

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**IDENTIFICAÇÃO DE SINAIS RADAR PULSADOS POR
MEIO DE TRANSFORMADA DE WAVELET CONTÍNUA E
REDES NEURAS ARTIFICIAIS**

ANDRÉ LUIZ BANDEIRA MOLINA

ORIENTADOR: FRANCISCO ASSIS DE OLIVEIRA NASCIMENTO

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA

PUBLICAÇÃO: PPGENE.DM – 287A/06

BRASÍLIA/DF: DEZEMBRO – 2006

FICHA CATALOGRÁFICA

MOLINA, ANDRÉ LUIZ BANDEIRA

Identificação de Sinais Radar Pulsados Por Meio de Transformada de Wavelet Contínua e Redes Neurais Artificiais [Distrito Federal] 2006.

xviii, 94p., 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2006)

Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Elétrica.

1. Transformada de wavelet

2. Redes neurais artificiais

3. Identificação de assinatura espectral

4. Sinal radar pulsado

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

MOLINA, A. L. B. (2006). Identificação de Sinais Radar Pulsados Por Meio de Transformada de Wavelet Contínua e Redes Neurais Artificiais. Dissertação de Mestrado em Engenharia Elétrica, Publicação PPGENE.DM-287A/06, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 94p.

CESSÃO DE DIREITOS

AUTOR: André Luiz Bandeira Molina.

TÍTULO: Identificação de Sinais Radar Pulsados Por Meio de Transformada de Wavelet Contínua e Redes Neurais Artificiais.

GRAU: Mestre

ANO: 2006

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.

André Luiz Bandeira Molina
SQS 113, Bloco B, ap 605, Brasília.
70376-020 Brasília – DF – Brasil.

Às memórias de meus avós,
Waldemar da Costa Bandeira e
Felicidade Rodrigues da Silva Bandeira,
e de meu tio, Adenélcio da Costa Bandeira.
Desejo-lhes muita paz e harmonia,
e que esta dedicatória eternize
suas presenças e o que vocês
representaram para mim.

AGRADECIMENTOS

Depois desta vitória, faz-se necessário reconhecer a importância daqueles que estiveram presentes nesta caminhada.

Primeiramente, agradeço a Deus, por permitir a conclusão de mais uma jornada.

Agradeço à minha família, que sempre me incentivou e apoiou nos desafios aos quais me propus, e à minha querida Patrícia, pela paciência, carinho e compreensão sem limites.

Ao meu orientador, professor Francisco Assis, cuja ajuda foi imprescindível para a conclusão desta dissertação, e que mostrou ser um grande amigo em todas as horas.

Agradeço aos meus companheiros de trabalho do Centro de Computação da Aeronáutica de Brasília, que me apoiaram durante todos os momentos difíceis dessa jornada e, em especial, ao Tenente Coronel Selmo Maciel Tupinambá, cuja compreensão e incentivo permitiram a dedicação necessária para a conclusão deste Mestrado.

Agradeço, por fim, a todos os professores desta instituição e aos colegas que fiz.

Muito Obrigado.

RESUMO

IDENTIFICAÇÃO DE SINAIS RADAR PULSADOS POR MEIO DE TRANSFORMADA DE WAVELET CONTÍNUA E REDES NEURAIAS ARTIFICIAIS

Autor: André Luiz Bandeira Molina

Orientador: Francisco Assis de Oliveira Nascimento

Programa de Pós-graduação em Engenharia Elétrica

Brasília, dezembro de 2006

O *Deinterleaving* de seqüências de pulsos de sinais radar é uma tarefa essencial para a identificação de radares em guerra eletrônica. Algumas formas de se realizar o *deinterleaving* baseiam-se em técnicas temporais, que utilizam histogramações e recursividade, e em técnicas baseadas em transformadas ortogonais, que utilizam as assinaturas espectrais para detecção das seqüências. Qualquer uma das abordagens onera grande esforço computacional. Entretanto, as técnicas que utilizam transformadas ortogonais são mais robustas devido às propriedades de minimização da correlação entre as diversas fontes de sinais. As assinaturas espectrais no espaço transformado apresentam um comportamento típico mais evidente.

Este trabalho propõe uma técnica composta de três etapas distintas. Primeiramente, tem-se um pré-processamento pelo qual toda a faixa de valores de interesse é segmentada em sub-faixas por meio de subamostragens e filtragem digital. Em seguida, utiliza-se a análise tempo-freqüência por meio da transformada de *wavelet* contínua de forma a separar os padrões espectrais de interesse. Na última etapa, é implementado um processo de identificação de assinatura espectral baseado em redes neurais artificiais. Também é apresentada toda a formulação matemática necessária para o estabelecimento das condições de contorno e elaboração dos procedimentos dedicados.

Os algoritmos foram desenvolvidos em linguagem MATLAB para simulação e validação dos modelos. Os resultados de simulação computacional mostram um bom desempenho e robustez desses algoritmos quando uma grande quantidade de fontes de sinais radar pulsados está presente. O modelo proposto fornece boa precisão da assinatura espectral, o que possibilita a identificação dos sinais radar pelas redes neurais artificiais.

ABSTRACT

IDENTIFICATION OF PULSED RADAR SIGNALS BY MEANS OF WAVELET CONTINUOUS TRANSFORM AND ARTIFICIAL NEURAL NETWORKS

Author: André Luiz Bandeira Molina

Supervisor: Francisco Assis de Oliveira Nascimento

Programa de Pós-graduação em Engenharia Elétrica

Brasília, December of 2006

The deinterleaving of radar signal pulse sequences is an essential work for the radar identification in electronic warfare. Some ways of performing deinterleaving are based on temporal techniques, that use histogram and recursivity, and in techniques based on orthogonal transforms, that use spectral signatures for the detection of the sequences. Any of them require heavy computational power. However, the orthogonal transform techniques are more robust due to the properties of correlation minimization among the several signal fonts. The spectral signatures in the transformed space show a typical behavior more clearly.

This work proposes a technique composed of three distinct phases. First, there is a preprocessing by which the whole interval of interest is segmented into sub intervals by means of downsampling and digital filtering. After that, it is used a time-frequency analysis by means of continuous wavelet transform, separating the spectral patterns of interest. Finally, it is implemented an identification process of spectral signature based on artificial neural networks. It is also described all the necessary mathematics concepts to the establishment of the contour conditions and elaboration of the dedicated procedures.

The algorithms were developed in MATLAB for the simulation and validation of the models. The results of the computer simulation show a good performance and robustness of these algorithms when several sources of pulsed radar signals are present. The proposed

model gives a good precision for the spectral signature, what makes the identification of radar signals possible by means of the artificial neural networks.

SUMÁRIO

1 - INTRODUÇÃO	1
1.1 - APRESENTAÇÃO DO PROBLEMA	1
1.2 - ESTADO DA ARTE.....	2
1.3 - SOLUÇÃO PROPOSTA.....	4
1.4 - ORDEM DE APRESENTAÇÃO	5
2 - RADARES, RECEPTORES E PROCESSAMENTO EM GUERRA ELETRÔNICA	7
2.1 - RADARES.....	7
2.1.1 - Medição de Distância e Resolução	11
2.1.2 - Tipos de Radares em Guerra Eletrônica.....	13
2.2 - RECEPTORES E PROCESSAMENTO EM GE.....	14
2.2.1 - Receptores de GE	14
2.2.2 - Processamento em GE.....	16
2.3 - DEINTERLEAVING	18
2.3.1 - Técnicas de Deinterleaving	20
3 - FERRAMENTAS MATEMÁTICAS UTILIZADAS NA SOLUÇÃO DO PROBLEMA	22
3.1 - TRANSFORMADA DE WAVELET CONTÍNUA	22
3.1.1 - Técnica de <i>Deinterleaving</i> por meio da Transformada de <i>Wavelet</i> Contínua	26
3.2 - FILTRAGEM DIGITAL FIR	33
3.3 - SUBAMOSTRAGEM DE SINAIS	37
3.4 - REDES NEURAS ARTIFICIAIS.....	39
4 - IDENTIFICAÇÃO DE SINAIS RADAR PULSADOS POR MEIO DE TRANSFORMADA DE WAVELET CONTÍNUA E REDES NEURAS ARTIFICIAIS	46
4.1 - Problemas da Técnica de Deinterleaving por meio da Transformada de Wavelet Contínua	46
4.2 - Modificação da Técnica de Deinterleaving por Meio da Transformada de Wavelet Contínua	47
4.3 - Identificação de Sinais Radar por meio de Redes Neurais Artificiais.....	53

5 - IMPLEMENTAÇÃO, RESULTADOS DAS SIMULAÇÕES E DISCUSSÃO.....	58
5.1 - Implementação do Sistema de Identificação de Sinais Radar Proposto	58
5.2 - Simulações e Análise dos Resultados	65
6 - CONCLUSÕES E RECOMENDAÇÕES	73
REFERÊNCIAS BIBLIOGRÁFICAS	75
APÊNDICES	
A – LISTAGEM DOS PROGRAMAS	80

LISTA DE TABELAS

Tabela 2.1 - Designações Adotadas para Bandas de Freqüência Inferiores a 40GHz.....	7
Tabela 2.2 - Designações Adotadas para Bandas de Freqüência Superiores a 40GHz.	7
Tabela 2.3 – Formato típico de PDW (TSUI, 1995).	15
Tabela 2.4 – Informações disponíveis para cada pulso de acordo com o tipo de receptor (ADAMY, 2001).	19
Tabela 5.1 – Seqüências de pulsos de sinais radar ilustrativas.....	59
Tabela 5.2 – Simulação das seqüências de sinais radar que se deseja identificar.	65
Tabela 5.3 – Seqüências de pulsos radar que compõem o sinal $s_v(t)$ para validação.	68

LISTA DE FIGURAS

Figura 1.1 – Parâmetros geralmente medidos para um sinal pulsado (modificado – SCHLEHER, 1999).	3
Figura 2.1 – Deslocamento na frequência provocado pelo efeito Doppler entre os sinais transmitido e refletido (modificado – NOEL, 1998).	8
Figura 2.2 – Modulação em frequência para sinal CW com evidente marcação no tempo para os sinais emitido e refletido (modificado – NOEL, 1998).	9
Figura 2.3 – Parâmetros característicos de radares pulsados. Parâmetros Intrapulso: PA – Amplitude de Pulso (<i>Pulse Amplitude</i>), PW – Largura de Pulso (<i>Pulse Width</i>), RF – Radiofrequência, TOA – Tempo de Chegada (<i>Time of Arrival</i>) e AOA – Angulo de Chegada (<i>Angle of Arrival</i>). Parâmetros Interpulso: PRI – Intervalo de Repetição de Pulsos (<i>Pulse Repetition Interval</i>), PRF – Frequência de Repetição de Pulsos (<i>Pulse Repetition Frequency</i>), e SR – Taxa de Varredura (Scan Rate).	10
Figura 2.4 – Padrões clássicos de PRI e diferença dos tempos de chegada (TOA) entre pulsos consecutivos para: a.1 e a.2 – PRI tipo Simples de valor 10, sem variação; b.1 e b.2 – PRI <i>Jitter</i> de valor central 5 com variação de até 10% em torno do valor central; c.1 e c.2 – PRI <i>Stagger</i> de valores 2.0, 2.2, 2.4 e 2.6; d.1 e d.2 – PRI <i>Wobulated</i> , com valor central 5, variação de até 10% e 50 pulsos por ciclo.	11
Figura 2.5 – Sinais pulsados de três emissores (Sinal A, Sinal B e Sinal C) ao longo do tempo e o sinal interceptado pelo receptor (A+B+C).	18
Figura 3.1 – Resolução no plano tempo-frequência da <i>Short Time Fourier Transform</i> (modificado – MATLAB).	23
Figura 3.2 – Resolução no plano tempo-frequência obtida pela Transformada de <i>Wavelet</i> (modificado – MATLAB).	24
Figura 3.3 – Processo de análise <i>wavelet</i> de um sinal para uma função $\psi(t)$ deslocada e escalonada (modificado – MATLAB).	26
Figura 3.4 – Gráficos da <i>wavelet</i> -mãe $\psi(t)$ para M igual a 12: parte real, parte imaginária, módulo e espectro.	27
Figura 3.5 – Gráficos da função detectora $D(T,t)$ para 3 sinais, no instante de tempo $t=70$, antes e após a limiarização com valor 0,32.	29
Figura 3.6 – Sinal $s(t)$ formado pelos sinais $s_1(t)$, $s_2(t)$ e $s_3(t)$ superpostos.	30

Figura 3.7 – Gráfico da função detectora $D(T,t)$ para o sinal $s(t)$ contendo os sinais $s_1(t)$, $s_2(t)$ e $s_3(t)$ superpostos.	31
Figura 3.8 - Gráficos da <i>wavelet</i> -mãe Morlet para M igual a 12: parte real, parte imaginária, módulo e espectro.....	32
Figura 3.9 – Gráfico da função detectora $D(T,t)$ para o sinal $s(t)$ contendo os sinais $s_1(t)$, $s_2(t)$ e $s_3(t)$ superpostos.	32
Figura 3.10 – Estrutura básica de um filtro FIR (ROCHA, 2006).	34
Figura 3.11 – Janelas Bartlett, Hamming, Hanning e retangular.	36
Figura 3.12 – Componentes de frequência do sinal $x(t)=sen(2\cdot\pi\cdot0,2\cdot t)+ sen(2\cdot\pi\cdot4\cdot t)$, amostrado a uma taxa de 10 amostras/segundo.....	38
Figura 3.13 – Efeito superposição de espectros observado para as componentes de frequência de $x(t)$ quando subamostrado a uma taxa inferior à frequência de Nyquist.....	38
Figura 3.14 – Componentes de frequência de $x(t)$ após filtragem e subamostragem.....	39
Figura 3.15 – Neurônio booleano de McCulloch e Pitts.	40
Figura 3.16 – Perceptron de Rosenblatt.....	41
Figura 3.17 – Funções de ativação mais comuns: a) Linear; b) Sigmóide logística; c) Tangente hiperbólico; e d) Degrau.	42
Figura 3.18 – <i>Perceptron</i> com uma única camada (modificado – BAUCHSIPESS, 2004).	43
Figura 3.19 – Rede <i>perceptron</i> multicamadas, com duas camadas, dois neurônios na camada oculta e um neurônio na última (BAUCHSIPESS, 2004).....	43
Figura 4.1 – Gráfico representativo das janelas de tempo necessárias para análise das sub-faixas de PRI prático.	50
Figura 4.2 – Janelas do sinal recebido para análise das sub-faixas de PRI prático após subamostragens.....	50
Figura 4.3 – Recepção de sinais radar pulsados modulados e conversão em sinais de onda quadrada com largura de pulso.....	51
Figura 4.4 – Diagrama em blocos do sistema de <i>deinterleaving</i> de sinais radar pulsados por meio de técnica da transformada de <i>wavelet</i> contínua modificada com subamostragens e filtragem digital.	53
Figura 4.5 – Modelo de identificação de radares por meio de rede Fuzzy Artmap (modificado – GRANGER <i>et al</i> , 2001).	54
Figura 4.6 – Esquema de conexão entre as redes <i>perceptron</i> multicamadas e as saídas da técnica de <i>deinterleaving</i> modificada.	55

Figura 4.7 – Estrutura da rede MLP proposta. Há 91 entradas, uma camada intermediária com número de nodos definidos empiricamente e uma camada de saída com número de nodos definidos em função do número de sinais radar que se deseja identificar.	56
Figura 4.8 – Modelo do sistema de identificação de sinais radar por meio de transformada de <i>wavelet</i> contínua e redes neurais artificiais.....	57
Figura 5.1 – Seqüências de sinais radar ilustrativas e seus espectros de Fourier: Sinal A: Simples; Sinal B: <i>Stagger</i> ; Sinal C: <i>Wobulated</i> ; Sinal D: <i>Jitter</i> ; Sinal A+B+C+D: Superposição dos sinais.	59
Figura 5.2 - Detalhes do filtro FIR passa-faixa projetado pela técnica de projeto de fase linear por janelamento da resposta impulsional desejada, com janela Hamming e ordem 150 para freqüência de corte normalizada entre 0,02 e 0,2: a) Resposta ao impulso; b) Magnitude (dB) e fase (graus).	60
Figura 5.3 – Técnica de <i>deinterleaving</i> por meio da transformada de <i>Wavelet</i> Contínua: a) Parte real da <i>wavelet</i> -mãe centralizada; b) Parte imaginária da <i>wavelet</i> -mãe centralizada; c) Escalograma normalizado da energia espectral obtido pela função detectora $D(PRI,t)$ para 4095 elementos após técnica da FFT; d) Escalograma normalizado da energia espectral truncado em 2048 pontos e limiarizado em 15% do valor do máximo.	62
Figura 5.4 – Desenho da rede MLP designada para identificação dos sinais radar A, B, C e D na sub-faixa de PRI entre 1 e 10 μ s (MATLAB).	63
Figura 5.5 – Conjunto de treinamento da rede MLP associada à sub-faixa de PRI entre 1 e 10 μ s para identificação dos sinais radar A, B, C e D.	64
Figura 5.6 – Formato das redes MLP para identificação dos sinais radar $s_1(t)$, $s_2(t)$, $s_3(t)$ e $s_4(t)$	66
Figura 5.7 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 1 e 10 μ s formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_1(t)$; b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_1(t)$; d) Saída desejada da rede para $s(t)$	66
Figura 5.8 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 10 e 100 μ s formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_2(t)$; b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_2(t)$; d) Saída desejada da rede para $s(t)$	66
Figura 5.9 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 100 e 1000 μ s formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_3(t)$;	

b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_3(t)$; d) Saída desejada da rede para $s(t)$	67
Figura 5.10 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 1000 e 10000 μ s formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_4(t)$; b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_4(t)$; d) Saída desejada da rede para $s(t)$	67
Figura 5.11 – Performances de treinamento para: a) Rede MLP associada à sub-faixa de PRI entre 1 e 10 μ s; b) Rede MLP associada à sub-faixa de PRI entre 10 e 100 μ s; c) Rede MLP associada à sub-faixa de PRI entre 100 e 1000 μ s; d) Rede MLP associada à sub-faixa de PRI entre 1000 e 10000 μ s.....	68
Figura 5.12 – Validação do modelo proposto para o sinal $s_v(t)$ com treinamento para identificação dos sinais $s_1(t)$, $s_2(t)$, $s_3(t)$ e $s_4(t)$. Acima estão as detecções da técnica da transformada de <i>wavelet</i> contínua para cada sub-faixa de PRI, e abaixo, as respectivas saídas das redes MLP associadas.	69
Figura 5.13 – Resultado da detecção e identificação do sinal $s_v(t)$ após modificação dos parâmetros e conjunto de treinamento.....	71

LISTA DE SÍMBOLOS, NOMENCLATURAS E ABREVIACÕES

AOA – *Angle of Arrival*

DOA – *Direction of Arrival*

DSP – *Digital Signal Processor*

CME – *Contramedidas Eletrônicas*

CVR – *Cristal Video Receiver*

CW – *Continuous Wave*

CWT – *Continuous Wavelet Transform*

DWT – *Discrete Wavelet Transform*

EHF – *Extremely High Frequency*

FIR – *Finite Impulse-duration Response*

FFT – *Fast Fourier Transform*

GE – *Guerra Eletrônica*

IIR – *Infinite Impulse-duration Response*

MAE – *Medidas de Apoio Eletrônico*

MAGE – *Medidas de Apoio a Guerra Eletrônica*

MCP – *McCulloch e Pitts*

MPE – Medidas de Proteção Eletrônica

PA – *Pulse Amplitude*

PDW – *Pulse Descriptor Word*

PR – *Pulsed Radars*

PRI – *Pulse Repetition Interval*

PRF – *Pulse Repetition Frequency*

PW – *Pulse Width*

RADAR – *Radio Detection and Ranging*

RF – *Radio Frequency*

RGE – Receptor de Guerra Eletrônica

RWR – *Radar Warning Receiver*

SR – *Scan Rate*

STFT – *Short Time Fourier Transform*

TOA – *Time of Arrival*

TW – Transformada de Wavelet

UHF – *Ultra High Frequency*

VHF – *Very High Frequency*

1 - INTRODUÇÃO

1.1 - APRESENTAÇÃO DO PROBLEMA

Guerra Eletrônica (GE) é conceituada como um conjunto de ações e conceitos militares que tem por objetivo controlar e utilizar o espectro eletromagnético por forças amigas e negar o seu uso e controle pelas forças inimigas.

Aliada a essas ações e conceitos, a GE pode ser dividida em:

- a) Medidas de Apoio à Guerra Eletrônica (MAGE) – responsáveis por coletar e processar informações do ambiente eletromagnético;
- b) Medidas de Ataque Eletrônico (MAE) – responsáveis por evitar, degradar, neutralizar, danificar, interromper, destruir ou enganar a utilização do espectro eletromagnético por parte do inimigo de forma a diminuir a eficácia de suas ações;
- c) Medidas de Proteção Eletrônica (MPE) – responsáveis pela proteção contra a utilização de MAE pelo inimigo e por garantir o uso eficaz do espectro eletromagnético por parte das forças amigas.

As MAGE, em especial, focam em ações que têm por propósito o reconhecimento de ameaças, praticamente em tempo real, apoiando medidas imediatas envolvendo MAE, MPE, etc. As principais atividades realizadas são interceptar, identificar, analisar e localizar radiações inimigas.

As radiações eletromagnéticas são geralmente interceptadas por meio de receptores que cobrem as bandas de frequências associadas às ameaças significativas. A identificação envolve a comparação dos dados interceptados com assinaturas de ameaças armazenadas em uma biblioteca de ameaças. Localização é geralmente realizada por meio da combinação de interceptações espacialmente dispersas (SCHLEHER, 1999).

Um dos equipamentos com esses requisitos é o Receptor de Alerta Radar (*Radar Warning Receiver* – RWR), que é responsável por detectar, identificar e classificar sinais radar de acordo com uma biblioteca de parâmetros desses sinais disponível, possibilitando a ativação de contramedidas antes que essas ameaças se tornem letais.

O ambiente eletromagnético tem se tornado cada vez mais complexo, bem como as ameaças controladas por radar mais eficazes. A proliferação, letalidade e avanço dos sistemas de mísseis modernos têm feito deles a maior ameaça para todas as plataformas, seja no mar, na terra ou no ar.

O espectro eletromagnético utilizado por radares na GE é muito amplo, situando-se principalmente na região entre 2 e 18 GHz os sinais associados a armamentos, bem como são muito amplas as possíveis variações dos parâmetros dos sinais, como intervalo de repetição de pulsos, largura de pulso, variação da portadora, entre outros.

Além disso, a grande quantidade de radares existente em determinados ambientes (também chamados ambientes densos) implica em uma alta quantidade de sinais atingindo os sistemas receptores, causando a sobreposição e intercalação desses sinais, fenômeno que é conhecido como *Interleaving*. Em alguns ambientes, estima-se cerca de milhões de pulsos por segundo atingindo um equipamento MAGE (ADAMY, 2001).

Estes fatores têm exigido sistemas MAGE mais rápidos, eficientes, e com respostas automáticas, uma vez que não há tempo suficiente para intervenção humana (o tempo para ativação de contramedidas em aeronaves é da ordem de segundos) (SCHLEHER, 1999).

Diante o exposto, os equipamentos MAGE devem possuir um alto poder de processamento em tempo real para a identificação e reconhecimento dos emissores, principalmente para as etapas de extração de parâmetros, conversão dos parâmetros em palavras digitais, extração dos sinais radar ativos e identificação com a biblioteca de sinais radar conhecidos.

1.2 - ESTADO DA ARTE

Sistemas reais não são apresentados na bibliografia, além de serem classificados como confidenciais, o que impede o estudo detalhado dos sistemas atuais.

Contudo, o acesso a publicações e às bibliografias especializadas (ADAMY, 2000, DRISCOLL *et al*, 1999, GRANGER *et al*, 2001, KINGSLEY, 1992, MAHAFZA, 2000,

MARDIA, 1989, MILOJEVIĆ *et al*, 1992, NERI, 2001, NOEL *et al*, 1998, SCHLEHER, 1999, SKOLNIK, 1980, TSUI, 1992, TSUI, 1995, ZHANG *et al*, 2004) permite apresentar as técnicas sugeridas como atuais.

Sistemas modernos de GE são freqüentemente muito integrados, incluindo múltiplos sensores e múltiplas contramedidas. Cada sinal instantâneo interceptado pelo sistema MAGE deve ser caracterizado por um conjunto de parâmetros. Os parâmetros geralmente medidos para um sinal pulsado são freqüência da portadora (Radiofreqüência – RF), amplitude de pulso (*Pulse Amplitude* – PA), largura de pulso (*Pulse Width* – PW), tempo de chegada (*Time of Arrival* – TOA) e ângulo de chegada (*Angle of Arrival* – AOA) e, em alguns sistemas, a polarização do sinal. Modulação em freqüência no pulso é outro parâmetro que pode ser utilizado para identificar um emissor particular. A Figura 1.1 ilustra esses parâmetros.

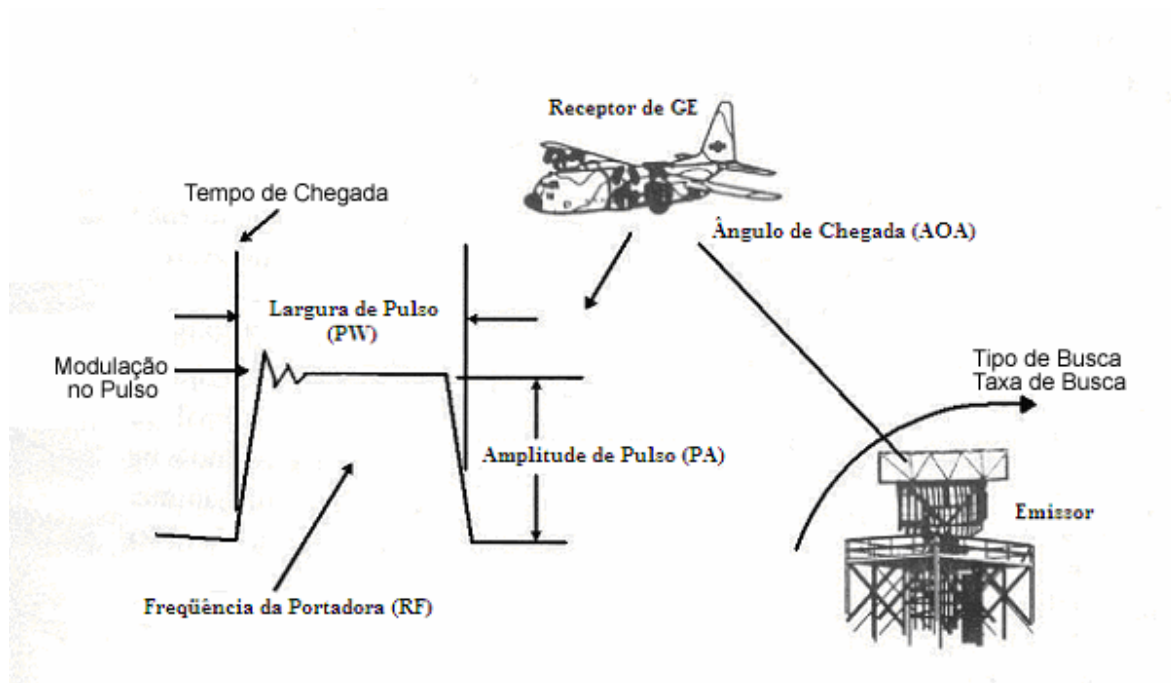


Figura 1.1 – Parâmetros geralmente medidos para um sinal pulsado (modificado – SCHLEHER, 1999).

Os parâmetros medidos em cada pulso interceptado são digitalizados em palavras digitais (*Pulse Descriptor Words* – PDWs), que formam um conjunto de vetores no espaço de parâmetros. Comparando vetores de múltiplos pulsos torna possível isolar os pulsos de um emissor em particular, processo conhecido como *Deinterleaving*.

Os parâmetros mais comuns para a realização do *Deinterleaving* são a frequência da portadora e o ângulo de chegada. Outros parâmetros como largura de pulso e tempo de chegada também são utilizados.

O ângulo de chegada é considerado um parâmetro valioso para o *deinterleaving* visto que um emissor não pode mudar rapidamente sua posição. Contudo, o ângulo de chegada é um dos parâmetros mais difíceis de medir, geralmente requerendo certa quantidade de antenas e receptores, todos com amplitude ou fase casados.

Há diversas publicações que utilizam como base para o *deinterleaving* apenas os tempos de chegada, permitindo a obtenção direta do intervalo de repetição de pulsos (*Pulse Repetition Interval* – PRI), como o histograma dos tempos de chegada (MARDIA, 1989, MILOJEVIĆ *et al*, 1992), periodograma (FOGEL *et al*, 1988, TSUI *et al*, 1995, ORSI *et al*, 1996), estatística circular (ELTON *et al*, 1994), redes neurais (Noone, 1995), transformada de Hough (PERKINS *et al*, 1992) e transformada de *Wavelet* contínua (DISCOLL *et al*, 1999). Essas técnicas são conhecidas também como algoritmos *Interval-only*, pois não consideram outros parâmetros.

