

Stephany Padilha Guimarães

Algoritmo de Identificação de Sinais Intermitentes no Espectro Radioelétrico

São José - SC

Dezembro, 2017

Stephany Padilha Guimarães

Algoritmo de Identificação de Sinais Intermitentes no Espectro Radioelétrico

Proposta de trabalho de conclusão de curso apresentada à Coordenação de Engenharia de Telecomunicações do Instituto Federal de Santa Catarina para a obtenção do diploma Bacharel em Engenharia de Telecomunicações.

Instituto Federal de Santa Catarina – IFSC

Campus São José

Engenharia de Telecomunicações

Orientador: Mário de Noronha Neto

Coorientador: Douglas Amorim Ferreira

São José - SC

Dezembro, 2017

Guimarães, Stephany Padilha

Algoritmo de Identificação de Sinais Intermitentes no Espectro Radioelétrico/
Stephany Padilha Guimarães. – São José - SC, Dezembro, 2017-
56 p.

Orientador: Mário de Noronha Neto Coorientador: Douglas Amorim Ferreira

Monografia (Graduação) – Instituto Federal de Santa Catarina – IFSC
Campus São José
Engenharia de Telecomunicações, Dezembro, 2017.

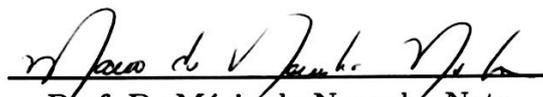
1. Atividades Intermitentes. 2. Rádio Definido por Software. 3. Monitoramento de Espectro. 4. GNU Radio. 5. Detecção por Energia I. Neto, Mario de Noronha. II. Ferreira, Douglas Amorim. III. Instituto Federal de Santa Catarina. IV. Campus São José. V. Graduação em Engenharia de Telecomunicações

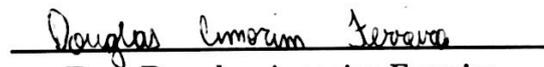
Stephany Padilha Guimarães

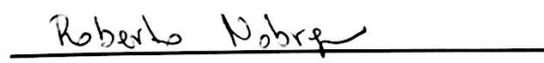
Algoritmo de Identificação de Sinais Intermitentes no Espectro Radioelétrico

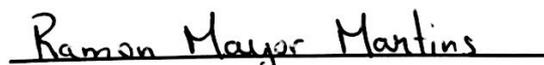
Proposta de trabalho de conclusão de curso apresentada à Coordenação de Engenharia de Telecomunicações do Instituto Federal de Santa Catarina para a obtenção do diploma Bacharel em Engenharia de Telecomunicações.

Trabalho aprovado. São José - SC. 17 de Dezembro de 2017:


Prof. Dr. Mário de Noronha Neto
Orientador
Instituto Federal de Santa Catarina


Eng. Douglas Amorim Ferreira
Coorientador
Dígitro Tecnologia S.A.


Prof. Dr. Roberto Wanderley da Nóbrega
Instituto Federal de Santa Catarina


Prof. Me. Ramon Mayor Martins
Instituto Federal de Santa Catarina

São José - SC
Dezembro, 2017

Agradecimentos

Primeiramente gostaria de agradecer a Deus, pela força sem explicação para continuar nos momentos difíceis e por deixar tudo acontecer nos momentos certos.

Agradeço à minha família por todo o apoio quando decidi fazer este curso e o suporte diante às dificuldades que enfrentei nos últimos 5 anos. Em especial aos meu pais e meu tios, Gedalva, Carmem, Genival e Emerson, que me tiveram como uma filha e sempre deram todo o suporte durante esse período.

Ao meu namorado Gabriel, que mesmo estando a pouco tempo em minha vida foi essencial para não me deixar desistir ao fim dessa caminhada.

A empresa Dígitro Tecnologia S.A. pelo estágio engrandecedor e a oportunidade de desenvolvimento deste trabalho. Também aos colegas de equipe que tive o prazer de conviver na empresa, estes me ensinaram muitas coisas, tanto para minha vida pessoal e profissional.

Ao engenheiro Sergio Schmiegelow por compartilhar todo seu conhecimento em diversas situações e fazer com que coisas absurdas pareçam simples.

Em especial ao meu coorientador Douglas Amorim Ferreira pelos ensinamentos e preocupação com o meu futuro, muitas vezes mais do que eu mesma.

Ao meu orientador Mario de Noronha Neto, por todas as sugestões e por não me deixar desistir da entrega deste trabalho, mesmo eu tendo decidido que isso era o certo a fazer.

E por fim, meus amigos por me incentivarem e acreditarem em mim até mesmo quando eu deixei de acreditar.

Resumo

Este trabalho de conclusão de curso apresenta a implementação de um algoritmo de detecção de atividades intermitentes no espectro radioelétrico. Para teste e utilização do algoritmo foi implementado um sistema composto por um periférico de aquisição, que utiliza rádio definido por software, e uma unidade de processamento de sinais. Esse sistema adquire sinais de rádio e faz o condicionamento necessário para possibilitar detecção de atividades intermitentes pelo algoritmo implementado. O algoritmo reconhece quando há intermitências nos sinais adquiridos através de uma variação do algoritmo de detecção por energia. A avaliação dos resultados do algoritmo acontece em diferentes faixas de frequência. Ela é realizada através da comparação entre os dados obtidos e as características do espectrograma de frequências (*waterfall*) dos sinais analisados.

Palavras-chave: Atividades Intermitentes. Rádio Definido por Software. Monitoramento de Espectro. GNU Radio. Detecção por Energia.

Abstract

This monograph presents the implementation of an algorithm for detecting intermittent activities in the radioelectric spectrum. For the test and use of the algorithm a system was composed of an acquisition peripheral, with software defined radio technology, and a signal processing unit. This system acquires radio signals and makes the necessary conditioning to enable detection of intermittent activities by the implemented algorithm. The algorithm recognizes when there are flashes on the acquired signals through a variation of the energy detection algorithm. The evaluation of the results of the algorithm happens in different frequency bands. It is performed by comparing the obtained data with the characteristics of the frequency spectrogram (waterfall) of the analyzed signals.

Keywords: Intermittent Activities. Software Defined Radio. Spectrum Monitoring. GNU-Radio. Power Detection.

Lista de ilustrações

Figura 1 – Diagrama do sistema das etapas do trabalho.	23
Figura 2 – Espectrograma de frequência de uma transmissão intermitente.(AGILENT TECHNOLOGIES, 2012)	25
Figura 3 – Diagrama de blocos de um SDR ideal.	27
Figura 4 – Diagrama de blocos de um SDR real.	27
Figura 5 – Representação um <i>flowgraph</i>	28
Figura 6 – Interface GNU Radio Companion com <i>flowgraph</i> para recepção <i>Narrowband Frequency Modulation</i> (NBFM).	29
Figura 7 – Diagrama de blocos da detecção por filtro casado.	32
Figura 8 – Diagrama de blocos da detecção por filtro casado.	33
Figura 9 – Ilustração de escolha incorreta de limiares de detecção	34
Figura 10 – Diagrama de blocos do sistema proposto.	35
Figura 11 – Diagrama de blocos ampliado da etapa de aquisição de amostras.	35
Figura 12 – <i>Universal Software Radio Peripheral</i> (USRP) N210 comercializado pela <i>Ettus</i> (ETTUS, 2017a).	36
Figura 13 – <i>Hardware</i> USRP N210	37
Figura 14 – Representação gráfica da equação 3.7 para surgimento de atividade.	41
Figura 15 – Máquina de estados do sinal Sd_k	41
Figura 16 – Gráficos dos estados possíveis de Sd_k	42
Figura 17 – Sinais Sd_k com limiares ideais para detecção de intermitências.	43
Figura 18 – Espectro de frequências <i>waterfall</i>	45
Figura 19 – Espectrograma <i>waterfall</i> do modelo de teste 1 com sinal gerado no GNU Radio	46
Figura 20 – Espectrograma <i>waterfall</i> do modelo de teste 2 com sinal gerado no GNU Radio	46
Figura 21 – Espectrograma <i>waterfall</i> do modelo de teste 3 com sinal gerado no GNU Radio	47
Figura 22 – Espectrograma <i>waterfall</i> do modelo de teste 4 com sinal gerado no GNU Radio	47
Figura 23 – Espectrograma <i>waterfall</i> do modelo de teste 5 com sinal gerado no GNU Radio	47
Figura 24 – Espectrograma <i>waterfall</i> do teste 6 com sinal gerado em 467 MHz	49
Figura 25 – Espectrograma <i>waterfall</i> do teste 7 com sinal gerado 462 MHz	49
Figura 26 – Espectrograma <i>waterfall</i> do teste 8 com sinal gerado 462 MHz	50
Figura 27 – Espectrograma <i>waterfall</i> do teste 9 com sinal gerado 459.3 MHz	50
Figura 28 – Espectrograma <i>waterfall</i> do teste 10 em 459.3 MHz	50

Figura 29 – Espectrograma <i>waterfall</i> do teste 11 em 859 MHz	51
Figura 30 – Espectrograma <i>waterfall</i> do teste 12 em 859 MHz	51
Figura 31 – Espectrograma <i>waterfall</i> do teste 13 em 859 MHz	51

Lista de tabelas

Tabela 1 – Opções de USRP disponíveis (ETTUS, 2017b)	30
Tabela 2 – Relação de placas filhas compatíveis com o USRP N210.	37
Tabela 3 – Resultados dos testes com sinais gerados.	48

Lista de abreviaturas e siglas

ADC *Analogic-Digital Converter*

ANATEL Agência Nacional de Telecomunicações

DAC *Digital-Analogic Converter*

DDC *Digital Down Converter*

DEP Densidade Espectral de Potência

DSP *Digital Signal Processing*

DUC *Digital Up Converter*

FCC *Federal Communications Commission*

FRS *Family Radio Service*

DFT *Discrete Fourier Transform*

FPGA *Field Programmable Gate Array*

FRS *Family Radio Service*

GPL *General Public License*

HF *High Frequency*

IFSC Instituto Federal de Santa Catarina

NBFM *Narrowband Frequency Modulation*

SDR *Software Defined Radio*

SWIG *Simplified Wrapper and Interface Generator*

TCC Trabalho de Conclusão de Curso

UHF *Ultra High Frequency*

USRP *Universal Software Radio Peripheral*

VHF *Very High Frequency*

IQ amostras complexas

USRP *Universal Software Radio Peripheral*

Lista de símbolos

$H_{mf}(f)$	Filtro casado ideal
$x(t)$	Sinal de entrada de um sistema
$y(t)$	Sinal de saída de um sistema
$X(f)$	Transformada de <i>Fourier</i> de $x(t)$
\mathcal{F}^{-1}	Transformada Inversa de <i>Fourier</i>
$h_{mf}(t)$	Resposta ao impulso do filtro casado
λ_{mf}	Limiar de detecção no método de filtro casado
λ_{en}	Limiar de detecção por energia
λ_{sup}	Limiar de detecção superior
λ_{inf}	Limiar de detecção inferior
T	Número de quadros lidos do sinal de entrada
N	Número de amostras em cada quadro
M	Tamanho do filtro janela
L	Tamanho da média temporal
J	Tamanho da média entre amostras
$S_x^\alpha(f)$	Função de densidade espectral cíclica
$R_x^\alpha(\tau)$	Função de autocorrelação cíclica
$x(n)$	Sinal de entrada de um sistema discreto
$y(n)$	Sinal de saída de um sistema discreto
$z(n)$	Ruído no sistema
$X(k)$	Transformada discreta de <i>Fourier</i> de $x(n)$
$w(n)$	Função filtro janela
$x_{x,t}(n)$	Sinal $x(n)$ filtrado

$\Phi(k)$	Densidade espectral de potência
$\Phi_{dB}(k)$	Logaritmo da densidade espectral de potência
$\Phi_{avg}(k)$	Média de T quadros processados
S_k	Sinal de saída pós etapa de tratamento
S_{k-1}	Sinal de saída anterior à S_k
Sd_k	Sinal diferença $S_k - S_{k-1}$
f_a	Frequência de amostragem
f_c	Frequência central
σ	Variância
μ	Média

Sumário

1	INTRODUÇÃO	21
1.1	Objetivos	22
1.1.1	Objetivo Geral	22
1.1.2	Objetivos Específicos	22
1.2	Contexto	22
1.3	Justificativa	22
2	FUNDAMENTAÇÃO TEÓRICA	25
2.1	Intermitências de Sinais Radioelétricos	25
2.2	Rádio definido por <i>Software</i>	26
2.2.1	Arquitetura do SDR Ideal	26
2.2.2	Arquitetura do SDR Real	27
2.2.3	GNU Radio	28
2.2.4	USRP	29
2.3	NumPy e SciPy	30
2.4	Sensoriamento Espectral	30
2.4.1	Detecção por Filtro Casado	31
2.4.2	Detecção Cicloestacionária	31
2.4.3	Detecção por Energia	32
3	DESENVOLVIMENTO	35
3.1	Etapas de Aquisição de Amostras	35
3.1.1	USRP N210	36
3.2	Etapas de Tratamento do Sinal	37
3.3	Etapas de Detecção de Atividades Intermitentes	39
3.3.1	Sinal Diferença	40
3.3.2	Limiares	40
4	RESULTADOS	45
4.1	Testes com Sinais Sintéticos	45
4.2	Testes com Sinais Reais	48
5	CONCLUSÕES	53
	REFERÊNCIAS	55

1 Introdução

O espectro radioelétrico abrange a faixa do espectro eletromagnético compreendida entre 3kHz e 30GHz. A faixa de 3MHz a 3GHz, que é vastamente utilizada para sistemas de comunicação, é comumente dividida em *High Frequency* (HF) (3MHz a 30MHz), *Very High Frequency* (VHF) (30MHz a 300MHz) e *Ultra High Frequency* (UHF) (300MHz a 3GHz) (FERREIRA, 2014).

