Raphael Philipe Mendes da Silva

**Algoritmos Genéticos Aplicados à Estimação Fasorial em
Sistemas Elétricos de Potência**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Mestre em Ciências pelo Programa de Engenharia Elétrica

Área de Concentração: Sistemas Elétricos de Potência

Orientador: Prof. Tit. Denis Vinicius Coury

São Carlos
2012

Ficha catalográfica preparada pela Seção de Tratamento
da Informação do Serviço de Biblioteca – EESC/USP

S586a Silva, Raphael Philipe Mendes da
Algoritmos genéticos aplicados à estimação fasorial
em sistemas elétricos de potência. / Raphael Philipe
Mendes da Silva ; orientador Denis Vinicius Coury. --
São Carlos, 2012.

Dissertação (Mestrado - Programa de Pós-Graduação
em Engenharia Elétrica e Área de Concentração em
Sistemas Elétricos de Potência)-- Escola de Engenharia
de São Carlos da Universidade de São Paulo, 2012.

1. Estimação fasorial. 2. Unidades de medição
fasorial. 3. Algoritmos genéticos. I. Título.

FOLHA DE JULGAMENTO

Candidato: Engenheiro **RAPHAEL PHILIPPE MENDES DA SILVA**.

Título da dissertação: "Algoritmos genéticos aplicados à estimação fasorial em sistemas elétricos de potência".

Data da defesa: 27/08/2012

Comissão Julgadora:

Resultado:

Prof. Titular **Denis Vinicius Coury (Orientador)**
(Escola de Engenharia de São Carlos/EESC)

APROVADO

Dr. **Silvio Aparecido de Souza**
(Companhia de Transmissão de Energia Elétrica Paulista/CTEEP)

APROVADO

Prof. Dr. **Antonio Carlos Siqueira de Lima**
(Universidade Federal do Rio de Janeiro/UFRJ)

APROVADO

Coordenador do Programa de Pós-Graduação em Engenharia Elétrica e Presidente da Comissão de Pós-Graduação:

Prof. Titular **Denis Vinicius Coury**

Dedicatória

Dedico este trabalho aos meus pais, Regina e Milton.

Agradecimentos

Aos meus pais, Regina e Milton, pelo amor, carinho, incentivo e suporte no desenvolvimento não só deste trabalho, mas de todas as outras conquistas de minha vida. Aos meus irmãos Paula e Luís Otávio, pelo apoio, incentivo e amor fraternal.

À minha namorada Carla, pelo apoio, incentivo e todo carinho. Por dividir comigo todos os bons e maus momentos, se mostrando uma ótima companheira.

Ao meu cunhado, professor Fábio, pelas caronas concedidas e por todas conversas a respeito da pós-graduação.

Ao professor Denis Vinicius Coury, pela orientação, confiança e atenção no desenvolvimento deste trabalho bem como pelas contribuições para meu crescimento intelectual e pessoal.

Ao professor Alexandre Cláudio Botazzo Delbem e ao Tiago Vieira da Silva, pela participação no desenvolvimento deste trabalho.

Aos professores Mário Oleskovicz e José Carlos de Melo Vieira Júnior pelos conselhos e pela atenção durante o período em que estive na universidade.

Aos meus avós, Marta e Mauro (Padrito), especialmente ao meu avô, que ao me ensinar a acender uma lampada utilizando pilhas e pedaços de fio quando pequeno, talvez tenha despertado em mim, já naquela época, a paixão pela engenharia.

Aos meus tios, em especial Adriani e Mauro, de quem sempre obtive incentivo para continuar estudando.

Aos amigos do laboratório LSEE, Monaro, Ulisses, Hermes, Daniel, Janison, Marcelo, Etienne dentre outros, com quem pude compartilhar conhecimentos e experiências de forma a crescer profissionalmente e pessoalmente.

Aos meus amigos da graduação pelas alegrias e conquistas compartilhadas.

Aos funcionários do Departamento de Engenharia Elétrica (SEL), da Escola de Engenharia de São Carlos (EESC), Universidade de São Paulo (USP), que de alguma forma contribuíram para o desenvolvimento do meu trabalho.

À CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pela concessão da bolsa de mestrado e para o financiamento do desenvolvimento deste trabalho.

"O conhecimento é o primeiro passo para o sucesso."

Nikola Tesla

Sumário

Lista de Siglas	xi
Lista de Tabelas	xiii
Lista de Figuras	xv
Resumo	xix
Abstract	xxi
1 Introdução	1
1.1 Estrutura do Trabalho	4
2 Unidades de Medição Fasorial - PMUs	5
2.1 Desenvolvimento Histórico	5
2.2 Funcionamento	6
2.3 Aplicações de PMUs	9
2.4 A Norma C37.118-2005	10
2.4.1 Precisão de Medidas	11
2.5 Estado da Arte em Estimação Fasorial	12
3 Algoritmos Aplicados à Estimação Fasorial - Fundamentos Teóricos	15
3.1 Os Algoritmos Genéticos	15
3.1.1 Codificação	16
3.1.2 Seleção	17
3.1.3 Cruzamento	17
3.1.4 Mutação	18
3.2 A Transformada Discreta de Fourier	18
3.2.1 Resumo para aplicação prática	24
3.3 <i>Phase-Locked Loop</i>	25

4	Dados Utilizados para Comparação dos Algoritmos Propostos	31
4.1	Dados Sintéticos	31
4.1.1	Teste de degrau de magnitude (10%)	32
4.1.2	Teste de degrau de fase (90°)	32
4.1.3	Teste de degrau de frequência (5 Hz)	33
4.2	Sistema Elétrico Proposto	34
4.2.1	Dinâmica dos Sistemas Elétricos de Potência	36
4.2.2	Reguladores de Velocidade	37
4.2.3	Reguladores de Tensão	39
4.2.4	O <i>software Alternative Transients Program</i> (ATP)	39
4.2.5	Situações Simuladas no <i>software ATP</i>	40
5	Arquiteturas Desenvolvidas	43
5.1	Implementação Computacional	43
5.2	Implementação dos AGs em <i>Hardware</i>	44
5.2.1	FPGAs	44
5.2.2	Considerações para a Implementação de AGs em FPGA	45
5.2.3	Arquitetura proposta para teste	49
6	Algoritmos Aplicados à Estimação Fasorial - Resultados	53
6.1	Os Parâmetros Utilizados	53
6.1.1	O Algoritmo Genético	54
6.1.2	O Algoritmo Genético compacto	54
6.1.3	A Transformada Discreta de Fourier	55
6.1.4	<i>Phasor-Locked Loop</i>	55
6.2	Resultados Utilizando Dados Sintéticos como Entrada	55
6.2.1	Influência do Ruído e da Taxa de Amostragem	56
6.2.2	Teste de Degrau de Amplitude	57
6.2.3	Teste de Degrau de Fase	64
6.2.4	Teste de Degrau de Frequência	70
6.2.5	Tempo de recuperação	75
6.3	Resultados Utilizando dados Simulados no <i>Software ATP</i>	77
6.3.1	Ilhamento	77
6.3.2	Entrada de Carga	79
6.3.3	Situação de Falta Elétrica Sustentada na Linha de Transmissão	81
6.4	Resultados utilizando FPGAs para implementação do AGc	83

6.4.1	Resultados para dados Sintéticos	83
6.4.2	Resultados para dados simulados no <i>software</i> ATP	87
7	Conclusões	93
7.1	Sugestões para continuidade na pesquisa	95
7.2	Publicações Decorrentes Desta Dissertação	96
7.2.1	Publicação em Congresso Internacional	96
7.2.2	Publicações em Periódicos	96
	Referências Bibliográficas	97
	Apêndice A Algoritmos Genéticos	101
A.1	Introdução	101
A.2	Os Algoritmos Genéticos	102
A.2.1	O Indivíduo	104
A.2.2	A População	104
A.2.3	A Codificação	105
A.2.4	A Avaliação da Aptidão	106
A.2.5	O Processo de Seleção	106
A.2.6	Operadores Genéticos	108
A.2.7	Elitismo	111
A.2.8	Parâmetros Genéticos	111
A.3	O Algoritmo Genético Compacto	112
	Referências Bibliográficas	113
	Apêndice B Arquivos ATP	115

Lista de Siglas

AG Algoritmo Genético

AGc Algoritmo Genético compacto

ASIC *Application Specific Integrated Circuit*

ATP *Alternative Transients Program*

CA Corrente Alternada

CE Computação Evolutiva

CIGRE *Conseil International des Grands Réseaux Electriques*

CORDIC *Coordinate Rotation Digital Computer*

EE energia elétrica

FPGA *field-programmable gate array*

GPS *Global Positioning System*

IEEE *Institute of Electrical and Electronics Engineers*

LSE *Least square error*

MSE *Mean Square Error*

PLL *Phase-Looocked Loop*

PMU *Phasor Measurement Unit*

RTDS *Real-time Digital Simulator*

RLS *Recursive Least Squares*

RMS *Root Mean Square*

SEP *Sistema Elétrico de Potência*

SNR *signal noise ratio*

STFT *Short-time Fourier Transform*

TDF *Transformada Discreta de Fourier*

TFTD *Transformada de Fourier de Tempo Discreto*

TVE *Total Vector Error*

VHDL *Very high speed integrated circuit Hardware Description Language*

Lista de Tabelas

4.1	Parâmetros utilizados no reguladores de velocidade das turbinas hidráulicas . . .	38
4.2	Parâmetros utilizados no regulador de velocidade da turbina a vapor	38
6.1	Parâmetros utilizados no Algoritmo Genético	54
6.2	Parâmetros utilizados Algoritmo Genético Compacto	55
6.3	Parâmetros utilizados no algoritmo baseado em <i>Phase-Locked Loop</i>	55
6.4	Tempo de recuperação dos algoritmos implementados computacionalmente . . .	76
6.5	Tempo de recuperação do AGc implementado em FPGA	87
A.1	Mapeamento por gene em um intervalo real	106
A.2	Aptidão de indivíduos em uma população	107

Lista de Figuras

2.1	Amostragem sincronizada para a medição de fasores em uma referência comum. .	7
2.2	Diagrama de blocos da arquitetura de <i>hardware</i> de um PMU.	8
2.3	Exemplo de sistema elétrico monitorado por PMUs.	9
3.1	Processo de ajuste de curva utilizado pelo AG para estimação fasorial.	17
3.2	Codificação Real.	17
3.3	Codificação Binária.	17
3.4	Processo de Cruzamento	18
3.5	Efeito da amostragem assíncrona na medida de $\bar{x}$	21
3.6	Estimação fasorial utilizando transformada discreta de Fourier	23
3.7	Diagrama de blocos do sistema <i>Phase-Locked Loop</i> (PLL) estudado.	28
4.1	Forma de onda do teste de degrau de Magnitude	33
4.2	Forma de onda do teste de Degrau de Fase	33
4.3	Forma de onda do teste de Degrau de Frequência	34
4.4	Sistema Elétrico proposto	35
4.5	Dinâmica dos Sistemas Elétricos de Potência	36
4.6	Sistema de Regulação de Velocidade de Turbina Hidráulica	38
4.7	Sistema de Regulação de Velocidade de Turbina Térmica com Reaquecimento . .	38
4.8	Regulador de Tensão	39
4.9	Falta Franca na linha de transmissão 1 a 75 Km da linha	41

4.10 Situação de Ilhamento	42
5.1 Arquitetura computacional implementada para testes	44
5.2 Estrutura de um FPGA	45
5.3 <i>Linear Feedback Shift Register</i>	46
5.4 Máquina de estados utilizada para modelar o AGc	49
5.5 Kit de Desenvolvimento Stratix III utilizado	50
5.6 Arquitetura implementada para testes	50
5.7 Comunicação com o kit de desenvolvimento	51
6.1 Precisão dos algoritmos para a taxa amostral 32 amostras por ciclo.	56
6.2 Precisão dos algoritmos para taxa amostral de 128 amostras por ciclo.	57
6.3 TVE para o caso de teste de degrau de amplitude com SNR 40.	58
6.4 Estimação de frequência para o caso de teste de degrau de amplitude com SNR 40.	58
6.5 Estimação de magnitude para o caso de teste de degrau de amplitude com SNR 40.	59
6.6 TVE para o caso de teste de degrau de amplitude com SNR 60.	60
6.7 Estimação de frequência para o caso de teste de degrau de amplitude com SNR 60.	60
6.8 Estimação de magnitude para o caso de teste de degrau de amplitude com SNR 60.	61
6.9 TVE para o caso de teste de degrau de amplitude com SNR 80.	62
6.10 Estimação de frequência para o caso de teste de degrau de amplitude com SNR 80.	62
6.11 Estimação de magnitude para o caso de teste de degrau de amplitude com SNR 80.	63
6.12 TVE para o caso de teste de degrau de fase com SNR 40.	64
6.13 Estimação de frequência para o caso de teste de degrau de fase com SNR 40.	65
6.14 Estimação de magnitude para o caso de teste de degrau de fase com SNR 40.	65
6.15 TVE para o caso de teste de degrau de fase com SNR 60.	66
6.16 Estimação de frequência para o caso de teste de degrau de fase com SNR 60.	67
6.17 Estimação de magnitude para o caso de teste de degrau de fase com SNR 60.	67

6.18	TVE para o caso de teste de degrau de fase com SNR 80.	68
6.19	Estimação de frequência para o caso de teste de degrau de fase com SNR 80. . .	68
6.20	Estimação de magnitude para o caso de teste de degrau de fase com SNR 80. . .	69
6.21	TVE para o caso de teste de degrau de frequência com SNR 40.	70
6.22	Estimação de frequência para o caso de teste de degrau de frequência com SNR 40.	71
6.23	Estimação de magnitude para o caso de teste de degrau de frequência com SNR 40.	71
6.24	TVE para o caso de teste de degrau de frequência com SNR 60.	72
6.25	Estimação de frequência para o caso de teste de degrau de frequência com SNR 60.	73
6.26	Estimação de magnitude para o caso de teste de degrau de frequência com SNR 60.	73
6.27	TVE para o caso de teste de degrau de frequência com SNR 80.	74
6.28	Estimação de frequência para o caso de teste de degrau de frequência com SNR 80.	74
6.29	Estimação de magnitude para o caso de teste de degrau de frequência com SNR 80.	75
6.30	Frequência estimada pelos algoritmos no subsistema 1.	77
6.31	Amplitude estimada pelos algoritmos no subsistema 1.	78
6.32	Frequência estimada pelos algoritmos no subsistema 3.	78
6.33	Amplitude estimada pelos algoritmos no subsistema 3.	79
6.34	Frequência estimada pelos algoritmos durante a situação de entrada de carga. . .	80
6.35	Amplitude estimada pelos algoritmos durante a situação de entrada de carga. . .	80
6.36	Frequência estimada pelos algoritmos no durante a situação de falta.	81
6.37	Amplitude estimada pelos algoritmos no durante a situação de falta.	82
6.38	TVE para o caso de teste de degrau de magnitude com SNR 60.	83
6.39	Estimação de frequência para o caso de teste de degrau de magnitude com SNR 60.	84
6.40	Estimação de Magnitude para o caso de teste de degrau magnitude com SNR 60.	84
6.41	TVE para o caso de teste de degrau de fase com SNR 60.	84
6.42	Estimação de frequência para o caso de teste de degrau de fase com SNR 60. . .	85
6.43	Estimação de magnitude para o caso de teste de degrau de fase com SNR 60. . .	85

6.44	TVE para o caso de teste de degrau de frequência com SNR 60.	86
6.45	Estimação de frequência para o caso de teste de degrau de frequência com SNR 60.	86
6.46	Estimação de magnitude para o caso de teste de degrau de frequência com SNR 60.	86
6.47	Frequência estimada no subsistema 1 pelo AGc implementado em FPGA.	88
6.48	Amplitude estimada no subsistema 1 pelo AGc implementado em FPGA.	88
6.49	Frequência estimada no subsistema 3 pelo AGc implementado em FPGA.	88
6.50	Amplitude estimada no subsistema 3 pelo AGc implementado em FPGA.	89
6.51	Frequência estimada durante a entrada de carga pelo AGc implementado em FPGA.	89
6.52	Amplitude estimada durante a entrada de carga pelo AGc implementado em FPGA.	90
6.53	Frequência estimada durante a situação de falta pelo AGc implementado em FPGA.	90
6.54	Amplitude estimada durante a situação de falta pelo AGc implementado em FPGA.	91
A.1	Fluxograma de um algoritmo genético clássico.	103
A.2	Representação de um indivíduo.	104
A.3	Seleção pelo método da roleta.	108
A.4	Seleção pelo método de torneio	108
A.5	Cruzamento de Um-Ponto	110
A.6	Cruzamento Multipontos	110
A.7	Processo de mutação.	111
A.8	Fluxograma de um AGc	113
A.9	Modelo de probabilidade para a população de um AGc	113

Resumo

Silva, R. P. M. **Algoritmos Genéticos Aplicados à Estimação Fasorial em Sistemas Elétricos de Potência.** 2012. 126 f. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2012.

Esta trabalho apresenta a análise e implementação de uma técnica inteligente, o algoritmo genético (AG), para implementação de unidades de medição fasoriais, denominadas PMUs (*Phasor Measurement Units*). A disponibilidade dos fasores em diversos pontos de um sistema elétrico de potência (SEP) é importante, tanto para monitoramento quanto para controle, proteção e estudo do sistema. Entretanto, a obtenção de tais fasores só têm sentido se os mesmos possuírem o mesmo referencial no tempo. Este referencial é conseguido através de sinais de satélites GPS (Global Positioning System) que sincronizam as PMUs instaladas nos pontos de interesse. Existe uma vasta quantidade de métodos que podem ser utilizados para que, de posse das formas de onda discretizadas de tensão e corrente, estime-se os fasores correspondentes e as frequências locais. Este projeto apresenta os AGs como ferramenta de estimação para a obtenção de uma PMU com todas as vantagens relativas a tais algoritmos. Além disso, uma versão do AG que utiliza menos recursos computacionais, o algoritmo genético compacto (AGc) também será estudado. Um estudo norteado pela norma internacional C37.118 compara o desempenho dos AGs com dois métodos tradicionais de medição fasorial, um baseado na transformada discreta de Fourier e outro baseado em um filtro *phase-locked loop*. Dados sintéticos e provenientes de simulações são utilizados para avaliar o desempenho dos algoritmos desenvolvidos. Para tirar vantagem da natureza paralela dos algoritmos genéticos, um estudo da implementação do AGc em FPGA (*field programmable gate array*) utilizando a linguagem VHDL é realizado a fim de estudar a implementação embarcada em PMUs.

Palavras-chave: estimação fasorial, unidades de medição fasorial, algoritmos genéticos

Abstract

Silva, R. P. M. **Genetic Algorithms Applied to Power Systems Phasor Measurement.** 2012. 126 f. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2012.

This work presents the implementation and analysis of an intelligent technique, the genetic algorithm (GA), for the implementation of phasor measurement units (PMUs). The estimation of phasors in several spots in an electrical power system is important for the monitoring, control, protection and study of this system. However, these phasors must be in a common time reference in order to be usefull. This reference is achieved by using signals provided by the Global Positioning System (GPS) that synchronize the PMUs installed in the system. There are several techniques that can be used to estimate the phasors and local frequency using current and voltage wave signals. This project introduces the GAs as a phasor estimation tool applied to PMUs. Besides that, a version of the GA that demands less computational resources, the compact Genetic Algorithm is studied and implemented. A detailed study is performed using the international standard C37.118 as a guide comparing the GAs with two traditional techniques. The two traditional techniques are based on the DFT (Discrete Fourier transform) and a phase-locked loop filter (PLL). Synthetic and simulated data is used to evaluate the performance of the implemented algorithms. In order to take advantage of the parallel behavior of the genetic algorithms, a study of its implementation in FPGA (field programmable gate array) using the VHDL language is performed to make the genetic algorithms useful in real PMUs.

Keywords: phasor measurement, phasor measurement unit, genetic algorithms

Capítulo 1

Introdução

O homem atualmente necessita de grande quantidade de energia de diferentes origens para a sua sobrevivência e seu bem-estar. Nos últimos séculos presenciou-se um desenvolvimento tecnológico acelerado pela utilização do petróleo e da energia elétrica (EE) como fontes de energia em larga escala. Em 1885 nos Estados Unidos, foi instalado o primeiro sistema de distribuição baseado em Corrente Alternada (CA) que alimentava 150 lâmpadas (Stevenson, 1986). Mais tarde, em 1890, a primeira linha de transmissão em CA dos Estados Unidos foi colocada em operação para transportar EE por uma distância de 20 km. Estes eventos podem ser descritos como marcos iniciais dos Sistemas Elétricos de Potência (SEPs) como são conhecidos hoje. Notoriamente a EE, devido a sua eficiência de transporte e versatilidade na transformação para outros tipos de energia, é de suma importância para o homem atualmente.

Dessa forma, o contínuo crescimento da população mundial, da economia de países desenvolvidos e em desenvolvimento alavancados pelos avanços tecnológicos atuais tem impulsionado uma demanda acelerada por EE. Tal aumento ocasionou a expansão progressiva dos SEPs desde sua data de concepção até os dias atuais. A fim de evidenciar esse crescimento, toma-se como exemplo o Brasil, onde o consumo total de energia elétrica passou de 18.346 GWh em 1960 (Schmidt e Lima, 2004) para 430.106 GWh em 2011 (Energética, 2012), o que representa uma taxa de crescimento média anual no período de 8,2% .

Levando-se em conta o crescimento no consumo de energia elétrica, espera-se que os SEPs sejam operados cada vez mais perto de sua capacidade máxima, de forma a aproveitar eficientemente os equipamentos instalados e fontes de energia disponíveis. A operação perto do limite exige a habilidade de controle em tempo-real do sistema, a fim de se evitar *blackouts* de larga

escala em cascata ou danos aos equipamentos (Adamiak, Premerlani e Kasztenny, 1993). Aliado a isso, existe a necessidade da melhoria contínua da qualidade da EE e serviços relacionados, de forma a satisfazer os consumidores e seus processos de produção cada vez mais sensíveis a distúrbios relacionados com a qualidade da EE (Santoso et al., 2002). Dessa forma, é imperativo a existência de um sistema de monitoramento preciso que possibilite investigar a operação dinâmica dos SEPs, e determinar seu estado de operação.

Neste contexto, fasores são ferramentas básicas de análise de circuitos de CA, utilizadas como meio de representação da forma de onda senoidal fundamental em regime permanente. Mesmo quando o SEP não está em regime permanente, fasores são úteis para a descrição do seu comportamento (Phadke, 1993). Por exemplo, quando o SEP está sob influência de oscilações eletromecânicas, as formas de onda não estão em regime permanente, nem a frequência do sistema está em valor nominal. Sob essas condições, como as variações de tensão e corrente são relativamente lentas, fasores ainda podem ser utilizados para analisar a rede, com as variações sendo tratadas como um conjunto de estados de regime permanente.

Para uma análise correta, quando várias formas de onda de tensão e corrente são medidas e convertidas para fasores em um SEP, estes devem ser amostradas em um mesmo instante de tempo, isto é, devem estar em uma referência comum. Isto é facilmente alcançado em uma subestação, onde uma fonte sincronizadora comum entre os equipamentos pode ser utilizada. Entretanto a tarefa de extrair fasores com referência comum entre subestações separadas por grandes distâncias não é uma tarefa trivial.

Ao longo dos anos, reconhecendo a importância da medição dos fasores e da diferença do ângulo de fase entre pontos remotos do sistema, muitas tentativas foram feitas a fim de se atingir a sincronização. Muitos sistemas de comunicação como linhas dedicadas, micro-ondas, ou ondas de rádio AM tem um desempenho pobre quanto a qualidade da sincronização. Fibras óticas, quando disponíveis, podem ser utilizadas para prover um sinal de sincronização de alta precisão, se houver uma fibra dedicada a esse propósito. No entanto, o seu alto custo impossibilita essa utilização. A fim de se baratear o sistema, se um canal de fibra multiplexado é utilizado, erros de sincronização da ordem de 100 microsegundos são atingidos. No entanto, erros dessa ordem de grandeza não são aceitáveis para a medição fasorial sincronizada em SEPs (Phadke, 1993).

A solução veio com a utilização do *Navstar Global Positioning System* (GPS), uma constelação de 24 satélites orbitando a Terra 2 vezes ao dia em 6 planos orbitais a uma altitude aproximada de 200.200 km (Hart et al., 2001). Concebido primariamente para propósitos nave-

gacionais, este sistema também fornece um sinal pulsante que tem precisão de 1 microsegundo em qualquer localização na Terra, ocasionando um erro de $0,021^\circ$ em um sistema de 60Hz, o que é suficiente para qualquer aplicação atual. Apenas um satélite precisa estar visível pela antena de um equipamento que utilize este sinal. Esta antena é pequena e pode ser facilmente montada no topo de instalações em uma subestação, revelando assim as vantagens na utilização desta ferramenta como forma de sincronização.

