

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELÉTRICA
CURSO DE ENGENHARIA ELÉTRICA

VINICIUS GALLERT DOS SANTOS

**SEGMENTAÇÃO AUTOMÁTICA DE REGISTROS DE OSCILOGRAFIA
FASORIAL COM ÊNFASE EM SISTEMAS DE GERAÇÃO**

TRABALHO DE CONCLUSÃO DE CURSO

PATO BRANCO

2014

VINICIUS GALLERT DOS SANTOS

**SEGMENTAÇÃO AUTOMÁTICA DE REGISTROS DE OSCILOGRAFIA
FASORIAL COM ÊNFASE EM SISTEMAS DE GERAÇÃO**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 2, do Curso de Engenharia Elétrica da Coordenação de Engenharia Elétrica – COELT – da Universidade Tecnológica Federal do Paraná – UTFPR, Câmpus Pato Branco, como requisito parcial para obtenção do título de Engenheiro.

Orientador: Prof. Me. Johnny Werner

Coorientador: Prof. Dr. Miguel Moreto

PATO BRANCO

2014

TERMO DE APROVAÇÃO

O trabalho de Conclusão de Curso intitulado **SEGMENTAÇÃO AUTOMÁTICA DE REGISTROS DE OSCILOGRAFIA FASORIAL COM ÊNFASE EM SISTEMAS DE GERAÇÃO**, do aluno **VINICIUS GALLERT DOS SANTOS** foi considerado **APROVADO** de acordo com a ata da banca examinadora N° 51 de 2014.

Fizeram parte da banca os professores:

PROF. Me. JOHNNY WERNER

PROF. Me. CESAR AUGUSTO PORTOLANN

PROF. Dr. RICARDO VASQUES DE OLIVEIRA

A ATA assinada encontra-se na Coordenação do Curso de Engenharia Elétrica.

DEDICATÓRIA

Aos meus pais, que acreditaram em mim e não mediram esforços para que eu alcançasse meus objetivos.

AGRADECIMENTOS

A Deus primeiramente, por me dar a oportunidade de viver e tornar meus sonhos possíveis. A meus pais pela confiança e apoio incondicional em todos os momentos. A esta universidade, seu corpo docente, direção e administração pela oportunidade e pelo apoio no decorrer deste curso. Ao Prof. Dr. Miguel Moreto e ao Prof. Me. Johnny Werner pela orientação e ajuda no desenvolvimento deste trabalho. Aos amigos, que sempre estiveram presentes e a todos que estiveram direta ou indiretamente presentes durante o desenvolvimento do trabalho.

EPÍGRAFE

Que os vossos esforços desafiem as impossibilidades, lembrai-vos de que as grandes coisas do homem foram conquistadas do que parecia impossível. (Charles Chaplin)

RESUMO

SANTOS, Vinicius G. Segmentação Automática de Registros de Oscilografia Fasorial com Ênfase em Sistemas de Geração. 2014. 51 f. Trabalho de Conclusão de Curso – Curso de Engenharia Elétrica, Universidade Tecnológica Federal do Paraná. Pato Branco, 2014.

Com a necessidade cada vez mais crescente de se obter confiabilidade nos sistemas de geração de energia elétrica, o estudo dos distúrbios que ocorrem nesses sistemas tem se tornado ponto chave na prevenção de paradas no fornecimento. Com o desenvolvimento de aparelhos capazes de monitorar esses distúrbios, como os Registradores Digitais de Perturbação (RDPs) surgiram as oscilografias digitais, que são sinais que registram as grandezas do sistema elétrico (corrente e tensão) em formas de onda ao longo do tempo ou em forma fasorial. Quando da ocorrência de uma falta no sistema, essas oscilografias registram o distúrbio. Nas centrais de comando dos sistemas de geração existem bancos de dados de oscilografias que estão conectados em rede com os RDPs, que enviam sequencialmente registros. Um grande fluxo de informação é enviado, porém, na maioria dos casos, com informação que não necessita ser analisada. A segmentação faz parte da análise automática de oscilografias, que tem a função de analisar automaticamente quais oscilografias requerem análise detalhada, sendo que as irrelevantes são descartadas. Este trabalho apresenta um programa que implementa uma ferramenta estatística chamada desvio padrão para identificar variações de amplitude anormais de um dado sinal e possui capacidade de identificar múltiplos índices de segmentação. A partir da segmentação pode-se iniciar um processo de análise aprofundado, gerando um fluxo de registros menor e propiciando agilidade no estudo dos distúrbios.

Palavras-chave: Oscilografias, segmentação automática, sistemas de geração, desvio padrão.

ABSTRACT

SANTOS, Vinicius G.. Automatic Segmentation of Oscillography Records with Emphasis in Generation Systems. 2014. 51 f. Trabalho de Conclusão de Curso – Curso de Engenharia Elétrica, Universidade Tecnológica Federal do Paraná. Pato Branco, 2014.

With the ever increasing need to obtain the reliability of electric power generation systems, the study of disturbances that occur in these systems has become a key point in preventing the supply stops. With the development of devices able to monitor these disturbances such as Digital Fault Recorders (DFRs), digital oscillography emerged, which are signs that record the magnitudes of the electric system (current and voltage) on waveforms over time or in phasor form. Upon the occurrence of a system fault, these oscillography record the disturbance. In the command centers of the generation systems, oscillography database are networked with DFRs, which sends sequentially records. A large flow of information is sent, however, in most cases, that information cannot be analyzed. The segmentation is part of the automatic analysis oscillography, which has the function of automatically analyzing oscillographys which require detailed analysis, discarding irrelevant records. This work presents a program which implements a tool called standard deviation to identify abnormal amplitude variations of the signal and has ability to identify multiple levels of segmentation. From the segmentation can begin a process of thorough analysis, generating a stream of lower records and providing agility in the study of disturbances.

Keywords: Oscillographys, Automatic Segmentation, Generation Systems, Standard Deviation.

LISTA DE FIGURAS

Figura 1 - Inserção do RDP na rede de oscilografias.....	16
Figura 2 - Registro de oscilografia fasorial.....	17
Figura 3 - Fluxograma da análise automática de oscilografias.....	18
Figura 4 - Segmentação de um registro de oscilografia fasorial.....	19
Figura 5 - Diagrama de blocos de execução do código.....	32
Figura 6 - Segmentação utilizando a metodologia dos mínimos quadrados.....	33
Figura 7 - Registro segmentado em escala maior.....	34
Figura 8 - Segmentação utilizando o método do desvio padrão.....	34
Figura 9 - Registro segmentado em escala maior.....	35
Figura 10 - Segmentação utilizando a metodologia dos mínimos quadrados.....	36
Figura 11 - Registro segmentado em escala maior.....	37
Figura 12 - Segmentação utilizando o método do desvio padrão.....	37
Figura 13 - Registro segmentado em escala maior.....	38
Figura 14 - Indicação do programa quando não ocorre nenhum evento.....	39

LISTA DE QUADROS

Quadro 1 – Bases de cálculo das usinas.....	22
---	----

LISTA DE ABREVIACOES

DWT – *Discrete Wavelet Transform* (Transformada Discreta Wavelet)

EMD – *Empirical Mode Decomposition* (Modo de Decomposio Emprica)

I - Corrente

IMF – *Intrinsic Mode Functions* (Funes de Modo Intrnseco)

UHMA – Usina Hidreltrica de Machadinho

UHPF – Usina Hidreltrica de Passo Fundo

UTCH – Usina Termoeltrica de Charqueadas

pu – Por Unidade

RDP – Registrador Digital de Perturbaes

RMS – *Root Mean Square* (Raiz Mdia Quadrtica)

V – Tenso

LISTA DE TABELAS

Tabela 1 – Exemplo 1: Comparação entre mínimos quadrados (<i>id2</i>) e desvio Padrão (<i>id3</i>).....	35
Tabela 2 - Exemplo 2: Comparação entre mínimos quadrados (<i>id2</i>) e desvio padrão (<i>id3</i>).....	38

SUMÁRIO

1. INTRODUÇÃO.....	10
2. REVISÃO BIBLIOGRÁFICA	12
3. REVISÃO TEÓRICA.....	16
4. METODOLOGIAS DE TRABALHO	20
4.1 Metodo dos Minimos Quadrados.....	20
4.2 Metodo do Desvio Padrão	23
5. RESULTADOS E DISCUSSÕES.....	33
6. CONCLUSÕES.....	40
REFERÊNCIAS.....	42
APÊNDICE A.....	43

1. INTRODUÇÃO

As máquinas elétricas que constituem as unidades de geração de energia elétrica são componentes vitais para o suprimento da demanda. Como peças importantes do sistema elétrico, esses equipamentos necessitam de observação continuada por meio de complexos sistemas de proteção para manutenção de sua integridade e disponibilidade do serviço de geração de forma ininterrupta. Essa observação acerca da proteção do sistema pode se dar por meio de oscilografias que são registros das grandezas do sistema elétrico, tais como correntes e tensões.

A utilização de equipamentos modernos de registro de perturbações possibilitou a centralização dos dados de registros de oscilografias, fazendo com que as análises pudessem ser mais minuciosas. No entanto, essa centralização gerou uma dificuldade para os profissionais que necessitam lidar com esses dados por causa do excesso de informação (BRITO, *et al.*, 2009). A segmentação automática serve para reduzir esse fluxo de dados identificando quais registros merecem atenção dos especialistas das centrais de comando. Esse recurso separa o sinal de oscilografia em segmentos pré-falta, falta e pós-falta, tornando a análise mais rápida.

Existem alguns métodos de segmentação sendo utilizados atualmente, como os mínimos quadrados, a análise do tensor e por transformada *wavelet*. Nesse trabalho será aplicada uma nova metodologia que é baseada no desvio padrão das amostras de um registro de oscilografia. O desvio padrão apresenta um valor médio do quanto as amostras estão distantes da média do sinal analisado.

Como base utiliza-se o método dos mínimos quadrados, que consiste no valor da diferença entre duas amostras consecutivas ou em uma janela de amostras. Os índices baseados nesse método possuem certos problemas, como a incapacidade de detectar múltiplos índices de segmentação e ainda detectar apenas distúrbios bruscos no sinal. O método baseado no desvio padrão pode conter múltiplos segmentos e detectar distúrbios que não são bruscos.

Entre os objetivos do trabalho está efetuar a segmentação automática de forma eficiente, identificando distúrbios em oscilografias reais de sistemas de geração e a junção de segmentos muito próximos no tempo. Está também a identificação de múltiplos segmentos, bem como identificar múltiplos distúrbios no

mesmo registro. Espera-se que ainda um programa seja desenvolvido e que implemente o desvio padrão e que os resultados acima citados sejam obtidos através do uso desse programa.

2. REVISÃO BIBLIOGRÁFICA

Neste capítulo é apresentada uma revisão bibliográfica acerca das metodologias de análise e segmentação automática de registros de oscilografia. O processo de análise de oscilografias envolve passos como tomada de decisão, etc. A ênfase da revisão se dará sobre a etapa de segmentação.

