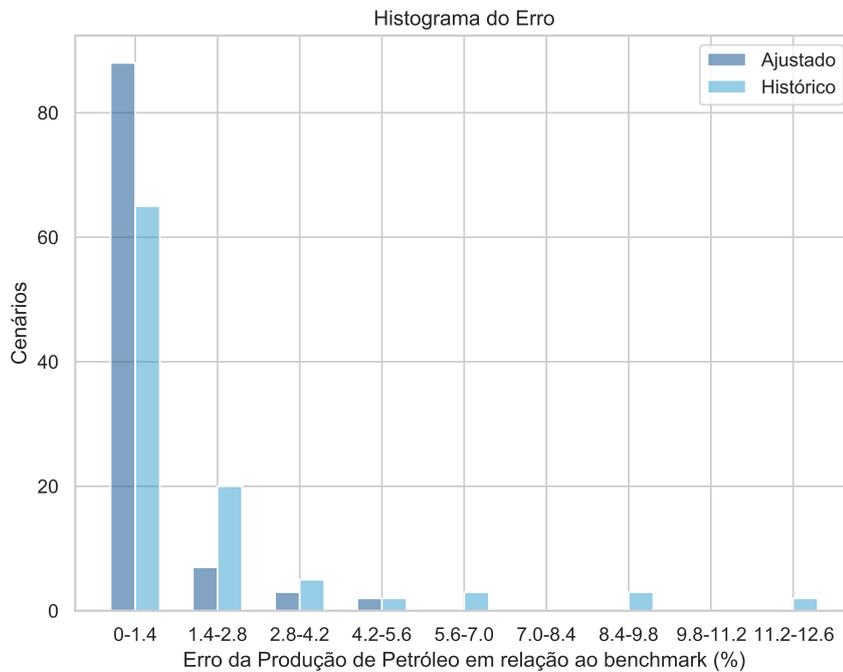
Com o objetivo de validar a metodologia de sintonia de parâmetros propõem-se os experimentos abaixo, em que para resolução dos problemas de otimização utilizou-se a biblioteca Gurobi no python.

### 5.6.1 Comparação de Cenários com Variação de 5%

A produção de petróleo final (todos os poços do cenário) obtida com os parâmetros-chave sintonizados (“ajustados”), encontrada pela metodologia proposta, é comparada com a produção final obtida com os parâmetros-chave históricos. Com base na Seção 5.3, gera-se 100 cenários, no

qual para cada cenário encontra-se o máximo de petróleo solucionando o problema apresentado na seção 5.5.2 para cada caso (sintonizado, histórico e *benchmark*), encontrando as produções de petróleo dos poços que compõem o cenário. No entanto, para encontrar pontos factíveis, conforme explicado na Seção 5.5.5, o problema de projeção de controle é solucionado para os casos sintonizado e histórico, utilizando parâmetros-chave *benchmark*, permitindo comparar as produções finais de petróleo dos casos. Na Figura 23 apresenta-se um gráfico do histograma do erro da produção de petróleo final dos casos (sintonizado e histórico) em relação ao caso *benchmark* (parâmetros-chave variados em até 5% em relação aos histórico), considerado caso de referência.

Figura 23 – Gráfico da Quantidade de Cenários em Relação às Faixas de Erros (entre Produção Final Ajustada e Histórica em Relação à Produção Final *Benchmark*).

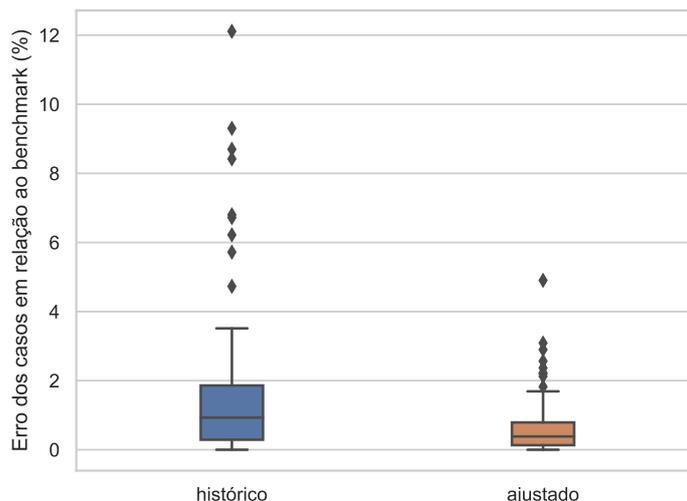


Fonte: O autor.

