

**UNIVERSIDADE SANTA CECÍLIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
MESTRADO EM ENGENHARIA MECÂNICA**

MÁRCIO PEREIRA CORRÊA

**LÓGICA PARACONSISTENTE ANOTADA APLICADA AO MONITORAMENTO DE
CONDIÇÃO E PREVENÇÃO DE FALHA EM ATIVOS DA INDÚSTRIA COM BASE
EM ASSINATURAS DE VIBRAÇÃO**

SANTOS/SP

2022

MÁRCIO PEREIRA CORRÊA

**LÓGICA PARACONSISTENTE ANOTADA APLICADA AO MONITORAMENTO DE
CONDIÇÃO E PREVENÇÃO DE FALHA EM ATIVOS DA INDÚSTRIA COM BASE
EM ASSINATURAS DE VIBRAÇÃO**

Dissertação apresentada a Universidade Santa Cecília como parte dos requisitos para obtenção de título de mestre no Programa de Pós-Graduação em Engenharia Mecânica, sob a orientação do Prof. Dr. João Inácio da Silva Filho.

SANTOS/SP

2022

Autorizo a reprodução parcial ou total deste trabalho, por qualquer que seja o processo, exclusivamente para fins acadêmicos e científicos.

511.31 Corrêa, Márcio Pereira.

C844L Lógica paraconsistente anotada aplicada ao monitoramento de condição e prevenção de falha em ativos da indústria com base em assinaturas de vibração. /

Márcio Pereira Corrêa.

2022.

116 f.

Orientador: Prof. Dr. João Inácio da Silva Filho.

Dissertação (Mestrado)-- Universidade Santa Cecília, Programa de Pós-Graduação em Engenharia Mecânica, Santos, SP, 2017.

1. Lógica paraconsistente anotada. 2. Manutenção. 3. Preventiva. 4. Prescritiva. 5. Corretiva. 6. Ativos. 7. Inteligência artificial. I. da Silva Filho, João Inácio. II. Lógica paraconsistente anotada aplicada ao monitoramento de condição e prevenção de falha em ativos da indústria com base em assinaturas de vibração.

DEDICATÓRIA

Dedico este trabalho à minha mãe, minha esposa e minhas filhas que tiveram paciência para suportar minha ausência.

AGRADECIMENTOS

Agradeço imensamente ao meu orientador professor João Inácio da Silva Filho, pelo suporte, direcionamento, incentivo e dedicação. Obrigado pelo ensinamento, unindo a experiência acadêmica com o conhecimento técnico da indústria, minha expectativa é levar esses ensinamentos para toda a vida profissional.

O apoio e suporte da Sandra e Imaculada, sempre com muita paciência, ajudando com as questões burocráticas.

Todos os professores que lecionaram as disciplinas cursadas, em especial a professora Dora e Maurício, pela dedicação, durante as aulas remotas.

Aos colegas de trabalho e amigos, Aldecir Menezes e Ayslan Machado pelo apoio e amizade.

Finalizando agradeço, a toda a minha família: Simone minha esposa e parceira de uma vida, minha mãe pela formação de caráter, pela memória do meu pai, meus queridos irmãos Érico e Kelly, minhas filhas Gabriela e Beatriz, vocês são a motivação me alimenta durante toda a minha jornada.

RESUMO

Neste estudo será apresentado um sistema especialista paraconsistente para monitoramento de riscos de vibração (SEvibrPAL2v) de ativos. O SEvibrPAL2v é baseado na análise de assinatura de vibração por meio de um conjunto de algoritmos fundamentados em Lógica Paraconsistente Anotada – LPA. A LPA é uma lógica não clássica, que possui como principal característica suportar sinais contraditórios em sua fundamentação e, portanto, adequada para a construção de modelos algorítmicos capazes de realizar um tratamento eficiente para sinais complexos, como os provenientes de vibração de máquinas. O SEvibrPAL2v foi construído em um microcontrolador ATmega2560 funcionando em conjunto com um programa, onde os sinais de vibração são capturados das estruturas mecânicas por sensores e, após receberem tratamento especial através da Transformada de Fourier Discreta (TFD), são então, com os algoritmos da LPA, na sua forma estendida de dois valores (LPA2v), modelados adequadamente para sinais lógicos paraconsistentes. Após essa modelagem inicial é aplicado um treinamento com o algoritmo de recorrência Extrator de efeitos da Contradição (ParaExtract) onde os sinais lógicos paraconsistentes são transformados em padrões paraconsistentes de assinaturas de vibração. Usando os fundamentos da lógica paraconsistente, os padrões de assinatura de vibração foram rotulados para diferentes problemas de vibração possíveis e conhecidos. Os padrões, resultantes do aprendizado supervisionado de máquina, foram armazenados no SEvibrPAL2v onde as entradas e saídas permitem que os valores sejam continuamente comparados através de configurações compostas por redes de algoritmos paraconsistentes, detectando anomalias e gerando sinais que informam o *status* de risco atual em tempo real da máquina. Os ensaios de comprovação da eficiência do SEvibrPAL2v foram feitos em análises inicialmente realizadas em pequenos protótipos e, após os ajustes iniciais, foram realizados os testes em um conjunto mecânico de média potência construído especificamente para este estudo. Os resultados são apresentados ao final deste estudo e possuem um alto índice de acurácia na identificação de assinatura e detecção de risco de falha, o que justifica o método utilizado e futuras aplicações, mesmo considerando que o SEvibrPAL2v ainda está em sua primeira versão.

Palavras-chave: Lógica paraconsistente anotada; Manutenção; Preventiva; Prescritiva, Corretiva; Ativos; Inteligência artificial; Indústria 4.0.

ABSTRACT

In this study, we introduced an expert system (SEvibrPAL2v), responsible for monitoring assets based on vibration signature analysis through a set of algorithms based on the Paraconsistent Annotated Logic –PAL. Being a non-classical logic, the main feature of the PAL is to support contradictory inputs in its foundation. It is therefore suitable for building algorithmic models capable of performing out appropriate treatment for complex signals, such as those coming from vibration. The SEvibrPAL2v was built on an ATmega2560 microcontroller, where vibration signals were captured from the mechanical structures of the machines by sensors and, after receiving special treatment through the Discrete Fourier Transform (DFT), then properly modeled to paraconsistent logic signals and vibration patterns. Using the PAL fundamentals, vibration signature patterns were built for possible and known vibration issues stored in SEvibrPAL2v and continuously compared through configurations composed by a network of paraconsistent algorithms that detects anomalies and generate signals that will report on the current risk status of the machine in real time. The tests to confirm the efficiency of SEvibrPAL2v were performed in analyses initially carried out on small prototypes and, after the initial adjustments, tests were carried out on bearings of a group of medium-power motor generators built specifically for this study. The results are shown at the end of this study and have a high index of signature identification and risk of failure detection. These results justify the method used and future applications considering that SEvibrPAL2v is still in its first version.

Keywords: Paraconsistent annotated logic; Maintenance; Preventive; Prescriptive, Corrective; Assets; Artificial intelligence; Industry 4.0

LISTA DE FIGURAS

Figura 1: Reticulado de quatro vértices (Hasse).....	18
Figura 2: Reticulado τ com o sistema de coordenadas.....	19
Figura 3: Reticulado τ composto de um novo sistema de coordenadas.....	20
Figura 4: Aumento de escala de $\sqrt{2}$	20
Figura 5: Rotação de 45° em relação a origem $\sqrt{2}$	21
Figura 6: Translação de valores entre QUPC e o reticulado da LPA2v $\sqrt{2}$	21
Figura 7: Ponto de Interpolação (G_c , G_{ct}) e distância D	23
Figura 8: Grau de Certeza resultante real, G_{cr} no reticulado.....	24
Figura 9: Valor do Grau de Evidência com variação linear e diretamente proporcional a grandeza medida.....	25
Figura 10: Valor do Grau de Evidência com variação linear e indiretamente proporcional a grandeza medida.....	25
Figura 11: Símbolo do NAP – Nó de Análise Paraconsistente.....	26
Figura 12: Símbolos da Célula Neural Artificial Paraconsistente.....	29
Figura 13: Rede de Análise Paraconsistente em Configuração Simples.....	30
Figura 14: Diagrama de blocos do SEvibrPAL2v.....	34
Figura 15(a): Vistas do Conjunto Mecânico.....	35
Figura 15(b): Fluxo de Aquisição de Dados.....	36
Figura 16(a): RPM / Tacômetro	36
Figura 16(b): Espectro obtido pelo CIS 2120	36
Figura 17: Pinagem Arduino Mega 2560.....	37
Figura 18: Pinagem Acelerômetro SparkFun MMA8452Q.....	38
Figura 19: Esquema eletrônico de montagem.....	39
Figura 20: Filtro de sinal LPA2V com 10 CNAPaps.....	40
Figura 21: Exemplo de saída do SEvibrPAL2v.....	46
Figura 22: Resultados gráficos dos sinais obtidos após a Transformada de Fourier.....	47
Figura 23: Resultados gráficos do Filtro de Sinal.....	48
Figura 24: Resultados da Normalização LPA2V.....	48
Figura 25(a): Espectro de operação normal.....	49
Figura 25(b): Padrão LPA2V de Operação Normal.....	49

Figura 26(a): Espectro de Falha de Desbalanceamento.....	49
Figura 26(b): Padrão LPA2V de Falha de Desbalanceamento.....	50
Figura 27(a): Espectro de falha de fixação.....	50
Figura 27(b): Padrão LPA2V de Falha de Fixação.....	50
Figura 28: Saída do SEvibrPAL2v, reportando Estado de Operação Normal.....	51
Figura 29: Saída do SEvibrPAL2v reportando Estado de Falha de Desbalanceamento.....	52
Figura 30: Saída do SEvibrPAL2v reportando Estado de Falha de Fixação.....	52

LISTA DE TABELAS

Tabela 1: Pinagem Acelerômetro SparkFun MMA8452Q	38
Tabela 2: Conjunto de Entrada do ParaExtract	42

LISTA DE ABREVIATURAS E SIGLAS

μ_{ER}	–	Grau de Evidência Resultante Real
Gc	–	Grau de Certeza
G _{CR}	–	Grau de Certeza Real
G _{ct}	–	Grau de Contradição
Gc	–	Grau de Certeza
LPA2V	–	Lógica Paraconsistente Anotada com anotação de 2 valores
LPA's	–	Lógicas Paraconsistentes Anotadas
NAP	–	Nó de Análise Paraconsistente
ParaExtract	–	Algoritmo Extrator de Efeitos de Contradição
LP	–	Lógica Paraconsistente
QUPC	–	Quadrado Unitário no Plano Cartesiano
RAP	–	Rede de análise paraconsistente
CNAP	–	Célula Neural Artificial Paraconsistente
CNAPap	–	Célula Neural Artificial Paraconsistente de Aprendizagem
V _{min}	–	Valor Mínimo contido em um conjunto
V _{max}	–	Valor Máximo contido em um conjunto
M _{max}	–	Máximo valor de Grau de Evidência
M _{min}	–	Mínimo valor de Grau de Evidência

LISTA DE SÍMBOLOS

\mathcal{T}	–	Reticulado de Hasse
\mathfrak{R}	–	Conjunto número reais
μ_E	–	Grau de evidência resultante
μ_{ER}	–	Grau de evidência resultante real
E_τ	–	Lógica Anotada Evidencial
φ	–	Intervalo de evidência
\mathcal{L}	–	Reticulado finito munido novo sistema de coordenadas
λ	–	Grau de evidência desfavorável
μ	–	Grau de evidência favorável
(μ, λ)	–	Anotação
V	–	Estado lógico verdadeiro
F	–	Estado lógico falso
\perp	–	Estado lógico paracompleto ou indeterminado
\top	–	Estado lógico inconsistente
D	–	Distância
F	–	Falso
P	–	Proposição P
\diamond	–	Operação do Nó de Análise Paraconsistente

SUMÁRIO

1.	INTRODUÇÃO	15
1.1	Importância.....	15
1.2	Problemática.....	16
1.3	Fundamentação teórica	16
1.3.1	Lógica Paraconsistente.....	16
1.3.2	Lógica Paraconsistente Anotada com anotação de dois valores (LPA2v) .	17
1.3.3	Relações algébricas entre o QUPC e o Reticulado da LPA2V	19
1.3.4	Graus de certeza e contradição.....	22
1.3.6	Nó de Análise Paraconsistente (NAP).....	26
1.3.7	Célula Neural Artificial Paraconsistente (CNAP).....	27
1.3.8	Célula Neural Artificial Paraconsistente de Aprendizado (CNAPap).....	28
1.3.10	Algoritmo Extrator de Efeito da Contradição (ParaExtract)	30
1.4	Justificativa	32
1.5	Objetivo.....	32
1.5.1	Objetivo geral	33
1.5.2	Objetivos específicos	33
2.	MATERIAIS E MÉTODOS.....	34
2.1	Aquisição de Dados	35
2.1.1	Calibração do Modelo de Aquisição de Dados	36
2.1.2	Microcontrolador Arduino Mega 2560	37
2.2	Acelerômetro SparkFun MMA8452Q	37
2.3	Esquema Eletrônico	38
2.4	TFD – Transformada de Fourier Discreta	39
2.5	Normalização.....	40
2.6	Filtro de Sinal Paraconsistente LPA2V	40
2.7	Treinamento e Aprendizado.....	41
2.7.1	Padronização LPA2V	41
2.7.2	Padrões	42
2.7.2.1	Padrão de Operação Normal.....	43
2.7.2.2	Padrão de Falha de Desbalanceamento.....	43
2.7.2.3	Padrão de Falha de Fixação.....	43
2.8	Monitoramento em Tempo Real	44
2.8.1	Padrão de Monitoramento	44
2.8.2	Análise LPA2V	44

2.8.3	Saída	46
3.	RESULTADOS E DISCUSSÃO	47
3.1	Resultados do Estágio de Treinamento e Aprendizagem	47
3.3	Resultados em Operação Normal	51
3.4	Resultados para Falha de Desbalanceamento	52
3.5	Resultados para Falha de Fixação	52
4.	CONCLUSÕES	53
4.1	Trabalhos futuros	53
	REFERÊNCIAS	55
	APÊNDICE A – Código Arduino Mega 2560	57
	APÊNDICE B – Código MatLab	59
	APÊNDICE C – Análise de Vibração do Conjunto Mecânico	71

1. INTRODUÇÃO

Em uma organização considera-se o conceito de Ativos como sendo os bens e direitos da empresa que podem ser convertidos em meios monetários. No caso de uma indústria de produção, além de seus bens estabelecidos como recurso financeiros, dinheiro em caixa, estoque, etc., esses Ativos são as máquinas, motores e demais equipamentos instalados na fábrica, essenciais para a produção. Seja qual for a indústria ou setor, o monitoramento das condições dos ativos é vital e vem sendo aprimorado com novas tecnologias e metodologias visando a previsão de falhas, otimização de tempo e custos relacionados a manutenções corretivas e preventivas. Vários fatores podem ser levados em consideração ao determinar a condição de um ativo, desde parâmetros elétricos, como consumo de energia e corrente, até parâmetros mecânicos, como vibração e térmicos como temperatura ambiente e do ativo. Com muitos dados, é necessário um modelo computacional que possa determinar o que e quando um determinado ativo irá falhar. Independentemente do setor, a confiabilidade é a capacidade de um ativo funcionar dentro dos requisitos de desempenho em um determinado período e condições de uso. É impossível eliminar completamente o tempo de inatividade, mas reduzi-lo é essencial para que a planta alcance uma operação cada vez mais lucrativa (GIANTOMASSI et al., 2015) (SONG et al., 2018).

1.1 Importância

Atualmente existem estudos e aplicações para detecção de falhas utilizando uma gama de aspectos como Análise Visual, Acústica, Elétrica e Térmica (HEMMATI et al., 2015) (WEIJTJENS et al., 2017).

No diagnóstico moderno de falhas de rolamentos de alta velocidade, métodos baseados em sinais de vibração são amplamente utilizados e o monitoramento *on-line* contínuo de máquinas rotativas é necessário para avaliar as condições de saúde em tempo real, reduzindo a possibilidade de tempo de inatividade (KWON et al., 2016) (CHEN et al., 2016) (INCE et al., 2016) (LEI & WU, 2020) (JANSSENS et al., 2016).

1.2 Problemática

Um sistema de monitoramento de vibração de rolamento deve ser preciso ao detectar o estado de operação do equipamento. O sistema deve ser capaz de coletar e analisar os dados corretamente e oferecer um diagnóstico eficiente. Também precisa ser capaz de evitar perdas e tempo de inatividade excessivo nos equipamentos de produção. Um diagnóstico incorreto causará substituição incorreta e/ou paralisação do equipamento ou até mesmo uma estimativa incorreta para uma manutenção causando custos desnecessários às plantas e empresas (WEIJTJENS et al., 2017) (ZHANG et al., 2017).

1.3 Fundamentação teórica

Nesta pesquisa foram desenvolvidos processos e técnicas de sensoriamento e monitoração de vibração em ativos onde os dados são tratados com algoritmos fundamentados em uma lógica não clássica, denominada de Lógica Paraconsistente - LP. Portanto, a seguir serão mostrados os principais fundamentos da LP, suas extensões na forma de anotações, principais equações e descrição dos algoritmos utilizados.

1.3.1 Lógica Paraconsistente

A lógica paraconsistente (LP) é uma lógica não clássica, proposicional e evidencial, cujos fundamentos se opõem a alguns princípios da lógica clássica binária. Estas características fundamentais da LP permitem equacionar sinais contraditórios sem enfraquecer as conclusões lógicas, tornando-a capaz de lidar com condições de incertezas (DA COSTA & ABE, 2000).

Existem diversas famílias de lógicas paraconsistentes e nesse trabalho utiliza-se uma extensão da LP denominada lógica paraconsistente anotada (LPA), que possui um reticulado associado onde anotações são representadas em seus vértices (DA SILVA FILHO et al., 2010).

O reticulado associado à LPA pode ser o de quatro vértices (Reticulado de Hasse), conforme mostrado na figura 1, onde uma proposição P pode ser analisada por graus de evidência, ou anotações (DE CARVALHO JR et al., 2021) (GARCIA et al., 2019) (ABE et al., 2018).

A figura 1 apresenta este tipo de representação da LPA com o reticulado associado e representações dos estados lógicos nos vértices.

1.3.2 Lógica Paraconsistente Anotada com anotação de dois valores (LPA2v)

Conforme descrito anteriormente, as lógicas Paraconsistentes Anotadas (LPA's) são classes de lógicas Paraconsistentes com reticulado associado τ , onde símbolos representativos de estados lógicos são considerados em seus vértices.

No reticulado associado a LPA2v, são atribuídos valores às suas fórmulas proposicionais correspondentes, onde as constantes notacionais representarão o conjunto de estados lógicos extremos $\{T, V, F, \perp\}$, correspondentes respectivamente a Inconsistente (T), Verdadeiro (V), Falso (F) e Indeterminado ou Paracompleto (\perp) (DE CARVALHO JR et al., 2021) (GARCIA et al., 2019) (ABE et al., 2018).

Quando se utiliza duas fontes de informação distintas para obtenção dos graus de evidência relacionados a uma única proposição P , onde uma fonte gera o grau de evidência favorável a proposição, cujo símbolo é μ e outra o grau de evidência desfavorável, ou contrária, a proposição, cujo símbolo é λ , diz-se que se trata de uma Lógica Paraconsistente Anotada com anotação de dois valores LPA2v.

A figura 1 mostra o Reticulado da LPA2v com os estados lógicos paraconsistentes extremos representados em seus vértices e as anotações compostas de graus de evidência.

