



UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE ENERGIAS ALTERNATIVAS E RENOVÁVEIS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA



Thais Christine Borges da Silva

Algoritmo Baseado em Lógica Fuzzy Para Detecção e Classificação de *Misfire* em Veículos Automotivos

João Pessoa

2018

Thais Christine Borges da Silva

Algoritmo Baseado em Lógica Fuzzy Para Detecção e Classificação de *Misfire* em Veículos Automotivos

Monografia apresentada a Coordenação do Curso de Engenharia Elétrica da Universidade Federal da Paraíba, como requisito parcial para obtenção do título de Engenheiro Eletricista.

Universidade Federal da Paraíba

Centro de Energias Alternativas e Renováveis

Departamento de Engenharia Elétrica

Orientador: Prof. Dr. Abel Cavalcanti Lima Filho

Coorientador: Prof. Msc. Thyago Leite de Vasconcelos Lima

João Pessoa

2018

Thais Christine Borges da Silva

Algoritmo Baseado em Lógica Fuzzy Para Detecção e Classificação de *Misfire* em Veículos Automotivos

Monografia apresentada a Coordenação do Curso de Engenharia Elétrica da Universidade Federal da Paraíba, como requisito parcial para obtenção do título de Engenheiro Eletricista.

Trabalho aprovado. João Pessoa, 15 de junho de 2018:

Prof. Dr. Abel Cavalcanti Lima Filho
Orientador

Prof. Msc. Thyago Leite de Vasconcelos Lima
Coorientador

Prof. Dr. Francisco Antônio Belo
Avaliador Interno

Prof. Dr. Koje Daniel Vasconcelos Mishina
Avaliador Interno

João Pessoa
2018

Agradecimentos

Agradeço à minha família, pelo apoio contínuo ao longo da vida, por todo esforço que me permitiu seguir em frente e pelas risadas constantes quando as adversidades surgiam.

Ao meu namorado pela paciência e compreensão incansáveis ao longo dessa jornada, por estar sempre presente e me incentivando a fazer o meu melhor e principalmente pelo amor.

À minha segunda família, Tânia, Jessé e Tayná, que tenho tanta sorte de fazer parte, pelo apoio e por sempre me receberem de braços abertos e me fazer sentir tão acolhida.

Ao meu orientador, Abel Cavalcante Lima Filho, pelo apoio, atenção e oportunidade.

Ao meu co-orientador, Thyago Leite de Vasconcelos Lima, pelos ensinamentos, paciência, companhia nos feriados e no laboratório, dedicação e, principalmente, pela amizade.

Ao professor Francisco Antônio Belo, pelas conversas e observações que ajudaram na construção desse trabalho.

Ao professor Koje Daniel Vasconcelos Mishina, pelas contribuições para a realização desse trabalho.

Aos meus amigos Giselle, Nicolas, Igor, Isabela, Rafael, por todo pensamento positivo, pelas conversas profundas e outras nem tanto, viagens e momentos tão preciosos, e à Georgia, pela companhia tão importante e especial nessa jornada tão longa pela graduação, do começo ao fim.

À Lupin, Boomer, Black e Lino, pela leveza, teimosia, brincadeira e amor que deixam tudo mais leve e dissipam o estresse.

Aos companheiros de laboratório, pela contribuição, companhia e alguns cafés.

*"No dia que a universidade me deu um diploma
e uma ciência que estava longe de carregar no cérebro.
Confesso que me senti ao mesmo tempo enganado e orgulhoso."
Machado de Assis*

Resumo

Motores de combustão interna presentes em veículos automotivos apresentam um funcionamento cíclico, ou seja, padrões se repetem dentro de um determinado intervalo de tempo, e liberam energia em forma de som. Partindo desse princípio, o presente trabalho tem como objetivo detectar e classificar a falha de *misfire* a partir do som emitido pelo carro, fazendo uso de técnicas de processamento de sinais e inteligência artificial. Inicialmente foram determinados os comportamentos padrão para o funcionamento normal e para o funcionamento com falhas, com o intuito de estabelecer uma referência. É proposto um parâmetro baseado na densidade espectral de potência como forma de identificar e classificar a falha, sendo este a entrada do sistema *fuzzy*. Os resultados obtidos com essa técnica foram considerados satisfatórios, uma vez que o algoritmo de diagnóstico apresentou alta taxa de acerto.

Palavras-chave: Processamento de Sinais. Lógica Fuzzy. Transformada de Fourier. Arduino. Densidade Espectral de Potência. Motor de Combustão.

Abstract

Internal combustion engines in automotive vehicles have a cyclic operation, patterns that repeat themselves within a certain time interval, and release energy in the form of sound. Based on this principle, the present work aims to detect and classify misfire from the sound produced by the car, making use of signal processing and artificial intelligence. Initially, the standard behaviors for normal operation and faulty operation were determined in order to establish a reference. It is proposed a parameter based on the power spectral density as a way to identify and classify the fault, this being the input of the fuzzy system. The results obtained with this technique were considered satisfactory, since the diagnosis' algorithm presented a high rate of correct classification.

Keywords: Signal Processing. Fuzzy. Fourier Transform. Arduino. Spectral Density of Power. Combustion Engine.

Lista de ilustrações

Figura 1 – Componentes do Motor	17
Figura 2 – Ciclo de quatro tempos do motor de combustão	18
Figura 3 – STFT de um sinal $x(n)$	22
Figura 4 – Funções Janela	23
Figura 5 – Diagrama H-E do conjunto <i>fuzzy</i> em um universo real $U:(-10,10)$	25
Figura 6 – Arquitetura simplificada de um sistema de classificação <i>fuzzy</i>	26
Figura 7 – Etapas do trabalho	27
Figura 8 – Posição para aquisição do sinal sonoro	28
Figura 9 – Sistema de Aquisição de Áudio e Cálculo da FFT	29
Figura 10 – FFT do sinal adquirido para condição normal	30
Figura 11 – FFT do sinal adquirido para acionamento sem a vela 1	31
Figura 12 – FFT do sinal adquirido para acionamento sem a vela 3	31
Figura 13 – FFT do sinal adquirido para acionamento sem o par de vela 1 e 4	32
Figura 14 – FFT do sinal adquirido para acionamento sem o par de vela 2 e 3	32
Figura 15 – Densidade espectral de potência para o pico em 13,33 Hz	33
Figura 16 – Densidade espectral de potência para o pico em 20 Hz	34
Figura 17 – Densidade espectral de potência para o pico na frequência fundamental	34
Figura 18 – Configuração dos conjuntos de entrada da lógica <i>fuzzy</i>	35
Figura 19 – Conjunto de saída da lógica <i>fuzzy</i>	35
Figura 20 – <i>Toolbox</i> para implementação <i>fuzzy</i> no <i>Matlab</i>	37
Figura 21 – Configuração dos conjuntos de entrada da lógica <i>fuzzy</i> no <i>Matlab</i>	37
Figura 22 – Regras da lógica <i>fuzzy</i> para implementação no <i>Matlab</i>	38
Figura 23 – Comparação entre o sistema <i>fuzzy</i> no Arduino e no <i>Matlab</i> para comportamento normal	39
Figura 24 – Comparação entre o sistema <i>fuzzy</i> no Arduino e no <i>Matlab</i> para acionamento sem 1 vela	40
Figura 25 – Comparação entre o sistema <i>fuzzy</i> no Arduino e no <i>Matlab</i> para acionamento sem 1 par de velas	40

Lista de tabelas

Tabela 1 – Características do sistema de aquisição de áudio.	28
Tabela 2 – Características do sistema de aquisição de áudio.	30
Tabela 3 – Matriz de confusão do sistema <i>fuzzy</i> implementado no Arduino	41
Tabela 4 – Matriz de confusão do sistema <i>fuzzy</i> implementado no <i>Matlab</i>	41

Lista de abreviaturas e siglas

OBD	<i>On-Board Diagnostic</i> : Diagnóstico de Bordo
GPRS	<i>General Packet Radio Service</i>
FFT	<i>Fast Fourier Transform</i> : Transformada Rápida de Fourier
CWT	<i>Continuous Wavelet Transform</i> : Transformada Wavelet Contínua
PMA	Ponto morto alto
PMB	Ponto morto baixo
FS	<i>Fourier Series</i> : Série de Fourier
FT	<i>Fourier Transform</i> : Transformada de Fourier
DTFS	<i>Discrete-Time Fourier Series</i> : Serie de Fourier no Tempo Discreto
DTFT	<i>Discrete-Time Fourier Transform</i> : Transformada de Fourier em Tempo Discreto
DC	<i>Direct Current</i>
IA	Inteligência Artificial
H-E	Hassi-Euler
PSD	<i>Power Spectral Density</i> : Densidade Espectral de Potência

Sumário

1	APRESENTAÇÃO	11
1.1	Introdução	11
1.2	Estado da Arte	13
1.3	Premissas e Hipóteses	15
1.4	Objetivos	15
1.4.1	Objetivo Geral	15
1.4.2	Objetivos Específicos	15
2	FUNDAMENTAÇÃO TEÓRICA	16
2.1	Motor de Combustão Interna	16
2.2	Análise Espectral de Sinais	18
2.2.1	Transformada de Fourier	19
2.2.2	Transformada de Fourier em Tempo Discreto	20
2.2.3	Transformada de Fourier de Curta Duração	21
2.3	Inteligência Artificial e Sistemas Inteligentes	23
2.3.1	Lógica Fuzzy	24
3	MATERIAIS E MÉTODOS	27
3.1	Veículo	27
3.2	Descrição do Sistema de Aquisição de Áudio e Cálculo da FFT	28
3.3	Obtenção dos Padrões de Falha	30
3.4	Obtenção do Parâmetro de Falha	32
3.5	Classificação das Falhas	34
4	RESULTADOS	39
5	CONCLUSÃO	42
	REFERÊNCIAS	43

1 Apresentação

1.1 Introdução

O processo de identificação e classificação de falha consiste em decidir se uma falha aconteceu ou não, com a geração de um diagnóstico, e identifica-lá a partir de comparação com comportamentos ou variáveis os quais já exista conhecimento do que é esperado ou normal (BELO, 2003). Muitas vezes cabe ao mecânico usar sua memória e experiência para jogar se há ou não uma falha no veículo, por vezes desmontando o carro desnecessariamente para descobrir qual foi a falha e onde ela ocorreu, o que muitas vezes se revela uma perda de tempo, além de causar danos às peças mais facilmente e aumentar o custo da manutenção. Somado a isso, existe o fato que os usuários de veículos tendem a ignorar pequenos problemas e prestam pouca, ou nenhuma, atenção à manutenção do carro, tornando ainda mais difícil o diagnóstico de falha que na maioria dos casos só ocorre após a quebra do veículo (SHEN; LUO, 2014).

Atualmente, a detecção de falhas em veículos automotivos também pode ser feita pelo OBD (*On-Board Diagnostic*), um sistema de auto-diagnóstico desenvolvido para veículos automotores que consiste em um conjunto de sensores previamente instalados, de forma que as informações levantadas pela leitura dos sensores sejam usadas para diagnóstico de falhas. Para análise dos dados levantados pelos sensores se torna necessário o uso de processamento digital de sinais, que desempenha um papel cada vez mais importante no controle do motor para atender aos requisitos cada vez mais rigorosos de eficiência de combustível e regulamentações de emissões. Na indústria automotiva, o desenvolvimento do controle e diagnóstico do motor está superando o desenvolvimento do próprio motor. Algoritmos sofisticados, baseados na teoria de controle e em processamento de sinais, estão sendo integrados ao controle do motor (WU; NAIK, 2017).

A detecção de faltas por análise sonora parte do princípio que alguns componentes presentes no motor de veículos automotivos apresentam um funcionamento cíclico, ou seja, se repetem dentro de um determinado intervalo de tempo, e liberam energia em forma de som. Quando peças presentes nesses componentes apresentam defeito, a energia liberada em forma de som apresenta outra característica e para diferentes falhas diferentes ruídos são emitidos. Por se apresentarem de forma cíclica, os ruídos produzidos pela peça defeituosa possuem outras frequências predominantes quando analisados no domínio da frequência que difere das frequências predominantes do funcionamento normal (LIMA, 2017).

Uma das dificuldades encontradas nesse tipo de análise é o fato que os motores

de combustão interna possuem várias fontes de ruído: o sistema de exaustão, o sistema de admissão, o ventilador usado para resfriamento e a superfície do bloco do motor. O ruído pode ser gerado por efeitos aerodinâmicos, pode ser devido a forças que resultam do processo de combustão, ou pode resultar de excitação mecânica por componentes do motor rotativos (HEYWOOD, 1988).

Misfire é uma falha que acontece quando a mistura de combustível e ar não pôde ser queimada em um único cilindro ou em vários cilindros de uma combustão interna. Muitas razões podem resultar em *misfire*, como falha do sistema de ignição, falha do sistema de injeção de combustível, falha nas vedações do cilindro, etc. O *misfire* pode resultar na diminuição da potência de saída, aumentando o consumo de combustível, descarregando poluentes excessivos e até danificando o catalisador (YINHUI et al., 2011).