A crescente necessidade do uso de tecnologias de comunicação estimulou a demanda por serviços de comunicação sem fio. O resultado foi a mudança completa do cenário de utilização do espectro radioelétrico. Como o espectro é um recurso limitado, torna-se relevante sua gestão e sua utilização de forma eficiente (FERREIRA, 2014; TELECO, 2017).

A gestão do espectro é feita de forma soberana por cada país. No Brasil, por exemplo, o órgão responsável por esta gestão é a Agência Nacional de Telecomunicações (ANATEL), já nos Estados Unidos é a *Federal Communications Commission* (FCC). Segundo a ANATEL, mesmo com os esforços da indústria e dos pesquisadores para melhoria do aproveitamento espectral, a oferta fixa deste recurso traz escassez devido a alta demanda por novas frequências (TELECO, 2017; COIMBRA, 2005).

As faixas de frequência sob responsabilidade das agências reguladoras são em maior parte reservadas. Em alguns casos usuários não licenciados fazem o uso dessas faixas, interferindo nas comunicações autorizadas.

Diante deste cenário, o monitoramento do espectro se mostra necessário para que seja garantido o uso devido das faixas regularizadas (FERREIRA, 2014). O monitoramento da faixa de 3MHz à 3GHz é extremamente complexo, dado a grande diversidade de tipos de transmissões presentes nessa parte do espectro. O uso de *Software Defined Radio* (SDR) tem se mostrado muito eficiente no monitoramento do espectro, principalmente por sua flexibilidade em analisar grandes porções do espectro (ALMEIDA, 2010; STEWART et al., 2015; WYGLINSKI et al., 2010).

Existem diversas técnicas para o monitoramento de espectro, as quais podem ser utilizadas para diversas finalidades, tais como: identificação de ociosidade do espectro, existência de sinais ilegais, entre outros (ALMEIDA, 2010; LIMA, 2014). Neste trabalho de conclusão do curso Trabalho de Conclusão de Curso (TCC) é apresentado um método para detectar sinais intermitentes no espectro radioelétrico.

1.1 Objetivos

1.1.1 Objetivo Geral

Desenvolver um método capaz de identificar sinais de rádio intermitentes no espectro radioelétrico utilizando tecnologia de rádio definido por *software*.

1.1.2 Objetivos Específicos

- Adquirir amostras de sinais intermitentes.
- Aplicar técnicas de processamento de sinais para tratamento das amostras.
- Definir método para a identificação de atividades de rádio intermitentes.
- Implementar algoritmos para identificação de variações no espectro.
- Avaliar os resultados obtidos pelo método desenvolvido.

1.2 Contexto

Para que o curso de Engenharia de Telecomunicações possa ser concluído no Instituto Federal de Santa Catarina (IFSC), a disciplina TCC29010 exige a apresentação de um trabalho de conclusão de curso. Para este propósito o presente trabalho foi desenvolvido na empresa Dígitro Tecnologia S.A., sob as orientações do professor Mario de Noronha Neto (IFSC) e coorientação do Engenheiro Douglas Amorim Ferreira (Dígitro). Neste trabalho houve a implementação de um algoritmo para detecção de atividades intermitentes no espectro radioelétrico. O sistema que representa as etapas deste trabalho está representado na Figura 1. Na primeira etapa o sinal na frequência desejada é capturado por um receptor de rádio e deslocado para banda base. O sinal adquirido na primeira etapa é encaminhado para a etapa de processamento no *host*. Durante a etapa de processamento, o sinal é condicionado e encaminhado para o algoritmo de detecção de atividades intermitentes.

1.3 Justificativa

Este trabalho está sendo desenvolvido devido ao interesse em identificar comunicações de rádio no espectro, mas também pela procura por inovação tecnológica na empresa Dígitro Tecnologia S.A.

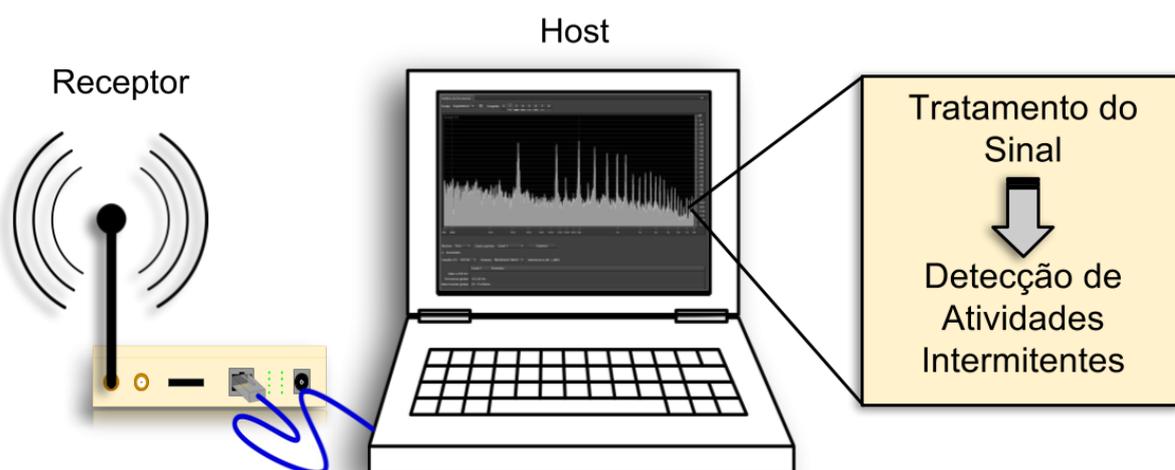


Figura 1 – Diagrama do sistema das etapas do trabalho.

2 Fundamentação Teórica

Neste capítulo serão abordadas as principais tecnologias e conceitos aplicados no desenvolvimento deste trabalho. O capítulo inicia com a descrição de intermitência e como sinais desse tipo se comportam. Em seguida é apresentada a tecnologia de rádio definido por *software* e ferramentas para sua implementação. Por fim, são apresentadas as principais técnicas de sensoriamento de espectro.

2.1 Intermitências de Sinais Radioelétricos

Intermitência pode ser relacionada a algo que cessa e recomeça por intervalos. Ou seja, ser intermitente é não ser contínuo, ter interrupções (GREGORIM et al., 2008). Um sinal de rádio pode surgir em algumas faixas do espectro radioelétrico por um determinado tempo, cessar e volta a aparecer em um tempo distinto. Esse tipo de transmissão é conhecida com intermitente. A Figura 2 mostra o espectrograma de um sinal onde o transmissor funciona intermitentemente. Quanto mais intensa a cor do espectrograma, maior é a amplitude do sinal naquela frequência (AGILENT TECHNOLOGIES, 2012).

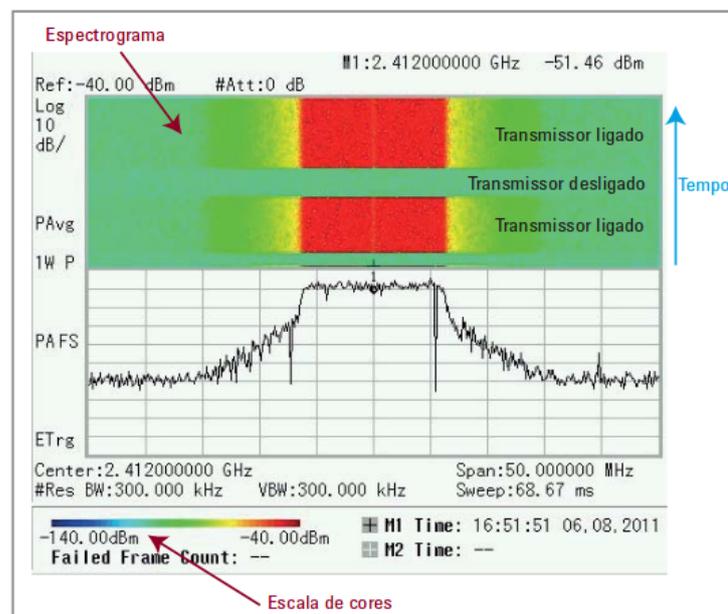


Figura 2 – Espectrograma de frequência de uma transmissão intermitente.(AGILENT TECHNOLOGIES, 2012)

Podemos destacar alguns exemplos de uso de transmissões intermitentes, tais como rádio *Talkabout* e *WalkieTalkie*, telégrafo (utilizando código Morse) e rádios que utilizam da técnica de *frequency hopping* (AGILENT TECHNOLOGIES, 2012).

2.2 Rádio definido por *Software*

Nas últimas décadas o SDR tem sido um tema de grande interesse para pesquisas e desenvolvimento. O SDR se refere a um sistema de rádio, onde quase todas as funcionalidades relacionadas a um rádio analógico convencional são implementadas em software (STEWART et al., 2015).

A implementação de um SDR é feita através de algoritmos de processamento digital de sinais, ou *Digital Signal Processing* (DSP), apropriados para (de)modular, (de)codificar e extrair informações de um sinal (STEWART et al., 2015). Dessa forma, com uma simples alteração de software é possível modificar todo o funcionamento do sistema de rádio, sem haver a necessidade de alteração do hardware (INÁCIO, 2011). Adicionalmente, podemos citar outras vantagens associadas ao uso de SDR, tais como (FERREIRA, 2014):

- *Multi-Band*: pode-se operar em diferentes bandas de frequência, simultânea ou sequencialmente;
- *Multi-Carrier*: possui flexibilidade para operar em várias frequências simultaneamente, seja na mesma banda ou não;
- *Multi-Rate*: trabalha com diferentes taxas de amostragem;
- Largura de banda variável: a largura de banda do sinal é determinada por filtros digitais, podendo alterá-los facilmente e como consequência a largura de banda também é facilmente alterada;

2.2.1 Arquitetura do SDR Ideal

No funcionamento ideal de um SDR, todo o processamento do sinal de rádio deve ser realizado em software (FERREIRA, 2014; WYGLINSKI et al., 2010). O diagrama de blocos ilustrado na Figura 3 apresenta um sistema considerado ideal para um SDR. A interface analógica é composta apenas pelos conversores: *Analogic-Digital Converter* (ADC) e *Digital-Analogic Converter* (DAC) (FERREIRA, 2014; INÁCIO, 2011). Na interface digital, está o processador, responsável por todas funcionalidades de processamento de sinais de rádio, como: modulação e demodulação, codificação e decodificação, deslocamento de sinal, entre outras (INÁCIO, 2011).

No entanto, existem algumas limitações de *hardware* que impedem a implementação do SDR descrito acima. Os ADC e DAC que existem na prática não suprem todos os requisitos referentes a taxa de amostragem, largura de banda e a faixa dinâmica. Se fosse possível a implementação de um SDR como descrito acima, além das limitações dos conversores, o custo computacional para processar os sinais seria muito alto (FERREIRA, 2014).

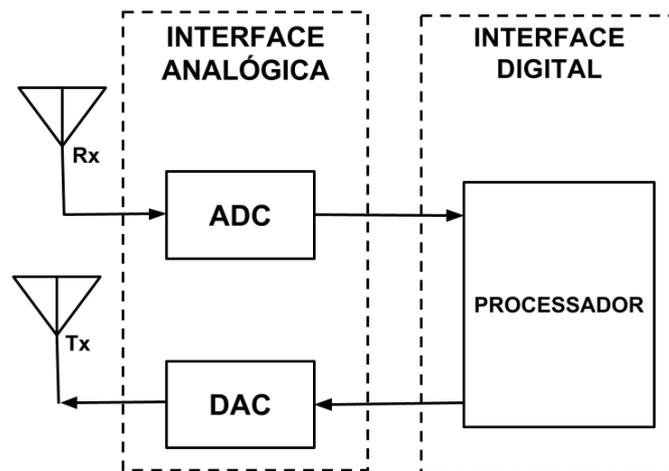


Figura 3 – Diagrama de blocos de um SDR ideal.

2.2.2 Arquitetura do SDR Real

O SDR real utiliza a combinação de sistemas de rádio analógico com sistemas de rádio digital. Essa combinação corrige algumas das limitações do SDR ideal descritas anteriormente (INÁCIO, 2011). O diagrama de blocos da Figura 4 mostra um sistema de SDR real.