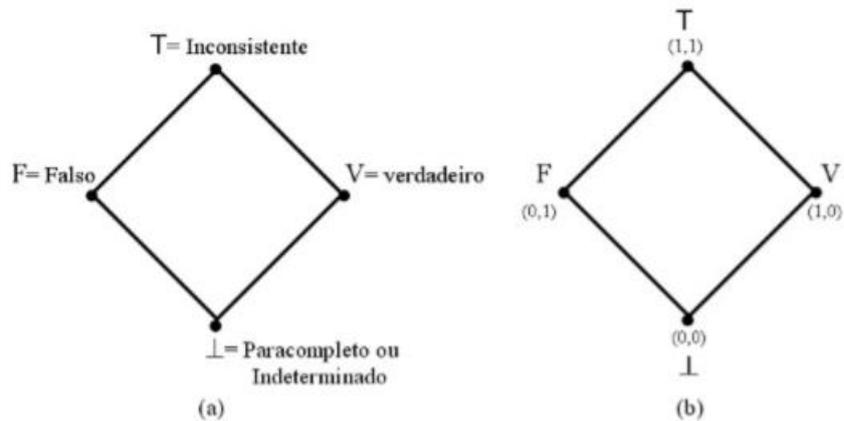


Figura 1. Reticulado de quatro vértices (Hasse)

Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

No reticulado de quatro vértices da LPA2v os estados lógicos podem ser utilizados como as saídas do algoritmo e neste caso são denominados estados lógicos resultantes. Portanto, no reticulado mostrado na Figura 2, tem-se:

- $P(\mu, \lambda) = P(1,0)$: indicando ‘existência de evidência favorável total e evidência desfavorável nula’, atribuindo uma conotação de Verdade à proposição;
- $P(\mu, \lambda) = P(0,1)$: indicando ‘existência de evidência favorável nula e evidência desfavorável total’, atribuindo uma conotação de Falsidade à proposição;
- $P(\mu, \lambda) = P(1,1)$: indicando ‘existência de evidência favorável total e evidência desfavorável total’ atribuindo uma conotação de Inconsistência à proposição;
- $P(\mu, \lambda) = P(0,0)$: indicando ‘existência de evidência favorável nula e evidência desfavorável nula’, atribuindo uma conotação de Indeterminação à proposição (DA SILVA FILHO, 2010).

A LPA2v, permite e admite, o tratamento de dados contraditórios, ambíguos e incompletos, com capacidade de conclusões e decisões em sistemas de análises (DA SILVA FILHO, 2010). Este modo de análise baseado em LPA2v tem se mostrado adequado quando se utiliza o conhecimento de evidências para ajudar a resolver conflitos, mesmo quando várias inferências inconsistentes ocorrem.

Neste trabalho serão utilizados algoritmos que utilizam equações interpretadas em um reticulado associado à LPA2v. As interpretações que permitiram a obtenção das equações serão mostradas a seguir.

1.3.3 Relações algébricas entre o QUPC e o Reticulado da LPA2V

A representação da anotação em LPA2v e a metodologia de interpretação do reticulado τ , permite o emprego da Lógica Paraconsistente para o tratamento de incertezas, com isso, são necessárias algumas interpretações algébricas, envolvendo um Quadrado no Plano Cartesiano (QUPC) e reticulado representativo da LPA2v (TORRES et al., 2011). Para tal, adota-se um sistema de coordenadas cartesianas para o plano, com isso, as anotações, da proposição dada, serão representadas por pontos do plano (DA SILVA FILHO, 2010). Uma primeira análise para se obter as equações da LPA2v inicia-se com a representação dos graus de evidência através de um quadrado unitário no plano cartesiano, conforme mostrado na figura 2.

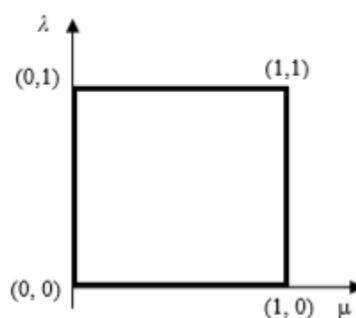


Figura 2. Reticulado τ com o sistema de coordenadas.

Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

Na Figura 3 identifica-se a anotação (μ, λ) onde, cada ponto de intersecção pode ser encontrado em outro sistema de coordenadas. Dessa forma, sistema pode ser fixado para τ , com isso, define-se a transformação entre QUPC e \mathcal{L} , implementando outro sistema de coordenadas no reticulado τ (TORRES et al., 2011).

Analogamente, como feito no QUPC, nesse reticulado \mathcal{L} , associa-se **T** a $(0, 1)$, **L** a $(0, -1)$, **F** a $(-1, 0)$ e **V** a $(1, 0)$. Desse modo, o reticulado \mathcal{L} será composto do seguinte sistema de coordenadas (DA SILVA FILHO, 2012), conforme exibido na Figura 3.

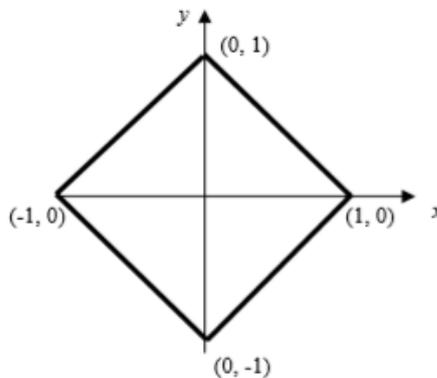


Figura 3. \mathcal{L} : Reticulado τ composto de um novo sistema de coordenadas
Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

As transformações entre QUPC e \mathcal{L} são definidas da seguinte forma onde \mathcal{L} pode ser obtido a partir de QUPC através de três fases: 1 – mudança de escala com aumento de $\sqrt{2}$, 2 – rotação 45 graus e; 3 – translação no eixo de origem (DA SILVA FILHO, 2012). Estas fases originam uma transformação linear, e seus passos são representados nas Figuras 4,5 e 6.

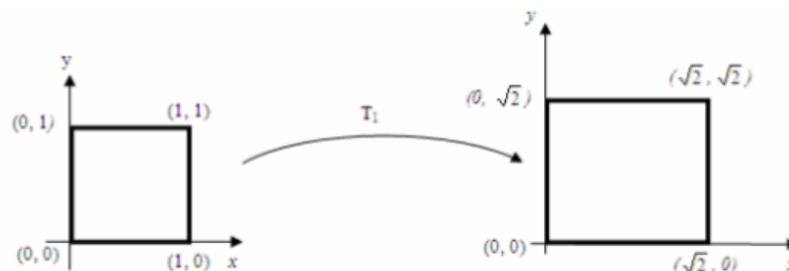


Figura 4. Aumento de escala de $\sqrt{2}$
Fonte: Torres et al. (2011).

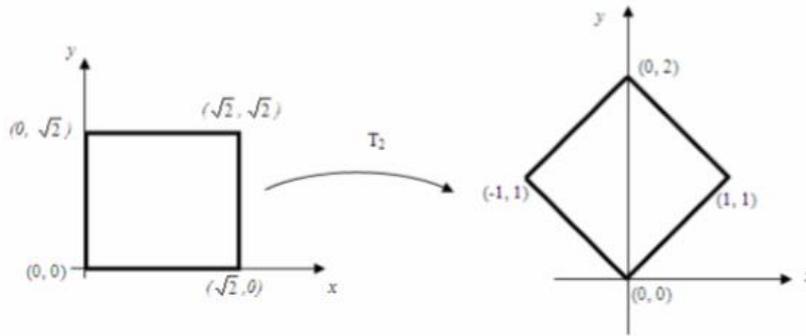


Figura 5. Rotação de 45° em relação a origem $\sqrt{2}$
 Fonte: Torres et al. (2011).

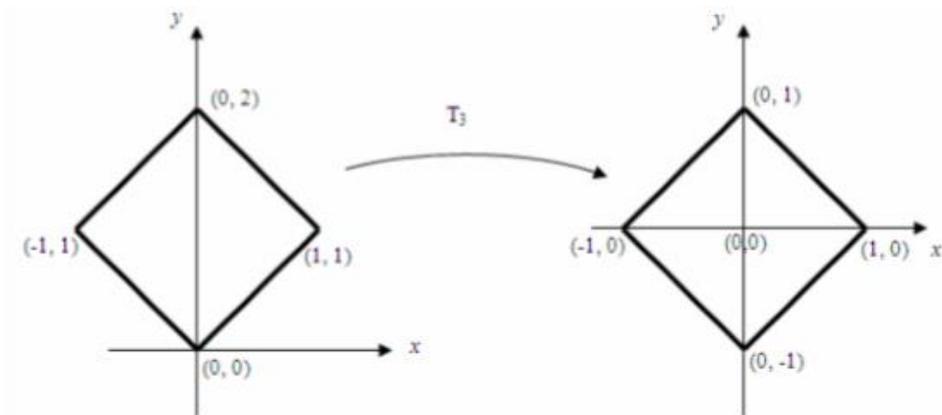


Figura 6. Translação de valores entre QUPC e o reticulado da LPA2v $\sqrt{2}$
 Fonte: Torres et al. (2011).

Através de transformações lineares no quadrado unitário no plano cartesiano e o Reticulado representativo da LPA, obtém-se a seguinte transformação (TORRES et al., 2011):

$$T(X,Y) = (x - y, x + y - 1) \quad (1)$$

A partir da equação 1 obtém-se os graus de certeza e de contradição conforme expostos a seguir.

1.3.4 Graus de certeza e contradição

Na transformação $T(X, Y)$ os componentes estão relacionados, conforme nomenclatura usual na LPA2v (DA SILVA FILHO, 2012), onde:

- $x = \mu \rightarrow$ Grau de evidência favorável;
- $y = \lambda \rightarrow$ Grau de evidência desfavorável, obtido:

Do primeiro termo, no par ordenado, da equação de transformação, determinando o Grau de certeza (G_c), com isso, é obtido pela seguinte equação:

$$G_c = \mu - \lambda \quad (2)$$

E seus valores, que pertencem ao conjunto \mathfrak{R} , variam no intervalo fechado +1 e -1, no eixo horizontal do reticulado, denominado “Eixo do Grau de Certeza”.

$G_c \rightarrow$ Resultando em +1, significa que o estado lógico, resultante da análise paraconsistente, é verdadeiro (V), quando resulta em -1 o estado lógico resultante da análise é falso (F).

O segundo termo obtido, no par ordenado da equação da transformação, é representado por: $x + y - 1 = \mu + \lambda - 1 \rightarrow$, determinando o grau de contradição G_{ct} , obtendo a seguinte equação:

$$G_{ct} = \mu + \lambda - 1 \quad (3)$$

E seus valores, que pertencem ao conjunto \mathfrak{R} , variam no intervalo fechado +1 e -1, no eixo horizontal do reticulado, denominado “Eixo do Grau de Contradição” (TORRES et al., 2011).

G_{ct} Resultando em +1, representa o estado lógico da análise paraconsistente é inconsistente (T).

G_c Resultando em -1 representa o estado lógico da análise paraconsistente é indeterminado (\perp).

Pode-se calcular o estado lógico paraconsistente, utilizando os graus de contradição (G_{ct}) e os graus de certeza (G_c), gerando o ponto (G_c, G_{ct}) (DA SILVA FILHO; ABE; LAMBERT-TORRES, 2008).

Os valores, quando plotados, no quadrado unitário do plano cartesiano, pode-se estender a análise em uma representação de 2 eixos. Os dois eixos são sobrepostos com o reticulado do LPA2v, onde pode-se delimitar regiões para comparações e determinação dos estados lógicos paraconsistente (DA SILVA FILHO, 2012). Com isso, a análise paraconsistente é interpretada, através de dados que são recebidos como grau de evidência desfavorável (descrença) representados por μ e grau de evidência favorável (crença) representado por λ , produzindo resultados denominados como grau de contradição e grau de certeza, subsidiando o sistema para a decisão de um estado lógico (TORRES et al., 2011).

A análise pode ser aprofundada com o Grau de Certeza Real G_{cr} que é obtido, como um valor projetado no eixo dos graus de certeza do reticulado, através das equações conforme mostrado geometricamente nas figuras 7 e 8 (DA SILVA FILHO, 2012).

Com base nos cálculos descritos é possível traçar uma reta com distância D até o limite V , para o caso de $G_C > 0$ e até o limite F , para o caso de $G_C < 0$. A interpolação em conjunto com a distância é apresentada na Figura representado no vértice direito do grau de certeza real (DA SILVA FILHO; ABE; LAMBERT-TORRES, 2008).

A distância é calculada, conforme equação 4. Após calculada a distância D , será obtido, através da projeção da reta D o grau de certeza, valor real, no eixo do grau de certeza, conforme Figuras 7 e 8.

$$D = \sqrt{(1 - |G_C|)^2 + G_{CT}^2} \quad (4)$$

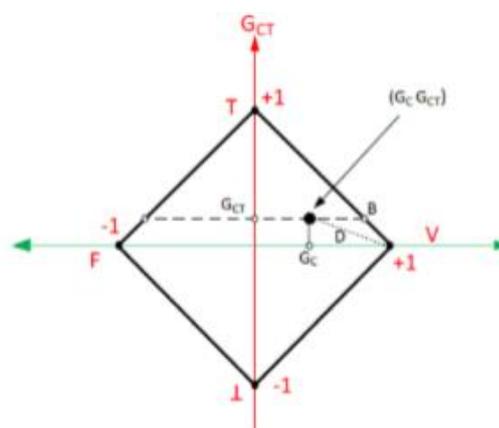


Figura 7. Ponto de Interpolação (G_c , G_{ct}) e distância D .
Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

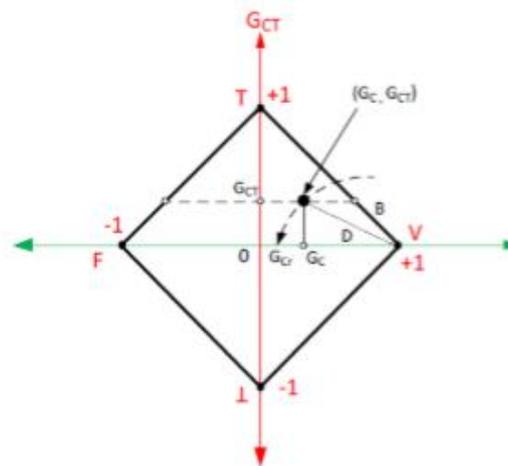


Figura 8. Grau de Certeza resultante real, Gcr no reticulado.

Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

$$\text{Se } G_c > 0: \quad G_{cr} = 1 - D \quad (5)$$

$$\text{Se } G_c < 0: \quad G_{cr} = D - 1 \quad (6)$$

Após a obtenção de G_{cr} , é possível, encontrar o valor normalizado, denominado como Grau de Evidência Resultante μ_{ER} (DA SILVA FILHO, 2012), portanto:

$$\mu_{ER} = \frac{G_{cr} + 1}{2} \quad (7)$$

1.3.5 Geração dos graus de evidência (Modelagem com variação Linear)

A extração do conhecimento é feita a partir da fonte de informação. Pode-se considerar que a variação das evidências de uma determinada proposição seja linear, de tal forma que a sua variação ocorra em um universo do discurso, compreendido, desde um valor do limite inferior de medição da grandeza, simbolizado por a_1 , até o valor do limite superior da grandeza, cujo símbolo, é a_2 . Dessa forma a evidência possui variação linear e proporcionalidade direta ao valor da grandeza (DA SILVA FILHO; ABE; LAMBERT-TORRES, 2008).

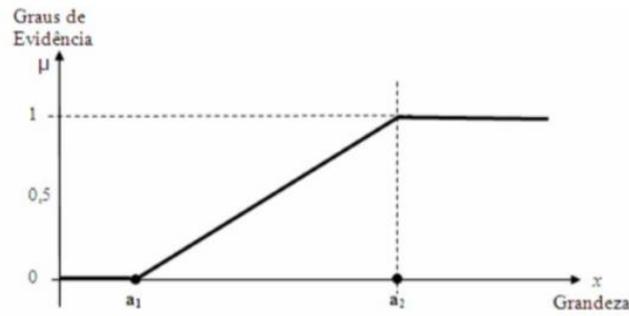


Figura 9. Valor do Grau de Evidência com variação linear e diretamente proporcional a grandeza medida

Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

Baseado no universo de discurso, caso a variação das evidências de determinada proposição for linear de modo indiretamente proporcional, a valoração dos Graus de Evidência de entrada será calculada conforme descrito a seguir.

No universo de discurso, compreendido entre o valor limite inferior da medição da grandeza simbolizado por a_1 e o valor do limite superior da medição da grandeza simbolizado por a_2 , na qual, a variação é linear e indiretamente proporcional ao valor da grandeza.

O gráfico com a variação do Grau de Evidência é apresentado, conforme a Figura 10.

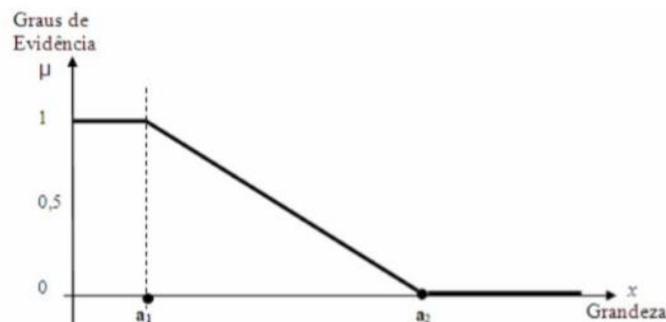


Figura 10. Valor do Grau de Evidência com variação linear e indiretamente proporcional a grandeza medida

Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

1.3.6 Nó de Análise Paraconsistente (NAP)

Das equações da LPA2v são obtidos algoritmos para tratamento e análise de sinais de informação, com aplicação em forma de Graus de Evidência. Os métodos aplicados no tratamento lógico paraconsistente poderão estar relacionados à análise efetuada por um algoritmo denominado de NAP (Figura 11) (DA SILVA FILHO et al., 2012), conforme detalhado a seguir.

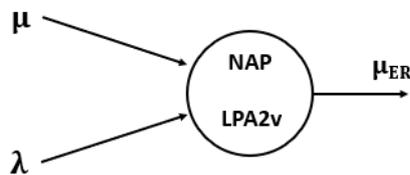


Figura 11. Símbolo do NAP – Nó de Análise Paraconsistente
Fonte: da Silva Filho et al. (2012).

a) valores de entrada:

- μ Grau de Evidência favorável $0 \leq \mu \leq 1$;
- λ Grau de Evidência desfavorável $0 \leq \lambda \leq 1$;

b) cálculo do grau de certeza:

$$Gc = \mu - \lambda \quad (8)$$

c) cálculo do grau de contradição:

$$Gct = \mu + \lambda - 1 \quad (9)$$

d) cálculo do Intervalo de Certeza $\phi(\pm)$:

$$\phi(\pm) = 1 - |Gct - 1| \quad (10)$$

e) cálculo da distância D:

$$D = \sqrt{(1 - |Gc|)^2 + Gct^2} \quad (11)$$

f) cálculo do Grau de Contradição normalizado:

$$\mu_{ctr} = \frac{\mu + \lambda}{2} \quad (12)$$

g) determinação do sinal de saída:

– se $s_e > 1$, então faça:

$$S1 = 0,5 \text{ e } S = \varphi_{E(\pm)} \quad (13)$$

– Indefinição, finaliza o algoritmo, caso contrário, ir para o próximo item;

h) determinação do grau de certeza real:

$$\text{Se } Gc > 0 \rightarrow Gcr = (1 - D)$$

$$\text{Se } Gc < 0 \rightarrow Gcr = (D - 1)$$

i) cálculo do grau de evidência resultante real:

$$\mu_{ER} = \frac{G_{CR} + 1}{2} \quad (14)$$

j) apresentação dos resultados de saída:

$$S1 = \mu_{ER} \text{ e } S = \varphi_{E(\pm)} \quad (15)$$

1.3.7 Célula Neural Artificial Paraconsistente (CNAP)

Uma Célula Neural Artificial Paraconsistente (CNAP) é a estrutura LPA2v capaz de, após apresentar um par de evidências favoráveis e desfavoráveis (μ , λ) em sua entrada, fornecer um resultado em sua saída, composto por um grau resultante de valor de evidência da análise (μE) e um valor de Grau de Contradição Normalizado (μCtr) (MARIO et al., 2021), (DE CARVALHO JR et al., 2018).

$$\mu E = \frac{(\mu - \lambda) + 1}{2}, \quad (16)$$

Onde, μE = Grau de Evidência de Saída.

$$\mu Ctr = \frac{(\mu + \lambda)}{2}, \quad (17)$$

Onde, μ_{Ctr} = Grau de Contradição Normalizado.

1.3.8 Célula Neural Artificial Paraconsistente de Aprendizagem (CNAPap)

A Célula Neural Artificial de Aprendizagem Paraconsistente (CNAPap) é basicamente uma CNAP comum tendo seus valores iniciais de entrada (definidos como 0,5 e sua saída (μ_{ER}) ligada às suas entradas de evidência desfavoráveis, posteriormente referenciadas como (DA SILVA FILHO et al., 2010).

$$\mu_E = \frac{(\mu_{1A} - \mu_{1BC}) + 1}{2} \quad (18)$$

$$\mu_{Ctr} = \frac{(\mu_{1A} + \mu_{1BC})}{2} \quad (19)$$

Onde $\mu_{1BC} = 1 - \mu_{1B}$.