De acordo com o gráfico acima, empregando a metodologia de sintonia nos parâmetros-chave obtém-se vantagens, uma vez que dos 100 cenários gerados, mais de 80 cenários apresentaram um erro percentual entre a produção final de petróleo do caso sintonizado, em relação ao caso *benchmark*, na faixa de 0 – 1.4(%). A produção final de petróleo do caso histórico, em relação ao caso *benchmark*, apresentou em 60 cenários um erro no valor de 0 – 1.4(%), demonstrando que a produção de petróleo do caso sintonizado esteve mais próxima da produção do caso *benchmark* (referência) em uma maior quantidade de cenários. Além disso, considerando as outras faixas de erro a magnitude da quantidade de cenários do caso sintonizado é menor em relação ao caso histórico.

A partir do erro percentual entre a produção final de petróleo dos casos sintonizado e histórico, em relação à produção do caso *benchmark*, de cada cenário, um gráfico é gerado, conforme apresentado na Figura 24.

Figura 24 – Gráfico da Distribuição do Erro da Produção Final Ajustada e Histórica em Relação à Produção Final *Benchmark*.



Fonte: O autor.

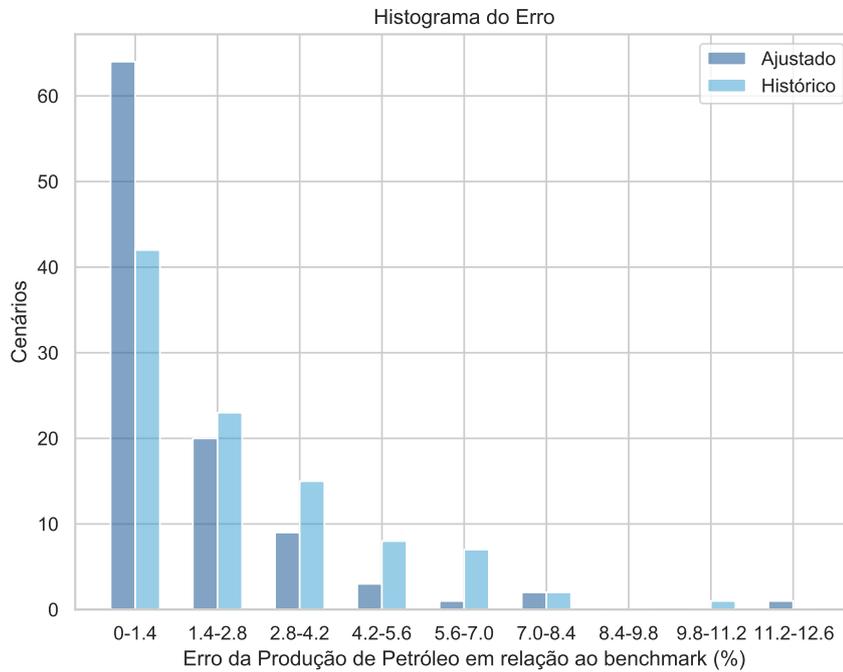
Nota-se que a mediana do erro percentual do caso sintonizado é menor em relação ao histórico, além dos erros serem menos dispersos. Com isso, a produção final obtida com os parâmetros-chave do caso sintonizado, ou seja encontrado pela metodologia proposta, aproximou-se mais da produção final do caso *benchmark* em relação à produção final obtida com os parâmetros-chave do caso histórico (obtidos no último teste de separação).

