



# IMPLEMENTAÇÃO EM SISTEMAS DE AUTOMAÇÃO DE SENSOR VIRTUAL BASEADO EM REDES NEURONAIS

**Rafael Pinto Tiradentes**

## **Projeto de Final de Curso**

Orientadores

Andrea Valdman, D.Sc.

Maurício Bezerra de Souza Júnior, D.Sc.

**Janeiro de 2020**

# IMPLEMENTAÇÃO EM SISTEMAS DE AUTOMAÇÃO DE SENSOR VIRTUAL BASEADO EM REDES NEURONAIS

*Rafael Pinto Tiradentes*

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

Aprovado por:

---

Rossana Odette Mattos Folly, D.Sc.

---

Andréa Pereira Parente, D.Sc.

---

Manuella Cristina Correia de Sá, M.Sc.

Orientado por:

---

Andrea Valdman, D.Sc.

---

Maurício Bezerra de Souza Júnior, D.Sc.

Rio de Janeiro, RJ - Brasil  
Janeiro de 2020

Tiradentes, Rafael Pinto

Implementação em sistemas de automação de sensor virtual baseado em redes neuronais /  
Rafael Pinto Tiradentes. Rio de Janeiro: UFRJ/EQ, 2020.

xiii, 69 p.; il.

(Monografia) - Universidade Federal do Rio de Janeiro, Escola de Química, 2019.

Orientadores: Andrea Valdman e Maurício Bezerra de Souza Júnior.

1. Sensor Virtual. 2. Recuperação de amônia. 3. Rede Neuronal.

4. Monografia. (Graduação - UFRJ/EQ). 5. Andrea Valdman e Maurício Bezerra de Souza Júnior. I. Título.

*Que a tua vida não seja uma vida estéril. - Sê útil. - Deixa rasto. - Ilumina com o  
resplendor da tua fé e do teu amor.*

*Apaga, com a tua vida de apóstolo, o rasto viscoso e sujo que deixaram os  
semeadores impuros do ódio. - E incendeia todos os caminhos da terra com o fogo de  
Cristo que levas no coração.*

*Josemaria Escrivá, Caminho.*

*Por ela, eu me desfaceto em vários,*

*Por ela, eu me reparto em emoções,*

*Por ela, eu me dou por inteiro,*

*Por ela, eu me transformo em canções.*

*Rodrigo Pinto Tiradentes, O desfacetado e o telefone que não toca.*

## AGRADECIMENTOS

Agradeço primeiramente a Deus pelo dom da vida e por ter me conservado no caminho da justiça e da caridade. Também a minha mãe Maria santíssima pelo auxílio e pelas graças que me concedeu durante a execução deste trabalho. E a minha família por ter me feito cristão, dado todo o suporte necessário e ensinado a amar a Igreja.

De modo especial, a minha mãe e meu pai que doaram suas vidas e me ensinaram o valor do estudo. Aos meus avós pela valiosa intercessão. Ao meu irmão pela atenção e sugestões e a minha futura esposa Ana Carolina pela paciência e carinho.

A todos do LADEQ e do Laboratório de Instrumentação e Controle que trabalhei, por terem compartilhado o conhecimento e tantos momentos alegres. Especialmente, às professoras Andrea Valdman e Rossana Folly por terem me ensinado a amar a engenharia e me formado desde o início do curso nos âmbitos acadêmico e profissional.

Aos meus orientadores por toda atenção e dedicação. Àqueles que tornaram este trabalho possível, especialmente ao Leonardo da TI que com tanta bondade sempre me atendeu. Também à Andréa Parente, Andressa Nakao e Alisson Martins pela amizade, os ensinamentos e conselhos. À Maria, pelo sorriso e alegria de sua acolhida diária calorosa.

À FCCSA, meus supervisores Manuella Correia e Ricardo Silva, e todos os meus amigos da Otimização de Processos pela compreensão do tempo e esforço que foi dedicado a este trabalho. Em especial, à Renê Landim pela confiança neste projeto e pelas inúmeras orientações ofertadas.

Agradeço a banca pela disponibilidade em participar da avaliação deste trabalho e pelas correções e conselhos que engradeceram este trabalho.

Ao final, continua a difícil tarefa de sem falhas recordar tantas pessoas queridas que, de alguma forma, contribuíram para este trabalho. Meus sinceros agradecimentos, Hanna Tanaka, Guilherme Freire, Alexandre Augusto, Maria Carolina, Maria Clara, Fernanda Gama, Ana Carolina Rozendo, Marcus Lins, Prof<sup>o</sup> Paulo Eduardo, Matheus Ziderick, Alexandre Cury, Lucas Valentim, Débora Lage, Kese Pontes, André Alberton, João Malheiro e a todos da EQ Hands-On e da LiDA que tive oportunidade de trabalhar.

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

## **IMPLEMENTAÇÃO EM SISTEMAS DE AUTOMAÇÃO DE SENSOR VIRTUAL BASEADO EM REDES NEURONAIS**

Rafael Pinto Tiradentes

Janeiro, 2020

Orientadores: Prof<sup>ª</sup>. Andrea Valdman, D.Sc

Prof Maurício Bezerra de Souza Júnior, D.Sc

O acompanhamento de variáveis de processo é fundamental para o conhecimento e correção de condições operacionais em plantas industriais de processos químicos e bioquímicos. Embora os sensores comerciais e os modelos fenomenológicos forneçam uma solução convencional, a complexidade e não linearidade de alguns processos impõe dificuldades à medição de algumas variáveis de interesse. Em resposta a estes problemas, soluções baseadas em aprendizado de máquina – tais como redes neurais – são propostas. No atual contexto *big data*, historiadores de processo tornaram as indústrias modernas em um ambiente rico e propício para modelos preditivos baseados em dados. A literatura indica não somente a competitividade na resolução de problemas das redes neurais em relação aos métodos clássicos, como também sua capacidade generalista e a modelagem matemática clara e de fácil compreensão ao engenheiro de processos. É de grande importância o estabelecimento de metodologias de implementação destes modelos preditivos, alcançando benefícios como a redução de custos operacionais e o incremento da confiabilidade operacional. Este trabalho propõe metodologias para implementação e diagnóstico da qualidade de sensores virtuais *offline* e *online* baseado em redes neurais Multilayer Perceptron (MLP), que forneçam tanto a flexibilidade de implementação quanto a simplicidade de configuração pelo engenheiro de processos. A metodologia foi aplicada para o modelo de predição do teor de amônia na saída de fundo da coluna *stripper* da Fábrica Carioca de Catalisadores. Uma proposta de sensor *offline* e duas propostas de sensor *online* são apresentadas, utilizando softwares comerciais amplamente disponíveis em plantas industriais: planilha eletrônica e um historiador industrial de processos. As ferramentas foram validadas com os padrões da etapa de validação da rede neuronal, não indicando desvios significativos. O funcionamento dos sensores não apresentou interrupções inesperadas ou ciclos de cálculo superiores a 1 segundo. Testes com sensor *offline* alcançaram a taxa de cálculo de 1,7 milissegundo por padrão. Embora o foco deste trabalho esteja na proposta da metodologia para implementação do sensor virtual, a rede neuronal foi validada para o seu ano de treinamento, evidenciando correspondência dos resultados com as tendências operacionais. Testes de performance indicaram ainda a possibilidade de redução do intervalo de cálculo de treinamento (20 minutos) em um quarto, com ganhos de qualidade nos indicadores  $R^2$  e erro quadrático médio. Resultados para as atuais condições operacionais evidenciaram a necessidade de manutenção da rede neuronal.

# ÍNDICE

CAPÍTULO I — INTRODUÇÃO .....	1
I.1 Motivação .....	1
I.2 Objetivo .....	2
I.3 Organização do trabalho .....	3
CAPÍTULO II — REVISÃO BIBLIOGRÁFICA .....	4
II.1 Big Data .....	4
II.1.1 Caracterização.....	4
II.1.2 Desafios .....	5
II.1.3 Fábrica Inteligente .....	6
II.2 Automação de Processos .....	7
II.2.1 Evolução .....	7
II.2.2 Arquitetura de Automação.....	9
II.3 PIMS .....	12
II.3.1 Infraestrutura de Dados.....	13
II.3.2 Fluxo de Dados .....	13
II.3.3 Aplicativos para Análise de Dados.....	16
II.4 Redes Neurais .....	16
II.4.1 Processamento Matemático .....	17
II.4.2 Função de Ativação .....	18
II.4.3 Topologia.....	19
II.4.4 Treinamento.....	20
II.4.5 Rede Multilayer Perceptron (MLP) .....	21
II.5 Sensor Virtual .....	22

II.5.1	Implementação em Sistemas Adquiridos.....	24
II.5.2	Implementação em Sistemas Próprios .....	26
CAPÍTULO III	— METODOLOGIA.....	28
III.1	Descrição do Processo .....	28
III.1.1	Arquitetura de Automação Instalada.....	31
III.1.2	Sensor virtual de teor de amônia no fundo da coluna .....	35
CAPÍTULO IV	— ALGORITMO MODULAR.....	37
IV.1	Importação dos Parâmetros.....	37
IV.2	Predição .....	38
IV.2.1	Verificações.....	38
IV.2.2	Procedimento Matemático da Rede Neuronal.....	38
IV.2.3	Diagnóstico.....	41
IV.3	Reinicialização.....	41
IV.4	Validação .....	41
CAPÍTULO V	— SENSOR VIRTUAL <i>OFFLINE</i> .....	43
V.1	Desenvolvimento do Algoritmo e da Interface.....	45
V.2	Validação do Sensor <i>Offline</i> .....	46
V.3	Testes de Performance da Rede Neuronal .....	48
V.3.1	Qualidade da Predição .....	48
V.3.2	Dinâmica da Predição.....	52
CAPÍTULO VI	— SENSOR VIRTUAL <i>ONLINE</i> .....	56
VI.1	Desenvolvimento do Algoritmo e da Interface do sensor virtual em planilha eletrônica .....	56
VI.2	Desenvolvimento do Algoritmo e da Interface do sensor virtual na Interface Homem-Máquina.....	58
VI.3	Validação do Sensor <i>Online</i> .....	60



VI.4	Análise Estrutural dos Sensores <i>Online</i> e <i>Offline</i> .....	62
CAPÍTULO VII	— CONCLUSÃO .....	63
VII.1	Conclusões .....	63
VII.2	Sugestões para Trabalhos Futuros.....	64
REFERÊNCIAS BIBLIOGRÁFICAS	.....	65
APÊNDICE A1	.....	69

## ÍNDICE DE FIGURAS

Figura II-1 Etapas básicas em automação industrial para um engenheiro de processo.....	9
Figura II-2 Pirâmide da Automação. ....	10
Figura II-3 Modelo não-linear de um neurônio. ....	18
Figura II-4 Arquitetura de rede <i>feedforward</i> . ....	20
Figura III-1 Principais etapas do tratamento de efluente amoniacal. ....	28
Figura III-2 Representação esquemática da coluna. ....	30
Figura III-3 Ilustração do teste de exceção com seus parâmetros principais. ....	34
Figura III-4 Algoritmo de compactação de dados Swinging Doors. ....	35
Figura IV-1 Sequência de execução das sub-rotinas. ....	37
Figura IV-2 Módulos do algoritmo modular. ....	42
Figura V-1 Planilha desenvolvida para configuração dos parâmetros principais do sensor virtual.....	43
Figura V-2 Fluxograma das etapas de configuração do sensor <i>offline</i> . ....	44
Figura V-3 Tela desenvolvida para configuração da primeira etapa do sensor <i>offline</i> . ....	46
Figura V-4 Tela desenvolvida para configuração da segunda etapa do sensor <i>offline</i> . ....	47
Figura V-5 Resultado do teste de validação do sensor <i>offline</i> . ....	47
Figura V-6 Gráfico de dispersão entre o resultado do sensor virtual obtido pelo STATISTICA e obtido pela interface <i>offline</i> (PIMS/EXCEL) desenvolvida. ....	48
Figura V-7 Resultado do teste de aquisição e exportação de dados do sensor <i>offline</i> . ....	49
Figura V-8 Gráfico de tendência temporal dos resultados do sensor virtual e do analisador de amônia para o dia 05/06/2016. ....	50
Figura V-9 Gráfico de tendência temporal dos resultados do sensor virtual e do analisador de amônia para o dia 14/06/2016. ....	51
Figura V-10 Predição do analisador e do sensor virtual para 3 frequências diferentes para o dia 05/06/2016. ....	53
Figura V-11 Predição do analisador e do sensor virtual para 3 frequências diferentes para o dia 14/06/2016. ....	53
Figura VI-1 Tela de acionamento do sensor virtual <i>online</i> . ....	57
Figura VI-2 Tela de diagnóstico do sensor <i>online</i> . ....	57

Figura VI-3 Tela de configuração e acionamento do sensor <i>online</i> implementado na IHM. .....	59
Figura VI-4 Gráfico de tendência temporal dos resultados do sensor virtual e do analisador de amônia para o dia 08/01/2020.....	61

## ÍNDICE DE TABELAS

Tabela II-1 Comparação entre características típicas das redes MLP e RBF.....	22
Tabela III-1 Variáveis de processo da coluna de recuperação de amônia.....	31
Tabela IV-1 Principais funções de transferência para redes neuronais.....	40
Tabela V-1 Resultados dos parâmetros quantitativos do teste para o dia 05/06/2016. ....	54
Tabela V-2 Resultados dos parâmetros quantitativos do teste para o dia 14/06/2016. ....	54
Tabela VI-1 Resultados dos parâmetros quantitativos do teste para o dia 05/06/2016.....	60
Tabela A-1 Pesos e biases da rede neuronal MLP 10-12-1 para predição do teor de amônia do fundo da coluna. ....	69

# NOMENCLATURA

## LISTA DE ABREVIATURAS

<b>Sigla</b>	<b>Descrição</b>
<b>BCBS</b>	Boxcar/Backslope
<b>CLP</b>	Controlador Lógico Programável
<b>EPS</b>	Enterprise Production System
<b>ERP</b>	Enterprise Resource Planning
<b>IHM</b>	Interface Homem-Máquina
<b>MES</b>	Manufacturing Execution System
<b>MLP</b>	Multilayer Perceptron
<b>OPC</b>	Object Linking and Embedding for Process Control
<b>PIMS</b>	Plant Information Management System
<b>RNA</b>	Rede Neuronal Artificial
<b>SCADA</b>	Sistema de Supervisão e Aquisição de Dados
<b>SDCD</b>	Sistema Digital de Controle Distribuído
<b>RBF</b>	Função de Base Radial
<b>DLL</b>	Dynamic Link Library
<b>FCC</b>	Fábrica Carioca de Catalisadores
<b>SP</b>	Set Point
<b>LIMS</b>	Laboratory Information Management System

## LETRAS LATINAS

<b>Símbolo</b>	<b>Descrição</b>
$w_{kj}$	Peso da sinapse que liga o neurônio j ao neurônio posterior k
$u_k$	Resultado da junção aditiva das entradas do neurônio k
$x_j$	Sinal da entrada j de um neurônio k
$b_k$	Bias do neurônio k
$v_k$	Ativação do neurônio k
$F$	Função de ativação
$y_k$	Amplitude de saída do neurônio k
$C_j$	Valor central da entrada j

## LETRAS GREGAS

<b>Símbolo</b>	<b>Descrição</b>
$\sigma_k$	Desvio padrão do neurônio k.

# CAPÍTULO I — INTRODUÇÃO

## I.1 Motivação

A modelagem de processos químicos é um instrumento fundamental para entender, e até prever, o comportamento dinâmico de variáveis-chave de uma planta industrial. Embora a abordagem fenomenológica inicial apareça como o método convencional, baseado no balanço mássico e energético do volume de controle, a complexidade e não linearidade de diversos sistemas impõem dificuldades à aplicação da mesma (Baughman e Liu, 1995). Neste cenário, soluções baseadas em análises de grande série de dados – tais como algoritmos de aprendizado de máquina – foram buscadas e se tornaram grandes aliadas na busca pela eficiência operacional, qualidade do produto, segurança do processo e proteção do meio ambiente (Shu *et al.*, 2016; Qin, 2014).

Especialmente, redes neuronais artificiais (RNAs) surgiram como uma abordagem capaz de não somente simular a resposta dinâmica do sistema, mas também considerar em seu modelo a instabilidade e a operação em estado não estacionário, tão comuns na indústria química (Baughman e Liu, 1995). Sua modelagem matemática clara, de fácil compreensão para engenheiros químicos, e a capacidade generalista e flexível das mesmas foram atestadas diversas vezes na literatura. Da mesma forma, se provaram competitivas na resolução de problemas frente aos métodos tradicionais. Pela união dessas virtudes se justifica, em muitos casos, uma relação custo-benefício favorável à utilização das redes em diversas situações, tais como: detecção de falhas, otimização de processos e sensores virtuais (Åström & McAvoy, 1992; Prieto *et al.*, 2016).

Apesar do grande potencial da ferramenta, sua premissa baseada essencialmente na coleta de uma grande quantidade de dados – e na sua qualidade – impôs, de início, um entrave à sua exploração em diversos setores marcados pela escassez dos mesmos. Plantas industriais, por outro lado, são atualmente fontes ricas em dados, uma vez que diversos fatores são monitorados continuamente (Venkatasubramaniam, 2009; Qin, 2014). Com efeito, o desenvolvimento dos sistemas de controle distribuído e sistemas supervisórios industriais forneceram as séries temporais que o estudo de diversas redes em âmbito acadêmico demandava (Valdman, 2010; Prieto *et al.*, 2016).

De toda forma, estes sistemas são dedicados ao controle e supervisão do processo. O atual processo de digitalização em plantas químicas insere em seus sistemas servidores dedicados especialmente ao historiamto de dados de processo e análises avançadas, favorecendo a descentralização das funções de automação e o ambiente propício para a implementação de algoritmos complexos calculados em tempo real. Dentre esses, destacam-se os *softwares* do tipo PIMS (*Plant Information Management System*). Isto colocou a indústria de processos químicos em posição de pioneirismo na adoção de ferramentas orientadas a dados (Chiang, Lu, Castillo, 2017; James, 2012).

A esse atual processo se une o amadurecimento de diversas tecnologias, apresentando-se prontas para aplicação no ambiente industrial (James, 2012). Ao se tratar de sensores virtuais on-line, redes neurais do tipo *multilayer perceptron* (MLP) destacam-se nesse processo por sua simplicidade e capacidade de generalização frente às demais. Landim (2016) propõe ainda a transportabilidade de redes MLP entre diferentes modelos preditivos para unidades de processo, alcançando boa capacidade preditiva nos casos avaliados.

A implementação de tais redes neurais para o monitoramento de variáveis-chave de processo em tempo real de difícil medição, portanto, é vista nesse cenário como potencial redutora de custos operacionais e de ganhos em confiabilidade operacional (Fortuna *et al.*, 2007). Entre o potencial apresentado na literatura e a aplicação do mesmo na indústria estão, entre os demais fatores, os engenheiros de processo. Como usuários-chave, tanto para desenvolvimento das redes quanto para interpretação do resultado, é necessário que a instância de automação utilizada para implementá-las suporte o processamento computacional associado e seja amigável e adequada às atividades das equipes de engenharia da planta (James, 2012).

## **I.2 Objetivo**

Este trabalho teve por objetivo principal desenvolver ambientes e metodologias para implementação de sensores virtuais, em *software* PIMS, baseados em redes neurais artificiais do tipo MLP. Foi colocada como premissa a escolha de aplicativos clientes amigáveis e disponíveis para configuração pelo engenheiro de processos.



Dentre os objetivos específicos estão:

- Desenvolvimento de um algoritmo modular adaptável quanto à implementação de rede neuronal MLP em diferentes aplicativos clientes;
- Desenvolvimento de ferramentas de diagnóstico do resultado predito;
- Desenvolvimento do sensor *offline* em planilha eletrônica;
- Desenvolvimento do sensor *online* em planilha eletrônica e interface homem – máquina do sistema PIMS;
- Implementação e validação dos sensores e ferramentas de diagnóstico desenvolvidos para predição de teor de amônia em fundo de coluna de recuperação de amônia.

### **I.3 Organização do trabalho**

Os capítulos deste trabalho são dispostos da seguinte forma:

A revisão bibliográfica (capítulo II) discute inicialmente a relação entre o uso e disponibilidade de dados na indústria com a arquitetura de automação apresentada na literatura. Em seguida, apresenta os fundamentos teóricos relacionados ao funcionamento do sensor virtual e à topologia das redes neuronais.

A metodologia (capítulo III) descreve brevemente o processo, a arquitetura de automação e a rede neuronal utilizados como estudo de caso. As etapas executadas no projeto são resumidas neste capítulo e apresentadas em detalhes, com seus resultados, nos 3 capítulos posteriores.

O capítulo IV apresenta o funcionamento de cada módulo do algoritmo desenvolvido e o detalhamento do procedimento matemático da rede neuronal MLP. Em seguida, o capítulo V apresenta o desenvolvimento e validação do sensor *offline* em planilha eletrônica comentando as adaptações feitas no algoritmo modular. Realiza e discute resultados de teste de performance do sensor virtual com a rede neuronal de Landim (2016). Por fim, o capítulo VI apresenta o desenvolvimento e validação do sensor virtual *online* em planilha eletrônica e na interface homem-máquina do sistema PIMS.

A conclusão (capítulo VII) encerra o trabalho com as considerações finais e sugestões para trabalhos futuros.

## CAPÍTULO II — REVISÃO BIBLIOGRÁFICA

### II.1 Big Data

Os recentes avanços nos campos da internet, dispositivos móveis, comunicação sem fio, internet das coisas, entre outros fatores, tornam notório o aumento abrupto de dados adquiridos e circulantes na sociedade atual. Ademais, as novas formas de coletá-los, armazená-los e processá-los que se sucederam – como os dispositivos digitais inteligentes, a computação em nuvem e a inteligência artificial – revelam também um estímulo ao seu uso (Chiang, Lu, Castillo, 2017; Qin, 2014).

No cenário industrial, esses dados advêm atualmente das centenas de sensores e atuadores que tornaram as plantas largamente instrumentadas. Antes mesmo da larga adoção de sensores inteligentes e da internet das coisas industriais, é possível ver que a quantidade de dados disponíveis nas salas de controle e de engenharia alcança um volume tal que sobrecarrega os funcionários. Assim, a análise de dados se torna escassa e, em geral, aparece somente para detecção de falhas e em caso de emergência (Chiang, Lu e Castillo, 2017; Qin, 2014).

#### II.1.1 Caracterização

A definição de *big data* segundo a abordagem de Qin (2014) consiste em “um conjunto de dados com tamanho e variedade desafiantes para as ferramentas tradicionais de captura, armazenamento, gerenciamento e análise”. Chiang, Lu e Castillo (2017) apresentam uma forma clara de identificar o ambiente *big data* através de três características:

1. Volume: grande quantidade de dados gerada e coletada.
2. Velocidade: necessidade para coletar e processar mais rapidamente grandes volumes de dados.
3. Variedade: contextualização de todos os tipos de dados, incluindo os estruturados e não estruturados.