Esse tipo de falha é uma das mais difíceis de monitorar e identificar. Alcançar o desempenho robusto de detecção de *misfire* necessário, mantendo baixos os custos de implementação, é uma tarefa difícil enfrentada pelos fabricantes automotivos, especialmente porque há uma variedade de configurações de motores emergentes e estratégias de controle implantadas para melhorar a economia de combustível. O principal desafio no desenvolvimento de um detector de *misfire* em tempo real é encontrar um algoritmo robusto e unificado capaz de detectar falhas e cilindros defeituosos com alta precisão de detecção, com pouco ou nenhum alarme falso, sob todas as condições de operação do motor, padrões de *misfire* e fontes potenciais de interferência de ruído. (WU; NAIK, 2017).

Existe basicamente três formas de detecção de *misfire*: uma é a observação da rotação do virabrequim, a segunda é a observação da corrente iônica e a última é a observação da pressão do cilindro. A primeira maneira é a mais popular devido ao seu custo. Os fatores que podem tornar a detecção de falha de ignição observando a rotação do virabrequim difíceis são o ruído na estrada, a aceleração do motor, o desvio do sistema de detecção da rotação do eixo de manivelas, a vibração do virabrequim e assim por diante. A vibração do virabrequim torna-se um problema maior quando a velocidade do motor é alta e dificulta a detecção de falha de ignição em condições de alta velocidade do motor. Devido a essa dificuldade, a detecção de falha de ignição só é obrigatória sob condições de baixa velocidade do motor (AONO; FUKUCHI, 2005).

O presente trabalho pretende desenvolver uma técnica para diagnóstico de *misfire* via análise sonora, de forma não invasiva e em tempo real. O sistema desenvolvido será responsável por analisar o som emitido pelo motor por meio de processamento digital de sinais e realizar o diagnóstico do estado atual do motor utilizando inteligência artificial. A principal vantagem apresentada pelo sistema é a redução substancial no número de sensores utilizados, além de não precisar haver contato entre o sistema e o elemento monitorado.

Primeiramente serão definidos os padrões de comportamento para os sinais sonoros emitidos pelo veículo para funcionamento normal e também para funcionamento defeituoso.

A análise do sinal sonoro será feita no domínio da frequência e a classificação da falha será feita por um sistema inteligente.

1.2 Estado da Arte

Através da resolução nº 354, de 13 de dezembro de 2004, o CONAMA "dispõe sobre os requisitos para adoção de sistemas OBD nos veículos automotores leves objetivando preservar a funcionalidade dos sistemas de controle de emissão", tornando obrigatória a adoção de sistemas OBD para veículos leves de passageiros e leves comerciais destinados ao mercado brasileiro. Inicialmente, a primeira geração OBDBr-I tinha como objetivo diagnosticar falhas em veículos relacionadas com a emissão de gases. A última geração do padrão nacional OBDBr-II permite um rápido diagnóstico de falhas relacionadas com emissão de gases, além de ter padronizado os protocolos de comunicação, meios de proteção e linguagem das informações armazenadas (BRASIL, 2004). Entretanto, segundo Barkai (2001), apenas 20% das vezes que o OBD identifica uma falha, ele fornece informação necessária para se identificar um componente defeituoso, pois o sistema monitora um sistema completo e não componentes separadamente. Em falhas mais complexas, que envolva múltiplos componentes, as informações podem ser ainda mais imprecisas.

Um dos problemas apresentados pelo OBD é o fato que as informações de falhas são apresentadas via aviso luminoso ou de forma mais detalhada com o auxílio de um escâner e os usuários podem continuar a usar o veículo sem saber qual a extensão da falha. Apresentando uma solução para esse problema, Chou et al. (2001) propôs um método e uma estrutura para extrair informações do sistema de controle do veículo e transferir para um local remoto, provendo meios e métodos para análise e comparação dessas informações, além de prover informações oportunas ao motorista do veículo.

Belo (2003) propõe um sistema para automatizar o diagnóstico de falhas em veículos fazendo uso do sistema de diagnósticos fornecidos pelo OBD-II. As informações adquiridas pelo sistema embarcado do carro seriam comparadas com informações obtidas a partir de modelos da estrutura e do comportamento dos componentes do veículos e partir disso seria possível a identificação e diagnóstico da falha de maneira automatizada.

No trabalho apresentado por Lowrey et al. (2003), um sistema *wireless* foi proposto utilizando o mesmo barramento de dados do OBD-II com as informações enviadas para um local remoto em tempo real e oferecendo um diagnóstico de falha enquanto o veículo está em uso para consulta em um website. Um outro sistema que envia dados dos sensores para um servidor local foi proposto por Al-Tae, Khader e Al-Saber (2007) com o diferencial de também enviar a localização do veículo utilizando sistema por satélite GPRS (*General Packet Radio Service*).

Explorando a dificuldade de detecção de *misfire* quando o veículo está em alta

velocidade, [Aono e Fukuchi \(2005\)](#) simularam essa situação para desenvolver uma nova forma de diagnóstico e propuseram um filtro, robusto para a aceleração, capaz de detectar não só as falhas periódicas, mas também falhas isoladas de *misfire*. Em um trabalho semelhante, [Yinhui et al. \(2011\)](#) observaram que os algoritmos de diagnóstico de *misfire* existentes podem detectar com precisão a falha contínua de disparo de cilindro único ou cilindro múltiplo, mas, ocasionalmente, pode não detectar com precisão a falha aleatória de vários cilindros e pode julgar erroneamente o cilindro onde ocorreu o *misfire*. A partir disso, sugeriram um método avançado de detecção de *misfire* do motor a gasolina usando vários filtros.

Alguns outros trabalhos foram desenvolvidos utilizando a mesma técnica de análise para detecção de *misfire* utilizando a medida da velocidade do virabrequim e utilizando como solução técnicas diferentes de filtragem ([NAIK, 2004](#); [LEE; LOH; WU, 2003](#)).

No trabalho elaborado por [Ma et al. \(2015\)](#) a detecção de *misfire* foi feita utilizando as flutuações de velocidade do virabrequim, utilizando uma combinação de análise na frequência, com aplicação da Transformada Rápida de Fourier (FFT - *Fast Fourier Transform*) para extrair as características de frequência específica dos sinais de *misfire*, com lógica *fuzzy* para reduzir possíveis falsos alarmes causada pela interrupção do ruído e perturbação de estrada irregular.

[Montani e Speciale \(2006\)](#) descreveram em seu trabalho uma nova técnica de análise baseada na abordagem *wavelet* que permite a extração dos componentes de frequência relacionados a um evento de falha de ignição e sua localização no domínio do tempo. O processo de detecção é realizado analisando a medição da velocidade angular do virabrequim, para sublinhar as desacelerações devido a eventos de *misfire*, e investigar o padrão pós-falha em um domínio adequado (domínio *wavelet*). O algoritmo proposto também permite um fácil isolamento do cilindro responsável pela falha de ignição.

O número de trabalhos sobre o diagnóstico de falhas de motores usando análise sonora ainda é pequeno. [Wu et al. \(2009\)](#) propuseram um algoritmo de transformada Wavelet contínua (CWT - *Continuous Wavelet Transform*), para processamento do sinal de áudio emitido em diferentes condições de funcionamento, combinado com um método de seleção de características é proposto para a análise de sinais de falha em um sistema de diagnóstico de falhas de *scooters*. Os resultados experimentais indicaram que o sistema proposto obteve uma taxa de reconhecimento de falhas superior a 95% na plataforma experimental do sistema de diagnóstico de falhas de *scooters*.

[Anami e Pagi \(2014\)](#) investigou a detecção de falhas de motocicletas usando o *chaincode* do pseudospectro. Os sinais sonoros da motocicleta são analisados para variações espectrais e essas variações são traçadas por um *chaincode*. Os recursos de *chaincode* são usados para classificar a amostra em saudável ou com falha usando a técnica de distorção de tempo dinâmico. Os resultados de classificação obtidos são superiores a 91% e 93%,

respectivamente, para motocicletas defeituosas e saudáveis.

1.3 Premissas e Hipóteses

- É possível desenvolver um método não invasivo para a detecção de *misfire* em motor de combustão em veículos automotivos de maneira satisfatória.
- Com o uso de um sistema embarcado é possível adquirir e processar os sinais sonoros adquiridos durante o funcionamento do motor e realizar o diagnóstico em tempo real.
- É possível realizar o diagnóstico da falha precocemente, antes da parada total do funcionamento do veículo.

1.4 Objetivos

1.4.1 Objetivo Geral

Desenvolver, validar, implementar e analisar um algoritmo baseado em lógica fuzzy para diagnóstico de *misfire* em veículos automotivos para funcionamento em sistemas embarcados.

1.4.2 Objetivos Específicos

- Pesquisar as técnicas existentes para diagnóstico da falha de misfire em veículos automotivos;
- Determinar os padrões de funcionamento do veículo com base nos sinais de áudio adquiridos;
- Determinar e extrair parâmetro(s) relevantes para o processo de classificação dos sinais;
- Desenvolver um algoritmo baseado em lógica fuzzy em linguagem de programação C/C++ para a classificação das falhas;
- Integrar o algoritmo desenvolvido com o sistema embarcado utilizado na pesquisa;
- Realizar testes do sistema, validando os resultados.

2 Fundamentação Teórica

Neste capítulo serão apresentados os conceitos de ferramentas de análises para processamento de sinais utilizados nessa pesquisa, mais aprofundadamente os aspectos teóricos sobre a Transformada de Fourier (FT - *Fourier Transform*). Também será apresentado nesse capítulo os conceitos do algoritmo de classificação aplicado nesse trabalho, a Lógica Fuzzy, para a interpretação das análises dos sinais de áudio.

2.1 Motor de Combustão Interna

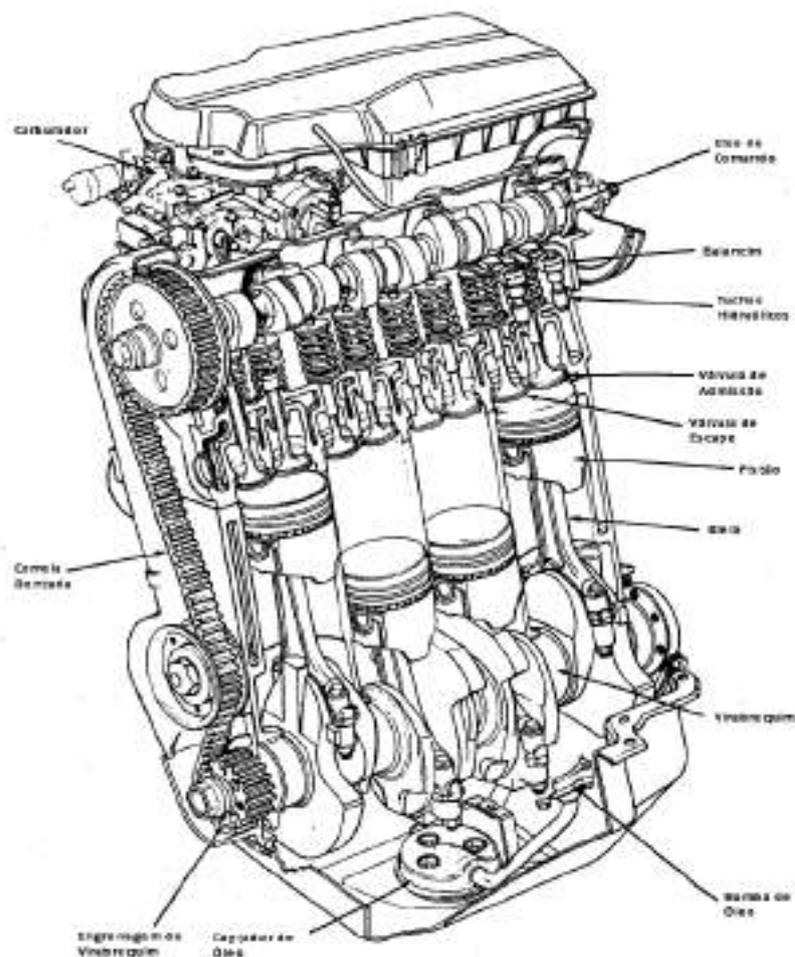
A energia mecânica fornecida para o funcionamento de um motor de combustão interna vem da energia química contida no combustível quando este é queimado ou oxidado dentro do motor. A mistura ar-combustível antes da combustão e os produtos queimados após a combustão são os fluidos de trabalho propriamente ditos, as transferências de trabalho que fornecem a potência de saída desejada ocorrem diretamente entre esses fluidos de trabalho e os componentes mecânicos do motor (HEYWOOD, 1988).