No trabalho de (STYVAKTAKIS; BOLLEN ; GU, 2001) é proposta uma metodologia de segmentação baseada na forma de onda de tensão trifásica (ou corrente), sendo que sua caracterização se dá por meio da análise da magnitude da frequência fundamental e das harmônicas. Os índices de segmentação são encontrados por meio dos quadrados médios dos resíduos de três filtros de Kalman, para cada uma das barras das fases do sistema. Com base nos índices de segmentação, o sinal é separado em evento pré-falta, falta e pós-falta. Os dados utilizados no trabalho são de tensão e corrente, por meio de fasores que contém valores de módulo e ângulo. A metodologia proposta leva em consideração a análise de dados de frequência e distorção harmônica, elementos esses indisponíveis ao trabalho.

Distúrbios com transitórios são geralmente não estacionários tanto no domínio do tempo quanto no da frequência. Uma das ferramentas mais utilizadas para extrair e segmentar sinais desse tipo é a Transformada *Wavelet* Discreta (do Inglês *Discrete Wavelet Transform*). A DWT é útil na análise de sinais com diferentes faixas de frequências por meio da dilatação e da translação de uma *wavelet* mãe. Costa, *et al.* (2011) propõe um método de segmentação baseado no algoritmo de Mallat (análise multirresolução), no qual a DWT utiliza um conjunto de filtros passa-altas (filtro *wavelet*) e passa-baixas (filtro de escala), dividindo a frequência do sinal de entrada em componentes de alta frequência (coeficientes *wavelets* de detalhe) e componentes de baixa frequência (coeficientes *wavelet* de aproximação), na primeira escala. A análise de distúrbios é feita com base em passos pré-definidos, os quais envolvem, por exemplo, detectar os coeficientes *wavelet* das tensões e correntes em várias escalas e avaliar ainda a energia calculada dos coeficientes *wavelet* de tensão e corrente. A análise sobre um sinal senoidal torna a metodologia proposta inviável no desenvolvimento do projeto, sendo que fasores de tensão e corrente estão disponíveis.

JAGUA *et al.* (2010) considera um sistema com n (n é o número de fases) fases e faz a segmentação baseada na teoria da potência instantânea de um tensor. Consideram-se os valores de tensão e corrente instantâneos utilizando o produto das componentes de um tensor e obtém-se a potência instantânea do tensor, que é chamada de componente de tensor e representa um ponto no eixo tridimensional de um cubo. Sendo assim, cada componente de potência pode ser decomposta em três tensores que são chamados de: isotrópico, desvio e antissimétrico. A metodologia propõe a segmentação do sinal analisando-se o ângulo de deformação do cubo. Essa análise é possível por meio de técnicas de álgebra linear, utilizando-se dos vetores com valores de corrente e tensão instantâneos medidos em instantes de tempo sequenciais, de modo a se obter o ângulo desejado. A deformação do cubo se dá em períodos pré-falta, falta e pós-falta. São coletadas amostras de variação de ângulo a cada ciclo da janela de coleta de dados, sendo que um limiar deve ser escolhido, de modo que caso o ângulo ultrapasse determinado valor, considera-se que houve um distúrbio, baseado em casos já conhecidos. Essa metodologia não é adequada ao trabalho proposto, pois trata de valores instantâneos de grandezas elétricas e obtidos a partir de registros senoidais e não fasoriais.

(UKIL; ZIVANOVIC, 2006) propõe uma metodologia baseada na aproximação por meio de estudos de probabilidades, utilizando filtros de Kalman em paralelo. Para chegar a segmentação, o sinal é decomposto em uma DWT, obtendo-se os coeficientes wavelet de detalhe e aproximação. Em seguida utiliza-se o chamado Método do Gatilho, o qual implica na comparação dos coeficientes wavelet com uma média pré-estabelecida, ou seja, o período de tempo onde ocorre alguma variação brusca no sinal é onde os coeficientes wavelet ultrapassam um dado limiar. É difícil aplicar essa metodologia pelo fato de a mesma analisar registros de curta duração e com elevado número de amostras, fazendo com que sejam considerados apenas transitórios rápidos.

(MUSARUDDIN; ZIVANOVIC, 2011) propõe um método chamado de Modo de Decomposição Empírica (EMD, do inglês, *Empirical Mode Decomposition*), que decompõe uma série no tempo em um número finito de Funções de Modo Intrínseco (IMF, do inglês, *Intrinsic Mode Functions*) que são os resíduos utilizados na comparação das amostras. A partir das amostras obtidas no formato COMTRADE, são identificados os extremos das curvas de tensão e corrente de modo a efetuar uma interpolação entre os dados e criar os chamados “envelopes”,

sendo um máximo e um mínimo. É então feita uma comparação entre a média dos envelopes e o valor obtido por meio do processo de "sifting" nos IMFs, gerando resíduos. Entre os passos para a finalização do processo de segmentação, está a aplicação da Transformada de Hilbert, que é utilizada para gerar o gráfico de distribuição de tempo-frequência-energia, também chamado de Espectro de Hilbert. O Espectro de Hilbert é utilizado para identificar as energias em determinados períodos de tempo. Então é feita uma comparação dos valores das energias com limiares pré-determinados, sendo que quando os valores forem maiores que o limiar, ocorre um evento que deve ser considerado. Essa metodologia não é adequada ao trabalho, pois os dados utilizados na metodologia levam em conta registros de curta duração e valores instantâneos de grandezas elétricas.

Um método de segmentação utiliza índices de detecção baseados na diferença entre duas amostras consecutivas de tensão RMS. O índice de detecção é calculado para cada fase e o índice selecionado para a segmentação é o máximo valor obtido entre as fases. O processo de identificação de eventos ocorre a partir de um valor de gatilho, o qual é ajustado em um valor mínimo. Um segmento é identificado quando esse valor é ultrapassado, em uma janela de análise (baseada em ciclos de tempo), de acordo com classificações pré-definidas de distúrbios. Apesar de utilizar valores de grandezas RMS, pode ser adaptada para utilização no projeto proposto, pois o módulo dos fasores no registro de longa duração (ou fasorial) é o valor RMS (STYVAKTAKIS; BOLLEN ; GU, 2002).

(MORETO, 2011) propõe um método de segmentação que avalia índices de detecção de eventos para caracterizar transitórios e/ou mudanças de estado nas grandezas analisadas. É feita uma adaptação para o cálculo dos índices baseando-se na diferença entre os valores absolutos de amostras consecutivas. Quando o índice ultrapassar um limiar pré-definido, considera-se que um evento foi detectado. Uma análise com base no desvio padrão também é feita, dada uma janela de amostras, visando obter um tratamento melhor do sinal com relação a ruídos e resultados inesperados. Sendo assim, dentro de uma janela de amostras, a análise leva em conta quais índices calculados ultrapassam um determinado limiar e indicando, pois, quais as amostras que fazem parte ou não de um transitório no sinal. Utilizam-se os módulos das grandezas e também o número de amostras em um período para que transitórios mais lentos sejam identificados. Essa metodologia

trabalha com os mesmos dados disponíveis ao trabalho proposto, tendo sua aplicação viável.

Algumas das metodologias apresentadas nesta revisão são para registros de curta duração, considerando transitórios rápidos e partindo de dados instantâneos de tensão e corrente. Duas metodologias puderam ser pré-selecionadas para análise, a de (STYVAKTAKIS; BOLLEN ; GU, 2002) e (MORETO, 2011).

Como os dados que serão trabalhados são valores de módulo e ângulo em registros de longa duração, a metodologia proposta por(MORETO, 2011) é a mais viável e pode ser adaptada para segmentar um sinal com múltiplos índices de segmentação.

3. REVISÃO TEÓRICA

As exigências cada vez maiores sobre os serviços prestados pelas empresas que gerem o fornecimento de energia elétrica no que tange a disponibilidade do sistema fez com que a análise de faltas se tornasse crucial. Para que se tenha um sistema confiável deve-se possuir uma grande quantidade de informações a respeito das faltas que ocorrem, tanto para remediar quanto principalmente, prevenir.

A *oscilografia* constitui-se na aquisição e registro de grandezas do sistema elétrico (corrente e tensão, de fase, linha ou de campo), além de sinalizações provenientes dos dispositivos de proteção quando da ocorrência de uma falta no sistema (BRITO, *et al.*, 2009). O *oscilógrafo* é o equipamento utilizado para gerar as oscilografias.

O Registrador Digital de Perturbações (RDP) é um oscilógrafo digital e é um equipamento muito utilizado no monitoramento de sistemas de geração. O RDP armazena vários tipos de grandezas em uma grande quantidade de canais, analógicos e digitais.

Os RDPs possuem canais analógicos da ordem de 8 a 64 canais e uma quantidade maior de canais digitais que variam entre 32 a 384 canais. Armazenam também estados de contadores auxiliares de relés, chaves e disjuntores. (MORETO, 2011)

Entre as grandezas que podem ser monitoradas pelo RDP encontram-se correntes de carga, tensões de fase, correntes de fase e neutro de transformadores (lado de Alta Tensão – AT) bem como correntes e tensão de excitação (MORETO, 2011). A figura 1 ilustra onde o RDP é inserido em uma rede de oscilografias.



Figura 1 - Inserção do RDP na rede de oscilografias.
 Fonte: Adaptado de (BRITO, *et al.*, 2009).

Um arquivo de registro de oscilografias é armazenado em um formato padrão chamado COMTRADE (IEEE) o qual pode conter vários canais registrando várias grandezas diferentes de várias unidades de geração.

Os registros gerados pelos RDPs podem ser de curta duração, fasorial ou contínuos. O registro de curta duração compreende um total de até 20 segundos e possui elevado número de amostras. O registro fasorial, que será tratado neste trabalho, representa os módulos e defasagem angular de tensões e correntes em períodos da ordem de minutos. O registro contínuo, ou histórico do dia, representa valores médios de tensão e corrente em intervalos de 1 minuto a partir de registros fasoriais e compreende várias horas (MORETO, 2011).

A figura 2 ilustra um registro de oscilografia fasorial.

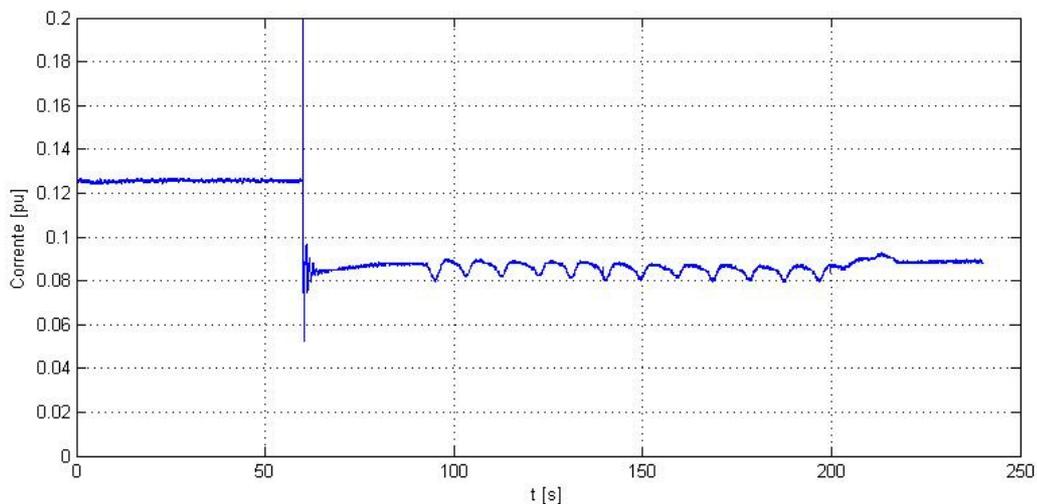


Figura 2 - Registro de oscilografia fasorial.
Fonte: Autoria própria.