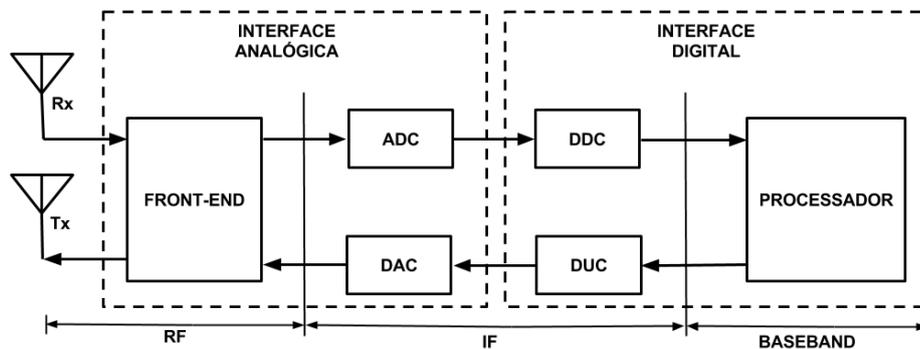


Figura 4 – Diagrama de blocos de um SDR real.

Um SDR real possui interfaces de condicionamento dos sinais analógicos antes do ADC e após o DAC. Essas interfaces podem ser chamadas de *front-ends* de RF. O condicionamento dos sinais ocorre através de ampliações, controle de ganho, filtragem e deslocamento na frequência (FERREIRA, 2014). Durante a recepção o sinal é deslocado para uma frequência intermediária e após o ADC passa por um *Digital Down Converter* (DDC), que basicamente desloca o sinal da frequência intermediária para banda-base, para ser processado. Na transmissão o sinal em banda-base vindo do processador é convertido para a frequência intermediária através de um *Digital Up Converter* (DUC), passa pelo DAC e é deslocado para a frequência de RF (FERREIRA, 2014; STEWART et al., 2015).

2.2.3 GNU Radio

O GNU Radio é um kit de ferramentas de *software* livre para desenvolvimento de SDR. Esse kit pode ser utilizado em computadores pessoais combinado a um hardware para conversão e aquisição de sinais radioelétricos (FERREIRA, 2014; GNURADIO, 2017a). O GNU Radio é licenciado pela *General Public License* (GPL) versão 3 ou posterior, permitindo que seja utilizado e modificado por qualquer pessoa. Todo seu código pertence a *Free Software Foundation* (GNURADIO, 2017a).

As aplicações do GNU Radio são blocos de processamento conectados e reconfiguráveis por onde há um fluxo de dados (GNURADIO, 2014). Os blocos conectados são chamados *flowgraphs*. Os blocos possuem algoritmos de filtros, codificação de canal, elementos de sincronização, (de)codificadores, (de)moduladores, equalizadores, entre outros elementos normalmente encontrados em sistemas de rádio (FERREIRA, 2014). A Figura 5 ilustra um modelo de *flowgraph*, onde amostras de áudio salvas em um arquivos são lidas (File Source), demoduladas e chegando ao fim o áudio é reproduzido (Audio Sink) .

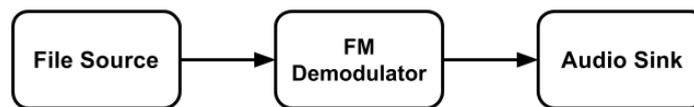


Figura 5 – Representação um *flowgraph*.

Os blocos GNU Radio podem ser implementados em C++ ou *Python*, no entanto, toda infra-estrutura da biblioteca é desenvolvida em C++. Para possibilitar o desenvolvimento dos *flowgraphs* nessas duas linguagens, o GNU Radio usa o *Simplified Wrapper and Interface Generator* (SWIG) (FERREIRA, 2014).

Existem formas de utilizar o GNU Radio sem a necessidade de escrever um código C++ ou *Python*, sendo uma delas a partir do uso do GNU Radio Companion, que é uma interface gráfica para facilitar a criação dos *flowgraphs* utilizando blocos já implementados. Ao finalizar as implementações no GNU Radio Companion é possível gerar um *script* em *Python* que representa o fluxo projetado (GNURADIO, 2014). Na Figura 6 é apresentado a construção de um *flowgraph* na interface do GNU Radio Companion.

O *flowgraph* construído na Figura 6 trata da implementação do diagrama da Figura 5, utilizando os blocos do GNU Radio: File Sink, NBFM Receive e Audio Sink. Os blocos necessitam de configuração, nesse caso, taxa de amostragem e taxa de áudio, para permitir o funcionamento desejado. Os blocos são agrupados por tipo e facilmente selecionados na lista ao lado direito da interface. A geração do arquivo *Python* ou execução do *flowgraph* é feita apenas clicando os botões da barra superior. A execução pode ser acompanhada no terminal na parte inferior da interface ou com janelas que abrem durante a simulação (dependendo dos blocos utilizados) (GNURADIO, 2017b).

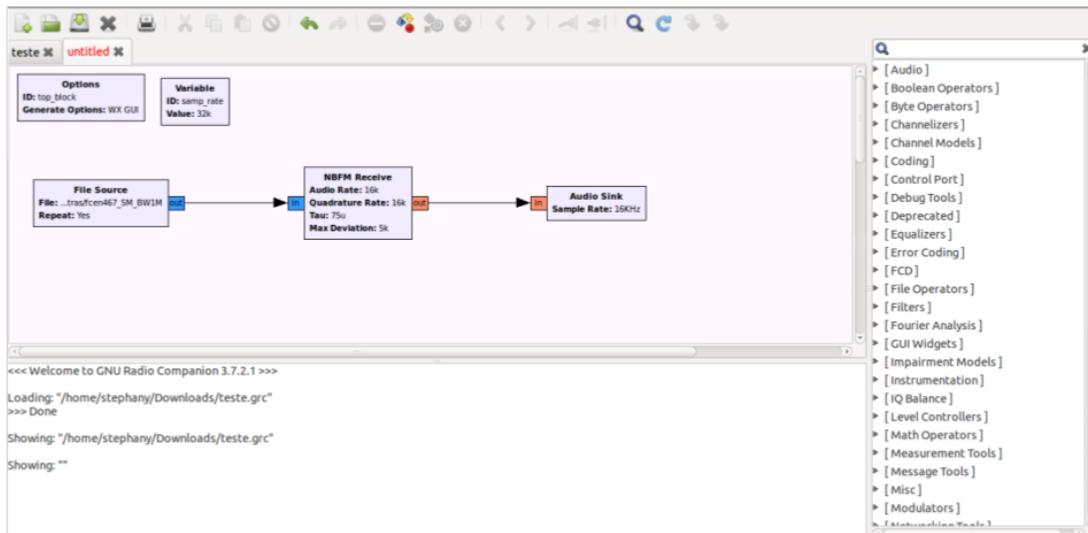


Figura 6 – Interface GNU Radio Companion com *flowgraph* para recepção NBFM.

O GNU Radio oferece vantagens relacionadas ao aumento de produtividade, devido a vasta quantidade de bibliotecas com algoritmos para processamento de sinais. Por outro lado, as aplicações geralmente são executadas em computadores pessoais que possuem barramentos compartilhados para comunicação com os periféricos de rádio, implicando em atrasos de processamento que podem impossibilitar sistemas com requisitos de tempo críticos (FERREIRA, 2014; JANSON, 2014).

2.2.4 USRP

O USRP é um *hardware* para desenvolvimento de SDR fornecido pela empresa Ettus Research, da National Instruments. Existem diversos modelos de USRP no mercado que possuem uma arquitetura similar, composta por *Field Programmable Gate Array* (FPGA), ADC e DAC (ETTUS, 2017b).

Em alguns modelos de USRP os *front-ends* de RF são integrados, porém em outros é necessário fazer a integração com circuitos de RF separados. Os circuitos presentes dentro do USRP são chamadas de placas mãe e os *front-ends* de RF vendidos pela Ettus para integração com o USRP são conhecidos como placas filhas (ETTUS, 2017b).

Na Tabela 1 são apresentadas três dos USRPs disponíveis no mercado, juntamente da descrição de suas principais características, como largura de banda máxima, interface de conexão com o *host* e se há *front-end* de RF integrado ou não.

A escolha da utilização de um USRP depende dos requisitos do sistema. Cada sistema possui uma especificação, e o USRP escolhido deve ser aquele que atende melhor a esses requisitos. Um exemplo de requisito é a faixa de frequências (FERREIRA, 2014).

Tabela 1 – Opções de USRP disponíveis (ETTUS, 2017b)

Nome	Front-End de RF	Largura de Banda Máxima	Interface
USRP1	Placas Filas (DC à 6GHz)	8MHz	USB 2.0
USRP B210	Integrado (70MHz à 6GHz)	56MHz	USB 3.0
USRP N210	Placas Filas (DC à 6GHz)	50MHz	Gigabit Ethernet

2.3 NumPy e SciPy

O Numpy é um pacote *Python* para computação científica que possibilita utilizar matrizes e vetores de diversas dimensões. Ele fornece a opção trabalhar com funções mais sofisticadas para álgebra linear, transformada de Fourier, geração randômicas de valores e etc (NUMPY, 2017).

O Scipy é uma coleção de bibliotecas matemáticas e científicas de código livre em *Python*. O Scipy foi construído sob a biblioteca Numpy e fornece ao usuário várias opções para manipulação e visualização de dados. A utilização do Scipy amplia a capacidade de desenvolvimento em *Python*, pois facilita o desenvolvimento em diversos nichos, como: processamento de sinais, *clustering*, funções e distribuições estatísticas, entre outros (SCIPY, 2008-2016).

2.4 Sensoriamento Espectral

O sensoriamento espectral refere-se tradicionalmente a medições que resultam em características do espectro eletromagnético. Exemplos de utilização do sensoriamento espectral seriam o rádio cognitivo e analisadores de espectro (ALMEIDA, 2010; FERREIRA, 2014; YUCEK; ARSLAN, 2009).

Para se obter bons resultados durante o sensoriamento espectral, algumas características devem ser avaliadas em técnicas de sensoriamento, como (ALMEIDA, 2010):

- Robustez: capacidade de identificar características e mudanças do espectro mesmo diante das piores condições de propagação e ruído;
- Rapidez: as atividades no espectro variam a cada instante, então o algoritmo de sensoriamento deve ser rápido o suficiente para detectar qualquer mudança no espectro;
- Precisão: a identificação de características e mudanças no espectro deve ter alta probabilidade de acerto;

Há casos em que o sensoriamento espectral é realizado em duas etapas: grosseira e refinada. A primeira para identificação de atividades e a segunda para classificação das características da atividade identificada (ALMEIDA, 2010).

Na literatura existem diversas técnicas de sensoriamento espectral, porém algumas estão mais difundidas. As três técnicas que se destacam, são: detecção por filtro casado, cicloestacionária e energia (FERREIRA, 2014).

2.4.1 Detecção por Filtro Casado

A detecção por filtro casado utiliza um filtro linear projetado para retornar a máxima relação sinal ruído para uma dada entrada, necessitando o conhecimento do sinal a ser detectado (YUCEK; ARSLAN, 2009).

Um filtro ideal para se obter máxima relação sinal ruído para o sinal de entrada $x(t)$ é dado por

$$H_{mf}(f) = X^*(f)e^{-j2\pi fT} \quad (2.1)$$

onde $X(f)$ é a transformada de Fourier do sinal analisado e o operador $*$ representa a conjugação complexa (ALMEIDA, 2010) e

$$h_{mf}(t) = \mathcal{F}^{-1}(H_{mf}(f)) \quad (2.2)$$

representa a resposta ao impulso do filtro, onde \mathcal{F}^{-1} é a transformada de Fourier inversa (ALMEIDA, 2010).

Na Figura 7 o diagrama de blocos apresenta uma forma de implementação da técnica de detecção por filtro casado. A saída $y(t)$ é obtida através da equação 2.2. A presença de atividade no sinal $x(t)$ é detectada através da comparação da saída $y(t)$ com um limiar λ_{mf} , onde a ausência de atividade acontece quando $y(t) < \lambda_{mf}$ e a presença quando $y(t) > \lambda_{mf}$ (FERREIRA, 2014). Esta técnica possui a vantagem de um curto tempo de observação do sinal de entrada associado à alta precisão para a detecção da atividade. No entanto, para cada atividade distinta é necessária uma nova implementação, que por sua complexidade apresenta grande desvantagem para o método (ALMEIDA, 2010; FERREIRA, 2014).