Os equipamentos conhecidos, que realizam as tarefas de identificação e classificação dos sinais radar detectados, realizam principalmente operações de histogramação dos parâmetros detectados para posterior filtragem.

1.3 - SOLUÇÃO PROPOSTA

A utilização de processamento de sinais representa uma oportunidade para a extração das informações de forma mais rápida, o que facilitaria tarefas posteriores como identificação, ou até mesmo as histogramações, dado que as informações principais já estariam disponíveis.

Os Processadores Digitais de Sinais (*Digital Signal Processor* – DSP) são otimizados para lidar com o processamento de sinais. A capacidade que os DSPs têm de repetir, em extrema velocidade, uma instrução complexa, faz com que sejam rapidamente resolvidas,

por exemplo, as Transformadas Rápidas de Fourier (*Fast Fourier Transform* – FFT). Isso representa mais um atrativo para a utilização de técnicas de processamento de sinais.

Além disso, uma alternativa para a identificação dos sinais radar seria a utilização de Redes Neurais. As redes neurais são particularmente úteis para classificar dados ligados a valores reais em que se deseja obter uma saída dotada de valor real. As redes neurais irão realizar uma classificação por graus, e serão capazes de expressar valores equivalentes a "não conhecido com certeza". Desta forma, podem-se associar as redes neurais ao processamento de sinais, bem como às histogramações, a fim de que o tempo necessário para a identificação do sinal radar seja minimizado, supondo a existência de um aprendizado prévio nas redes neurais.

Assim, a proposta deste trabalho é apresentar um modelo de sistema que, por meio de um pré-processamento, promove a segmentação de toda a faixa de interesse, para em seguida utilizar a Transformada de *Wavelet* Contínua a fim de isolar os padrões espectrais de interesse, e, por fim, faz uso de redes neurais artificiais para identificação dos padrões espectrais obtidos, extraindo regiões de atividades dos sinais radar identificados.

Embora seja necessária a utilização de valores práticos, bem como a inserção de fenômenos intrínsecos a esses sistemas, como ruídos, perdas, etc., este trabalho se resume a análises por meio de simulações com valores hipotéticos, dentro dos limites práticos.

Contudo, verificou-se que este trabalho é condizente, em geral, com os modelos propostos pela bibliografia consultada, no que tange ao modelamento matemático e valores hipotéticos utilizados.

1.4 - ORDEM DE APRESENTAÇÃO

No primeiro capítulo foi apresentada uma introdução para a contextualização do leitor no assunto abordado e no problema endereçado. Foi descrita uma proposta de solução, que será adotada como objetivo deste trabalho.

No segundo capítulo há uma revisão bibliográfica sobre alguns conceitos de radares, receptores e processamento em guerra eletrônica, a fim de apresentar ao leitor algumas definições necessárias ao entendimento do assunto abordado.

No terceiro capítulo, são apresentadas as ferramentas matemáticas utilizadas e condições de contorno que permitem o desenvolvimento da solução.

O quarto capítulo descreve alguns problemas da técnica de *deinterleaving* por meio da transformada de *wavelet* contínua para detecção de seqüências de pulsos radar, e apresenta uma proposta de solução por meio de subamostragens e filtragem digital FIR, incluindo a identificação de sinais radar utilizando redes neurais artificiais.

Por fim, o quinto capítulo apresenta as conclusões e recomendações do autor.

2 - RADARES, RECEPTORES E PROCESSAMENTO EM GUERRA ELETRÔNICA

Neste capítulo são apresentados de forma sucinta alguns conceitos necessários à familiarização do leitor no contexto deste trabalho.

2.1 - RADARES

RADAR é uma abreviação para *Radio Detection and Ranging*. Sistemas RADAR utilizam formas de ondas particulares e antenas direcionais para transmitir energia em um volume específico do espaço a fim de procurar objetos, os quais, se estiverem dentro daquele volume, refletirão porções desta energia (ecos) de volta para o RADAR. Esses ecos são processados pelo receptor do RADAR obtendo-se informações como distância, velocidade, posição angular, e outras características.

Os radares são freqüentemente classificados pelas formas de onda que utilizam, ou pela freqüência de operação. Pelas formas de onda, podem ser de Onda Contínua (*Continuous Wave – CW*) ou Radares Pulsados (*Pulsed Radars – PR*). Pela freqüência de operação, a Tabela 2.1 e a Tabela 2.2 apresentam as possíveis classificações:

Tabela 2.1 - Designações Adotadas para Bandas de Freqüência Inferiores a 40GHz.

Designação atual de freqüência	C	D	E	F	G	H	I	J	K
Faixa de freqüência (GHz)	0.5 – 1.0	1.0 – 2.0	2.0 – 3.0	3.0 – 4.0	4.0 – 6.0	6.0 – 8.0	8.0 – 10.0	10.0 – 20.0	20.0 – 40.0

Tabela 2.2 - Designações Adotadas para Bandas de Freqüência Superiores a 40GHz.

Designação atual de freqüência	Q	U	V	E	W	F	D	G
Faixa de freqüência (GHz)	33 – 50	40 – 60	50 – 75	60 – 90	75 – 110	90 – 140	110 – 170	140 – 220

Os radares CW emitem energia eletromagnética continuamente, utilizando antenas transmissoras e receptoras separadamente. Os radares CW mais simples emitem um sinal senoidal de amplitude, frequência e fase constantes. Esses radares mais simples podem medir precisamente a velocidade relativa, por meio do efeito Doppler, e a posição angular do alvo, sem, contudo, extrair a informação de distância (NOEL, 1998).

A Figura 2.1 ilustra o processo de deslocamento em frequência provocado pelo efeito Doppler.

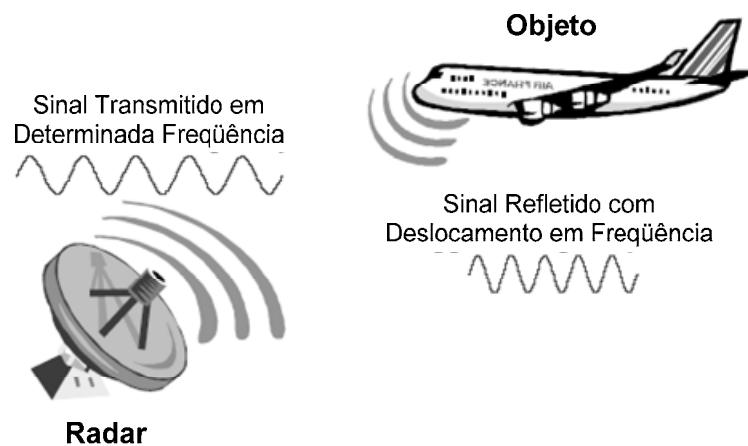


Figura 2.1 – Deslocamento na frequência provocado pelo efeito Doppler entre os sinais transmitido e refletido (modificado – NOEL, 1998).

A incapacidade de um radar CW simples medir distância ocorre devido ao estreito espectro (largura de banda) de sua forma de onda transmitida.

Caso a informação de distância necessite ser obtida, alguma marcação no tempo deve ser aplicada na portadora CW. Uma forma simples de marcação é o processo de modulação em frequência, que é a variação controlada da frequência do sinal transmitido. Medindo-se a frequência variante do sinal refletido e o intervalo de tempo entre a transmissão e recepção, é possível obter a distância.

A Figura 2.2 ilustra o processo de modulação em frequência, bem como as marcações no tempo do sinal emitido e refletido, o que possibilita o cálculo da distância entre o objeto e o radar.

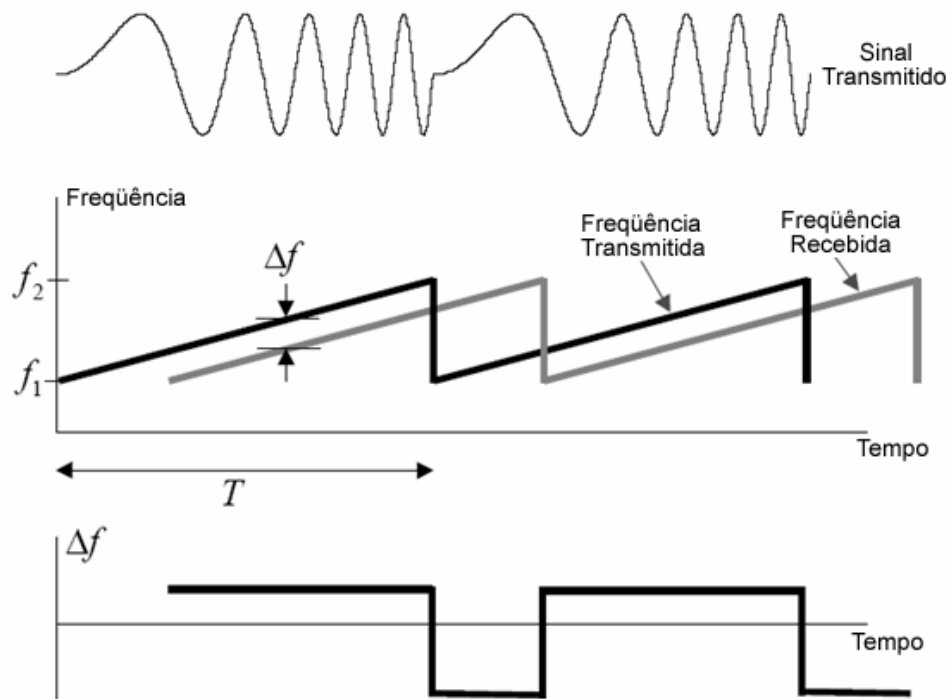


Figura 2.2 – Modulação em frequência para sinal CW com evidente marcação no tempo para os sinais emitido e refletido (modificado – NOEL, 1998).

Radars Pulsados utilizam trens de ondas pulsadas, podendo ser classificados com base em sua Frequência de Repetição de Pulso (*Pulse Repetition Frequency – PRF*) como baixa PRF, média PRF e alta PRF. Radars de baixa PRF são utilizados primariamente para obtenção de distância. Os Radars de alta PRF são utilizados principalmente para mensuração de velocidade do alvo. A classificação também pode ser realizada com o análogo do PRF, que é o Intervalo de Repetição de Pulso (*Pulse Repetition Interval – PRI*) (MAHAFZA, 2000). A Figura 2.3 ilustra um sinal radar pulsado, bem como seus parâmetros característicos.

A maioria dos radares é do tipo pulsado convencional. Aquelos relacionados a armamentos possuem a frequência da portadora variando entre 2 a 18GHz, havendo alguma atividade em torno de 35 GHz, 94 GHz, 140 GHz, e 220 GHz (SCHELEHER, 1999). Os valores típicos de PRF variam entre 100 Hz e 500 kHz, os valores de PW entre 0.1 a 250 μ s, e os valores de SR entre poucos a milhares de RPM (SCHELEHER, 1999, TSUI, 1995). A combinação desses parâmetros define a operação de determinado radar.

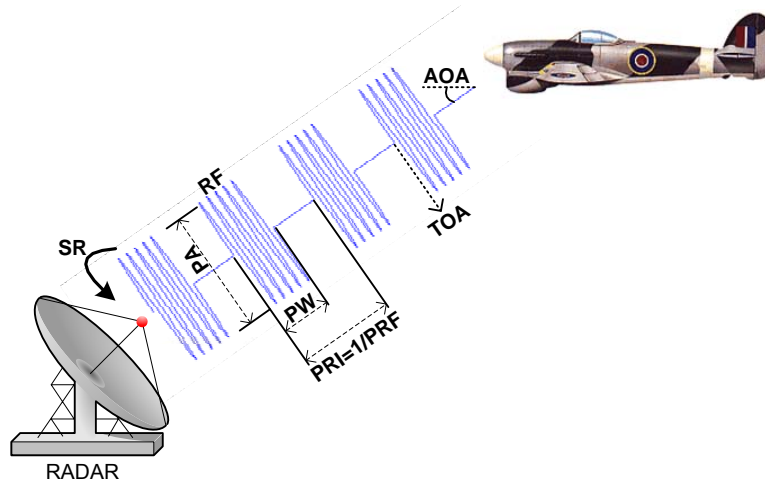


Figura 2.3 – Parâmetros característicos de radares pulsados. Parâmetros Intrapulso: PA – Amplitude de Pulso (*Pulse Amplitude*), PW – Largura de Pulso (*Pulse Width*), RF – Radiofrequência, TOA – Tempo de Chegada (*Time of Arrival*) e AOA – Angulo de Chegada (*Angle of Arrival*). Parâmetros Interpulso: PRI – Intervalo de Repetição de Pulsos (*Pulse Repetition Interval*), PRF – Frequência de Repetição de Pulsos (*Pulse Repetition Frequency*), e SR – Taxa de Varredura (*Scan Rate*).

Além de auxiliar na identificação dos pulsos, alguns radares empregam técnicas mais sofisticadas de transmissão para dificultar sua detecção. A variação da PRF é uma dessas técnicas. Aqui será utilizado como parâmetro de análise o PRI, que é o inverso da PRF.

As formas clássicas de variação do PRI são:

- a) *Jitter* – O PRI varia aleatoriamente até um limite percentual em torno de um valor de PRI central. Como exemplo, pode-se ter um PRI de valor central $5\mu\text{s}$, e variações aleatórias de até 10% ($0.5\mu\text{s}$) em torno desse valor entre os pulsos;
- b) *Stagger* – O PRI varia segundo um padrão seqüencial que se repete em ciclos. Um exemplo seria uma seqüência de quatro valores de PRI, $2\mu\text{s}$ $2.2\mu\text{s}$ $2.4\mu\text{s}$ e $2.6\mu\text{s}$, onde entre pulsos consecutivos, a cada momento é utilizado um desses valores, sempre se repetindo a seqüência. Nesse caso, diz-se que o PRI do frame é de $9,2\mu\text{s}$, a soma dos valores individuais, e o PRI médio é $2,3\mu\text{s}$; e
- c) *Wobulated* – Aqui as variações do PRI obedecem a um padrão senoidal, sempre em torno de um valor central com um limite percentual máximo de variação. Assim, se tivéssemos um PRI central de $5\mu\text{s}$ e uma variação máxima de 10% do valor central ($0,5\mu\text{s}$), os valores de PRI entre pulsos consecutivos oscilariam entre $4,5\mu\text{s}$ e $5,5\mu\text{s}$, conforme um padrão senoidal.

A Figura 2.4 ilustra os padrões clássicos de PRI conhecidos, bem como a primeira diferença entre os tempos de chegada (TOA) dos pulsos.

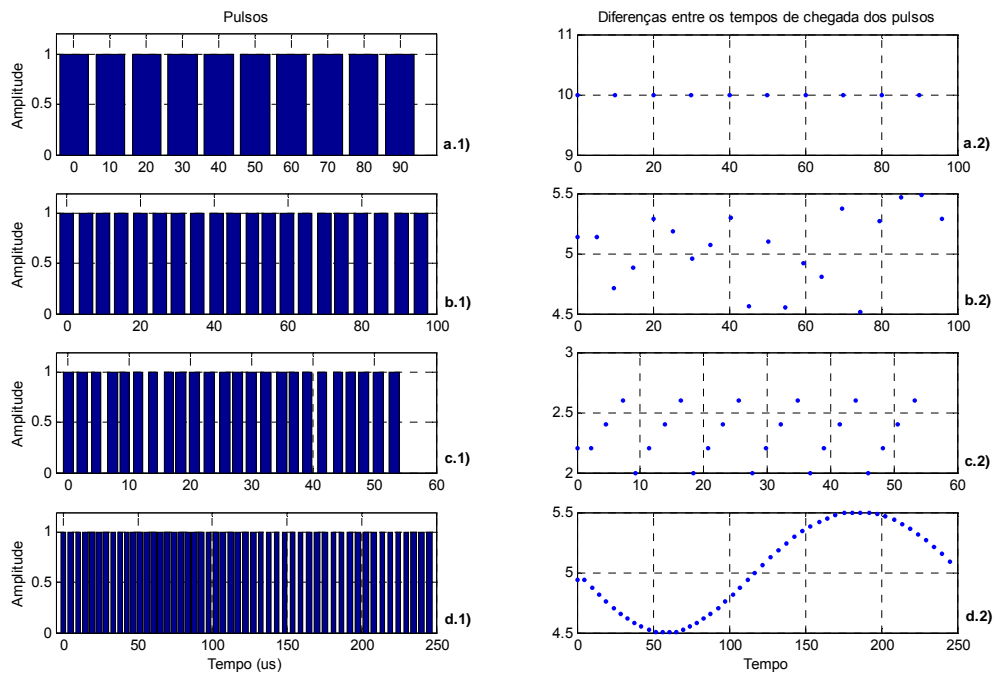


Figura 2.4 – Padrões clássicos de PRI e diferença dos tempos de chegada (TOA) entre pulsos consecutivos para: a.1 e a.2 – PRI tipo Simple de valor 10, sem variação; b.1 e b.2 – PRI *Jitter* de valor central 5 com variação de até 10% em torno do valor central; c.1 e c.2 – PRI *Stagger* de valores 2.0, 2.2, 2.4 e 2.6; d.1 e d.2 – PRI *Wobulated*, com valor central 5, variação de até 10% e 50 pulsos por ciclo.

Outra técnica utilizada por radares mais modernos é o salto em frequência, onde ocorre a seleção randômica de frequências, o que é também conhecido como agilidade em frequência. Existe ainda a seleção randômica de um número fixo de frequências, também conhecida como diversidade em frequência.

2.1.1 - Medição de Distância e Resolução

A taxa na qual os pulsos são transmitidos é chamada Frequência de Repetição de Pulsos (PRF). A taxa máxima na qual os pulsos Radar podem ser transmitidos ocorre quando um pulso é transmitido antes que o pulso anterior tenha completado o trajeto de ida e volta ao alvo. Nesta situação não fica claro qual pulso originou qual eco, o que gera ambigüidade no reconhecimento do alvo (KINGSLEY, 1992).

A equação que define a distância em relação ao alvo, considerando que um pulso viaja na velocidade da luz é:

$$R = ct_d / 2. \quad (2.1)$$

R é a distância, c é a velocidade da luz e t_d é o tempo de atraso de propagação do sinal (ida e volta) (KINGSLEY, 1992).

Pode-se simplificar a interpretação utilizando a expressão “150m por microssegundo”, que indica que, depois da transmissão, cada microssegundo extra implica que o alvo está a mais 150m distante.

Dessa forma, se o intervalo de tempo entre dois pulsos (PRI) for $1\mu\text{s}$ (PRF de 1MHz), então distâncias de até 150m poderão ser cobertas pelo radar. Distâncias acima poderão ocasionar uma situação na qual não é evidente qual pulso originou o eco (KINGSLEY, 1992).

A distância máxima R_{max} resultante para que não haja ambigüidade é dada por:

$$R_{\text{max}} = c \cdot PRI / 2. \quad (2.2)$$

Esta é a distância a qual um pulso pode ser emitido e seu eco recebido sem que o próximo pulso seja emitido.

Há diversas formas de reduzir o PRI (aumentar a PRF) sem a necessidade de reduzir-se a distância máxima sem ambigüidade. Uma delas é utilizar rajadas de pulsos em PRIs diferentes, o que é conhecido também como PRIs *staggered*.

A resolução de um radar indica o quão distante dois objetos devem estar para que sejam detectados realmente dois objetos ao invés de um único e grande. A resolução leva em consideração a duração do pulso, τ , e o intervalo entre os pulsos, PRI. Se o intervalo de tempo entre os ecos de dois objetos é maior que a duração do pulso τ , então dois ecos separados são visualizados. Se os ecos estão mais próximos que τ , os ecos irão se unir, parecendo um único objeto.

A expressão que nos dá a resolução ΔR é dada por:

$$\Delta R = c\tau/2. \quad (2.3)$$

Os valores de precisão e resolução são melhorados, na prática, com boas taxas sinal-ruído, interpolação dos valores de distância obtidos em várias medições e com a aproximação do caminho estimado do objeto, que é o acompanhamento.

2.1.2 - Tipos de Radares em Guerra Eletrônica

A utilização dos radares em GE está intimamente ligada à obtenção de informações de direção ou posição, e velocidade de objetos. De acordo com a aplicação dessas informações, há dois tipos principais de radares: radares de vigilância ou busca, e radares de acompanhamento (KINGSLEY, 1992).

Os radares de vigilância ou busca visam a esclarecer um determinado volume do espaço, determinando a existência ou não de alvos, e, em caso positivo, obtendo informações como posição, proa, velocidade e altitude. São incluídos nessa categoria os radares de controle de tráfego aéreo.

A maioria dos radares mais modernos deste tipo é denominada multifunção devido à capacidade de executar outras atividades simultaneamente ao processo de procurar por novos alvos. Daí surge um subtipo, que são os radares de aquisição, que têm como objetivo prover informação prévia de posição para um radar de acompanhamento.

Os radares de acompanhamento medem continuamente as coordenadas de um alvo móvel a fim de determinar seu caminho e prever sua trajetória. O processo de acompanhamento pode ser realizado utilizando a distância, ângulo ou informação Doppler. Contudo, o acompanhamento em ângulo é o atributo característico dos radares de acompanhamento.

De certa forma, os radares de vigilância podem ser considerados como sistemas de acompanhamento, visto que estimam a posição do alvo toda vez que o processo de busca

retoma para determinadas posições – processo conhecido como *track-while-scan*. Embora eles possam manter o acompanhamento de vários alvos simultaneamente, a precisão dos dados obtidos pode não ser adequada para determinados propósitos.

Por outro lado, os radares de acompanhamento são dedicados a um alvo em particular, e o observam continuamente com alta precisão, razão pela qual estão frequentemente associados a armamentos.

Alguns radares de acompanhamento têm dispositivos próprios de busca, como, por exemplo, um radar de aquisição. Contudo, a forma de operação mais comum é um radar de busca ou vigilância alertar sobre quaisquer alvos representando uma ameaça, transferindo as coordenadas do alvo para o sistema de acompanhamento, que então procura na região informada para obter o alvo antes de iniciar o procedimento de acompanhamento.

2.2 - RECEPTORES E PROCESSAMENTO EM GE

2.2.1 - Receptores de GE

Conforme já mencionado, as MAGE têm como objetivo coletar e processar informações do ambiente eletromagnético. As principais atividades realizadas são interceptação, identificação, análise e localização das radiações inimigas.

Para cumprir essas atividades, no que tange a armamentos guiados a RF, os seguintes passos são necessários (SCHLEHER, 1999):

- a) Interceptação das radiações associadas com o armamento;
- b) Separação dos sinais de interesse;
- c) Medição dos parâmetros dos sinais selecionados;
- d) Comparação desses parâmetros com um conjunto de parâmetros armazenados para identificação dos tipos de sensores associados com os sinais interceptados;
- e) Identificação do armamento associado com o sensor identificado; e
- f) Obtenção da direção angular, ou, quando possível, da localização do sinal.

Nesse sentido, os receptores de GE são sistemas especialmente criados para tratar com a diversidade do ambiente eletromagnético. Eles diferem dos receptores de comunicação no sentido de que no projeto destes, a frequência, os tipos de modulação, a largura de banda e o sinal de chegada são conhecidos, ao passo que para aqueles, não somente a informação do sinal de entrada é desconhecida, mas o sinal transmitido pode ser especialmente gerado com a finalidade de evitar a sua interceptação (TSUI, 1995).

Outra grande diferença entre um receptor de GE e outros tipos de receptores reside na forma de sua saída, que consiste de palavras digitais (*Pulse Descriptor Words* – PDW) que descrevem as características de cada pulso radar interceptado. Dependendo do projeto do receptor, cada tipo produz um formato único de PDW.

Geralmente, para um sinal radar pulsado, a PDW inclui os seguintes parâmetros: frequência portadora (RF); amplitude de pulso (PA); largura de pulso (PW); tempo de chegada (TOA); e ângulo de chegada (AOA). A resolução dos parâmetros da PDW é determinada pelo número de bits alocados. A Tabela 2.3 ilustra um exemplo de PDW.

Tabela 2.3 – Formato típico de PDW (TSUI, 1995).

Parâmetros	Faixa	Nº de Bits
Frequência (RF)	Até 32 GHz	15 (Resolução de 1 MHz)
Amplitude de Pulso (PA)	Até 128 dB	7 (Resolução de 1 dB)
Largura de Pulso (PW)	Até 204 µs	12 (Resolução de 0.05 µs)
Tempo de Chegada (TOA)	Até 50 s	30 (Resolução de 0.05 µs)
Ângulo de Chegada (AOA)	360 graus	9 (Resolução de 1 grau)
Flag de Sinal BPSK		1
Flag de Sinal Chirp		1
Nº Total de Bits		75

Os receptores de GE podem ser classificados quanto à frequência de operação, à aplicação e suas estruturas.

De acordo com sua frequência de operação, subdividem-se em: muito alta frequência (*Very High Frequency* – VHF); ultra alta frequência (*Ultra High Frequency* – UHF); microondas; e extremamente alta frequência (*Extremely High Frequency* – EHF). Ainda podem ser subdivididos em dois grupos: receptores de interceptação de comunicação e receptores de interceptação de radar.

Quanto à aplicação, os receptores de GE podem ser Receptores de Alerta Radar (*Radar Warning Receivers – RWR*), Receptores de Alerta e Guiagem de Radar (*Radar Homing and Warning Receivers – RHWR*), Receptores de Ataque Eletrônico, Receptores de Apoio a GE e Receptores de Inteligência Eletrônica.

Quanto à estrutura, tradicionalmente os receptores de GE são classificados como (ADAMY, 2001):

- a) Cristal Vídeio;
- b) Super-heteródino;
- c) Medida de Frequência Instantânea (*Instantaneous Frequency Measurement – IFM*);
- d) Frequência de Rádio Sintonizada (*Tunned Radio Frequency – TRF*);
- e) Sintonia Fixa;
- f) Canalizado;
- g) Compressivo;
- h) Bragg Cell;
- i) Digital.

2.2.2 - Processamento em GE

Freqüentemente os receptores de GE estão associados com processadores de GE. Na medida em que o ambiente eletromagnético se tornou mais complexo, as armas controladas por radar mais letais, e o tempo de resposta mais curto, foi necessário detectar e identificar ameaças automaticamente. Identificação de ameaças é uma tarefa primária do processamento em GE em quase todos os sistemas (ADAMY, 2001). Localização de emissores também é uma tarefa básica das operações de processamento em GE.

Identificação, em sistemas modernos, é uma tarefa complexa. Em geral, é necessário saber que tipo de ameaça está presente, sua localização e seu modo de operação. Algumas generalizações podem ser feitas com relação à identificação de ameaças:

- a) Tarefas mais simples são realizadas primeiro;
- b) Dados obtidos de análises prévias e simples são removidos, permitindo análises mais complexas em quantidades de dados menores; e

- c) As análises são finalizadas assim que todas as ambigüidades necessárias foram resolvidas.

O processamento em GE é responsável por realizar as seguintes tarefas (TSUI, 1995):

- a) *Deinterleaving* – é a separação dos trens de pulsos interceptados em trens de pulsos individuais por radar. Para isso, compara-se cada pulso interceptado a fim de verificar se são originários do mesmo radar. Dada a importância deste tópico em relação ao tema do trabalho, foi reservada a próxima seção para o seu detalhamento;
- b) Cálculo do PRI – uma vez identificados os trens de pulsos individuais por radar, a informação de Tempo de Chegada (TOA) pode ser utilizada para a obtenção do PRI, que é a diferença dos TOAs entre os pulsos de um trem de pulsos já separado. Se a informação de diferenças entre TOAs foi utilizada para o *deinterleaving*, a informação de PRI já está disponível;
- c) Identificação do radar – caracterizam-se RF, PRI e PW como parâmetros intrínsecos do radar, pois são gerados pelo radar, e podem ser utilizados para determinação do tipo do radar. Por outro lado, PA e AOA não são geradas pelo radar, mas são funções da posição relativa do radar e do receptor;
- d) Rastreamento (*Tracking*) – o processador de GE pode processar uma quantidade limitada de pulsos, a qual é geralmente menor que a quantidade de pulsos que um receptor de GE pode interceptar. Contudo, uma vez que o trem de pulsos radar é identificado, não é mais necessário realizar o *deinterleaving* desses pulsos. Rastreadores são projetados para prevenir que pulsos relacionados a radares já identificados alcancem a porção onde é realizado o *deinterleaving* no processador;
- e) *Revisiting* – uma vez que um trem de pulsos é identificado como ameaça, alguma contramedida pode ser acionada para interferir no sinal, como um *jammer*. Ao mesmo tempo, como o trem de pulsos é rastreado, a informação não chegará à porção responsável pelo *deinterleaving*. Como resultado, não se sabe se o sinal ainda está sendo transmitido, o que pode implicar em aplicar um *jamming* contra um sinal que não existe. Assim, para saber se o sinal ainda está sendo interceptado, o rastreador para temporariamente de rastrear, liberando a informação para o processador. Este processo é chamado *revisiting*. Se o sinal ainda está sendo interceptado, os parâmetros medidos nos pulsos estarão presentes no processador, e

o rastreador poderá continuar a rastrear o sinal. Caso o sinal não esteja mais sendo interceptado, o *jammer* e as atividades de rastreamento podem parar.