A utilização de equipamentos modernos de registro de perturbações possibilitou a centralização dos dados, fazendo com que as análises pudessem ser mais minuciosas. No entanto, essa centralização gerou uma dificuldade para os profissionais que necessitam lidar com esses dados por causa do excesso de informação (BRITO, *et al.*, 2009).

Para a filtragem desse excesso de informação, são utilizadas técnicas de processamento automático dessas oscilografias para que apenas registros com informações úteis sejam enviadas para o banco de dados.

A segmentação automática de oscilografias faz parte de uma metodologia automática de análise de faltas em sistemas de geração de energia elétrica. Essa metodologia envolve os passos constantes no fluxograma ilustrado na figura 3, onde o passo analisado neste trabalho encontra-se destacado.

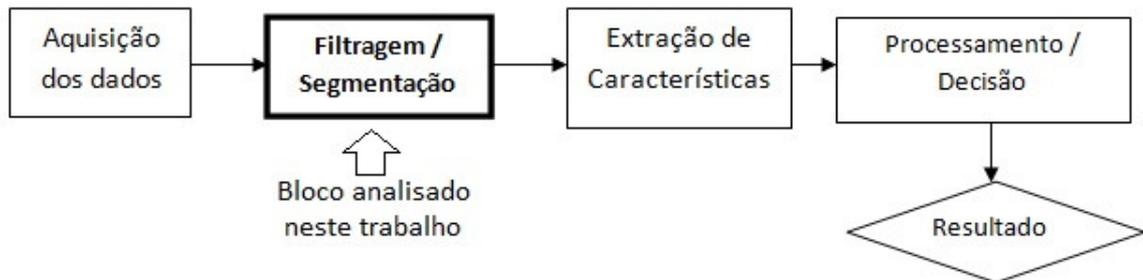


Figura 3 - Fluxograma da análise automática de oscilografias.
 Fonte: Adaptado de (MORETO, 2011).

A *aquisição dos dados* constitui o bloco que adquire os dados de algum sistema de geração de oscilografias (RDP). O bloco da *segmentação* será detalhado a frente. A *extração de características* constitui o bloco em que os registros segmentados são transformados em informações úteis ao tipo de análise que será efetuada. No *processamento/decisão* a informação é processada e o diagnóstico final é obtido com base em definições prévias (BOLLEN; GU, 2006).

A *segmentação* faz parte do processo de filtragem dos registros. Essa filtragem seleciona apenas os registros que contém informações relevantes para análise de distúrbios ocorridos em determinado sistema. A segmentação tem o intuito de separar o registro de oscilografia em múltiplos segmentos, os quais irão distinguir os pontos em que nada foi detectado dos quais ocorreu algum distúrbio. A figura 4 representa um registro de oscilografia fasorial na qual um distúrbio foi identificado e 3 segmentos foram obtidos, isolando a faixa do gráfico que contém a parte relevante. No segmento inicial e final não houve variação significativa de tensão. O segmento central indica um transitório, uma variação da grandeza monitorada. Desta forma, a segmentação consiste em separar um sinal não estacionário em porções que podem ser consideradas estacionárias, permitindo a utilização de técnicas de processamento tradicionais, como a transformada de Fourier.

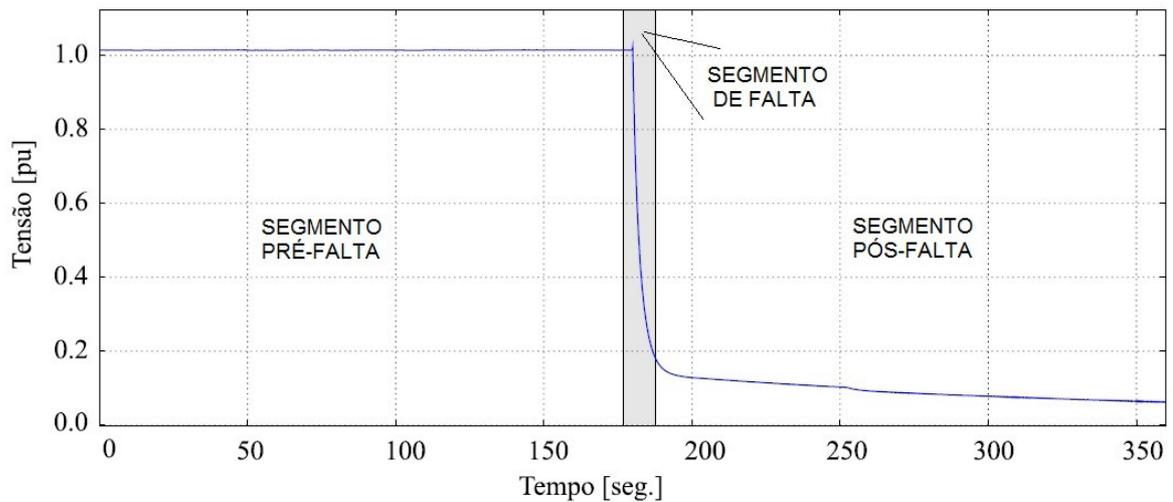


Figura 4 - Segmentação de um registro de oscilografia fasorial.
Fonte: Adaptado de (MORETO; ROLIM, 2010).

Na figura 4, a região hachurada representa a parte em que ocorreu algum tipo de distúrbio. As regiões anterior e posterior da figura representam o segmento inicial e final do registro respectivamente, e indicam a parte do sinal onde a informação a ser analisada é irrelevante. A segmentação é parte importante nas metodologias de diagnóstico automático. Sendo assim, uma segmentação bem feita permite um melhor desempenho das metodologias de extração de características e tomada de decisão que serão executadas posteriormente no processo de análise (figura 3).

4. METODOLOGIAS DE TRABALHO

4.1 METODO DOS MINIMOS QUADRADOS

Para a segmentação automática dos registros de oscilografia são necessários dois programas desenvolvidos para serem utilizados no software Matlab®. Um programa é necessário para interpretar os dados provenientes da biblioteca COMTRADE. Esse programa não será alterado no decorrer do trabalho, haja vista que constitui um módulo básico para o desenvolvimento da metodologia.

O programa que será adaptado foi desenvolvido por (MORETO, 2011) e será aprimorado para poder comportar os requisitos deste trabalho. Entre os objetivos que devem ser desenvolvidos neste programa está a interpretação de múltiplos índices de segmentação, ou seja, múltiplos segmentos em um mesmo registro e a união de segmentos adjacentes.

O trabalho proposto por (STYVAKTAKIS; BOLLEN ; GU, 2002) segmenta o registro de oscilografia com base na diferença absoluta entre duas amostras consecutivas. Quando a diferença entre as duas amostras for maior que um limiar ϵ considera-se que houve um segmento. A equação (1) representa a metodologia proposta:

$$id_1(n) = |u_{rms}(n + 1) - u_{rms}(n)| \text{ para } n = 1 \dots N - 1 \quad (1)$$

Na equação (1), n é o número da amostra e N é o número total de amostras do sinal u_{rms} amostrado. O método (1) possui uma desvantagem. O cálculo da diferença entre amostras consecutivas considera um intervalo de tempo muito pequeno em um registro fasorial, podendo assim gerar segmentos de forma precoce em pontos onde não ocorrem variações.

O trabalho proposto por (MORETO, 2011) segmenta o sinal com base na diferença entre os módulos de duas grandezas fasoriais subsequentes para uma determinada janela de amostras. A equação (2) representa essa diferença:

$$id_2(n) = |u_{rms}(n + \Delta) - u_{rms}(n)| \text{ para } n = 1 \dots N - \Delta \quad (2)$$

Na equação (2), o símbolo Δ é um número inteiro que representa o número de amostras da janela de análise de *id2*.

Um segmento é identificado quando a diferença das duas amostras é maior que certo limiar. A janela de amostragem é variável, ou seja, pode ser ajustada de acordo com a necessidade. Essa variação se faz necessária quando da ocorrência de transitórios mais lentos e não apenas transitórios bruscos. Quanto menos amostras na janela, menor a porção do sinal é analisada e apenas transitórios bruscos são identificados. Os valores da janela podem variar de 30 a 480 amostras por janela. Essa variação se dá com base na frequência de amostragem que é obtida por meio da biblioteca COMTRADE. Para os testes será utilizada uma janela de 30 amostras, que é o valor mínimo para se obter um resultado satisfatório.

O limiar utilizado para disparo da segmentação pode ser tomado como sendo entre *0,01 pu* e *0,05 pu*. Esses valores são obtidos do trabalho de (MORETO, 2011) e tem base em testes e observações em oscilografias reais de unidades geradoras. O valor do limiar deve ser comparado com o resultado obtido anteriormente. Caso o valor obtido for maior que o do limiar, há a geração de um segmento. O limiar pode ainda ser tomado de acordo com o caso em que for aplicado, não precisando seguir um padrão.

Para a comparação dos resultados, foram utilizados 2 programas em conjunto: *ler_fasorial.m* e *EventDetect2_adaptado.m*. O programa *ler_fasorial.m* serve como entrada para o programa principal. Ele é responsável pela leitura de arquivos COMTRADE. Possui como entrada o canal a ser lido e a pasta onde os arquivos com os registros estão localizados. Como saída, retorna uma matriz de *n* linhas por 3 colunas, podendo retornar ainda um vetor com as configurações COMTRADE utilizadas na leitura (opcional). A primeira coluna representa o número da amostra, a segunda coluna representa o tempo onde ela ocorre e a terceira coluna representa o valor do fasor do registro. No programa, o nome dessa matriz ficou padronizada como *dados*. O programa *EventDetect2_adaptado.m* utiliza os dados obtidos do programa *ler_fasorial.m* para analisar o registro e desenhar os segmentos pelo método dos mínimos quadrados em sua segunda versão (*id2*).

Para os cálculos serão utilizados dados em base *Por Unidade (pu)*. Os registros de oscilografia utilizados para os testes foram obtidos da Usina Termoelétrica de Charqueadas (UTCH), no Rio Grande do Sul. Sendo assim, são

utilizadas três bases diferentes para representar três sistemas diferentes. Para isso, no quadro 1 são expostas as bases de tensão e potência utilizadas para cada ponto.

Quadro 1 – Bases de cálculo das usinas.

Fonte: Autoria própria.

USINA	TENSÃO DE LINHA (V)	POTENCIA (MVA)	TENSÃO BASE TRANSFORMADOR LADO AT (V)
UHPF	13800	113	230000
UHMA	16000	418,5	525000
UTCH	6000	24	66000

Os outros valores base citados no quadro 1 dizem respeito a Usina Hidrelétrica de Passo Fundo (UHPF) e a Usina Hidrelétrica de Machadinho (UHMA). Como citado acima, as oscilografias utilizadas para os testes são da UTCH, mas podem muito bem ser das outras duas usinas, haja vista que suas bases também são listadas. O quadro 1 expõe a versatilidade do programa de poder ser utilizado para mais de uma usina, porém, para os testes deste trabalho será utilizada apenas a base da UTCH.