2.4.2 Detecção Cicloestacionária

Devido a modulação dos sinais por portadoras senoidais periódicas, a grande maioria dos sinais digitais de comunicação possui características ciclo estacionárias. Um sinal é dito ciclo estacionário quando possui parâmetros estatísticos variando ciclicamente com o tempo (ALMEIDA, 2010; FERREIRA, 2014). A detecção cicloestacionária explora essas

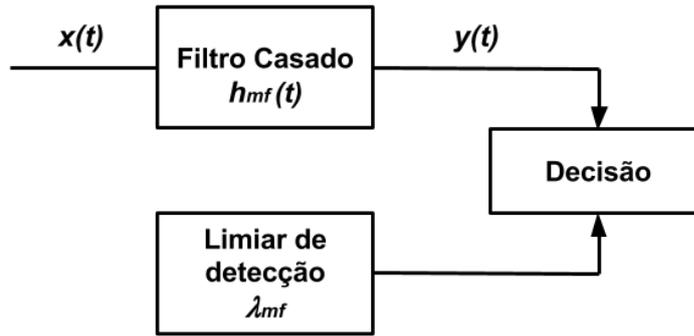


Figura 7 – Diagrama de blocos da detecção por filtro casado.

propriedades nos sinais de comunicação (ALMEIDA, 2010). Para identificar a presença de atividades no espectro radioelétrico essa técnica utiliza a função de correlação cíclica. Através dos algoritmos de detecção cicloestacionários é possível diferenciar ruído de transmissão, devido aos ruídos não serem ciclo estacionários (YUCEK; ARSLAN, 2009). Essa técnica considera o cálculo da função da densidade espectral cíclica, expressa pela equação

$$S_x^\alpha(f) = \sum_{\tau=-\infty}^{\infty} R_x^\alpha(\tau) e^{-j2\pi f\tau} \quad (2.3)$$

onde

$$R_x^\alpha(f) = E[x(n + \tau)x^*(n - \tau)e^{j2\pi\alpha n}] \quad (2.4)$$

é a função de autocorrelação, α é a frequência cíclica e $x(n)$ o sinal de entrada.

Através da função de densidade espectral cíclica são gerados valores de pico quando a frequência cíclica é igual a frequência do sinal de entrada $x(n)$ que possui atividades (YUCEK; ARSLAN, 2009).

Esta técnica de detecção exige um tempo elevado de observação do sinal de entrada, além do alto custo computacional (FERREIRA, 2014).

2.4.3 Detecção por Energia

A técnica de detecção por energia utiliza a energia espectral do sinal de entrada para a identificação de atividades. Devida a sua baixa complexidade de implementação, é a técnica mais utilizada para sensoriamento de espectro (URKOWITZ, 1967). A detecção acontece com base em duas hipóteses,

$$H_0 : y(n) = z(n) \quad (2.5)$$

quando não há presença de atividade e o sinal recebido $y(n)$ possui apenas amostras de ruído $z(n)$,

$$H_1 : y(n) = x(n) + z(n) \quad (2.6)$$

quando há a presença de atividade $x(n)$ juntamente das amostras de ruído (ALMEIDA, 2010).

Na Figura 8 está ilustrado o diagrama de bloco correspondente à técnica de detecção por energia.

A hipótese H_0 é verdadeira quando $X(k) \leq \lambda_{en}$ e H_1 é verdadeira quando $X(k) > \lambda_{en}$, onde $X(k)$ é a transformada discreta de Fourier do sinal de entrada $x(n)$ e λ_{en} é o limiar de detecção no domínio da frequência (FERREIRA, 2014). O limiar de detecção depende diretamente da relação sinal ruído do sinal analisado. A escolha do limiar tem impacto direto no desempenho do detector. É de grande importância garantir que tenha alta probabilidade de a hipótese H_1 acontecer. A sensibilidade à variação de parâmetros e a incerteza de ruído degrada a probabilidade de H_1 , podendo ocasionar alarmes falsos prejudiciais à detecção (ALMEIDA, 2010).

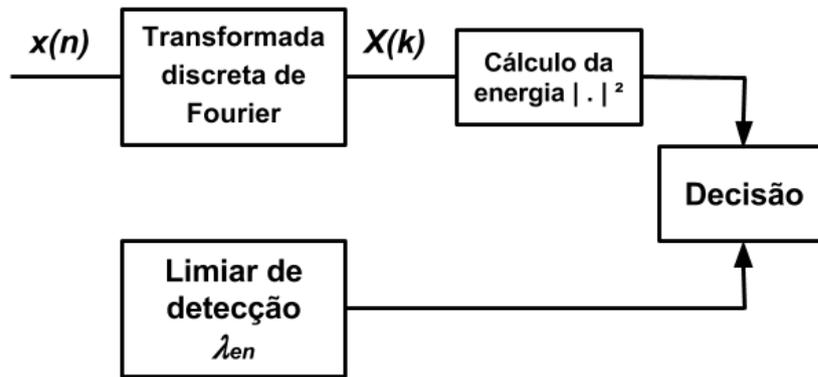


Figura 8 – Diagrama de blocos da detecção por filtro casado.

A Figura 9 ilustra os erros cometidos na escolha do limiar de detecção. Na Figura 9a o limiar de detecção é baixo, ocasionando falsas detecções devido ao ruído presente no sinal. Na Figura 9b o limiar de detecção é muito alto e acaba não detectando algumas atividades (FERREIRA, 2014).

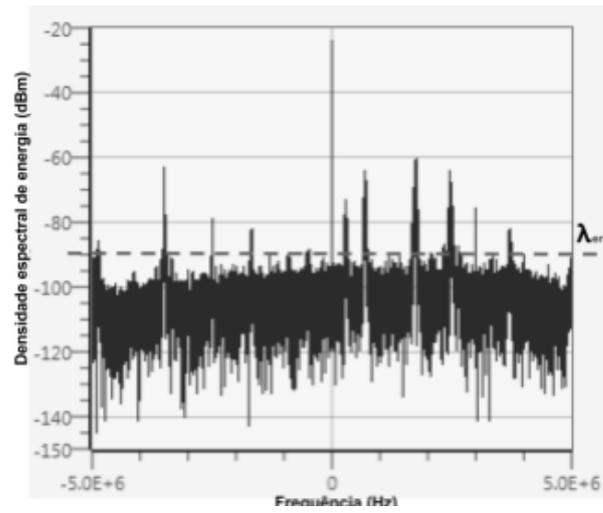
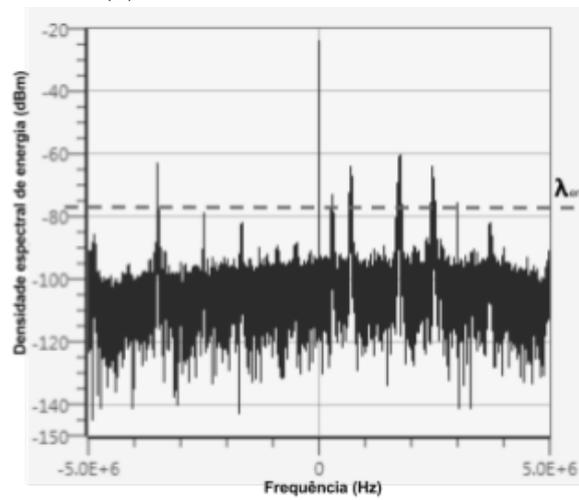
(a) Escolha do limiar λ_{en} baixo.(b) Escolha do limiar λ_{en} alto.

Figura 9 – Ilustração de escolha incorreta de limiares de detecção

3 Desenvolvimento

Neste capítulo será apresentada a implementação do algoritmo de detecção de atividades intermitentes proposto por esse trabalho de conclusão de curso. Inicialmente teremos a explicação da etapa de aquisição de amostras de sinais de rádio. Em seguida, a etapa de tratamento das amostras de sinais de rádio. E, por fim, a implementação realizada para a detecção de atividades de rádio intermitentes. As etapas abordadas neste capítulo estão ilustradas na Figura 10.

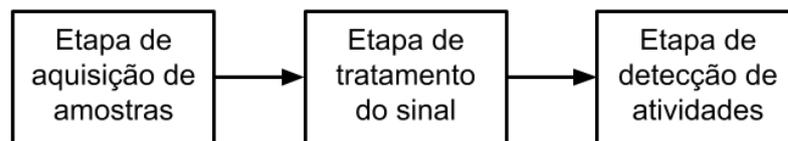


Figura 10 – Diagrama de blocos do sistema proposto.

3.1 Etapa de Aquisição de Amostras

A etapa de aquisição das amostras é a primeira etapa do sistema proposto, onde os sinais são adquiridos e armazenados em arquivos. O diagrama de blocos da Figura 11 ilustra a redistribuição de processos que acontecem nesta etapa.



Figura 11 – Diagrama de blocos ampliado da etapa de aquisição de amostras.

A aquisição de sinais acontece através de uma antena conectada a um USRP em conjunto com um *flowgraph* construído através do GNU Radio Companion. Este *flowgraph* recebe as amostras complexas (IQ) de sinais do USRP e pode tratá-las de diversas formas. Uma delas é salvar em um arquivo comum, que pode ser utilizado depois. Outra tratativa é

criar um arquivo do tipo *pipe*. Arquivos *pipe* possibilitam escrever nele as amostras e ler as mesmas em tempo de execução do *flowgraph*, ou seja, em tempo real. Além das duas formas citadas anteriormente é interessante abordar a ideia da criação de um bloco no GNU Radio com uma implementação que utiliza as amostras. A ideia da criação de um bloco é utilizar as amostras em *stream* sem a necessidade de criar arquivos. Na última opção não se segue a ideia do diagrama da Figura 11, pois o *stream* acontece diretamente dentro do GNU Radio. Para esse trabalho a segunda e a terceira forma são mais interessante, pois é ideal que a identificação das atividades ocorram em tempo real.

3.1.1 USRP N210

O USRP utilizado neste trabalho foi o N210, pertencente a classe de *hardware* de alta performance da Ettus Research. Ele fornece a capacidade de processamento de alta largura de banda com faixa de frequência dinâmica (ETTUS, 2017a). A Figura 12 mostra a fotografia do equipamento comercializado pela Ettus. Esse equipamento é uma caixa metálica onde encontra-se o *hardware* conhecido como USRP N210.



Figura 12 – USRP N210 comercializado pela *Ettus* (ETTUS, 2017a).

O *hardware* do USRP N210 é composto por uma placa-mãe com dois ADC de 14 bits de 100MS/s, dois DAC de 16 bits de 400MS/s, além de DDC e DUC com 25MHz de resolução (ETTUS, 2017c). Os componentes listados anteriormente são controlados por um FPGA. A interface de comunicação com o *host* é um cabo Gigabit Ethernet (ETTUS, 2017a). Por ela usar cabo Gigabit Ethernet, a largura de banda máxima da N210 é limitada para 50MHz (ETTUS, 2017c). Essa placa não possui *front-end* de RF integrado, então ela depende de placas filhas (FERREIRA, 2014). A Figura 13 apresenta o *hardware* da placa mãe N210 sem a presença de placas filhas.

As faixas de atuação do N210 dependem somente das placas filhas conectadas nela. Uma vantagem é a existência de diversas placas filhas que são compatíveis com este



Figura 13 – Hardware USRP N210

modelo. A Tabela 2 apresenta a relação de placas filhas disponíveis para o USRP N210, detalhando as faixas de atuação de cada uma e suas larguras de banda.

Tabela 2 – Relação de placas filhas compatíveis com o USRP N210.

Placa	Frequência Inicial	Frequência Final	Largura de Banda
Basic Tx/Rx	1MHz	250MHz	50MHz
LFTX/LFRX	DC	50MHz	50MHz
TVRX2	50MHz	860MHz	10MHz
DBSRX2	800MHz	2.35GHz	1-60MHz
WBX	50MHz	2.35GHz	40MHz
SBX	400MHz	4.4GHz	40MHz
CBX	1.2GHz	6GHz	40MHz
UBX40	10MHz	6GHz	40MHz

A placa filha utilizada neste trabalho é a WBX, com o intuito de cobrir a faixa UHF.

3.2 Etapa de Tratamento do Sinal

A etapa de tratamento do sinal tem como função o condicionamento do sinal digitalizado proveniente da etapa de aquisição. Esta etapa é necessária para a preparação do sinal para a etapa de detecção de atividades intermitente descrita na seção 3.3.

O sinal de entrada recebido é digitalizado em banda base e fornece como saída a Densidade Espectral de Potência (DEP). O processamento realizado nessa etapa foi imple-

mentado na linguagem de programação *Python* com a ajuda de módulos de processamentos de sinais nativos.

O sinal de entrada $x(n)$ possui um bloco de amostras IQ que são lidas em T números de quadros com N amostras. Em seguida, cada quadro é multiplicado por uma função janela $w(n)$,

$$x_{w,t}(n) = w(n)x_t(n), \quad (3.1)$$

onde $x_t(n)$ são os termos dos quadros de amostras do sinal de entrada. A janela $w(n)$ utilizada foi do tipo Hanning e é representada por:

$$w(n) = 0.5 - 0.5 \cos\left(\frac{2\pi n}{m}\right), 0 \leq n \leq M - 1, \quad (3.2)$$

em que M é o tamanho da janela. O intuito é utilizar essa janela para minimizar as perdas espectrais causadas pela descontinuidade dos quadros de amostras.

No próximo passo é calculada a *Discrete Fourier Transform* (DFT) $X_t(k)$ do quadro janelado $x_{w,t}(n)$ através de:

$$X_t(k) = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} x_{w,t}(n) e^{-j\frac{2\pi kn}{N}}, \quad (3.3)$$

para obter a resposta em frequência instantânea de cada quadro, onde N é o número de *taps* da DFT.