Embora seja possível enquadrar a indústria química neste ambiente, esse ainda é um tema emergente na indústria de processos químicos. O crescente interesse de empresas nacionais por termos como indústria 4.0, inteligência artificial, *machine learning* e

digitalização evidencia o espaço que as estratégias baseadas fortemente em análise de dados ainda têm para crescer (Chiang, Lu e Castillo, 2017).

Apresenta-se de fato um novo paradigma para a história, a mesma que evidencia a indústria química como uma das pioneiras no uso de controle computacional, muito em parte pela razão de que a operação segura e eficiente da planta requer o monitoramento de variáveis de processo analógicas e discretas, normalmente abundantes. A literatura não defende que *big data* seja uma resposta universal, contudo enxerga os dados como ativos indispensáveis para tomadas de decisão (especialmente em condições afastadas do projeto) mais assertivas em tempo real (Chiang, Lu e Castillo, 2017; Qin, 2014).

Diversos exemplos têm apontado a posse de uma enorme quantidade de dados como fonte de grande vantagem se aliada a grande poder de processamento computacional e métodos eficazes de análise dos mesmos (Qin, 2014). Dessa forma, encontramos uma segunda definição do termo *big data* como sendo uma jornada para transformar dados em soluções intuitivas para a informação de negócios e decisões operacionais (Chiang, Lu e Castillo, 2017).

Nessa jornada, a instância operacional se beneficia por ganhar novos recursos para lidar com processos complexos e matérias primas variadas. Pode-se alcançar também um controle com múltiplos objetivos, agregando mais valor aos dados armazenados e provindo da instrumentação inteligente. Não obstante, na instância gerencial, o ambiente *big data* cria valor para a empresa ao tornar os dados coletados (mais acurados e detalhados) em informação transparente e utilizável para o desenvolvimento de novos produtos e serviços personalizados ao cliente (Qin, 2014).

### **II.1.2 Desafios**

O estudo para implementação dos novos métodos de análise, em especial o aprendizado de máquina e a inteligência artificial, teve de lidar com diversas dificuldades. O pré-processamento dos dados requereu especial atenção, uma vez que as abordagens mais tradicionais e bem estabelecidas para indústria o exigem (Prieto *et al.*, 2016; Qin, 2014).

Dentre os desafios mais comumente encontrados encontra-se a limitação de dados para aprendizado como expoente. Em muitos casos, encontra-se disponível uma quantidade de

dados inferior à necessária, o que requisita estratégias adicionais. Técnicas adicionadas à metodologia básica de inteligência artificial aparecem para enfrentar outros empecilhos, tais como dados não balanceados (percentual muito superior de amostras em uma condição ou região do espaço amostral em detrimento das outras), dados incompletos (variáveis desconhecidas ou séries temporais perdidas) e alta dimensionalidade. Este último, caracterizado muitas vezes pelo envolvimento de variáveis em excesso no modelo, destaca-se ao longo da história recente (Prieto *et al.*, 2016).

Com o advento do *big data* novas demandas precisam ser solucionadas. A avaliação da maior magnitude dos dados depende das habilidades de distinguir significado dos mesmos e identificar sua escassez para o aprendizado. Depende também da habilidade de interagir com a veracidade dos dados, isto é, com a incerteza do dado coletado. Mais uma vez, a velocidade é o ponto crucial para diminuir o ciclo de tomada de decisão, passando necessariamente pela seleção da frequência ideal de captura de dados (Chiang, Lu, Castillo, 2017).

Por fim, validar o conhecimento já existente, obtido da combinação das múltiplas fontes de dados, é essencial. Isto exige a administração das múltiplas fontes de coleta de forma eficiente para não se perder no denominado “dilúvio de dados”. Ultrapassar a era de informação limitada a longos experimentos e simulações de processo clama à indústria, portanto, por uma estrutura física e computacional orientada às novas necessidades (Venkatasubramaniam, 2009).

### **II.1.3 Fábrica Inteligente**

A influência e imersão da tecnologia computacional pode ser encontrada facilmente no cotidiano de médias e grandes cidades de todo o mundo. Ademais, o conceito da indústria 4.0 surge da necessidade de estabelecer abordagem similar nas fábricas (James, 2012).

A alteração profunda nas práticas de operação, marcada por dispositivos inteligentes conectados por redes, e no modelo de negócios, baseado fortemente na integração da produção, logística e o gerenciamento de toda a cadeia de valor, são fatores anunciados ao se abordar o tema da quarta revolução industrial (Li, 2016). A fábrica inteligente é vista como estandarte da indústria 4.0.

O objetivo da fábrica inteligente é a operação autônoma através de sensores, atuadores e dispositivos dotados de suficiente poder de processamento e capazes de agregarem-se a sistemas inteligentes auto organizados (James, 2012). Do ponto de vista da indústria 4.0, ela é centrada na excelência operacional garantida, dentre outros aspectos, pela previsão e otimização do processo. Sua fundamentação está na união da automação, digitalização, visualização da informação, modelagem dos dados e integração de sistemas (Li, 2016).

O sistema físico-computacional indicado pela literatura para lidar com esse ambiente dinâmico é baseado na informação, na comunicação tecnológica, e na computação distribuída. São os denominados sistemas cyber-físicos, capazes de abordar em conjunto os três aspectos atingidos separadamente na arquitetura atual: os dados (séries temporais de cada variável), a informação (relação entre variáveis) e o gerenciamento do conhecimento (modelos que explicam a informação) (Venkatasubramaniam, 2009).

É preciso ressaltar que o uso de tecnologias como redes sem fio e celulares em ambiente industrial ainda são fonte de discussão quanto à sua segurança e confiabilidade (Parente *et al.*, 2018). Atualmente, o quantitativo de fábricas realmente inteligentes é pequeno e as que são de fato têm, em maioria, sua tecnologia inteiramente baseada em um único fornecedor (James, 2012). Diante deste cenário, para uma aplicação imediata da inteligência artificial na indústria é preciso conhecer a arquitetura de automação atualmente implementada.

## **II.2 Automação de Processos**

### **II.2.1 Evolução**

A história da automação de processos é recente e acompanha a evolução natural da indústria de processos, que foi ganhando complexidade aos longos dos anos e por isso foi demandando cada vez mais um controle preciso da produção. Sua atuação contemporânea vai além da regulação das condições operacionais, atingindo a otimização do processo, planejamento da produção e a integração de negócios (Parente, 2019).

Como marco crucial da história moderna, podemos apontar a década de 1940, onde surgem os primeiros instrumentos pneumáticos. A operação do processo, até então essencialmente manual, passa a ser equipada com tubulações específicas para transmissão de informação (Gutierrez e Pan, 2008).

O avanço da década de 1940 permite a criação da sala de controle do processo, reunindo em si os controladores. Nas duas décadas seguintes, a distância permitida entre a sala de controle e os elementos do campo aumenta com a introdução da eletrônica analógica (Gutierrez e Pan, 2008). É no final deste período que aparecem os transistores, conferindo grande impulso para a automação, e no qual se inicia a transição para sistemas de controle digital (Souza, 2005; Colombo *et al.*, 2014).

Na década de 1970 surgem os sistemas de controle distribuídos. Em especial, foram lançados os primeiros CLPs (Controladores Lógico Programáveis) e, pouco tempo depois, os primeiros SDCDs (Sistemas Digitais de Controle Distribuído). Os primeiros, substituem o uso de relés, temporizadores e sequenciadores mecânicos, enquanto os segundos surgem com a finalidade de ser um sistema integrado para monitorar múltiplos subsistemas. Em muitos casos, devido ao potencial tecnológico de cada um desses sistemas, se manteve a utilização em conjunto dos mesmos (Gutierrez e Pan, 2008).

É ainda na década de 1970 que ocorre o desenvolvimento de microprocessadores. Em virtude disto, na década de 1980 os computadores digitais são introduzidos nas fábricas agregando as funções de troca de informação em rede. Os painéis sinópticos e mesas de controle são substituídos por computadores, contendo muitas vezes sistemas supervisórios. Naturalmente, é neste período que surgem os Sistemas de Supervisão e Aquisição de Dados (SCADA) (Souza, 2005; Valdman, 2010).

A adoção da comunicação em rede gera significativos esforços para padronizar o barramento de campo. Assim, na década de 1990 surge a instrumentação baseada em redes industriais padronizadas (instrumentação *Fieldbus e Profibus*). Com o avanço da tecnologia de microprocessadores nas décadas seguintes, os instrumentos são dotados de “inteligência”, ou seja, ganhando capacidade de processamento digital (Souza, 2005; Gutierrez e Pan, 2008).

Em paralelo ao desenvolvimento da instrumentação, plataformas operacionais baseadas em sistema Windows da Microsoft são desenvolvidas (Valdman, 2010). Através delas foi se promovendo a interligação dos sistemas de monitoramento e controle de processos com os sistemas de gestão da empresa, especialmente com uso da rede Ethernet (Gutierrez e Pan, 2008). Novas camadas de automação focadas em traduzir os dados de

processo em informação gerencial surgem, tendo os EPSs (*Enterprise Production System*) e ERP (*Enterprise Resource Planning*) como seus expoentes (Souza, 2005).

## II.2.2 Arquitetura de Automação

A interligação de sistemas cria um fluxo de informações que conecta o processo aos sistemas corporativos de mais alto nível. Esse fluxo percorre não somente etapas em série (ascendentes e descendentes), mas também em paralelo, e acaba por apresentar uma hierarquia bem definida (Valdman, 2010).

Do ponto de vista da engenharia de processo, a Figura II-1 ilustra a transmissão de informação focando nos elementos de controle.

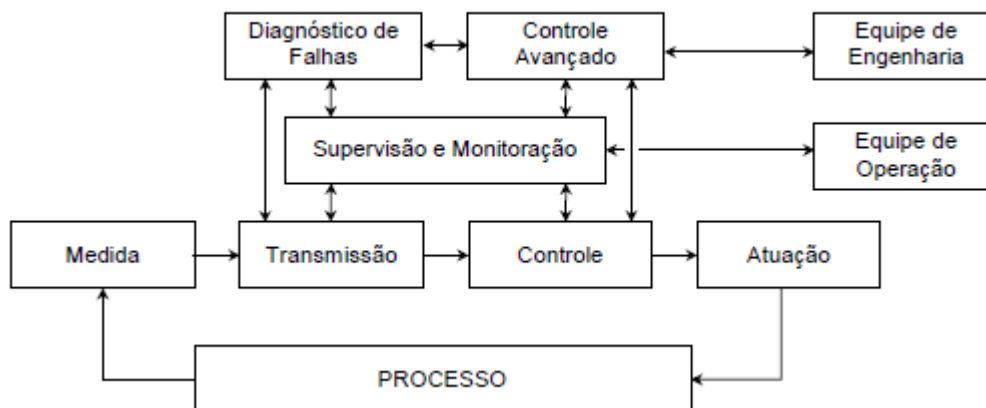


Figura II-1 Etapas básicas em automação industrial para um engenheiro de processo.

Fonte: Valdman, 2010.

A primeira instância de qualquer sistema de automação é o próprio processo ao qual se relaciona. Nele ocorrem os fenômenos físicos, químicos ou bioquímicos, as transferências de massa e energia e também os fenômenos mecânicos dos equipamentos; a fonte da maior parte dos dados está ali representada e disponível para serem analisadas e manipuladas pela hierarquia superior (Valdman, 2010).

Acima temos a instrumentação do processo, responsável pelo ciclo contínuo de controle através de sensores (medição das variáveis de processo), transmissores (conversão

do sinal do sensor para outro adequado à transmissão), controladores (tomada de decisão na magnitude da manipulação) e atuadores (execução dos comandos do controlador na variável manipulada). Esses elementos são indispensáveis para o sistema de automação da empresa, de forma que a qualidade da informação obtida e transmitida neles é fundamental para a análise correta das equipes de operação e engenharia e da validade dos relatórios gerenciais elaborados (Valdman, 2010, Souza, 2005).

Acima da instrumentação está a supervisão e monitoração do processo. Ela é a responsável por acompanhar o ciclo da instância inferior em tempo real, coletando seus dados e os dispondo em interfaces homem-máquina (IHMs). Por meio destas que a equipe de operadores monitora, identifica, quantifica os fenômenos e atua no processo (Valdman, 2010; Souza, 2005).

É também possível que a equipe de engenharia se sirva dessa instância para acompanhar o processo e trabalhar sua informação. A Figura II-2 apresenta a pirâmide organizacional da automação baseadas na norma IEC 62264 (2007) – ou originalmente ISA-95 – onde as camadas acima do controle operacional da planta – focadas no gerenciamento da informação – são ressaltadas (Colombo *et al.*, 2014).

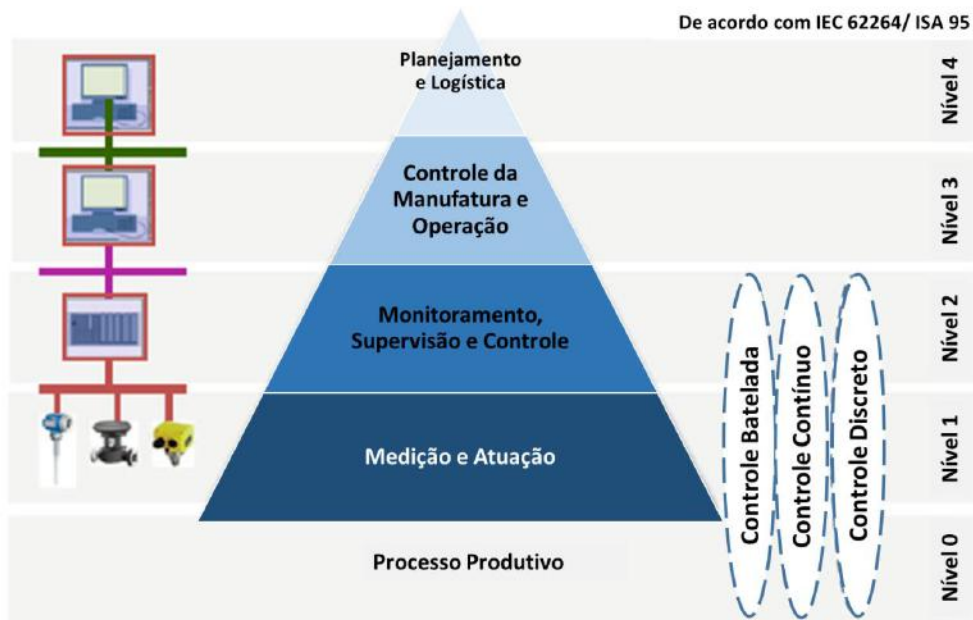


Figura II-2 Pirâmide da Automação.

Fonte: Colombo *et al.*, 2014 (Adaptado).



Na pirâmide da automação proposta por Colombo *et al.* (2014) o processo produtivo está no nível 0 e a instrumentação do processo no nível 1. O controlador, por sua vez, destacado dos atuadores e sensores em instância superior, é representado no mesmo nível da supervisão do processo. Nessa representação os elementos CLP, SCADA e SDCD são incluídos juntos, assim como muitas vezes se encontram em indústrias de processo, no nível 2.

Tradicionalmente, o controle discreto é realizado no CLP enquanto o SDCD ou o SCADA são mais adequados para o desenvolvimento de IHMs, configuração de alarmes, atuação no processo e coleta direta dos dados de processo (garantindo integridade do dado e as taxas apropriadas de frequência de captura).

No nível 3 se encontram os softwares de controle de produção do tipo EPS, em especial o MES (*Manufacturing Execution System*) e o PIMS (*Plant Information Management System*). São responsáveis pelo gerenciamento de toda a cadeia de produção, concentrando todas as informações relevantes da planta em um só lugar. Estes sistemas coletam informações dos SCADAs, SDCDs, PLCs e demais sistemas de supervisão, controle e informação da empresa, preparando grandes bases de dados históricos. Em seguida, manipulam a informação de maneira adequada para a tomada de decisões estratégicas de caráter operacional, econômico e financeiro (Souza, 2005).

A última instância é a logística e planejamento do negócio. Neste nível, dentre os diversos sistemas existentes, o ERP se destaca fortemente no mercado ao transformar os dados dos níveis anteriores em informação de negócio. Esse amplo sistema multimodular, integrado em uma base de dados única, interage com um conjunto integrado de aplicações nos campos de fabricação, compras, estoque, logística, finanças, dentre outros (Souza, 2005; Lamb, 2013).

O fluxo de informações pela pirâmide da automação depende da comunicação entre os diferentes níveis. Entre os níveis 1 e 2 se destacam o sinal analógico 4 a 20 mA, a Ethernet e as redes de campo, em especial a rede Fieldbus (termo genérico para os protocolos de alta velocidade, baseados em redes de transmissão digital). A comunicação entre os níveis 2, 3 e 4, por sua vez, é usualmente realizada através de protocolo OPC (Colombo *et al.*, 2014; Parente *et al.*, 2018).

## II.3 PIMS

Progressos em controle avançado, baseados em métodos mais complexos, requerem o desenvolvimento de sistemas robustos capazes de trocar informações com níveis inferiores na hierarquia, usando ou não a instância de supervisão e monitoramento (Valdman, 2010). Engenheiros de processo necessitam de ferramentas de acompanhamento de processo mais robustas – independentemente da fonte de dados – para dados em tempo real e históricos (Carvalho *et al.*, 2005).

Atendendo as demandas, surgiram os *softwares* PIMS, que segundo Carvalho *et al.* (2005) são sistemas de aquisição de dados que essencialmente recuperam os dados de processo residentes em fontes distintas, os armazenam em banco de dados único e os disponibilizam através de suas ferramentas clientes. Propõem-se a gerenciar as informações de produção para transformar a massa de dados em conhecimento (Souza, 2005).

Dedicando esse nível da pirâmide da automação para realizar análises e gerar relatórios, reduz-se o consumo de recursos computacionais dos sistemas prioritários à supervisão e monitoração do processo. As análises geradas nessa instância ainda podem ser integradas a outras fontes de dados e se transformar em informação útil às equipes de engenharia, onde cada usuário do sistema pode acessar os dados de forma independente e instantânea. Os frutos da centralização de dados (históricos e em tempo real) de processo e a democratização da visualização da informação *online* se traduzem em maior interatividade com o processo e agilidade na tomada de decisões para sua otimização (Parente, 2019; Carvalho *et al.*, 2005).

Devido às dificuldades em limitar os campos de ação de cada ferramenta EPS, os sistemas do tipo PIMS e MES foram incorporando funções um do outro a tal ponto que algumas literaturas mais recentes não os distinguem. Entretanto, para indústrias químicas e bioquímicas, o papel dos dois é bastante diferente. Sistemas do tipo MES recebem as informações do PIMS e SCADA/SDCD para fazer interface com os sistemas de gestão (Souza, 2005). Enquanto as principais virtudes do PIMS estão associadas às ferramentas de análise e visualização do extenso banco de dados histórico, o MES fornece visão holística da produção e dos equipamentos para decisão de negócios com bancos de dados relacionais diversos, quer nos campos de qualidade e capacidade do processo e produto, quer no rastreamento de material e planejamento de produção (Kletti, 2007).

### II.3.1 Infraestrutura de Dados

A partir desta seção este trabalho apresenta uma visão geral de um sistema PIMS, baseando-se fortemente no sistema PIMS PI System da OSIsoft, utilizado na metodologia e também um dos principais representantes do gênero no mercado nacional. Carvalho *et al.* (2005) descreve um sistema PIMS como sendo constituído por cinco elementos: servidor principal, servidor de comunicação, estações clientes, banco de dados relacional e infraestrutura de rede (corporativa e de automação).

O servidor principal é o elemento responsável pela centralização das informações, onde os dados de processo recebidos pelos servidores de comunicação são armazenados e disponibilizados para aplicações posteriores. Como medida de segurança, alguns fornecedores desenvolvem seu sistema PIMS com mecanismo de “*Store&Forward*” que permite que os dados recentes sejam temporariamente armazenados no servidor de comunicação caso haja perda de comunicação com o servidor principal (Carvalho *et al.*, 2005).

Os servidores de comunicação, por sua vez, são os responsáveis por interligar o servidor principal com as fontes de dados. Apesar dos principais PIMS possuírem interfaces dedicadas para a maioria dos sistemas de importância comercial, o protocolo OPC é o convencionalmente escolhido, tornando-se o PIMS o cliente OPC e os sistemas SCADA, SDCD e CLPs, os servidores OPC (Carvalho *et al.*, 2005).

Nas estações clientes são disponibilizados os diferentes *softwares* do PIMS desenvolvidos para seus usuários-chave. Já o banco de dados relacional é composto por um conjunto de bancos de dados externos utilizados pelo PIMS ao acessar módulos especiais para processos em bateladas e geradores de eventos. Por fim, quanto à infraestrutura de rede mencionada, em geral se utilizam as redes corporativas da empresa (Carvalho *et al.*, 2005).

### II.3.2 Fluxo de Dados

Os dados recebidos através das interfaces de comunicação do PI System são enviados ao servidor principal (PI Data Archive) na forma de eventos. Cada evento apresenta 3 elementos: *timestamp* (data completa de coleta do dado com precisão em milissegundos), valor e *status* (texto indicando a qualidade do dado coletado) (OSIsoft, 2018). A partir de

então o sistema PIMS realiza uma série de procedimentos para armazenar informações instantâneas durante uma longa quantidade de tempo no menor espaço de disco possível.

A redução em tempo real do tamanho do banco de dados armazenados é feita automaticamente através de algoritmos de exceção e compressão. Logo após a interface coletar o evento, ela mesma realiza um teste de exceção para avaliar a relevância do mesmo e descartar os reprovados. O evento, quando aprovado neste primeiro teste, é enviado junto ao seu predecessor a um subsistema do servidor principal e assume o valor de *snapshot* atual. O *snapshot* anterior ao atual é avaliado por um teste de compressão e descartado caso seja reprovado. Os aprovados são enviados à fila de eventos deste subsistema (OSISoft, 2018).

Um segundo subsistema do servidor principal lê os eventos na fila de eventos e atua como uma memória e *buffer* de disco entre esses dois subsistemas. Os eventos validados são retirados pelo segundo subsistema e armazenados em cache de memória. Por fim, periodicamente ele move os dados do cache para os arquivos *archive* onde os eventos são armazenados definitivamente (OSISoft, 2018).

Em resumo, o fluxo de dados segue as seguintes etapas:

1. Interface coleta os dados e cria um evento.
2. A interface realiza teste de exceção para decidir se o evento é significativo.
3. Envia eventos significativos para servidor principal (PI Data Archive) e descarta os não significativos.
4. Servidor principal executa teste de compressão para determinar a necessidade de armazenar o evento.
5. O evento aprovado no teste de compressão é enviado à fila de eventos, de onde são movidos para armazenamento final, e descarta os reprovados.

Esta compressão é uma das características mais importantes de sistemas PIMS. Carvalho *et al.* (2005) indica uma taxa de compressão na ordem de 10 valores coletados para 1 armazenado (10:1) e 20:1. A perda de informação deste procedimento é minimizada pela utilização de algoritmos de compressão de amostragem variável, uma vez que amostram a curva nos pontos de mudanças significativas (Souza *et al.*, 2005).

