

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E DE
COMPUTAÇÃO

Israel Jacob Galego Cassimiro

Identificação de Falhas em Máquinas Elétricas Rotativas Usando Sistemas Inteligentes

São Carlos
2013

Israel Jacob Galego Cassimiro

Identificação de Falhas em Máquinas Elétricas Rotativas Usando Sistemas Inteligentes

Trabalho de Conclusão de Curso apresentado
à Escola de Engenharia de São Carlos, da
Universidade de São Paulo

Curso de Engenharia Elétrica com ênfase
em Sistemas de Energia e Automação

ORIENTADOR: Prof. Dr. Ivan Nunes da Silva

São Carlos
2013

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

C345i Cassimiro, Israel Jacob Galego
Identificação de falhas em máquinas elétricas
rotativas usando sistemas inteligentes / Israel Jacob
Galego Cassimiro; orientador Ivan Nunes da Silva. São
Carlos, 2013.

Monografia (Graduação em Engenharia Elétrica com
ênfase em Sistemas de Energia e Automação) -- Escola de
Engenharia de São Carlos da Universidade de São Paulo,
2013.

1. Motor de indução trifásico. 2. Sistemas
inteligentes. 3. Redes neurais artificiais. 4.
Classificador de falhas. 5. Transformada discreta
wavelet. 6. Variáveis estatísticas. I. Título.

FOLHA DE APROVAÇÃO

Nome: Israel Jacob Galego Cassimiro

Título: "Identificação de falhas em máquinas elétricas rotativas usando sistemas inteligentes"

Trabalho de Conclusão de Curso defendido e aprovado
em 18 / 11 / 2013,

com NOTA 100 (DEZ, ZERO), pela Comissão Julgadora:

Prof. Associado Ivan Nunes da Silva - (Orientador - SEL/EESC/USP)

Prof. Dr. Danilo Hernane Spatti - SEL/EESC/USP

Prof. Dr. Marcelo Suetake - (UFSCar)

Coordenador da CoC-Engenharia Elétrica - EESC/USP:
Prof. Associado Homero Schiabel

*When prayers turn to promises,
not even fate can stand in their way.*

Oerba Yun Fang – Final Fantasy XIII

Agradecimentos

A conclusão deste Trabalho de Conclusão de Curso, bem como toda a graduação teve a colaboração de diversas pessoas que se envolveram de forma direta ou indiretamente.

Em primeiro lugar gostaria de agradecer à minha família que foi a base do que sou hoje. Ao meu pai **José Jacob Cassimiro**, que, da forma mais simples possível, mostrou que podemos conseguir tudo que queremos com muito trabalho e dedicação. À minha mãe **Nilce Aparecida Galego Cassimiro**, que sempre me incentivou a estudar e crescer na vida, sempre esteve ao meu lado quando eu precisei, e pelas longas noites acordadas ao leito de minha cama quando estava enfermo. À minha irmã **Isabela Galego Cassimiro**, na qual sempre me inspirei, e que foi um exemplo pra mim desde cedo em relação aos estudos. E por fim ao meu irmão **Inácio José Galego Cassimiro**, pelo grande companheirismo e por ser o segundo jogador sempre que eu precisei, mesmo que isso significasse minha derrota no jogo.

Ao meu orientador **Prof. Dr. Ivan Nunes da Silva**, que, com sua paciência e dedicação, contribuiu tanto para a realização deste trabalho quanto para a minha formação ética e profissional.

Ao **Prof. Dr. José Carlos de Melo Viera Júnior** que sempre lecionou para a minha turma com muita dedicação desde o início da graduação, contribuindo para minha formação.

Ao grande amigo, mais que amigo, ao irmão **Eleandro Marcondes de Almeida**, por sua amizade sem igual, pelas longas jornadas de estudo durante e aos finais de semana e por ter me aturado por todos estes anos da graduação.

Ao **Artur Lustre Gonçalves**, por todas as vezes que morreu por conta da *witch* que eu ativei, pelas jogatinas em grupo, pelo carisma e pela grande amizade.

Ao **Gabriel Rogatto**, que, como veterano, sempre deu boas dicas tanto para a graduação quanto para a vida, pelas duradouras risadas e conversas fiadas, e por ter sido, acima de tudo, um grande amigo.

À **Lara Bertini Vieira**, por todas as situações cômicas que passamos juntos devido ao que ela falava sem pensar (T), pelos cafés às quintas-feiras, pelos *cappuccinos* do IFSC e por sempre estar disposta a ajudar quando eu precisei.

Ao **Luís Fernando Navarro**, pela sorveteira, por estatística (kkk), pelos bons momentos, por ser o eterno bixão que pega o suco e pela amizade.

Ao **Lucas Rodrigues Borges**, por ser um dos poucos que sempre dizia que eu estava errado, me deixando furioso, pelos eventos de anime, pelo Mario Kart DS na aulas (isso era segredo!!), e pela amizade.

Ao **Murilo Eduardo Casteroba Bento**, o qual eu conheci no meu primeiro dia na universidade e que seguiu comigo até o final, sempre trabalhando juntos, rindo juntos, e por ter me aturado nas viagens de volta pra casa.

Ao **Osmar Carlos Pinheiro Junior**, pelos bons momentos, pelas tardes de video-game, pelas risadas no bandejão, pelas pizzarias, pelas recomendações de animes, pelas reclamações no LoL, e pela grande amizade. Arigatou!

E por fim ao **Rafael Atique Claudio**, pelas conversas, tanto presenciais quanto pelo facebook, pela zoação, pelos momentos, por sempre aturar minhas reclamações, pela grande pessoa que é e pela grande amizade que eu pude compartilhar.

Resumo

CASSIMIRO, I. J. G. (2013). “**Identificação de Falhas em Máquinas Elétricas Rotativas Usando Sistemas Inteligentes**”. Trabalho de Conclusão de Curso - Escola de Engenharia de São Carlos, Universidade de São Paulo, 2013.

Atualmente, a maioria das indústrias utilizam motores em seus processos de fabricação de produtos. Sendo assim, é plausível que sejam feitas manutenções contínuas nestes motores a fim de evitar uma possível quebra, interrompendo toda a linha de produção e gerando grandes prejuízos. Dessa forma, busca-se em pesquisas uma melhor forma de tentar prever que um motor está com uma falha, antes que este possa ficar seriamente danificado. A literatura aponta muitas maneiras de se encontrar possíveis problemas, e uma delas, que vem ganhando bastante destaque, é a utilização de sistemas inteligentes para a classificação destas falhas. Sendo assim, este trabalho tem como objetivo implementar um classificador de falhas de motores de indução trifásicos utilizando sistemas inteligentes, mais especificamente redes neurais artificiais.

Palavras-chave: Motor de Indução Trifásico, Sistemas Inteligentes, Redes Neurais Artificiais, Classificador de Falhas, Transformada Discreta *Wavelet*, Variáveis Estatísticas.

Abstract

CASSIMIRO, I. J. G. (2013). “**Identifying Faults in Rotating Electrical Machines Using Intelligent Systems**”. Trabalho de Conclusão de Curso - Escola de Engenharia de São Carlos, Universidade de São Paulo, 2013.

Nowadays, most industries use motors in their manufacturing processes of products. Thus, it is desirable that maintenance on these motors be made to prevent a possible breakage, stopping all the production lines and generating large losses. Therefore, research is made in order to find a better way to predict (forecast) if a motor is in malfunction (is not working well) before it can be seriously damaged. The literature shows many ways to find potential problems, and one of them, which is receiving a lot of attention is the use of intelligent systems for the classification of these failures. Thus, this study aims to implement a fault classifier for three phase induction motors using intelligent systems, more specifically artificial neural networks.

Keywords: Three Phase Induction Motor, Intelligent Systems, Neural Networks, Fault Classifier, Discrete Wavelet Transform, Statistical Variables.

Lista de Figuras

1.1	Classificação das Falhas em Motores de Indução Trifásicos	24
2.1	Banco de filtros da <i>wavelet</i>	28
2.2	Banco de filtros de 2 aplicações da <i>wavelet</i>	29
2.3	Sinal de 60 Hz somado a outro sinal de frequência 8 Hz	30
2.4	Sinal de 60 Hz somado a outro sinal de frequência 2 Hz	30
2.5	Coeficientes de Aproximação de ambos os sinais (Filtro Passa-baixa)	31
2.6	Coeficientes de Detalhe de ambos os sinais (Filtro Passa-alta)	31
2.7	Exemplos do coeficiente de <i>skewness</i>	33
2.8	Neurônio Artificial	36
2.9	Funções de ativação de um neurônio	38
2.10	Estrutura da PMC	39
2.11	Estrutura da PMC pronta para ser equacionada	40
3.1	Etapas do Sistema de Classificação de Falhas	44
3.2	Sinais das Correntes Trifásicas do MIT normal acionado pela rede elétrica	46
3.3	Sinais das Correntes Trifásicas do MIT normal acionado pelo inversor trifásico	46
3.4	Sinais das Correntes Trifásicas do MIT acionado pela rede elétrica com 1 barra danificada	47
3.5	Sinais das Correntes Trifásicas do MIT acionado pelo inversor trifásico com 1 barra danificada	47
3.6	Coeficiente de aproximação da primeira aplicação da TW	48
3.7	Coeficiente de aproximação da segunda aplicação da TW	48
3.8	Coeficiente de aproximação da primeira aplicação da TW	49
3.9	Coeficiente de aproximação da segunda aplicação da TW	49
3.10	Delimitação das amostras	50
3.11	Princípio dos segmentos proporcionais	51
3.12	Rede PMC utilizada	52
3.13	Relação entre erro quadrático médio e número de neurônios da camada escondida	53
3.14	Relação entre erro quadrático médio e número de neurônios da camada escondida	53
3.15	Relação entre erro médio quadrático e número de neurônios da primeira camada escondida	54
4.1	Porcentagem de acertos para o primeiro conjunto de testes	56
4.2	Porcentagem de acertos para o segundo conjunto de dados	56
4.3	Média da Corrente I_a para todos os casos de testes	58
4.4	Mediana da Corrente I_a para todos os casos de testes	58

4.5	Porcentagem de acertos	59
-----	----------------------------------	----

Lista de Tabelas

3.1	Dados adquiridos	45
3.2	Padrão de Saídas Desejadas da MLP	51
3.3	Topologias Finais para o Classificador de Falhas	54
4.1	Matriz de validação resultante para falhas no rotor acionado pela rede elétrica - Conjunto de dados 1	55
4.2	Tabela de Porcentagem de Acertos	57
4.3	Matriz de validação resultante para falhas no rotor acionado pela rede elétrica - Conjunto de dados 2	57
4.4	Tabela de Porcentagem de Acertos	59
4.5	Matriz de Validação Resultante para Falhas no Motor Acionado pelo Inversor Trifásico	59

Lista de Siglas

KT	Coeficiente de <i>Kurtosis</i>
MIT	Motor de Indução Trifásico
MLP	<i>MultiLayer Perceptron</i>
PMC	Perceptron Multicamadas
RMS	<i>Root Mean Square</i>
RNA	Rede Neural Artificial
SK	Coeficiente de <i>Skewness</i>
TDF	Tranformada Discreta de Fourier
TDW	Tranformada Discreta Wavelet
TF	Tranformada de Fourier
TW	Transformada Wavelet

Sumário

1	Introdução	23
1.1	Motivação e Relevância do Trabalho	23
1.2	Objetivos e Justificativas	24
1.3	Organização do Trabalho	24
2	Metodologias Empregadas	27
2.1	Introdução	27
2.2	Transformada Discreta Wavelet	27
2.3	Variáveis Estatísticas	32
2.4	Redes Neurais Artificiais	34
2.4.1	Principais Características	34
2.4.2	Potenciais Aplicações das RNAs	35
2.4.3	O Neurônio Artificial	35
2.4.4	Funções de Ativação Neural	37
2.4.5	Redes <i>Perceptron</i> Multicamadas	38
2.4.6	Processamento da PMC	39
2.4.7	Treinamento da PMC	40
2.4.8	Projetando uma rede PMC	40
3	Sistema de Identificação de Barras Danificadas	43
3.1	Introdução	43
3.2	Construção da Base de Dados	44
3.3	Aplicação da Transformada Wavelet	47
3.4	Delimitação das Amostras	50
3.5	Aplicação das Variáveis Estatísticas e Normalização dos Dados	50
3.6	Separação dos Conjuntos de Teste e Treinamento	51
3.6.1	MIT acionado pela rede elétrica	52
3.6.2	MIT acionado pelo inversor trifásico	53
3.6.3	Topologias Finais	54
4	Validação das RNA's	55
4.1	Introdução	55
4.2	Validação para a Partida Direta	55
4.3	Validação para a Partida Utilizando Inversor Trifásico	58
5	Conclusão	61

Capítulo 1

Introdução

1.1 Motivação e Relevância do Trabalho

Atualmente, as máquinas elétricas são utilizadas para importantes trabalhos tanto na geração de energia quanto nas indústrias. Máquinas trifásicas se dividem em duas grandes classes: Máquinas Síncronas e Máquinas Assíncronas. As Máquinas Síncronas são mais utilizadas para a geração de energia elétrica, a tensão gerada está em sincronia com o campo, 60 Hz no caso do Brasil. Já os Máquinas Assíncronas, ou os Motores de Indução Trifásicos, são utilizados para os mais diversos fins na indústria brasileira. Em 2005, a Empresa de Pesquisa Energética (EPE) levantou que 47% da energia produzida no Brasil é destinada à classe industrial, sendo deste montante 50% destinado aos motores, o que os tornam bastantes importantes para as indústrias [1]. Com isso, estes motores recebem atenção especial das indústrias, pois uma falha em um motor de grande porte pode parar uma linha de produção inteira, gerando muitos prejuízos.

Entre os tipos de motores de indução, o Motor de Indução com Rotor em Gaiola de Esquilo merece um destaque maior, pois apresenta alta robustez, longa vida útil e quase não precisa de manutenções [1, 2]. Neste trabalho motores deste tipo serão estudados, já que estão mais presentes nas indústrias.

Como dito anteriormente, quando acontece uma falha de um motor que não estava prevista, linhas de produção inteiras cessam e trazem enormes prejuízos para as empresas. É com este intuito que estudos são desenvolvidos a fim de analisar o motor e avisar se uma possível falha, para que este possa sofrer uma manutenção antes de causar problemas mais severos.

Conforme visto em [3], as falhas de MIT's mais comuns podem ser classificadas conforme mostra a Figura 1.1.

Neste trabalho focou-se nas falhas geradas por barras quebradas do rotor. Estas falhas ocorrem quando há um aumento na oscilação da carga, extrema vibração ou pobre dinâmica de partida [3].

Sendo assim, este trabalho tem como objetivo criar um classificador de falhas para que seja possível verificar se o motor está ou não em bom funcionamento, evitando paradas bruscas.