Através do treinamento por iteração, que consiste em aplicar sucessivamente um padrão na entrada do sinal de grau de evidência favorável (μ) até que as contradições diminuam, e um grau de evidência resultante igual a um seja obtido como saída. No processo de aprendizagem, uma equação para os valores do grau de evidência resultante sucessiva, $\mu_E(k)$, é considerada até adquirir o valor de um. Portanto, para um valor inicial de $\mu_E(k)$, os valores de $\mu_E(k+1)$ são obtidos até $\mu_E(k+1) = 1$.

Considerando o processo de aprendizagem do padrão de verdade, a equação de aprendizagem é obtida através do cálculo da equação do grau de evidência resultante (DA SILVA FILHO et al., 2010):

$$\mu_E(k+1) = \frac{\{(\mu_{1A}) - (\mu_E(k)C * FL)\} + 1}{2}, \quad (20)$$

Onde FL é um valor Real, contido no intervalo [0, 1] que ajusta a velocidade de aprendizado da CNAPap.

A Figura 12 apresenta o símbolo de uma Célula Neural Artificial Paraconsistente de Aprendizagem (CNAPap).

a) Célula Neural Artificial Paraconsistente – (CNAP).

b) Célula Neural Artificial Paraconsistente de Aprendizado – (CNAPap).

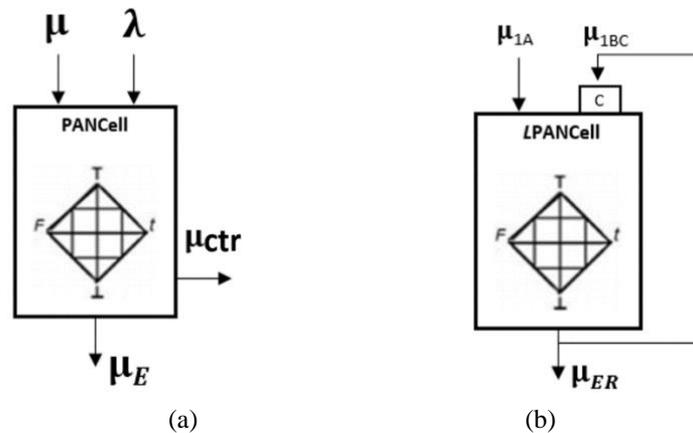


Figura 12: Símbolos da Célula Neural Artificial Paraconsistente
Fonte: (Da Silva Filho et al., 2010).

Neste estudo, um filtro de sinal foi construído, composto por 10 CNAPaps interconectadas em cascata, conforme será apresentado na seção Materiais e Métodos.

1.3.9 Rede de análise paraconsistente (RAP)

A rede RAP é composta por dois ou mais algoritmos NAPs interligados, de tal forma que cada nó execute a análise de uma única proposição. Com o objetivo de obter valores suficientes para a tomada de decisão sobre a proposição, o resultado da análise paraconsistente produzido em cada NAP é associado aos resultados dos outros NAP's (DA SILVA FILHO; ABE; LAMBERT-TORRES, 2008), (DA CRUZ, CLOVIS MISSENO, 2015).

As combinações dos NAPs em uma RAP devem considerar três circunstâncias:

1. As proposições analisadas em cada nó podem ser logicamente combinadas através de Graus de Certeza reais resultantes originando diferentes interligações;
2. Os valores dos Graus de Certeza reais resultantes e os Intervalos de Certeza Reais advindos dos NAPs de diferentes proposições poderão fazer uso de conectivos “E” e disjunção “OU” conforme desenho topológico da RAP e;
3. Os valores dos Graus de Certeza reais resultantes poderão ser

normalizados em valores entre 0 e 1 no intervalo real e assim considerados como Graus de Evidência usados na entrada de outras proposições que estão sendo analisados por diferentes NAPs (DA SILVA FILHO, 2012; DA SILVA FILHO; ABE; LAMBERT-TORRES, 2008).

Dessa forma, na configuração básica, a análise preliminar no NAP apresenta um intervalo de certeza sinalizado $\varphi(\pm)$ e o grau de certeza real G_{cr} fazem referência a uma única proposição, conseqüentemente, produzirão um único valor de evidência (DA SILVA FILHO, 2012; DA SILVA FILHO; ABE; LAMBERT-TORRES, 2008), conforme mostrado na Figura 13.

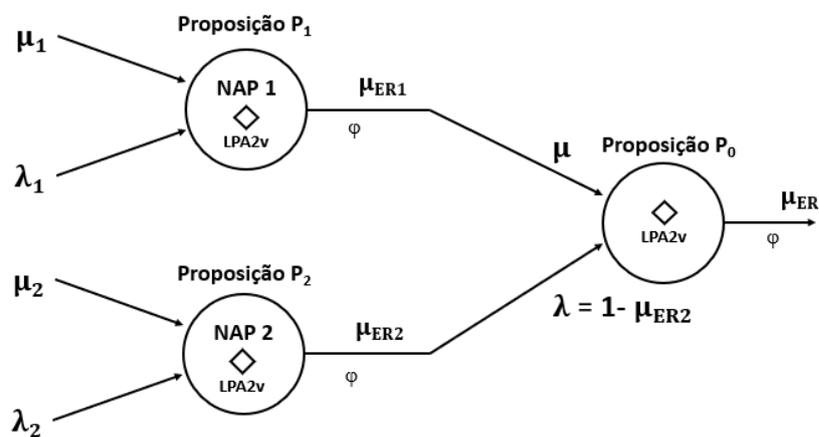


Figura 13. Rede de Análise Paraconsistente em Configuração Simples.
Fonte: da Silva Filho, Abe e Lambert-Torres (2008).

1.3.10 Algoritmo Extrator de Efeito da Contradição (ParaExtract)

O Algoritmo Extrator de Efeitos de Contradição (ParaExtract) é composto por conexões entre NAPs. Essa configuração forma uma Rede de Análise Paraconsistente capaz de extrair gradativamente os efeitos da contradição das informações provenientes do Banco de Dados de Conhecimento Incerto. A hipótese de extração dos efeitos da contradição tem como princípio que, se os primeiros sinais tratados forem os mais contraditórios, então o resultado da análise paraconsistente irá convergir para um valor consensual. Nesta operação típica, o ParaExtract recebe um conjunto de sinais de informação representados por graus de Evidência (μ_E) a respeito de determinada proposição P e, independentemente de outras informações externas, faz análises paraconsistentes em seus valores onde, gradativamente, vai

extraindo os efeitos da contradição permanecem como saída de um único Grau de Evidência Real (μ_{ER}) resultante (DA SILVA FILHO et al., 2010).

O μ_{ER} é o valor representativo do grupo de sinais de entrada após o processo de extração dos efeitos da contradição.

O Algoritmo é descrito conforme o procedimento:

1) Apresentar os N valores de Graus de Evidencia contidos no conjunto

$$G_{\mu} = (\mu_A, \mu_B, \mu_C, \dots, \mu_n) \quad */\text{Graus de Evidência } 0 \leq \mu \leq 1*/$$

2) Selecione o maior valor contido no conjunto de Graus de Evidência

$$\mu_{\max A} = \max (\mu_A, \mu_B, \mu_C, \dots, \mu_n)$$

3) Considere o maior valor do conjunto como sendo o Grau de Evidência Favorável

$$\mu_{\max A} = \mu_{\text{sel}}$$

4) Considere o menor valor do conjunto como sendo o Grau de Evidência Desfavorável

$$\mu_{\min A} = \min (\mu_A, \mu_B, \mu_C, \dots, \mu_n)$$

5) Transforme o menor valor do conjunto em Grau de Evidência Desfavorável

$$1 - \mu_{\min A} = \lambda_{\text{sel}}$$

6) Faça a análise paraconsistente entre os valores selecionados

$$\mu_{R1} = \mu_{\text{sel}} \diamond \lambda_{\text{sel}} \quad */ \text{Onde } \diamond \text{ é a ação paraconsistente do NAP } */$$

7) Adicione o valor obtido de μ_{R1} de volta ao conjunto, excluindo os valores μ_{\max} e μ_{\min} , selecionados anteriormente.

$$G_{\mu} = (\mu_A, \mu_B, \mu_C, \dots, \mu_n, \mu_{R1}) - (\mu_{\max A}, \mu_{\min A})$$

8) Retorne ao passo 2 até que o conjunto de valores tenha apenas um 1 elemento resultante

$$\text{Ir para passo 2 até } G_{\mu} = (\mu_{ER})$$

1.4 Justificativa

A crescente demanda do mercado por produtos industrializados com alto nível de qualidade e baixo impacto ambiental exige que as máquinas estejam funcionando em plenas condições de normalidade. Para isto programações de manutenção preventiva, preditivas e prescritivas são feitas para que reparos sejam preventivamente efetuados evitando paradas e danos que possam afetar a produção ou provocar acidentes que impactam o meio ambiente. Para alcançar níveis confiáveis a pesquisa de novas técnicas computacionais para a monitoração de equipamentos industriais tem se tornado atualmente um tema muito importante. Dessa forma entende-se que esforços devem ser feitos para se encontrar novas técnicas de monitoramento das condições dos ativos, visto que é vital a previsão de falhas, otimização de tempo e custos relacionados a manutenções corretivas e preventivas.

Na pratica uma monitoração eficiente pode ser complexa, porque a métrica da condição de um ativo não é claramente especificada, pois podem ser obtidas por parâmetros elétricos, como consumo de energia e corrente, bem como por parâmetros mecânicos, como vibração e térmicos que consideram temperatura ambiente e a própria temperatura do ativo. Dessa forma, os sistemas de monitoração devem ser aptos a trabalhar com informações representadas através de muitos sinais provenientes de banco de dados incertos. Isto torna sistemas convencionais baseados em lógica clássica binária, pouco eficientes e sujeitos a falhas computacionais e alto tempo de processamento, o que inviabiliza respostas em tempo real.

Com base nestas considerações, esta pesquisa, que consiste em encontrar novas técnicas de monitoração de vibração aplicando algoritmos baseados em lógica paraconsistente, se justifica, pois na extração de informações sobre estado de riscos de ativos, vários fatores podem ser levados em consideração configurando um conjunto de fontes evidenciais que geralmente trazem informações ambíguas, incompletas e inconsistentes.

1.5 Objetivo

Os objetivos deste estudo foram divididos entre Objetivo Geral e Específicos. Ambos descritos a seguir.

1.5.1 Objetivo geral

O objetivo deste estudo consiste em pesquisar, projetar e construir um dispositivo microprocessado tal que o *hardware* seja constituído de um robusto sistema de monitoramento, apresentando funcionamento baseado nos conceitos da Lógica Paraconsistente (LPA) e dedicado as análises da vibração em equipamentos industriais.

1.5.2 Objetivos específicos

Aplicar uma estrutura algorítmica baseada em Lógica Paraconsistente (LPA) funcionando como um sistema especialista (SEvibrPAL2v) capaz aprender padrões de assinaturas de vibração e, a partir desta estruturação criar um programa para que um sistema microprocessado possa classificar estados de riscos e monitorar continuamente rolamentos e demais componentes alertando a eminência de possíveis danos a um equipamento industrial.

2. MATERIAIS E MÉTODOS

O Sistema Especialista Paraconsistente para análises de vibração - SEvibrPAL2v foi desenvolvido com um conjunto de algoritmos paraconsistentes construindo unidades de análise interligando dois segmentos. A figura 4 mostra uma visão geral do SEvibrPAL2v através do diagrama em blocos, onde pode-se observar dois principais estágios: 1) Estágio de aprendizado de máquina e; 2) Estágio de monitoramento e classificação.

O primeiro segmento dedicado ao treinamento e aprendizado é composto por uma unidade de Aquisição de Dados, uma unidade de Modelagem de Dados LPA2V, uma unidade que aplica a análise LPA2V para criar assinaturas com padrões paraconsistentes e uma unidade que armazena esses padrões classificados em tipos de riscos para os ativos.

O segundo segmento SEvibrPAL2v que é dedicado ao monitoramento e classificação é composto por uma unidade que monitora as informações em tempo real, uma unidade que compara os valores capturados com as assinaturas armazenadas e a unidade de saída que apresenta os resultados de acordo com o risco de falha do ativo com base na vibração.

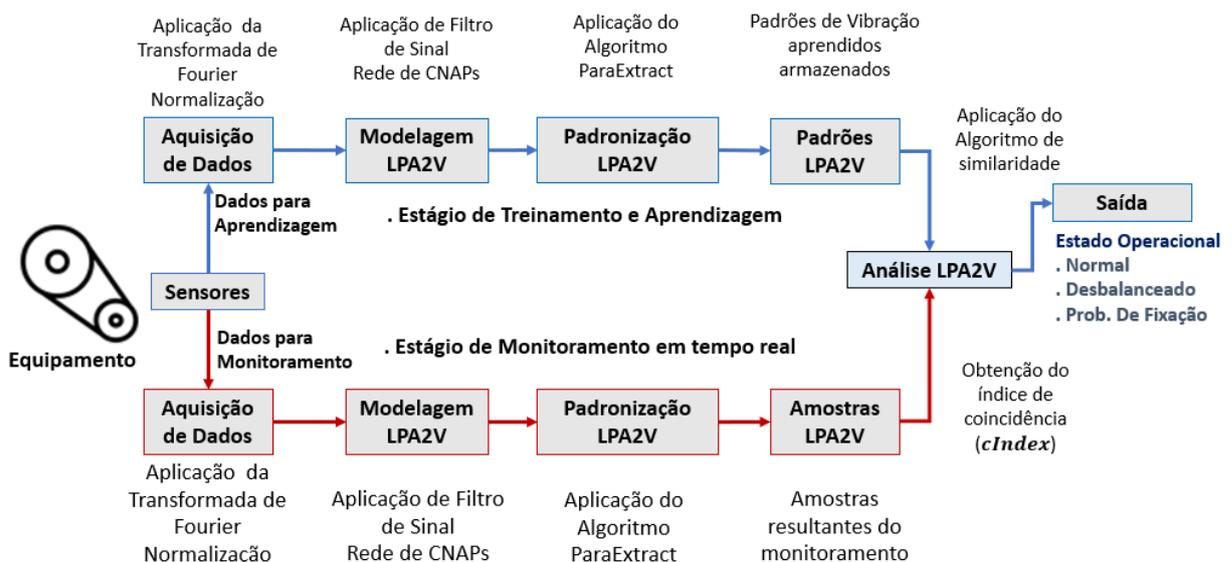


Figura 14. Diagrama de blocos do SEvibrPAL2v

Os blocos referentes a figura 14 serão descritos com detalhes, a seguir.

2.1 Aquisição de Dados

Para este experimento foi construído um conjunto mecânico equipado com motor 1/3 hp modelo Schultz JetMaster2, 1750 rpm, 0,25 kW a 220V; uma polia motriz de 110 mm de diâmetro; uma polia movida de 110 mm de diâmetro, unidas por uma correia tipo V de 300 mm de diâmetro; a polia movida foi montada em um rolamento de esferas modelo de SKF Explorer 6305-2Z/C3.

Este rolamento foi montado em um eixo em balanço de 130 mm de comprimento; um Acelerômetro triaxial Sparkfun modelo MMA8452Q foi instalado no eixo em balanço.

A Figura 15(a) mostra os detalhes dos equipamentos utilizados na aquisição de dados e geração de estudos de riscos causados por vibrações.

O acelerômetro Sparkfun foi conectado a um microcontrolador Arduino Mega 2560 equipado com microprocessador ATmega2560, através das interfaces SDA e SCL. Este monitorava e enviava as leituras de vibração para um computador onde um script de processamento de sinal escrito em MatLab lia e processava as leituras, bem como identificava possíveis distúrbios vibracionais, posteriormente persistindo os dados em mídia apropriada. Desta forma, todas as leituras de vibração dos dados foram enviadas para o MatLab usando o protocolo de comunicação serial (RS-232) a uma velocidade de 115200 bauds.

A Figura 15(b) mostra o fluxo de dados na etapa de aquisição de dados.

- a) Detalhes do equipamento.
- b) Fluxo de Aquisição de Dados.

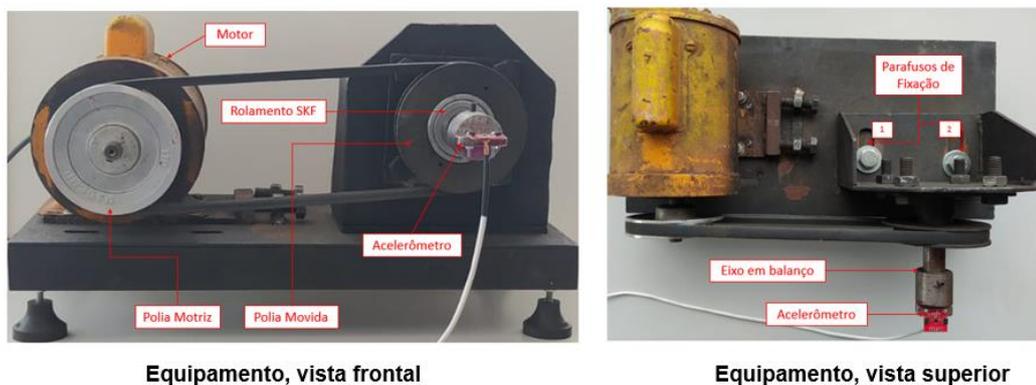


Figura 15(a), Vistas do Conjunto Mecânico

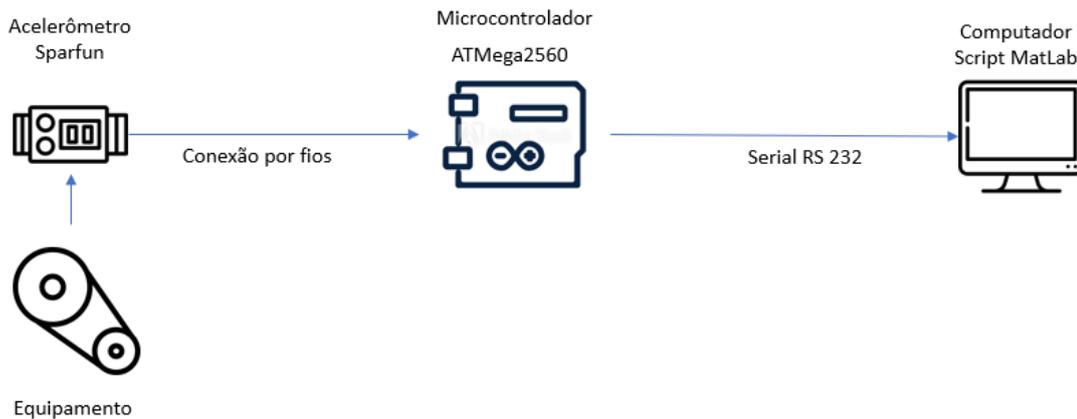


Figura 15(b), Fluxo de Aquisição de Dados

2.1.1 Calibração do Modelo de Aquisição de Dados

Com a finalidade de garantir a eficiência e eficácia do modelo construído e adotado para este estudo, foram utilizados equipamentos de medição e análise profissionais. Durante os ensaios realizados, para avaliar o RPM do motor JetMaster2 foi utilizado um tacômetro digital, marca Minipa, modelo MDT-2245C. Para analisar o espectro de vibração do conjunto, foi utilizado um analisador de vibração CIS, modelo 2120.

A rotação do motor foi aferida em 1785 RPM, medida diretamente no eixo do motor, junto a polia motriz. Figura 16(a).

O espectro obtido pelo analisador de vibração foi compatível com os espectros resultantes da aquisição de dados do SEvibrPAL2v. Figura 16(b).



Figura 16(a), RPM / Tacômetro

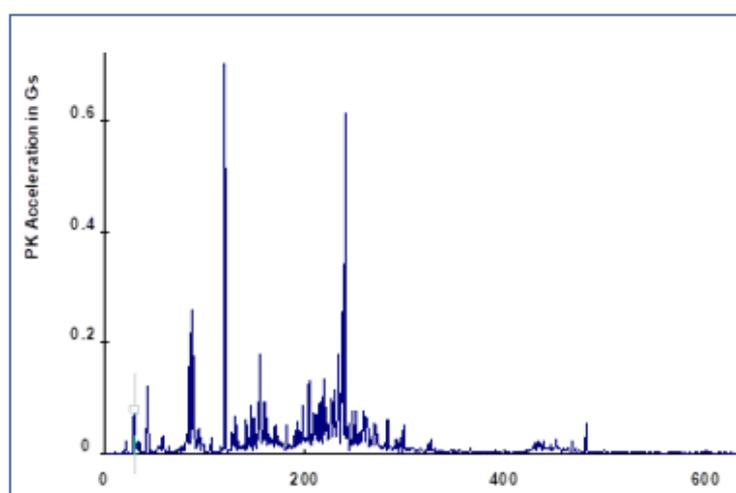


Figura 16(b), Espectro obtido pelo CIS 2120

Os ensaios de vibração completos são detalhados no Apêndice C – Análise de Vibração do Conjunto Mecânico.