Entretanto, a fidelidade da resposta dinâmica resultante da operação de restauração do sinal original a partir dos dados armazenados (reconstrução) é assunto de interesse da academia (Thornhill, Choudhury, Shah, 2004). Os métodos de consulta a base de dados devem ser escolhidos com cautela para não gerar equívocos na interpretação da informação (Valdman e Tiradentes, 2019).

Quanto a esses aspectos, Souza (2005) define 5 características essenciais de um sistema PIMS. As duas primeiras são quanto à velocidade: o algoritmo de compressão deve ser simples e rápido assim como a reconstrução do sinal (especialmente útil para visualização dos gráficos de tendência). Coloca também altas taxa de compressão e fidelidade de reconstrução dos dados como pontos relevantes (minimização do erro entre o valor descompactado e original). Por último acrescenta a segurança dos dados coletados, para que não se percam em caso de pane ou queda de energia. Os procedimentos do PI System para este último caso foram omitidos nesta seção, mas podem ser consultados no manual do fornecedor (OSIsoft, 2018).

Os algoritmos de compressão de amostragem variável disponíveis podem ser divididos em diretos (conhecidos também como *piecewise linear trending methods*) e de transformação. Ainda que o método *wavelet* (mais convencional dentre os de transformação) seja superior aos métodos diretos, estes são os aplicados industrialmente pela sua capacidade de aplicação em tempo real (Aslmeyer, 2006; Thornhill, Choudhury, Shah, 2004). Dois algoritmos deste grupo são frequentemente adotados: o *Boxcar/Backslope* (BCBS) e o *swinging doors*. Ambos os métodos são explorados em Souza *et al.* (2005). Enquanto o PI System adota o segundo, concorrentes como o InfoPlus.21 da AspenTech utilizam o primeiro (Thornhill, Choudhury, Shah, 2004; Silveira *et al.*, 2012; Souza *et al.*, 2005).

### II.3.2.1 Armazenamento da Informação

As informações são armazenadas em bancos de dados temporais contendo basicamente as informações do evento (*timestamp*, valor e *status*) junto ao identificador do dado (nome do ponto ou *tag*). Quanto aos tipos de dados armazenados, no PI System são separados em 5 classes padrão:

- **Alarm:** informações de alarmes;
- **Base:** inclui atributos do sistema e de seus usuários;
- **Classic:** dados em tempo real provenientes de interfaces;
- **SQC Alarm:** pontos de alarme baseados em controle estatístico de processo;
- **Totalizador:** pontos que representam um valor integrador de um determinado conjunto de dados.

### II.3.3 Aplicativos para Análise de Dados

Dentro do PIMS algumas ferramentas são disponibilizadas para análise de dados, gerando novos dados a serem historiados no servidor principal. Inicialmente dentro servidor de dados, dois instrumentos para geração de cálculos se apresentam. Enquanto a configuração do primeiro é mais simples, mas apresenta limitações, o segundo oferece ambiente de programação amigável ao programador básico com uma maior disponibilidade de recursos (OSIsoft, 2019).

Análises de dados mais avançadas também podem ser realizadas nos aplicativos clientes. Através de aplicativos mais amigáveis ao engenheiro de processo, é possível acessar o módulo de programação em Visual Basic para coletar e guardar valores em pontos do servidor principal. Por outro lado, é disponibilizado um serviço de análise de dados com funções pré-definidas e baseadas em diversos métodos, como busca em tabelas, cálculos simples sequenciamento de expressões matemáticas com operadores lógicos (OSIsoft, 2019).

## II.4 Redes Neurais

Redes neurais artificiais fazem parte de um campo de estudos multidisciplinar, envolvendo áreas distintas da ciência como: neurociência, matemática, estatística, física, ciência da computação e engenharia. Em torno de sua principal virtude – a habilidade de aprender – se estabeleceram aplicações nos campos da modelagem, análise de séries temporais, reconhecimento de padrões, entre outros (Haykin, 2001).

De modo geral, o desenvolvimento desta tecnologia baseou-se em dois principais objetivos: primeiro entender melhor o sistema nervoso humano, e então, construir informação em sistemas de processamento inspirados nele. Ainda que em algumas tarefas o computador seja altamente eficiente, não é capaz de alcançar o cérebro humano em trabalhos

cognitivos. A demanda por flexibilidade, robustez e eficiência em problemas complexos encontrou no cérebro humano um modelo de processamento a ser simulado (Prieto *et al.*, 2016).

Por esta ótica, o cérebro é modelado como um computador altamente complexo, não-linear e com atividades em paralelo. Ele é o centro de um sistema que recebe estímulos, os converte em entradas para uma rede neural repleta de neurônios simples que toma decisões apropriadas e envia as decisões para atuadores, que convertem as saídas da rede em respostas discerníveis. Dessa forma, Haykin (2001) define uma rede neural como um processador paralelamente distribuído, constituído de unidades de processamento simples (neurônios), que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para uso. A simulação do cérebro humano se completa pelo processo de aprendizagem com o ambiente e que é armazenado nas conexões entre os neurônios.

A aprendizagem fornece capacidade de generalização e adaptabilidade a mudanças no ambiente para a rede, mas não é este o único benefício das RNAs. Sua não linearidade distribuída e a uniformidade com que se constitui permitem resolver problemas complexos em aplicações diversas. Mais ainda, sua natureza paralela a torna potencialmente rápida na computação (Haykin, 2001).

#### II.4.1 Processamento Matemático

Uma unidade de processamento de informação (neurônio ou nó  $k$ ) do tipo *perceptron*, representado na Figura II-3, é constituída basicamente por 4 elementos (Haykin, 2001):

1. conjunto de sinapses caracterizadas com peso próprio  $w_{kj}$ ; referente ao neurônio  $k$  que enviam informação proveniente da entrada  $j$ .
2. somador  $u_k$  que combina linearmente os  $x_j$  sinais de entrada, ponderados pelos seus respectivos pesos  $w_{kj}$ .
3. *bias*  $b_k$  que se adiciona ao resultado do somador para aumentar ou diminuir a ativação  $v_k$ .
4. função de ativação  $F$ , que restringe a amplitude da saída do neurônio  $y_k$ .

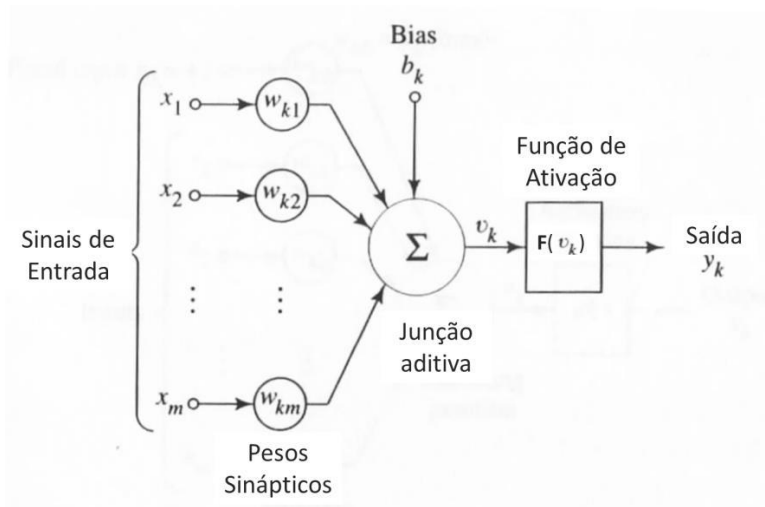


Figura II-3 Modelo não-linear de um neurônio.

Fonte: Haykin, 2001 (Adptado).

O uso do bias tem o efeito de aplicar uma transformação afim à saída  $u_k$ , alterando o potencial de ativação  $v_k$ . Isto se faz deslocando verticalmente a função de  $v_k$  em função de  $u_k$ , impedindo que intercepte a origem. Posteriormente, a função de ativação utiliza este potencial para definir a saída do neurônio.

#### II.4.2 Função de Ativação

Modelos pioneiros, como o de McCulloch e Pitts (1943), utilizam funções de ativação do tipo *Heaviside*. Entretanto, o desenvolvimento histórico das RNAs exigiu a utilização de funções de ativação diferenciáveis para sua parametrização. Desta adaptação surgiram funções sigmóides, portando equilíbrio entre os comportamentos lineares e não lineares. As principais expoentes desse tipo de função são as funções logística (limitada entre 0 e 1) e tangente hiperbólica (limitada entre -1 e 1) (Haykin, 2001).

Outras funções de ativação globais, ativas em todo o domínio de entrada, podem ser utilizadas, como as funções exponencial, seno e identidade (Haykin, 2001). Em contraste, funções de ativação de natureza local também podem ser abordadas. A principal representante desta classe é a função de base radial, em especial a função gaussiana. Em seu comportamento apresentam máximo de resposta ao centro e resposta nula ao se aproximar das extremidades (Himmelblau, 2000).



Redes neuronais baseadas nesta classe de função acrescentam novos parâmetros: um centro  $C_j$  para cada entrada e o desvio-padrão  $\sigma_k$  do neurônio. O processamento matemático é, então, sutilmente alterado. Em seu modelo mais simples (1 camada escondida e 1 camada de saída) não há pesos para as sinapses que levam aos neurônios da camada escondida e nestes somente aplicam a função de ativação. Em seguida, os neurônios da camada de saída recebem através de cada sinapse o resultado de  $F(x_j, C_j, \sigma_k)$  e os peso sináptico associados para aplicar somatório análogo à  $u_k$  e então calcular  $v_k$ . Em última análise, a escolha da classe de função de ativação é fundamental para a definição da topologia e modelagem matemática da rede neuronal (Himmelblau, 2000).

### II.4.3 Topologia

A rede neuronal artificial é formada por uma coleção de neurônios conectados uns aos outros, de forma que o sinal de um vai sendo transmitido aos demais pelas diferentes camadas da rede. Cada camada contém uma quantidade  $m$  de neurônios distribuídos paralelamente. A camada de entrada é a que recebe os sinais de entrada de uma fonte externa, e em geral o único processamento nesta camada é uma alteração de escala. O grupo de neurônios que retorna o sinal ao ambiente externo é, por sua vez, a camada de saída. Quaisquer camadas existentes entre os dados de entrada e de saída são denominadas camadas ocultas (Himmelblau, 2000).

De acordo com a quantidade de camadas e o sentido do fluxo de informação, as RNAs se dividem em 3 tipos de arquiteturas: redes *feedforward* (FF) de camada única, redes FF com múltiplas camadas e redes *feedback* (FB) ou recorrentes. Para esta classificação, a camada de entrada é desconsiderada. Nas redes FF (Figura II-4) há um único sentido para o fluxo de informação, sendo este das entradas para a primeira camada oculta e desta em diante até a camada de saída; no caso da rede FF de camada única, esta é a própria camada de saída. Redes FB, em contraposição, tem pelo menos um laço de realimentação (Haykin, 2001).

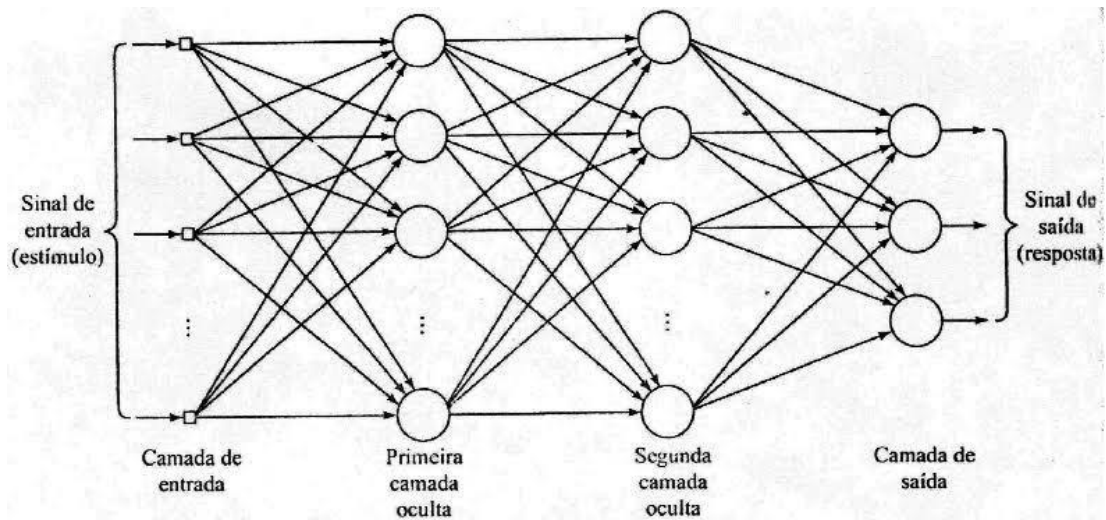


Figura II-4 Arquitetura de rede *feedforward*.

Fonte: Haykin, 2001.

#### II.4.4 Treinamento

Em geral, não é possível utilizar métodos analíticos para calcular os parâmetros numéricos de uma rede neuronal (seus pesos de cada sinapse e *bias* de cada neurônio). Ela precisa ser treinada com um conjunto de dados obtidos do processo a ser modelado. Nesse treinamento é estabelecida toda a topologia da rede, o que inclui uma etapa de aprendizagem para estimação dos parâmetros numéricos por meio de um algoritmo de otimização (Himmelblau, 2000; Prieto *et al.*, 2016).

Haykin (2001) define, portanto, a aprendizagem como “um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual a rede está inserida”. Existe uma variedade de algoritmos de aprendizagem, oferecendo vantagens distintas e diferindo na formulação de ajuste do peso sináptico e na maneira pela qual a própria rede se relaciona com o ambiente.

Existem três tipos básicos de aprendizado: supervisionado, não-supervisionado e o *reinforcement learning*. No primeiro se fornece o conjunto de entradas e saídas da rede, para que, de posse da resposta desejada, o erro seja minimizado. Por outro lado, o treinamento não supervisionado dispensa os valores de saída da rede, uma vez que adquire conhecimento dos conjuntos de padrões e os agrupa em classes. O terceiro tipo se baseia num mecanismo sem

dados prévios de saída que recompensa ou pune suas respostas mediante interação contínua com ambiente e maximiza o reforço positivo. (Prieto *et al.*, 2016)

Escolhido o tipo de aprendizado, o procedimento padrão recomenda a divisão dos dados em um conjunto de treinamento e um conjunto de testes. Ao testar a rede neuronal treinada com dados diferentes avalia-se sua capacidade de predição e generalização. No entanto, é necessário que seu conjunto de treinamento represente todo o domínio de interesse. Caso contrário, a predição dos resultados para entradas extrapoladas de fora do domínio pode ser prejudica (Himmelblau, 2000).

#### **II.4.5 Rede Multilayer Perceptron (MLP)**

As redes neuronais mais consolidadas para aplicação na indústria de processo são as redes *multilayer perceptron* (MLP) e as redes de função de base radial (RBF). Estes dois tipos de redes se enquadram na classe de redes FF com múltiplas camadas. Entretanto, diferem entre si no tipo de neurônio utilizado, nas funções de ativação e nos métodos de aprendizagem.

Redes MLP são baseadas em neurônios do tipo *perceptron*, tanto para as camadas ocultas quanto para a camada de saída. Há pelo menos uma camada escondida com funções de ativação não lineares e utilizam treinamento supervisionado para a determinação dos parâmetros (pesos sinápticos e *biases*). Dentre os diversos algoritmos de aprendizado disponíveis para esta rede, o mais abordado é o *backpropagation* (BP). Nele, a rede propaga as entradas recebidas para gerar os valores de saída. O desvio em relação ao valor esperado é calculado e o erro é retropropagado até a camada de entrada, ajustando os parâmetros (Haykin, 2001).

Ao estruturar a rede em torno de funções de ativação de base radial é necessário ajustar o processamento matemático envolvido e adicionar novos parâmetros. Em redes RBF os neurônios ocultos e de saída diferem quanto à modelagem, ao passo que os presentes nas camadas ocultas recebem treinamento não supervisionado e os da camada de saída recebe treinamento supervisionado. Apesar da sua maior complexidade, o treinamento desta rede é, em geral, mais rápido que de uma rede MLP (Himmelblau, 2000).

Uma comparação resumida entre configurações típicas para redes MLP e RBF é apresentada na Tabela II-1. Contudo, é preciso destacar que a rede neuronal de maior aplicação industrial é a MLP pois reúne as vantagens de ter ampla capacidade de generalização e menor número de parâmetros (Kadlec, Gabrys e Strandt, 2009).

Tabela II-1 Comparação entre características típicas das redes MLP e RBF.

Rede	Função de Ativação Típica			Aprendizagem
	Camada de Entrada	Camada Escondida	Camada de Saída	
MLP	Linear	Logística ou Tangente hiperbólica	Linear ou Logística	Supervisionada
RBF	Linear	Radial Gaussiana	Linear ou Logística	Não supervisionada (camadas ocultas) Supervisionada (camada de saída)

Fonte: Salvatore, 2007 (Adptado).

## II.5 Sensor Virtual

Sensores virtuais (também conhecidos como *soft sensors*) são modelos computadorizados que utilizam grande quantidade de dados mensurados e historiados para prever resultados não disponíveis. Na indústria de processos, são largamente utilizados para prever variáveis de processo em tempo real a partir dos dados recebidos de sistemas supervisórios e análises laboratoriais (Kadlec, Gabrys e Strandt, 2009).

Podem ser divididos em duas classes gerais: orientados a modelos (*model-driven*) e empíricos (*data-driven*). A primeira classe apoia toda sua metodologia em princípios fenomenológicos, surgindo modelos focados na descrição de estados estacionários ideais do processo. Os modelos baseados somente nos dados experimentais obtidos, posteriormente apresentados, carregam a vantagem de representar mais fielmente as condições da planta (Kadlec, Gabrys e Strandt, 2009).

A implementação de sensores virtuais no contexto industrial ganha espaço nas atuais discussões de fábricas inteligentes, vinculados pelos seus objetivos mútuos. Tomando como

exemplo as redes neuronais, encontramolas aplicadas em problemas de detecção de falhas, estimação de variáveis de processo, previsão de qualidade do produto e até mesmo em controle avançado de processos (Prieto *et al.*, 2016; Himmelblau, 2000).

Fortuna *et al.* (2007) apresenta como essas técnicas se tornam atrativas pelo ponto de vista empresarial. Em primeiro lugar custos podem ser significativamente reduzidos. Sensores físicos de alto custo podem ser substituídos por redes de predição, assim como a rotina de trabalho de analistas químicos pode ser reduzida. Em segundo lugar destaca-se o ganho de confiabilidade no processo, pois podem ser implementados em paralelo aos sensores físicos existentes.

Os sensores virtuais fornecem dados não somente para os cenários de falha do instrumento, mas também em tempo real. Isto é, conferem grande vantagem frente aos longos intervalos de amostragem padrão – característico de variáveis de complexa medição. Por fim, não há necessidade de investimento em sistemas físico (*hardware*) para a sua implementação, visto que a atual arquitetura de automação já é capaz de receber essas técnicas.

Entretanto, o processo de implementação de sensores virtuais passa pela escolha de um *software* adequado, visto que ainda há uma deficiência de ferramentas industriais apropriadas disponíveis comercialmente. Não obstante, soluções empíricas são intrinsecamente individuais e personalizadas caso a caso. Nesse cenário deve-se administrar o dilema entre adquirir sistemas padronizados e genéricos – que impõem limitações à algoritmos avançados – ou desenvolver soluções próprias que necessitam de manutenção contínua (Chiang, Lu, Castillo, 2017).

Basear-se nas virtudes de generalização e adaptabilidade das redes neuronais é uma possibilidade. Landim (2016) estudou a transportabilidade de uma mesma metodologia de desenvolvimento de RNAs para diferentes unidades produtivas de uma planta industrial, alcançando bons resultados para redes MLP com uma camada escondida. Dessa forma, seria possível economizar tanto no tempo e esforço para desenvolvimento das redes quanto na adaptação de ferramentas já existentes.

## II.5.1 Implementação em Sistemas Adquiridos

Industrialmente se encontram alguns relatos de adaptação de sistemas já adquiridos comercialmente; especialmente de *softwares* de automação do segundo e terceiro níveis. A limitação destes em relações a utilização de algoritmos avançados é então contornada, quer pela busca de modelos matemáticos simples, a fim de consumir menos processamento digital, quer pela extração do maior potencial de análise e comunicação de sistemas externos (Zhou, Chai e Sun, 2013; Oliveira *et al.*, 2014).

Buscando uma implementação direta, Oliveira e colaboradores (2014) desenvolveram um sensor virtual para consumo específico da caldeira de uma central termoelétrica diretamente no PIMS. A predição é realizada com rede neuronal MLP com 7 variáveis de entrada, 2 neurônios na camada escondida e 1 variável de saída. A rede foi implantada em linguagem SQL, disponibilizada pelo PIMS, e em 4 semanas de execução apresentou 97% de confiabilidade e erro percentual máximo de -3,42%.

Tendo também o sistema PIMS como receptor final, Salvatore (2007) desenvolveu um analisador virtual para teor de enxofre total em correntes de diesel oriundas de uma unidade de hidrotreatamento na Refinaria Duque de Caxias (REDUC) da Petrobras. Apesar dos melhores resultados para redes do tipo RBF, o número de parâmetros requisitado (pelo menos 10 vezes menor para MLP) foi utilizado como critério de decisão, uma vez que o processamento computacional seria mais exigido.

Assim, uma rede MLP com 33 neurônios na camada de entrada, 7 neurônios na camada escondida e 1 variável de saída foi implementada em Microsoft Visual C++.. Foi encontrado um tempo de execução menor que 1 segundo, indicando baixo custo de processamento. Entretanto, a resposta é calculada a cada 3 horas e meia, respeitando a modelagem do processo. O resultado do algoritmo é historiado no servidor da Infraestrutura de Dados, disponibilizada pela equipe de gerenciamento do PIMS (Salvatore, 2007).

No mesmo sistema PIMS, Ghergherehchi (2011) explora um ambiente de programação que funciona como uma fonte de dados interna ao próprio servidor. Neste trabalho, foi desenvolvido um sistema de alarmes baseado em lógica *Fuzzy* para monitorar bombas submersas da filial Norueguesa da companhia de energia Talisman. Para tal fim, uma base

de dados modular foi configurada e uma biblioteca externa importada. Foi verificado que o sistema é capaz de verificar 1000 eventos por segundo.

Uma abordagem amigável ao engenheiro de processos foi desenvolvida por Valdman (2010) para monitorar poluentes de caldeira piloto da Escola de Química/UFRJ. Com o módulo de programação VBA da Interface Homem-Máquina (IHM) de um sistema SCADA, uma tela de configuração foi criada. Como teste, uma rede neuronal do tipo MLP foi implementada por meio da mesma. Redes RBF foram descartadas neste trabalho tanto pelo custo de processamento já comentado quanto pela necessidade de muitos pares de treinamento.