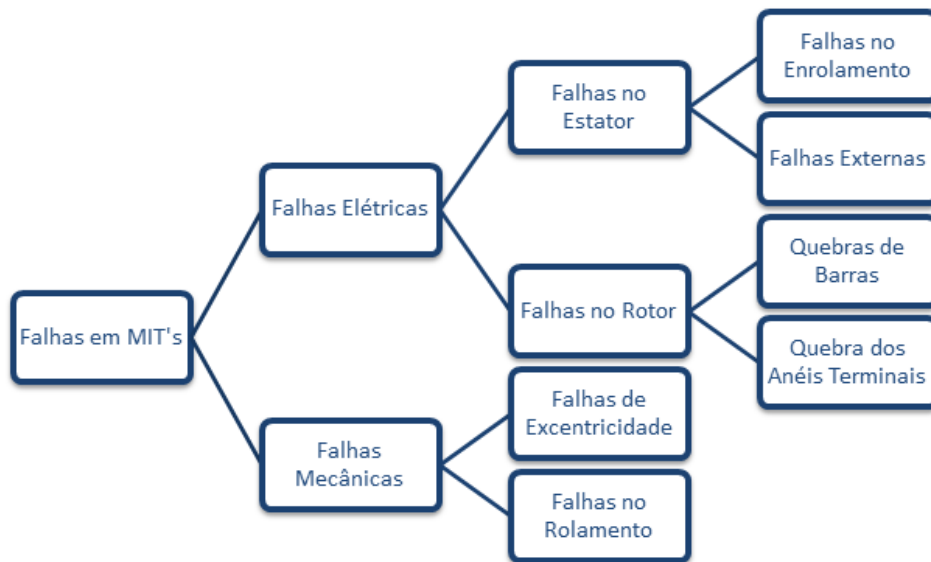


Figura 1.1: Classificação das Falhas em Motores de Indução Trifásicos

1.2 Objetivos e Justificativas

O objetivo deste trabalho é verificar a validade da utilização de sistemas inteligentes para classificar falhas nos rotores de motores de indução trifásicos.

- Desenvolver um sistema computacional que seja responsável por analisar falhas em motores de indução trifásicos através das informações de suas correntes trifásicas;
- Investigar a Transformada Discreta *Wavelet* e mostrar que esta consegue exercer a etapa de pré-processamento do sinal das correntes elétricas;
- Utilizar variáveis estatísticas para extrair as características de cada sinal de corrente;
- Identificar a melhor topologia para a identificação de falhas utilizando uma rede neural artificial;
- Verificar se a metodologia proposta para a identificação de falhas é válida e pode ser usada.

1.3 Organização do Trabalho

Este Trabalho de Conclusão está organizado em cinco capítulos que demonstram os passos para a obtenção dos resultados finais.

O Capítulo 1 apresentou a relevância e a dificuldade de se analisar as falhas em motores de indução trifásicos (MIT). Mostrou também as principais falhas que ocorrem nestes tipos de motores, bem como a relevância e as justificativas para este trabalho.

No Capítulo 2 serão apresentadas e detalhadas as metodologias empregadas para montar um classificador de falhas de um MIT. Estas metodologias são: a Transformada Discreta *Wavelet* e as Variáveis Estatísticas, ambas para extrair as características dos sinais de corrente do MIT, e, por fim, as Redes Neurais Artificiais, mais precisamente a Perceptron MultiCamadas, que foi utilizada para classificar as falhas.

No Capítulo 3 serão apresentados os passos do classificador de falhas, desde a medição das correntes trifásicas do motor até a classificação final. Aqui será mostrado com detalhes como as metodologias foram utilizadas para extrair as características dos sinais de corrente possibilitando a classificação da falha.

No Capítulo 4 apresenta os resultados obtidos no treinamento e validação da Rede Neural, tanto para motores acionados diretamente da rede como motores acionados por um inversor trifásico.

E, por fim, o Capítulo 5 apresenta as conclusões gerais deste trabalho.

Capítulo 2

Metodologias Empregadas

2.1 Introdução

Para a montagem de um sistema classificador de problemas em um MIT é necessário utilizar algumas metodologias para extrair as informações dos sinais de corrente do MIT e, posteriormente, classificar a situação do motor em cima destas características.

As três metodologias utilizadas neste trabalho foram: A Transformada Discreta *Wavelet*, as Variáveis Estatísticas e o Perceptron Multicamadas. Cada uma destas metodologias será explicada mais detalhadamente nas seções seguintes.

2.2 Transformada Discreta Wavelet

Muitas vezes faz-se necessário a troca do domínio de um sinal/amostra para que a análise destes dados seja feita de forma satisfatória. É o que acontece com a Transformada de Fourier (TF), que utiliza funções bases como seno e cosseno para transformar um sinal do domínio do tempo para o domínio da frequência, permitindo, de forma mais clara, visualizar as frequências presentes naquele sinal. A TF necessita de um intervalo do sinal que se repete ao longo do tempo para que esta seja capaz de retirar as frequências presentes neste sinal, isto é, um sinal estacionário, o que, em alguns casos, pode ser uma característica ruim.

Como pode ser visto em [4, 5], a TF consegue extrair as características de um sinal, sendo possível classificar o problema que o MIT apresenta, porém, em todos estes casos, o sinal era estacionário. Em sistemas *online*, em que a corrente do estator é dinâmica e varia com o passar do tempo, a TF já não é a melhor opção [6].

O sinal de corrente dos MIT's podem apresentar pequenas mudanças com o passar do tempo, causado por ruídos externo ou possíveis falhas, sendo assim, é provável que não seja possível escolher um intervalo de tempo que se repete. É possível escolher uma parte do sinal que o tente representar como um todo, e com isso a TF irá dar bons resultados, mas pode ser que alguma frequência seja perdida neste processo.

É com este intuito que neste trabalho optou-se por estudar e aplicar a Transformada *Wavelet* (TW). A TW não possui um único conjunto de funções bases (seno e cosseno no caso da TF), mas sim um conjunto infinito de funções bases, todas derivadas da *wavelet* mãe $\psi(t)$, quando esta é dilatada ou transladada. Isto permite que a função *wavelet* translade um sinal no domínio do tempo não retirando apenas as características de frequência dele (como acontece com a FT), mas

também as características do tempo [7], isto é, a *wavelet* é capaz de verificar se as frequências do sinal alteraram com o passar do tempo (domínio tempo-frequência). Transladar aqui significa a movimentação da janela através do sinal.

Ambas as transformadas, TDF e TDW, apresentam grandes semelhanças do ponto de vista funcional. São invertíveis, as matrizes da transformada inversa são as transpostas das originais e ambas as transformadas são convoluções [8].

As operações de dilatação e translação da *wavelet* mõe permite decompor o sinal em diferentes frequências em suas escalas correspondentes permitindo estudar cada componente separadamente. Estas operações podem ser representadas por:

$$\psi\left(\frac{x-b}{a}\right), (a, b) \in R^+ \times R \quad (2.1)$$

em que $a = 2^{-j}$ e $b = k \cdot 2^{-j}$ sendo k e j inteiros [9].

Sendo assim, os parâmetros a e b são alterados a fim de selecionar todas as faixas do domínio frequência-tempo que sejam importantes para um estudo específico.

A Transformada Discreta *Wavelet* (TDW) difere um pouco da contínua, utilizando bancos de filtros na análise de multiresolução. O banco de filtros é responsável por separar o sinal em bandas de frequências, isto é, frequências acima de um limiar e frequências abaixo de um limiar. O sinal pode passar por diversos filtros passa-alta para a análise das altas frequências, e por diversos filtros passa-baixa para análise das baixas frequências. A Figura 2.1 apresenta um diagrama para o cálculo da TDW.

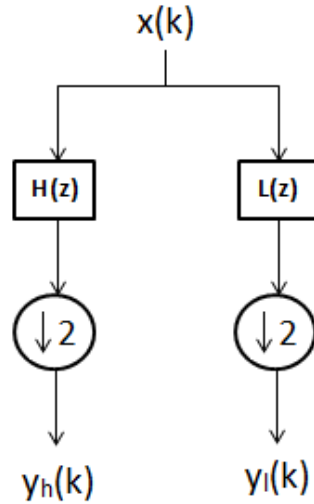


Figura 2.1: Banco de filtros da *wavelet*

Note pela Figura 2.1 que o sinal $x(k)$ passa por 2 filtros, um $H(z)$ representando um filtro passa-alta e um $L(z)$ representando o filtro passa baixa. Após a filtragem do sinal há outro operador, o *downsampling*. Este operador é útil para diminuir as amostras que os sinais filtrados apresentam, já que estes sinais estão com amostras a mais do que o necessário para sua correta representação.

A TDW pode ser aplicada diversas vezes nos sinais resultantes, conforme o nível de detalhamento necessário da aplicação em que esta é utilizada. A Figura 2.2 apresenta a TDW sendo

aplicada a mais de um nível.

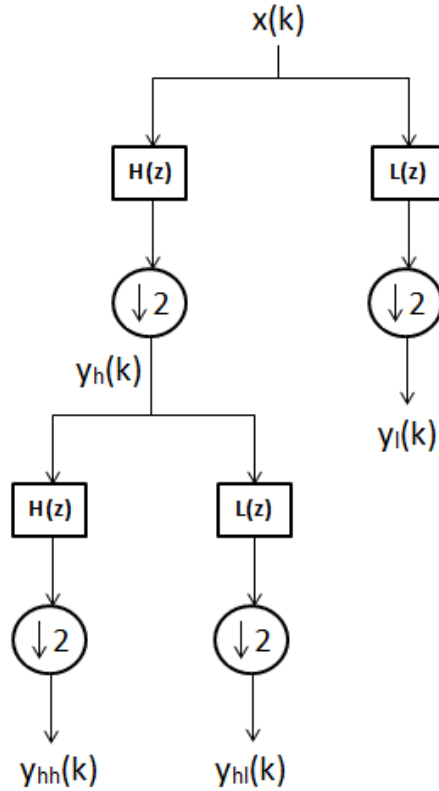


Figura 2.2: Banco de filtros de 2 aplicações da *wavelet*

Neste trabalho, será utilizada a TDW de Haar, em que [10] mostra como seu uso é simples, trazendo bons resultados para o trabalho.

Por meio da *Wavelet* de Haar, é possível decompor um sinal $x(k)$ em um nível j , em coeficientes que representam as baixas frequências ($c_n(k)$) e as altas frequências do sinal ($d_n(k)$). Esta decomposição acontece através da convolução de $x(k)$ com a transformada *wavelet* de Haar (W_t), podendo ser feita através de uma simples equação matricial, juntamente com o *downsampling*.

$$\begin{bmatrix} c_j(k) \\ d_j(k) \end{bmatrix} = W_T \cdot y_{signal} \quad (2.2)$$

sendo que

$$y_{signal} = \begin{bmatrix} x(1) & x(3) & x(5) & \dots & x(n-1) \\ x(2) & x(4) & x(6) & \dots & x(n) \end{bmatrix} \quad (2.3)$$

e

$$W_T = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \quad (2.4)$$

O *downsampling* ocorre quando se ajusta o sinal conforme mostra (2.3). Este ajuste, juntamente com a multiplicação matricial por W_T resulta na TW do sinal. A TW pode ser aplicada diversas vezes em sinal, conforme o nível de detalhamento necessário.

Os coeficientes representados por d_j são conhecidos como coeficientes de detalhe, e apresentam as informações das altas frequências do sinal. Já os coeficientes c_j são conhecidos como coeficientes de aproximação e trazem informações sobre as baixas frequências que o sinal apresenta.

A Figura 2.3 apresenta um sinal senoidal de 60 Hz somado a outro sinal senoidal de aproximadamente 8 Hz. Já a figura 2.4 apresenta o mesmo sinal senoidal de 60 Hz, porém somado a um sinal de, aproximadamente, 2 Hz.

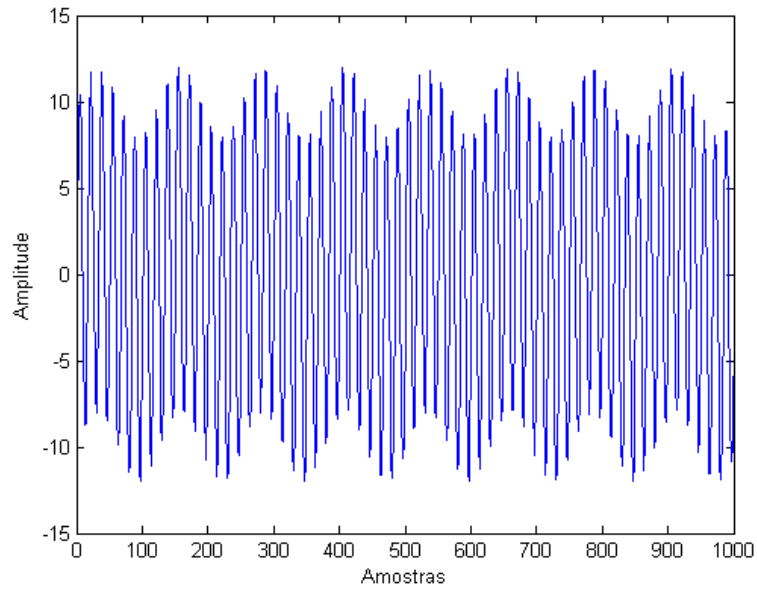


Figura 2.3: Sinal de 60 Hz somado a outro sinal de frequência 8 Hz

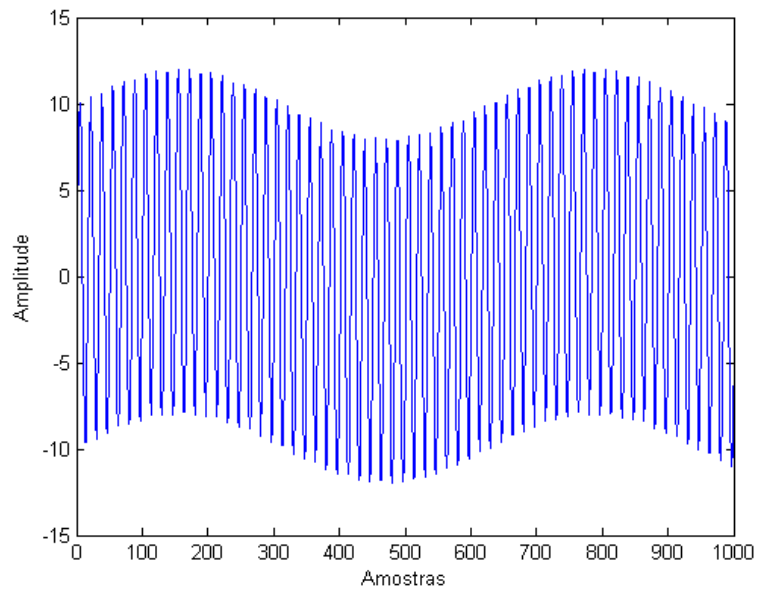


Figura 2.4: Sinal de 60 Hz somado a outro sinal de frequência 2 Hz

A TDW será aplicada uma vez em ambos os sinais a fim de verificar sua funcionalidade. A Figura 2.5 apresenta os coeficientes de aproximação da TWD e a Figura 2.6 os coeficientes de

detalhe.