Fazendo uso do GPS e utilizando os avanços disponibilizados pelas primeiras gerações de relés computadorizados, surge em 1988 na *Virginia Polytechnic Institute and State University* o primeiro protótipo de uma ferramenta utilizada especificamente para extrair fasores sincronizados: o *Phasor Measurement Unit* (PMU), ou unidade de medição fasorial (Phadke, 2002). A aplicação mais natural de uma PMU é a monitoração de um SEP pela determinação de seu estado através da medição em diversos pontos de interesse (Phadke, 1993). Esta abordagem pode ainda ser utilizada para validar modelos computacionais que são simulados. Existem outras muitas aplicações que podem ser exploradas pela utilização de PMUs (Burnett Jr et al., 1994; Hart et al., 2001; Phadke, 1993).

Para que ocorresse a padronização das PMUs e definição de seus requisitos, foi lançado pelo *Institute of Electrical and Electronics Engineers* (IEEE) em 1995 a norma 1344-1995 (IEEE, 1995), e mais tarde revisada e lançada na forma da norma C37.118-2005 (IEEE, 2005). A norma atualizada C37.118 define convenções para a realização de medições fasoriais, disponibiliza um método para determinar a precisão destas medições, além de requisitos de performance em situações de regime permanente. Ela também define o formato dos dados para comunicação de tempo real, de forma que PMUs de diferentes fabricantes possam ser compatíveis entre si. No entanto, o padrão não especifica o método para efetuar medidas, bem como outros fatores como taxa amostral, algoritmos e método de sincronização (Martin et al., 2008).

Na literatura podem ser encontrados inúmeros algoritmos para a estimação fasorial em SEPs. Dentre os algoritmos disponíveis para esta finalidade, pode-se citar os que fazem uso da Transformada Discreta de Fourier (TDF) (Wang e Sun, 2004, 2006) e os que utilizam o PLL para a extração dos fasores das formas de onda. Além destes, pode-se utilizar Algoritmos Genéticos (AGs) num processo de *curve fitting*, isto é, um ajuste de curvas através de uma fórmula matemática a fim de se extrair parâmetros que compõem os fasores (Carvalho et al., 2009).

Algoritmos genéticos são meta-heurísticas de busca e pesquisa utilizados em problemas de otimização, tendo como vantagem não haver a necessidade de conhecimentos sobre o domínio

do problema abordado para que seja feito o seu uso. O uso de AGs tem se tornado popular devido a sua flexibilidade e eficiência sendo assim, matéria de pesquisas e desenvolvimento de trabalhos. No entanto, a utilização destes algoritmos como forma de estimação fasorial não foi totalmente explorada, podendo assim ser alvo de pesquisas.

Desta forma, este trabalho apresenta o estudo e aplicação de AGs como ferramenta para a estimação fasorial em SEPs. Primeiramente, o histórico e funcionamento das PMUs, bem como os requisitos definidos pela norma c37.118-2005 são apresentados. Os fundamentos que definem os AGs são abordados e dois tipos distintos destes algoritmos são apresentados e implementados computacionalmente. Além disso, algoritmos descritos na literatura que utilizam TDF e PLL também são implementados a fim de se criar um comparativo entre as diferentes técnicas. O comparativo entre os métodos implementados é norteado pelos requisitos definidos na norma c37.118-2005 e outros artigos relacionados. Um SEP para estudo é proposto e implementado em simulação computacional com a utilização do *software* ATP, utilizando modelos completos de geradores e reguladores de velocidade e de tensão, de forma a se observar a dinâmica do sistema. Por fim, o AGc é implementado com a utilização de um FPGA, de forma a se tirar proveito do paralelismo provido por estes dispositivos.

1.1 Estrutura do Trabalho

O presente trabalho é composto por 7 capítulos. O Capítulo 1 visa introduzir o assunto discutido no trabalho, evidenciando os fatos que levaram ao desenvolvimento de PMUs, introduz a norma C37.118-2005 citando alguns métodos de estimação fasorial e acaba por propor a utilização de algoritmos genéticos para este fim. O desenvolvimento histórico, funcionamento e principais utilizações de PMUs são discutidos no capítulo 2. Neste mesmo Capítulo, também é apresentada uma revisão bibliográfica sobre o estado da arte com relação às pesquisas relacionadas à estimação fasorial. No capítulo 3 é demonstrada a estimação fasorial para os métodos escolhidos para estudo neste trabalho, dentre eles os próprios AGs, a TDF e PLL. A criação de dados para análise e o estudo e simulação de um SEP no *software* ATP são discutidos no capítulo 3. Os métodos desenvolvidos para a elaboração deste trabalho são descritos no capítulo 5, onde são apresentados as formas como os algoritmos foram implementados computacionalmente e em *hardware* dedicado utilizando um FPGA. Os resultados obtidos pela aplicação dos métodos são discutidos no capítulo 6. Por fim, no capítulo 7 são apresentadas as conclusões feitas durante o desenvolvimento do trabalho.

Capítulo 2

Unidades de Medição Fasorial - PMUs

Este capítulo tem por objetivo definir de forma precisa as unidades de medição fasorial - PMUs, descrevendo o histórico de sua criação, seu funcionamento e possíveis aplicações. Ao fim, será apresentada a norma IEEE C37.118, que define requisitos para estes dispositivos devem atender. Tal norma será utilizada neste trabalho para quantificar os algoritmos desenvolvidos com relação a precisão das estimativas fasoriais.

2.1 Desenvolvimento Histórico

A aplicação de relés computacionais como campo de pesquisa foi iniciada na década de 60 (Phadke, 2002). Os computadores disponíveis naquela época não eram rápidos nem acessíveis economicamente o suficiente para a construção completa de um relé de proteção. No entanto, grandes passos foram dados nas décadas de 70 e 80 no desenvolvimento de algoritmos para relés computacionais. Um desses esforços foi o desenvolvimento do Relé de Distância de Componente Simétrica para a proteção de linhas de transmissão de alta tensão. A principal inovação neste relé foi o algoritmo de proteção baseado nas medições da tensão de sequência positiva, negativa e zero além das correntes no terminal da linha de transmissão. A vantagem do novo algoritmo foi que ele requeria o processamento de apenas uma equação para determinar a localização da falta independentemente do tipo desta. Diante da falta de poder computacional disponível na época, isso foi uma grande avanço.

Um dos resultados deste desenvolvimento foi a criação de um algoritmo recursivo para o cálculo das componentes simétricas de tensão e corrente, a Transformada Discreta de Fourier de componente simétrica. Desta forma, tornou-se aparente que uma técnica para a medição de tensão e corrente de sequência positiva com precisão e tempo de resposta satisfatório seria de interesse para outras aplicações, principalmente se as medições pudessem ser sincronizadas ao longo do SEP. Isto levou ao próximo estágio de desenvolvimento: a sincronização das amostras feitas no sinais de tensão e corrente.

A sincronização precisa das medidas se tornou possível com a utilização do sistema de satélites GPS. Este sistema provê um sinal de pulso com precisão de 1 microssegundo possibilitando alcançar um erro de precisão em um sistema de 60 Hz de $0,021^\circ$, precisão adequada para que haja confiabilidade nas medidas.

Estes avanços convergiram e permitiram a criação de uma ferramenta que agrega as características de sincronização das medições: o PMU. Este dispositivo foi desenvolvido pela Virginia Tech sendo financiado por diversas agências que tinham interesses em seu desenvolvimento. Este interesse se deu em decorrência de certos fatos ocorridos, como o blecaute de 1965 no nordeste dos Estados Unidos, evidenciando o desejo por um controle e monitoramento mais preciso do sistema. O primeiro protótipo foi lançado em 1988.

Baseado no protótipo desenvolvido na Virginia Tech, fabricantes começaram a manufaturar PMUs como um produto comercial. Neste passo foram adicionadas inúmeras inovações, como um receptor de GPS interno, um conversor analógico-digital sigma-delta de 16-bits para cada canal de entrada e interfaces para comunicação remota para acesso ao PMU. Além disso, foram criados concentradores de dados, que coletam dados de diversos PMUs, sincronizando-os de acordo com suas referências de tempo, de forma que os fasores pudessem ser analisados posteriormente por um *software*.

Com a instalação destes dispositivos em campo, ficou claro a sua eficiência e aplicações para o controle e monitoramento do sistema elétrico, sendo que hoje encontram-se milhares destes dispositivos instalados pelo mundo.

2.2 Funcionamento

Considerando-se o problema da medição da tensão de sequência positiva em duas subestações separadas por grandes distâncias, se as amostras de dados usadas nas duas subestações forem

sincronizadas precisamente, e o tempo absoluto do processo de amostragem gravado, então estas medições podem ser enviadas para um ponto remoto comum acompanhadas de estampas de tempo. Pelo alinhamento das estampas de tempo destas duas medições obtidas das duas subestações, medições de sequência positiva podem estar disponíveis a cada instante, de forma sincronizada.

Figura 2.1: Amostragem sincronizada para a medição de fasores em uma referência comum.

A figura 2.1 ilustra este princípio graficamente. Os pulsos sincronizados são utilizados para amostrar tensões e correntes nas duas subestações. O tempo preciso é relacionado com as amostras, de forma que seja possível ter medições de dois locais distantes em uma mesma estampa de tempo.

Como foi mencionado, a precisão na sincronização da amostragem das medições foi possível com a utilização de um sistema GPS. Além disso, o PMU é constituído de outros componentes, como descrito na figura 2.2, onde pode-se observar a o diagrama de blocos da arquitetura de *hardware* de um PMU (Hart et al., 2001; Phadke e Thorp, 2006).

O filtro *anti-aliasing* é utilizado para filtrar frequências dos sinais de entrada de tensão e de corrente acima da frequência de Nyquist. O oscilador divide o pulso de período de 1 segundo proveniente do GPS em pulsos com período de duração menor de forma a controlar o conversor analógico digital. Com os sinais digitalizados, o processador irá aplicar um algoritmo sobre estes dados de forma a extrair informações de interesse. Por fim, os dados são disponibilizados por uma interface de comunicação de forma que possam ser acessados remotamente.

Figura 2.2: Diagrama de blocos da arquitetura de *hardware* de um PMU.

O grande benefício provido por um sistema monitorado por PMUs é a possibilidade de se efetuar medições sincronizadas em tempo real. Por exemplo, os componentes de sequência positiva da frequência fundamental podem ser usados por um centro de controle avançado que coleta informações de diversos dispositivos simultaneamente, possibilitando a análise de contingência e fluxo de carga em tempo-real. A figura 2.3 demonstra o esquema de funcionamento de um SEP monitorado por diversos PMUs interligados a um centro de controle. Deve-se ressaltar que o sistema deve prover a infra-estrutura para a comunicação entre os dispositivos (Nuqui, 2001).

Figura 2.3: Exemplo de sistema elétrico monitorado por PMUs.

2.3 Aplicações de PMUs

Como mencionado no capítulo 1, fasores são ferramentas básicas de análise de circuitos de CA, usualmente introduzidos para representação do regime permanente senoidal do sistema elétrico. Porém, mesmo que este não esteja completamente em regime permanente, os fasores são úteis na descrição de seu comportamento (Phadke, 1993). Alguns fenômenos elétricos observados em SEPs são relativos a variações muito lentas das formas de onda de tensão e de corrente. Assim, os fasores podem ser usados assumindo-se que tais variações sejam equivalentes a um conjunto de estados permanentes consecutivos. Além disso, até quando as variações são súbitas, ou quando os sinais possuem componentes transitórias significativas, os fasores fornecem informações importantes para estudo do SEP (Phadke, 1993).

A importância da medição sincronizada de fasores pode ser evidenciada com os quatro próximos exemplos de aplicações (Hart et al., 2001):

- Estimação de Estado: um dos mais importantes elementos de um sistema de gerenciamento

de energia é a estimação de estados de um SEP através de medições em tempo real, permitindo ao operador tomar decisões melhores, mantendo assim o sistema mais seguro quando em situações de contingência;

- **Proteção Avançada de Redes:** da teoria de proteção, sabe-se que a lógica diferencial tornou-se o principal meio de proteção para os grandes transformadores de potência e também para quase todos os sistemas onde a saída de corrente se encontra relativamente próxima à entrada (Coury, Oleskovicz e Giovanini, 2007);
- **Controle Avançado:** dispositivos controladores são projetados para atuar no SEP, de tal forma que otimizem uma função objetiva. Desde que os fenômenos a serem controlados são definidos em função de variáveis de barramentos distantes do sistema, a medição sincronizada oferece a oportunidade de buscar essas medidas “remotas” para o controlador;
- **Validação de Modelos Computacionais:** Um último, mas não menos importante exemplo de aplicação de PMUs citado aqui, consiste na verificação da modelagem dos sistemas de transmissão e geração através da comparação com os dados dos sistemas reais (Burnett Jr et al., 1994).

2.4 A Norma C37.118-2005

O IEEE publicou dois padrões que cobrem medições fasoriais sincronizadas em SEPs. O padrão IEEE (1995) foi publicado em 1995 e mais tarde revisado e publicado na forma do padrão IEEE (2005). O padrão define convenções para as medidas, fornece um método para determinar a precisão das mesmas, e fornece requisitos para performance das medidas em regime permanente. Ele também define formatos para comunicação de dados para transmissão em tempo-real. O documento introduz o conceito de PMU, a construção deste dispositivo, comunicações, e potencial armazenamento de medições. Dispositivos que são denominados PMUs precisam ser intercambiáveis entre si, provendo medidas equivalentes quando alimentados com sinais de entrada iguais em regime permanente, e comunicando estas medidas de uma maneira padronizada (Martin et al., 2008).

Para o desenvolvimento deste trabalho, deve-se chamar a atenção para um conceito importante definido no padrão, a precisão de medidas, que é a forma como os algoritmos implementados são comparados.

2.4.1 Precisão de Medidas

O padrão define precisão como uma diferença fasorial entre o fasor estimado ($\vec{X}_{MEDIDO(k)}$) e o esperado ($\vec{X}_{IDEAL}$) para uma medição em um determinado instante de tempo k . Ele é definido como o erro vetorial total - *Total Vector Error* (TVE)

$$TVE_{(k)} = 100\% \cdot \frac{|\vec{X}_{MEDIDO(k)} - \vec{X}_{IDEAL}|}{|\vec{X}_{IDEAL}|} \quad (2.1)$$

O *TVE* agrega três possíveis fontes de erro: magnitude, ângulo e tempo de sincronização. Por exemplo, se um determinado PMU sincronizado perfeitamente com a referência de tempo e sem erro de ângulo, 1% de erro na magnitude irá se traduzir em 1% de erro no *TVE*. Sem erros de sincronização e magnitude, um erro no ângulo de 0.573 ° corresponde a 1% de *TVE*. Quando a única fonte de erro é sincronização, uma imprecisão de 26,5 μs em 60 Hz causa 1% no *TVE*.

O padrão permite compatibilidade com dois níveis de precisão: Nível 1 e Nível 0. Ambos exigem *TVE* abaixo de 1%, mas o nível 1 é mais exigente, impondo mais condições de testes para referência em termos de variação de frequência, magnitude, distorções no sinal, entre outros.

O padrão não especifica o método para medição, isto é, não especifica a frequência de amostragem, algoritmos de estimação fasorial, ou método de sincronização. Ele somente define que um mesma saída - 1% de *TVE* - deve ser obtida sob uma ampla quantidade de situações de referência. Isto permite que os fabricantes usem diferente métodos de medição e estimação enquanto garante a conformidade. Este é a principal motivação para o desenvolvimento deste trabalho.

Ainda com relação à precisão das estimações fasoriais, a norma faz uma consideração que foi levada em conta no desenvolvimento deste trabalho. Para manter compatibilidade entre os diferentes algoritmos existentes, a norma considera que a referência de tempo deve ser localizada no centro da janela de dados do algoritmo.

A precisão das medidas de frequência (incluindo a variação de frequência) não é especificado no padrão. No entanto, tem-se mostrado que os PMUs estão entre melhores medidores de frequência disponíveis, com precisão de alguns milihertz (tipicamente 1-3 mHz) com a janela de medição de alguns ciclos.

O padrão não especifica os requisitos de performance de PMUs sob condições transitórias. Isto significa que sob condições tais como uma rápida variação de frequência, vários PMUs irão produzir saídas que diferem entre si, mesmo que estejam em conformidade com a norma. No

entanto é necessário saber o tempo que o algoritmo leva para voltar a informar uma resposta com TVE inferior a 1% após o transitório, isto é, quando encontra-se em regime permanente novamente.

2.5 Estado da Arte em Estimação Fasorial

Como a norma C37.118 (IEEE, 2005) não especifica os métodos de estimação fasorial, pode-se utilizar diferentes métodos desde que a conformidade com a precisão seja garantida. Assim, aqui são apresentados os principais métodos para estimação fasorial disponíveis na literatura.

Dentre os vários métodos existentes aplicáveis à estimação fasorial, alguns se destacam, quer por simplicidade ou complexidade, elegância, desempenho, ou outros fatores quaisquer. A seguir, quatro métodos são comentados e algumas avaliações iniciais de cada método são realizadas.

A aplicação da TDF pura e simplesmente em janelas de sinais do SEP pode produzir resultados insatisfatórios devido ao fato da frequência do sistema sofrer pequenas variações na prática. Desvios na frequência fundamental ocasionam oscilações na amplitude e fase no resultado da TDF, assim como desvio do valor médio destes parâmetros. Se o sinal possui componentes harmônicas, que seriam totalmente atenuadas se não houvesse desvio, o resultado é ainda pior, pois estas componentes passam a influenciar a estimativa do fasor da fundamental.

Dada a incapacidade da TDF de lidar com frequência variante no tempo, muitos autores propõem metodologias de correção (ou compensação) de seu resultado. Harris (1978) analisa a utilização de diferentes janelas como a de Hamming, Kaiser, etc. para a redução do efeito *leakage*, ou vazamento espectral e consequente melhoria de resultados. Este método é também conhecido como *Short-time Fourier Transform* (STFT). Phadke, Thorp e Adamiak (1983) introduzem um método de correção de amplitude e fase e estimação de frequência. Entretanto, o erro de fase considerado na aproximação proposta não é completo, o que prejudica a exatidão das estimativas. Hart et al. (1997) desenvolveram uma aproximação que adapta o tamanho de janela para obter melhores estimativas de frequência, mas não corrige a fase. Funaki e Tanaka (2002) propuseram correções precisas para a fase e frequência, assim como Nguyen e Li (2006), sendo que estes últimos desenvolveram um método baseado em interpolação da janela de dados analisados. Finalmente, Wang e Sun (2004, 2006) propuseram um método com a mesma abordagem de Phadke, Thorp e Adamiak (1983), mas com a diferença de considerar o erro total da fase para obtenção das equações de correção. Como resultado um método de alto desempenho é obtido

para estimação do fasor fundamental e da frequência.

Baseados em modelos de otimização, surgem os algoritmos do tipo PLL, como alternativa ao uso da TDF. Estes modelos consideram variantes no tempo os parâmetros (amplitude, frequência e fase) dos sinais de tensão e corrente de um SEP. Desta forma, um conjunto de equações não lineares é obtido através do método do Gradiente Descendente para estimação de tais parâmetros.

Ziarani e Konrad (2004) apresentam uma estrutura simples e eficaz para estimação de parâmetros. O interessante deste método é que a resposta em regime permanente pode ser melhorada diminuindo as constantes que determinam o passo do algoritmo. Esse procedimento torna a resposta mais lenta, fazendo com que em cada aplicação particular seja necessário o ajuste do binômio velocidade/exatidão conforme desejado. Karimi-Ghartemani, Karimi e Iravani (2004) apresentam uma estratégia para diminuir o efeito do ruído nos sinais elétricos na estimativa da amplitude. Com isso, uma nova estrutura PLL é obtida já que além da inserção de um filtro passa-baixa para suavizar a resposta da amplitude, há a inserção de um filtro na estrutura interna do PLL apresentado por Ziarani e Konrad (2004). Também Karimi-Ghartemani e Iravani (2005) apresentam uma estrutura PLL onde dois filtros passa-baixas são inseridos nos laços internos do PLL para aumento da imunidade à presença de ruídos e componentes harmônicas, estas últimas sendo responsáveis principalmente por erros de *offset* nas estimativas da fundamental. Karimi, Karimi-Ghartemani e Iravani (2004); Karimi-Ghartemani e Iravani (2004) apresentam uma nova estrutura PLL baseada na extração das componentes em fase e em quadratura do sinal elétrico analisado. Uma avaliação detalhada é feita sobre o erro na estimativa da componente fundamental, devido à presença de ruído e harmônicos.

Métodos baseados nos mínimos quadrados - *Least square error* (LSE) - são também uma alternativa para medição de parâmetros de um sinal elétrico de um SEP. Sachdev e Baribeau (1979) apresentam um algoritmo onde os coeficientes de um filtro digital são encontrados baseados no LSE e então usados para determinação das partes real e imaginária dos fasores de tensão e corrente. Já Sachdev (1985) apresentam um algoritmo para determinação da frequência baseado também no método dos mínimos quadrados. Kamwa (1991) propuseram dois métodos, um baseado no algoritmo recursivo - *Recursive Least Squares* (RLS) - e outro no erro quadrático médio - *Mean Square Error* (MSE) - para a obtenção das estimativas de amplitude e fase dos fasores de tensão e corrente do SEP. O detalhe destes algoritmos é que a frequência é obtida através da derivação numérica da fase estimada e filtros de média móvel são então utilizados

para suavizar o resultado, muito susceptível a variações espúrias, principalmente em razão de ruídos nos sinais. Outros algoritmos utilizando o critério LSE foram propostos por Terzija, Djuric e Kovacevic (1994) e De Almeida e Lima (1998). Neste último trabalho, o detalhe está na técnica apresentada para modificação da matriz de covariância, o que resulta em estimativas mais rápidas.

Os algoritmos genéticos, também conhecidos como algoritmos evolutivos, são algoritmos de busca baseados em mecanismo de seleção natural e genética natural. Dado um conjunto de indivíduos, a filosofia de um AG é a geração de descendentes a partir dos melhores indivíduos da população ou conjunto. Os indivíduos são classificados como strings (cromossomos) e para cada cromossomo há uma função de avaliação (*fitness*). Quanto melhor for a representação de um cromossomo a uma possível solução para o problema, maior será sua chance de sobreviver e gerar novos indivíduos. Os mecanismos de geração de indivíduos podem ser divididos de duas formas: genético e evolutivo. Os mecanismos genéticos, chamados de cruzamento e de mutação, são responsáveis pela troca das informações genéticas entre os indivíduos selecionados de modo a produzirem novos indivíduos. Os operadores evolutivos tratam em determinar quais os indivíduos sofrerão cruzamento ou mutação. Essencialmente, um AG tenta otimizar (maximizar ou minimizar) uma função objetivo. Em muitos casos, o desenvolvimento dessa função pode estar baseado no valor retornado pela mesma, o qual pode representar apenas uma avaliação parcial do problema. O algoritmo deve ser rápido, uma vez que o mesmo irá avaliar cada indivíduo de uma dada população em suas sucessivas gerações. É importante salientar que o paralelismo implícito dos AGs, devido à vasta diversidade de indivíduos (soluções) na população inicial, os torna uma poderosa ferramenta de busca, prevenindo convergências para soluções locais.

Os pioneiros na utilização de computação evolutiva para estimação de parâmetros e aplicação em relés de frequência foram EL-Naggar e Youssef (2000). Já De Macedo et al. (2003); Souza et al. (2006) desenvolveram um método de estimação de harmônicos baseados em algoritmos genéticos. Uma característica relevante nos resultados de Souza et al. (2006) é a obtenção de resultados precisos com janelas de 1/2 ciclo, fator relevante para muitas aplicações em SEP. Estes dois últimos trabalhos apresentam resultados de pesquisas desenvolvidas na EESC-USP no tema computação evolutiva.

Capítulo 3

Algoritmos Aplicados à Estimação Fasorial - Fundamentos Teóricos

Este capítulo apresenta os fundamentos teóricos dos métodos aplicados para estimação fasorial escolhidos para implementação neste trabalho. Primeiramente é apresentado o algoritmo que é o principal foco deste trabalho, o algoritmo genético. Em seguida são apresentados o algoritmo baseado em TDF e por fim o algoritmo baseado em PLL. É importante observar que utiliza-se o estado da arte com relação aos assuntos abordados.

3.1 Os Algoritmos Genéticos

AGs são algoritmos de busca e pesquisa utilizados na otimização de problemas, sendo um mecanismo de busca heurística inspirado na seleção natural e genética populacional. Esses algoritmos operam sobre um conjunto (ou população) de possíveis soluções (os indivíduos) para a otimização do problema, geralmente com inicialização aleatória. Com o processo de evolução, as características dos indivíduos são modificadas utilizando-se operadores genéticos e a melhoria do conjunto de soluções do sistema é alcançada. Ao fim de certo número de gerações, espera-se que boas soluções para o problema estejam disponíveis na população. Uma descrição mais detalhada sobre este tipo de algoritmos pode ser encontrada no Apêndice A.

Existem diversos tipos de codificação para os indivíduos, critérios de seleção, *crossover*, mutação além dos AGs baseados em probabilidade de distribuição. Nesta seção serão apresentados os fundamentos utilizados para a implementação deste trabalho.