Para determinar a distribuição de potência do sinal é calculada a DEP através de

$$\Phi(k) = |X_t(k)|^2, \quad (3.4)$$

facilitando a identificação do picos de sinal onde há transmissões.

No caso de haver um "piso" de ruído alto os sinais de baixa potência podem ser prejudicados e informações podem ser perdidas. Para suavização do ruído foi utilizada uma média temporal, calculada através de:

$$\Phi_{avg}(k) = \frac{1}{L} \sum_{t=0}^{L-1} \Phi(k)_t, \quad (3.5)$$

onde L é o número de quadros pré processados e armazenados para cálculo da média. Essa média proporciona maior robustez, pois gera uma estabilidade no piso de ruído e realça os picos. Por outro lado, quanto maior o número de quadros utilizados para a média, ou seja, quanto maior L, se um pico deixar de existir nos quadros futuros pode acabar atenuando as atividades do presente.

As variações de frequência podem ocasionar a identificação de vários picos por um algoritmo, mesmo quando trata-se de apenas um. Em seguida uma média é aplicada no

quadro resultante da equação 3.5 para proporciona maior robustez para o sinal e reduz a variação em frequência do espectro. A equação

$$S_k = \frac{1}{J+1} \sum_{i=-\frac{J}{2}}^{\frac{J}{2}} \Phi_{avg}(k+i) \quad (3.6)$$

calcula a média entre amostras do quadro, onde J é o número de amostras utilizadas para a média. Quanto maior o valor de J menor a resolução em frequência do sinal.

Os tratamentos descritos nesta seção são simples porém essenciais para a eficiência das próximas etapas. Apesar de simples a determinação dos valores de T, J, L e J determinam a eficiência dos resultados obtidos, pois a potência do sinal lido pode ser diferentes em cada caso, e o processamentos das amostras pode prejudicar as informações. Não existem valores pré determinados a serem utilizados, pois existem inúmeras situações de aquisição de sinais, mas é possível determinar essas variáveis para os principais casos de uso.

3.3 Etapa de Detecção de Atividades Intermitentes

A etapa de detecção de atividades intermitentes é a última realizada. Nela o espectro tratado na etapa anterior é sensoreado. Durante esta etapa as atividades presentes na porção de espectro analisada são detectadas e identificadas como intermitentes ou não.

Na Seção 2.4 foram apresentadas algumas das técnicas de detecção de atividades no espectro. Porém, algumas técnicas possuem características pouco interessantes para a proposta deste trabalho. Na técnica de detecção por filtro casado (Seção 2.4.1) é necessário o conhecimento prévio das características das atividades a serem detectadas. Esse fator torna necessário o conhecimento de todo o espectro para que seja possível monitorá-lo. A segunda técnica (Seção 2.4.2) depende do cálculo da correlação espectral. A inviabilidade desta técnica é devida ao alto poder computacional exigido, em conjunto ao elevado tempo de observação. Estes dois pontos prejudicam principalmente sistemas em tempo real, que necessitam de resultados rápidos. Por fim, a técnica de detecção por energia (Seção 2.4.3) necessita do conhecimento do "piso" de ruído, que pode ser estimado. Através desta estimativa fica mais simples a determinação do limiar de detecção.

Este trabalho propõe uma variação da técnica de detecção por energia, que possibilita a detecção e a classificação das atividades simultaneamente. Na Seção 3.3.1 será detalhado como o sinal é preparado, através do cálculo da diferença entre amostras, para a identificação de intermitências. Por fim, na Seção 3.3.2, será explicado como ocorre a detecção a partir de limiares.

3.3.1 Sinal Diferença

Na Seção 3.2 foi descrita a etapa do tratamento de sinais, que trouxe como resultado o sinal S_k . Para a identificação de atividades intermitentes há dependência de um histórico do sinal S_k , ou seja, o sinal anterior, S_{k-1} e o sinal atual S_k . A detecção acontece calculando-se o sinal diferença, Sd_k , entre o sinal atual S_k e o sinal anterior S_{k-1} , pois esse cálculo traz a informação das variações do espectro. O sinal Sd_k , pode ser descrito pela equação

$$Sd_k = S_k - S_{k-1} \quad (3.7)$$

A equação 3.7 só traz resultados perceptíveis quando desaparecem ou surgem atividades no espectro. Para exemplificar supomos que S_k e S_{k-1} tenha valor 0 e 1. Se considerarmos o sinal $k = 0$, onde $S_k = 1$ e $S_{k-1} = 0$, pela equação 3.7 $Sd_k = 1$. Se no próximo instante $k = 1$, temos $S_k = 1$ e $S_{k-1} = 1$, então $Sd_k = 0$. Com essa analogia é possível notar que quando há uma mudança no sinal atual essa mudança é refletida em Sd_k , mas caso de $S_k = S_{k-1}$ nada acontece em Sd_k .

A Figura 14 apresenta um gráfico que corresponde o resultado esperado de Sd_k quando surge uma atividade no espectro. Na Figura 14a o sinal S_{k-1} é mostrado e na Figura 14b o sinal S_k , quando surge uma atividade. O resultado da diferença é apresentado na Figura 14c.

O sinal Sd_k pode apresentar quatro estados distintos, representados na máquina de estados da Figura 15. No estado 1, Sd_k não existe transmissão. No estado 2, acontece o início de uma transmissão. O estado 3, é quando a transmissão está acontecendo. Por fim, o estado 4 representa o fim da atividade.

Para entender melhor o significado dos estados, na Figura 16 há a representação de uma possível situação do sinal Sd_k em cada um dos estados. Na Figura 16a um exemplo do estado 1, em 16b o estado 2, em 16c o estado 3 e em 16d o estado 4.

As informações dos estados vão possibilitar determinar se o sinal é intermitente ou não. Essas informações são observadas através dos picos positivos e negativos na Figura 16b e 16c respectivamente. Os picos serão analisados a partir de dois limiares de detecção λ_{inf} e λ_{sup} , com escala linear. A detecção acontecerá através do ajuste dos limiares entre os picos e o "pisso" de ruído restante. A etapa de detecção será descrita na seção 3.3.2.

3.3.2 Limiares

Os limiares de detecção devem ser determinados para o funcionamento adequado da detecção. O limiar superior λ_{sup} é relativo a um valor mínimo que possibilita estimar o início de uma atividade. O limiar inferior λ_{inf} tem relação a um valor máximo que

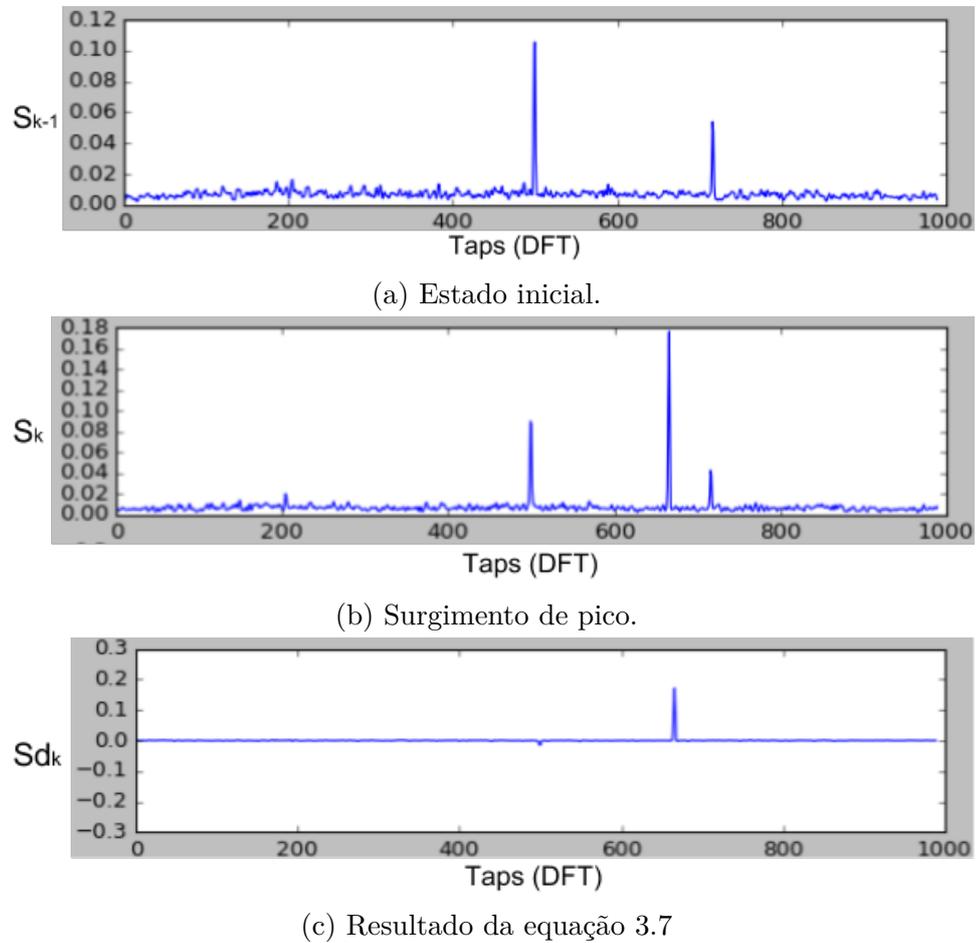


Figura 14 – Representação gráfica da equação 3.7 para surgimento de atividade.

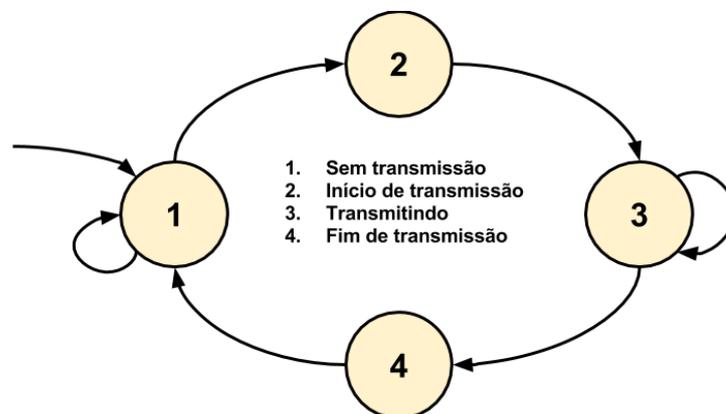
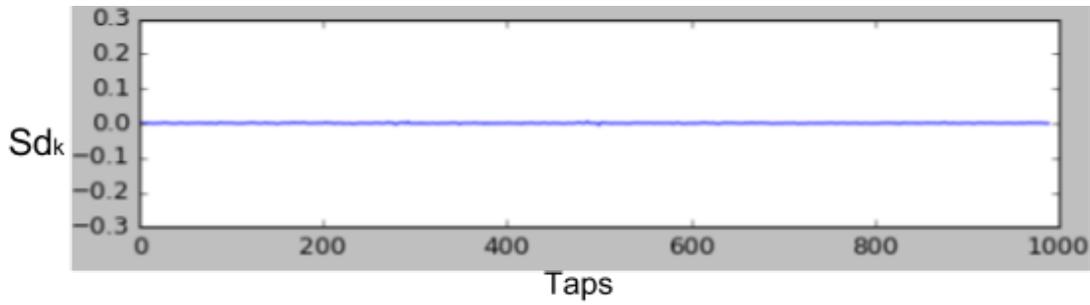
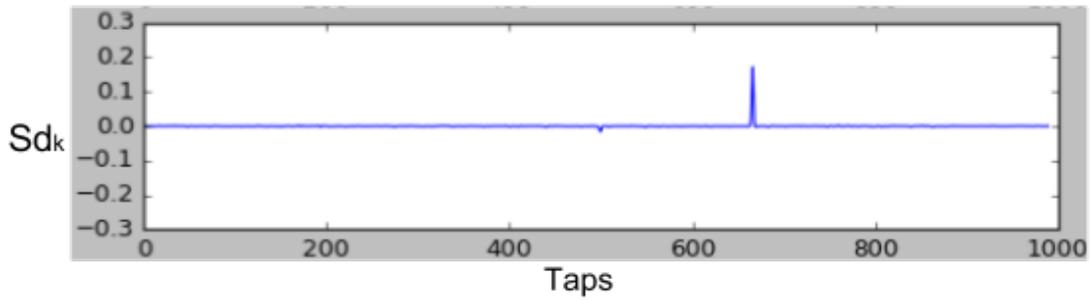


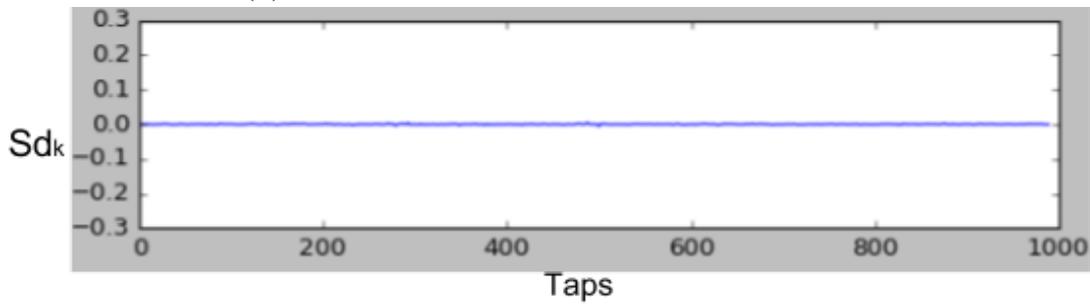
Figura 15 – Máquina de estados do sinal S_{d_k} .



