



AMADEU SOARES FERLIN

**Monitoramento do processo de junção por
Friction Stir Welding para detecção de falhas**

80/2013

CAMPINAS
2013



UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA

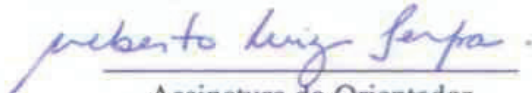
AMADEU SOARES FERLIN

**Monitoramento do processo de junção por
Friction Stir Welding para detecção de falhas**

Orientador: Prof. Dr. Alberto Luiz Serpa

Dissertação de Mestrado apresentada à Faculdade de Engenharia Mecânica da Universidade Estadual de Campinas para a obtenção do título de Mestre em Engenharia Mecânica, na Área de Mecânica dos Sólidos e Projeto Mecânico.

ESTE EXEMPLAR CORRESPONDE À VERSÃO
FINAL DA DISSERTAÇÃO DEFENDIDA PELO
ALUNO AMADEU SOARES FERLIN, E ORIEN-
TADO PELO PROF. DR. ALBERTO LUIZ SERPA.


Assinatura do Orientador

CAMPINAS
2013

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Rose Meire da Silva - CRB 8/5974

F384m Ferlin, Amadeu Soares, 1986-
Monitoramento do processo de junção por Friction Stir Welding para detecção de falhas / Amadeu Soares Ferlin. – Campinas, SP : [s.n.], 2013.

Orientador: Alberto Luiz Serpa.
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Monitoramento. 2. Soldadura por fricção. 3. Fourier, Transformada de. 4. Wavelets. I. Serpa, Alberto Luiz, 1967-. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Friction stir welding monitoring for failure detection

Palavras-chave em inglês:

Monitoring
Friction Stir Welding
Fourier Transform
Wavelet

Área de concentração: Mecânica dos Sólidos e Projeto Mecânico

Titulação: Mestre em Engenharia Mecânica

Banca examinadora:

Alberto Luiz Serpa [Orientador]
Janito Vaqueiro Ferreira
Vicente Lopes Júnior

Data de defesa: 29-07-2013

Programa de Pós-Graduação: Engenharia Mecânica

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE MECÂNICA COMPUTACIONAL

DISSERTAÇÃO DE MESTRADO ACADÊMICO

**Monitoramento do processo de junção por
Friction Stir Welding para detecção de falhas**

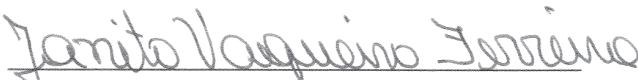
Autor: Amadeu Soares Ferlin

Orientador: Prof. Dr. Alberto Luiz Serpa


A Banca Examinadora composta pelos membros abaixo aprovou esta Dissertação:



Prof. Dr. Alberto Luiz Serpa, Presidente
DMC/FEM/Unicamp



Prof. Dr. Janito Vaqueiro Ferreira
DMC/FEM/Unicamp



Prof. Dr. Vicente Lopes Júnior
DEM/FEIS/UNESP

Campinas, 29 de julho de 2013

AGRADECIMENTOS

Ao Prof. Dr. Alberto Luiz Serpa, meu orientador, pela confiança e auxílio desde o início dos trabalhos, em tudo que foi necessário para a finalização desta etapa.

Ao Prof. Dr. Renato Pavanello e ao Prof. Dr. Janito Vaqueiro Ferreira que, participando das reuniões do grupo FSW da Unicamp, ajudaram a sanar diversas dúvidas e apoiaram os processos de pesquisa e experimentais.

Ao programa de pós graduação em Engenharia Mecânica da Universidade Estadual de Campinas.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo auxílio financeiro durante o primeiro ano de pesquisas no mestrado.

À Empresa Brasileira de Aeronáutica (Embraer), pelo auxílio financeiro, reuniões, *workshops*, visitas e pela oportunidade da área de pesquisa.

Aos membros da banca, pela disponibilidade.

Aos funcionários do Departamento de Mecânica Computacional, em especial à Beth e ao Geraldo, que tanto me auxiliaram no dia a dia.

Aos parceiros de laboratório Fuga, Daniel, Alysson, Kellen, Mariane, Ana, Ruth, Camilo, Luan e Alan.

Aos amigos de longa data Labaki, Emílio, Itamar, Renato, Renan, Sunti, Danilo Pires, Gustavo e Leandro, que fizeram de Campinas praticamente uma Ilha Solteira.

Aos amigos de curta data, mas não por isso menos amigos, Wendell, Piveta, Dani, André, Cris, Marcela, Priscilla, Danilo, Miguel e Manuel.

Aos meus pais e às minhas irmãs, por todo o apoio, incentivo e paciência durante todos esses

anos de estudo.

À minha namorada, Eliane, que compartilhou cada momento comigo, dos piores aos melhores; assim como toda sua família, que sempre confiou em mim.

Por fim, a todas as pessoas que conheci, pois sem elas a história seria totalmente diferente.

RESUMO

Esta dissertação discute aspectos do monitoramento do processo de soldagem por *Friction Stir Welding* (FSW) de chapas de alumínio utilizando acelerômetros fixados à superfície das chapas. Os sinais adquiridos durante o processo de soldagem foram processados utilizando dois métodos diferentes - a Transformada de Fourier de Tempo Curto (STFT) e a Transformada *Wavelet* - que foram capazes de detectar e localizar falhas criadas através da remoção de material nas chapas. A aplicação destas técnicas nos sinais adquiridos durante o processo de soldagem permitem a identificação de mudanças no comportamento destes sinais através de resultados gráficos contendo informações de frequência e de tempo. Através destas informações é possível determinar o momento em que as mudanças de comportamento ocorreram e, conseqüentemente, prever as posições correspondentes à ocorrência da falha no respectivo cordão da solda.

Palavras-Chave: Monitoramento; Soldagem por Fricção; Transformada de Fourier de Tempo Curto; Transformada *Wavelet*.

ABSTRACT

This dissertation discusses aspects of the Friction Stir Welding (FSW) process monitoring of aluminum plates using accelerometers attached to the plates surface. The signals acquired during the welding process were processed using two different methods - Short Time Fourier Transform (STFT) and Wavelet Transform - that were able to detect and locate failures created by removing material of the plates. The application of these techniques on the signals acquired during the welding process allows the identification of changes in the signals pattern by graphical results containing frequency and time information. Thus it is possible to determine the moment at which the changes in the behavior occurred and therefore predict the positions corresponding to the occurrence of the failure in the weld line.

Keywords: Monitoring; Friction Stir Welding; Short Time Fourier Transform; Wavelet Transform.

LISTA DE FIGURAS

1.1	Painéis produzidos pela Sapa (Sapa (2012a)).	2
1.2	Processo FSW (The Research Laboratory of Civil Engineering and Mechanics of Ecole Centrale de Nantes (2011), modificada).	3
1.3	Ferramentas do processo FSW (Capelari e Mazzaferro (2009), modificada).	3
1.4	Características da junta unida por FSW (James et al. (2004), modificada).	3
1.5	Janelamento: (a) STFT e (b) Transformada <i>Wavelet</i> (Gallegati (2008), modificada).	6
2.1	Experimentos: (a) peça sólida, (b) chapas encostadas e (c) chapas com espaçamento conhecido (Yang et al. (2006), modificada).	10
2.2	Monitoramento da força axial: peça sólida (Yang et al. (2006), modificada).	10
2.3	Monitoramento da força axial (a) com espaçamento conhecido e (b) com as chapas encostadas (Yang et al. (2006), modificada).	11
2.4	Monitoramento das forças axiais para diferentes tamanhos de falha (Fleming (2009), modificada).	12
2.5	Mapa de contorno para chapa uniformemente soldada (Sinha et al. (2008)).	13
2.6	Mapa de contorno para falta de penetração da ferramenta (Sinha et al. (2008)).	14
2.7	Dimensões das falhas (Chen et al. (2003)).	15
2.8	Processo com as falhas: (a) perfil tridimensional e (b) mapa de contornos da Transformada <i>Wavelet</i> (Chen et al. (2003)).	16
2.9	Mapas de Quadrados Médios: (a) sem falhas e (b) com falhas (Moraes (1996)).	17
2.10	Ensaio de aceleração: (a) domínio do tempo e (b) Mapa de Quadrados Médios (Moraes (1996)).	18
2.11	STFT para ensaio de aceleração (Moraes (1996)).	19
3.1	Sinais formados por (a) soma das senóides e (b) posicionamento das senóides.	23
3.2	FFTs dos sinais da Fig. 3.1.	23
3.3	Janelamento do sinal (Sethares (2012), modificada).	25
3.4	STFTs dos sinais da Fig. 3.1.	26
3.5	FFT e STFT do Sinal (b) da Fig. 3.1 utilizando 3 divisões.	27

3.6	FFT e STFT do Sinal (b) da Fig. 3.1 utilizando 20 divisões.	27
3.7	FFT e STFT do Sinal (b) da Fig. 3.1 utilizando 10 divisões.	28
4.1	Família Daubechies de <i>wavelets</i> (Oliveira (2012), modificada).	29
4.2	Expansão e translação da <i>wavelet</i> -mãe. (Oliveira (2012)).	30
4.3	Onda quadrada de dois ciclos.	30
4.4	Níveis da decomposição da onda quadrada.	31
4.5	<i>Wavelets</i> D4 em nível 3.	32
4.6	Função expandida.	33
4.7	Oito iterações da função escala D4.	34
4.8	Forma final da função escala D4.	35
4.9	<i>Wavelet</i> D4 a partir da função escala da Fig. 4.8.	36
4.10	<i>Wavelet</i> D4 calculada por iteração.	37
4.11	Base para as amplitudes <i>wavelet</i> quadradas.	41
4.12	Mapa de Quadrados Médios de uma senóide ($f = 1024$ Hz).	42
4.13	Mapa de Quadrados Médios de uma senóide ($f = 512$ Hz).	43
4.14	Mapa de Quadrados Médios de uma senóide com amplitudes diferentes ($f = 512$ Hz).	43
5.1	Interface gráfica criada em GUI (Matlab [®]) para análise dos sinais.	45
5.2	Sinais (a) sem falhas e (b) com falhas.	47
5.3	FFTs dos sinais (a) sem falhas e (b) com falhas.	47
5.4	Divisões de um sinal sem <i>overlap</i> e com janela retangular.	48
5.5	Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,1, janela retangular, 0% <i>overlap</i>).	49
5.6	Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,5, janela retangular, 0% <i>overlap</i>).	49
5.7	Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=1,1, janela retangular, 0% <i>overlap</i>).	50
5.8	Divisões do sinal com 50% de <i>overlap</i> e janela Hanning.	51
5.9	Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,1; janela Hanning, 50% <i>overlap</i>).	52

5.10	Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,5; janela Hanning, 50% <i>overlap</i>).	52
5.11	Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=1,0; janela Hanning, 50% <i>overlap</i>).	53
5.12	<i>Zoom</i> da falha em $L/2$	53
5.13	Níveis de decomposição (2 coeficientes) dos sinais (a) sem falhas e (b) com falhas. . .	54
5.14	Mapas de Quadrados Médios (2 coeficientes) dos sinais (a) sem falhas e (b) com falhas.	55
5.15	Níveis de decomposição (8 coeficientes) dos sinais (a) sem falhas e (b) com falhas. . .	55
5.16	Mapas de Quadrados Médios (8 coeficientes) dos sinais (a) sem falhas e (b) com falhas.	56
5.17	Níveis de decomposição (12 coeficientes) dos sinais (a) sem falhas e (b) com falhas. . .	56
5.18	Mapas de Quadrados Médios (12 coeficientes) dos sinais (a) sem falhas e (b) com falhas.	57
5.19	Níveis de decomposição (20 coeficientes) dos sinais (a) sem falhas e (b) com falhas. . .	57
5.20	Mapas de Quadrados Médios (20 coeficientes) dos sinais (a) sem falhas e (b) com falhas.	58
5.21	Superfícies diferentes para atrito do microfone.	59
5.22	Sinal sonoro adquirido.	59
5.23	STFT do sinal sonoro adquirido.	60
5.24	Mapa de Quadrados Médios do sinal sonoro adquirido.	60
6.1	Ferramenta utilizada nos ensaios.	64
6.2	Uma das chapas utilizadas nos ensaios.	64
6.3	Fixação das chapas à mesa da máquina.	65
6.4	Posicionamento dos acelerômetros 1, 2 e 3 nas chapas.	65
6.5	Chapas unidas no ensaio 1.	66
6.6	Falhas feitas nas chapas para o ensaio 2.	67
6.7	Aproximação da ferramenta.	67
6.8	Chapas unidas no ensaio 2.	68
6.9	Sinais adquiridos no ensaio.	69
6.10	Mapas de cores da STFT dos sinais.	70
6.11	Mapas de Quadrados Médios dos sinais.	71

LISTA DE TABELAS

3.1	Tamanho das Janelas (Chan e Zhu (2012), Quatieri (2001b))	26
6.1	Dimensões da ferramenta e das chapas.	64
6.2	Características dos acelerômetros utilizados	65

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Estrutura do Trabalho	7
2	REVISÃO BIBLIOGRÁFICA	9
2.1	Força axial	9
2.2	Processamento de imagens	12
2.3	Sensores de emissão acústica	14
2.4	STFT e Transformada <i>Wavelet</i>	17
3	TRANSFORMADA DE FOURIER DE TEMPO CURTO	21
3.1	Introdução	21
3.2	Transformada de Fourier	21
3.3	Transformada de Fourier de Tempo Curto	23
4	TRANSFORMADA WAVELET	29
4.1	Introdução	29
4.2	Equações de Dilatação	32
4.3	<i>Wavelets</i> de Dilatação	35
4.4	Transformada <i>Wavelet</i> Circular	37
4.5	Transformada <i>Wavelet</i> Discreta	38
4.6	Mapa de Quadrados Médios	40
5	VALIDAÇÃO DAS TÉCNICAS DE DETECÇÃO	45
5.1	Introdução	45
5.2	Simulação da vibração de um sistema rotativo	46
5.2.1	Transformada de Fourier de Tempo Curto (STFT)	47
5.2.2	Transformada <i>Wavelet</i>	54
5.3	Teste com medição de som	59
5.4	Conclusões do capítulo	61

6	RESULTADOS EXPERIMENTAIS	63
6.1	Introdução	63
6.2	Materiais	63
6.3	Testes	66
6.3.1	Ensaio 1: determinação dos parâmetros	66
6.3.2	Ensaio 2: detecção e localização de falhas	67
6.4	Resultados	68
6.4.1	Transformada de Fourier de Tempo Curto (STFT)	70
6.4.2	Transformada <i>Wavelet</i>	71
6.5	Conclusões do capítulo	72
7	CONCLUSÕES	73
	REFERÊNCIAS	75
	ANEXO A - ALGORITMOS	81
A.1	Programa Principal	81
A.2	Carregando o sinal	93
A.3	Adequação do tamanho do sinal	94
A.4	Mapa de Quadrados Médios	94
A.5	Coefficientes <i>wavelet</i>	95
A.6	Normalização	99
A.7	Divisão do sinal	100

1 INTRODUÇÃO

O processo de soldagem por fricção linear, também conhecido como *friction stir welding* (FSW), foi inventado por Wayne Thomas e um time de pesquisadores em 1991, no Reino Unido. *The Welding Institute* (TWI), onde a técnica foi criada, promove pesquisas e consultorias em tecnologia e engenharia para empresas do mundo todo (TWI, 2012a; TWI, 2012b).

Inicialmente utilizado por indústrias automotivas e aeroespaciais, o processo passou de uma simples técnica de soldagem de chapas finas de ligas de alumínio para um processo adaptável de grande utilização na união de ligas de metais leves, titânio, aço e outros metais duros de pequenas espessuras (TWI, 2012b).

Atualmente, além dos setores automotivo e aeroespacial, indústrias de construção naval, ferroviárias e de componentes mecânicos estão investindo no desenvolvimento do processo FSW para suas linhas de produção. Além de diminuir o tempo e os custos de produção, FSW é uma técnica que supera diversos problemas associados às técnicas tradicionais de soldagem (TWI, 2012a; TWI, 2012b).

Algumas vantagens em relação aos processos convencionais são a baixa distorção e as excelentes propriedades mecânicas do material após a junção. Por ser um processo de união no estado sólido, existem menos chances de contaminação do cordão e a microestrutura é bastante fina, aumentando a resistência à tração e à fadiga (SINHA et al., 2008).

A Sapa, indústria especializada em perfis de alumínio, iniciou uma produção em série de painéis unidos através do processo FSW em 1996 (SAPA, 2012a). Atualmente é capaz de produzir painéis de até 26 metros de comprimento por 3,5 metros de largura (SAPA, 2012b). Estes painéis são aplicados principalmente em trens e navios. A Alston LHB utiliza esses painéis pré-fabricados nos trens de Copenhague desde 1997 (KALLEE et al., 2012). A Fig. 1.1 mostra alguns dos painéis produzidos pela Sapa.

Os veículos de lançamento Delta II e Delta IV, da Boeing, também utilizam o processo FSW.



Figura 1.1 - Painéis produzidos pela Sapa (Sapa (2012a)).

O primeiro desses com um módulo interestágios unido pelo processo foi lançado em 1999. Segundo John Tracy, vice-presidente de Engenharia para Sistemas de Defesa Integrados em 2004, a união por FSW diminuiu os custos do processo para estes veículos pela metade (POLT, 2012).

A indústria aeronáutica avalia a viabilidade deste processo na produção da fuselagem dos aviões em substituição ao processo de rebatamento. Além de ser um processo mais rápido, há uma redução de peso na aeronave, significando também uma redução de custos (GENICOLO, 2007).

Pode-se ter uma ideia de como este processo é realizado através do esquema mostrado na Fig. 1.2. Este processo consiste em unir duas chapas utilizando uma ferramenta cilíndrica (Fig. 1.3), composta por um ombro (de diâmetro maior) e um pino (de diâmetro menor). Ao girar, a ferramenta é colocada em contato com a superfície na linha de junção das duas chapas.

De acordo com James et al. (2004), uma vez que a zona a ser soldada atinge um estado termo-mecânico apropriado, a ferramenta é transladada ao longo da linha de junção das chapas e a união é feita pela transferência do material ao redor da ferramenta e sua deposição atrás dela. A Fig. 1.4 mostra uma junta unida pelo processo.

O lado do avanço é definido onde a velocidade de translação ao longo da linha de junção soma-se à velocidade de rotação da ferramenta. O lado do retrocesso é o oposto, onde a velocidade de rotação é subtraída da velocidade de translação (JAMES et al., 2004).

Em peças de alumínio, FSW é um processo onde não há fusão do material e, com isso, não

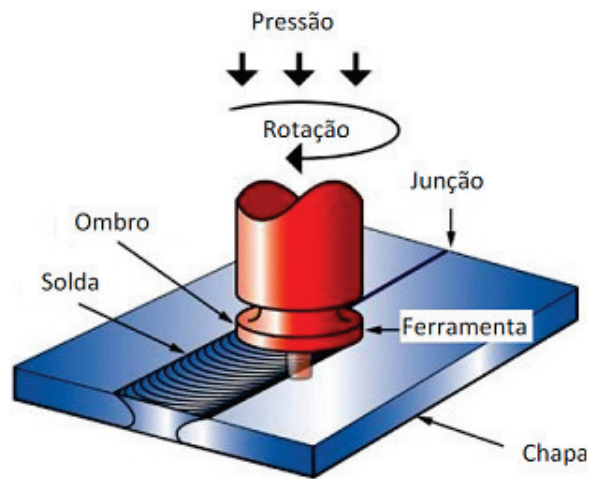


Figura 1.2 - Processo FSW (The Research Laboratory of Civil Engineering and Mechanics of Ecole Centrale de Nantes (2011), modificada).



Figura 1.3 - Ferramentas do processo FSW (Capelari e Mazzaferro (2009), modificada).

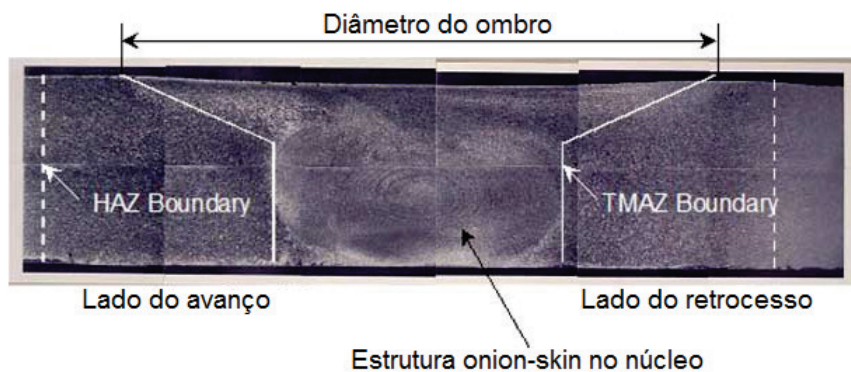


Figura 1.4 - Características da junta unida por FSW (James et al. (2004), modificada).

há geração de arcos, fumos ou faíscas. Segundo ABAL (2011), a ausência de uma fase de fusão nesses processos elimina defeitos como porosidade, bolhas, rebaixos, inclusões e microestruturas indesejáveis, que frequentemente aparecem na solda e nas zonas afetadas pelo calor em processos de soldagem por fusão. Além disso, produz cordões de alta qualidade em materiais de difícil soldagem, como o alumínio, magnésio e cobre (TWI, 2012a).

Contudo, o processo apresenta algumas limitações. Para uma correta junção, o sistema de fixação deve ser muito robusto; o equipamento não é portátil e para cada espessura a se soldar existe uma ferramenta adequada, pois o comprimento do pino muda de acordo com a espessura da chapa.

Apesar do processo não ser livre de defeitos, James et al. (2004) afirma que a quantidade de defeitos em processos de soldagem por atrito geralmente é pequena se comparada à quantidade encontrada em processos por fusão. Algumas dessas falhas são a falta de penetração do pino, a formação de cavidade e a presença de *kissing bonds*.

A falta de penetração ocorre quando o comprimento do pino é consideravelmente menor que a espessura da chapa. A penetração da ferramenta é, geralmente, cerca de 90% desta espessura e pode haver falhas se houver alguma variação neste valor ao longo da linha de junção (JAMES et al., 2004; BIRD; KLEINER, 2012). Já a cavidade, forma-se abaixo da superfície, não contém material e geralmente é alinhada com a direção da junção (BIRD; KLEINER, 2012). Os defeitos na raiz, ou *kissing bonds*, ocorrem quando não há deformação suficiente para a formação de novas ligações químicas e a raiz da junção fica parcialmente unida (OOSTERKAMP et al., 2004).

A existência dessas falhas traz a necessidade de uma forma de monitoramento do processo de junção por FSW. As técnicas de monitoramento evitam que chapas com falhas passem pela análise visual ou outros métodos de inspeção dos cordões sem serem detectadas e, dessa forma, minimizam o descarte de materiais e evitam problemas estruturais.

Pesquisas feitas para aplicar ou melhorar o monitoramento do processo utilizam a aquisição de dados de corrente elétrica e potência das máquinas, forças axial e horizontal da ferramenta, processamento de imagens, sinais de emissão acústica e vibrações como principais técnicas.

Alguns trabalhos mostram que há variação da corrente elétrica e da potência da máquina quando a ferramenta passa por uma falha na chapa. Sinha et al. (2008) calcularam a potência de saída do motor de uma fresadora modificada para soldar chapas por FSW. Nos experimentos realizados, houve uma diminuição no consumo de potência entre um processo de soldagem sem falhas e um processo de soldagem com falhas.

As forças axial e horizontal também sofrem variações em seus módulos em casos semelhantes. Experimentos feitos por Yang et al. (2006) e por Fleming (2009) mostram que existe uma variação na magnitude da força axial conforme a ferramenta passa por uma falha já existente na chapa.

A emissão acústica (EA) tem sido usada em processos de usinagem, pois possui resposta rápida, com pouco ruído, fácil operação e baixo custo (BOMBONATO et al., 2011). Sensores de EA acoplados à peça podem adquirir um sinal acústico durante todo o processo de soldagem. Através do processamento deste sinal podem-se detectar as possíveis falhas ocorridas durante a soldagem.

De maneira semelhante, acelerômetros podem adquirir sinais de vibração das chapas durante a soldagem. Estes sinais, processados de maneira correta, podem fornecer informações da ocorrência e localização de falhas nos processos de produção.

Outro método, utilizado para monitorar a qualidade superficial das uniões FSW, utiliza técnicas de processamento de imagens. Segundo Sinha et al. (2008), a quebra acidental de uma ferramenta ou mudanças repentinas nas condições de soldagem podem ser detectadas por este método.

O presente trabalho apresenta o monitoramento do processo de soldagem por FSW de chapas de alumínio utilizando acelerômetros, fixados na superfície das chapas. Os sinais adquiridos durante o processo foram processados utilizando dois métodos diferentes (Transformada de Fourier de Tempo Curto (STFT) e Transformada *Wavelet*), que foram capazes de detectar e localizar falhas criadas nas chapas.

As técnicas selecionadas para o processamento dos sinais adquiridos nos ensaios permitem a verificação de mudanças repentinas nos sinais. Porém, como característica principal, estes métodos apresentam resultados gráficos contendo informações de frequência e tempo. Estas informações são essenciais para determinar o momento em que essas mudanças ocorreram e, conseqüentemente, suas posições na chapa soldada.

De acordo com Quatieri (2001b), a aplicação da STFT deve-se ao fato de que a simples análise de um sinal através da Transformada de Fourier não é suficiente para caracterizar mudanças

no conteúdo espectral em um certo intervalo de tempo. A STFT, entretanto, consiste em aplicar uma Transformada de Fourier separada para cada instante de tempo. Dessa forma, é possível associar cada instante da Transformada de Fourier com os instantes vizinhos, possibilitando traçar a variação espectral do sinal no tempo. Esta associação permite uma análise do comportamento do sinal como um todo e em instantes determinados.

Da mesma forma, a desvantagem de se perder as informações de tempo na análise de um sinal (assim como a Transformada de Fourier) é superada utilizando a Transformada *Wavelet*. Segundo Newland (1993), esta ferramenta apresenta uma maneira alternativa de decompor o sinal em suas formas constituintes.

Quatieri (2001a) afirma que a Transformada *Wavelet* pode ser interpretada como uma colagem de peças de espectrogramas baseados em diferentes janelas de análise e, dessa forma, diferentes resoluções tempo-frequência. De maneira simplificada, as janelas de curta duração são usadas para analisar as componentes de alta frequência, tendo assim uma boa resolução no tempo. As janelas de longa duração são usadas para as componentes de baixa frequência, para uma boa resolução em frequência.

A Fig. 1.5 mostra uma comparação entre as janelas aplicadas na STFT e na Transformada *Wavelet*. Na STFT as janelas são fixas, enquanto que na Transformada *Wavelet* mudam-se os comprimentos de acordo com o nível de decomposição.