Em motores do tipo ignição com faísca, ar e combustível são misturados no sistema de entrada, antes de entrarem no cilindro do motor, a energia é produzida pela queima de combustível dentro de uma câmara de combustão ou um cilindro contendo um pistão que sobe e desce em um movimento alternativo resultante da combustão. Estendendo-se para baixo do pistão, há uma biela que liga o pistão ao virabrequim. A biela e o virabrequim convertem o movimento recíproco do pistão em um movimento rotativo (DHS, DEPARTMENT OF HOMELAND SECURITY, 2012). Em motores de automóveis a temperatura do ar que está entrando é controlada misturando ar do ambiente com o ar que foi esquentado por contato com o coletor de escape. Os componentes do motor são apresentados na Figura 1.

O ciclo termodinâmico mais comum em motores de veículos leves é o ciclo de quatro tempos ou ciclo de Otto que descreve o funcionamento de um típico motor de pistão de ignição com faísca (WU, 2003), onde o motor aproveita a energia da queima de combustível no interior de um cilindro e os gases impulsionam um pistão. O ciclo de quatro tempos requer, para cada cilindro do motor, duas revoluções do virabrequim para cada ciclo de explosão. De maneira simplificada, o ciclo de Otto de um motor de combustão interna consiste em quatro etapas (HEYWOOD, 1988), que estão ilustradas na Figura 2:

- **Admissão:** que se inicia com o pistão no PMA (ponto morto alto) e termina com o pistão no PMB (ponto morto baixo), ao fazer isso a nova mistura de ar-combustível

Figura 1 – Componentes do Motor



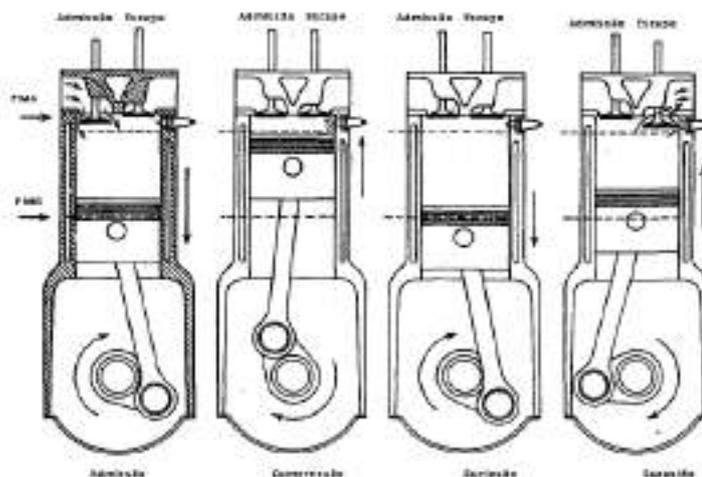
Fonte: Adaptado de [Heywood \(1988\)](#)

é puxada para dentro do cilindro. Para aumentar a massa introduzida, a válvula de entrada abre pouco antes da fase começar e fecha depois de terminar;

- **Compressão:** quando ambas as válvulas estão fechadas e a mistura dentro do cilindro é comprimida a uma pequena fração de seu volume inicial. No final da fase de compressão, a combustão é iniciada e a pressão do cilindro sobe mais rapidamente;
- **Explosão:** começa com o pistão em PMA e termina em PMB à medida que os gases de alta temperatura e alta pressão empurram o pistão para baixo e forçam a manivela a girar. Cerca de cinco vezes mais trabalho é feito no pistão durante a fase de explosão do que o pistão teve que fazer durante a compressão. Quando o pistão se aproxima do PMB, a válvula de exaustão abre para iniciar o processo de exaustão e soltar a pressão do cilindro para se aproximar da pressão de exaustão;
- **Exaustão:** onde os gases queimados remanescentes saem do cilindro: primeiro,

porque a pressão do cilindro pode ser substancialmente mais alta que a pressão de exaustão, assim os gases são varridos pelo pistão à medida que ele se move em direção ao PMA. Quando o pistão se aproxima de PMA, a válvula de entrada é aberta. Logo após o pistão chegar no PMA, as válvulas de escape se fecham e o ciclo recomeça.

Figura 2 – Ciclo de quatro tempos do motor de combustão



Fonte: Adaptado de [Heywood \(1988\)](#)

2.2 Análise Espectral de Sinais

Em sua grande maioria, as técnicas de processamentos de sinais são empregadas para extração de informações e alguns tipos de sinais apresentam mais explicitamente informações quando mapeados no domínio da frequência. A Transformada de Fourier (FT - *Fourier Transform*) é a técnica mais clássica para processamento de sinais determinísticos no domínio do tempo contínuo e de energia limitada, sendo responsável pelo mapeamento entre os domínios do tempo e da frequência ([MENEZES, 2014](#)). Apresenta-se como um método efetivo e largamente utilizado em processamento de sinais e diagnóstico de falhas, sendo mais adequado para sinais de natureza estacionária, ou seja, que não variam no tempo ([SANTIAGO; PEDERIVA, 2003](#)).

Segundo [Haykin e Veen \(2007\)](#) existem quatro formas de representação em Transformada de Fourier, dependendo do tipo de sinal a ser analisado, sendo eles:

- Série de Fourier (FS - *Fourier Series*); para sinais periódicos e contínuos;
- Transformada de Fourier (FT - *Fourier Transform*): para sinais aperiódicos e contínuos;

- Série de Fourier no Tempo Discreto (DTFS - *Discrete-Time Fourier Series*): para sinais periódicos em tempo discreto;
- Transformada de Fourier em Tempo Discreto (DTFT - *Discrete-Time Fourier Transform*): para sinais aperiódicos em tempo discreto.

2.2.1 Transformada de Fourier

De acordo com [Smith \(1997\)](#), a Transformada de Fourier para sinais contínuos é dividida em duas formas. Para sinais contínuos e periódicos no tempo, utiliza-se a Série de Fourier e para sinais contínuos, porém aperiódicos no tempo, é usada a forma da Transformada de Fourier simplesmente chamada de Transformada de Fourier.

A forma trigonométrica da Série de Fourier para um sinal periódico pode ser expressa segundo a Equação 2.1 ([LATHI, 2005](#)), também conhecida como equação de síntese da forma trigonométrica da Série de Fourier:

$$x(t) = a_0 + \sum_{k=1}^{\infty} [a_k \cos(k\omega_0 t) + b_k \sen(k\omega_0 t)] \quad (2.1)$$

onde $\omega_0 = \frac{2\pi}{T_0}$ e a_0 , a_k e b_k , os coeficientes de Fourier são expressos de acordo com as Equações 2.2, 2.3 e 2.4, respectivamente.

$$a_0 = \frac{1}{T_0} \int_0^{T_0} x(t) dt \quad (2.2)$$

$$a_k = \frac{2}{T_0} \int_0^{T_0} x(t) \cos(k\omega_0 t) dt \quad (2.3)$$

$$b_k = \frac{2}{T_0} \int_0^{T_0} x(t) \sen(k\omega_0 t) dt \quad (2.4)$$

Se o sinal $x(t)$ for uma tensão ou uma corrente, o termo a_0 representa o nível DC (*direct current*) do sinal periódico ([MAGALHÃES, 2007](#)).

A Série de Fourier em sua forma exponencial é a representação mais utilizada, apresentada na Equação 2.5, por meio da qual um sinal periódico pode ser representado como uma soma de exponenciais complexas.

$$x(t) = \sum_{k=-\infty}^{\infty} X_k e^{jk\omega_0 t} \quad (2.5)$$

onde X_k é chamado coeficiente espectral e pode ser obtido pela Equação 2.6. A série exponencial de Fourier é apenas outra forma da série trigonométrica de Fourier, onde

cada senoide de frequência ω pode ser expresso por uma soma de duas exponenciais complexas: $e^{j\omega t}$ e $e^{-j\omega t}$.

$$X_k = \frac{1}{T_0} \int_0^{T_0} x(t) e^{jk\omega t} dt \quad (2.6)$$

Um sinal aperiódico pode ser expresso como uma soma contínua de exponenciais, aplicando um processo de limitação, de forma a forçar o sinal aperiódico em um periódico. Sendo um sinal $x(t)$ aperiódico, ao determinar o período desse sinal como $T_0 \rightarrow \infty$, o mesmo irá se repetir sempre em intervalos infinitos de tempo, assim sua frequência fundamental será $\omega_0 \rightarrow 0$ e o espectro do sinal se tornará contínuo. A soma contínua de exponenciais se torna integral chamada Integral de Fourier, sendo a Integral Direta de Fourier expressa na Equação 2.7 e a Integral Inversa de Fourier expressa na Equação 2.8 (LATHI, 2005).

$$X(\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt \quad (2.7)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(\omega) e^{j\omega t} d\omega \quad (2.8)$$

2.2.2 Transformada de Fourier em Tempo Discreto

A Transformada de Fourier em Tempo Discreto (DTFT - *Discrete-Time Fourier Transform*) é importante para o processamento de sinais digitais onde os dados precisam ser discretizados e quantizados para que possam ser manipulados (MENEZES, 2014). Enquanto a Transformada de Fourier representa um sinal contínuo como uma soma de senoides ou exponenciais contínuas, a DTFT representa o sinal discreto como uma soma de senoides ou exponenciais discretas, sendo essa a principal diferença entre as duas (LATHI; GREEN, 2014).

A Transformada de Fourier em Tempo Discreto é definida segundo a Equação 2.9 (MITRA; KUO, 2006).

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} \quad (2.9)$$

em que $\omega = \frac{2\pi f}{f_s}$ é a frequência normalizada em radianos e $X(e^{j\omega})$ é a notação adotada para diferenciar a DTFT da TF (MENEZES, 2014). Por essa equação é possível observar que $X(e^{j\omega})$ é uma função periódica com um período de 2π , ou seja, $X(e^{j\omega+2\pi}) = X(e^{j\omega})$ (MITRA; KUO, 2006).

No geral, $X(e^{j\omega})$ é uma função complexa da variável real ω e pode ser escrita na forma retangular como na Equação 2.10 ou na forma polar, mostrada na Equação 2.11.

$$X(e^{j\omega}) = X_{re}(e^{j\omega}) + jX_{im}(e^{j\omega}) \quad (2.10)$$

na qual $X_{re}(e^{j\omega})$ é a parte real de $X(e^{j\omega})$ e $X_{im}(e^{j\omega})$ é a parte imaginária.

$$X(e^{j\omega}) = |X(e^{j\omega})|e^{j\theta(\omega)} \quad (2.11)$$

onde $\theta(\omega) = \arg\{X(e^{j\omega})\}$. A magnitude de $X(e^{j\omega})$ é $|X(e^{j\omega})|$ e a sua fase é $\theta(\omega)$.

A Transformada Inversa de Fourier é mostrada na Equação 2.12 e pode ser interpretada como os coeficientes na representação da função periódica $X(e^{j\omega})$.

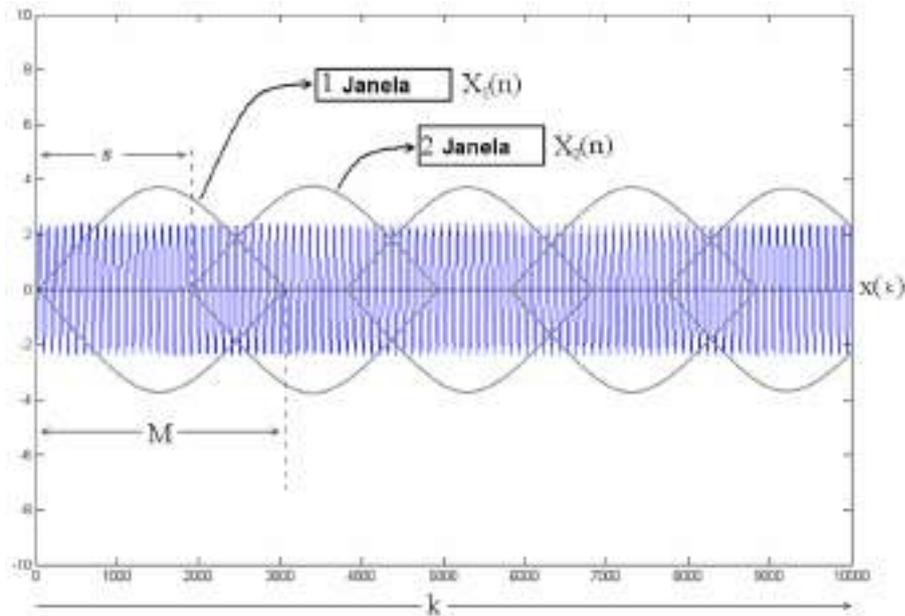
$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n} d\omega \quad (2.12)$$

2.2.3 Transformada de Fourier de Curta Duração

A transformada de Fourier é uma ferramenta matemática potente, mas não diretamente relevante para o processamento prático de sinais analógicos, porque a integração deve ser realizada desde o início do tempo até bem depois do observador deixar de se preocupar com a resposta, além de ser necessário assumir que o sinal não varia com o tempo, mesmo no infinito. Essa afirmação não é verdadeira para a maior parte dos sinais reais, o que torna o uso da FT pouco frequente em processamento de sinais digitais na prática (STEIN, 2000).