2.1.2 Microcontrolador Arduino Mega 2560

O Arduino Mega 2560 é um microcontrolador de baixo custo e extremamente flexível, se adaptando à maioria das necessidades técnicas de diferentes projetos. Equipado com um processador ATmega2560 é comumente utilizado em projetos de automação. Sua versatilidade é decorrente da grande variedade e componentes que podem ser integrados à placa principal e pelo fato de seu firmware ser programado através da linguagem C++. O mesmo conta com pinos de entradas e saídas analógicas e digitais, figura 16



Figura 17: Pinagem Arduino Mega 2560

2.2 Acelerômetro SparkFun MMA8452Q

O MMA8452Q é um acelerômetro capacitivo de baixa potência, três eixos, com 12 bits de resolução. Este acelerômetro possui funções incorporadas com opções programáveis, e se comunica com os microcontroladores Arduino através do protocolo I²C (*Inter-Integrated Circuit*) e utilizando os pinos SDA (*Serial Data*) e SCL (*Serial Clock*), figura 17.

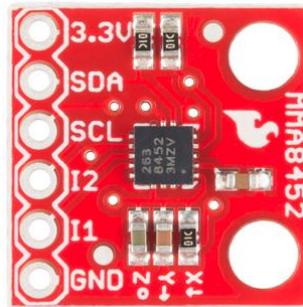


Figura 18: Pinagem Acelerômetro SparkFun MMA8452Q

A Tabela 1 apresenta a pinagem, bem como as principais função de cada pino.

Pino	Função	Input/Output	Observações
3.3V	Power Supply	Input	Entre 1.95 - 3.6V
SDA	I ² C Data Signal	Bi-directional	Bi-direcional. Tensão não deve exceder a alimentação (e.g. 3.3V).
SCL	I ² C Clock Signal	Input	Sinal de Clock. Tensão não deve exceder a alimentação (e.g. 3.3V).
I2	Interrupt 2	Output	Interrupção programável — Indica: data ready, orientation change, tap, entre outros
I1	Interrupt 1	Output	Interrupção programável — Indica: data ready, orientation change, tap, entre outros
GND	Ground	Input	0V

Tabela 1: Pinagem Acelerômetro SparkFun MMA8452Q

2.3 Esquema Eletrônico

O Acelerômetro Sparkfun MMA8452Q é acoplado ao microcontrolador Arduino Mega através dos pinos SDA (SerialData) e SCL (Clock), pela qual toda a informação é transmitida entre os dispositivos. Para tal, foi utilizado um “Level Shift” simples, com resistores de 330R para ambos os pinos. A alimentação do Acelerômetro SparkFun é fornecida pelos 3.3V do Arduino. A figura 18 detalha o esquema de montagem.

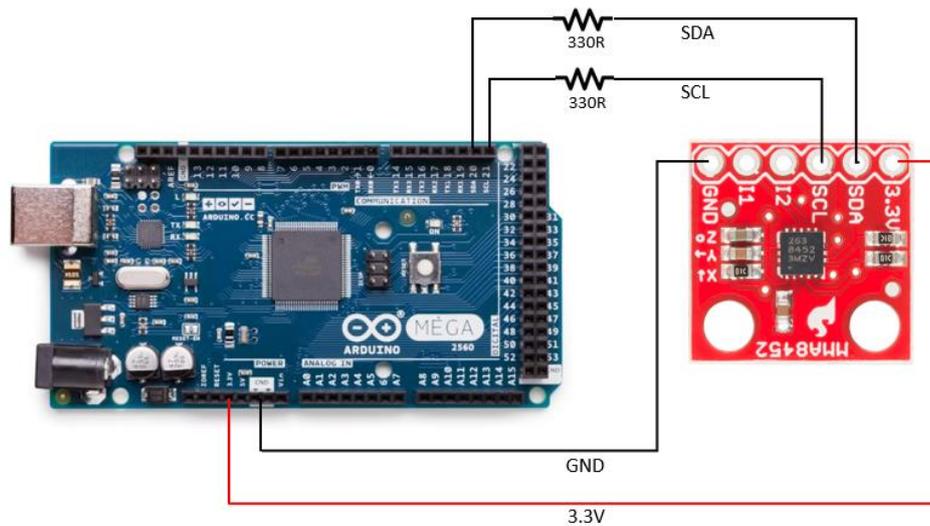


Figura 19: Esquema eletrônico de montagem

2.4 TFD – Transformada de Fourier Discreta

Todas as leituras de vibrações foram inicialmente processadas no domínio do tempo. Para monitorar falhas de frequência, o sistema precisou transformar todas as leituras para o domínio da frequência.

A Transformada de Fourier Discreta é uma função interna do Matlab, e seu resultado é a amplitude de aceleração/vibração em função da frequência. Isso permitiu a análise no domínio da frequência para obter uma compreensão mais profunda das leituras de vibração.

A função MatLab FFT(X) é dada pela equação:

$$X(j) = \frac{1}{n} \sum_{k=1}^n Y(k) W_n^{-(j-1)(k-1)}, \quad (21)$$

Onde: $W_n = e^{(-2\pi i)/n}$ é uma das N raízes.

Os resultados gráficos dos sinais obtidos após a aplicação da Transformada de Fourier são mostrados no capítulo de Resultados e Discussão.

2.5 Normalização

A LPA2V requer valores de entrada como Graus de Evidência e esses valores devem ser normalizados em valores infinitos entre Zero e Um (0 e 1). Dentro de um conjunto de valores de leitura, já transformados do domínio do tempo para a frequência, o sistema identifica os valores mínimo (Vmin) e máximo (Vmax), e estes são considerados ainda como 0 e 1 respectivamente. Portanto, uma vez recebidos, todos os dados foram normalizados como Graus de Evidência (Valores entre 0 e 1) através da equação de Normalização LPA2V:

$$N = \frac{(V - V_{\min})}{(V_{\max} - V_{\min})}, \quad (22)$$

Onde V é o valor lido dos sensores

Os resultados gráficos dos sinais obtidos pela Normalização são mostrados no capítulo de Resultados e Discussão.

2.6 Filtro de Sinal Paraconsistente LPA2V

Para beneficiar a análise, um filtro de sinal LPA2V foi construído utilizando Células Neurais Artificiais Paraconsistentes de Aprendizagem (CNAPap) para obter um espectro mais linear, isso foi especialmente benéfico ao criar e comparar padrões. Para isso, foi implementado um bloco de 10 CNAPap para realizar um filtro de sinal Paraconsistente em todas as leituras.

A Figura 19 mostra a configuração das CNAPap utilizadas como filtro de sinal neste trabalho.

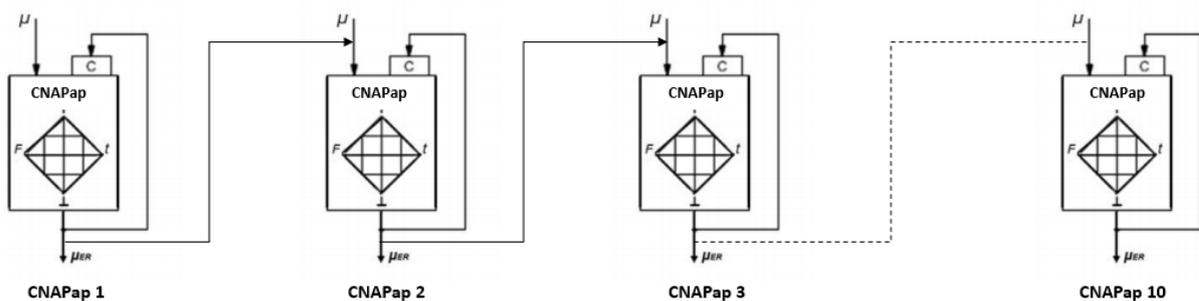


Figura 20: Filtro de sinal LPA2V com 10 CNAPap.

Equação utilizada pelo Filtro de Sinal LPA2V

$$\mathbf{M} = \frac{\mu_a - (1 - \mu) + 1}{2}, \quad (23)$$

Onde μ_a é o grau de evidência anterior e μ é o grau de evidência atual, ambos contidos no mesmo conjunto de valores.

Os resultados gráficos obtidos através do Filtro de Sinal são mostrados no capítulo de Resultados e Discussão.

2.7 Treinamento e Aprendizado

O treinamento e aprendizado inicia-se com uma padronização dos dados para que possam ser aplicados através dos algoritmos da LPA2v e uma modelação de padrões conforme os itens a seguir.

2.7.1 Padronização LPA2V

A padronização é o processo usado para criar um padrão único baseado em vários conjuntos de leituras semelhantes, mas não idênticos. Nesta implementação, após as leituras serem processadas, transformadas e normalizadas elas são submetidas ao algoritmo ParaExtract, e a saída é um padrão que representa a condição de vibração em um determinado momento do ciclo de vida do equipamento. Os 3 passos anteriores (DFT, Normalização e Filtro de sinal LPA2V) são repetidos 30 vezes e o resultado é uma matriz de 30 linhas por 750 colunas (Tabela 2). Esta matriz serve como entrada para ParaExtract onde cada coluna representa um grupo de estudo.

O padrão de saída é obtido após a iteração de todas as 30 linhas. A Tabela 2 exemplifica os dados e demonstra o procedimento executado pelo algoritmo ParaExtract.

	Leitura 1	Leitura 2	Leitura 3	Leitura 750
Linha 1	0.17	0.17	0.05	0.05	0.4	0.4
Linha 2	0.01	0.01	0.01	0.04	0.04	0.25
Linha 3	0.23	0.23	0.08	0.08	0.17	0.17
...	0.12	0.12	0.12	0.18	0.18	0.11
...	0.41	0.26	0.26	0.02	0.02	0
Linha 30	0.01	0.01	0.01	0.04	0.04	0.25

Padrão	X	Y	Z	W
--------	---	---	---	-----	-----	---

Obs: Segunda Coluna - valor min. = 0.01 e valor máx. = 0.41

Tabela 2: Conjunto de Entrada do ParaExtract

O algoritmo ParaExtract processa cada subconjunto descrito na tabela 2, e um único grau de evidência permanecerá para cada posição da linha. O resultado é uma segunda matriz de 3 linhas por 750 colunas. A linha 1 armazena o grau de evidência resultante para esse subconjunto. As linhas 2 e 3 armazenam os graus de evidência Min e Max de cada subconjunto. Os graus Min e Max são salvos e utilizados na fase de monitoramento. Quando este processo é finalizado, o padrão é considerado aprendido e persistido no sistema – os padrões são utilizados no segmento de monitoramento.

2.7.2 Padrões

Para distinguir entre possíveis falhas de vibração, o SEvibrPAL2v precisou aprender tais falhas antes da fase de monitoramento para que pudesse comparar com as leituras de vibração em tempo real. Para cada tipo de falha, um padrão único foi criado e persistido. Para este experimento, através de testes laboratoriais com estimulação de defeito nos mancais, foram classificados um total de 3 tipos de falhas.

Os padrões aprendidos usados neste trabalho são:

- A. Padrão de Operação Normal
- B. Padrão de falha de Fixação
- C. Padrão de falha de Desbalanceamento.

Os três tipos de falhas abordados neste estudo, bem como os métodos de obtenção serão descritos em detalhes a seguir.

2.7.2.1 Padrão de Operação Normal

Condição em que o equipamento está livre de problemas e considerado ótimo para operação normal. Um técnico de manutenção certifica as condições do equipamento. O padrão aprendido é descrito no item 3.2. Resultados de Classificação de padrões.

2.7.2.2 Padrão de Falha de Desbalanceamento

Condição em que o centro de massa do rotor não corresponde ao centro de rotação. Falhas de desbalanceamento podem ocorrer devido a defeitos de fabricação, por ex. rotores da bomba não balanceados durante a fabricação, bem como durante a operação. por exemplo, rotor de exaustão com muito material particulado; perda de material do rotor devido à erosão ou corrosão; e danos na hélice. O padrão aprendido é descrito no item 3.2. Resultados de Classificação de padrões.

Método:

Um corpo metálico foi fixado na borda da polia movida e então a vibração foi medida. Essa falha foi observada no espectro de vibração como um sinal agudo, com maior amplitude de vibração, comumente na frequência de rotação da máquina, expressa como 1X, denotando 1 vez a velocidade de rotação/rpm.

2.7.2.3 Padrão de Falha de Fixação

As falhas mecânicas de fixação são causadas por falta de aperto ou falta de torque adequado dos parafusos, porcas soltas, desgaste do eixo e dimensionamento incorreto. O padrão aprendido é descrito no item 3.2. Resultados de Classificação de padrões.

Método:

A porca de fixação 2, conforme indicado na figura 15(a), foi completamente

afrouxada; o motor foi iniciado com segurança e, em seguida, os sinais de vibração foram lidos. Amplitudes de vibração de 0,02g foram observadas em frequências inferiores a 30Hz, caracterizando frouxidão.

2.8 Monitoramento em Tempo Real

A principal premissa do SEvibrPAL2v é a de viabilizar não apenas o aprendizado de possíveis falhas utilizando a vibração como fonte de dados, mas também a de permitir que os padrões aprendidos possam ser comparados dinamicamente com leituras efetuadas em um determinado momento de operação do equipamento. Para tal o SEvibrPAL2v foi desenhado de forma que tais leituras pudessem ser adquiridas em tempo real, sem a necessidade de outros equipamentos de leitura.

2.8.1 Padrão de Monitoramento

O processo de padronização descrito durante a unidade de modelagem LPA2V consiste em 30 leituras subsequentes do dispositivo para que a amostra do padrão possa representar com mais precisão as condições do equipamento. Para o monitoramento, optou-se por coletar um conjunto de dados mais curto com 10 leituras subsequentes para que a amostra de processamento de sinal obtida seja o mais próximo possível do tempo real.

As etapas descritas para Transformação de Fourier, Normalização, Filtro de Sinal Paraconsistente LPA2V e Padronização são repetidas para cada conjunto de dados de monitoramento. O resultado é novamente um padrão de amostra que pode mudar ao longo do tempo chamado Padrão de Monitoramento.

2.8.2 Análise LPA2V

Esta unidade é crítica no sistema, onde o padrão de monitoramento é comparado aos padrões aprendidos usando técnicas LPA2V. Esta comparação é feita aplicando-se o Algoritmo de Similaridade LPA2V que resulta no Índice de Coincidência (**cIndex**) conforme mostrado abaixo.

Algoritmo de Similaridade LPA2V

- 1) Iterar o Padrão de Monitoramento (750 posições) e coletar o Grau de Evidência para cada posição.
- 2) Normalize o grau de Evidência com base nos valores Mín e Máx salvos obtidos da mesma posição, a partir dos padrões aprendidos:

Faça μ_{mon} = Grau de Evidência da posição N do Padrão de Monitoramento

Faça μ_{learn} = Grau de Evidência da posição N do Padrão Aprendido

Faça μ_{Min} = Grau mínimo da posição N do Padrão Aprendido

Faça μ_{Max} = Grau máximo da posição N do Padrão Aprendido

$$\mu_N = \frac{(\mu_{\text{mon}} - \mu_{\text{min}})}{(\mu_{\text{max}} - \mu_{\text{learn}})} \quad (24)$$

Onde μ_N é o Grau de Evidência Normalizado

- 3) Descarte o valor se for menor que 0 (zero) ou maior que 1 (um)

Se $\mu_N < 0$ ou $\mu_N > 1$, faça $\mu_N = 0$

- 4) Verificar a coincidência dos graus considerando uma tolerância entre os Padrão Aprendido e o de Monitoramento

Se $\mu_N = \mu_{\text{Learn}} \pm \text{Tolerancia}$, faça $\text{index_true} = \text{index_true} + 1$

Se não, faça $\text{index_false} = \text{index_false} + 1$

- 5) Consolidar os Resultados

Após iterar as 750 colunas, calcule o Índice de Coincidência do padrão de falha

$$cIndex = \frac{index_true}{index_true + Index_false} * 100 \quad (25)$$

- 6) Repita os passos de 1 a 5 para cada um dos padrões de falhas conhecidos
- 7) Apresente o resultado na saída

2.8.3 Saída

A saída SEvibrPAL2v foi baseada na Análise PAL entre o padrão de monitoramento e os padrões de falhas conhecidos. O resultado é um índice de coincidência, expresso em porcentagem.

A figura 20 mostra um exemplo da saída apresentada pelo SEvibrPAL2v com os índices de coincidência para cada falha conhecida neste estudo.

```

Command Window
SEvibrPAL2v Saída
=====
Índice de Operação Normal = 42.24 %
Índice de Falhas de Fixação = 96.14 %
Índice de Falhas de Desbalanceamento = 43.69 %

```

Figura 21: Exemplo de saída do SEvibrPAL2v

Com esta abordagem, o sistema foi capaz de identificar adequadamente a probabilidade de o padrão de monitoramento parecer um padrão de falha conhecido

3. RESULTADOS E DISCUSSÃO

Os resultados deste estudo foram divididos e organizados com base no desenho inicial do SEvibrPAL2v, observado no capítulo Materiais e Métodos, Figura 14: Resultados dos Estágios de Treinamento e Aprendizagem, e Resultados do Estágio de Monitoramento.

3.1 Resultados do Estágio de Treinamento e Aprendizagem

Os resultados das fases de Treinamento e Aprendizagem, compostos por: Transformada de Fourier, Filtro de Sinal, Normalização e Classificação de Padrões, são descritos abaixo:

3.1.1 Transformada de Fourier

Durante o processo de aprendizagem, a Transformada de Fourier se mostrou de fundamental importância na modelagem de dados do SEvibrPAL2v uma vez que o conjunto de dados coletados representam a aceleração e estavam dispostos no domínio do tempo (m/s^2). Para a classificação dos padrões aprendidos, se fez necessário transladar os dados para o domínio da frequência. A figura 21 exibe os resultados obtidos pelo SEvibrPAL2v na utilização da Transformada de Fourier Discreta.

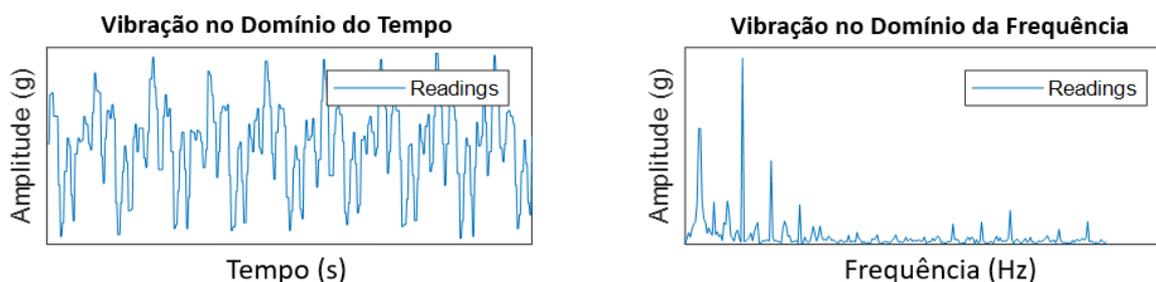


Figura 22: Resultados gráficos dos sinais obtidos após a Transformada de Fourier

3.1.2 Filtro de Sinal LPA2v

O Filtro de Sinal Paraconsistente implementado no SEvibrPAL2v obteve significantes resultados ao tratar o espectro de frequência, facilitando assim a

classificação dos padrões aprendidos. Conforme mostrado a figura 22, o resultado é um sinal mais linear que mantém os picos de frequência críticos.