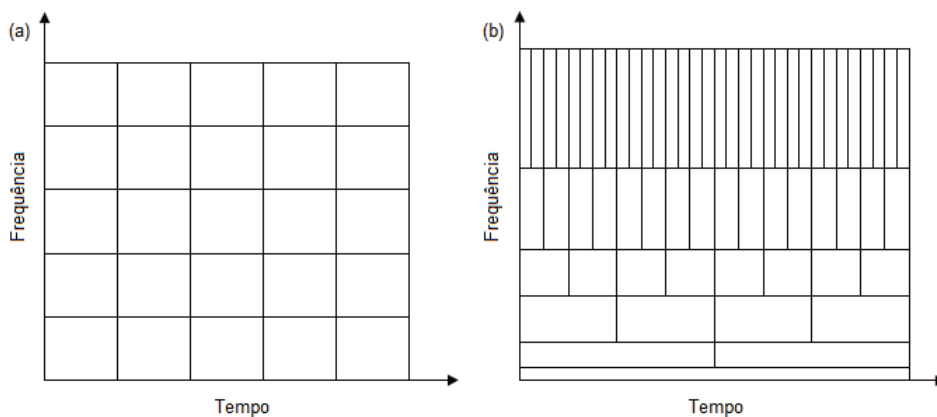


Figura 1.5 - Janelamento: (a) STFT e (b) Transformada *Wavelet* (Gallegati (2008), modificada).

1.1 Estrutura do Trabalho

O Capítulo 2 apresenta uma revisão bibliográfica com os principais trabalhos e artigos referentes ao monitoramento de processos de produção e às técnicas de união por FSW.

O Capítulo 3 mostra detalhadamente a teoria do método da Transformada de Fourier de Tempo Curto, também conhecida como STFT (*Short Time Fourier Transform*). Alguns exemplos são mostrados para demonstrar a maneira como os resultados são apresentados por este método.

O Capítulo 4 apresenta a teoria do método da Transformada *Wavelet*, explicando os parâmetros envolvidos nos cálculos e demonstrando a teoria apresentada por Newland (1993) para a elaboração de programas capazes de trabalhar com esta ferramenta.

O Capítulo 5 demonstra a utilização dos programas desenvolvidos com sinais simulados, fazendo uma comparação entre os resultados obtidos com os dois métodos.

O Capítulo 6 mostra a aplicação dos programas em sinais obtidos em ensaios, através da união por FSW de duas chapas de alumínio.

O Capítulo 7 apresenta as conclusões desta dissertação e algumas propostas para trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo, será apresentado um resumo da revisão bibliográfica realizada a respeito do monitoramento de processos de junção por FSW, métodos de detecção de falhas e o processamento dos sinais adquiridos no processo. Este estudo teve como objetivo analisar o que vem sendo feito nesta área, quais as possíveis formas de monitoramento e a melhor maneira para implementá-las.

O número de técnicas testadas até o momento não é grande. As pesquisas que tratam deste assunto não são muitas e os autores citam uns aos outros em diferentes trabalhos. Mesmo assim, diferentes técnicas são utilizadas e apresentam bons resultados.

2.1 Força axial

Os dados obtidos a respeito da força axial aplicada pela ferramenta à chapa soldada são muito importantes para monitorar a condição do cordão durante o processo. É uma técnica que permite a implementação para um monitoramento *on-line*, ou seja, permite que o processo seja monitorado e envie respostas sobre a qualidade da solda no momento em que esta é realizada.

Em seu trabalho, Yang et al. (2006) desenvolveram um algoritmo de monitoramento para detecção de falhas no processo FSW. Em estudos experimentais, determinaram como os parâmetros do processo e o tamanho das falhas afetariam o resultado final da soldagem, onde a força axial foi o principal parâmetro estudado. O algoritmo proposto examina a força axial filtrada, no domínio da frequência, para determinar a existência de uma falha. Utilizando uma máquina FSW com dois eixos equipados com células de força (um perpendicular ao eixo da ferramenta e na direção da solda e o outro paralelo ao eixo da ferramenta), os ensaios foram feitos para entender qualitativamente o problema, soldando peças com falhas e sem falhas. Três situações diferentes foram experimentadas. Na primeira, a ferramenta soldou uma peça sólida. Na segunda, duas chapas foram firmemente posicionadas, uma encostada na outra, de maneira a não existir nenhum espaço entre elas. Na

terceira, as chapas foram fixadas contendo um espaço de tamanho conhecido entre elas. Estas três situações são mostradas na Fig. 2.1.

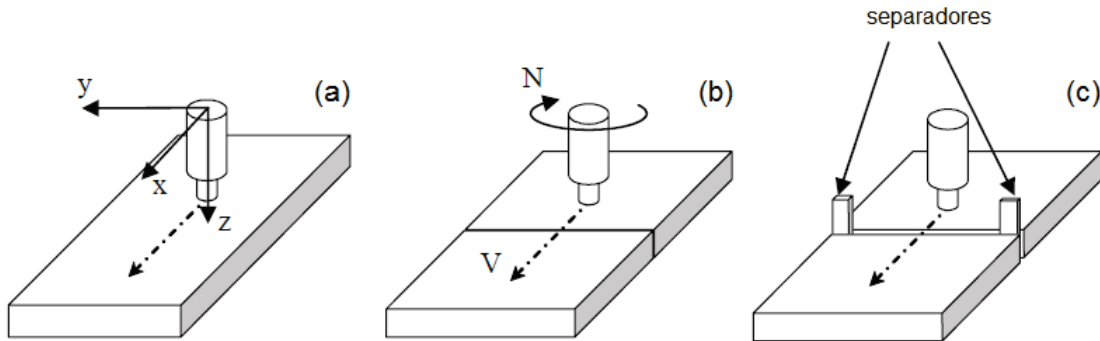


Figura 2.1 - Experimentos: (a) peça sólida, (b) chapas encostadas e (c) chapas com espaçamento conhecido (Yang et al. (2006), modificada).

A primeira situação mostrou que há um aumento na magnitude da força conforme o pino penetra a chapa, seguido de uma diminuição e estabilização em um valor constante conforme a ferramenta gira e a temperatura da chapa aumenta, como mostra a Fig. 2.2. A força aumenta novamente quando a soldagem começa até atingir um valor máximo, que se mantém até o final do processo.

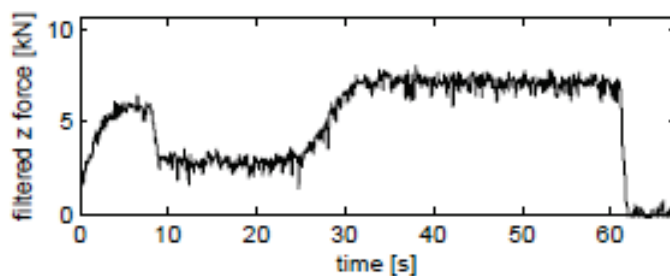


Figura 2.2 - Monitoramento da força axial: peça sólida (Yang et al. (2006), modificada).

As outras duas situações apresentam características semelhantes, como mostra a Fig. 2.3. Entretanto, há uma mudança aparente na força axial quando a ferramenta passa pela junção das chapas. Os autores mostraram que, quanto maior a largura do intervalo entre as chapas, maior será a queda na magnitude da força.

Aplicou-se aos sinais de força a Transformada de Fourier Discreta, possibilitando o cálculo da

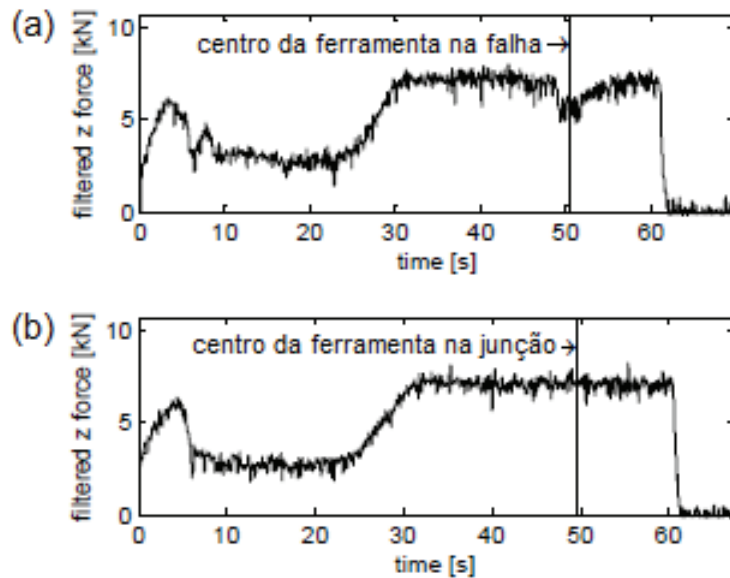


Figura 2.3 - Monitoramento da força axial (a) com espaçamento conhecido e (b) com as chapas encostadas (Yang et al. (2006), modificada).

Densidade Espectral de Potência, utilizada para monitorar automaticamente a presença de falhas. Segundo Yang et al. (2006), assim como a força axial, a magnitude da Densidade Espectral de Potência aumenta conforme a largura da falha aumenta.

Estudos semelhantes foram feitos por Fleming (2009). Em seus experimentos, a velocidade de rotação utilizada foi de 2000 RPM e a velocidade de avanço de 16 polegadas por minuto (406,4 mm/min). Falhas com diferentes profundidades (0,0004", 0,0008", 0,0012", 0,0016", 0,0020", 0,0030", 0,0040", 0,0050") foram feitas em uma chapa e então outra chapa, sem falhas, foi presa acima da primeira. Desta maneira, as falhas simularam vazios no interior do material. O monitoramento do processo foi feito utilizando dinamômetros. Entretanto, a primeira análise após a realização do processo foi visual, para tentar identificar algum problema no cordão, porém nenhuma falha pode ser encontrada.

Analisando, então, os sinais de força axial adquiridos no experimento, Fleming (2009) notou que, para falhas com profundidade acima de 0,002", houve uma notável queda nos valores de força, como mostra a Fig. 2.4. Segundo o autor, quedas repentinas na intensidade da força axial podem monitorar bem os processos FSW, podendo-se relacionar a intensidade da variação da força com o tamanho da falha. Entretanto, também é possível que esta informação, se não processada

corretamente, possa ser insuficiente para a detecção acurada de falhas, ou seja, existem outros motivos que podem fazer com que a intensidade da força varie.

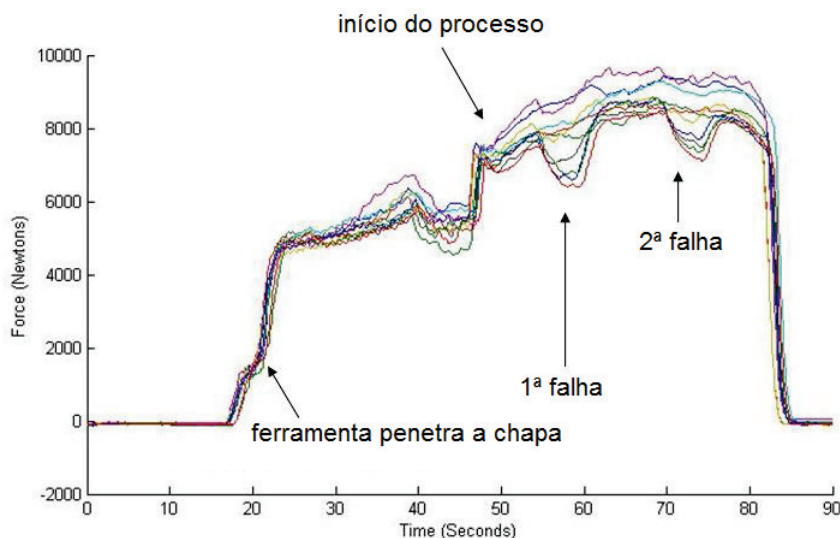


Figura 2.4 - Monitoramento das forças axiais para diferentes tamanhos de falha (Fleming (2009), modificada).

2.2 Processamento de imagens

A obtenção de imagens digitais da superfície das juntas de chapas unidas por FSW também é uma técnica com bons resultados. É possível a implementação deste método para monitoramento *on-line*, porém os resultados obtidos mostram falhas apenas na superfície da junção. Nos casos em que as falhas são internas, este método é incapaz de detectá-las.

Em seu artigo, Sinha et al. (2008) discutiram um método para monitoramento *on-line* do processo FSW utilizando técnicas de processamento de imagens. Diversos experimentos foram feitos utilizando diferentes parâmetros, onde imagens do primeiro modo da zona de solda foram capturadas. Segundo Muthukumaran e Mukherjee (2006), o primeiro modo de transferência de metal se dá através da ação do ombro da ferramenta, que gera calor e faz com que o material deformado seja movido de maneira a ser depositado em camadas atrás da ferramenta. Já o segundo

modo acontece ao redor do pino, onde o metal se deforma por extrusão ao seu redor.

Nos experimentos feitos por Sinha et al. (2008), o processo FSW foi feito utilizando uma fresadora modificada, onde o ângulo de inclinação da ferramenta utilizado foi de 1° . A velocidade de rotação utilizada foi de 1120 RPM e a velocidade de avanço de 200 mm/min.

As imagens foram capturadas com uma câmera digital durante o processo. A primeira etapa consistiu em aplicar filtros às imagens, para eliminar ruídos. Em seguida, as cores foram passadas para uma escala de cinza, possibilitando a criação de um histograma do nível de cinza das imagens, mostrando a distribuição das intensidades.

Uma seção transversal foi selecionada ao longo da imagem em escala de cinza e as variações nas intensidades dos *pixels* desta seção foram apresentadas em um gráfico. Por fim, o contorno da imagem (seção onde os valores de intensidade são iguais) foi feito, gerando um mapa de contorno que possibilita a extração de dados úteis da imagem.

O primeiro experimento, feito com uma chapa uniformemente soldada, mostrou variações mínimas das intensidades dos *pixels* ao longo da seção transversal. Além disso, apresentou linhas contínuas e uniformes no mapa de contorno, como mostra a Fig. 2.5.

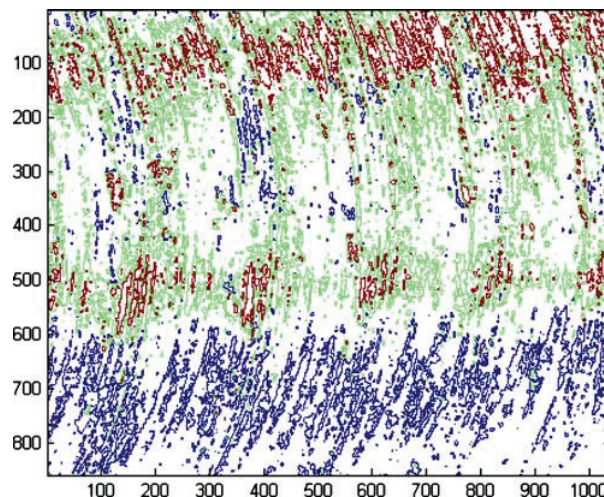


Figura 2.5 - Mapa de contorno para chapa uniformemente soldada (Sinha et al. (2008)).

Simulando uma falha no pino, o segundo experimento mostrou grandes variações nas intensidades dos *pixels* ao longo da seção transversal. No entanto, pequenas variações nas intensidades

da escala de cinza foram observadas antes da falha ocorrer. Segundo Sinha et al. (2008), pode-se dizer que antes da falha do pino, a transferência de metal se dá em dois modos, mas após a falha o primeiro modo fica afetado devido à falta do segundo modo.

Em um terceiro experimento, simulando a falta de penetração da ferramenta, os mapas de contorno apresentaram falhas, ou seja, os contornos foram quebrados e descontínuos por toda a zona soldada, como mostra a Fig. 2.6.

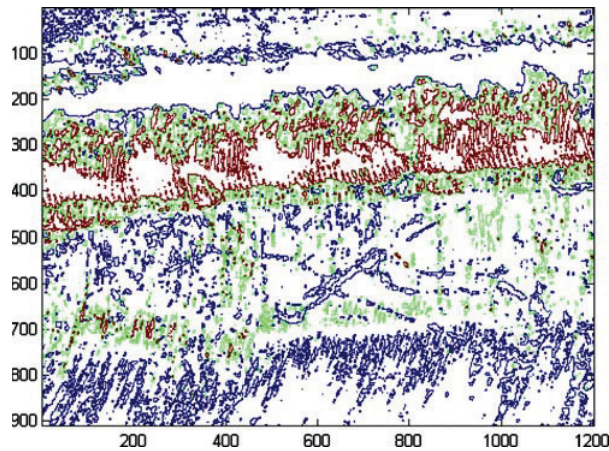


Figura 2.6 - Mapa de contorno para falta de penetração da ferramenta (Sinha et al. (2008)).

Neste mesmo trabalho, Sinha et al. (2008) também discutem a possibilidade de monitorar o processo através do consumo de potência da máquina. A potência de saída do motor, segundo os autores, é calculada através da corrente elétrica e do sinal de voltagem do eixo do motor. Nos experimentos em questão, o consumo de potência durante uma soldagem sem falhas foi de aproximadamente 3,5 kW, enquanto que a potência consumida em um processo falhado foi de aproximadamente 2,0 kW.

2.3 Sensores de emissão acústica

O método mais utilizado dentre os estudados é o monitoramento através de sinais de emissão acústica (EA). Sensores EA acoplados às chapas adquirem os sinais de emissão acústica durante o processo FSW e, após o término do processo, são processados computacionalmente.

Em seu trabalho, Chen et al. (2003) investigam a possibilidade de aplicar a técnica de emissão acústica para o monitoramento do processo FSW. Testes experimentais foram feitos em chapas de liga de alumínio 6061, onde três falhas igualmente espaçadas foram feitas, como mostra o esquema da Fig. 2.7. Uma ferramenta com diâmetro de ombro de 24 mm, diâmetro de pino de 6 mm e altura de pino de 5,5 mm foi utilizada. Os sensores EA foram posicionados simetricamente em ambos os lados da junta, a uma distância de 70 mm.

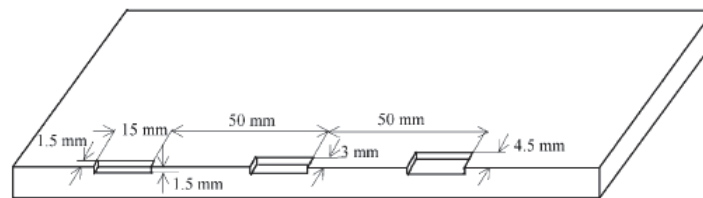


Figura 2.7 - Dimensões das falhas (Chen et al. (2003)).

A primeira aquisição de dados foi feita com a máquina sem soldar nenhuma chapa, para detectar os ruídos de funcionamento da máquina e ajustar os filtros necessários. Após ajustados os filtros, o sinal EA foi coletado em um processo de soldagem comum, com as chapas contendo as falhas conhecidas.

Para decompor o sinal EA, Chen et al. (2003) utilizaram a Transformada *Wavelet*. Segundo os autores, existem mudanças significativas na banda de energia no momento em que o pino penetra ou sai da junta soldada, assim como quando o ombro da ferramenta encosta ou se desprende das chapas. A variação da banda de energia durante a translação da ferramenta através da região com a falha permite detectar a existência, a localização e o tamanho das falhas, como mostra a Fig. 2.8. Através de um mapa de contorno, este método pode ser efetivamente utilizado para monitorar o estado do processo e rapidamente identificar os defeitos.

Na Fig. 2.8, os picos maiores indicam a entrada da ferramenta nas falhas, enquanto que os picos menores indicam a chegada da ferramenta na borda final da falha.

Em outro artigo importante, Soundararajan et al. (2006) discutem a análise de sinais de emissão acústica para investigar a possibilidade de aplicação da técnica no monitoramento do processo FSW. Os experimentos foram feitos utilizando diferentes velocidades de rotação, de translação e

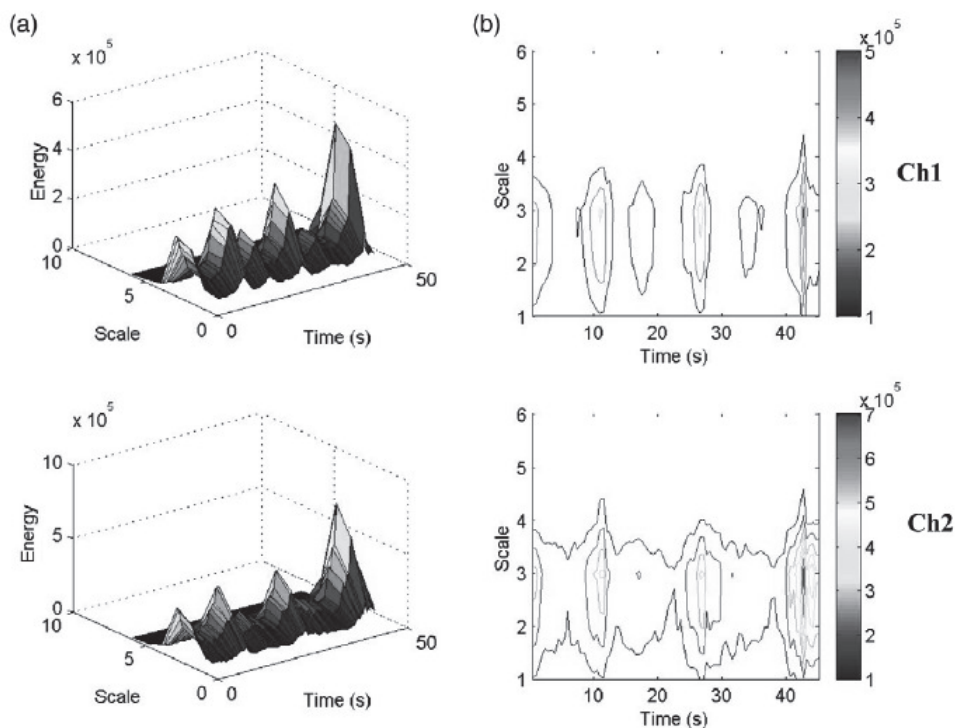


Figura 2.8 - Processo com as falhas: (a) perfil tridimensional e (b) mapa de contornos da Transformada *Wavelet* (Chen et al. (2003)).

diferentes profundidades de penetração da ferramenta.

Neste trabalho, Soundararajan et al. (2006) realizaram três experimentos: o primeiro envolveu a variação da velocidade de rotação da ferramenta (500, 750 e 1000 RPM); o segundo apresentou mudança nas velocidades de translação e rotação (400 RPM e 75 mm/min para 750 RPM e 50 mm/min) e o objetivo foi observar as mudanças no sinal de emissão acústica devido à rápida variação nas condições termomecânicas na zona de solda; o terceiro sofreu mudanças na profundidade de penetração da ferramenta (0,2 mm, 0,4 mm e 0,6 mm). De acordo com Soundararajan et al. (2006), no terceiro experimento, o contato do ombro da ferramenta com a peça não foi alcançado inicialmente, mas uma condição de contato muito boa foi conseguida nos próximos dois estágios.

Resultados da aplicação da FFT nos sinais de emissão acústica obtidos nos experimentos de Soundararajan et al. (2006) mostraram que a amplitude do sinal no domínio da frequência foi sensível à mudança de profundidade de penetração da ferramenta. Sinais de certas frequências desapareceram quando a ferramenta perdeu o contato com a peça durante o processo.

Segundo Soundararajan et al. (2006), a Transformada *Wavelet* discreta mostrou mudanças

abruptas significantes no sinal decomposto nos níveis de decomposição mais altos, quando o ombro entrava em contato ou se desprendia da peça. Identificando as frequências durante o processo e analisando os sinais decompostos pela *wavelet* em vários níveis, foi possível monitorar efetivamente o estado da soldagem e identificar rapidamente mudanças no processo.

2.4 STFT e Transformada *Wavelet*

Como forma de processamento dos sinais adquiridos em ensaios ou processos, a STFT e a Transformada *Wavelet* são ferramentas capazes de gerar resultados com informações de tempo e frequência, importantes para a detecção e localização de mudanças no sinal.

Em seu trabalho, Moraes (1996) analisou a aceleração e desaceleração de um rotor montado em bancada. Inicialmente, simulou dois sinais idênticos, compostos por componentes harmônicas com diferentes frequências e ponderações, de forma a exemplificar a vibração mecânica de um sistema rotativo. Em um dos sinais, somou uma sequência de oito pulsos transitórios, representando as falhas em 0,5s. Visualmente, os dois sinais simulados não apresentaram diferenças. Entretanto, ao aplicar a Transformada *Wavelet*, foi possível notar diferenças nas escalas 10 e 11 do Mapa de Quadrados Médios, em 0,5s, como mostra a Fig. 2.9.

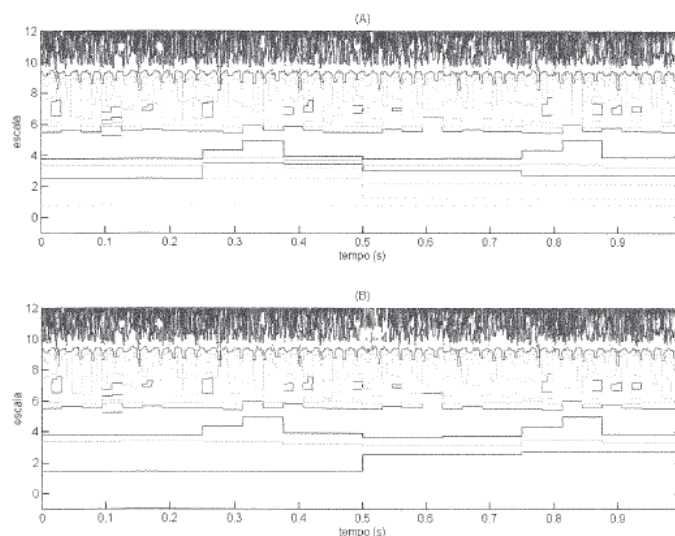


Figura 2.9 - Mapas de Quadrados Médios: (a) sem falhas e (b) com falhas (Moraes (1996)).

O Mapa de Quadrados Médios é uma representação gráfica da aplicação da Transformada *Wavelet*, e será explicado no Capítulo 4.

Na análise da aceleração de rotores, a montagem experimental consistiu em um eixo montado em mancais de rolamentos suportados por réguas flexíveis de aço, e as medições foram feitas através de acelerômetros colados nas direções vertical e horizontal.

Foram obtidos os Mapas de Quadrados Médios deste experimento e, segundo Moraes (1996), além da identificação das passagens por ressonâncias, foi possível perceber um comportamento fora do padrão no intervalo de 2,6s a 3,1s, nas escalas de 8 a 10, sendo relacionado a características estruturais ou de flexão do rotor. A Fig. 2.10 mostra este comportamento.