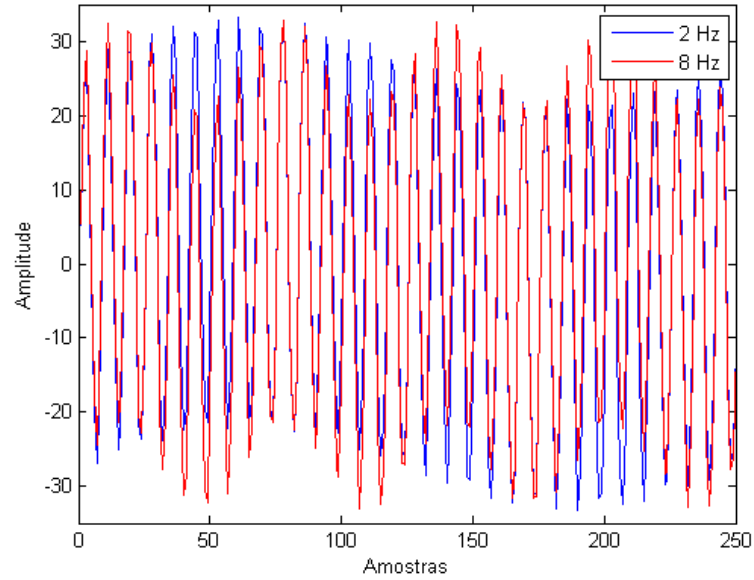


Figura 2.5: Coeficientes de Aproximação de ambos os sinais (Filtro Passa-baixa)

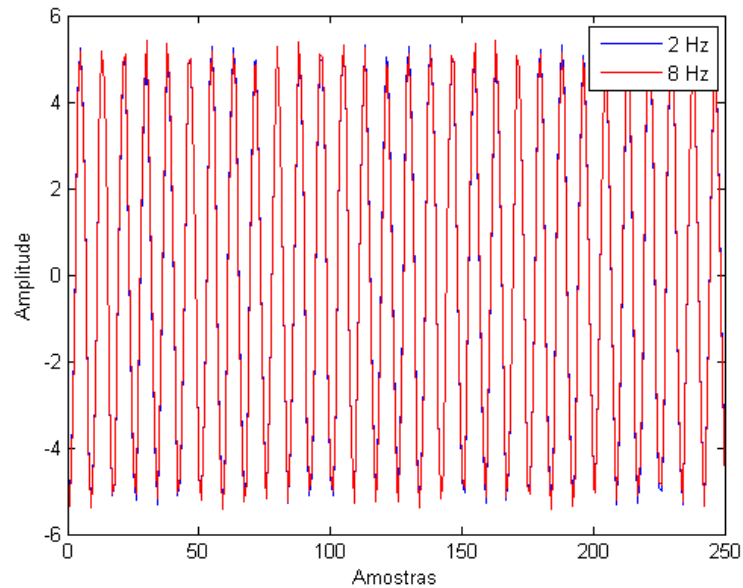


Figura 2.6: Coeficientes de Detalhe de ambos os sinais (Filtro Passa-alta)

Note que na Figura 2.5 os sinais da TDW são diferentes para as amostras que contêm 2 Hz e 8 Hz, isto porque o filtro passa-baixa diferencia estas frequências. Na Figura 2.6 praticamente não há diferenciação dos sinais, já que os sinais não apresentam alta frequência. O contrário pode ser observado somando-se altas frequências no sinal e analisando novamente os coeficientes.

Como o intuito é encontrar as baixas frequências que dominam o sinal através do tempo, fato causado por haver problemas no MIT, os coeficientes de aproximação serão utilizados daqui em diante, e o coeficiente de detalhe será descartado.

2.3 Variáveis Estatísticas

Conforme visto na seção anterior, após a TW ser aplicada nos sinais trifásicos de corrente, tem-se como resultado sinais ao longo do tempo. Sendo assim, é inviável ter este sinal como entrada para um PMC, já que se necessitariam de muitas entradas e a rede ficaria complexa demais. Para que a rede consiga obter êxito no seu treinamento e teste para a localização de falhas em um motor, é necessário que as características dos sinais analisados sejam extraídos da melhor forma possível, tornando único cada sinal analisado.

Já que este fator apresenta uma importância elevada para o correto treinamento da rede, e, conseqüentemente, o correto diagnóstico, buscou-se variáveis estatísticas que apresentam uma metodologia bastante eficiente, sendo capazes de generalizar as informações contidas em um dado sinal. As variáveis estatísticas, em geral, são utilizadas para generalizar um modelo em cima de uma amostra de dados, o que é exatamente o intuito aqui.

Dentre as diversas variáveis estatísticas existentes, foram escolhidas cinco para extraírem as características das correntes trifásicas do motor. Estas cinco variáveis estão dispostas abaixo:

- **Média:** é uma variável que apresenta um único valor que melhor representa o conjunto de dados como um todo. Esta variável pode ser calculada como:

$$\mu^i = \frac{1}{N} \sum_{j=1}^N x_j^i \quad (2.5)$$

em que N é o número total de amostras e x_j^i é a j -ésima amostra do conjunto i .

- **Mediana:** variável que apresenta um único valor capaz de separar a parte inferior das amostras da parte superior. Assim, metade dos elementos do conjunto serão inferiores ou iguais a mediana, e a outra metade serão iguais ou superiores a mediana. Ela indica o valor central do conjunto de dados.

Para amostras de tamanho n ímpar, a mediana é:

$$mediana = x \left(\frac{n+1}{2} \right) \quad (2.6)$$

Já para amostras de tamanho n par tem-se:

$$mediana = \frac{1}{2} \cdot \left[x \left(\frac{n}{2} \right) + x \left(\frac{n}{2} + 1 \right) \right] \quad (2.7)$$

- **Root Mean Square:** também conhecido como **valor quadrático médio** ou **valor eficaz**, a *rms* apresenta uma medida de magnitude da variável. Pode ser calculada utilizando a seguinte função:

$$rms = \sqrt{\frac{1}{N} \sum_{j=1}^N (x_j^i)^2} \quad (2.8)$$

- **Skewness:** variável que representa a forma da distribuição. Pode ser calculada como:

$$SK = \frac{1}{(N-1)s^3} \sum_{j=1}^N (x_j^i - \mu^i)^3 \quad (2.9)$$

em que s é o desvio padrão do conjunto de amostras.

- **Kurtosis:** outra variável que representa a forma da distribuição.

$$KT = \frac{1}{(N-1)s^4} \sum_{j=1}^N (x_j^i - \mu^i)^4 \quad (2.10)$$

Em [11], os coeficientes de *skewness* e *kurtosis* são chamados de terceiro e quarto momento do conjunto, relacionando a potência em que os dados são elevados, sendo *skewness* à terceira potência e *kurtosis* à quarta potência, como pode ser observado em (2.9) e (2.10).

O Coeficiente de *Skewness* (SK) [12], ou em português, **obliquidade**, indica o quão simétrico os valores do conjunto analisado estão em torno de sua média. Este pode ser separado em três situações:

- $SK < 0$: informa que existem mais amostras acima do valor da média no conjunto observado;
- $SK > 0$: informa que existem mais amostras abaixo do valor da média;
- $SK = 0$: Os dados do conjunto estão distribuídos simetricamente em torno da média.

As Figuras 2.7(a) e 2.7(b) apresentam casos em que o SK é maior e menor que zero, respectivamente. Note que nas figuras há um traço que separa os dados que tem valores acima da média e os dados com valores abaixo da média. Na Figura 2.7(a) há mais dados acima da média, o que resultou em um SK abaixo de zero ($SK = -0,5634$). Já na 2.7(b) há mais valores abaixo da média, resultando em um SK acima de zero ($SK = 3,2738$).

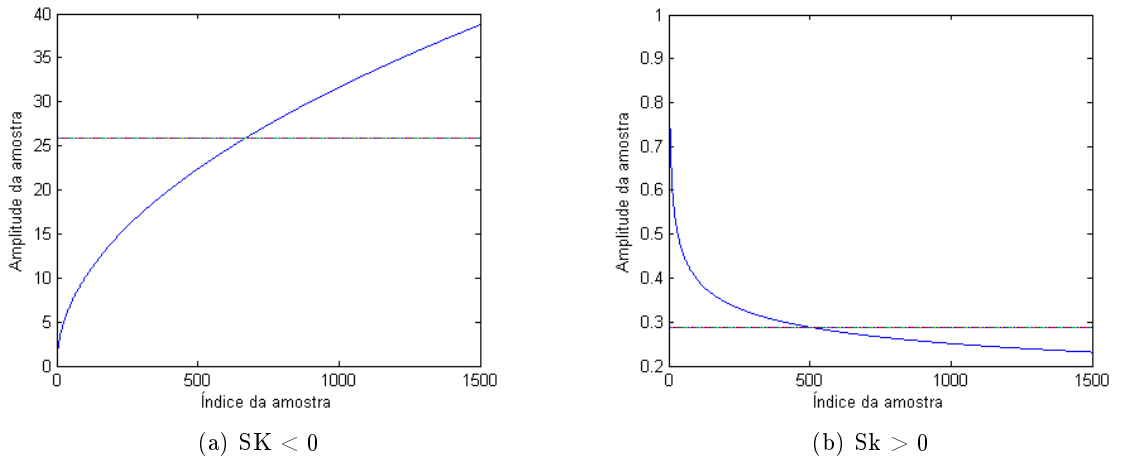


Figura 2.7: Exemplos do coeficiente de *skewness*

O Coeficiente de *Kurtosis* é utilizado para verificar se a maioria dos dados se encontra no pico da média. Para uma distribuição normal, por exemplo, o valor de KT é aproximadamente 3, e para uma distribuição exponencial, o KT vale 9.

Skewness e *Kurtosis*, que medem a variabilidade e a forma da distribuição, são comumente utilizados juntos para verificar o quão próxima o conjunto analisado é de uma distribuição normal. Em conjunto com a média, mediana e *rms*, é possível caracterizar o sinal de corrente das máquinas de uma forma única, como citado em [12]. Desta forma o conjunto de dados torna-se seguro.

2.4 Redes Neurais Artificiais

As RNAs (Redes Neurais Artificiais) são modelos computacionais que tentam simular a forma com que o cérebro humano processa informações e resolve problemas, isto é, entre erros e acertos. Os neurônios são representados por unidades de processamento associados a alguns pesos, chamados de pesos sinápticos, e a conexões que as ligam em outras unidades, chamadas de conexões sinápticas. O treinamento da rede se dá ajustando os pesos sinápticos de cada neurônio, repetidamente, até que esta forneça resultados em sua saída que sejam compatíveis com os dados de entrada [13].

2.4.1 Principais Características

Muitas vezes, modelos matemáticos que regem os sistemas não lineares e variantes no tempo são complexos ou quase impossíveis de serem determinados. Uma grande característica das RNA é, justamente, a habilidade de mapear estes tipos de sistemas com facilidade, alterando os pesos sinápticos dependendo do algoritmo de treinamento utilizado. Ela também é capaz de trabalhar com um enorme número de variáveis e conjunto de dados sem perder sua eficiência. Abaixo se encontram as principais características das RNAs [13]:

- **Capacidade de aprendizagem:** utilizando um conjunto de dados do processo para o treinamento da RNA, esta consegue ajustar seus pesos sinápticos de forma a se adaptar ao processo;
- **Capacidade de se adaptar:** sistemas variantes no tempo mudam constantemente. As RNAs são capazes de seguir estas alterações sem que haja nenhuma alteração estrutural interna (como número de neurônios), bastando apenas a apresentação de um novo conjunto de dados;
- **Capacidade de generalização:** utilizando o conjunto de amostras entregues para o treinamento da RNA, esta é capaz de efetuar um mapeamento e generalizar casos desconhecidos, isto é, dados de entrada que não foram utilizados para o treinamento da rede;
- **Agrupar ou organizar dados e informações:** são capazes de extrair informações importantes dos dados de entrada, reorganizando sua estrutura, a fim de organizá-los em classe separadas, isto é, formar padrões;
- **Tolerância à falhas:** mesmo com parte de sua estrutura comprometida (perda de um neurônio, por exemplo), a RNA ainda é capaz de operar informações, já que o processo de treinamento foi distribuído para inúmeros neurônios artificiais. Isto mostra o quanto as RNAs são robustas.

- **Facilidade de prototipagem:** após o treinamento da rede, sua utilização acontece por simples operações matemáticas elementares. Portanto, esta pode ser facilmente embarcada em um hardware dedicado.

2.4.2 Potenciais Aplicações das RNAs

Atualmente, muitas áreas estão utilizando RNAs para realizar processos. A seguir está listado algumas destas áreas de atuação das RNAs [13]:

- **Aproximador de Funções:** utilizando um conjunto de dados que apresenta o relacionamento de variáveis de entrada e saída, as RNAs são treinadas com o intuito de mapear estes relacionamentos e generalizar suas repostas. É comumente utilizado para mapear processos que apresentam modelos analíticos complexos, desconhecidos ou de difícil obtenção;
- **Controle de Processo:** basicamente consistem em identificar ações de controle que permitam o alcance dos requisitos de qualidade, de eficiência e de segurança de certo processo. São comumente utilizados em controles empregados em robótica, aeronaves, elevadores, eletrodomésticos, satélites, etc;
- **Classificação de Padrões:** esta aplicação consiste em associar cada amostra de entrada da rede à uma classe previamente predefinida. É geralmente utilizado em reconhecimento de imagens, voz, escrita, entre outros;
- **Agrupamento de Dados/*Clustering*:** O intuito é identificar e detectar similaridades e particularidades entre as diversas amostras de entrada do processo, objetivando-se o agrupamento das mesmas. São aplicados em identificação automática de classes, compressão e garimpagem de dados (*data mining*);
- **Sistemas de Previsão:** a rede consegue estimar dados futuros levando em consideração diversas medidas anteriormente observadas em seu domínio. Estes tipos de redes são utilizadas para prever mercados financeiros, demanda de energia, previsões climáticas, entre outros;
- **Otimização:** consiste em minimizar ou maximizar uma função objetivo, também conhecida como função custo, obedecendo restrições do problema. São utilizadas em problemas de otimização restrita, otimização combinatorial, programação dinâmica, problemas de sequenciamento de produção, etc;
- **Memórias Associativas:** consegue recuperar padrões corretos mesmo que este esteja incompleto ou distorcido. São utilizadas em processamento de imagens, transmissão de sinais, etc.

2.4.3 O Neurônio Artificial

Proposto em 1943 no trabalho de Warren McCulloch, psiquiatra e neuroanatomista, em conjunto com Walter Pitts, matemático, o primeiro modelo artificial de um neurônio nasceu [14]. Neste trabalho foi apresentado o modelo de neurônio como sendo uma unidade básica de processamento que recebe vários estímulos de entrada (dados) e, somando-os ponderadamente,

resulta em uma resposta na saída. A Figura 2.8, retirada de [14], apresenta o neurônio artificial proposto no trabalho de McCulloch e Pitts.

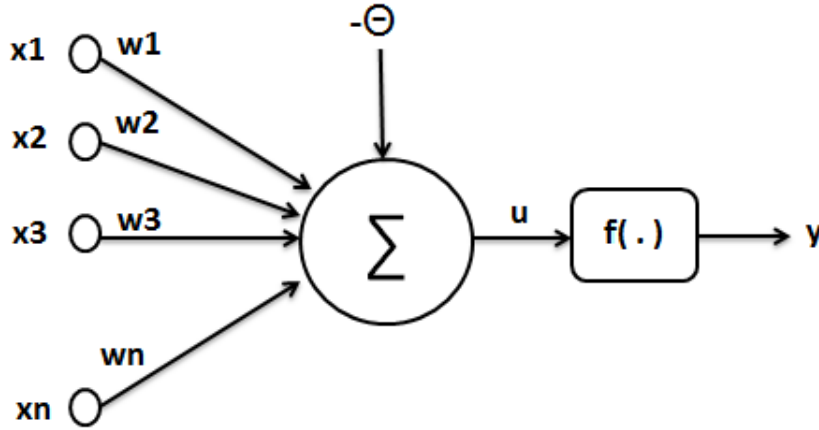


Figura 2.8: Neurônio Artificial

Na figura é possível observar as seguintes variáveis:

- **Sinais de entrada** $\{x_1, x_2, x_3, \dots, x_n\}$;
- **Pesos sinápticos** $\{w_1, w_2, w_3, \dots, w_n\}$;
- **Combinador linear** $\{\Sigma\}$;
- **Limiar de ativação** $\{\theta\}$;
- **Potencial de ativação** $\{u\}$;
- **Função de ativação** $\{f(\cdot)\}$;
- **Sinal de saída** $\{y\}$.