O problema de otimização utilizando AGs para estimar o fasor fundamental e rastrear a frequência de um SEP é definido como um problema otimização de *curve-fitting* (ajuste de curvas), em que busca-se a curva mais semelhante à presente no sinal elétricos através de seus parâmetros (frequência, amplitude e ângulo de fase). Considera-se um modelo senoidal para o sinal elétrico

$$v(t) = A(t)\text{sen}(2\pi f(t)t + \theta(t)), \quad (3.1)$$

em que $A(t)$ é a amplitude, $f(t)$ é a frequência e $\theta(t)$ é o ângulo de fase. Pode-se observar que os parâmetros fasoriais (A, f, θ) são variantes no tempo. Tal consideração deve ser levado em fato devido a dinâmica que domina os SEPs.

Desta forma, pode-se definir uma função de erro aplicada sobre uma janela deslizante do sinal do sistema elétrico discretizado $u[n]$ a ser analisado. Dentro desta janela, considera-se os parâmetros fasoriais são invariantes, de forma que

$$e[n] = C(n, A, f, \theta) = \sum_{k=0}^{N-1} [u[n-k] - A\text{sen}(2\pi f k T_s + \theta)]^2, \quad (3.2)$$

em que T_s é o período de amostragem (isto é $T_s = 1/f_s$), N é o número de amostras em uma janela, $u[n]$ é o sinal analisado e $\psi = \{A, f, \theta\}$ é o conjunto dos parâmetros a ser determinado.

O processo de ajuste de curvas pode ser visto no exemplo da figura 3.1. Pode-se observar diversos indivíduos (curvas contínuas em diferentes cores), que são soluções válidas no espaço de busca. Observando a legenda, o *fitness*, ou adaptação de cada uma das curvas está indicada. A curva que tem um melhor ajuste é aquela com o menor erro, ou maior *fitness*, evidenciando que é uma melhor solução para o problema.

3.1.1 Codificação

Neste trabalho são utilizadas duas codificações diferentes. Para a análise computacional utilizando o AG tradicional, variáveis do tipo ponto-flutuante são utilizadas (codificação real). Para a utilização do AG compacto é utilizado a codificação binária. Na figura 3.2 pode ser observado um indivíduo com codificação real, e na figura 3.3 um indivíduo com codificação binária com N_A , N_f e N_θ bits disponíveis para a codificação da amplitude, frequência e fase respectivamente.

Figura 3.1: Processo de ajuste de curva utilizado pelo AG para estimação fasorial.

Amplitude	Frequência	Fase
$0 \leq A \leq 1, 1$	$58 \leq f \leq 62$	$0 \leq \theta \leq 2\pi$

Figura 3.2: Codificação Real.

Amplitude	Frequência	Fase
$A_{N_A-1} A_{N_A-2} A_{N_A-3} \dots A_0$	$f_{N_f-1} f_{N_f-2} f_{N_f-3} \dots f_0$	$\theta_{N_\theta-1} \theta_{N_\theta-2} \theta_{N_\theta-3} \dots \theta_0$
N_A	N_f	N_θ

Figura 3.3: Codificação Binária.

3.1.2 Seleção

O processo de seleção consiste na escolha aleatória de indivíduos para geração de indivíduos para reprodução. O operador de seleção por torneio é utilizado nesta etapa.

3.1.3 Cruzamento

A etapa de cruzamento direciona o processo evolucionário no sentido de se obter soluções potencialmente melhores. Este operador troca material genético de dois cromossomos resultantes do processo de seleção dos cromossomos p_1 e p_2 para a criação de um filho que pode se beneficiar das características de seus pais.

O cruzamento realizado neste trabalho é baseado no cruzamento real. O processo é ilustrado na figura 3.4.

Figura 3.4: Processo de Cruzamento

ψ representa o conjunto de parâmetros $\{A, f, \theta\}$. Cada parâmetro é combinado separadamente como mostrado nesta figura., isto é, através do cálculo do valor médio do parâmetro médio e de suas diferenças. O parâmetro filho é definido através da escolha aleatória dos cinco valores possíveis, ou seja, dos dois parâmetros pertencentes aos pais, bem como os novos parâmetros gerados. Este procedimento é efetuado três vezes, uma para cada parâmetro.

3.1.4 Mutação

A mutação é a operação responsável pela geração de diversidade do material genético na população. Basicamente, esta é aplicada separadamente a cada parâmetro do filho resultante do cruzamento, de acordo com uma taxa de mutação especificada. O operador de mutação utilizado é baseado no operador real, com a diferença de não ser diâmico. Basicamente é adicionada ou subtraída uma constante $\Delta\psi$ a cada $\psi = \{A, f, \theta\}$, ou seja,

$$\psi \leftarrow \psi \pm \Delta\psi. \quad (3.3)$$

3.2 A Transformada Discreta de Fourier

A TDF tem sido aplicada com sucesso em muitas aplicações para medição em SEPs. No entanto, a TDF é precisa somente quando o teorema de amostragem é satisfeito e a frequência de amostragem está sincronizada com a frequência do sinal analógico (número inteiro de amostras em um número inteiro de ciclos). Em termos práticos, a amostragem sincronizada é teórica, devido a imprecisões durante o processo de amostragem e flutuações desconhecidas na frequência do sinal analógico. Quando não ocorre esta sincronização, erros são inseridos devido

principalmente ao efeito de *leackage*.

Uma forma de estimar fasores mais precisamente mediante essa discrepância na amostragem é pela utilização do erro inserido no ângulo de fase. Wang e Sun (2004, 2006) apresentam um método de compensação preciso para estimação correta da frequência, amplitude e ângulo de fase utilizando informações do erro de ângulo de fase proveniente da aplicação TDF em duas janelas de dados distintas. A descrição detalhada é dada a seguir.

Considerando um sinal senoidal desconhecido com período T ,

$$x_a(t) = 2F \cos(2\pi ft + \varphi) = Fe^{j(\Omega t + \varphi)} + Fe^{-j(\Omega t + \varphi)} \quad (-\infty < t < +\infty), \quad (3.4)$$

em que $2F$ é a amplitude, φ é o ângulo de fase e $\Omega = 2\pi/T$ é a velocidade angular.

Amostrando $x_a(t)$ com frequência de amostragem $f_s = 1/T_s$ obtêm-se

$$x(n) = x_a(t)|_{t=nT_s} = Fe^{j\varphi}e^{jn\omega_0} + Fe^{-j\varphi}e^{-jn\omega_0} \quad (-\infty < n < +\infty), \quad (3.5)$$

em que $\omega_0 = \Omega_0 T_s = 2\pi T_s/T$.

Amostrando $x(n)$ em uma janela retangular, uma sequencia de N amostras é obtida,

$$x_d(n) = x(n)d(n),$$

em que $d(n)$ é uma janela retangular com comprimento N ($N \geq 4$),

$$x(n) = \begin{cases} 1, 0 \leq n \leq N-1 \\ 0, n < 0, n \geq N. \end{cases} \quad (3.6)$$

A Transformada de Fourier de Tempo Discreto (TFTD) de $x_d(n)$ é

$$X_d(\omega) = \sum_{n=-\infty}^{+\infty} x(n)d(n)e^{-jn\omega} = \sum_{n=0}^{N-1} x(n)e^{-jn\omega} = Fe^{j\varphi}D(\omega - \omega_0) + Fe^{-j\varphi}D(\omega + \omega_0), \quad (3.7)$$

em que $D(\omega)$ é a TFTD de $d(n)$,

$$D(\omega) = \sum_{n=0}^{N-1} e^{-jn\omega} = e^{-j\omega(N-1)/2} \frac{\text{sen}(\frac{\omega N}{2})}{\text{sen}(\frac{\omega}{2})}. \quad (3.8)$$

A TDF de $x_d(n)$ é obtida amostrando $X_d(\omega)$ no domínio da frequência,

$$\begin{aligned} X(l) &= \sum_{n=0}^{N-1} x(n) e^{-j(2\pi/N)nl} = X_d(\omega) \Big|_{\omega=l(2\pi/N)} \\ &= F e^{j\varphi} D\left(\frac{2\pi l}{N} - \omega_0\right) + F e^{-j\varphi} D\left(\frac{2\pi l}{N} + \omega_0\right) \quad (l = 0, \dots, N-1). \end{aligned} \quad (3.9)$$

O sinal senoidal $x_a(t)$ dado pela equação 3.4 pode ser representado por um fasor,

$$\bar{x} = N F e^{j\varphi} \quad (3.10)$$

Na equação 3.9, o coeficiente da componente fundamental $X(1)$ é o valor medido de $\bar{x}$,

$$X(1) = \sum_{n=0}^{N-1} x(n) e^{-j(2\pi/N)n} = F e^{j\varphi} D\left(\frac{2\pi l}{N} - \omega_0\right) + F e^{-j\varphi} D\left(\frac{2\pi l}{N} + \omega_0\right). \quad (3.11)$$

Se ocorre a sincronização na da frequência de amostragem com a frequência analógica, então $T/T_s = N$, assim

$$X(1) = \sum_{n=0}^{N-1} x(n) e^{-j(2\pi/N)n} = F e^{j\varphi} D(0) + F e^{-j\varphi} D\left(\frac{4\pi}{N}\right) = N F e^{j\varphi}. \quad (3.12)$$

Pela equação 3.12 pode-se observar que TDF é preciso quando a frequência de amostragem está sincronizada com a frequência do sinal analógico. No entanto esta sincronia é puramente teórica, devido à fatores apresentados anteriormente. Baseado em uma análise mais detalhada do erro do ângulo de fase, um método prático e preciso para estimação dos parâmetros de frequência, amplitude e ângulo de fase é apresentado por Wang e Sun (2004, 2006).

Define-se

$$\lambda = \frac{N T_s}{T} = \frac{N f}{f_s} = 1 + \frac{(N f - f_s)}{f_s} = 1 + \Delta\lambda, \quad (3.13)$$

em que f é a frequência do sinal analógico, f_s é a frequência de amostragem e N é o número de amostras. $\Delta\lambda$ representa o grau de desvio da frequência nominal. Desta forma têm-se

$$\omega_0 = \frac{2\pi T_s}{T} = \frac{2\pi\lambda}{N}. \quad (3.14)$$

De acordo com a equação 3.14, o valor medido de $\bar{x}$ é

$$\begin{aligned}
X(1) &= NF_m e^{j\varphi_m} = F e^{j\varphi} D\left(\frac{-2\pi\Delta\lambda}{N}\right) + F e^{-j\varphi} D\left(\frac{2\pi(2+\Delta\lambda)}{N}\right) \\
&= F e^{j\varphi} \cdot e^{j\Delta\lambda\pi(N-1)/N} \cdot \frac{\text{sen}(\Delta\lambda \cdot \pi)}{\text{sen}(\frac{\Delta\lambda \cdot \pi}{N})} + F e^{-j\varphi} \cdot e^{-j\Delta\lambda\pi(N-1)/N + j2\pi/N} \cdot \frac{\text{sen}(\Delta\lambda \cdot \pi)}{\text{sen}(\frac{\Delta\lambda \cdot \pi}{N} + \frac{2\pi}{N})}, \quad (3.15)
\end{aligned}$$

em que F_m e φ_m representam os valores medidos de F e φ .

O efeito da amostragem assíncrona ($\Delta\lambda \neq 0$) para a medida de $\bar{x}$ pode ser observado na figura 3.5. Pode-se observar os vetores formados pelas duas parcelas da equação 3.15, resultando no vetor medido.

Figura 3.5: Efeito da amostragem assíncrona na medida de $\bar{x}$.

São utilizadas algumas aproximações para a resolução do problema, como $\tan(\alpha) \approx \alpha$ quando α é pequeno. Desta forma, da figura 3.5 têm-se que

$$\Delta\varphi_1 = \Delta\lambda \cdot \frac{(N-1)}{N} \quad (3.16)$$

e

$$\Delta\varphi_2 \approx \frac{\text{sen}(\frac{\Delta\lambda \cdot \pi}{N})}{\text{sen}(\frac{\Delta\lambda \cdot \pi}{N} + \frac{2\pi}{N})} \cdot \text{sen}\left(\frac{2\pi}{N} - 2\varphi - 2\Delta\varphi \frac{N-1}{N}\right). \quad (3.17)$$

O erro total do ângulo de fase $\Delta\varphi$ é

$$\Delta\varphi = \Delta\varphi_1 + \Delta\varphi_2 \quad (3.18)$$

Portanto, uma formula prática para calcular o ângulo de fase φ verdadeiro é obtida por

$$\begin{aligned}
\varphi &= \varphi_m - \Delta\varphi_1 - \Delta\varphi_2 \\
&\approx \varphi_m - \Delta\lambda\pi \cdot \frac{N-1}{N} - \frac{\Delta\lambda \cdot \pi}{N \cdot \text{sen}(\frac{2+\Delta\lambda}{N}\pi)} \cdot \text{sen}\left(\frac{2\pi}{N} - 2\varphi - 2\Delta\varphi\pi \frac{N-1}{N}\right) \\
&\approx \varphi_m - \Delta\lambda\pi \cdot \frac{N-1}{N} - \frac{\Delta\lambda \cdot \pi}{N \text{sen}(\frac{2+\Delta\lambda}{N}\pi)t} \cdot \text{sen}\left(\frac{2\pi}{N} - 2\varphi_m\right)
\end{aligned} \tag{3.19}$$

Desta forma, a partir da equação 3.19, é possível extrair informações através de dois sinais amostrados em janelas em instantes de tempo diferente, como será mostrado a seguir.

Amostrando o sinal $x(n)$ com janelas retangulares $d(n)$ e $d(n-M)$ respectivamente, são obtidas duas sequências de tamanho N

$$x_d(n) = x(n) \cdot d(n) \tag{3.20}$$

e

$$x_d^M(n) = x(n) \cdot d(n-M). \tag{3.21}$$

As representações fasoriais de $x_d(n)$ e $x_d^M(n)$ são

$$\bar{x}_1 = NF e^{j\varphi_1} \tag{3.22}$$

e

$$\bar{x}_2 = NF e^{j\varphi_2}, \tag{3.23}$$

em que φ_1 e φ_2 são os ângulos de fase verdadeiros de $\bar{x}_1$ e $\bar{x}_2$, que satisfazem

$$\varphi_2 - \varphi_1 = \frac{2\pi MT_s}{T} = (1 + \Delta\lambda) \cdot \frac{2\pi M}{N}. \tag{3.24}$$

Aplicando a TDF em $x_d(n)$ e $x_d^M(n)$, de acordo com a equação 3.19 obtêm-se

$$\varphi_{1m} \approx \varphi_1 + \Delta\lambda\pi \cdot \frac{N-1}{N} + \frac{\Delta\lambda \cdot \pi}{N \cdot \text{sen}(\frac{2\pi}{N})} \cdot \text{sen}\left(\frac{2\pi}{N} - 2\varphi_{1m}\right) \tag{3.25}$$

e

$$\varphi_{2m} \approx \varphi_2 + \Delta\lambda\pi \cdot \frac{N-1}{N} + \frac{\Delta\lambda \cdot \pi}{N \cdot \text{sen}(\frac{2\pi}{N})} \cdot \text{sen}\left(\frac{2\pi}{N} - 2\varphi_{2m}\right), \tag{3.26}$$

em que φ_{1m} e φ_{2m} são respectivamente os valores medidos de φ_1 e φ_2 .

O processo pode ser observado em detalhes na figura 3.6, em que são observadas duas janelas de dados, com os respectivos ângulo de fase medidos, separados por M amostras.

Figura 3.6: Estimação fasorial utilizando transformada discreta de Fourier

Sabendo que $\varphi_1 = \varphi_2 + \frac{2\pi M}{N}$, e utilizando as equações 3.25 e 3.26, obtêm-se

$$\Delta\lambda = \frac{(\varphi_{2m} - \varphi_{1m}) - K_3}{K_3 + K(K_2 - K_1)}, \quad (3.27)$$

em que

$$K = \frac{\pi}{(N \cdot \text{sen}(\frac{2\pi}{N}))} \quad (3.28)$$

$$K_1 = \text{sen}\left(\frac{2\pi}{N} - 2\varphi_{1m}\right) \quad (3.29)$$

$$K_2 = \text{sen}\left(\frac{2\pi}{N} - 2\varphi_{2m}\right) \quad (3.30)$$

$$K_3 = \frac{2\pi M}{N}. \quad (3.31)$$

Desta forma, a frequência real de $x_a(t)$ é

$$f = \frac{f_s(1 + \Delta\lambda)}{N}. \quad (3.32)$$

A fim de se tornar o resultado mais preciso, pode-se modificar K com $\Delta\lambda$ calculado pela

equação 3.27,

$$K = \frac{\pi}{N \cdot \text{sen}\left(\frac{2+\Delta\lambda}{N}\pi\right)}, \quad (3.33)$$

e então utilizar a equação 3.27 o novo $\Delta\lambda$.

A amplitude é obtida substituindo-se $\Delta\lambda$ obtido pela equação 3.27 na equação 3.15,

$$F = \frac{NF_m e^{j\varphi_m}}{e^{j\varphi} D\left(\frac{-2\pi\Delta\lambda}{N}\right) + e^{-j\varphi} D\left(\frac{2\pi(2+\Delta\lambda)}{N}\right)}. \quad (3.34)$$

No entanto, a avaliação desta equação é muito onerosa computacionalmente, desta forma, pode-se aplicar simplificações, tornando a avaliação de F mais viável,

$$F \approx \frac{F_m}{1 - \frac{\pi^2}{6}\Delta\lambda^2 + K\Delta\lambda \cdot \cos\left(\frac{2\pi}{N} - 2\varphi_m\right)} \approx F_m \left[1 + \frac{\pi^2}{6}\Delta\lambda^2 - K\Delta\lambda \cdot \cos\left(\frac{2\pi}{N} - 2\varphi_m\right) \right] \quad (3.35)$$

3.2.1 Resumo para aplicação prática

Anteriormente foram apresentados os fundamentos para a estimação dos três parâmetros que compõem um fasor: amplitude, fase e frequência. Um resumo prático para a implementação do algoritmo TDF aplicado para estimação fasorial desenvolvido por Wang e Sun (2004, 2006) é apresentado em alguns passos.

1. Determine o tamanho N das janelas de dados, o deslocamento de M amostras entre estas e calcule:

$$\begin{aligned} K &= \frac{\pi}{(N \cdot \text{sen}\left(\frac{2\pi}{N}\right))} \\ K_3 &= \frac{2\pi M}{N} = 2\pi \\ K_4 &= \frac{\pi(N-1)}{N} \end{aligned}$$

Estas constantes são calculadas somente uma vez no início do algoritmo, não tendo influência no restante.

2. Calculo da TDF recursiva

$$X^{(r+1)} = \left(X^{(r)} - x(0) \right) e^{j(2\pi/N)} + x(N) \cdot e^{j(2\pi/N)}. \quad (3.36)$$

3. Estimação dos parâmetros fasoriais

$$\begin{aligned} K_1 &= \text{sen} \left(\frac{2\pi}{N} - 2\varphi_{1m} \right) \\ K_2 &= \text{sen} \left(\frac{2\pi}{N} - 2\varphi_{2m} \right) \\ \Delta\lambda &= \frac{(\varphi_{2m} - \varphi_{1m}) - K_3}{K_3 + K(K_2 - K_1)} \\ \varphi_2 &\approx \varphi_{2m} - \Delta\lambda \cdot K_4 - \Delta\lambda \cdot K \cdot K_2 \\ f &= \frac{f_s(1 + \Delta\lambda)}{N} \\ F &= F_m \left[1 + \frac{\pi^2}{6} \Delta\lambda^2 - K \Delta\lambda \cdot \cos \left(\frac{2\pi}{N} - 2\varphi_m \right) \right] \end{aligned} \quad (3.37)$$

Volte para o passo 2.

3.3 *Phase-Locked Loop*

Um outro algoritmo capaz de extrair os parâmetros de frequência, amplitude e fase é apresentado por Ziarani e Konrad (2004) e mais tarde melhorado por Karimi-Ghartemani, Karimi e Iravani (2004). Um conjunto de equações diferenciais não lineares governa a dinâmica deste algoritmo. O método do gradiente descendente é usado para minimizar o erro mínimo quadrático entre o sinal de entrada e o sinal senoidal desejado. As deduções são apresentadas de forma resumida baseando-se no trabalho de Carvalho (2008).

Seja $u(t)$ a representação de tensão ou corrente de um sinal. Esta função geralmente é contínua e quase sempre periódica. Um componente senoidal desta função, $y(t) = A \sin \phi(t)$, é de interesse, em que A é a amplitude e $\phi(t)$ representa a fase total deste componente. Quando a frequência é fixa, o termo $\phi(t)$ pode ser expressado como $\omega t + \delta$, em que ω é a frequência expressa em radianos e δ é a constante de fase. Idealmente, ou em regime permanente, os parâmetros A , ω e δ são quantidades fixas. No entanto, o mesmo não ocorre na prática. Em uma

situação típica, $u(t)$ tem a forma

$$u(t) = \sum_{i=0}^{\infty} A_i \text{sen} \phi_i + n(t), \quad (3.38)$$

em que $n(t)$ denota o ruído superimposto.

No algoritmo proposto, o objetivo é extrair a componente senoidal de $u(t)$. Tal componente desejado pode ser expresso como $y(t) = A \text{sen} \phi(t)$, incorporando todas as variações de fase no termo $\phi(t)$. Para incorporar explicitamente o conceito de frequência instantânea na fase global, pode-se definir

$$y(t) = A(t) \text{sen} \left(\int^t \omega(\tau) d\tau + \delta(t) \right) \quad (3.39)$$

Seja $\mathcal{M}$ um conjunto contendo todos os sinais senoidais definidos como

$$\mathcal{M} = \left\{ A(t) \text{sen} \left(\int^t \omega(\tau) d\tau + \delta(t) \right) \left| \begin{array}{l} A(t) \in [A_{min}, A_{max}] \\ \omega(t) \in [\omega_{min}, \omega_{max}] \\ \delta(t) \in [\delta_{min}, \delta_{max}] \end{array} \right. \right\} \quad (3.40)$$

Desta forma, procura-se uma componente em $\mathcal{M}$ que mais se assemelhe a componente de interesse presente em $u(t)$. Define-se o vetor de parâmetros a ser definido como

$$\Psi(t) = [A(t) \ \omega(t) \ \delta(t)]^T. \quad (3.41)$$

A saída é definida como o componente senoidal desejado,

$$y(t, \Psi(t)) = A(t) \text{sen} \left(\int^t \omega(\tau) d\tau + \delta(t) \right). \quad (3.42)$$

Pode-se definir uma função de custo J , quadrado do erro instantâneo,

$$\begin{aligned} J(t, \Psi(t)) &= \{e(t)\}^2 \\ &= \{u(t) - A(t) \text{sen}[\phi(t)]\}^2 \\ &= \{u(t) - y(t, \psi(t))\}^2 \end{aligned} \quad (3.43)$$

Utilizando o método dos Gradientes Descendentes, o vetor dos parâmetros pode ser estimado

$$\frac{d\Psi(t)}{dt} = -\mu \frac{\partial J(t, \Psi(t))}{\partial \Psi(t)}, \quad (3.44)$$

em que a matriz diagonal positiva μ é a constante de regulação do algoritmo, que controla a velocidade de convergência. O método do Gradiente Descendente garante a convergência para a solução desejada se a função de custo é globalmente quadrática nos parâmetros. Caso contrário, se a forma da função de custo não é quadrática - como é o caso descrito anteriormente - ou não claramente definida, deve-se averiguar a convergência pela utilização de uma prova matemática. Esta garantia é apresentada por Ziarani e Konrad (2004).

Procedendo com a expansão da equação 3.43,

$$\begin{bmatrix} \frac{dA(t)}{dt} \\ \frac{d\omega(t)}{dt} \\ \frac{d\delta(t)}{dt} \end{bmatrix} = - \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial A(t)} \left[u(t) - A(t) \text{sen} \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \right]^2 \\ \frac{\partial}{\partial \omega(t)} \left[u(t) - A(t) \text{sen} \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \right]^2 \\ \frac{\partial}{\partial \delta(t)} \left[u(t) - A(t) \text{sen} \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \right]^2 \end{bmatrix}. \quad (3.45)$$

Calculando as derivadas parciais e utilizando a definição de erro da equação 3.43, obtêm-se

$$\begin{aligned} \frac{dA(t)}{dt} &= 2m_1 e(t) \cdot \text{sen} \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \\ \frac{d\omega(t)}{dt} &= 2m_2 e(t) A(t) t \cdot \cos \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \\ \frac{d\delta(t)}{dt} &= 2m_3 e(t) A(t) \cdot \cos \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \end{aligned} \quad (3.46)$$

A variável t aparece explicitamente na segunda linha do conjunto de equações 3.46, o que implica em um sistema variante no tempo. Ziarani e Konrad (2004) observou que este sistema é instável, e portanto, sem utilidade prática. A solução heurística empregada para tornar o sistema invariante no tempo é substituir a variável t por uma constante m_4 , sendo estável e com desempenho satisfatório para aplicações com sinais elétricos.