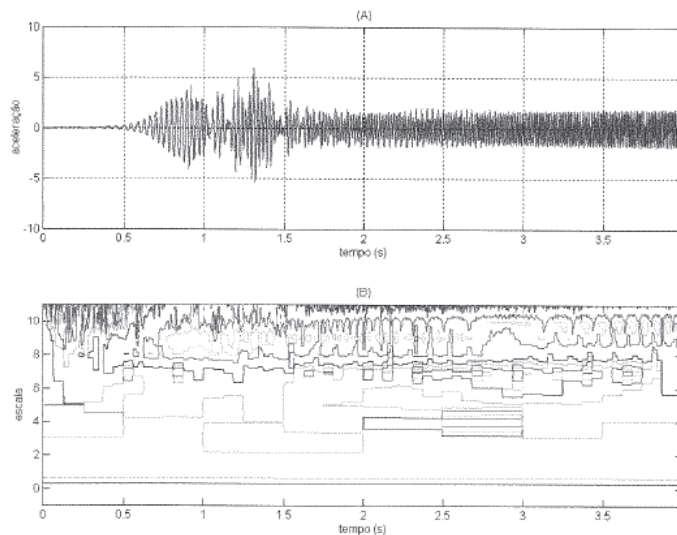


Figura 2.10 - Ensaio de aceleração: (a) domínio do tempo e (b) Mapa de Quadrados Médios (Moraes (1996)).

Aplicando-se, então, a STFT, Moraes (1996) buscou caracterizar melhor o fenômeno (Fig. 2.11). Segundo o autor, no instante em que a anormalidade foi detectada, surgiram duas componentes, uma com frequência aproximada de 95 Hz e a outra cuja frequência corresponde à soma da componente da rotação com a componente de 95 Hz.

Com estes resultados, o autor mostra que ambos os métodos foram capazes de detectar variações no padrão de sinais simulando um processo mecânico assim como sinais obtidos experimentalmente.

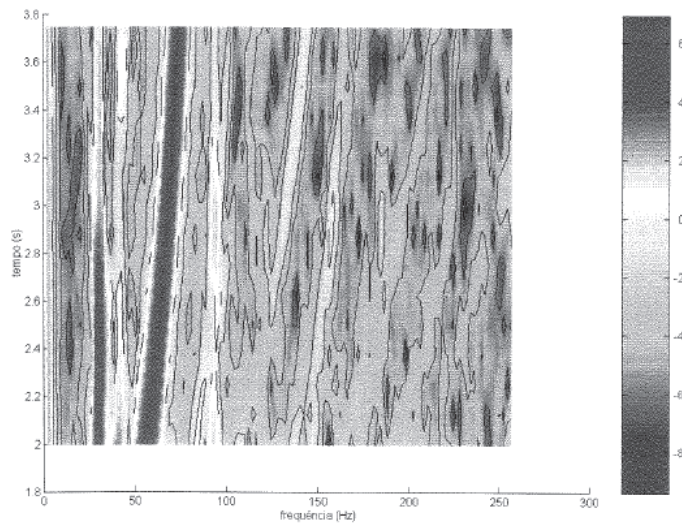


Figura 2.11 - STFT para ensaio de aceleração (Moraes (1996)).

3 TRANSFORMADA DE FOURIER DE TEMPO CURTO

3.1 Introdução

Uma das principais formas de monitoramento de processos é a aquisição dos sinais envolvidos nestes. Através de acelerômetros, sensores de emissão acústica, dinamômetros ou qualquer outro método adequado ao processo, é possível monitorar os sinais e guardá-los como arquivos. Análises computacionais são, em geral, rápidas e por isso ganham destaque.

Para que os resultados sejam devidamente utilizados e tenham validade, devem ser processados de maneira a se obter o máximo aproveitamento do processo de monitoramento realizado. Existem diversas maneiras de processar um sinal e cada uma delas existe para um diferente propósito. As diferentes características dos processamentos dos sinais fazem com que alguns sejam mais indicados para um determinado processo e outros não acrescentem nenhuma informação à análise.

Uma maneira muito eficaz para obter dados de um sinal é trabalhar com seu espectro de frequências. Informações muito úteis são geradas com esta análise, e algumas delas serão discutidas nas seções que se seguem.

3.2 Transformada de Fourier

Uma ferramenta muito útil em processamento de sinais é a Transformada de Fourier, que decompõe um sinal em diversas componentes senoidais de diferentes frequências. Dessa forma é possível obter as informações espectrais do sinal. Um problema com este tipo de análise é que a informação temporal é perdida. Matematicamente, a Transformada de Fourier Contínua é definida como:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt. \quad (3.1)$$

Os sinais adquiridos no monitoramento de processos podem ser utilizados e processados computacionalmente. Dessa forma, é necessário que os sinais a serem analisados pela Transformada de Fourier sejam discretos e, para processá-los, é necessário que a Transformada de Fourier também seja discreta. A Eq. (3.2) apresenta a Transformada de Fourier Discreta.

$$f_j = \sum_{k=0}^{n-1} x_k e^{-\frac{2\pi i}{n} jk}, k = 0, \dots, n - 1. \quad (3.2)$$

A Transformada de Fourier apresenta uma análise global das frequências que compõem um sinal, sendo muito útil para sinais onde estas componentes não variam com o tempo. Contudo, para sinais onde há distúrbios ou falhas, esta ferramenta não é a mais indicada.

Sem a informação de tempo, é impossível apontar o instante em que o distúrbio ocorreu. Em algumas análises, nem mesmo é possível dizer que houve algum distúrbio utilizando apenas a Transformada de Fourier.

Para exemplificar melhor esta situação, a Fig. 3.1 apresenta dois sinais diferentes que serão analisados utilizando a Transformada Rápida de Fourier (*Fast Fourier Transform* - FFT), que é um algoritmo eficiente para se calcular a Transformada de Fourier Discreta. Ambos os sinais são compostos por três frequências. O primeiro é formado pela soma de três senóides, uma em cada frequência (50, 100 e 200 Hz). O segundo é formado pelas mesmas três frequências, mas as três senóides não foram somadas e sim posicionadas uma ao lado da outra, ou seja, cada terço do sinal apresenta apenas uma frequência, diferente dos outros terços. É possível notar visualmente as diferenças entre estes sinais. Aplicando-se a Transformada Rápida de Fourier aos dois sinais, os resultados obtidos são as frequências que os compõem, mostradas na Fig. 3.2. Como dito anteriormente, sem a informação temporal, as informações dadas pela análise não permitem qualquer distinção entre os diferentes sinais.

Como os sinais provenientes dos processos de união por FSW podem possuir variações na frequência ou na amplitude devido a distúrbios ou falhas, é preciso utilizar outra forma de pro-

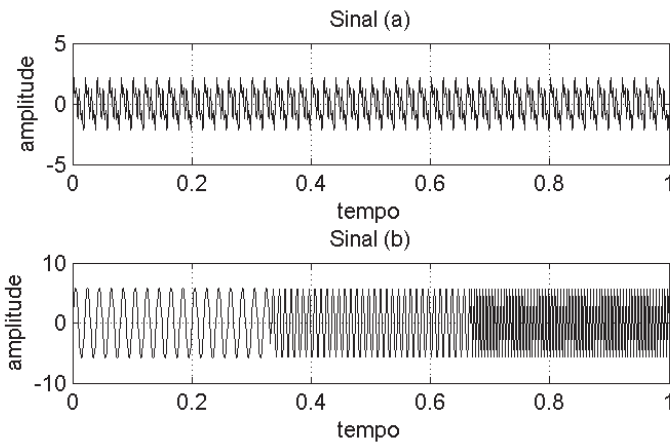


Figura 3.1 - Sinais formados por (a) soma das senóides e (b) posicionamento das senóides.

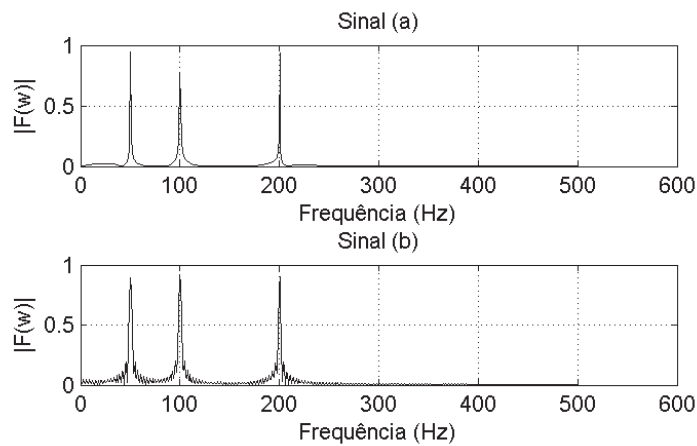


Figura 3.2 - FFTs dos sinais da Fig. 3.1.

cessamento que possibilite a identificação do instante de tempo em que essas variações ocorreram. Dessa forma, baseando-se na Transformada de Fourier, foi desenvolvida a Transformada de Fourier de Tempo Curto.

3.3 Transformada de Fourier de Tempo Curto

A Transformada de Fourier de Tempo Curto (*Short Time Fourier Transform - STFT*), também conhecida como Transformada de Fourier Janelada ou Transformada de Gabor tenta adequar a necessidade de tratar os sinais não apenas no domínio do tempo ou da frequência, mas em ambos.

Segundo Quatieri (2001b), na análise de sinais utilizando a Transformada de Fourier não é possível caracterizar mudanças do conteúdo espectral variante no tempo. A STFT consiste, portanto, de uma Transformada de Fourier separada para cada instante de tempo. Dessa forma, é possível associar cada instante da Transformada de Fourier com os instantes vizinhos, possibilitando traçar a variação espectral do sinal no tempo.

Na STFT Contínua a função a ser transformada é multiplicada por uma função janela não nula por um período de tempo curto. Assim, a Transformada de Fourier do sinal resultante é calculada conforme a janela é deslizada ao longo do eixo do tempo, resultando em uma representação bidimensional do sinal. Matematicamente esse processo é dado por:

$$F(\tau, \omega) = \int_{-\infty}^{\infty} f(t) \omega(t - \tau) e^{-i\omega t} dt, \quad (3.3)$$

onde $\omega(t)$ é a função janela e $f(t)$ é o sinal a ser transformado. As janelas mais utilizadas são a Hanning e a Gaussiana.

Na STFT Discreta, mais importante em termos computacionais, os dados a serem transformados podem ser divididos em diferentes partes. Para evitar problemas com os contornos (as extremidades) dos sinais, estas partes são sobrepostas umas às outras. Esta sobreposição é denominada *overlap*. A porcentagem de *overlap* deve ser definida no início do processo. De acordo com Chan e Zhu (2012) e Gutierrez-Osuna (2012), a equação da STFT Discreta é dada por:

$$F(m, \omega) = \sum_{n=-\infty}^{\infty} f[n] \omega[m - n] e^{-i\omega n}, \quad (3.4)$$

sendo $f[n]$ o sinal discreto e $\omega[n]$ a janela.

Após a divisão do sinal, tem-se diversos intervalos de tempo e a Transformada de Fourier é aplicada a cada um destes intervalos, como mostra a Fig. 3.3. Gera-se, a partir destes intervalos, uma matriz que será utilizada para criar um mapa de cores, onde o eixo horizontal pode ser representado pelo tempo ou espaço, o eixo vertical representado pelas frequências e as cores mostram as intensidades (os picos) das frequências.

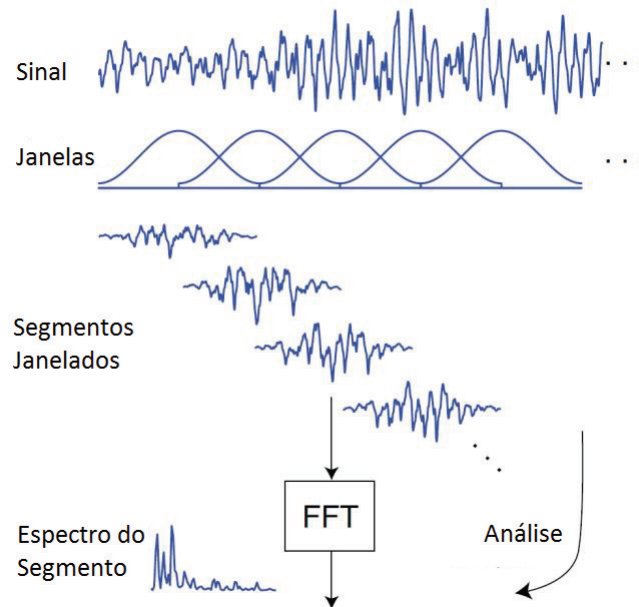


Figura 3.3 - Janelamento do sinal (Sethares (2012), modificada).

Tomando como exemplo os mesmos sinais mostrados na Fig. 3.1, é possível mostrar como a STFT trabalha de maneira a preservar os dados de tempo, possibilitando a diferenciação na análise dos sinais e localização (no tempo ou espaço) das mudanças ocorridas. A Fig. 3.4 mostra o mapa de cores de ambos os sinais, onde o eixo horizontal representa o tempo e o eixo vertical representa as frequências dos trechos do sinal. As cores mostram a intensidade dos valores de frequência, onde os valores mais baixos aproximam-se do azul e os valores mais altos aproximam-se do vermelho. Nota-se, claramente, a diferença entre os dois sinais na análise de suas frequências. O primeiro sinal possui as frequências de 50, 100 e 200 Hz em toda sua extensão enquanto que o segundo sinal apresenta apenas uma frequência em cada porção. O mais importante é notar a escala de tempo presente, possibilitando a identificação do instante em que as frequências mudam.

Para uma boa utilização do método, deve-se levar em consideração alguns parâmetros da análise. Como dito, a porcentagem de *overlap* deve ser definida, assim como o tipo de janela utilizado. O tamanho da janela também deve ser definido e não será modificado até que a análise seja feita. Esta definição afeta a resolução tempo/frequência da STFT, assim como mostra a Tabela 3.1.

Nota-se, através das informações da Tabela 3.1, que deve existir um equilíbrio entre as resoluções de tempo e frequência, pois ao priorizar uma delas, a outra fica comprometida. Se o sinal é dividido em intervalos muito pequenos, a análise terá uma boa resolução no tempo, entretanto não

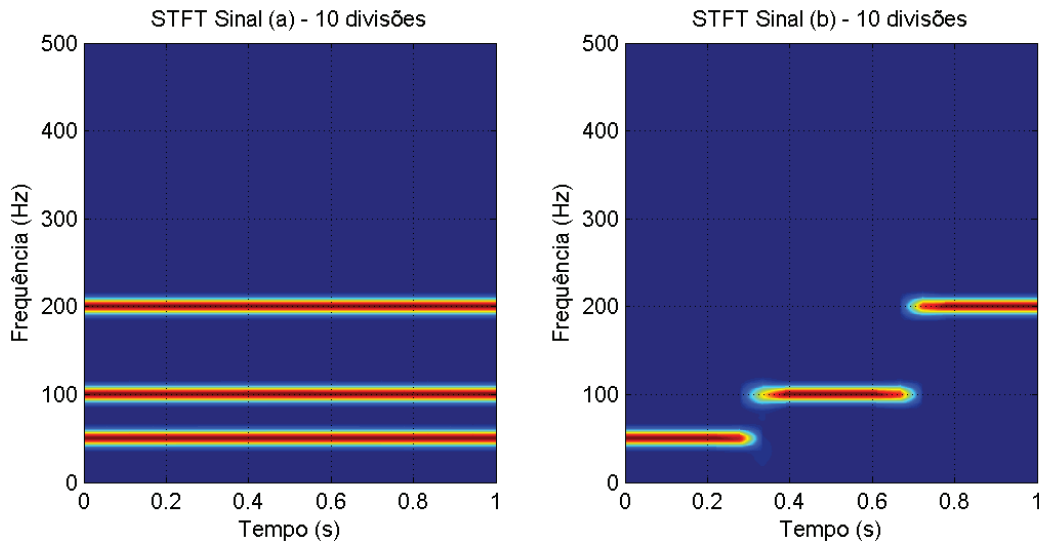


Figura 3.4 - STFTs dos sinais da Fig. 3.1.

Tabela 3.1 - Tamanho das Janelas (Chan e Zhu (2012), Quatieri (2001b))

Tamanho da Janela	Banda	Resolução no Tempo	Resolução na Frequência	Utilização
Longa	Estreita	Ruim	Boa	Componentes senoidais (harmônicos)
Curta	Larga	Boa	Ruim	Componentes de rápida variação no tempo

há pontos suficientes para uma boa análise pela Transformada de Fourier. Por outro lado, se o sinal for dividido em grandes intervalos, as informações espectrais serão boas, mas a resolução no tempo não será adequada para a análise.

Segundo Quatieri (2001b), $|F(n, \omega)|^2$ é o espectrograma de um sinal. Para uma janela curta, $|F(n, \omega)|^2$ é denominado “espectrograma de banda larga” (*wideband spectrogram*). Quando a janela utilizada é longa, $|F(n, \omega)|^2$ é denominado “espectrograma de banda estreita” (*narrowband spectrogram*). O autor também afirma que uma janela longa representa melhor a harmonicidade espectral, enquanto que a janela curta prejudica esta harmonicidade e degrada as amplitudes harmônicas, mas captura melhor as mudanças na harmonicidade.

Para demonstrar as diferentes resoluções, utilizar-se-á o mesmo exemplo dos casos anteriores. O que será mudado é o número de divisões do sinal. Utilizando 3 divisões - um número pequeno - as frequências ficam bem definidas, porém as informações temporais se misturam, como mostra a Fig. 3.5.

Com 20 divisões, um número de divisões relativamente grande para o número de pontos deste sinal, as informações de tempo são muito bem definidas, porém nota-se uma dificuldade em calcular corretamente as frequências que compõem o sinal, como mostra a Fig. 3.6.

Um exemplo de um bom número de divisões destes sinais é mostrado na Fig. 3.7. Para esta análise em questão, 10 divisões é um valor onde as informações de tempo são boas e as informações de frequência são preservadas.

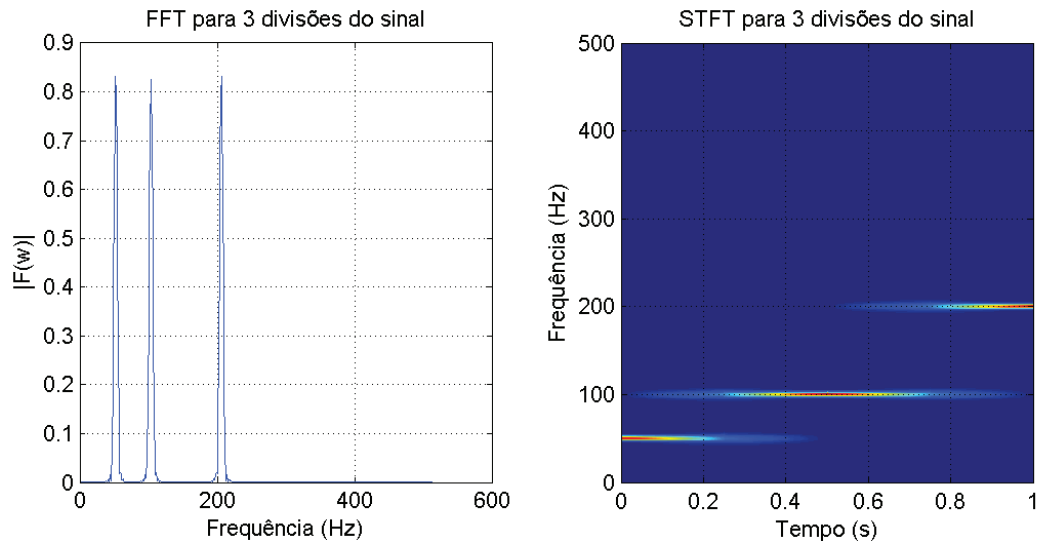


Figura 3.5 - FFT e STFT do Sinal (b) da Fig. 3.1 utilizando 3 divisões.

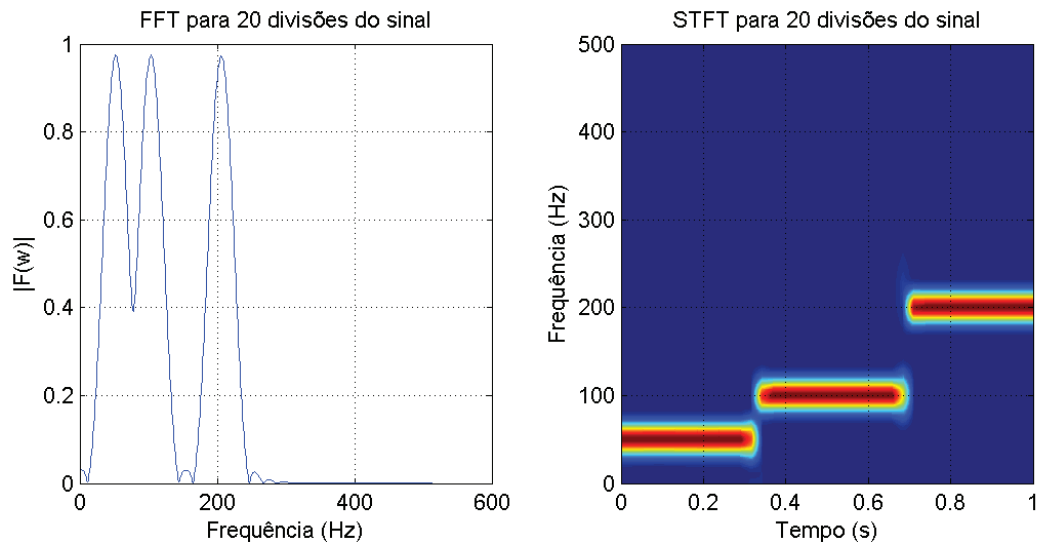


Figura 3.6 - FFT e STFT do Sinal (b) da Fig. 3.1 utilizando 20 divisões.

Este método pode ser aplicado aos sinais adquiridos nos processos FSW, visto que estes sinais

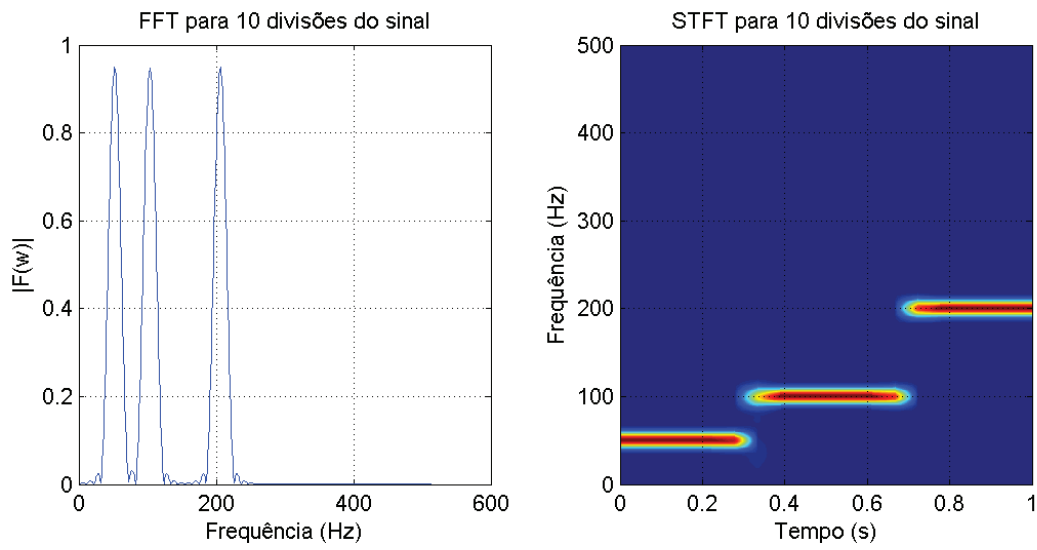


Figura 3.7 - FFT e STFT do Sinal (b) da Fig. 3.1 utilizando 10 divisões.

não são periódicos e a informação de tempo é importante para a localização das falhas. É possível que, no momento em que a ferramenta passar por uma falha, haja alguma variação das frequências componentes do sinal, que serão mostradas no mapa de cores resultante da aplicação do método da STFT.

4 TRANSFORMADA WAVELET

4.1 Introdução

Outra ferramenta muito importante no processamento de sinais com variações em suas frequências é a Transformada *Wavelet*. Esta ferramenta apresenta uma análise multirresolução onde um sinal é analisado em diferentes níveis de resolução de tempo e frequência. A Transformada *Wavelet* é considerada uma boa opção para detecção e localização de transitórios com componentes de baixa e alta frequência (CAPELARI; MAZZAFERRO, 2009).

Alfred Haar foi o primeiro a mencionar sobre as *wavelets*, no ano de 1909 em sua tese de Ph.D. Jean Morlet e Alex Grossman desenvolveram a decomposição de um sinal em seus coeficientes, assim como sua reconstrução em 1981. Em 1986, Stephane Mallat e Yves Meyer usaram *wavelets* para desenvolver uma análise multirresolução, mencionando a função escala das *wavelets* pela primeira vez. Dessa maneira, pesquisadores puderam construir suas próprias famílias de *wavelets*. No ano de 1998, Ingrid Daubechies usou a análise multirresolução para construir sua própria família de *wavelets*, mostrada na Fig. 4.1, muito importante atualmente (MERRY, 2012).

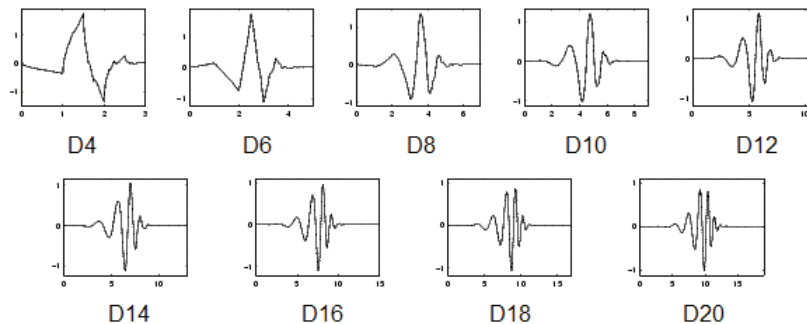


Figura 4.1 - Família Daubechies de *wavelets* (Oliveira (2012), modificada).

A Transformada *Wavelet* decompõe um sinal adquirido $f(t)$ em componentes cuja forma depende da forma da *wavelet* de dilatação (também conhecida como *wavelet*-mãe) escolhida. A *wavelet* de dilatação é uma função escolhida de uma família de *wavelets* e é utilizada em ver-

sões expandidas ou comprimidas e deslocada no eixo horizontal, assim como mostra a Fig. 4.2 (MELLO, 2011; OLIVEIRA, 2012). Expandindo ou comprimindo a função é possível varrer todas as frequências do sinal. Deslocando-a é possível varrer todo o comprimento do sinal.

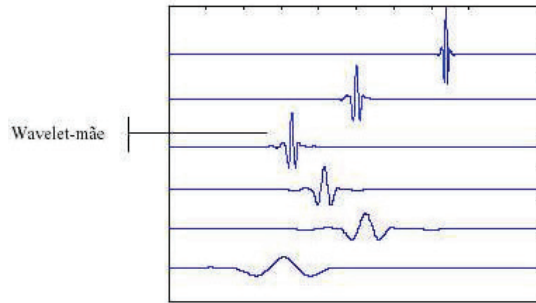


Figura 4.2 - Expansão e translação da *wavelet*-mãe. (Oliveira (2012)).