Os sinais $\{x_1, x_2, x_3, \dots, x_N\}$ são os estímulos que o neurônio recebe do meio externo, tais como medidas de sensores ou dados de comprimento. Já as variáveis $\{w_1, w_2, w_3, \dots, w_N\}$ são os pesos sinápticos, isto é, são os valores que representam as conexões sinápticas, e são utilizados para ponderar os dados de entradas, dando maior importância para algum dado em especial. O símbolo Σ representa a junção dos sinais de entrada ponderados, isto é, este faz uma combinação linear dos sinais ponderados. A variável θ representa o limiar de ativação para que o resultado do combinador linear possa disparar um estímulo para outro neurônio. O símbolo u é o valor da junção do combinador linear com o limiar de ativação, representando o potencial de ativação. Por fim, tem-se uma função $f(\cdot)$, conhecida como função de ativação, que tem como objetivo limitar o valor de saída dentro de um determinado intervalo ou efetuar o mapeamento linear [13].

Sendo assim, tem-se:

$$u = \sum_{i=1}^N w \cdot x_i - \theta \quad (2.11)$$

A saída do neurônio é calculada conforme abaixo:

$$y = f(u) \quad (2.12)$$

2.4.4 Funções de Ativação Neural

Conforme [13], as principais funções de ativação são as do tipo linear, degraus ou sinais, e sigmoidal. As funções degraus ou sinais são descontínuas e retornam valores iguais a 0 ou -1 para potenciais de ativação negativos ($u < 0$) ou 1 para potenciais de ativação nulos ou positivos ($u \geq 0$).

$$f_{sinddeg}(u) = \begin{cases} 0 & \text{ou} & -1, & \text{se } x < k \\ 1, & & & \text{se } x \geq k \end{cases} \quad (2.13)$$

Ambas as funções apresentadas anteriormente não são diferenciáveis ao longo de todo o seu domínio, e geralmente são empregadas como classificador de padrões após o processamento de uma RNA.

A função linear produz valores equivalentes ao potencial de ativação u , sendo definida por:

$$f_{linear}(u) = u \quad (2.14)$$

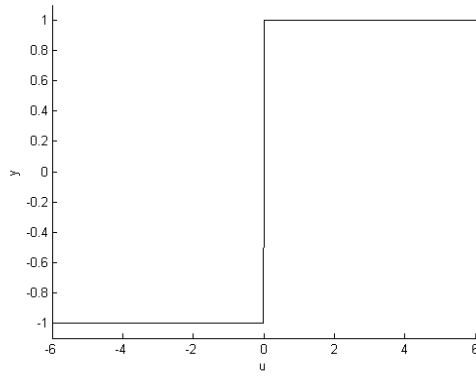
Já as funções sigmóides são diferenciáveis ao longo de todo o seu domínio e retornam valores entre -1 e 1 no caso da função tangente hiperbólica, ou entre 0 e 1, no caso da função logística. Estas funções são definidas como se segue:

$$f_{tanh}(u) = \frac{1 - e^{-\beta u}}{1 + e^{-\beta u}} \quad (2.15)$$

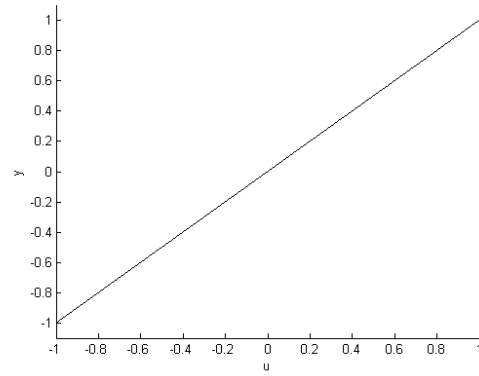
$$f_{logsig}(u) = \frac{1}{1 + e^{-\beta u}} \quad (2.16)$$

em que β é o parâmetro que indica a inclinação da função.

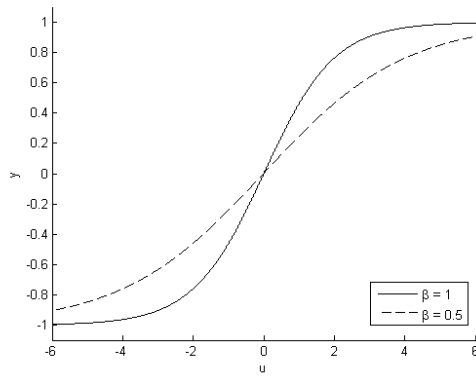
A Figura 2.9 apresenta os formatos das funções de ativação comentadas no texto acima.



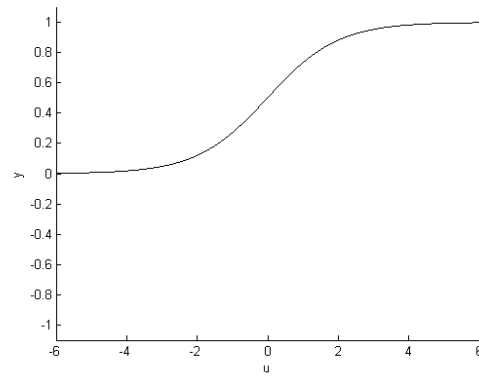
(a) Função sinal



(b) Função linear



(c) Função tangente hiperbólica



(d) Função logística

Figura 2.9: Funções de ativação de um neurônio

2.4.5 Redes *Perceptron* Multicamadas

As redes *Perceptron Multicamadas* (PMC) são constituídas de vários neurônios interconectados pelas conexões sinápticas. A Figura 2.10 ilustra uma rede PMC, mostrando que os neurônios são divididos em várias camadas, chamadas de camadas neurais [13].

As camadas neurais são divididas em três tipos:

- **Camada de Entrada:** esta camada é responsável por receber os sinais de entrada e repassá-los para todos os neurônios da primeira camada oculta. Sendo assim, os neurônios da camada de entrada não operam sobre os dados recebidos do meio exterior, apenas os repassam;
- **Camada Oculta ou Escondida:** estas camadas são responsáveis por processar os dados de entrada extraindo suas características.
- **Camada de Saída:** esta camada é também constituída de neurônios, sendo responsável por produzir/apresentar os resultados finais da rede.

Existem algumas técnicas utilizadas para encontrar a melhor quantidade de neurônios para cada camada escondida existente, assim como o número de camadas escondidas. Porém, estas técnicas são heurísticas, não permitindo a completa garantia de que a arquitetura escolhida é a melhor possível para o desempenho global.

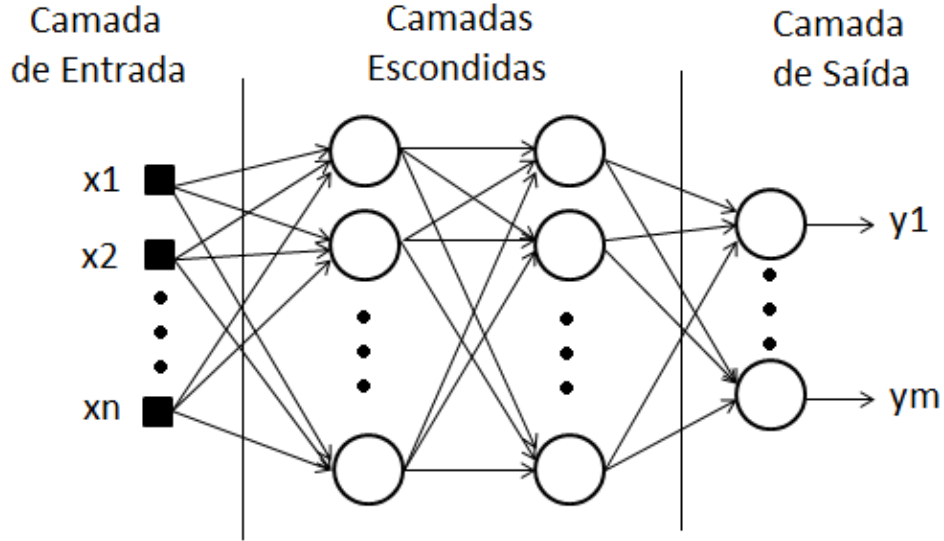


Figura 2.10: Estrutura da PMC

2.4.6 Processamento da PMC

Os sinais que chegam na primeira camada oculta são processados utilizando os pesos sinápticos associados a cada conexão entre estes neurônios e os neurônios da próxima camada oculta.

Os sinais que chegam na camada de entrada são processados utilizando os pesos sinápticos ($W_{ji}^{(l)}$) que ligam os neurônios da camada de entrada até a primeira camada oculta, é feita a combinação linear e a função de ativação é aplicada. Com isso, estes sinais estão prontos para passarem para a próxima camada oculta, seguindo o mesmo processo. Os sinais vão sendo transportados de camada a camada até a camada de saída da rede. Esta camada processa as informações obtidas da última camada oculta e fornece a resposta da rede referente aos estímulos da camada de entrada. A Figura 2.11 apresenta um PMC com variáveis definindo entradas, saídas e pesos sinápticos.

Observando a Figura 2.11, temos que:

- $W_{ji}^{(l)}$ é o peso sináptico do j -ésimo neurônio da camada l conectado ao i -ésimo neurônio da camada $(l - 1)$;
- $I_j^{(l)}$ é o potencial de ativação do j -ésimo neurônio da camada l ;
- $\theta_j^{(l)}$ é o limiar de ativação do j -ésimo neurônio da camada (l) .

$$I_j^{(1)} = \sum_{i=1}^N W_{ji}^{(1)} \cdot x_i - \theta_j^{(1)} \quad (2.17)$$

$$I_j^{(2)} = \sum_{i=1}^N W_{ji}^{(2)} \cdot Y_i^{(1)} - \theta_j^{(2)} \quad (2.18)$$

$$I_j^{(3)} = \sum_{i=1}^N W_{ji}^{(3)} \cdot Y_i^{(2)} - \theta_j^{(3)} \quad (2.19)$$

- $Y_j^{(l)}$ é a saída do j -ésimo neurônio da camada l :

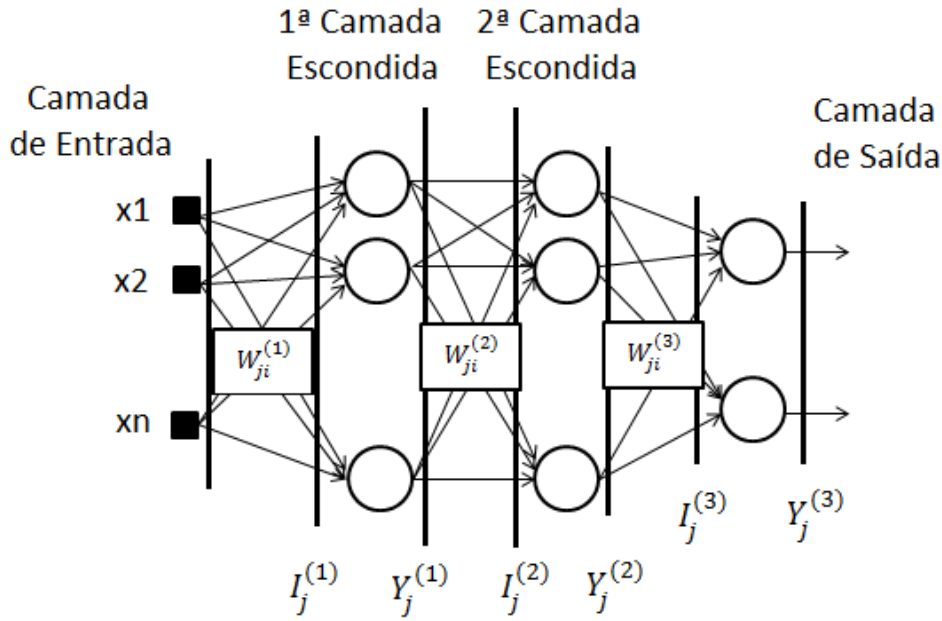


Figura 2.11: Estrutura da PMC pronta para ser equacionada

$$Y_j^{(1)} = f(I_j^{(1)}) \quad (2.20)$$

$$Y_j^{(2)} = f(I_j^{(2)}) \quad (2.21)$$

$$Y_j^{(3)} = f(I_j^{(3)}) \quad (2.22)$$

em que $f(\cdot)$ é uma função diferenciável.

2.4.7 Treinamento da PMC

No processo de aprendizagem de uma rede PMC existem duas fases: a fase *forward* e a fase *backward*. A primeira fase tem como objetivo levar as informações de entrada até a saída da rede, e a segunda fase faz exatamente o oposto, o sentido do fluxo de dados passa a ser da saída para a entrada.

Na primeira etapa, isto é, no passo *forward*, o intuito é propagar os dados de entrada através dos neurônios existentes na rede, cada um aplicando seus pesos. Ao final desta etapa, o algoritmo compara os resultados finais com a resposta esperada e calcula um erro. Caso este erro esteja fora dos limites impostos pelo treinamento, o algoritmo entra na segunda etapa, o *backward*. Esta etapa tem como objetivo retropropagar o erro camada por camada afim de ajustar os pesos sinápticos dos neurônios existentes nelas [13].

2.4.8 Projetando uma rede PMC

O projeto de uma rede PMC requer a manipulação de diversas variáveis. O número de camadas escondidas, o número de neurônios por camada e as funções de ativação de cada neurônio são variáveis importantes, que, se ajustadas corretamente, faz com que a rede responda de forma muito exata [15].

Quando uma rede já treinada, que apresenta erro baixo, recebe uma nova informação de treinamento que causa um alto erro, houve um problema de generalização, conhecido como *overfitting*. O *overfitting* acontece quando a rede, ao invés de generalizar as informações que recebe, simplesmente decora as informações de entrada e as relacionam com a saída. Assim a rede não foi ajustada corretamente e não consegue generalizar a resposta para casos desconhecidos. O número de neurônios está diretamente ligado a isso. Caso muitos neurônios sejam usados inadequadamente, é bem provável que um *overfitting* venha a ocorrer [13].

Capítulo 3

Sistema de Identificação de Barras Danificadas

3.1 Introdução

O objetivo deste trabalho é utilizar uma RNA para verificar quebra de barras do rotor de um MIT. Para isso serão utilizadas as correntes trifásicas do MIT, que após processadas, serão passadas como parâmetros para a rede neural afim de esta classificar se há barras quebradas (ou quantas barras quebradas existem) no MIT. Sendo assim, as correntes serão a fonte de dados do sistema.

Foram utilizados dois tipos de partida para o MIT, acionamento pela rede elétrica e acionamento por um inversor trifásico. Na partida direta, o MIT é alimentado com uma tensão de 220 V equilibrada com uma frequência de 60 Hz. Na partida utilizando um inversor trifásico, foram aplicadas tensões com formato de onda retangulares e moduladas por largura de pulso (PWM). Neste caso foram utilizadas frequências de acionamento entre 40 Hz e 60 Hz.

A montagem da bancada de testes e os dados são provenientes da Tese de Doutorado apresentada em [5]. As informações da montagem da bancada, bem como as ferramentas utilizadas para colher os dados amostrais, podem ser encontrados neste trabalho de conclusão de curso.

A Figura 3.1 apresenta as etapas que a corrente irá passar até chegar na entrada da MLP.