Des de que

$$\phi(t) = \int_0^t \omega(\tau) d\tau + \delta(t), \quad (3.47)$$

derivando-se em ambos os lados da equação 3.47, obtêm-se

$$\frac{d\phi(t)}{dt} = \omega(t) + \frac{d\delta(t)}{dt}. \quad (3.48)$$

Isolando-se $d\delta/dt$ na equação 3.48 e substituindo-se no conjunto de equações 3.46 obtêm-se

$$\begin{aligned} \frac{A(t)}{dt} &= 2m_1 e(t) \text{sen} \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \\ \frac{\omega(t)}{dt} &= 2m_2 m_4 e(t) A(t) \cos \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \\ \frac{\phi(t)}{dt} &= \omega(t) + 2m_3 e(t) A(t) \cos \left(\int_0^t \omega(\tau) d\tau + \delta(t) \right) \end{aligned} \quad (3.49)$$

Karimi-Ghartemani, Karimi e Iravani (2004) utilizam uma simplificação do conjunto de equações 3.49 que consiste em incorporar $A(t)$, das equações de frequência e fase total, nas constantes das respectivas equações. Considera-se esse procedimento como coerente, levando em conta a convergência deste parâmetro, após o transitório do algoritmo. Desta forma, agrupando-se as constantes e utilizando-se a equação 3.48, chega-se ao conjunto de equações diferenciais que regem o comportamento do sistema PLL utilizado neste trabalho

$$\begin{aligned} \frac{A(t)}{dt} &= \mu_1 e(t) \text{sen}(\phi(t)) \\ \frac{\omega(t)}{dt} &= \mu_2 e(t) \cos(\phi(t)) \\ \frac{\phi(t)}{dt} &= \omega(t) + \mu_3 e(t) \cos(\phi(t)), \end{aligned} \quad (3.50)$$

cujo diagrama de blocos pode ser observado na figura 3.7.

Figura 3.7: Diagrama de blocos do sistema PLL estudado.

O sistema apresentado pela figura 3.7 pode ser implementado em um circuito analógico. Para sua utilização digital, o conjunto de equações diferenciais 3.50 deve ser discretizado. Tal

processo é efetuado utilizando-se o método *Euler Forward*, que resulta

$$\begin{aligned}A[n+1] &= A[n] + \mu_1 T_s e[n] \text{sen}(\phi[n]) \\ \omega[n+1] &= \omega[n] + \mu_2 T_s e[n] \cos(\phi[n]) \\ \phi[n+1] &= \phi[n] + T_s \omega[n] + \mu_3 T_s e[n] \cos(\phi[n]),\end{aligned}\tag{3.51}$$

em que T_s é o período de amostragem e os fatores μ_1 , μ_2 e μ_3 são as constantes positivas que ditam o comportamento do algoritmo em regime transitório e permanente. O sinal de erro discretizado é

$$\begin{aligned}e[n] &= u[n] - y[n] \\ &= u[n] - A[n] \text{sen}(\phi[n]),\end{aligned}\tag{3.52}$$

em que $u[n]$ é o sinal de entrada amostrado com período T_s . Deve-se notar que os valores de $A[0]$, $\omega[0]$ e $\phi[0]$ devem ser inicializados com valores próximos aos valores nominais.

Capítulo 4

Dados Utilizados para Comparação dos Algoritmos Propostos

Para avaliar de forma correta os algoritmos desenvolvidos, é necessário criar dados para teste em que obtêm-se controle sobre as características das formas de onda a serem analisadas, a fim de se avaliar a precisão dos mesmos. Além disso, para testar a viabilidade prática dos algoritmos, são necessários dados provenientes de um sistema elétrico. Este capítulo apresenta a metodologia desenvolvida para a criação de dados sintéticos para teste bem como dados provenientes de simulações utilizando-se do *software* ATP, em que diversas situações de funcionamento de um SEP serão testadas.

4.1 Dados Sintéticos

Dados sintéticos são dados aplicáveis a uma determinada situação que não são obtidos por medição direta. Estes dados são geralmente utilizados para atingir certas necessidades ou condições que não estão presentes em dados reais. Além disso, pode-se utiliza-los quando o acesso à dados reais é restrito ou não se tem controle sobre as qualidades destes.

Para o desenvolvimento deste trabalho, dados sintéticos são utilizados para testar a precisão e comportamento dos métodos de estimação fasorial implementados. Neste contexto, tem-se controle sobre os parâmetros da forma de onda que é um parâmetro de entrada para os algoritmos. Desta forma, a partir de dados sintéticos em que se tem certeza sobre os parâmetros de frequência, amplitude e angulo de fase pode-se estimar adequadamente a precisão dos métodos

avaliando-se o TVE.

Os dados sintéticos utilizados foram gerados de acordo com o Apêndice G da norma IEEE (2005), que detalha testes de desempenho mediante situação transitórias. Tal escolha se deu pelo fato de se poder avaliar simultaneamente o TVE e o tempo necessário para que o algoritmo, após o transitório, volte a ter TVE inferior a 1%, conforme descrito no Capítulo 2.

Os testes de desempenho propostos pelo Apêndice G da norma IEEE (2005), base para a geração dos dados sintéticos neste trabalho, são os seguintes:

- Degrau de magnitude de 10%;
- Degrau de fase de 90°;
- Degrau frequência de 5 Hz.

A seguir as equações para a geração dos dados sintéticos são devidamente detalhadas.

4.1.1 Teste de degrau de magnitude (10%)

A forma de onda de teste de degrau de magnitude está descrita na Equação 4.1

$$\begin{aligned}x(t < 0) &= X_{m1}\cos(\omega_0 t) \\x(0) &= \frac{X_{m1} + X_{m2}}{2}\cos(\omega_0 t) \\x(t > 0) &= X_{m2}\cos(\omega_0 t)\end{aligned}\tag{4.1}$$

, em que X_{m1} representa a tensão ou corrente inicial e $X_{m2} = 0,9X_{m1}$ para o caso estudado. O valor pode ser modificado para melhor estudar o comportamento do algoritmos testados. A Figura 4.1 apresenta a forma de onda do teste de degrau de magnitude de 10%.

Pode-se notar que após o instante 0, ocorre uma queda na amplitude da forma de onda.

4.1.2 Teste de degrau de fase (90°)

A forma de onda de teste de degrau de fase está descrita na Equação 4.2.

Figura 4.1: Forma de onda do teste de degrau de Magnitude

$$\begin{aligned}
 x(t < 0) &= X_m \cos(\omega_0 t) \\
 x(0) &= X_m \cos\left(\omega_0 t + \frac{\pi}{4}\right) \\
 x(t > 0) &= X_m \cos\left(\omega_0 t + \frac{\pi}{2}\right)
 \end{aligned} \tag{4.2}$$

, em que X_m é o sinal de entrada de tensão ou corrente. Pode-se escolher outro desvio de fase a fim de se avaliar o desempenho dos algoritmos. A Figura 4.2 apresenta a forma de onda do teste de degrau de fase de 90° .

Figura 4.2: Forma de onda do teste de Degrau de Fase

Pode-se observar que após o instante 0, ocorre a inversão de fase do sinal, ficando visível.

4.1.3 Teste de degrau de frequência (5 Hz)

A forma de onda de teste de degrau de frequência está descrita na Equação 4.3

$$\begin{aligned}
x(t < 0) &= X_m \cos(\omega_0 t) \\
x(0) &= X_m \\
x(t > 0) &= X_m \cos[2\pi(f_0 + 5)t]
\end{aligned} \tag{4.3}$$

, em que X_m é o sinal de entrada de tensão ou corrente. Pode-se escolher outro desvio de frequência a fim de se avaliar o desempenho dos algoritmos. A Figura 4.3 apresenta a forma de onda do teste de degrau de frequência de 5Hz.

Figura 4.3: Forma de onda do teste de Degrau de Frequência

Observando-se a Figura 4.3, pode-se notar que após o instante 0, o sinal tem seu período diminuído, evidenciando o degrau de frequência.

4.2 Sistema Elétrico Proposto

Para a realização dos testes, com a consequente validação do algoritmo proposto, será utilizado o modelo completo de um SEP, no qual serão incluídos elementos de geração, transmissão e distribuição de energia elétrica. A figura 4.4 mostra um esquema representativo do sistema elétrico considerado.

O sistema elétrico consiste em três subsistemas interligados. O subsistema superior, enumerado por 3, possui um gerador síncrono de 13,8 kV com potência de 90MVA conectado a um transformador elevador delta-estrela aterrada com relação de 13,8kV/230kV. Esse transformador o conecta a linha de transmissão 3 de 230kV. A linha 17 termina no barramento de interligação 3 onde, através de transformadores abaixadores 230/13,8kV delta-estrela aterrada será conectado

Figura 4.4: Sistema Elétrico proposto

as cargas do subsistema 3.

O subsistema logo abaixo, enumerado por 1, possui 3 geradores 15 kV com potência 445 MVA. Cada gerador está conectado a um transformador elevador delta-estrela aterrada com a relação de 15/525kV com potência de 476 MVA. Os transformadores conectam os geradores a barra de geração 1 que será ligada a linha de transmissão 1 de 525 kV. Essa linha levará ao barramento de interligação 1, onde haverá outro transformador igual ao utilizado nos geradores, porém agora abaixador, para conectar o barramento com a carga.

O subsistema inferior, enumerado por 2, consiste de um gerador de 13,8kV com potência de 200MVA, 4 transformadores (1 elevador, 3 abaixadores) 13,8/230 kV, uma linha de transmissão de 230kV e cargas cuja tensão é 13,8kV. O subsistema 1 é interligado com o subsistema 2 e 3 através das linhas de transmissão de interligação de 230kV.

O sistema proposto foi implementado no *software* digital de simulações de sistemas elétricos ATP. Fora utilizados modelos completos de linhas de transmissão, geradores e reguladores de tensão e velocidade, descritos apropriadamente a seguir.

4.2.1 Dinâmica dos Sistemas Elétricos de Potência

Um sistema elétrico de potência se comporta como um sistema mecânico em rotação. A Figura 4.5 mostra um sistema elétrico simplificado a fim de se explicar a dinâmica deste sistema mecânico. Potência mecânica é produzida pela água ou vapor, causando um torque mecânico T_m no eixo que une a turbina ao gerador. O gerador transforma a potência mecânica em potência elétrica. A carga conectada ao gerador causa um torque elétrico T_e no eixo. Como mostrado na Figura 4.5, uma mudança no consumo de potência ou na sua produção causa uma flutuação na velocidade do conjunto turbina-gerador, e por consequência em uma flutuação na frequência do sistema.

Figura 4.5: Dinâmica dos Sistemas Elétricos de Potência

Uma diferença entre o torque elétrico e o mecânico resulta em um torque de aceleração T_a , como mostra a Equação 4.4:

$$T_a = T_m - T_e \quad (4.4)$$

A equação do movimento mostra a relação entre o torque de aceleração e a aceleração angular da turbina, dado pela Equação 4.5, como segue:

$$T_a = T_m - T_e = J \frac{\partial \omega}{\partial t}, \quad (4.5)$$

em que:

- J é o momento de inércia ($kg - m^2$)
- ω é a velocidade angular (rad/s)
- $\frac{\partial \omega}{\partial t}$ é a aceleração angular (rad/s^2)
- T_a é o torque resultante ($N - m$ ou J/rad)

Para pequenas variações pode-se considerar:

$$P_m \approx T_m$$

$$P_e \approx T_e$$

Então:

$$P_m - P_e = J \frac{\partial \omega}{\partial t} \quad (4.6)$$

Observando-se a Equação 4.6, pode-se concluir que qualquer variação no equilíbrio carga-geração ($P_m - P_e \neq 0$) resulta em uma variação na frequência do sistema.

A fim de se estudar esse fenômeno por meio de simulações, o *software* ATP será empregado. As simulações farão uso dos modelos completos de geradores e modelos de reguladores de tensão e de velocidade.

4.2.2 Reguladores de Velocidade

O papel dos reguladores de velocidade é corrigir possíveis déficits entre geração e carga controlando a potência aplicada na turbina pela vazão de água ou vapor. Desta forma, o regulador de velocidade pode ser modelado como um sistema em malha fechada como observado nas Figuras 4.6 e 4.7 em que observa-se um regulador de uma turbina hidráulica e uma turbina térmica com reaquecimento, respectivamente Vieira Filho (1984).

Os parâmetros adotados para o regulares de velocidade são típicos de conjuntos de sistemas hidráulicos, e pode ser observados na Tabela 4.1.

Os parâmetros utilizados para a turbina a vapor pode ser observados na Tabela 4.2.

Os sistemas mostrados anteriormente recebem como entrada a diferença de frequência $\Delta F(s)$ entre a frequência nominal de referência e a frequência do sistema. A saída do regulador de

Figura 4.6: Sistema de Regulação de Velocidade de Turbina Hidráulica

Tabela 4.1: Parâmetros utilizados no reguladores de velocidade das turbinas hidráulicas

Descrição	Parâmetros	
	Unidades de Geração 1	Unidades de Geração 3
Constante de Tempo (T_g)	0,2 (s)	0,4(s)
Constante de Tempo <i>Dashpot</i> (T_t)	3 (s)	1,2(s)
Estatismo Transitório (r)	2,3	2
Estatismo Permanente (R)	0,1	0,1
Constante de Tempo de Partida da Água (T_w)	0,15 (s)	0,15 (s)

Figura 4.7: Sistema de Regulação de Velocidade de Turbina Térmica com Reaquecimento

Tabela 4.2: Parâmetros utilizados no regulador de velocidade da turbina a vapor

Descrição	Parâmetros Unidade de Geração 2
Constante de Tempo do Regulador (T_1)	0,02 (s)
Constante de Tempo da Turbina (T_s)	0,25(s)
Estatismo Permanente (R)	0,2
Constante de Tempo de Reaquecimento T_r	4,5 (s)
Constante de Reaquecimento C	0,4

velocidade é a diferença de abertura ΔA que deve ser aplicada na turbina. A saída da turbina é a diferença de potência.

4.2.3 Reguladores de Tensão

O regulador de tensão controla a excitação no estator do gerador, controlando assim a tensão. O sistema de controle descrito na Figura 4.8 foi obtido de Barbosa (2010).

Figura 4.8: Regulador de Tensão

4.2.4 O *software* ATP

Os SEPs são planejados para disponibilizar energia com qualidade, confiabilidade e continuidade. No entanto, os SEPs estão constantemente expostos às várias contingências, tais como, descargas atmosféricas e outras intempéries, curtos-circuitos, falhas na operação e dispositivos (geradores, transformadores, cabos, disjuntores, chaves de manobra, barramentos, relés e motores, entre outros). Das consequências, tais contingências podem prejudicar os sistemas que estejam interligados.

Neste contexto, torna-se necessário a execução de um prévio estudo dos possíveis transitórios que podem ocorrer em um sistema elétrico. Este estudo poderá ter o intuito, tanto para a definição de uma adequada especificação dos dispositivos de proteção, quanto para se determinar os motivos que ocasionaram uma anomalia comportamental do sistema elétrico (Amon Filho e Pereira, 1996). Dentre os mecanismos existentes para a realização de tal estudo, os simuladores digitais têm se mostrado uma ferramenta em potencial devido ao aumento no poder de processamento dos computadores atuais.

Atualmente, o ATP possibilita a utilização de configurações arbitrárias em simulações de transitórios eletromagnéticos em redes polifásicas. Além destas, outras características constituem o grupo de definições passíveis de utilização, tais como não-linearidades, elementos com parâmetros concentrados e distribuídos, diversos tipos de chaves, transformadores e reatores, dentre outros.

Vale frisar que por ser um programa digital, pelo ATP não é possível se obter uma resposta

continua no tempo, sendo assim, caracterizada por valores a intervalos de tempo discretos. Entretanto, este fato não reduz a precisão nos resultados obtidos pelo ATP. Este software apresenta precisão suficiente para ser utilizado em diversas categorias de estudos. Conforme pode ser comprovado, realizando-se uma comparação dos resultados obtidos pela utilização do ATP com dados reais, percebe-se uma boa correlação entre os mesmos (Amon Filho e Pereira, 1996).

Contudo, a documentação do software ATP consiste basicamente de um manual (*ATP Rule Book*), o qual possui todas as informações sobre os modelos disponíveis. A utilização do ATP abrange tanto a área acadêmica quanto empresarial. Os resultados apresentados por sua utilização possuem uma elevada aceitação e validação, o que pode ser verificado, por exemplo, por sua grande utilização em trabalhos publicados nos periódicos do *Conseil International des Grands Réseaux Electriques* (CIGRE) e do IEEE.

4.2.5 Situações Simuladas no *software* ATP

A seguir são apresentados exemplos de funcionamento do *software* ATP para duas situações distintas de funcionamento do sistema elétrico proposto. A primeira retrata uma falta franca em uma linha de transmissão e a segunda uma situação de ilhamento de uma porção do sistema. Os cartões do ATP utilizados para realizar os testes a seguir estão disponíveis no Apêndice B.

Falta Franca na Linha de Transmissão 1

A seguir são evidenciadas as formas de onda de tensão e corrente medidas no barramento BGER3 para uma falta franca do tipo fase-terra aplicada a 75 Km da linha de transmissão LT3.

A partir da Figura 4.9 pode-se observar as formas de onda tensão e corrente. Na Figura 4.9(b) pode-se observar a contribuição da Fase A para a corrente de falta, evidenciando que a situação foi corretamente simulada.

Ilhamento

Para testar a dinâmica simulada no sistema proposto, foi simulado uma situação de ilhamento, em que uma porção do sistema é isolada, ocorrendo um déficit entre geração e consumo de potência nesta porção. Em um determinado instante, a linha de transmissão LT13 foi desligada, formando dois subsistemas. A porção conectada diretamente ao gerador GER31 possui

(a) Tensões

(b) Correntes

Figura 4.9: Falta Franca na linha de transmissão 1 a 75 Km da linha

mais cargas do que potência conectada, enquanto a porção restante possui mais potência de geração. O resultado pode ser observado na Figura 4.10.

Pode-se observar que após o desligamento da linha de transmissão LT13, ocorre uma queda na frequência do gerador GER31, evidenciando o desbalanço formado entre carga e geração.

Figura 4.10: Situação de Ilhamento

Para o restante do sistema, a frequência é aumentada.

Capítulo 5

Arquiteturas Desenvolvidas

Este capítulo tem por finalidade descrever as ferramentas utilizadas para o desenvolvimento deste trabalho. A descrição da arquitetura de *software* e *hardware* criadas utilizando estas ferramentas para a implementação dos algoritmos de estimação fasorial também é descrita. Optou-se pela abordagem da criação de "arquiteturas" devido a quantidade de dados a serem tratados bem como número de algoritmos a serem implementados. Além disso, tal abordagem tem como vantagem a flexibilização da pesquisa, pela utilização de modularização, permitindo que novos algoritmos possam ser testados com facilidade e que os algoritmos existentes também possam ser alterados sem grandes esforços. Para a implementação computacional dos algoritmos, foi utilizada a linguagem de programação C++. Tal escolha se deve ao fato desta linguagem ser orientada a objetos, permitindo assim uma melhor organização do código e a modularização dos algoritmos implementados. Para a implementação do AGc em *hardware*, foi utilizada uma FPGA da Altera, lançando-se mão da linguagem *Very high speed integrated circuit Hardware Description Language* (VHDL) para definir a lógica por ela implementada. Foi também utilizado um processador *softcore* NIOS II nessa implementação para facilitar o teste dos algoritmos, que são implementados como lógica "customizada" pela FPGA.

5.1 Implementação Computacional

Como citado anteriormente, os algoritmos citados neste trabalho foram implementados computacionalmente pela utilização da linguagem de programação C++ (Deitel e Deitel, 2005). O uso desta linguagem permitiu a modularização da arquitetura de testes, descrita na Figura 5.1

Figura 5.1: Arquitetura computacional implementada para testes

Os algoritmos abordados neste trabalho, AG, AGc, TDF e PLL foram implementados por classes com uma interface de entrada e saída comum entre estas. Esta interface de comunicação permite o acesso a canais de dados, que estão ligadas a arquivos de dados de entrada e saída.

Os arquivos de entrada e de saídas são arquivos de dados dispostos em colunas separados por tabulação em que os arquivos de entrada contém as formas de onda a serem analisadas e os arquivos de saída contém os parâmetros fasoriais estimados.

A cada iteração da arquitetura de teste computacional, uma amostra é lida do arquivo de entrada e disponibilizada no canal de entrada. Os algoritmos, fazendo uso da interface de entrada comum fazem o processamento da nova amostra e disponibilizam o resultado no canal de saída, que por fim é gravado no arquivo de saída para posterior análise.

5.2 Implementação dos AGs em *Hardware*

A seguir são apresentadas as ferramentas utilizadas para a implementação em *hardware* do AGc bem como a arquitetura proposta que faz uso destas ferramentas.

5.2.1 FPGAs

Um FPGA é um circuito integrado projetado para ser programado após o sua fabricação (Kilts, 2007). A programação de um FPGA é usualmente feita utilizando-se uma linguagem de descrição de *hardware*, dentre as quais pode-se citar VHDL (Chu, 2006) e Verilog. FPGAs podem ser usados para implementar qualquer função lógica que poderia ser implementada por um

circuito integrado de aplicação específica. A possibilidade de reconfigurar um FPGA em contraste com os custos de alteração de um circuito *Application Specific Integrated Circuit* (ASIC) oferece vantagens para muitas aplicações, especialmente onde existe uma baixa demanda por unidades de circuitos lógicos, não justificando assim a utilização de um ASIC. A Figura 5.2 apresenta a estrutura de um FPGA.

Figura 5.2: Estrutura de um FPGA

FPGAs são compostos por componentes de lógica programável chamados "blocos lógicos", e uma hierarquia de interconexões reconfiguráveis que permitem os blocos se ligarem uns aos outros. Além disso, existem pinos de I/O que podem ser configurados por meio das interconexões reconfiguráveis de forma acessar sinais de entrada e saída externos. Os blocos lógicos podem ser configurados para implementar de funções combinacionais complexas a funções lógicas simples como "E" e "OU" lógicos. Em muitos FPGAs, também são incluídos elementos de memórias que podem ser simples *Flip-Flops* ou blocos de memórias mais complexas.

5.2.2 Considerações para a Implementação de AGs em FPGA

A utilização de linguagens de alto nível em conjunto com abordagem que utiliza lógica combinacional e máquinas de estado permitem a implementação dos algoritmos genéticos em FPGA, possibilitando que estes sejam executados de maneira mais eficiente do que computadores com arquitetura de processadores convencionais, pois são executados de forma paralela e não sequencial. No entanto, dado as limitações impostas pelas funcionalidades dos blocos lógicos presentes nos FPGA, bem como em sua quantidade disponível, algumas considerações devem ser feitas para que ocorra a implementação de AGs em FPGA de forma bem sucedida. Para a

implementação do AGc, este algoritmo foi implementado através de uma máquina de estados finitos. Além disso, foram feitas outras importantes considerações para a utilização do AGc para o problema de estimação fasoriais tais como a geração de sequências pseudo aleatórias, implementação da função trigonométrica seno e simplificações para avaliação da função objetivo. Os procedimentos adotados para a implementação do AGc estão descritas a seguir.

Geração de Sequências Pseudoaleatórias

A utilização de sequências aleatórias é essencial ao funcionamento dos AGs, no caso estudados, no processo de criação de novos indivíduos pelo modelo de probabilidade de distribuição (AGc). As implementações computacionais de AGs geralmente fazem uso de uma função disponibilizada pela biblioteca padrão de programação, que gera uma sequência pseudoaleatória através de métodos numéricos. Entretanto, esta função não pôde ser utilizada devido as limitações dos FPGAs citadas anteriormente. Uma alternativa é a utilização de uma tabela com uma sequência pseudoaleatória grande o suficiente para permitir o funcionamento dos algoritmos genéticos. No entanto, esta estratégia demanda a utilização de grande número de blocos lógicos. A solução utilizada para suprir este problema é a utilização de uma estrutura combinacional conhecida como *linear feedback shift register* (LFSR) que gera de forma on-line uma sequência de números pseudoaleatória. Tal solução foi utilizada devido a sua simplicidade e não necessidade de uma sequência de números muito grande. A estrutura de funcionamento do LFSR é descrita na Figura 5.3.

Figura 5.3: *Linear Feedback Shift Register*

A cada utilização do LFSR ocorre um deslocamento dos bits para a esquerda e o bit que será posicionado na posição 1 será o resultante da lógica combinacional composta por postas XOR (ou exclusivo). Esta estratégia composta de deslocamento de bits e realimentação pela lógica combinacional permite a criação de uma sequência pseudoaleatória de tamanho 2^N em que N é o tamanho da estrutura de bits do LFSR. Para a implementação deste trabalho foram utilizados LFSR de tamanho 16.

Implementação da Função Trigonométrica Seno

A função trigonométrica seno é essencial a avaliação da função objetivo do problema de estimação fasorial, como descrito na Equação 3.2. Funções trigonométricas são implementadas por bibliotecas computacionais pela expansão destas em séries de Taylor. Entretanto, novamente, devido as limitações do FPGAs, não é possível a utilização desta abordagem. Outra abordagem conhecida no meio digital, onde têm-se restrições de uso de lógica e poder de processamento é a utilização de um algoritmo conhecido como *Coordinate Rotation Digital Computer* (CORDIC) (Altera, 2011). No entanto, a utilização deste método para o problema abordado é inviável devido ao excessivo número de avaliações da função objetivo pelo fato de cada avaliação do método tomar alguns ciclos de *clock*. A solução empregada foi a utilização de uma *Lookup table* ou vetor de associação que contém os valores da função seno discretizados. Tal abordagem permite a avaliação da função objetivo sem restrições de tempo.