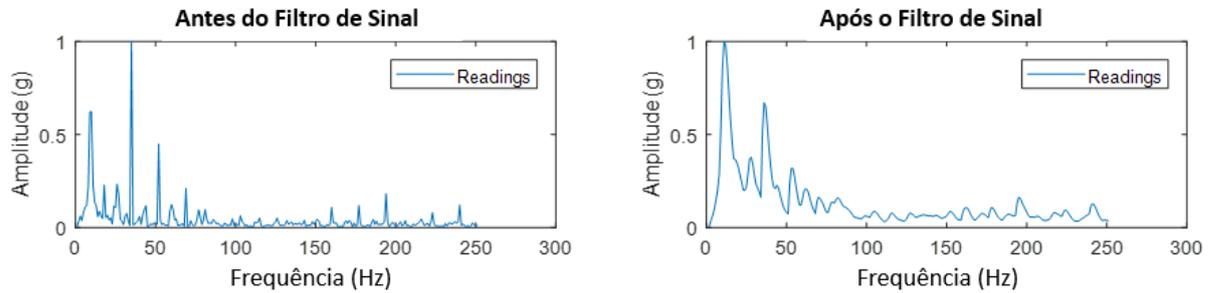


Figura 23: Resultados gráficos do Filtro de Sinal

3.1.3 Normalização LPA2v

Ainda no processo de Treinamento e Aprendizagem, a Normalização é o processo de entrada para toda e qualquer análise LPA2v. O SEvibrPAL2v foi capaz de realizar a normalização de forma eficiente. A figura 23 exibe o resultado da normalização do espectro de frequência.

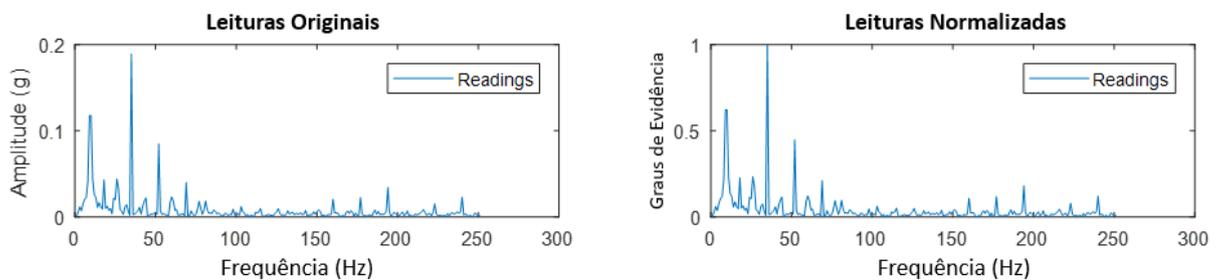


Figura 24: Resultados da Normalização LPA2V

3.1.4 Resultados de Classificação de Padrões

Durante o estágio de Treinamento e Aprendizagem, o SEvibrPAL2v foi capaz de aprender com grande eficiência, os padrões de falhas abordados neste estudo e posteriormente persistindo os dados para serem utilizados no estágio de monitoramento.

Os resultados relacionados aos padrões dos três tipos de falhas estão mostrados a seguir:

3.1.4.1 Padrão de Operação Normal

A Figura 24(a) mostra o espectro de Operação Normal e a Figura 24(b) exibe o gráfico Padrão de Operação Normal LPA2V.

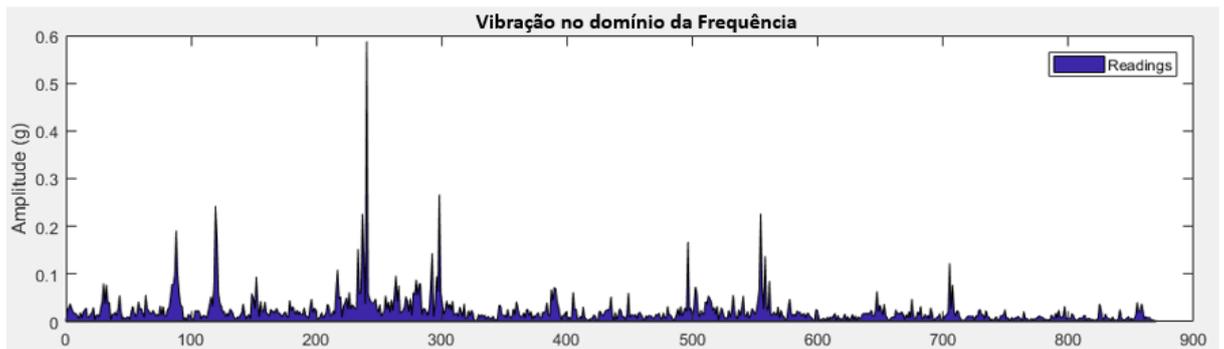


Figura 25 (a) - Espectro de operação normal

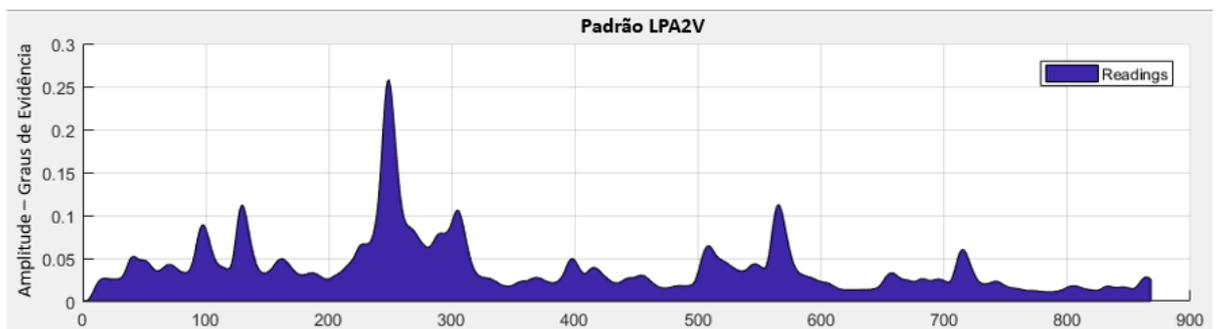


Figura 25 (b) - Padrão LPA2V de Operação Normal

3.1.4.2 Padrão de Falha de Desbalanceamento

A Figura 25(a) mostra o espectro de falha desbalanceada e a Figura 25(b) exibe o gráfico do Padrão LPA2V da falha de desbalanceamento.

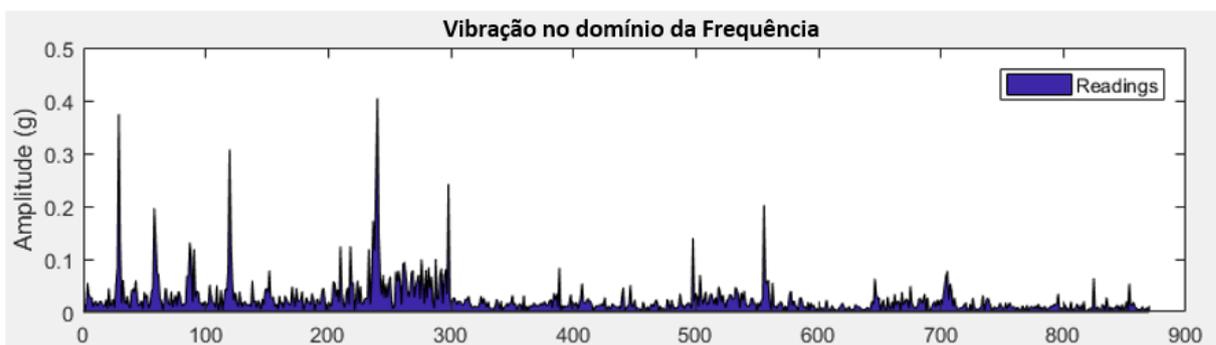


Figura 26 (a) - Espectro de Falha de Desbalanceamento

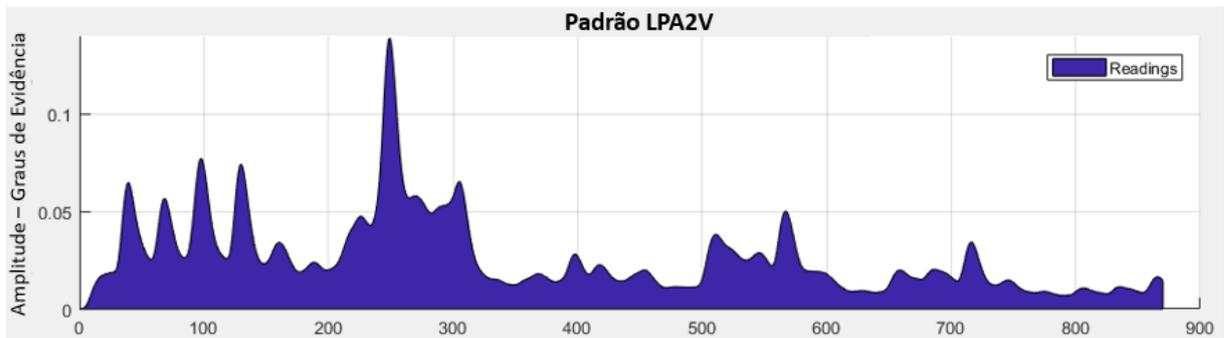


Figura 26 (b) – Padrão LPA2V de Falha de Desbalanceamento

3.1.4.3 Padrão de Falha de Fixação

A Figura 26(a) mostra o espectro de falha de Fixação e a Figura 26(b) exibe o gráfico do padrão LPA2V da falha de fixação.

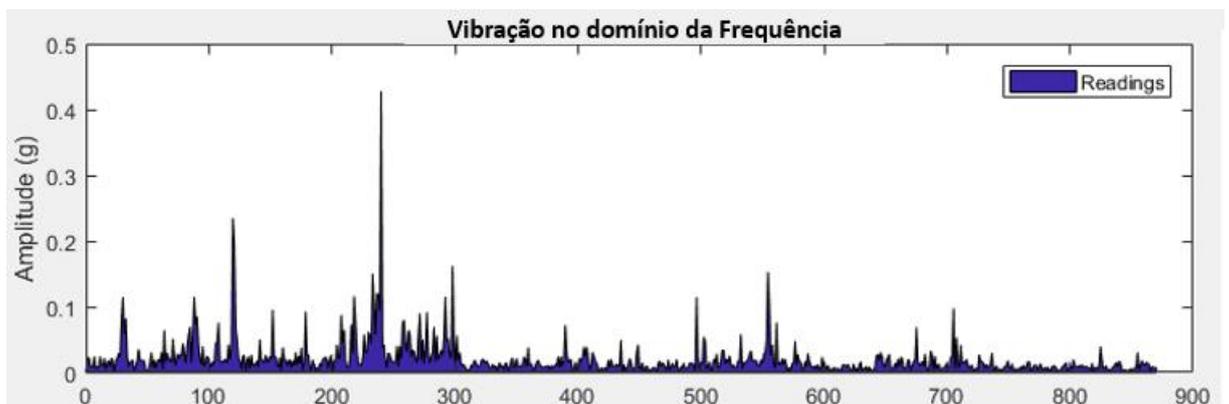


Figura 27 (a) - Espectro de falha de fixação

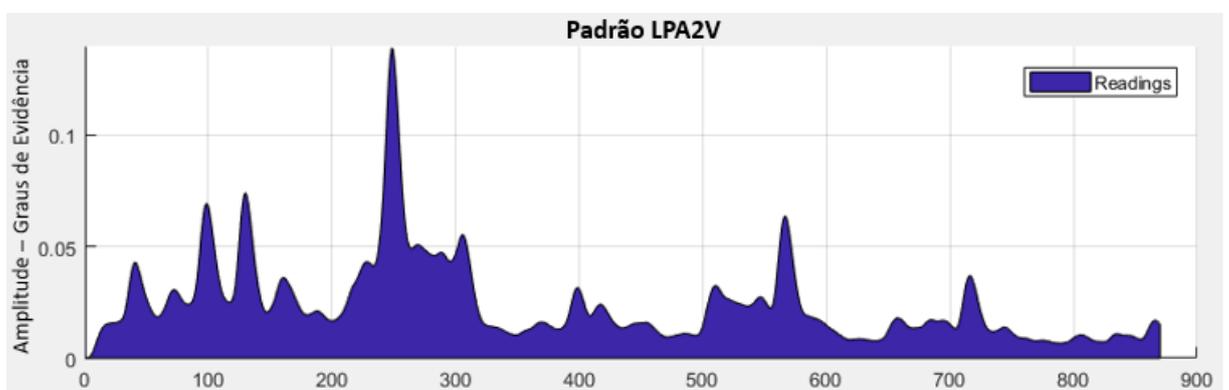


Figura 27 (b) - Padrão LPA2V de Falha de Fixação

3.2 Resultados do Estágio de Monitoramento

Os resultados deste estudo podem ser divididos pelos tipos de falhas conhecidos que o SEvibrPAL2v foi capaz de identificar e reportar adequadamente. Essa abordagem permitiu focar em cada índice de coincidência de padrões, aqui também considerado como índice de sucesso.

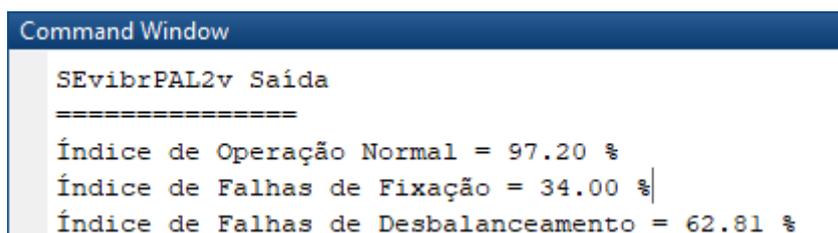
Como esperado, o SEvibrPAL2v conseguiu identificar corretamente quando o equipamento estava operando em condições normais, relatando um índice de coincidência maior de 97,2%. A identificação adequada das condições normais é tão importante quanto as condições de falha conhecidas. Embora o índice de Falha de Fixação também esteja acima de 62%, o índice de Operação Normal ainda é maior. (Ver Figura 14).

Com base nos padrões aprendidos, com foco situações específicas, o SEvibrPAL2v conseguiu identificar falhas de desbalanceamento com um índice de coincidência muito alto. O sistema alcançou um alto índice de coincidência de 84,42%. (Ver Figura 15). Usando a mesma abordagem para a condição Fixação, SEvibrPAL2v também obteve um índice de coincidência mais alto, SEvibrPAL2v foi capaz de identificar e relatar adequadamente um índice de 96,14% (ver a Figura 16).

3.3 Resultados em Operação Normal

Como parte deste estudo, o estado de Operação Normal é tão importante quanto todos os padrões de falhas conhecidos aprendidos pelo SEvibrPAL2v.

A Figura 27 mostra um resultado de análise em que o índice de coincidência foi de 97,2%, indicando que o equipamento se encontra em um estado de operacional Normal.



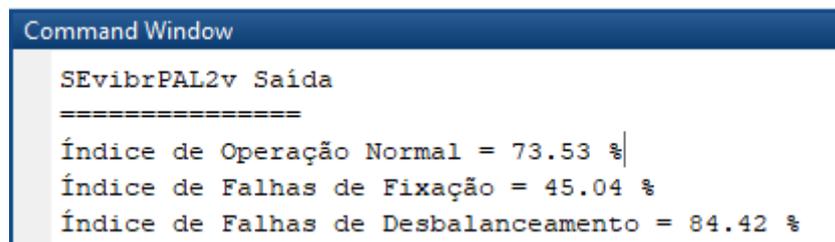
```
Command Window
SEvibrPAL2v Saída
=====
Índice de Operação Normal = 97.20 %
Índice de Falhas de Fixação = 34.00 %
Índice de Falhas de Desbalanceamento = 62.81 %
```

Figura 28: Saída do SEvibrPAL2v, reportando Estado de Operação Normal.

O índice de coincidência tem 97,2% de similaridade com "Estado de Operação Normal"; 34,0% de similaridade com "Estado de Falha de Fixação" e 62,81% de similaridade com "Estado de Falha de Desbalanceamento".

3.4 Resultados para Falha de Desbalanceamento

A Figura 28 exibe o resultado da análise, onde o índice de coincidência foi de 84,42% indicando uma Falha de Desbalanceamento.



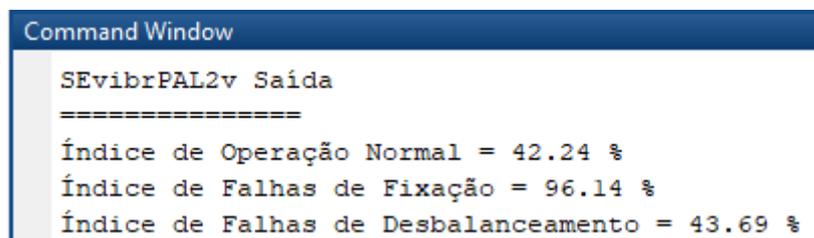
```
Command Window
SEvibrPAL2v Saída
=====
Índice de Operação Normal = 73.53 %
Índice de Falhas de Fixação = 45.04 %
Índice de Falhas de Desbalanceamento = 84.42 %
```

Figura 29: Saída do SEvibrPAL2v reportando Estado de Falha de Desbalanceamento.

O índice de coincidência tem 84,42% de similaridade com "Estado de Falha de Desbalanceamento", 73,53% de similaridade com "Estado de Operação Normal", e 45,04% de similaridade com "Estado de Falha de Desbalanceamento".

3.5 Resultados para Falha de Fixação

A Figura 29 exibe o resultado da análise em que o índice de coincidência foi de 97,2%, indicando uma Falha de Fixação.



```
Command Window
SEvibrPAL2v Saída
=====
Índice de Operação Normal = 42.24 %
Índice de Falhas de Fixação = 96.14 %
Índice de Falhas de Desbalanceamento = 43.69 %
```

Figura 30: Saída do SEvibrPAL2v reportando Estado de Falha de Fixação.

O índice de coincidência tem 96,14% de similaridade com "Estado de Falhas de Fixação", 42,24% de similaridade com "Estado de Operação Normal" e 43,69% de similaridade com "Estado de Falhas de Desbalanceamento".

4. CONCLUSÕES

O principal objetivo deste estudo foi aplicar a metodologia da Lógica Anotada Paraconsistente com Anotação de dois Valores (LPA2V), juntamente com os conceitos de Inteligência Artificial, em Análise Vibracional, construindo condições de obter sinais de evidências em tempo real e comparando com padrões de falhas conhecidos a partir de assinaturas de vibração. Ao se avançar no encontro deste objetivo verificou-se que a metodologia da Lógica Paraconsistente Anotada com Anotação de dois valores (LPA2V) é capaz de processar assinaturas de vibração, construir padrões e realizar comparação de padrões com alta taxa de sucesso. Os resultados comprovaram que o Sistema Especialista Paraconsistente (SEvibrPAL2v) foi capaz de aprender, analisar e identificar de forma autônoma e em tempo real, as falhas mecânicas propostas neste estudo e apresentou resultados satisfatórios com suporte mecânico. Também as técnicas baseadas em microcontrolador Arduino® Mega e o acelerômetro Sparkfun® MMA8452Q, se mostraram eficientes ao coletar e processar dados oriundos do equipamento em estudo. O software Matlab® demonstrou processamento e velocidade compatíveis com a proposta de monitoramento em tempo real. O Conjunto mecânico construído para a análise, atingiu a compatibilidade com os equipamentos industriais propostos neste estudo. Como em qualquer sistema de Inteligência Artificial, algumas preocupações foram consideradas durante este estudo, como viés, transparência, confiança e explicabilidade, e dentro deste contexto, verificou-se o SEvibrPAL2v totalmente compatível com tais conceitos devido ao código simples e claro e à metodologia usada durante seu desenvolvimento.

4.1 Trabalhos futuros

Este estudo focou em aplicar a Lógica Paraconsistente Anotada com Anotação de 2 valores (LPA2V) na coleta de informações, processamento e comparação de tipos de falhas preestabelecidos pelo seu próprio estágio de aprendizado.

Os resultados obtidos comprovaram a eficácia do modelo e a capacidade futura em analisar outros tipos de falhas oriundas de bases de dados públicas e/ou compartilhadas, formando uma rede de conhecimento estruturados pela LPA2V.

Neste estudo, o SEvibrPAL2v utilizou a aceleração (G) como parâmetro para monitoramento da condição do ativo. Trabalhos futuros devem abordar parâmetros de velocidade RMS (Root Mean Square) como fonte de informação afim de explorar outros tipos de falhas. Para tal, dois métodos podem ser utilizados:

- 1) Através das técnicas e metodologias da Lógica Paraconsistente Anotada com Anotação de 2 valores (LPA2V), obter e processar os parâmetros da velocidade a partir da aceleração coletados por meio de acelerômetro e calculando a velocidade RMS posteriormente.
- 2) Obter os informação diretamente do equipamento utilizando sensores de velocidade, calculando a velocidade RMS posteriormente.