Em um primeiro momento, os sinais de corrente do MIT serão lidos. Estes sinais irão passar por um pré-processamento, que compreende a Transformada textitWavelet e as Variáveis estatísticas. Após estas etapas as amostras serão normalizadas e, por fim, separadas para um grupo de treinamento e um grupo de validação da RNA. Em seguida estes dados são entregues à MLP. Cada passo será exposto detalhadamente nas seções seguintes.

Os sinais de correntes trifásicas eram medidos após o MIT atingir seu regime permanente. Isto é importante, já que, no regime permanente, há a presença de frequências que aparecem por conta das barras quebradas.

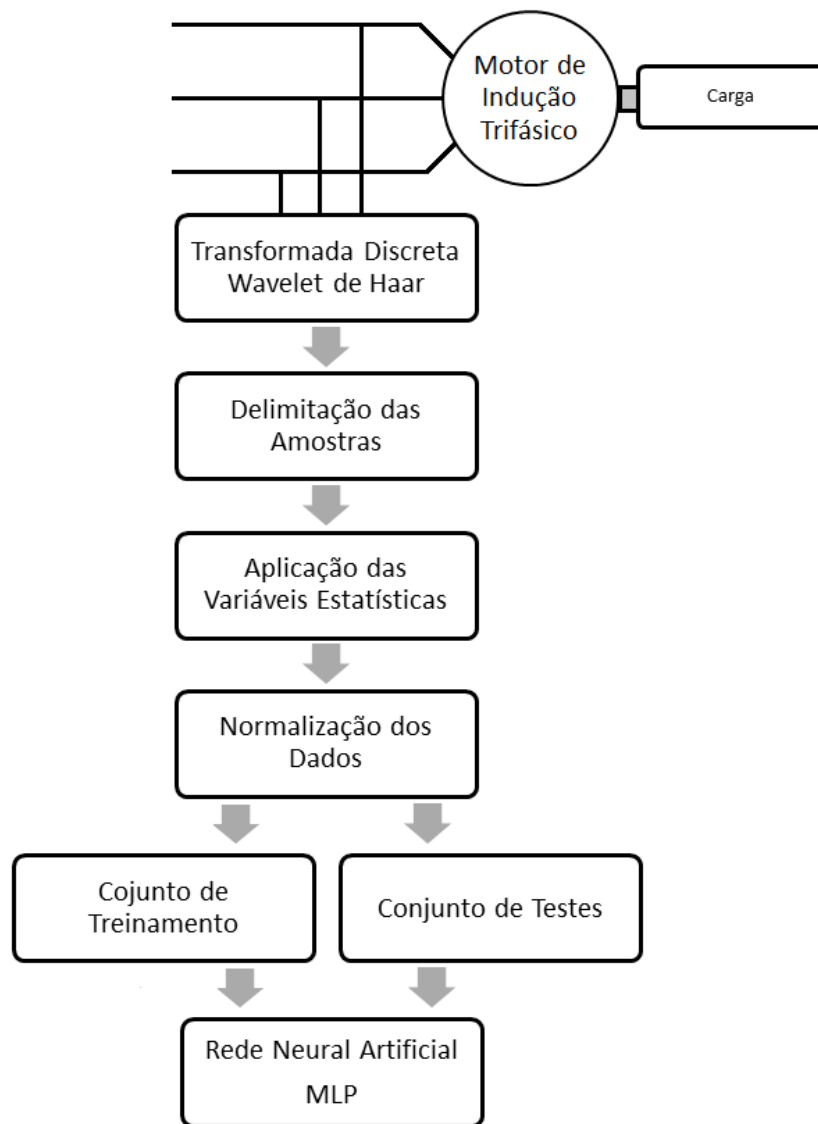


Figura 3.1: Etapas do Sistema de Classificação de Falhas

3.2 Construção da Base de Dados

A aquisição e a transformação dos sinais de analógico para digital foi realizada através da placa NIDAQmx PCIc-6259 e por um computador ao qual esta estava conectada. A frequência de amostragem utilizada para a coleta de dados foi de 3840 Hz. Os dados coletados dizem respeito a MIT's saudáveis e MIT's que apresentam barras danificadas.

É importante utilizar diversos tipos de motores sadios, a fim de deixar o classificador robusto o suficiente para se adequar a situações reais. Os seguintes motores sadios foram utilizados:

- **Motor Normal:** Um motor com adaptação para rotores sadios ou com barras quebradas;
- **Motor de Fábrica:** Um motor novo que não apresenta nenhum tipo de alteração.
- **Motor de Alto Rendimento:** Este motor utiliza materiais diferentes em sua construção comparado ao motor de fábrica;

O classificador tem como objetivo classificar se o motor está sadio, ou se ele apresenta alguma barra de seu rotor com defeito. Os tipos de classificação serão:

- 1 barra danificada;
- 2 barras danificadas;
- 4 barras danificadas;
- Motor em condições normais de funcionamento.

Para cada motor citado acima, juntamente com os possíveis problemas, foram aplicadas diversas cargas em situações de acionamento diferentes.

Para o acionamento direto, isto é, pela rede elétrica, foram utilizadas cargas de 0,5 Nm, 1,0 Nm, 1,5 Nm, 2,0 Nm, 2,5 Nm, 3,0 Nm, 3,5 Nm e 4,0 Nm. Já para os ensaios com os motores sendo acionados pelo inversor, foram utilizadas as mesmas cargas, porém, os motores foram alimentados com sinais que apresentavam frequência de 40 Hz, 50 Hz e 60 Hz.

É importante que o banco de dados seja diversificado, pois assim, a MLP terá informações suficientes para generalizar os casos de falha no motor.

Para cada aquisição feita, isto é, para cada tipo de carga ou frequência+carga no caso do acionamento por inversor, tem-se um arquivo com as informações organizadas conforme a Tabela (3.1).

Tabela 3.1: Dados adquiridos

<i>Tempo</i>	V_a	V_b	V_c	I_a	I_b	I_c
<i>tempo₁</i>	XX	XX	XX	XX	XX	XX
<i>tempo₂</i>	XX	XX	XX	XX	XX	XX

No caso deste trabalho, apenas as colunas referentes às correntes trifásicas serão consideradas.

A Figura 3.2 apresenta as correntes trifásicas de um MIT sem falhas acionado através da rede elétrica em 60 Hz. A literatura de máquinas elétricas diz que as correntes trifásicas de um motor são senoidais e equilibradas quando este é alimentado por tensões senoidais equilibradas. De fato, isso não ocorre devido à distribuição de fluxo magnético não ser uniforme e entre outros fatores.

A Figura 3.3 apresenta as correntes trifásicas de um MIT sem falhas acionado pelo inversor trifásico. Note que estes sinais de corrente são mais distorcidos em relação ao anterior.

Quando há algum problema no MIT, um curto-circuito ou uma barra do rotor danificada, as correntes trifásicas irão apresentar um comportamento ligeiramente diferente, aparecendo sinais de baixa frequência que se mantêm ao longo do tempo.

A Figura 3.4 apresenta as formas de onda das correntes trifásicas de um MIT acionado pela rede elétrica que apresenta 1 barra danificada no rotor.

Note que os sinais da Figura 3.4 que representam o motor com 1 barra quebrada são muito parecidos com os sinais da Figura 3.2. Isto acontece porque as frequências presentes no sinal com o rotor danificado são muito baixas, quase imperceptíveis, mostrando a dificuldade de se identificar os motores com barras quebradas apenas utilizando os sinais puros. Esse problema será resolvido utilizando a TW que permite ajustar o tamanho da janela de processamento por várias componentes de frequência [10].

A Figura 3.5 apresenta as formas de onda da corrente para o motor com 1 barra danificada acionado por um inversor trifásico. Novamente, o mesmo problema ocorre, porém, aqui é mais

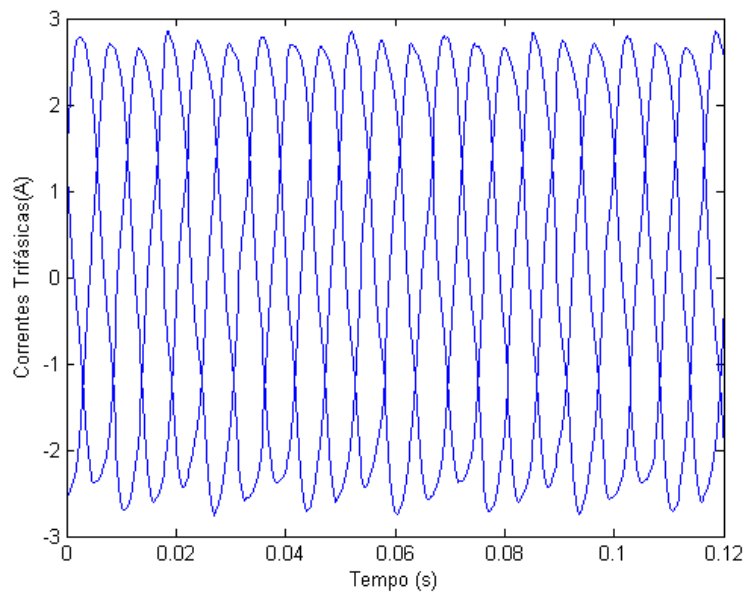


Figura 3.2: Sinais das Correntes Trifásicas do MIT normal acionado pela rede elétrica

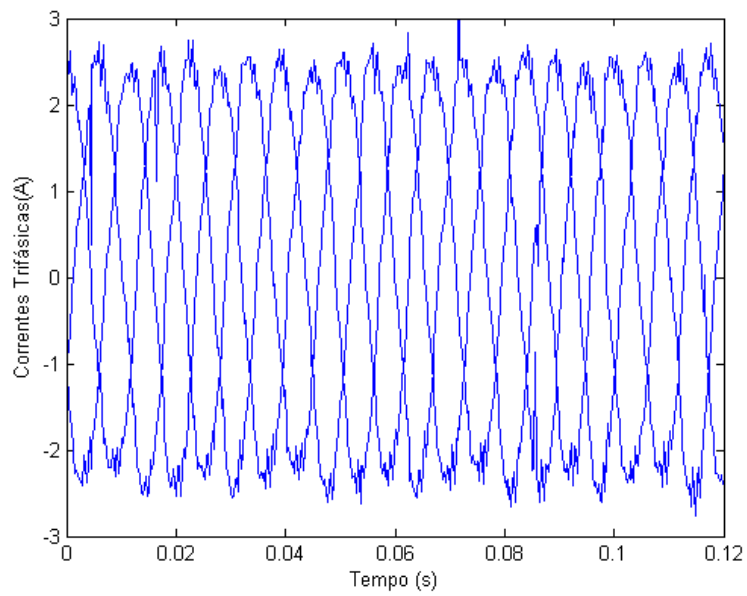


Figura 3.3: Sinais das Correntes Trifásicas do MIT normal acionado pelo inversor trifásico

grave, já que há o aparecimento de ruídos no sinal quando o MIT é acionado por um inversor [16].

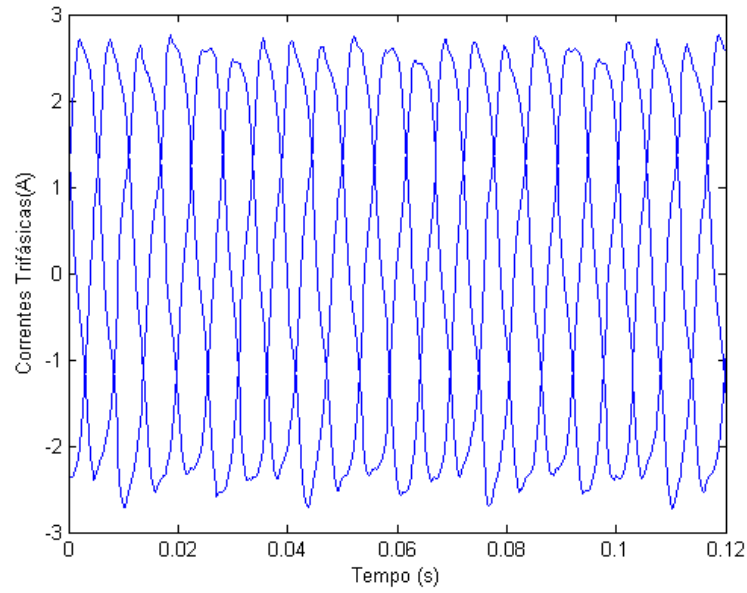


Figura 3.4: Sinais das Correntes Trifásicas do MIT acionado pela rede elétrica com 1 barra danificada

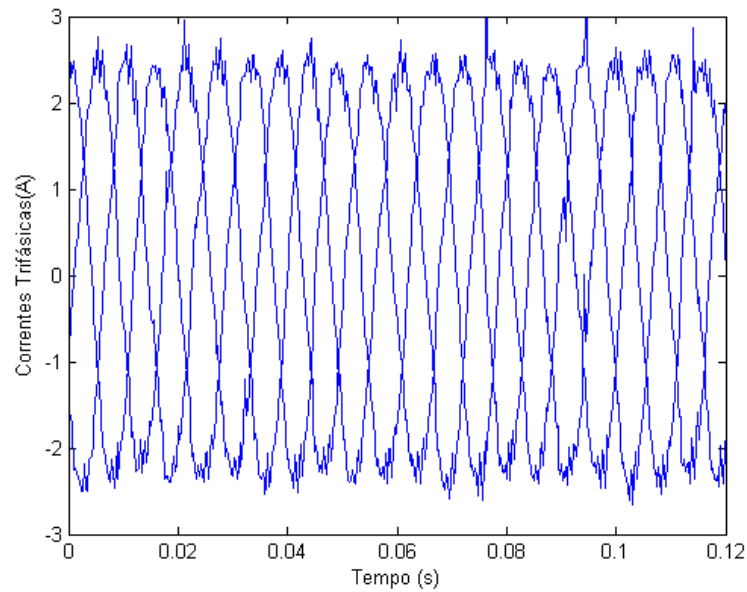


Figura 3.5: Sinais das Correntes Trifásicas do MIT acionado pelo inversor trifásico com 1 barra danificada

3.3 Aplicação da Transformada Wavelet

Como dito anteriormente, a utilização da Transformada Discreta *Wavelet* deve-se ao fato de que os sinais de MIT com e sem problemas são bastante parecidos, e sem um processamento do sinal, não é possível captar as informações que a forma de onda pode nos oferecer. A TDW realça os espectros do sinal que exibem as falhas do motor [10]. Neste trabalho, utilizou-se a Transformada *Wavelet* de Haar, já que esta é bastante simples e confiável, e já foi utilizada em diversas resoluções de problemas [17].

Para os motores acionados através da rede elétrica, a TDW foi aplicada duas vezes nos seus

sinais de corrente, isto é, após a primeira aplicação da TW no sinal original, gerando coeficientes de aproximação e detalhe, uma segunda aplicação foi feita nos coeficientes de aproximação. A Figura 3.6 mostra os coeficientes de aproximação da primeira aplicação da TW e a Figura 3.7 apresenta o mesmo coeficiente, porém com a segunda aplicação da TW. Estes casos apresentados são referentes a um MIT com 1 barra danificada.