A transição da FT para a STFT requer forçar sinais arbitrários a se tornarem sinais finitos no tempo, então o sinal é separado em segmentos centrados em torno do índice de tempo no espectro de interesse e esses segmentos devem ser escolhidos em um tamanho suficientemente pequeno para serem considerados estacionários. Para evitar falsos efeitos devido à borda de cada segmento, uma função de suavização geralmente é aplicada a cada segmento, isto é normalmente feita pela multiplicação do segmento de entrada por uma função de janela, antes que seu espectro seja calculado (SORENSEN; BURRUS, 1988).

Para um sinal $x(n)$, a SFTF do sinal é realizada ao selecionar uma pequena subseção do sinal com tamanho de M amostras, em seguida é feito o janelamento ao se multiplicar o sinal por uma função janela adequada e por fim uma DTFT ou uma FFT é realizada no sinal janelado, gerando um vetor dos coeficientes espectrais, X_m . O tamanho do salto é determinado (por s amostras) e processo se repete até todo o sinal ter sido processado (ARABACI; BILGIN, 2009). O processo é ilustrado na Figura 2

Figura 3 – STFT de um sinal $x(n)$ 

Fonte: Adaptado de Arabaci e Bilgin (2009)

A STFT de uma sinal $x(n)$ é definida por (YEAP; UKIL, 2016):

$$X(n, K) = \sum_{n=0}^{N-1} x(n)w(n - ls)e^{-j\frac{2\pi}{N}kn} \quad (2.13)$$

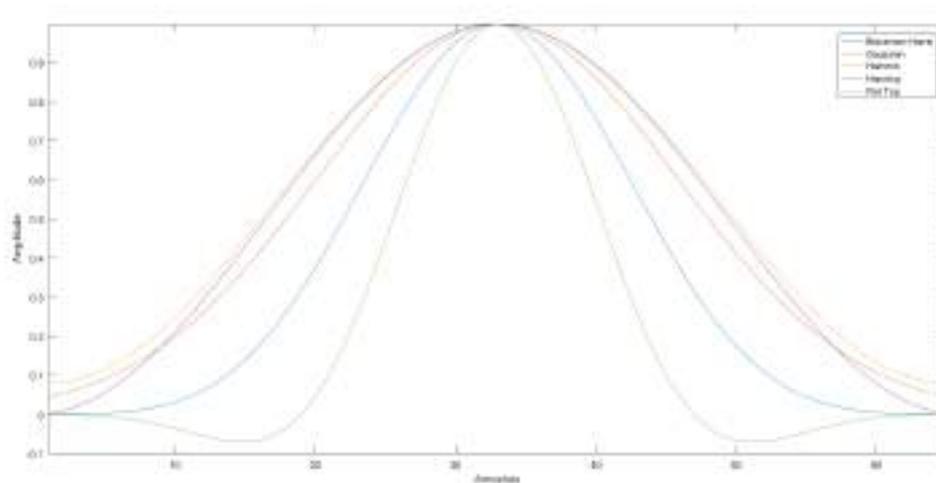
onde k é o índice de frequência, N é o tamanho da FFT, $w(n)$ é a função janela, l é a posição da janela e s é o tamanho do salto entre janelas.

O resultado da STFT é um sinal com representação bidimensional (tempo e frequência), o que requer um comprometimento entre as resoluções de cada domínio simultaneamente. Esta situação é abordada pelo princípio de incerteza de Heisenberg, para se ter uma boa ideia da composição da frequência, é necessário amostrar o sinal por um longo período, mas se ao invés disso uma pequena região do sinal for analisada, é difícil determinar o comportamento na frequência do sinal naquela região (ADDISON, 2017). Ou seja, quanto maior o tempo durante o qual observamos um sinal, mais precisas serão nossas previsões da distribuição de frequência; mas quanto maior a duração da janela, menos precisas são as alterações na frequência que podem estar ocorrendo no sinal. A desigualdade de incerteza não permite medir simultaneamente com precisão arbitrária os dois: a estrutura espectral e os tempos nos quais esta estrutura muda (STEIN, 2000).

A função de janela é uma função $w(t)$ que é estritamente zero fora do tempo de interesse. A função de janela em si não deve introduzir nenhum artefato no espectro do sinal, comumente usados são as funções de janelas que sobem suavemente e continuamente

de zero a unidade e, em seguida, retornam simetricamente para zero (STEIN, 2000). Os tipos mais comuns de função janela utilizados estão ilustrados na Figura 4.

Figura 4 – Funções Janela



Fonte: Autora

2.3 Inteligência Artificial e Sistemas Inteligentes

Segundo Barr, Feigenbaum e Roads (1982), a Inteligência Artificial (IA) "é a parte da ciência da computação envolvida no projeto de sistemas que exibem características que associamos com inteligência no comportamento humano". Ela é usada para automatizar e sistematizar processos que exijam pensamento e decisão, tornando-se útil para as mais diversas aplicações que façam uso da atividade intelectual humana (RUSSELL; NORVIG, 2004).

Uma forma de definir de forma satisfatória o funcionamento de uma IA é com o teste de Turing, também conhecido como Jogo da Imitação, proposto por Alan Turing nos anos 50. De forma simplificada, o teste consiste em um interrogador que fará perguntas para uma pessoa e uma máquina localizados em outra sala. Se a máquina conseguir imitar um humano satisfatoriamente, de forma que o interrogador não consiga dizer qual dos dois é a pessoa ou o computador, ela será considerada inteligente (TURING, 1950).

As técnicas baseadas em IA mais utilizadas são as seguintes (CAMPOS; SAITO, 2004):

- Sistemas Especialistas;
- Sistemas baseados na lógica *fuzzy*;

- Sistemas baseados em redes neurais;
- Sistemas baseados em algoritmos genéticos.

Dessa forma, um sistema inteligente é capaz de raciocinar, decidir e aprender, de forma a simular o processo de pensamento humano durante a resolução de determinados problemas e para isso eles fazem uso da Inteligência Artificial (CAMPOS; SAITO, 2004).

2.3.1 Lógica Fuzzy

Zadeh (1973), partindo do princípio que os elementos chaves do pensamento humano não são números, mas rótulos de conjuntos difusos, em classes de objetos onde a transição de um rótulo para outro acontece gradualmente e não de forma abrupta, propôs uma abordagem alternativa para lidar com solução de problemas de sistemas mais complexos: a lógica *fuzzy*.

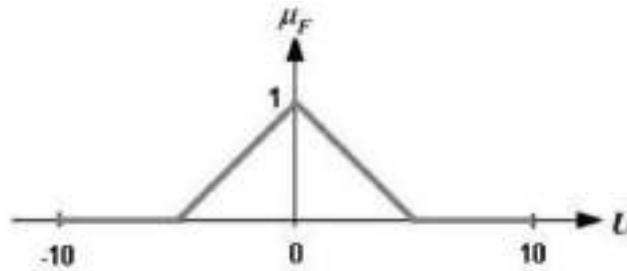
A Lógica *Fuzzy* representa a forma humana de pensar, incerta, imprecisa, difusa ou nebulosa (em inglês "*fuzzy*") em um sistema de controle, tentando imitar o processo que as pessoas utilizam para inferir conclusões que não podem ser classificadas como "sim" ou "não" (SIMOES; SHAW, 2007). Segundo Rezende (2003), ao invés de definir precisamente uma solução em termos de números exatos, a perspectiva humana de encarar problemas escolhe uma classificação ou agregação qualitativa em categorias gerais ou conjuntos de possíveis soluções. Esse tipo de classificação traz a ideia de uma variável linguística, uma entidade utilizada para representar de modo impreciso um conceito ou uma variável e apenas admite como valores expressões linguísticas, como "frio", "quente", "grande" etc. Ela pode ser representada por um conjunto *fuzzy* associado a um conceito linguístico que classifica ou define um valor impreciso.

Os conjuntos *fuzzy* são utilizados para incorporar a forma difusa de pensamento humano, evitando a passagens bruscas entre duas classes, ou a forma incerta, impedindo que os elementos não pertençam a apenas uma ou outra classe (CAMPOS; SAITO, 2004). De maneira análoga à teoria clássica de conjuntos, um elemento associado a um conjunto *fuzzy* possui um grau de pertinência associado, podendo variar entre a pertinência total e a não pertinência. O grau de pertinência de um elemento a um conjunto é caracterizado com uma função real, que assume como valor qualquer número no intervalo fechado $[0,1]$. A função de pertinência, representada na Equação 2.14 associa a cada $x \in U$ um número real $\mu_F(x)$ no intervalo $[0,1]$, representando o grau de pertinência de x em F , sendo F um conjunto *fuzzy*:

$$\mu_F(x) : U \rightarrow [0, 1] \quad (2.14)$$

A função de pertinência pode ser representada em forma de gráfico, quando o universo do conjunto é contínuo ou possui uma grande quantidade de elementos discretos, esse gráfico é chamado de Diagrama de Hassi-Euler (H-E) (REZENDE, 2003). Um exemplo é apresentado na Figura 5.

Figura 5 – Diagrama H-E do conjunto *fuzzy* em um universo real $U:(-10,10)$



Fonte: Rezende (2003)

Os conjuntos *fuzzy*, assim como os conjuntos da teórica clássica, podem se relacionar através de operações de interseção, união e complemento, definidas pelas Equações 2.15, 2.16 e 2.17, respectivamente (CAMPOS; SAITO, 2004).

$$(\forall x \in X) \Rightarrow \mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad (2.15)$$

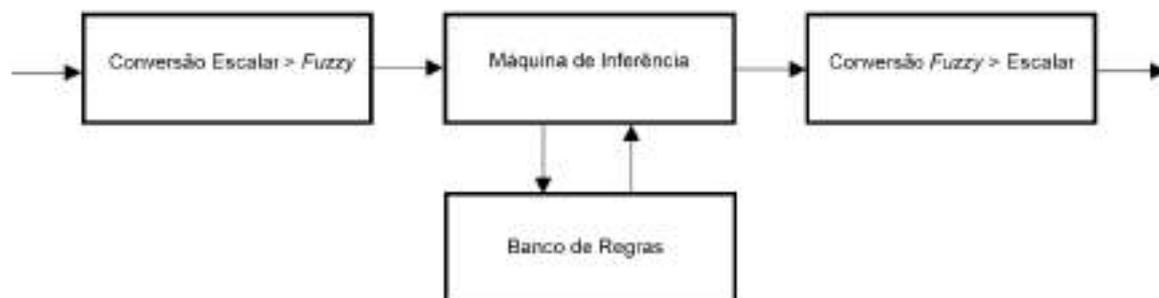
$$(\forall x \in X) \Rightarrow \mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad (2.16)$$

$$(\forall x \in X) \Rightarrow \mu_{A^c}(x) = 1 - \mu_A(x) \quad (2.17)$$

O uso mais comum da lógica *fuzzy* está no projeto de controladores, porém também pode ser usada no projeto de sistemas de classificação (BELAOUT; KRIM; MELLIT, 2016). Uma arquitetura simplificada de um sistema de classificação *fuzzy* é mostrada na Figura 6.

O processo de classificação começa com a conversão de números reais para o domínio *fuzzy*, isto é, associando a cada número real um grau de pertinência apropriado para o conjunto que dado número pertence, de forma que valores escalares sejam convertidos para um grau de associação de varia de $[0,1]$, essa etapa também é conhecida como fuzzificação (SIMOES; SHAW, 2007). O banco de regras, ou base de conhecimento, possui todas as informações dos universos de referência, dos valores linguísticos e das funções de pertinência de todas as variáveis do sistema, Ele age como o cérebro do sistema *fuzzy* através de um conjunto de regras que associam as variáveis linguísticas de entradas com

Figura 6 – Arquitetura simplificada de um sistema de classificação fuzzy



Fonte: Adaptado de [Rezende \(2003\)](#)

as de saída ([CAMPOS; SAITO, 2004](#)). A máquina de inferência avalia as regras da base de conhecimento e um conjunto *fuzzy* de saída é gerado correspondente a cada variável de saída. A etapa final do processo é conhecida por defuzzificação, onde uma informação qualitativa vinda da máquina de inferência é transformada em uma informação quantitativa para a saída ([REZENDE, 2003](#)).

3 Materiais e Métodos

No presente trabalho, o procedimento adotado para a identificação e a classificação da falha de *misfire* consiste em quatro etapas, conforme ilustrado na Figura 7. Cada etapa é descrita de forma sucinta a seguir e ilustradas no fluxograma da Figura 7:

- **Aquisição de dados** - os sinais sonoros emitidos pelo veículo são adquiridos, conforme descrito no item 3.1;
- **Cálculo da FFT** – os dados provenientes da etapa de aquisição passam por uma operação de janelamento e, em seguida, são submetidos ao processo de obtenção da FFT. A janela de Hamming foi adotada no presente trabalho;
- **Extração da característica** – São observados dois intervalos provenientes da FFT, cujos valores de pico encontrados são utilizados no cálculo da densidade espectral de potência (PSD);
- **Diagnóstico da falha** – um classificador fuzzy é utilizado para identificar e classificar a amostra de áudio analisada.