Forma de Avaliação da Função Objetivo

A avaliação da função objetivo envolve uma série de operações matemáticas que são custosas. Para que a avaliação da função objetivo em *hardware* fosse possível foi necessário realizar algumas simplificações e considerações. Considerando-se na Equação 3.2, o termo que está dentro da função seno:

$$2\pi f k T + \theta = W + \theta \quad (5.1)$$

Para a implementação em *hardware*, foi considerado o cromossomo responsável por calcular a frequência como W , ao invés de simplesmente f , como na implementação computacional. Desta forma, esse cromossomo irá representar um ângulo e não a frequência em si. Dessa forma é possível mapear o intervalo $[0 : 2\pi]$ dentro do intervalo inteiro $[0 : 2^{16} - 1]$. Outra simplificação foi considerar o cromossomo com $k = 8$. Tal estratégia facilita o cálculo dos vários termos da função objetivo, como observado a seguir.

$$\begin{aligned} W &= 2\pi f 8T \\ W/2 &= 2\pi f 4T \\ W/4 &= 2\pi f 2T \\ W/8 &= 2\pi f T \\ W/8 + W/4 &= 2\pi f 3T \\ W/8 + W/2 &= 2\pi f 5T \\ W/4 + W/2 &= 2\pi f 6T \\ W/8 + W/4 + W/2 &= 2\pi f 7T \\ &\dots \end{aligned} \tag{5.2}$$

As operações descritas a seguir podem ser realizadas por simples rotacionamentos de bits e somas, facilitando assim a avaliação da função objetivo pois os termos para todos os ks poderão ser calculados.

Maquina de Estados do Algoritmo Genético compacto

O AGc foi implementado em FPGA por meio de uma modelagem de uma máquina de estados finitos, como observado pela Figura 5.4. Em cada um dos estados um procedimento do AGc é efetuado. No estado de início o vetor de probabilidades é iniciado e em seguida no estado de geração dois indivíduos são gerados com base no vetor de probabilidades. Logo após, os dois indivíduos são avaliados com base na função de avaliação. Em seguida, os indivíduos são comparados no estado de comparação. Por fim, o vetor de probabilidades é atualizado. Este processo é iterativo e o estado de resultados, onde os resultados são publicados em um barramento de dados só é alcançado quando o número de gerações for maior ou igual ao número máximo de gerações. O estado final da máquina de estados é o estado fim, em que a máquina de estados fica permanentemente até ser reiniciada.

É importante ressaltar que o processo de geração dos indivíduos e avaliação da função objetivo é paralelo para os dois indivíduos.

Figura 5.4: Máquina de estados utilizada para modelar o AGc

5.2.3 Arquitetura proposta para teste

A implementação prática da arquitetura aqui descrita se utilizou de uma FPGA Stratix III da fabricante Altera. Foi utilizado o kit de desenvolvimento "*Stratix III FPGA Development Kit*". O FPGA utilizado, Stratix III possui as seguintes características:

- 142.500 equivalentes de elementos lógicos;
- 744 pinos de I/O configuráveis;
- 384 multiplicadores de 18X18.

A foto do kit utilizado pode ser observado na Figura 5.5. O kit dispõe de diversas interfaces de entrada e saída, dentre as quais ressalta-se a porta *ethernet*, que permite que o kit seja conectado a uma rede de computadores.

A codificação do AGc em VHDL quando compilada pelo *software* Quartus II da Altera, irá gerar descrição de *hardware* que pode ser gravada no *chip* de FPGA. Entretanto, é necessário uma arquitetura de apoio para que dados de entrada possam ser enviados para a lógica sintetizada relativa ao AGc e os resultados de sua análise obtidos. Para suprir essa necessidade, foi utilizado a solução da Altera de *System on a Programmable Chip*, ou Sistema em um *Chip* Programável. Tal solução dispõe de um processador, o *softcore* NIOS II, em um barramento de dados que pode ser compartilhado com diversos outros dispositivos. O processador NIOS II é chamado *softcore* pois é descrito em linguagem de descrição de *hardware* e logo após gravado. Dentre os dispositivos que podem ser alocados no barramento de dados estão: controladores de rede, controladores de LCD, controladores de memória e lógicas customizadas.

Figura 5.5: Kit de Desenvolvimento Stratix III utilizado

Desta forma, foi criado um sistema onde o AGc implementado em VHDL foi alocado no barramento de dados do processador NIOS II, como descrito na Figura 5.6. Além do CPU e do AGc também foi alocados controladores de rede, controladores de memória, controladores de rede, dentre outros elementos. Tal arquitetura possibilita o suporte deste *hardware* a executar um sistema operacional que pode executar programas, flexibilizando a comunicação com o *hardware* desenvolvido.

Figura 5.6: Arquitetura implementada para testes

O sistema operacional uC/COS-II foi utilizado no sistema de *hardware* desenvolvido. Neste passo, é utilizada uma arquitetura cliente servidor, em que o cliente é o kit de desenvolvimento e o servidor é um computador pessoal. A descrição desta arquitetura pode ser vislumbrada na Figura 5.7.

No lado do servidor, foi desenvolvido um programa escrito na linguagem C que acessa o barramento de dados e se comunica com o *hardware* relativo ao AGc. Quando este programa é executado, os dados a serem processados pelo *hardware* do AGc chegam por rede advindas do cliente, através de uma conexão *TCP/IP* via *sockets*. Os dados são processados e logo após os resultados são enviados ao cliente.

No lado do cliente, os dados são enviados continuamente e os resultados são gravados em arquivos para posterior análise.

Figura 5.7: Comunicação com o kit de desenvolvimento

Capítulo 6

Algoritmos Aplicados à Estimação Fasorial - Resultados

Este capítulo apresenta os resultados da aplicação dos algoritmos propostos e implementados computacionalmente, tendo como entrada dados sintéticos e provenientes de simulações de diversas situações no *software* ATP. Também são apresentados resultados para a implementação do AGc em *hardware* dedicado utilizando FPGAs . Inicialmente os parâmetros utilizados em cada um dos algoritmos é apresentado de forma que seja possível replicar os resultados obtidos. Os resultados são apresentados na formas de figuras comparativas com relação as estimações de magnitude e frequência bem como a métrica TVE. Por fim, o tempo de recuperação de cada um dos algoritmos é apresentado na forma de tabelas, evidenciando as qualidades de cada um dos algoritmos.

6.1 Os Parâmetros Utilizados

A seguir são apresentados os parâmetros utilizados em cada um dos algoritmos implementados. Os parâmetros foram escolhidos empiricamente, como foi o caso dos algoritmos genético. Para o caso dos outros algoritmos (TDF e PLL), os parâmetros foram escolhidos com base na literatura existente.

6.1.1 O Algoritmo Genético

Na utilização do AG, deve-se escolher os parâmetros genéticos de taxa de *crossover* e taxa de mutação, que representam respectivamente a probabilidade de ocorrer cruzamento entre dois indivíduos e de estes dois indivíduos sofrerem mutação. Além disso, deve-se fixar os valores iniciais de frequência, amplitude e fase, possibilitando restringir a área de busca do AG. A precisão do algoritmo é influenciada pelo número de indivíduos na população e pelo número máximo de gerações. Os parâmetros utilizados podem ser observados na Tabela 6.1

Tabela 6.1: Parâmetros utilizados no Algoritmo Genético

Parâmetro	Valor
Taxa de <i>Crossover</i>	80%
Taxa de Mutação	2 %
Inicialização Frequência	[59,5:60,5] Hz
Inicialização Amplitude	[0,9:1,1] p.u.
Inicialização Fase	[0:2 π]
Tamanho População	60
Número máximo de gerações	100

Observando os dados da tabela 6.1, pode-se constatar que o número máximo de avaliações para cada estimativa fasorial é de 6.000 avaliações da função objetivo. Foi constatado que um aumento do número de avaliações da função objetivo pelo aumento da população ou pelo aumento do número máximo de gerações não proveu uma melhora significativa nas estimações, sendo os valores apresentados os mais próximos do adequado.

6.1.2 O Algoritmo Genético compacto

Para o Algoritmo Genético compacto (AGc), os parâmetros que devem ser definidos para o algoritmo funcionar são os seguintes: tamanho da população, número máximo de gerações, número de *bits* utilizados para a codificação de frequência, amplitude e fase além de um intervalo inicial onde os genes são mapeados. Os parâmetros utilizados podem ser observados na Tabela 6.2.

É importante lembrar que o tamanho da população no AGc controla a velocidade com que o algoritmo irá convergir. O número de avaliações da função objetivo foi de 6.000, visto que o número máximo de gerações é 3000 e a cada geração são gerados 2 indivíduos.

Tabela 6.2: Parâmetros utilizados Algoritmo Genético Compacto

Parâmetro	Valor
Tamanho da População	600
Número Máximo de Gerações	3000
Bits de Codificação de Frequência	24
Bits de Codificação de Amplitude	24
Bits de Codificação de Fase	24
Frequência Inicial	[59,9:60,1]
Amplitude Inicial	[0,9:1,1]
Fase Inicial	[0:2 π]

6.1.3 A Transformada Discreta de Fourier

O único parâmetro utilizado na configuração do algoritmo que utiliza TDF é o deslocamento M entre as janelas de dados de tamanho N , utilizadas para estimativa fasoriais, como discutido no Capítulo 3. Para este trabalho, o deslocamento utilizado foi de um quarto do tamanho da janela de dados. Como exemplo, pode-se tomar uma janela dados de 32 amostras. O deslocamento utilizado nesse caso seria de 8 amostras entre as janelas.

6.1.4 *Phasor-Locked Loop*

Os parâmetros que são necessários para iniciar o algoritmo PLL foram obtidos de acordo com Karimi-Ghartemani, Karimi e Iravani (2004). As estimativas iniciais de frequência, amplitude, e ângulo de fase foram definidas de acordo com o sinal de entrada. Todos os valores dos parâmetros utilizados podem ser observados na Tabela 6.3.

Tabela 6.3: Parâmetros utilizados no algoritmo baseado em *Phase-Locked Loop*

Parâmetro	Valor
μ_1	200
μ_2	12000
μ_3	200
f_0	60 Hz
A_0	1 p.u.
ϕ_0	0

6.2 Resultados Utilizando Dados Sintéticos como Entrada

A seguir são apresentados os resultados para os algoritmos aplicados sobre os dados sintéticos descritos anteriormente. Inicialmente é avaliado a influencia da taxa de amostragem sobre

a precisão das estimações dos algoritmos, pela aplicação de sinais em regime com diferentes relações de sinal ruído *signal noise ratio* (SNR) para as taxas de 32 e 128 amostras por ciclo da onda fundamental. Os sinais de entrada tem duração de cerca de 1s. A métrica TVE é utilizada para avaliar a precisão das estimações. Após, os três testes de performance descritos na norma C37.118 utilizando-se diferentes SNRs (40dB, 60dB e 80dB) são aplicados como entrada para os algoritmos. Gráficos contendo o TVE, estimativa de frequência e estimativa de magnitude são apresentados para cada teste de performance e para cada SNR. Ao fim, será apresentado uma tabela onde estão dispostos os tempos de recuperação de cada algoritmo, isto é, o tempo para que, depois da ocorrência do evento transitório, o algoritmo volte a apresentar estimativas fasoriais com erro abaixo de 1%.

6.2.1 Influência do Ruído e da Taxa de Amostragem

A seguir são apresentados as precisões de cada um dos algoritmos mediante a aplicação de sinais de entrada contendo situações de regime permanente de duração de 1s com SNR de diferentes intensidades. Além disso, para verificar a influência da taxa de amostragem sobre a precisão dos algoritmos, os algoritmos são testados com sinais de entrada de 32 e 128 amostras por ciclo respectivamente.

Figura 6.1: Precisão dos algoritmos para a taxa amostral 32 amostras por ciclo.

Observando-se as Figuras 6.1 e 6.2, é possível observar que, como esperado, quanto menor o SNR, maior é o erro nas estimativas dos algoritmos. Para o caso de 20 dB, nenhum dos algoritmos foi capaz de prover um TVE inferior a 1%, em ambas as taxas de amostragem. Para os demais SNRs, 40 dB, 60 dB e 80 dB, foram obtidos TVEs inferiores a 1 %. Por esse motivo, nos

Figura 6.2: Precisão dos algoritmos para taxa amostral de 128 amostras por ciclo.

próximos testes não serão utilizados sinais com SNR inferior a 40 dB. Pode-se observar também que não houve grandes variações nas precisões dos algoritmos se levado em conta as duas taxas de amostragem testadas. Desta forma será fixado a taxa de amostragem de 32 amostras por ciclo para os próximos testes. Os algoritmos baseados na TDF e PLL tiveram uma melhor precisão para todos os sinais de entrada, no entanto, os AGs tiveram um desempenho muito semelhante.

6.2.2 Teste de Degrau de Amplitude

A seguir são avaliados os algoritmos de estimação pela aplicação de um degraude magnitude de 10%, conforme especificado pela equação 4.1. Os sinais utilizados tem SNR de 40, 60 e 80 dB. A precisão dos algoritmos é avaliada pela utilização do TVE, estimação de frequência e estimação de magnitude.

SNR 40

As Figuras 6.3, 6.4 e 6.5 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degraude amplitude de 10% com SNR de 40 dB.

Pode ser observado que todos os algoritmos conseguiram prover uma estimativa confiável após a ocorrência do evento de degraude amplitude, sendo que os algoritmos genéticos conseguiram ter um valor inferior a 1% de TVE mais rapidamente que os outros algoritmos, tendo o desempenho parecido entre estes. O algoritmo baseado na TDF teve o terceiro melhor desempenho, seguido pelo algoritmo baseado em PLL. É possível observar que a variação de

Figura 6.3: TVE para o caso de teste de degrau de amplitude com SNR 40.

Figura 6.4: Estimação de frequência para o caso de teste de degrau de amplitude com SNR 40.

Figura 6.5: Estimação de magnitude para o caso de teste de degrau de amplitude com SNR 40.

amplitude tem uma pequena influência sobre as estimativas de frequência dos algoritmos, sendo mais pronunciada na estimativa de frequência do algoritmo baseado em TDF.

SNR 60

As figuras 6.6, 6.7 e 6.8 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de amplitude de 10% com um SNR de 60 dB.

O mesmo comportamento dos algoritmos foi observado nos testes descritos nas Figuras 6.6, 6.6 e 6.6, que no teste anterior de SNR 40. Os algoritmos genéticos foram os mais rápidos a atingir o valor de 1% de TVE, seguidos pela TDF e PLL. Novamente foi observado que, apesar da amplitude ser o único parâmetro da forma de onda modificado durante o degrau, foi observado desvios da estimativa de frequência do algoritmo com relação a frequência nominal, sendo mais pronunciada no algoritmo baseado na TDF.

Figura 6.6: TVE para o caso de teste de degrau de amplitude com SNR 60.

Figura 6.7: Estimação de frequência para o caso de teste de degrau de amplitude com SNR 60.

Figura 6.8: Estimação de magnitude para o caso de teste de degrau de amplitude com SNR 60.

SNR 80

As figuras 6.9, 6.10 e 6.11 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de amplitude de 10% com um SNR de 80 dB.

O resultado observado nas Figuras 6.6, 6.6 e 6.6 foi semelhante aos testes obtidos para os casos com SNR de 40 e 60 dB. Observa-se que, devido à menor intensidade de ruídos, os algoritmos conseguem ter uma resposta menos ruidosa. Novamente, foi observado, principalmente na estimativa de frequência do algoritmo baseado na TDF uma influência do degrau de magnitude sobre a estimativa de frequência.

Figura 6.9: TVE para o caso de teste de degrau de amplitude com SNR 80.

Figura 6.10: Estimação de frequência para o caso de teste de degrau de amplitude com SNR 80.

Figura 6.11: Estimação de magnitude para o caso de teste de degrau de amplitude com SNR 80.

6.2.3 Teste de Degrau de Fase

A seguir são avaliados os algoritmos de estimação pela aplicação de um degraú de fase de 90° , conforme especificado pela equação 4.2. Os sinais utilizados tem SNR de 40, 60 e 80 dB. A precisão dos algoritmos é avaliada pela utilização do TVE, estimação de frequência e estimação de magnitude.

SNR 40

As figuras 6.12, 6.13 e 6.14 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degraú de fase de 90° com um SNR de 40 dB.

Figura 6.12: TVE para o caso de teste de degraú de fase com SNR 40.

Para o caso de degraú de fase com SNR 40 dB, o desempenho dos algoritmos foi o seguinte: ambos os algoritmos genéticos apresentaram um desempenho semelhante, sendo o melhor desempenho dentre os algoritmos. Eles apresentaram o menor tempo para que a estimativa estivesse adequada ($TVE < 1\%$). Novamente, o algoritmo baseado em TDF obteve o terceiro melhor desempenho seguido pelo algoritmo baseado em PLL. Apesar de a fase ser o único parâmetro que foi modificado no sinal de entrada durante o degraú, foi observado a influência deste evento sobre as estimativas de amplitude e frequência, que desviaram sobre seus valores nominais em

Figura 6.13: Estimação de frequência para o caso de teste de degrau de fase com SNR 40.

Figura 6.14: Estimação de magnitude para o caso de teste de degrau de fase com SNR 40.

todos algoritmos.

SNR 60

As figuras 6.15, 6.16 e 6.17 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de fase de 90° com um SNR de 60 dB.

Figura 6.15: TVE para o caso de teste de degrau de fase com SNR 60.

De forma semelhante ao teste de degrau de fase com SNR de 40 dB, no teste com SNR de 60 dB, todos algoritmos conseguiram apresentar uma estimativa adequada depois da decorrência do evento. Os algoritmos genéticos obtiveram o melhor desempenho dentre os algoritmos testados, com desempenho semelhante dentre si. O algoritmo baseado na TDF obteve o segundo melhor desempenho enquanto o algoritmo baseado no filtro PLL apresentou o maior tempo de recuperação. Novamente, foi observado a influência do degrau de fase sobre as estimativas de magnitude e frequência em todos algoritmos.

SNR 80

As figuras 6.18, 6.19 e 6.20 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de fase de 90° com um SNR de 80 dB.

Figura 6.16: Estimação de frequência para o caso de teste de degrau de fase com SNR 60.

Figura 6.17: Estimação de magnitude para o caso de teste de degrau de fase com SNR 60.

Figura 6.18: TVE para o caso de teste de degrau de fase com SNR 80.

Figura 6.19: Estimação de frequência para o caso de teste de degrau de fase com SNR 80.

Figura 6.20: Estimação de magnitude para o caso de teste de degrau de fase com SNR 80.

No caso do degrau de fase com SNR de 80 dB, os resultados foram similares aos já encontrados. Novamente os algoritmos genéticos apresentaram desempenho semelhante e o melhor dentre os algoritmos estudados para este caso. O algoritmo baseado na TDF obteve o terceiro melhor resultado enquanto o algoritmo baseado em PLL apresentou o maior tempo para apresentar uma estimativa adequada. Mais uma vez foi observado a influência do degrau de fase sobre as estimativas de magnitude e frequência de todos algoritmos.

6.2.4 Teste de Degrau de Frequência

A seguir são avaliados os algoritmos de estimação pela aplicação de um degraú de frequência de 5 Hz, conforme especificado pela equação 4.3. São adicionados ruídos de magnitudes de 20, 40, 60 e 80 dB a este teste a fim de se avaliar a influência ao ruído. A precisão dos algoritmos é avaliada pela utilização do TVE, estimação de frequência e estimação de magnitude.

SNR 40

As figuras 6.21, 6.22 e 6.23 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degraú de frequência de 5 Hz com um SNR de 40 dB.

Figura 6.21: TVE para o caso de teste de degraú de frequência com SNR 40.

Para o caso de degraú de frequência com SNR 40 dB, o desempenho dos algoritmos foi o seguinte: ambos os algoritmos genéticos apresentaram um desempenho semelhante, sendo o melhor desempenho dentre os algoritmos. Eles apresentaram o menor tempo para que a estimativa estivesse adequada ($TVE < 1\%$). O algoritmo baseado em PLL obteve o terceiro melhor desempenho. O algoritmo baseado na TDF não foi capaz de apresentar estimativas adequadas após a decorrência do degraú de frequência, fato explicado pela grande excursão da frequência do sinal com relação a frequência nominal, causando vazamento espectral. Como nos outros

Figura 6.22: Estimação de frequência para o caso de teste de degrau de frequência com SNR 40.

Figura 6.23: Estimação de magnitude para o caso de teste de degrau de frequência com SNR 40.

casos, a alteração de um único parâmetro, neste caso a frequência, provocou alterações sobre as estimativas de magnitude do algoritmo baseado em PLL.

SNR 60

As figuras 6.24, 6.25 e 6.26 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de frequência de 5 Hz com um SNR de 60 dB.

Figura 6.24: TVE para o caso de teste de degrau de frequência com SNR 60.

Assim como para o caso de degrau de frequência com SNR de 40 dB, o algoritmo baseado na TDF não foi capaz de apresentar estimativas confiáveis após a decorrência do evento. Os algoritmos genéticos obtiveram o melhor desempenho, tendo uma recuperação para um TVE inferior à 1 % em menos tempo que os outros algoritmos e desempenho semelhante entre si. O degrau de frequência teve uma pequena influência sobre a estimativa de magnitude do algoritmo PLL.

SNR 80

As figuras 6.27, 6.28 e 6.29 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de frequência de 5 Hz com um SNR de 80 dB.

Figura 6.25: Estimação de frequência para o caso de teste de degrau de frequência com SNR 60.

Figura 6.26: Estimação de magnitude para o caso de teste de degrau de frequência com SNR 60.

Figura 6.27: TVE para o caso de teste de degrau de frequência com SNR 80.

Figura 6.28: Estimação de frequência para o caso de teste de degrau de frequência com SNR 80.

Figura 6.29: Estimação de magnitude para o caso de teste de degrau de frequência com SNR 80.

Mais uma vez, para o caso de degrau de frequência com SNR de 80 dB, o algoritmo baseado em TDF não foi capaz de apresentar uma estimativa adequada depois da decorrência do evento. Os algoritmos genéticos obtiveram novamente o melhor desempenho, tendo o menor tempo de recuperação do evento, seguido pelo algoritmo baseado em PLL. Também foi notado uma influência do degrau de frequência sobre a estimativa de magnitude do algoritmo PLL

6.2.5 Tempo de recuperação

A fim de se quantificar o tempo de convergência dos algoritmos testados, a seguir serão apresentados os tempos de convergência destes para cada um dos casos de teste, variando-se a relação sinal ruído SNR. O tempo de convergência dos algoritmos, já denominado tempo de recuperação é o tempo necessário para que, depois da ocorrência do evento transitório, os algoritmos voltem a apresentar estimativas favoráveis com TVE inferior a 1%.

Observando-se a Tabela 6.4, é possível analisar o desempenho dos algoritmos frente aos eventos transitórios testados. Nota-se que em todos os algoritmos que existe uma relação entre a intensidade do ruído no sinal de entrada e o tempo de recuperação. Quanto maior intensidade

Tabela 6.4: Tempo de recuperação dos algoritmos implementados computacionalmente

Caso de Teste	Algoritmo	Relação Sinal Ruído (dB)		
		40	60	80
Degrau de Magnitude	AG	0,011979 s	0,000521	0,000000
	AGc	0,011458 s	0,001042	0,001042
	PLL	0,018750 s	0,025000	0,025000
	TDF	0,018750 s	0,018229	0,018229
Degrau de Fase	AG	0,013020	0,010417	0,010417
	AGc	0,013020	0,010417	0,010417
	PLL	0,101562	0,101562	0,101562
	TDF	0,020833	0,020833	0,020833
Degrau de Frequência	AG	0,009896	0,008333	0,008333
	AGc	0,009896	0,008333	0,008333
	PLL	0,066146	0,075000	0,075521
	TDF	—	—	—

do ruído, maior o tempo de recuperação. Também é possível observar que o teste de degrau de amplitude tem a menor influência sobre o tempo de recuperação.

Dentre os algoritmos testados, os AGs tiveram o menor tempo de recuperação para todos os casos testados e os desempenho dos dois tipos destes algoritmos foi muito semelhantes. O maior tempo de recuperação do AG foi de 0,013020 s para o caso de degrau de fase com SNR 20 dB e o menor foi o caso de degrau de magnitude com SNR 80 dB, obtendo um tempo de recuperação 0. O maior e menor tempos de recuperação do AGc ocorreu nos mesmos casos de teste do AG, com tempos de 0,013020 s e 0,001042 s respectivamente.