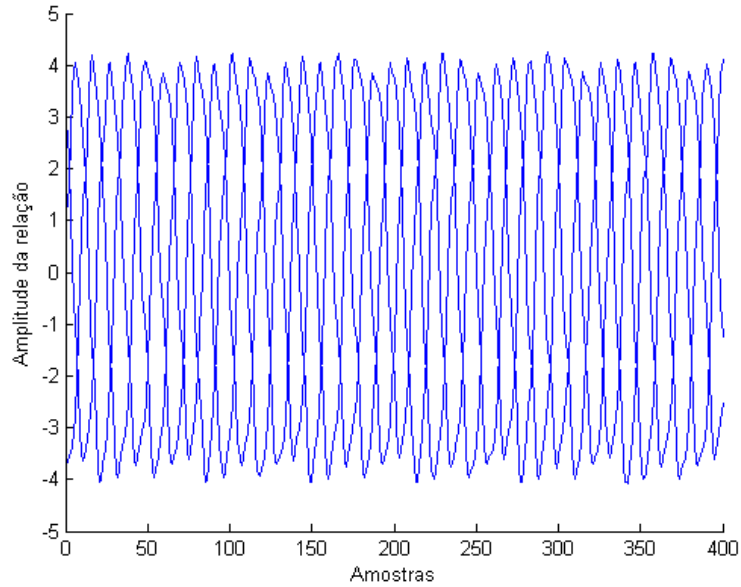


Figura 3.6: Coeficiente de aproximação da primeira aplicação da TW

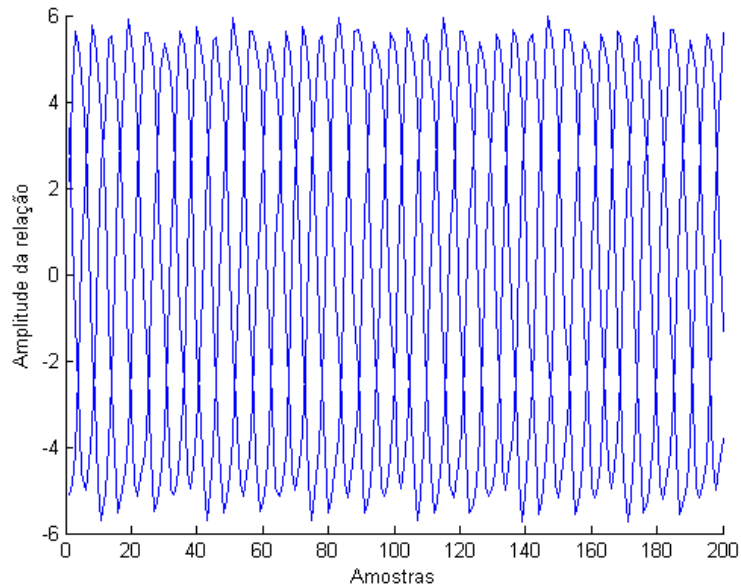


Figura 3.7: Coeficiente de aproximação da segunda aplicação da TW

Note que o número de amostras diminuiu entre a Figura 3.6 e a Figura 3.7, sendo resultado do *downsampling*.

O mesmo procedimento foi feito para as correntes de um MIT acionado pelo inversor. Os resultados dos coeficientes de aproximação podem ser visualizados nas Figuras 3.8 e 3.9, sendo

a primeira e a segunda aplicação da TW, respectivamente.

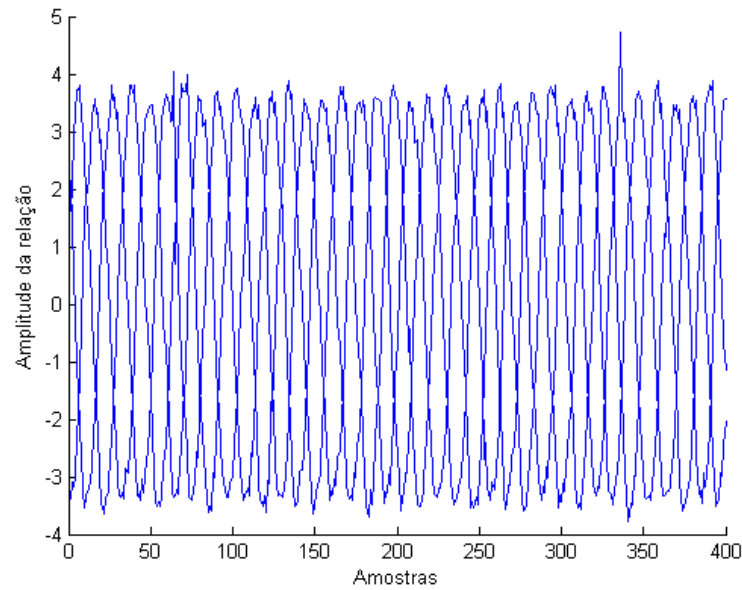


Figura 3.8: Coeficiente de aproximação da primeira aplicação da TW

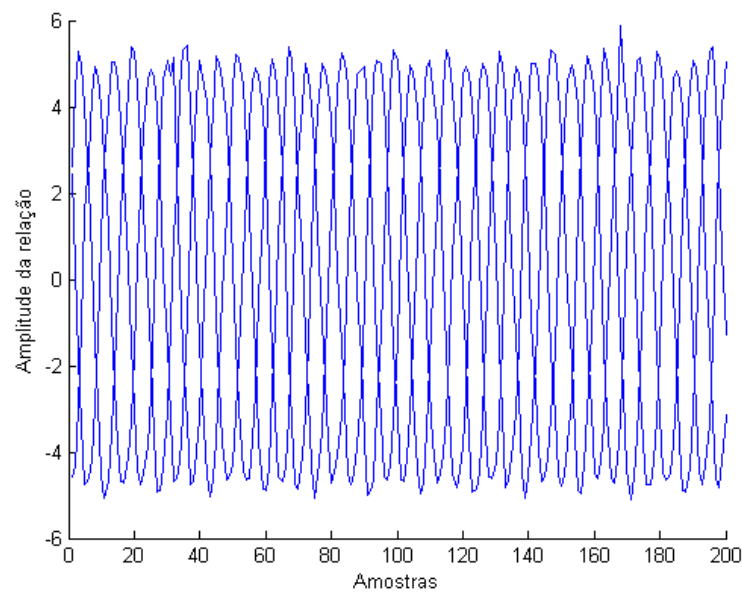


Figura 3.9: Coeficiente de aproximação da segunda aplicação da TW

Note que no segundo caso, em que o MIT foi acionado pelo inversor, os sinais de corrente trifásicos que antes apresentavam muitas deformações ficaram mais visíveis, devido a aplicação da TDW. Isto mostra o quão importante é a TDW, já que esta proporciona a visualização de pequenas características do sinal.

Feitas as aplicações da TDW, o próximo passo foi dividir os sinais em várias amostras, que será explicitado na próxima seção.

3.4 Delimitação das Amostras

Após a aplicação da TW, para cada tipo de acionamento em conjunto com a carga aplicada, obteve-se três sinais que relacionam as correntes elétricas trifásicas com a *mother function* da Transformada de Haar. Estes sinais apresentam 220 ciclos, e estes ciclos serão divididos em amostras.

Cada amostra conterá 10 ciclos completos do sinal. É importante frisar que se deve selecionar ciclos completos para reduzir o efeito de borda causando a menor perda possível de dados importantes. Como cada amostra contém 10 ciclos, têm-se, no total, 22 amostras para cada caso de acionamento em conjunto com a carga.

A Figura 3.10 apresenta a delimitação de uma amostra.

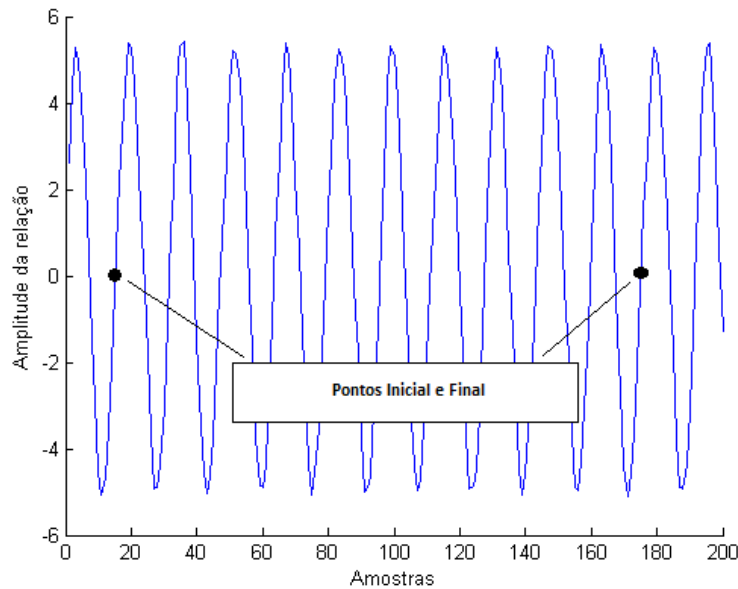


Figura 3.10: Delimitação das amostras

3.5 Aplicação das Variáveis Estatísticas e Normalização dos Dados

As variáveis estatísticas, responsáveis por retirarem informações importantes de cada sinal, foram aplicadas em cima das amostras obtidas anteriormente. Foram aplicadas três tipos de conjuntos de variáveis estatísticas.

Para o acionamento utilizando a rede elétrica, foram aplicados dois conjuntos. No primeiro deles foi utilizado a combinação média, *rms*, *skewness* e *kurtosis*. Estas variáveis foram aplicadas nos três sinais de corrente, resultando em 12 variáveis que serão entregues para as entradas da MLP. No segundo conjunto foram utilizadas mediana, *rms*, *skewness* e *kurtosis*, resultando novamente em 12 variáveis.

Para o acionamento utilizando o inversor, o conjunto média, *rms*, *skewness* e *kurtosis* foi utilizado.

O próximo passo foi normalizar os dados. Para isso foi utilizado uma técnica muito famosa, conhecida como Teorema de Tales. Este utiliza o princípio dos segmentos proporcionais, conforme

mostra a Figura 3.11. A Equação (3.1) mostra como o cálculo deve ser feito.

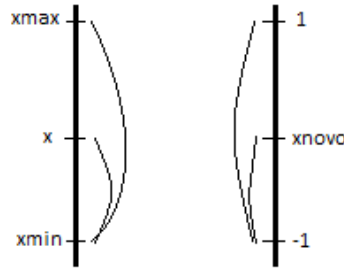


Figura 3.11: Princípio dos segmentos proporcionais

$$x_{novo} = 2 \cdot \left(\frac{x - x_{min}}{x_{max} - x_{min}} \right) - 1 \quad (3.1)$$

Com isso os dados ficaram compreendidos entre o intervalo de $[-1 \ 1]$.

3.6 Separação dos Conjuntos de Teste e Treinamento

Como dito anteriormente, tem-se 22 amostras para cada conjunto de tipo de motor e carga. Destas 22 amostras, 6 foram retiradas para a validação da rede e 16 serão utilizadas para o seu treinamento.

Para o MIT acionado pela rede elétrica são 52 casos possíveis, cada um contendo 22 amostras, totalizando 1144 amostras. Dentre estas 936 serão utilizadas para o treinamento e 312 para os testes, isto é, aproximadamente 27% do banco será utilizado para a validação da rede.

Já para o MIT acionado pelo inversor trifásico tem-se 220 casos possíveis, cada um contendo 22 amostras, totalizando 480 amostras. 3520 serão utilizadas para o treinamento da rede e 1320 para a validação.

A rede neural escolhida para o classificador de falhas foi a Perceptron Multi-Camadas (PMC), do inglês *Multlayer Perceptron* (MLP). As topologias escolhidas seguiram o critério do *cross-validation*, em que se altera a quantidade de neurônios das camadas afim de encontrar a melhor topologia possível para o problema. Para o desenvolvimento da MLP foi utilizado o *toolbox* do *MATLAB* disponível na universidade. A função de ativação para a camada de entrada escolhida foi a tangente hiperbólica, e para a camada de saída foi a linear. Todas as redes foram treinadas com o algoritmo *backpropagation*, utilizando *Levenberg-Marquardt*. As redes apresentam 12 neurônios de entrada, já que todos os três possíveis conjuntos de dados passam a mesma quantidade de informações, e 4 neurônios na saída. A classificação do motor na saída da rede se dará conforme a Tabela 3.2.

Tabela 3.2: Padrão de Saídas Desejadas da MLP

Situação do Motor	Saída Desejada
1 barra quebrada	[1 -1 -1 -1]
2 barras quebradas	[-1 1 -1 -1]
4 barras quebradas	[-1 -1 1 -1]
Motor Normal	[-1 -1 -1 1]

A Figura 3.12 apresenta uma MLP conforme os padrões adotados anteriormente.

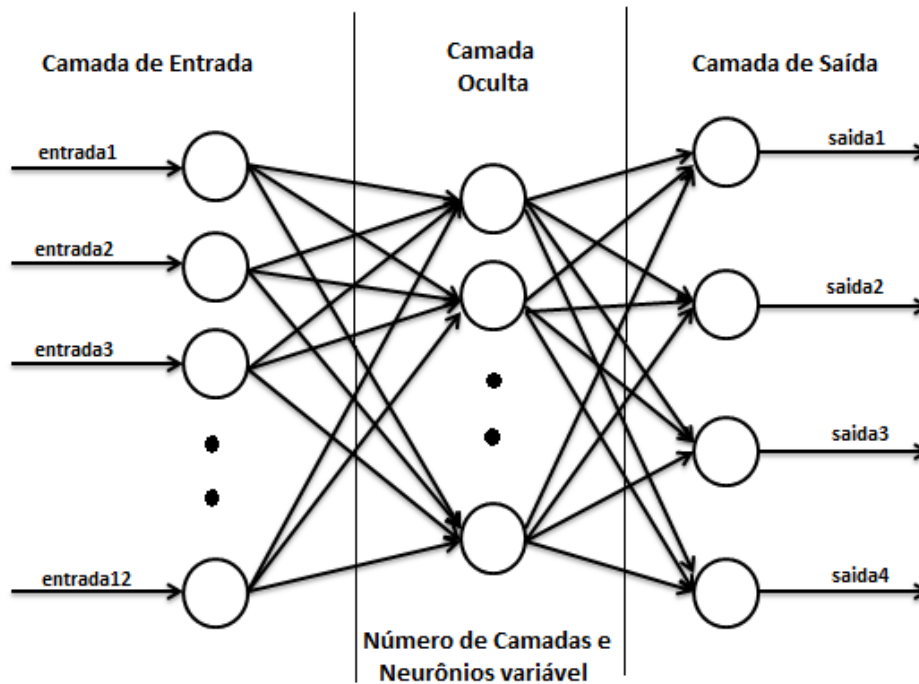


Figura 3.12: Rede PMC utilizada

3.6.1 MIT acionado pela rede elétrica

A primeira rede neural treinada foi para os MITs alimentados diretamente pela rede elétrica. Como dito anteriormente, para este caso há dois conjuntos de dados, um que utiliza a média e outro que utiliza a mediana. Em ambos os casos, a estrutura da rede apresenta: 12 neurônios na camada de entrada, uma camada escondida variando de 17 a 23 neurônios e uma camada de saída com quatro neurônios.

Conjunto de Dados 1

Neste primeiro conjunto, as informações de entrada são: média, *rms*, *skewness* e *kurtosis*. Neste treinamento foi utilizada apenas 1 camada oculta com quantidade de neurônios variando entre 17 e 23. Isto foi feito para encontrar qual a melhor topologia que apresentaria o menor erro médio quadrático. A Figura 3.13 apresenta um gráfico com estas informações. Neste caso foi utilizado um critério de parada referente ao erro quadrático médio no valor de 10^{-10} .

Note que o valor do erro ficou próximo para todas as topologias testadas. O ponto realçado no gráfico da Figura 3.13 foi que apresentou o menor erro, e portanto, como topologia final foi escolhida a que apresenta 19 neurônios na camada escondida.