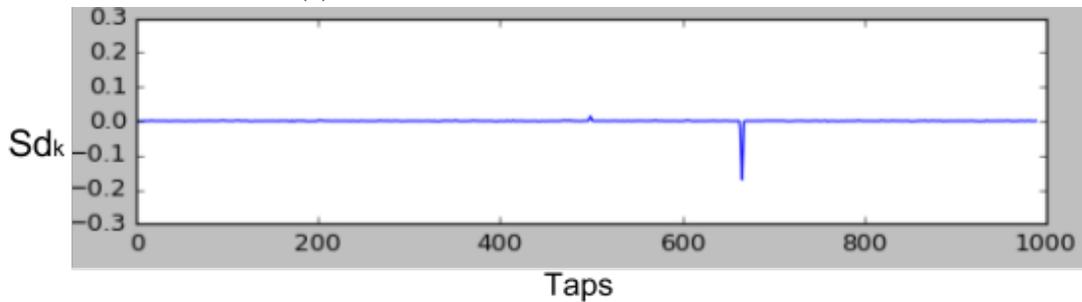
(a) Sinal no estado 1 - "sem transmissão".



(b) Sinal no estado 2 - "início de transmissão".



(c) Sinal no estado 3 - "transmitindo".



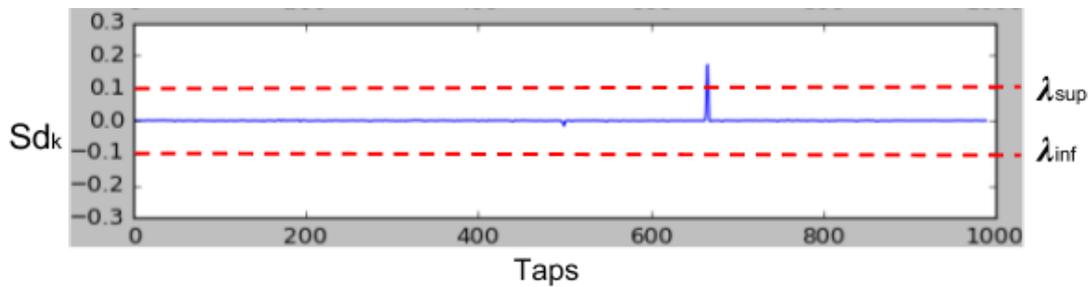
(d) Sinal no estado 4 - "fim de transmissão".

Figura 16 – Gráficos dos estados possíveis de S_{dk} .

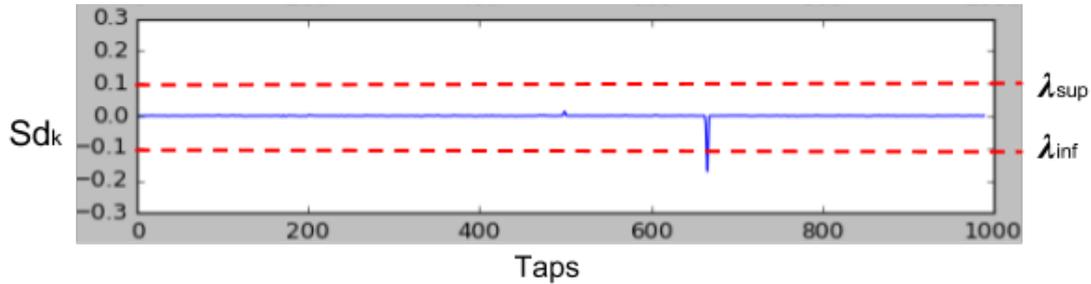
possibilita estimar o fim de uma atividade. Considerando o sinal de entrada Sd_k , temos a seguinte análise:

$$\begin{cases} \text{Início de transmissão, se} & Sd_k > \lambda_{sup} \\ \text{Fim de transmissão, se} & Sd_k < \lambda_{inf} \\ \text{Transmitindo ou sem transmissão,} & C.C. \end{cases} \quad (3.8)$$

A Figura 17 mostra como são aplicados os limiares para a detecção de atividades intermitentes a partir da expressão 3.8, a qual o pode ser aplicada para qualquer entrada Sd_k .



(a) Limiar λ_{sup} ideal para detectar surgimento de atividades.



(b) Limiar λ_{inf} ideal para detectar fim de atividades.

Figura 17 – Sinais Sd_k com limiares ideais para detecção de intermitências.

A determinação do limiar deve ser o melhor valor entre o "piso" de ruído e o pico considerado atividade. As Figuras 17a e 17b ilustram as melhores aplicações para os limiares superior e inferior neste sinal. Neste trabalho cada limiar foi determinado empiricamente a partir da análise dos gráficos de cada sinal adquirido. Em sinais conhecidos há maior facilidade na determinação dos limiares, porém em casos reais a grande variação das condições do espectro torna a escolha dos limiares complexa e imprecisa.

4 Resultados

Neste capítulo serão apresentados os resultados obtidos com este trabalho de conclusão de curso. A comparação para avaliar se uma atividade é intermitente ou não se dará a partir da análise do espectrograma, ou *waterfall*, dos sinais. O *waterfall* possibilita visualizar e identificar facilmente onde há sinais intermitentes no espectro como exemplificado na Figura 18, onde o eixo da horizontal representa os *taps* da DFT e o vertical o histórico temporal de sinais.

Serão analisados os resultados considerando sinais sintéticos e reais. Os sinais sintéticos foram gerados a partir de senoides utilizando o GNU Radio. Já os sinais reais foram coletados em uma faixa de UHF contendo comunicações intermitentes conhecidas.

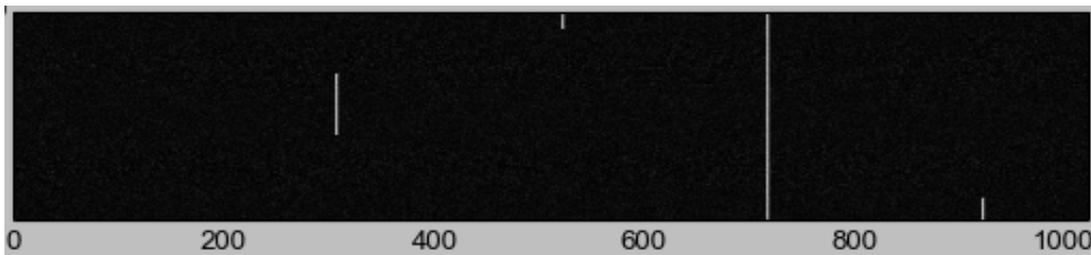


Figura 18 – Espectro de frequências *waterfall*

O funcionamento adequado para o algoritmo necessita de ajustes para cada um dos sinais, pois existem vários fatores que podem interferir nas características dos mesmos. Estes fatores podem ser: o nível de potência, o "pisso" de ruído e até o tratamento realizado na recepção.

Para avaliação dos resultados foram gerados espectrogramas dos sinais analisados. Esses espectrogramas são construídos por vetores, onde cada posição corresponde às amostras processadas na etapa de tratamento descrita na Seção 3.2, ou seja, um *tap*. A concentração de frequência por posição do vetor depende diretamente da largura de banda do sinal analisado.

Em todos os testes realizados, foram considerados somente sinais com largura de banda de 2MHz.

4.1 Testes com Sinais Sintéticos

Para comparação dos resultados nestes testes foi considerado uma DFT com tamanho de 1024 amostras, resultando em uma resolução aproximada de 2kHz por *tap*. Além disso, foram analisadas apenas 7 DFTs por segundo de forma a não onerar os recursos

de processamento do equipamento. A frequência de amostragem (f_a) é de 2MHz e os tamanhos das médias (J e L) serão fixados em 5 e 10 respectivamente. Por fim, os limiares de detecção (λ_{sup} e λ_{inf}) serão configurados com diferentes valores.

As amostras utilizadas foram geradas através de um *flowgraph* no GNU Radio. Esse *flowgraph* gera senoides e as soma a um sinal de distribuição gaussiana com média (μ) e variância (σ) 1, que representa o "piso" de ruído. Estas senoides tem a alternância de suas amplitudes entre 0 e 1, para simular quando há ou não atividade no espectro.

Nos próximos parágrafos serão apresentados alguns dos modelos de transmissão testados, e por fim o desempenho do algoritmo sobre esses testes referente a limiares de detecção distintos.

No modelo de teste 1, da Figura 19, foi avaliado o desempenho do algoritmo nos casos onde há a presença de sinais intermitentes juntamente de um sinal contínuo. Neste teste, por exemplo seriam esperadas detecções de intermitências nos *taps* de número 300, 500 e 900 aproximadamente.

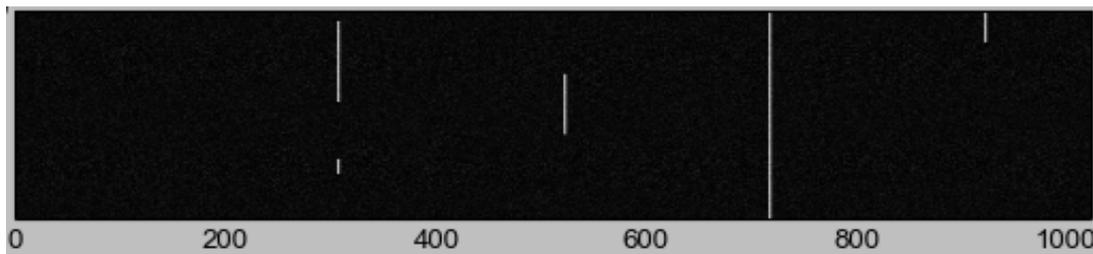


Figura 19 – Espectrograma *waterfall* do modelo de teste 1 com sinal gerado no GNU Radio

Nas situações presentes na Figura 20 espera-se que não haja detecção devido a presença apenas de sinais contínuos.

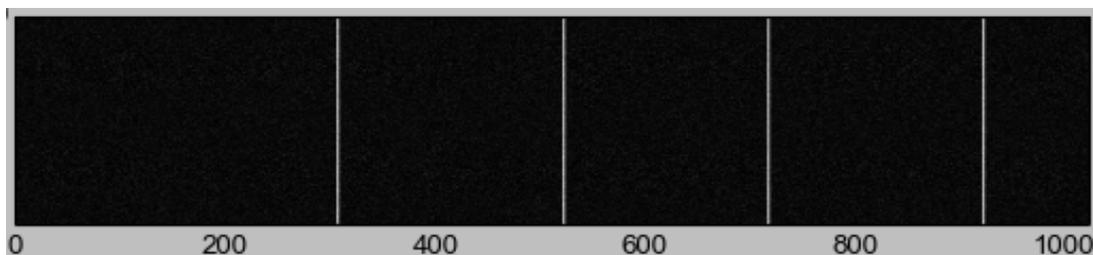


Figura 20 – Espectrograma *waterfall* do modelo de teste 2 com sinal gerado no GNU Radio

O modelo 3 foi simulado para situações onde há o surgimento de atividades, porém elas possuem uma duração elevada e podem ser consideradas atividades não intermitentes. A Figura 21 exemplifica essas situações. O esperado era que as atividades entre os *taps* 200 e 400, 400 e 600, 600 e 800 fossem consideradas intermitentes, porém o algoritmo deve

falhar nessas situações, pois as transmissões tem uma duração superior ao limite de espera do algoritmo para considerar o surgimento de alguma atividade nova nesses *taps*.

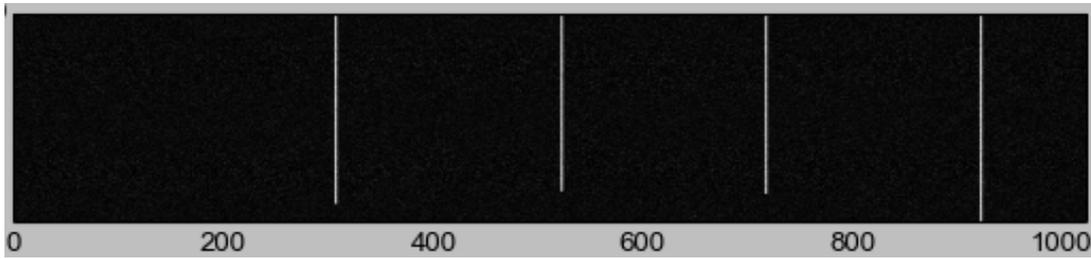


Figura 21 – Espectrograma *waterfall* do modelo de teste 3 com sinal gerado no GNU Radio

Existem algumas atividades que possuem a duração muito inferior a 1 segundo, estas são transmissões de sinalização no espectro. Essas atividades tem a característica de um "pulso" de sinal no espectro. O teste 4, da Figura 22, sugere um caso com "pulsos" de sinais, no qual o esperado era detectar as atividades nos *taps* 300, 500 e 700 aproximadamente. Em casos de "pulsos" o tamanho da DFT interfere no acerto do algoritmo, pois se a precisão for baixa um "pulso" pode se misturar ao ruído no espectro e não ser identificado.