Figura 7 – Etapas do trabalho



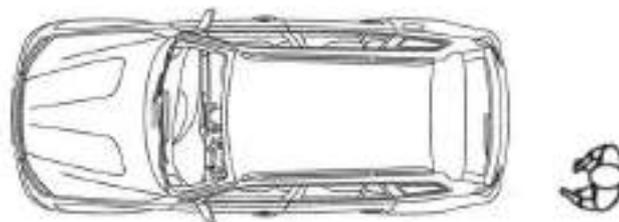
Fonte: Autora

3.1 Veículo

O veículo utilizado para as aquisições foi um carro cedido pela Universidade Federal da Paraíba (UFPB), modelo Ford Fiesta 1.6, fabricado em 2006 e sua velocidade de rotação de marcha lenta é de $850 \text{ rpm} \pm 50 \text{ rpm}$ (FORD, 2005).

A aquisição dos sinais de áudio se deu na região do escapamento do veículo, tendo sido adotado como padrão uma distância de cerca de um metro e um ângulo de aproximadamente 45° com o mesmo, como ilustrado na Figura 8, e na presença de um especialista.

Figura 8 – Posição para aquisição do sinal sonoro



Fonte: Autora

3.2 Descrição do Sistema de Aquisição de Áudio e Cálculo da FFT

O sistema de aquisição de áudio é baseado no Arduino Due, que em comparação com outros modelos da empresa, possui mais capacidade de processamento e memória (LEWIS, 2012). Mediante configuração prévia, a taxa de aquisição foi ajustada para 44,1 kHz (44100 amostras por segundo) e a resolução é de 16 bits, para que seja possível obter um sinal o mais próximo possível ao sinal analógico vindo do microfone. O elemento transdutor adotado é um microfone condensador de eletreto, CMA-4544PF-W, que está acoplado à placa por meio de uma *breakout board*, que também conta com um amplificador operacional MAX4466.

A gravação do áudio é feita no formato WAVE, formato que permite o armazenamento de sinais sem comprimi-los, garantindo que não há perda de informação, o que permite a utilização do áudio para análises e processamentos.

As informações sobre o sistema de aquisição de áudio estão resumidas na Tabela 1 e a estrutura do sistema é mostrada na Figura 9.

Tabela 1 – Características do sistema de aquisição de áudio.

Placa de Desenvolvimento	Modelo	Arduino Due
	Taxa de aquisição	44,1 kHz (44100 amostras/s)
	Resolução	16 bits
Microfone	Modelo	CMA-4445PF-W
	Sensibilidade	-44 dB
	Frequência de operação	20 Hz - 20 kHz
Áudio	Formato	WAVE

Fonte: Produzida pela autora

Os dados provenientes da etapa de aquisição passam por uma operação de janelamento e, em seguida, são submetidos ao processo de obtenção da FFT. Muitos algoritmos para a implementação da FFT encontram-se descritos na literatura, como o PFA (*Prime-Factor Algorithm*), Good-Thomas e Algoritmo de Winograd-Fourier (MAGALHÃES, 2007).

Figura 9 – Sistema de Aquisição de Áudio e Cálculo da FFT



Fonte: Lima (2017)

O algoritmo utilizado no presente trabalho se vale da técnica apresentada por Cooley e Tukey (1965). Essencialmente, a solução proposta se constitui em dividir as amostras em termos de índices ímpares e de índices pares, de maneira que após a repetição desse processo, conhecido como decimação no tempo, se tenham apenas dois valores para o cômputo da DFT, resultando assim em redução do esforço computacional (PRESS et al., 1996).

Após a decimação no tempo, o próximo passo é a realização da operação conhecida como *Butterfly*, que se vale da simetria das amostras presente na lógica da FFT para elevar o ganho computacional (PRESS et al., 1996). O algoritmo da FFT implementado no presente projeto requer $\log_2(n)$ estágios de *Butterfly* (MCKEOWN, 2010), cada estágio consistindo de $n/2$ "butterflies". Esta operação tem como entrada dois números complexos, a_k e b_k , e como saída dois novos números complexos $a_k + W_n^k$ e $a_k - W_n^k$. O fator W_n^k é chamado de fator de giro (*twiddle factor*) e, por sua vez, não depende do sinal em que a FFT é aplicada. Em síntese, para um valor fixo de n , os valores de W_n^k podem ser computados previamente para todos os valores de k e armazenados em um vetor (RIVEST; LEISERSON, 1990). A lógica do *Butterfly* resulta em uma diminuição de operações complexas, o que resulta em uma diminuição do esforço computacional, conforme ilustrado na Tabela 2.

O algoritmo utilizada para o cálculo da FFT no presente trabalho encontra-se também descrito em (LIMA, 2017), com detalhes de implementação e testes de validação.

Tabela 2 – Características do sistema de aquisição de áudio.

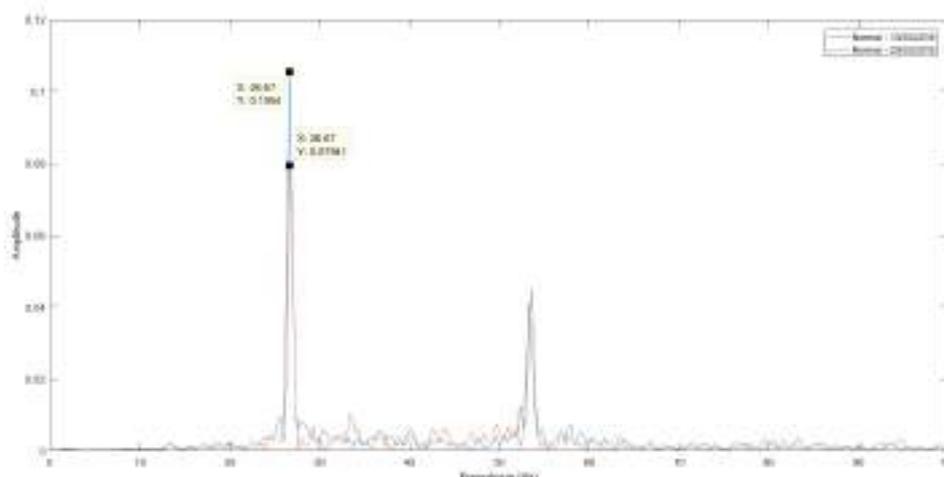
Comprimento da Amostra	Cálculo Direto da DFT		Cálculo da FFT	
	Multiplicações Complexas	Adições Complexas	Multiplicações Complexas	Adições Complexas
128	16384	16256	448	896
256	65536	65280	1024	2048
512	262144	264632	2304	4608
1024	1048576	1047552	5120	10240

Fonte: Adaptado de [McKeown \(2010\)](#)

3.3 Obtenção dos Padrões de Falha

Para determinação do padrão dos sinais sonoros emitidos pelo veículo em funcionamento normal foram feitas duas aquisições em dias diferentes, com o veículo em ponto morto, os sinais sonoros emitidos pelo veículo foram adquiridos, conforme descrito no item 3.1. Os dois sinais adquiridos são mostrados na Figura 10. É possível observar que a frequência fundamental do sinal acontece em 26,65 Hz, aproximadamente duas vezes a frequência de rotação do motor, isso ocorre porque para cada cilindro do motor, duas revoluções do virabrequim são necessárias em cada ciclo de explosão.

Figura 10 – FFT do sinal adquirido para condição normal



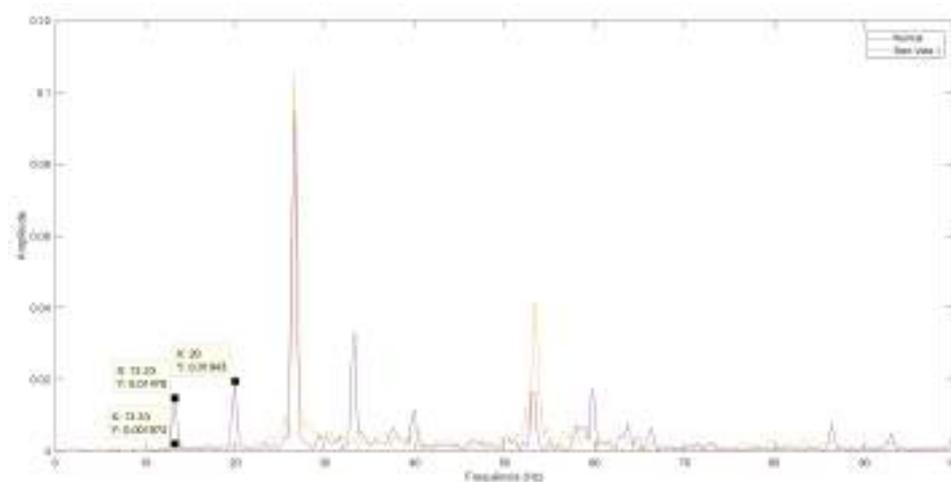
Fonte: Autora

As conexões das velas de ignição foram retiradas para levantamento do padrão do comportamento defeituoso do carro, gerando uma falha no ciclo de explosão do cilindro correspondente, uma das situações de ocorrência de *misfire*. Inicialmente foram interrompidas conexões de uma vela por vez, em seguida, foram interrompidas aos pares, velas 1 e 4, depois velas 2 e 3.

Pela Figura 11, que apresenta uma comparação entre o sinal normal e o sinal de acionamento sem a vela 1, é possível observar o surgimento de dois outros picos em uma frequência diferente da frequência fundamental, em uma distância de aproximadamente

seu valor dividido por 2.

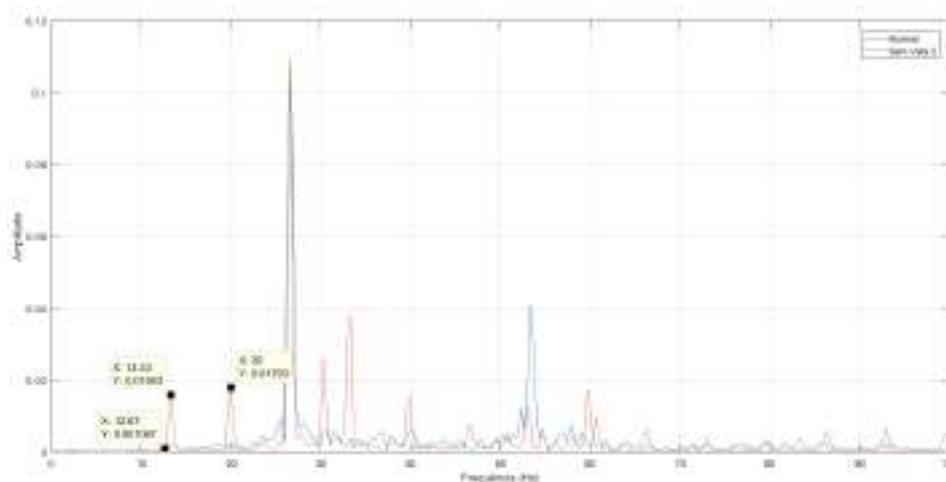
Figura 11 – FFT do sinal adquirido para acionamento sem a vela 1



Fonte: Autora

Em seguida, foram analisados os sinais de áudio para o acionamento sem a vela 3. Foi observado que o comportamento dos picos se repete, como é demonstrado na Figura 12, caracterizando um possível padrão para desconexão de uma única vela.

Figura 12 – FFT do sinal adquirido para acionamento sem a vela 3



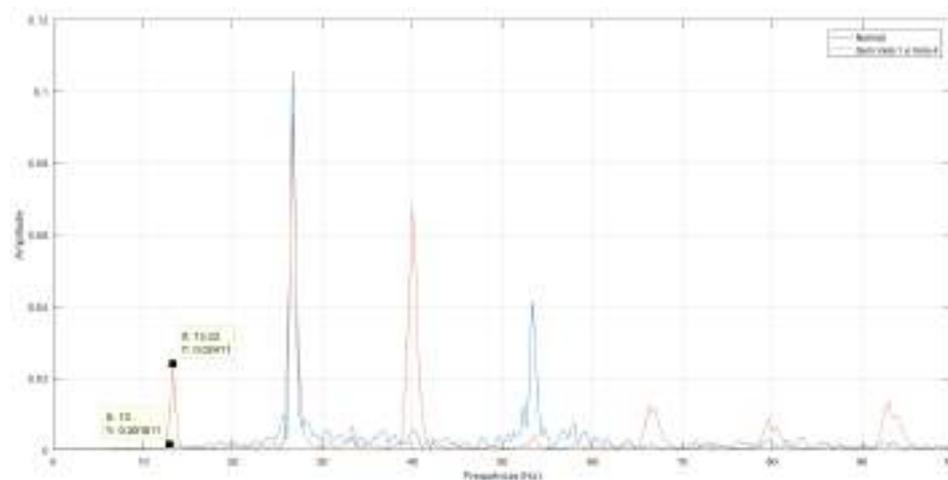
Fonte: Autora

O mesmo processo foi repetido para a desconexão do par de velas, sendo o primeiro par vela 1 e vela 4, que se situam nas extremidades, e o segundo par vela 2 e vela 3, situadas no meio do motor.