Conjunto de Dados 2

Neste segundo conjunto, as informações de entrada são: mediana, *rms*, *skewness* e *kurtosis*. Novamente, todas as características do primeiro treinamento foram repetidas para este. A Figura 3.14 apresenta um gráfico com os erros para este conjunto de dados.

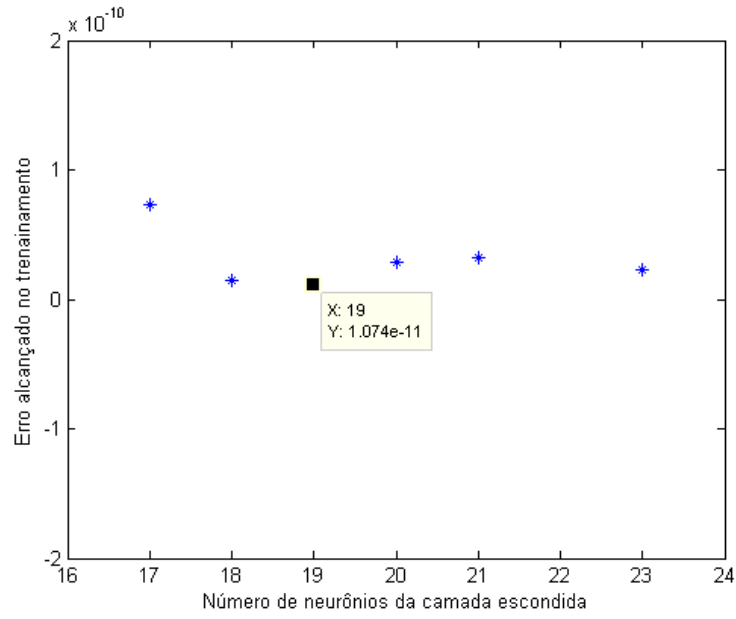


Figura 3.13: Relação entre erro quadrático médio e número de neurônios da camada escondida

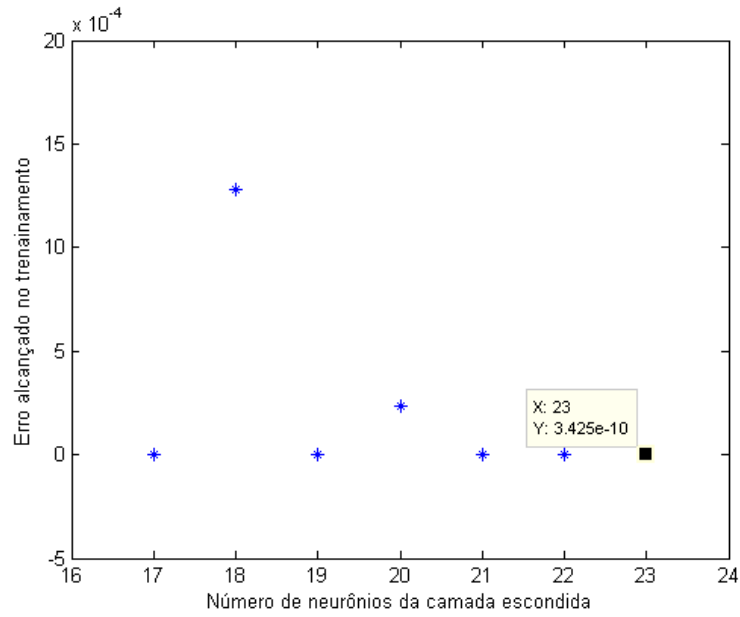


Figura 3.14: Relação entre erro quadrático médio e número de neurônios da camada escondida

Neste caso, foi escolhida a topologia que apresenta 23 neurônios na camada escondida, conforme está realçado no gráfico da Figura 3.14

3.6.2 MIT acionado pelo inversor trifásico

No caso do MIT acionado pelo inversor trifásico, foi considerado apenas um conjunto de informações: média, *rms*, *skewness* e *kurtosis*. Aqui, a PMC/MLP apresenta suas camadas escondidas. A primeira camada escondida teve seu número de neurônios variado entre 17 e 23 a fim de encontrar o menor erro possível. Já a segunda camada escondida teve seu número de neurônios fixos em dez. A Figura 3.15 apresenta os erros para este caso de treinamento. No caso do MIT acionado pelo inversor trifásico, foi considerado apenas um conjunto de informações:

média, *rms*, *skewness* e *kurtosis*. Aqui, a PMC/MLP apresenta suas camadas escondidas. A primeira camada escondida teve seu número de neurônios variado entre 17 e 23 afim de encontrar o menor erro possível. Já a segunda camada escondida teve seu número de neurônios fixos em dez. A Figura 3.15 apresenta os erros para este caso de treinamento.

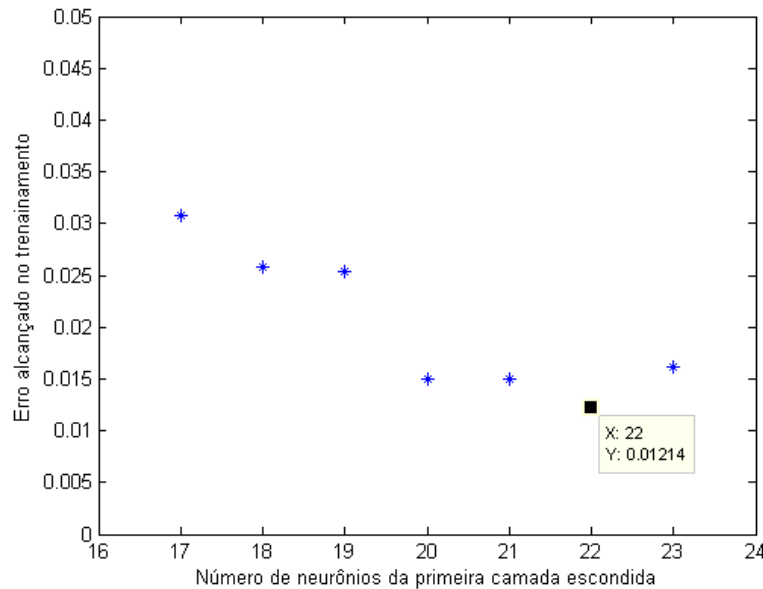


Figura 3.15: Relação entre erro médio quadrático e número de neurônios da primeira camada escondida

Aqui, a melhor topologia possível apresenta 22 neurônios na primeira camada escondida, como realçado na Figura 3.15.

3.6.3 Topologias Finais

A Tabela 3.3 apresenta as topologias finais escolhidas para cada RNA treinada. São estas topologias que serão validadas no próximo capítulo.

Tabela 3.3: Topologias Finais para o Classificador de Falhas

Conjunto de Dados	Número de Neurônios	Menor Erro Quadrático Médio
Acionamento Direto 1	[19,4]	$3,4 \cdot 10^{-10}$
Acionamento Direto 2	[23,4]	$1,7 \cdot 10^{-11}$
Acionamento pelo Inversor Trifásico	[22,10,4]	0,01214

Capítulo 4

Validação das RNA's

4.1 Introdução

Este capítulo tem como intuito validar as RNA's treinadas nos capítulos anteriores, a fim de verificar se estas são capazes de incorporar um sistema de classificação de falhas. As possíveis classificações que as RNA's farão são: Rotor com 1 barra danificada, Rotor com 2 barras danificadas, Rotor com 4 barras danificadas ou Motor Normal. Os motores foram acionados de duas formas diferentes, diretamente pela rede elétrica e por um inversor trifásico.

Todo o sistema foi, mais uma vez, implementado no MATLAB, disponível na universidade, já que este se mostra um interpretador de alto nível.

As seções seguintes apresentam os resultados das validações.

4.2 Validação para a Partida Direta

Esta seção tem como objetivo apresentar a validação da RNA PMC para os motores acionados através da rede elétrica. Como comentado anteriormente, foram utilizados dois conjuntos de dados, a fim de comparar quais variáveis estatísticas que melhor retiraram as características de cada sinal. O sistema aqui criado deve ser capaz de encontrar falhas nas máquinas mostrando quantas barras estão danificadas, ou se as máquinas estão sadias.

A Figura 4.1 mostra a validação da RNA com o primeiro conjunto de testes. Este conjunto contém média, *rms*, *skewness* e *kurtosis*.

Note que todos os testes feitos obtiveram êxito, isto é, a MLP conseguiu classificar 100% dos casos. A Tabela 4.1 apresenta a situação real do MIT e como a RNA o classificou.

Tabela 4.1: Matriz de validação resultante para falhas no rotor acionado pela rede elétrica - Conjunto de dados 1

Situação Real do Motor	Classificação da PMC				Total
	1 Barra Danificada	2 Barras Danificadas	4 Barras Danificadas	Motor Normal	
1 Barra Danificada	30	0	0	0	30
2 Barras Danificadas	0	30	0	0	30
4 Barras Danificadas	0	0	48	0	48
Motor Sadio	0	0	0	204	204
Total	30	30	48	204	312

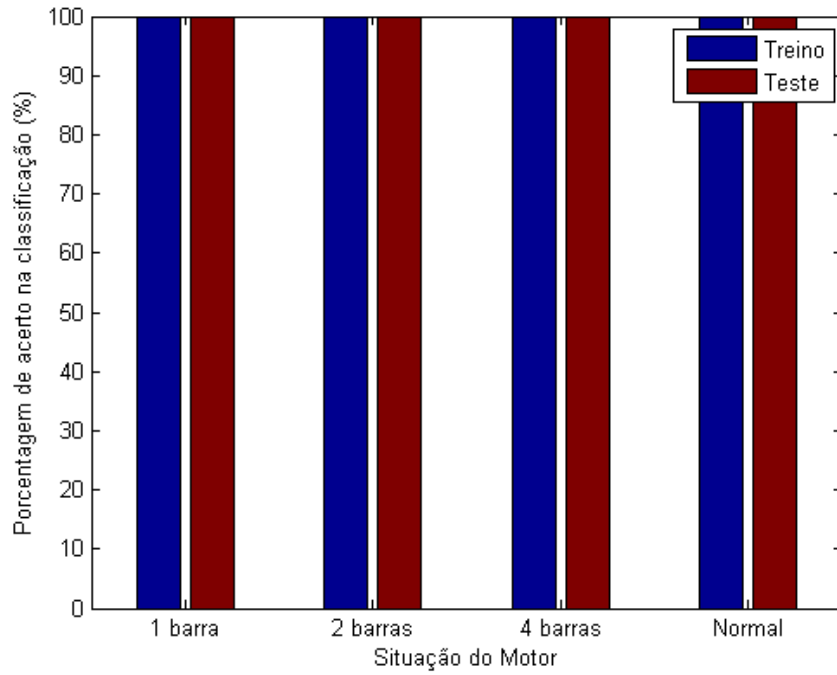


Figura 4.1: Porcentagem de acertos para o primeiro conjunto de testes

O 100% de acerto da RNA deve-se ao fato de que as variáveis estatísticas foram bem escolhidas, isto é, este conjunto de informações consegue retirar as características de cada sinal e distingui-las extremamente bem.

A Figura 4.2 mostra a validação da RNA com o primeiro conjunto de testes. Este conjunto contém mediana, *rms*, *skewness* e *kurtosis*.

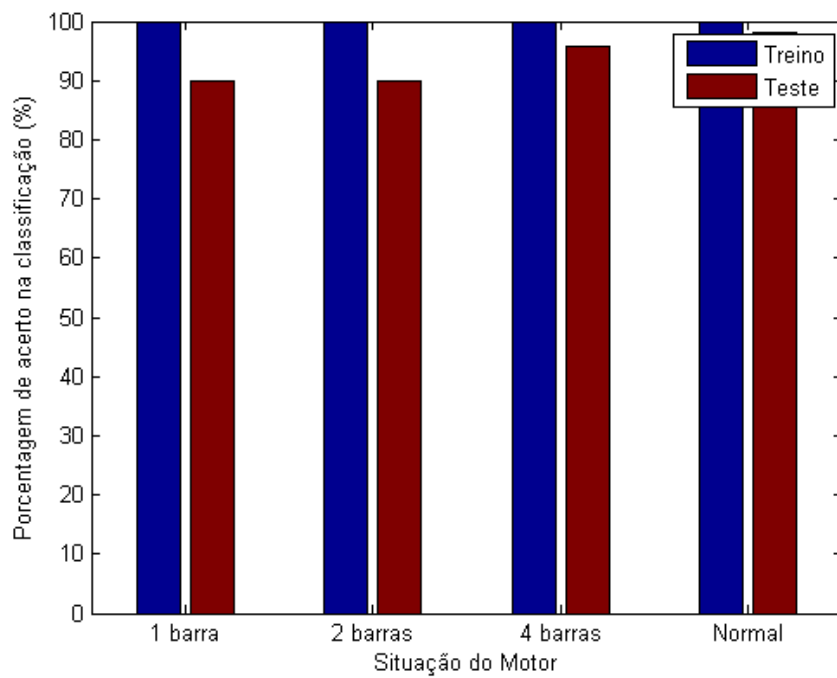


Figura 4.2: Porcentagem de acertos para o segundo conjunto de dados

Note que para este conjunto de informações, a RNA não conseguiu 100% de acertos. A Tabela 4.2 mostra a porcentagem de acertos para cada caso, e a Tabela 4.3 mostra a situação real do MIT e como a RNA o classificou.

Tabela 4.2: Tabela de Porcentagem de Acertos

1 Barra Danificada	2 Barras Danificadas	4 Barras Danificadas	Motor Normal
90,00%	90,00%	95,83%	98,04%

Tabela 4.3: Matriz de validação resultante para falhas no rotor acionado pela rede elétrica - Conjunto de dados 2

Situação Real do Motor	Classificação da PMC				Total
	1 Barra Danificada	2 Barras Danificadas	4 Barras Danificadas	Motor Normal	
1 Barra Danificada	27	0	2	1	30
2 Barras Danificadas	3	27	0	0	30
4 Barras Danificadas	2	0	46	0	48
Motor Sadio	0	0	3	201	204
Total	32	27	51	202	312

Mesmo que o resultado não tenha sido 100% de acerto, a RNA ainda conseguiu classificar muito bem a situação de cada motor, errando apenas 10% ou menos em cada caso.

Aqui há algo interessante a se observar. No primeiro caso a rede acertou 100% dos testes, o que não aconteceu no segundo. Isto significa que o primeiro conjunto de variáveis estatísticas, responsáveis por extrair as características do sinal, está trazendo características que diferenciam mais os sinais do que o segundo conjunto. Do primeiro para o segundo conjunto, trocou-se apenas uma variável, a média pela mediana, portanto foi feito um estudo sobre estas variáveis.

O gráfico da Figura 4.3 apresenta a variável média aplicada em uma das correntes trifásicas do motor. Note que esta tem intervalos bem definidos. A primeira parte, em azul, representa motores com 1 barra quebrada. Note que os valores da média se mantêm em uma mesma posição. Na segunda parte, em vermelho, são representados os motores com 2 barras danificadas. Novamente os valores se mantêm próximos. Na terceira parte, em rosa, há 2 intervalos bem definidos representando 4 barras quebradas, e o último intervalo, em verde, representa o motor sem problemas. Note que todos os intervalos estão bem definidos e diferentes um em relação ao outro. Isso mostra que a **média** foi uma ótima variável para a extração de características.