Os valores base são identificados pelo próprio arquivo COMTRADE utilizado, que informa de qual ponto o registro foi obtido. Para o caso de não ser identificada a usina, utiliza-se a UHMA como referência. A obtenção de uma usina como referência pode ser alterada diretamente no programa, alterando a saída conforme interesse. Conceitos básicos de sistemas de potência são aplicados no programa para calcular a conversão das grandezas para pu , tais como a obtenção de tensões, correntes e potências de base. Este trabalho irá aplicar um terceiro índice de segmentação que envolve a utilização do desvio padrão para uma janela de amostras. A utilização do desvio padrão torna o sistema menos suscetível a erros.

4.2 METODO DO DESVIO PADRÃO

Antes da introdução do equacionamento do desvio padrão, deve-se observar a equação que obtém a média da porção do sinal dentro da janela de amostragem. Em (3) encontra-se o cálculo da média (SMITH, 1997).

$$\mu = \frac{1}{N} \sum_{i=0}^{N-1} x_i \quad (3)$$

Na equação (3), x_i é a amostra i no tempo. Sendo assim, o índice utilizado calculado pelo desvio padrão está representado por (4):

$$id_3^2(n) = \sigma_\Delta^2(n) = \frac{1}{\Delta-1} \sum_{i=N}^{N+\Delta} (|\vec{y}(i)| - \mu)^2 \quad (4)$$

Em (4), N é o índice da amostra no registro de oscilografia, y representa o módulo do fasor na amostra i , μ representa a média calculada para a janela de análise, Δ representa o número total de amostras na janela de análise, σ_Δ representa o desvio padrão calculado para esta janela. Para os testes serão utilizados 30 amostras por janela de amostragem, o que registra 60 fasores por segundo.

O cálculo do desvio padrão consiste em obter-se o valor médio do sinal dentro da janela de amostragem. Após esse passo, utiliza-se o módulo de cada amostra N da janela (que possui módulo e ângulo) e subtrai-se do valor médio μ . A partir disso, é calculada a média desse resultado, informando um valor médio do quanto as amostras estão distantes da média do sinal (SMITH, 1997).

Dois programas bases foram utilizados para teste da metodologia proposta. Os nomes dos arquivos são *ler_fasorial.m* e *índice_3.m*. Os dois arquivos necessitam estar na mesma pasta para que o programa de segmentação funcione, haja vista que um depende do outro.

O programa *ler_fasorial.m* teve sua função descrita anteriormente. O programa *índice_3.m* é utilizado para ajustar os cálculos e efetuar a segmentação dos registros de oscilografia por meio da metodologia do desvio padrão.

A matriz *dados* obtida por *ler_fasorial.m* é utilizada em todo o desenvolvimento do código de *id3* assim como em *id2*. Possui as mesmas

características e informações mencionadas anteriormente e é de extrema importância no decorrer da execução do programa.

Para obtenção das bases é feita a comparação da variável *usina* que contém uma informação vinda do programa *ler_fasorial*. De acordo com a usina listada, utilizam-se os valores do Quadro 1 para o cálculo das bases. Em seguida, deve-se calcular as bases conforme equacionamento a seguir.

$$V_{bF} = \frac{V_{bL}}{\sqrt{3}} \quad (5)$$

$$S_{b1} = \frac{S_{b3}}{3} \quad (6)$$

$$I_{b1} = \frac{S_{b1}}{V_{bF}} \quad (7)$$

Na equação (5), V_{bF} é a tensão de base de fase, V_{bL} é a tensão de base de linha ambos em V. Na equação (6), S_{b1} é a potência de base monofásica e S_{b3} é a potência aparente de base trifásica, ambos em MVA. Na equação (7) I_{b1} é a corrente de base monofásica em A.

Com base nesses cálculos e numa variável chamada *unidade* que armazena qual tipo de grandeza representa o sinal de oscilografia, determina-se qual a base que será utilizada para o resto do programa. O próximo passo será calcular o desvio padrão.

A metodologia proposta para o índice de detecção 3 (*id3*) consiste em substituir o método dos mínimos quadrados (módulo das diferenças entre os fasores de duas amostras) implementado por (MORETO, 2011) pelo desvio padrão. Essa substituição de metodologia tende a precisar mais a obtenção dos valores onde deverá ocorrer a segmentação, bem como pode-se selecionar melhor os segmentos. Os resultados obtidos entre as metodologias não são correspondentes, mas podem ser comparadas para um mesmo limiar. O código do programa *índice_3.m* está comentado e em sua forma completa no Apêndice A. A seguir será apresentado e comentado cada ponto importante do código que implementa o *id3*. As descrições do código serão feitas por linha.

```
1 - canal_rdp = 31;
```

```
2 - intervalo = [3100 4100];
```

```

3 - limiar = 0.004;
4 - DeltaT = 30;

5 - [dados cfg] = ler_fasorial('E:\Oscilografias\', canal_rdp);

6 - canal = find(cfg.analogico_numero == canal_rdp);

7 - usina = cfg.nome(1:4);

8 - disp(sprintf('%s, Canal: %d,
%s', usina, canal_rdp, cfg.analogico_identificacao(canal, :)));

```

No código acima, a linha 1 recebe o número do canal do RDP a ser lido; a linha 2 recebe o intervalo de amostragem; a linha 3 recebe o limiar para disparo da segmentação; a linha 4 recebe o intervalo da janela de amostragem; a linha 5 recebe o caminho para o programa *ler_fasorial.m* que irá obter os valores numéricos e configurações do registro de oscilografia do respectivo canal; a linha 6 faz com que a variável *canal* receba o valor de configuração do programa *ler_fasorial.m*; na linha 7 a variável *usina* recebe um valor de 1 a 4 que corresponde a usina indicada no arquivo COMTRADE para o cálculo posterior das bases; a linha 8 simplesmente imprime os dados obtidos na tela.

```

->     if usina == 'UHPF'

->         VbaseL = 13800; .
->         Sbase3 = 113;
->         VbaseTR = 230000;
->     elseif usina == 'UHMA'

->         VbaseL = 16000;
->         Sbase3 = 418.5;
->         VbaseTR = 525000;
->     elseif usina == 'UTCH'

->         VbaseL = 6000;
->         Sbase3 = 24;
->         VbaseTR = 66000;
->     else

->         disp('Usina não identificada. Serão utilizadas as bases da UHMA.');
```

```

->         VbaseL = 16000;
->         Sbase3 = 418.5;
->         VbaseTR = 525000;
->     end

```

O trecho de código acima não será analisado linha a linha. Esse trecho compara a variável *usina* e seleciona uma base para efetuar os cálculos que serão explicitados a seguir.

```

9 -> T = 1/cfg.frequencia;

10 -> VbaseF = VbaseL/sqrt(3);
11 -> Sbase1 = Sbase3*1e6/3;
12 -> Ibase1 = Sbase1/VbaseF;
13 -> IbaseTR = Sbase1/(VbaseTR/sqrt(3));

14 -> N = size(dados,1);

15 -> unidade = cfg.analogico_unidade(canal,:);

16 ->     if  unidade == 'A '
17 ->         base = Ibase1;
18 ->         grandeza = 'Corrente [pu]';
19 ->     elseif unidade == 'kV'
20 ->         base = VbaseF/1000;
21 ->         grandeza = 'Tensão [pu]';
22 ->     end

```

Na linha 9 é obtido o período de amostragem em unidade de tempo; nas linhas 10 a 13 são calculadas as bases para a usina selecionada; na linha 14 N recebe a quantidade de amostras presentes em dado registro; na linha 15, *unidade* recebe a unidade monitorada por meio de consulta ao arquivo COMTRADE para o canal selecionado; nas linhas 16 a 22, por meio da unidade informada, é definida a base que será utilizada para os cálculos posteriores. A base obtida já está em valor *pu*.

```

23 -> k = 1;
24 -> m = 1;
25 -> t = 1;
26 -> f = 1;

27 -> while (k * DeltaT <= N)
28 ->     m = k * DeltaT;

29 ->     if (k == 1)
30 ->         n = 1;
31 ->     else
32 ->         n = (k-1)*DeltaT;
33 ->     end

```

Na implementação do desvio padrão, pode-se considerar um grande bloco de código, o Bloco A. Esse bloco possui várias iterações internas responsáveis pelo resultado final de *di3*. Cada iteração será analisada a seguir. Vale ressaltar que os códigos analisados dentro do bloco A são atualizados a cada vez que a janela de amostragem é incrementada de *DeltaT* amostras.

Nas linhas 23 a 26 são feitas definições de variáveis que serão utilizadas no laço *while*. A linha 27 indica o início do bloco A. A rotina será executada até que o número total de amostras seja alcançado. Na linha 28, m assume o valor de k vezes o número de amostras da janela em análise, sendo sempre o final da janela de amostragem.

Nas linhas 29 a 33 é feita uma lógica para incrementar n . No caso, n precisa ser incrementado de ΔT toda vez, assumindo o valor inicial da próxima janela. Exemplificando numericamente, caso ΔT seja igual a 30, n inicia em 1 e logo após a primeira iteração, n irá assumir o valor 30 enquanto m , na linha 28, irá assumir o valor 60 e assim por diante até o laço atingir o valor menor ou igual a N .

```
34 -> t2(k,1) = dados(n,2); % Vetor de tempo.
35 -> n_amostras(k,1) = n;
```

Na linha 34 é criado o vetor de tempo baseado na segunda coluna da matriz *dados*, que contém o tempo em que cada amostra ocorre. Na linha 35, *n_amostras* corresponde ao número de amostras final.

A variável μ na equação (3) é obtida a partir do seguinte trecho de código:

```
36 -> for l = n:m
37 ->     v(l,1) = abs(dados(l,3))/base;
38 -> end
```

Na linha 36, a variável criada l varia dentro do laço *for* desde a amostra inicial (n) até a amostra final da janela (m), ou seja, caso a janela de amostragem seja de 30 amostras, n será igual a 1 e m será igual a 30. É obtido o módulo da amostra l dividido pela base utilizada, fazendo m iterações. Em seguida, na linha 37 é efetuada a soma dos valores obtidos em no vetor v e o resultado é dividido pelo número de amostras dentro da janela (no caso, ΔT).

O seguinte trecho de código executa a soma e a divisão, resultando na média μ da equação (3):

```
39 -> for l = n:m
40 ->     soma = soma + v(l,1);
```

```
41 -> end
42 -> media = soma/30;
```

A linha 39 realiza a mesma iteração da linha 36. O procedimento de soma na linha 40 possui o mesmo número de iterações m , porém agora, a cada iteração o vetor *soma* pega o valor anteriormente armazenado e soma com o valor da próxima amostra até o final do laço.

O próximo passo é obter a parte interna ao somatório da equação (4). Essa parte do programa envolve subtrair os valores dos módulos de cada amostra dentro da janela do valor da média das amostras dentro da mesma janela. O valor resultante deve ser elevado ao quadrado. Em seguida é efetuada a soma de todos os termos, o que resultará em *somadev*. O trecho de código a seguir representa esse cálculo:

```
43 -> for l = n:m
44 ->     dif(l,1) = (v(l,1) - media)^2;
45 ->     somadev = somadev + dif(l,1);
46 -> end
```

Na linha 42, a variável l varia conforme anteriormente citado sobre as linhas 36 e 39, *dif* na linha 43 representa a parte interna do somatório da equação (4) e *somadev* representa o somatório de todos os termos da janela executando uma função parecida com a soma da linha 40.