A rede obtida utiliza 6 variáveis na camada de entrada, 10 neurônios com função tangente hiperbólica na camada escondida e retorna 6 variáveis na camada de saída. Reportes sobre a qualidade dos dados, confiabilidade da informação e identificação da causa de falha foram implementadas em conjunto. A taxa de cálculo foi definida em 10 segundos e os resultados finais são historiados em tags do SCADA definidas pelo usuário (Valdman, 2010).

Encontra-se ainda na literatura implementações de sensores virtuais para ajuste do controle do processo. Zhou, Chai e Sun (2013) utiliza um sensor virtual baseado em rede neuronal RBF para ajustar parâmetros do controlador da unidade de moagem de uma planta de processamento mineral na China. O módulo de programação VBA do sistema SCADA foi escolhido para executar o algoritmo, em comunicação com um software comercial onde cálculos matriciais são executados e enviar os resultados para um SDCD.

Aplicações com objetivo de sintonizar parâmetros de SDCD comumente recorrem ao processamento do sensor em ambiente externo. Analogamente ao último caso, Li *et al.* (2013) prediz 2 resultados de qualidade de planta de alumina chinesa para sintonizar controle em SDCD. Informações de processo são enviadas diretamente do PLC para unidade externa onde sensor virtual baseado em técnica *support vector machine* executa os cálculos. As duas respostas obtidas são utilizadas para cálculo do set-point do controlador e então alimentam o SDCD. Quanto a este exemplo, pode-se já considerá-lo um híbrido entre a implementação no sistema SDCD já adquirido pela empresa e o desenvolvimento de sistema própria.

## II.5.2 Implementação em Sistemas Próprios

Na contramão aos métodos que se baseiam nos sistemas de automação comerciais estão os sistemas personalizados. Diferenciam das soluções próprias, como visto em alguns casos anteriores, por serem verdadeiras ferramentas digitais externas aos sistemas adquiridos e construídas com o objetivo de hospedar e configurar o sensor virtual.

Popov e colaboradores (2014) apresentam uma abordagem sistemática para criação própria de sensor virtual baseado em redes neurais utilizando tecnologia .NET. Desenvolve um servidor no sistema operacional Windows que obtém data periodicamente através de comunicação OPC e a envia para módulo de predição. Outro módulo, construído em paralelo, faz a interface gráfica com o usuário, onde este pode criar a rede, treiná-la e monitorar sua saída. O módulo de treinamento utiliza método Levenberg-Marquardt, permite configurar parâmetros do treinamento como o tempo de coleta de dados para o mesmo e ainda agenda novos treinamentos periódicos.

Como estudo de caso, este trabalho implementa um sensor virtual para monitorar CaO livre no clínquer e os finos de cimento em fábrica de cimento da Lafarge em Beočin (Sérvia). Para estimação do CaO, a rede MLP conta 7 variáveis de entrada e 2 neurônios na camada escondida, enquanto que para prever os finos de cimento são utilizadas outras 7 variáveis de entrada e 6 neurônios na camada escondida. Ambas as redes são retreinadas a cada 2 horas (Popov *et al.*, 2014).

Outra metodologia é apresentada por Rezende, Teixeira e Mendes (2015) para implementação de sensor virtual em ambiente Windows para a Petrobras. A partir de uma abordagem orientada a objeto, o sensor é encapsulado como uma Dynamic Link Library (DLL). Utiliza módulo do SCADA, desenvolvido pela própria empresa, responsável por gerenciar as tarefas dos usuários do sistema.

Uma das funções é gerenciar as funções dos usuários implementadas como DLLs e atuar como camada de comunicação entre essas tarefas e os demais sistemas da empresa como o SCADA e o PIMS. A configuração do cálculo a ser executado é realizada em ambiente de programação inerente ao sensor, enquanto as tarefas de configuração e os



resultados do sensor são realizados em interface gráfica desenvolvida. Testes relataram tempos de execução inferiores a 1 segundo.

De posse deste mesmo ambiente, Aguirre e colaboradores (2017) implementaram rede neuronal MLP com 5 variáveis de entrada e 10 neurônios na camada escondida (função de ativação tangente hiperbólica) para a estimação única da pressão de fundo do poço de petróleo. Apesar da acurácia da inteligência artificial, os autores ressaltam a necessidade de retrainar a rede neuronal periodicamente e após mudanças bruscas no processo.

Programar novos treinamentos dos modelos implementados no sensor virtual é uma das claras necessidades (Rezende, Teixeira e Mendes, 2015; Aguirre *et al.*, 2017). Alguns trabalhos apresentados discutem as vantagens de estimação de parâmetros físico-químicos disponíveis em intervalos de tempo longos por meio de análises laboratoriais ou analisadores digitais. Não obstante, os sensores virtuais recorrem a este tipo de dado laboratorial para compor as variáveis de entrada do sensor (Valdman, 2010; Salvatore, 2007; Li *et al.*, 2013).

Embora a literatura sobre desenvolvimento de inteligências artificiais para sensores virtuais seja vasta, ainda é pequeno o número de trabalhos que discutem procedimentos claros de implementação destas tecnologias em âmbito industrial (Rezende, Teixeira e Mendes, 2015). No cenário industrial nacional, os esforços para utilização das ferramentas ainda se encontram concentrados em empresas de grande porte como a Petrobras (Rezende, Teixeira e Mendes, 2015; Aguirre *et al.*, 2017) e BRASKEM (Neumann *et al.*, 2019).

## CAPÍTULO III — METODOLOGIA

O presente trabalho foi desenvolvido na Fábrica Carioca de Catalisadores (FCC) e implementa, no sistema PIMS adquirido PI System, um sensor virtual baseado em rede neuronal MLP. A validação é realizada com a rede neuronal MLP 10-12-1 para predição do teor de amônia na saída de fundo em coluna *stripper* desenvolvida por Landim (2016). A metodologia adotada é uma adaptação para sistema PIMS daquela proposta por Valdman (2010) para SCADA.

### III.1 Descrição do Processo

A coluna *stripper* de recuperação de amônia em operação na Fábrica Carioca de Catalisadores tem por objetivo obter na corrente de topo a maior quantidade possível de amônia na fase gasosa proveniente dos processos de produção. Por questões de controle de qualidade do efluente o teor de amônia na saída de fundo deve estar abaixo de um limite fixado por norma ambiental.

A Figura III-1 situa a coluna *stripper* dentro do processo de tratamento de efluente. A coluna é utilizada para separar a amônia do efluente do processo – evitando contaminação do corpo hídrico que o recebe – e, então, reintegrá-la ao processo. O processo em questão é uma etapa intermediária do tratamento de efluente amoniacal, procedendo à etapa de sedimentação para retirada de sólidos. O clarificado é alimentado à coluna, onde o produto de topo, rico em amônia, é destinado para reciclo e o produto de fundo segue para as etapas de tratamento final.

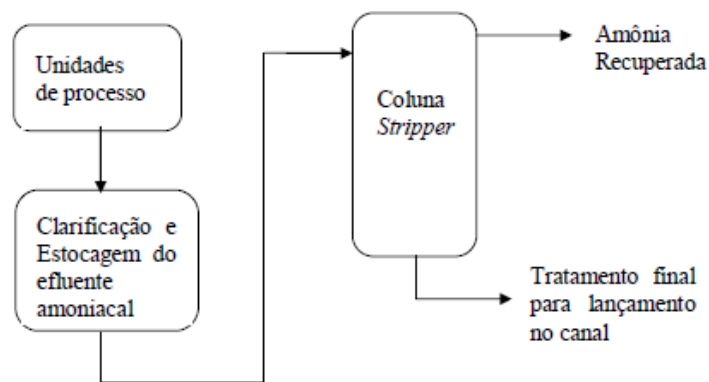


Figura III-1 Principais etapas do tratamento de efluente amoniacal.

Fonte: Landim, 2016.

A coluna é dividida em duas seções (enriquecimento e esgotamento) e entre elas se situa a carga da coluna. A alimentação da coluna recebe tanto tratamento cáustico – para ajuste de pH com adição de hidróxido de sódio a 50% – quanto tratamento térmico por meio de trocador de calor com a saída de fundo da coluna em maior temperatura, formando a carga.

A seção de enriquecimento é localizada entre a alimentação e o topo, concentrando a saída de topo em amônia na fase gasosa, e sendo composta por 4 pratos com borbulhadores. Esta seção recebe ainda a entrada do refluxo de topo, obtido após a condensação da saída de topo, através de uma corrente de água de resfriamento.

A seção de esgotamento é localizada entre a alimentação e o fundo da coluna e sua função é evitar a saída de amônia no líquido de fundo. Esta etapa é composta por dois leitos recheados aleatoriamente com anéis do tipo Pall Rings, um redistribuidor de líquido entre os leitos, um vaso de flash operando em pressão negativa após a saída de fundo e o refluxo de fundo (arraste), composto pelo vapor do flash e pelo vapor proveniente das caldeiras da central de utilidades.

Devido à adição de vapor aquecido como arraste no fundo da coluna, o gradiente de temperatura da coluna é decrescente do fundo para o topo. O vapor que ascende entra em contato com líquido descendente cedendo-lhe calor, liberando e arrastando a amônia existente na fase vapor. A perda de calor provoca ainda a condensação de parte do vapor, saindo com o produto de fundo.

A operação da coluna requer ainda o monitoramento do diferencial de pressão (mantido constante pelo alívio dos não condensáveis direcionados para uma coluna absorvedora) indicando indiretamente o grau de obstrução que bandejas e recheios sofrem pela deposição e ou cristalização de materiais insolúveis e aderentes. Este efeito negativo à eficiência da coluna ocorre majoritariamente devido à presença de sólidos na carga.

A Figura III-2 apresenta esquema do equipamento, enquanto a Tabela III-1 lista as principais variáveis conforme serão nomeadas neste trabalho. A numeração indica ainda onde as variáveis apresentadas na Tabela III-1 são medidas. As variáveis 5 e 6 (Razão carga vapor e Arraste) não aparecem na Figura III-2 por serem valores obtidos de uma relação e não por

meio de uma medição. As demais variáveis são todas medidas *online*. A seguir são apresentadas as equações para cálculo destas tags:

$$\text{Razão Carga Vapor} = \frac{\text{Carga}}{\text{Vapor}} [=] [1] \quad \text{Eq. III-1}$$

$$\text{Arraste} = \frac{\text{Vazão Arraste}}{\text{Vapor}} [=] [1] \quad \text{Eq. III-2}$$

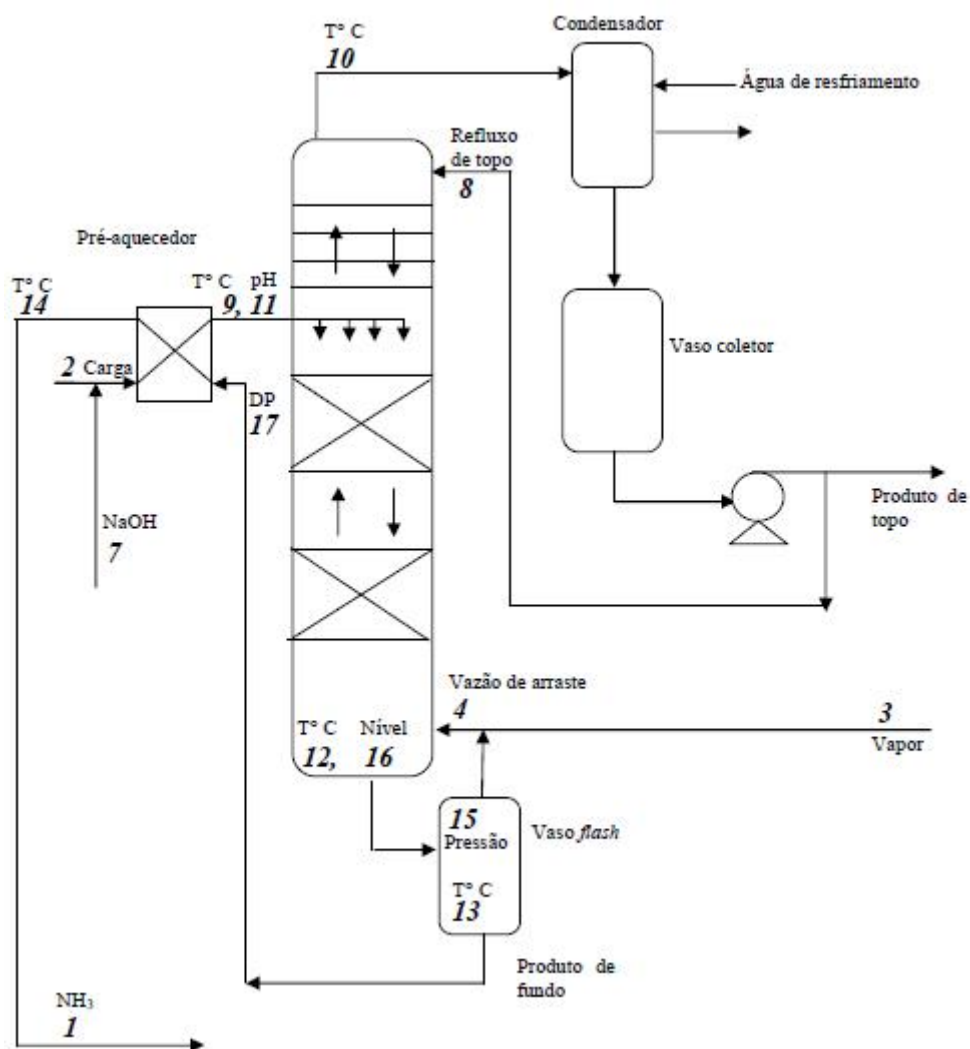


Figura III-2 Representação esquemática da coluna.

Fonte: Landim, 2016.

Tabela III-1 Variáveis de processo da coluna de recuperação de amônia.

Nº	Nome	Descrição
1	NH <sub>3</sub> fundo	Teor de amônia no fundo da coluna (ppm)
2	Carga	Vazão de alimentação da coluna (kg/h)
3	Vapor	Vazão de vapor vivo – proveniente da caldeira (kg/h)
4	Vazão arraste	Vazão de arraste na entrada da coluna (kg/h)
5	Razão carga vapor	Relação das vazões de alimentação e vapor vivo
6	Arraste	Relação das vazões de arraste e vapor vivo
7	Soda	Vazão de soda cáustica (l/h)
8	Refluxo	Vazão de refluxo no topo (l/h)
9	T carga	Temperatura da carga (°C)
10	T topo	Temperatura de topo (°C)
11	pH carga	pH da alimentação
12	T fundo	Temperatura do fundo (°C)
13	T <i>flash</i>	Temperatura do vaso de <i>flash</i> (°C)
14	T saída trocador	Temperatura do fundo após o pré aquecedor (°C)
15	P <i>flash</i>	Pressão do vaso <i>flash</i> (kgf/cm <sup>2</sup> )
16	Nível fundo	Nível do fundo da coluna (%)
17	DP recheio sup	Diferencial de pressão do leito no recheio superior (mmH <sub>2</sub> O)

Fonte: Landim, 2016 (adaptado).

### III.1.1 Arquitetura de Automação Instalada

O sistema de automação da planta onde é desenvolvido o presente trabalho é baseado na pirâmide de automação. Sendo assim as variáveis mensuradas listadas na Tabela III-1 (todas à exceção da Razão Carga Vapor e Arraste) são obtidas por instrumentos na planta. De todas elas, a mais crítica é o teor de amônia no fundo da coluna, uma vez que tem grande potencial negativo de impacto ambiental.

Esta variável é monitorada por um analisador de amônia instalado na corrente de fundo, após a passagem pelo pré aquecedor da carga. Este equipamento possui uma limitada frequência de análise, configurada para fornecer um resultado a cada 20 minutos, devido às etapas de limpeza e calibração efetuadas entre as medições. A mesma variável ainda é inferida por análise laboratorial em intervalos de 4 horas e historiada separadamente da medida pelo analisador. A medida do laboratório não foi utilizada neste trabalho devido ao grande intervalo de resultados.

A planta apresenta 5 malhas de controle implementadas: 3 em modo cascata da vazão de alimentação com vazão de vapor, pH com vazão de soda e temperatura de topo com vazão de refluxo; 2 malhas *feedback* para a pressão do vaso *flash* e nível do fundo da coluna.

A vazão de alimentação é controlada através de um SDCD em modo cascata com a vazão de vapor, sendo esta última a malha principal, onde o *Set Point* (SP) da carga é obtido a partir da relação ótima com a vazão de vapor. Outra malha de controle em cascata é realizada com o sensor de pH no loop principal e a vazão de soda na malha secundária. Nestes dois casos se estabelecem relações entre variáveis, principalmente o efeito diretamente proporcional entre o pH da carga e a quantidade de soda cáustica alimentada.

Um terceiro controle em modo cascata é realizado para manter a temperatura de topo constante por meio do ajuste na vazão de refluxo de topo. Sendo o refluxo de topo líquido, em temperatura menor que o vapor de saída, a entrada deste na coluna reduz a temperatura do último prato. Devido ao caráter inversamente proporcional do processo, um aumento do SP de temperatura implica na redução no SP da vazão de refluxo. Também são controladas por malhas *feedback* a pressão do vaso *flash* e o nível do fundo da coluna.

O arquivamento destas informações é realizado em historiador de processos PI System instalado na rede corporativa. Através de fibra óptica, o sistema PIMS recolhe dados do analisador de amônia em intervalos de 5 segundos. Embora nenhuma das *tags* tenha o mecanismo de compressão dos dados ativado, algumas possuem um pequeno percentual de exceção configurado. Isto possibilita que a informação disponível (arquivada) não seja a última recebida.

O PIMS recebe ainda os dados do laboratório (vindo do sistema LIMS), de bases de dados relacionais corporativas e de sistemas supervisórios instalados na planta. Os dados de processo são então trabalhados pelo engenheiro de processos por meio dos aplicativos clientes do historiador – a saber: a extensão para Microsoft Excel (PI DataLink), a IHM para monitoramento gráfico do processo (PI ProcessBook) e o configurador da base de dados relacional (PI System Explorer) – e pelo sistema MES configurado paralelamente. Os dois sistemas, PIMS e MES, são conectados pela rede corporativa e podem receber dados das mesmas fontes. Os testes de exceção e compressão realizados pelo PIMS são descritos a seguir.

#### III.1.1.1 Algoritmo de Exceção

O teste de exceção é realizado para avaliar a necessidade de enviar os eventos coletados aos procedimentos seguintes, visto que estes podem sobrecarregar o servidor principal. O teste utilizado pelo PI System é um simples algoritmo de banda morta que ignora os valores dentro da mesma, à exceção do último valor contido nela. Este algoritmo utiliza 4 atributos de exceção, que devem ser configurados de forma individual para cada ponto do sistema (OSISoft, 2018).

A Figura III-3 exemplifica os atributos e funcionamento do algoritmo. O valor A é o último valor reportado ao servidor principal. Seguindo o objetivo de eliminar os pontos dentro da banda morta, a interface de comunicação relata o primeiro valor fora dos limites da banda e também envia o seu valor imediatamente anterior para o teste de compressão. Ou seja, a interface descarta o valor B e relata o valor D junto com seu valor prévio C. O teste de compressão determina se A e D são suficientes para reconstruir o sinal original ou se C necessita ser preservado (OSISoft, 2018).

O algoritmo se baseia em parâmetros de tempo e desvios absolutos e relativos. Os tempos máximo e mínimo entre eventos reportados ao servidor principal são, respectivamente, a exceção máxima (ExcMax) e a exceção mínima (ExcMin). O desvio absoluto (ExcDev) especifica a quantidade, em unidade de engenharia, que o valor recebido pode diferir do prévio. Finalmente, o desvio relativo (ExcDevPercent) é ajustado em porcentagem de span da variável (OSISoft, 2018).

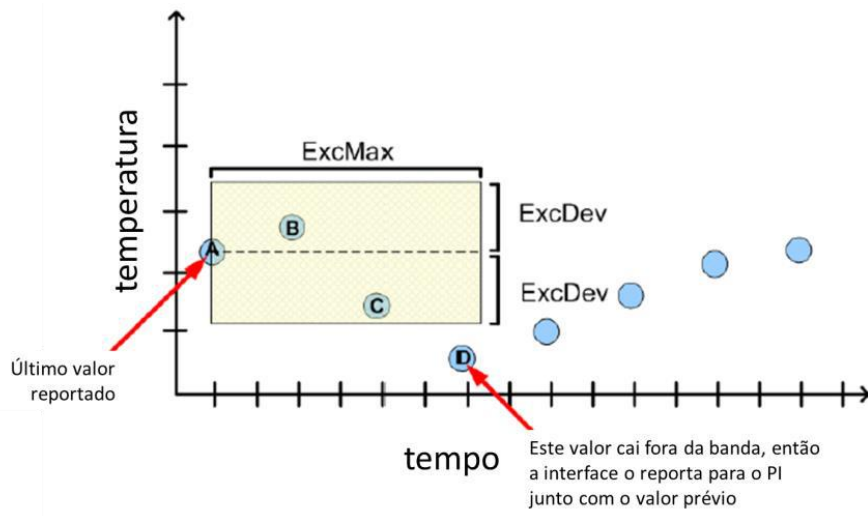


Figura III-3 Ilustração do teste de exceção com seus parâmetros principais.

Fonte: OSISoft, 2018 (Adptado).

### III.1.1.2 Teste de Compressão

O teste de compressão garante o armazenamento apenas dos dados históricos necessários à reprodução do sinal de origem com precisão (OSISoft, 2018).

Dentro do fluxo de dados do PI System, quando um novo valor *snapshot* é reportado junto com seu valor predecessor cria-se uma área de cobertura no formato de paralelogramo entre o último evento armazenado e o *snapshot* atual. Caso algum valor a partir do armazenado estiver fora dos limites do paralelogramo caracteriza-se um evento; o valor predecessor ao *snapshot* atual é armazenado no sistema e todos os valores entre este e o anteriormente armazenado são descartados. Se o limite no eixo do tempo for excedido, o *snapshot* atual é armazenado (OSISoft, 2018).

De modo análogo à banda do teste de exceção, os tempos máximo e mínimo de compressão (CompMax e CompMin, respectivamente), a metade da largura do paralelogramo (CompDev) e o desvio de compressão relativo (CompDevPercent) são os parâmetros a serem configurados no sistema. A Figura III-4 ilustra o procedimento de compressão.



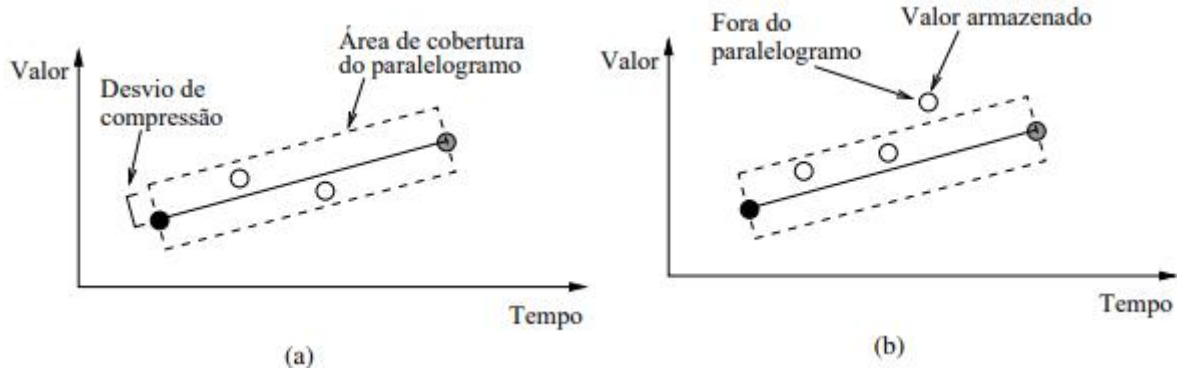


Figura III-4 Algoritmo de compactação de dados Swinging Doors.