Em seu livro, Newland (1993) apresenta a teoria para a decomposição *wavelet* e o Mapa de Quadrados Médios, que serão explicados adiante. Esta teoria foi utilizada neste trabalho, encaixando-se bem aos propósitos de análise dos sinais gerados no monitoramento do processo FSW, e será detalhada neste capítulo. Os sinais analisados devem ter como comprimento L (número de pontos) uma potência de 2 ($L = 2^n$). Uma boa maneira de se entender como funciona a decomposição *wavelet* é através de exemplos. Assim, para exemplificar a decomposição, utilizar-se-á uma onda quadrada com dois ciclos formada por $L = 2^9 = 512$ pontos, como mostra a Fig. 4.3.

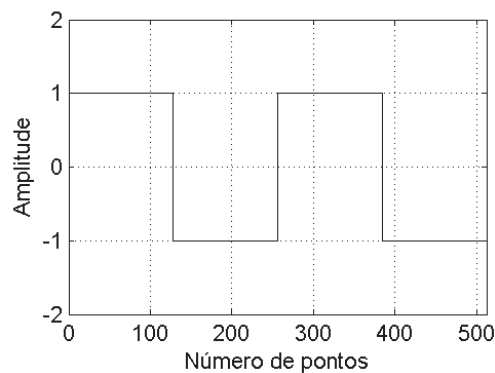


Figura 4.3 - Onda quadrada de dois ciclos.

O primeiro passo na decomposição é escolher uma *wavelet*-mãe. Neste exemplo, a *wavelet* D4 será utilizada. A *wavelet* D4 faz parte da família Daubechies de *wavelets*, por isso sua denominação "D"; e é construída utilizando 4 coeficientes *wavelet*, por isso a numeração "4".

É possível ver na Fig. 4.4 as componentes do sinal, onde cada uma delas representa um nível. A quantidade de níveis de uma decomposição é determinada pelo tamanho do sinal ($L = 2^n$), apresentando sempre $n + 1$ níveis, numerados de -1 a $n - 1$. No exemplo em questão, existem 10 níveis, numerados de -1 a 8.

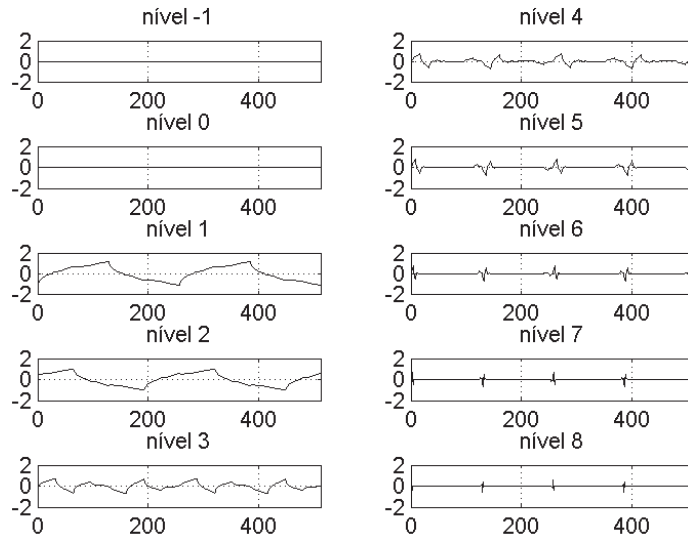


Figura 4.4 - Níveis da decomposição da onda quadrada.

Por definição, em cada nível j existem 2^j *wavelets*, uma ao lado da outra na escala horizontal. A forma da *wavelet* D4 utilizada para decompor a onda quadrada do exemplo, em nível $j = 3$, pode ser vista no primeiro gráfico da Fig. 4.5. Neste nível, a *wavelet* ocupa apenas parte do comprimento do sinal. Seguindo a definição da transformada *wavelet*, para cobrir toda a extensão do sinal, deve-se adicionar *wavelets* a este nível até que contenha $2^3 = 8$ *wavelets*. Estas estarão espaçadas em $L/2^j = 512/8 = 64$ posições em relação à *wavelet* vizinha.

O segundo gráfico da Fig. 4.5 mostra três *wavelets* adjacentes sobrepostas e o terceiro gráfico mostra as mesmas três *wavelets* somadas. O quarto gráfico mostra todas as oito *wavelets* desta escala somadas.

A estrutura da Transformada *Wavelet* determina a posição e espaçamento de cada *wavelet* ao longo do eixo horizontal. Porém a amplitude de cada *wavelet* pode ser alterada. O objetivo é adquirir o sinal original discreto $f(n)$ e convertê-lo em uma nova sequência de números reais

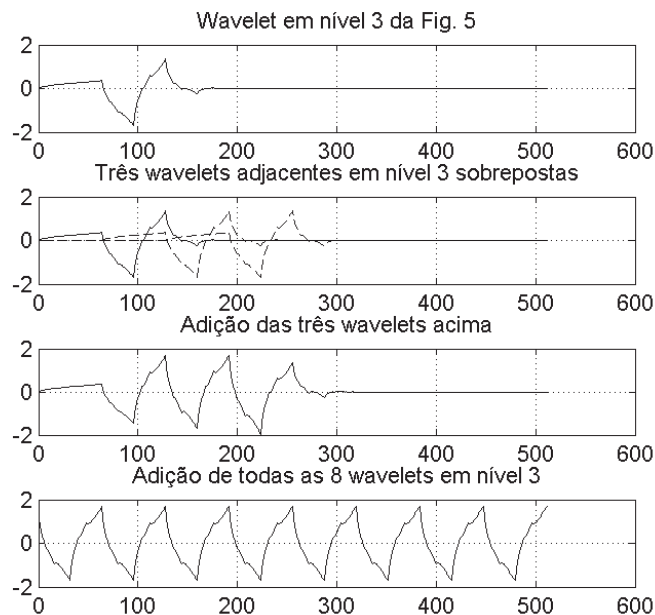


Figura 4.5 - *Wavelets* D4 em nível 3.

$a(n)$, de mesmo tamanho L e que define a amplitude de cada *wavelet* em cada nível de maneira a assegurar que, ao final da transformada, a soma de cada componente reproduza fielmente o sinal original. Para obter essa sequência de números reais $a(n)$, faz-se operações matriciais utilizando os coeficientes *wavelet*. Estes coeficientes já foram calculados e é possível encontrá-los tabelados. Para entender melhor as operações da Transformada *Wavelet*, é necessário apresentar a ideia das equações de dilatação (Seção 4.2) e *wavelets* de dilatação (Seção 4.3).

4.2 Equações de Dilatação

As equações de dilatação dão origem às *wavelets* de dilatação. O termo dilatar significa expandir, ou seja, esta equação pode expandir ou comprimir as *wavelets* de dilatação. A função $\phi(x)$ é uma versão expandida de $\phi(2x)$, como mostra a Fig. 4.6. A amplitude é a mesma (neste caso, unitária), porém a escala horizontal de $\phi(x)$ é o dobro de $\phi(2x)$.

É possível expressar $\phi(x)$ como uma série finita, onde cada termo envolve a função compri-

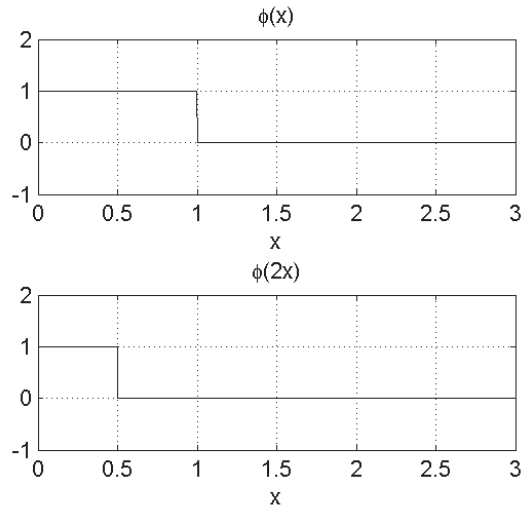


Figura 4.6 - Função expandida.

mida $\phi(2x)$. Através do argumento $(2x - k)$, é possível posicionar cada um dos termos comprimidos no eixo horizontal através de um valor inteiro, positivo ou negativo, k . Dessa maneira, a equação de dilatação básica tem a seguinte forma:

$$\phi(x) = \sum_{k=0}^{N-1} c_k \phi(2x - k), \quad (4.1)$$

onde c_k são os coeficientes *wavelet*, com k variando de 0 a $N - 1$ e N o número de coeficientes (N é sempre par). A maneira mais simples para resolver a Eq. (4.1) é através de um algoritmo iterativo, onde cada nova aproximação $\phi_j(x)$ é calculada baseando-se na anterior, $\phi_{j-1}(x)$, como na Eq. (4.2). O critério de parada do processo iterativo se dá quando ϕ_j atingir sua forma limitante, ou seja, não há mais diferenças entre ϕ_j e ϕ_{j-1} .

$$\phi_j(x) = \sum_{k=0}^{N-1} c_k \phi_{j-1}(2x - k). \quad (4.2)$$

A Fig. 4.7 mostra oito iterações do processo para determinação da função escala D4. Esta função faz parte da família Daubechies, e utiliza quatro coeficientes para sua determinação. Por isso a denominação D4, como dito anteriormente.

As iterações seguem uma operação matricial, exemplificada para a segunda iteração na

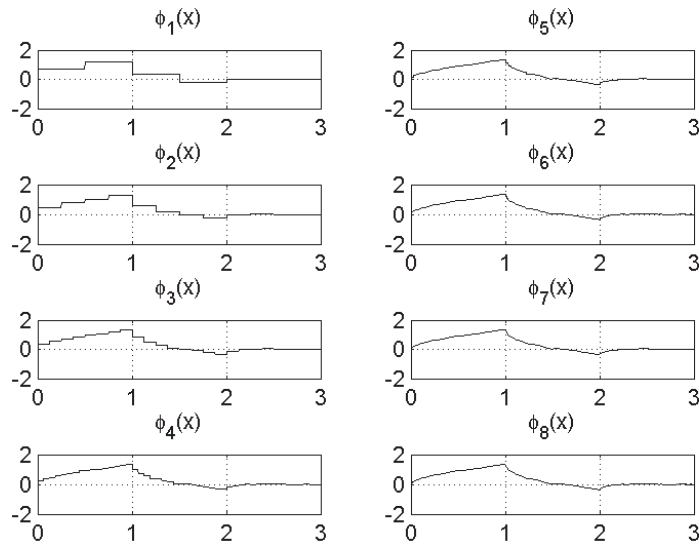


Figura 4.7 - Oito iterações da função escala D4.

Eq. (4.3), utilizando quatro coeficientes:

$$[\phi_2] = \begin{bmatrix} c_0 \\ c_1 \\ c_2 & c_0 \\ c_3 & c_1 \\ & c_2 & c_0 \\ & c_3 & c_1 \\ & & c_2 & c_0 \\ & & c_3 & c_1 \\ & & & c_2 \\ & & & c_3 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} [1] = M_2 M_1 [1], \quad (4.3)$$

onde M_r denota uma matriz genérica ($r = 1, 2, \dots$) onde cada coluna possui uma submatriz dos coeficientes (matriz M_1) posicionada duas posições abaixo da submatriz à sua esquerda. O número $[1]$ mostrado na Eq. (4.3) equivale à amplitude da função que, neste caso, é unitária. A implementação deste esquema de multiplicações de matrizes é crucial na Transformada *Wavelet* discreta. A iteração não é o método mais eficiente para gerar as funções escala ϕ , mas é simples e de fácil programação. Além disso, para *wavelets* ortogonais (escolhas particulares dos coeficientes c_k), a

matriz M_r possui propriedades especiais essenciais para o funcionamento da Transformada *Wavelet* discreta. A forma final da equação de dilatação para quatro coeficientes é mostrada na Fig. 4.8.

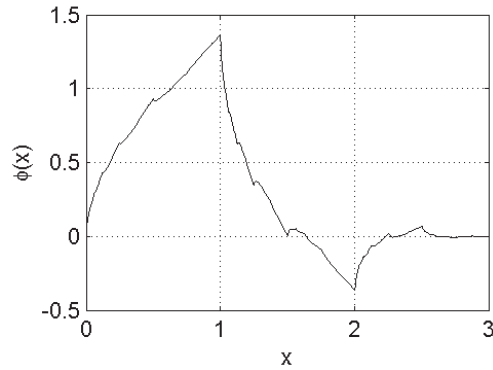


Figura 4.8 - Forma final da função escala D4.

4.3 Wavelets de Dilatação

A wavelet $W(x)$, também chamada de *wavelet*-mãe ou *wavelet* de dilatação é construída a partir da função escala correspondente, mostrada na Seção 4.2. Para a função escala definida na Eq. (4.1), utilizando quatro coeficientes, a *wavelet* de dilatação correspondente é:

$$W(x) = -c_3\phi(2x) + c_2\phi(2x - 1) - c_1\phi(2x - 2) + c_0\phi(2x - 3). \quad (4.4)$$

A Eq. (4.5) mostra a forma geral da *wavelet* de dilatação.

$$W(x) = \sum_{k=0}^{N-1} (-1)^k c_k \phi(2x + k - N + 1). \quad (4.5)$$

Os coeficientes utilizados são os mesmos usados para a equação de dilatação. Entretanto, para esta função, são usados em ordem reversa e com termos alternados tendo seus sinais trocados. Fazendo-se os cálculos da Eq. (4.4) para a função $\phi(x)$ definida na seção anterior e mostrada na Fig. 4.8, obtém-se o gráfico mostrado na Fig. 4.9.

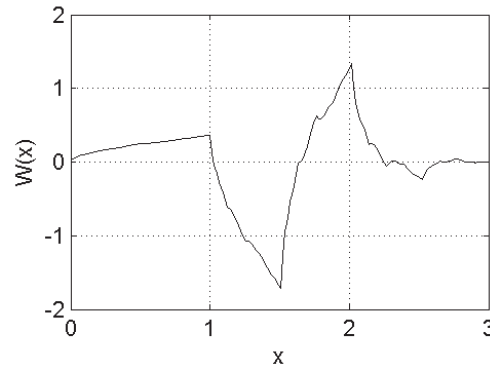


Figura 4.9 - *Wavelet* D4 a partir da função escala da Fig. 4.8.

Esta é a *wavelet* D4 usada para a decomposição da onda quadrada da Fig. 4.3, mostrada na Fig. 4.4. Considerando-se as operações matriciais no desenvolvimento de $\phi(x)$ pelo método iterativo e supondo que a função esteja completamente desenvolvida, isto é, o critério de parada foi atingido, pode-se gerar $W(x)$ do exemplo utilizado na seção anterior através da Eq. (4.4). Matricialmente para a segunda iteração, utilizando quatro coeficientes, tem-se:

$$[W_2] = \begin{bmatrix} c_0 \\ c_1 \\ c_2 & c_0 \\ c_3 & c_1 \\ & c_2 & c_0 \\ & c_3 & c_1 \\ & & c_2 & c_0 \\ & & & c_3 \end{bmatrix} \begin{bmatrix} -c_3 \\ c_2 \\ -c_1 \\ c_0 \end{bmatrix} [1] = M_2 G_1 [1]. \quad (4.6)$$

A Fig. 4.10 mostra as oito primeiras iterações no desenvolvimento da *wavelet* D4, através das operações matriciais.

É importante notar que a diferença entre as operações para calcular a equação de dilatação e a *wavelet* de dilatação está na matriz M_1 , que para o cálculo da *wavelet* de dilatação é substituída pela matriz G_1 :

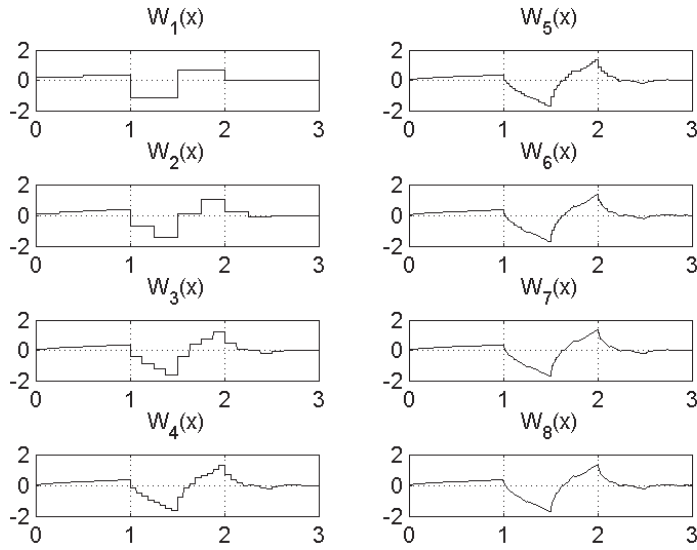


Figura 4.10 - *Wavelet* D4 calculada por iteração.

$$G_1 = \begin{bmatrix} -c_3 \\ c_2 \\ -c_1 \\ c_0 \end{bmatrix}. \quad (4.7)$$

4.4 Transformada *Wavelet* Circular

Segundo Newland (1993), para ter um algoritmo da Transformada *Wavelet* discreta, é conveniente limitar o intervalo da variável independente x a um intervalo unitário, para que $f(x)$ esteja definida apenas para $0 \leq x < 1$.

Deve-se, portanto, considerar uma expansão *wavelet* que esteja dentro do intervalo $0 \leq x < 1$, pois algumas *wavelets* $W(2^j x - k)$ sobrepõem uma ou ambas as extremidades do intervalo.

Para evitar os problemas criados por esta sobreposição, assume-se que $f(x)$ é um período de um sinal periódico no intervalo $0 \leq x < 1$ e repetido nos intervalos unitários adjacentes.

Usando como exemplo a *wavelet* D4, mostrada na Fig. 4.9, pode-se notar que $W(x)$ ocupa três intervalos unitários, $0 \leq x < 3$. No intervalo $0 \leq x < 1$, $f(x)$ receberá contribuições do primeiro terço de $W(x)$, do segundo terço de $W(x + 1)$ e do terceiro terço de $W(x + 2)$. É como se $W(x)$ envolvesse o intervalo unitário. Dessa forma, quando qualquer *wavelet* sair do intervalo $0 \leq x < 1$, deve ser interpretada como "envolvida" no intervalo, quantas vezes forem necessárias.

4.5 Transformada *Wavelet* Discreta

Na Transformada *Wavelet* discreta (DWT), assume-se que $f(x)$ seja um período de um sinal periódico, como dito anteriormente. O algoritmo da DWT foi desenvolvido por Mallat (1989), e por isso é chamado Algoritmo Piramidal de Mallat. O resultado final da aplicação deste algoritmo é o vetor $a = [a_0 \ a_1 \ a_2 \ a_3 \ \dots \ a_{2^j+k} \ \dots]$, contendo os coeficientes da expansão *wavelet*, ou seja, as amplitudes de cada *wavelet* decomposta do sinal original. O termo a_0 , por exemplo, é a amplitude da função escala $\phi(x)$. Como nas seções anteriores, o cálculo desse vetor depende apenas dos coeficientes *wavelet* tabelados, c_k . É importante notar que o número de elementos no vetor a é o mesmo do sinal f . As matrizes M e G , definidas na seção anterior, sofrem uma pequena mudança neste algoritmo. Para que se encaixem em um intervalo unitário, as duas últimas linhas devem ser somadas às duas primeiras. Esta operação é definida como o "envolvimento", brevemente explicado na Seção 4.4. A matriz M_1 , mostrada na Eq. (4.3), torna-se:

$$M_1 = \begin{bmatrix} c_0 + c_2 \\ c_1 + c_3 \end{bmatrix}. \quad (4.8)$$

Por conta das condições impostas aos coeficientes para assegurar que as *wavelets* e suas funções escala fossem todas mutuamente ortogonais, tem-se que as matrizes M_r são ortogonais. Assim:

$$\begin{aligned}
\frac{1}{2}M_r^t M_r &= I, \\
G_r^t M_r &= 0, \\
M_r^t G_r &= 0, \\
\frac{1}{2}G_r^t G_r &= I.
\end{aligned}
\tag{4.9}$$

Para melhor demonstrar estas operações matriciais, definem-se novos símbolos para as transpostas de G e M como H e L , respectivamente. Tem-se, a partir da Eq. (4.9):

$$\begin{aligned}
\frac{1}{2}LM &= I, \\
HM &= 0, \\
LG &= 0, \\
\frac{1}{2}HG &= I.
\end{aligned}
\tag{4.10}$$

Finalmente, pelo Algoritmo Piramidal de Mallat, é possível definir o vetor a a partir do sinal f . Assim, tem-se:

$$\begin{aligned}
a_1 &= \frac{1}{2}H_1 \frac{1}{2}L_2 \frac{1}{2}L_3 f, \\
\begin{bmatrix} a_2 \\ a_3 \end{bmatrix} &= \frac{1}{2}H_2 \frac{1}{2}L_3 f, \\
\begin{bmatrix} a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} &= \frac{1}{2}H_3 f.
\end{aligned}
\tag{4.11}$$

Representando em forma de diagrama, tem-se:

$$\begin{array}{ccccccc}
 f'(1) & \xleftarrow{\frac{1}{2}L_1} & f'(1:2) & \xleftarrow{\frac{1}{2}L_2} & f'(1:4) & \xleftarrow{\frac{1}{2}L_3} & f'(1:8) = f \\
 \downarrow & & \frac{1}{2}H_1 \downarrow & & \frac{1}{2}H_2 \downarrow & & \frac{1}{2}H_3 \downarrow \\
 a = [& a_0 & a_1 & a_{2:3} & a_{4:7} &] &
 \end{array} \tag{4.12}$$

A notação L para a matriz transposta de M foi usada pois esta matriz pode ser interpretada como um filtro passa baixa (*low-pass filter*) e a notação H para a matriz transposta de G por ser interpretada como um filtro passa alta (*high-pass filter*). Em seu livro, Newland (1993) mostra um importante exemplo para entendimento desta teoria.

4.6 Mapa de Quadrados Médios

A expansão do sinal pode ser dada por:

$$f(x) = a_0\phi(x) + \sum_j \sum_k a_{2^j+k} W(2^j x - k), \quad 0 \leq x < 1. \tag{4.13}$$

O inteiro j descreve os diferentes níveis das *wavelets* começando em $j = 0$ e o inteiro k cobre o número de *wavelets* em cada nível, cobrindo o intervalo de 0 a $2^j - 1$. Pode-se calcular o valor quadrado médio de $f(x)$ elevando os dois lados da Eq. (4.13) ao quadrado e integrando no intervalo unitário $0 \leq x < 1$. Segundo Newland (1993), tem-se que:

$$\int_0^1 W(2^j x - k) dx = 0 \tag{4.14}$$

e, a não ser que $r = j$ e $s = k$:

$$\int_0^1 W(2^j x - k) W(2^r x - s) dx = 0 \tag{4.15}$$

Dessa forma, chega-se a uma equação cujos primeiros termos são:

$$\int_0^1 f^2(x)dx = a_0^2 + a_1^2 + \frac{1}{2}(a_2^2 + a_3^2) + \frac{1}{4}(a_4^2 + a_5^2 + a_6^2 + a_7^2) + \dots$$

(4.16)

$$\dots + \frac{1}{8}(a_8^2 + a_9^2 + \dots + a_{15}^2) + \frac{1}{16}(a_{16}^2 + \dots + a_{31}^2) + \dots$$

Isto mostra como o quadrado médio de $f(x)$ é distribuído entre os diferentes níveis *wavelet* e entre as diferentes *wavelets* em cada nível. A Eq. (4.16) pode ser ilustrada em um gráfico, com os valores das amplitudes *wavelet* quadradas, como mostra a Fig. 4.11. Esta representação em cores é mostrada da mesma forma como na STFT, mostrada no Capítulo 3.

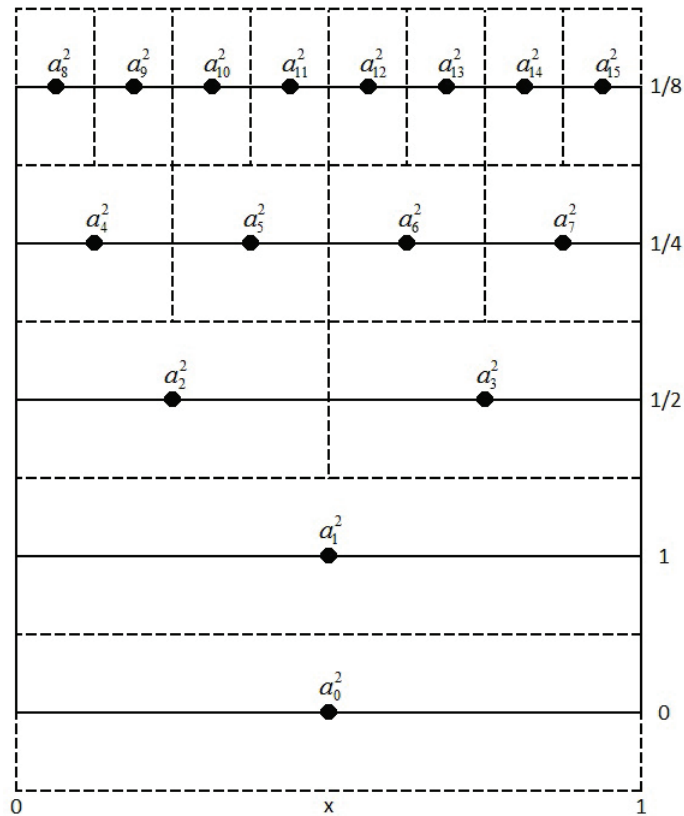


Figura 4.11 - Base para as amplitudes *wavelet* quadradas.

No nível j , existem 2^j *wavelets* em cada intervalo unitário. Para um trem de *wavelets* unitárias, todas no mesmo nível, estas são espaçadas $1/2^j$ umas das outras, sendo a frequência fundamental 2^j ciclos por intervalo unitário. Dessa forma, o nível *wavelet* é uma medida de frequência onde a frequência fundamental dobra conforme se passa de um nível para o próximo. Uma vez

que, para registros de vibração, a variável independente x geralmente é medida de tempo, o mapa de quadrados médios pode ser interpretado como um gráfico de frequência (crescendo em oitavas) pelo tempo.

Um exemplo simples, mas que mostra de forma eficiente as características da aplicação do método da Transformada *Wavelet* é apresentado a seguir. A Fig. 4.12 mostra o Mapa de Quadrados Médios resultante da aplicação do método em uma senóide de frequência 1024 Hz.