REFERÊNCIAS

- Abe, J. M., Akama, S., Nakamatsu, K., Da Silva Filho, J.I. (2018). Some Aspects on Complementarity and Heterodoxy in Non-Classical Logics. *Procedia Computer Science*, 126, pp. 1253–1260. <https://doi.org/10.1016/j.procs.2018.08.068>
- Chen, J., Pan J., Li Z., Zi, Y, Chen, X. (2016). Generator bearing fault diagnosis for wind turbine via empirical wavelet transform using measured vibration signals. *Renewable Energy* VL.89. DOI: 10.1016/j.renene.2015.12.010
- Da Costa N. C.A., Abe J. M. (2000). Paraconsistência em informática e inteligência artificial, *Ciência • Estud.* av. 14 (39) • <https://doi.org/10.1590/S0103-40142000000200012>
- Da Silva Filho, J. I., Lambert-Torres, G., Abe, J. M. (2010). Uncertainty Treatment Using Paraconsistent Logic - Introducing Paraconsistent Artificial Neural Networks. *IOS Press*, 328, 211, *Frontiers in Artificial Intelligence and Applications*, Amsterdam, Netherlands,
- Da Cruz, Clovis Misseno (2015). *Aplicação da Rede Neural Artificial Paraconsistente em Controle Estatístico de Processo, Mestrado em Engenharia Mecânica.*
- De Carvalho, A., Justo, J.F., Angelico, B.A., De Oliveira, A.M., Da Silva Filho, J.I. (2021) Rotary Inverted Pendulum Identification for Control by Paraconsistent Neural Network. *IEEE Access*, 9, pp. 74155–74167, 9430548. doi: 10.1109/ACCESS.2021.3080176.
- De Carvalho, J. I. Da Silva Filho, M. C. Mario, M. F. Blos and C. M. da Cruz, "A Study of Paraconsistent Artificial Neural Cell of Learning Applied as PAL2v Filter," in *IEEE Latin America Transactions*, vol. 16, no. 1, pp. 202-209, Jan. 2018, doi: 10.1109/TLA.2018.8291474.
- Garcia, D. V., Da Silva Filho, J. I., Silveira Jr., L., Pacheco, M. T. T., Abe, J. M., Carvalho Jr., A., Blos, M. F., Pasqualucci, C. A. G. & Mario, M.C. (2019). Analysis of Raman spectroscopy data with algorithms based on paraconsistent logic for characterization of skin cancer lesions. *Vibrational Spectroscopy*, 103, <https://doi.org/10.1016/j.vibspec.2019.102929>.
- Giantomassi A., Ferracuti F., Iarlori S., Ippoliti G. and Longhi S., (2015). Electric Motor Fault Detection and Diagnosis by Kernel Density Estimation and Kullback–Leibler Divergence Based on Stator Current Measurements, in *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1770-1780, doi: 10.1109/TIE.2014.2370936.
- Hemmati, F., Orfali W. and Gadala, M. S. (2015). Roller bearing acoustic signature extraction by wavelet packet transform, applications in fault detection and size estimation. *Applied Acoustics*. pp. 101 – 118. doi.org/10.1016/j.apacoust.2015.11.003
- Ince T., Kiranyaz S., Eren L., Askar M. and Gabbouj M., (2016). Real-Time Motor Fault Detection by 1-D Convolutional Neural Networks, in *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7067-7075, doi: 10.1109/TIE.2016.2582729.

Janssens O., Slavkovikj V., Vervisch B. , Stockman K. , Loccufier M., Verstockt S., Van de Walle R. and Van Hoecke S. (2016). Convolutional neural network based fault detection for rotating machinery. *Journal of Sound and Vibration*. 377. P.331-345.

Kwon D., Hodkiewicz M. R., Fan J., T. Shibutani and M. G. Pecht, (2016). IoT-Based Prognostics and Systems Health Management for Industrial Applications. In *IEEE Access*, vol. 4, pp. 3659-3670doi: 10.1109/ACCESS.2016.2587754.

Lei, X., Wu, Y. (2020). Research on mechanical vibration monitoring based on wireless sensor network and sparse Bayes. *J Wireless Com Network* 2020, 225 <https://doi.org/10.1186/s13638-020-01836-9>

Mario, M. C., Garcia, D. V., Da Silva Filho, J. I., Silveira Júnior, L., Barbuy, H. S. (2021). Characterization and classification of numerical data patterns using Annotated Paraconsistent Logic and the effect of contradiction. *Research, Society and Development*, [S. l.], v. 10, n. 13, p. e283101320830, DOI: 10.33448/rsd-v10i13.20830. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/20830>.

Song L., Wang H. and Chen P. (2018). Vibration-Based Intelligent Fault Diagnosis for Roller Bearings in Low-Speed Rotating Machinery, in *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 8, pp. 1887-1899, doi: 10.1109/TIM.2018.2806984.

Weijtjens, W., Verbelen, T., Capello, E., & Devriendt, C. (2017). Vibration based structural health monitoring of the substructures of five offshore wind turbines. In *Procedia Engineering: X International Conference on Structural Dynamics, EURO DYN 2017* (Vol. 199, pp. 2294-2299). (Procedia Engineering). <https://doi.org/10.1016/j.proeng.2017.09.187>

Zhang, M., Feng, K., & Jiang, Z. (2017). Research on variational mode decomposition in rolling bearings fault diagnosis of the multistage centrifugal pump. *Mechanical Systems and Signal Processing*, 93(1), 460-493. <https://doi.org/10.1016/j.ymssp.2017.02.013>

APÊNDICE A – Código Arduino Mega 2560

```

#include <Wire.h>
// Must include Wire library for I2C

#include "SparkFun_MMA8452Q.h"
// to get the library: http://librarymanager/All#SparkFun_MMA8452Q

const int BUFFER = 500 * 3; // Buffer size
float waveArray[BUFFER]; // Array to store waveform

MMA8452Q accel; // create instance of the MMA8452 class

int read = 1;
int count = 0;

int matlabData;
unsigned long now, prev = 0;
float t = 0;
int fs = 0;

void enviarFs()
{
  Serial.print(fs);
  Serial.print('\n');
}

void enviarOnda()
{
  if (accel.available()) // Wait for new data from accelerometer
  {
    prev = millis();

    while (count < BUFFER )
    {
      waveArray[count] = accel.getCalculatedZ();
      count++;
    }

    now = millis();

    t = ((now - prev) / 1000.0) / BUFFER ;
    fs = 1 / t;
    count = 0;

    for (int i = 0; i < BUFFER; i++)
    {
      Serial.print(" ");
      Serial.print(waveArray[i]);
      waveArray[i] = 0;
    }

    Serial.print('\n');
  }
}

```

```
void setup() {
  Serial.begin(115200);
  Wire.begin();

  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite(LED_BUILTIN, LOW);

  if (accel.begin() == false) {
    Serial.println("Not Connected. Please check connections and read the hookup guide.");
    while (1);
  }
}

void loop() {
  while (Serial.available() > 0)
    // Adds input char to the string "input" while there still is characters to read
    {
      digitalWrite(LED_BUILTIN, HIGH);
      matlabData = Serial.read(); // read data

      if (matlabData == 'W')
      {
        enviarOnda();
      }

      if (matlabData == 'F')
      {
        enviarFs();
      }
    }
}
```

APÊNDICE B – Código MatLab

```

% Menu Principal
% main_menu.m
% Escrito por Marcio Correa e Dra. Dora Vilanova
% 2022

clear;
clc;

disp('+-----+');
disp('|           |');
disp('|         M E N U         |');
disp('|           |');
disp('|  1. Aprender Padrão    |');
disp('|  2. Plotar Padrão     |');
disp('|  3. Monitorar         |');
disp('|  4. Sair              |');
disp('|           |');
disp('+-----+');
disp(' ');
op=input('Dê sua opção: ');
if op==1
    Aprendizado2
elseif op==2
    plot_padrao
elseif op==3
    Monitorar4
elseif op==4
    clc;
    disp('Encerrando o programa');
    priorPorts=instrfind;
    delete(priorPorts);
    pause (1);
    close;

else
    disp('Opção errada....');
    pause (1);
    main_menu
end

% Codigo de Aprendizado
%Aprendizado2.m
% Escrito por Marcio Correa e Dra. Dora Vilanova
% 2022

disp("Realizando Leituras - COM");

priorPorts=instrfind;
delete(priorPorts);

se=serial('COM3','BaudRate',115200);
se.InputBufferSize = 4096 * 3;

disp("Abrindo Porta Serial");
fopen(se);

```

```

pause(2);

disp("Porta Serial Aberta");

% apagar o padrao anterior
temp = xlsread('Dados.xlsx');
if ~isempty(temp)
    xlswrite('Dados.xlsx', zeros(size(temp))*nan)
end

amostras=30;

dados = zeros (751,amostras+1);

for cp=1:amostras

    disp("Enviando Comando de Aprendizagem: " + cp);
    fprintf(se, 'W');

    disp("Aguardando Processamento");
    pause(1);

    % Le os dados da serial e armazena na matriz input
    input = str2double (strsplit(fscanf(se, '%c'), ' '));

    % salvar_amostras(cp,:) = input;

    % Remove a coluna 1 pois é sempre nula.
    input(1) = [];

    amostras_salvas(cp,:) = input;

    % disp("Frequencia de Amostragem");
    fprintf(se, 'F');

    % pause(1);
    Fs = fscanf(se, '%d');

    N = length(input);

    %Determine FFT and Plot
    tic
    freq = 0:Fs/length(input):Fs/2; %frequency array for FFT

    xdft = fft(input); %Compute FFT
    xdft = 1/length(input).*xdft; %Normalize
    xdft(2:end-1) = 2*xdft(2:end-1);

    % obtemos apenas o espectro, sem as frequencias
    spec = abs(xdft(1:floor(N/2)+1));

    dados(:,cp+1) = spec'; % transpose line to column;

```

```

end

csvwrite('amostras_salvas.csv', amostras_salvas);

dados(:,1) = freq'; % transpose line to column;

xlswrite('dados.xlsx', dados);

%plot(freq, spec, 'r');

spec(1) = 0;
area(freq, spec);
title('Vibration in Frequency-Domain');
legend('Readings');
xlabel('Frequency (Hz)');
ylabel('Amplitude (g)');
%set(gca, 'XTick', [], 'YTick', []);

% Normalizacao LPA2V
[lin, col] = size(dados);

for cc=2:col

    spec = dados(:, cc);
    % Normalizacao LPA2V
    minVal = min(spec);
    maxVal = max(spec);
    norm_spec = (spec - minVal) / (maxVal - minVal) ;

    % comprimento do das leituras do espectro normalizadas;
    lNorm = length(norm_spec);

    % obtendo media movel LPA2V com 10 CNAPs, eliminando ruidos;
    for cn=1:10
        Me_ant = 0;
        norm_spec(1) = 0;
        for c = 2:lNorm
            Me = (norm_spec(c) - (1-Me_ant)+1) / 2;
            norm_spec(c) = Me;
            Me_ant = Me;
        end
    end

    padrao(cc-1,:) = norm_spec'; % tranpose column to line

end

[pLin, pCol] = size(padrao);

```

```

for cc = 1:pCol

    aux1 = padrao(:,cc);

    pMax = max(aux1);
    pMin = min(aux1);

    s = pLin;

    while s >= 2

        % indentifica Mi e Lambida e salva suas posicoes;

        max_n = max(aux1);
        min_n = min(aux1);

        mi = max_n;
        la = 1 - min_n;

        % calculo LPA2V

        gc = mi - la;
        gct = (mi + la) - 1;

        x = (1-abs(gc))^2;
        y = gct^2;

        d = sqrt(x+y);

        if d>=1
            gcr=0;
        else
            if gc>0
                gcr=1-d;
            else
                gcr=d-1;
            end
        end

        Mier = (gcr + 1) / 2;

        % encontra posicao dos valores e remove dos padrao.
        posMi = find(aux1==max_n,1);
        aux1(posMi) = [];

        posLa = find(aux1==min_n,1);
        aux1(posLa) = [];

        i = length(aux1);

        aux1(i+1) = Mier;

        s = s -1;
    end
end

```

```

end

pFinal(1,cc) = Mier;
pFinal(2,cc) = pMax;
pFinal(3,cc) = pMin;

end

xlswrite('padrao.xlsx',pFinal);
xlswrite('freq.xlsx',freq);

disp("Padrão de Vibração Aprendido");
% hold on;
% plot(freq,pFinal,'r');

fclose(se);
main_menu;

%plot(freq,padrao(1,:));

% Codigo de Plot de Padrões
% Plot_padrao.m
% Escrito por Marcio Correa e Dra. Dora Vilanova
% 2022

padrao_comp = xlsread('padrao.xlsx');

padrao = padrao_comp(1,:);

p_freq = xlsread('freq.xlsx');
hold on;
grid on;

%plot(p_freq,padrao,'r');
area(p_freq,padrao);%
title ('PAL2V Pattern');
legend('Readings');
xlabel('Frequency (Hz)');
ylabel('Amplitude - Evidence Degrees');
%set(gca,'XTick',[],'YTick',[]);

main_menu;

% Codigo de Monitoramento e Comparação
% Monitorar4.m
% Escrito por Marcio Correa e Dra. Dora Vilanova
% 2022

disp("Monitoramento por Padrão");

% Carrega padrao completo salvo em arquivo

```

```

padrao_comp = xlsread('padrao_normal.xlsx');

% Padrao de frequencia para plotar;
p_freq = xlsread('freq.xlsx');

% Apenas o padrao normalizado
padrao_apren = padrao_comp(1,:);

% Apenas o valor maximo
padrao_max = padrao_comp(2,:);

% Apenas o valor minimo
padrao_min = padrao_comp(3,:);

% CARGA PROBLEMAS e CONDICOES

% 1 - Carrega padrao com problemas de fixacao
padrao_comp_fixacao = xlsread('padrao_fixacao.xlsx');

% Apenas o padrao normalizado - problemas de fixacao
padrao_fixacao = padrao_comp_fixacao(1,:);

% Apenas o valor maximo
padrao_fixacao_max = padrao_comp_fixacao(2,:);

% Apenas o valor minimo
padrao_fixacao_min = padrao_comp_fixacao(3,:);

% 2 - Carrega padrao com problemas de balacimento
padrao_comp_desbalanceado = xlsread('padrao_desbalanceado.xlsx');

% Apenas o padrao normalizado - problemas de fixacao
padrao_desbalanceado = padrao_comp_desbalanceado(1,:);

% Apenas o valor maximo
padrao_desbalanceado_max = padrao_comp_desbalanceado(2,:);

% Apenas o valor minimo
padrao_desbalanceado_min = padrao_comp_desbalanceado(3,:);

disp("Realizando Leituras - COM");

priorPorts=instrfind;
delete(priorPorts);

se=serial('COM3','BaudRate',115200);
se.InputBufferSize = 4096 * 3;

disp("Abrindo Porta Serial");
fopen(se);
pause(2);

disp("Porta Serial Aberta");

```

```

%apenas uma pequena amostragem para o padrao de monitoramento
amostras=10;
desvio = 0.09

% inicia looping the N monitoramentos;
for loop=1:50

    for cp=1:amostras

        %disp("Enviando Comando de Aprendizagem: " + cp);
        fprintf(se, 'W');

        %disp("Aguardando Processamento");
        pause(1);

        % Le os dados da serial e armazena na matriz input
        input = str2double (strsplit(fscanf(se, '%c'), ' '));

        condicao(cp,:) = input;

        % Remove a coluna 1 pois é sempre nula.
        input(1) = [];

        %disp("Frequencia de Amostragem");
        fprintf(se, 'F');

        %pause(1);
        Fs = fscanf(se, '%d');

        N = length(input);

        %Determine FFT and Plot
        tic
        freq = 0:Fs/length(input):Fs/2; %frequency array for FFT

        xdft = fft(input); %Compute FFT
        xdft = 1/length(input).*xdft; %Normalize
        xdft(2:end-1) = 2*xdft(2:end-1);

        % obtemos apenas o espectro, sem as frequencias
        spec = abs(xdft(1:floor(N/2)+1));

        dados(:,cp+1) = spec'; % transpose line to column;

    end

    dados(:,1) = freq'; % transpose line to column;

    %csvwrite('condicao.csv',condicao);

    % Normalizacao LPA2V
    [lin,col] = size(dados);

    for cc=2:col

```

```

spec = dados(:,cc);
% Normalizacao LPA2V
minVal = min(spec);
maxVal = max(spec);
norm_spec = (spec - minVal) / ( maxVal - minVal) ;

% comprimento do das leituras do espectro normalizadas;
lNorm = length(norm_spec);

% obtendo media movel LPA2V com 6 CNAPs, eliminando ruidos;
for cn=1:10
    Me_ant = 0;
    norm_spec(1) = 0;
    for c = 2:lNorm
        Me = (norm_spec(c)-(1-Me_ant)+1) / 2;
        norm_spec(c) = Me;
        Me_ant = Me;
    end
end

padrao(cc-1,:) = norm_spec'; % tranpose column to line

end

[pLin, pCol] = size(padrao);

for cc = 1:pCol

    aux1 = padrao(:,cc);

    pMax = max(aux1);
    pMin = min(aux1);

    s = pLin;

    while s >= 2

        % indentifica Mi e Lambida e salva suas posicoes;

        max_n = max(aux1);
        min_n = min(aux1);

        mi = max_n;
        la = 1 - min_n;

        % calculo LPA2V

        gc = mi - la;
        gct = (mi + la) - 1;

        x = (1-abs(gc))^2;
        y = gct^2;

```

```

d = sqrt(x+y);

if d>=1
    gcr=0;
else
    if gc>0
        gcr=1-d;
    else
        gcr=d-1;
    end
end

Mier = (gcr + 1) / 2;

% encontra posicao dos valores e remove dos padrao.
posMi = find(aux1==max_n,1);
aux1(posMi) = [];

posLa = find(aux1==min_n,1);
aux1(posLa) = [];

i = length(aux1);

aux1(i+1) = Mier;

s = s -1;
end

pMonitor(1,cc) = Mier;
pMonitor(2,cc) = pMax;
pMonitor(3,cc) = pMin;

end

clc;
%disp("Padrão de Monitoramento Pronto");
disp("ESPALmon Output");
disp("=====");
disp("");

padrao_mon = pMonitor(1,:);

% comparando pontos com o padrao aprendido em condicoes normais
% var de comparacao em porcentual;

comp = 0;

index_true = 0;
index_false = 0;
total = 0;

```

```

for cp=1:lNorm

    if ( padrao_mon(cp) >= padrao_min(cp) && padrao_mon(cp) <=
padrao_max(cp))

        desvio_min = padrao_apren(cp) - ( padrao_apren(cp) * desvio );
        desvio_max = padrao_apren(cp) + ( padrao_apren(cp) * desvio );

        if (padrao_mon(cp) >= desvio_min && padrao_mon(cp) <=
desvio_max)

            index_true = index_true + 1;

        else

            index_false = index_false + 1;
        end

    end

end

%cla reset; % Do a complete and total reset of the axes
%hold on;

%plot(p_freq,padrao_apren,'r');
%plot(p_freq,padrao_mon,'k');

total = index_true + index_false;

index = (index_true / total) * 100;
fprintf('Normal Operation Index = %4.2f %%\n',index);

% comparando pontos com o padrao aprendido em condicoes de fixacao
% var de comparacao em porcentual;
comp = 0;

index_true = 0;
index_false = 0;
total = 0;

%for cp=1:lNorm

for cp=1:lNorm

    if ( padrao_mon(cp) >= padrao_fixacao_min(cp) && padrao_mon(cp) <=
padrao_fixacao_max(cp))

        desvio_min = padrao_fixacao(cp) - ( padrao_fixacao(cp) *
desvio );
        desvio_max = padrao_fixacao(cp) + ( padrao_fixacao(cp) *
desvio );

```

```

        if (padrao_mon(cp) >= desvio_min && padrao_mon(cp) <=
desvio_max)

            index_true = index_true + 1;

        else

            index_false = index_false + 1;
        end

    end

end

total = index_true + index_false;

index = (index_true / total) * 100;
fprintf('Looseness Failure Index = %4.2f %%\n',index);

% comparando pontos com o padrao aprendido em condicoes de
% balanceamento
% var de comparacao em porcentual;
comp = 0;

index_true = 0;
index_false = 0;
total = 0;

for cp=1:lNorm

    if ( padrao_mon(cp) >= padrao_desbalanceado_min(cp) &&
padrao_mon(cp) <= padrao_desbalanceado_max(cp) )

        desvio_min = padrao_desbalanceado(cp) -
( padrao_desbalanceado(cp) * desvio );
        desvio_max = padrao_desbalanceado(cp) +
( padrao_desbalanceado(cp) * desvio );

        if (padrao_mon(cp) >= desvio_min && padrao_mon(cp) <=
desvio_max)

            index_true = index_true + 1;

        else

            index_false = index_false + 1;
        end

    end

end

total = index_true + index_false;