Fonte: Souza *et al.*, 2005 (adaptado).

### III.1.2 Sensor virtual de teor de amônia no fundo da coluna

Com o objetivo de modelar um analisador virtual para prever o teor de amônia no fundo da coluna, Landim (2016) desenvolveu uma rede neuronal MLP com 10 neurônios de entrada, 12 neurônios na única camada escondida e um neurônio de saída. Estas variáveis foram previamente selecionadas a partir de um processo sistemático de redução de dimensionalidade por PCA e matriz de correlação e multicolinearidade. As funções de ativação para as camadas de entrada, escondida e de saída são respectivamente identidade, tangente hiperbólica e exponencial. A descrição completa da rede com seus valores de biases e pesos está disponível na Tabela A-1 (Apêndice A1).

No trabalho de Landim (2016), foram utilizados 1177 padrões repartidos entre 70% para treinamento, 15% para validação e 15% para teste no programa STATISTICA. O coeficiente de correlação  $R^2$  obtido foi de 0,834, com maior precisão para teores de amônia entre 0 a 8 ppm. No teste de verificação, realizado com novos 589 padrões, foi obtido um  $R^2$  igual a 0,690. A rede MLP 10-12-1 não foi implementada por Landim (2016).

O desenvolvimento da aplicação do sensor virtual neste trabalho foi planejado para oferecer flexibilidade e adaptabilidade tanto no que diz respeito à configuração de diferentes redes neuronais pelo usuário, quanto à implementação do modelo gerado nos aplicativos clientes. Dessa forma, foi priorizada a utilização de ferramentas com acesso à configuração pelo engenheiro de processos e o desenvolvimento de um algoritmo modular e eficiente a ser facilmente adaptado ao novo ambiente.

Novos testes e validações foram realizados com a rede neuronal desenvolvida e parametrizada por Landim (2016). Para o arquivamento dos resultados de predição e diagnóstico no sistema PIMS foi necessário a criação de 2 tags. A primeira, para resultados de predição do teor de amônia, configurada como *float* e a segunda, para receber a informação de qualidade da predição, configurada como *digital* (tag que recebe valor inteiro e o associa a uma classificação em texto).

Além destas duas tags, outras duas do tipo *float* foram criadas para armazenar valores das variáveis relacionais Razão Carga Vapor e Arraste. Os detalhes da implementação são discutidos ao longo dos próximos capítulos. Ao todo 3 etapas distintas foram executadas:

1. Desenvolvimento do algoritmo modular em linguagem de programação Visual Basic for Applications (VBA), escolhida por ser a predominante nos aplicativos clientes disponíveis. A validação do procedimento matemático da rede neuronal foi comparada com os resultados do teste de verificação fornecidos pelo *software* STATISTICA, obtidos por Landim (2016).
2. Implementação e validação do sensor *offline* através da extensão para Microsoft Excel. Consistiu no desenvolvimento da metodologia para configuração dos parâmetros de uma rede neuronal previamente treinada, configuração do método de aquisição de dados de entrada ao sensor virtual, e na implementação e validação dos mecanismos de diagnóstico e arquivamento dos resultados. Testes de performance da rede neuronal treinada por Landim (2016) também foram realizados.
3. Implementação e validação do sensor *online* através de 2 aplicativos clientes diferentes no sistema PIMS. Etapa composta pela criação de duas tags para arquivar dados das variáveis relacionais, pelo desenvolvimento das interfaces com o usuário, e pela validação *online* da aquisição dos dados da planta, do arquivamento da predição no historiador e do histórico de diagnósticos. Foi também desenvolvida a metodologia de arquivamento do resultado de diagnóstico.

## CAPÍTULO IV — ALGORITMO MODULAR

O procedimento de cálculo matemático da rede neuronal foi planejado e desenvolvido para ser adaptado para qualquer aplicação baseada em VBA. Para isso, ele foi isolado dentro de um módulo em comunicação com as etapas anteriores e posteriores. À semelhança do procedimento matemático, as demais etapas foram desenvolvidas também em módulos.

Ademais, para a configuração de cálculo de predição *online* e em tempo real essa estratégia foi expandida para a estrutura geral do código, fornecendo três sub-rotinas realizando separadamente a importação da parametrização da rede, a predição das variáveis de saída e a reinicialização do procedimento de predição até que o processo seja abortado. A mesma sequência de execução é mantida na predição *offline* na forma de uma única sub-rotina, uma vez que uma quantidade fixa de ciclos é identificada no início da execução. A sequência de execução é apresentada na Figura IV-1.

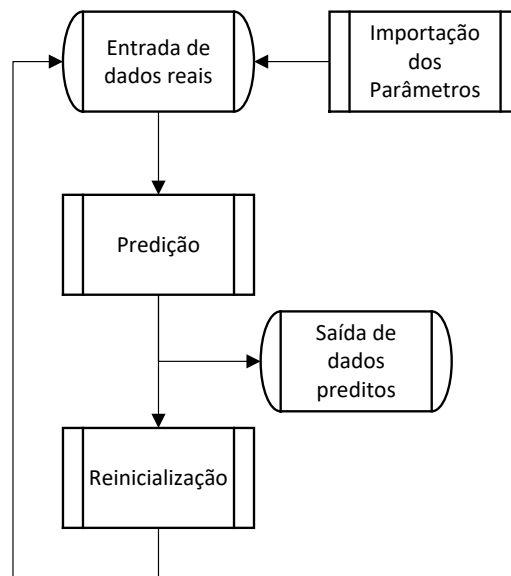


Figura IV-1 Sequência de execução das sub-rotinas.

Fonte: própria (2020).

### IV.1 Importação dos Parâmetros

A etapa de importação dos dados tem a dupla função de receber e preencher vetores e matrizes com os principais parâmetros da rede neuronal: quantidade de neurônios em cada camada (de entrada, escondida e principal); valores mínimos e máximos de treinamento de cada variável; valores de todos os biases e pesos; funções de ativação de cada camada.

São definidos também vetores com os nomes das tags de entrada para consulta no historiador e com os nomes das tags de saída para armazenamento no historiador e uma variável com nome da tag de diagnóstico para arquivamento dos resultados de diagnóstico. A definição dessas informações é realizada a cada sensor virtual implementado.

A estrutura cíclica permite reduzir o esforço computacional, uma vez que o preenchimento das variáveis é realizado uma única vez. Não obstante, a alteração dos parâmetros somente é permitida ao final da sequência de execução ou mediante interrupção da mesma.

## **IV.2 Predição**

A etapa de predição corresponde à maior parte do algoritmo, executando a identificação do modo de execução selecionado (somente no sensor *online*), importação do valor de cada variável de entrada, verificação primária, cálculo da predição, verificação secundária, diagnóstico e exportação do resultado.

### **IV.2.1 Verificações**

Enquanto a verificação primária avalia se algum valor de entrada é não numérico – e no caso da configuração *online*, se o tempo de atualização é não nulo ou inadequado – a verificação secundária avalia a ocorrência de erro no resultado da função de ativação. Em caso de uma resposta positiva para qualquer uma das duas verificações, a exportação do resultado é pulada e o ciclo reinicializado. No entanto, caso a verificação primária tenha resposta positiva para o teste do tempo de atualização o código é interrompido.

### **IV.2.2 Procedimento Matemático da Rede Neuronal**

O procedimento matemático da rede neuronal é estruturado para executar a cada camada as etapas de cálculo das sinapses entre os neurônios correspondentes. Os dados enviados à camada de entrada passam por pré-tratamento, em que as variáveis de entrada são escalonadas de 0 a 1, para normalizar a sua influência nas variáveis de saída. Portanto, os sinais de entrada correspondem às entradas normalizadas dentro do intervalo dos dados utilizados para treinamento da rede neuronal. A equação IV-1 ilustra o pré-tratamento realizado em cada neurônio da camada de entrada.

$$NE_i = \frac{E_i - Min_i}{Max_i - Min_i} \quad \text{Eq. IV-1}$$

Em que:

$E_i$ , é o valor da variável de entrada  $i$  em unidade de engenharia;

$Min_i$ , é o valor mínimo da variável de entrada  $i$  encontrado no conjunto de dados de treinamento da rede;

$Max_i$ , é o valor máximo da variável de entrada  $i$  encontrado no conjunto de dados de treinamento da rede;

$NE_i$ , é o valor normalizado da variável de entrada  $i$ ;

As saídas dos neurônios das camadas de entrada e escondida (nomeados genericamente de  $x_j$ ) são encaminhadas para os neurônios da camada seguinte, onde são processadas conforme as equações IV-2, IV-3 e IV-4. No módulo desenvolvido foram implementadas a possibilidade de utilização de todas as funções de ativação da Tabela IV-1, à exceção da gaussiana de base radial

O neurônio  $k$ , da camada oculta ou de saída, pode ser descrito matematicamente então pelo seguinte trio de equações:

$$u_k = \sum_{j=1}^m w_{kj} \cdot x_j \quad \text{Eq. IV-2}$$

$$v_k = u_k + b_k \quad \text{Eq. IV-3}$$

$$y_k = F(v_k) \quad \text{Eq. IV-4}$$

Em que:

$x_j$ , é o sinal de entrada  $j$  das camadas oculta e de saída;

$w_{kj}$ , é o peso da sinapse que liga o neurônio  $j$  ao neurônio da camada posterior  $k$ ;

$u_k$ , é o resultado da junção aditiva das entradas do neurônio  $k$ ;

$b_k$ , é o bias do neurônio  $k$ ;

- $v_k$ , é a ativação do neurônio k;
- $y_k$ , é a amplitude de saída do neurônio k;
- $F$  é a função de ativação.

Tabela IV-1 Principais funções de transferência para redes neuronais.

Função	Equação	
Logística	$F(v_k) = \frac{1}{1 + e^{-v_k}}$	Eq. IV-5
Tangente Hiperbólica	$F(v_k) = \tanh(v_k)$	Eq. IV-6
Exponencial	$F(v_k) = e^{v_k}$	Eq. IV-7
Seno	$F(v_k) = \text{sen}(v_k)$	Eq. IV-8
Identidade	$F(v_k) = v_k$	Eq. IV-9
Gaussiana de Base Radial	$F(x_j, C_j, \sigma_k) = \exp\left(-\frac{\sum_{j=1}^n [(x_j - C_j)^2]}{\sigma_k^2}\right)$	Eq. IV-10

Por fim, os sinais de saída dos neurônios da última camada ( $NS_j$ ) devem ser re-escalados em suas respectivas unidades de engenharia na faixa utilizada para treinamento da rede. A Eq. IV-11 apresenta o procedimento de pós-tratamento dos dados de saída.

$$S_j = NS_j (Max_j - Min_j) + Min_j \quad \text{Eq. IV-11}$$

Em que:

$S_j$ , é o valor da variável de saída j em unidade de engenharia;

$Min_j$ , é o valor mínimo da variável de saída j encontrado no conjunto de dados de treinamento da rede;

$Max_j$ , é o valor máximo da variável de saída j encontrado no conjunto de dados de treinamento da rede;

$NS_i$ , é o valor normalizado da variável de saída j.

### IV.2.3 Diagnóstico

Esta etapa é baseada na proposta de diagnóstico apresentada por Valdman (2010) tanto na ordem de verificações quanto nas possíveis classificações. O módulo de diagnóstico possui unicamente a função de identificar a qualidade da predição do sensor virtual e detalhar um possível tipo de erro.

As classificações para qualidade foram definidas como “erro” para caso de falha na função de ativação ou entrada não numérica, “ruim” para valores de entrada ou saída fora da faixa de treinamento da rede e “bom” se passar por todas as verificações. No caso da qualidade “ruim” o módulo ainda detalha quais variáveis estão fora da faixa.

Apesar de a visualização do diagnóstico pelo usuário diferir entre as aplicações desenvolvidas foi estabelecida uma classificação numérica para o tipo de erro, assumindo valores inteiros de 0 a 5 conforme abaixo:

- 0) Ausência de falha;
- 1) Tempo de atualização negativo ou não numérico (sensor *online*);
- 2) Valor de entrada não numérico;
- 3) Valor de entrada ou saída abaixo do valor mínimo de treinamento;
- 4) Valor de entrada ou saída acima do valor máximo de treinamento;
- 5) Falha na função de ativação.

### IV.3 Reinicialização

Por fim, a reinicialização aciona novamente a etapa de predição até que a quantidade fixa de padrões seja calculada (sensor *offline*) ou a execução seja abortada (sensor *online*). O algoritmo ainda permite que o intervalo de cálculo (sensor *online*) seja alterado sem pausar a execução. O detalhamento do algoritmo se encontra na Figura IV-2.

### IV.4 Validação

O algoritmo modular é validado nos 2 próximos capítulos a partir dos testes realizados com os sensores *offline* e *online*. Em complemento, testes de performance da rede neuronal desenvolvida por Landim (2016) são realizados com os sensores virtuais validados.

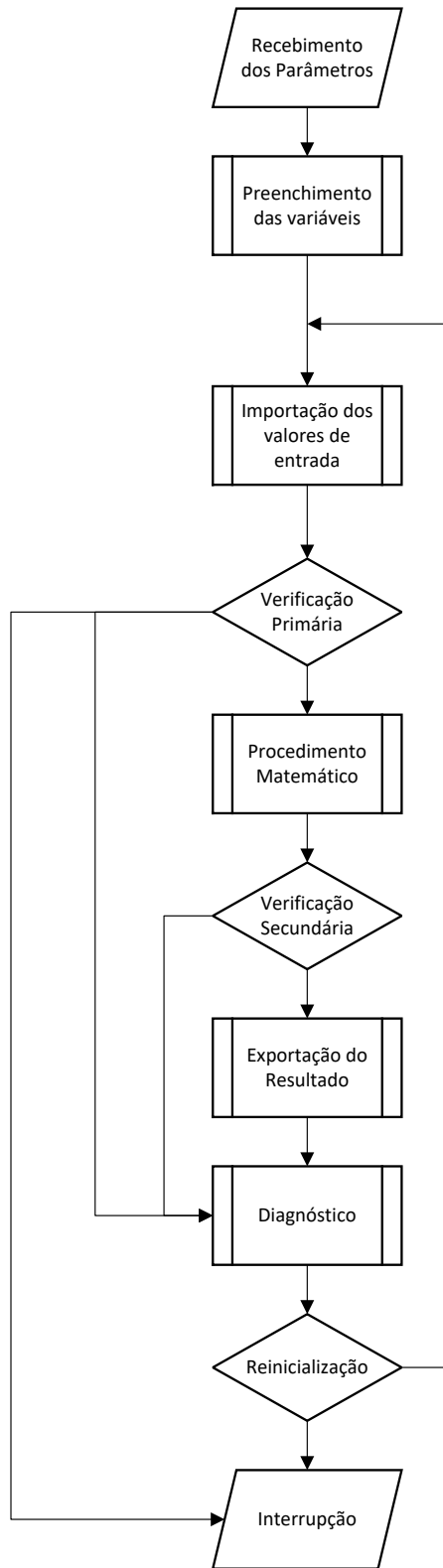


Figura IV-2 Módulos do algoritmo modular.

Fonte: própria (2020).



## CAPÍTULO V — SENSOR VIRTUAL *OFFLINE*

O sensor virtual *offline* foi planejado para oferecer a possibilidade de testar a rede neuronal em ambiente isolado, antes de arquivar os resultados no banco de dados histórico, em uma ferramenta de ampla utilização industrial para um engenheiro de processos: o Microsoft Excel e o PI DataLink, uma extensão do PI System disponibilizada para o Microsoft Excel.

A interface foi desenvolvida com o intuito de ser utilizada por um engenheiro de processos, familiarizado com o processo, a planta industrial e o sistema de automação vigente na planta, sem a necessidade de amplo conhecimento de programação computacional ou mesmo do processamento matemático utilizado por redes neurais. Na Figura V-1 podemos ver a lista suspensa para seleção da função de ativação e os botões de auxílio ao usuário.

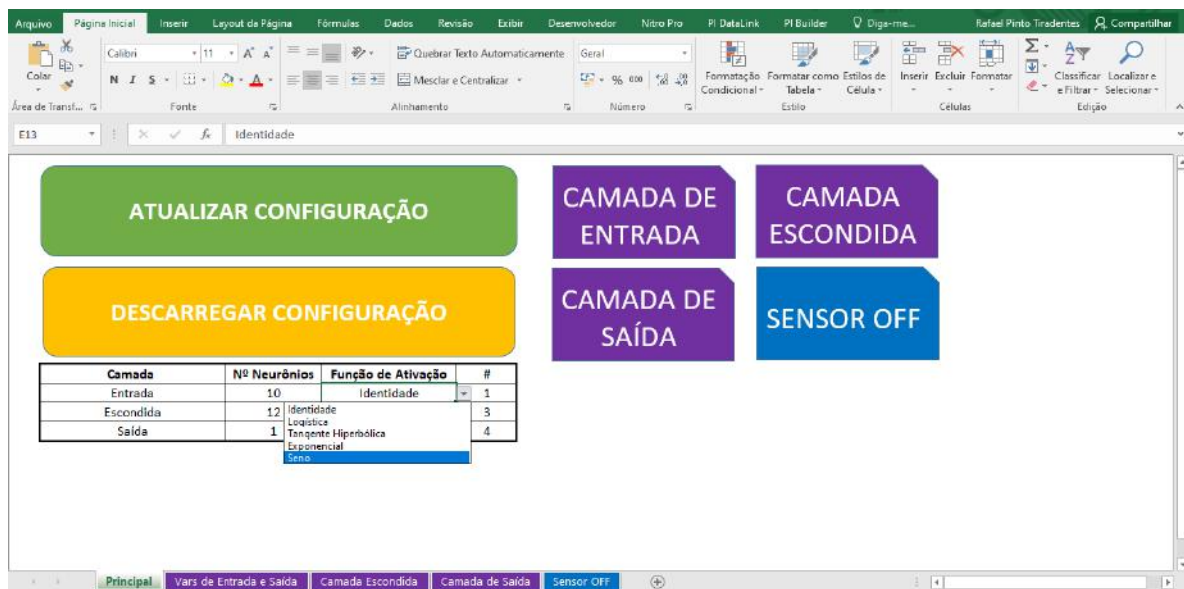


Figura V-1 Planilha desenvolvida para configuração dos parâmetros principais do sensor virtual.

Fonte: própria (2020)

Sendo assim, a partir da utilização do software de planilha eletrônica, a configuração do sensor *offline* é definida em 3 etapas. Primeiro é realizada a escolha na planilha “principal” dos parâmetros gerais da rede (número de neurônios e a função de ativação de cada camada). Em seguida, aciona-se um botão para limpar e, a partir dessas informações fornecidas na etapa anterior, são construídas as planilhas que vão receber os valores de biases, pesos, nomes das tags e a faixa dos valores de treinamento de cada variável. Nesta etapa, 3 planilhas são

disponibilizadas para edição: uma para o preenchimento dos nomes das tags e os intervalos de treinamento; outra para os pesos e biases da camada escondida; e a terceira, análoga à segunda, para a camada de saída.

A terceira e última etapa corresponde a preparação do ambiente final do sensor virtual para funcionar em comunicação com o sistema PIMS. Na planilha “sensor off”, são selecionados os tags correspondentes às informações fornecidas aos neurônios de entrada e os tags que recebem o valor fornecido pelo sensor virtual, todos disponíveis no sistema PIMS. Por fim, são extraídos da base de dados do historiador os padrões de entrada da rede neuronal.

As etapas de configuração são ilustradas pelo fluxograma da Figura V-2.

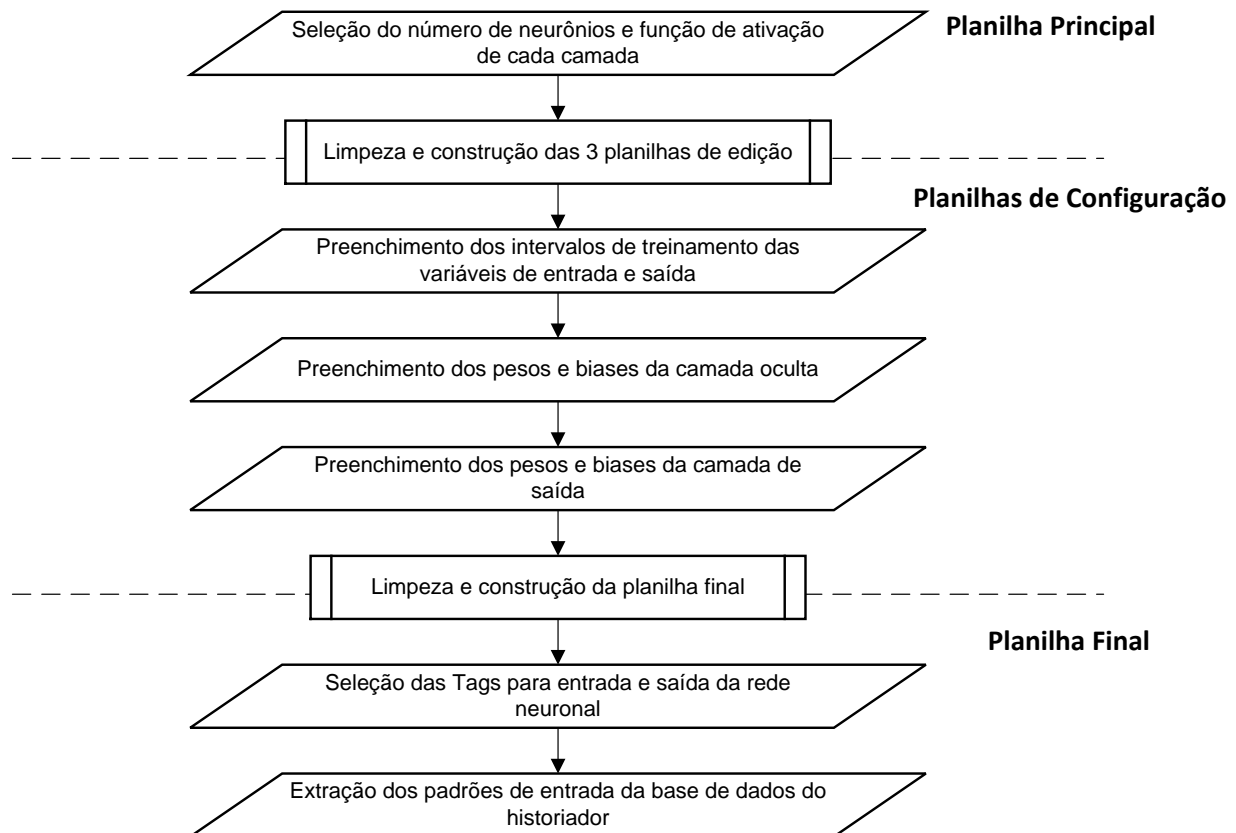


Figura V-2 Fluxograma das etapas de configuração do sensor *offline*.