Os sinais obtidos pela desconexão dos dois pares de velas são mostrados nas Figuras 13 e 14, onde pode ser percebido que este comportamento apresenta apenas um pico de

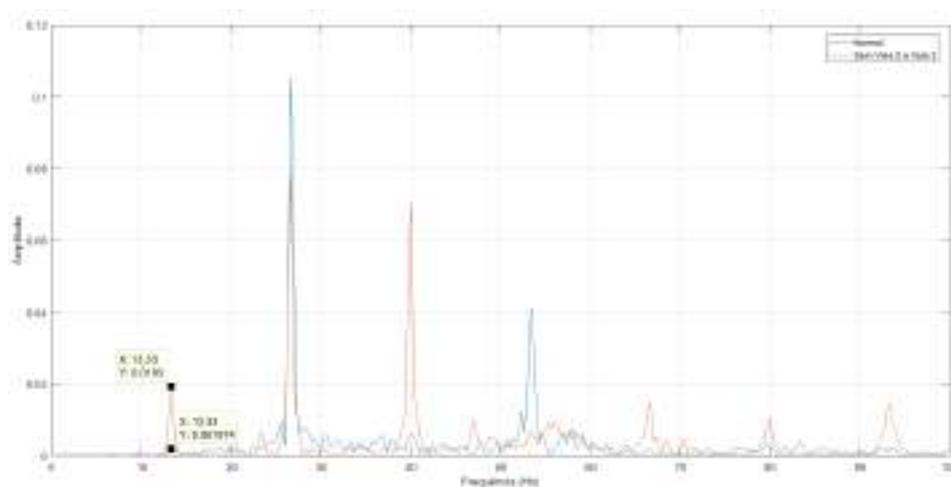
frequência diferente da frequência fundamental em uma distância de aproximadamente seu próprio valor.

Figura 13 – FFT do sinal adquirido para acionamento sem o par de vela 1 e 4



Fonte: Autora

Figura 14 – FFT do sinal adquirido para acionamento sem o par de vela 2 e 3



Fonte: Autora

3.4 Obtenção do Parâmetro de Falha

São observados dois intervalos provenientes da FFT (0 Hz – 15 Hz e 20 Hz – 25 Hz), cujos valores de pico encontrados são utilizados no cálculo da densidade espectral de

potência (PSD - *Power Spectral Density*), conforme a Equação 3.1 (HOWARD, 2004):

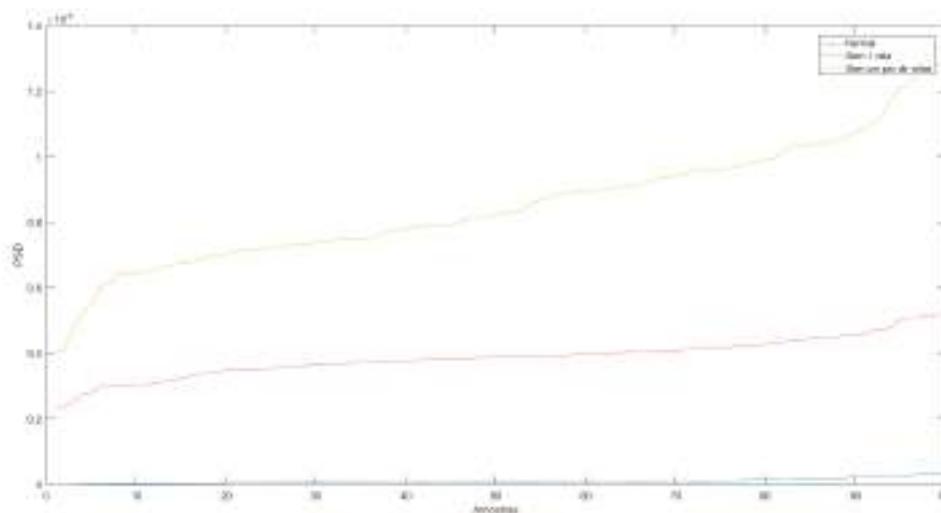
$$PSD = \frac{G_{rms}^2}{f} \quad (3.1)$$

onde G_{rms} é o valor eficaz da FFT do sinal de áudio em uma dada frequência f .

As densidades espectrais de potência para os picos localizados nos intervalos de frequência observados, que foram utilizadas como parâmetro de classificação, são mostradas nas Figura 15 e 16, respectivamente. As situações de desconexão de uma vela e de um par de velas apresentaram densidades espectrais semelhantes para o primeiro pico, já no comportamento normal a densidade espectral foi quase nula.

Para o segundo pico as densidades espectrais de potência se mostraram semelhantes para o sinal normal e o sinal com acionamento sem um par de velas, também quase nulos. O sinal para acionamento sem uma das velas apresentou uma PSD relativamente maior.

Figura 15 – Densidade espectral de potência para o pico em 13,33 Hz

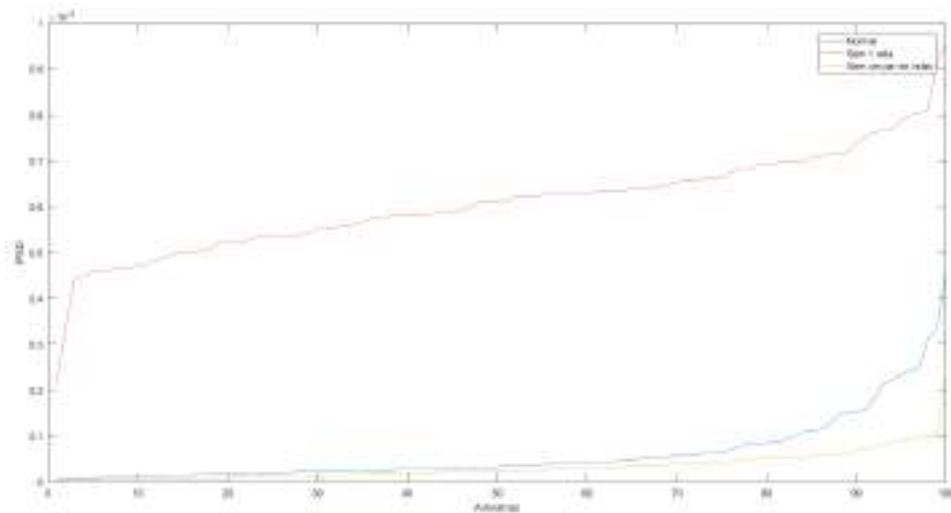


Fonte: Autora

Para o pico da frequência fundamental, as densidades encontradas foram muito próximas, como pode ser observado na Figura 17, e por isso foram descartadas para classificação da falha.

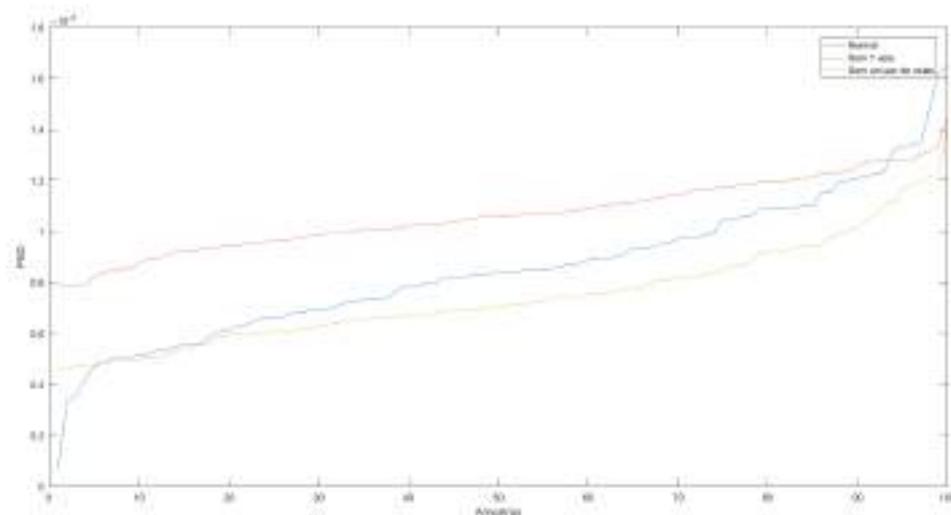
Com os resultados obtidos foi possível encontrar um parâmetro determinante para classificação do comportamento normal e do comportamento com falhas.

Figura 16 – Densidade espectral de potência para o pico em 20 Hz



Fonte: Autora

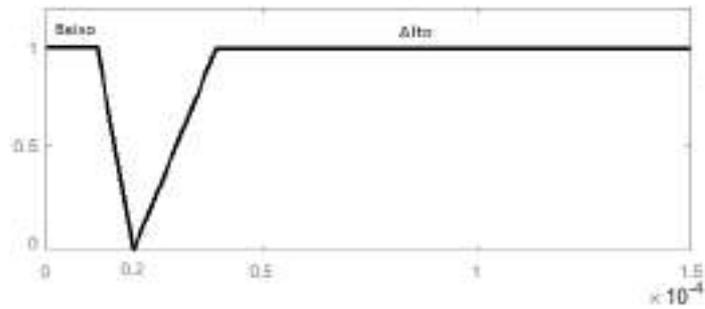
Figura 17 – Densidade espectral de potência para o pico na frequência fundamental



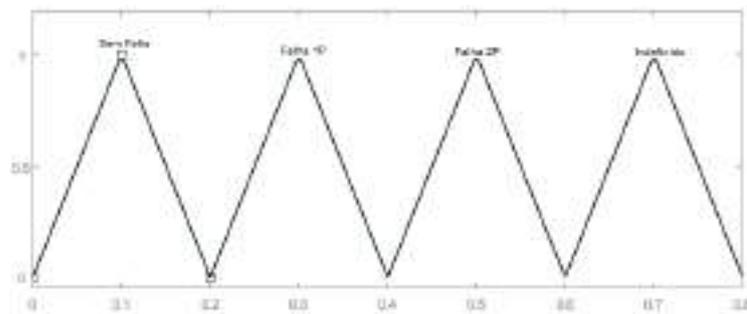
Fonte: Autora

3.5 Classificação das Falhas

Um sistema de classificação baseado em lógica *fuzzy* é utilizado nessa etapa, cujo resultado contém informação da presença ou não de falta, bem como a severidade da mesma. Para isto foram criados 2 conjuntos de entrada *fuzzy* idênticos chamados Pico 1 e Pico 2, contendo 2 subconjuntos cada, e 1 conjunto de saída, com 4 subconjuntos, chamado Diagnóstico. O conjunto de entrada é mostrado na Figura 18 e o de saída na Figura 19.

Figura 18 – Configuração dos conjuntos de entrada da lógica *fuzzy*

Fonte: Autora

Figura 19 – Conjunto de saída da lógica *fuzzy*

Fonte: Autora

Ao analisar os valores de PSD para os dois picos, é possível observar que eles se agrupam em dois intervalos de valores distintos. A função de entrada *Baixo* refere-se aos valores de PSD abaixo do limiar estipulado de 0.2×10^{-4} , já a função de entrada *Alto* retrata os valores que estão acima do limiar estipulado. O propósito do sistema é fornecer um diagnóstico sobre o estado de funcionamento do motor, por isso as variáveis de saída se remetem a um dos estados possíveis do motor.

Com um limiar criado por *fuzzy*, a saída do sistema de diagnóstico não é apenas uma falha ou não uma falha, mas um grau relativo de como é provável que uma falha tenha ocorrido (MA et al., 2015). Baseado nisso, as seguintes regras foram elaboradas baseadas na observação dos gráficos:

- **Regra 1:** Se **Pico 1** é *Baixo* e **Pico 2** é *Baixo* então **Diagnóstico** é *Sem Falha*;
- **Regra 2:** Se **Pico 1** é *Alto* e **Pico 2** é *Alto* então **Diagnóstico** é *Falha 1P*;
- **Regra 3:** Se **Pico 1** é *Alto* e **Pico 2** é *Baixo* então **Diagnóstico** é *Falha 2P*;
- **Regra 4:** Se **Pico 1** é *Baixo* e **Pico 2** é *Alto* então **Diagnóstico** é *Indefinido*.