2.3 - DEINTERLEAVING

Deinterleaving de pulsos é o processo de isolar os pulsos interceptados de dois ou mais emissores em conjuntos de pulsos individuais por emissor.

Em ambientes considerados densos estima-se cerca de milhões de pulsos por segundo atingindo os sistemas de recepção, o que faz da tarefa de *deinterleaving* uma das mais custosas no processamento em GE. Além disso, outro fator complicador é a superposição de pulsos, que ocorre quando dois pulsos atingem os sistemas de recepção ao mesmo tempo.

A Figura 2.5 ilustra os sinais de três radares pulsados ao longo do tempo, bem como o sinal recebido pelo receptor, que é resultante da superposição dos sinais. Para melhor visualização os sinais foram criados com valores altos de *duty cycle* (largura de pulso dividida pelo intervalo de repetição de pulso). Normalmente, para sinais pulsados, esse valor é 0,1%.

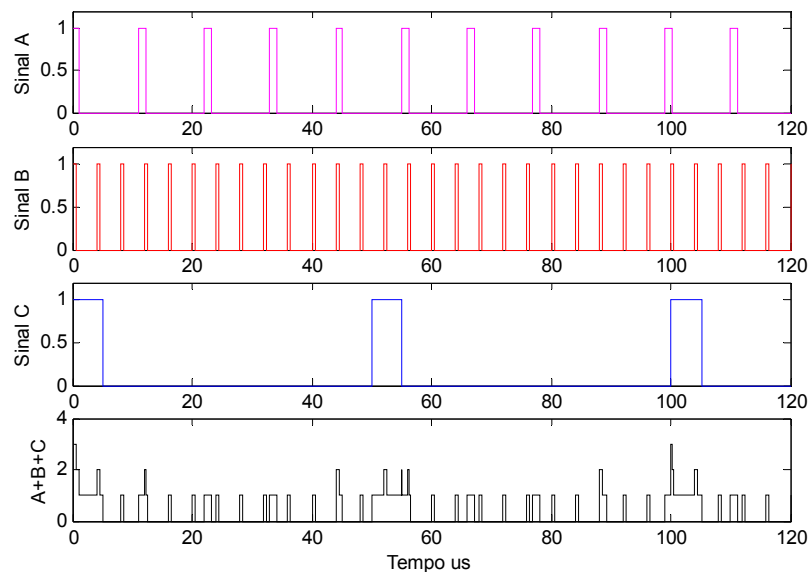


Figura 2.5 – Sinais pulsados de três emissores (Sinal A, Sinal B e Sinal C) ao longo do tempo e o sinal interceptado pelo receptor (A+B+C).

Observa-se a ocorrência da superposição de pulsos, conhecida como *Pulse on Pulse* (POP) *Problem*. Caso o sistema observe apenas um pulso, um dos pulsos será eliminado. Dependendo do número de pulsos eliminados e do processamento de identificação do sinal em seguida, pode haver um efeito negativo na performance do sistema.

O processo de *deinterleaving* envolve todas as informações conhecidas sobre os pulsos recebidos, o que depende da configuração do sistema de recepção. A Tabela 2.4 ilustra as informações disponíveis para os sistemas de acordo com os tipos de receptores utilizados.

Tabela 2.4 – Informações disponíveis para cada pulso de acordo com o tipo de receptor (ADAMY, 2001).

Tipo de receptor ou subsistema	Informações obtidas em cada pulso
Receptor cristal vídeo	Largura de pulso, potência do sinal, tempo de chegada e amplitude por tempo
Sistema monopulso <i>Direction Find</i> (DF)	Direção de chegada
Receptor IFM	Frequência RF
Receptor com discriminador AM e FM	Largura de pulso, potência do sinal, tempo de chegada, frequência RF, amplitude e frequência por tempo
Receptor digital	Largura de pulso, potência do sinal, tempo de chegada, frequência RF, e modulação FM ou digital no pulso
Receptor Canalizado	Largura de pulso, potência do sinal, tempo de chegada, frequência (somente por canal)

Os receptores IFM representaram uma ferramenta poderosa para o *deinterleaving* dos pulsos com base nas frequências associadas a sinais individuais. Contudo, se há agilidade em frequência (alteração da frequência da portadora) pulso a pulso, essa técnica não é eficiente.

Um sistema de alta precisão para obtenção de direção (*Direction-finding* – DF) que disponibilize informação de AOA estável pulso a pulso possibilita a realização do *deinterleaving*. Em muitas circunstâncias, esse método de *deinterleaving* é preferível, pois não importa a complexidade da modulação do sinal. Uma vez isolados os pulsos de um único sinal, análises estatísticas podem ser realizadas para obtenção das informações.

Diversas técnicas de *deinterleaving* de pulsos foram propostas por meio de publicações. Serão mostradas apenas algumas que utilizam somente os tempos de chegada como parâmetro disponível.

2.3.1 - Técnicas de Deinterleaving

Técnicas de *deinterleaving* baseadas apenas no parâmetro de tempo de chegada, não considerando filtragens prévias dos pulsos com relação a seus outros parâmetros, são conhecidas como algoritmos *interval-only*. Essas técnicas também são conhecidas por possibilitarem a reconstrução do PRI dos sinais, visto que extraem essa informação imediatamente.

Sete técnicas principais desse conjunto de algoritmos foram levantadas e serão aqui apresentadas superficialmente:

- a) Histograma dos tempos de chegada – Técnica mais comum que faz uso das diferenças entre os TOA dos pulsos, realizando a histogramação dessas informações (MARDIA, 1989, MILOJEVIĆ *et al*, 1992);
- b) Periodograma – Faz uso da transformada de Fourier do sinal composto dos TOA de cada pulso e atribui pesos à saída, produzindo picos nas PRF apresentadas graficamente (FOGEL *et al*, 1988, TSUI *et al*, 1995, ORSI *et al*, 1996);
- c) *Folding histogram* – Algoritmo que empacota os dados de TOA em um intervalo ajustável que resulta na detecção do PRI em vários intervalos de empacotamento (STAELIN, 1969);
- d) Estatística circular – Técnica que utiliza o algoritmo *Folding Histogram* combinado com o uso de estatística circular (ELTON *et al*, 1994);
- e) Redes Neurais – Utiliza uma técnica de busca modificada, combinada com uma abordagem de redes neurais recursivas (Noone, 1995);
- f) Transformada de Hough – Técnica que faz o mapeamento dos dados unidimensionais de TOA para dados bidimensionais lineares, aplicando em seguida a transformada de Hough (PERKINS *et al*, 1992); e

- g) Transformada de *Wavelet* Contínua – Técnica que aplica a transformada de *wavelet* contínua, com uma *wavelet*-mãe apropriada e um fator de resolução específico, determinando os valores de PRI das seqüências de pulsos radar por meio de limiarização (DRISCOLL *et al*, 1999).

Este trabalho faz uso da técnica da transformada de *wavelet* contínua para a detecção de seqüências de pulsos radar. Dessa forma, o próximo capítulo apresenta as ferramentas matemáticas necessárias para o entendimento da técnica e das modificações propostas.

3 - FERRAMENTAS MATEMÁTICAS UTILIZADAS NA SOLUÇÃO DO PROBLEMA

Neste capítulo são apresentados os fundamentos e conceitos matemáticos necessários para a formulação da solução proposta.

3.1 - TRANSFORMADA DE WAVELET CONTÍNUA

Processamento de sinais está relacionado com a representação, transformação e manipulação de sinais e da informação que eles contêm.

O processamento de sinais digitais surgiu em meados de 1960 (OPPENHEIM; SCHAFER, 1989) como um novo ramo do processamento de sinais devido ao potencial e viabilidade de implementação de processamento de sinais em tempo real utilizando computadores digitais. Um aspecto fundamental do processamento de sinais digitais reside no processamento de seqüências de amostras.

Os sinais utilizados no processamento de sinais digitais podem ser estacionários quando suas propriedades são estatisticamente invariantes no tempo. Estes sinais são muito bem estudados por meio da Transformada de Fourier – TF, pois são decompostos em combinações lineares de senos e co-senos por meio das funções $e^{-i2\pi ft}$. Para um sinal $f(t)$, a TF é dada por:

$$F(f) = \int_{-\infty}^{\infty} f(t)e^{-j2\pi ft} dt. \quad (3.1)$$

Os sinais não estacionários apresentam transientes que não podem ser previstos. A TF não é adequada para a análise desses sinais, bem como não é eficiente para a análise de sinais de duração finita, pois sua base tem duração infinita. Em 1946, D. Gabor criou a transformada conhecida como *Short Time Fourier Transform* – STFT, que faz uso de uma “função janela” $g(t-\tau)$, sendo definida por:

$$STFT(\tau, f) = \int_{-\infty}^{\infty} f(t)g^*(t-\tau)e^{-j2\pi ft} dt, \quad (3.2)$$

onde $g^*(t)$ é o complexo conjugado de $g(t)$.

Contudo, a função janela $g(t)$ tem comprimento fixo, o que torna a STFT inadequada para analisar sinais que apresentem ao mesmo tempo componentes de alta e baixa frequência. A Figura 3.1 ilustra a resolução tempo-frequência obtido pela STFT.

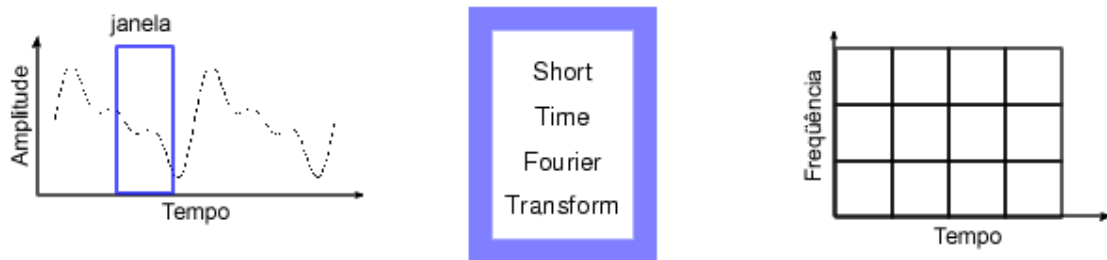


Figura 3.1 – Resolução no plano tempo-frequência da *Short Time Fourier Transform* (modificado – MATLAB).

A Transformada de *Wavelet* – TW oferece uma solução para esta restrição. Assim como a transformada de Fourier (TF), a Transformada de *Wavelet* decompõe um sinal em convenientes conjuntos de bases de funções. *Wavelets* literalmente significa pequenas ondas (*small waves*).

Como seu nome sugere, elas são ondas, oscilam e suas curvas têm um decaimento a zero em área algébrica. A pequenez refere-se ao fato de que elas são localizadas em tempo. Isso contrasta com o comportamento das bases de Fourier, pois essas bases são construídas por senos e co-senos infinitos. Estas funções senoidais e co-senoidais estão perfeitamente localizadas no espaço de frequência, mas não decrescem a zero como uma função do tempo. Por outro lado, as *Wavelets* caem a zero quando $t \rightarrow \pm\infty$ e desfrutam de boa propriedade de localização no tempo.

A primeira menção registrada ao termo “*wavelet*” ocorreu em 1909, na tese do matemático alemão Alfred Haar (ZANANDREA, 2002). O conceito de ondeletas na forma teórica presente foi inicialmente proposto por Jean Morlet e Alex Grossmann, na França.

Os métodos de análise de *wavelet* foram desenvolvidos principalmente por Y. Meyer e seus colegas, que asseguraram a disseminação destes métodos (ZANANDREA, 2002). O algoritmo principal data do trabalho de Stephanie Mallat em 1988 (ZANANDREA, 2002).

A construção de uma *wavelet* é realizada a partir de uma função básica, de suporte compacto e oscilatória, denominada *wavelet*-mãe, $\psi(t)$, que gera as bases a partir de deslocamentos e dilatações $\psi(\frac{t-b}{a})$, onde a é a escala, responsável pela dilatação, e b é o termo responsável pelo deslocamento.

Assim, a duração da *Wavelet* $\psi(t)$ pode ser ajustada em função do conteúdo de frequências do sinal a ser analisado, ou da ênfase que se deseja colocar na análise das baixas ou altas frequências do sinal.

Existem três possibilidades para a TW, quais sejam, a transformada de *wavelet* contínua (do inglês, *Continuous Wavelet Transform* – CWT), a expansão em série de *wavelet* e a transformada de *wavelet* discreta (do inglês, *Discrete Wavelet Transform* – DWT), sendo efetivamente utilizadas no processamento de sinais a CWT e a DWT.

A DWT consiste dos parâmetros de escala a e deslocamento b restritos a um conjunto discreto, usualmente dado por $a = 2^j$ e $b = k2^j$, com j e k inteiros. Já a CWT permite a escolha de quaisquer valores de escala e deslocamento, estando associada a funções de variável contínua.

A estrutura da CWT e DWT permite um particionamento não uniforme do plano tempo-frequência, permitindo a sua utilização no processamento de sinais não estacionários, como, por exemplo, as imagens. Outra consequência importante da dilatação é a possibilidade de uma representação hierárquica do conjunto de dados. Essa abordagem é também chamada de análise de multiresolução. A Figura 3.2 ilustra a resolução tempo-frequência obtida pela TW.



Figura 3.2 – Resolução no plano tempo-frequência obtida pela Transformada de *Wavelet* (modificado – MATLAB).

Tendo em vista que este trabalho faz uso da CWT, somente o detalhamento matemático inerente a esta transformada será apresentado.

Seja $\psi(t)$ uma função de valor real, chamada *wavelet*-mãe, cuja transformada de Fourier satisfaz a condição de admissibilidade

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty, \quad (3.3)$$

onde C_ψ é uma constante e $\Psi(\omega)$ é a TF da *wavelet*-mãe $\psi(t)$. Essa condição de admissibilidade equivale a exigir que a função $\Psi(\omega)$ se anule na origem. Assim, dado que:

$$\Psi(\omega) = \int_{-\infty}^{\infty} \psi(t) e^{-i\omega t} dt, \quad (3.4)$$

temos:

$$\Psi(0) = 0 = \int_{-\infty}^{\infty} \psi(t) dt, \quad (3.5)$$

significando que $\psi(t)$ deve ser oscilante e de média nula.

A *wavelet*-mãe $\psi(t)$ deve ser uma função janela, ou seja:

$$\int_{-\infty}^{\infty} |\psi(t)| dt < \infty, \quad (3.6)$$

de tal forma que $\psi(t)$ é contínua em \mathfrak{R} .

Considerando-se então $\psi(t)$ uma função oscilatória e de curta duração, a Transformada de *Wavelet* Contínua é definida pela seguinte expressão:

$$CWT(a, b) = a^{-1/2} \int_{-\infty}^{\infty} f(t) \psi\left(\frac{t-b}{a}\right) dt. \quad (3.7)$$

Conforme já mencionado, b é o parâmetro responsável pelo deslocamento de $\psi(t)$ ao longo do tempo, e o parâmetro de escala a é responsável pela compressão ou dilatação.

A TW não envolve implicitamente o conceito de frequência, mas sim o resultado da análise do sinal para uma gama de valores de a , ou seja, dos valores da duração da janela $\psi(t)$. A Figura 3.3 ilustra o processo de análise do sinal pela CWT em função de um valor de $\psi(t)$ escalonada de a e deslocada de b .

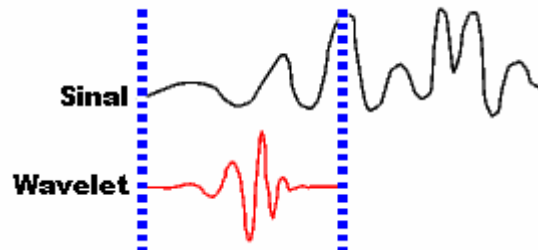


Figura 3.3 – Processo de análise *wavelet* de um sinal para uma função $\psi(t)$ deslocada e escalonada (modificado – MATLAB).

Os valores obtidos correspondem, de certa forma, à similaridade da *wavelet*-mãe com o trecho do sinal analisado, para aquele valor de escala e deslocamento. Assim, decorre outro aspecto importante deste processo de análise: a escolha da *wavelet*-mãe.

A escolha da *wavelet*-mãe está intimamente ligada com o formato do sinal que desejamos analisar. Assim, em muitos casos torna-se interessante utilizar uma *wavelet*-mãe que se assemelhe ao formato de sinal que será estudado.

3.1.1 - Técnica de *Deinterleaving* por meio da Transformada de *Wavelet* Contínua

A técnica aqui apresentada foi proposta por DRISCOL e HOWARD (1999), e baseia-se na CWT para a detecção de seqüências de pulsos radar intercaladas (*interleaved*). O parâmetro de escala, T , está associado à característica periódica dos sinais (Intervalo de Repetição de Pulsos), e o parâmetro de deslocamento, t , à posição no tempo. Assim, a função detectora possui dois argumentos, T e t , como na CWT.

Os sinais radar são representados como um conjunto de impulsos superpostos em função de seus tempos de chegada (TOA),

$$s(t) = \sum_j \delta(t - t_j). \quad (3.8)$$

Cabe observar que a informação de largura de pulso não é utilizada nessa técnica.

A *wavelet*-mãe utilizada é:

$$\psi(t) = M^{1/2} \chi\left(\frac{t}{M}\right) e^{2\pi i t}, \quad (3.9)$$

onde $\chi(t)$ é uma janela retangular de comprimento unitário, e M é um parâmetro ajustável relativo à resolução da função detectora. Este parâmetro determina o número de ciclos da exponencial complexa contida na janela $\chi(t)$, e representa o menor número de pulsos que se espera receber ou se deseja detectar. A resolução espectral obtida é $(1/M)T$.

Um valor prático para o parâmetro M é 12. Radares de busca recebem por volta de dez a doze pulsos refletidos. A Figura 3.4 ilustra os valores real, imaginário, do módulo e do espectro da *wavelet*-mãe para M igual a 12.

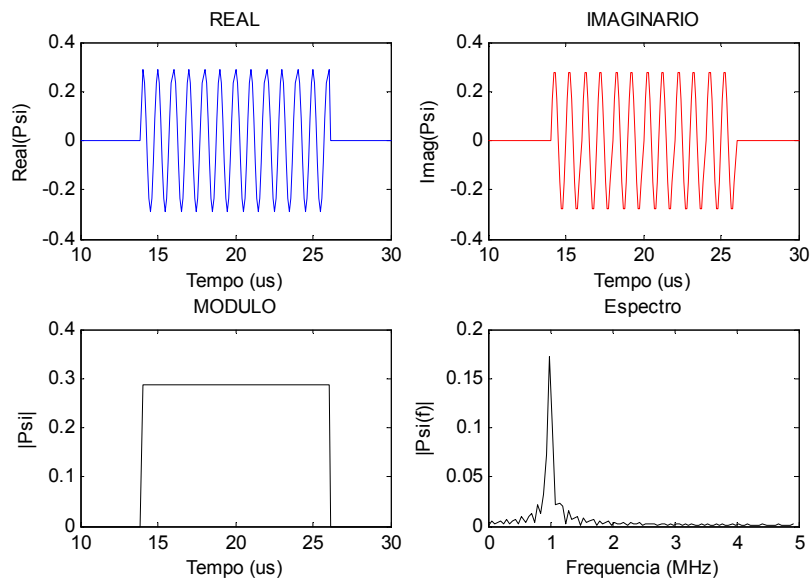


Figura 3.4 – Gráficos da *wavelet*-mãe $\psi(t)$ para M igual a 12: parte real, parte imaginária, módulo e espectro.

A *wavelet*-mãe $\psi(t)$ satisfaz a condição de admissibilidade da equação 3.3 quando M é um inteiro positivo, e é uma função janela, como na equação 3.6. Ainda, ela é oscilante e de média nula, satisfazendo a equação 3.5.

Isto posto, temos que a função detectora $D(T,t)$, formada pela CWT do sinal $s(t)$ com a *wavelet*-mãe $\psi(t)$ deslocada de t e escalonada de T , é dada por:

$$D(T,t) = \frac{T}{M} \left| \int s(t') \frac{1}{\sqrt{T}} \psi^* \left(\frac{t'-t}{T} \right) dt' \right|^2. \quad (3.10)$$

Uma detecção ocorre quando o valor da função $D(T,t)$ ultrapassa um determinado limiar a (*threshold*), e representa a ocorrência de uma seqüência de pulsos radar com valor de PRI igual a T no instante t . Um valor de limiar a sugerido pelo autor da técnica é 0,6.

Esta função detectora é sensível a variações de PRI de seqüências de pulsos radar de três tipos, a saber:

- a) Simples – o valor do PRI é constante;
- b) Stagger – as variações do PRI obedecem a uma seqüência de valores simples;
- c) Complexa – as variações do PRI são do tipo *jitter* ou *wobulated*, entre outras.

A lei de formação dos tempos de chegada dos pulsos para uma seqüência do tipo simples, com PRI igual a T_s , é dada por:

$$t_j = (j-1)T_s + t_\phi, \quad (3.11)$$

onde t_ϕ representa a fase ou tempo de referência.

Para uma seqüência do tipo *stagger* com K fases, $T_1, T_2, T_3, \dots, T_K$, que se repetem a cada K ciclos, a lei de formação da seqüência dos pulsos é dada por:

$$t_{j+K} = t_j, \quad (3.12)$$

onde os tempos de chegada da seqüência são:

$$t_1 = t_\phi; \quad t_j = T_{j-1} + T_j, \quad j = 2, 3, \dots, N. \quad (3.13)$$

Uma seqüência do tipo complexa pode ser de diversos tipos. Um exemplo é uma seqüência *wobulated*, cuja lei de formação dos valores de PRI é dada por:

$$T_j = A \cos(B_j + C), \quad (3.14)$$

onde A, B e C são constantes.

Uma propriedade importante resulta da equação 3.10. Uma seqüência de pulsos intercalados, $s(t)$, resulta da soma dos termos individuais $s_1(t) + s_2(t) + s_3(t) + \dots + s_N(t)$,

correspondendo às N seqüências individuais de pulsos radar. O produto interno $\int \psi^* s$ é uma transformação linear. Assim, se os termos de ordem quadrada forem pequenos se comparados com o valor 1, a função detetora $D(T,t)$ comporta-se como a soma das funções detetoras das seqüências individuais $D_1(T,t) + D_2(T,t) + D_3(T,t) + \dots + D_k(T,t)$.

Dessa forma, as características de período (PRI) das seqüências individuais de pulsos podem ser obtidas de $D(T,t)$ sem a necessidade de se realizar o *deinterleaving* da seqüência $s(t)$.

A Figura 3.5 ilustra uma detecção da função $D(T,t)$ no instante de tempo $t=70$ para 3 sinais intercalados, antes e após a limiarização (*threshold*) com valor 0,32.

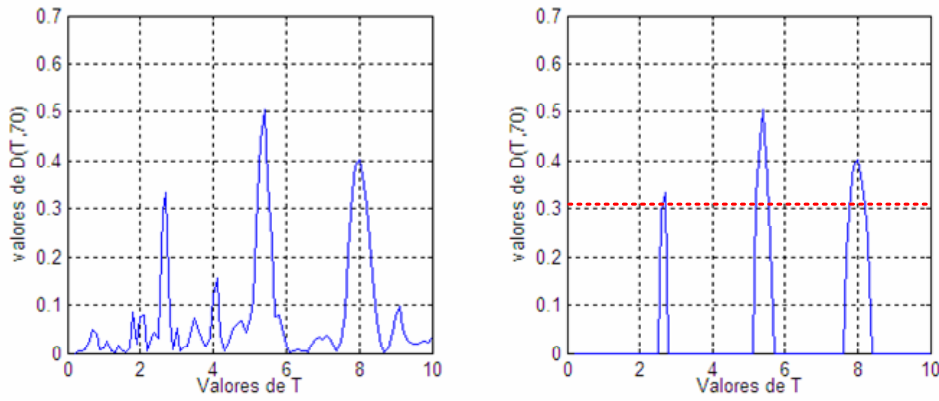


Figura 3.5 – Gráficos da função detetora $D(T,t)$ para 3 sinais, no instante de tempo $t=70$, antes e após a limiarização com valor 0,32.

A implementação dessa técnica foi realizada por MOLINA (2002) com uma modificação matemática. Seja a função $\tilde{\psi}_T(t)$ dada por:

$$\tilde{\psi}_T(t) = \frac{1}{\sqrt{T}} \psi\left(-\frac{t}{T}\right). \quad (3.15)$$

Dessa forma, a equação 3.10 pode ser reescrita como:

$$D(T,t) = \frac{T}{M} \left| \int s(t') \frac{1}{\sqrt{T}} \psi^*\left(\frac{t'-t}{T}\right) dt' \right|^2 = \frac{T}{M} \left| \int s(t') \tilde{\psi}_T^*(t-t') dt' \right|^2, \quad (3.16)$$

o que nos dá:

$$D(T,t) = \frac{T}{M} \left| s(t') * \tilde{\psi}_T^*(t') \right|^2 = \frac{T}{M} \left| \frac{1}{\sqrt{T}} s(t') * \psi^*\left(-\frac{t'}{T}\right) \right|^2, \quad (3.17)$$

permitindo, assim, a utilização da convolução como forma de calcular a CWT.

Como exemplo de aplicação da técnica, seja $s(t)$ um sinal formado de 3 seqüências de pulsos de sinais radar individuais, quais sejam:

- a) Seqüência de pulsos de sinal radar $s_1(t)$, com variação de PRI do tipo simples, com valor de PRI igual a $8 \mu\text{s}$, fase t_ϕ igual a $12 \mu\text{s}$ e 12 pulsos interceptados;
- b) Seqüência de pulsos de sinal radar $s_2(t)$, com variação de PRI do tipo *stagger* com 4 níveis, com valores de PRI iguais $2,2$ $2,4$ $2,8$ e $3,2 \mu\text{s}$, fase t_ϕ igual a $12 \mu\text{s}$ e 20 pulsos interceptados; e
- c) Seqüência de pulsos de sinal radar $s_3(t)$, com variação de PRI do tipo complexa *wobulated*, com valor de PRI central igual a $5 \mu\text{s}$, variação de até 10% em torno do valor central, com 22 pulsos por ciclo, fase t_ϕ igual a $0 \mu\text{s}$ e 44 pulsos interceptados.

A Figura 3.6 ilustra o sinal $s(t)$ obtido pela superposição dos sinais $s_1(t)$, $s_2(t)$ e $s_3(t)$.

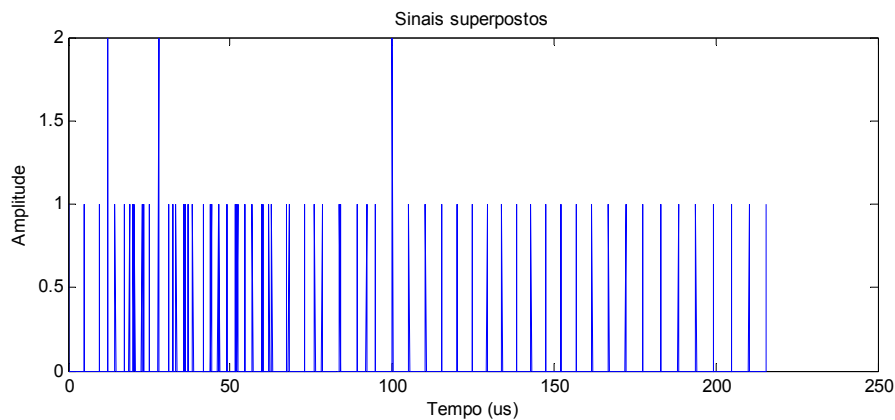


Figura 3.6 – Sinal $s(t)$ formado pelos sinais $s_1(t)$, $s_2(t)$ e $s_3(t)$ superpostos.

Utiliza-se, então, a função detectora no sinal $s(t)$, obtendo como resultado a energia espectral ao longo do tempo para cada valor de escala (PRI) testado nessa função. Essa energia espectral obtida pela função detectora pode ser apresentada graficamente, como na Figura 3.7, que é o escalograma limiarizado ilustrando a energia espectral obtida para o sinal $s(t)$. A limiarização permite melhor representação dos sinais presentes.