O algoritmo baseado na TDF teve o segundo melhor desempenho nos casos testados. Entretanto, para o caso do teste de degrau de frequência, o algoritmo não convergiu após a decorrência do evento transitório. O motivo para que não haja a convergência é o fato de a transformada discreta de Fourier estar sofrendo uma grande excursão quanto a frequência fundamental do sinal de entrada, provocando um efeito de vazamento espectral (Oppenheim, Willsky e Nawab, 1983). Mesmo efetuando as correções propostas por Wang e Sun (2004, 2006), não foi possível suprir este efeito. O maior tempo de recuperação deste algoritmo foi para o caso de degrau de frequência, onde não conseguiu apresentar estimativas fasoriais com TVE inferior a 1%. O menor tempo de recuperação foi para o caso de degrau de magnitude com 0,018229 s.

O algoritmo baseado em PLL teve os maiores tempo de recuperação dentre os algoritmos testados. O maior tempo de recuperação foi de 0,101562 para o caso de degrau de fase, e o menor tempo de recuperação foi de 0,025000 s para o caso de degrau de magnitude.

6.3 Resultados Utilizando dados Simulados no *Software* ATP

A seguir são apresentados resultados da aplicação de dados simulados no *software* ATP nos algoritmos implementados. A frequência estimada é utilizada como forma de qualificar os resultados dado que o ATP provê uma referência de frequência que é a velocidade de rotação dos geradores. Para comparação com a magnitude estimada, será utilizado o cálculo do valor *Root Mean Square* (RMS).

6.3.1 Ilhamento

Nas Figuras 6.30 e 6.32 é possível observar as estimações de frequência dos algoritmos para o caso de Ilhamento do sistema, descrito na Figura 4.4. Este caso se caracteriza pela abertura da linha de transmissão LT13, formando dois subsistemas isolados, sendo o subsistema 1 alimentado pelos geradores GER11, GER12, GER13 e GER21 e o subsistema 3 alimentado pelo gerador GER31. Foram utilizados como ponto de medida os barramentos BINT1 e BINT3. Nas Figuras 6.31 e 6.33 observa-se a estimativa de amplitude para os mesmos casos.

Figura 6.30: Frequência estimada pelos algoritmos no subsistema 1.

Observando-se as Figuras 6.30 e 6.32 pode-se notar que os algoritmos se mantiveram fiéis á

Figura 6.31: Amplitude estimada pelos algoritmos no subsistema 1.

Figura 6.32: Frequência estimada pelos algoritmos no subsistema 3.

Figura 6.33: Amplitude estimada pelos algoritmos no subsistema 3.

referência provida pelo *software* ATP, não sendo possível discriminar qual algoritmo obteve a melhor estimativa. No evento observado, nota-se que no subsistema 1, houve um aumento da frequência logo após a ocorrência do evento, em função da excesso de geração provocado pela situação de ilhamento. No subsistema 3, houve um déficit de geração, evidenciado pela queda de frequência.

6.3.2 Entrada de Carga

As Figuras 6.34 e 6.35 apresentam respectivamente as estimações de frequência e amplitude dos algoritmos para a situação simulada de entrada de uma carga de 450 MVA no barramento de cargas BCAR1A, descrito na Figura 4.4.

Pode-se observar pela queda de frequência que com a entrada de carga, houve um desbalanço entre a relação geração-consumo. Entretanto, com a atuação dos reguladores de velocidade e de tensão, os valores de frequência e magnitude dos sinais retornaram a valores próximos do nominal.

Figura 6.34: Frequência estimada pelos algoritmos durante a situação de entrada de carga.

Figura 6.35: Amplitude estimada pelos algoritmos durante a situação de entrada de carga.

6.3.3 Situação de Falta Elétrica Sustentada na Linha de Transmissão

A Figura 6.36 e 6.37 apresentam respectivamente os resultados da aplicação de dados simulados de uma situação em que ocorreu uma falta franca sustentada a 50% da linha de transmissão LT1 no sistema descrito pela Figura 4.4.

Figura 6.36: Frequência estimada pelos algoritmos no durante a situação de falta.

A ocorrência da falta foi a situação mais severa aplicada sobre os algoritmos, pois as estimções de frequência apresentaram mais ruído do que os casos de teste anteriores. O algoritmo que foi menos influenciado pela situação de falta foi o algoritmo de PLL.

Figura 6.37: Amplitude estimada pelos algoritmos no durante a situação de falta.

6.4 Resultados utilizando FPGAs para implementação do AGc

Esta seção irá apresentar os resultados dos dados sintéticos e simulados no *software* ATP quando analisados por meio da implementação do AGc em FPGA.

6.4.1 Resultados para dados Sintéticos

Para a análise dos AGs implementados em FPGA foram utilizados os sinais sintéticos descritos anteriormente com um SNR de 60 dB, como apresentados a seguir. Podem ser observados para cada um dos testes (degrau de magnitude, fase e frequência) os respectivos TVE, estimação de magnitude e de frequência.

Degrau de Magnitude

As Figuras 6.38, 6.39 e 6.40 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de magnitude de 10% com um SNR de 60 dB.

Figura 6.38: TVE para o caso de teste de degrau de magnitude com SNR 60.

Pode-se notar que o algoritmo, depois da ocorrência do evento, apresentou precisão de medidas aceitáveis, com TVE inferior a 1%.

Degrau de Fase

As Figuras 6.41, 6.42 e 6.43 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de fase de 90° com um SNR de 60 dB.

Figura 6.39: Estimação de frequência para o caso de teste de degrau de magnitude com SNR 60.

Figura 6.40: Estimação de Magnitude para o caso de teste de degrau magnitude com SNR 60.

Figura 6.41: TVE para o caso de teste de degrau de fase com SNR 60.

Figura 6.42: Estimação de frequência para o caso de teste de degrau de fase com SNR 60.

Figura 6.43: Estimação de magnitude para o caso de teste de degrau de fase com SNR 60.

Pode-se notar que o algoritmo, depois da ocorrência do evento, apresentou precisão de medidas aceitáveis, com TVE inferior a 1%.

Degrau de Frequência

As Figuras 6.44, 6.45 e 6.43 apresentam o TVE, estimação de frequência e estimação de magnitude para o teste de degrau de frequência de 60 Hz com um SNR de 60 dB.

Pode-se notar que o algoritmo, depois da ocorrência do evento, apresentou precisão de medidas aceitáveis, com TVE inferior a 1%.

Figura 6.44: TVE para o caso de teste de degrau de frequência com SNR 60.

Figura 6.45: Estimação de frequência para o caso de teste de degrau de frequência com SNR 60.

Figura 6.46: Estimação de magnitude para o caso de teste de degrau de frequência com SNR 60.

Tempo de recuperação

A fim de se quantificar o tempo de convergência dos algoritmos testados, a seguir são apresentados os tempos de convergência do algoritmo AGc implementado em FPGA para cada um dos casos testados.

Tabela 6.5: Tempo de recuperação do AGc implementado em FPGA

Caso de Teste	Relação Sinal Ruído (dB)
	60
Degrau de Magnitude	0,008 s
Degrau de Fase	0,007 s
Degrau de Frequência	0,06 s

O tempos descritos na Tabela 6.5 resumizam os resultados encontrados pela aplicação AGc implementado em FPGA. Comparando-se as Tabelas 6.5 e 6.4 pode-se notar que os tempos da implementação computacional e foram semelhantes, da mesma ordem de grandeza.

6.4.2 Resultados para dados simulados no *software* ATP

A seguir são apresentados resultados da aplicação de dados simulados no *software* ATP. A frequência estimada é utilizada como forma de qualificar os resultados dado que o ATP provê uma referência de frequência que é a velocidade de rotação dos geradores.

Ilhamento

Nas Figuras 6.47 e 6.49 é possível observar as estimações de frequência dos algoritmos para o caso de Ilhamento do sistema. Foram utilizados como ponto de medida os barramentos BINT1 e BINT3, visíveis na Figura 4.4, como descrito anteriormente. Nas Figuras 6.48 e 6.50 observa-se a estimacão de amplitude para os mesmos casos.

De forma similar a que foi observado anteriormente, nas Figuras 6.30 e 6.32 pode-se notar que os algoritmos se mantiveram fiéis á referência provida pelo *software* ATP, não sendo possível discriminar qual algoritmo obteve a melhor estimativa. No evento observado, nota-se que no subsistema 1, houve um aumento da frequência logo após a decorrência do evento, em função da excesso de geração provocado pela situação de ilhamento. No subsistema 3, houve um déficit de geração, evidenciado pela queda de frequência.

Figura 6.47: Frequência estimada no subsistema 1 pelo AGc implementado em FPGA.

Figura 6.48: Amplitude estimada no subsistema 1 pelo AGc implementado em FPGA.

Figura 6.49: Frequência estimada no subsistema 3 pelo AGc implementado em FPGA.

Figura 6.50: Amplitude estimada no subsistema 3 pelo AGc implementado em FPGA.

Entrada de Carga

As Figuras 6.51 e 6.52 apresentam respectivamente as estimções de frequência e amplitude dos algoritmos para a situação simulada de entrada de uma carga de 450 MVA no barramento de cargas BCAR1A da Figura 4.4.

Figura 6.51: Frequência estimada durante a entrada de carga pelo AGc implementado em FPGA.

Assim como foi observado pelas implementações computacionais, nota-se que, pela queda de frequência que com a entrada de carga, houve um desbalanço entre a relação geração-consumo. Entretanto, com a atuação dos reguladores de velocidade e de tensão, os valores de frequência e magnitude dos sinais retornaram a valores próximos do nominal.

Figura 6.52: Amplitude estimada durante a entrada de carga pelo AGc implementado em FPGA.

Situação de Falta Sustentada na Linha de Transmissão

A Figura 6.53 e 6.54 apresentam respectivamente os resultados da aplicação de dados simulados de uma situação em que ocorreu uma falta franca sustentada a 50% da linha de transmissão LT1 da Figura 6.50.

Figura 6.53: Frequência estimada durante a situação de falta pelo AGc implementado em FPGA.

Da mesma forma que nos algoritmos implementados computacionalmente, ocorrência da falta foi a situação mais severa aplicada sobre o algoritmo, pois a estimação de frequência apresentaram mais ruído do que os casos de teste anteriores. O algoritmo que foi menos influenciado pela situação de falta foi o algoritmo de PLL.

Figura 6.54: Amplitude estimada durante a situação de falta pelo AGc implementado em FPGA.

Capítulo 7

Conclusões

Este trabalho apresentou a utilização de uma técnica inteligente, os algoritmos genéticos, como ferramenta de estimação fasorial em sistemas elétricos de potência. A teoria básica relativa a algoritmos genéticos foi apresentada e foram introduzidos dois tipos de algoritmos: algoritmo genético simples, e algoritmo genético compacto. Além disso, foram estudados e implementados dois outros métodos para estimação fasorial: um algoritmo baseado na Transformada Discreta de Fourier e outro baseado em um filtro *phase-locked loop*.

Os algoritmos citados foram implementados computacionalmente utilizando a linguagem de programação C++. A justificativa no uso desta linguagem está no fato desta ser uma linguagem orientada objetos e que executa os algoritmos implementados com eficiência. Os algoritmos implementados foram testados com dados sintéticos, em que se tem controle sobre os parâmetros da forma de onda gerada. Também foram feitos testes com dados simulados provenientes do *software* ATP. A fim de se tirar vantagem do paralelismo inerente dos algoritmos genéticos, a implementação em *hardware* do Algoritmo Genético Compacto utilizando a linguagem VHDL em um FPGA também foi abordada. As considerações necessárias para a implementação embarcada deste tipo de algoritmo, tais como a forma de avaliação da função objetivo e geração de números pseudo-aleatórios foram apresentadas.

Para a análise dos algoritmos utilizando dados sintéticos, foram utilizados sinais de situações de regime permanente com diferentes intensidades de ruído bem como com taxas amostrais de 32 e 128 amostras por ciclo. Para testar a resposta dos algoritmos mediante a situações transitórias, foram gerados sinais de acordo com o apêndice G da norma IEEE C37.118, que descreve testes de degrau de amplitude, frequência e fase. Tais testes visaram medir o tempo de recuperação dos

algoritmos. Um sistema elétrico simulado no *software* ATP foi proposto, contendo 5 geradores de diferentes portes, a fim de se evidenciar a dinâmica dos sistemas elétricos de potência. Foram simuladas situações de faltas e de ilhamentos de porções do sistema elétrico, estas últimas feitas a fim de se testar a influência destes distúrbios sobre a estimação de frequência dos algoritmos.

Algumas vantagens e desvantagens relativas às técnicas estudadas são apresentadas a seguir:

1. O AG e AGc destacam-se por requererem menor tempo para recuperar/aproximar os parâmetros do sinal (amplitude, frequência e fase) conforme cada nova amostra é apresentada após o evento. Para o correto entendimento dessa vantagem é importante fazer clara distinção entre esse tempo de recuperação do sinal e o tempo de computação que o algoritmo requer para convergir a cada nova amostra apresentada após um evento, isto é, o tempo de convergência. Ambos os algoritmos genéticos apresentam tempo de convergência maior comparado aos outros métodos, uma vez que o processo evolutivo desses algoritmos calcula um número relativamente grande de vezes a função objetivo do problema na busca pelos parâmetros mais adequados do sinal.
2. As estimativas do AG e do AGc correspondem a valores mais realísticos que os dos outros dois métodos avaliados para os instantes logo após o evento, isto é durante o período de recuperação do sinal, o que de certa forma explica o menor tempo de recuperação. Para os instantes em que todos os métodos já estão estimando razoavelmente os parâmetros do sinal em regime, esses algoritmos genéticos apresentam saídas menos precisas que dos métodos TDF e PLL. Mesmo nesse caso, é importante observar que os valores de erro de precisão do AG e do AGc são inferiores aos recomendados por norma ($<1\%$);
3. A TDF mostra desempenho intermediário em termos de tempo de recuperação quando comparado com o algoritmo genético e o PLL. O tempo de convergência é similar ao do PLL e, como dito anteriormente, bem inferior ao AG. No entanto, a TDF apresenta limitações já conhecidas na literatura para ocorrência de variação da frequência fundamental.
4. O PLL apresenta desempenho inferior ao dos outros métodos em termos de tempo de recuperação, porém possui a vantagem de maior precisão para estimativas após a recuperação do sinal.

Observando-se os resultados dos dados sintéticos em regime aplicados como entrada aos algoritmos, foi possível observar que não existe grande diferença quanto a precisão provida pelos

algoritmos variando-se a taxa de amostragem, sendo esta então fixada em 32 pontos por ciclo. Também foi observado que os algoritmos em regime não possuem estimativas confiáveis (TVE menor que 1%) para sinais com SNR inferior a 40 dB.

Durante a aplicação de dados provenientes de simulações utilizando-se o *software* ATP dos algoritmos a, foram constatadas boas estimações de frequência mediante a ocorrência de eventos no sistema elétrico simulado. Foram testadas três situações distintas: falta do tipo fase-terra, ilhamento de uma porção do sistema e entrada de um grande bloco de carga. Em todos os casos apresentados, a referência de velocidade dos geradores foram utilizadas como comparação para a estimativa de frequência dos algoritmos. Foi possível observar que os algoritmos genéticos são adequados ao estudo da dinâmica destes eventos, pois as estimativas de frequência se mantiveram próximas às referências providas pelo ATP.

A implementação em *hardware* utilizando um FPGA do Algoritmo Genético compacto foi testada utilizando-se os mesmos dados de entrada dos algoritmos simulados computacionalmente. Os tempos de recuperação para os testes de degrau de magnitude, fase e frequência deste tipo de implementação foram semelhantes aos tempos encontrados nas simulações. O implementação embarcada também proveu uma boa estimativa de frequência quando da aplicação dos dados provenientes do sistema elétrico simulado no *software* ATP.

7.1 Sugestões para continuidade na pesquisa

Como sugestões de continuidade da pesquisa são apresentados os seguintes itens:

- Utilização de dados de entrada digitalizados a partir de um sistema de aquisição ligado a um simulador digital de sistema elétrico de tempo-real *Real-time Digital Simulator* (RTDS);
- Estudo das formas de comunicação utilizadas pelas unidades de medição fasorial, descritas na norma IEEE C37.118;
- Estudo de uma arquitetura embarcada que inclua o sistema de comunicação com as estimativas fasoriais em um único chip de FPGA, pela implementação dos algoritmos de estimação fasorial em FPGA utilizando a linguagem VHDL e a comunicação controlada pelo *softcore* Nios II.
- Efetuar um novo estudo levando em conta as normas C37.118.1-2011 e C37.118.2-2011, relativas a sincrofasores, que foram lançadas recentemente enquanto este trabalho estava

sendo desenvolvido.

7.2 Publicações Decorrentes Desta Dissertação

A seguir são evidenciados publicações que foram conseguidas em decorrência do desenvolvimento deste trabalho. Dentre as publicações estão presentes uma publicação em congresso internacional e duas publicações em periódicos, um nacional e outro internacional.

7.2.1 Publicação em Congresso Internacional

SILVA, R. P. M. ; DELBEM, A. C. B. ; COURY, D. V. . "Algoritmos Genéticos Aplicados à Estimação Fasorial em Sistemas Elétricos de Potência". Em: *IX CLAGTEE (Congresso Latino Americano de Geração e Transmissão de Energia Elétrica)*, 2011, Mar del Plata - Argentina.

7.2.2 Publicações em Periódicos

Periódico Nacional:

SILVA, R. P. M. ; DELBEM, A. C. B. ; COURY, D. V. "Técnicas Evolutivas Aplicados ao Projeto de Unidades de Medição Fasorial". *Revista Controle & Automação da Sociedade Brasileira de Automação*, 2013.

Periódico Internacional:

SILVA, R. P. M. ; DELBEM, A. C. B. ; COURY, D. V. "Genetic Algorithm Applied to Phasor Estimation and Frequency Tracking in PMU Development". *International Journal of Electrical Power & Energy Systems*, Elsevier, 2013.

Referências Bibliográficas

- Adamiak, Mark, William Premerlani e B. Kasztenny (1993). “Synchrophasors: Definition, Measurement, and Application”. Em: *Proceedings of 2nd Carnegie Mellon Conference in Electric Power Systems*. Vol. 13, pp. 57–62.
- Altera (2011). *Advanced Synthesis CookBook*.
- Amon Filho, J. e M. Pereira (1996). “ATP Alternative Transient Program: curso básico sobre utilização do ATP”. Em: *Comitê Latino Americano de Usuários do EMTP/ATP*.
- Barbosa, Daniel (2010). “Sistema Híbrido Inteligente para o Monitoramento e Proteção de Transformadores de Potência”. Tese de doutorado. Universidade de São Paulo, Escola de Engenharia de São Carlos.
- Burnett Jr, RO, MM Butts, PS Sterlina, G.P. Co e GA Atlanta (1994). “Power system applications for phasor measurement units”. Em: *IEEE Computer Applications in Power* 7.1, pp. 8–13.
- Carvalho, Janison Rodrigues (2008). “Estimação de Harmônicos/Inter-Harmônicos: uma Abordagem Multitaxa”. Diss. de mestrado. Universidade Federal de Juiz de Fora, p. 113.
- Carvalho, J.R., D.V. Coury, D.C. Jorge, A.C.B. Delbem e E.V. Simões (2009). “Algoritmos Genéticos Aplicados à Medição Fasorial e Projeto de PMUs”. Em: *CBQEE 2009*. Blumenau - SC, pp. 1–6.
- Chu, P.P. (2006). *RTL hardware design using VHDL: coding for efficiency, portability, and scalability*. Wiley-IEEE Press.
- Coury, D.V., M. Oleskovicz e R. Giovanini (2007). *Proteção Digital de Sistemas Elétricos de Potência: dos relés eletromecânicos aos microprocessados inteligentes*. 1ª ed. São Carlos: EESC-USP.

- De Almeida, L.A.L. e AC Lima (1998). “Covariance management based RLS algorithm for phasor estimation in severely noisy systems”. Em: *Power Delivery, IEEE Transactions on* 13.4, pp. 1067–1072.
- De Macedo, RA, D. da Silva Filho, D.V. Coury e A.a. de Franca Mendes Carneiro (2003). “An evolutionary optimization approach to track voltage and current harmonics in electrical power systems”. Em: *Power Engineering Society General Meeting, 2003, IEEE*. Vol. 2. IEEE, pp. 1250–1255.
- Deitel, HM e PJ Deitel (2005). *C++ How to Program, 5th*. Edition USA Prentice Hall.
- EL-Naggar, K e H. K. M. Youssef (set. de 2000). “A genetic based algorithm for frequency-relaying applications”. Em: *Electric Power Systems Research* 55.3, pp. 173–178.
- Energética, Empresa de Pesquisa (jan. de 2012). “Consumo de energia elétrica no brasil cresce 3,6% em 2011”. Em: *Resenha*, pp. 1–4.
- Funaki, T. e S. Tanaka (2002). “Error estimation and correction of DFT in synchronized phasor measuremen”. Em: *IEEE/PES Transmission and Distribution Conference and Exhibition*, pp. 448–453.
- Harris, F.J. (1978). “On the use of windows for harmonic analysis with the discrete Fourier transform”. Em: *Proceedings of the IEEE* 66.1, pp. 51–83.
- Hart, David, Damir Novosel, Yi Hu, Bob Smith e Mike Egolf (1997). “A new frequency tracking and phasor estimation algorithm for generator protection”. Em: *Power Delivery, IEEE Transactions on* 12.3, pp. 1064–1073.
- Hart, D.G., D. Uy, V. Gharpure, D. Novosel, D. Karlsson e M. Kaba (2001). “PMUs- a new approach to power network monitoring”. Em: *ABB Review*, pp. 58–61.
- IEEE (1995). *IEEE-Std.1344: IEEE Standard for Synchrophasors for Power Systems*. New York, United States: IEEE (Institute of Electrical e Electronics Engineers).
- (2005). *IEEE-Std.C37.118: IEEE Standard for Synchrophasors for Power Systems*. New York, United States: IEEE (Institute of Electrical e Electronics Engineers).
- Kamwa, I. (1991). “Fast adaptive schemes for tracking voltage phasor and local frequency in power transmission and distribution systems”. Em: *Transmission and Distribution*, pp. 930–936.

-
- Karimi, H., M. Karimi-Ghartemani e MR Iravani (2004). “Estimation of frequency and its rate of change for applications in power systems”. Em: *IEEE Transactions on Power Delivery* 19.2, pp. 472–480.
- Karimi-Ghartemani, M. e M.R. Iravani (2004). “Robust and frequency-adaptive measurement of peak value”. Em: *Power Delivery, IEEE Transactions on* 19.2, pp. 481–489.
- (jan. de 2005). “Measurement of harmonics/inter-harmonics of time-varying frequencies”. Em: *Power Delivery, IEEE Transactions on* 20.1, pp. 23–31.
- Karimi-Ghartemani, M., H. Karimi e M.R. Iravani (abr. de 2004). “A Magnitude/Phase-Locked Loop System Based on Estimation of Frequency and In-Phase/Quadrature-Phase Amplitudes”. Em: *IEEE Transactions on Industrial Electronics* 51.2, pp. 511–517.
- Kilts, S. (2007). *Advanced fpga design: architecture, implementation, and optimization*. Wiley-IEEE Press.
- Martin, K.E., D. Hamai, M.G. Adamiak, S. Anderson, M. Begovic, G. Benmouyal, G. Brunello, J. Burger, J.Y. Cai, B. Dickerson, V. Gharpure, B. Kennedy, D. Karlsson, A.G. Phadke, J. Salj, V. Skendzic, J. Sperr, Y. Song, C. Huntley, B. Kasztenny e E. Price (out. de 2008). “Exploring the IEEE Standard C37.118–2005 Synchrophasors for Power Systems”. Em: *IEEE Transactions on Power Delivery* 23.4, pp. 1805–1811.
- Nguyen, T.T. e X.J. Li (2006). “A fast and accurate method for estimating power system phasors using DFT with interpolation”. Em: *Proc. 2006 Power Engineering Society General Meeting*. Ieee, pp. 18–22.
- Nuqui, Reynaldo Francisco (2001). “State Estimation and Voltage Security Monitoring Using Synchronized Phasor Measurements Reynaldo Francisco Nuqui”. Tese de doutorado. Virginia Polytechnic Institute e State University.
- Oppenheim, A.V., A.S. Willsky e S.H. Nawab (1983). *Signals and systems*. Vol. 2. Prentice-Hall Englewood Cliffs, NJ.
- Phadke, AG (1993). “Synchronized phasor measurements in power systems”. Em: *IEEE Computer Applications in Power* 6.2, pp. 10–15.
- Phadke, A.G. (2002). “Synchronized phasor measurements-a historical overview”. Em: *IEEE/PES Transmission and Distribution Conference and Exhibition*. Vol. 1. Asia Pacific: IEEE, pp. 476–479.