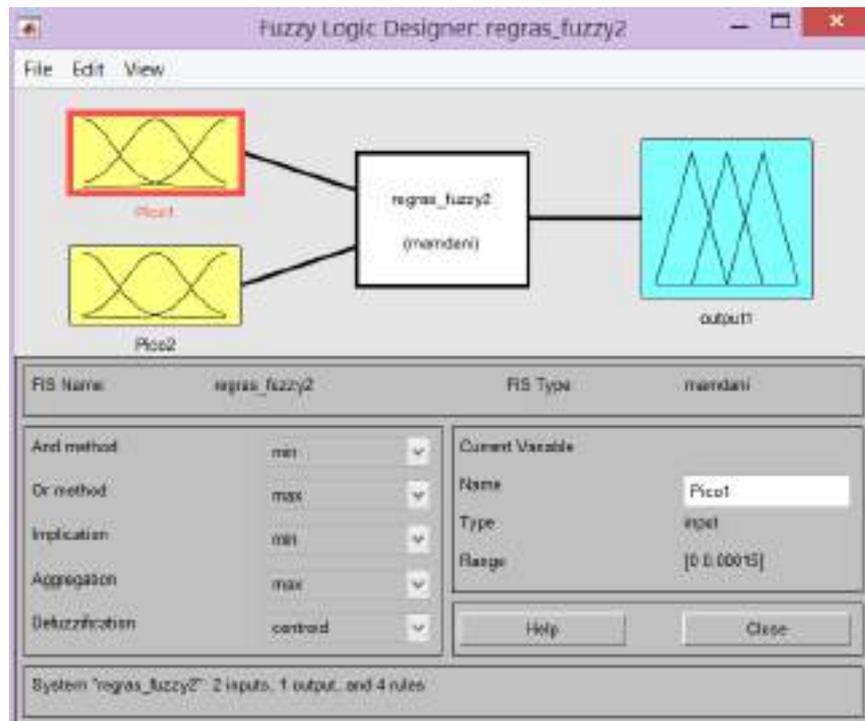
Para implementação da lógica *fuzzy* no Arduino foi utilizada uma biblioteca uma biblioteca escrita em C++ que auxilia na implementação dessa lógica em sistemas embarcados, criada pelo *Robotic Research Group* (RRG) na Universidade Estadual do Piauí (UESPI-Teresina). A sua implementação é simples e direta, de não possuir limitações explícitas de quantidade de conjuntos *fuzzy*, regras *fuzzy*, entradas ou saídas, estas limitam-se a capacidade de processamento e armazenamento de cada microcontrolador (KRIDI et al., 2013).

A criação das regras *fuzzy* na biblioteca mencionada é feita adicionando um antecedente para cada consequência, isto é, montar a expressão condicional do antecedente de uma regra *fuzzy* do tipo SE-ENTÃO com o comando *new FuzzyRuleAntecedent*. As condições podem ser agrupadas com *JoinSingle*, para condições únicas, *JoinWithAND*, para condições que devem ocorrer juntas, e *JoinWithOR*, para condições que é suficiente apenas uma delas ocorrer. A consequência para as condições impostas na primeira parte do código é adicionada com o comando *new FuzzyRuleConsequent*. Os dois argumentos anteriores são relacionado pela criação de uma nova regra com o comando *new FuzzyRule*. Um exemplo para a criação da regra para diagnóstico da falha sem uma vela é mostrado a seguir:

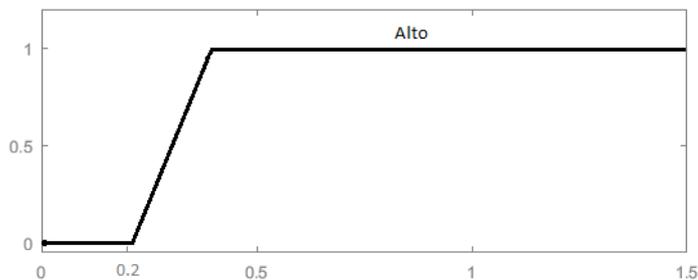
```
1 FuzzyRuleAntecedent* ifPicoUmAltoAndPicoDoisAlto = new
   FuzzyRuleAntecedent();
2 ifPicoUmAltoAndPicoDoisAlto->joinWithAND(alto, alto2);
3 FuzzyRuleConsequent* thenDiagnosticoFaltaUmP = new
   FuzzyRuleConsequent();
4 thenDiagnosticoFaltaUmP->addOutput(faltaump);
5 FuzzyRule* fuzzyRule02 = new FuzzyRule(2,
   ifPicoUmAltoAndPicoDoisAlto, thenDiagnosticoFaltaUmP);
```

Com o intuito de validar o sistema *fuzzy* implementado no Arduino, o mesmo sistema foi feito no Matlab pelo *toolbox* de lógica *fuzzy*. O *Fuzzy Logic Designer* é a ferramenta principal de para implementação do sistema *fuzzy* no *Matlab*, nele são determinadas as quantidades de entradas e saídas, quantas e quais funções de pertinência serão usadas em cada variável, a faixa de valor para cada função de pertinência, as regras do sistema, o tipo de inferência, entre outras coisas. Uma visão geral dessa ferramenta é mostrada na Figura 20.

Houve uma diferença nos conjuntos de entrada feitos no *Matlab* e no Arduino. Por não existir a opção de criar uma regra baseada no fato de um valor **não** pertencer a um conjunto, houve a necessidade de criação de 2 subconjuntos nos conjuntos de entrada, no *Matlab* isso não foi necessário e apenas um subconjunto foi criado para os conjuntos de entrada, como ilustrado na Figura 21. Os conjuntos de saídas permaneceram iguais para as duas implementações.

Figura 20 – *Toolbox* para implementação *fuzzy* no *Matlab*

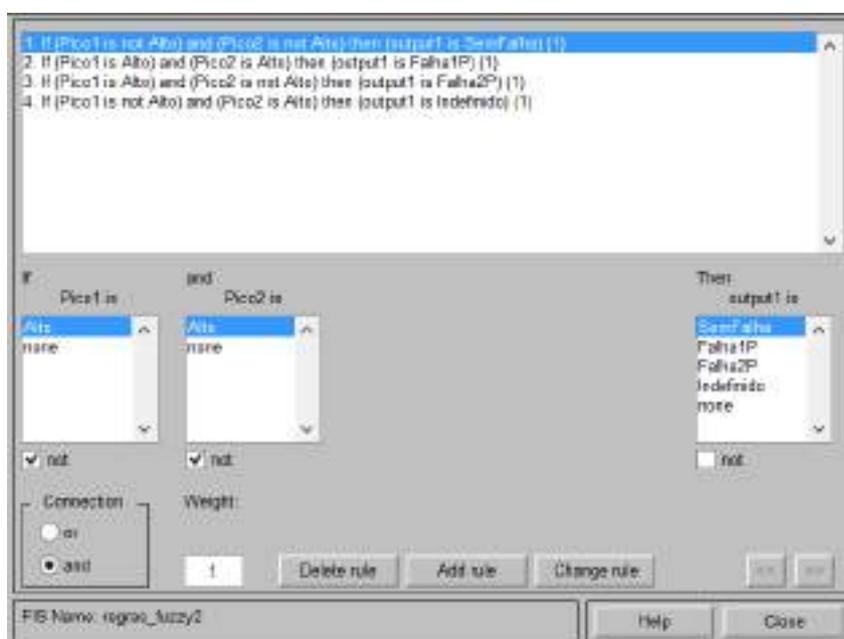
Fonte: Autora

Figura 21 – Configuração dos conjuntos de entrada da lógica *fuzzy* no *Matlab*

Fonte: Autora

O *rule editor* permite criar e editar as regras que definem o comportamento do sistema. As regras do sistema *fuzzy* para implementação no *Matlab* foram modificadas e estão ilustradas na Figura 22 com o *rule editor*, com a diferença que para essa aplicação é considerado se um valor **não** pertence a um subconjunto ao invés de se pertence a um ou a outro subconjunto para classificação.

Figura 22 – Regras da lógica *fuzzy* para implementação no *Matlab*



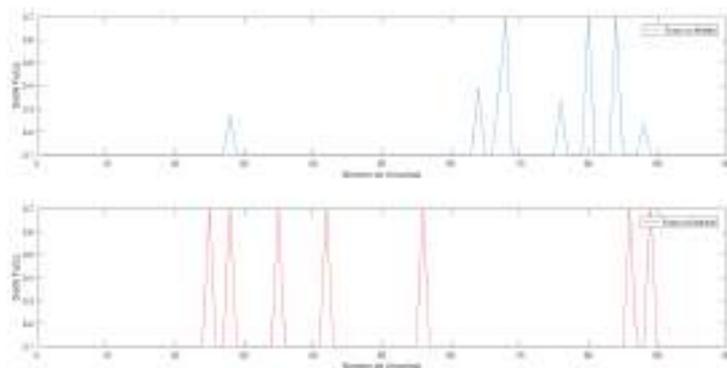
Fonte: Autora

4 Resultados

As aquisições dos sinais apresentados à rede *fuzzy* como entrada foram gravados em dias diferentes, todas com 1 minuto de duração. Foram inicialmente testados diferentes intervalos de tempo para as amostras a serem analisadas e foi determinado que 1s de áudio seria suficiente para detecção da falha. Conforme o funcionamento do algoritmo da FFT, o vetor submetido ao processo necessita conter um total de 2^n pontos. Dessa forma, o valor mais próximo do total de amostras presentes em 1s e que atendesse aos parâmetros de exigência da FFT foi de 65536 (2^{16}) pontos. Dessas amostras, 300 foram escolhidas aleatoriamente para cálculo da densidade espectral de potência e para saber qual o resultado do diagnóstico, elas foram subdivididos da seguinte forma: 100 para o comportamento normal, 100 para a falha sem 1 vela e 100 para a falha sem 1 par de velas, apenas 0,46% do número total de amostras por segundo foram necessárias para um diagnóstico correto da falha. Isso indica que a aquisição de um áudio mais curto já seria suficiente para a detecção e classificação.

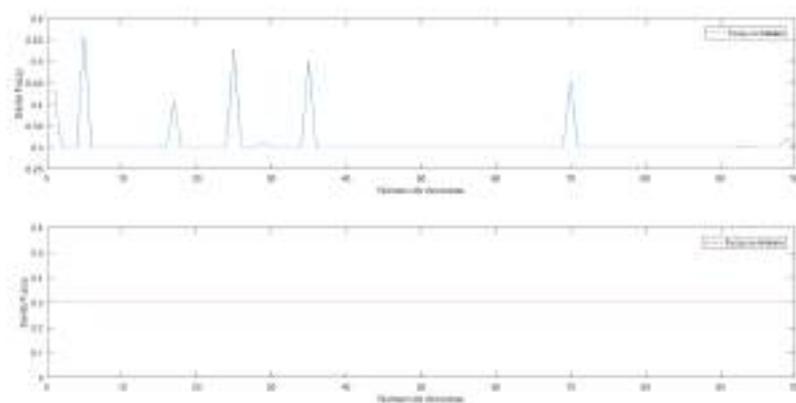
As Figuras 23, 24 e 25 mostram a comparação da saída do sistema *fuzzy* implementado no *Matlab* e no Arduino para o comportamento normal, comportamento sem uma vela e sem um par de velas, respectivamente. É possível observar que a performance do sistema no sistema embarcado teve uma taxa maior de acertos do que quando implementado no *Matlab*. Em ambos os casos, a situação que se apresenta mais problemática é a de comportamento normal, onde se apresenta mais erros na forma de classificação da falha como indefinida.

Figura 23 – Comparação entre o sistema *fuzzy* no Arduino e no *Matlab* para comportamento normal



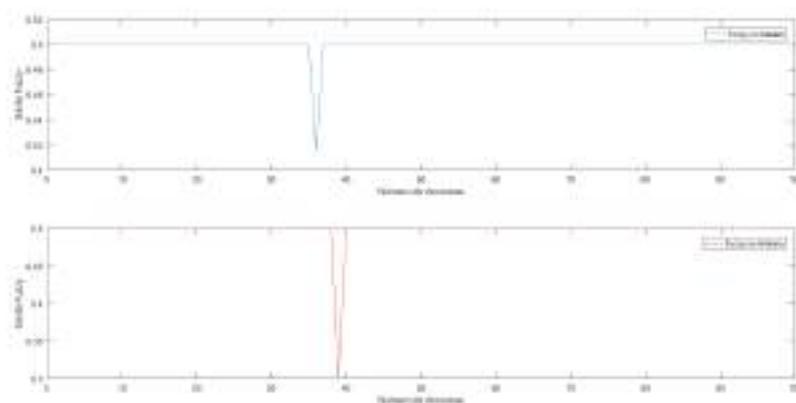
Fonte: Autora

Figura 24 – Comparação entre o sistema *fuzzy* no Arduino e no *Matlab* para acionamento sem 1 vela



Fonte: Autora

Figura 25 – Comparação entre o sistema *fuzzy* no Arduino e no *Matlab* para acionamento sem 1 par de velas



Fonte: Autora

Para melhor visualização dos resultados *fuzzy* implementado no Arduino e no *Matlab*, os dados de saída foram expostos na Tabela 3 e 4, respectivamente. Reiterando o que foi observado por imagens, a situação que apresentou uma porcentagem menor de acerto foi para o comportamento sem falha. Isso acontece pois há uma sobreposição dos valores de PSD para o comportamento normal e o comportamento com acionamento sem 1 vela para o segundo pico que pode causar o ativamento da 4ª regra do sistema *fuzzy*. Com uma taxa média de acerto de 97,75%, a lógica *fuzzy* no Arduino é considerada satisfatória.