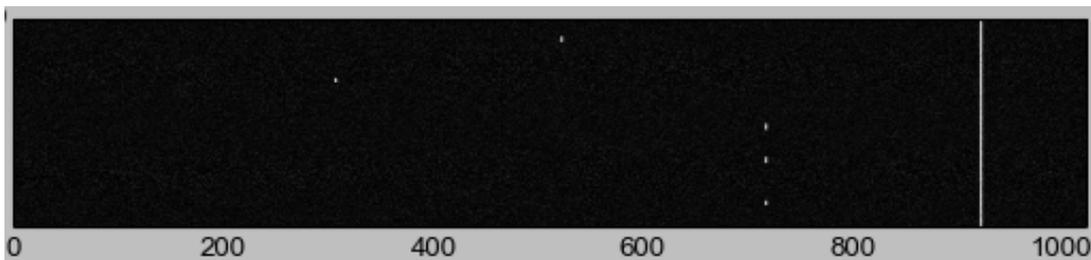


Figura 22 – Espectrograma *waterfall* do modelo de teste 4 com sinal gerado no GNU Radio

O modelo 5 testa o desempenho do algoritmo no surgimento e o cessar de atividades simultaneamente. Esse modelo está representado na Figura 23, na qual o espectro foi gerado com um único sinal em vários *taps*.

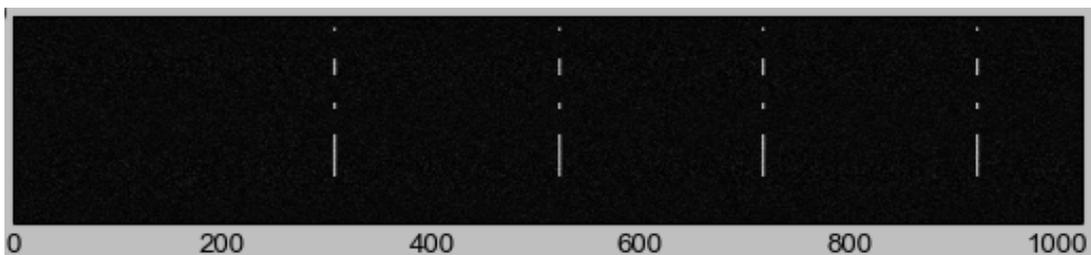


Figura 23 – Espectrograma *waterfall* do modelo de teste 5 com sinal gerado no GNU Radio

Nos testes descritos anteriormente, por se tratarem de um casos "ideais" a identificação das atividades como intermitentes acontece logo no segundo aparecimento de pico

da atividade, pois se trata de um cenário mais controlado, onde não há variações bruscas no "piso" de ruído e nas características dos sinais. Devido ao maior conhecimento sobre as características dos sinais analisados, a Tabela 3 apresenta os resultados dos testes com a variação dos limiares de detecção, considerando somente o ajuste de λ_{sup} e replicando-o para λ_{inf} negativamente. É importante lembrar que os limiares de detecção estão em escala linear.

Os resultados dos testes foram satisfatórios, porém algumas detecções ocorriam com atrasos. O ideal seria identificar o sinal na segunda intermitência que no *tap* correspondente. Embora a informação do atraso não apareça na Tabela 3, algumas atividades foram classificadas como intermitentes após vários pulsos surgirem e cessarem. Devido a potência dos picos das atividades, mesmo com os atrasos o percentual de acerto foi alto. Mas na medida em que o limiar era reduzido, os ruídos eram confundidos pelo algoritmo como atividade. Por outro lado quanto maior o limiar maior o número de atividades que deixavam de ser identificadas.

Tabela 3 – Resultados dos testes com sinais gerados.

Limiar λ_{sup}	Limiar λ_{inf}	Percentual de Acertos
3,5	-3,5	20%
4,5	-4,5	89%
5	-5	90%
6	-6	85%
7,5	-7,5	72%
8,5	-8,5	50%

4.2 Testes com Sinais Reais

Em casos reais a análise se torna mais complexa, pois depende de vários fatores. O limiar é uma das maiores fragilidades do algoritmo, pois sua determinação depende da posição do transmissor e do receptor, bem como a potência do sinal recebido. Dessa forma o limiar se torna um parâmetro muito incerto e específico para cada caso.

A seguir estão descritos alguns testes com sinais reais, estes adquiridos a partir do uso do USRP e do GNU Radio. Foram selecionadas algumas frequências centrais da faixa UHF. Em cada caso são descritas as configurações do algoritmo e discutidos os resultados da classificação.

Os testes a seguir foram realizados de duas formas distintas. Na primeira, foram escolhidos canais em um rádio *Talkabout* que opera nas frequências reservadas para *Family Radio Service* (FRS) e geradas transmissões. Na segunda, foram utilizadas transmissões

aleatórias que surgiam em algumas frequências da faixa UHF. Para realização dos testes uma pré configuração coerente com a faixa do espectro analisada é aplicada, para determinar os limiares e tamanhos das médias J e L . É importante enfatizar a redução dos limiares de detecção, pois a potência do sinal é inferior a dos sinais produzidos com o GNU Radio.

Os próximos três testes descritos seguem o primeiro formato. No primeiro caso de teste foi realizada a configuração do algoritmo com uma frequência central $f_c = 467MHz$, $\lambda_{sup} = 0.5$, $\lambda_{inf} = -0.5$, frequência de amostragem $f_a = 1MHz$, tamanho da média de janela $J = 4$ e tamanho da média temporal de $L = 10$. Na Figura 24 é apresentado o *waterfall* da banda com a transmissão adquirida. É possível observar que devido ao transmissor estar muito próximo, apareceram bandas laterais do mesmo sinal com uma potência menor. Estes sinais podem ser confundidos pelo algoritmo como outras transmissões e prejudicar a detecção. Nesse caso os limiares foram bem determinados e o algoritmo detectou atividades intermitentes nas posições 177 e 206, que são valores aproximados aos sinais da Figura 24.

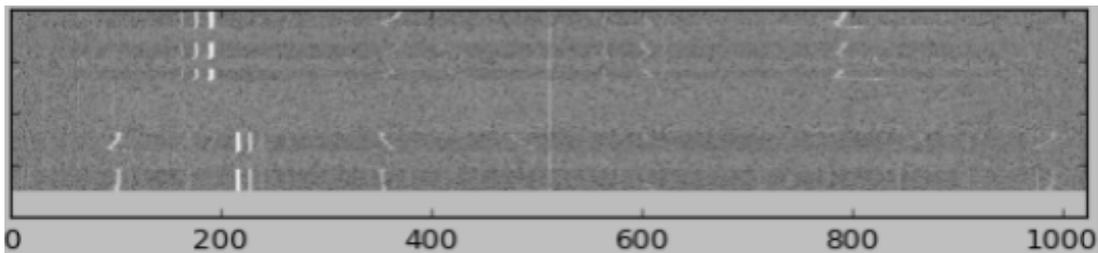


Figura 24 – Espectrograma *waterfall* do teste 6 com sinal gerado em 467 MHz

No próximo teste o valor de f_c é alterado, pois o rádio foi configurado com outros dois canais FRS. Para esse teste foi setado no algoritmo $f_c = 462MHz$, $\lambda_{sup} = 0.5$, $\lambda_{inf} = -0.5$, $f_a = 1MHz$, $J = 4$ e $L = 10$. Nesse teste alguns fatores prejudicaram a detecção de todas as atividades. Na Figura 25 é possível observar três atividades distintas para o lado esquerdo do *waterfall*. Porém, o algoritmo detectou intermitência apenas na posição 74 e sinalizou detecção de pico para as posições 51 e 70. Na posição 51 não houve a identificação de intermitência pois surgiu apenas um pico e durante a análise. Já na posição 70 a potência do sinal não foi alta o suficiente para o limiar configurado.

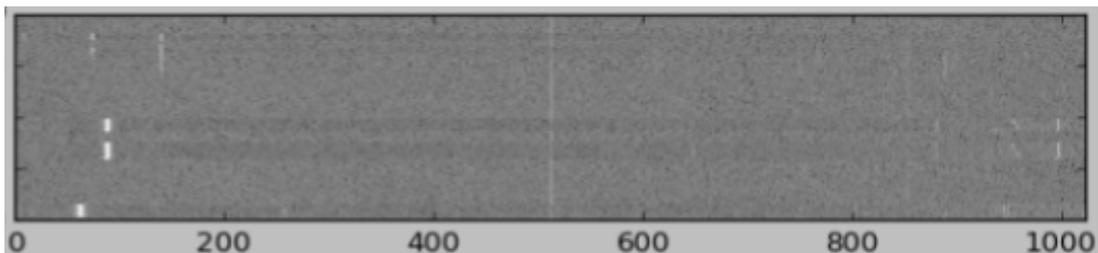


Figura 25 – Espectrograma *waterfall* do teste 7 com sinal gerado 462 MHz

O próximo caso de teste segue as mesmas configurações do teste anterior. Na

Figura 26 é visível apenas uma atividade no *waterfall*. Por não aparentar nenhum fator prejudicial o algoritmo detectou intermitência na posição 50.

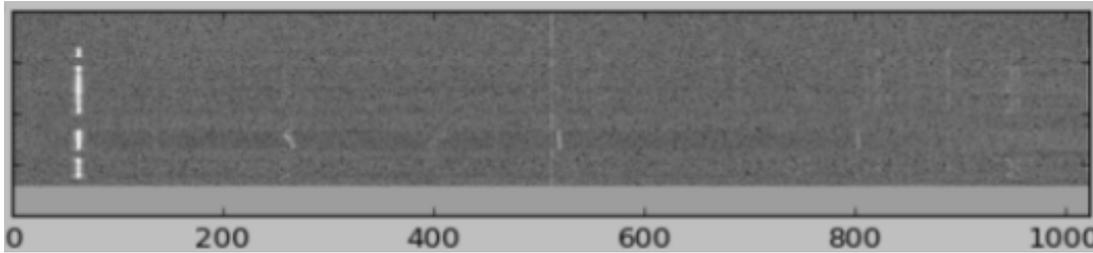


Figura 26 – Espectrograma *waterfall* do teste 8 com sinal gerado 462 MHz

Os próximos testes seguem o segundo formato descrito no início desta seção. O primeiro teste, Figura 27, foi realizado com frequência central $f_c = 459.3\text{MHz}$, $\lambda_{sup} = 0.2$, $\lambda_{inf} = -0.2$, frequência de amostragem $f_a = 5\text{MHz}$, tamanho do filtro de janela $J = 5$ e tamanho do filtro temporal $L = 9$. Nesse teste o algoritmo identificou apenas duas atividades intermitentes, em 430 e 494. Apesar das condições adversas do espectro, nenhum dos parâmetros do algoritmo precisou ser alterado. A identificação condiz aproximadamente com as informações presentes na Figura 27.

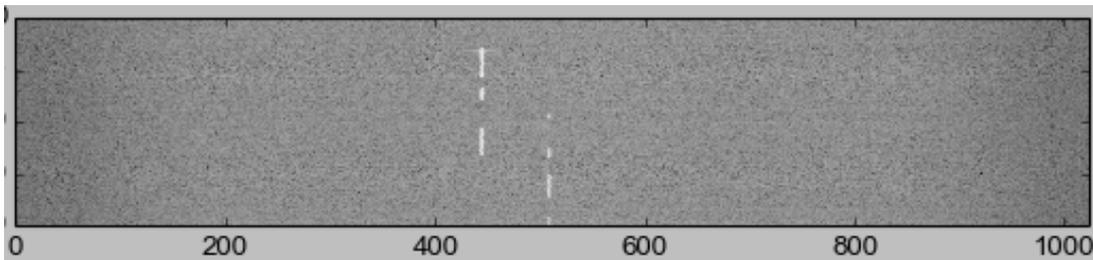


Figura 27 – Espectrograma *waterfall* do teste 9 com sinal gerado 459.3 MHz

O próximo teste seguiu as mesmas configurações do teste anterior (Figura 27). O algoritmo identificou intermitência na posição 494 e a Figura 28 confirma a detecção com uma atividade aparente próxima à posição identificada pelo algoritmo.

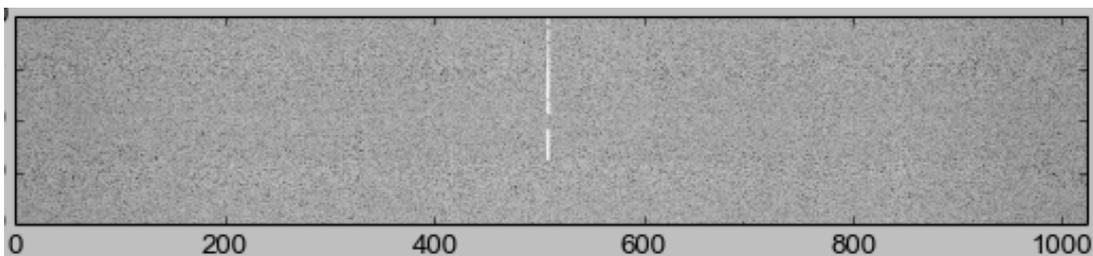


Figura 28 – Espectrograma *waterfall* do teste 10 em 459.3 MHz

Na Figura 29 temos um caso onde a potência do sinal identificado pelo algoritmo na posição 152 era suficientemente alta para atingir o limiar, porém os pulsos não foram identificados como intermitência.