```

```
index = (index_true / total) * 100;
fprintf('Unbalancing Failure Index = %4.2f %%\n',index);

%     cla reset; % Do a complete and total reset of the axes
%     hold on;

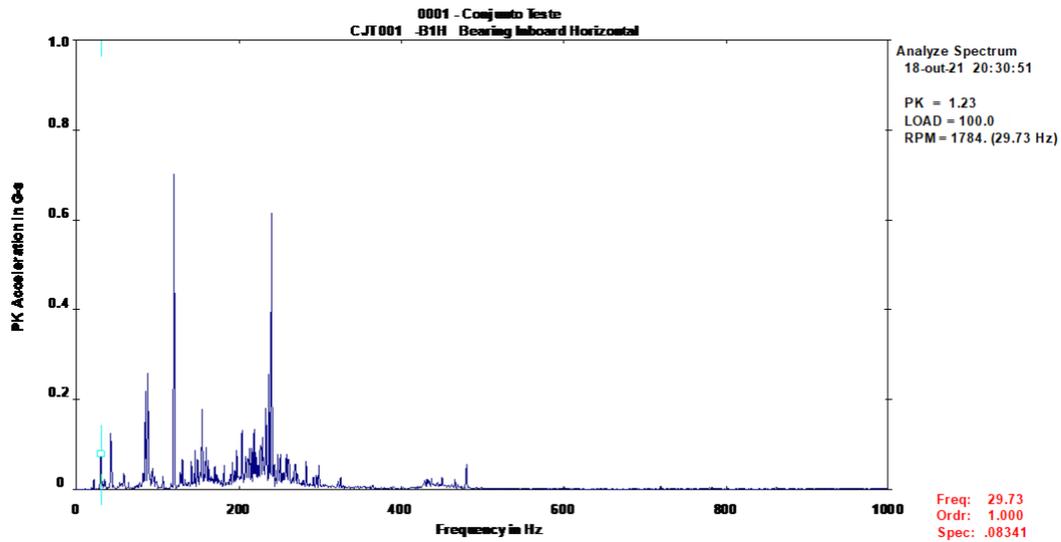
% plot(p_freq,padrao_apren,'r');
% plot(p_freq,padrao_fixacao,'g');
%     plot(p_freq,padrao_desbalanceado,'y');
%     plot(p_freq,padrao_mon,'k');

end

fclose(se);
%main_menu;
```

APÊNDICE C – Análise de Vibração do Conjunto Mecânico

- Software – MHM – Machinery Health Manager / Emerson - US
- Coletor de Dados CSI 2120



Frequência de Rotação 29.73 Hz (1N);
Coleta na direção horizontal (X);
Amplitude de vibração mancal polia movida B1H .083 Gs

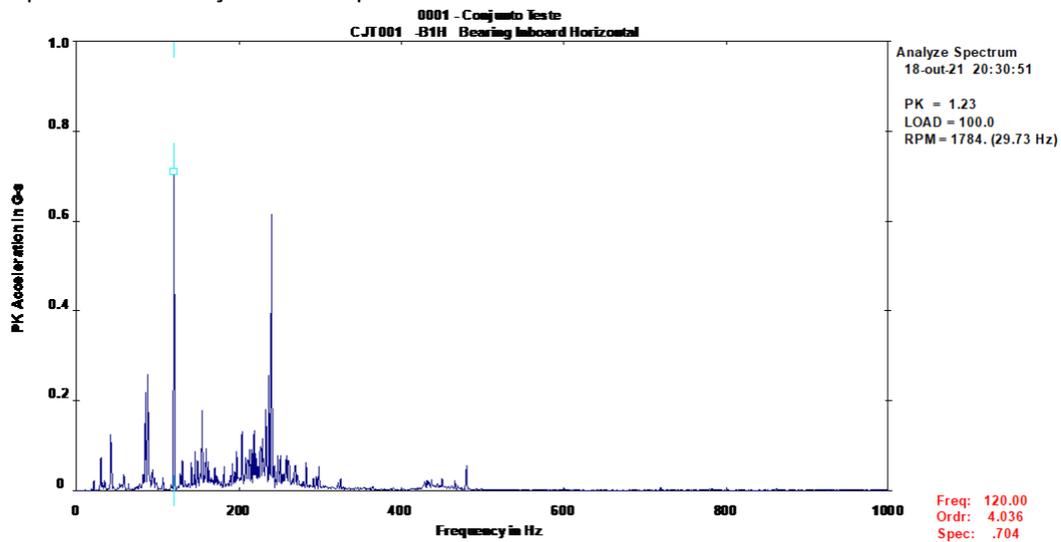


Gráfico com plotagem em 120 Hz;
Coleta na direção horizontal (X);
Polia Movida B1H .704 Gs;

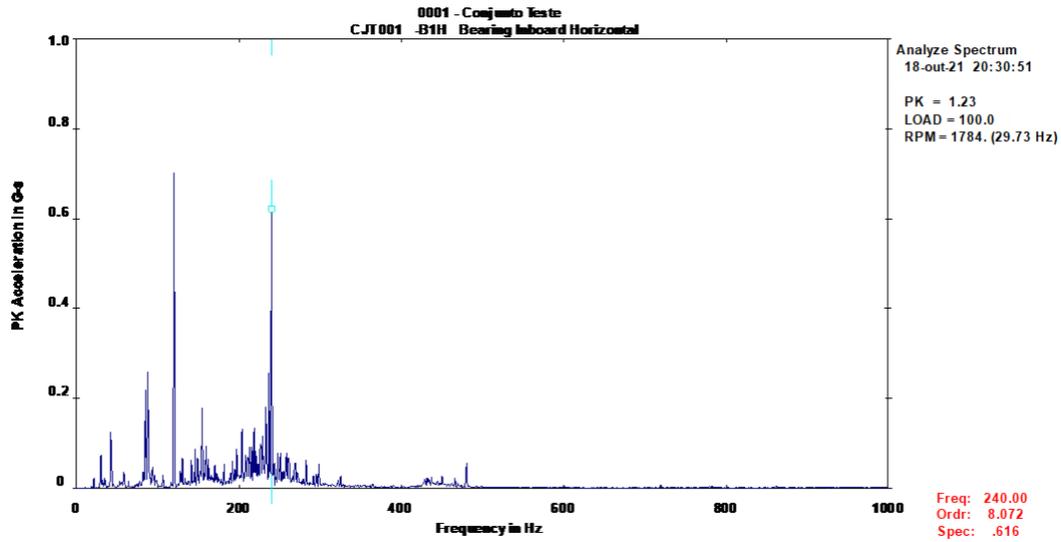
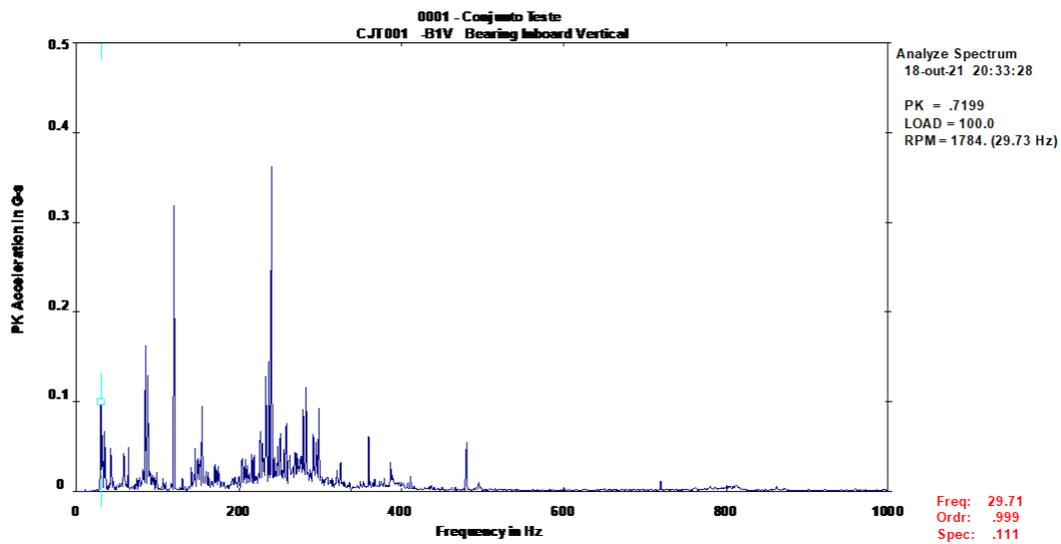


Gráfico com plotagem em 240 Hz.
Coleta na direção Horizontal (X);
Polia Movida B1H .616Gs;



Frequência de Rotação 29.71 Hz (1N);
Coleta na direção vertical(Z);
Amplitude de vibração mancal polia movida B1V .111 Gs

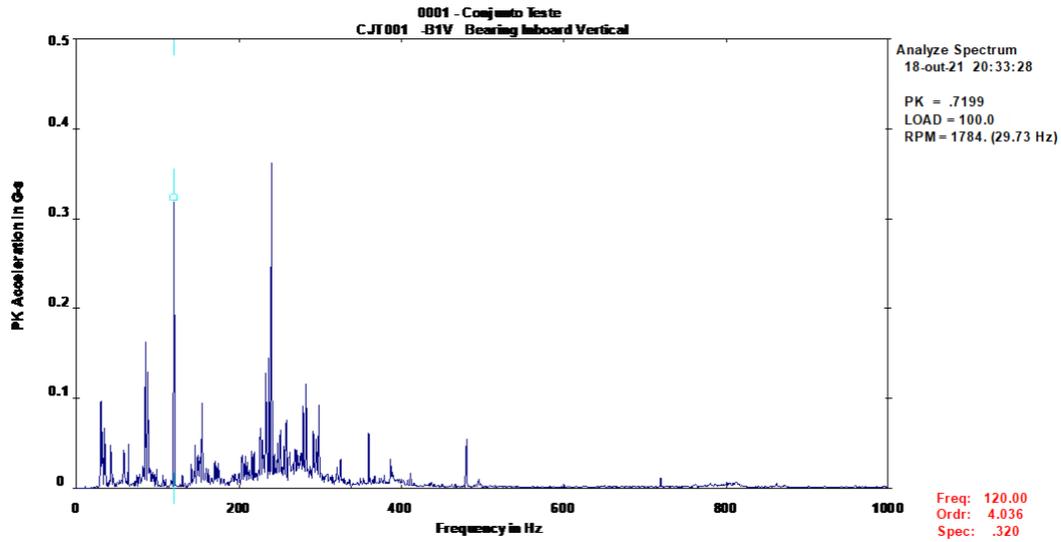


Gráfico com plotagem em 120 Hz;
Coleta na direção Vertical (Z);
Polia Movida B1V .320Gs

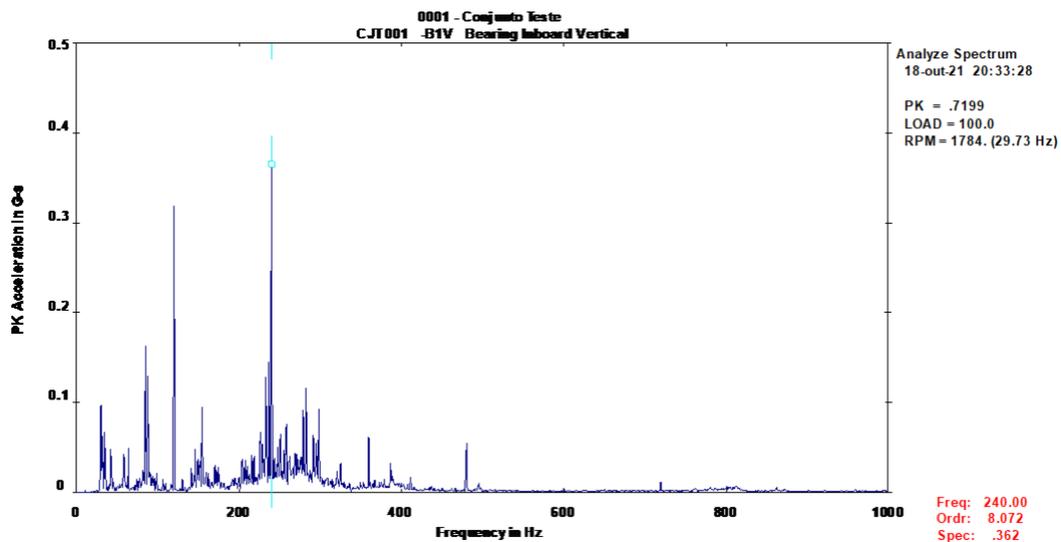
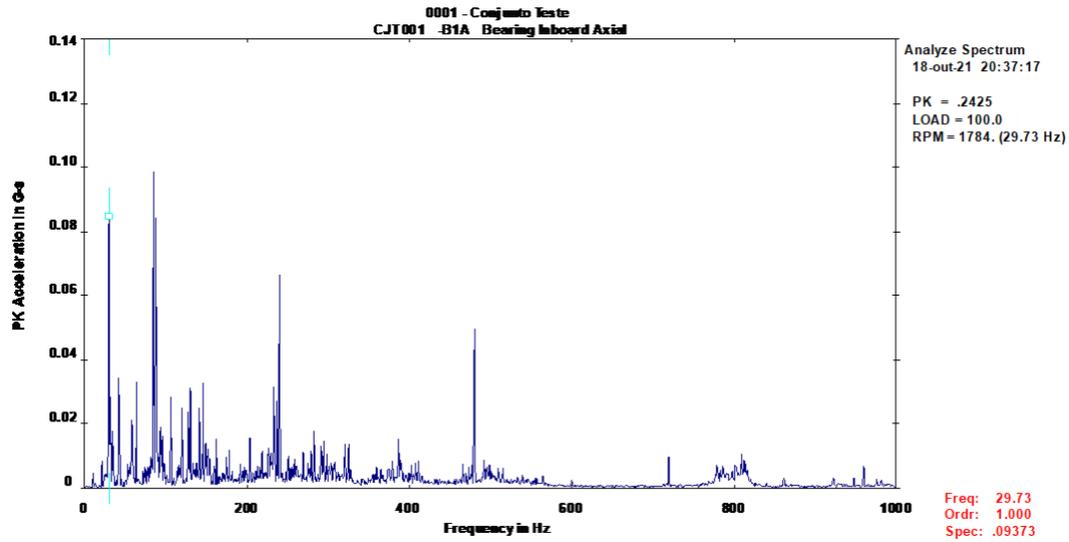


Gráfico plotagem em 240 Hz;
Coleta na direção Vertical (Z);
Polia Movida B1V .362Gs;



Frequência de Rotação 29.73 Hz (1N);
Coleta na direção axial (Y);
Amplitude de vibração mancal polia movida B1A .093 Gs

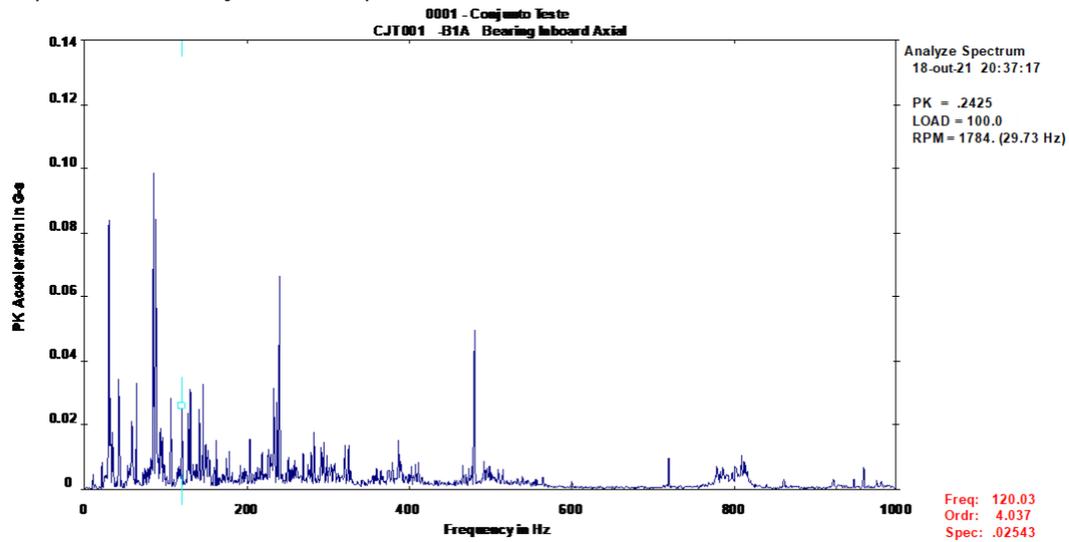


Gráfico com plotagem em 120 Hz;
Coleta na direção axial (Y);
Polia Movida B1A .025Gs

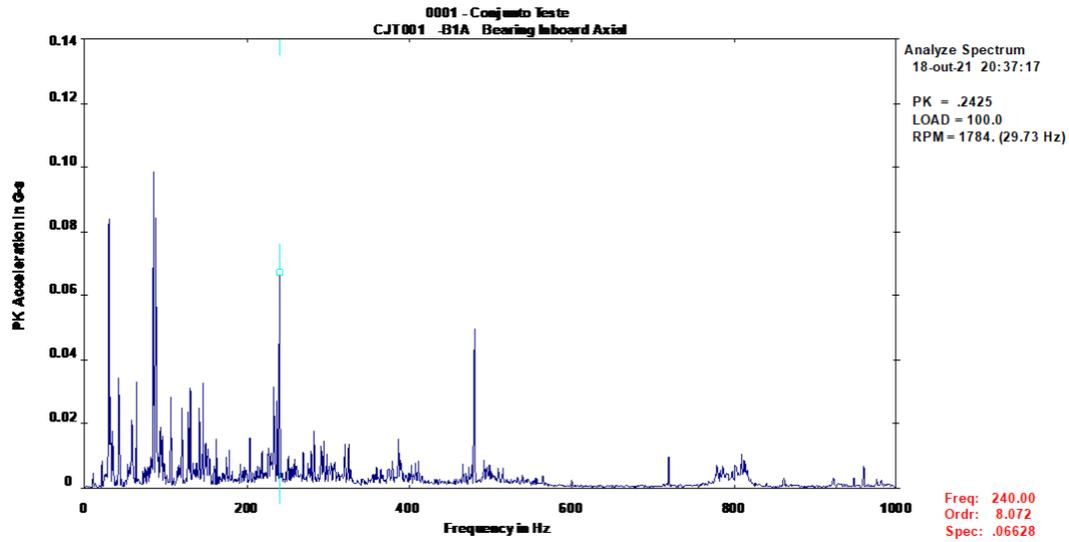
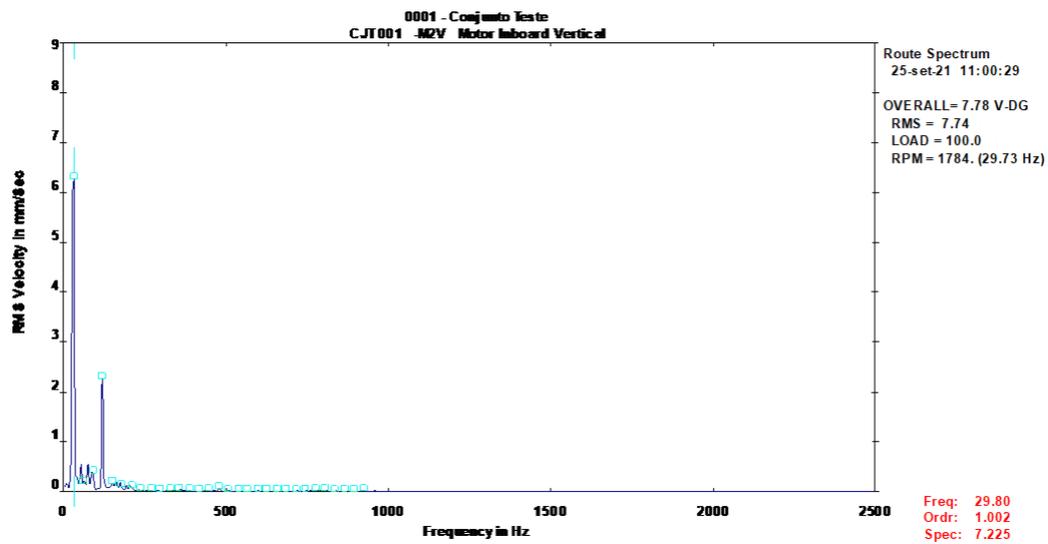
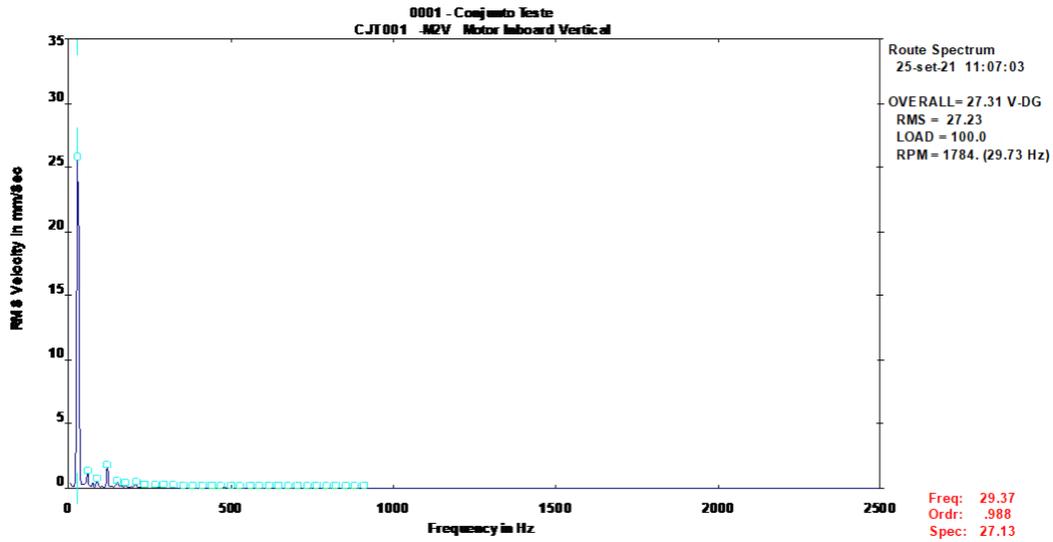