### 5.6.2 Comparação de Cenários com Variação de 10%

Com base na seção 5.6.1, compara-se a produção final de petróleo para os casos histórico e sintonizado, em relação ao caso *benchmark*, no qual os parâmetros-chave são obtidos com uma variação de 10 % em relação aos parâmetros-chave históricos. Na Figura 25 apresenta-se um gráfico do histograma do erro percentual da produção final de petróleo dos casos (sintonizado e histórico) em relação ao caso *benchmark*. Nota-se que, de acordo com o gráfico, a quantidade de cenários em que a produção final de petróleo do caso sintonizado, está na faixa de 0 – 1.4(%), diminui em relação ao experimento anterior, porém ainda assim há mais cenários em que o erro percentual é menor no caso sintonizado.

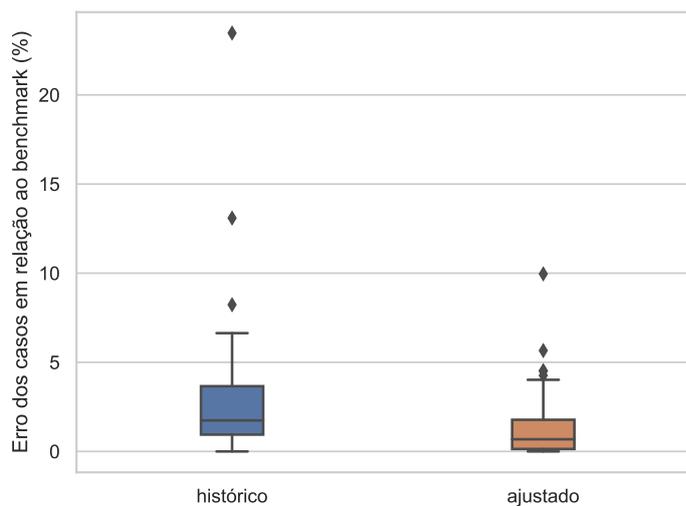
A partir do erro percentual entre a produção final de petróleo dos casos sintonizado e histórico, em relação à produção do caso *benchmark*, um gráfico é gerado com os erros percentuais dos 100 cenários gerados, conforme apresentado na Figura 26.

Figura 25 – Gráfico da Quantidade de Cenários em Relação às Faixas de Erros (entre Produção Final Ajustada e Histórica em Relação à Produção Final Benchmark).



Fonte: O autor.

Figura 26 – Gráfico da Distribuição do Erro da Produção Final Ajustada e Histórica em Relação à Produção Final Benchmark.



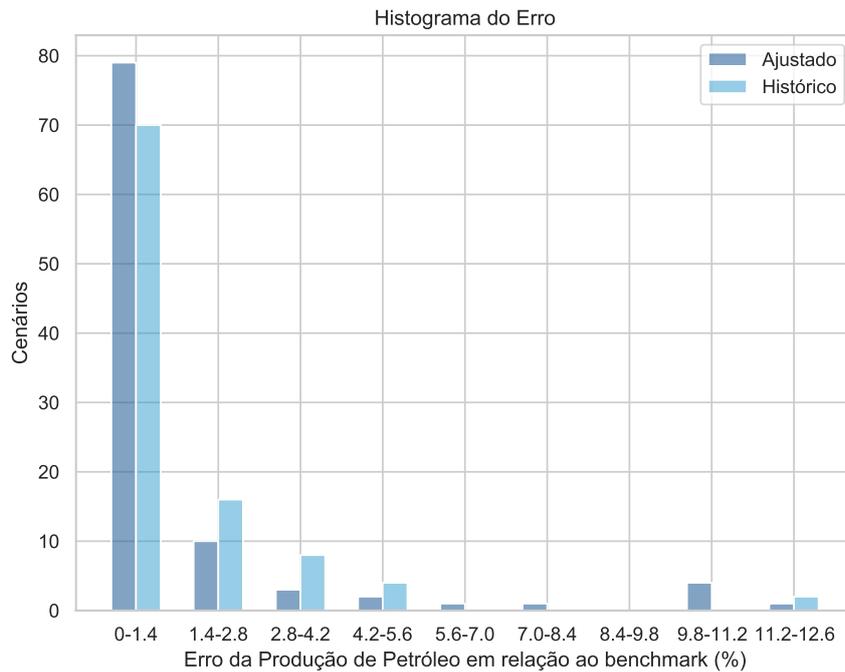
Fonte: O autor.