Para obtenção do índice de detecção em sua forma completa, deve-se extrair a raiz quadrada do somatório obtido na função anterior. Sendo assim, o seguinte trecho de código obtém o valor final de *id3* para o índice k .

```
47 -> di3 = somadev/29;
48 -> id3(k,1) = sqrt(di3);
```

Na linha 46, *di3* representa o resultado de *somadev* dividido pelo número de amostras menos 1 conforme indicado em (4). Na linha 47, o índice *id3* representa o índice de segmentação 3 para um índice k , para k variando de 1 até o número

máximo de amostras registrados, que no caso dos testes efetuados chega a 14480 amostras (60 fasores por segundo). Após esse passo, k é incrementado de uma unidade e varia até atingir o valor máximo de janelas de 30 amostras no período analisado do sinal.

Posteriormente, o código utiliza funções do próprio software para traçar os parâmetros do gráfico, tais como legendas, se o gráfico irá possuir um *grid* para demarcação dos tempos, a escala das legendas, etc.

Em seguida, é implementada uma função que encontra os pontos na linha do tempo em que ocorre a extrapolação do limiar pré-definido. A seguinte função executa esse passo:

```
49 -> indices_di = find(id3 > limiar);
50 -> c = size(indices_di,1);
```

O processo de obtenção dos índices é semelhante nas duas metodologias. Na linha 48, a função *find* retorna os valores onde *id3* é maior que o limiar. Em seguida, na linha 49 é criada uma variável *c* que será utilizada no Bloco B. Essa variável *c* contém a quantidade de termos presentes no vetor *índices_di*, ou seja, utiliza a função *size*.

O bloco B contém o trecho do código que identifica os índices onde ocorre a extrapolação do limiar e traça uma linha vertical na amostra inicial e final do segmento.

A lógica do bloco B consiste em traçar uma linha vertical identificando os locais onde inicia e termina determinado segmento. Para o caso da ocorrência de um distúrbio maior que o limiar um *if* é implementado:

```
51 -> if (isempty(indices_di) == 0)
```

Na linha 50 a função *isempty* verifica se *índices_di* está vazio ou não. Caso o valor seja 1 (verdadeiro), ele sai do *if* e retorna a mensagem de não ocorrência de evento. Para o caso em que o valor é igual a zero, significa que *índices_di* não é vazio e possui índices em que o limiar é extrapolado, indicando a ocorrência de um evento. Sendo assim, a rotina entra em um laço *for* que irá percorrer uma contagem equivalente ao valor do tamanho do vetor *índices_di* por

meio da função *size*. Essa função conta quantos elementos determinado vetor possui.

```
52 -> for i = 1:c
```

Na linha 51 é definido o laço que implementa a inserção dos segmentos é definida. Esse laço irá variar desde 1 até o valor de índices encontrados *c*. A função *line* traça a linha vertical nos tempos dos índices selecionados. Sendo assim, por padrão, são traçadas linhas verticais no primeiro e no último valor de *c*, ou seja, no primeiro e último índice de detecção. Sendo assim, o seguinte trecho efetua esse procedimento:

```
53 ->         if (i == c)
54 ->             line([t2(indices_di(i)) t2(indices_di(i))],
                    altura, 'color', 'r', 'LineStyle', '-');
55 ->         end
56 ->         if (i == 1)
57 ->             line([t2(indices_di(i)) t2(indices_di(i))],
                    altura, 'color', 'r', 'LineStyle', '-');
58 ->         end
```

O programa tem a funcionalidade de unir segmentos em períodos de tempo próximos. Sendo assim, quando índices de valores seguidos são detectados, eles são ignorados e apenas os índices iniciais e finais são segmentados. Nas linhas 51 a 56 a linha traçada representa os segmentos inicial e final de determinado registro com distúrbio.

A seguir é implementada a lógica que identifica se dois índices de detecção são subsequentes ou não. Essa lógica é necessária para poder separar o registro em vários segmentos porém ignorando segmentos em sequência. Para isso, é necessário comparar *i* e *c*. Caso *i* seja diferente de *c*, o índice *n* é intermediário, necessitando análise. Duas variáveis são criadas, *a* e *b*. Essa recebe o valor do índice *i+1* e aquela recebe o valor do índice *i*. Se o valor da subtração desses dois índices resultar em um valor maior que 1 ($h > 1$) é traçada uma linha vertical em *a* e *b*. Essa lógica representa o traçado da linha reta no ultimo índice do primeiro

segmento (*a*) e no primeiro índice do segundo segmento (*b*). O seguinte trecho representa essa lógica:

```
59 -> if (i ~= c)
60 ->         a = indices_di(i);
61 ->         b = indices_di(i+1);
62 ->         h = b - a;
63 ->         if (h > 1)
64 ->             line([t2(a) t2(a)], altura, 'color', 'r', 'LineStyle',
65 ->                 '-');
66 ->             line([t2(b) t2(b)], altura, 'color', 'r', 'LineStyle',
67 ->                 '-');
68 ->         end
69 -> end
```

Para o caso do *if* não ser executado, o laço *for* inicial continua sendo executado até que *i* se torne igual a *c*.

Chegado o ponto final do código, a segmentação ocorre para o caso de um registro com algum distúrbio e uma mensagem informando, caso não ocorra nenhuma falta.

A figura 5 ilustra um diagrama de blocos que representa os passos de execução do programa.

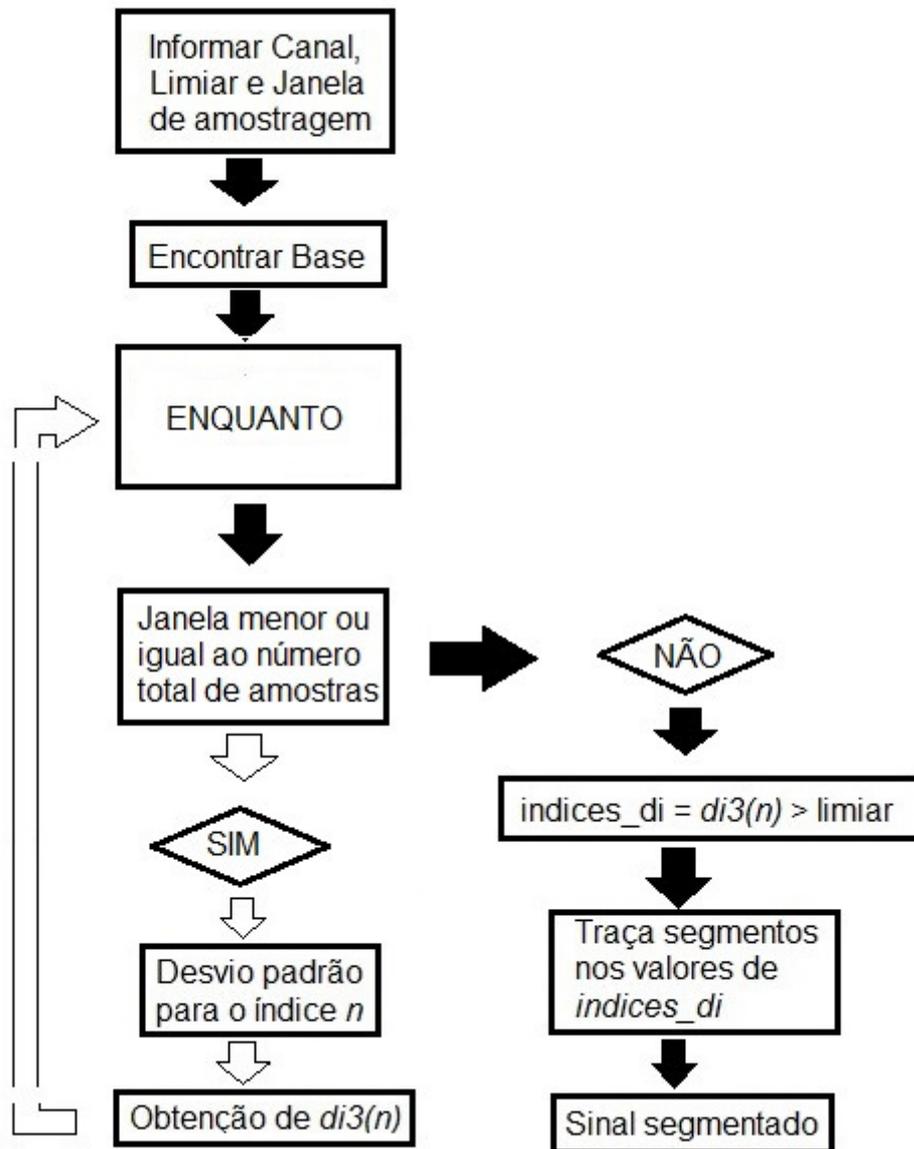


Figura 5 - Diagrama de blocos de execução do código.
 Fonte: Autoria própria.

5. RESULTADOS E DISCUSSÕES

Nesse tópico serão efetuadas comparações entre os programas utilizando a metodologia dos mínimos quadrados e do desvio padrão. Os valores dos limiares utilizados nos testes estão dentro da margem citada anteriormente.

Pelas informações obtidas no arquivo COMTRADE, o primeiro registro analisado está na base UTCH, foi gerado no RDP 2 e é um dado de corrente na fase C do gerador 3. O número do canal é 15, o limiar utilizado nesse caso foi de 0,0178 pu e são levadas em conta 30 amostras na janela de amostragem.

Utilizando o programa EventDetect2_adaptado.m obteve-se o resultado ilustrado na figura 6, que representa o registro de oscilografia sendo segmentado pelo método dos mínimos quadrados.

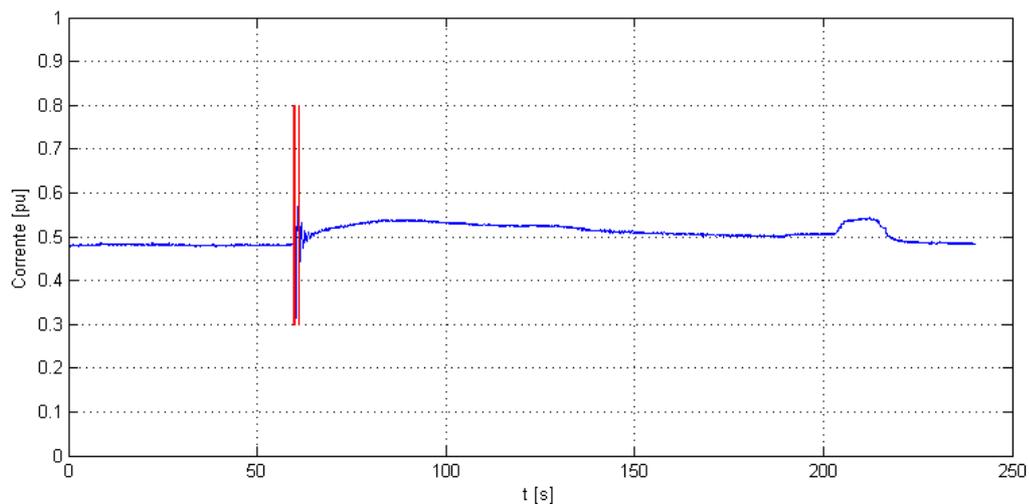


Figura 6 - Segmentação utilizando a metodologia dos mínimos quadrados.
Fonte: Autoria própria.

O intuito do programa é gerar segmentos no início e no final da parte do sinal onde ocorreu algum distúrbio. A figura 7 ilustra o registro gerado em escala ampliada. Ocorreram três variações significativas no sinal. Uma mais significativa, que foi sinalizada com os traços vermelhos, e dois transitórios mais lentos, a direita, que não foram identificados. O programa conseguiu gerar segmentos apenas no primeiro transitório brusco.