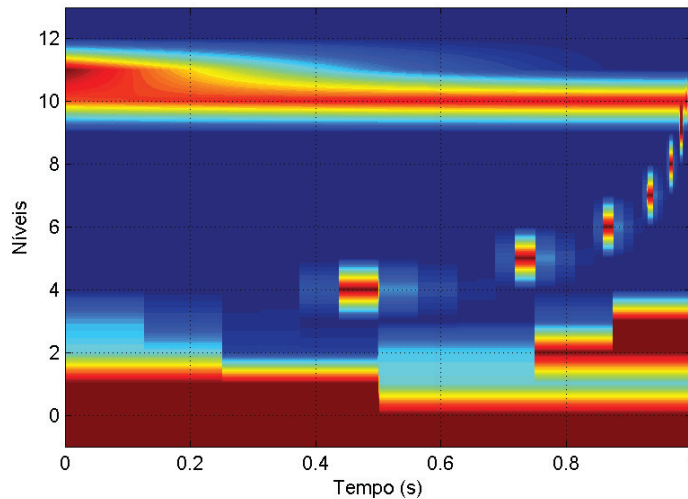


Figura 4.12 - Mapa de Quadrados Médios de uma senóide ($f = 1024$ Hz).

Nota-se uma intensidade das cores constante no nível de decomposição 10. Um sinal com as mesmas características, porém com frequência de 512 Hz, apresenta um Mapa de Quadrados Médios semelhante, como mostra a Fig. 4.13.

Houve uma mudança do nível 10 de decomposição para o nível 9. Mostra-se, assim, que mudanças no nível de decomposição do sinal significam mudanças na frequência. Assim como definido anteriormente, diminui-se um nível conforme a frequência diminui pela metade.

Mudanças na amplitude do sinal correspondem a mudanças nas cores do mapa de quadrados médios, ou seja, na intensidade das *wavelets* em cada nível. A Fig. 4.14 mostra um sinal onde a primeira metade apresenta menor amplitude que a segunda metade.

Assim como a STFT, este método apresenta informações de tempo, tornando-se muito im-

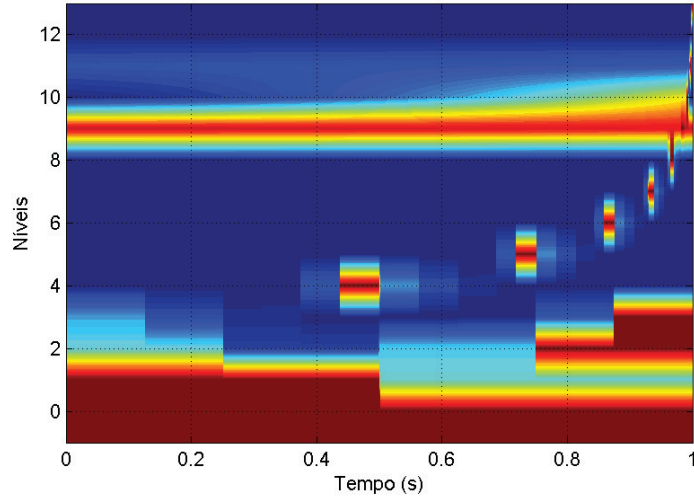


Figura 4.13 - Mapa de Quadrados Médios de uma senóide ($f = 512$ Hz).

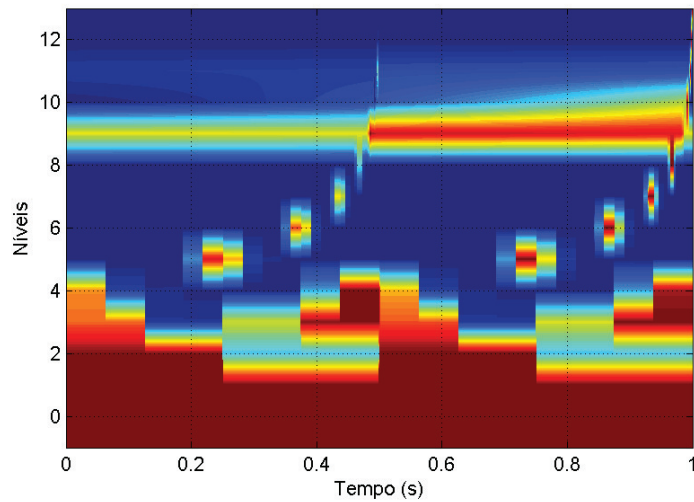


Figura 4.14 - Mapa de Quadrados Médios de uma senóide com amplitudes diferentes ($f = 512$ Hz).

portante para a análise dos sinais de monitoramento do processo FSW. Além disso a utilização dos diferentes níveis no processamento do sinal permite com que este seja analisado de maneira geral, ou seja, de maneira ampla ou de maneira detalhada, possibilitando a identificação das falhas.

5 VALIDAÇÃO DAS TÉCNICAS DE DETECÇÃO

5.1 Introdução

Com a criação de uma interface gráfica (Fig. 5.1), foi possível juntar as teorias apresentadas nos Capítulos 3 e 4 em apenas um programa, de maneira a facilitar a comparação e compreensão dos resultados dados pelos mapas de cores. O Anexo A apresenta os algoritmos de todos os programas desenvolvidos para que a interface possa ser utilizada.

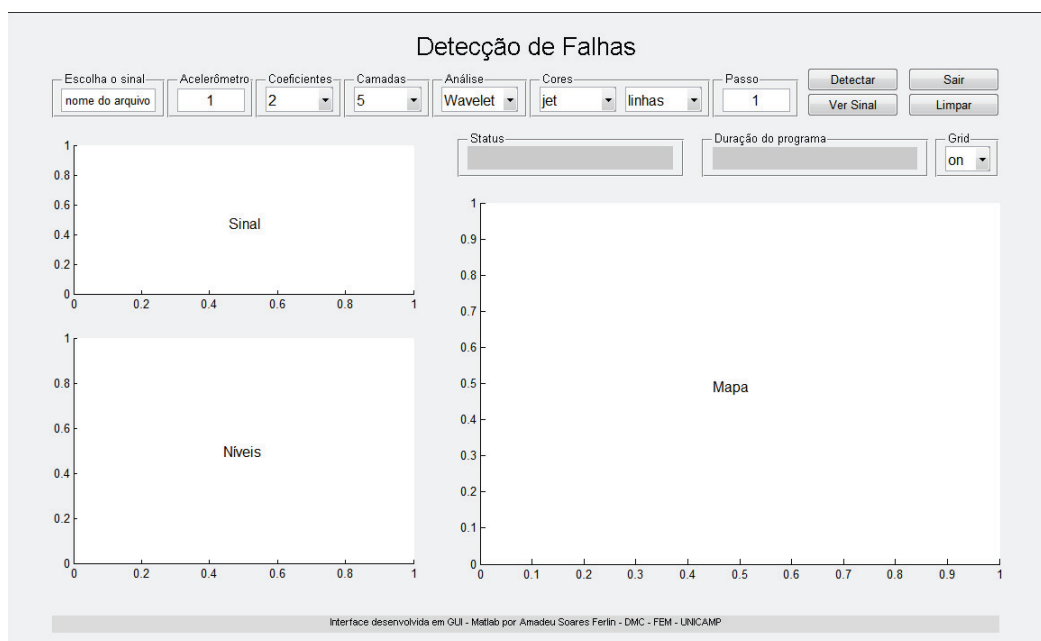


Figura 5.1 - Interface gráfica criada em GUI (Matlab[®]) para análise dos sinais.

O primeiro passo para a utilização do programa é inserir o nome do arquivo cujos dados do processo foram gravados no campo “Escolha o sinal”. Se durante o processo de aquisição de dados foi usado mais de um acelerômetro, é possível selecionar no campo “Acelerômetro” qual deles se deseja analisar o sinal. Características como o número de coeficientes *wavelet* e o método de análise são escolhidas, respectivamente, nos campos “Coeficientes” e “Análise” e então dadas como entrada para o programa. A análise feita através da STFT apresenta como resultado apenas uma imagem, disposta no espaço de maiores dimensões (denominado “Mapa”, na Fig. 5.1). A outra

opção de análise, feita através da Transformada *Wavelet*, apresenta duas imagens como resultado. Uma delas é o Mapa de Quadrados Médios, apresentado no espaço denominado “Mapa”. No espaço denominado “Níveis”, aparecem os diferentes níveis de decomposição do sinal. Para facilitar as análises neste documento, os gráficos serão apresentados fora da interface gráfica.

5.2 Simulação da vibração de um sistema rotativo

De maneira a entender como os resultados são mostrados pelo programa e validá-los, criou-se um sinal com as mesmas características do sinal utilizado por Moraes (1996), exemplificando a vibração mecânica de um sistema rotativo.

Este sinal é composto por três componentes harmônicas, de frequências 100, 60 e 25 Hz, ponderadas por 1,0, 0,4 e 0,6, respectivamente. A amplitude da terceira componente foi modulada por uma frequência de 12 Hz. Para deixá-lo mais próximo de um sinal real, acrescentou-se um ruído branco normalizado, com nível RMS=0,01 e desvio padrão $s=0,1$. A frequência de amostragem deste sinal é de 4096 Hz.

Para conclusões mais concretas, é importante que haja uma comparação entre um sinal normal, ou seja, sem falhas, e um sinal com falhas. Dessa forma, foram utilizados dois sinais. O primeiro (Fig. 5.2a) é exatamente como sugere a descrição acima. O segundo (Fig. 5.2b) apresenta falhas, adicionadas a ele em forma de pulsos. As falhas adicionadas aparecem em $L/3$, $L/2$ e $3L/4$, onde L é o comprimento do sinal. No sinal da Fig. 5.2b, estes pulsos possuem um nível RMS=0,1 .

É importante lembrar que estas falhas foram criadas com o intuito de validar o método, seguindo o trabalho de Moraes (1996). As falhas encontradas em um cordão de solda por FSW não apresentam, necessariamente, este tipo de padrão ou comportamento.

Visualmente não é possível detectar qualquer falha ou diferença entre os sinais mostrados na Fig. 5.2. Desta forma, para uma verificação mais detalhada, serão aplicados os diferentes métodos para análise das respostas apresentadas para esta situação.

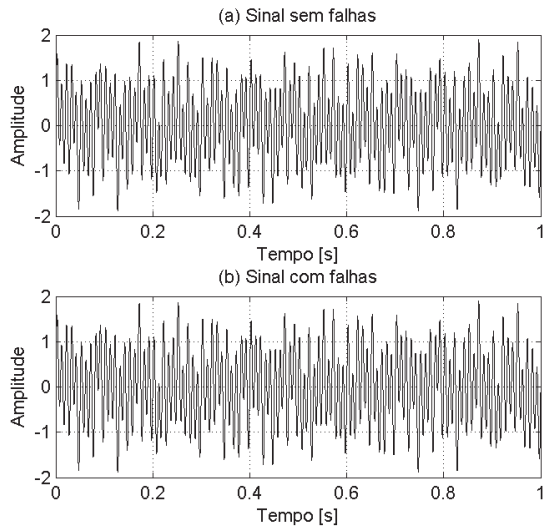


Figura 5.2 - Sinais (a) sem falhas e (b) com falhas.

5.2.1 Transformada de Fourier de Tempo Curto (STFT)

A primeira análise feita nos sinais foi a aplicação da FFT para verificação das frequências que os compõem. Na Fig. 5.3, para ambos os sinais, aparecem as frequências de 100 Hz, 60 Hz e outras duas, resultantes da modulação da frequência de 25 Hz pela frequência de 12 Hz. Nota-se que nenhuma outra frequência que possa ser associada aos pulsos adicionados como falhas aparece na análise.

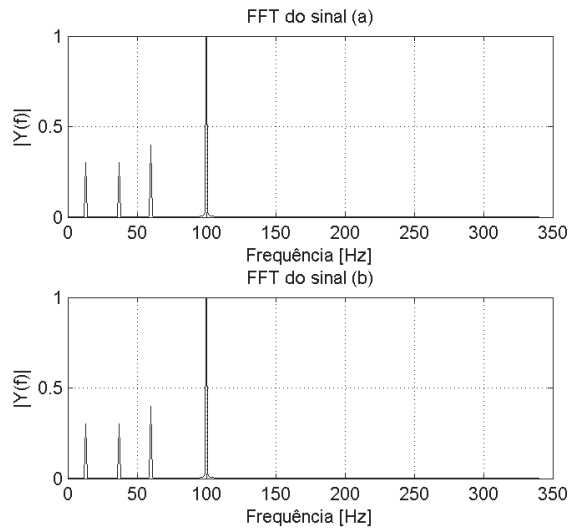


Figura 5.3 - FFTs dos sinais (a) sem falhas e (b) com falhas.

Como visto no Capítulo 3, a FFT é uma boa ferramenta de análise para uma visão geral do sinal. Entretanto, para mostrar pequenas mudanças e informações do instante em que ocorreram estas mudanças, a STFT é mais completa.

Para aplicar a STFT, deve-se definir a janela a ser utilizada e a porcentagem de *overlap* desejada. Para este capítulo, em uma análise comparativa, duas situações diferentes foram consideradas. A primeira situação utilizou janela retangular, sem *overlap*. A segunda, janela Hanning e 50% de *overlap*.

Situação 1: janela retangular e 0% de *overlap*

A Fig. 5.4 mostra a divisão de um sinal em 5 intervalos, sem *overlap*, multiplicados pela função janela retangular, apenas para exemplificar estas operações.

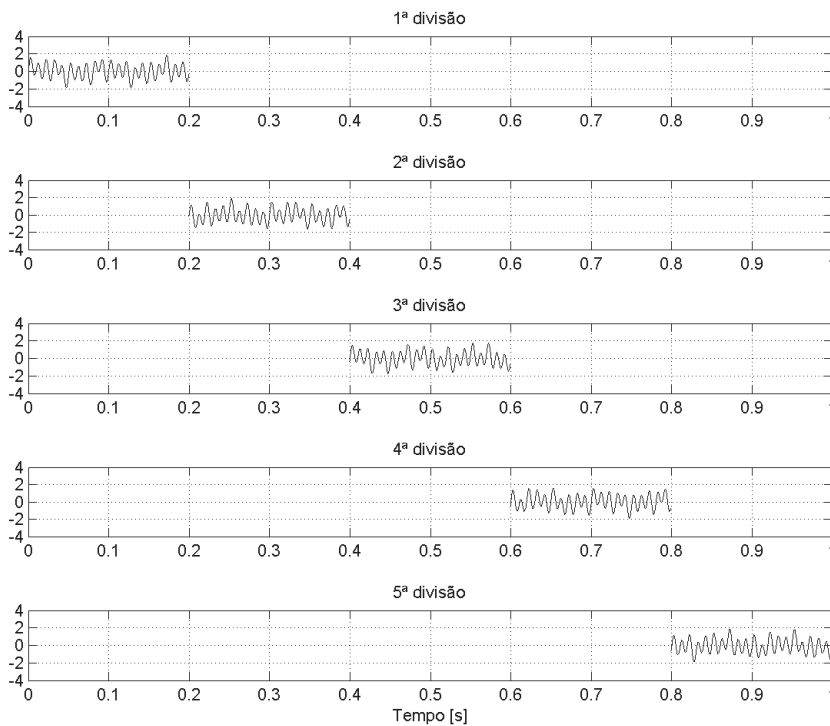


Figura 5.4 - Divisões de um sinal sem *overlap* e com janela retangular.

Para a situação da análise em questão, o sinal foi dividido em 35 intervalos e a cada um deles foi aplicada uma janela retangular. Com uma frequência de amostragem de 4096 Hz e período de 1 segundo, o sinal apresenta $L = 4096$ pontos no total, o que dá $l = 117$ pontos por intervalo. Com as definições de janela e *overlap* utilizadas nesta situação, para um nível RMS=0,1 para os pulsos, não foi possível detectar falha alguma comparando-se os dois mapas, mostrados na Fig. 5.5.

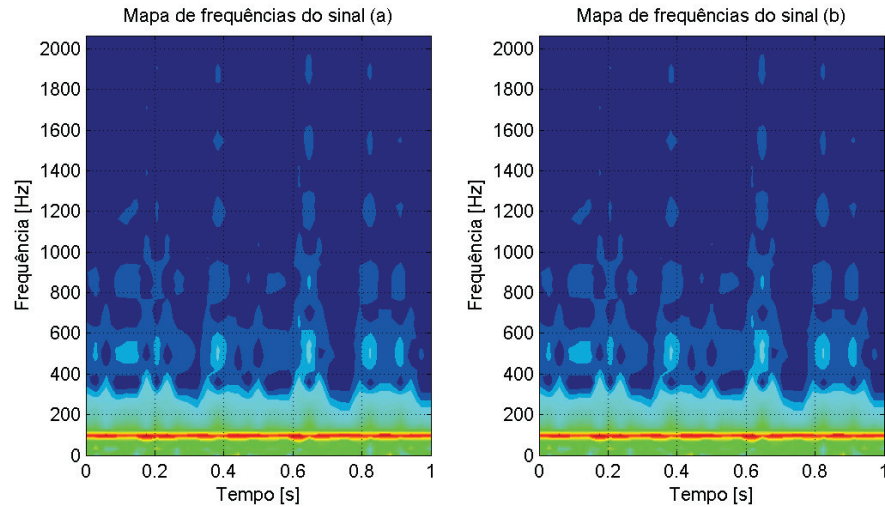


Figura 5.5 - Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,1, janela retangular, 0% *overlap*).

Aumentando, então, o nível RMS das falhas para 0,5, pôde-se notar pequenas diferenças entre os mapas de cores (Fig. 5.6).

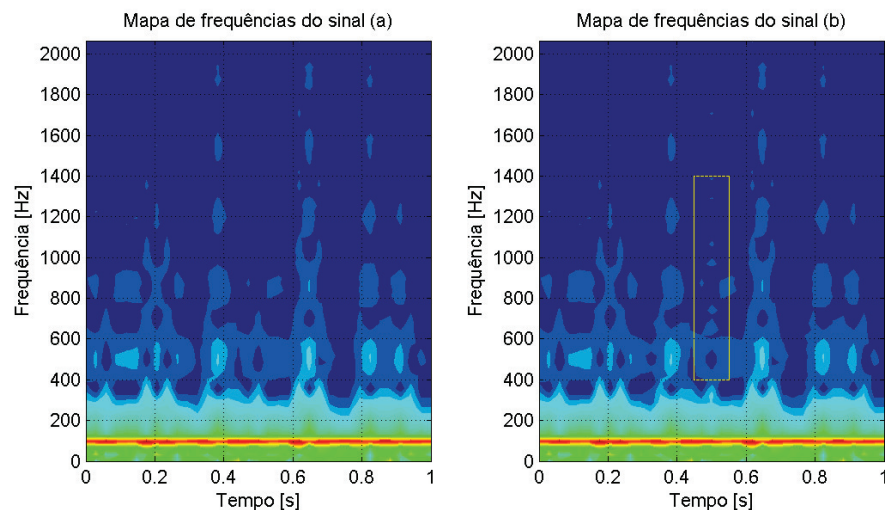


Figura 5.6 - Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,5, janela retangular, 0% *overlap*).

No entanto, estas falhas são detectadas somente se houver uma comparação entre os mapas de cores de um sinal sem falhas e um sinal com falhas, ou seja, é necessário seguir uma referência.

As três falhas são detectadas no mapa somente quando o nível RMS é superior a 1,1, como mostra a Fig. 5.7. Ainda assim é necessário um sinal de referência.

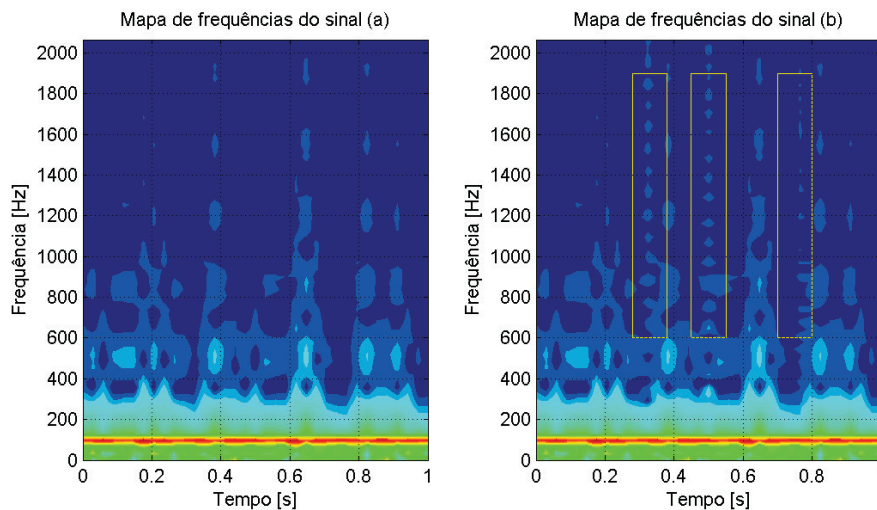


Figura 5.7 - Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=1,1, janela retangular, 0% *overlap*).

Situação 2: janela Hanning e 50% de *overlap*

Nesta situação o sinal em questão também foi dividido em 35 intervalos e, a cada um deles, foi aplicada uma janela Hanning. Como o *overlap* foi de 50%, o número de intervalos passou a ser de 69 ($2n - 1$, onde n é o número de intervalos escolhido). A Fig. 5.8 mostra a divisão do sinal em 5 intervalos, com *overlap* de 50%, multiplicados pela função janela Hanning, apenas para exemplificar estas operações.

Como a frequência de amostragem e o período do sinal não mudaram, cada intervalo apresenta o mesmo número de pontos da situação anterior. Com as definições de janela e *overlap* utilizadas nesta situação, para um nível RMS=0,1 para os pulsos, também não foi possível detectar falha alguma comparando-se os dois mapas mostrados na Fig. 5.9.

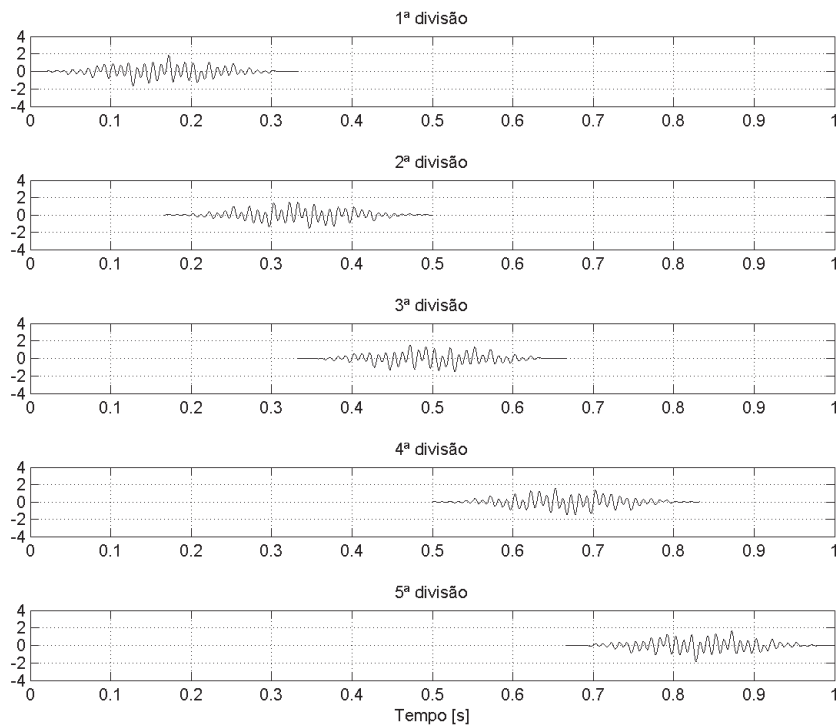


Figura 5.8 - Divisões do sinal com 50% de *overlap* e janela Hanning.

Aumentando o valor do nível RMS das falhas para 0,5, é possível notar algumas pequenas diferenças (Fig. 5.10). Diferentemente da situação anterior, é possível detectar uma falha sem a comparação com algum sinal que não tenha falhas. Entretanto as mudanças são tão pequenas que o resultado pode não ser tão confiável quanto o desejado.

Quando o nível RMS=0,9, torna-se possível detectar as três falhas do sinal. Porém, quando este nível é superior a 1,0, que foi o valor utilizado para os resultados da Fig. 5.11, as conclusões ficam realmente boas e é possível dizer com maior certeza a localização das falhas no sinal.

A utilização da janela Hanning, uma das mais utilizadas neste tipo de análise, e de 50% de *overlap* fez com que o mapa de cores da STFT ficasse mais “limpo” e melhorou a localização temporal das falhas. No entanto, pulsos com os níveis RMS detectados para este método são falhas de tamanhos consideráveis, chegando a cerca de 30% da amplitude do sinal. A Fig. 5.12 mostra, em um *zoom* do sinal na posição $L/2$, a amplitude da falha em relação à amplitude do sinal.

Dessa forma, buscam-se métodos que possam dar resultados satisfatórios para mudanças com

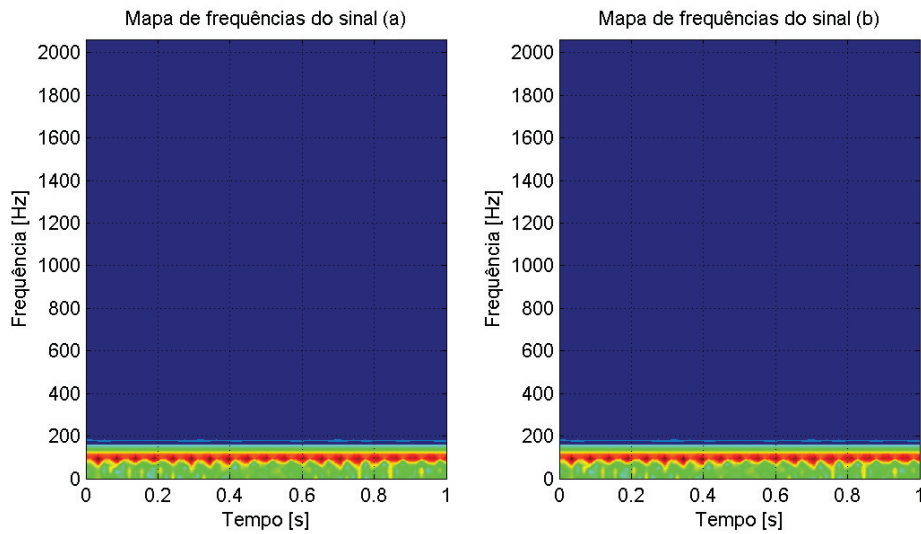


Figura 5.9 - Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,1; janela Hanning, 50% *overlap*).