Fonte: própria (2020).

## V.1 Desenvolvimento do Algoritmo e da Interface

A implementação do algoritmo modular do capítulo anterior para sensor *offline* é acompanhada de ajustes para adequação ao aplicativo Excel. No entanto, a estrutura básica do algoritmo é mantida.

Para garantir o caráter de ambiente de teste da ferramenta *offline*, o algoritmo foi implementado de modo a tomar como interface as planilhas descritas na seção anterior. Desse modo, os parâmetros são preenchidos a partir das planilhas de configuração (etapa 2 da configuração); a importação das variáveis é feita diretamente da planilha final (etapa 3 da configuração); e a exportação do resultado é realizada diretamente para a planilha final.

As funções pré-definidas pela extensão do historiador para Excel oferecem diferentes métodos para a aquisição de dados. Para garantir a consistência da predição nas fases de treinamento e implementação, é utilizado o método de busca de valores arquivados não interpolados a partir de um dado instante de tempo.

Além disso, durante a implementação da rede neuronal de Landim (2016), os valores para as variáveis relacionais razão carga vapor e arraste são obtidos por meio de expressões matemáticas, utilizando como variáveis os pontos individuais de vazão de vapor e vazão de arraste arquivados no historiador de processos.

Destaca-se ainda que o usuário tem a liberdade para colocar valores quaisquer para o cálculo. Assim, abre-se a possibilidade de utilizar a interface também como simulador de novas condições operacionais, desde que as variáveis de entrada estejam dentro do intervalo de treinamento.

Também o diagnóstico foi adaptado para apresentar visualmente seus resultados, por meio de mensagens ao usuário informando o número total de falhas. A identificação das falhas é apresentada nas células correspondentes aos dados de entrada de cada padrão, colorindo de vermelho valores fora da faixa de treinamento e de amarelo entradas não numéricas.

O processo de arquivamento dos resultados é realizado em duas etapas, permitindo ao engenheiro de processos confirmar os dados que serão historiados. A primeira etapa é responsável pelo procedimento do algoritmo já apresentado no capítulo anterior, dispondo o

resultado na planilha final. A segunda exporta os valores para a base de dados histórica do sistema PIMS. Somente datas em tempo real ou precedentes são armazenadas, sobrescrevendo valores com a mesma data.

## V.2 Validação do Sensor *Offline*

A validação do algoritmo modular foi realizada em conjunto com a validação do sensor *offline* (ambos desenvolvidos neste trabalho). O sensor *offline* foi configurado para receber a rede neuronal MLP 10-12-1 desenvolvida por Landim (2016). Para o teste foram utilizados os 589 padrões do teste de verificação utilizados por Landim (2016).

As etapas de configuração do sensor *offline* são representadas pelas figuras a seguir. Após o preenchimento do quadro principal na Figura V-3, o botão “atualizar configuração” prepara as planilhas de configuração (em roxo) para serem preenchidas na segunda etapa (como pode ser visto na Figura V-4). Finalmente, o botão “descarregar configuração” realiza a etapa 3, disponibilizando a planilha final para aquisição de dados, execução da rede neuronal e exportação dos resultados.

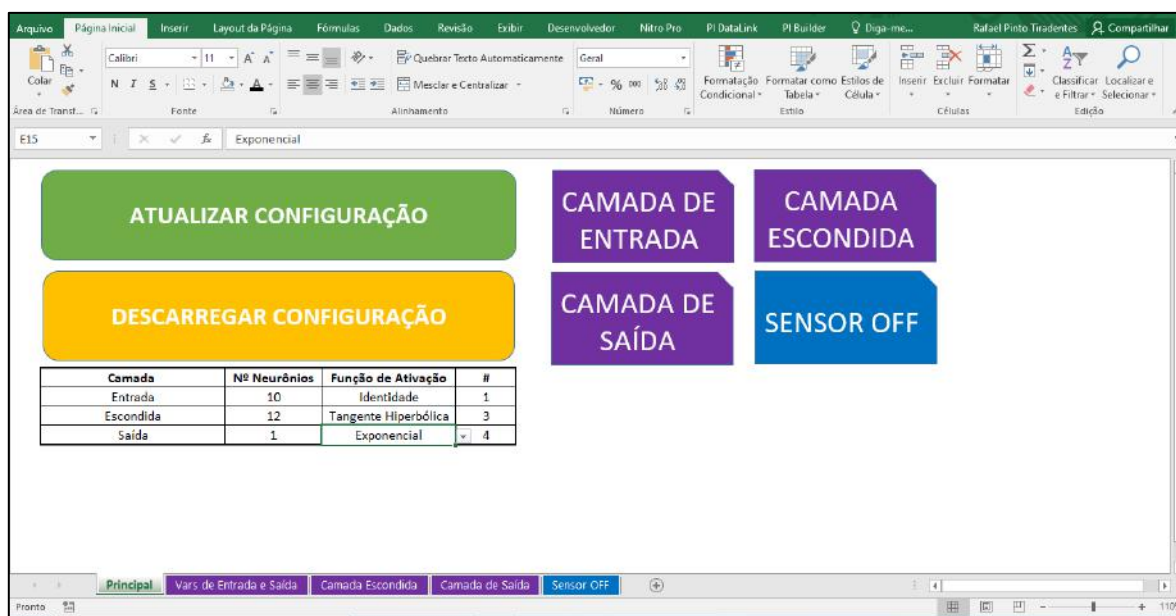


Figura V-3 Tela desenvolvida para configuração da primeira etapa do sensor *offline*.

Fonte: própria (2020).

Sinapse (j)	Neurônio (k)											
j	1	2	3	4	5	6	7	8	9	10	11	
0 (bias)	-0,44929736	-0,59742633	-0,39201827	-2,26342626	-0,02657249	0,77294533	-0,36408444	0,335788	0,114741	-0,71829	-1,03524	-0,11921
1	-1,4012073	1,4594014	2,50963833	-0,59222138	-2,41152362	-0,46444694	0,76770193	-0,01849	0,287588	-2,19953	1,804035	-0,08056
2	-0,13691332	-1,96582842	-0,07033479	1,49173524	-1,99838205	0,06716543	0,82089488	-0,04445	0,126811	0,867203	-0,2207	-0,22184
3	1,0671133402	-0,18389631	-1,3732496	0,64950965	-1,49647143	0,26904761	-0,26985714	0,710684	-0,31166	1,429018	0,626094	-0,11224
4	0,313002317	-1,90994442	0,79084563	-0,71957123	-1,74751127	0,14846395	-0,01144898	-0,10811	0,091031	-1,47905	0,49251	-0,03264
5	-0,76424924	0,83153759	-0,49609897	-0,46429509	-1,2734171	0,44882316	-0,20663608	-0,03311	0,061084	-0,56071	1,792261	0,132794
6	1,36334496	-1,78166441	-0,00372605	-0,66337835	-1,22765941	0,42324435	0,63872218	0,229294	-0,22571	5,489199	0,004027	-0,01562
7	-0,56572003	-2,17227334	-0,81313226	0,69073124	2,548184413	0,38159173	-1,05941442	0,20862	0,210114	0,434614	-1,40674	0,061843
8	1,15505975	-0,81557983	1,2718408	1,62555856	-0,156103571	-0,08317112	1,59254673	0,292231	-0,07463	1,860871	0,691561	0,235256
9	-0,70797716	-0,49478574	-0,85261469	1,35752569	2,125715015	0,05224051	-1,42081169	-0,48417	0,437492	-2,66727	1,038114	0,08389
10	-1,10877338	0,4562602	0,59832199	3,04906441	0,874966226	0,16281757	-0,81900148	-0,1079	0,278867	1,940756	1,595233	-0,06265

Figura V-4 Tela desenvolvida para configuração da segunda etapa do sensor *offline*.

Fonte: própria (2020).

Os 589 padrões foram processados em 3,2 segundos representando em média um tempo de 5,4 milissegundos por padrão. Foram encontradas 74 variáveis de entrada fora do intervalo de treinamento (Figura V-5).

REDE NEURONAL		Variáveis de Entrada										Variáveis de saída
EXPORTAR		1	2	3	4	5	6	7	8	9	10	1
130	126	45388,13281	12,60219797	1,38994441	89,7654953	2942,844727	67,93360138	12,01264668	90,1770477	-0,240792409	344,7709656	4,444661424
131	127	45325,94141	12,56751043	1,3919054					8619	-0,245081857	351,463501	4,161629092
132	128	45237,70313	12,57519209	1,3668813					8683	-0,233751774	362,1811523	3,518012326
133	129	45496,75	12,63522667	1,3772366					8372	-0,239411816	357,919342	3,903079556
134	130	45348,87891	12,58516087	1,381437					8555	-0,240019143	348,390686	3,781686003
135	131	45204,87109	12,56512655	1,3799878					7527	-0,237201259	341,9621582	3,461504503
136	132	45333,44141	12,5982102	1,376004					8412	-0,23837689	343,004303	3,389781778
137	133	45185,14453	12,58439976	1,3652996					8697	-0,22501187	325,7613525	2,2621701
138	134	46178,70313	12,55175909	1,3770308					8829	-0,285073817	329,5219116	4,396242131
139	135	46016,02734	12,37150176	1,3843671	105,0028381	4120,889648	69,44141388	12,14580727	89,0616608	-0,292569786	333,309906	3,745156599
140	136	45834,40234	12,71723764	1,4003092	91,28984813	4121,943359	69,80652618	12,14469147	89,9489212	-0,26992318	310,4107666	4,09791658
141	137	45625,96494	12,87421808	1,3701899	91,18087769	4107,428222	69,69972992	12,15037441	90,1472083	-0,241305366	281,112915	4,409054817
142	138	45625,33984	12,68527768	1,407587	101,2302017	4061,32866	70,34809113	12,079904816	90,2208633	-0,257011473	305,7763367	4,223667364
143	139	45278,60938	13,04245896	1,3554539	97,50876617	2202,085693	71,73619843	12,16743374	91,4405365	-0,203589663	271,4147949	4,555858744
144	140	45046,66406	12,621735	1,3805997	90,01512146	3245,66748	71,21852875	12,12527752	91,0595994	-0,231372312	292,7107849	4,609960604
145	141	45329,76172	13,30872711	1,3749795	112,041954	3322,599961	71,29592896	12,04333687	90,869812	-0,212148488	283,6644287	5,602887621
146	142	44922,57081	12,51379032	1,3888052	92,78952026	3588,837402	71,70243835	11,97928524	91,033699	-0,229865402	288,9911194	5,016875794
147	143	45052,75391	12,75722243	1,3873937	90,23561859	3349,492676	71,73886108	11,93010426	91,1995087	-0,224312561	281,0089111	5,907451044
148	144	45131,64944	12,91815734	1,3663378	91,14953613	3913,244873	71,35536957	11,9302063	91,1427841	-0,211701885	270,7679443	5,49514174

Figura V-5 Resultado do teste de validação do sensor *offline*.

Fonte: própria (2020).

Os valores preditos obtidos através do software STATISTICA – originalmente utilizado no trabalho de Landim (2016) – e através da interface desenvolvida no presente trabalho são comparados na Figura V-6. O desvio médio obtido foi nulo.

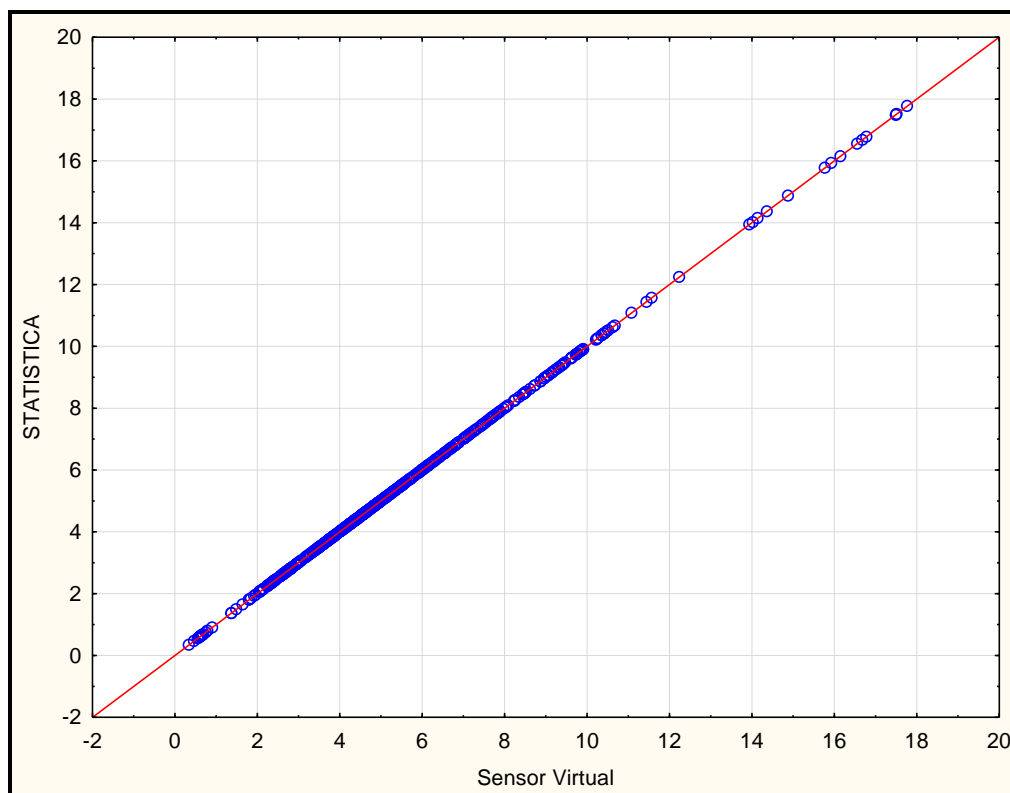


Figura V-6 Gráfico de dispersão entre o resultado do sensor virtual obtido pelo STATISTICA e obtido pela interface *offline* (PIMS/EXCEL) desenvolvida.

Fonte: própria (2020).

### V.3 Testes de Performance da Rede Neuronal

#### V.3.1 Qualidade da Predição

Para testar a performance da rede neuronal MLP 10-12-1 desenvolvida por Landim (2016) foi realizada a predição do teor de amônia através do sensor *offline* em intervalos de 5 minutos entre os dias 4 e 15 de junho de 2016. O período corresponde àquele utilizado por Landim (2016) para treinar a rede. O teste possibilitou ainda validar as etapas de aquisição de dados e exportação de resultados do sensor *offline* desenvolvido neste trabalho.

No total foram 3134 padrões aquisitados diretamente da base de dados histórica do PIMS através da extensão para Excel. Todo o conjunto de dados foi exportado em menos de 1 minuto. O tempo de execução da rede neuronal para todos os padrões foi de 8,4 segundos, representando em média um tempo de 1,7 milissegundos por padrão. O tempo de cálculo por padrão quase 3 vezes menor que do teste anterior é fruto da estrutura cíclica do algoritmo, que mantém os vetores e matrizes estáticos na memória, e que se revela no aumento do número de padrões a serem processados.

Um recorte do resultado é visto na Figura V-7. Os padrões que não tiveram predição calculada indicam a atuação dos módulos de verificação primária e secundária. Padrões que passaram pelas verificações e foram identificados pelo diagnóstico como “ruins” geram predições com elevado grau de incerteza. Resultados de teor de amônia maiores que 15 ppm – como observado – reforçam a importância da ferramenta de diagnóstico, pois não somente estão fora do intervalo de treinamento da variável de saída como também são bastante improváveis devido à característica de controle do processo.

		Variáveis de Entrada										Variáveis de Saída
		1	2	3	4	5	6	7	8	9	10	1
		Carga	Razão carga vapor	Arraste	Soda	Refluxo	T carga	pH carga	T flash	P flash	DP recheio sup	NH3 fundo
1731	10/06/2016 18:49	39050,83594	11,19557598	1,42922	177,3867645	914,5075684	57,46390533	11,96124458	84,5217438	-0,5286237	393,5718079	2,394252233
1732	10/06/2016 18:54	19,32178497	0,005335386	1,40765	0,813756943	1,722618937	57,46590042	11,96124458	84,5468521	-0,530416965	821,9571533	19,50745572
1733	10/06/2016 18:59	13,51700401	-3,423388828	-479,91	0,813756943	1,722618937	56,54297638	11,92308235	88,5986023	-0,173378706	0,624168694	
1734	10/06/2016 19:04	16,97647858	-4,997586431	-557,82	0,222535327	1,486918926	56,75507355	11,92308235	88,7802048	-0,090230405	0,624168694	
1735	10/06/2016 19:09	16,75636673	23,29070828	2637	0,222535327	1,486918926	57,46668243	11,63384247	88,95066836	-0,088829577	0,434905052	
1736	10/06/2016 19:14	14,38232231	-2,206380145	-290,91	0,24176991	1,486332059	56,55942535	11,63384247	88,95066836	-0,088800073	0,434905052	
1737	10/06/2016 19:19	17,36995697	47,76753684	5212,52	0,24176991	1,486332059	55,87863922	11,90633011	88,8550262	-0,090600014	0,449212968	
1738	10/06/2016 19:24	17,54950333	11,3841406	1232,52	0,229712188	1,342792153	55,22737885	11,90633011	88,8550262	-0,090920597	0,449212968	
1739	10/06/2016 19:29	16,29055786	0,0453922	5,28406	0,229712188	1,342792153	54,03903198	12,11474991	88,7689819	-0,089924663	0,437823951	25,56522557
1740	10/06/2016 19:34	13,95237064	0,085528078	11,6194	0,2532745	1,516616583	53,60159683	12,11474991	88,7689819	-0,089526288	0,437823951	26,45910506
1741	10/06/2016 19:39	14,81412697	0,014531673	1,86281	0,2532745	1,516616583	53,20243835	12,24939537	88,6721497	-0,089648038	0,449282587	4,432755766
1742	10/06/2016 19:44	14,91993713	0,014196701	1,80459	0,250756353	1,484287143	52,8848877	12,24939537	88,6721497	-0,088079058	0,449282587	3,308615749
1743	10/06/2016 19:49	14,83662605	0,027942961	3,56127	0,250756353	1,484287143	52,57082748	12,35025978	88,5888672	-0,087670863	0,460862994	21,41784193
1744	10/06/2016 19:54	13,22348499	0,012501848	1,78496	0,225880951	1,462686181	50,10949326	12,35025978	88,5888672	-0,086536914	0,460862994	3,678394183
1745	10/06/2016 19:59	13,51970577	0,012694274	1,77291	0,225880951	1,462686181	48,46434021	12,3889761	88,4987488	-0,088791162	0,48828125	3,932893913
1746	10/06/2016 20:04	12,67622948	0,01282901	1,91731	0,219790429	1,280608177	47,32512283	12,3889761	88,4987488	-0,091393881	0,48828125	8,616760215
1747	10/06/2016 20:09	15,25148392	0,014423496	1,79057	0,219790429	Bad	46,26294327	12,40806293	88,4342194	-0,089149609	0,457018614	
1748	10/06/2016 20:14	12,37471294	0,011580685	1,7723	0,220701277	Bad	45,29800415	12,40806293	88,4342194	-0,088784598	0,457018614	
1749	10/06/2016 20:19	15,63105106	0,014647645	1,7746	0,220701277	Bad	44,54769135	12,40779686	88,3414764	-0,08894235	0,485628814	

Figura V-7 Resultado do teste de aquisição e exportação de dados do sensor *offline*.

Fonte: própria (2020).

Vale ressaltar que cada padrão é composto por 10 variáveis de entrada. Dos 31340 dados de entrada, 5070 foram identificados como fora do intervalo de treinamento pelo diagnóstico; o equivalente a 16% do total. Isso representa 1262 dos 3134 padrões, ou 40% dos padrões. Dentre esses 40%, 51% (ou 648 padrões) são advindos de instantes de tempo em que a coluna ou estava inoperante ou estava iniciando ou finalizando sua operação. Desconsiderando estes padrões, temos uma porcentagem final de 21% de padrões fora do intervalo de treinamento.

A fim de investigar o comportamento temporal do sensor virtual, os resultados preditos foram exportados ao sistema PIMS e comparados graficamente com as medições do analisador industrial de amônia mediante gráficos de tendência criados na IHM do sistema PIMS (Figura V-8 e Figura V-9). A escala de tempo escolhida, de 1 dia, é a frequentemente utilizada por engenheiros de processo para avaliar a tendência e a dinâmica de variáveis de processo.

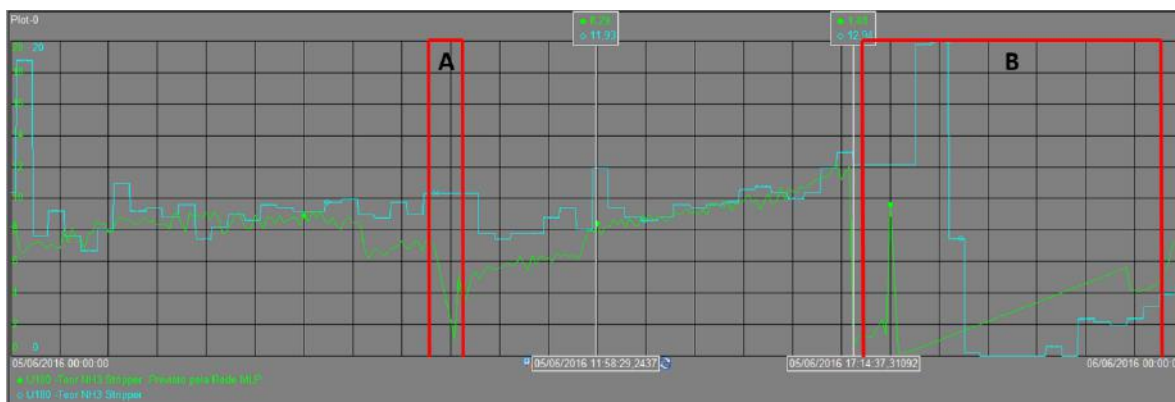


Figura V-8 Gráfico de tendência temporal dos resultados do sensor virtual e do analisador de amônia para o dia 05/06/2016.

Fonte: própria (2020).

A Figura V-8 dispõe em verde os resultados do sensor virtual, calculados a cada 5 minutos, para o dia 5 de junho de 2016, escritos na base de dados do historiador através da sub-rotina de exportação dos dados. Em azul estão os valores de teor de amônia inferidos pelo analisador a cada 20 minutos. Nas regiões sinalizadas com bordas vermelhas encontram-se os períodos (A e B) diagnosticados com alguma incerteza na medição.