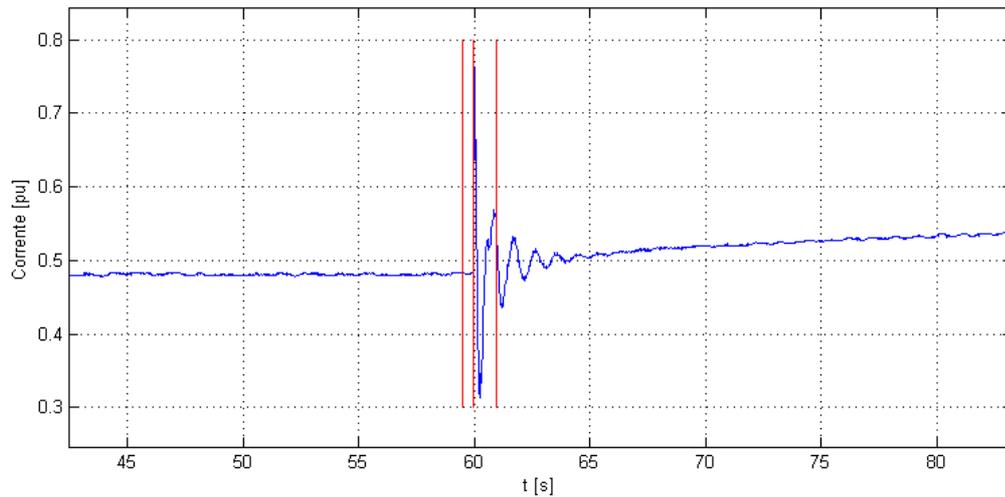


Figura 7 - Registro segmentado em escala maior.
Fonte: Autoria própria.

Pela figura 7 nota-se que foram gerados segmentos onde houve a extrapolação do limiar pela diferença entre amostras. A figura 8 representa o mesmo caso segmentado pelo índice de detecção 3.

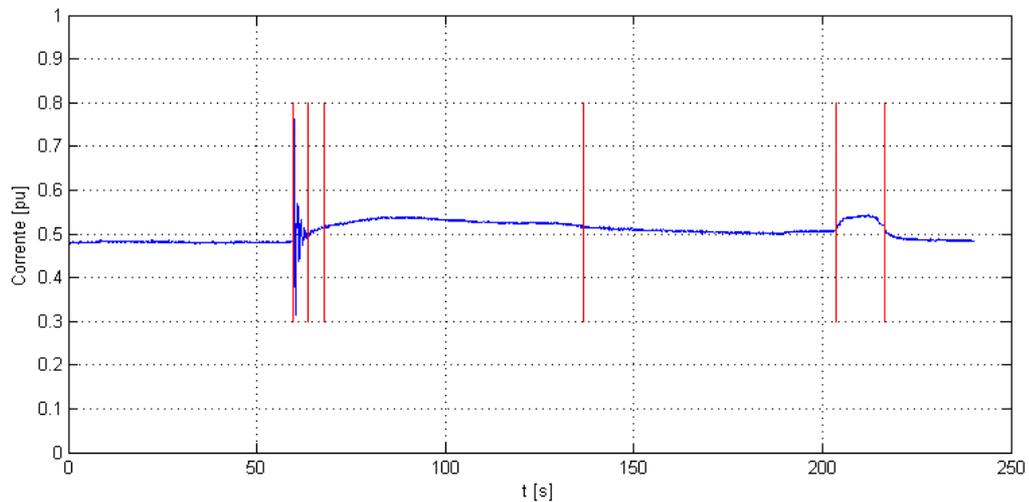


Figura 8 - Segmentação utilizando o método do desvio padrão.
Fonte: Autoria própria.

A análise da figura 8 sugere que mais segmentos foram gerados, agora em três possíveis transitórios. Foram gerados segmentos iniciais e finais para cada um dos três transitórios. Ampliando a escala da figura 8, nota-se que os segmentos agora cobrem toda a porção do sinal onde ocorreu algum tipo de distúrbio. Essa constatação pode ser feita a partir da figura 9.

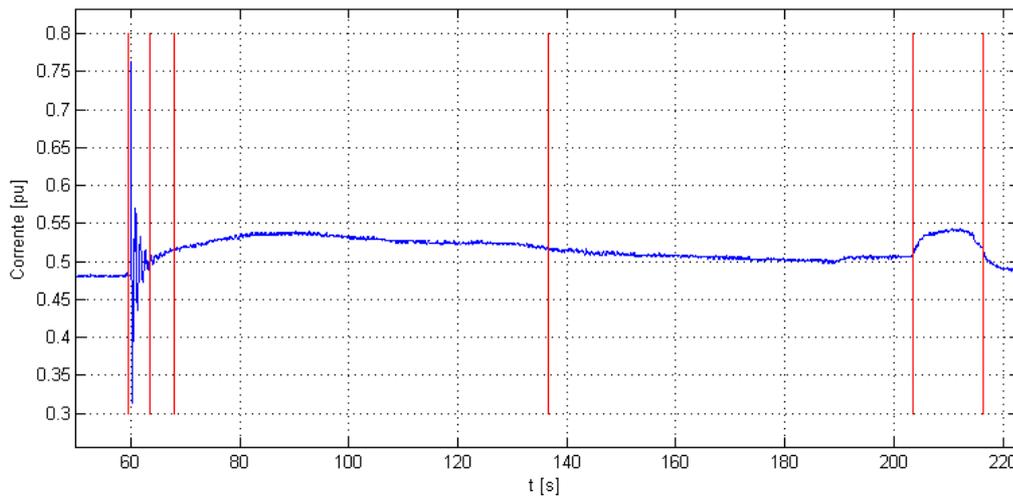


Figura 9 - Registro segmentado em escala maior.
Fonte: Autoria própria.

Para o exemplo citado acima, obteve-se ainda uma tabela que compara os valores obtidos para os índices encontrados pelos dois métodos testados para o mesmo limiar. Os resultados comparativos estão listados na tabela 1.

Tabela 1 - Exemplo 1: Comparação entre mínimos quadrados (*id2*) e desvio padrão (*id3*).

SEGMENTOS	INDICES DE SEGMENTAÇÃO			
	<i>ID2(n)</i>		<i>ID3(n)</i>	
	Indice (<i>n</i>)	Valor (<i>pu</i>)	Indice (<i>n</i>)	Valor (<i>pu</i>)
INICIAL	120	0,1431	120	0,0339
FINAL	121	0,1177	128	0,0179
INICIAL	123	0,307	137	0,0178
FINAL			274	0,0178
INICIAL			408	0,0180
FINAL			434	0,0179

Fonte: Autoria própria

A tabela 1 indica uma comparação entre os métodos, indicando em quais índices *n* ocorreram os segmentos e qual o valor obtido em *pu* para cada segmento. A tabela indica ainda o índice onde ocorreu o segmento inicial e final de cada transitório.

Analisando a tabela 1 pode-se concluir que o número de índices segmentados na nova metodologia para uma mesmo limiar (0,0178 *pu*) é bem

superior a quantidade obtida pelo método dos mínimos quadrados. A diferença se dá principalmente pelo fato de a diferença obtida no índice 2 ser absoluta entre uma amostra e outra, não levando em conta uma média dentro da janela de amostragem, enquanto no índice 3 é levada em conta a média das amostras na janela. A tabela ainda mostra que para o índice 3 há um segmento inicial e final para cada um dos três transitórios, enquanto que no índice 2, apenas um transitório possui início e fim. A grande distância entre valores presente na tabela representa que os resultados intermediários foram omitidos e apenas o final e inicial foram levados em conta.

Em consulta ao arquivo COMTRADE, o registro analisado a seguir está na base UTCH, foi gerado no RDP 2 e é um dado de tensão na fase B do gerador 1. O número do canal é 5, o limiar utilizado nesse caso foi de 0,0358 pu e são levadas em conta 30 amostras na janela de amostragem.

Utilizando o programa EventDetect2_adaptado.m obteve-se o resultado ilustrado na figura 10. Tal como na figura 6, o registro de oscilografia foi segmentado pelo método dos mínimos quadrados.

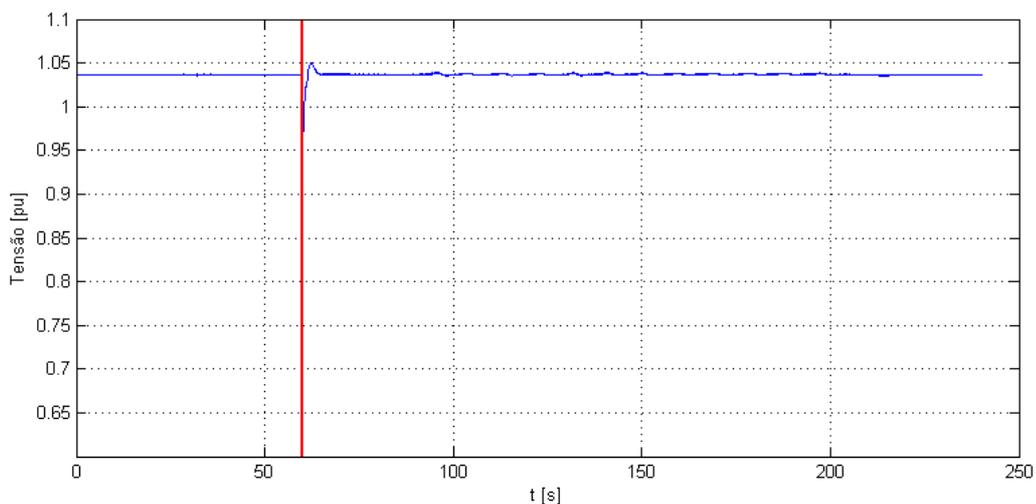


Figura 10 - Segmentação utilizando a metodologia dos mínimos quadrados.
Fonte: Autoria própria.

O trecho destacado em vermelho identifica que ocorreu um distúrbio brusco no registro. Deve-se notar que o trecho em vermelho não cerca totalmente a porção do sinal que contém o transitório. A figura 11 ilustra o registro gerado em escala ampliada.

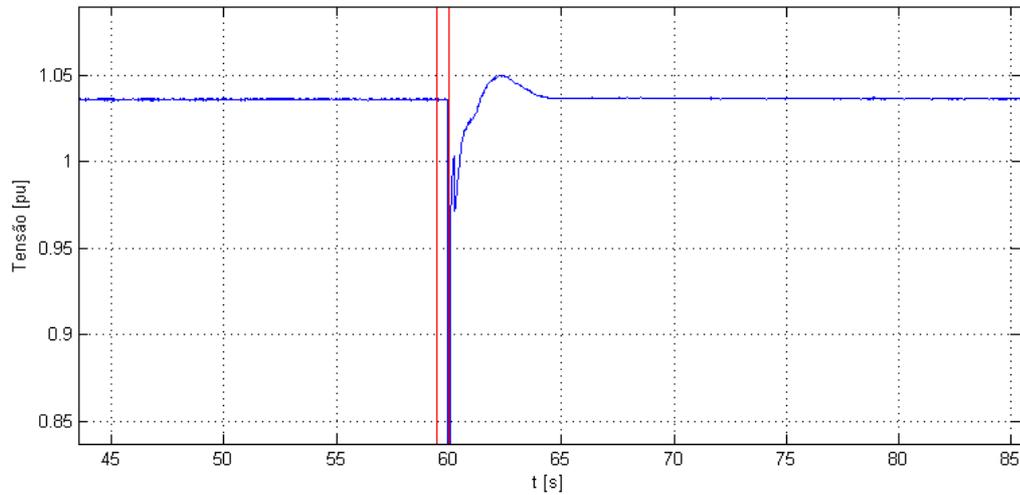


Figura 11 - Registro segmentado em escala maior.
Fonte: Autoria própria.