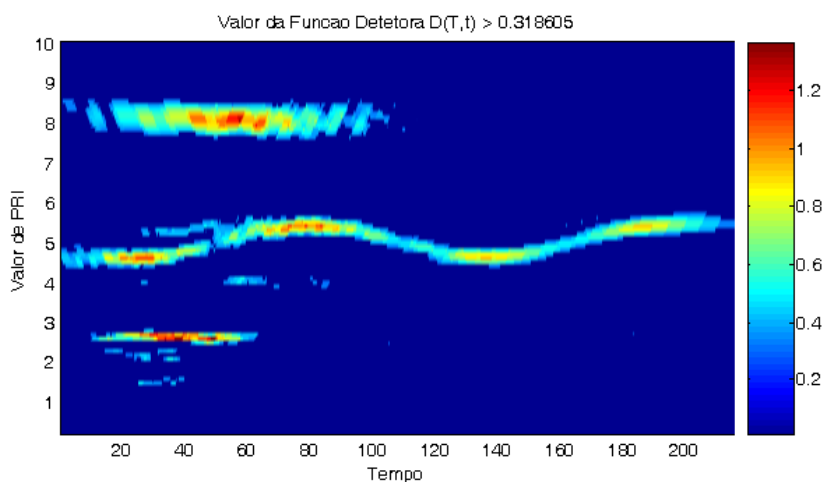


Figura 3.7 – Gráfico da função detectora $D(T,t)$ para o sinal $s(t)$ contendo os sinais $s_1(t)$, $s_2(t)$ e $s_3(t)$ superpostos.

A escolha do valor de limiarização proposto por MOLINA (2002) baseou-se em um valor empírico baseado nos valores de máximo, mínimo e desvio padrão dos valores de $D(T,t)$.

Observa-se que é possível a identificação das seqüências individuais de sinais radar $s_1(t)$, $s_2(t)$ e $s_3(t)$ e sua localização no tempo.

ZANANDREA, SAOTOME e FREITAS (2005) apresentaram uma variação da técnica, propondo a utilização da transformada rápida de Fourier (do inglês, *Fast Fourier Transform* – FFT) para o cálculo da CWT, e a modificação da *wavelet*-mãe.

A partir da equação 3.17, pode-se utilizar a FFT para o cálculo da convolução:

$$D(T,t) = \frac{T}{M} \left| \frac{1}{\sqrt{T}} s(t') * \psi^* \left(-\frac{t'}{T} \right) \right|^2 = M^{-1/2} \mathfrak{F}^{-1} [S(f) \cdot \Psi(f)]. \quad (3.18)$$

A *wavelet*-mãe proposta é a Morlet, que é dada pela seguinte equação:

$$\psi(t) = \frac{1}{(\sigma^2 \pi)^{1/4}} e^{-\frac{t^2}{2\sigma^2}} e^{2\pi i t}, \quad (3.19)$$

onde σ é o desvio padrão, e que é substituído pelo parâmetro M , o que nos dá:

$$\psi(t) = \frac{1}{(M^2 \pi)^{1/4}} e^{-\frac{t^2}{2M^2}} e^{2\pi i t}. \quad (3.20)$$

A Figura 3.8 ilustra os valores real, imaginário, do módulo e do espectro da *wavelet*-mãe Morlet para M igual a 12.

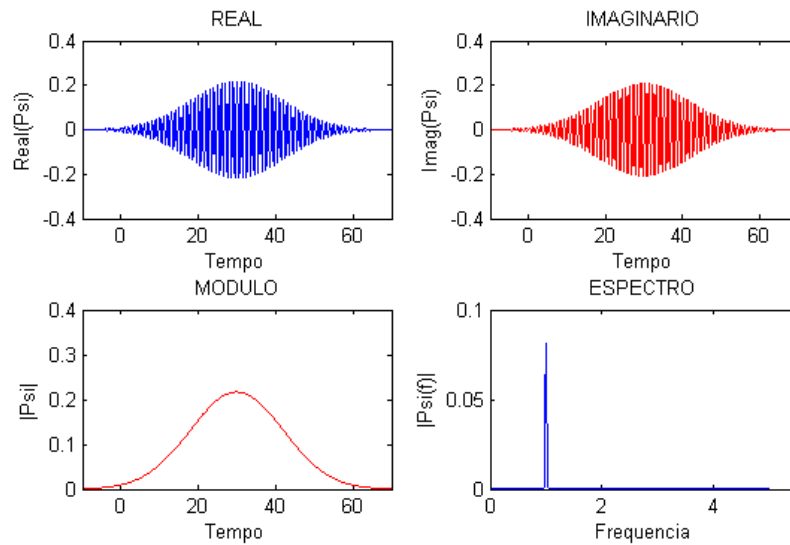


Figura 3.8 - Gráficos da *wavelet*-mãe Morlet para M igual a 12: parte real, parte imaginária, módulo e espectro.

Aplicando-se a função detectora $D(T,t)$ no sinal $s(t)$ criado acima, temos o resultado apresentado na Figura 3.9. A determinação do valor de limiarização foi a mesma para a obtenção da Figura 3.7.

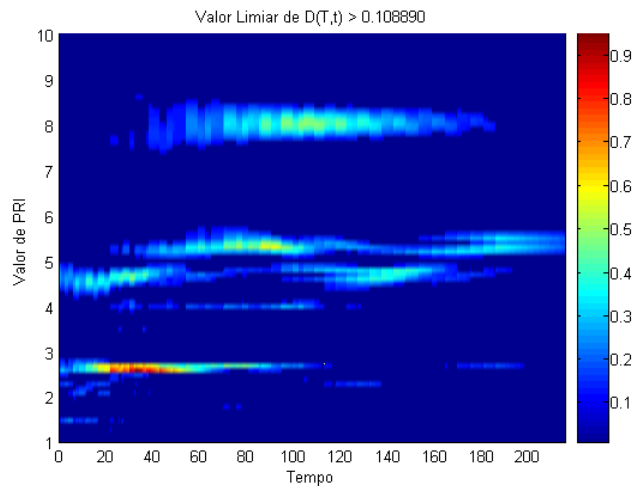


Figura 3.9 – Gráfico da função detectora $D(T,t)$ para o sinal $s(t)$ contendo os sinais $s_1(t)$, $s_2(t)$ e $s_3(t)$ superpostos.

A partir do que foi apresentado, observa-se que a técnica de *deinterleaving* por meio da transformada de *wavelet* contínua é eficiente na identificação dos valores de PRI das seqüências de pulsos radar recebidos. A técnica, por não ser muito precisa nos valores obtidos, é interessante num primeiro momento para a identificação, localização no tempo e

determinação da quantidade de emissores presentes. A partir do seu resultado, podem-se associar algoritmos de histogramação, entre outros, para a determinação precisa dos parâmetros desejados.

3.2 - FILTRAGEM DIGITAL FIR

Sistemas cuja resposta ao impulso tenha um número finito de amostras diferentes de zero são conhecidos como sistemas de resposta finita ao impulso (*Finite Impulse-duration Response* – FIR). Estes sistemas sempre serão estáveis desde que cada valor de resposta ao impulso seja finito (OPPENHEIM; SCHAFER, 1989).

Filtros digitais FIR são filtros digitais que, da mesma forma que os sistemas FIR, são caracterizados por uma resposta ao impulso que se torna nula após um intervalo de tempo finito. Além disso, são filtros não recursivos, e suas saídas dependem apenas da entrada atual e entradas anteriores (sistemas causais).

Há várias razões para o estudo de projetos de filtros digitais FIR, dentre as quais (NASCIMENTO, 2002):

- a) Filtros digitais FIR podem ser projetados com fase exatamente linear (atraso de grupo constante) para todas as frequências, o que simplifica o processo de aproximação, visto que envolve somente a aproximação de amplitude;
- b) Em aplicações de fase não linear, a implementação garante estabilidade;
- c) Filtros digitais FIR apresentam realizações eficientes, tanto em estruturas não-recursivas como em recursivas;
- d) As realizações não-recursivas são intrinsecamente estáveis;
- e) O ruído interno ocasionado pela precisão aritmética finita (quantização dos coeficientes do filtro e dos sinais) pode ser facilmente minimizado em realizações não-recursivas.

Esses filtros podem ser do tipo passa-baixa, passa-alta, passa-faixa ou rejeita-faixa, de acordo com sua seletividade em frequência.

Sua função de transferência é dada por

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^M b_k z^{-k} = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_M z^{-M}, \quad (3.21)$$

onde b_k são os coeficientes do filtro.

A estrutura básica de um filtro FIR é apresentada na Figura 3.10.

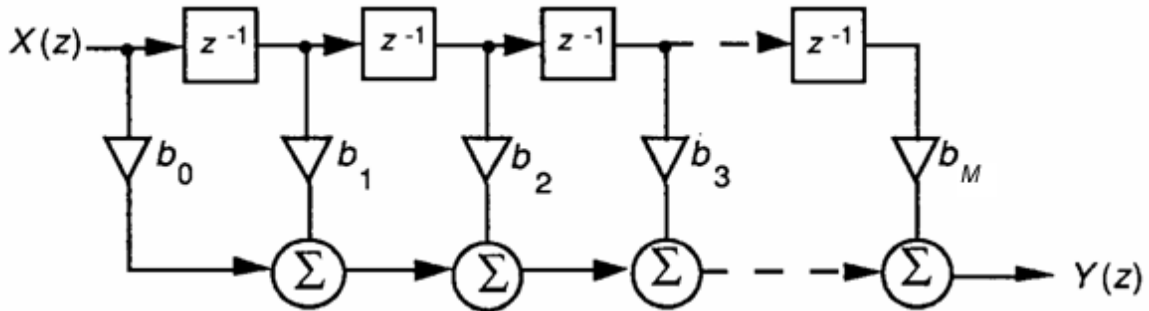


Figura 3.10 – Estrutura básica de um filtro FIR (ROCHA, 2006).

A partir da equação 3.21 ou da Figura 3.10 pode-se escrever a saída do filtro como

$$y[n] = \sum_{k=0}^M b_k x[n-k] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] + \dots + b_M x[n-M], \quad (3.22)$$

que representa a equação das diferenças lineares que descreve o filtro digital. Para o caso específico de filtros FIR, a equação 3.22 é identicamente igual à convolução linear.

Há diversas técnicas de projetos de filtros digitais FIR. Neste trabalho será abordada apenas a técnica de projeto de fase linear por janelamento da resposta impulsional desejada.

Supondo-se que se conheça a resposta impulsional desejada e que a mesma seja infinita, sua implementação seria por meio de um sistema de resposta infinita ao impulso (*Infinite Impulse-duration Response* – IIR). Por se tratar de um processo estável, a seqüência IIR deve tender para amplitude zero à medida que cresce o número de amostras. Assim, se truncarmos essa seqüência em algum ponto, teremos uma nova seqüência que é finita e pode ser realizada por meio de uma estrutura FIR que aproxima, no domínio das freqüências, o processo IIR original.

O método de janelamento inicia com a representação da resposta desejada em freqüência dada por

$$H_d(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h_d[n]e^{-j\omega n}, \quad (3.23)$$

onde $h_d[n]$ corresponde à resposta impulsional desejada, que é infinita. $h_d[n]$ pode ainda ser escrita em termos de $H_d(e^{j\omega})$ por meio da transformada inversa de Fourier em tempo discreto como

$$h_d[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\omega}) e^{j\omega n} d\omega. \quad (3.24)$$

A forma mais simples de se obter um filtro FIR a partir de $h_d[n]$ é multiplicando essa resposta desejada por uma janela retangular, ou seja,

$$h[n] = \begin{cases} h_d[n]; & 0 \leq n \leq M \\ 0; & \text{outros} \end{cases} \quad (3.25)$$

Genericamente, pode-se definir $h[n]$ como o produto da resposta impulsional desejada $h_d[n]$ por uma janela de duração finita $w[n]$ com $M+1$ elementos, ou seja,

$$h[n] = \begin{cases} h_d[n]w[n]; & n = 0, 1, \dots, M \\ 0; & \text{outros} \end{cases} \quad (3.26)$$

A multiplicação temporal leva a uma convolução no domínio das freqüências, podendo-se escrever

$$H(e^{j\omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\theta}) W(e^{j(\omega-\theta)}) d\theta. \quad (3.27)$$

onde

$$W(e^{j\omega}) = \sum_{n=0}^M w[n] e^{-j\omega n} \quad (3.28)$$

Assim, $H(e^{j\omega})$ é, no domínio da freqüência, a convolução periódica da resposta desejada pela transformada de Fourier da janela utilizada, implicando em que quanto mais próxima estiver $W(e^{j\omega})$ de um impulso, mais próxima estará $H(e^{j\omega})$ de $H_d(e^{j\omega})$.

O truncamento do sinal no domínio do tempo traz como consequência a ocorrência do fenômeno de Gibbs, que é uma oscilação do domínio das freqüências.

Algumas janelas conhecidas são Bartlett, Hamming e Hanning e retangular. A Figura 3.11 ilustra essas janelas no domínio do tempo.

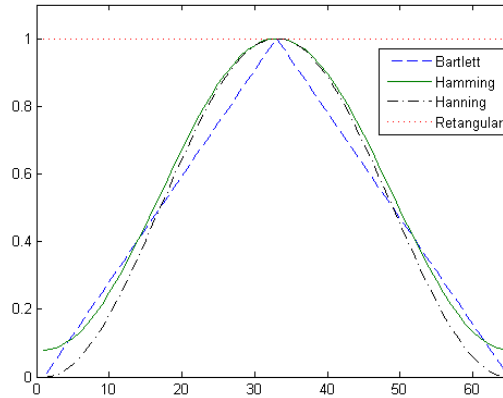


Figura 3.11 – Janelas Bartlett, Hamming, Hanning e retangular.

Para uma janela retangular, temos que a função janela, no domínio da frequência, é dada por

$$W(e^{j\omega}) = \sum_{n=0}^M e^{-j\omega n} = e^{-j\omega M/2} \frac{\sin[\omega(M+1)/2]}{\sin(\omega/2)}. \quad (3.29)$$

Assim, a amplitude da função janela, no domínio da frequência, é dada por

$$|W(e^{j\omega})| = \left| \frac{\sin[\omega(M+1)/2]}{\sin(\omega/2)} \right|. \quad (3.30)$$

A fim de que esses filtros sejam realizáveis, é necessário que sejam causais, exigindo um atraso de grupo positivo, dado por

$$\text{tg}(\omega) = -\frac{d\theta(\omega)}{d\omega} = +\alpha \quad (3.31)$$

onde $\theta(\omega)$ é a resposta de fase do filtro e $+\alpha$ corresponde ao atraso (temporal) a que todas as frequências serão submetidas quando aplicadas à entrada do filtro. O critério de fase linear exige que a resposta impulsional desejada $h_d[n]$ tenha simetria par ou ímpar. A função janela é simétrica em relação ao ponto $M/2$, o que nos permite reescrever a equação 3.29, para uma janela genérica, como

$$W(e^{j\omega}) = W_p(e^{j\omega})e^{-j\omega M/2} \quad (3.32)$$

onde $W_p(e^{j\omega})$ é uma função par em ω . Caso a resposta ao impulso desejada $h_d[n]$ seja também simétrica em relação a $M/2$, temos que a resposta ao impulso janelada também será simétrica, e sua resposta em frequência terá fase linear, ou seja,

$$H(e^{j\omega}) = A_p(e^{j\omega})e^{-j\omega M/2}, \quad (3.33)$$

onde $A_e(e^{j\omega})$ é uma função par em ω . Caso a resposta ao impulso desejada será anti-simétrica em relação a $M/2$, a resposta em frequência resultante será também anti-simétrica em relação a $M/2$, e a resposta em frequência terá fase linear com 90° de deslocamento constante, ou seja,

$$H(e^{j\omega}) = jA_p(e^{j\omega})e^{-j\omega M/2}, \quad (3.34)$$

onde $A_p(e^{j\omega})$ é uma função ímpar em ω .

3.3 - SUBAMOSTRAGEM DE SINAIS

Sinais digitais são amostras tomadas a intervalos regulares. Ocorre que algumas vezes a informação é digitalizada em uma taxa acima da necessária, sendo necessária a remoção de amostras do sinal.

Entretanto, nem sempre é possível apenas remover as amostras obtidas de acordo com a taxa de amostragem desejada. Fenômenos de superposição de espectros (do inglês, *aliasing*) podem ocorrer quando componentes do sinal possuem frequências superiores à frequência de Nyquist da nova taxa de amostragem.

Dessa forma, antes da realização da subamostragem, torna-se necessária uma filtragem passa-baixa no sinal original a fim de remover as componentes de frequência que podem causar superposição de espectros no sinal subamostrado.

Como exemplo, seja um sinal $x(t)$ dado pela soma de dois senos com frequências de 0,2 e 4 Hz, amostrados a uma taxa de 10 amostras/segundo. A Figura 3.12 ilustra as componentes em frequência do sinal amostrado.

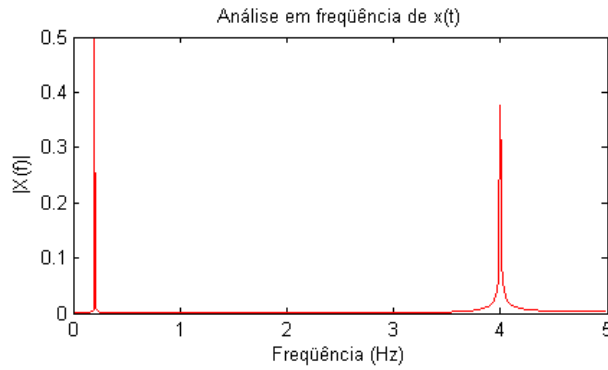


Figura 3.12 – Componentes de frequência do sinal $x(t)=sen(2\cdot\pi\cdot0,2\cdot t)+ sen(2\cdot\pi\cdot4\cdot t)$, amostrado a uma taxa de 10 amostras/segundo.

Supondo que o sinal $x(t)$ seja subamostrado a um fator de 2, ou seja, 5 amostras/segundo, ocorreria um efeito de superposição de espectros causado pela componente de 4 Hz, criando uma falsa componente no sinal $x(t)$ com 1 Hz. Isto ocorre pelo fato de a frequência obtida pela taxa de amostragem, 5 Hz, ser inferior à frequência de Nyquist necessária para o sinal em questão. A Figura 3.13 ilustra o efeito superposição de espectros obtido para a subamostragem descrita do sinal $x(t)$.

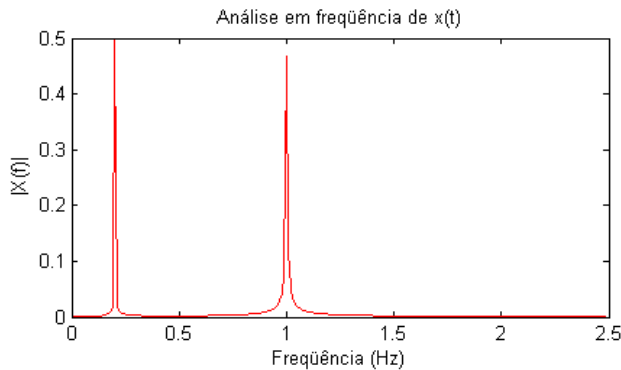


Figura 3.13 – Efeito superposição de espectros observado para as componentes de frequência de $x(t)$ quando subamostrado a uma taxa inferior à frequência de Nyquist.

Como solução, torna-se necessária a utilização de um filtro passa-baixa com frequência de corte f_c dada por

$$f_c = f_s / 2M , \quad (3.35)$$

onde f_s é a frequência de amostragem inicial, e M é o fator de subamostragem. A filtragem minimiza o efeito de superposição de espectros eliminando as frequências superiores ao valor máximo suportado pela nova frequência de amostragem.

No caso descrito, teríamos que a frequência de corte do filtro passa-baixa seria $f_c = 2,5\text{Hz}$. Realizando a filtragem antes da subamostragem, o fenômeno de superposição de espectros é minimizado, e a componente de 4 Hz é removida pela filtragem

A Figura 3.14 ilustra as componentes de frequência do sinal $x(t)$ após a filtragem passa-baixa com frequência de corte $f_c = 2,5\text{ Hz}$ e subamostragem de um fator de 2.

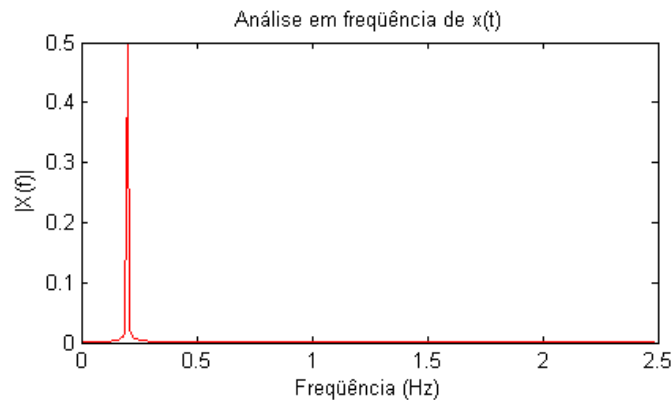


Figura 3.14 – Componentes de frequência de $x(t)$ após filtragem e subamostragem.

3.4 - REDES NEURAIS ARTIFICIAIS

As Redes Neurais Artificiais (RNA) tiveram como origem modelos matemáticos e de engenharia baseados nos neurônios biológicos. Contudo, convencionou-se chamar redes neurais artificiais a toda topologia de processamento de sinais constituída de vários elementos processadores simples altamente interconectados, isto é, estruturas baseadas no conexionismo (BAUCHSPIESS, 2004).

RNAs são sistemas paralelos distribuídos compostos por unidades de processamento simples (nodos) que calculam determinadas funções matemáticas (normalmente não-lineares). Tais unidades são dispostas em uma ou mais camadas e interligadas por um grande número de conexões, geralmente unidirecionais. Na maioria dos modelos estas conexões estão associadas a pesos, os quais armazenam o conhecimento representado no modelo e servem para ponderar a entrada recebida por cada neurônio da rede. O funcionamento destas redes é inspirado em uma estrutura física concebida pela natureza: o cérebro humano (BRAGA *et al*, 2000).

A solução de problemas através de RNAs é bastante atrativa, já que a forma como estes são representados internamente pela rede e o paralelismo natural inerente à arquitetura das RNAs criam a possibilidade de um desempenho superior ao dos modelos convencionais. A sua capacidade de aprender através de exemplos e de generalizar a informação aprendida é o atrativo principal da solução de problemas por meio de RNAs.

Em 1943, Warren McCulloch e Walter Pitts propuseram o primeiro modelo de neurônio artificial, conhecido como neurônio booleano. Logo em seguida, em 1949, Donald Hebb introduziu o conceito de aprendizado baseado no reforço. Mais tarde, Widrow e Hoff sugeriram uma regra de aprendizado conhecida como regra de Widrow-Hoff, ou regra delta, que ainda hoje é utilizada (BRAGA *et al*, 2000, KOVÁCS, 1996, BAUCHSIPESS, 2004). A Figura 3.15 ilustra o neurônio booleano de McCulloch e Pitts.

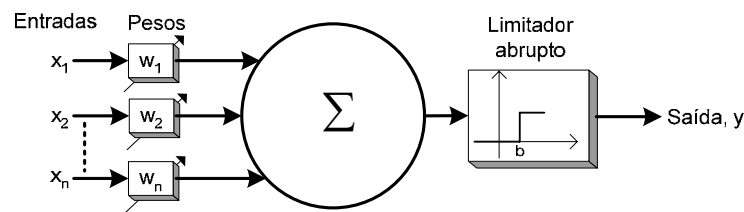


Figura 3.15 – Neurônio booleano de McCulloch e Pitts.

A saída do neurônio de McCulloch e Pitts é dada por:

$$y = f\left(\sum_{i=1}^m w_i x_i - b\right) = f(W^t X - b) \rightarrow y \in [0;1] \text{ degrau.} \quad (3.36)$$

Assim, o neurônio de McCulloch e Pitts (MCP) faz a divisão do espaço euclidiano \mathfrak{R}^m em duas regiões A e B tal que:

$$\begin{aligned} W^t X - b > 0 &\Rightarrow X \in A(y=1) \\ W^t X - b < 0 &\Rightarrow X \in B(y=0) \end{aligned} \quad (3.37)$$

Já em 1957, Frank Rosenblatt, dando prosseguimento às idéias de McCulloch, criou uma genuína rede de discriminadores lineares e chamou esta rede de *perceptron*. Os neurônios que recebem diretamente as entradas da rede constituem o que se chama de camada de entrada.

Na realidade, a camada de entrada nada mais é do que o próprio sinal de entrada. Os neurônios que recebem como entradas as saídas daqueles da camada de entrada constituem a segunda camada e assim sucessivamente, até a camada final que é a camada de saída.

As camadas internas que não são nem a de entrada e nem a de saída são geralmente referidas como camadas ocultas. A dimensão do vetor de saída amarra o número de neurônios na camada de saída. A Figura 3.16 ilustra o *perceptron* de Rosenblatt.

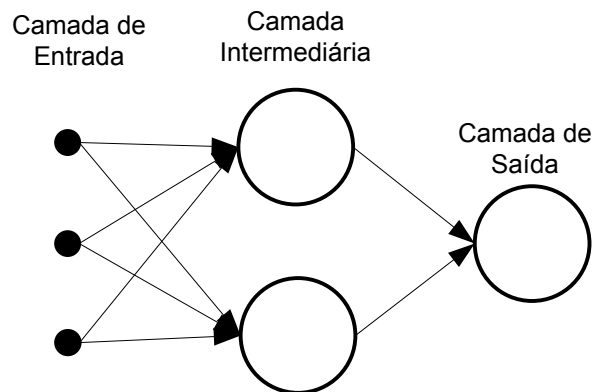


Figura 3.16 – Perceptron de Rosenblatt.

O modelo proposto por Rosenblatt tinha como unidades básicas os neurônios MCP e era composto de uma regra de aprendizado (BRAGA *et al*, 2000).

Em 1969, Minsky e Papert mostraram que o *perceptron* não era capaz de resolver problemas que não fossem linearmente separáveis. Isso intensificou o desinteresse nos trabalhos com RNAs, até que, em 1982, John Hopfield apresentou um artigo sobre as propriedades associativas das RNAs, criando novamente um interesse na comunidade acadêmica.

Em 1986, Rumelhart, Hinton & Williams mostraram por meio do algoritmo de treinamento *back-propagation* para o *perceptron* multicamadas (*Multilayer Perceptron* – MLP) que as RNAs de múltiplas camadas são capazes de resolver problemas não linearmente separáveis.

Teoricamente, redes com duas camadas intermediárias podem implementar qualquer função, seja ela linearmente separável ou não. A precisão obtida e a implementação da função objetivo dependem do número de nodos utilizados nas camadas intermediárias.

A utilização de funções de ativação na saída dos nodos derivou do modelo proposto por McCulloch e Pitts. Trata-se de funções aplicadas às saídas dos nodos permitindo quaisquer valores de saídas. Algumas funções conhecidas são linear, rampa, degrau e sigmoidal.

Um dos principais aspectos relacionados ao projeto de redes MLP diz respeito à escolha da função de ativação, as quais devem ser não lineares e diferenciáveis. A não linearidade é requerida para a resolução de problemas não linearmente separáveis. Já a necessidade de serem diferenciáveis justifica-se pelo cálculo do gradiente, que direcionará o ajuste dos pesos para minimização do erro. A Figura 3.17 ilustra as curvas das funções de ativação mais comuns.

Pesquisadores investigaram o número de camadas intermediárias necessárias para a implementação de classes de funções em uma RNA, e observaram que uma camada intermediária é suficiente para aproximar qualquer função contínua. Já duas camadas intermediárias são suficientes para aproximar qualquer função matemática.

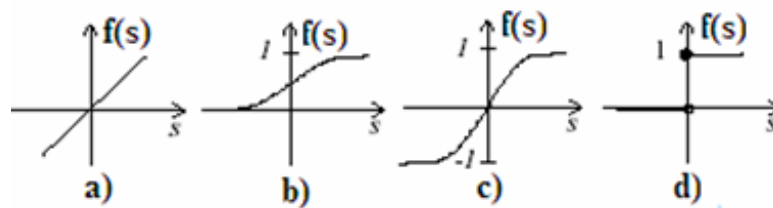


Figura 3.17 – Funções de ativação mais comuns: a) Linear; b) Sigmóide logística; c) Tangente hiperbólico; e d) Degrau.

A Figura 3.18 ilustra uma rede MLP com uma única camada e um neurônio, também conhecida apenas como *perceptron*.

Observa-se na Figura 3.18 que há uma entrada de polarização, o que permite uma soma ponderada não nula das entradas quando a soma destas é zero. Assim, o parâmetro de deslocamento b da função de ativação na equação 3.36 torna-se desnecessário.

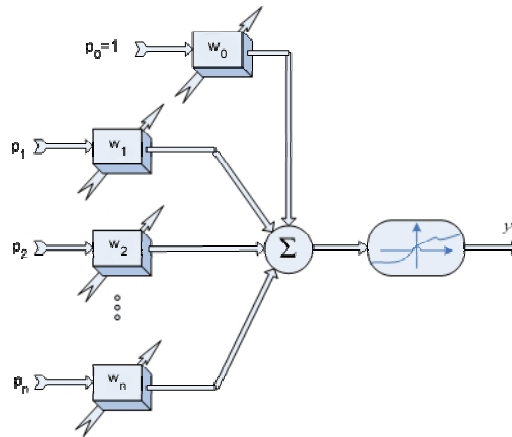


Figura 3.18 – *Perceptron* com uma única camada (modificado – BAUCHSIPESS, 2004).