Nos instantes de tempo A e B o resultado do sensor virtual é mais distante do alvo e também não apresenta grande correspondência com a tendência temporal dos dados. Pelo contrário, as demais regiões do gráfico apresentam menores desvios e maior correspondência no seguimento da tendência do analisador. A análise da dinâmica deste teste será aprofundada na seção seguinte.

A Figura V-9 apresenta mais variações na condição operacional. Os intervalos D, E e F identificados em vermelho correspondem ao estado da coluna fora de operação, onde não há novos resultados do analisador e poucos dados do sensor virtual, que são automaticamente interpolados pelo historiador. O único intervalo identificado em vermelho correspondente a valores de entrada fora da faixa de treinamento é o C. Nesse curto intervalo não é possível ver, novamente, uma correspondência com a tendência da variável alvo.

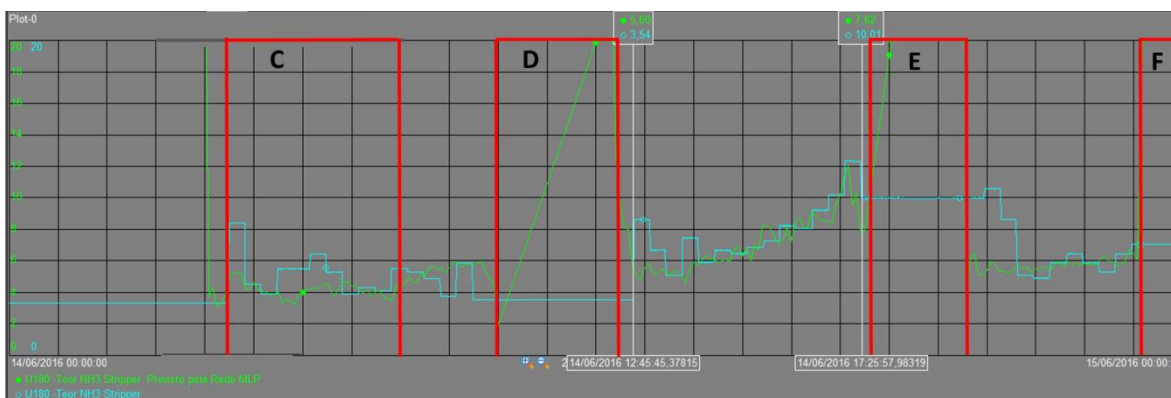


Figura V-9 Gráfico de tendência temporal dos resultados do sensor virtual e do analisador de amônia para o dia 14/06/2016.

Fonte: própria (2020).

O comportamento da predição do sensor virtual para os demais instantes é bastante similar ao do primeiro gráfico. Destaca-se uma particular correspondência da tendência dos dados em período de acréscimo contínuo da variável alvo. A investigação da dinâmica das variáveis será realizada na próxima seção para as datas discriminadas em branco das Figura V-8 e Figura V-9.

### V.3.2 Dinâmica da Predição

Os períodos de tempo correspondentes aos dias 5 de junho entre 12:00 e 17:00 e 14 de junho de 2016 entre 12:40 e 17:20 foram selecionados para um teste de dinâmica da resposta devido à similaridade na tendência crescente do teor de amônia, na faixa de variação do analisador e no intervalo de tempo que o evento ocorre (tanto na parte do dia quanto em duração).

É importante ressaltar que a rede neuronal utilizada neste trabalho somente é treinada para reconhecer padrões, ignorando qualquer representação da dinâmica do processo. Portanto, o que se buscou nesse teste foi extrair o máximo de informação, ou seja, maior número de predições significativas para o mesmo intervalo de 20 minutos do analisador industrial.

Foram testados 3 intervalos de cálculo: 1, 5 e 20 minutos. À redução do intervalo em 4 e 20 vezes do original se espera alcançar maior conhecimento dinâmico. Intervalos menores que 1 minuto não foram considerados devido ao incremento significativo do caráter ruidoso. Resultados dos testes são dispostos nas Figura V-10 e Figura V-11. Mais uma vez os valores do analisador (dados a cada 20 minutos) são apresentados em azul. As predições do sensor virtual testadas a cada 1, 5 e 20 minutos aparecem respectivamente em amarelo, preto e vermelho.

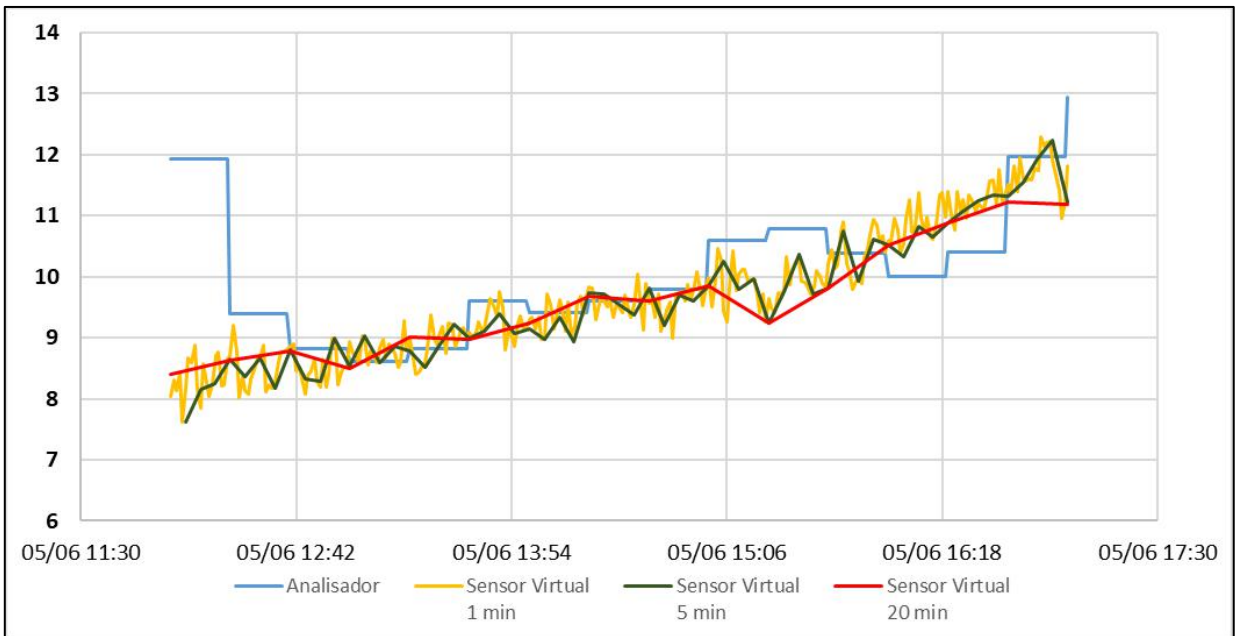


Figura V-10 Predição do analisador e do sensor virtual para 3 frequências diferentes para o dia 05/06/2016.

Fonte: própria (2020).

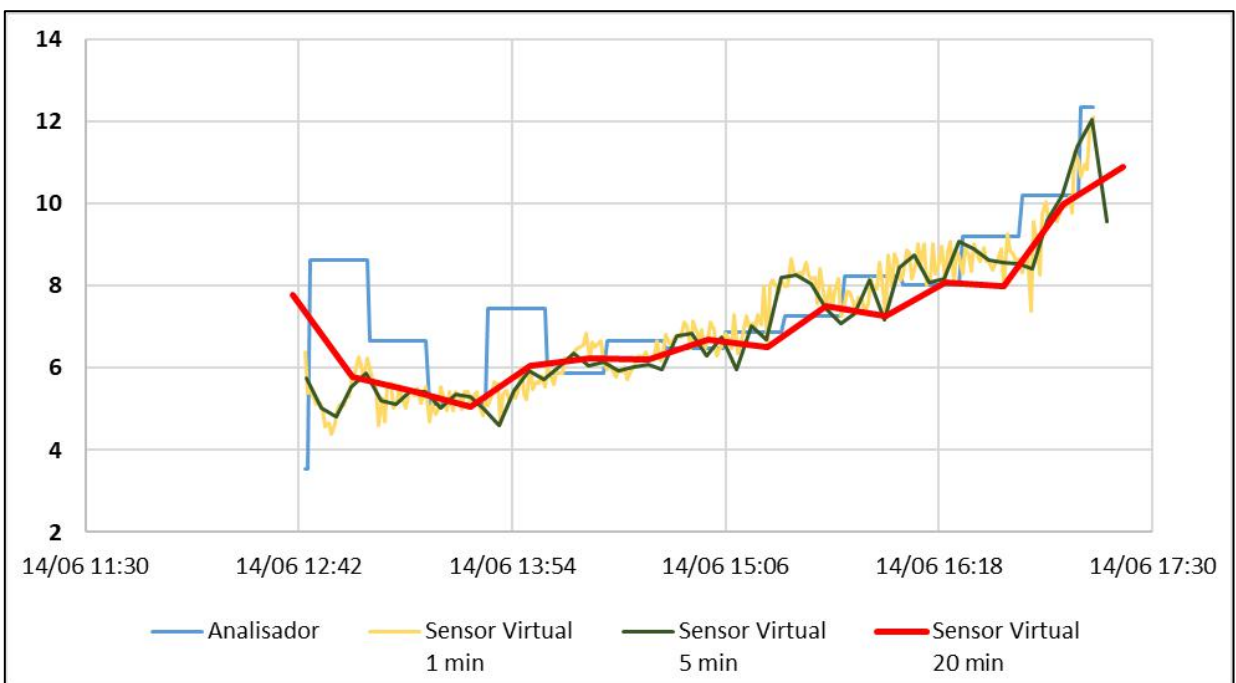


Figura V-11 Predição do analisador e do sensor virtual para 3 frequências diferentes para o dia 14/06/2016.

Fonte: própria (2020).

Todos os intervalos de tempo responderam satisfatoriamente, ou seja, sem apresentar desvios excessivos e repentinos em sua tendência. Como já observado no teste anterior, a predição a cada 20 minutos é capaz de simular uma dinâmica próxima do real ainda que a estimativa dos dados não seja de grande acurácia.

Nestes 2 períodos, foram calculados 2 indicadores de desempenho para cada intervalo de cálculo utilizado: o quadrado do coeficiente de correlação ( $R^2$ ) entre a resposta do sensor virtual e do analisador industrial e o erro quadrático médio (calculado pelo somatório dos desvios quadrados divididos pelo número de pares utilizados). A resposta do analisador industrial foi considerada constante durante o intervalo de 20 minutos. As Tabela V-1 e Tabela V-2 apresentam os resultados obtidos.

Tabela V-1 Resultados dos parâmetros quantitativos do teste para o dia 05/06/2016.

<b>Intervalo de Cálculo</b>	<b><math>R^2</math></b>	<b>Erro Quadrático Médio</b>
<b>1 minuto</b>	0,2297	1,2157
<b>5 minutos</b>	0,2057	1,2922
<b>20 minutos</b>	0,3526	1,3154

Fonte: própria (2020)

Tabela V-2 Resultados dos parâmetros quantitativos do teste para o dia 14/06/2016.

<b>Intervalo de Cálculo</b>	<b><math>R^2</math></b>	<b>Erro Quadrático Médio</b>
<b>1 minuto</b>	0,4863	1,6941
<b>5 minutos</b>	0,5388	1,8065
<b>20 minutos</b>	0,4799	2,3113

Fonte: própria (2020)

O período do dia 5 apresenta um desvio negativo no  $R^2$  em 49% ao valor obtido de 0,690 por Landim (2016) no teste de verificação. Apesar do menor erro quadrático médio registrado para o dia 5, a correlação de todos os intervalos é muito distante da condição esperada. Por outro lado, o período do dia 14 alcançou desvio negativo no  $R^2$  de 22%. Para a avaliação quantitativa a seguir somente será considerado o período do dia 14, devido ao menor desvio.

A expectativa quanto ao aumento da frequência de cálculo se mostra parcialmente verdadeira. Ainda que a utilização de frequências maiores que a de treinamento possibilitem mais previsões intermediárias entre os resultados do analisador (e conseqüentemente informação adicional), se observa pequena melhora na previsão expressa no melhor valor de  $R^2$ . Tão pouco há perda de qualidade quanto ao erro quadrático médio, o que legitima ainda a escolha por uma frequência maior.

Qualitativamente, dentre os dois intervalos de cálculo propostos com frequências maiores destaca-se um comportamento central do intervalo de 5 minutos frente ao de 1. Em conseqüência, o intervalo de 5 minutos se aproxima de uma média móvel do intervalo de 1 minuto. O mesmo não ocorre ao intervalo de 20 minutos frente ao de 5. Outra característica marcante é já o caráter ruidoso dos resultados em amarelo (1 minuto).

Além disto, para o dia 14 o  $R^2$  do intervalo de 5 minutos mostrou-se superior ao de 1 minuto mantendo erro quadrático médio ainda menor que para o intervalo de 20 minutos. Estas observações conduziram a decisão de estabelecer o intervalo de cálculo do sensor *online* em 5 minutos.

## CAPÍTULO VI — SENSOR VIRTUAL *ONLINE*

A implementação do sensor *online* foi desenvolvida através de 2 ferramentas clientes do sistema PIMS, seguindo a direção do ambiente mais externo para o mais interno ao historiador de processos. As duas ferramentas – extensão para Microsoft Excel (PI DataLink) e a IHM (PI ProcessBook) – oferecem suporte à programação baseada em VBA.

O procedimento matemático da rede neuronal manteve-se o mesmo das 2 etapas anteriores. No entanto, como o sensor *online* busca as informações diretamente da base de dados histórica do PIMS, foi necessário criar 2 novas tags no sistema PIMS; uma para cada variável relacional da rede neuronal (razão carga vapor e arraste). Estes pontos foram configurados sem exceção e compressão e foram definidos para calcular e historiar um novo valor a cada novo valor adquirido para as variáveis de processo correspondentes (Vazão de Carga, Vazão de vapor, Vazão de arraste).

### VI.1 Desenvolvimento do Algoritmo e da Interface do sensor virtual em planilha eletrônica

Uma das alternativas para implementação do sensor virtual *online* foi desenvolvida utilizando o software de planilha eletrônica Microsoft Excel (Versão 2019), baseado no sensor *offline* previamente desenvolvido. O que compete ao sensor *online* especificamente são 2 novas planilhas: uma para ativar o sensor e selecionar o modo de execução e outra para arquivar os resultados do diagnóstico. As duas são independentes do sensor *offline*. A primeira planilha (“sensor ON”) fornece os modos de execução automático ou manual, enquanto a segunda planilha (“Diagnóstico”) registra os resultados do algoritmo principal.

Por meio da primeira planilha (Figura VI-1) o usuário seleciona primeiro o modo de cálculo (Automático/Manual). No modo automático, procedimento matemático da rede neuronal é reiniciado automaticamente com uma frequência de atualização definida pelo usuário. Neste modo, o botão para início (“start”) ou parada (“stop”) do sensor é habilitado e o usuário deve configurar o tempo de atualização nas caixas de texto. No modo manual, o botão de cálculo (“calcular”) é habilitado e a cada vez que é pressionado, o procedimento matemático da rede neuronal é executado uma única vez. O estado atualizado do sensor virtual é disposto abaixo dos 3 botões.

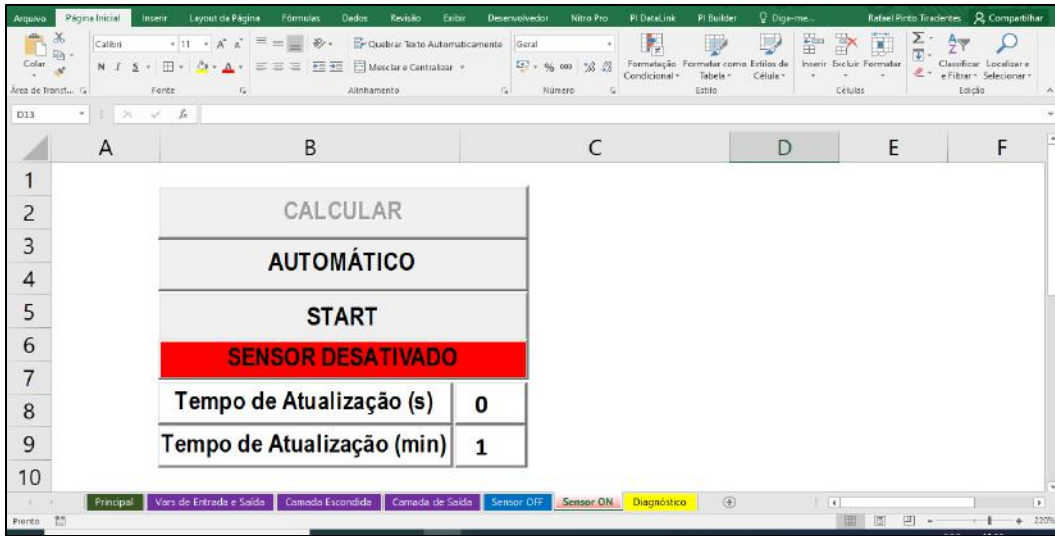


Figura VI-1 Tela de acionamento do sensor virtual *online*.

Fonte: própria (2020).

Na segunda planilha (Figura VI-2) o resultado do diagnóstico do cálculo de predição é disposto através de colunas (Data/hora, qualidade da predição da rede neuronal, tipo e descrição) cada vez que o procedimento matemático é solicitado. Essa interface permite que as informações sejam filtradas em função da data, qualidade, tipo de erro e descrição.

	A	B	C	D	E
	#	Data	Qualidade	Tipo de Erro	Descrição
2	1	14/11/2019 14:05:29	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
3	2	14/11/2019 14:05:46	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
4	3	14/11/2019 14:06:32	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
5	4	14/11/2019 14:06:44	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
6	5	14/11/2019 14:08:50	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
7	6	14/11/2019 14:09:43	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
8	7	14/11/2019 14:09:48	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
9	8	14/11/2019 14:09:53	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
10	9	14/11/2019 14:09:58	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
11	10	14/11/2019 14:10:03	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
12	11	14/11/2019 14:10:08	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
13	12	14/11/2019 14:10:13	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
14	13	14/11/2019 14:10:18	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
15	14	14/11/2019 14:10:23	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
16	15	14/11/2019 14:10:28	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
17	16	14/11/2019 14:10:33	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
18	17	14/11/2019 14:10:38	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
19	18	14/11/2019 14:10:43	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
20	19	14/11/2019 14:10:48	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
21	20	14/11/2019 14:10:53	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
22	21	14/11/2019 14:10:58	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.
23	22	14/11/2019 14:11:03	RUIM	4	Variável de entrada de número 3 acima do valor máximo de treinamento.

Figura VI-2 Tela de diagnóstico do sensor *online*.

Fonte: própria (2020).

O sensor *online* implementado na ferramenta Excel utiliza todos os módulos do algoritmo proposto, sem a necessidade de execução em dois passos como visto no sensor *offline*, incluindo os módulos de verificação primária e de reinicialização.

A importação dos valores das variáveis de entrada e exportação dos resultados é realizada diretamente com o historiador de dados por meio dos nomes das tags previamente configuradas no PIMS, fornecidas na configuração inicial. A consulta aos resultados pode ser feita utilizando os recursos da extensão do PIMS para planilha eletrônica.

O módulo de diagnóstico é disposto na própria ferramenta, de forma a trabalhar com os dados relacionais calculados pelo software em comparação aos dados obtidos a partir da base de dados histórica do sistema PIMS.

## **VI.2 Desenvolvimento do Algoritmo e da Interface do sensor virtual na Interface Homem-Máquina**

A segunda ferramenta em que foi implementado o sensor virtual *online* é a interface gráfica para acompanhamento de processos do sistema PIMS. A Figura VI-3 apresenta a tela desenvolvida para configuração e execução da rede neuronal no sistema PIMS (PI System da fabricante OSIsoft, versão 2016), seguindo metodologia análoga àquela adotada pelo o sensor *offline*. Os principais parâmetros a serem configurados são:

**1º) Parâmetros Principais:** nome do nó do servidor PIMS disponível na rede; número de neurônios e a função de ativação de cada camada da rede neuronal; nomes das tags de entrada, de saída e de diagnóstico.

**2º) Configuração das camadas:** intervalos de treinamento das variáveis de entrada e de saída; pesos e biases da camada escondida; pesos e biases da camada de saída.

**3º) Seleção do modo de cálculo:** automático ou manual; início (“start”) ou interrupção (“stop”); tempo de atualização.



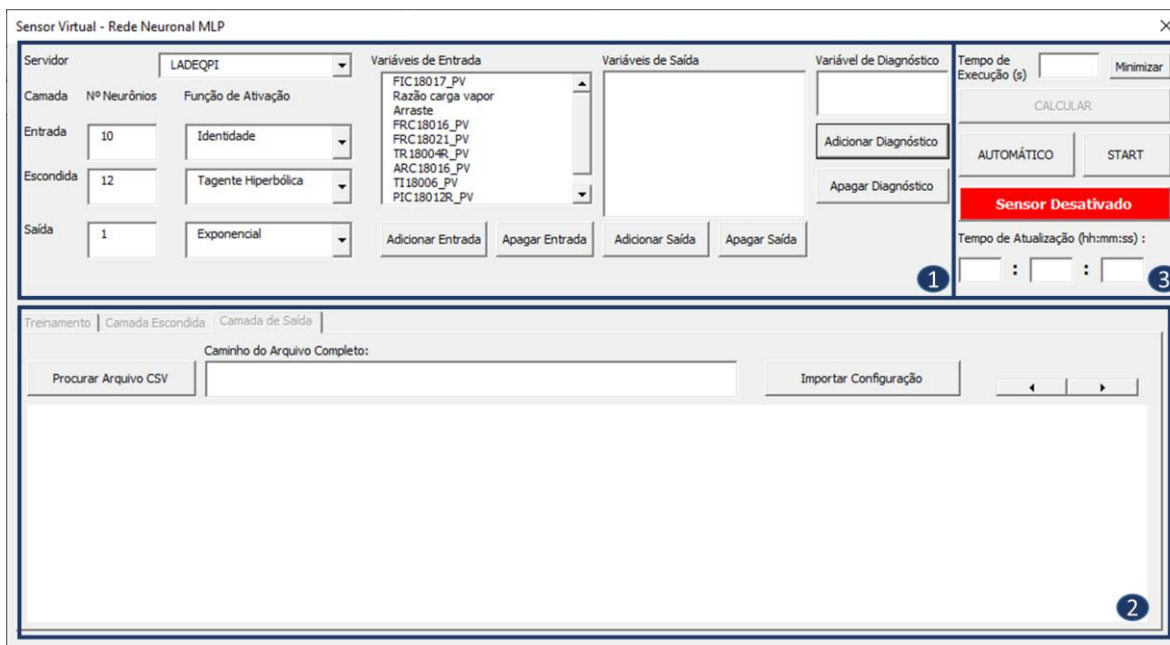


Figura VI-3 Tela de configuração e acionamento do sensor *online* implementado na IHM.

Fonte: própria (2020).

Para agilizar a configuração das camadas foi desenvolvido um módulo para importar os dados das planilhas preenchidas através da ferramenta desenvolvida para o sensor *offline*, salvas em formato CSV. Quanto à seleção do modo de cálculo e acionamento do sensor, foi reproduzida a metodologia da ferramenta anterior.