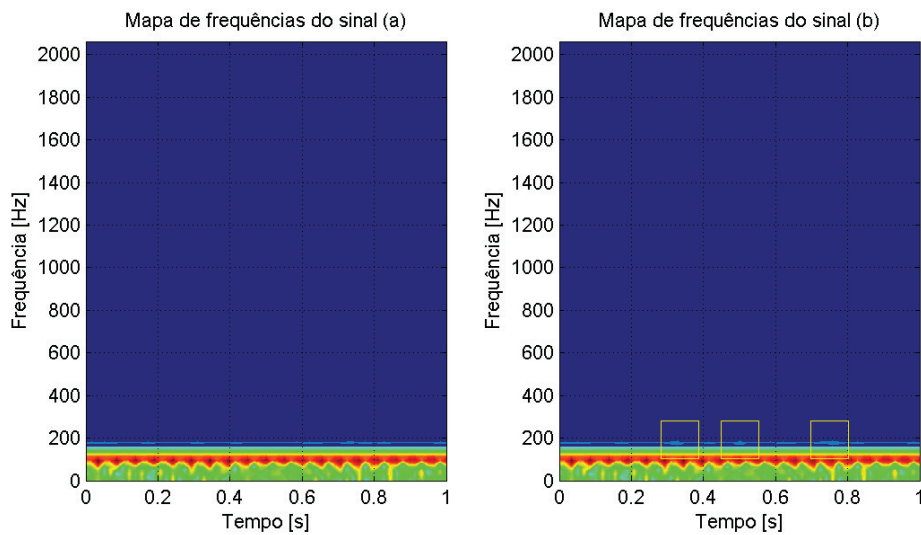


Figura 5.10 - Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=0,5; janela Hanning, 50% *overlap*).

amplitudes ainda menores. Considera-se a Transformada *Wavelet* uma boa opção para se avaliar esta condição.

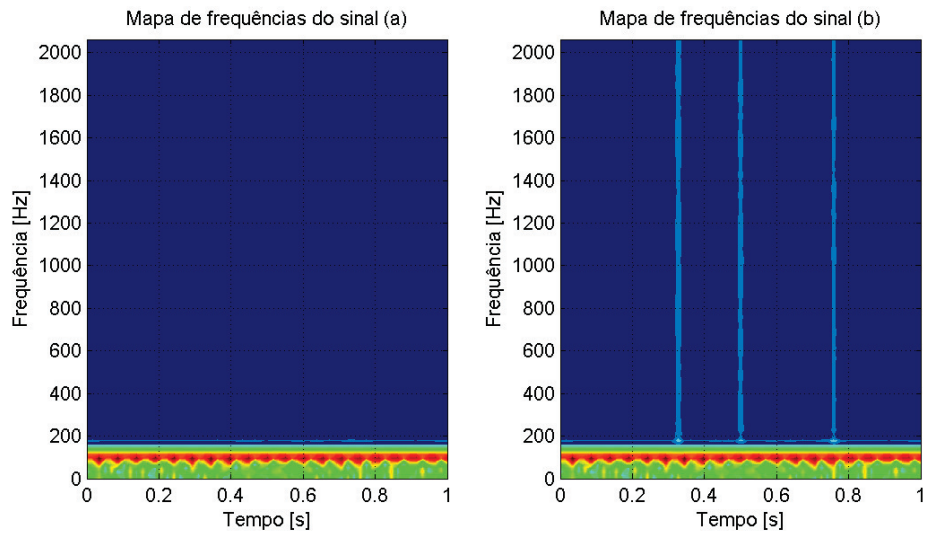


Figura 5.11 - Mapas de cores dos sinais (a) sem falhas e (b) com falhas (RMS=1,0; janela Hanning, 50% *overlap*).

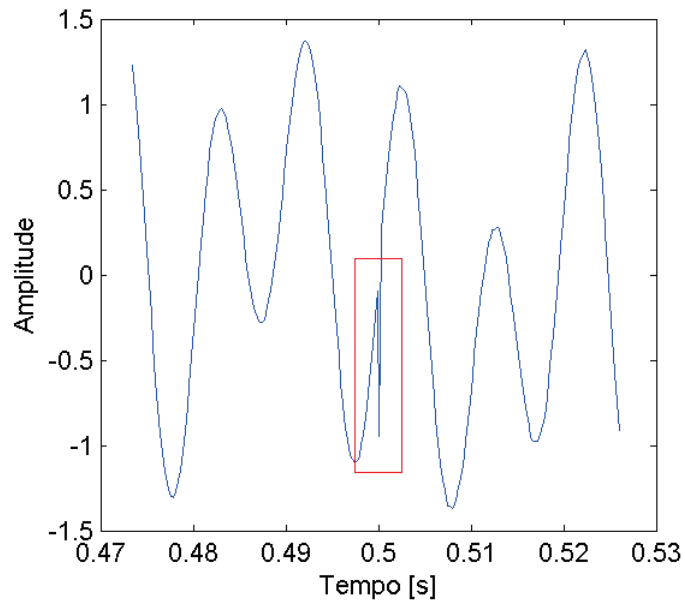


Figura 5.12 - *Zoom* da falha em $L/2$.

5.2.2 Transformada *Wavelet*

A análise feita através da Transformada *Wavelet* apresenta, como resultado, duas imagens. Uma delas denomina-se Mapa de Quadrados Médios. A outra, apresenta os níveis de decomposição do sinal. A utilização dos dois gráficos na mesma análise proporciona uma maior confiança na avaliação dos resultados.

O número de coeficientes *wavelet* utilizado na análise é escolhido pelo usuário. De forma a criar uma análise comparativa, serão utilizados quatro valores diferentes para o número de coeficientes: 2, 8, 12 e 20.

Com a utilização de 2 coeficientes *wavelet* e nível RMS=0,1, apenas uma falha (na posição $3L/4$) foi detectada no gráfico dos níveis de decomposição (Fig. 5.13b). É muito provável que esta falha também tenha sido apresentada no mapa de quadrados médios da Fig. 5.14b. Porém, com poucos coeficientes, o mapa apresenta muita variação nos valores, ficando com muitas cores e difícil de se analisar.

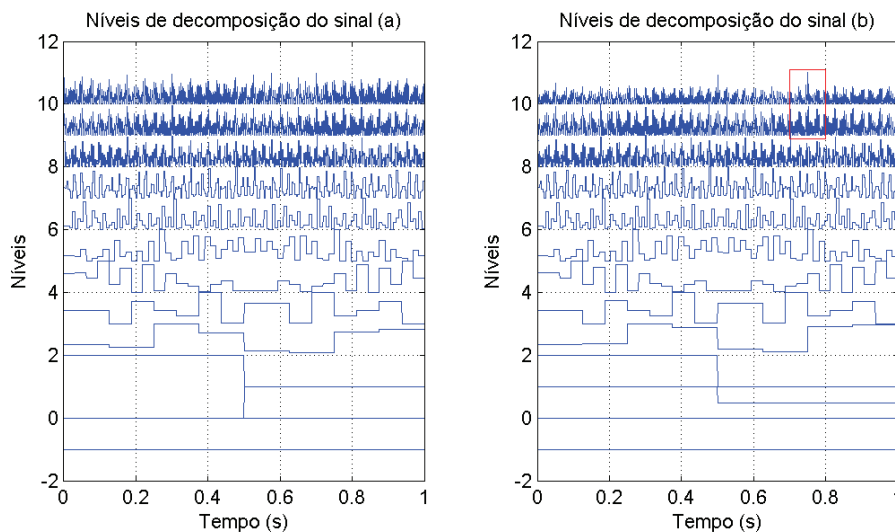


Figura 5.13 - Níveis de decomposição (2 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

Utilizando 8 coeficientes *wavelet* e nível RMS=0,1, as três falhas foram detectadas tanto no gráfico dos níveis de decomposição (Fig. 5.15b) quanto no Mapa de Quadrados Médios (Fig. 5.16b), com bastante clareza. Assim como a análise pela STFT utilizando janela Hanning e 50% de

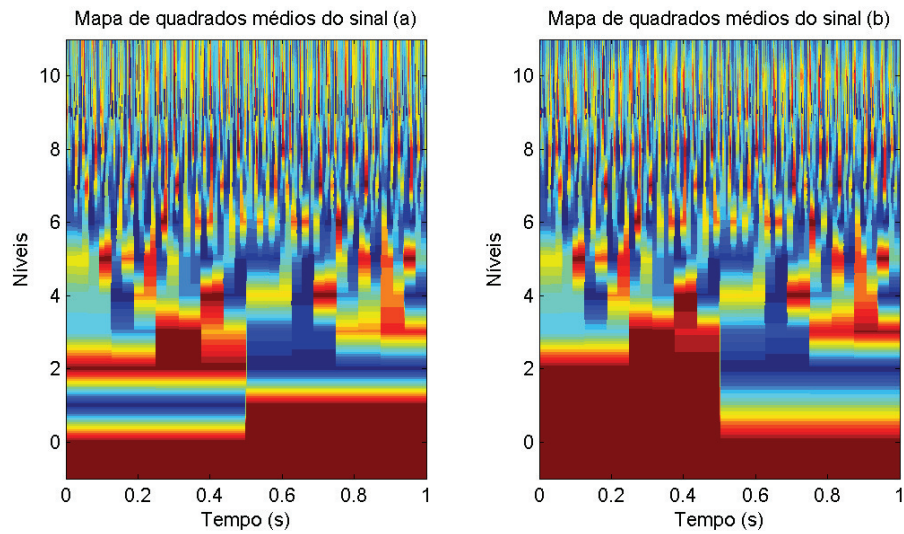


Figura 5.14 - Mapas de Quadrados Médios (2 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

overlap (Seção 5.2.1), esta situação também não necessita de comparação do sinal analisado com um sinal de referência.

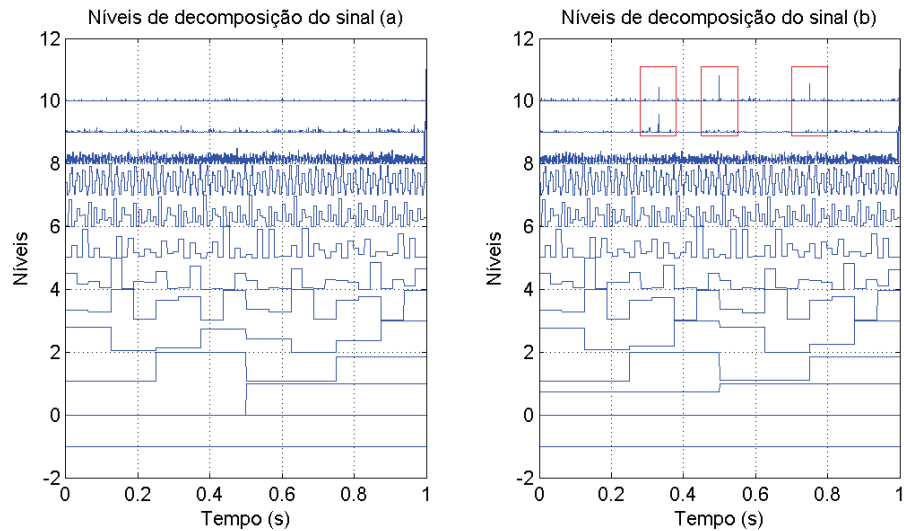


Figura 5.15 - Níveis de decomposição (8 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

Com 12 coeficientes *wavelet*, as três falhas com nível $RMS=0,1$ foram detectadas e mostradas mais claramente que para 8 coeficientes. Diminuindo o nível RMS dos pulsos para $0,08$, a escolha deste número de coeficientes fez com que a detecção das três falhas fosse um pouco difícil, como mostra as Figs. 5.17 e 5.18.

Utilizando 20 coeficientes *wavelet*, as três falhas com nível $RMS=0,08$ foram detectadas e

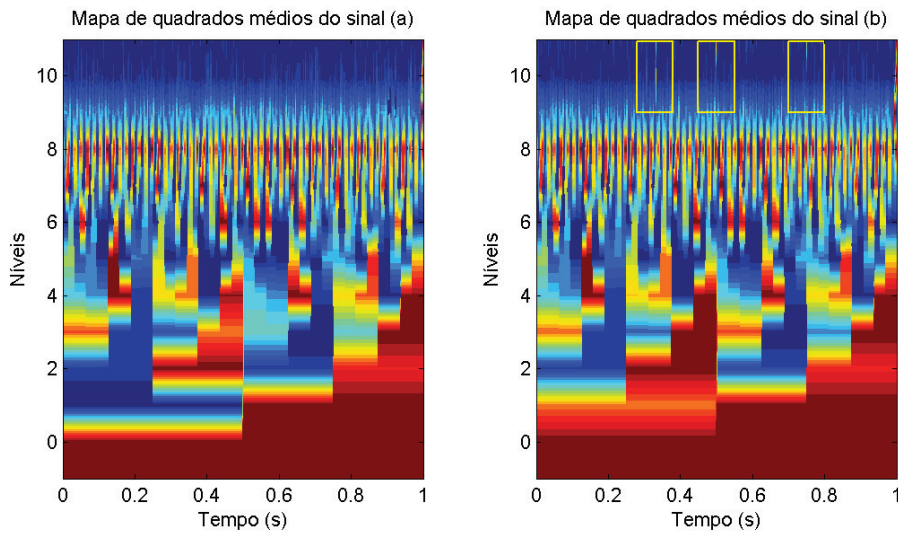


Figura 5.16 - Mapas de Quadrados Médios (8 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

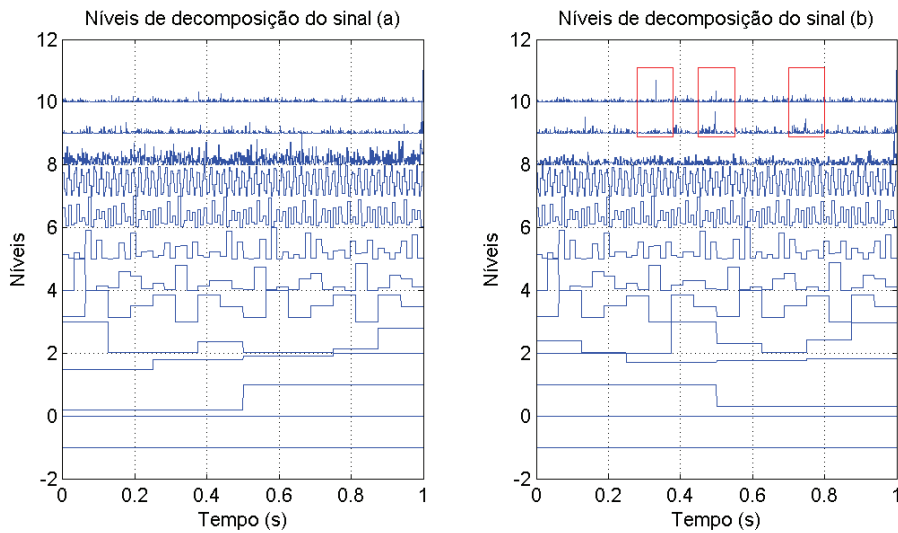


Figura 5.17 - Níveis de decomposição (12 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

mostradas mais nitidamente que para 12 coeficientes, como mostram as Figs. 5.19 e 5.20.

Na comparação dos gráficos dos níveis da decomposição para os casos analisados é possível notar que as mudanças, neste caso, ocorrem sempre nos níveis mais altos. Como mostrado no Capítulo 4, os níveis mais baixos analisam as frequências mais baixas, enquanto que os níveis mais altos analisam as frequências mais altas. Como as falhas são na forma de pulsos, possuem altas frequências e, por esta razão, aparecem nos níveis mais altos da decomposição.

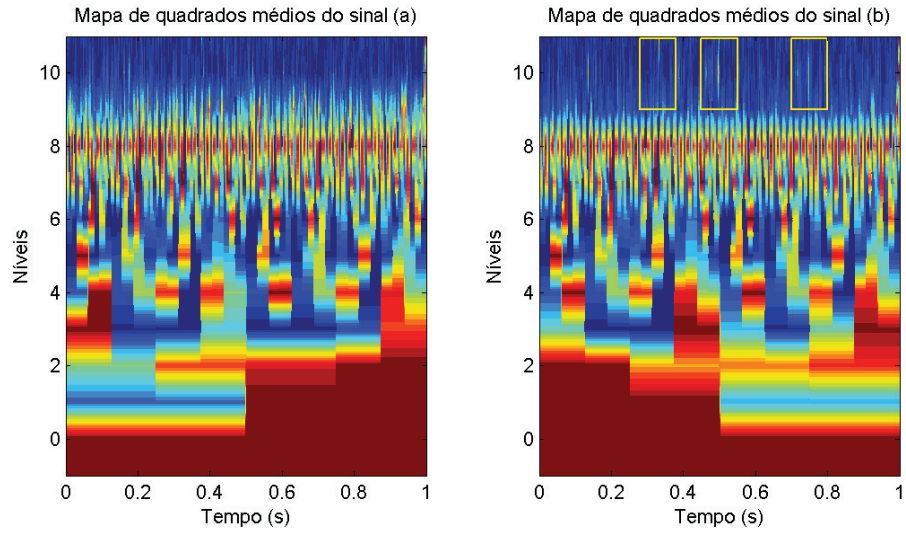


Figura 5.18 - Mapas de Quadrados Médios (12 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

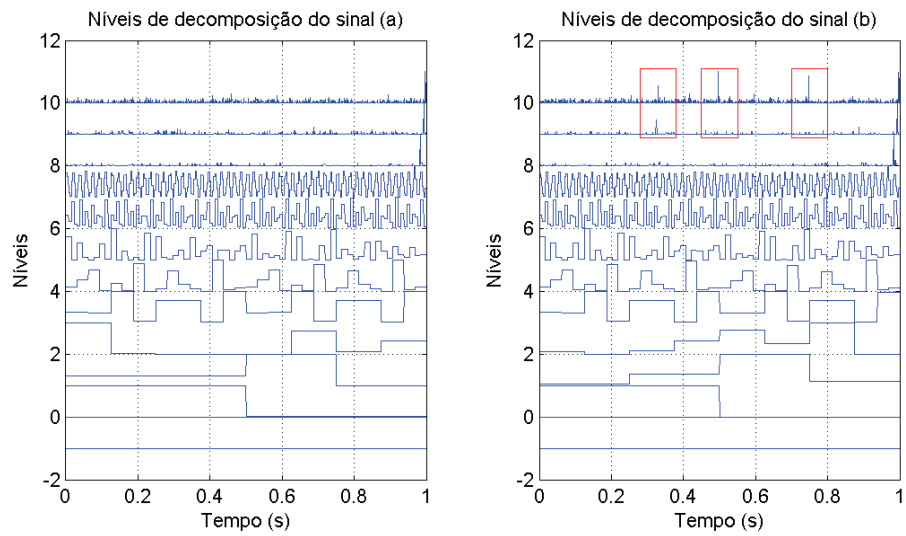


Figura 5.19 - Níveis de decomposição (20 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

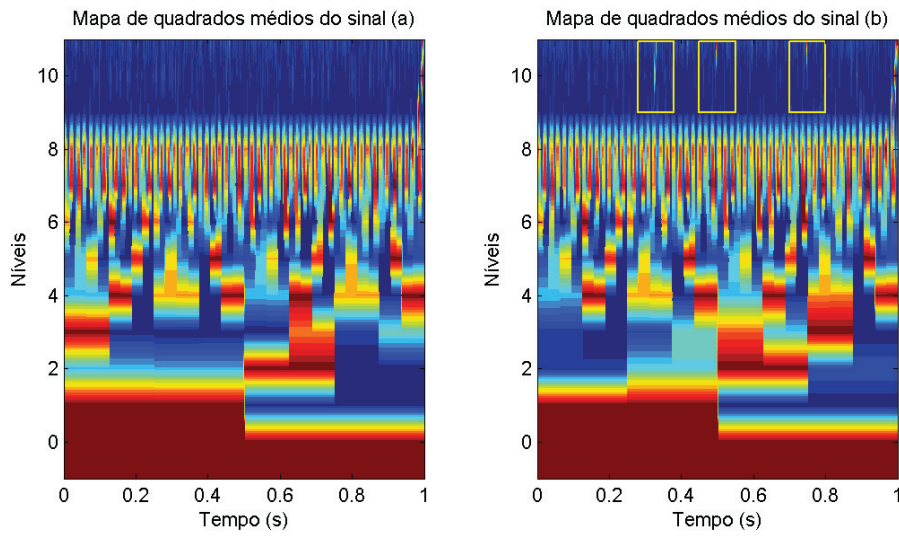


Figura 5.20 - Mapas de Quadrados Médios (20 coeficientes) dos sinais (a) sem falhas e (b) com falhas.

5.3 Teste com medição de som

Para verificar as respostas dadas pelos métodos a um sinal real, fez-se um teste onde o sinal adquirido é de origem sonora. Através do atrito de um microfone comum com uma superfície com diferentes rugosidades, esquematizada na Fig. 5.21, foi possível obter um sinal variante no tempo, mostrado na Fig. 5.22.

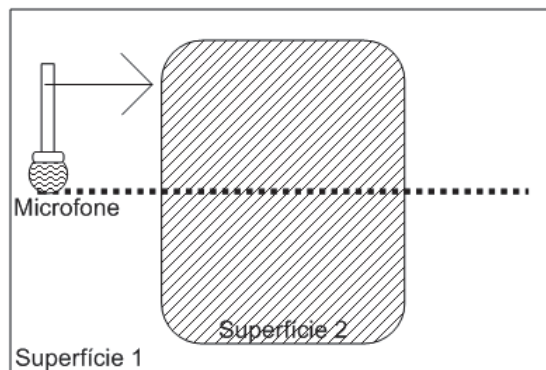


Figura 5.21 - Superfícies diferentes para atrito do microfone.

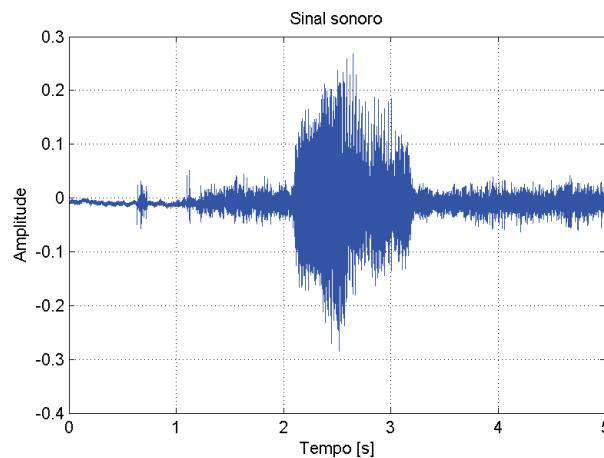


Figura 5.22 - Sinal sonoro adquirido.

Uma análise visual do sinal permite a verificação de mudanças na amplitude no intervalo de 2,2s a 3,2s. No entanto, é desejável analisar quais as características dos resultados dados pela Transformada de Fourier de Tempo Curto (STFT) e pela Transformada *Wavelet*. Aplicando-se a STFT ao sinal mostrado na Fig. 5.22, tem-se o mapa de cores apresentado na Fig. 5.23.

Nota-se, no intervalo de 2,2s a 3,2s da Fig. 5.23, um aumento na intensidade da frequência

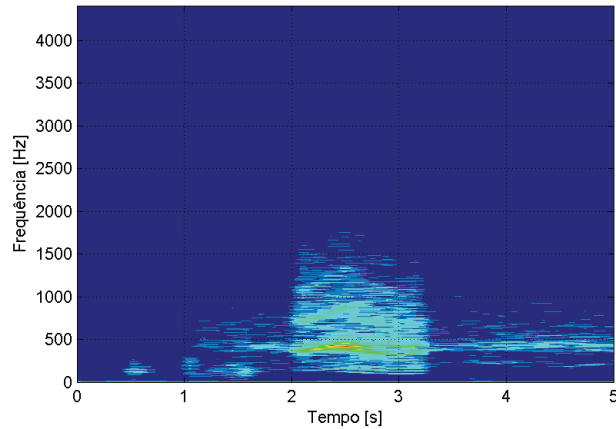


Figura 5.23 - STFT do sinal sonoro adquirido.

de 500 Hz, caracterizado pela mudança de cores do mapa. Além disso, houve o aparecimento de frequências não existentes anteriormente, caracterizado pela nuvem de frequências até 1500 Hz. Estas características são suficientes para apontar uma mudança no comportamento do sinal neste intervalo, podendo significar uma falha em um sinal de monitoramento.

Aplicando-se a Transformada *Wavelet* ao sinal da Fig. 5.22, tem-se resultados semelhantes. No intervalo de 2,2s a 3,2s, há um aumento na intensidade das *wavelets* do nível 10 ao nível 16, como se pode ver na Fig. 5.24. Além disso, é possível notar que as pequenas variações de amplitude em 0,7s e 1,2s também foram detectadas pelo Mapa de Quadrados Médios.

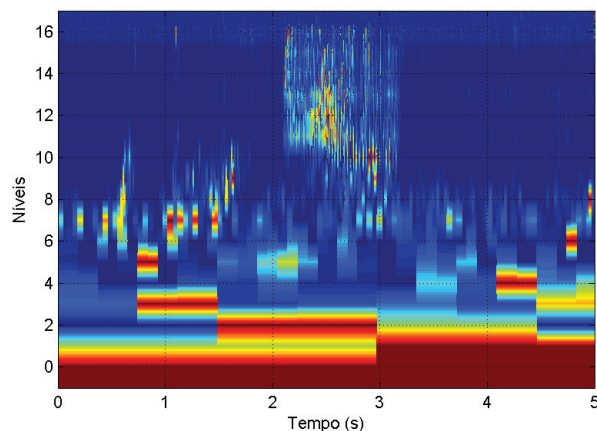


Figura 5.24 - Mapa de Quadrados Médios do sinal sonoro adquirido.

5.4 Conclusões do capítulo

Ambos os métodos apresentados foram capazes de detectar as falhas introduzidas no sinal de teste e as mudanças no sinal sonoro real. Embora a utilização de uma janela mais adequada (janela Hanning) e do *overlap* de 50% terem melhorado a apresentação dos mapas de cores e diminuído a amplitude necessária (nível RMS) para que a falha fosse detectada no sinal de teste, o método da STFT não demonstrou um bom desempenho. Como dito anteriormente, para que haja a detecção utilizando este método, as falhas devem ter amplitudes de aproximadamente 30% das amplitudes do sinal.

Nos sinais reais, as análises foram eficientes em ambos os métodos. As mudanças no sinal são perceptíveis visualmente. No entanto, a utilização dos métodos neste sinal teve como intuito a verificação do tipo de resposta dado pelos mapas de cores e de quadrados médios.

Pelo fato de o método da Transformada *Wavelet* ter detectado falhas muito menores (inclusive menores que as falhas propostas inicialmente, na Fig. 5.2), acredita-se que este seja um bom método para detecção de falhas no processo de junção por FSW.

6 RESULTADOS EXPERIMENTAIS

6.1 Introdução

Os métodos implementados foram testados com sinais simulados e apresentaram bons resultados. A etapa final deste trabalho é aplicar estes métodos na detecção de falhas em ensaios reais, ou seja, monitorando o processo de união por FSW.

Os ensaios foram realizados na oficina mecânica do Departamento de Mecânica Computacional (DMC) da Faculdade de Engenharia Mecânica (FEM) da Universidade Estadual de Campinas (Unicamp). Os primeiros testes foram feitos com o intuito de adequar os parâmetros do processo como velocidade de avanço (~ 30 mm/min), velocidade de rotação (1100 RPM) e inclinação da ferramenta ($0,5^\circ$).

6.2 Materiais

A máquina utilizada neste processo foi uma fresadora comum, com uma ferramenta simples para o processo FSW, confeccionada na Unicamp (Fig. 6.1), cujas dimensões são apresentadas na Tabela 6.1. A ferramenta foi inclinada em $0,5^\circ$ para que, ao se deslocar, seu ombro não retirasse material da junção das chapas.