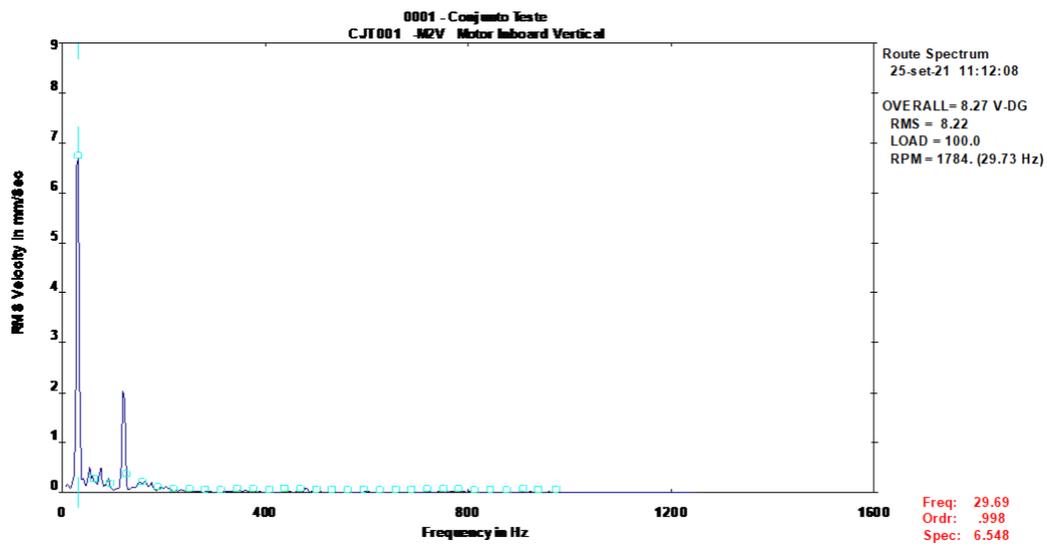
Gráfico com plotagem em 240 Hz;
Coleta na direção axial (Y);
Polia Movida B1A .025Gs



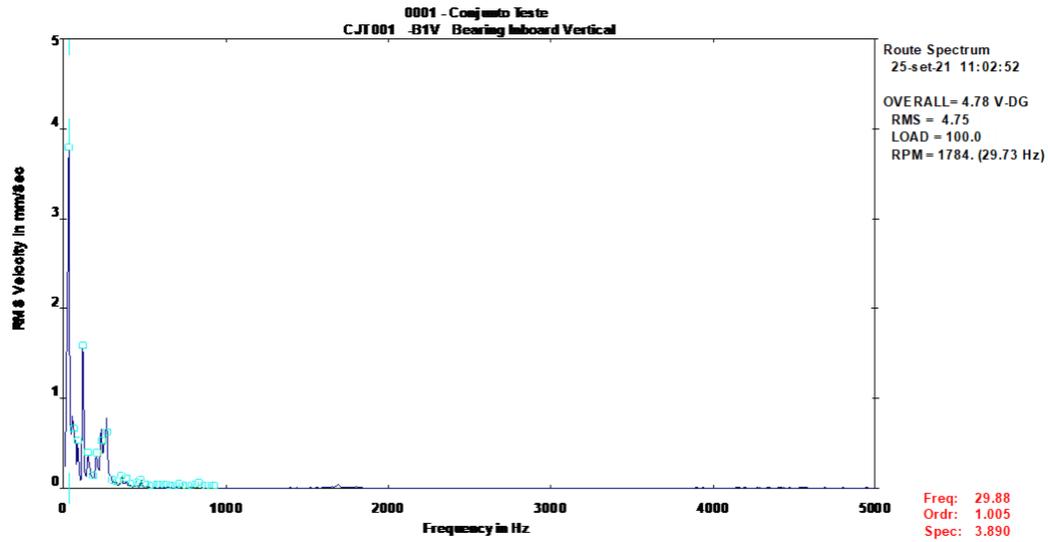
- Mancal – (LA) do motor – M2V 7.22 mm/s.
- Protótipo em condições normais de operação;
- Acelerômetro na direção Vertical;



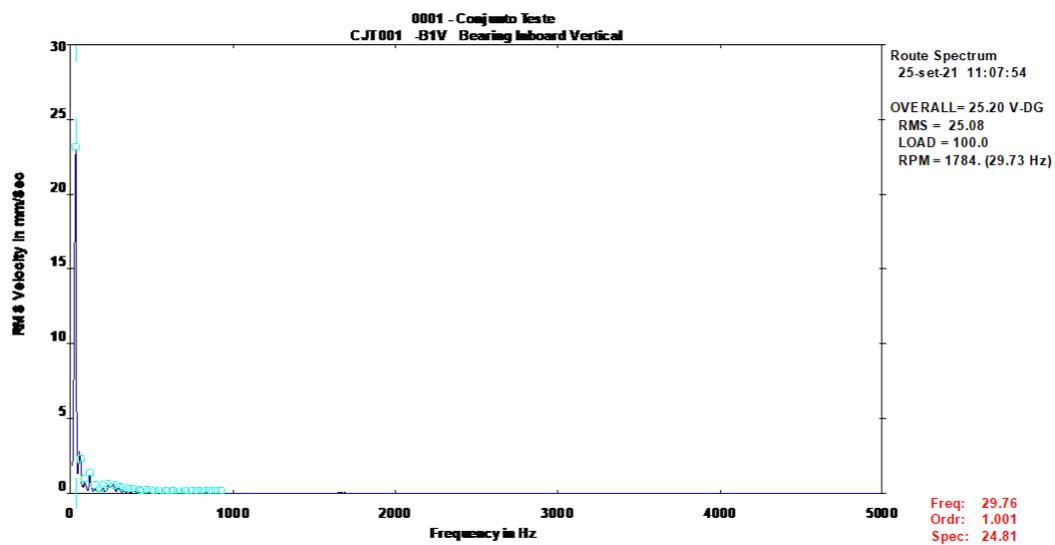
- Mancal – (LA) do motor – M2V **27.13** mm/s.
- Protótipo operando com (1N) predominante no espectro indicando desbalanceamento;
- Acelerômetro da direção Vertical;



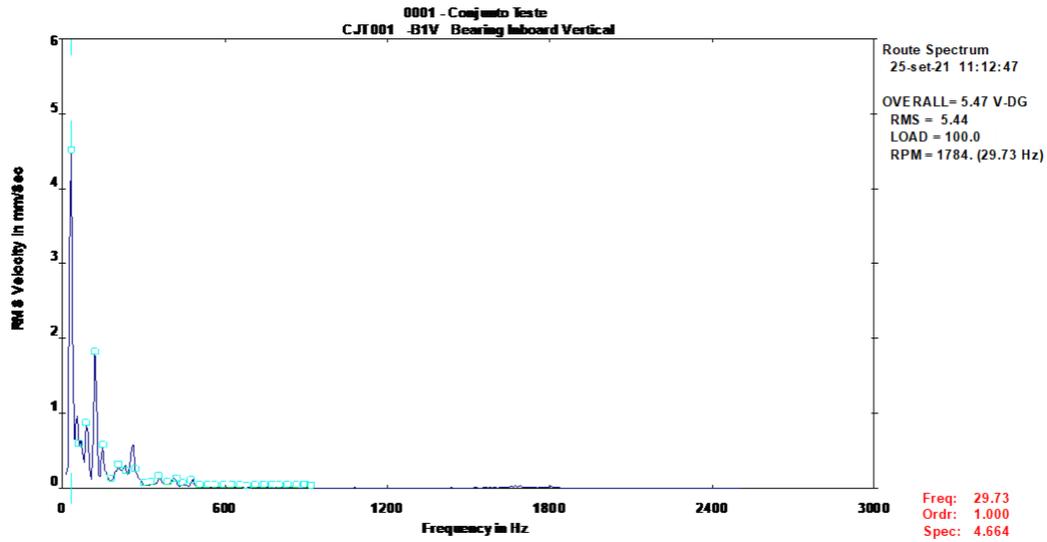
- Mancal – (LA) do motor – M2V 6.54 mm/s.
- Protótipo operando com pé manco espectro indicando baixa rigidez;
- Acelerômetro na direção vertical;



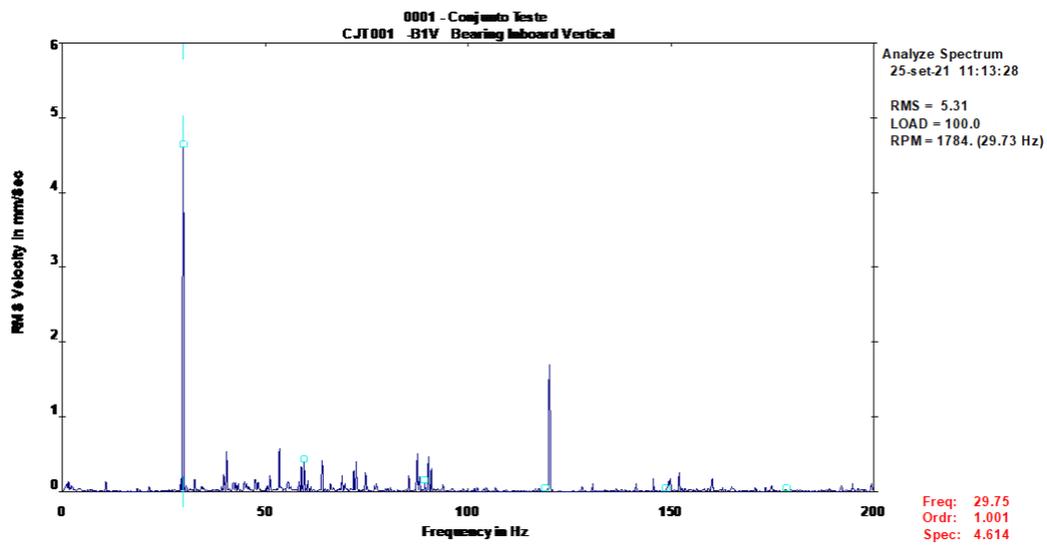
- Mancal – (B1) polia movida – B1V 3.89 mm/s.
- Protótipo em condições normais de operação;
- Acelerômetro na direção vertical;



- Mancal – (B1) polia movida – B1V **24.81** mm/s.
- Protótipo operando com (1N) predominante no espectro indicando desbalanceamento;
- Acelerômetro na direção vertical;

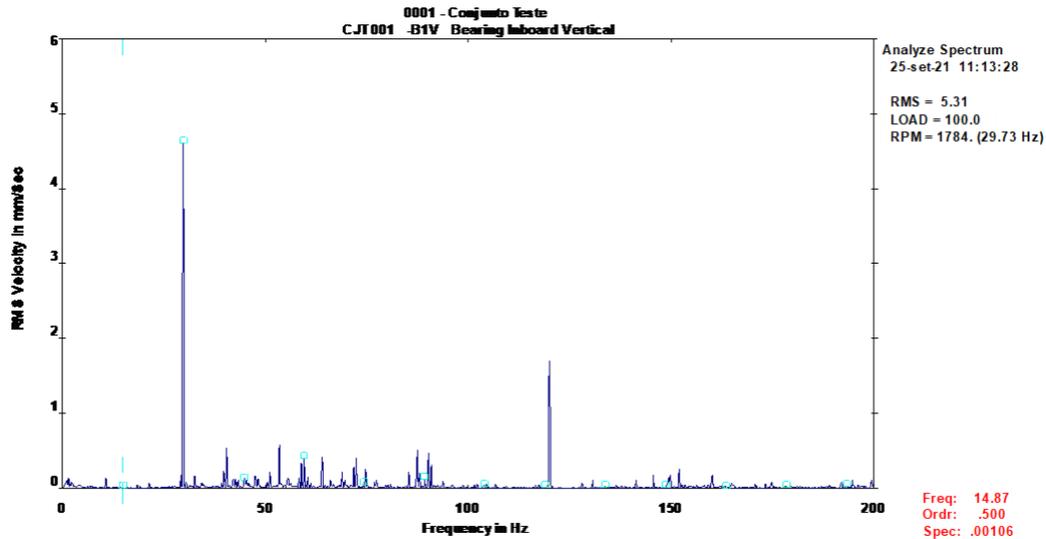


Mancal – (B1) polia movida – B1V 4.61 mm/s. Protótipo operando com pé manco espectro indicando baixa rigidez;

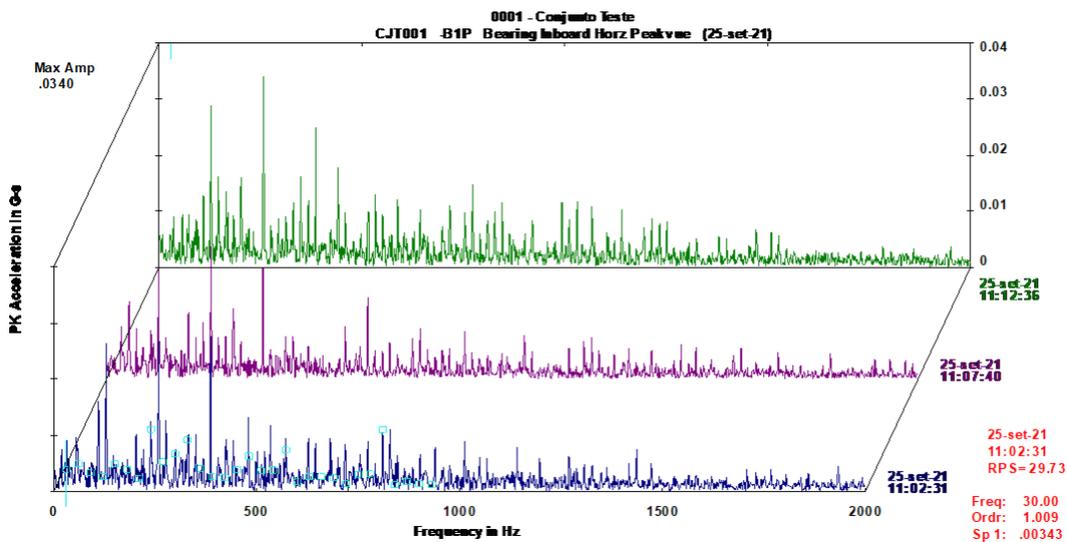


COLETA COM MONITORAMENTO ESPECIAL – RESOLUÇÃO COM 200 HZ DE FUNFO DE ESCALA

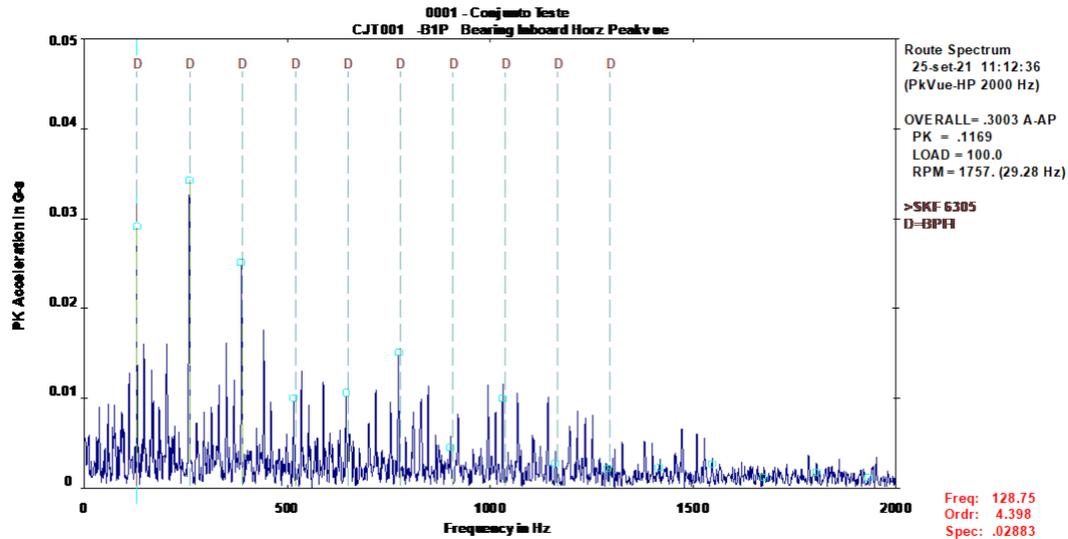
Mancal – (B1) polia movida – B1V 4.61 mm/s. Protótipo operando com pé manco espectro indicando baixa rigidez;



COLETA COM MONITORAMENTO ESPECIAL – RESOLUÇÃO COM 200 HZ DE FUNFO DE ESCALA
Mancal – (B1) polia movida – B1V plotagem em (0.5 N) com harmônicos. Protótipo operando com pé manco
espectro indicando baixa rigidez;



- Mancal – (B1) polia movida – B1P.
- Espectros em cascata coletas em níveis de envelope (aceleração);
- Na sequência de horário de coleta:
 - 11h02min. Protótipo em condições normais de operação;
 - 11h07min. Protótipo operando com desbalanceamento;
 - 11h12min. Protótipo operando com pé manco;



- Mancal – (B1) polia movida – B1P.
- Espectros em níveis de envelope (aceleração);
- Coleta realizada com o pé manco;
- Protótipo apresentando nos espectros frequência de falha (BPH) Pista interna do rolamento SKF – 6305, porém, em níveis de amplitude incipiente (0.30 Gs).

NOTA

1. Em uma avaliação comparativa, nota-se que, os espectros e amplitudes de vibrações na condição de operação normal do protótipo e com pé manco da polia movida, estão semelhantes, isso porque são características do conjunto, no entanto, é importante ressaltar que, na condição com a polia movida desbalanceada, há uma alteração significativa em ambos os mancais, mas, exemplos a seguir são da polia movida:
 - a. Medição normal – B1V 3.89 mm/s.
 - b. Medição com polia desbalanceada – B1V **24.81** mm/s.
 - c. Medição com pé manco – B1V 4.61 mm/s.
2. Em aceleração (níveis de envelope), nas três condições, normal, com desbalanceamento e, pé manco os espectros do mancal – (B1) são semelhantes e com amplitudes também incipientes, variando entre (0.20 Gs e 0.30 Gs) apresentando Pista interna do Rolamento SKF-6305.
3. Além disso, em todas as condições também há frequência de rotação e harmônicos, com uma pequena variação na amplitude conforme (item 2);
4. Outro sim, isso poderá estar ocorrendo em função do dimensional, alojamento do rolamento, pré carga em relação ao eixo e, também base do sensor tri - axial;