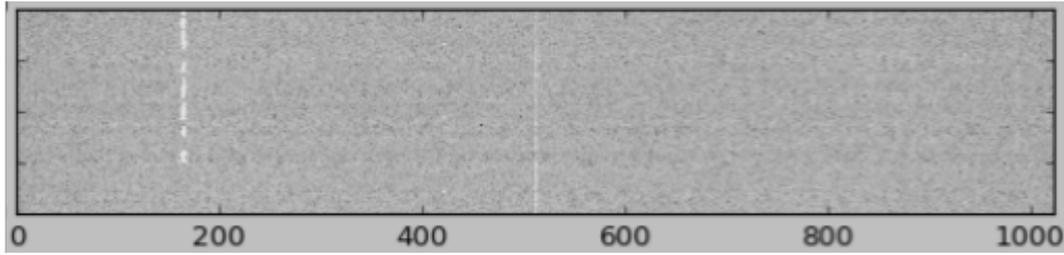


Figura 29 – Espectrograma *waterfall* do teste 11 em 859 MHz

Para a aquisição da Figura 30 o algoritmo foi configurado com $f_c = 859\text{MHz}$, $\lambda_{sup} = 0.5$, $\lambda_{inf} = -0.5$, $f_a = 1\text{MHz}$, $J = 5$ e $L = 10$. Nesse cenário foi identificada a intermitência na posição 152, porém em 665 só identificou o início da atividade, o que não a classifica como intermitente.

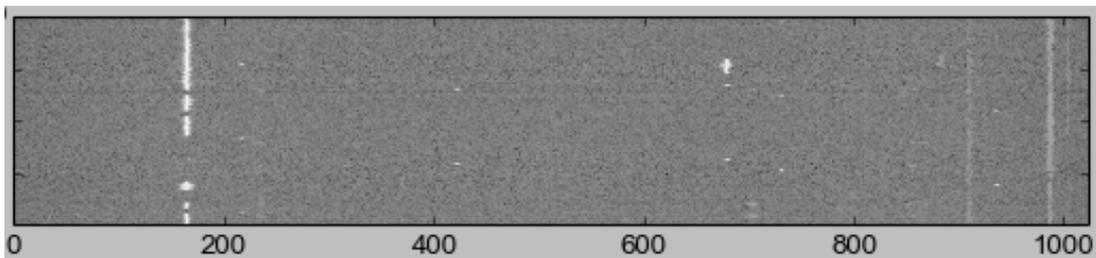


Figura 30 – Espectrograma *waterfall* do teste 12 em 859 MHz

Na aquisição da Figura 31 os limiares de detecção foram alterados para $\lambda_{sup} = 0.4$, $\lambda_{inf} = -0.4$. Nesse caso identificou o início da transmissão na posição 153 e na 206, mas devido ao baixo nível de potência da atividades da posição 106, não houve o reconhecimento desta intermitência. Por outro lado, ao reduzir o limiar, os “pulsos” passam a ser detectados, mas sob os riscos de interferência do “pisso” de ruído.

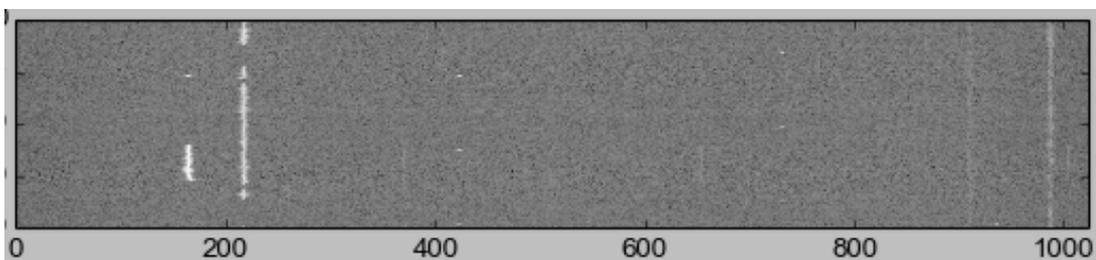


Figura 31 – Espectrograma *waterfall* do teste 13 em 859 MHz

Nesta seção foi possível observar o funcionamento do algoritmo. Ele reconhece as atividades intermitentes no espectro em casos controlados. Porém, vários fatores influenciam para o mau funcionamento, como nível de potência do sinal, escolha de limiares, “pisso” de ruído, qualidade do receptor, entre outras já citadas anteriormente. O algoritmo poderia sofrer melhorias para casos reais. As melhorias poderiam ocorrer através do tratamento das amostras, para redução das interferências e sinais ruidosos. Outro ponto a ser atingido

seria uma melhor definição de parâmetros utilizados para a identificação, ou até mesmo um ajuste automático dos parâmetros.

5 Conclusões

Neste trabalho de conclusão de curso foi proposta a implementação de um sistema que possibilita a identificação das intermitências de sinais de rádio no espectro. Essa identificação acontece através de quatro estados, são estes: sem transmissão, início de transmissão, transmitindo e fim de transmissão. Cada um dos estados gera uma característica diferente no sinal resultante do algoritmo. Esse sinal é avaliado através de dois limiares, superior e inferior. Dependendo das sequências e duração desses estados, são obtidas as informações de intermitência.

Para tornar possível a identificação dessas atividades foi utilizado um sistema que realiza a aquisição de amostras. Tal sistema, é composto por um receptor SDR e um *host*. As amostras são coletadas pelo receptor SDR e, posteriormente, processadas no *host*. Os processamentos realizados no *host* incluem o tratamento do sinal adquirido e o processo de detecção.

O tratamento do sinal condiciona as amostras adquiridas para a realização da detecção de intermitências. Ao final do tratamento, o sinal é passado para a etapa de detecção no domínio da frequência. O fato de o sinal estar no domínio da frequência facilita a identificação, pois a determinação visual dos limiares é simplificada, considerando que é possível visualizar os picos de concentração de energia do sinal.

O método de detecção de atividades intermitentes foi avaliado levando em consideração o espectrograma, ou *waterfall*, dos sinais. Através do conhecimento do comportamento das intermitências no espectro, houve a comparação dos resultados do algoritmo à imagem do espectrograma. Com a execução dos testes em cenários distintos foi possível concluir que o algoritmo funciona muito bem em sinais gerados com características ideais, mas não traz bons resultados em cenários reais. Os fatores, como potência dos sinais, outras atividades na banda analisada, e ajuste dos limiares de detecção podem prejudicar a identificação da intermitência. Outro problema observado no algoritmo é o atraso. Em alguns testes há a identificação correta das intermitências, porém acontece após vários pulsos surgirem e cessarem. Esse atraso se torna prejudicial em casos de necessidade do conhecimento imediato da informação. Esses resultados ressaltam a necessidade de ajustes no algoritmo, como uma determinação de parâmetros de configuração de forma automática, pois atualmente a determinação é feita através de análises visuais.

Para futuros trabalhos sugere-se a melhoria do algoritmo, para torná-lo mais robusto às variações do ambiente, além de uma implementação mais específica para cada faixa de frequências analisada. Outra ideia é a implementação de um forma de identificação de intermitências utilizando aprendizado de máquina ou outros métodos de detecção, para

comparação de desempenho entre métodos.

Referências

AGILENT TECHNOLOGIES. *Testando interferências com analisadores de espectros de mão*. [S.l.], 2012. 22 p. Disponível em: <<http://literature.cdn.keysight.com/litweb/pdf/5990-9074PTBR.pdf>>. Acesso em: 20 nov 2017. Citado 2 vezes nas páginas 11 e 25.

ALMEIDA, E. P. L. de. *Desenvolvimento de Técnica de Sensoriamento do Espectro Embasada em Detecção de Energia para Aplicações em Sistemas Rádio-Cognitivos*. Dissertação (Mestrado) — Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, Brasil, fev. 2010. Citado 5 vezes nas páginas 21, 30, 31, 32 e 33.

COIMBRA, T. R. *Regulação do Espectro de Radiofrequências: Análise técnica do modelo brasileiro*. 2005. Disponível em: <<http://www.teleco.com.br/pdfs/tutorialespeccradio.pdf>>. Acesso em: 27 jun 2017. Citado na página 21.

ETTUS. In: USRP N210 KIT. Ettus Research, 2017. Disponível em: <<https://www.ettus.com/product/details/UN210-KIT>>. Acesso em: 20 out 2017. Citado 2 vezes nas páginas 11 e 36.

ETTUS. In: PRODUCTS. Ettus Research, 2017. Disponível em: <<https://www.ettus.com/product>>. Acesso em: 21 out 2017. Citado 3 vezes nas páginas 13, 29 e 30.

ETTUS. In: USRP N200/N210 Datasheet. Ettus Research, 2017. Disponível em: <https://www.ettus.com/content/files/07495_Ettus_N200-210_DS_Flyer_HR_1.pdf>. Acesso em: 21 out 2017. Citado na página 36.

FERREIRA, D. A. *Sistema de detecção de atividade em tempo real no espectro radioelétrico com reconhecimento automático de modulação analógica*. Monografia — Curso de Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, Brasil, 2014. Citado 10 vezes nas páginas 21, 26, 27, 28, 29, 30, 31, 32, 33 e 36.

GNURADIO. In: GNU Radio. John Gilmore, 2014. Disponível em: <https://en.wikipedia.org/wiki/GNU_Radio>. Acesso em: 10 out 2017. Citado na página 28.

GNURADIO. In: WHAT is GNU Radio? Wiki GNU Radio, 2017. Disponível em: <https://wiki.gnuradio.org/index.php/What_is_GNU_Radio>. Acesso em: 10 out 2017. Citado na página 28.

GNURADIO. In: GNU Radio Companion. Wiki GNU Radio, 2017. Disponível em: <<https://wiki.gnuradio.org/index.php?title=GNURadioCompanion&action=history>>. Acesso em: 10 out 2017. Citado na página 28.

GREGORIM, C. O. et al. *MICHAELIS: Dicionário Prático Língua Portuguesa*. São Paulo, SP, Brasil: Editora Melhoramentos, 2008. v. 1. Citado na página 25.

INÁCIO, J. C. *Implementação de um Rádio Definido por Software usando MATLAB e um par Transmissor/Receptor FM*. Monografia — Curso de Tecnologia em Sistemas de Telecomunicações, Instituto Federal de Santa Catarina, Campus São José, São José SC, Brasil, 2011. Citado 2 vezes nas páginas 26 e 27.

JANSON, J. *Rádio Definido por Software: Estudo e realização de Testes com Plataforma livre*. Monografia — Curso de Tecnologia em Sistemas de Telecomunicações, Instituto Federal de Santa Catarina, São José, SC, Brasil, 2014. Citado na página 29.

LIMA, A. D. de. *Arquitetura Eficiente para Sensoriamento Espectral e Classificação Automática de Modulação Utilizando Características Cicloestacionárias*. Monografia — Programa de Pós-graduação em Engenharia Elétrica e Computação, Universidade Federal do rio Grande do Norte, 2014. Citado na página 21.

NUMPY. In: NUMPY. Num Focus, 2017. Disponível em: <<https://www.numpy.org/>>. Acesso em: 17 out 2017. Citado na página 30.

SCIPY. In: TUTORIAL. Enthought, 2008–2016. Disponível em: <<https://docs.scipy.org/doc/scipy/reference/tutorial/general.html>>. Acesso em: 17 out 2017. Citado na página 30.

STEWART, B. et al. *Software Defined Radio using MATLAB & Simulink and the RTL-SDR*. Glasgow, Scotland, UK: Department of Electronic and Electrical Engineering University of Strathclyde, 2015. v. 1. Citado 3 vezes nas páginas 21, 26 e 27.

TELECO. *Estatísticas de Celulares no Brasil*. 2017. Disponível em: <<http://www.teleco.com.br/ncel.asp>>. Acesso em: 27 jun 2017. Citado na página 21.

URKOWITZ, H. Energy detection of unknown deterministic signals. *Proceedings of the IEEE*, v. 55, n. 4, p. 523–531, April 1967. ISSN 0018-9219. Citado na página 32.

WYGLINSKI, A. M. et al. *Cognitive Radio Communications and Networks: Principles and Practice*. MA, Burlington, USA: Academic Press is an imprint of Elsevier, 2010. 2-6 p. Citado 2 vezes nas páginas 21 e 26.

YUCEK, T.; ARSLAN, H. A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Communications Surveys Tutorials*, v. 11, n. 1, p. 116–130, First 2009. ISSN 1553-877X. Citado 3 vezes nas páginas 30, 31 e 32.