Nota-se que a mediana do erro percentual do caso sintonizado é menor em relação ao histórico, além dos erros serem menos dispersos.

### 5.6.3 Comparação Entre os Cenários Considerando Erros nas Medições

Considerando que possa haver medições imprecisas para o total de petróleo, gás e água, dos poços que estão conectados a uma plataforma, no qual são obtidos na geração dos cenários apresentado na Seção 5.5.1, nesse experimento as equações 4.3a, 4.3b e 4.3c poderá ter um erro aleatório de até 5 %. Os parâmetros-chave do caso *benchmark* são obtidos a partir de uma variação de 5 % dos parâmetros-chave históricos. Portanto, foram gerados 100 cenários novamente, para cada cenário compara-se a produção final de petróleo dos casos sintonizado e histórico, em relação ao caso *benchmark*, conforme os experimentos anteriores. Na Figura 27 apresenta-se um gráfico do histograma do erro da produção final de petróleo dos casos (sintonizado e histórico) em relação ao caso *benchmark*, considerando os erros nas medições de até 5 %.

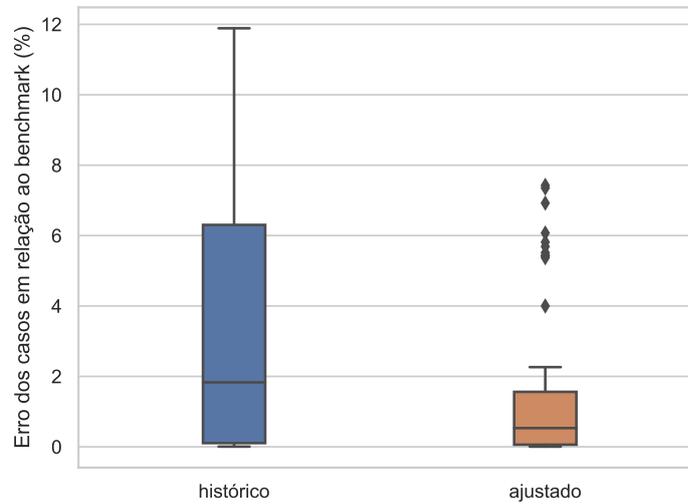
Figura 27 – Gráfico da Quantidade de Cenários em Relação às Faixas de Erros (entre Produção Final Ajustada e Histórica em Relação à Produção Final Benchmark).



Fonte: O autor.

A partir do erro percentual entre a produção final de petróleo dos casos sintonizado e histórico, em relação à produção final do caso *benchmark*, um gráfico é gerado com os erros percentuais dos 100 cenários gerados, conforme apresentado na Figura 28.

Figura 28 – Gráfico da Distribuição do Erro da Produção Final Ajustada e Histórica em Relação à Produção Final Benchmark.



Fonte: O autor.

Nota-se que a mediana do erro percentual do caso sintonizado é menor em relação ao histórico, além dos erros serem menos dispersos.

## 6 CONSIDERAÇÕES FINAIS

### 6.1 SUMÁRIO

Nesta dissertação apresentou-se uma metodologia para identificação de parâmetros-chave, tais como RGO, BSW e IP, de poços de petróleo conectados a uma plataforma, que compartilham um separador, e considerando restrições críticas da plataforma.

Alguns experimentos numéricos foram realizados para validação dos ganhos da metodologia desenvolvida, demonstrando que os parâmetros-chave sintonizados permitem uma maximização mais eficiente da produção de petróleo, em relação aos valores dos parâmetros-chave históricos (obtidos no último teste de separação). Cenários foram gerados para aplicação da metodologia, no intuito de sintonizar os parâmetros-chave em diferentes situações (cenários), e na maioria dos cenários gerados os parâmetros-chave sintonizados se aproximaram dos parâmetros-chave *benchmark*, ou seja apresentaram valores mais coerentes com os parâmetros considerados “reais”.