Tabela 3 – Matriz de confusão do sistema *fuzzy* implementado no Arduino

	Sem falta	Falta 1P	Falta 2P	Indefinido	Taxa de Acerto (%)
Sem Falta	93	4	0	3	93%
Falta 1P	0	100	0	0	100%
Falta 2P	1	0	99	0	99%
Indefinido	0	1	0	99	99%
Performance geral:					97,75%

Fonte: Produzida pela autora

Tabela 4 – Matriz de confusão do sistema *fuzzy* implementado no *Matlab*

	Sem falta	Falta 1P	Falta 2P	Indefinido	Taxa de Acerto (%)
Sem Falta	93	0	0	7	93%
Falta 1P	0	100	0	0	95%
Falta 2P	1	0	99	0	99%
Indefinido	0	1	0	99	99%
Performance geral:					96,5%

Fonte: Produzida pela autora

5 Conclusão

O presente trabalho tratou do desenvolvimento de um algoritmo inteligente para diagnóstico da falha de *misfire* em veículos automotivos equipados com motores de combustão interna. Conforme observado na literatura, a proposta apresenta relevância por se tratar de uma técnica não invasiva, o que a difere da maioria dos trabalhos pesquisados, que utilizam métodos que exigem o contato direto com o objeto a ser diagnosticado.

Os resultados demonstram que padrões de funcionamento normal e de falhas podem ser catalogados para as condições estudadas, o que consiste em um forte indício de que tal situação possa se repetir para outras falhas de natureza cíclica. A mesma metodologia aplicada no decorrer do trabalho pode ser expandida e abordar a ocorrência de outras falhas em condição simultânea, bem como falhas em regime dinâmico.

O algoritmo de classificação proposto integrado com sucesso ao sistema embarcado utilizado para aquisição sonora, constituindo-se assim em uma melhoria significativa no dispositivo final de diagnóstico. A taxa de acerto para classificação atingiu um total de 97,77%, considerado um bom desempenho dentro das limitações do que foi proposto.

Como continuidade do trabalho podem ser integradas novas funcionalidades, como a utilização de outras técnicas de processamento de sinais, como a transformada de *Wavelet*, ou a utilização de outras ferramentas de classificação, como redes neurais artificiais ou classificadores KNN. Além disso, um refinamento do que foi proposto pode levar ao desenvolvimento de aplicações móveis, o que atingiria um patamar comercial ao se levar em conta a área de estudo e o alcance populacional e capacidades de memória e processamento dos *smartphones*.

Referências

- ADDISON, P. S. *The illustrated wavelet transform handbook: introductory theory and applications in science, engineering, medicine and finance*. [S.l.]: CRC press, 2017. Citado na página 22.
- AL-TAEE, M. A.; KHADER, O. B.; AL-SABER, N. A. Remote monitoring of vehicle diagnostics and location using a smart box with global positioning system and general packet radio service. In: IEEE. *Computer Systems and Applications, 2007. AICCSA '07. IEEE/ACS International Conference on*. [S.l.], 2007. p. 385–388. Citado na página 13.
- ANAMI, B. S.; PAGI, V. B. Acoustic signal-based approach for fault detection in motorcycles using chaincode of the pseudospectrum and dynamic time warping classifier. *IET Intelligent Transport Systems*, IET, v. 8, n. 1, p. 21–27, 2014. Citado na página 14.
- AONO, T.; FUKUCHI, E. Misfire detection method robust against crankshaft vibration and acceleration. In: IEEE. *Control Applications, 2005. CCA 2005. Proceedings of 2005 IEEE Conference on*. [S.l.], 2005. p. 1218–1221. Citado 2 vezes nas páginas 12 e 14.
- ARABACI, H.; BILGIN, O. Neural network classification and diagnosis of broken rotor bar faults by means of short time fourier transform. In: *Proceedings of the International MultiConference of Engineers and Computer Scientists*. [S.l.: s.n.], 2009. v. 1. Citado 2 vezes nas páginas 21 e 22.
- BARKAI, J. *Vehicle diagnostics-are you ready for the challenge?* [S.l.], 2001. Citado na página 13.
- BARR, A.; FEIGENBAUM, E.; ROADS, C. *The Handbook of Artificial Intelligence, Volume 1*. [S.l.]: JSTOR, 1982. Citado na página 23.
- BELAOUT, A.; KRIM, F.; MELLIT, A. Neuro-fuzzy classifier for fault detection and classification in photovoltaic module. In: IEEE. *8th International Conference on Modelling, Identification and Control (ICMIC), 2016*. [S.l.], 2016. p. 144–149. Citado na página 25.
- BELO, V. P. *Sistema para Diagnóstico Automático de Falhas em Veículos Automotores OBD-2*. Dissertação (Mestrado) — Universidade Federal de Minas Gerais, Belo Horizonte, jul. 2003. Citado 2 vezes nas páginas 11 e 13.
- BRASIL. *CONAMA: Decreto n. 354, de 13 de dez. de 2004*. 2004. Citado na página 13.
- CAMPOS, M. M. de; SAITO, K. *Sistemas inteligentes em controle e automação de processos*. [S.l.]: Ciência Moderna, 2004. Citado 4 vezes nas páginas 23, 24, 25 e 26.
- CHOU, P. B.-L. et al. *System and method for vehicle diagnostics and health monitoring*. [S.l.]: Google Patents, 2001. US Patent 6,330,499. Citado na página 13.
- COOLEY, J. W.; TUKEY, J. W. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, v. 19, n. 90, p. 297–301, 1965. Citado na página 29.

- DHS, DEPARTMENT OF HOMELAND SECURITY. *What Every Member of the Trade Community Should Know About: Internal Combustion Piston Engines*. EUA, 2012. Citado na página 16.
- FORD. *Motor 1.6L Zetec Rocam: EcoSport e Fiest*. [S.l.], 2005. Citado na página 27.
- HAYKIN, S.; VEEN, B. V. *Signals and systems*. [S.l.]: John Wiley & Sons, 2007. Citado na página 18.
- HEYWOOD, J. B. *Internal combustion engine fundamentals*. Mcgraw-hill New York, 1988. Citado 4 vezes nas páginas 12, 16, 17 e 18.
- HOWARD, R. M. *Principles of random signal analysis and low noise design: The power spectral density and its applications*. [S.l.]: John Wiley & Sons, 2004. Citado na página 33.
- KRIDI, D. S. et al. Desenvolvimento de uma biblioteca fuzzy para o controle autônomo de um robô móvel em ambiente desconhecido. *Mostra Nacional de Robótica*, v. 1, 2013. Citado na página 36.
- LATHI, B. P. *Linear systems and signals*. [S.l.]: Oxford University Press New York., 2005. v. 2. Citado 2 vezes nas páginas 19 e 20.
- LATHI, B. P.; GREEN, R. A. *Essentials of digital signal processing*. [S.l.]: Cambridge University Press, 2014. Citado na página 20.
- LEE, A.; LOH, R.; WU, J. Automotive engine misfire detection using kalman filtering. In: IEEE. *IEEE 58th Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003*. [S.l.], 2003. v. 5, p. 3377–3381. Citado na página 14.
- LEWIS, J. *Arduino Comparison Chart*. 2012. Disponível em: <<https://www.baldengineer.com/arduino-comparison-chart.html>>. Acesso em: 22 mai 2018. Citado na página 28.
- LIMA, T. L. d. V. *Desenvolvimento de um sistema embarcado para detecção de falhas em veículos automotivos por meio de análise sonora*. Tese (Relatório de Qualificação de Doutorado) — Universidade Federal da Paraíba, João Pessoa, nov. 2017. Citado 2 vezes nas páginas 11 e 29.
- LOWREY, L. H. et al. *Internet-based vehicle-diagnostic system*. [S.l.]: Google Patents, 2003. US Patent 6,611,740. Citado na página 13.
- MA, X. et al. *Combined frequency domain analysis and fuzzy logic for engine misfire diagnosis*. [S.l.], 2015. Citado 2 vezes nas páginas 14 e 35.
- MAGALHÃES, H. d. O. *Análise de sinais para engenheiros: Uma abordagem via Wavelet*. [S.l.]: Rio de Janeiro: Brasport, 2007. Citado 2 vezes nas páginas 19 e 28.
- MCKEOWN, M. Fft implementation on the tms320vc5505, tms320c5505, and tms320c5515 dsps. *Application report. Texas instruments, (SPRABB6A)*, 2010. Citado 2 vezes nas páginas 29 e 30.
- MENEZES, A. S. *Uma contribuição à análise espectral de sinais estacionários e não estacionários*. Tese (Doutorado) — Universidade Federal de Juiz de Fora, Juiz de Fora, set. 2014. Citado 2 vezes nas páginas 18 e 20.

- MITRA, S. K.; KUO, Y. *Digital signal processing: a computer-based approach*. [S.l.]: McGraw-Hill Higher Education New York, 2006. v. 2. Citado na página 20.
- MONTANI, M.; SPECIALE, N. Multiple misfire identification by a wavelet-based analysis of crankshaft speed fluctuation. In: IEEE. *IEEE International Symposium on Signal Processing and Information Technology, 2006*. [S.l.], 2006. p. 144–148. Citado na página 14.
- NAIK, S. Advanced misfire detection using adaptive signal processing. *International Journal of Adaptive Control and Signal Processing*, Wiley Online Library, v. 18, n. 2, p. 181–198, 2004. Citado na página 14.
- PRESS, W. H. et al. *Numerical recipes in C*. [S.l.]: Cambridge university press Cambridge, 1996. v. 2. Citado na página 29.
- REZENDE, S. O. *Sistemas inteligentes: fundamentos e aplicações*. [S.l.]: Editora Manole Ltda, 2003. Citado 3 vezes nas páginas 24, 25 e 26.
- RIVEST, R. L.; LEISERSON, C. E. *Introduction to algorithms*. [S.l.]: McGraw-Hill, Inc., 1990. Citado na página 29.
- RUSSELL, S. J.; NORVIG, P. *Inteligencia Artificial: un enfoque moderno*. [S.l.: s.n.], 2004. Citado na página 23.
- SANTIAGO, D. F. d. A.; PEDERIVA, R. Influência da resolução tempo-frequência da wavelet de morlet no diagnóstico de falhas de máquinas rotativas. *Mecânica Computacional*, v. 22, p. 2538–2550, 2003. Citado na página 18.
- SHEN, Y.; LUO, J. Design and implementation of vehicle fault diagnosis system for intelligent travel. In: IEEE. *Cloud Computing and Intelligence Systems (CCIS), 2014 IEEE 3rd International Conference on*. [S.l.], 2014. p. 553–558. Citado na página 11.
- SIMÕES, M. G.; SHAW, I. S. Controle e modelagem fuzzy. *Blucher: FAPESP, São Paulo, BRA*, 2007. Citado 2 vezes nas páginas 24 e 25.
- SMITH, S. W. *The scientist and engineer's guide to digital signal processing*. [S.l.]: California Technical Pub. San Diego, 1997. Citado na página 19.
- SORENSEN, H. V.; BURRUS, C. S. Efficient computation of the short-time fast fourier transform. In: IEEE. *International Conference on Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988*. [S.l.], 1988. p. 1894–1897. Citado na página 21.
- STEIN, J. Y. *A Computer Science Perspective*. [S.l.]: Wiley, 2000. Citado 3 vezes nas páginas 21, 22 e 23.
- TURING, A. Mind. *Mind*, v. 59, n. 236, p. 433–460, 1950. Citado na página 23.
- WU, C. *Thermodynamic cycles: computer-aided design and optimization*. [S.l.]: CRC Press, 2003. Citado na página 16.
- WU, J.-D. et al. Fault classification of a scooter engine platform using wavelet transform and artificial neural network. In: CITESEER. *Proceedings of the International MultiConference of Engineers and Computer Scientists*. [S.l.], 2009. v. 1, p. 18–20. Citado na página 14.

- WU, Z. J.; NAIK, S. M. Dsp applications in engine control and onboard diagnostics: Enabling greener automobiles. *IEEE Signal Processing Magazine*, IEEE, v. 34, n. 2, p. 70–81, 2017. Citado 2 vezes nas páginas 11 e 12.
- YEAP, Y. M.; UKIL, A. Fault detection in hvdc system using short time fourier transform. In: IEEE. *Power and Energy Society General Meeting (PESGM), 2016*. [S.l.], 2016. p. 1–5. Citado na página 22.
- YINHUI, W. et al. Advanced gasoline engine misfire diagnostic method based on crankshaft speed multiple filtering. In: IEEE. *International Conference on Electric Information and Control Engineering (ICEICE), 2011*. [S.l.], 2011. p. 1964–1968. Citado 2 vezes nas páginas 12 e 14.
- ZADEH, L. A. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on systems, Man, and Cybernetics*, IEEE, n. 1, p. 28–44, 1973. Citado na página 24.