As chapas de alumínio utilizadas (Fig. 6.2), cujas dimensões são mostradas na Tabela 6.1, foram preparadas antes de serem unidas por FSW. A preparação consiste em fresar e retirar as rebarbas das laterais que serão unidas, pois o processo exige que as chapas estejam em contato e que não haja espaços entre elas.

A fixação das chapas na mesa da máquina deve ser firme e em vários pontos, para evitar empenamentos devido à força que a ferramenta aplica na direção do avanço. A Fig. 6.3 mostra

Tabela 6.1 - Dimensões da ferramenta e das chapas.

Dimensões da Ferramenta		
Comprimento do Pino	Diâmetro do Pino	Diâmetro do Ombro
1,5 mm	5,0 mm	8,0 mm

Dimensões das Chapas		
Espessura	Largura	Comprimento
1,6 mm	50,0 mm	300,0 mm



Figura 6.1 - Ferramenta utilizada nos ensaios.

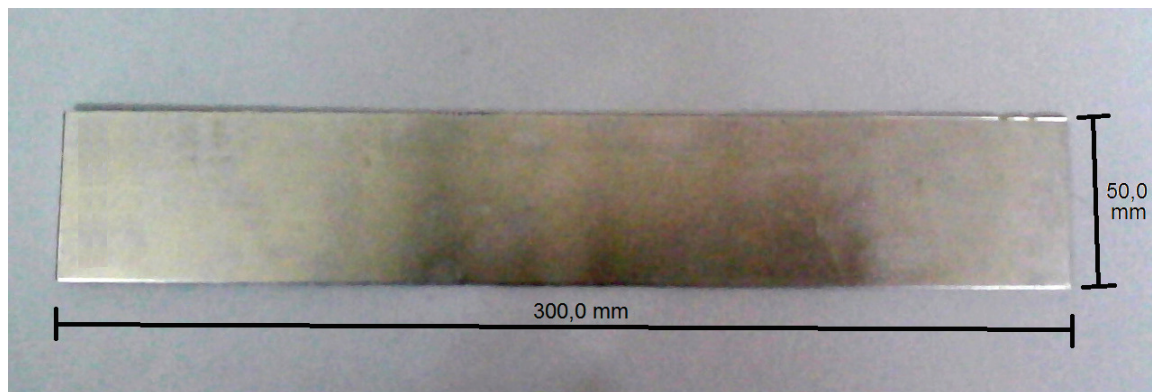


Figura 6.2 - Uma das chapas utilizadas nos ensaios.

como as chapas foram fixadas no ensaio realizado. É importante, também, que a linha de união das duas chapas esteja localizada sobre um apoio liso.

Para adquirir os sinais, foram utilizados três acelerômetros PCB Piezotronics modelo PCB 353B68 (Tabela 6.2), sendo um deles posicionado na barra fixadora (acelerômetro 1) e os outros dois posicionados nas chapas (acelerômetros 2 e 3), como mostra a Fig. 6.4.

Tabela 6.2 - Características dos acelerômetros utilizados

Modelo	Sensibilidade
PCB 353B68	98,5930 m/s ² /v
PCB 353B68	101,6580 m/s ² /v
PCB 353B68	100,2043 m/s ² /v

Estes acelerômetros foram ligados em um aparelho LMS SCADAS SCR05, que é um sistema compatível com diversos tipos de sensores e é integrado a um programa da própria empresa (LMS Test.Lab) capaz de realizar algumas análises aos sinais adquiridos. Através deste sistema, os sinais foram adquiridos, processados e enviados ao computador para salvá-los em arquivo para que, só então, pudessem ser pós-processados pelos programas desenvolvidos.

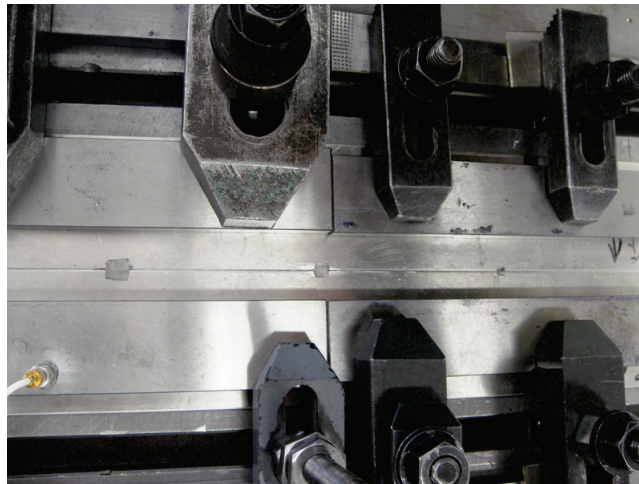


Figura 6.3 - Fixação das chapas à mesa da máquina.

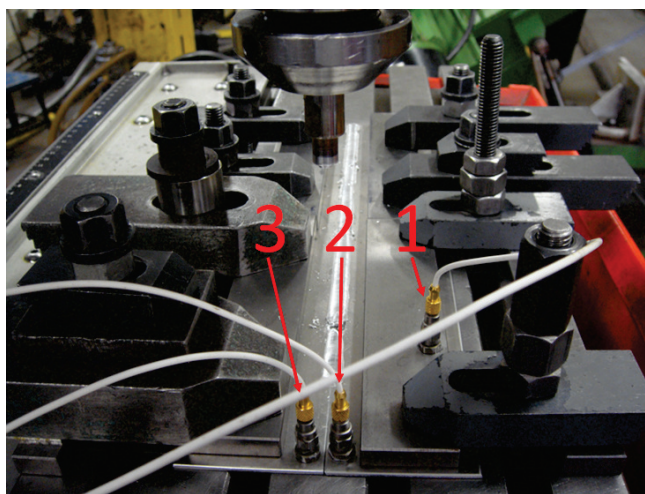


Figura 6.4 - Posicionamento dos acelerômetros 1, 2 e 3 nas chapas.

6.3 Testes

6.3.1 Ensaio 1: determinação dos parâmetros

O primeiro teste realizado foi feito utilizando um par de chapas sem falhas. Estas chapas foram devidamente preparadas e fixadas à máquina. Os acelerômetros não foram utilizados, pois como dito anteriormente, o motivo deste ensaio era apenas de fixar parâmetros para o processo, como velocidade de avanço, rotação e inclinação da ferramenta.

A Fig. 6.5 mostra a chapa soldada neste ensaio. As marcações "i" e "f" simbolizam o início e o fim do processo, respectivamente. A rotação da ferramenta foi no sentido horário, o que caracteriza a parte inferior do cordão, na figura, como avanço (onde a velocidade de rotação e translação se somam) e a parte superior como retrocesso (onde a velocidade de rotação é contrária à velocidade de translação).



Figura 6.5 - Chapas unidas no ensaio 1.

Pela qualidade superficial do cordão, considerou-se o ensaio como válido, adotando os parâmetros utilizados como referência para o segundo ensaio, sendo: velocidade de avanço $v_f \approx 30 \text{ mm/min}$, velocidade de rotação $\omega = 1100 \text{ RPM}$ e inclinação da ferramenta de $0,5^\circ$.

6.3.2 Ensaio 2: detecção e localização de falhas

Neste ensaio foram feitas três falhas em ambas as chapas, caracterizando vazios no material. As falhas eram quadradas e cada uma delas possuía um tamanho diferente. A primeira tinha lado 4,0 mm, a segunda 3,0 mm e a terceira 2,0 mm. A Fig. 6.6 mostra estas falhas.

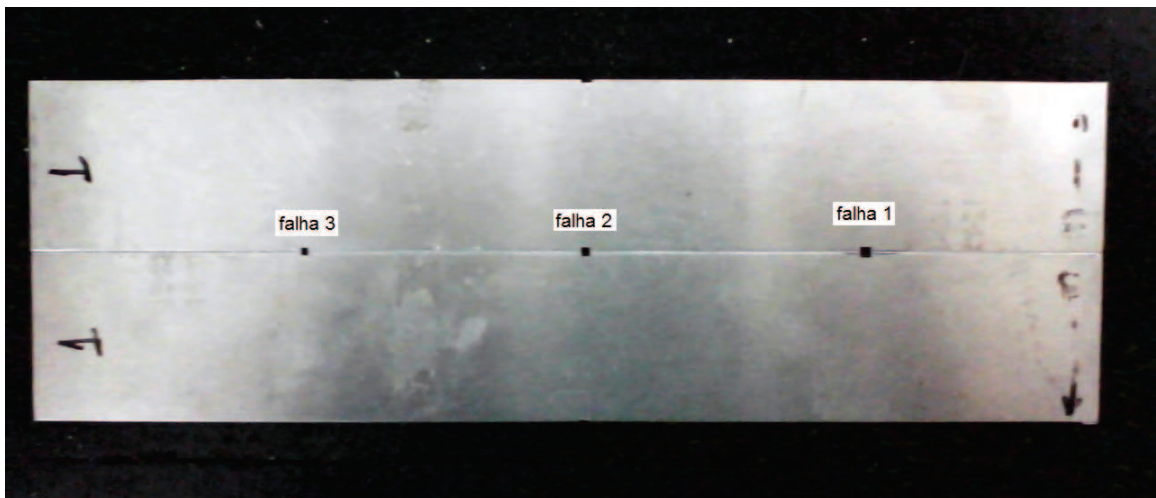


Figura 6.6 - Falhas feitas nas chapas para o ensaio 2.

As laterais em contato foram devidamente preparadas e as chapas devidamente fixadas para o processo. Os acelerômetros foram posicionados da mesma maneira como a mostrada na Fig. 6.4.

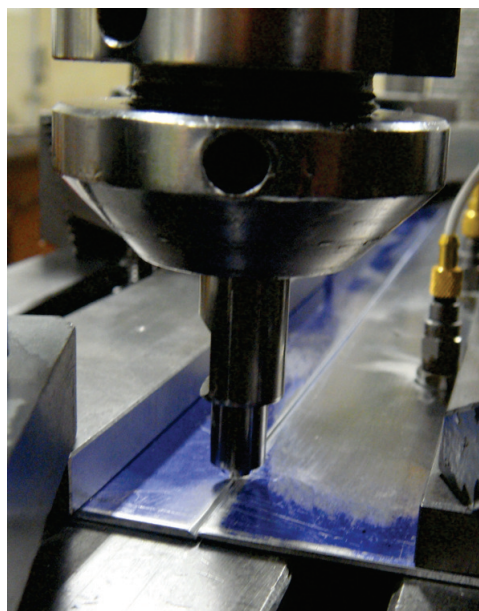


Figura 6.7 - Aproximação da ferramenta.

A ferramenta foi, então, posicionada no início da chapa e seu alinhamento com a linha de junção das chapas foi verificado, como na Fig. 6.7. Ao iniciar o processo, o pino da ferramenta encostou-se à chapa e permaneceu girando até que a chapa atingisse temperatura suficiente para que o pino fosse forçado contra ela e penetrasse a junção.

Considerando-se que o comprimento do pino é de 1,5 mm, percebe-se que a ferramenta não atravessou a chapa, deixando 0,1 mm de material entre o pino e a mesa da máquina. É importante citar que o ombro da ferramenta apenas tangencia a superfície da peça. O avanço da mesa foi ligado e a união começou a ser feita.

A Fig. 6.8 mostra o resultado final deste ensaio. As marcações "i" e "f" simbolizam o início e o fim do processo, respectivamente. A rotação também foi no sentido horário, como na Seção 6.3.1. As marcações 1, 2 e 3 mostram onde estavam a primeira, a segunda e a terceira falha, respectivamente. As marcações superiores (120, 275 e 430) mostram os tempos (em segundos) em que a ferramenta passou por cada falha desde o início da aquisição do sinal.

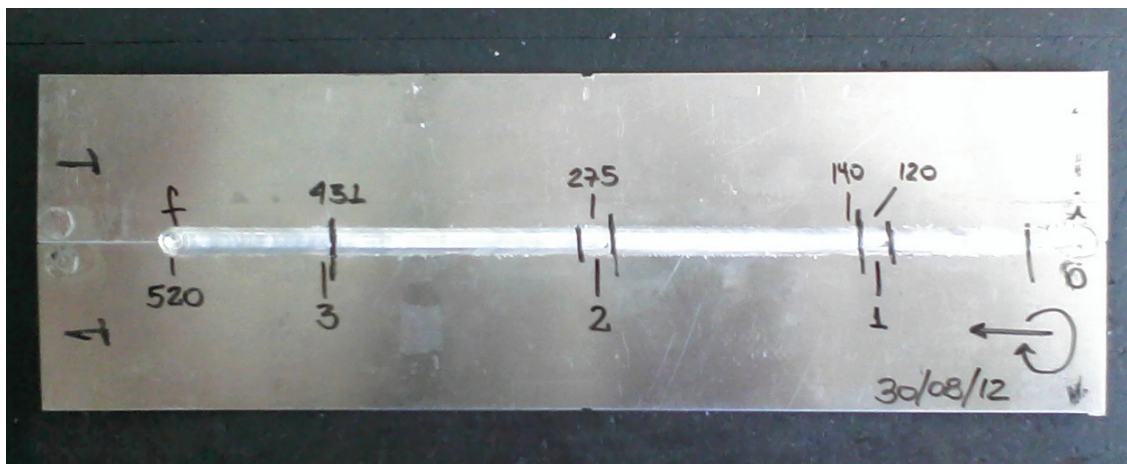


Figura 6.8 - Chapas unidas no ensaio 2.

6.4 Resultados

As chapas unidas no ensaio 2 apresentaram um cordão com acabamento superficial liso e sem falhas, apresentando falta de preenchimento apenas onde havia falta de material, como era

esperado. Os sinais adquiridos por cada acelerômetro possuem pouco menos de $7,35 \cdot 10^6$ pontos e são mostrados na Fig. 6.9.

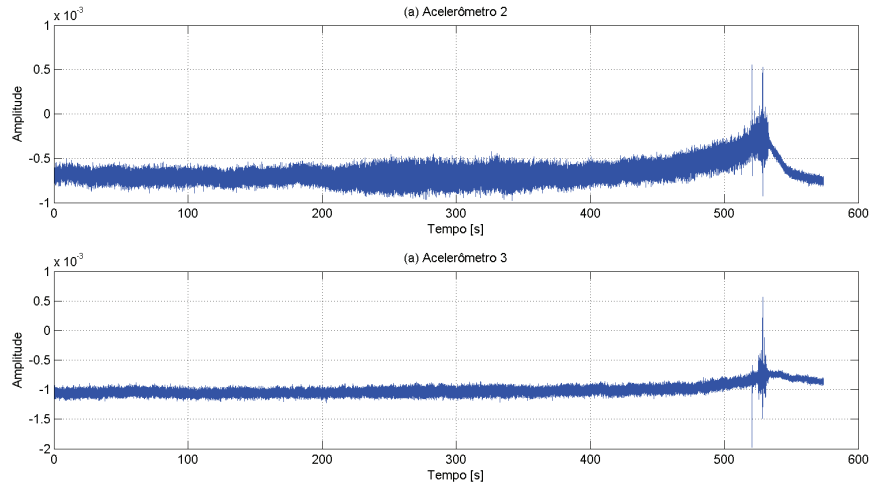


Figura 6.9 - Sinais adquiridos no ensaio.

Analisando visualmente os sinais medidos, não é possível detectar as falhas. Somente o término do processo é evidente nos sinais, com amplitudes maiores, em aproximadamente $t = 520$ s. Desta maneira, os sinais foram armazenados e utilizados nos dois métodos desenvolvidos para o programa.

É importante que os instantes de tempo do início e do término do processo sejam conhecidos. Com a velocidade de avanço da máquina e os valores de tempo é possível localizar a posição da chapa em que a falha foi detectada, como mostra a Eq. (6.1), onde p indica a posição da falha, em mm; v_f a velocidade de avanço, em mm/min; e t o instante de tempo em que a falha foi detectada, em s.

$$p = \frac{v_f t}{60} \quad (6.1)$$

6.4.1 Transformada de Fourier de Tempo Curto (STFT)

O método da STFT foi eficiente na localização das falhas do ensaio realizado. Apesar do acelerômetro localizado na barra fixadora não revelar nenhuma falha e de um dos acelerômetros das chapas apresentar apenas uma falha, um dos acelerômetros mostrou a ocorrência e localização de todas as falhas e dos momentos em que o processo efetivamente começou e terminou.

A Fig. 6.10 mostra os resultados obtidos para os dois acelerômetros que detectaram as falhas (acelerômetros 2 e 3). O acelerômetro 2 detectou de maneira eficiente todas as falhas, como se vê na Fig. 6.10a. A primeira em 120 segundos, a segunda em 275 segundos e a terceira em 430 segundos. O acelerômetro 3 foi capaz de detectar apenas a segunda falha, como visto na Fig. 6.10b no instante 275 segundos.

Os sinais foram divididos em 1000 intervalos cada um. Tem-se, portanto, pouco menos de 7350 pontos por intervalo; um valor que permite a aplicação do método sem perdas de informação. Os resultados mostrados referem-se à análise utilizando janela Hanning e 50% de *overlap*.

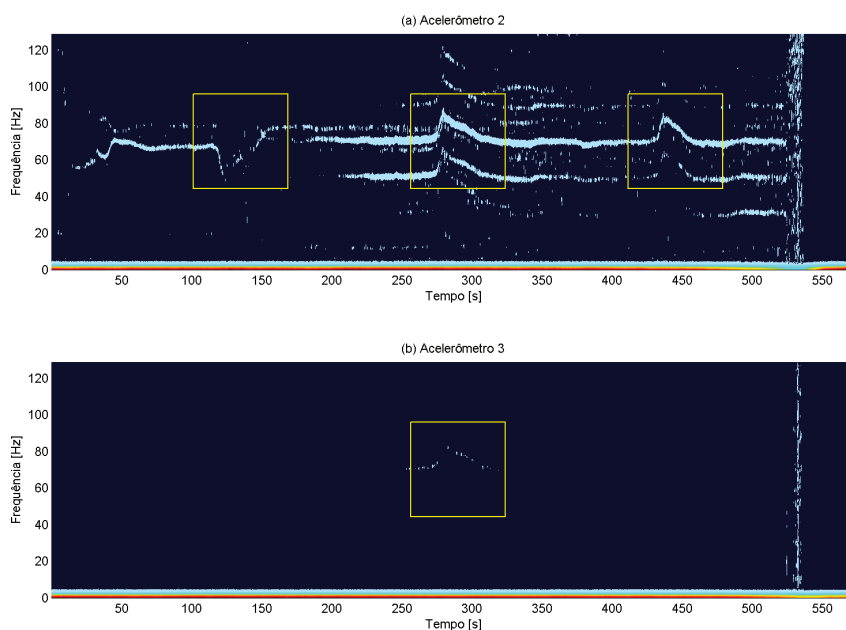


Figura 6.10 - Mapas de cores da STFT dos sinais.

Notam-se variações na frequência em ambos os mapas da Fig. 6.10. O instante de tempo em que cada variação começa refere-se à entrada do ombro da ferramenta na falha. As posições conhecidas das falhas coincidem com as posições das variações mostradas na Fig. 6.10 e, comparando-se com os intervalos sem falha, existem mudanças significativas no comportamento do sinal.

6.4.2 Transformada *Wavelet*

A aplicação da Transformada *Wavelet* também permitiu a localização das falhas no ensaio realizado. A Fig. 6.11 mostra os resultados obtidos para os dois acelerômetros que detectaram as falhas (acelerômetros 2 e 3). Assim como na seção anterior, o acelerômetro 2 detectou todas as falhas, como se vê na Fig. 6.11a. O acelerômetro 3 não foi capaz de detectar as falhas de maneira nítida, como visto na Fig. 6.11b. Para ambos os mapas, utilizou-se 76 coeficientes *wavelet*, já que a utilização de um número menor apresentou resultados menos significativos que o da Fig. 6.11.

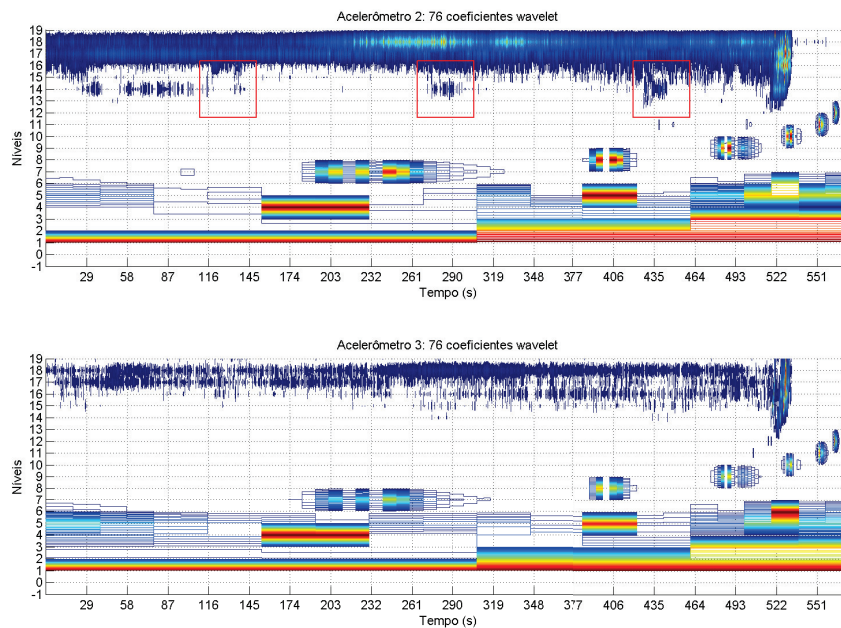


Figura 6.11 - Mapas de Quadrados Médios dos sinais.

6.5 Conclusões do capítulo

Diferentemente dos sinais simulados no Capítulo 5, os sinais adquiridos nos ensaios apresentam muito mais ruído e influência de outras fontes. Dessa forma, os mapas de cores apresentam mais componentes, que dificultam a interpretação dos resultados. Ainda assim os métodos foram capazes de detectar e localizar as falhas criadas nas chapas.

No método da STFT, o mapa de cores obtido para o acelerômetro 2 apresenta todas as informações necessárias para a análise do processo. Analisando-se apenas a Fig. 6.10a é possível determinar os instantes de tempo em que o processo se inicia ($t_i = 45$ s) e termina ($t_f = 520$ s), além de localizar todas as falhas.

O método da Transformada *Wavelet* apresentou as três falhas para o acelerômetro 2, assim como a STFT. O final do processo, em $t_f = 520$ s também é evidente. Mostra-se, assim, que a Transformada *Wavelet* pode ser utilizada no monitoramento do processo FSW.

Nota-se, aqui, a importância das informações de tempo nos resultados apresentados. Com estas informações, juntamente com os parâmetros do processo, é possível determinar a posição de cada falha no cordão através da Eq. (6.1) podendo, inclusive, substituir o tempo no eixo das abscissas pela posição. Também é importante que os dois métodos sejam utilizados de maneira complementar, ou seja, para a análise de um sinal, deve-se utilizar ambos os métodos para se tirar as conclusões.

7 CONCLUSÕES

O processo de junção por *Friction Stir Welding* (FSW) vem sendo amplamente aplicado nas indústrias, principalmente no ramo aeronáutico, para união das chapas da fuselagem dos aviões. O monitoramento deste processo é importante para a detecção e localização de falhas que possam surgir no cordão, contribuindo para a redução dos custos e do tempo de produção.

Por ser um processo relativamente novo, existem poucos trabalhos publicados relacionados à detecção de falhas em FSW, o que torna os estudos apresentados nesta dissertação importantes para o desenvolvimento do monitoramento do processo.

Este trabalho apresentou dois métodos para o monitoramento do processo FSW: o método da Transformada de Fourier de Tempo Curto (STFT) e o método da Transformada *Wavelet*. A aplicação destes métodos em sinais simulados permitiu verificar a capacidade de detecção de falhas em sinais simples e padronizados. Através da aplicação destes métodos em sinais adquiridos por acelerômetros, buscou-se analisar a capacidade de cada um deles na detecção e localização de falhas feitas na linha de junção de chapas de alumínio.

O método da STFT apresenta uma vantagem em relação à simples aplicação da Transformada Rápida de Fourier (FFT): as informações de tempo não são perdidas e são importantes para a localização das falhas no cordão. A aplicação deste método nos sinais simulados mostrou ser capaz de identificar falhas com grandes amplitudes (aproximadamente 30% do valor das amplitudes do sinal). São falhas grandes, capazes de serem também detectadas por uma simples análise visual dos sinais.

No entanto, a aplicação deste método nos ensaios realizados mostrou-se eficiente na detecção e localização das falhas. Em um dos acelerômetros posicionados nas chapas (acelerômetro 2), o método da STFT foi capaz de detectar as três falhas existentes, além de apresentar os instantes de tempo do início e do término do processo.

O método da Transformada *Wavelet* também apresenta as informações de tempo em sua aná-

lise e sua aplicação nos sinais simulados apresentou resultados melhores que a STFT. As amplitudes das falhas detectadas foram menores que as amplitudes propostas inicialmente. A aplicação deste método nos ensaios apresentou resultados onde, em um dos acelerômetros, as três falhas puderam ser identificadas e localizadas.

Sinais simulados apresentam menos variações e menos ruído, fazendo com que a detecção das falhas seja mais simples, aumentando a possibilidade de resultados positivos. Porém, para os sinais obtidos nos ensaios, o nível de ruído e interferências externas é muito alto. A má fixação dos acelerômetros e dos cabos e vibração excessiva da máquina podem prejudicar a aquisição dos sinais.

A interface gráfica criada apresenta a opção de análise pelos dois métodos apresentados. Como visto nos resultados, a utilização de ambos permite uma análise mais completa e aumenta a certeza na detecção das falhas.

Apesar dos resultados dos ensaios apresentarem interferências externas, os métodos testados puderam fornecer as informações necessárias para a localização das falhas. Sabendo-se a posição das falhas antes de aplicar os métodos de detecção permitiu constatar que existem diferenças significativas entre os intervalos do sinal sem falhas e os intervalos onde as falhas foram detectadas. Dessa forma, torna-se possível a detecção da falha analisando se há ou não mudanças no padrão do sinal analisado, mesmo sem prévias informações sobre suas posições.

Em trabalhos futuros, falhas com dimensões menores podem ser ensaiadas e outros tipos de sensores podem ser utilizados como, por exemplo, os sensores de emissão acústica.

O monitoramento *on-line* também é muito importante para o processo FSW pois, ao detectar uma falha durante o processo, pode-se tomar as providências necessárias para que haja uma correção ou até mesmo que o processo seja interrompido ou terminado.