Foram gerados apenas dois segmentos, sendo eles gerados apenas em índices que identificam o início do transitório. Apenas em dois segmentos do sinal o programa detectou que a diferença é maior que o limiar ($0,0358 \text{ pu}$). A figura 12 representa o mesmo caso, porém agora segmentado pelo índice de detecção 3.

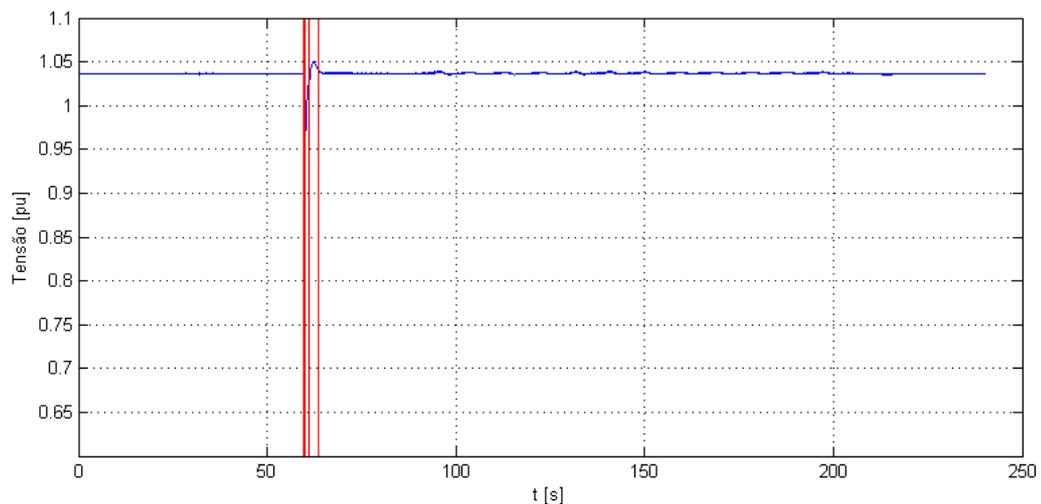


Figura 12 - Segmentação utilizando o método do desvio padrão.
Fonte: Autoria própria.

Ampliando a escala da figura 12, nota-se que os segmentos agora cobrem toda a porção do sinal onde ocorreu algum tipo de distúrbio. Essa constatação pode ser feita a partir da figura 13.

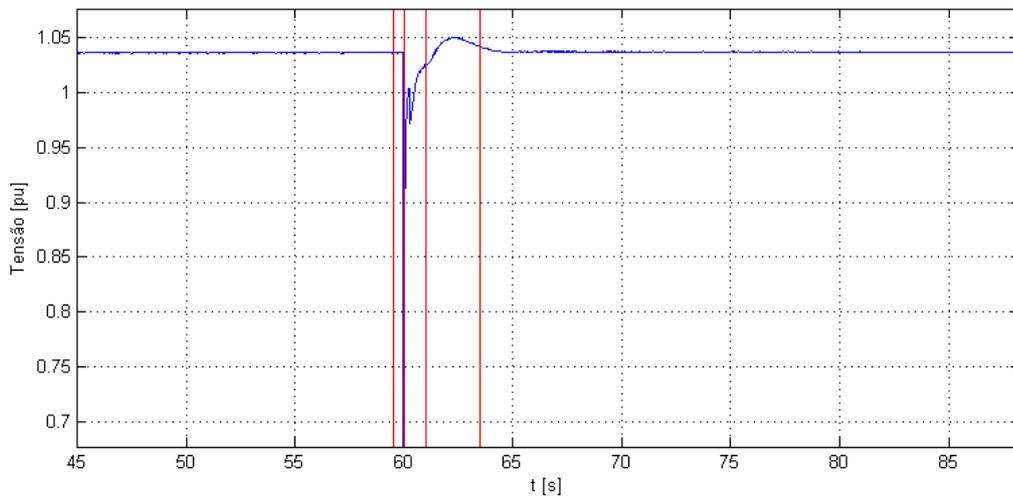


Figura 13 - Registro segmentado em escala maior.
Fonte: Autoria própria.

Foram gerados quatro segmentos. Os dois gerados pelo método anterior, porém agora, com mais dois segmentos gerados indicando o final do transitório. Nesse caso, considera-se o primeiro e o último segmento como válidos e que varrem toda a porção do sinal em que ocorreu o transitório. Pelo índice 3 pode-se observar que agora o primeiro segmento e o último indicam o início e o fim do transitório de forma correta. Os segmentos intermediários não foram ignorados porque houveram índices que não ultrapassaram o limiar.

Para o exemplo citado acima, obteve-se ainda uma tabela que compara os valores obtidos para os índices de cada um dos dois métodos testados para o mesmo limiar. Os resultados comparativos estão listados na tabela 2.

Tabela 2 - Exemplo 2: Comparação entre mínimos quadrados (*id2*) e desvio padrão (*id3*).

SEGMENTOS	ÍNDICES DE SEGMENTAÇÃO			
	<i>ID2(n)</i>		<i>ID3(n)</i>	
	Índice (<i>n</i>)	Valor (<i>pu</i>)	Índice (<i>n</i>)	Valor (<i>pu</i>)
INICIAL	120	0,4161	120	0,1127
FINAL	121	0,3821	121	0,1181
INICIAL			123	0,0358
FINAL			128	0,0359

Fonte: Autoria própria.

Pela tabela 2 conclui-se que pelo índice 1, apenas dois índices extrapolaram o limiar. No índice 3 ocorreram 7 extrapolações, levando em conta que entre 123 e 128 ocorreram 5 variações.

Quando o programa não detecta nenhum distúrbio, ou seja, o limiar de detecção não é ultrapassado, uma mensagem é exibida informando que não ocorreu nenhum evento. A figura 14 ilustra essa funcionalidade.

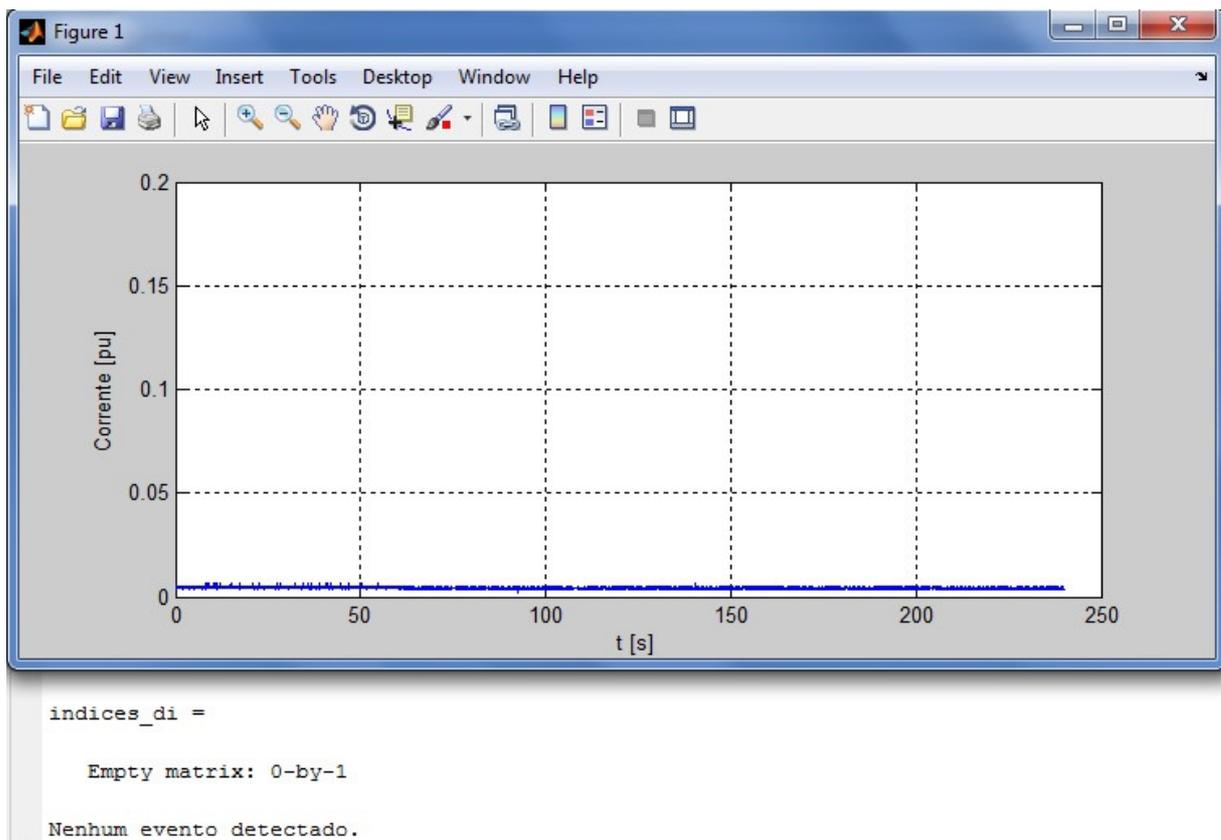


Figura 14 - Indicação do programa quando não ocorre nenhum evento.
Fonte: Autoria própria.

No caso da figura 14, como nenhum distúrbio foi identificado, o registro é descartado e não é enviado a central de comando. Para o caso de não haver a metodologia de segmentação automática na rede de oscilografias, essa função de identificar se o registro possui informações úteis ou não seria de um analista, o que iria tomar um tempo precioso. Com o advento das metodologias de segmentação, esse serviço é feito de forma automática, tornando o trabalho dos analistas mais focado ao estudo dos distúrbios.

6. CONCLUSÕES

A análise de registros de oscilografias em sistemas de geração tem se tornado cada vez mais importante na análise de faltas para tomadas de decisão com relação a proteção do sistemas. Um registro de oscilografia pode conter informações muito relevantes para que sejam tomadas medidas no âmbito da manutenção dos sistemas, bem como a predição de possíveis falhas.

A segmentação foi testada em registros de oscilografia tanto de corrente quanto de tensão e os resultados obtidos foram satisfatórios. Sendo assim, o programa executa sua função em sinais de qualquer uma das duas grandezas, estando sujeito apenas a escolha de um limiar apropriado de acordo com as bases em que se deseja trabalhar.

Pode-se afirmar que a metodologia implementada atendeu aos objetivos principais do projeto segmentando os registros de forma automática, identificando múltiplos segmentos e unindo segmentos próximos em um único segmento, tudo isso respeitando o limite de amostras da janela de análise. A identificação automática de oscilografias em que não ocorre algum distúrbio reduz potencialmente o trabalho dos analistas de proteção de sistemas de geração.

A geração de segmentos do programa implementado no trabalho possui a vantagem de identificar múltiplos distúrbios em um mesmo registro, tornando ele mais versátil do que outros programas que apenas identificam três segmentos (pré-falta, falta e pós-falta). Quanto mais segmentos gerados, mais transitórios são detectados e o sinal fica mais fácil de ser analisado em partes por metodologias de processamento de sinais.