Com os parâmetros-chave *benchmark*, sintonizado e histórico encontra-se a produção de petróleo máxima, considerando restrições críticas das plataformas, para efeito de comparação. Para a maioria dos cenários a produção de petróleo utilizando os parâmetros-chave sintonizados apresentou um valor maior em relação a produção utilizando os parâmetros-chave históricos.

A metodologia desenvolvida possibilita manter os parâmetros-chave dos poços de petróleo atualizados, fazendo com que as condições operacionais dos poços reflitam as condições reais. Com isso, a metodologia pode auxiliar os operadores das plataformas a encontrarem valores de RGO, BSW e IP mais próximos dos valores reais, não necessitando esperar até o próximo teste de separação para atualizar os parâmetros-chave dos poços.

### 6.2 TRABALHOS FUTUROS

Como trabalho futuro sugere-se estudos envolvendo sistemas de produção mais complexos, considerando outros parâmetros-chave de poços e outras variáveis de medição como, por exemplo, pressões, para garantir que o modelo sintonizado esteja mais próximo da situação real. Além disso, pretende-se aplicar e avaliar essa metodologia em campos reais de petróleo.



## REFERÊNCIAS

- BEALE, E.; FORREST, J. Global optimization using special ordered sets. *Mathematical Programming*, v. 10, p. 52–69, 1976.
- BEASLEY, J. E. et al. Scheduling aircraft landings—the static case. *Transportation Science*, v. 34, p. 180–197, 2000.
- BORBA, B. S. et al. Energy-related climate change mitigation in Brazil: Potential, abatement costs and associated policies. *Energy Policy*, v. 49, p. 430–441, 2012.
- CAMPONOGARA, E.; NAKASHIMA, P. Optimizing gas-lift production of oil wells: piecewise linear formulation and computational analysis. *IIE Transactions*, v. 38, p. 173–182, 2006.
- CAMPONOGARA, E. et al. An automation system for gas-lifted oil wells: Model identification, control, and optimization. *Journal of Petroleum Science and Engineering*, v. 70, n. 3-4, p. 157–167, 2010.
- CASTRIGNANO, A. et al. *Agricultural Internet of Things and Decision Support for Precision Smart Farming*. [S.l.]: Academic Press, 2020.
- CATCHPOLE, A. R. The application of linear programming to integrated supply problems in the oil industry. *Journal of the Operational Research Society*, v. 3, p. 161–169, 2017.
- CHACHUAT, B.; SRINIVASAN, B.; BONVIN, D. Adaptation strategies for real-time optimization. *Computers and Chemical Engineering*, v. 33, p. 1557–1567, 2009.
- CLAUSEN, J. *Branch and Bound Algorithms – Principles And Examples*. 1999.
- COTTLE, R. W.; THAPA, M. N. *Linear and Nonlinear Optimization*. [S.l.]: Springer, 2017.
- DIEHL, F. C. *Controle Ativo de Golfadas em Poços de Petróleo Offshore*. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, 2022.
- FAHMI, I.; CREMASCHI, S. Process synthesis of biodiesel production plant using artificial neural networks as the surrogate models. *Computers & Chemical Engineering*, Elsevier, v. 46, p. 105–123, 2012.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <<http://www.deeplearningbook.org>>.
- GRIMSTAD, B.; ANDERSSON, H. ReLU networks as surrogate models in mixed-integer linear programs. *Computers & Chemical Engineering*, Elsevier, v. 131, n. 106580, 2019.
- GRIMSTAD, B. A. et al. Global optimization of multiphase flow networks using spline surrogate models. *Computers and Chemical Engineering*, v. 84, p. 237–254, 2016.
- GUNNERUD, V.; FOSS, B. Oil production optimization—a piecewise linear model, solved with two decomposition strategies. *Computers and Chemical Engineering*, v. 34, p. 1803–1812, 2010.
- HAYKIN, S. *Redes Neurais: Princípios e Prática*. [S.l.]: Tata McGraw-Hill Education, 2008.