A saída da rede da Figura 3.18 é dada por:

$$y = f\left(\sum_{k=0}^n p_k w_k\right), \quad (3.38)$$

onde f é a função de ativação aplicada à saída do nodo.

Em redes MLP com mais de uma camada, a equação 3.38 é aplicada à saída de todos os neurônios em todas as camadas. A Figura 3.19 ilustra uma rede MLP com duas camadas, dois neurônios na camada oculta e um neurônio na última.

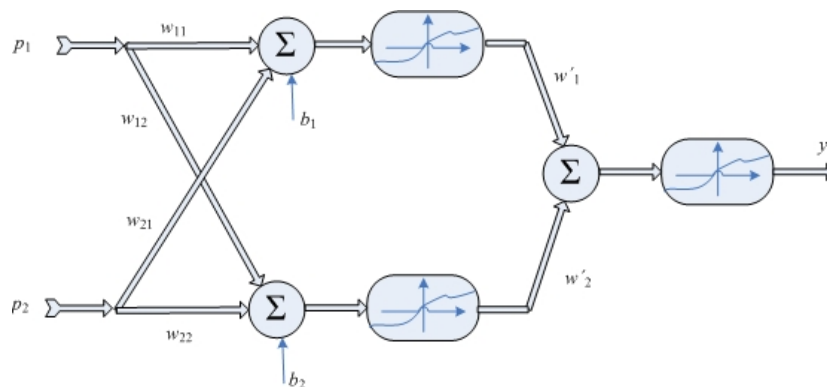


Figura 3.19 – Rede *perceptron* multicamadas, com duas camadas, dois neurônios na camada oculta e um neurônio na última (BAUCHSIPESS, 2004).

Atualmente, o aprendizado das redes MLP pode ser realizado por diversos algoritmos, os quais são geralmente do tipo supervisionado, ou seja, que requerem um agente externo que indique à rede um comportamento bom ou ruim.

Segundo BRAGA *et al* (2000), a utilização de uma RNA na solução de uma tarefa passa inicialmente por uma fase de aprendizagem, quando a rede extrai informações relevantes de padrões de informação apresentados para ela, criando assim uma representação própria para o problema. A etapa de aprendizagem consiste em um processo iterativo de ajuste de parâmetros da rede, os pesos das conexões entre as unidades de processamento, que guardam, ao final do processo, o conhecimento que a rede adquiriu do ambiente em que está operando. Há dois paradigmas principais para o aprendizado: aprendizado supervisionado e aprendizado não-supervisionado.

O aprendizado supervisionado tem esse nome em razão de a(s) entrada(s) e saída(s) desejada(s) para a rede serem fornecidas por um supervisor externo. Com isso, o ajuste dos parâmetros da rede é realizado de forma a se criar uma ligação entre os pares de entrada e saída fornecidos. O supervisor externo indica explicitamente um comportamento bom ou ruim para a rede, visando a direcionar o processo de treinamento. Os exemplos mais conhecidos de algoritmos para aprendizado supervisionado são a regra delta e a sua generalização para redes de múltiplas camadas, o algoritmo *backpropagation*.

No aprendizado não-supervisionado não há um supervisor externo para acompanhar o processo de aprendizado. Para estes algoritmos, somente os padrões de entrada estão disponíveis para a rede. A partir do momento em que a rede estabelece uma harmonia com as regularidades estatísticas da entrada de dados, desenvolve-se nela uma habilidade de formar representações internas para codificar características da entrada e criar novas classes ou grupos automaticamente. Há diversos métodos para implementação de aprendizado não-supervisionado. Contudo, não são objeto de estudo nesta dissertação.

O algoritmo de aprendizado mais conhecido para treinamento das redes MLP é o *backpropagation*, de forma que a maioria dos métodos de aprendizado para essas redes utiliza variações desse algoritmo.

Esse algoritmo é supervisionado, utilizando pares compostos por entrada e saída desejada para que, por meio de um mecanismo de correção de erros, os pesos da rede sejam ajustados. O treinamento ocorre em duas fases, conhecidas como fase *forward*, que define a saída da rede para um dado padrão de entrada, e fase *backward*, que a saída fornecida e a saída desejada para atualizar os pesos das conexões.

O algoritmo *back-propagation* é baseado na regra delta proposta por Widrow e Hoff, sendo por isto também chamada regra delta generalizada (BRAGA *et al*, 2000). Tem por objetivo a minimização de uma função de custo baseada na soma dos erros quadráticos:

$$E = \frac{1}{2} \sum_p \sum_{i=1}^k (d_i^p - y_i^p)^2, \quad (3.39)$$

onde E é a medida do erro total, p é o número de padrões, k é o número de unidades de saída, d_i é a i -ésima saída desejada e y_i é a i -ésima saída gerada pela rede.

Esse algoritmo é, em geral, muito lento e suscetível a problemas de treinamento, podendo ocorrer falsos pontos de convergência em regiões de gradiente próximo a zero (mínimos locais). Para acelerar a convergência, utilizam-se variantes do *back-propagation* (BRAGA *et al*, 2000), que utilizam tendências recentes da superfície de erro e momento, ou taxa de aprendizagem adaptativa e momento, entre outros, como por exemplo, os algoritmos *Quickprop*, *Resilient Backpropagation* e *Levenberg-Marquardt Backpropagation*.

Esse capítulo apresentou as ferramentas matemáticas necessárias à formulação da solução proposta. O próximo capítulo aborda alguns problemas da técnica da transformada de *wavelet* contínua e propõem um modelo para solucionar esses problemas e identificar a atividade de sinais radar por meio das ferramentas matemáticas apresentadas.

4 - IDENTIFICAÇÃO DE SINAIS RADAR PULSADOS POR MEIO DE TRANSFORMADA DE WAVELET CONTÍNUA E REDES NEURAIIS ARTIFICIAIS

Este capítulo aborda alguns problemas da técnica de transformada de *wavelet* contínua proposta por DRISCOLL e HOWARD (1999) e sugere um modelo para solucionar esses problemas e realizar a identificação de sinais radar por meio das redes neurais artificiais.

4.1 - Problemas da Técnica de Deinterleaving por meio da Transformada de Wavelet Contínua

A técnica proposta por DRISCOLL e HOWARD (1999), apesar de fundamentada na transformada de *wavelet* contínua, tem sua aplicação voltada para sistemas discretos, que requerem a amostragem dos sinais. A técnica requer a utilização da transformada contínua de *wavelet* uma vez que os valores de escalas testados são os valores de PRI que desejamos detectar, não sendo interessante utilizar a transformada discreta de *wavelet* com as escalas dadas por 2^j , com j inteiro, pois grande parte dos valores de PRI não seria testada.

Conforme apresentado na seção 2.1, os radares pulsados associados a armamentos têm sua frequência portadora principalmente na faixa entre 2 e 18 GHz, o que exigiria sistemas com taxa de amostragem da ordem de nano segundos de acordo com o Teorema de Nyquist.

Contudo, se a informação de frequência da portadora não é necessária ou não pode ser obtida (como nos receptores de cristal vídeo), o período de amostragem necessário é determinado pelos valores de frequência de repetição dos pulsos, PRF, ou o seu análogo, o PRI. Assim, sabendo-se que a faixa de PRI prático varia entre 2 e 10000 μ s, um período de amostragem aceitável seria da ordem de décimos de microssegundos.

A técnica proposta por DRISCOLL e HOWARD (1999) não é interessante para a análise de toda a faixa de PRI prático, pois não considera a complexidade computacional do cálculo da CWT para valores de PRI elevados.

Como exemplo, seja $t_x = 0,1 \mu\text{s}$ o período de amostragem de um sistema de recepção de sinais radar. Conforme apresentado na seção 3.1.1, o parâmetro M da técnica indica o menor número de pulsos que esperamos receber ou desejamos detectar em uma rajada de pulsos radar. Assim, considerando-se um radar com variação de PRI do tipo simples, com um valor de PRI pequeno, como $10 \mu\text{s}$, sua detecção seria possível por meio de um tempo de recepção de, aproximadamente,

$$12 \times 10 \mu\text{s} = 120 \mu\text{s}.$$

A partir do período de amostragem t_x apresentado acima, a função detectora $D(T,t)$ proposta por DRISCOLL e HOWARD (1999) trabalharia com um vetor contendo por volta de 1200 (120/0,1) elementos, o que é aceitável para aplicações em DSP.

Entretanto, se é pretendido detectar um segundo radar com variação de PRI do tipo simples, mas com valor de PRI alto, como $8000 \mu\text{s}$, a detecção seria possível somente com um tempo de recepção de, aproximadamente,

$$12 \times 8000 \mu\text{s} = 96000 \mu\text{s}.$$

Esse tempo de recepção, com o período de amostragem t_x , iria gerar um vetor contendo por volta de 960000 elementos (96000/0,1), o que tornaria inviável a utilização da função detectora $D(T,t)$ proposta por DRISCOLL e HOWARD (1999) em elementos de processamento digital.

Ainda que utilizando a transformada rápida de Fourier para o cálculo da função detectora $D(T,t)$, como apresentado por ZANANDREA, SAOTOME e FREITAS (2005), existe a limitação de performance em razão do tamanho do vetor do sinal com um período de amostragem t_x que satisfaça o critério de Nyquist para os valores práticos de PRI.

4.2 - Modificação da Técnica de Deinterleaving por Meio da Transformada de Wavelet Contínua

A ordem de grandeza da faixa de PRI prático utilizado em GE associada ao tempo necessário para a recepção de um número mínimo de pulsos de sinal radar para detecção representa um desafio para os sistemas computacionais baseados em digitalização.

Pode-se realizar a divisão dessa faixa de PRI prático em sub-faixas de PRI prático de acordo com o número mínimo de pulsos que se espera receber ou se deseja detectar para o maior valor de PRI de cada sub-faixa. Senão vejamos.

Seja $M=12$ o número mínimo de pulsos que se espera receber ou se deseja detectar para a realização do *deinterleaving* segundo a técnica da transformada de *wavelet* contínua. Assim, o menor tempo de recepção necessário para que um sinal com valor de PRI igual a T seja detectado fica da ordem de:

$$janela = M \cdot T = 12 \cdot T . \quad (4.1)$$

Uma proposta para a divisão da faixa de PRI prático em sub-faixas seria uma escala logarítmica, onde a precisão para os valores maiores de PRI seria menor em função de um período de amostragem maior. Isso poderia ser obtido dividindo-se toda a faixa de PRI prático em décadas, como de 1 a 10 μs , 10 a 100 μs , 100 a 1000 μs e 1000 a 10000 μs , com períodos de amostragem de 0,1 μs , 1 μs , 10 μs e 100 μs , respectivamente.

Dessa forma, torna-se possível designar janelas de tempo de recepção para cada sub-faixa de PRI prático de forma a permitir que a função detectora trabalhe com valores de tamanho aceitável. As janelas são determinadas pelo maior valor de PRI de cada janela, $12T_{maior,sub-faixa}$.

Primeiro, vejamos as definições das janelas de recepção e seus vetores.

Para a primeira sub-faixa, 1 a 10 μs , temos que a janela de recepção é dada por:

$$janela_{1,10} = 12 \cdot T_{maior1,10} = 12 \cdot 10 \mu s = 120 \mu s .$$

Essa janela, a um período de amostragem de 0,1 μs , fornece um vetor de tamanho $120/0,1=1200$.

Para a segunda sub-faixa de PRI, 10 a 100 μs , temos que a janela de recepção é dada por:

$$janela_{10,100} = 12 \cdot T_{maior10,100} = 12 \cdot 100 \mu s = 1200 \mu s .$$

Como ainda temos um período de amostragem de 0,1 μs , o vetor recebido teria $1200/0,1=12000$ elementos.

Para as sub-faixas de PRI entre 100 a 1000 μs e 1000 a 10000 μs , teríamos que as janelas de recepção seriam $janela_{100,1000} = 12000 \mu\text{s}$ e $janela_{1000,10000} = 120000 \mu\text{s}$, e os vetores de recepção teriam 120000 e 1200000 elementos, respectivamente.

Agora que foram definidas janelas de recepção adequadas ao número mínimo de pulsos $M=12$ para cada sub-faixa, ainda resta resolver o problema do tamanho dos vetores a serem submetidos à função detectora.

A subamostragem de sinais é uma alternativa para isto. Tendo em vista que as sub-faixas de PRI variam de um fator de 10, também poderíamos realizar subseqüentes subamostragens a um fator de 10, fazendo com que todos os vetores de recepção tivessem um tamanho de 1200 elementos. Senão vejamos.

Para a primeira sub-faixa, 1 a 10 μs , não seria realizada nenhuma subamostragem. Já para a segunda sub-faixa, 10 a 100 μs , realizaríamos uma primeira subamostragem na janela de recepção a um fator de 10, o que nos daria um período de amostragem de 1 μs e, para um tempo de recepção de 1200 μs conforme apresentado acima, um total de $1200/1=1200$ elementos.

Para terceira sub-faixa, 100 a 1000 μs , seria realizada uma segunda subamostragem do sinal já recebido. Assim, teríamos um período de amostragem de 10 μs e, para o tempo de recepção de 12000 μs dessa sub-faixa, um total de 1200 elementos.

Para a última sub-faixa, 1000 a 10000 μs , teríamos, analogamente, um período de amostragem de 100 μs e um vetor de recepção com 1200 elementos.

A implementação de algoritmos rápidos para processamento digital de sinais (PDS), como transformada rápida de Fourier e transformada rápida de *wavelet*, é realizada com vetores de tamanhos múltiplos de $N = 2^K$; $K \in \mathbb{Z}_+$. Assim, o menor múltiplo para o nosso sistema de recepção, que trabalha a um período de amostragem de 0,1 μs , seria 2048 (2^{11}). Esse valor satisfaz os requisitos de janela mínima de recepção baseada no valor do parâmetro $M=12$.

A Figura 4.1 representa graficamente as janelas de recepção sugeridas para análise das sub-faixas de PRI sem as subamostragens. Já a Figura 4.2 representa graficamente as janelas de recepção para as sub-faixas de PRI após as subamostragens sucessivas.

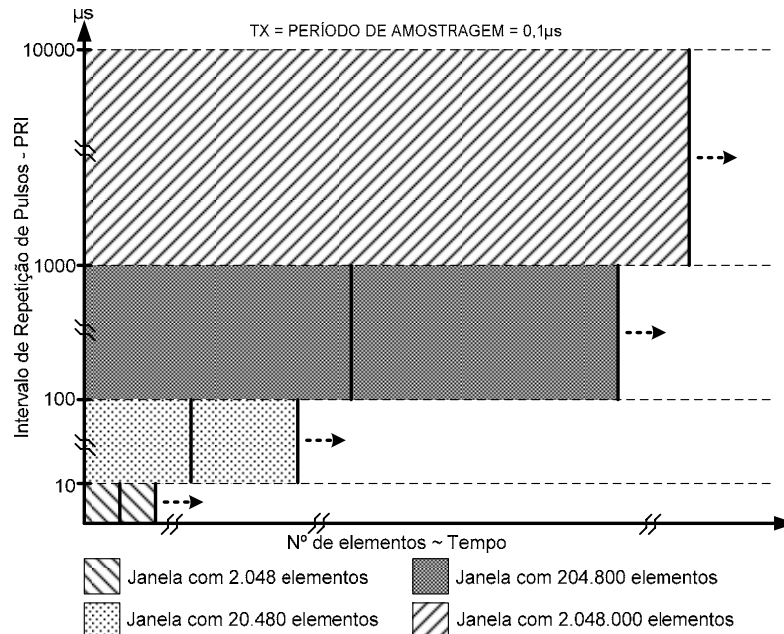


Figura 4.1 – Gráfico representativo das janelas de tempo necessárias para análise das sub-faixas de PRI prático.

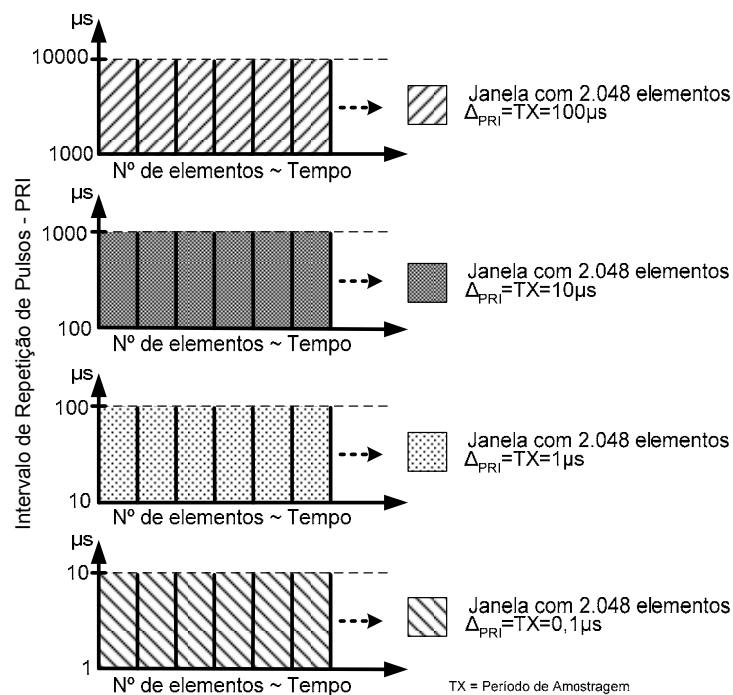


Figura 4.2 – Janelas do sinal recebido para análise das sub-faixas de PRI prático após subamostragens.

A partir das janelas de recepção com 2048 elementos, fica computacionalmente viável a utilização da técnica da transformada de wavelet contínua para o *deinterleaving*. Cabe lembrar que se faz necessária uma filtragem prévia à subamostragem para que não ocorra o efeito de superposição de espectros, conforme apresentado na seção 3.3.

Contudo, conforme apresentado na equação 3.8, os sinais recebidos são representados como superposição de impulsos. A utilização de subamostragens nos sinais recebidos pode representar um problema no momento em que os pulsos de um determinado sinal podem ser totalmente excluídos.

Dessa forma, torna-se interessante a inclusão da informação da largura de pulso (PW) dos pulsos recebidos, o que, inclusive, é mais condizente com os sinais obtidos dos receptores de guerra eletrônica baseados em cristal vídeo. Com isso, temos que um sinal $s(t)$ de um determinado radar, formado por uma seqüência de N pulsos com tempo de chegada t_j e largura de pulso τ , são representados como:

$$s(t) = \sum_{j=1}^N w\left(\frac{t-t_j}{\tau}\right) \quad (4.2)$$

onde $w(t)$ corresponde à janela retangular.

A Figura 4.3 ilustra a forma de representação proposta dos sinais radar em função da recepção dos sinais originais modulados.

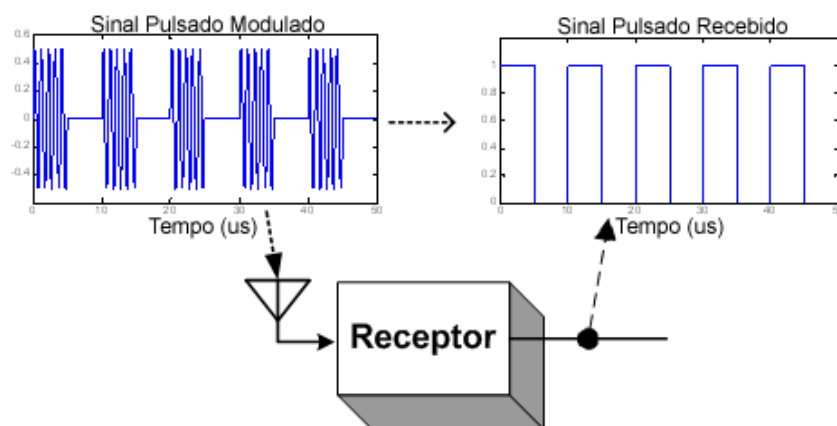


Figura 4.3 – Recepção de sinais radar pulsados modulados e conversão em sinais de onda quadrada com largura de pulso.

Com a inserção da informação da PW, ainda que as seqüências de pulsos tenham informação removida nas subamostragens, esse efeito é minimizado.

A separação em sub-faixas de PRI no sinal recebido requer o uso de outra ferramenta: a filtragem digital.

Pode-se projetar um filtro passa-faixa capaz de filtrar as interferências e ruídos provocados pelas sub-faixas de PRI que não estão sendo analisadas.

Sejam f_{c1} e f_{c2} as frequências de corte para a primeira sub-faixa de PRI, 1 a 10 μ s. Os valores de f_{c1} e f_{c2} são determinados pelo inverso dos limites da sub-faixa de PRI, e são 0,1 e 1 MHz, respectivamente. O valor da frequência de amostragem f_s dessa sub-banda é dado por $1/t_x = 10$ MHz.

Se ω_{c1} e ω_{c2} são as frequências de corte normalizadas, teremos:

$$\omega_{c1} = \frac{f_{c1}}{f_s/2} = 2 \cdot \frac{f_{c1}}{f_s} = 2 \cdot \frac{0,1\text{MHz}}{10\text{MHz}} = 0,02$$

$$\omega_{c2} = \frac{f_{c2}}{f_s/2} = 2 \cdot \frac{f_{c2}}{f_s} = 2 \cdot \frac{1\text{MHz}}{10\text{MHz}} = 0,2$$

Para as demais sub-faixas, analogamente ao cálculo apresentado, as frequências de corte normalizadas serão as mesmas.

Dessa forma, pode-se projetar um filtro digital FIR pelo método do janelamento com fase linear para isolar as sub-faixas de PRI desejadas.

Com isso, será empregado o pré-processamento apresentado para a segmentação da faixa de PRI prático em sub-faixas por meio das subamostragens e filtragem digital FIR, para em seguida utilizarmos a técnica de *deinterleaving* por meio da transformada de *wavelet* contínua para obtenção das características espectrais dos sinais radar pulsados para cada sub-faixa.

A Figura 4.4 representa como seria o sistema de detecção de seqüências de pulsos radar baseado nas janelas de recepção para as sub-faixas de PRI, nas subamostragens e nas filtragens digitais FIR com a utilização da transformada de *wavelet* contínua.

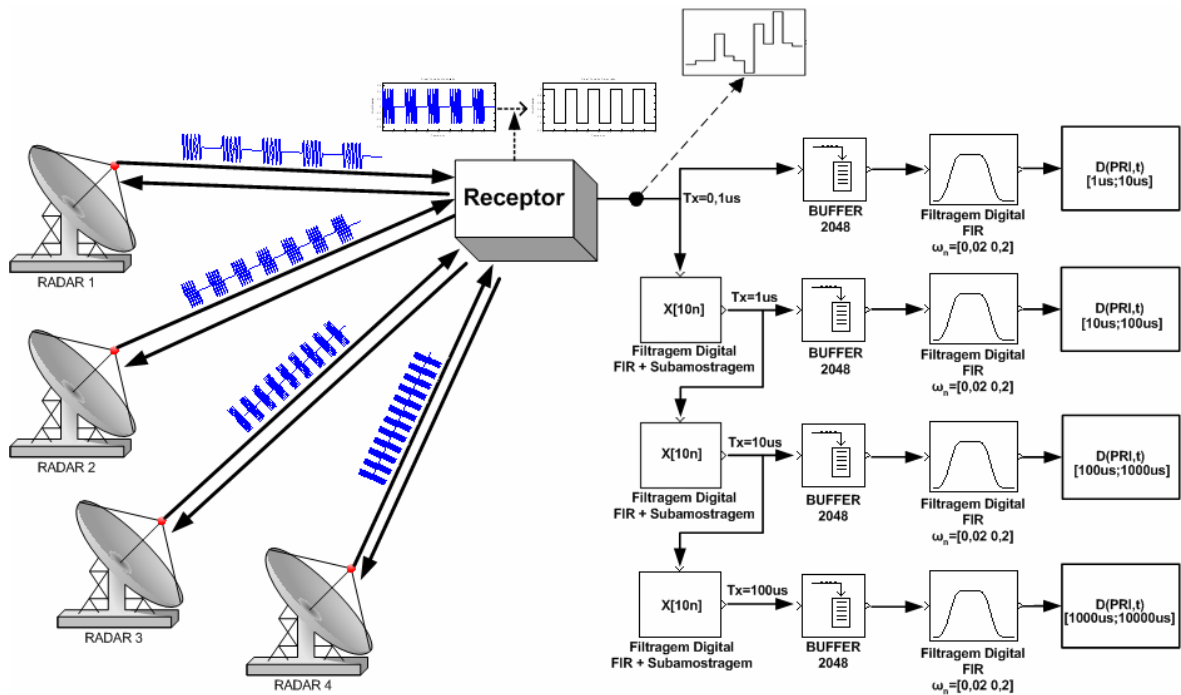


Figura 4.4 – Diagrama em blocos do sistema de *deinterleaving* de sinais radar pulsados por meio de técnica da transformada de *wavelet* contínua modificada com subamostragem e filtragem digital.

4.3 - Identificação de Sinais Radar por meio de Redes Neurais Artificiais

A identificação de radiações constitui uma das principais tarefas das Medidas de Apoio à Guerra Eletrônica – MAGE, conforme apresentado na Introdução, e tem como um dos objetivos permitir contra-resposta imediata no caso de presença de sinais que constituem ameaças à plataforma no ambiente analisado. A etapa de identificação deve ser rápida, a fim de que as contra-medidas aplicadas sejam eficientes.

Alguns sistemas como o Receptor de Alerta Radar possuem bibliotecas internas que armazenam assinaturas de sinais radar, possibilitando o confronto desses valores com os parâmetros obtidos por meio das técnicas próprias dos equipamentos, como as histogramações (MOLINA, 2002).

Durante a elaboração deste trabalho não se teve registro de nenhum equipamento que, na prática, utiliza redes neurais artificiais para a identificação dos emissores. Contudo, diversos trabalhos foram propostos nesse campo, como ZHANG *et al* (2004), que propõem

o uso de uma transformada *wavelet* adaptativa baseada em rede neural para extrair informações e reconhecer sinais radar com modulações intrapulso diferentes.

GRANGER *et al* (2001) propõem um modelo baseado em uma rede Fuzzy Artmap para classificação de sinais radar após a formação de conjuntos de pulsos em uma arquitetura “O quê e Onde” (do inglês, *What-and-Where*). A rede Fuzzy Artmap tem treinamento supervisionado, e é treinada para identificar os sinais de interesse. Sinais não reconhecidos pela rede possuem uma resposta padronizada. A Figura 4.5 ilustra o modelo proposto por GRANGER *et al* (2001).

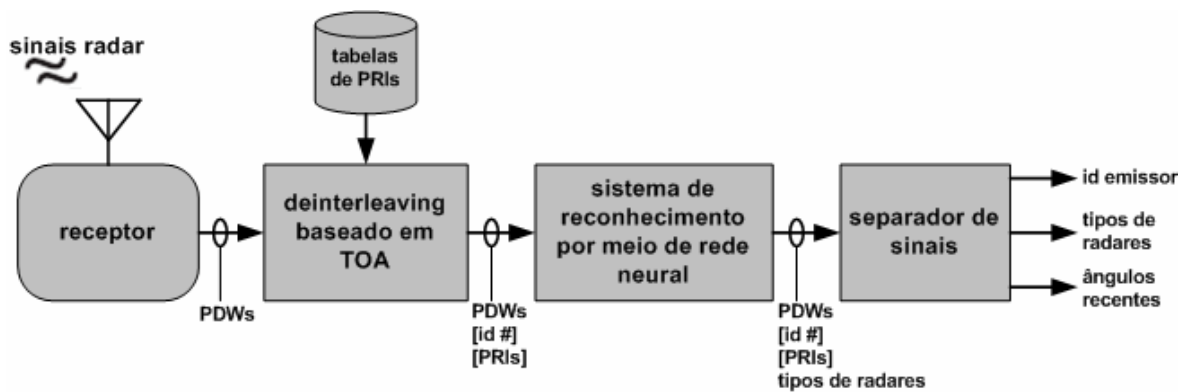


Figura 4.5 – Modelo de identificação de radares por meio de rede Fuzzy Artmap (modificado – GRANGER *et al*, 2001).

Neste trabalho é proposta a utilização de uma rede neural *perceptron* multicamadas (MLP) para a identificação da presença de sinais radar previamente conhecidos pela rede.

São construídas quatro redes MLP independentes, uma para cada sub-faixa de PRI. Cada rede tem como entrada o resultado da técnica de *deinterleaving* modificada para a sub-faixa em questão, e sua saída é definida pelo número de sinais radar que se deseja identificar, o que é definido na etapa de treinamento.

O esquema ilustrativo da conexão entre as redes MLP e as saídas da técnica de *deinterleaving* modificada é apresentado na Figura 4.6.