Já o gráfico da Figura 4.4 apresenta a variável mediana aplicada, novamente, em uma das correntes do motor. Note que aqui, não há como diferenciar que intervalo pertence a qual tipo de classificação, já que a mediana da corrente apresenta valores próximos para qualquer tipo de falha. Sendo assim a classificação utilizou as outras três variáveis estatísticas para separar os tipos de falhas. Isto mostra que a **mediana** não extrai bem as características de um sinal.

Concluindo, o melhor conjunto para extrair as características dos sinais que passaram pela TW compreende: média, *rms*, *skweness* e *kurtosis*.

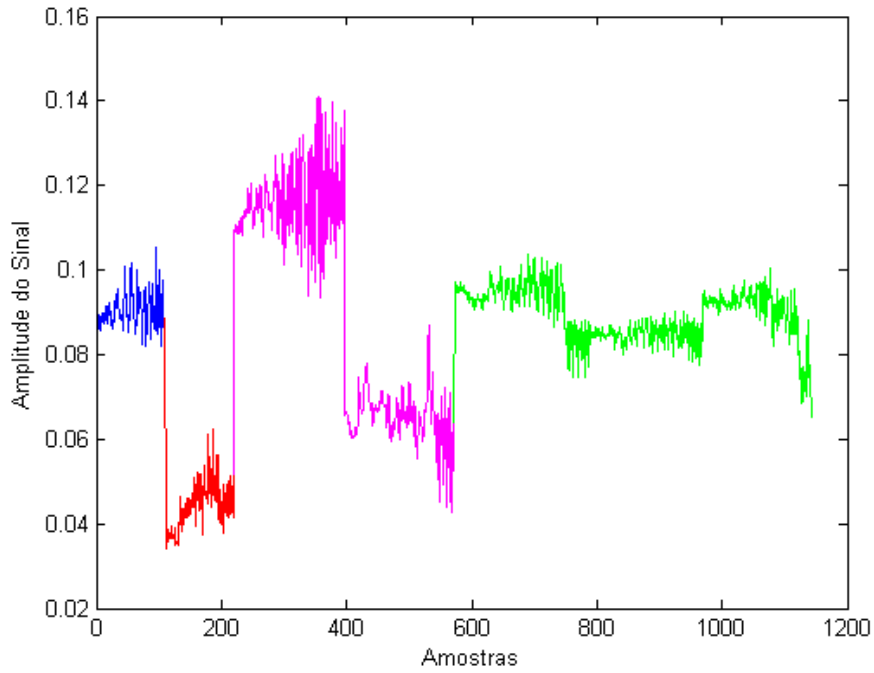


Figura 4.3: Média da Corrente I_a para todos os casos de testes

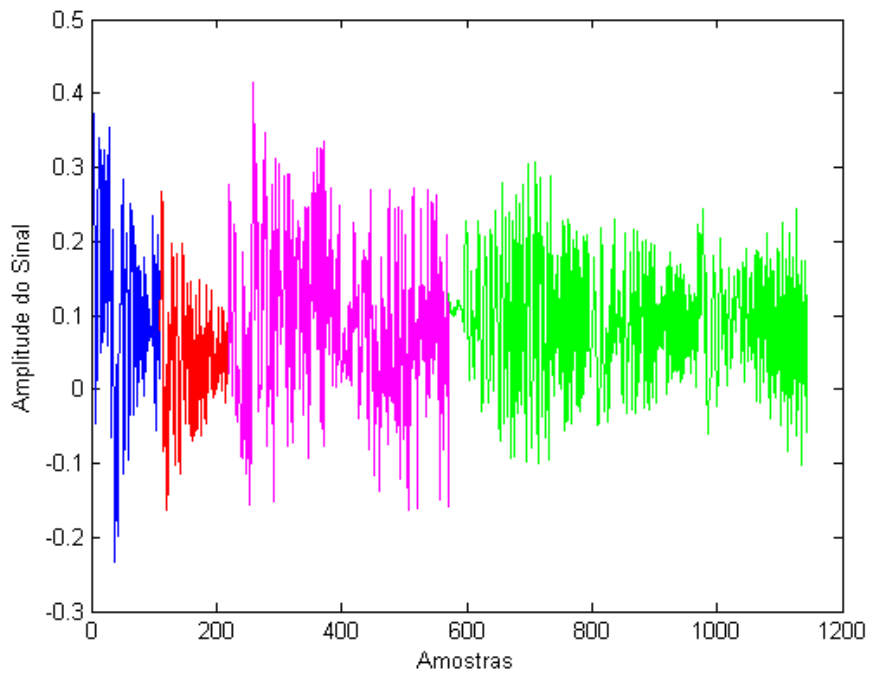


Figura 4.4: Mediana da Corrente I_a para todos os casos de testes

4.3 Validação para a Partida Utilizando Inversor Trifásico

Esta seção tem como objetivo apresentar a validação da RNA PMC para os motores acionados pelo inversor trifásico. Visto que, na seção anterior, a variável média se mostrou melhor para extrair as características do sinal em relação a mediana, apenas um conjunto de dados foi então utilizado aqui. O sistema agora criado deve ser capaz de encontrar falhas nas máquinas mostrando

quantas barras estão danificadas, ou se as máquinas estão sadias. Dentre as diferentes cargas possíveis de serem aplicadas, os sinais de tensão tiveram sua frequência variada, a fim de deixar o sistema de classificação mais robusto.

A Figura 4.5 mostra a porcentagem de acertos para cada situação.

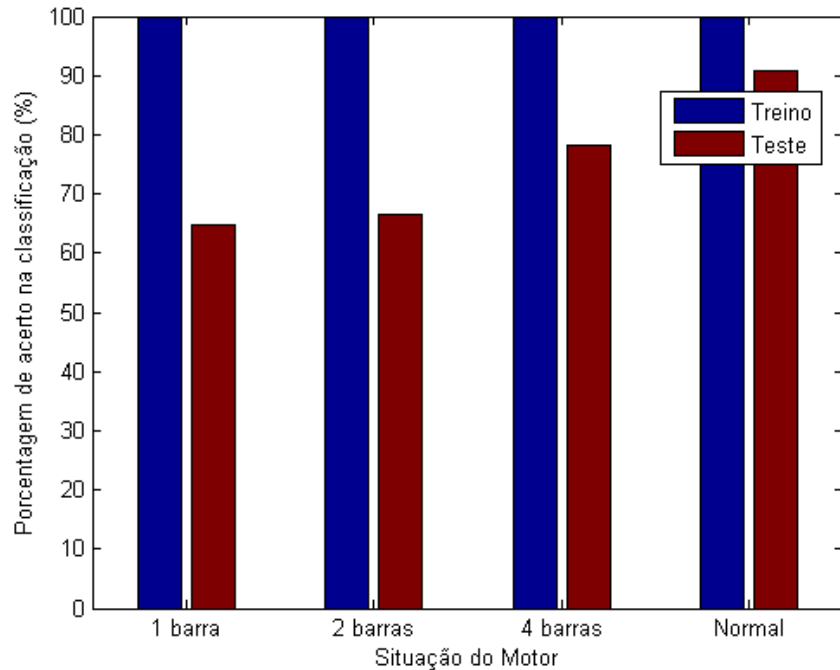


Figura 4.5: Porcentagem de acertos

Note que, neste caso, a RNA não conseguiu classificar bastante parte dos dados, principalmente quando o motor apresentava 1 barra quebrada. Nesta validação, a RNA chegou dar resultados incompatíveis com os quatro possíveis, isto é, ela classifica que um motor está ao mesmo tempo com 1 barra e 2 barras quebradas. O grande número de erros pode ter acontecido porque os motores acionados por um inversor apresentam muitos ruídos em seus sinais de corrente, dificultado a extração de características. Informações mais detalhadas podem ser encontradas na Tabela 4.4 e na Tabela 4.5.

Tabela 4.4: Tabela de Porcentagem de Acertos

1 Barra Danificada	2 Barras Danificadas	4 Barras Danificadas	Motor Normal
62,96%	66,67%	78,33%	90,73%

Tabela 4.5: Matriz de Validação Resultante para Falhas no Motor Acionado pelo Inversor Trifásico

Situação Real do Motor	Classificação da PMC				Total	Total Real
	1 Barra Danificada	2 Barras Danificadas	4 Barras Danificadas	Motor Normal		
1 Barra Danificada	42	0	8	11	61	54
2 Barras Danificadas	0	44	0	22	66	66
4 Barras Danificadas	10	0	190	48	248	240
Motor Sadio	10	17	62	871	960	60
Total	62	61	260	952	1335	1320

Como visto, a PMC não apresentou um resultado muito satisfatório para a classificação de defeitos nas barras de motores acionados por um inversor trifásico. Isto pode ter sido causado pelos altos graus de ruídos que um inversor adiciona no sinal de corrente do motor. Algumas possíveis soluções seriam:

- Investigar outras variáveis estatísticas que caracterizem melhor cada sinal;
- Aumentar o número de aplicações da TW no sinal de corrente;
- Aumentar o número de camadas escondidas do RNA;
- Separar redes neurais que classificam motores utilizados para certos valores de carga;
- Alterar a PMC para alguma outra topologia, como, por exemplo a RBF;
- Adição de filtros de corrente nos sinais.

Mesmo assim, a RNA conseguiu classificar alguns casos, o que já a torna viável para a implementação em indústrias.

Dentre todas as opções, a investigação por outras variáveis estatísticas talvez seja a mais viável. É importante analisar bem o conjunto de dados que está sendo tratado, e com isso, a escolha de certas variáveis estatísticas podem facilitar ou dificultar o trabalho da RNA na classificação das falhas.

Capítulo 5

Conclusão

O objetivo deste Trabalho de Conclusão de Curso foi implementar um classificador de falhas para MIT's que pudesse trazer informações de possíveis falhas antes de uma quebra permanente do motor, causando prejuízos. O intuito era medir os valores das correntes trifásicas do motor e com isso classificar uma possível falha.

As correntes trifásicas do motor precisam passar por um pré-processamento do sinal antes de serem utilizadas, pois este pré-processamento separa informações importantes para a implementação do sistema. Para isso foi estudado e utilizado a Transformada Discreta *Wavelet*, mais precisamente a Transformada de Haar. Esta transformada separa, utilizando blocos de filtros, as frequências altas e as frequências baixas do sinal para posterior utilização. Motores que apresentam barras quebradas tendem a apresentar frequências próximas à frequência fundamental do sinal, e, portanto, a transformada foi utilizada para realçar estas frequências.

Após o processamento da transformada, os sinais passaram por diversas variáveis estatísticas, as quais foram capazes de extrair as informações importantes dos sinais de corrente. Foram utilizadas as variáveis: média, mediana, *rms*, *skewness* e *kurtosis*. Estas variáveis foram divididas em conjuntos, com o intuito de encontrar a melhor combinação de variáveis que extraíssem as informações do sinal.

Estas informações foram separadas em conjuntos de treino e validação. O conjunto de treino fez parte dos dados de entrada de uma RNA do tipo PMC/MLP para que fosse possível encontrar a melhor topologia para a rede. E por fim, o conjunto de validação foi colocado na rede para verificar as informações.

Foram utilizados dois tipos de acionamento dos motores. O acionamento direto e o acionamento através de um inversor trifásico. Os resultados para o acionamento direto ficaram extremamente bons, ressaltando que, com a utilização da média como uma variável estatística, a rede acertou 100% dos casos testados. Já para o caso do acionamento através do inversor trifásico, a rede não obteve tanto êxito. Isso deve-se ao fato do inversor adicionar muitos ruídos aos sinais de corrente da máquina, dificultando a extração das características. Mesmo assim, ela ainda obteve um resultado satisfatório, podendo ser melhorado realizando pequenas mudanças conforme explicitado anteriormente.

Todos estes resultados mostraram que, a Transformada Discreta *Wavelet* aliada às variáveis estatísticas extraem informações importantes e suficientes para que a classificação das falhas seja realizada com sucesso. Também foi possível observar que a MLP é um ótimo algoritmo classificador, mas que pode ser melhorado se o número de neurônios e o número de camadas

escondidas for alterado.

Para um trabalho futuro, que visa o objetivo de melhorar a classificação quando o motor for acionado por inversores, pode-se aplicar mais vezes a TW e utilizar outras topologias de redes neurais, aliando-se a algoritmos genéticos para encontrar o melhor número de neurônios possível em cada camada.

Referências Bibliográficas

- [1] TRZYNADLOWSKI, M. A. **Control of Induction Motors**. [S.l.]: Academic Press, 2001.
- [2] GOEDTEL, A. **Estimador Neural de Velocidade para Motores de Indução Trifásicos**. 2007. Tese de Doutorado, Universidade de São Paulo (USP/EESC)
- [3] YEH, C.-C.; DEMERDASH, N. A. O. Induction Motor-Drive Systems with Fault Tolerant Inverter-Motor Capabilities. In: IEEE INTERNATIONAL ELECTRIC MACHINES & DRIVES CONFERENCE (IEMDC), 2007. **Proceedings...** [S.l.: s.n.], 2007. v.2, p.1451-1458.
- [4] FILIPPETTI, F.; FRANCESCHINI, G.; TASSONI, C.; VAS, P. Recent development of induction motor drives fault diagnosis using AI techniques. **IEEE Transactions on Industrial Electronics**. [S.l.], v.47, n.5, p.994-1004, 2000.
- [5] SUETAKE, M. **Sistemas inteligentes para monitoramento e diagnóstico de falhas em motores de indução trifásicos**, São Carlos, 2012.
- [6] BELLINI, A.; FILIPPETTI, F.; TASSONI, C.; CAPOLINO, G. A. Advances in diagnostic techniques for induction machines. **IEEE Transactions on Industrial Electronics**. [S.l.], v.55, n.12, p4109-4126, December 2008a.
- [7] CODY, M. A., **The Fast Wavelet Transform Beyond Fourier Transforms**, Dr. Dobb's Journal, v. 17, n. 4, April, 1992.
- [8] EDWARDS, T., **Discrete Wavelet Transform: Theory and Implementation**, Tech. Rep., Stanford University, September, 1991.
- [9] VIDAKOVIC, B.; MUELLER, P. *Wavelet for kids: a tutorial introduction*. [S.l.]: Duke University, 1991.
- [10] SANTOS, F. M. da C.; **Identificação de Falhas em Motores de Indução Trifásicos Usando Sistemas Inteligentes** São Carlos, 2013, Tese de Doutorado, Universidade de São Paulo (USP/EESC)
- [11] COCHRAN, W. G. Statistical methods. The Iowa State University; Ames, Iowa, 1967, 593p.
- [12] LIVINSTONE, D. **A practical guide to scientific analysis**. [S.l.]: John Wiley and Sons, 2009, 341p.
- [13] SILVA, I. N. da; SPATTI, D. H.; FLAUZINO, R. A. **Redes Neurais Artificiais para Engenharia e Ciências Aplicadas**. São Paulo, SP: Artliber, 2010.

- [14] MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **Bulletin of Mathematical Biophysics**, [S.l.], v.5, p.115-133, 1943.
- [15] BEALE, M. H.; HAGAN, M. T.; DEMUTH, H. B. **Neural Network Toolbox: user's guide**. Natick, MA: The MathWorks, Inc., 2011.
- [16] WU, S.; CHOW, T. Induction machine fault detection using SOM-based RBF neural networks. **IEEE Transactions on Industrial Electronics**, [S.l.], v.51, n.1, p.183-194, Feb. 2004.
- [17] MALLAT, S. **A Wavelet Tour of Signal Processing: the sparse way**. 3.ed. [S.l.]: Elsevier, 2009, 832p.