- Phadke, AG e J.S. Thorp (2006). “History and applications of phasor measurements”. Em: *Power Systems Conference and Exposition*, pp. 331–335.
- Phadke, AG, JS Thorp e MG Adamiak (1983). “A new measurement technique for tracking voltage phasors, local system frequency, and rate of change of frequency”. Em: *Power Apparatus and Systems, IEEE Transactions on* 5, pp. 1025–1038.
- Sachdev, MS (1985). “A least error squares technique for determining power system frequency”. Em: *Power Apparatus and Systems, IEEE* 2, pp. 437–444.
- Sachdev, MS e MA Baribeau (1979). “A new algorithm for digital impedance relays”. Em: *Power Apparatus and Systems, IEEE Transactions on* 6, pp. 2232–2240.
- Santoso, Surya, H. Wayne Beaty, Roger C. Dugan e Mark F. McGranaghan (2002). *Electrical Power Systems Quality*. McGraw-Hill Professional, p. 528.
- Schmidt, Cristiane Alkmin Junqueira e Marcos A. M. Lima (mar. de 2004). “A demanda por energia elétrica no Brasil”. Em: *Revista Brasileira de Economia* 58.1, pp. 67–98.
- Souza, SA, RA Macêdo, ET Vargas, DV Coury e M. Oleskovicz (2006). “Estimação de Parâmetros de um Sistema Elétrico de Potência Utilizando Algoritmos Genéticos”. Em: *IEEE Latin America Transactions* 4.1, pp. 47–54.
- Stevenson, Willian D. (1986). *Elementos de Análise de Sistemas de Potência*. 2 ed. São Paulo: McGraw-Hill, p. 458.
- Terzija, V.V., MB Djuric e BD Kovacevic (1994). “Voltage phasor and local system frequency estimation using Newton type algorithm”. Em: *Power Delivery, IEEE Transactions on* 9.3, pp. 1368–1374.
- Vieira Filho, X (1984). “Operação de Sistemas de Potência com Controle Automático de Geração”. Em: *Editora Campus*.
- Wang, Maohai e Y. Sun (2004). “A practical, precise method for frequency tracking and phasor estimation”. Em: *Power Delivery, IEEE Transactions on* 19.4, pp. 1547–1552.
- (2006). “A practical method to improve phasor and power measurement accuracy of DFT algorithm”. Em: *Power Delivery, IEEE Transactions on* 21.3, pp. 1054–1062.
- Ziarani, AK e A. Konrad (2004). “A method of extraction of nonstationary sinusoids”. Em: *Signal Processing* 84.8, pp. 1323–1346.

Apêndice A

Algoritmos Genéticos

Neste apêndice são apresentados os conceitos teóricos relativos à AGs definidos por diversos autores na área, bem como as bases teóricas biológicas, que são os conceitos que inspiraram a Computação Evolutiva (CE), da qual os AGs fazem parte. Em seguida é apresentada a terminologia referente a essa classe de algoritmos exemplificado sob a uma forma de um AG simples. Na sequência são apresentados os operadores de seleção, recombinação (*crossover*) e mutação. Por fim, é apresentado um AG baseado em modelos probabilísticos, que tem funcionamento diferente dos algoritmos tradicionais.

A.1 Introdução

A teoria da seleção natural de Darwin propõe que plantas e animais que existem hoje são o resultado de milhões de anos de adaptação as mudanças no meio-ambiente. A qualquer momento, diferente organismos podem coexistir e competir por recursos em um ecossistema. Os organismos que são mais capazes de se adaptar ao meio onde vivem, adquirir mais recursos e procriar, são aqueles cujos os descendentes serão mais numerosos no futuro. Organismos que são menos capazes, serão aqueles que terão menos ou nenhum descendentes. O primeiro é dito como mais apto do que o último, e as características distintas que proporcionaram a aptidão do primeiro são ditas como selecionadas sobre as características do último. Ao longo do tempo, a população inteira do ecossistema irá evoluir e conter organismos, que na média, são mais aptos do que os organismos da geração anterior da população, pois eles possuem mais características que tendem a promover a sobrevivência.

A CE abstrai os princípios evolucionários citados anteriormente em algoritmos que podem ser usados para pesquisa de soluções ótimas de um problema. Em um algoritmo de pesquisa, um número de possíveis soluções para um problema estão disponíveis, e a tarefa é encontrar a melhor solução possível em um determinado instante de tempo. Para um espaço de busca com um número pequeno de soluções, todas as soluções podem ser avaliadas em um tempo razoável de forma a se encontrar o ótimo. No entanto, essa pesquisa exaustiva se torna inviável conforme o espaço de busca cresce. Algoritmos de busca tradicionais amostram o espaço de busca aleatoriamente ou heurísticamente (gradiente descendente) uma solução por vez de forma a encontrar o ótimo. O aspecto mais marcante que distingue uma busca evolutiva destes algoritmos tradicionais é que a primeira é baseada em população. Através da adaptação de sucessivas gerações de um grande número de indivíduos, um algoritmo evolutivo realiza uma busca direta eficiente.

Na computação CE, existem quatro paradigmas históricos que são responsáveis pelas pesquisas na área: algoritmos genéticos, programação genética, estratégias evolutivas e programação evolutiva. As diferenças básicas entre estes paradigmas reside na natureza nos esquemas de representação, operadores de reprodução e métodos de seleção (Sivanandam e Deepa, 2010).

A.2 Os Algoritmos Genéticos

Os AGs têm sido usados na ciência e na engenharia como algoritmos adaptativos de busca de soluções em problemas e como modelo computacional de sistemas naturais evolucionários. Vale lembrar que os AGs são uma classe de procedimentos com um conjunto de passos distintos e bem especificados, no qual cada um destes passos possui muitas possíveis variações.

As tarefas de busca ou otimização possuem vários componentes, entre eles o espaço de busca, no qual são consideradas todas as possibilidades de solução de um determinado problema; e a função de avaliação, a qual avalia as soluções no espaço de busca (Winston, 1992).

Os métodos e técnicas de busca e otimização tradicionais iniciam-se com um único candidato que é manipulado utilizando, geralmente, métodos determinísticos para resolução de um problema. Por outro lado, as técnicas mais recentes, fundamentadas na computação evolucionária, operam sobre uma população de candidatos em paralelo. Assim, elas podem fazer a busca em diferentes áreas do espaço de soluções, alocando um número de soluções apropriadas para a busca em várias regiões. Portanto, essas técnicas têm maiores chances de atingir as áreas mais

promissoras do espaço de busca.

Segundo Back (1994), O AG difere dos métodos tradicionais de busca e otimização em quatro aspectos:

- O AG trabalha com uma codificação do conjunto de parâmetros e não com os próprios parâmetros;
- O AG explora um espaço de busca, no qual estão todas as possíveis soluções do problema;
- O AG utiliza informações de custo e não derivadas ou conhecimento auxiliar; e
- O AG utiliza regras de transição probabilísticas e determinísticas.

O AG se baseia, inicialmente, na geração de uma população formada por um conjunto aleatório de indivíduos que podem ser vistos como possíveis soluções do problema. Durante o processo evolutivo, esta população é avaliada de maneira que para cada indivíduo é dado um índice, o qual reflete sua habilidade de adaptação a determinado ambiente. Uma porcentagem dos mais adaptados é mantida, enquanto os outros são descartados. Os membros mantidos pela seleção podem sofrer modificações em suas características, através dos operadores genéticos, gerando descendentes para a próxima geração. Este processo, chamado de reprodução, é repetido até que um conjunto de soluções satisfatórias seja encontrado, conforme apresentado na figura A.1 (Souza, 2008). Cada uma das fases do algoritmo será comentada com mais detalhes a seguir.

Figura A.1: Fluxograma de um algoritmo genético clássico.

Embora possam parecer simplistas do ponto de vista biológico, esses algoritmos são suficientemente complexos para fornecer mecanismos poderosos e robustos de busca adaptativa (Goldberg, 1989). Ressalta-se que o desempenho da implementação do AG dependerá da escolha da melhor

representação do problema, do tamanho da população inicial, dos tipos de operadores genéticos escolhidos, das probabilidades de tais operações ocorrerem e da escolha da função de avaliação.

A.2.1 O Indivíduo

Um indivíduo, é uma possível solução para o problema. Indivíduos agrupam duas formas de solução:

- O cromossomo, que é a informação genética com a qual o AG aplica suas operações;
- O fenótipo, que é a expressão do cromossomo nos termos do modelo abordado.

Na figura A.2 pode ser observado um indivíduo, ou cromossomo. Um indivíduo é composto por genes. Cada gene é a representação de um fator, em que o conjunto de fatores compõem o fenótipo. Deste modo, cada gene contém informações sobre a solução que ele representa, enquanto os fatores são as soluções propriamente ditas.

Figura A.2: Representação de um indivíduo.

A grosso modo, os genes são codificações de variáveis em um problema de otimização. O fenótipo é a expressão do gene no modelo, ou seja, é o valor da variável propriamente dito. As formas de codificações mais comumente usadas são string de bits, e codificação por valor, as quais são discutidas a diante.

A.2.2 A População

A população é um conjunto de indivíduos presente em um espaço de busca e que são testados através da avaliação de seu fenótipo. Dois aspectos com relação a população devem ser considerados:

- A população inicial;

- O tamanho da população.

O tamanho da população para que o AG tenha uma boa resposta irá depender da complexidade do problema abordado. Comummente utiliza-se uma população inicializada aleatoriamente. Para o caso de uma população composta por cromossomos de *strings* de bits, isso significa que os bits serão inicializados aleatoriamente em 0 ou 1. No entanto, pode-se utilizar algum conhecimento do domínio do problema para melhor inicializar a população.

Idealmente, a população inicial deve ter uma diversidade genética grande de modo que seja possível explorar o espaço de busca adequadamente. Novamente, recorrendo ao caso dos cromossomos binários, isso significa que deve-se haver todas as combinações de alelos nos genes possíveis.

A.2.3 A Codificação

Como foi dito, o cromossomo que representa o indivíduo, é constituído de genes, que tem seu efeito refletido pela criação de diferentes fenótipos. Ou seja, cada gene diferente poderá criar diferentes fenótipos.

Na prática, isso é realizado com a utilização de codificação, ou seja, o cromossomo codifica um valor que é avaliado através da função de *fitness* ou de aptidão. Dois tipos de codificação, que são os tipos mais comuns serão utilizados neste trabalho: codificação binária, e codificação por valor.

Na codificação binária, cada gene é constituído por uma string de bits, em que cada combinação de bits geram diferentes fenótipos. A grosso modo, considerando-se um problema de otimização, a codificação é feita pelo mapeamento de um intervalo de uma variável dentro de das possíveis combinações. Isso é muito interessante do ponto de vista computacional, já que a memória de um computador é uma imensa matriz de 0s e 1s. Para o caso da codificação binária precisa-se de uma função que irá efetuar a conversão da codificação binária para os valores desejados. Toma-se como exemplo a tabela A.1, em que o intervalo real $[59 : 61]$ está mapeado em um gene de 10 bits.

Uma formula prática para a codificação efetuada em um intervalo genérico $[lb : ub]$ é

$$V = G \cdot \frac{(ub - lb)}{(2^N - 1)}, \quad (A.1)$$

Tabela A.1: Mapeamento por gene em um intervalo real

Gene (Codificado)	Valor Decodificado
0000000000 (0)	59,0
0000000001 (1)	59,001955034
0000000010 (2)	59,003910068
0000000011 (3)	59,005865103
$\vdots$	$\vdots$
1111111110 (1022)	60,998044966
1111111111 (1023)	61,0

em que G é o valor inteiro do cromossomo, ub é o limite superior do intervalo, lb é o limite inferior do intervalo e N é o comprimento da *string* de bits de um gene.

A codificação por valor é mais simples, pois não exige a conversão. O próprio valor que o gene guarda já é o valor da variável no problema de otimização. Usualmente, utiliza-se codificação por valor com genes guardando valores de ponto flutuante.

A.2.4 A Avaliação da Aptidão

Após a decodificação dos genes, a avaliação da aptidão é realizada por uma função matemática, a função objetivo, a qual avalia a evolução dos indivíduos da população a cada geração. O valor de saída da função classifica cada indivíduo em relação à solução encontrada pelo AG. A partir desta classificação, os melhores indivíduos obtidos serão selecionados na função Seleção e originarão os futuros indivíduos da próxima geração.

A função de aptidão normalmente é modelada baseada em um erro calculado pela diferença entre os valores tomados como base e aqueles estimados pelo AG. O resultado da função, calculado a cada iteração, indicará a qualidade da resposta encontrada pelo AG em relação aos valores de referência. Do ponto de vista adaptativo, a avaliação da aptidão determina a capacidade do indivíduo sobreviver perante as variações dos parâmetros envolvidos.

A.2.5 O Processo de Seleção

A seleção é um dos mais importantes operadores que agem sobre uma população em um AG. A seleção tem por objetivo selecionar os melhores indivíduos (melhores soluções) em uma população de forma a replicar e modificar estes melhores indivíduos para gerações futuras através de operadores genéticos. Analogamente, este processo ocorre na natureza, em que indivíduos mais adaptados ao ambiente possuem mais condições para se reproduzir do que indivíduos menos

adaptados (Souza, 2008).

O processo de seleção leva em conta a qualificação dos indivíduos por parte da função aptidão, em que indivíduos são selecionados para a função de cruzamento. Deve ser notada a importância deste processo, pois guia o sucesso do AG de forma a selecionar as melhores soluções, sendo desconhecida a modelagem do problema, caracterizando assim esse operador como universal.

Existem vários operadores de seleção, entre os quais são destacados o Método da Roleta e o Método do Torneio. A seguir considera-se um exemplo a ser aplicado cada para cada um dos operadores de seleção destacados anteriormente. Leve em conta os indivíduos destacados na Tabela A.2. São apresentados 8 indivíduos, cada um sua respectiva aptidão. Indivíduos são selecionados de acordo com sua aptidão de acordo com cada método, lembrando que, quanto maior a aptidão, mais adaptado é o indivíduo (melhor solução).

Tabela A.2: Aptidão de indivíduos em uma população

Indivíduo	Aptidão
I1	1
I2	1,5
I3	1,7
I4	9
I5	3
I6	10
I7	8
I8	5,1

O Método da Roleta

O funcionamento do método da roleta é similar ao funcionamento de uma roleta utilizada em jogos de azar. Cada indivíduo é particionado de acordo com sua partição sobre um disco, de forma que cada partição seja proporcional à aptidão de cada indivíduo. Desta forma, indivíduos que tiverem uma aptidão grande ocuparão uma maior partição na roleta, sendo mais facilmente escolhidos. A Figura A.3 ilustra o funcionamento do método da roleta utilizando os valores de aptidão descritos na Tabela A.2.

A cada giro da roleta, um indivíduo apontado pela seta é selecionado. Como os indivíduos de maior aptidão tem uma maior partição na roleta, estes tem mais chances de serem escolhidos.

Figura A.3: Seleção pelo método da roleta.

O Método do Torneio

No método do torneio, n indivíduos são selecionados aleatoriamente na população. O melhor indivíduo dentre estes n será o selecionado. A Figura A.3 apresenta o funcionamento do método da roleta para $n = 3$ utilizando os valores de aptidão descritos na Tabela A.2.

Figura A.4: Seleção pelo método de torneio

A.2.6 Operadores Genéticos

Após a aplicação da seleção, são necessários mecanismos que melhorem as soluções disponíveis. No caso dos AGs estes mecanismos são os operadores de cruzamento (*crossover*) e mutação.

O cruzamento é baseado no mecanismo de reprodução dos seres vivos, em que características de dois indivíduos previamente selecionados são recombinadas para a criação de um novo indivíduo.

O operador de mutação tem o objetivo de manter a variabilidade genética em uma determinada população. Neste caso são feitas pequenas alterações em cada indivíduo, introduzindo assim novas características na população.

Espera-se que a partir da aplicação sucessiva dos operadores descritos anteriormente em conjunto com o operador de seleção leve a geração de indivíduos com alto valor de aptidão. A seguir são explicados a implementação de operadores de seleção e mutação para o caso de codificação real e binária.

Operadores Genéticos Reais

A forma como os operadores são aplicados está intimamente ligada ao tipo de codificação utilizada pelos indivíduos. Melhores resultados para codificação real é obtida pela utilização de operadores aritméticos (Haupt e Haupt, 2004). A seguir são apontados alguns operadores reais.

- Cruzamento de média Aritmética: os genes (g_n) de um novo indivíduo é gerado a partir dos genes de dois indivíduos selecionados (g_1 e g_2)

$$g_n = \frac{g_1 + g_2}{2}. \quad (\text{A.2})$$

- Cruzamento Média Geométrica: É uma variação do cruzamento por média aritmética, no qual o novo gene é dado por

$$g_n = \sqrt{g_1 + g_2}. \quad (\text{A.3})$$

estes tipos de cruzamento podem causar perda de diversidade, uma vez que o mesmo tende a levar genes para o meio do intervalo permitido. Este tipo de problema pode ser evitado com o cruzamento BLZ- α .

- Cruzamento BLX- α : dados os genes de dois indivíduos selecionados (g_1 e g_2), é produzido um novo gene g_n a partir de

$$g_n = g_1 + \beta(g_1 + g_2), \quad (\text{A.4})$$

em que $\beta \in (-\alpha; 1 + \alpha)$. Para o caso de $\alpha = 0,5$, a tendência de gerar filhos próximos ao centro do intervalo é balanceada, evitando a perda de diversidade.

- Mutação uniforme real: Ocorre devido a simples soma de um gene selecionado para mutação com um número aleatório em um intervalo definido.

$$g = g + U(a, b), \quad (\text{A.5})$$

em que $U(a, b)$ representa um número aleatório com distribuição uniforme entre o intervalo

$(a; b)$.

- Mutação gaussiana real: Ocorre devido soma de um gene selecionado para mutação com um número aleatório de distribuição normal.

$$g = g + N(p, \sigma), \quad (\text{A.6})$$

em que $N(p, \sigma)$ é uma distribuição normal de média p e desvio padrão σ .

Operadores Genéticos Binários

Com cruzamento de dois indivíduos progenitores, uma ou mais soluções são criadas a partir do intercâmbio de suas informações genéticas. Este operador pode ser utilizado de várias maneiras. As mais utilizadas estão descritas a seguir.

- Cruzamento de Um-Ponto: um ponto de cruzamento é escolhido e a partir dele as informações genéticas dos progenitores são trocadas, como observado na Figura A.5.

Figura A.5: Cruzamento de Um-Ponto

- Cruzamento Multipontos: é a generalização da ideia de troca de material genético através de vários pontos. O funcionamento deste operador para dois pontos é demonstrado na Figura A.6.

Figura A.6: Cruzamento Multipontos

- Mutação: Após a operação de cruzamento, o operador de mutação é aplicado, com uma determinada probabilidade, em cada bit dos filhos gerados. O operador de mutação inverte o valor dos bits selecionados. O processo de mutação pode ser observado na Figura A.7

Figura A.7: Processo de mutação.

A.2.7 Elitismo

Quando se aplicam os operadores genéticos, todos os indivíduos de uma população são substituídos a cada geração. Devido a esse processo, corre-se o risco de que boas soluções sejam descartadas. Para que isso não ocorra, aplica-se uma técnica denominada elitismo, que consiste em manter os melhores indivíduos de uma população presentes nas próximas. Tal técnica garante que os melhores indivíduos pesquisados até o momento estejam presentes em cada geração.

A.2.8 Parâmetros Genéticos

O desempenho de um AG é influenciado pela definição dos parâmetros a serem utilizados no algoritmo. Portanto, é importante medir como esses parâmetros afetam o problema abordado, bem como os recursos computacionais disponíveis. Os principais parâmetros são descritos a seguir.

- **Tamanho da população:** O tamanho da população afeta o desempenho dos AGs. Uma população que tenha uma boa representação do espaço de busca irá solucionar um problema com maior eficiência do que uma população pequena. No entanto, uma grande população exige mais recursos computacionais.
- **Taxa de cruzamento:** representa a probabilidade do operador de cruzamento (*crossover*) ser aplicado entre dois indivíduos selecionados (pais). Se a taxa de cruzamento for muito alta, pode-se eliminar indivíduos com boa aptidão rapidamente. Caso seja muito baixa, pode ocorrer a estagnação da busca.
- **Taxa de mutação:** representa a probabilidade do operador de mutação ser aplicado, para a manutenção da variabilidade genética. Uma taxa de mutação muito baixa irá resultar em baixa variabilidade genética. Uma taxa muito alta pode implicar na destruição de estruturas com boa aptidão.

A.3 O Algoritmo Genético Compacto

A população em um AG tem um papel fundamental no sucesso deste tipo de algoritmo, pois representa as possíveis soluções para o problema abordado. Quanto melhor for a representação do espaço de busca por parte da população, melhor será a qualidade das soluções encontradas. No entanto, existe um custo computacional associado ao armazenamento da população. Dependendo do tamanho da população necessária para a resolução de um problema, a aplicabilidade computacional é inviável. Desta forma, surgem estratégias para minimizar o custo computacional do armazenamento da população.

Harik, Lobo e Goldberg (1999) apresentam uma estratégia que representa a população como uma estimativa de distribuição sobre o conjunto de soluções e que é operacionalmente equivalente ao AG simples (Goldberg, 1989) com recombinação uniforme. Este algoritmo, denominado AGc, processa cada gene independentemente bit a bit utilizando uma memória finita, dessa forma economizando memória computacional. Esta memória finita é denominada vetor de probabilidade, em que cada posição desta guarda a probabilidade de um bit (locus) de um gene ser 1. Desta forma o vetor de probabilidade representa a proporção de 0 e 1 em uma determinada população para cada alelo dos genes da população.

O processo de execução do AGc baseia-se na atualização do vetor de probabilidade a medida que novos indivíduos são gerados e avaliados. Inicialmente o vetor de probabilidade é setado com probabilidade de 0,5 para todos os bits. Em seguida são gerados dois indivíduos com probabilidades obedecendo o vetor. Os dois indivíduos são avaliados e classificados como ganhador e perdedor. A grosso modo, o indivíduo ganhador contém melhores esquemas do que o indivíduo perdedor, de forma que deseja-se manter estes esquemas, enquanto o indivíduo perdedor tem esquemas tidos como piores, a serem evitados. Este esquema é semelhante a uma seleção binária por torneio. Em uma população de tamanho n , a proporção de alelos ganhadores vai aumentar a uma proporção de $1/n$. Desta forma, deve-se atualizar o vetor de probabilidades de acordo com este critério. Após a atualização do vetor de probabilidade, dois novos indivíduos são gerados e o processo recomeça a partir deste ponto. O processo termina quando ocorre a convergência de todas as posições do vetor de probabilidade. O processo pode ser observado mais claramente na Figura A.8, em que n representa o tamanho da população e p representa o vetor de probabilidade.

O processo de convergência do modelo, vetor de probabilidades p , pode ser observado na Figura A.9, onde observa-se a convergência dos alelos dos genes. .

Figura A.8: Fluxograma de um AGc

Figura A.9: Modelo de probabilidade para a população de um AGc

Deve-se levar em conta que o AGc é binário e portanto necessita de codificação para extrair o valor das variáveis no problema estudado. Pelo fato de ser binário e não necessitar de grande quantidade de memória, o AGc pode ser utilizado para a implementação de AGs de forma embarcada.

Referências Bibliográficas

- Back, T. (1994). *Selective pressure in evolutionary algorithms: a characterization of selection mechanisms*. IEEE, pp. 57–62.
- Goldberg, David E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, p. 412.

Harik, G. R., F. G. Lobo e David E. Goldberg (out. de 1999). “The Compact Genetic Algorithm”.

Em: *Evolutionary Computation, IEEE Transactions on* 3.4, pp. 287–297.

Haupt, Randy L. e Sue Ellen Haupt (2004). *Practical Genetic Algorithms*. Wiley-Interscience, p. 253.

Sivanandam, S.N. e S. N. Deepa (2010). *Introduction to Genetic Algorithms*. Springer, p. 442.

Souza, S.A. (2008). “Algoritmos Genéticos Aplicados À Proteção e Estimação de Harmônicos em Sistemas Elétricos de Potência”. Tese de doutorado. Universidade de São Paulo.

Winston (1992). *Artificial Intelligence*. 3ª ed. Addison Wesley, p. 750.