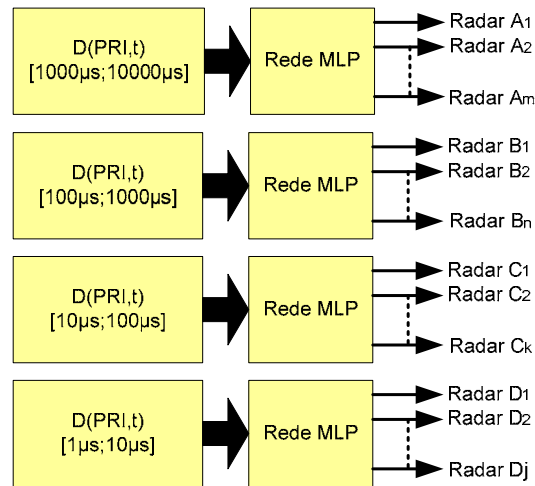


Figura 4.6 – Esquema de conexão entre as redes *perceptron* multicamadas e as saídas da técnica de *deinterleaving* modificada.

Cada rede MLP é capaz de identificar um conjunto de sinais radar de uma sub-faixa de PRI em função do treinamento realizado. Assim, para a sub-faixa de PRI entre 1 e 10 μ s, caso se queira identificar presença dos sinais radar D_1 , D_2 , ..., D_j (vide Figura 4.6), deve-se treinar a rede MLP associada a essa sub-faixa com as saídas esperadas da técnica para esse conjunto de sinais. Para as outras sub-faixas de PRI o esquema é análogo.

O número de entradas das redes MLP fica definido pela quantidade de valores de PRI testados pela técnica modificada na sub-faixa em questão. No esquema proposto, para a sub-faixa de PRI entre 1 e 10 μ s, o incremento do PRI nas janelas na transformada de *wavelet* é igual ao valor do período de amostragem para essa sub-faixa, 0,1 μ s, o que nos dá um total de 91 valores de PRI testados. Para a segunda sub-faixa de PRI, 10 a 100 μ s, o incremento também é igual ao valor do período de amostragem para essa sub-faixa, 1 μ s, também resultando num total de 91 valores de PRI testados. De forma análoga, para as outras sub-faixas, o número de valores de PRI testados é 91. Dessa forma, fica definido nesse sistema proposto que o número de entradas das redes MLP será 91.

A especificação do número de camadas ocultas e neurônios de cada camada não é um processo simples. Como o problema em questão tem características não lineares, será proposta a utilização de duas camadas, sendo uma camada intermediária e uma de saída.

Com relação ao número de nodos da camada intermediária, este será definido empiricamente em função de simulações, procurando-se evitar os problemas de

memorização dos padrões de treinamento (*overfitting*) e não convergência da rede (*underfitting*).

A estrutura da rede MLP proposta para a sub-faixa de PRI entre 1 e 10 μs é apresentada na Figura 4.7.

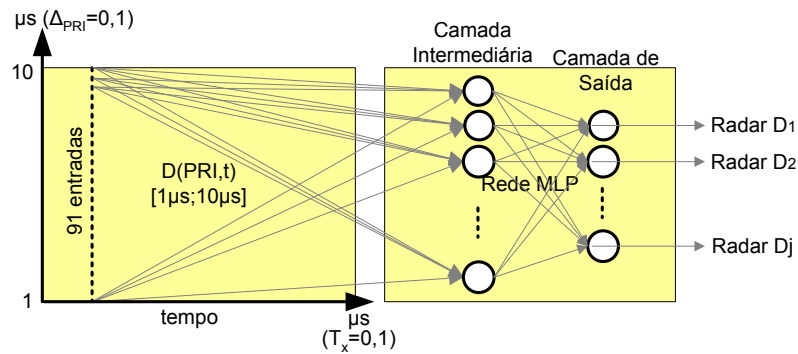


Figura 4.7 – Estrutura da rede MLP proposta. Há 91 entradas, uma camada intermediária com número de nodos definidos empiricamente e uma camada de saída com número de nodos definidos em função do número de sinais radar que se deseja identificar.

Dessa forma, aplica-se à rede MLP um conjunto inteiro de valores da função detectora em cada instante de tempo, a fim de verificar a presença ou ausência dos sinais aos quais a rede foi submetida a treinamento.

O modelo do sistema de identificação de sinais radar por meio da técnica modificada da transformada de *wavelet* contínua e redes neurais artificiais é apresentado na Figura 4.8.

Este capítulo apresentou toda a formulação necessária para a apresentação de um sistema de identificação de sinais radar que utiliza a técnica modificada da transformada de *wavelet* contínua e redes neurais artificiais. O próximo capítulo apresentará a implementação desse sistema no MATLAB e a análise das simulações.

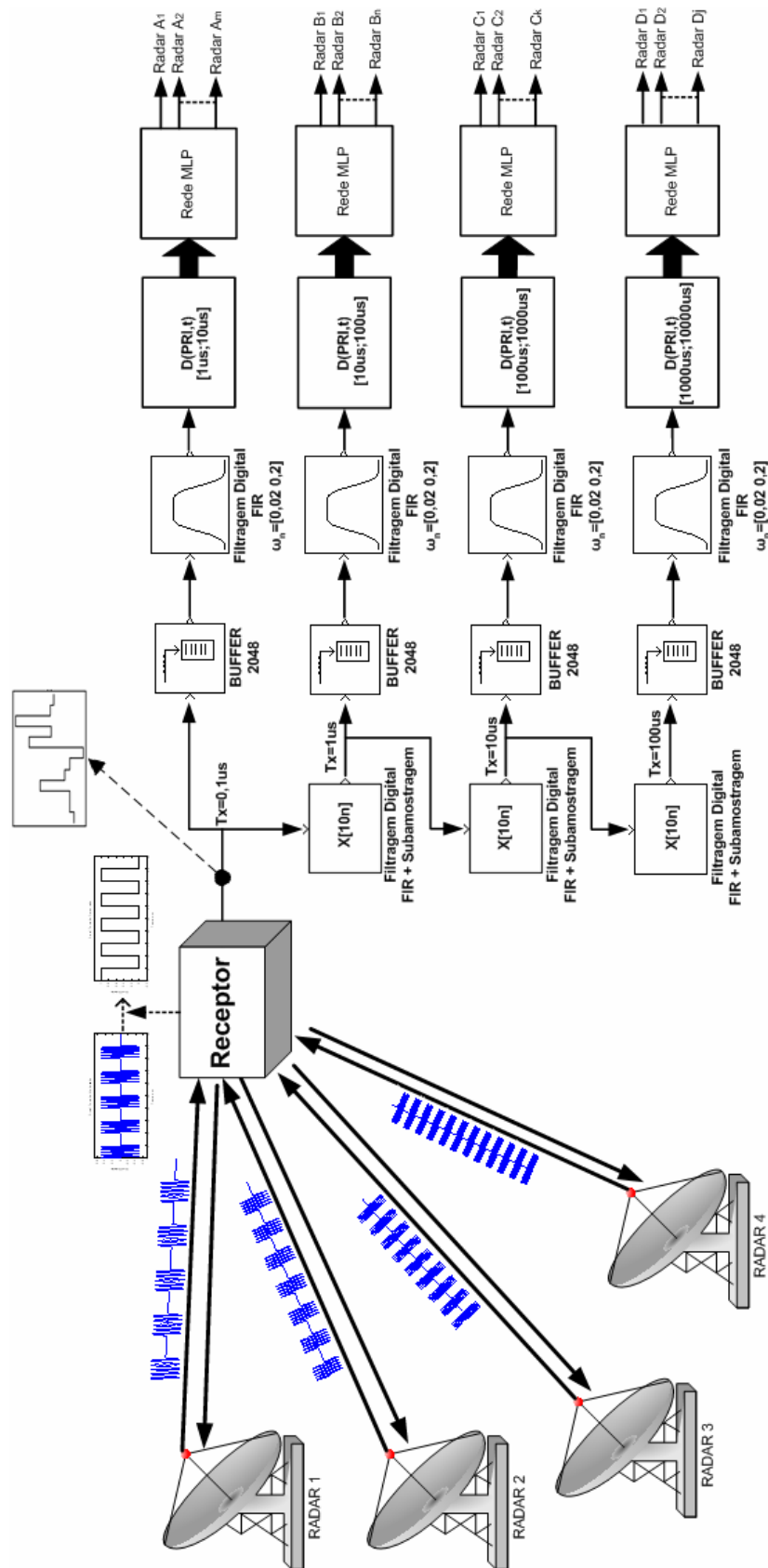


Figura 4.8 – Modelo do sistema de identificação de sinais radar por meio de transformada de *wavelet* contínua e redes neurais artificiais.

5 - IMPLEMENTAÇÃO, RESULTADOS DAS SIMULAÇÕES E DISCUSSÃO

Este capítulo apresenta os detalhes da implementação do sistema proposto, simulações e análise dos resultados.

5.1 - Implementação do Sistema de Identificação de Sinais Radar Proposto

O modelo de sistema de identificação de sinais radar apresentado foi implementado na plataforma MATLAB versão 6.5.

Os algoritmos de criação das seqüências de sinais radar seguem as leis de formação apresentadas na seção 3.1.1 e foram implementados de forma a receberem como parâmetros básicos os valores de tempo do início e fim do sinal, intervalo de repetição de pulsos (PRI), largura de pulso e taxa de amostragem. Esses parâmetros são básicos para todos os tipos de sinais criados. Contudo, sinais mais complexos exigem a existência de outros parâmetros, quais sejam:

- a) Seqüência de pulsos de sinal radar com PRI do tipo simples – Não requer parâmetros especiais;
- b) Seqüência de pulsos de sinal radar com variação de PRI do tipo *stagger* – O valor do parâmetro de PRI é substituído por um vetor de valores de PRI;
- c) Seqüência de pulsos de sinal radar com variação de PRI do tipo *wobulated* – Requer o valor de variação percentual em torno do PRI central e número de pulsos por ciclo, permitindo o cálculo dos valores de PRI para cada ciclo de variação senoidal; e
- d) Seqüência de pulsos de sinal radar com variação de PRI do tipo *jitter* – Requer o valor de variação percentual máxima em torno do PRI central, permitindo o cálculo de valores pseudo-randômicos de PRI em torno de um valor central.

O modelo de sistema proposto supõe um período de amostragem de $0,1 \mu\text{s}$ para os sinais após o receptor. Assim, todas as seqüências de pulsos de sinal radar, ao serem criadas, terão esse período de amostragem.

Para ilustrar os tipos dos sinais implementados, foram criadas quatro seqüências de pulsos de sinais radar, dos quatro tipos apresentados, conforme a Tabela 5.1.

Tabela 5.1 – Seqüências de pulsos de sinais radar ilustrativas.

Seqüência	Tipo de Sinal	PRI (μ s)	Δ PRI	PW (μ s)	Início (μ s)	Término (μ s)
A	Simple	2	-	0.2	0	120
B	<i>Stagger</i>	3.8, 4.0, 4.2 e 4.4	-	0.4	0	120
C	<i>Wobulated</i>	6	Até 10%, com 20 pulsos por ciclo	0.6	0	180
D	<i>Jitter</i>	8	Até 5%	0.8	0	160

Essas seqüências de pulsos, que aqui chamaremos de sinais A, B, C e D, a superposição desses sinais, o que é conhecido na literatura como *interleaving*, e os respectivos espectros de Fourier são apresentados graficamente na Figura 5.1. A superposição dos sinais é realizada mediante a simples soma dos sinais criados.

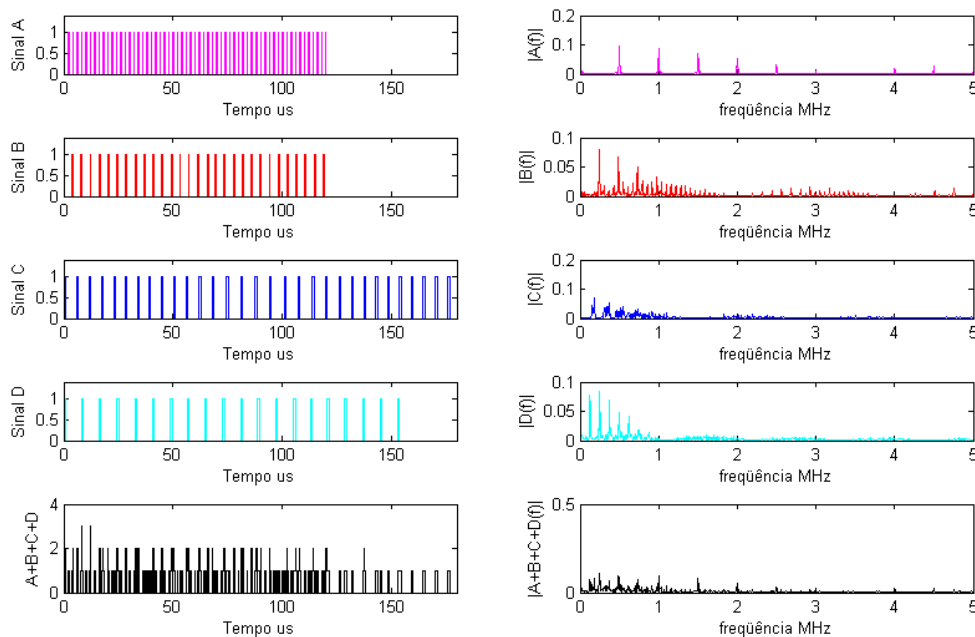


Figura 5.1 – Seqüências de sinais radar ilustrativas e seus espectros de Fourier: Sinal A: Simple; Sinal B: *Stagger*; Sinal C: *Wobulated*; Sinal D: *Jitter*; Sinal A+B+C+D: Superposição dos sinais.

O modelo de sistema de identificação de sinais radar apresentado no capítulo anterior prevê a utilização de subamostragens para a análise das sub-faixas de intervalo de repetição de pulsos (PRI) entre 10 e 100 μ s, 100 e 1000 μ s, e 1000 e 10000 μ s. Conforme exposto na seção 3.3, a utilização da subamostragem requer uma filtragem prévia do sinal a fim de minimizar o efeito de superposição de espectros (do inglês, *aliasing*). A subamostragem do sinal foi realizada utilizando previamente uma filtragem digital FIR

passa-baixa por meio de um filtro de ordem 30, com frequência de corte normalizada dada por $1/r$, com r igual ao valor do fator de subamostragem, que em nosso caso é 10.

Para a criação das janelas de tamanhos 2048 pontos utilizou-se um *buffer*, que toma o vetor do sinal subamostrado (ou não, para o caso da sub-faixa de PRI entre 1 e 10 μ s), e o divide em segmentos de tamanho $n = 2048$. O procedimento retorna uma matriz com cada coluna representando um vetor de 2048 elementos. Caso o vetor não seja múltiplo de 2048, o restante dos elementos da última coluna da matriz é completado com zeros.

Para a análise de todas as sub-faixas de PRI, existe a necessidade da filtragem de cada uma delas, conforme o modelo apresentado, a fim de minimizar as interferências e ruídos de outras sub-faixas.

Para isso, utilizou-se um filtro digital FIR com frequência de corte normalizada dada por ω_n , que em nosso caso é o intervalo 0,02 a 0,2 para um filtro passa-faixa, conforme apresentado na seção 4.2, e ordem 150. O filtro utiliza a técnica de projeto de fase linear por janelamento da resposta impulsional desejada com janela Hamming, com amplitude de resposta na frequência central da banda passante igual a 0 dB e atraso de grupo igual a 75 amostras. A Figura 5.2 apresenta a resposta impulsional, amplitude em decibéis e fase em graus para o filtro projetado.

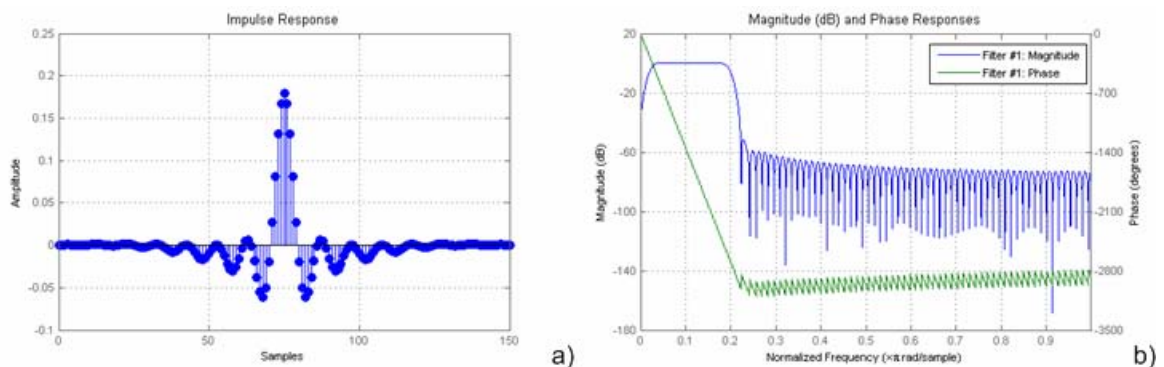


Figura 5.2 - Detalhes do filtro FIR passa-faixa projetado pela técnica de projeto de fase linear por janelamento da resposta impulsional desejada, com janela Hamming e ordem 150 para frequência de corte normalizada entre 0,02 e 0,2: a) Resposta ao impulso; b) Magnitude (dB) e fase (graus).

Com os procedimentos de subamostragens e filtragens temos os sinais separados em sub-faixas prontos para serem analisados pela técnica de *deinterleaving* por meio da transformada de *wavelet* contínua.

A implementação da técnica de *deinterleaving* por meio da transformada de *wavelet* contínua (CWT) foi aqui empregada com a utilização da transformada rápida de Fourier (FFT) conforme sugerido por ZANANDREA *et al* (2005), com a *wavelet*-mãe proposta pela técnica original de DRISCOLL e HOWARD (1999). Dessa forma calculam-se a transformada discreta de Fourier do sinal e a transformada discreta inversa de Fourier do vetor obtido, respectivamente.

Os 2048 coeficientes complexos da transformada contínua de *wavelet* para um determinado valor de escala, que em nosso caso é o valor de intervalo de repetição de pulsos (PRI), são obtidos por meio da função detectora $D(PRI, t)$ apresentada na equação 3.18. Estes coeficientes são normalizados e elevados ao quadrado, resultando num valor espectral de energia, que pode ser apresentado graficamente.

Ocorre que a utilização da FFT no cálculo da convolução linear implica na utilização de vetores do sinal e *wavelet*-mãe com 4095 elementos, que são obtidos por meio de acréscimo de zeros no final desses vetores originais. O resultado é então truncado tomando-se os 2048 elementos centrais, uma vez que a janela da *wavelet*-mãe é centralizada.

A Figura 5.3 ilustra as partes real e imaginária da *wavelet*-mãe proposta por DRISCOLL e HOWARD (1999) centralizada, bem como os escalogramas da energia espectral normalizada obtida pela função detectora $D(PRI, t)$, sem limiarização, com 4095 elementos e depois do truncamento em 2048 elementos, para os sinais superpostos A+B+C+D criados acima, após serem filtrados pelo filtro digital FIR passa-faixa proposto. Cabe lembrar que, por se tratar de sinais na sub-faixa de PRI entre 1 e 10 μ s, não foi necessário realizar a subamostragem.

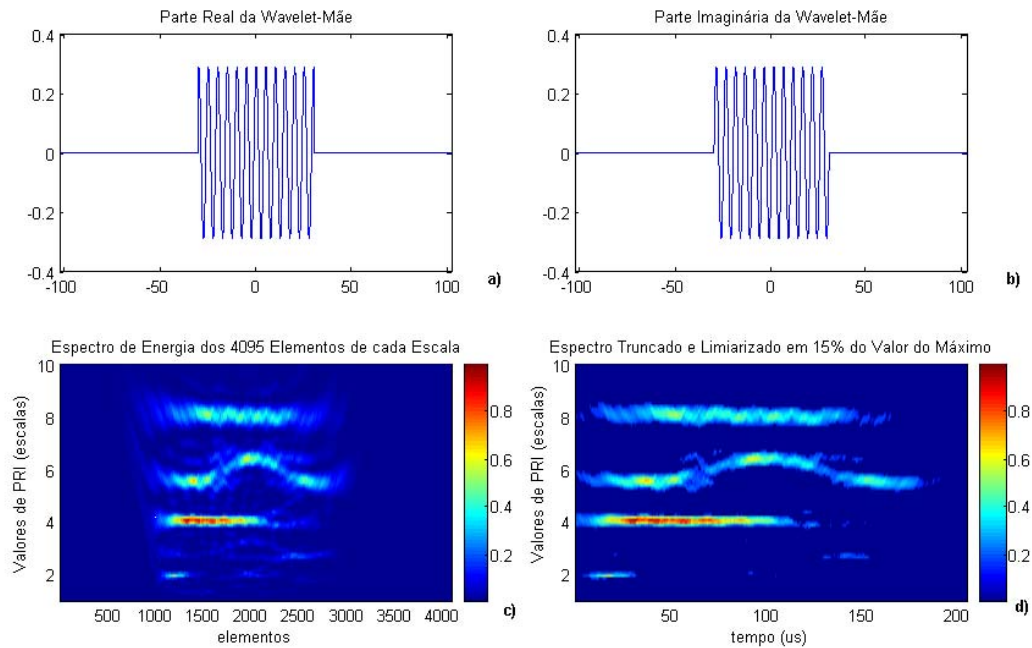


Figura 5.3 – Técnica de *deinterleaving* por meio da transformada de *Wavelet* Contínua: a) Parte real da *wavelet*-mãe centralizada; b) Parte imaginária da *wavelet*-mãe centralizada; c) Escalograma normalizado da energia espectral obtido pela função detectora $D(PRI, t)$ para 4095 elementos após técnica da FFT; d) Escalograma normalizado da energia espectral truncado em 2048 pontos e limiarizado em 15% do valor do máximo.

Neste ponto entra a identificação dos sinais radar por meio das redes *perceptron* multicamadas (MLP). Conforme apresentado no capítulo anterior, as redes MLP têm 91 entradas, representando a variação de PRI (escalas) utilizada no cálculo da função detectora para cada sub-faixa de PRI.

Há uma rede para cada sub-faixa de PRI, e cada rede possui 91 entradas e 2 camadas. A escolha do número de nodos da camada intermediária foi realizada empiricamente, tendo o cuidado de não ser um número pequeno, o que pode causar não convergência da rede (*underfitting*), nem muito grande, podendo ocorrer a memorização dos padrões de treinamento (*overfitting*). Dessa forma, o número de nodos escolhido inicialmente para a primeira camada será 40.

A função de transferência escolhida para a primeira e segunda camadas foi a sigmoideal logística, uma vez que ela é não-linear e diferenciável. A saída dessa função varia entre 0 e 1, e uma identificação de presença de um sinal radar será dada por um valor de saída acima de 0,5, e a ausência do sinal, por um valor abaixo de 0,5.

O algoritmo de treinamento da rede será uma variação do algoritmo *backpropagation*, conhecida como algoritmo Rprop (BRAGA *et al*, 2000), de *resilient backpropagation*. Essa função foi escolhida devido à moderada necessidade de memória e maior velocidade de convergência do treinamento, comparada com o algoritmo *backpropagation* original. O número máximo de épocas determinado para o treinamento dependerá do número de exemplos fornecidos.

O treinamento das redes MLP para cada sub-faixa é realizado com os padrões de energia espectral obtidos pela função detectora sem limiarização. Torna-se interessante a rede aprender a limiarizar o que é caracterizado como sinal e o que não é, visto que na medida em que vários sinais são inseridos, a interferência entre eles prejudica uma limiarização pré-determinada.

Para cada rede, cria-se um conjunto de treinamento formado pela simulação dos sinais isolados e superpostos que se deseja detectar. Para exemplificar, tomemos os sinais A, B, C e D acima.

Supondo que se queira identificar a presença desses quatro sinais radar, teríamos que a rede MLP designada para a sub-faixa de PRI entre 1 e 10 μ s teria quatro saídas, cada uma correspondendo a um sinal radar. Chamaremos então as quatro saídas como Saída Radar A, Saída Radar B, Saída Radar C e Saída Radar D.

A Figura 5.4 ilustra a representação da rede MLP designada para a identificação dos 4 sinais radar A, B, C e D na sub-faixa de PRI entre 1 e 10 μ s.

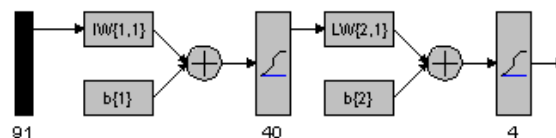


Figura 5.4 – Desenho da rede MLP designada para identificação dos sinais radar A, B, C e D na sub-faixa de PRI entre 1 e 10 μ s (MATLAB).

Seriam formados 5 conjuntos de treinamento encadeados, compostos por seqüências simuladas dos sinais individuais A, B, C e D, e a superposição destes, A+B+C+D. As saídas seriam a indicação de presença de cada sinal radar, de acordo com a ordem em que

esses são apresentados nas entradas do conjunto de treinamento. Esse tipo de treinamento é necessário uma vez que as redes MLP não possuem aprendizado incremental.

A Figura 5.5 apresenta o conjunto de treinamento que seria utilizado na rede MLP designada para a sub-faixa de PRI entre 1 e 10 μ s.

Como a filtragem digital FIR passa-faixa apenas minimiza a interferência das outras sub-faixas de PRI, é interessante que o último conjunto de treinamento mencionado seja criado incluindo também a superposição das outras seqüências de sinais simulados que se deseja identificar para as outras sub-faixas de PRI, que neste exemplo não existem.

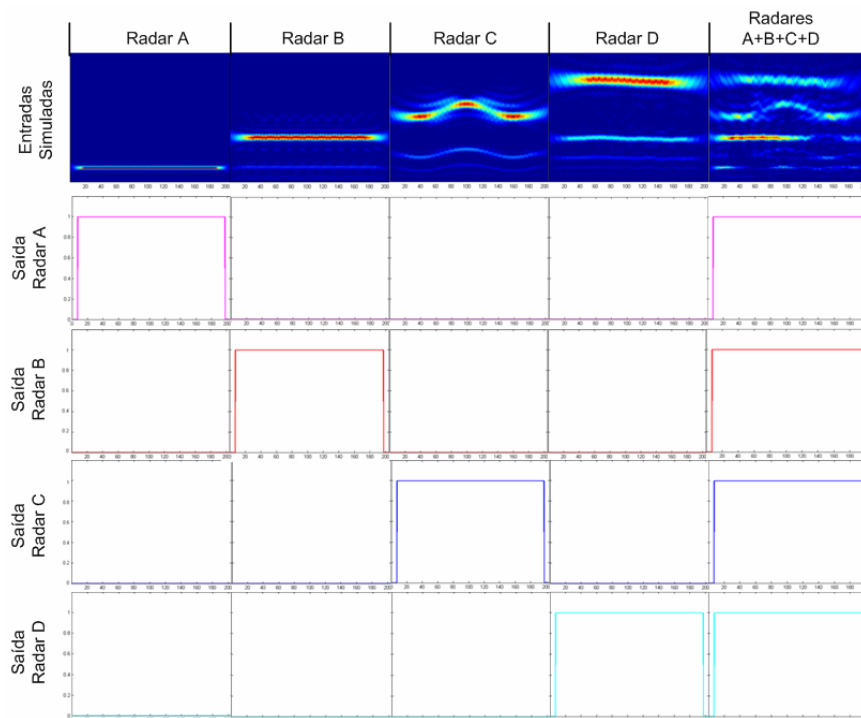


Figura 5.5 – Conjunto de treinamento da rede MLP associada à sub-faixa de PRI entre 1 e 10 μ s para identificação dos sinais radar A, B, C e D.

Outro ponto importante do treinamento é a consideração do atraso de grupo inserido pela filtragem digital FIR, que é refletido em um deslocamento dos espectros de energia obtidos pela função detectora. As filtragens digitais FIR fornecem um atraso de grupo de $n/2$, onde n é a ordem do filtro. Assim, para a filtragem passa-faixa, como a ordem do filtro é 150, o atraso de grupo inserido é de 75 amostras. Esses atrasos devem ser considerados nas saídas desejadas das redes neurais de cada sub-faixa de PRI. A filtragem FIR realizada nas subamostragens não insere atraso de grupo, uma vez que o algoritmo implementa a filtragem nos dois sentidos do vetor do sinal, removendo a distorção de fase.

O atraso inserido na sub-faixa de PRI entre 1 e 10 μs pela filtragem passa-faixa é de

$$75 \text{ amostras} \cdot 0,1 \mu\text{s/amostra} = 7,5 \mu\text{s}.$$

O atraso inserido na sub-faixa de PRI entre 10 e 100 μs pela filtragem passa-faixa é de

$$75 \text{ amostras} \cdot 1 \mu\text{s/amostra} = 75 \mu\text{s}.$$

Analogamente, os atrasos inseridos nas sub-faixas de PRI entre 100 e 1000 μs e entre 1000 e 10000 μs são, respectivamente, 750 μs e 7500 μs .