A implementação do diagnóstico é feita mediante a seleção de uma tag *digital* (que recebe valor inteiro e o associa a uma classificação em texto) para receber os resultados do módulo homônimo. Embora não seja possível a visualização da qualidade da predição em modo tão detalhado quanto nas ferramentas anteriores, os recursos do sistema PIMS permitem que a variável diagnóstico seja apresentada em conjunto com as de entrada e de saída em gráfico de tendência ao longo do tempo.

Assim como as ferramentas anteriores, o sensor *online* implementado na IHM do sistema PIMS foi desenvolvido em ambiente VBA. São utilizados os mesmos módulos do algoritmo proposto. Entretanto, o código foi realocado na nova ferramenta com pequenos ajustes de compatibilidade ao novo ambiente. A configuração dos parâmetros é feita diretamente na tela desenvolvida e a aquisição de dados é feita através dos recursos do sistema PIMS.

A consulta dos resultados nesta ferramenta é realizada através de gráficos de tendência, o que motivou a alteração estrutural da disposição do diagnóstico. A tag digital, fornecida pelo usuário, recebe os valores de qualidade (*erro*, *ruim* e *bom*) propostos na metodologia do presente trabalho. Essa modificação se alia as potencialidades da IHM, possibilitando uma visualização do diagnóstico integrada com as demais variáveis de interesse do engenheiro de processos.

### VI.3 Validação do Sensor *Online*

O sensor virtual *online* foi mantido em execução por 3,5 horas no dia 8 de janeiro de 2020. Foi mantido em modo automático com o intervalo de cálculo de 5 minutos, conforme resultado do teste de performance obtidos a partir da análise do sensor offline. Os sensores *online* implementados em planilha eletrônica e na IHM do sistema PIMS foram mantidos em execução no mesmo computador simultaneamente, exportando o resultado para tags distintas.

As duas ferramentas funcionaram corretamente, sem interrupções inesperadas, durante todo o teste. Não foram observados ciclos de cálculo superiores a 1 segundo. A facilidade de implementação e alteração dos parâmetros foi alcançada junto com a interface amigável e compatível com o PIMS.

Ao todo, 43 valores de teor de amônia preditos para cada sensor *online* e 11 valores do analisador industrial foram arquivados na base de dados do historiador. Novamente os parâmetros  $R^2$  e erro quadrático médio foram calculados em relação aos valores do analisador industrial. Foi assumido que durante o intervalo de 20 minutos os valores do analisador industrial são iguais ao primeiro valor adquirido. A Tabela VI-1 apresenta os parâmetros calculados para cada sensor *online*.

Tabela VI-1 Resultados dos parâmetros quantitativos do teste para o dia 08/01/2020.

<b>Sensor <i>Online</i></b>	<b><math>R^2</math></b>	<b>Erro Quadrático Médio</b>
<b>Planilha Eletrônica</b>	0,0029	11,3166
<b>IHM</b>	0,0029	11,3166

Fonte: própria (2020)

A Figura VI-4 dispõe os resultados graficamente. Em verde está representado o analisador industrial, em azul os resultados do sensor virtual *online* via planilha eletrônica e em amarelo do sensor virtual *online* via IHM do sistema PIMS.

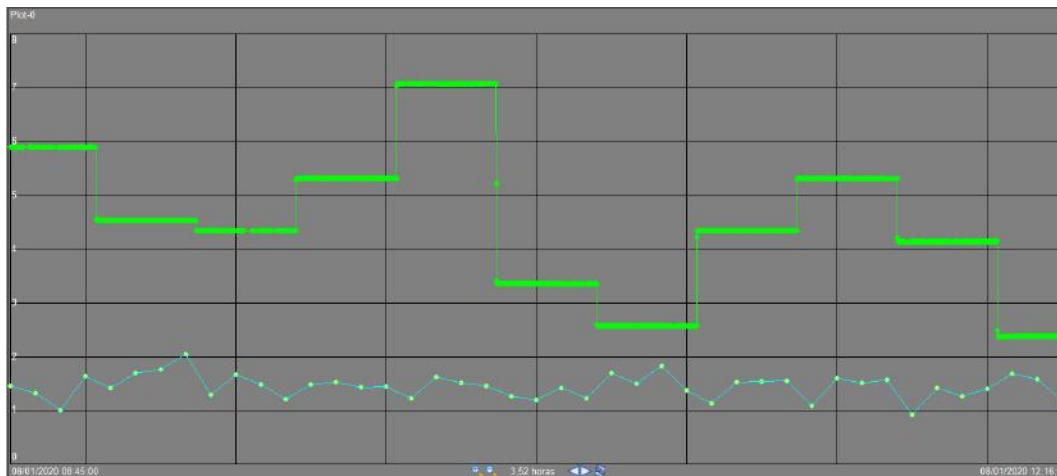


Figura VI-4 Gráfico de tendência temporal dos resultados do sensor virtual e do analisador de amônia para o dia 08/01/2020.

Fonte: própria (2020)

Na Figura VI-4 é possível observar que os resultados para os sensores *online* são coincidentes. Por isto, os resultados dos parâmetros  $R^2$  e erro quadrático médio são idênticos. Portanto, a adaptação do algoritmo para a IHM do sistema PIMS foi bem-sucedida. Também o módulo de diagnóstico teve papel relevante no teste.

Durante todo o período do teste foram observadas variáveis de entrada fora do intervalo de treinamento e indicado resultado ruim (ou seja, pouco confiável) pelo módulo de diagnóstico. Conseqüentemente, o valor de  $R^2$  para o período atual é muito baixo e bastante inferior ao encontrado para o ano de 2016. Também o erro quadrático médio é bastante superior ao identificado para o ano de 2016. Isto não somente indica que o sensor virtual pode estar descalibrado, mas valida o funcionamento da ferramenta de diagnóstico.

## VI.4 Análise Estrutural dos Sensores *Online* e *Offline*

As três implementações desenvolvidas neste trabalho possuem suas próprias particularidades, inerentes à estrutura escolhida. Particularmente, a distinção entre os sensores *offline* e *online* surge do objetivo de cada ferramenta. Enquanto a primeira foi desenvolvida para execução de ajustes finais da rede neuronal, a segunda foi idealizada para o cálculo cíclico da predição.

O aplicativo Excel, mais externo ao sistema PIMS, é vantajoso ao sensor *offline* na medida em que há prévia familiaridade do usuário e todos os dados de entrada e de saída podem ali ser dispostos. Cria-se um ambiente seguro para testes do sensor. Embora o sucesso deste, a potencialidade máxima dos modos de cálculo e diagnóstico só é alcançada no sensor *online*.

Os dois sensores *online* desenvolvidos utilizam os mesmos módulos de cálculo, acionando repetidamente sub-rotinas em linguagem de programação VBA, funcionando de forma automática e sem a interferência do usuário. Devido a estratégia proposta para o desenvolvimento das ferramentas há a limitação conhecida de que é necessário manter o aplicativo cliente aberto. A solução quanto a isto estaria em implementar o sensor *online* em instâncias mais internas ao sistema PIMS.

Para uma migração do sensor *offline* para *online* simplificada e de curta duração, a ferramenta *online* em Excel se revela mais vantajosa. O arquivamento dos resultados de diagnóstico é bastante rico em informação, sendo útil para detalhamento da saúde do sensor virtual. Entretanto, esses resultados se acumulam rapidamente, tornando o tamanho do arquivo Excel excessivamente volumoso.

Por outro lado, uma aplicação mais robusta e que consuma menos esforço computacional é fornecida pela implementação *online* na IHM do sistema PIMS, pois a comunicação com o servidor principal do historiador é mais simples e confiável.

O arquivamento dos resultados de diagnóstico no próprio historiador reduz ainda o volume do arquivo, ao mesmo tempo que fornece a opção de visualizar a qualidade do sensor ao longo do tempo. Esta implementação também permite trabalhar com outras telas gráficas da IHM simultaneamente.

# CAPÍTULO VII — CONCLUSÃO

## VII.1 Conclusões

Este trabalho propôs 3 implementações diferentes de um sensor virtual para redes neurais MLP com uma camada escondida, partindo de uma mesma metodologia de flexibilidade de configuração apoiada em um algoritmo modular. Os módulos deste algoritmo foram adaptados caso a caso para adequação ao aplicativo e melhor usufruto do ambiente. Como estudo de caso, foi utilizada a rede MLP 10-12-1 desenvolvida por Landim (2016) para predição do teor de amônia na saída de fundo de uma coluna *stripper*.

O algoritmo modular se comprovou adaptável e genérico. Enquanto o processamento computacional da rede se manteve intacto, os módulos de importação e exportação de resultados foram ajustados isoladamente. A estrutura cíclica permitiu reduzir o esforço computacional das ferramentas, principalmente para o sensor *offline*. Módulos de verificação e diagnóstico se mostraram versáteis ao longo do trabalho. O teste de validação do sensor *offline*, realizado com os dados do teste de verificação da rede (Landim, 2016), reproduziu os resultados obtidos na literatura e não indicou desvios significativos.

O sensor virtual *offline* provou-se bastante eficaz como ambiente de ajustes finais da rede. O *software* escolhido para implementação (Microsoft Excel) une a potencialidade do historiador de dados – possibilitando previsões para dados antigos ou simular condições operacionais – com os recursos nativos familiares ao usuário final.

Dentre os intervalos de cálculo testados, foi identificado o de 5 minutos como mais adequado. Ao mesmo tempo que fornece mais estimativas entre as medições do analisador, não gera uma visualização oscilatória, como vista para intervalo de 1 minuto. O intervalo selecionado foi implementado nos dois sensores virtuais *online* desenvolvidos.

Metodologias análogas de configuração e execução foram utilizadas nas ferramentas desenvolvidas para o sensor *online* de modo a garantir facilidade de migração da ferramenta *offline* e o cálculo contínuo das previsões. A implementação do sensor *online* em planilha eletrônica revelou-se útil para o detalhamento da qualidade da predição e utilização de curta duração, uma vez que o tamanho do arquivo cresce continua e rapidamente. Por sua vez, a implementação do sensor *online* na Interface Homem-Máquina oferece funcionamento mais

robusto do sensor, adequação à operação contínua e suporte à visualização gráfica das estimativas. O teste de validação dos sensores *online* não apresentou interrupções inesperadas ou ciclos de cálculo superiores a 1 segundo.

O foco deste trabalho esteve mais nas ferramentas desenvolvidas, do que na rede neuronal utilizada como estudo de caso. De toda forma, testes de performance da rede de Landim (2016) foram realizados. Quanto à qualidade de sua estimativa, a avaliação para o mês de junho de 2016 (período do treinamento) indicou o percentual de 21% dos 3134 padrões com pelo menos uma variável fora do intervalo de treinamento e com a coluna em operação. Contudo, para este período foi observada correspondência com a tendência da resposta do analisador.

Por outro lado, a avaliação da performance da ferramenta online na planta industrial durante o período analisado de 8 de janeiro de 2020, revelou uma baixa correlação com a tendência do processo evidenciada previamente pelo diagnóstico do sensor virtual. Logo, a utilização do sensor virtual *online* para a predição do teor de amônia no fundo da coluna de recuperação requer etapa de manutenção. Ou seja, necessita o recolhimento de padrões atuais e o retreinamento da rede neuronal, além da atualização da configuração do sensor virtual, conforme metodologia abordada neste trabalho.

## **VII.2 Sugestões para Trabalhos Futuros**

A princípio sugere-se o estudo aprofundado da ferramenta de diagnóstico dos sensores *online*. Quanto a rede neuronal utilizada neste trabalho é indicada a sua manutenção por meio das metodologias desenvolvidas neste trabalho e por Landim (2016). Em relação às ferramentas desenvolvidas neste trabalho é sugerida a sua validação com testes de maior duração e realizados por usuários finais (engenheiros de processo). Propõe-se ainda a implementação da metodologia e ferramentas em outros sistemas PIMS.

No âmbito da arquitetura de automação, recomenda-se o estudo da implementação do sensor em instâncias mais internas ao historiador de processos – aplicativos clientes para análise de dados agendadas ou configuração de tags complexas. Desse modo, seria possível alcançar o cálculo ininterrupto da previsão; independente do acionamento do usuário final e da permanência do aplicativo cliente aberto no computador do engenheiro.

## REFERÊNCIAS BIBLIOGRÁFICAS

- Aguirre, L.A et al. **Development of soft sensors for permanente downhole gauges in deepwater oil wells**. Control Engineering Practice, vol 65, p. 83-89. 2017.
- Aslmeyer, F. **Automatic adjustment of data compression in process information management systems**. 16<sup>th</sup> European Symposium on Computer Aided Process Engineering and 9<sup>th</sup> Internacional Symposium on Process System Engineering, p 1533 – 1538. 2006.
- Âström, K.J.; McAvoy, T. J. **Intelligent control**. Full papers, vol 2, n° 3, p. 115-127, 1992.
- Baughman, D. R.; Liu, Y. A. **Neural Networks in Bioprocessing and Chemical Engineering**. Academic Press, INC. 1995.
- Carvalho, F.B.; Torres, B.S.; Fonseca, M. O.; Seixas, C. F. **Sistemas PIMS – conceituação, usos e benefícios**. Tecnologia em Metalurgia e Materiais, vol 4, abril-jun 2005, p 1-5.
- Chiang, L.; Lu, B.; Castillo, I. **Big Data Analytics in Chemical Engineering**. Annual review of chemical and biomolecular engineering, vol 8, p. 63-85, 2017.
- Colombo, A.W. et al. **Industrial cloud-based cyber-physical systems**. Suíça: Springer, 2014.
- Fortuna, L. et al. **Soft sensors for monitoring and control of industrial processes**. London: Springer, 2007.
- Ghergherehchi, R. **Benefits of using stream insight and fuzzy logic with PI system for industrial alarm system**. Dissertação de mestrado em ciência da computação. University of Stavanger, 2011.
- Gutierrez, R. M. V.; Pan, S. S. K. **Complexo Eletrônico: Automação do Controle**. BNDES Setorial, Rio de Janeiro, v. 28, p. 189-232, setembro, 2008.
- Haykin, S. **Redes Neurais: princípios e práticas**. 2<sup>a</sup> edição. Porto Alegre: Bookman, 2001.
- Himmelblau, D.M. **Applications of Artificial Neural Networks in Chemical Engineering**. Korean J. Chem. Eng, vol 17, p. 373-392, 2000.

IEC. **Enterprise-control system integration — part 3: activity models of manufacturing operations management**. 2007.

James, T. **Smart Factories**. Engineering & Tecnology, 2012.

Kadlec, P.; Gabrys, B.; Strandt, S. **Data-driven soft sensors in the process industry**. Computer and Chemical Engineering, v. 33, p. 795-814, 2009.

Kletti, J. **Manufacturing Execution Systems – MES**. Berlin: Springer, 2007.

Lamb, F. **Automação industrial na prática**. Porto Alegre: Bookman, 2015.

Landim, R. de O. **Sobre a transportabilidade de modelos preditivos baseados em redes neurais: aplicações a uma indústria química**. Dissertação de Mestrado, Programa de Pós-Graduação em Tecnologia de Processos Químicos e Bioquímicos, Escola de Química, UFRJ, 2016.

Li, Defang. **Perspective for smart factory in petrochemical industry**. Computers and Chemical Engineering, vol 91, p. 136-148, 2016.

Li, Y.G.; Gui, W., H.; Yang, C. H.; Xie, Y. F. **Soft sensor and expert control for blending and digestion process in alumina metallurgical industry**. Journal of Process Control, 23, 1012 – 1021. 2013.

Neumann, G. et al. **Development of a virtual analyzer for real-time monitoring of a polymer slurry process using gradient boosting algorithms**. I congresso brasileiro em engenharia de sistemas de processo (PSE-BR). 2019.

Oliveira, C.A.; Dummer, F.S.; Salomão, J.M.; Kulitz, H.R. **Implantação de um sensor virtual no sistema PIMS baseado em redes neurais artificiais para fins de diagnóstico de processo**. 18º Seminário de Automação e TI industrial, São Paulo. 2014.

OSIsoft. **Live Library**. 2019. Disponível em: < <https://livelibrary.osisoft.com>>. Acesso em: 1 set. 2019.

OSIsoft. **PI Data Archive 2018. Guia de Gerenciamento de Sistema**. 2018.



Parente, A. P. **Ferramentas de automação e machine learning para operação segura e ecoeficiente de processos de combustão**. Tese de Doutorado, Programa de Pós-Graduação em Tecnologia de Processos Químicos e Bioquímicos, Escola de Química, UFRJ, 2019.

Parente, A. P., Valdman, A., Folly, R. O. M., Souza Jr, M. B., & Nascimento, I. C. S. **Automação de Processos Industriais: do Pneumático à Indústria 4.0**. Revista Processos Químicos, 12(24), 101-108, 2018.

Popov, N. S.; Tepić, Z.; Stanišić, D.; Lazarević, S. **Implementation of a neural network based soft sensor**. INFOTEH-JAHORINA vol 13, março, 2014.

Prieto, A.; Prieto, B.; Ortigosa, E.M.; Ros, E.; Pelayo, F.; Ortega, J.; Rojas, I. **Neural networks: An overview of early research, current frameworks and new challenges**. Neurocomputing, vol 214, p. 242-268, 2016.

Qin, S. J. **Process Data Analytics in the Era of Big Data**. AIChE Journal, vol 60, nº 9, p. 3092 – 3100, 2014.

Rezende, E. F. A.; Teixeira, A. F.; Mendes, E. M. A. M. **BR-Sensor: na on-line data-driven soft sensor of downhole pressure**. IFAC, vol 48, p 311-316. 2015.

Salvatore, L. **Inferência do teor de enxofre em unidades de hidrotreatamento de diesel baseada em modelagem heurística**. Dissertação de Mestrado, Programa de Pós-Graduação em Tecnologia de Processos Químicos e Bioquímicos, Escola de Química, UFRJ, 2007.

Shu, Y.; Ming, L.; Cheng, F.; Zhang, Z.; Zhao, J. **Abnormal situation management: Challenges and opportunities in the big data era**. Computers and Chemical Engineering, vol 91, p. 104-113, 2016.

Silveira, R.P.; Trierweiler, J.O.; Farenzena, M.; Teixeira, H.C. **Systematic Approaches for PI System Data Compression Tuning**. 8<sup>th</sup> IFAC Symposium on Advanced Control of Chemical Processes, p. 309 – 313. Furama Riverfront, Singapore, jun 10-13, 2012.

Souza, A. J. **Sistema de gerência de informação de processos industriais via WEB**. Dissertação, Programa de Pós-Graduação em Engenharia Elétrica, Centro de Tecnologia, UFRN, 2005.

Souza, A. J.; Feijó, R. H., Leitão, G. B. P.; Medeiros, A. A. D.; Bezerra, C. G.; de Andrade, W. L. S.; Guedes, L. A., e Maitelli, A. L. **Gerência de informação de processos industriais: um estudo de caso na produção de petróleo e gás**. VII SBAI / II IEEE LARS, São Luís. 2005.

Thornhill, N.F.; Choudhury, S.M.A.A.; Shah, S.L. **The impact of compression on data-driven process analyses**. Journal of Process Control, vol 14, p 389-398. 2014

Valdman, A. **Sistema de automação para monitoramento on-line de gases residuais e diagnóstico de uma caldeira operada com misturas diesel/biodiesel**. Dissertação de Mestrado, Programa de Pós-Graduação em Tecnologia de Processos Químicos e petroquímicos, Escola de Química, UFRJ, 2010.

Valdman, A.; Tiradentes, R.P. **Sala de aula 4.0: treinando um engenheiro de processos em um ambiente multidisciplinar**. The Journal of Engineering and Exact Sciences, vol. 05 N. 05, 2019.

Venkatasubramanian, V. **Drowning in data: Informatics and Modeling Challenges in a Data-Rich Networked World**. AIChE Journal, vol 55, nº 1, p. 2-8, 2009.

Zhou, P.; Chai, T.; Sun, J. **Intelligence-based supervisory control for optimal operation of a DCS-controlled grindind system**. IEEE Trans. on Control System Technology, 21 (1), 162-175. 2013.

## APÊNDICE A1

Tabela A-1 Pesos e biases da rede neuronal MLP 10-12-1 para predição do teor de amônia do fundo da coluna.

Pesos dos neurônios entre a camada de entrada e a camada escondida												
Variável	1° Neurônio	2° Neurônio	3° Neurônio	4° Neurônio	5° Neurônio	6° Neurônio	7° Neurônio	8° Neurônio	9° Neurônio	10° Neurônio	11° Neurônio	12° Neurônio
Carga	-1,40121	1,45940	2,50964	-0,59222	-2,41352	-0,46445	0,76770	-0,01849	0,28757	-2,19953	1,80404	-0,08056
Razão carga vapor	-0,13691	1,96583	-0,07053	1,49174	-1,99838	0,06717	0,82086	-0,04445	0,12681	0,86720	-0,22070	-0,22184
Arraste	1,06713	-0,18390	-1,37325	0,64951	-1,49647	0,26905	-0,26986	0,71068	-0,31166	1,42902	0,62609	-0,11224
Soda	0,31300	1,90994	0,79085	-0,71957	-1,74751	0,14846	-0,01145	-0,10811	0,09103	-1,47905	0,49251	-0,03264
Refluxo	-0,76425	0,83154	-0,49610	-0,46430	-1,27342	0,44882	-0,20664	-0,03311	0,06108	-0,56071	1,79226	0,13279
T carga	1,36334	-1,78166	-0,00373	-0,66338	-1,22766	0,42324	0,63872	0,22929	-0,22571	5,48920	0,00403	-0,01562
pH carga	-0,56572	-2,17227	-0,81313	0,69073	2,54818	0,38159	-1,05941	0,20862	0,21011	0,43461	-1,40674	0,06184
T flash	1,15506	-0,81558	1,27184	1,62554	0,15610	-0,08317	1,59255	0,29223	-0,07463	1,86087	0,69156	0,23526
P flash	-0,70798	-0,49479	-0,85261	1,35753	2,12572	0,05224	-1,42081	-0,48417	0,43749	-2,66727	1,03811	0,08389
DP recheio sup	-1,10877	0,45626	0,59832	3,04906	0,87497	0,16282	-0,81900	-0,10790	0,27887	1,94076	1,59523	-0,06265
Bias de entrada	-0,44930	-0,93743	-0,39202	-2,26343	-0,02657	0,77295	-0,36408	0,33579	0,11474	-0,71829	-1,03524	-0,11921

Pesos dos neurônios entre a camada escondida e a camada de saída												
1° Neurônio	2° Neurônio	3° Neurônio	4° Neurônio	5° Neurônio	6° Neurônio	7° Neurônio	8° Neurônio	9° Neurônio	10° Neurônio	11° Neurônio	12° Neurônio	
1,08696	-0,37071	0,69620	1,81603	-2,32245	0,17668	0,65303	0,36560	-0,14648	-1,15276	-2,83058	-0,40696	
Bias de entrada	-0,70318											

Fonte: Landim, 2016.