REFERÊNCIAS

- ABAL. **Soldagem por fricção linear**. 2011. Disponível em: <http://www.abal.org.br/aluauto/ed10/mundoautomotivo4.asp>. Acesso em: 4 de mai. 2011.
- BIRD, C. R.; KLEINER, D. **The phased array inspection of friction stir welded aluminium plant**. 2012. Disponível em: <http://www.twi.co.uk/technical-knowledge/published-papers/the-phased-array-inspection-of-friction-stir-welded-aluminium-plant/>. Acesso em: 6 de ago. 2012.
- BOMBONATO, S. G.; LUSTOSA, A. A.; SILVA, M. B. **Análise do desgaste de ferramentas via emissão acústica com aplicações em redes neurais**. 2011. Disponível em: <http://www.seer.ufu.br/index.php/horizontecientifico/article/viewFile/4061/3024>. Acesso em: 23 de out. 2011.
- CAPELARI, T. V.; MAZZAFERRO, J. A. E. Avaliação da geometria de ferramenta e parâmetros do processo FSW na soldagem da liga de alumínio AA 5052. **Soldagem & Inspeção**, v. 14, n. 3, p. 215–227, 2009. Disponível em: <http://www.scielo.br/scielo.php?pid=S0104-92242009000300005&script=sci_arttext>.
- CHAN, S. C.; ZHU, Z. Y. **Short-time Fourier Transform**. 2012. Disponível em: <http://www.eee.hku.hk/~work3220/Speech%20analysis%20and%20synthesis-2%20STFT.pdf>. Acesso em: 10 de set. 2012.
- CHEN, C.; KOVACEVIC, R.; JANDGRIC, D. Wavelet transform analysis of acoustic emission in monitoring friction stir welding of 6061 aluminum. **International Journal of Machine Tools and Manufacture**, v. 43, n. 13, p. 1383–1390, 2003. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0890695503001305>>.
- FLEMING, P. A. **Monitoring and control of friction stir welding**. Tese (Doutorado) — Graduate School of Vanderbilt University, Nashville, Tennessee, May 2009. Dispo-

nível em: <<http://etd.library.vanderbilt.edu/ETD-db/available/etd-01222009-003129/>>.

GALLEGATI, M. Wavelet analysis of stock returns and aggregate economic activity. **Computational Statistics & Data Analysis**, v. 52, n. 6, p. 3061–3074, 2008. ISSN 0167-9473. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167947307002915>>.

GENICOLO, M. A. C. **Estudo de viabilidade para implementação de uniões soldadas pelo método friction stir welding em estruturas aeronáuticas**. Dissertação (Mestrado em Engenharia Mecânica de Projeto de Fabricação) — Escola Politécnica, Universidade de São Paulo, São Paulo, 2007. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/3/3151/tde-08012008-162326/>>.

GUTIERREZ-OSUNA, R. **Short-time Fourier analysis and synthesis**. 2012. Disponível em: <http://research.cs.tamu.edu/prism/lectures/sp/l6.pdf>. Acesso em: 10 de set. 2012.

JAMES, M. N. et al. The relationship between process mechanisms and crack paths in friction stir welded 5083-H321 and 5383-H321 aluminium alloys. **Fatigue & Fracture of Engineering Materials & Structures**, v. 28, sep 2004. Disponível em: <<http://onlinelibrary.wiley.com/doi/10.1111/j.1460-2695.2004.00830.x/pdf>>.

KALLEE, S. W.; J., D.; NICHOLAS, E. D. **Railway Manufacturers Implement Friction Stir Welding**. 2012. Disponível em: <http://www.aws.org/wj/oct02/feature.html>. Acesso em: 8 de nov. 2012.

MALLAT, S. A theory for multiresolution signal decomposition: the wavelet representation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 11, p. 674–93, 1989. Disponível em: <<http://www.cs.sfu.ca/CourseCentral/820/li/material/source/papers/Mallat-wavelet.pdf>>.

MELLO, C. A. **Análise wavelet**. 2011. Disponível em: <http://www.cin.ufpe.br/~cabm/pds>. Acesso em:

26 de fev. 2011.

MERRY, R. J. E. **Wavelet theory and applications: a literature study**. 2012. Disponível em: <http://alexandria.tue.nl/repository/books/612762.pdf>. Acesso em: 27 de mar. 2012.

MORAES, F. J. V. **Teoria wavelet aplicada à análise de vibrações**. 105 f. Dissertação (Mestrado em Engenharia Mecânica) — Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas, 1996. Disponível em: <<http://www.bibliotecadigital.unicamp.br/document/?code=vtls000111027>>.

MUTHUKUMARAN, S.; MUKHERJEE, S. Two modes of metal flow phenomenon in friction stir welding process. **Sci Technol Weld Joi**, v. 11, p. 337–340, 2006.

NEWLAND, D. E. **An introduction to: random vibrations, spectral and wavelet analysis**. Third. Longman House, Burnt Mill, Harlow, Essex, CM20 2JE, England: Longman Scientific & Technical, 1993.

OLIVEIRA, H. M. **Wavelets: entrando na onda**. 2012. Disponível em: <http://www2.ee.ufpe.br/codec/slidesseminario.pdf>. Acesso em: 19 de jun. 2012.

OOSTERKAMP, A.; OOSTERKAMP, L. D.; NORDEIDE, A. **'Kissing Bond' phenomena in solid-state welds of aluminum alloys**. 2004. Disponível em: <http://www.aws.org/wj/supplement/08-2004-OOSTERKAMP-s.pdf>. Acesso em: 5 de nov. 2012.

POLT, W. J. **A little friction at Boeing**. 2012. Disponível em: http://www.boeing.com/news/frontiers/archive/2004/september/i_tt.html. Acesso em: 8 de nov. 2012.

QUATIERI, T. F. Filter-bank analysis/synthesis. In: **Discrete-Time Speech Signal Processing: Principles and Practice**. [S.l.]: Prentice Hall PTR, 2001. cap. 8, p. 363–426.

_____. Short-time fourier transform analysis and synthesis. In: **Discrete-Time Speech Signal Processing: Principles and Practice**. [S.l.]: Prentice Hall PTR, 2001. cap. 7, p. 309–362.

SAPA. **Friction Stir Welding**. [S.l.], 2012. Disponível em: <http://www.sapagroup.com/pages/797/FSW_folder_Eng.pdf>.

_____. **Joining by Friction Stir Welding (FSW)**. 2012. Disponível em: <http://www.sapagroup.com/en/sapa-rail/our-offer1/production/friction-stir-welding/>. Acesso em: 8 de nov. 2012.

SETHARES, W. A. **A phase vocoder in Matlab**. 2012. Disponível em: <http://sethares.engr.wisc.edu/vocoders/phasevocoder.html>. Acesso em: 5 de nov. 2012.

SINHA, P. et al. Condition monitoring of first mode of metal transfer in friction stir welding by image processing techniques. **The International Journal of Advanced Manufacturing Technology**, Springer London, v. 36, p. 484–489, 2008. ISSN 0268-3768. Disponível em: <<http://dx.doi.org/10.1007/s00170-006-0854-2>>.

SOUNDARARAJAN, V.; ATHARIFAR, H.; KOVACEVIC, R. Monitoring and processing the acoustic emission signals from the friction-stir-welding process. **Proceedings of the Institution of Mechanical Engineers. Part B, Journal of Engineering Manufacture**, 2006. Disponível em: <<http://pib.sagepub.com/content/220/10/1673.short>>.

The Research Laboratory of Civil Engineering and Mechanics of Ecole Centrale de Nantes. **Processing technology**. 2011. Disponível em: <http://www.ec-nantes.fr/version-anglaise/research/gem/research-group-polymers-dynamics-and-processing-technology-77700.kjsp>. Acesso em: 11 de jul. 2011.

TWI, T. W. I. **Friction Stir Welding**. 2012. Disponível em: <http://www.twi.co.uk/technologies/welding-coating-and-material-processing/friction-stir-welding>. Acesso em: 8 de mai. 2012.

____. **Friction stir welding enables worldwide engineering success.** 2012. Disponível em: <http://www.twi.co.uk/news-events/news/2012-11-friction-stir-welding-enabling-worldwide-engineering-success/>. Acesso em: 8 de mai. 2012.

YANG, Y. et al. Automatic gap detection in friction stir welding processes. In: **Proceedings of 2006 ASME International Conference on Manufacturing Science and Engineering.** [s.n.], 2006. Disponível em: <<http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA458171>>.

ANEXO A - ALGORITMOS

A.1 Programa Principal

```
%% Start initialization code - DO NOT EDIT
function varargout = interface_wavelet2(varargin)

warning('off')

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
    'gui_Singleton',  gui_Singleton, ...
    'gui_OpeningFcn', @interface_wavelet2_OpeningFcn, ...
    'gui_OutputFcn',  @interface_wavelet2_OutputFcn, ...
    'gui_LayoutFcn',  [] , ...
    'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

%%
```

```

% --- Executes just before interface_wavelet2 is made visible.
function interface_wavelet2_OpeningFcn(hObject, eventdata, ...
    handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.
function varargout = interface_wavelet2_OutputFcn(hObject, ...
    eventdata, handles)
varargout{1} = handles.output;

% --- Executes on button press in botao_exit.
function botao_exit_Callback(hObject, eventdata, handles)
close

% --- Executes on selection change in menu_camadas.
function menu_camadas_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function menu_camadas_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), ...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in menu_construtores.
function menu_construtores_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function menu_construtores_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), ...

```

```

        get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in menu_sinais.
function menu_sinais_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function menu_sinais_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in menu_analise.
function menu_analise_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function menu_analise_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in cores.
function cores_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function cores_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))

```



```

        set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in menu_grid.
function menu_grid_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function menu_grid_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in menu_preenchimento.
function menu_preenchimento_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function menu_preenchimento_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function menu_passo_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function menu_passo_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

function arquivo_entrada_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function arquivo_entrada_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function acelerometro_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function acelerometro_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in botao_ver_sinal.
function botao_ver_sinal_Callback(hObject, eventdata, handles)
cla(handles.axes3);
cla(handles.axes);
set(handles.aviso,'String',' ');
set(handles.tempo_gasto,'String',' ');

% turning off the other buttons
set(handles.botao_exit,'Enable','off')
set(handles.botao_limpar,'Enable','off')
set(handles.botao_detectar,'Enable','off')

```

```

set(handles.aviso,'String','Carregando o sinal.....');
set(handles.tempo_gasto,'String',' ');
drawnow

nome = get(handles.arquivo_entrada,'String');
acel = get(handles.acelerometro,'String');
acel = str2double(acel);

[y,tempo] = loading(nome);
sinal = y{acel};

%% Adjusting the size of the signal
[sinal_cf,comprimento,tamanho_original] = adequacao_tamanho(sinal);
sinal_sf = sinal_cf;

plot(tempo,sinal_cf(1:tamanho_original),'Parent',handles.axes2)
title('Sinal','Parent',handles.axes2)
xlabel('tempo (s)','Parent',handles.axes2)
ylabel('aceleração (m/s^{2})','Parent',handles.axes2)
grid(handles.axes2)

set(handles.aviso,'String',' ');
set(handles.tempo_gasto,'String',' ');
drawnow % to show the message at the moment

% turning on the other buttons
set(handles.botao_exit,'Enable','on')
set(handles.botao_limpar,'Enable','on')
set(handles.botao_detectar,'Enable','on')

% --- Executes on button press in botao_limpar.

```

```

function botao_limpar_Callback(hObject, eventdata, handles)
cla(handles.axes3);
cla(handles.axes2);
cla(handles.axes);
set(handles.avisos,'String',' ');
set(handles.tempo_gasto,'String',' ');

% --- Executes on button press in botao_detectar.
function botao_detectar_Callback(hObject, eventdata, handles)
%% Important: "detectar" function
tic;

cla(handles.axes3);
cla(handles.axes2);
cla(handles.axes);
set(handles.avisos,'String',' ');
set(handles.tempo_gasto,'String',' ');

% turning off the other buttons
set(handles.botao_exit,'Enable','off')
set(handles.botao_limpar,'Enable','off')
set(handles.botao_ver_sinal,'Enable','off')

set(handles.avisos,'String','Calculando.....');
set(handles.tempo_gasto,'String',' ');
drawnow

passo = get(handles.menu_passo,'String');
passo = str2double(passo);

nome = get(handles.arquivo_entrada,'String');

```

```

accel = get(handles.acelerometro,'String');
accel = str2double(accel);

[y,t] = loading(nome);

tempo = t(1:passo:end);
sinal = y{accel}(1:passo:end);

opcao_analise = get(handles.menu_analise,'Value');

%%
if opcao_analise == 1
    %% Matching the size of the signal
    [sinal_cf,comprimento,tamanho_original] = adequacao_tamanho(sinal);
    sinal_sf = sinal_cf;

    %% Calculating the Mean Square Map
    n_constr = get(handles.menu_construtores,'Value');
    n_constr = n_constr*2;
    mapa = map_amadeu(sinal_cf,n_constr);

    %% Signal Normalization
    mapa_norm = normalizacao(mapa);

    set(handles.aviso,'String','Gerando Mapa.....');
    drawnow

    %% Plots
    grade = get(handles.menu_grid,'Value');
    %-----
    % Plot - signals

```

```

plot(tempo,sinal_cf(1:tamanho_original),'Parent',handles.axes2)
title('Sinal','Parent',handles.axes2)
xlabel('tempo (s)','Parent',handles.axes2)
ylabel('aceleração (m/s^{2})','Parent',handles.axes2)
set(handles.axes2,'XTick',0:round(tempo(end)/20):tempo(end))
grid(handles.axes2)

%% Plot map
escala = linspace(tempo(1),tempo(end),tamanho_original/2);

for i = 1:size(mapa_norm,1)
    escala2(i,:) = escala;
    mapa_norm2(i,:) = mapa_norm(i, :)+(i-3);
end
plot(escala2.',mapa_norm2(:,(1:length(sinal)/2)).',...
    'b-','Parent',handles.axes3)
ylabel('Níveis','Parent',handles.axes3)
xlabel('Tempo (s)','Parent',handles.axes3)
set(handles.axes3,'XTick',0:round(tempo(end)/20):tempo(end))
set(handles.axes3,'YTick',-1:1:size(mapa_norm,1)-2)
grid(handles.axes3)

%% Plot contour map
escala = linspace(tempo(1),tempo(end),tamanho_original/2);

camadas = get(handles.menu_camadas,'Value');
camadas = 5*camadas;

opcao_cor = get(handles.cores,'Value');
if opcao_cor == 1
    cor = 'jet';

```

```

elseif opcao_cor == 2
    cor = 'summer';
elseif opcao_cor == 3
    cor = 'lines';
end

preenchimento = get(handles.menu_preenchimento,'Value');
if preenchimento == 1
    contour(escala,-1:(size(mapa_norm,1)-2),...
        mapa_norm(:,1:tamanho_original/2),camadas,...
        'Parent',handles.axes) %contour
elseif preenchimento == 2
    contourf(escala,-1:(size(mapa_norm,1)-2),...
        mapa_norm(:,1:tamanho_original/2),camadas,...
        'LineColor','none','Parent',handles.axes) %contourf
end
colormap(cor)
title(['Mapa de quadrados médios - ',num2str(n_constr),...
    ' construtores - ',num2str(camadas),' camadas'],...
    'Parent',handles.axes)
xlabel('Tempo (s)','Parent',handles.axes)
ylabel('Níveis','Parent',handles.axes)
set(handles.axes,'XTick',0:round(tempo(end)/20):tempo(end))
set(handles.axes,'YTick',-1:1:size(mapa_norm,1)-2)
if grade == 1
    grid(handles.axes)
end

elseif opcao_analise == 2
    tamanho_original = length(sinal);

```

```

%% Signal Division
[s_part,S_part,f_part,pt] = divisao(sinal);

%% Z-Matrix
matriz_z = zeros(size(S_part,2),size(S_part{1},2));
for i = 1:size(S_part,2)
    matriz_z(i,:) = S_part{i};
end

%% Signal Normalization
[matriz_z_norm] = normalizacao(matriz_z);

set(handles.aviso,'String','Gerando Mapa.....');
drawnow

%% Plots
cla(handles.axes3);
grade = get(handles.menu_grid,'Value');

%% Plot - signals
plot(tempo,sinal(1:tamanho_original),'Parent',handles.axes2)
title('Sinal','Parent',handles.axes2)
xlabel('tempo (s)','Parent',handles.axes2)
ylabel('aceleração (m/s^{2})','Parent',handles.axes2)
set(handles.axes2,'XTick',0:round(tempo(end)/20):tempo(end))
grid(handles.axes2)

%% Plot contour map
camadas = get(handles.menu_camadas,'Value');
camadas = 5*camadas;

```



```

opcao_cor = get(handles.cores,'Value');
if opcao_cor == 1
    cor = 'jet';
elseif opcao_cor == 2
    cor = 'summer';
elseif opcao_cor == 3
    cor = 'lines';
end

preenchimento = get(handles.menu_preenchimento,'Value');
if preenchimento == 1
    contour(linspace(tempo(1),tempo(end),pt),f_part{1},...
        matriz_z_norm(:,1:length(f_part{1})).',camadas,...
        'Parent',handles.axes) %contour
elseif preenchimento == 2
    contourf(linspace(tempo(1),tempo(end),pt),f_part{1},...
        matriz_z_norm(:,1:length(f_part{1})).',camadas,...
        'LineColor','none','Parent',handles.axes) %contourf
end
colormap(cor)
title(['Mapa de quadrados médios - ',num2str(pt),...
    ' divisões - ',num2str(camadas),' camadas'],...
    'Parent',handles.axes)
ylabel('Frequência (Hz)','Parent',handles.axes)
xlabel('Tempo (s)','Parent',handles.axes)
set(handles.axes,'XTick',0:round(tempo(end)/20):tempo(end))
if grade == 1
    grid(handles.axes)
end
end
end

```

```

set(handles.aviso,'String','Pronto!');

format short
tElapsed = toc/60;

set(handles.tempo_gasto,'String',[num2str(tElapsed),' minutos']);

% turning on the other buttons
set(handles.botao_exit,'Enable','on')
set(handles.botao_limpar,'Enable','on')
set(handles.botao_ver_sinal,'Enable','on')

clear camadas comprimento f i y1 NFFT % cleaning

```

A.2 Carregando o sinal

```

%% Programa Loading

function[y,t] = loading(nome)

load (nome)

for i = 1:size(Signal.y_values.values,2)
    y{i} = Signal.y_values.values(:,i);
end

t0 = Signal.x_values.start_value;
delta_t = Signal.x_values.increment;
n = Signal.x_values.number_of_values;

```

```

t = t0:delta_t:n*delta_t;
t = t.';

clear i j n t0 Signal delta_t

```

A.3 Adequação do tamanho do sinal

```

% Auxiliar Program
% Signal Length Adequation

function [sinal, comprimento, tamanho_original] =...
    adequacao_tamanho(sinal)

tamanho_original = length(sinal);
tam = ceil(log(length(sinal))/log(2));
if round(tam/2) ~= tam/2
    tam = tam+1;
end
sinal_cf = zeros(2^tam,1);
sinal_cf(1:length(sinal)) = sinal;
comprimento = length(sinal_cf)/length(sinal);
sinal = sinal_cf;

```

A.4 Mapa de Quadrados Médios

```

% Mean Square Map
% Based on Newland (1993)

```

```

function A = map_amadeu(f,N)

M = length(f); % "f" signal length
n = round(log(M)/log(2));
a = wavedn(f,N); % wavelet transform coeffs

a_quad = a.*conj(a);

A = zeros(length(f)/2,n+1);
A(1:M/2,1) = a_quad(1);

cont = 2;
for j = 2:n+1
    cols_cont = 0;
    for i = 1:2^(j-2)
        cols = (M/2)/(2^(j-2));
        A(cols_cont+1:cols_cont+cols,j) = a_quad(cont);
        cont = cont+1;
        cols_cont = cols_cont+cols;
    end
    A(:,j) = circshift(A(:,j),cols);
end

A = A.';

```

A.5 Coeficientes *wavelet*

% To generate the N coefficients required by the wavelet transform.

```

function c = dcoeffs_amadeu(N)

if N == 2
c(1) = 7.07106781186547524400844362104849039284835937688474e-01;
c(2) = 7.07106781186547524400844362104849039284835937688474e-01;
end

if N == 8
c(1) = 2.30377813308896500863291183044070850001615248248309e-01;
c(2) = 7.14846570552915647089921955273992603707608401099308e-01;
c(3) = 6.30880767929858907881716338300615220203222922677195e-01;
c(4) = -2.7983769416859854211413747180075385411987320224491e-02;
c(5) = -1.8703481171909308407957067278908141958454417437458e-01;
c(6) = 3.08413818355607636272193625349590501703148217200340e-02;
c(7) = 3.28830116668851997354075135492443886645419411375497e-02;
c(8) = -1.0597401785069032104883208524027229181099964906376e-02;
end

if N == 12
c = [1.1154074335010946362132391724092343904253959198442167e-01
4.94623890398453085677204176877855588637786382896274362e-01
7.51133908021095350678934498439731685580254783338261200e-01
3.15250351709197629085989654810926396649519923517294524e-01
-2.2626469396543982007631450066090346567054015397289699e-01
-1.2976686756726193556228960587658546084523374922358147e-01
9.75016055873230491023435525381253423398307474952551427e-02
2.75228655303057286255408395041932136573875878304345432e-02
-3.1582039317486029565079080699848669057479532373148423e-02
5.53842201161496139251918398046501220611026277386496429e-04
4.77725751094551063963597524682070705023050121658143429e-03

```

```
-1.0773010853084795648526216095872000352352336093344196e-03].';  
end
```

```
if N == 76
```

```
c = [1.4257766416741316720554202475678658032117843974641911e-06  
3.57625199426402301274256901488887621795830722794012641e-05  
4.21170266472711643224701444490646915530057320113054973e-04  
3.08308811925375177428874009026274191017732252062458286e-03  
1.56372493475721561727749010272408007048627002663262066e-02  
5.78899436128592564972766427931724195251324628776648121e-02  
1.60071993564110697348280086116659968516939546505504895e-01  
3.30775781411014651149363753440461175480076867704157703e-01  
4.96591175311718097659917114771870893935241483895172608e-01  
4.93356078517100797572848534699731706496951362359435909e-01  
2.13050571355578513828674335345856245125562466595116044e-01  
-1.8286766770833589079755485079462391352182231850414106e-01  
-3.2167563780899786284834717254069163619298419405281890e-01  
-6.2266506047824322266433601604787658475658621010455971e-02  
2.32125963835353108502870810428599499867161556366285807e-01  
1.49985119618717019958640345378892730729822602826260302e-01  
-1.4179568597305962167100531445223302763925910553758306e-01  
-1.5991256515824436182885332145235349378042088443861026e-01  
8.56381215561510574161221781436916531348712964553600185e-02  
1.41414734073382680088468311937917059409260617491575528e-01  
-5.6586458630727381456817876578433206468155094106351142e-02  
-1.1473117071074437523941440194589427797156654892301699e-01  
4.30958954330476428813787122361603062424656868359540879e-02  
8.72043982620397501191071416415445676207378612423308847e-02  
-3.6605103402874295673720710395067723725679387109434328e-02  
-6.1766208708413159936047367056132462418974977823733379e-02  
3.19898775315378063081838113636685902613703545057663113e-02
```

4.00549811051159482095208708624111430903857737936673295e-02
-2.6891493880894514385508517677159673134178903932872367e-02
-2.3114134020549316808569135535856212489253038655402033e-02
2.09046452556552434021598236535134209467026149152683167e-02
1.12904972786859648427008148776154423285111589144984396e-02
-1.4701882065398682137089864028166050456484812246624351e-02
-4.1313066560310892741232311033267457231881345485209381e-03
9.21478503219718051203153487018173400352286164590389450e-03
5.62571574840353200574156559488114875706670343721452210e-04
-5.0713145092183480939350614175056630020068213239587526e-03
7.16982182106401925778416536489491562188854149677337043e-04
2.40069778189097318389230691408259214398414055021013013e-03
-8.4486266655377750090689378514658569732513630109240033e-04
-9.4246140772273779640159422717800982839102306399080187e-04
5.81075975053286366202032106367819663340955570698147672e-04
2.81763925038067074601804896753560819012352318061296106e-04
-3.0310204607266119936006290203297846824964771064704277e-04
-4.5556826966684202746886830059877643606772171499279383e-05
1.26204335016617070538234653713181770136152238790491733e-04
-1.1554091038337171926284790479834609533819593426423741e-05
-4.1751416485403977972963250657757113091974119262894124e-05
1.33417614992135038254750345728606092221807003133013760e-05
1.03735918404559979563225833501006510352495984496609487e-05
-6.4567304284696191603799104396175754209869723941371219e-06
-1.5508443501186025758533801485259129994012924731855343e-06
2.14996026993966520778954819979077059689025240507639488e-06
-8.4870875860725930718698052660894266296064798769822218e-08
-5.1877337388741444260084746833785423680663100006028230e-07
1.39637754550835548122796158105996118451987250249346201e-07
8.40035104689596552693358717678127950795308066925931872e-08
-4.8847579374592867620821854116087639640410103921019148e-08

```

-5.4242748002872985111266841748544149284475217106644764e-09
1.03470453927485848092404649095280393732823953722290815e-08
-1.4363294877951357068545398569792759111836284765216362e-09
-1.3491977539834488218503817708897863012467413043079349e-09
5.26113255735759849453576663877262457210033220919897965e-10
6.73233649018930868574062696418262315975976753672484403e-11
-8.2782565225381347273306929381589911153353846117958747e-11
1.10169293459945455115083262216022423128019536291949854e-11
6.29153731703950858158091362085914083585288630898958419e-12
-2.4847892375636428570433612145027607236114685918332626e-12
2.62649650406525207048828287647052537985142953838948157e-14
1.80866123627453058226708484634395937708592201906780814e-13
-4.2498178195714630069666163715542065728631225627449167e-14
-4.5633971621273731091016916430479237477965634491940756e-16
2.04509967678898890780227256440231009539864109281936716e-15
-4.4053070424834613424490271398383016110068352854550501e-16
4.30459683955879001625186747712279150884969768805816905e-17
-1.7161524510887441887324042817379642777130260872242482e-18] .';

```

end

A.6 Normalização

```
%% Normalização
```

```
function [mapa_norm] = normalizacao(mapa)
```

```
mapa_norm = zeros(size(mapa));
```

```
for i = 1:size(mapa,1)
```

```
    linha = mapa(i,:);
```



```

    mapa_norm(i,:) = (linha.*1/max(linha));
end

```

A.7 Divisão do sinal

```

% Signal Division

function [s_part,S_part,f_part,pt] = divisao(sinal)

pt = round(length(sinal)/150);

l = round(length(sinal)/pt)-1; %partition length

cont1 = 1; cont2 = 1;
coll1 = 1; col2 = 2;
for i = 1:pt
    s_part{i} = sinal(cont1:cont2); % partitioned signal matrix
    cont1 = cont1+l;
    cont2 = cont2+l;

    NFFT = 2^nextpow2(l); % Next power of 2 from length of y
    S_part{i} = fft(s_part{i},NFFT)/l;
    S_part{i} = S_part{i}.';
    f_part{i} = length(sinal)/2*linspace(0,1,NFFT/2+1);
    f_part{i} = f_part{i}.';
end

```