Após o treinamento das redes MLP, o sistema está preparado para a identificação dos sinais radar utilizados no treinamento.

A validação do sistema proposto é feita após o treinamento das redes MLP com os sinais radar de interesse. Prepara-se um conjunto de validação composto de sinais que podem conter ou não os sinais do treinamento, com início em instantes de tempo diferentes, como em um ambiente real.

5.2 - Simulações e Análise dos Resultados

Para validarmos o sistema proposto, iremos supor que se deseja detectar apenas um sinal radar em cada sub-faixa de PRI.

Sejam $s_1(t)$, $s_2(t)$, $s_3(t)$ e $s_4(t)$ as seqüências de sinais radar correspondentes aos sinais que se deseja identificar. Para o treinamento das redes MLP, essas seqüências são definidas conforme a Tabela 5.2.

Tabela 5.2 – Simulação das seqüências de sinais radar que se deseja identificar.

Seqüência	Tipo de Sinal	PRI (μs)	ΔPRI	PW (μs)	Início (μs)	Término (μs)
$s_1(t)$	Simple	8	-	0.8	0	190
$s_2(t)$	Stagger	58, 60, 61 e 63	-	6	0	1900
$s_3(t)$	Wobulated	400	Até 10%, com 20 pulsos por ciclo	40	0	19000
$s_4(t)$	Jitter	2000	Até 5%	100	0	190000

As quatro redes MLP para as quatro sub-faixas de PRI possuem apenas uma saída cada, uma vez que se deseja identificar apenas um sinal radar por sub-faixa de PRI. A Figura 5.6 ilustra o formato dessas redes.

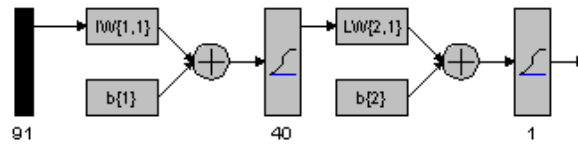


Figura 5.6 – Formato das redes MLP para identificação dos sinais radar $s_1(t)$, $s_2(t)$, $s_3(t)$ e $s_4(t)$.

O sinal formado pela superposição no tempo desses quatro sinais será chamado $s(t)$. Os conjuntos de treinamento criados para cada uma das redes MLP serão formados pelo encadeamento das detecções do sinal simulado isoladamente e do sinal $s(t)$ para cada sub-faixa de PRI. As Figura 5.7, Figura 5.8, Figura 5.9 e Figura 5.10 ilustram os conjuntos de treinamento para cada rede MLP associada a uma sub-faixa de PRI.

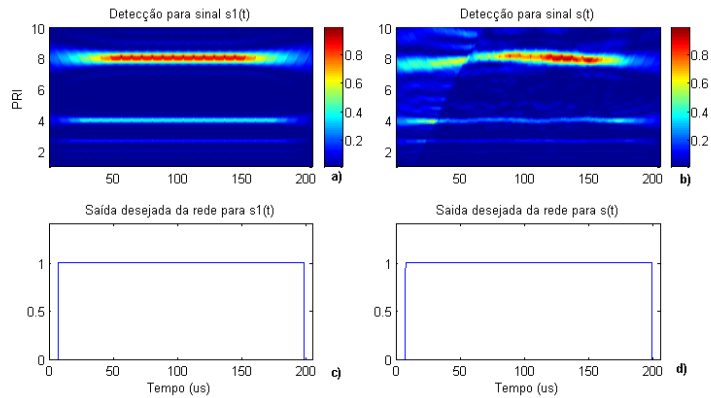


Figura 5.7 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 1 e 10 μ s formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_1(t)$; b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_1(t)$; d) Saída desejada da rede para $s(t)$.

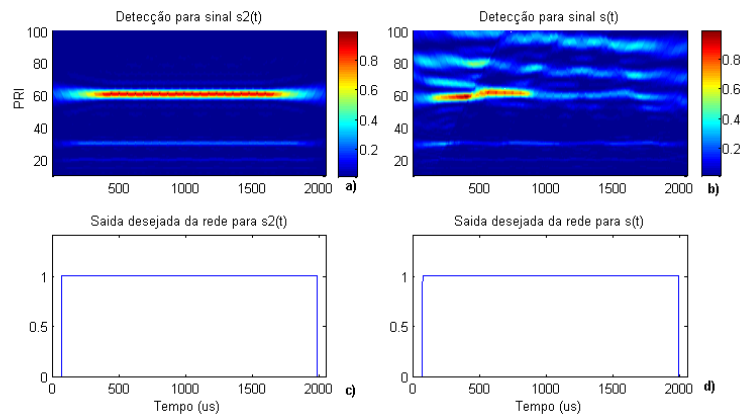


Figura 5.8 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 10 e 100 μ s formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_2(t)$; b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_2(t)$; d) Saída desejada da rede para $s(t)$.

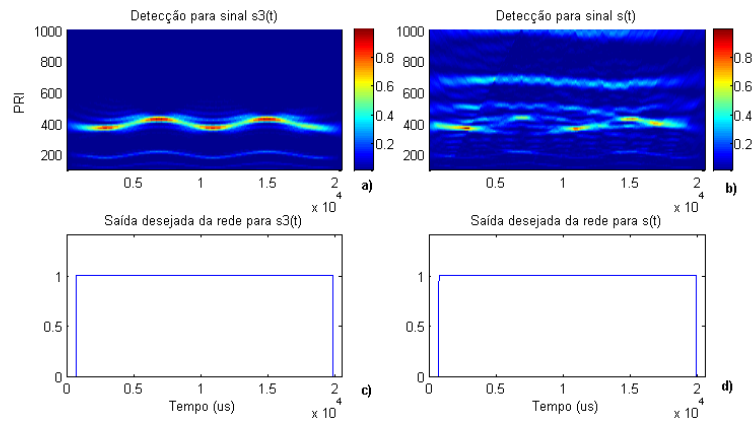


Figura 5.9 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 100 e 1000 μs formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_3(t)$; b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_3(t)$; d) Saída desejada da rede para $s(t)$.

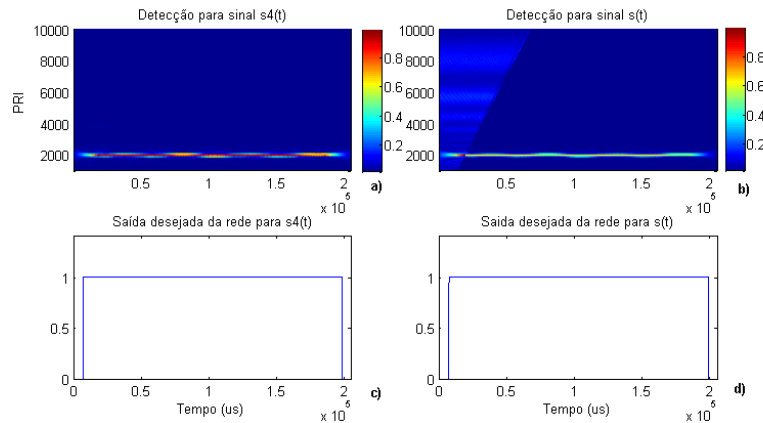


Figura 5.10 – Conjunto de treinamento para a rede MLP associada à sub-faixa de PRI entre 1000 e 10000 μs formado pelas detecções e saídas desejadas: a) Detecção para o sinal $s_4(t)$; b) Detecção para o sinal $s(t)$; c) Saída desejada da rede para $s_4(t)$; d) Saída desejada da rede para $s(t)$.

O treinamento foi realizado fixando-se o número de épocas em 200 e o objetivo de erro em zero. O número de exemplos de treinamento para cada entrada da rede, conforme os conjuntos apresentados acima, será de $2 \times 2048 = 4096$. Os gráficos das performances de treinamento obtidos no MATLAB para cada rede MLP são apresentados na Figura 5.11.

Após o treinamento das redes, podemos validá-las criando um conjunto de validação composto de seqüências de pulsos diversas, inclusive com seqüências de pulsos dos sinais radar que se deseja identificar.

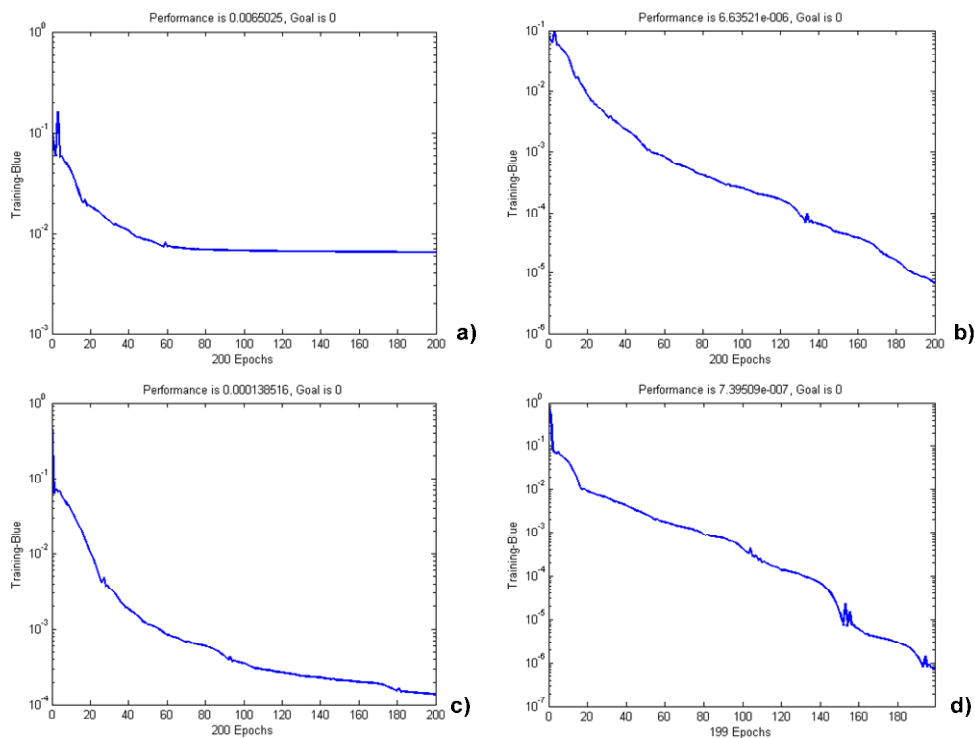


Figura 5.11 – Performances de treinamento para: a) Rede MLP associada à sub-faixa de PRI entre 1 e 10 μs ; b) Rede MLP associada à sub-faixa de PRI entre 10 e 100 μs ; c) Rede MLP associada à sub-faixa de PRI entre 100 e 1000 μs ; d) Rede MLP associada à sub-faixa de PRI entre 1000 e 10000 μs .

Seja $s_v(t)$ um sinal composto de 8 seqüências de pulsos superpostas ao longo do tempo, conforme a Tabela 5.3.

Tabela 5.3 – Seqüências de pulsos radar que compõem o sinal $s_v(t)$ para validação.

Seqüência	Tipo de Sinal	PRI (μs)	ΔPRI	PW (μs)	Início (μs)	Término (μs)
<i>A</i>	<i>Wobulated</i>	4	Até 5%, com 25 pulsos por ciclo	0.4	30	120
<i>B</i>	<i>Simples</i>	8	-	0.8	60	200
<i>C</i>	<i>Simples</i>	20	-	2	70	300
<i>D</i>	<i>Stagger</i>	58, 60, 61 e 63	-	6	50	2000
<i>E</i>	<i>Stagger</i>	700, 702, 706, 707	-	20	80	14000
<i>F</i>	<i>Wobulated</i>	400	Até 10%, com 20 pulsos por ciclo	40	140	17000
<i>G</i>	<i>Jitter</i>	5000	Até 5%	200	100	140000
<i>H</i>	<i>Jitter</i>	2000	Até 5%	100	220	160000

Observa-se que as seqüências de pulsos *B*, *D*, *F* e *H* correspondem, respectivamente, aos sinais radar $s_1(t)$, $s_2(t)$, $s_3(t)$ e $s_4(t)$ que desejamos identificar pelas redes MLP treinadas.

O sinal $s_v(t)$ criado é submetido ao modelo proposto, com as subamostragens, filtragens e função detectora para cada sub-faixa de PRI, para então ser validado pelas redes MLP

associadas a cada sub-faixa. Os resultados exibidos serão somente os de interesse em cada sub-faixa, uma vez que as sub-faixas de PRI menores têm sinais com duração menor do que as sub-faixas de PRI maiores. Assim, será apresentada somente a detecção e identificação dos primeiros 2048 elementos de cada janela de análise para cada sub-faixa de PRI. A Figura 5.12 ilustra as detecções e identificações do modelo proposto para o sinal $s_V(t)$.

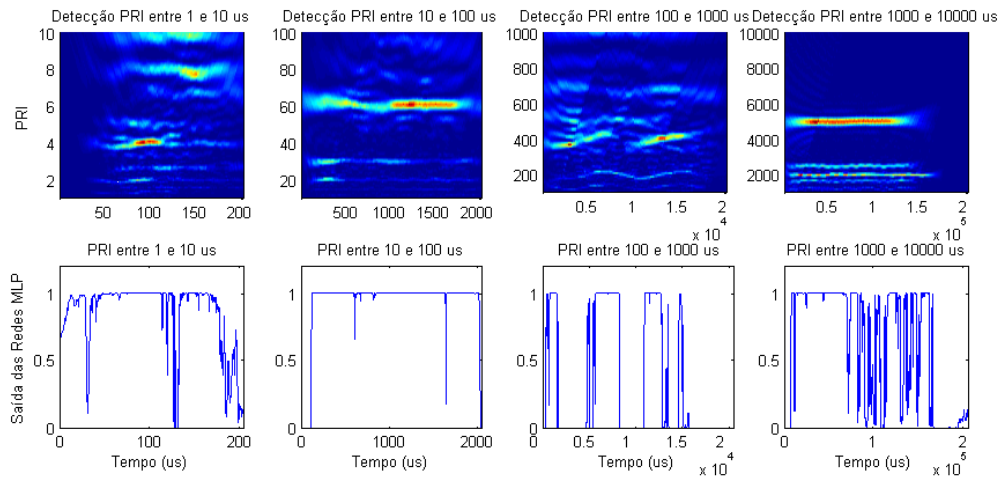


Figura 5.12 – Validação do modelo proposto para o sinal $s_V(t)$ com treinamento para identificação dos sinais $s_1(t)$, $s_2(t)$, $s_3(t)$ e $s_4(t)$. Acima estão as detecções da técnica da transformada de *wavelet* contínua para cada sub-faixa de PRI, e abaixo, as respectivas saídas das redes MLP associadas.

Pela análise dos resultados, observa-se que as detecções registram os espectros de energia dos sinais A , B , C , D , E , F , G e H com distorções mais acentuadas nos espectros de menor PRI, como pode ser observado comparando-se os espectros obtidos para as sub-faixas de PRI entre 1 e 10 μs (maior distorção) e entre 1000 e 10000 μs (menor distorção). Essas distorções são causadas pela interferência dos sinais presentes em sub-faixas de PRI superiores, uma vez que esses sinais incorporam harmônicos dos sinais presentes em sub-faixas de PRI inferiores.

As saídas das redes MLP associadas às detecções registram a presença dos sinais radar treinados, com algumas ocorrências de erros conhecidos como falsos positivos e negativos.

Os falsos positivos referem-se às saídas das redes MLP fixadas em um nível acima de 0,5, indicando a presença dos sinais com que as redes foram treinadas, sem contudo existir presença desses sinais. Esse tipo de erro pode ser observado entre o intervalo 0 a 60 μs na saída da rede MLP associada à sub-faixa de PRI entre 1 e 10 μs , uma vez que o sinal B ,

que corresponde ao sinal $s_1(t)$ com que a rede foi treinada, tem seu início somente no instante $t = 60 \mu\text{s}$.

Os falsos negativos referem-se às saídas das redes MLP fixadas em um nível abaixo de 0,5, indicando a ausência dos sinais com que as redes foram treinadas, havendo a presença desses sinais. Esse tipo de erro pode ser observado aproximadamente entre o intervalo 8500 e 10500 μs , uma vez que o sinal F , que corresponde ao sinal $s_3(t)$ com que a rede foi treinada, mantém-se presente nesse intervalo.

Os erros de identificação verificados podem ser causados por diversas razões, dentre elas:

- a) Número de nodos da camada intermediária abaixo ou acima do indicado;
- b) Treinamento insuficiente ou excessivo: se o número máximo de épocas para convergência for insuficiente, as redes não conseguem aprender a identificação. Caso contrário, as redes memorizam os padrões de treinamento, sendo incapazes de generalizar;
- c) Número de exemplos do conjunto de treinamento reduzido: se o número de exemplos do conjunto de treinamento das redes MLP for pequeno, o algoritmo de aprendizado pode não conseguir ajustar os pesos de forma a permitir que a rede generalize as diversificações dos conjuntos de validação.

Diante dessas razões, será proposta a modificação do conjunto de treinamento, número de nodos da camada intermediária, número máximo de épocas máximo do treinamento e objetivo de erro.

O novo conjunto de treinamento tem como única modificação o tempo de duração dos sinais simulados. Na Tabela 5.2, os instantes de término das seqüências $s_1(t)$, $s_2(t)$, $s_3(t)$ e $s_4(t)$ são, respectivamente, 190, 1900, 19000 e 190000 μs . No novo conjunto desses sinais, esses instantes serão 150, 1500, 15000 e 150000 μs .

O número de nodos da camada intermediária das redes MLP será aumentado para 80, assumindo que o número previamente escolhido, 40, é insuficiente para a rede aprender a classificar os padrões apresentados.

O número máximo de épocas para treinamento será reduzido para 180, e o objetivo de erro será fixado em 0,001. Com isso, o critério de parada do treinamento é antecipado, assumindo que as redes não sofrerão de memorização dos padrões de treinamento.

Após o treinamento das redes MLP, realizou-se a validação delas com o sinal $s_v(t)$, tendo como resultados as detecções e identificações apresentadas na Figura 5.13.

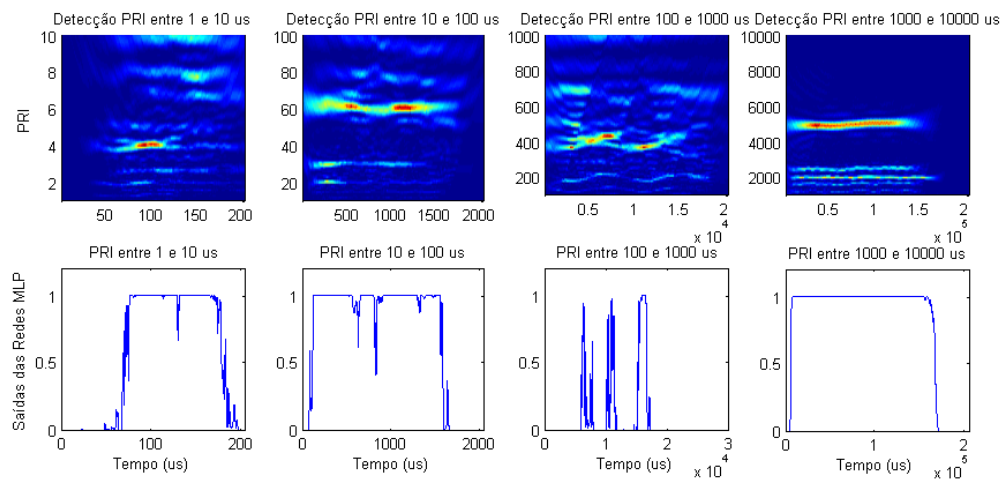


Figura 5.13 – Resultado da detecção e identificação do sinal $s_v(t)$ após modificação dos parâmetros e conjunto de treinamento.

A partir da Figura 5.13, observa-se que as identificações obtidas nas redes MLP associadas às sub-faixas de PRI entre 1 e 10 μs , 10 e 100 μs e 1000 e 10000 μs são satisfatórias, pois determinam com razoável precisão os instantes de início e término dos sinais que essas redes identificam. Contudo, a identificação apresentada pela rede MLP associada à sub-faixa de PRI entre 100 e 1000 μs apresentou uma alta taxa de falsos negativos. Com isso, verifica-se que a modificação nos parâmetros e conjunto de treinamento e número de nodos melhorou a identificação dos sinais radar, porém, os valores mais acertados não foram escolhidos.

A escolha adequada do conjunto de treinamento, número de nodos da camada intermediária, critérios de parada do treinamento requer um estudo aprofundado, não sendo o objetivo deste trabalho.

Os algoritmos desenvolvidos apresentaram um ótimo desempenho quando comparados com trabalhos anteriores referentes às técnicas da transformada de *wavelet* contínua padrão

e técnicas temporais de histogramação (MOLINA, 2002), inclusive com vários sinais radar pulsados presentes.

A partir dos resultados obtidos, acredita-se que é viável o emprego do modelo proposto para a detecção e identificação de sinais radar. Pode ser possível melhorar as detecções obtidas pela técnica da transformada de *wavelet* contínua com as subamostragens e filtragens digitais, uma vez que os efeitos de distorções e ruídos resultantes da superposição dos sinais dificultam a identificação promovida pelas redes MLP.

Um estudo sobre a escolha da *wavelet*-mãe pode ser promissor no sentido de adequar essas funções às características dos sinais após as superposições, subamostragens e filtragens digitais.

Este capítulo apresentou a implementação, simulações e resultados do modelo de identificação de sinais radar por meio da transformada de *wavelet* contínua e redes neurais artificiais. Foram discutidos os principais aspectos desse modelo, como necessidade de melhorias no treinamento e na técnica da transformada de *wavelet* contínua. O próximo capítulo apresentará as conclusões e considerações finais.

6 - CONCLUSÕES E RECOMENDAÇÕES

O problema endereçado neste trabalho foi propor a utilização de processamento de sinais digitais e redes neurais artificiais para a extração de características espectrais de sinais radar e promover sua identificação.

Para isso, foram apresentadas as características dos radares utilizados em guerra eletrônica (GE), com enfoque especial aos radares pulsados, formas clássicas de variação do Intervalo de Repetição de Pulsos (PRI), receptores e processamento em GE. Foi introduzido o problema da superposição de pulsos, que é solucionado pelas técnicas de *deinterleaving*. Algumas técnicas acadêmicas de *deinterleaving* foram apresentadas.

As ferramentas matemáticas necessárias para o desenvolvimento da solução foram detalhadas, justificando-se sua utilização no modelo proposto. A técnica de *deinterleaving* por meio da transformada de *wavelet* contínua foi apresentada, bem como modificações propostas por outros autores.

Os problemas existentes da técnica convencional de *deinterleaving* por meio da transformada de *wavelet* contínua foram discutidos, tendo sido proposto um pré-processamento por meio de subamostragens e filtragem digital FIR para segmentação da faixa de valores de interesse, viabilizando computacionalmente a utilização da transformada de *wavelet* contínua para extração das características espectrais dos sinais. Incluiu-se o reconhecimento dessas características espectrais por meio de redes neurais *perceptron* multicamadas (MLP) para a identificação dos sinais radar pulsados.

O modelo proposto foi implementado no MATLAB, permitindo verificar sua validade. Observou-se que o modelo atende razoavelmente sua finalidade de detectar e identificar sinais radar pulsados, pois as detecções por meio do pré-processamento e da análise pela transformada de *wavelet* contínua fornecem boa precisão espectral, permitindo a identificação das assinaturas espectrais dos sinais radar pulsados pelas redes MLP associadas a cada sub-faixa. Além disso, os algoritmos desenvolvidos têm complexidade computacional factível, o que proporcionou um bom desempenho nas simulações.

Um estudo aprofundado nas diversas características do modelo pode melhorar sua capacidade de detectar e identificar os sinais radar pulsados.

Para isso, sugerem-se como trabalhos futuros, a pesquisa de *wavelets*-mãe adequadas às características dos sinais radar após as filtrações e subamostragens, bem como um estudo da filtração digital utilizada para reduzir a interferência dos sinais radar nas sub-faixas analisadas. Também é conveniente um estudo mais aprofundado na escolha do número de nodos da camada intermediária e dos critérios de treinamento das redes neurais MLP, a fim de encontrar um ponto ótimo que permita uma excelente identificação dos sinais com que as redes foram treinadas.

REFERÊNCIAS BIBLIOGRÁFICAS

ADAMY, D. **EW 101: A First Course in Electronic Warfare**. Boston: Artech House, 2001. p. 1-6, 49-71, 73-97.

BAUCHSPIESS, A. **Introdução aos Sistemas Inteligentes – Aplicações em Engenharia de Redes Neurais, Lógica Fuzzy e Sistemas Neuro-Fuzzy**. Brasília: Universidade de Brasília, mar. 2004. 71 p. (Notas de aula).

BRAGA, A. P., et al. **Redes Neurais Artificiais: Teoria e Aplicações**. Rio de Janeiro: LTC, 2000. p. 1-73.

CHUI, C. K. **WAVELET ANALYSIS AND ITS APPLICATIONS. Wavelets: A Tutorial in Theory and Applications**. San Diego: Academic Press, 1992. v. 2, 729 p.

COELHO, L. S. **Introdução às Redes Neurais Artificiais**. [s.l.]: Pontifícia Universidade Católica do Paraná, 2004. Disponível em: <http://www.las.pucpr.br/leandro/control_e_avancado/A16_RN_Introdução.pdf>. Acesso em: 14 mar. 2004.

DRISCOLL, D. E., HOWARD, S. D. The Detection of Radar Pulse Sequences by Means of a Continuous Wavelet Transform. **IEEE Proceedings on International Conference, Acoustics, Speech, and Signal Processing**, [s.l.], vol. 3, p. 1389-1392, mar. 1999.

ELTON, S. D., GRAY, D. A. The Application of Circular Statistics to Specific Radar Pulse Train Detection. **Proceedings EUSIPCO '94**, [s.l.], vol. 1, p. 284-287, 1994.

FOGEL, E. GAVISH, M. Parameter estimation of quasi-periodic sequences. In: International Conference on Acoustics, Speech and Signal Processing, [s.l.], vol. 4, p. 2348-2351, 1988.

GOMES, J., VELHO, L., GOLDENSTEIN, S. **Wavelets: Teoria, Software e Aplicações**. 21º Colóquio Brasileiro de Matemática. Rio de Janeiro: IMPA, 1997. 216 p.

GRANGER, E., et al. Radar ESM with a What-and-Where Fusion Neural Network. **Proceedings of the 2001 IEEE Signal Processing Society Workshop – Neural Networks for Signal Processing XI, 2001**, North Falmouth, p. 539-548, 10-12 set. 2001.

HOLSCHNEIDER, M. **Wavelets: An Analysis Tool**. Oxford: Clarendon Press, 1995. 423 p.

KINGSLEY, S., QUEGAN, S. **Understanding Radar Systems**. Maidenhead: McGraw-Hill Book Company, 1992. p. 1-60.

KOVÁCS, Z. L. **Redes Neurais Artificiais: Fundamentos e Aplicações**. São Paulo: [s.n], 1996. p. 1-41, 55-76.

MAHAFZA, B. R. **Radar Systems Analysis and Design Using MATLAB**. Huntsville: CRC Press, 2000. p. 1-21, 117-134.

MARDIA, H. K. New Techniques for the Deinterleaving of Repetitive Sequences. **IEE Proceedings for Radar and Signal Processing**, [s.l.], vol. 136 (4), p. 149-154, ago. 1989.

MILOJEVIĆ, D. J., POPOVIĆ, B. M. Improved Algorithm for the Deinterleaving of Radar Pulses. **IEE Proceedings for Radar and Signal Processing**, [s.l.], vol. 139, p. 98-104, fev. 1992.

MOLINA, A. L. B. M. **Uma Investigação do Sistema de Software do Receptor de Alerta Radar RWR ELT-156X**. São José dos Campos, 2002. 170 p. Trabalho de Graduação – Instituto de Tecnológico de Aeronáutica. Confidencial.

NASCIMENTO, F. A. O. **Projeto de Filtros Digitais**. Brasília: Universidade de Brasília, 2002. 30 p. (Notas de aula).

NERI, F. **Introduction to Electronic Defense Systems**. 2nd ed. Boston: Artech House, 2001. 624 p.

NOEL S., SZU, H. Doppler Frequency Estimation with Wavelets and Neural Networks. **Proceedings of the 12th Annual International Symposium on Aerospace/Defense Sensing, Simulation and Controls**, Orlando, Florida, abr. 1998.

NOEL, S., SZU, H. **Wavelets and Neural Networks for Radar**. Disponível em: <<http://ise.gmu.edu/~snoel/publications.htm>>. Acesso em: 28 jul. 2006.

NOONE, G. Radar Pulse Train Parameter Estimation and Tracking using Neural Networks. **Proceedings of International Conference on Artificial Neural Networks and Expert Systems**, New Zealand, p. 95-98, nov. 1995

OPPENHEIN, A. V., SCHAFER, R. W. **Discrete-Time Signal Processing**. New Jersey: Prentice-Hall, 1989. p. 1-105, 112-114, 444-452.