HOFFMANN, A.; STANKO, M. Short-term model-based production optimization of a surface production network with electric submersible pumps using piecewise-linear functions. *Journal of Petroleum Science and Engineering*, v. 158, p. 570–584, 2017.

IBP. *Estudo mostra importância do setor de petróleo e gás para a economia do Brasil*. 2019. Available from Internet: <<https://www.ibp.org.br/noticias/estudo-mostra-importancia-do-setor-de-petroleo-e-gas-para-a-economia-do-brasil/>>.

KHAN, S. et al. Evaluation of an innovative hybrid gas lift technique. *SPE Artificial Lift Conference and Exhibition - Americas*, Day 3 Thu, November 12, 2020, 2020.

KOSMIDIS, V. D.; PERKINS, J. D.; PISTIKOPOULOS, E. N. Optimization of well oil rate allocations in petroleum fields. *Industrial & Engineering Chemistry Research*, v. 43, p. 3513–3527, 2005.

MARTINEZ, N. et al. Global optimization of non-convex piecewise linear regression splines. *Journal of Global Optimization*, Springer, v. 68, p. 563–586, 2017.

MISIC, V. V. Optimization of tree ensembles. *Optimization and Control*, arXiv, v. 1705.10883v2, 2019.

NAQA, I. E.; MURPHY, M. J. What is machine learning? In: NAQA, I. E.; LI, R.; MURPHY, M. J. (Ed.). *Machine Learning in Radiation Oncology: Theory and Applications*. Springer International Publishing, 2015. p. 3–11. ISBN 978-3-319-18305-3. Available from Internet: <[https://doi.org/10.1007/978-3-319-18305-3\\_1](https://doi.org/10.1007/978-3-319-18305-3_1)>.

PADBERG, M. Approximating separable nonlinear functions via mixed zero-one programs. *Operations Research Letters*, v. 27, n. 1, p. 1–5, 2000.

PETROBRAS. *A Cada Desafio, Novas Tecnologias*. 2015. Available from Internet: <<https://presal.petrobras.com.br/tecnologias-pioneiras/#7>>.

REDUTSKIY, Y. Conceptualization of smart solutions in oil and gas industry. *The 7th International Conference on Sustainable Energy Information Technology*, v. 109, p. 745 – 753, 2017.

RUMELHART, D.; HINTON, G.; WILLIAMS, R. Learning representations by back-propagating errors. *Nature*, v. 323, p. 533–536, 1986.

SAEPUDIN, D. et al. An investigation on gas lift performance curve in an oil-producing well. *International Journal of Mathematics and Mathematical Sciences*, v. 2007, p. 173–182, 2007.

SEMAN, L. O. et al. Tuning of oil well models with production data reconciliation. *Computers and Chemical Engineering*, v. 145, 2021.

SHARMA, S.; SHARMA, S. Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology*, v. 4, n. 12, p. 310–316, 2020.

SILVA, T. L. *Formulações inteiras mistas para modelos lineares por partes multidimensionais: aplicações na otimização da produção de petróleo sob restrições de pressão*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2012.

SILVA, T. L.; CAMPONOGARA, E. A computational analysis of multidimensional piecewise-linear models with applications to oil production optimization. *European Journal of Operational Research*, v. 232, p. 630–642, 2014.

- SILVA, T. L.; CODAS, A.; CAMPONOGARA, E. A computational analysis of convex combination models for multidimensional piecewise-linear approximation in oil production optimization. *IFAC Proceedings Volumes*, v. 45, p. 292–298, 2012.
- VIELMA, J. P. Mixed integer linear programming formulation techniques. *SIAM Review*, Society for Industrial and Applied Mathematics, v. 57, p. 3–57, 2015.
- VIELMA, J. P.; AHMED, S.; NEMHAUSER, G. Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Operations Research*, v. 58, p. 303–315, 2009.
- WOLSEY, L. A. *Integer Programming*. [S.l.]: John Wiley & Sons, Ltda, 2008.
- WUESTA, T. et al. Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research: An Open Access Journal*, v. 4, p. 23–45, 2016.