A utilização de registros de longa duração ainda é pouco difundida devido a dificuldade de lidar com transitórios lentos, sendo que as metodologias estudadas abordam apenas transitórios bruscos e registros de curta duração. A análise em sistemas de geração também se torna possível com essa abordagem, dando ainda mais campo de pesquisa acerca do assunto.

Sendo assim, com a aplicação do programa, os registros de oscilografia que possuírem algum tipo de variação (distúrbio) serão pré-selecionados automaticamente, sendo que os locais onde houver alguma variação serão

destacados. A partir disso, pode-se prever um fluxo de registros muito menor de registros gravados no sistema dos centros de controle.

Para trabalhos futuros, pode-se citar a identificação dos tipos de falhas a que com base nos registros de oscilografia. Já existem projetos de sistemas automáticos de análise de oscilografias, que, com base em experiências reais de faltas, identificam o tipo de falha do sistema, reduzindo ainda mais o tempo de análise dos registros. A segmentação de registros de oscilografia torna o sinal estacionário por partes, fazendo com que a aplicação de técnicas como a Transformada de Fourier sejam possíveis na hora de identificar o motivo dos distúrbios de forma automática. Com adaptações, esse programa pode servir também para identificar distúrbios em outros sistemas físicos, dado que a janela de amostragem é variável e a frequência de amostragem também.

REFERÊNCIAS

BOLLEN, Math H.; GU, Irene. **Signal Processing of Power Quality Disturbances**. 1^a. ed. [S.l.]: Wiley IEEE - Press, 2006.

BRITO, Núbia S. D. et al. **Rede de oscilografia da CHESF: da concepção até o estado atual**. XIII Encuentro Regional Iberoamericano de CIGRÉ. Puerto Iguazu, Argentina: [s.n.]. 2009. p. 8.

COSTA, Flávio B. et al. **A Tool for Fault and Power Quality Disturbance Analysis in Oscillographic Records**. 3rd International Youth Conference. [S.l.]: CNPq, Fapesq. 2011. p. 1-6.

JAGUA, Jorge et al. **Waveform Segmentation Based On Tensor Analysis**. IEEE Andean Conference, Exhibition and Industry Forum. Colombia: [s.n.]. 2010. p. 7.

MORETO, Miguel. **Análise automática de registros digitais de perturbações em unidades geradoras**. 2011. 203p. Tese (Doutorado em Engenharia Elétrica) - Universidade Federal de Santa Catarina, Florianópolis, 2011.

_____ ; ROLIM, Jacqueline G. Análise automática de oscilografias em sistemas elétricos de potência. **SBA Controle & Automação**, São José dos Campos, v. 21, n. 4, p. 347-362, Jul./Ago. 2010.

MUSARUDDIN, Mustarum; ZIVANOVIC, Rastko. **Signal Segmentation of Fault Records based on Empirical Mode Decomposition**. TENCON. [S.l.]: IEEE. 2011. p. 138-143.

SMITH, Stephen W. **The Scientist and Engineer's Guide to Digital Signal Processing**. California, USA: California Technical Publishing, 1997.

STYVAKTAKIS, Emmanouil; BOLLEN, Math H. J.; GU, Irene Y. H. Expert system for Voltage Dip Classification and Analysis. **IEEE**, Gothenburg, 2001. 671-676.

_____ ; BOLLEN, Math H. J.; GU, Irene Y. H. **Automatic Classification of Power System Events using rms Voltage Measurements**. 2002. 824-829p. IEEE Power Engineering Society Summer Meeting, USA, 2002.

UKIL, Abisek; ZIVANOVIC, Rastko. Automatic Signal Segmentation based on Abrupt Change Detection for Power System Applications. IEEE, 2006.

APÊNDICE A

A seguir é listado o código do programa *índice_3.m*

```
% INDICE DE DETECÇÃO 3 - DESVIO PADRÃO E MULTIPLOS SEGMENTOS
%
% Testes de uma rotina para detecção e identificação de eventos
% em registros fasoriais.
%
% Índice de detecção 3 criado por Vinicius Gallert dos Santos
%
% Base do programa criada por Miguel Moreto (id2)

clc;
clear all;

% Configurações:
canal_rdp = 21; % Canal a ser lido. Para ver todos os canais registrados
pelo
% RDP, consultar o arquivo Comtrade.inf no diretório da oscilo.
intervalo = [3100 4100];
limiar = 0.0358; % Limiar do índice de detecção de eventos.
DeltaT = 30; % Número de amostras do indice de deteccao.

% Leitura de um registro fasorial:
% Informar o caminho onde está o arquivo COMTRADE.
[dados cfg] =
ler_fasorial('E:\Oscilografias\2009_02_16_17_49_44L',canal_rdp);
%,intervalo);

%Obtém as configurações junto ao arquivo ler_fasorial.m
canal = find(cfg.analogico_numero == canal_rdp);

usina = cfg.nome(1:4);

disp(sprintf('%s, Canal: %d,
%s',usina,canal_rdp, cfg.analogico_identificacao(canal,:)));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% O trecho a seguir representa a padronização das bases utilizadas
% bem como os cálculos envolvidos.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if usina == 'UHPF'
    % Bases do sistema (UHPF):
    VbaseL = 13800; % Tensão de base de linha em V.
    Sbase3 = 113; % Potência aparente de base trifásica em MVA.
    VbaseTR = 230000; % Tensão do lado de alta do trafo.
elseif usina == 'UHMA'
    % Bases do sistema (UHMA):
    VbaseL = 16000; % Tensão de base de linha em V.
```

```

        Sbase3 = 418.5; % Potência aparente de base trifásica em MVA.
        VbaseTR = 525000; % Tensão do lado de alta do trafo.
elseif usina == 'UTCH'
    % Bases do sistema (UHMA):
    VbaseL = 6000; % Tensão de base de linha em V.
    Sbase3 = 24; % Potência aparente de base trifásica em MVA.
    VbaseTR = 66000; % Tensão do lado de alta do trafo.
else
    disp('Usina não identificada. Serão utilizadas as bases da UHMA.');
```

VbaseL = 16000;

Sbase3 = 418.5;

VbaseTR = 525000;

```
end

T = 1/cfg.frequencia; % Periodo de amostragem.

%base = 1;
VbaseF = VbaseL/sqrt(3); % Tensão de base de fase.
Sbase1 = Sbase3*1e6/3; % Potência de base monofásica.
Ibase1 = Sbase1/VbaseF;
IbaseTR = Sbase1/(VbaseTR/sqrt(3)); % Base de corrente do lado de alta do
trafo.

N = size(dados,1);

% Verifica se é tensão ou corrente:
unidade = cfg.analogico_unidade(canal,:);
if unidade == 'A '
    base = Ibase1;
    grandeza = 'Corrente [pu]';
elseif unidade == 'kV'
    base = VbaseF/1000; % A medida já é em kilovolt.
    grandeza = 'Tensão [pu]';
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Fim dos cálculos de bases e identificação de grandezas
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%                               BLOCO A
%
%   Nesta parte o código utilizado implementa o
%   desvio padrão para leitura e segmentação
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Variáveis criadas para execução da rotina para geração dos segmentos
k = 1;
m = 1;
t = 1;
f = 1;
while (k * DeltaT <= N) %Fica no laço até que acabem as amostras
    m = k * DeltaT; %Amostra final da janela analisada

    if (k == 1) %Se k for 1
        n = 1; %A primeira amostra começa em 1
    else

```

```

        n = (k-1)*DeltaT;    %Caso k for maior, n assume o valor final da
janela anterior (m - 1)
    end

    t2(k,1) = dados(n,2); % Vetor de tempo.

    % Números das amostras do sinal:
    n_amostras(k,1) = n;    %dados(n,1); % Vetor de amostras.

    v = 0;
    soma = 0;
    for l = n:m            %Laço que obtem a média na janela de amostragem

        v(l,1) = (abs(dados(l,3)))/base; %Encontra o módulo do fasor das
amostras no intervalo da janela

    end

    for l = n:m            %Efetua a soma de todos os módulos dos fasores das
amostras na janela

        soma = soma + v(l,1);

    end

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % soma é utilizado para obter a soma total dentro da janela
    % e media obtem o valor da média das amostras do sinal
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    soma;
    media = soma/30;

    somadev = 0;    %Variavel que representa a soma dentro do quadrado do
desvio padrao

    % O laço seguinte calcula o somatório dos elementos que irão compor
    % o indice de segmentação n.
    for l = n:m

        dif(l,1) = (v(l,1) - media)^2; %Representa a diferença dentro da
fórmula do desvio padrão

        somadev = somadev + dif(l,1);    %É o somatório que irá gerar o
indice n

    end

    %A função abaixo obtém a variância e obtém o valor final do índice n
    %É a divisão pelo número de amostras da janela menos 1 (m - 1) e
    %a obtenção da raiz quadrada.

    somadev;

```

```

    di3 = somadev/29;
    id3(k,1) = sqrt(di3);

    k = k + 1; %k controla a quantidade de iterações para não extrapolar o
valor de amostras total

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Aqui termina a função que implementa o desvio padrão
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Aqui começa a configuração das escalas e parâmetros do gráfico das
% oscilografias.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure(1);
set(1, 'Position', [100 300 900 400]);
h = axes;
plot(dados(:,2),abs(dados(:,3))/base);
hold on;

    % Ajuste das escalas do gráfico:
altura = get(h, 'YLim');
limite = 1;
if altura(2)<0.3
    set(h, 'YLim', [0 0.2]);
elseif altura(2)>limite
    set(h, 'YLim', altura);
else
    set(h, 'YLim', [0 limite]);
end

% Plota grafico do sinal:
xlabel('t [s]'); ylabel(grandeza);
grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Aqui termina a parte de parametrização do gráfico
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Encontra índices onde di3 é maior do que o limiar informado:
indices_di = find(id3 >limiar)

% Uma outra variável recebe a quantidade de índices onde ocorreu a falha
c = size(indices_di,1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% BLOCO B
% A função a seguir é utilizada para segmentar o sinal. Caso a distância

```

```

% entre os segmentos seja de mais de 1 indice de id3, buscasse o proximo
% segmento, tendo assim, multiplos segmentos.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%A função isempty retorna 0 caso o vetor indices_di possua qualquer valor
%armazenado. Sendo isempty = 0, o laço if é verdadeiro e entra na função.

    if (isempty(indices_di) == 0)
        for i = 1:c

            if (i == c) %Ultima amostra

                line([t2(indices_di(i))
t2(indices_di(i))],altura,'color','r','LineStyle','-');

            end

            if (i == 1) %Primeira amostra

                line([t2(indices_di(i))
t2(indices_di(i))],altura,'color','r','LineStyle','-');

            end

            if (i ~= c) %Amostras intermediarias

                a = indices_di(i);
                b = indices_di(i+1);

                h = b - a;

                if (h > 1)

                    line([t2(a) t2(a)],altura,'color','r','LineStyle','-');

                    line([t2(b) t2(b)],altura,'color','r','LineStyle','-');

                end

            end

        end

    end

    %Caso não ocorra nenhum indice onde o limiar é extrapolado, o programa
    %retorna que não houve nenhum evento.

    else
        disp('Nenhum evento detectado.')
    end
end

```