ORSI, R. J., Moore, J. B., MAHONY, R. E. Interleaved pulse train spectrum estimation. **Proceedings of The Fourth International Symposium on Signal Processing and its Applications**, [s.l.], vol. 1, p. 125-128, 1996.

PERKINS, J., COAT, I. Pulse train deinterleaving via the Hough transform. **IEEE International Conference on Acoustics, Speech, and Signal Processing**, Adelaide, vol. 3, 197-200, abr. 1994.

ROCHA, A. F. **Filtros FIR**. Brasília: Universidade de Brasília, [200-]. Disponível em <<http://www.ene.unb.br/~adson/FiltrosFIR.doc>>. Acesso em: 15 out. 2006.

ROCHA, A. F. **Filtros FIR**. Brasília: Universidade de Brasília, [200-]. Disponível em <<http://www.ene.unb.br/~adson/FiltrosDigitais.doc>>. Acesso em: 15 out. 2006.

SCHLEHER, D. C. **Electronic Warfare in the Information Age**. Boston: Artech House, 1999. p. 1-24, 333-396.

STAELIN, D. H. Fast folding algorithm for detection of periodic pulse trains. **Proceedings of the Second Workshop on Signal Processing Applications**, Brisbane, vol. 57, p. 724-725, abr. 1969.

SKOLNIK, M. **Introduction to Radar Systems**. Boston: McGraw-Hill, 1980. p.1-12, 68-95.

TSUI, J. **Digital Techniques for Wideband Receivers**. Boston: Artech House, 1995. p. 9-20, 155-172.

TSUI, J. B. Y. *et al.* Spectral Properties of Radar Time-of-Arrival Periodicities. US Patent 5,396,250, 1995.

TSUI, J. **Microwave Receivers with Electronic Warfare Applications**. Malabar: Krieger Publishing Company, 1992. p. 1-9, 81-111.

Wikipédia, a enciclopédia livre. Rede Neural. Disponível em: <http://pt.wikipedia.org/wiki/Redes_neurais>. Acesso em: 30 jun. 2006

Wikipédia, a enciclopédia livre. DSP. Disponível em: <<http://pt.wikipedia.org/wiki/DSP>>. Acesso em: 30 jun. 2006.

Yahoo! GeoCities. Redes Neurais Artificiais. Disponível em: <http://br.geocities.com/pablo_dalbem/rn1.htm>. Acesso em: 14 mar. 2004.

ZANANDREA, A., et al. Detecção de sinais de pulsos de radar através de transformada contínua de wavelets. In: XXVIII Congresso Nacional da Matemática Aplicada e Computacional - CNMAC, 2005, São Paulo, 2005. p. 478. Disponível em: <200.231.172.253/cnmac/storal2/ademilson_zanandrea_ST20.pdf>. Acesso em: 20 ago. 2006.

ZANANDREA, A., SAOTOME, O. **Transformada de Wavelets Aplicada a Sinais**. São José dos Campos: Instituto Tecnológico de Aeronáutica, 2002. v. 1, 2, 3. (Notas de aula).

ZHANG, G., et al. Application of Generalized Adaptive Wavelet Neural Network Base don Resemblance Coefficient to Signal Recognition. **Proceedings of the 2004 IEEE International Symposium on Intelligent Control**, Taipei, p. 443-447, set. 2004.

APÊNDICES

A – LISTAGEM DOS PROGRAMAS

Aqui se encontram os principais algoritmos desenvolvidos no MATLAB, necessários para simulações e validações do modelo proposto.

O algoritmo a seguir cria seqüências de pulso radar do tipo simples.

```
% Funcao que gera pulsos de radares do tipo "simples"

function vetor = simples_pw(Ti,Tf,PRI,PW,Ta)
% vetor = sinal gerado a partir dos valores parametrizados
% Ti = fase, Tempo de inicio do sinal
% Tf = Tempo final da ocorrencia do sinal
% PRI = Intervalo de Repeticao de Pulso para sinal Simples
% PW = Largura do Pulso
% Ta = Perodo de amostragem

PRI_round = round(PRI*10.00)/10.00; % arredonda para uma casa decimal o PRI
PW_round = round(PW*10.00)/10.00; % arredonda PW
Desloc_PRI = PRI_round/Ta; % Deslocamento provocado por PRI
Desloc_PW = PW_round/Ta; % Deslocamento provocado por PW
Ti_round = round(Ti*10.00)/10.00; % arredonda para uma casa decimal
%o valor do Ti
Ti_pos = Ti_round/Ta + 1; % Posicao no vetor do instante de tempo
%Ti (inicio do sinal)
Tf_round = round(Tf*10.00)/10.00; % arredonda para uma casa decimal
%o valor do tempo final
Tf_pos = Tf_round/Ta + 1; % Localizacao no vetor do instante de tempo Tf

n=1;
while (n < Tf_pos);
    if (n <= Ti_pos) % Enquanto sinal nao inicia,
        vetor(1:Ti_pos) = 0; % ate esse instante nao ha sinal
        vetor(Ti_pos:Ti_pos+Desloc_PW) = 1; % instantes em que ha pulso
        n = Ti_pos + 1;
    elseif ((n + Desloc_PRI - 1) <= Tf_pos)
        vetor(n + Desloc_PW:n+Desloc_PRI - 2)=0; % entre os pulsos nao
        %ha sinal
        n = n + Desloc_PRI; % atualiza o valor do indice do vetor
```

```

if ((n - 1 + Desloc_PW) <= Tf_pos) % Verifica se a largura de
    %pulso ainda cabe
    vetor(n-1:n-1+Desloc_PW) = 1; % instante em que ha pulso
else % Se PW nao coube
    vetor(n-1:Tf_pos) = 1; % instante em que ha pulso
    n = Tf_pos;
end;
else
    vetor(n+Desloc_PW:Tf_pos) = 0;
    n = Tf_pos;
end;
end;

```

O próximo algoritmo gera seqüências de pulsos do tipo *stagger*.

```

% Funcao que gera pulsos de radares do tipo "stagger"

```

```

function vetor = stagger_pw(Ti,Tf,vet_PRI,PW,Ta)
% Ti = fase, Tempo de inicio do sinal
% Tf = Tempo final da ocorrencia do sinal
% vet_PRI = valores de PRI stagger
% PW = Largura do Pulso
% Ta = Periodo de amostragem

vet_PRI_round = round(vet_PRI*10.00)/10.00; % arredonda para uma
%casa decimal os valores de PRI
PW_round = round(PW*10.00)/10.00; % arredonda PW
Desloc_vet_PRI = vet_PRI_round/Ta; % Deslocamento provocado pelos PRI
Desloc_PW = PW_round/Ta; % Deslocamento provocado por PW
Ti_round = round(Ti*10.00)/10.00; % arredonda para uma casa decimal o
%valor do Ti
Ti_pos = Ti_round/Ta + 1; % Posicao em termos de vetor do instante de %
tempo Ti
Tf_round = round(Tf*10.00)/10.00; % arredonda para uma casa decimal o %
valor do tempo final
Tf_pos = Tf_round/Ta + 1; % Localizacao do Tempo final no vetor

j = length(vet_PRI_round);
i = j;

```

```

n=1;

while (n < Tf_pos);
    if (n <= Ti_pos) % Enquanto sinal nao inicia,
        vetor(1:Ti_pos) = 0; % ate esse instante nao ha sinal
        vetor(Ti_pos:Ti_pos+Desloc_PW) = 1; % instantes em que ha pulso
        n = Ti_pos + 1;
    elseif ((n + Desloc_vet_PRI(mod(i,j)+1) - 1) <= Tf_pos)
        % entre os pulsos nao ha sinal
        vetor(n + Desloc_PW:n+Desloc_vet_PRI(mod(i,j)+1) - 2)=0;
        % atualiza o valor do indice do vetor
        n = n + Desloc_vet_PRI(mod(i,j)+1);
        % Verifica se a largura de pulso ainda cabe
        if ((n - 1 + Desloc_PW) <= Tf_pos)
            vetor(n-1:n-1+Desloc_PW) = 1; % instantes em que ha pulso
        else % Se PW nao coube
            vetor(n-1:Tf_pos) = 1; % instantes em que ha pulso
            n = Tf_pos;
        end;
        i = i+1; % atualiza indice do vetor stagger
    else
        vetor(n+Desloc_PW:Tf_pos) = 0;
        n = Tf_pos;
    end;
end;

```

O algoritmo a seguir gera seqüências de pulsos do tipo *wobulated*.

```
% Funcao que gera pulsos de radares do tipo "wobulated"
```

```

function vetor = wobulated(Ti,Tf,PRI,PW,var,ppc,Ta)
% Ti = fase, Tempo de inicio do sinal
% Tf = Tempo final da ocorrencia do sinal
% PRI = Valor Central de Intervalo de Repeticao de Pulso
% PW = Largura do Pulso
% var = valor percentual de variacao sobre o PRI central
% ppc = pulsos por ciclo
% Ta = Período de amostragem

```

```

Rad_inc = 2*pi/ppc; % fator de incremento em radianos
vet_PRI = PRI;
for n = pi/2+Rad_inc:Rad_inc:5*pi/2-Rad_inc % 360 graus para
    %ciclo completo
    vet_PRI = [vet_PRI (PRI + var*PRI*cos(n))]; % cria o vetor
    %de PRIs a serem usados, como stagger
end;

vet_PRI_round = round(vet_PRI*10.00)/10.00; % arredonda para uma
%casa decimal os valores de PRI
PW_round = round(PW*10.00)/10.00; % arredonda PW
Desloc_vet_PRI = vet_PRI_round/Ta; % Deslocamento provocado pelos PRI
Desloc_PW = PW_round/Ta; % Deslocamento provocado por PW
Ti_round = round(Ti*10.00)/10.00; % arredonda para uma casa decimal o
%valor da Ti
Ti_pos = Ti_round/Ta + 1; % Posicao em termos de vetor do instante de
%tempo Ti
Tf_round = round(Tf*10.00)/10.00; % arredonda para uma casa decimal o
%valor do tempo final
Tf_pos = Tf_round/Ta + 1; % Localizacao do Tempo final no vetor

j = length(vet_PRI_round);
i = j;
n=1;

while (n < Tf_pos);
    if (n <= Ti_pos) % Enquanto sinal nao inicia,
        vetor(1:Ti_pos) = 0; % ate esse instante nao ha sinal
        vetor(Ti_pos:Ti_pos+Desloc_PW) = 1; % instantes em que ha pulso
        n = Ti_pos + 1;
    elseif ((n + Desloc_vet_PRI(mod(i,j)+1) - 1) <= Tf_pos)
        vetor(n + Desloc_PW:n+Desloc_vet_PRI(mod(i,j)+1) - 2)=0; % entre
        %os pulsos nao ha sinal
        n = n + Desloc_vet_PRI(mod(i,j)+1); % atualiza o valor do indice
        %do vetor
    if ((n - 1 + Desloc_PW) <= Tf_pos) % Verifica se a largura de
        %pulso ainda cabe
        vetor(n-1:n-1+Desloc_PW) = 1; % instantes em que ha pulso
    else % Se PW nao coube
        vetor(n-1:Tf_pos) = 1; % instantes em que ha pulso
    end
end

```

```

        n = Tf_pos;
    end;
    i = i+1; % atualiza indice do vetor stagger
else
    vetor(n+Desloc_PW:Tf_pos) = 0;
    n = Tf_pos;
end;
end;
end;

```

O próximo algoritmo gera seqüências de pulsos radar do tipo *jitter*.

```

% Funcao que gera pulsos de radares do tipo "jitter"

function vetor = jitter_pw(Ti,Tf,PRI,PW,var,Ta)
% Ti = fase, Tempo de inicio do sinal
% Tf = Tempo final da ocorrencia do sinal
% PRI = Valor Central de Intervalo de Repeticao de Pulso
% PW = Largura do Pulso
% var = valor percentual de variacao sobre o PRI central
% Ta = Periodo de amostragem

PW_round = round(PW*10.00)/10.00; % arredonda PW
Desloc_PW = PW_round/Ta; % Deslocamento provocado por PW
Ti_round = round(Ti*10.00)/10.00; % arredonda para uma casa decimal
%o valor da Ti
Ti_pos = Ti_round/Ta + 1; % Posicao em termos de vetor do instante
%de tempo Ti
Tf_round = round(Tf*10.00)/10.00; % arredonda para uma casa decimal
%o valor do tempo final
Tf_pos = Tf_round/Ta + 1; % Localizacao do Tempo final no vetor

n=1;

while (n < Tf_pos);
    if (n <= Ti_pos) % Enquanto sinal nao inicia,
        vetor(1:Ti_pos) = 0; % ate esse instante nao ha sinal
        vetor(Ti_pos:Ti_pos+Desloc_PW) = 1; % instantes em que ha pulso
        n = Ti_pos + 1;
    else

```

```

delta = (round(PRI*(1+var*2*(rand-0.5))*10.00)/10.00)/Ta;
if ((n + delta - 1) <= Tf_pos)
    vetor(n + Desloc_PW:n + delta - 2)=0; % entre os pulsos nao
    %ha sinal
    n = n + delta; % atualiza o valor do indice do vetor
    if ((n - 1 + Desloc_PW) <= Tf_pos) % Verifica se a largura
        %de pulso ainda cabe
        vetor(n-1:n-1+Desloc_PW) = 1; % instantes em que ha pulso
    else % Se PW nao coube
        vetor(n-1:Tf_pos) = 1; % instantes em que ha pulso
        n = Tf_pos;
    end;
else
    vetor(n+Desloc_PW:Tf_pos) = 0;
    n = Tf_pos;
end;
end;
end;
end;

```

O próximo algoritmo realiza a detecção das seqüências de pulsos radar por meio da transformada de *wavelet* contínua.

```

function powxc_out = extract(x,PRI,dt,Tx,M,pf)
% x = sinal a ser estudado; dt = intervalo de incremento
% PRI = Range de valores de PRI a serem testados; M = Parametro de Ajuste
% Tx = Taxa de amostragem do sinal original;
% pf = Porcentagem de Threshold a ser utilizado

[R,N] = size(x);

meio = round(N/2);

t=(-N/2):(N/2-1)*Tx;
tt=(-t); % jah inverte na funcao psi para fazer a convolucao ou fft

for ind=1:R
    for a=1:length(PRI)

```

```

%Wavelet de Driscoll
    psi = ((PRI(a)*M)^-0.5) * 1 * (abs(tt/(PRI(a)*M)) <= 0.5) .* exp(2*pi*i*(tt/PRI(a)));
%Wavelet de Morlet
%    psi = ( ( (M*1)^2*pi )^(-1/4) ) .* exp( -( tt/PRI(a) )^2 / (2*(M^2)) ) .* exp(2*pi*i*(tt/PRI(a)));
    twx = ifft( fft([conj(psi) zeros(1, length(x))]) .* fft([x(ind,:) zeros(1,length(psi)-1)]));
    powxc(a,:) = (abs(twx).^2)/M;
end;
%subplot(1,2,1),plot_horizontal(powxc,dt,PRI);

%gws=mean(powxc');
%THRES=((max(gws)-min(gws))/2)+min(gws)+std(gws)

maximo=max(max(powxc))
powxc = powxc/maximo;
THRES = pf * 1;

for ind1=1:length(powxc(1,:))
    for ind2=1:length(powxc(:,1))
        if powxc(ind2,ind1) < THRES
            powxc(ind2,ind1) = 0.0;
        end;
    end;
end;

figure;
pcolor((1:N)*Tx,PRI,powxc(:,meio:meio+N-1)), shading interp, colorbar % imprime N elementos
%pcolor((1:length(powxc)),PRI,powxc), shading interp, colorbar % imprime 2N-1 elementos

xlabel('Tempo');
ylabel('Valor de PRI');
mess = sprintf('Valor Limiar de D(T,t) > %f, THRES);
title(mess);
end;
powxc_out = powxc(:,meio:meio+N-1);

```

O algoritmo a seguir cria as redes neurais e as treina com os conjuntos passados como parâmetros.


```

function net = cria_net(D,T,neurons)

%net = newff(PR,[S1 S2...SNI],{TF1 TF2...TFNI},BTF,BLF,PF)
%PR - Rx2 matrix of min and max values for R input elements.
%Si - Size of ith layer, for NI layers.
%TFi - Transfer function of ith layer, default = 'tansig'.
%BTF - Backprop network training function, default = 'trainlm'.
%BLF - Backprop weight/bias learning function, default = 'learngdm'.
%PF - Performance function, default = 'mse'.
%and returns an N layer feed-forward backprop network.

transfer = {'logsig' 'logsig'};

BTF = 'TRAINRP';
BLF = 'learngdm';
PF = 'mse';

%Limites dos valores de entrada da rede para as deteccoes normalizadas
PR = [0 1];
for i=1:90
    PR = [PR; 0 1];
end

net=newff(PR,neurons,transfer,BTF,BLF,PF);
net.trainParam.goal = 0.001;
net.trainParam.epochs = 180;
net = train(net,D,T);

```

O algoritmo seguinte cria os conjuntos de sinais e saídas para treinamento das redes MLP.

```

% Cria os sinais para treinamento das redes
clear all;
clear all;

```

Ta = 0.1; % Periodo de Amostragem escolhido simulando escala de microssegundos

%Radares com PRI entre 1 e 10000 us

```

%Radar A
A = simples_pw(0,150,8,0.8,Ta); % Inicio em t=10us e termino em t=100us, PRI=5us
La = length(A);
T_A = saida_neural(7.5,157.5,Ta); % Saida neural desejada (TARGET)

%Radar B
B = stagger_pw(0,1500,[58 60 61 63],6,Ta);
Lb = length(B);
T_B = saida_neural(75,1575,Ta);

%Radar C
C = wobullated_pw(0,15000,400,40,0.1,20,Ta);
Lc = length(C);
T_C = saida_neural(750,15750,Ta);

%Radar D
D = jitter_pw(0,150000,2000,100,0.05,Ta);
Ld = length(D);
T_D = saida_neural(7500,157500,Ta);

Lm = max([La Lb Lc Ld]);

A = [A zeros(1,Lm-La)];
T_A = [T_A zeros(1,Lm-La)];
B = [B zeros(1,Lm-Lb)];
T_B = [T_B zeros(1,Lm-Lb)];
C = [C zeros(1,Lm-Lc)];
T_C = [T_C zeros(1,Lm-Lc)];
D = [D zeros(1,Lm-Ld)];
T_D = [T_D zeros(1,Lm-Ld)];

ABCD = interleaving([A;B;C;D]);

save('train_4radares.mat','ABCD','A','B','C','D','T_A','T_B','T_C','T_D');

```

O próximo algoritmo cria os sinais para validação das redes MLP.

% Armazena 8 radares para validação das redes treinadas

```
clear all;
```

```
clear all;
```

```
Ta = 0.1; % Periodo de Amostragem escolhido simulando escala de microssegundos
```

```
%Radares com PRI entre 1 e 10000 us
```

```
%Radar A
```

```
A = wobullated_pw(30,120,4,0.4,0.05,25,Ta);
```

```
La = length(A);
```

```
%Radar B *Treinado
```

```
B = simples_pw(60,200,8,0.8,Ta); % Inicio em t=10us e termino em t=100us, PRI=5us
```

```
Lb = length(B);
```

```
%Radar C
```

```
C = simples_pw(70,300,20,2,Ta); % Inicio em t=10us e termino em t=100us, PRI=5us
```

```
Lc = length(C);
```

```
%Radar D *Treinado
```

```
D = stagger_pw(50,2000,[58 60 61 63],6,Ta);
```

```
Ld = length(D);
```

```
%Radar E
```

```
E = stagger_pw(80,14000,[700 702 706 707],70,Ta);
```

```
Le = length(E);
```

```
%Radar F *Treinado
```

```
F = wobullated_pw(140,17000,400,40,0.1,20,Ta);
```

```
Lf = length(F);
```

```
%Radar G
```

```
G = jitter_pw(100,140000,5000,200,0.05,Ta);
```

```
Lg = length(G);
```

```
%Radar H *Treinado
```

```
H = jitter_pw(220,160000,2000,100,0.05,Ta);
```

```
Lh = length(H);
```

```
Lm = max([La Lb Lc Ld Le Lf Lg Lh]);
```

```
A = [A zeros(1,Lm-La)];
```

```
B = [B zeros(1,Lm-Lb)];
```

```
C = [C zeros(1,Lm-Lc)];
```

```
D = [D zeros(1,Lm-Ld)];
```

```
E = [E zeros(1,Lm-Le)];
```

```
F = [F zeros(1,Lm-Lf)];
```

```
G = [G zeros(1,Lm-Lg)];
```

```
H = [H zeros(1,Lm-Lh)];
```

```
Sv8 = interleaving([A;B;C;D;E;F;G;H]);
```

```
save('valida_8radares.mat','Sv8','A','B','C','D','E','F','G','H');
```

O algoritmo seguinte realiza o treinamento das redes MLP com os sinais criados para esse fim.

```
% realiza o treinamento das redes MLP
```

```
clear all;
```

```
clear all;
```

```
%Para PRI entre 1 a 10 us
```

```
load train_4radares.mat
```

```
% Quantidade de Janelas de 2048 ptos do Buffer a serem analisadas
```

```
n=1;
```

```
% coeficientes para filtrar sinal com PRI qualquer entre w=0.02 e 0.2
```

```
h = fir1(150,[0.02 0.2]);%blackman(151));
```

```
%Subamostragens
```

```
% entre 10 e 100 us
```

```
B1 = decimate(B,10,'fir');
```

```
T_B1 = downsample(T_B,10); %saida da rede
```

```

% entre 100 e 1000 us
C1 = decimate(C,10,'fir');
C2 = decimate(C1,10,'fir');
T_C1 = downsample(T_C,10);
T_C2 = downsample(T_C1,10); %saida da rede

% entre 1000 e 10000 us
D1 = decimate(D,10,'fir');
D2 = decimate(D1,10,'fir');
D3 = decimate(D2,10,'fir');
T_D1 = downsample(T_D,10);
T_D2 = downsample(T_D1,10);
T_D3 = downsample(T_D2,10); %saida da rede

%todos os sinais juntos
ABCD1 = decimate(ABCD,10,'fir');
ABCD2 = decimate(ABCD1,10,'fir');
ABCD3 = decimate(ABCD2,10,'fir');

%para treino com o radar A (PRI entre 1 e 10 us)
A_1_10 = buffer(A,2048);
t_a = buffer(T_A,2048);
a_1_10 = filter(h,1,A_1_10);
sinal_detectado_A = extract(a_1_10(:,1:n)',[1:0.1:10],0.1,0.1,12,0);

%para treino com o radar B (PRI entre 10 e 100 us)
B_10_100 = buffer(B1,2048);
t_b = buffer(T_B1,2048);
b_10_100 = filter(h,1,B_10_100);
sinal_detectado_B = extract(b_10_100(:,1:n)',[10:1:100],1,1,12,0);

%treino com o radar C (PRI entre 100 e 1000 us)
C_100_1000 = buffer(C2,2048);
t_c= buffer(T_C2,2048);
c_100_1000 = filter(h,1,C_100_1000);
sinal_detectado_C = extract(c_100_1000(:,1:n)',[100:10:1000],10,10,12,0);

%treino com o radar D (PRI entre 1000 e 10000 us)

```

```

D_1000_10000 = buffer(D3,2048);
t_d = buffer(T_D3,2048);
d_1000_10000 = filter(h,1,D_1000_10000);
sinal_detectado_D = extract(d_1000_10000(:,1:n)',[1000:100:10000],100,100,12,0);

%treino com todos os sinais
%entre 1 e 10 us
ABCD_1_10 = buffer(ABCD,2048);
abcd_1_10 = filter(h,1,ABCD_1_10);
sinal_detectadoABCD_1_10 = extract(abcd_1_10(:,1:n)',[1:0.1:10],0.1,0.1,12,0);

%entre 10 e 100 us
ABCD_10_100 = buffer(ABCD1,2048);
abcd_10_100 = filter(h,1,ABCD_10_100);
sinal_detectadoABCD_10_100 = extract(abcd_10_100(:,1:n)',[10:1:100],1,1,12,0);

%entre 100 e 1000 us
ABCD_100_1000 = buffer(ABCD2,2048);
abcd_100_1000 = filter(h,1,ABCD_100_1000);
sinal_detectadoABCD_100_1000 = extract(abcd_100_1000(:,1:n)',[100:10:1000],10,10,12,0);

%entre 1000 e 10000 us
ABCD_1000_10000 = buffer(ABCD3,2048);
abcd_1000_10000 = filter(h,1,ABCD_1000_10000);
sinal_detectadoABCD_1000_10000 =
extract(abcd_1000_10000(:,1:n)',[1000:100:10000],100,100,12,0);

% criacao e treinamento das redes

net_1_10 = cria_net([sinal_detectado_A sinal_detectadoABCD_1_10], ...
    [t_a(:,1:n)' t_a(:,1:n)],[80 1]);
%pause;
net_10_100 = cria_net([sinal_detectado_B sinal_detectadoABCD_10_100], ...
    [t_b(:,1:n)' t_b(:,1:n)],[80 1]);
%pause;
net_100_1000 = cria_net([sinal_detectado_C sinal_detectadoABCD_100_1000], ...
    [t_c(:,1:n)' t_c(:,1:n)],[80 1]);
%pause;
net_1000_10000 = cria_net([sinal_detectado_D sinal_detectadoABCD_1000_10000], ...
    [t_d(:,1:n)' t_d(:,1:n)],[80 1]);

```

```
save('redes_4radares.mat','net_1_10','net_10_100','net_100_1000','net_1000_10000');
```

O algoritmo a seguir faz a validação das redes MLP treinadas.

```
%Realiza a validação das redes MLP
clear all;
clear all;

%Para PRI entre 1 a 10 us
load valida_8radares.mat
load redes_4radares.mat

% Quantidade de Janelas de 2048 ptos do Buffer a serem analisadas
n=1;

% coeficientes para filtrar sinal com PRI qualquer entre w=0.02 e 0.2
h = fir1(150,[0.02 0.2]);%blackman(151));

% Subamostragens
Sinal_10_100 = decimate(Sv8,10,'fir');
Sinal_100_1000 = decimate(Sinal_10_100,10,'fir');
Sinal_1000_10000 = decimate(Sinal_100_1000,10,'fir');

% Para analise entre 1 e 10 us
Y_1_10 = buffer(Sv8,2048);
y_1_10 = filter(h,1,Y_1_10);

% Bufferização e Filtragem

% Para analise entre 10 e 100 us
Y_10_100 = buffer(Sinal_10_100,2048);
y_10_100 = filter(h,1,Y_10_100);

% Para analise entre 100 e 1000 us
Y_100_1000 = buffer(Sinal_100_1000,2048);
y_100_1000 = filter(h,1,Y_100_1000);

% Para analise entre 1000 e 10000 us
```

```

Y_1000_10000 = buffer(Sinal_1000_10000,2048);
y_1000_10000 = filter(h,1,Y_1000_10000);

% Detecção das Seqüências de Pulsos
sinal_detectado_1_10 = extract(y_1_10(:,1:n)',[1:0.1:10],0.1,0.1,12,0);
sinal_detectado_10_100 = extract(y_10_100(:,1:n)',[10:1:100],1,1,12,0);
sinal_detectado_100_1000 = extract(y_100_1000(:,1:n)',[100:10:1000],10,10,12,0);
sinal_detectado_1000_10000 = extract(y_1000_10000(:,1:n)',[1000:100:10000],100,100,12,0);

saida_1_10 = sim(net_1_10,sinal_detectado_1_10);
saida_10_100 = sim(net_10_100,sinal_detectado_10_100);
saida_100_1000 = sim(net_100_1000,sinal_detectado_100_1000);
saida_1000_10000 = sim(net_1000_10000,sinal_detectado_1000_10000);

subplot(2,4,1), extract(y_1_10(:,1:n)',[1:0.1:10],0.1,0.1,12,0);
subplot(2,4,2), extract(y_10_100(:,1:n)',[10:1:100],1,1,12,0);
subplot(2,4,3), extract(y_100_1000(:,1:n)',[100:10:1000],10,10,12,0);
subplot(2,4,4), extract(y_1000_10000(:,1:n)',[1000:100:10000],100,100,12,0);
subplot(2,4,5), plot(0:0.1:204.7,saida_1_10);
subplot(2,4,6), plot(0:1:2047,saida_10_100);
subplot(2,4,7), plot(0:10:20470,saida_100_1000);
subplot(2,4,8), plot(0:100:204700,saida_1000_10000);

```

O algoritmo seguinte realiza a superposição (*interleaving*) das seqüências de pulsos radar.

```

%interleaving.m
function vet = interleaving(vetores)
% Matriz contendo os vetores dos sinais na forma [A; B; C; ...]

[m,n] = size(vetores); % tamanho da matriz dos sinais
vet = [zeros(1,n)]; % inicia variavel dos sinais

for i=1:m
    vet = vet + vetores(i,:); % realiza a soma (interleaving) dos sinais
end;

%vet = or(vet,vet); % limita a 1 os valores onde houve sobreposição

```