Apêndice B

Arquivos ATP

Neste apêndice são apresentados os arquivos utilizados no *software* ATP para realizar as simulações computacionais do sistema elétrico proposto para estudo.

```
1
2 BEGIN NEW DATA CASE
3 JMARTI SETUP, 1.0
4 $ERASE
5 C      1      2      3      4      5      6      7
6 C 34567890123456789012345678901234567890123456789012345678901234567890
7 BRANCH LT1IA2LT1FA2LT1IB2LT1FB2LT1IC2LT1FC2
8 LINE CONSTANTS
9 METRIC
10 C DADOS CONDUTORES POR FASE
11 C      1      2      3      4      5      6      7
12 C 34567890123456789012345678901234567890123456789012345678901234567890
13 C ><SKI><RESIS ><><REACT ><DIAM ><HORIZ ><VTOWER><VMID >
14 1 0.815 0.09136 4      2.48400 -8.9692 31.772 15.552
15 1 0.815 0.09136 4      2.48400 -9.4232 31.772 15.552
16 1 0.815 0.09136 4      2.48400 -9.4232 31.318 15.098
17 1 0.815 0.09136 4      2.48400 -8.9692 31.318 15.098
18 2 0.815 0.09136 4      2.48400 0.2270 35.772 19.552
19 2 0.815 0.09136 4      2.48400 -0.2270 35.772 19.552
20 2 0.815 0.09136 4      2.48400 -0.2270 35.318 19.098
21 2 0.815 0.09136 4      2.48400 0.2270 35.318 19.098
22 3 0.815 0.09136 4      2.48400 9.4232 31.772 15.552
23 3 0.815 0.09136 4      2.48400 8.9692 31.772 15.552
24 3 0.815 0.09136 4      2.48400 8.9692 31.318 15.098
25 3 0.815 0.09136 4      2.48400 9.4232 31.318 15.098
26 0 0.500 4.18000 4      0.91440 -9.1992 36.845 22.245
27 0 0.500 4.18000 4      0.91440 9.1992 36.845 22.245
28 BLANK card ending conductor cards
29 C DADOS FREQUENCIA
30 C      1      2      3      4      5      6      7
31 C 34567890123456789012345678901234567890123456789012345678901234567890
32 C RHO >< FREQ >< FCAR > < DIST > I <D><P> <>
33 1000.0      60.0      1      75      0
34 1000.0      0.001      1      75      9 10 0
35 BLANK card ending frequency cards
36 BLANK card ending LINE CONSTANTS imbedded usage
```

```

37 C      1      2      3      4      5      6      7
38 C 34567890123456789012345678901234567890123456789012345678901234567890
39 C      <IDEBUG><IPUNCH><KOUTPR><GMODE >
40      1      3.E-8
41 C      1      2      3      4      5      6      7
42 C 34567890123456789012345678901234567890123456789012345678901234567890
43 C NEXMI><EPSTOL><NORMAX><IECODE><IFWTA ><IFPLOT>< IFDAT><INELIM>
44      0.30      30      0      1      0      0
45      0.30      30      0      1      0      0      .0
46 BLANK card terminating JMARTI
47 $PUNCH
48 BEGIN NEW DATA CASE
49 BLANK card terminating all cases

```

Código B.1: Modelo de linhas no Jmarti

```

1 C -----
2 C MAQUINA 1 DA UNIDADE DE GERACAO 1
3 C -----
4 C Gerador Sincrono - Serra da Mesa 445 MVA 60 polos 15 kV 60 Hz
5 C < n 1><>< Tensao >< Freq >< Angulo >
6 59GER11A      12.24E+3      60.0      0.0
7 GER11B
8 GER11C
9 C      1      2      3      4      5      6      7      8
10 C 34567890123456789012345678901234567890123456789012345678901234567890
11 TOLERANCES      300
12 PARAMETER FITTING      2.0
13 C      1      2      3      4      5      6      7      8
14 C 34567890123456789012345678901234567890123456789012345678901234567890
15 C <><> <N>< SMOTVTP>< SMOVTQ >< RMVA >< RKV >< AGLINE >< S1D >< S2D >
16 1 1 1 60      1      1      445.0      15.0      3142
17      -1
18 C      1      2      3      4      5      6      7      8
19 C 34567890123456789012345678901234567890123456789012345678901234567890
20 C --- Electrical parameters - Manufacture's data
21 C RA >< XL >< XD >< XQ >< XDD >< XQQ >< XDDD >< XQQQ >
22 0.0019      0.150      1.0      0.930      0.280      0.600      0.180      0.180
23 C TDOP >< TQOP >< TDOPP >< TQOPP >< XO >< RN >< XN >< XCAN >
24 5.0      0.1      0.21      0.160
25 C 34567890123456789012345678901234567890123456789012345678901234567890
26 C --- Mechanical System data
27 C < >< EXTRS >< HICO >< DSR >< HSP >< DSB >
28 1      1.0      26.6
29 BLANK CARD ENDING MASS CARDS
30 C --- DATA OUTPUT
31 C GA< >< >< >< >< >< >< >< >< >< >
32 C 11
33 C 21
34 C 31
35 C 51
36 BLANK CARD ENDING OUTPUT REQUEST
37 71EFD11
38 72WAT11      1
39 74VEL11      2
40 FINISH
41 C -----
42 C MAQUINA 2 DA UNIDADE DE GERACAO 1
43 C -----
44 C Gerador Sincrono - Serra da Mesa 445 MVA 60 polos 15 kV 60 Hz
45 C < n 1><>< Tensao >< Freq >< Angulo >

```

```

46 59GER12A      12.24E+3      60.0      0.0
47 GER12B
48 GER12C
49 C      1      2      3      4      5      6      7      8
50 C 3456789012345678901234567890123456789012345678901234567890
51 TOLERANCES                                     300
52 PARAMETER FITTING                          2.0
53 C      1      2      3      4      5      6      7      8
54 C 3456789012345678901234567890123456789012345678901234567890
55 C <><> <N>< SMOTVTP>< SMOVTQ >< RMVA >< RKV >< AGLINE >< S1D >< S2D >
56 1 1 1 60      1      1      445.0      15.0      3142
57                                     -1
58 C      1      2      3      4      5      6      7      8
59 C 3456789012345678901234567890123456789012345678901234567890
60 C --- Electrical parameters - Manufacture's data
61 C RA >< XL >< XD >< XQ >< XDD >< XQQ >< XDDD >< XQQQ >
62 0.0019 0.150 1.0 0.930 0.280 0.600 0.180 0.180
63 C TDOP >< TQOP >< TDOPP >< TQOPP >< XO >< RN >< XN >< XCAN >
64 5.0 0.1 0.21 0.160
65 C 3456789012345678901234567890123456789012345678901234567890
66 C --- Mechanical System data
67 C < >< EXTRS >< HICO >< DSR >< HSP >< DSB >
68 1 1.0 26.6
69 BLANK CARD ENDING MASS CARDS
70 C --- DATA OUTPUT
71 C GA< >< >< >< >< >< >< >< >< >< >
72 C 11
73 C 21
74 C 31
75 C 51
76 BLANK CARD ENDING OUTPUT REQUEST
77 71EFD12
78 72WAT12      1
79 74VEL12      2
80 FINISH

```

Código B.2: Geradores do Subsistema 1

```

1 C      1      2      3      4      5      6      7      8
2 C 3456789012345678901234567890123456789012345678901234567890
3 C -----
4 C REGULADOR DE VELOCIDADE UNIDADE 1 GERADOR 1
5 C -----
6 C Medicao Frequencia na barra do Gerador
7 90GER11A
8 C 98FGER 50      +GER11A      6.00E1
9 C 33FGER
10 C CARREGANDO A VELOCIDADE DO GERADOR      < T INIT ><T FIM >
11 92VEL11      -1.0000000
12 77VEL11 94.2477796
13 C 33VEL11
14 C NIVEL DO SINAL
15 C <OUT ><>< A >< B >< C >      < T INIT ><T FIM >
16 11FSOLL1 6.000000E10.000000E00.000000E0      -1.0000000
17 C 33FSOLL1
18 C 11VREF 8.365805E30.000000E00.000000E0      -1.0000000 {1.05 PU}
19 C 33VREF
20 C CALCULO DA FREQUENCIA
21 88FGER11 = ((30 * VEL11) / (2 * PI))
22 C CALCULO DA VARIACAO DA FREQUENCIA - COMPARADOR
23 88FDEL11 = FGER11 - FSOLL1

```

```

24 C 33FDEL11
25 C CONTROLE - REGULADOR DE VELOCIDADE PRIMARIO 1
26 77RV111 0.0
27 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
28 1RV111 +RVRP11 1.000
29 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
30 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
31 1 3
32 1 70
33 C 33RV111 -> 7 e 100
34 C CONTROLE - REGULACAO DA VELOCIDADE EM RP (1/R)
35 77RVRP11 0.0
36 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
37 0RVRP11 -FDEL11 10.0
38 C 33RVRP11
39 C CONTROLE - REGULADOR DE VELOCIDADE PRIMARIO 2
40 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
41 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
42 1RV211 +RV111 1.000
43 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
44 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
45 1
46 1 0.2
47 C 33RV211
48 C CONTROLE - FUNCAO DE TRANSFERENCIA DA TURBINA
49 77CTUR11 0.0
50 77WAT11 1.0
51 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
52 1CTUR11 +RV211 1.000
53 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
54 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
55 1 -0.15
56 1 0.075
57 C 33RV211
58 C 33CTUR11
59 C CALCULO DA POTENCIA MAXIMA
60 98WAT11 = 1 + CTUR11
61 C 33WAT11
62 C -----
63 C REGULADOR DE TENSÃO UNIDADE 1 GERADOR 1
64 C -----
65 C INICIALIZANDO AS VARIÁVEIS
66 11VREF 8.660254E30.000000E00.000000E0 -1.0000000 {1 PU}
67 77EFD11 0.985
68 77VS11 0.0
69 C 90GERB
70 C 1 2 3 4 5 6 7 8
71 C 34567890123456789012345678901234567890123456789012345678901234567890
72 88VGR11 66+GER11A 60.000
73 C SOMATORIO 1 - DELTA V E A RETROALIMENTACAO
74 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
75 0VS11 -VGR11 -VF11 +VREF 1.000
76 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
77 1VR11 +VS11 1.000
78 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
79 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
80 1. 0.60
81 1. 0.05
82 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
83 1VRT11 +VR11 1.000 0 1.15
84 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >

```



```

85 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
86 46
87 1 0.06
88 C SOMATORIO 2
89 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
90 1EFD11 +VRT11 -VFE11 1.000
91 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
92 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
93 1
94 0 0.46
95 C RETROALIMENTACAO LOCAL
96 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
97 0VFE11 +EFD11 0.95
98 C RETROALIMENTACAO GERAL
99 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
100 1VF11 +EFD11 1.000
101 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
102 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
103 0 0.1
104 1 1.0
105 C PLOTANDO AS SAIDAS
106 C 33VGR11
107 C 33VRT11
108 C 33EFD11
109 C 33VR11
110 C -----
111 C REGULADOR DE VELOCIDADE UNIDADE 1 GERADOR 2
112 C -----
113 C Medicao Frequencia na barra do Gerador
114 90GER12A
115 C 98FGER 50 +GER12A 6.00E1
116 C 33FGER
117 C CARREGANDO A VELOCIDADE DO GERADOR < T INIT ><T FIM >
118 92VEL12 -1.0000000
119 77VEL12 94.2477796
120 C 33VEL12
121 C NIVEL DO SINAL
122 C 33VREF
123 C CALCULO DA FREQUENCIA
124 88FGER12 = ((30 * VEL12) / (2 * PI))
125 C 33FGER12
126 C CALCULO DA VARIACAO DA FREQUENCIA - COMPARADOR
127 88FDEL12 = FGER12 - FSOLL1
128 C 33FDEL12
129 C CONTROLE - REGULADOR DE VELOCIDADE PRIMARIO 1
130 77RV112 0.0
131 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
132 1RV112 +RVRP12 1.000
133 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
134 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
135 1 3
136 1 70
137 C 33RV112 -> 7 e 100
138 C CONTROLE - REGULACAO DA VELOCIDADE EM RP (1/R)
139 77RVRP12 0.0
140 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
141 0RVRP12 -FDEL12 10.0
142 C 33RVRP12
143 C CONTROLE - REGULADOR DE VELOCIDADE PRIMARIO 2
144 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
145 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>

```

```

146 1RV212                      +RV112                      1.000
147 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
148 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
149      1
150      1      0.2
151 C 33RV212
152 C CONTROLE - FUNCAO DE TRANSFERENCIA DA TURBINA
153 77CTUR12  0.0
154 77WAT12    1.0
155 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
156 1CTUR12      +RV212                      1.000
157 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
158 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
159      1      -0.15
160      1      0.075
161 C 33RV212
162 C 33CTUR12
163 C CALCULO DA POTENCIA MAXIMA
164 98WAT12  = 1 + CTUR12
165 C 33WAT12
166 C -----
167 C REGULADOR DE TENSÃO UNIDADE 1 GERADOR 2
168 C -----
169 C INICIALIZANDO AS VARIÁVEIS
170 77EFD12  0.985
171 77VS12    0.0
172 C 90GERB
173 C      1      2      3      4      5      6      7      8
174 C 34567890123456789012345678901234567890123456789012345678901234567890
175 88VGR12 66+GER12A                      60.000
176 C SOMATORIO 1 - DELTA V E A RETROALIMENTAÇÃO
177 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
178 0VS12    -VGR12 -VF12 +VREF                      1.000
179 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
180 1VR12    +VS12                      1.000
181 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
182 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
183      1.      0.60
184      1.      0.05
185 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
186 1VRT12    +VR12                      1.000 0      1.15
187 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
188 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
189      46
190      1      0.06
191 C SOMATORIO 2
192 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
193 1EFD12    +VRT12 -VFE12                      1.000
194 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
195 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
196      1
197      0      0.46
198 C RETROALIMENTAÇÃO LOCAL
199 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
200 0VFE12    +EFD12                      0.95
201 C RETROALIMENTAÇÃO GERAL
202 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
203 1VF12    +EFD12                      1.000
204 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
205 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
206      0      0.1

```

```

207      1          1.0
208 90BINT1A
209 90BINT1B
210 90BINT1C
211 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
212 0VINT1A +BINT1A 4.35-6
213 C 33VINT1A
214 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
215 0VINT1B +BINT1B 4.35-6
216 C 33VINT1B
217 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
218 0VINT1C +BINT1C 4.35-6
219 C 33VINT1C
220 33FGER11

```

Código B.3: Reguladores de Tensão e Velocidade dos Geradores do Subsistema 1

```

1 C -----
2 C MAQUINA 1 DA UNIDADE DE GERACAO 2
3 C -----
4 C Gerador Sincrono - Serra da Mesa 200 MVA 2 polos 13,8 kV 60 Hz
5 C < n l><>< Tensao >< Freq >< Angulo >
6 59GER21A 11.26E+3 60.0 21.1
7 GER21B
8 GER21C
9 C 1 2 3 4 5 6 7 8
10 C 34567890123456789012345678901234567890123456789012345678901234567890
11 TOLERANCES 300
12 PARAMETER FITTING 2.0
13 C 1 2 3 4 5 6 7 8
14 C 34567890123456789012345678901234567890123456789012345678901234567890
15 C <><> <N>< SMOTVTP>< SMOVTQ >< RMVA >< RKV >< AGLINE >< S1D >< S2D >
16 1 1 1 2 1 1 200.0 13.8 935
17 -1
18 C 1 2 3 4 5 6 7 8
19 C 34567890123456789012345678901234567890123456789012345678901234567890
20 C --- Electrical parameters - Manufacture's data
21 C RA >< XL >< XD >< XQ >< XDD >< XQQ >< XDDD >< XQQQ >
22 0.0030 0.100 1.5 0.750 0.370 0.750 0.240 0.340
23 C TDOP >< TQOP >< TDOPP >< TQOPP >< XO >< RN >< XN >< XCAN >
24 1.8 0.35 0.35 0.110
25 C 34567890123456789012345678901234567890123456789012345678901234567890
26 C --- Mechanical System data
27 C < >< EXTRS >< HICO >< DSR >< HSP >< DSB >
28 1 1.0 0.00763
29 BLANK CARD ENDING MASS CARDS
30 C --- DATA OUTPUT
31 C GA< >< >< >< >< >< >< >< >< >
32 C 11
33 C 21
34 C 31
35 C 51
36 BLANK CARD ENDING OUTPUT REQUEST
37 71EFD21
38 72STEM21 1
39 74VEL21 2
40 FINISH

```

Código B.4: Gerador do Subsistema 2

```

1 C -----

```

```

2 C REGULADOR DE VELOCIDADE UNIDADE 2 GERADOR 1
3 C -----
4 C CARREGANDO A VELOCIDADE DO GERADOR                < T INIT ><T FIM  >
5 92VEL21                                                -1.0000000
6 77VEL21    94.2477796
7 C 33VEL21
8 C NIVEL DO SINAL
9 C <OUT ><><  A    ><  B    ><  C    >                < T INIT ><T FIM  >
10 11FSOLL2  6.000000E10.000000E00.000000E0            -1.0000000
11 C 33FSOLL2
12 C 11VREF    8.365805E30.000000E00.000000E0        -1.0000000 {1.05 PU}
13 C 33VREF
14 C CALCULO DA FREQUENCIA
15 88FGER21 = ((1 * VEL21) / (2 * PI))
16 C CALCULO DA VARIACAO DA FREQUENCIA - COMPARADOR
17 88FDEL21 = FGER21 - FSOLL2
18 C 33FDEL21
19 C CONTROLE - REGULACAO DA VELOCIDADE EM RP (1/R)
20 77RVP21    0.0
21 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
22 0RVRP21                -FDEL21                    5.0
23 C 33RVRP21
24 C CONTROLE - REGULADOR DE VELOCIDADE
25 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
26 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
27 1RV21                +RVRP21                    1.000
28 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
29 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
30      1
31      1      0.02
32 C 33RV21
33 C CONTROLE - FUNCAO DE TRANSFERENCIA DA TURBINA - REAQUECIMENTO
34 77CTUR21    0.0
35 77STEM11    1.0
36 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
37 1CREC21                +RV21                    1.000
38 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
39 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
40      1      1.75
41      1      4.5
42 C 33RV21    0.75 e 3
43 C 33CTUR21
44 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
45 1CTUR21                +CREC21                    1.000
46 C  N0  ><  N1  ><  N2  ><  N3  ><  N4  ><  N5  ><  N6  ><  N7  >
47 C  D0  ><  D1  ><  D2  ><  D3  ><  D4  ><  D5  ><  D6  ><  D7  >
48      1
49      1      0.25
50 C CALCULO DA POTENCIA MAXIMA
51 98STEM21 = 1 + CTUR21
52 C 33STEM21
53 C -----
54 C REGULADOR DE TENSAO UNIDADE 2 GERADOR 1
55 C -----
56 C INICIALIZANDO AS VARIAVEIS
57 90GER21A
58 11VREF2    7.967433E30.000000E00.000000E0        -1.0000000 {1 PU}
59 77EFD21    0.985
60 77VS21     0.0
61 C 90GERB
62 C      1      2      3      4      5      6      7      8

```

```

63 C 34567890123456789012345678901234567890123456789012345678901234567890
64 88VGR21 66+GER21A 60.000
65 C SOMATORIO 1 - DELTA V E A RETROALIMENTACAO
66 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
67 0VS21 -VGR21 -VF21 +VREF2 1.000
68 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
69 1VR21 +VS21 1.000
70 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
71 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
72 1. 0.60
73 1. 0.05
74 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
75 1VRT21 +VR21 1.000 0 1.15
76 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
77 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
78 46
79 1 0.06
80 C SOMATORIO 2
81 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
82 1EFD21 +VRT21 -VFE21 1.000
83 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
84 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
85 1
86 0 0.46
87 C RETROALIMENTACAO LOCAL
88 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
89 0VFE21 +EFD21 0.95
90 C RETROALIMENTACAO GERAL
91 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
92 1VF21 +EFD21 1.000
93 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
94 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
95 0 0.1
96 1 1.0
97 33FGER21

```

Código B.5: Reguladores de Tensão e Velocidade do Gerador do Subsistema 2

```

1 C -----
2 C GERADORES -----
3 C 1 2 3 4 5 6 7 8
4 C 34567890123456789012345678901234567890123456789012345678901234567890
5 C --- Gerador Sincrono de 90MVA, 8polos, 13.8kV, 60Hz (ATP-RULE BOOK VIII)
6 C < n l><> Tensao >< Freq >< Angulo >
7 59GER31A 11267.6528 60. 20
8 GER31B
9 GER31C
10 C 1 2 3 4 5 6 7 8
11 C 34567890123456789012345678901234567890123456789012345678901234567890
12 TOLERANCES 300
13 PARAMETER FITTING 2.0
14 C --- Machine ratings
15 C 1 2 3 4 5 6 7 8
16 C 34567890123456789012345678901234567890123456789012345678901234567890
17 C <><> <N>< SMOTVTP>< SMOVTQ >< RMVA >< RKV >< AGLINE >< S1D >< S2D >
18 1 1 1 8 1.0 1.0 90. 13.80 280.0
19 -1.
20 C 1 2 3 4 5 6 7 8
21 C 34567890123456789012345678901234567890123456789012345678901234567890
22 C BLANK CARD - NO SATURATION MODELING
23 C --- Electrical parameters - Manufacture's data

```

```

24 C RA >< XL >< XD >< XQ >< XDD >< XQQ >< XDDD >< XQQQ >
25 0.038 0.122 1.200 0.700 0.350 0.700 0.250 0.230
26 C TDOP >< TQOP >< TDOPP >< TQOPP >< XO >< RN >< XN >< XCAN >
27 6.240 0.032 0.060 0.150
28 C 3456789012345678901234567890123456789012345678901234567890
29 C --- Mechanical System data
30 C < >< EXTRS >< HICO >< DSR >< HSP >< DSB >
31 1 1.00 3.3330650
32 BLANK CARD ENDING MASS CARDS
33 C --- DATA OUTPUT
34 C GA< >< >< >< >< >< >< >< >< >< >
35 C 11
36 C 21
37 C 31
38 C 51
39 BLANK CARD ENDING OUTPUT REQUEST
40 71EFD31
41 72WAT31 1
42 74VEL31 2
43 FINISH

```

Código B.6: Gerador do Subsistema 3

```

1 C -----
2 C REGULADOR DE VELOCIDADE MAQUINA 1 UNIDADE 3
3 C -----
4 C CARREGANDO A VELOCIDADE DO GERADOR < T INIT ><T FIM >
5 92VEL31 -1.0000000
6 77VEL31 94.2477796
7 C NIVEL DO SINAL
8 C <OUT ><>< A >< B >< C > < T INIT ><T FIM >
9 11FSOLL3 6.000000E10.000000E00.000000E0 -1.0000000
10 C CALCULO DA FREQUENCIA
11 88FGER31 = ((4 * VEL31) / (2 * PI))
12 C CALCULO DA VARIACAO DA FREQUENCIA - COMPARADOR
13 88FDEL31 = FGER31 - FSOLL3
14 C 33FDEL31
15 C CONTROLE - REGULADOR DE VELOCIDADE PRIMARIO 1
16 77RV311 0.0
17 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
18 1RV131 +RVRP31 1.000
19 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
20 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
21 1 1.2
22 1 24
23 C 33RV131
24 C CONTROLE - REGULACAO DA VELOCIDADE EM RP (1/R)
25 77RVRP31 0.0
26 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
27 0RVRP31 -FDEL31 10.0
28 C 33RVRP31
29 C CONTROLE - REGULADOR DE VELOCIDADE PRIMARIO 2
30 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
31 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
32 1RV231 +RV131 1.000
33 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
34 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
35 1
36 1 0.4
37 C 33RV231
38 C CONTROLE - FUNCAO DE TRANSFERENCIA DA TURBINA

```

```

39 77CTUR31 0.0
40 77WAT31 1.0
41 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
42 1CTUR31 +RV231 1.000
43 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
44 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
45 1 -0.15
46 1 0.075
47 C 33RV231
48 C 33CTUR31
49 C CALCULO DA POTENCIA MAXIMA
50 88WAT31T = 1 + CTUR31
51 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
52 0WAT31L +WAT31T 1.000 0 1.0
53 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
54 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
55 98WAT31 = WAT31L
56 C -----
57 C REGULADOR DE TENSAO UNIDADE 3 GERADOR 1
58 C -----
59 C INICIALIZANDO AS VARIABEIS
60 90GER31A
61 11VREF3 7.967433E30.000000E00.000000E0 -1.0000000 {1 PU}
62 77EFD31 0.985
63 77VS31 0.0
64 C 90GERB
65 C 1 2 3 4 5 6 7 8
66 C 34567890123456789012345678901234567890123456789012345678901234567890
67 88VGR31 66+GER31A 60.000
68 C SOMATORIO 1 - DELTA V E A RETROALIMENTACAO
69 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
70 0VS31 -VGR31 -VF31 +VREF3 1.000
71 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
72 1VR31 +VS31 1.000
73 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
74 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
75 1. 0.60
76 1. 0.05
77 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
78 1VRT31 +VR31 1.000 0 1.15
79 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
80 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
81 46
82 1 0.06
83 C SOMATORIO 2
84 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
85 1EFD31 +VRT31 -VFE31 1.000
86 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
87 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
88 1
89 0 0.46
90 C RETROALIMENTACAO LOCAL
91 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
92 0VFE31 +EFD31 0.95
93 C RETROALIMENTACAO GERAL
94 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
95 1VF31 +EFD31 1.000
96 C N0 >< N1 >< N2 >< N3 >< N4 >< N5 >< N6 >< N7 >
97 C D0 >< D1 >< D2 >< D3 >< D4 >< D5 >< D6 >< D7 >
98 0 0.1
99 1 1.0

```

```

100 C IMPRIMINDO EM PU
101 90BINT3A
102 90BINT3B
103 90BINT3C
104 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
105 0VINT3A +BINT3A 4.35-6
106 C 33VINT3A
107 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
108 0VINT3B +BINT3B 4.35-6
109 C 33VINT3B
110 C <NAME><> <IN1 > <IN2 > <IN3 > <IN4 > <IN5 > <GAIN><F LO><F HI><N LO><N HI>
111 0VINT3C +BINT3C 4.35-6
112 C 33VINT3C
113 C 34567890123456789012345678901234567890123456789012345678901234567890
114 99FVINT350+BINT3A 60.000
115 33FGER31

```

Código B.7: Reguladores de Tensão e Velocidade do Gerador do Subsistema 3